CRANFIELD UNIVERSITY

RUSSELL K. STRAND

**Smoothed Particle Hydrodynamics Modelling
for Failure in Metals**

SCHOOL OF ENGINEERING

PhD THESIS

CRANFIELD UNIVERSITY

SCHOOL OF ENGINEERING

PhD THESIS

Academic Year 2009-2010

RUSSELL K. STRAND

**Smoothed Particle Hydrodynamics Modelling
for Failure in Metals**

Supervisor: Dr. James Campbell

March 2010

– II –

To my beloved wife, my soul mate, my 'Squishamin'.

For her endless love, encouragement and support throughout these challenging years.
Quite simply it wouldn't have happened without you!

Thank you Nikki.

This page is intentionally blank

Abstract

It is generally regarded to be a difficult task to model multiple fractures leading to fragmentation in metals subjected to high strain rates using numerical methods. Meshless methods such as Smoothed Particle Hydrodynamics (SPH) are well suited to the application of fracture mechanics, since they are not prone to the problems associated with mesh tangling.

This research demonstrates and validates a numerical inter-particle fracture model for the initiation, growth and subsequent failure in metals at high strain rate, applicable within a Total Lagrangian SPH scheme. Total Lagrangian SPH performs calculations in the reference state of a material and therefore the neighbourhoods remain fixed throughout the computation; this allows the inter-particle bonds to be stored and tracked as material history parameters. Swegle (2000) showed that the SPH momentum equation can be rearranged in terms of a particle-particle interaction area. By reducing this area to zero via an inter-particle damage parameter, the principles of continuum damage mechanics can be observed without the need for an effective stress term, held at the individual particles.

This research makes use of the Cochran-Banner damage growth model which has been updated for 3D damage and makes the appropriate modifications for inter-particle damage growth. The fracture model was tested on simulations of a 1D flyer plate impact test and the results were compared to experimental data. The test showed that the model can recreate the phenomena associated with uniaxial spall to a high degree of accuracy. Some limited modelling was also conducted in 2 and 3 dimensions and promising results were observed.

Research was also performed into the mesh sensitivity of the explosively driven Mock-Holt experiment. 3D simulations using the Eulerian SPH formulation were conducted and the best results were observed with a radial packing arrangement.

An in-depth assessment of the Monaghan repulsive force correction was also conducted in attempt to eliminate the presence of the SPH tensile instability and stabilise the available Eulerian SPH code. Successful results were observed in 1D, although the results could not be replicated consistently in 2D. A further study was also conducted into an approach that makes use of a partition of unity weighting to two different SPH approximations of the same flow-field; one local and one non-local (or extended). Unfortunately this approach could not be made to stabilise the code.

This page is intentionally blank

Acknowledgements

I am indebted to my supervisors Dr. James Campbell and Prof. Rade Vignjevic for all the help, advice and direction that they gave me during this work. Thanks chaps.

Thanks to Cranfield University and the Atomic Weapons Establishment for giving me the opportunity to complete this PhD and for supporting me financially during this work.

Thanks to Nick Snelling for allowing me to pick his brains on a number of quite specific coding matters and for his general wisdom and insight on all manner of things.

Thanks to Adam Swales, Tom Fell and Andrew Gardner for reading and correcting various parts and iterations of this thesis.

Thanks to my friends, my brother and my sisters for the distraction.

Thanks to all my family for giving me the strength and encouragement to finish this thesis. Particularly I would like to thank Auntie Alicia and Zbys for giving me a good talking to and a well needed kick up the backside!

A special thank you to my Dad for his invaluable advice on many aspects of life and for his untiring moral support (and often financial as well).

Finally, thanks to all those in the School of Engineering and Cranfield University who have helped me throughout my PhD.

– VII –

This page is intentionally blank

Table of Contents

This page is intentionally blank

Table of Figures

List of Tables

This page is intentionally blank

Mathematical Notation, Abbreviations and Acronyms

1D	1 dimension/dimensional
2D	2 dimensions/dimensional
3D	3 dimensions/dimensional

$\in$	SPH estimate of unity [LNL]
$\in^L$	local SPH estimate of unity [LNL]
$\in^{NL}$	non-local SPH estimate of unity [LNL]

∇_i	gradient of the kernel function wrt a particle i [MCM, SPH, MRF, LNL]
∇_j	gradient of the kernel function wrt a particle j [SPH, LNL]

α	material parameter [PTW]
	dimensionless variable to obtain ε [MRF]

β	work hardening parameter [SG]
	transition parameter [PTW]

γ	dimensionless material parameter [PTW]
γ_0	thermodynamic constant at the reference state [SG]
	Gruneisen coefficient [Gr]
γ_i	initial plastic strain [SG]

ΔD	damage increment [LDM]
ΔD_{ij}	Cochran-Banner damage increment [CBD]
Δl_{ij}	current change in inter-particle distance between a particle i and its neighbour j [CBD]
Δp	average particle spacing in the neighbourhood of a particle i [MRF(1)]
	initial inter-particle distance [MRF(2), MH]
δ	Dirac delta function [SPH]

$\dot{\boldsymbol{\varepsilon}}$	total rate of deformation tensor [CBD]
$\dot{\boldsymbol{\varepsilon}}^{el}$	elastic component of the rate of deformation tensor [CBD]
$\dot{\boldsymbol{\varepsilon}}^{pl}$	plastic component of the rate of deformation tensor [CBD]
$\dot{\boldsymbol{\varepsilon}}_i$	rate of deformation tensor for a particle i [CBD]
$\dot{\boldsymbol{\varepsilon}}_j$	rate of deformation tensor for a particle j [CBD]
ε	dimensionless variable [MRF]
ε_0	constant [ABV]
ε_e	elasitic strain [CDM]
ε_{ij}	current inter-particle strain between a particle i and its neighbour j [CBD]
ε_p	plastic strain [CDM]
$\dot{\varepsilon}$	strain rate [SG, JC]

$\dot{\varepsilon}^*$	non-dimensional strain rate [JC]
$\dot{\varepsilon}^p$	plastic strain rate [MTS]
$\dot{\varepsilon}_0$	reference strain rate [MTS, JC]
$\dot{\varepsilon}_{0i}$	reference strain rate [MTS]
$\dot{\varepsilon}_{0s}$	reference strain rate [MTS]
$\dot{\varepsilon}_i$	strain rate wrt a particle i [LNL]
$\dot{\varepsilon}_{s0}$	reference strain rate [MTS]
$\dot{\bar{\varepsilon}}^p$	effective plastic strain rate [JC]
$\bar{\varepsilon}^p$	equivalent/effective plastic strain [SG, JC]
ζ	strain equation variable [LNL]
ζ^L	strain equation variable (calculated with local neighbours) [LNL]
ζ^{NL}	strain equation variable (calculated with non-local neighbours) [LNL]
η	current time-step [CBD]
Θ_0	hardening due to dislocation generation [MTS]
θ	Debye coefficient [SG]
	material parameter [PTW]
θ_0	shell thickness [SBF]
θ_D	damage angle [ABC]
κ	perturbation wave number [TL, SPH]
	dimensionless material parameter [PTW]
λ	local SPH smoothing length scale factor used to obtain non-local smoothing length [LNL]
μ	excess compression [Gr]
μ_{ij}	artificial bond viscosity parameter [ABV]
v	specific volume [SG]
v_0	initial specific volume [SG]
ξ	momentum equation variable [LNL]
ξ^L	momentum equation variable (calculated with local neighbours) [LNL]
ξ^{NL}	momentum equation variable (calculated with non-local neighbours) [LNL]
$\dot{\xi}$	reciprocal of the time for a transverse sound wave to cross an atom [PTW]
ξ_{ij}	momentum equation variable [SIA]
Π_{ij}	artificial bond viscosity [MRF]
π	mathematical constant, 3.141592653589

P	problem domain [SPH]
ρ	material density [SPH, TL, SPH, SG, MTS, PTW, MRF, IEPH, Gr, MPM]
ρ_0	initial density [TL, SPH, SG, Gr, PB]
ρ_i	density of a particle i [MCM, SPH, MRF]
ρ_{ij}	average density between each pair of particles [ABV]
ρ_j	density of a particle j [MCM, SPH, MRF, LNL]
ρ_{j^L}	density of a particle j^L (in the local neighbourhood) [LNL]
$\rho_{j^{NL}}$	density of a particle j^{NL} (in the non-local neighbourhood) [LNL]
Σ	spall stress of a material [SC]
$\boldsymbol{\sigma}$	stress tensor [MCM, SPH, SIA]
	Cauchy stress tensor [CDM]
$\boldsymbol{\sigma}^s$	specific stress tensor [MPM]
$\boldsymbol{\sigma}_i$	full-stress tensor of a particle i [MCM, SPH, LNL]
$\boldsymbol{\sigma}_j$	full-stress tensor of a particle j [MCM, SPH, LNL]
$\tilde{\boldsymbol{\sigma}}$	effective stress tensor [CDM]
σ	von Mises equivalent deviatoric stress [PTW]
σ_0	reference yield strength [SG, IEPH]
$\sigma_{i_{np}}$	the np component of the stress tensor for a particle i, where n and p are integers ranging from 1 to 3 [SG]
σ_m	yield stress work hardening limit [SG]
σ_y	yield strength, flow stress, yield stress in tension [MYC, JC, SG]
$\hat{\sigma}$	mechanical threshold stress [MTS]
$\hat{\sigma}_a$	dislocation interaction for long-range barriers [MTS]
$\hat{\sigma}_i$	dislocation interaction for interstitial atoms [MTS]
$\hat{\sigma}_s$	saturation threshold stress (MTS at zero strain hardening) [MTS]
$\hat{\sigma}_{s0}$	saturation threshold stress at 0K [MTS]
$\hat{\sigma}_s^*$	dislocation interaction for solute atoms [MTS]
ς	SPH kernel range constant [SPH]
τ	prescribed part of the traction on the surface $\partial\Omega$ [MPM]
	flow stress [PTW]
$\hat{\tau}$	dimensionless stress variable [PTW]
$\hat{\tau}_s$	saturation stress [PTW]
$\hat{\tau}_s^H$	saturation stress at high-strain rate [PTW]
$\hat{\tau}_y^H$	yield stress at high-strain rate [PTW]
$\hat{\tau}_s^L$	saturation stress at low-strain rate [PTW]
$\hat{\tau}_y^M$	yield stress at intermediate-strain rate [PTW]
$\hat{\tau}_y$	yield stress [PTW]

Φ	support domain or domain of influence (particle neighbourhood) [SPH]
ϕ	mapping function [TL, SPH]
ϕ_1	level set mapping function to describe the location of one end (tip) of a crack [X-FEM]
ϕ_2	level set mapping function to describe the location of one end (tip) of a crack [X-FEM]
Ψ_{ij}	momentum equation variable [SIA]
ψ	equivalent plastic strain [PTW]
	level-set maping function to describe the location of a crack [X-FEM]
$\dot{\psi}$	strain rate [PTW]
Ω	current configuration [MPM]
Ω_I	domain of influence [EFG]
ϖ	complex perturbation frequency [TL, SPH]
ω	nonlinear coefficient [JWL]
$\omega_{\Re}$	real part of a complex perturbation frequency [TL, SPH]
$\omega_{\Im}$	real part of a complex perturbation frequency [TL, SPH]
A	interaction area vector [SIA]
A	atomic weight [SG, PTW]
	material constant [MTS]
	linear coefficient, Mbar (GPa) [JWL]
	yield stress constant [JC]
A_{ij}	Swegle interaction area vector [SIA, CBD]
A_{vo}	Avogadro's constant, 6.025×10^{23} g mol^{-1} [PTW]
ABC	Angular Bond Break Criterion
ABV	Artificial Bond Viscosity
ALE	Arbitrary Lagrangian Eulerian
ARMCO	American Rolling and Milling COmpany
AWE	Atomic Weapons Establishment
a	thermodynamic constant [SG]
	first order volume correction coefficient [Gr]
	coefficients of p^T [EFG]
a_0	material constant [MTS]
a_1	material constant [MTS]
a_2	material constant [MTS]
$\bar{a}_i$	acceleration of a particle i [MRF]
a_i	acceleration of a particle i [LNL]
a_k	nodal degree of freedom of the displacement [X-FEM]

B	linear coefficient, Mbar (GPa) [JWL]
	strain hardening coefficient [JC]
	branch enrichment function [X-FEM]
b	kernel normailsation constant [SPH]
	shear modulus pressure constant [SG]
	magnitude of Burger's vector [MTS]
	body force per unit mass [MPM]
$b^{'}$	yield stress pressure constant [SG]
b_0	shear modulus at 0K [MTS]
b_1	material constant [MTS]
b_2	material constant [MTS]
b_j	nodal degree of freedom of the displacement [X-FEM]
$\mathbf{C}$	material tangent modulus [TL, SPH]
C	specific heat [SG]
	linear coefficient, Mbar (GPa) [JWL]
	shock velocity vs. particle velocity curve intercept [Gr]
	strain rate dependence coefficient [JC]
C_p	specific heat capacity at constant pressure [MTS]
C_L	linear coefficient [ABV]
C_Q	quadratic coefficient [ABV]
CBD	Cochran-Banner Damage growth model
CDM	Continuum Damage Mechanics
CM	Constitutive Model
c_{ij}	average speed of sound between each pair of particles [ABV]
$\mathbf{D}_{ij_n}$	damage matrix for a particle i with n neighbour particles j [CBD]
D	detonation velocity [PB]
	damage variable [CDM]
D_{CB}	Cochran-Banner damage parameter [CBD]
D_{LHS}	inter-particle damage parameter applied to all inter-particle bonds on the LHS of a particle i [CBD]
D_{RHS}	inter-particle damage parameter applied to all inter-particle bonds on the RHS of a particle i [CBD]
D_{crit}	critical damage [LDM, CBD]
D_{ij}	inter-particle damage parameter [CDM, LDM, CBD]
DYNA3D	a nonlinear, explicit, finite element solver developed by LLNL
d	initial inter-particle spacing [MRF]
dV_{ij}	change in volume between a particle i and its neighbour j [CBD]
dt	current time-step [CBD]
dia.	diameter

E	internal energy [PB, Gr]
	Young's Modulus [CDM]
	detonation energy per unit volume, $\left(\text{Mbar} \cdot \text{cm}^3\right)\big/\text{cm}^3$, $\left(\text{GPa} \cdot \text{m}^3\right)\big/\text{m}^3$ [JWL]
E_0	energy at 0K [SG]
	initial internal energy [PB, G]
E_K	(total) kinetic energy [MRF, LNL]
E_c	cold compression energy [SG]
E_i	energy of a particle i [MCM, SPH]
	internal energy [SG]
E_m	melting energy [SG]
E_p	plastic modulus [IEPH]
EFG	Element Free Galerkin method
EFG-P	Particle method within the EFG framework (Rabczuk and Belytschko, 2004)
EOS	Equation Of State
e	strain tensor [MPM]
F	force exerted on a surface due to stress [SIA]
	load acting on the overall section area [CDM]
	deformation gradient [TL, SPH]
$\tilde{\mathbf{F}}$	perturbed deformation gradient [TL, SPH]
F	burn fraction [PB]
	function or kernel estimate [LNL]
F_1	burn fraction component [PB]
F_2	burn fraction component [PB]
F_L	local approximation to a function or kernel estimate [LNL]
F_{NL}	non-local approximation to a function or kernel estimate [LNL]
F_i	force exerted by a particle i [MRF, SIA]
	function or kernel estimate of a particle i [LNL]
FE	Finite Element
f	energy exponential coefficient [SG]
$f_{c_{ij}}$	repulsive force scale function [RFC]
f_{ij}	repulsive force scale function [MRF]
G	shear modulus [SG, MTS, PTW, IEPH]
G_0	reference shear modulus [SG, MTS, JC]
	shear modulus at 0K [PTW]
G'_p	derivative of shear modulus wrt pressure at the reference state [SG]
G'_T	derivative of shear modulus wrt temperature at the reference state [SG]
Gr	Gruneisen EOS
g_0	normalised activation energy for a dislocation/dislocation interaction [MTS]
g_{0i}	normalised activation energy for a dislocation/interstitial interaction [MTS]
g_{0s}	normalised activation energy for a dislocation/solute interaction [MTS]

H	Heaviside enrichment function [X-FEM]
HCP	Hexagonal Close Packed
HE	High Explosive(s)
h	SPH smoothing length [MCM, SPH, MRF, LNL]
	energy coefficient [SG]
	characteristic dimension of the particle under consideration [PB]
h^L	local SPH smoothing length [LNL]
h^{NL}	non-local SPH smoothing length [LNL]
h_0	initial (reference) SPH smoothing length [TL, SPH]
h_j	SPH smoothing length of a contact neighbour particle j [RFC]
h_{ij}	average SPH smoothing length between a pair of particles [ABV]

IEPH	Isotropic Elastic-Plastic Hydrodynamic CM
$\mathbf{i}$	imaginary unit
i	current particle [SPH, MRF, LNL, CBD, LDM]
	current node [X-FEM]

J	determinant of the deformation gradient (Jacobian) [TL, SPH]
J_2	second invarient of the deviatoric stress tensor [MYC]
JC	Johnson Cook CM
JWL	Jones Wilkins Lee EOS
j	neighbour particle [SPH, MRF, LNL, CBD, LDM]
	node whose support is bisected by a crack [X-FEM]
j^L	local neighbour particle [LNL]
j^{NL}	non-local neighbour particle [LNL]
j_1	the first neighbour in the neighbourhood of a particle i [ABC]
j_n	the nth neighbour in the neighbourhood of a particle i [ABC]

K	(degrees) Kelvin
	material coefficient [CDM]
k	neighbour particle of a neighbour particle [TL, SPH]
	Boltzmann's constant [MTS]
	repulsive contact force scale factor [RFC]
	yield stress in pure shear [MYC]
	node of an element containing a crack tip [X-FEM]

LDM	Linear Damage growth Model
LHS	Left-Hand-Side
LLNL	Lawrence Livermore National Laboratory
LNL	Local/Non-Local (partition of unity) approach to approximating derivatives
LS-DYNA	a combined implicit/explicit finite element solver developed by LSTC
LSTC	Livermore Software Technology Corporation
l_d	particle distance from detonation point [PB]

l_{ij}	inter-particle distance between a particle i and its neighbour j [CBD]
l_{ij_p}	inter-particle distance between a particle i and its neighbour j for the time-step before an arbitrary time p [CBD]
l_{ij_q}	inter-particle distance between a particle i and its neighbour j for an arbitrary time q [CBD]
M	atomic mass [PTW]
	magnitude of the vector between a pair of SPH particles [SC]
	material coefficient [CDM]
MA	Maximum Absolute
MCM	Meshless Continuum Mechanics
MH	Mock-Holt experiment
MPM	Material Point Method
MRF	Monaghan's Repulsive Force
MTS	Mechanical Threshold Stress CM
MYC	von Mises Yield Criterion or J_2 plasticity (or J_2 flow) theory
m	total mass of a system [MRF, LNL]
	temperature dependence exponent [JC]
m_i	mass of a particle i [MCM, SPH, MRF]
m_j	mass of a particle j [MCM, TL, SPH, MRF, LNL]
m_{j^L}	mass of a particle j^L (in the local neighbourhood) [LNL]
$m_{j^{NL}}$	mass of a particle j^{NL} (in the non-local neighbourhood) [LNL]
N	number of spatial dimensions [MCM, SPH, MRF,CBD]
	shape function [X-FEM]
N_{nbr}	number of neighbours present in the neighbourhood of a particle i [MRF]
NTNU	Norwegian University of Science and Technology
$\mathbf{n}$	normal vector [CDM]
$\mathbf{n}_i$	unit vector for a particle i which projects onto a new direction [SC]
$\mathbf{n}_{i_x}$	x-component of a unit vector for a particle i which projects onto a new direction [SC]
$\mathbf{n}_{i_y}$	y-component of a unit vector for a particle i which projects onto a new direction [SC]
$\mathbf{n}_{i_z}$	z-component of a unit vector for a particle i which projects onto a new direction [SC]
$\mathbf{n}_j$	unit vector for a particle j which projects onto a new direction [SC]
$\mathbf{n}_{j_x}$	x-component of a unit vector for a particle j which projects onto a new direction [SC]
$\mathbf{n}_{j_y}$	y-component of a unit vector for a particle j which projects onto a new direction [SC]
$\mathbf{n}_{j_z}$	z-component of a unit vector for a particle j which projects onto a new direction [SC]

n	work hardening parameter [SG]
	exponential constant [MRF, RFC]
	strain hardening exponent [JC]
	number of neighbours or particles [SPH]
	neighbour number [CBD]
ODE	Ordinary Differential Equation
OFHC	Oxygen-Free High thermal Conductivity
$\mathbf{P}$	nominal stress tensor [TL, SPH]
$\tilde{\mathbf{P}}$	perturbed nominal stress tensor [TL, SPH]
P	pressure [MRF]
P_0	pressure at 0K [SG]
P_{CJ}	Chapman Jouguet pressure [PB]
P_i	pressure of a particle i [MRF]
P_j	pressure of a particle j [MRF]
PDE	Partial Differential Equation
PB	Prescribed Burn CM
PIC	Particle-In-Cell method
PMMA	Poly(Methyl MethAcrylate), $(C_5O_2H_8)_n$
PTW	Preston Tonks Wallace CM
p	pressure [JWL, Gr]
	arbitrary time [CBD]
p^T	polynomial of arbitrary order [EFG]
p_s	material constant [MTS]
$\mathbf{q}$	artificial viscosity tensor [MCM, MRF]
$\mathbf{q}_i$	artificial viscosity tensor of a particle i [MRF, LNL]
$\mathbf{q}_j$	artificial viscosity tensor of a particle j [MRF, LNL]
q	material constant [MTS]
	characteristic length [MCM]
	arbitrary time [CBD]
q_i	material constant [MTS]
q_s	material constant [MTS]
R	universal gas constant, $8.314472(15)\,\mathrm{J\,K^{-1}\,mol^{-1}}$ [SG]
	Monaghan's repulsive force [MRF]
R_1	nonlinear coefficient [JWL]
R_i	Monaghan's repulsive force due to a particle i [MRF]
R_j	Monaghan's repulsive force due to a particle j [MRF]
RFC	Repulsive Force Contact (algorithim)
RHS	Right-Hand-Side
RKPM	Reproducing Kernel Particle Method

r_e	explosive radius [SBF]
r_i	inner radius (shell) [SBF]
r_{ij}	distance between SPH particles [MCM, SPH, LNL, RFC]
	modulus of the x-coordinate difference between a pair of particles [ABV]
r_o	outer radius (shell) [SBF]
S	deviatoric stress tensor [TL, SPH]
S	surface [MPM]
	overall section area [CDM]
$\tilde{S}$	effective area [CDM]
S_1	first slope coefficient [Gr]
S_2	second slope coefficient [Gr]
S_3	third slope coefficient [Gr]
S_D	total area of micro-cracks and voids [CDM]
SBF	Slate Billings and Fuller experiment
SC	Spall Criterion (for damage model initialisation)
SG	Steinberg-Guinan CM
SIA	Swegle's Interaction Area
SIMLab	Structural Impact Laboratory, (sfi, NTNU)
SPH	Smoothed Particle Hydrodynamics
s_∞	value of $\hat{\tau}_s$ at very high temperature [PTW]
s_0	value of $\hat{\tau}_s$ at zero temperature [PTW]
s_{th}	constant-structure deformation variable [MTS]
s_{thi}	constant-structure deformation variable for interstitial atoms [MTS]
s_{ths}	constant-structure deformation variable for solute atoms [MTS]
sfi	Centre for Research-based Innovation (Department of Structural Engineering, NTNU)
T	usual stress tensor [CDM]
$\mathbf{T}^s$	specific tangent modulus tensor [MPM]
$\mathbf{T}_i$	traction vector of a particle i in the direction of a particle j [SC]
$\mathbf{T}_{i_x}$	x-component of a traction vector for a particle i [SC]
$\mathbf{T}_{i_y}$	y-component of a traction vector for a particle i [SC]
$\mathbf{T}_{i_z}$	z-component of a traction vector for a particle i [SC]
$\mathbf{T}_{ij}$	traction vector between a particle i and a particle j [SC]
$\mathbf{T}_j$	traction vector of a particle j in the direction of a particle i [SC]
$\tilde{\mathbf{T}}$	effective stress vector [CDM]
T	temperature [SG, PTW]
	absolute temperature [MTS]
T^*	homologous temperature [JC]
T_m	melt temperature [SG, PTW, JC]

T_{m0}	melt temperature at the initial density [SG]
T_r	room/reference temperature [MTS, JC]
T_{room}	room temperature (300K) [SG]
$\hat{T}$	dimensionless temperature variable [PTW]
TL	Total Lagrangian
TNT	Trinitrotoluene, explosive material, $C_6H_2(NO_2)_3CH_3$
t	problem (current) time [PB, MRF, CBD, TL, SPH, X-FEM]
t_L	lighting time (particle) [PB]
t_{det}	lighting time (detonation point) [PB]
u	displacement [TL, SPH]
	arbitrary function [EFG]
u_i	nodal degree of freedom of the displacement [X-FEM]
$\tilde{u}$	perturbed displacement [TL, SPH]
$\dot{u}_{ij}$	x-velocity difference between a pair of particles [ABV]
u^h	approximation to u [EFG]
V	volume [MPM]
	compression variable [SG]
	current relative volume [PB]
	volume of detonation products / volume of undetonated HE [JWL]
V_0	initial relative volume [Gr]
V_{0j}	initial volume of a particle j [TL, SPH]
V_i	volume of a particle i [SIA, CBD]
V_j	volume of a particle j [SIA, CBD]
V_{rel}	initial relative volume [MRF]
v	velocity [SPH, MPM]
v_T	total velocity of a system [MRF, LNL]
v_i	velocity of a particle i [MCM, SPH]
v_p	particle velocity [Gr]
v_s	shock velocity [Gr]
$\overline{v}_i$	velocity of a particle i [MRF]
v_j	velocity of a particle j [MCM, SPH]
$\mathbf{W}$	mass equation weighting function [MPM]
$\mathbf{W}^*$	energy equation weighting function [MPM]
W	SPH kernel function [TL, SPH, MRF]
W''	spatial second derivative of the gradient of the SPH kernel function
W_{ij}	SPH kernel function [SPH, MRF, LNL]
W_{ij^L}	local SPH kernel function [LNL]

$W_{ij^{NL}}$	non-local SPH kernel function [LNL]
WD_{MRF}	work done by the Monaghan's repulsive force [MRF]
w	momentum equation weighting function [MPM]
w	weight function [EFG]
wrt	with respect to
X	notation for co-ordinates in the material co-ordinate system [TL, SPH]
X-FEM	eXtended Finite Element Method
x	notation for co-ordinates in the spatial co-ordinate system [TL, SPH]
$\mathbf{x}_{ij_1}$	position vector between a particle i and its first neighbour j_1 [ABC]
$\mathbf{x}_{ij_n}$	position vector between a particle i and its nth neighbour j_n [ABC]
x	position vector [SPH]
x'	position vector [SPH]
$x_{(n-1)}$	x-position of SPH particle at time $n-1$ [MRF]
x_i	x-coordinate of a particle i [SC, ABC]
x_{ij}	x-coordinate difference between a pair of particles [ABV, ABC]
x_j	x-coordinate of a particle j [SC]
x_{j_1}	y-coordinate of a particle j_1 [ABC]
x_n	x-position of SPH particle at time n [MRF]
Y	yield strength [SG]
Y_0	reference yield strength [SG]
Y'_p	derivative of yield strength wrt pressure at the reference state [SG]
$y_{(n-1)}$	y-position of SPH particle at time $n-1$ [MRF]
y_∞	value of $\hat{\tau}_y$ at very high temperature [PTW]
y_0	value of $\hat{\tau}_y$ at zero temperature [PTW]
y_1	transition parameter [PTW]
y_2	transition parameter [PTW]
y_i	y-coordinate of a particle i [SC, ABC]
y_{ij}	y-coordinate difference between a pair of particles [ABC]
y_j	y-coordinate of a particle j [SC]
y_{j_1}	y-coordinate of a particle j_1 [ABC]
y_n	y-position of SPH particle at time n [MRF]
$z_{(n-1)}$	z-position of SPH particle at time $n-1$ [MRF]
z_i	z-coordinate of a particle i [SC, ABC]
z_{ij}	z-coordinate difference between a pair of particles [ABC]
z_j	z-coordinate of a particle j [SC]

z_{j_1} z-coordinate of a particle j_1 [ABC]

z_n z-position of SPH particle at time n [MRF]

This page is intentionally blank

The important thing is not to stop questioning.
Curiosity has its own reason for existing.
One cannot help but be in awe when he contemplates
the mysteries of eternity, of life,
of the marvellous structure of reality.
It is enough if one tries merely to comprehend
a little of this mystery every day.

(Albert Einstein)

If I have seen further than other men,
it is because I have stood upon the shoulders of giants.

(Isaac Newton)

This page is intentionally blank

1   Introduction

1.1   The Motivation for this Research

The response of metals to high stress and strain rate is of great importance in a range of applications; these include projectile impacts, explosive and blast loading. Numerical simulation provides a crucial tool for investigation and analysis in these applications. One primary limitation on the accuracy of the simulations is the ability of material models to correctly predict the material behaviour. Thus modelling and predicting the response of materials to impact and shock loading is a vital research area. The ultimate aim in this area is to develop predictive simulation tools based on advanced spatial discretisation techniques, combined with constitutive relations to numerically model the behaviour of materials under all loading conditions. Researchers have taken different paths in attempt to achieve accurate simulations of the process and have achieved mixed results due to the inherent challenges with the complex computational analysis that is required.

A large portion of crack-growth research is conducted with the use of 'meshed' methods such as the finite element technique. This is not surprising since these methods are well established and understood. The aim of this research programme is to develop a model for the initiation and growth of damage and subsequent failure in metals under shock loading by utilising the advantages of 'meshless' methods such as Smoothed Particle Hydrodynamics (SPH). Meshless (or meshfree) methods do not require the use of a fixed spatial grid to provide connectivity between nodes and this therefore eliminates the issues of mesh entanglement under large deformations, and also allows damage to grow in an arbitrary direction, reducing the influence on the spatial discretisation of the solution. This makes them particularly suited to the application of fracture mechanics.

1.2   The Fracture Model

The fracture model that has been developed takes its principles from the classical understanding of continuum damage mechanics and couples this to an existing area vector concept in the SPH method to apply damage. It makes use of an inter-particle

approach to apply the damage, i.e. the bonds between particles and their neighbours are affected directly and ultimately broken when a critical level has been reached. Existing damage growth concepts are adapted for use with this technique and the result is an approach which is able to accurately predict material failure in the case of plate impact experiments, on which the model has been demonstrated and validated. The development process of this fracture model (which included investigations into enhancements and modifications to the existing SPH method), is explained in detail in this thesis. The following sections first outline the original objectives of this PhD project and then detail the structuring of this thesis which has been laid out largely in a chronological format.

1.3   Summary of Objectives

The objectives of the proposed programme were as follows:

- To explore further developments or enhancements to the existing Cranfield SPH code to facilitate the development and implementation of a fracture model.
- To develop and implement a modelling methodology for the initiation and growth of damage and subsequent failure in metals under high strain rate loading into Cranfield University's SPH code.
- To demonstrate, through correlation with existing experimental data, that the developed fracture model can be used to describe the break-up of materials under high-strain rate loading.

1.4   Structure of the Thesis

This thesis is laid out as follows:

Chapter 2 contains a description of the type of material physics that is of interest to this research. It also contains a history of modelling crack initiation, growth and fracture within both meshed and meshless methods. This chapter contains the bulk of the

literature survey that was conducted for this research and was used to develop ideas and structure to the fracture modelling approach that was adopted.

Chapter 3 primarily discusses the meshless numerical method that has been used for this research, i.e. the SPH method. Two formulations were available for this research: the classical SPH formulation (referred to in this thesis as the Eulerian SPH formulation), developed for modelling of continuum solid mechanics problems, and the Total Lagrangian formulation, an updated SPH approach which addresses some of the draw-backs in Eulerian SPH. These draw-backs (such as the tensile instability problem and numerical fracture) are also identified and discussed in this chapter and an overview of how the two formulations have been utilised within this research is also provided. The chapter also contains a brief explanation of three other meshless numerical schemes (for comparative purposes), which were identified through the literature survey contained in Chapter 2.

Chapter 4 identifies the material models and equations of state that have been used in this research. Three constitutive models, the Johnson-Cook, Steinberg-Guinan and Mechanical Threshold Stress constitutive models are explained. The latter two models were implemented into the Cranfield SPH code as part of this research and so the implementations and the corresponding verification processes are also outlined. A justification of these choices is also provided. Chapter 4 also identifies an explosion model that has been implemented for this research and discusses the verification of the implementation.

Chapter 5 details an investigation into two approaches that were implemented to attempt to stabilise the Eulerian SPH formulation. Two methods that could potentially be implemented to eliminate the SPH draw-backs identified in Chapter 3 are discussed and tested. The first is termed the Monaghan Repulsive Force and is taken from literature. The second attempts a less documented approach which considers a partition of unity weighting to the approximation of the SPH derivatives.

Chapter 6 outlines the modelling of two explosively driven experiments that were taken from literature which describe the type of fracture that this research is concerned with.

Chapters 7 and 8 contain the detailed investigation and development of a novel inter-particle fracture model which has been designed throughout this research. Chapter 7 identifies the functionality required of the fracture model and focuses on the theory of the application of the damage and how it relates to the classical understanding of continuum damage mechanics. The modifications and implementations into the SPH code are also discussed and the effect of the model on SPH stability is examined from a theoretical point of view. An explanation of the position of the damage routine within the time integration loop is also provided. Chapter 8 provides the demonstration and validation of the fracture model by means of two damage growth models that were used to develop the damage parameter over time. It also identifies how the damage growth is initiated between particle bonds. The damage growth models were a linear damage growth model and a modified Cochran-Banner damage growth model which is investigated in detail. Tests are conducted using simulations of a copper plate impact experiment and the model is assessed on its ability to represent the experimental response. Extensive modelling in 1D has been performed and is discussed in detail. Limited modelling in 2 and 3 dimensions is also presented.

Chapter 9 provides the conclusions to this research project and makes some recommendations for further work.

Chapter 10 contains the references and bibliography.

Annex 1 contains some extra verification for the explosion model detailed in Chapter 4. Appendix A contains an extended abstract and set of presentation slides that were presented at an international conference in Trondheim, Norway in June 2008 about early developments in the fracture modelling process. Appendix B contains the submission format of a journal paper that has been submitted to the International Journal of Impact Engineering which also contains information on some of the fracture model developments. Appendix C explains the Preston-Tonks-Wallace constitutive model which was considered as one of the candidate models for this research. Finally Appendix D contains the source code for a program which generates a radial mesh of particles in the configuration of the Mock-Holt experiment as discussed in Chapter 6.

2   A Review of Dynamic Fracture in Solids

This research aims to develop a fracture model to accurately predict the initiation and growth of fracture at high strain rate in metals using computational mechanics. This is generally regarded as a difficult task (as will be discussed in this chapter) and many researchers have taken different paths in an attempt to achieve an accurate simulation of the process. To this end, many different discretisation techniques have been employed, where several of the fracture methods are specific to the individual technique, but some could be manipulated or modified to encompass others. It is the purpose of this review to identify and describe the physical problem that is addressed by this research and also to outline the various methods suggested by the literature.

2.1   A Brief Overview of Material Fracture Mechanics

The process of material failure is largely dependent on type of material and the loading the material experiences. The purpose of this research is to develop a numerical fracture model to predict the break-up of metals at high strain rate and it is therefore necessary to discuss the physics associated with this type of failure in order to gain an understanding of the problem.

2.1.1   Physics of Fracture at High Strain Rate in Metals

Zukas (1990) explains that fracture in metals at high strain rate occurs after the nucleation and growth of micro-cracks and voids in the material coalesce to form cracks. In brittle materials large quantities of micro-cracks and voids are assumed to always be present in the material. The distribution of these pre-existing micro-cracks and voids form predefined failure planes throughout the material. Cracks will propagate along these predefined failure planes once the stress has reached a material limit. In ductile materials micro-cracks and voids are initially present, although their existence is less prominent than in brittle materials. Ductile failure occurs by the slow growth and linking up of micro-cracks and voids. This process is shown diagrammatically in Figure 2.1-1.
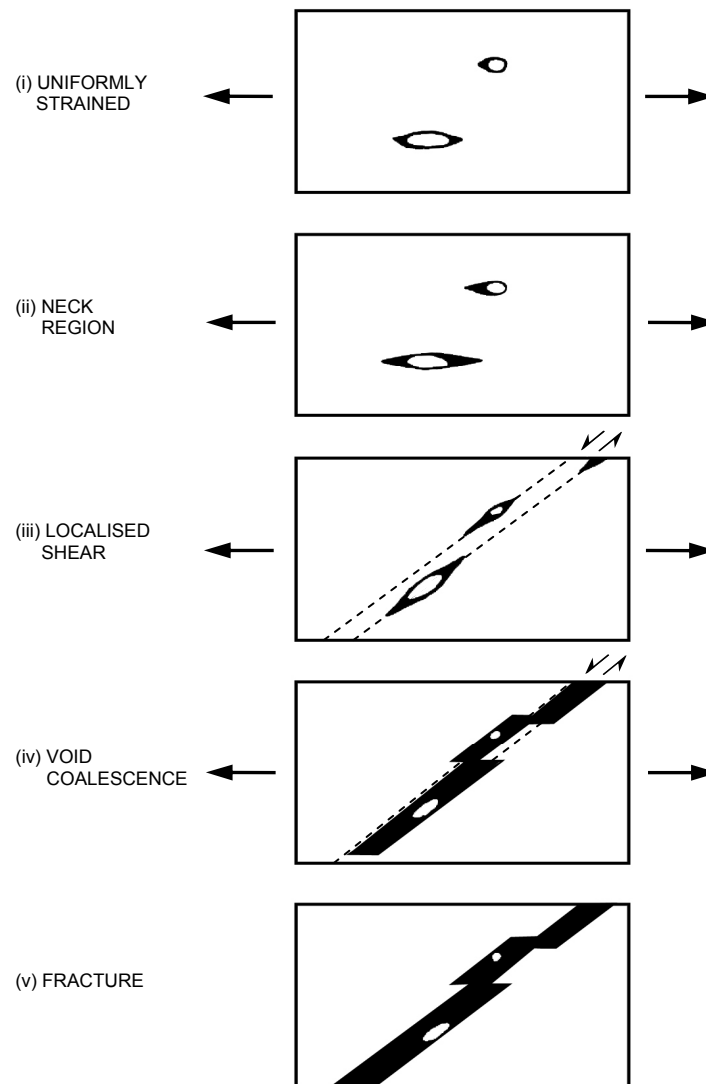
Figure 2.1-1: Typical ductile failure modes identifying the nucleation, growth and coalescence of voids in a material under tensile strain (French, Weinrich 1977).

When a material is subjected to some from of high strain rate loading, i.e. high velocity impact, blast loading, etc. shock waves (hydrostatic tension and compression waves) propagate through the material. The interaction of shock waves in a material can lead to regions of high tensile stress which lead to rapid nucleation, growth and coalescence of micro-cracks and voids and thus fracture. In this instance the type of fracture is termed 'spall fracture' or simply 'spallation.' Figure 2.1-2 clearly shows evidence of spallation in a copper target which was impacted by a copper flyer at a velocity of 304 m/s (Panov 2006).
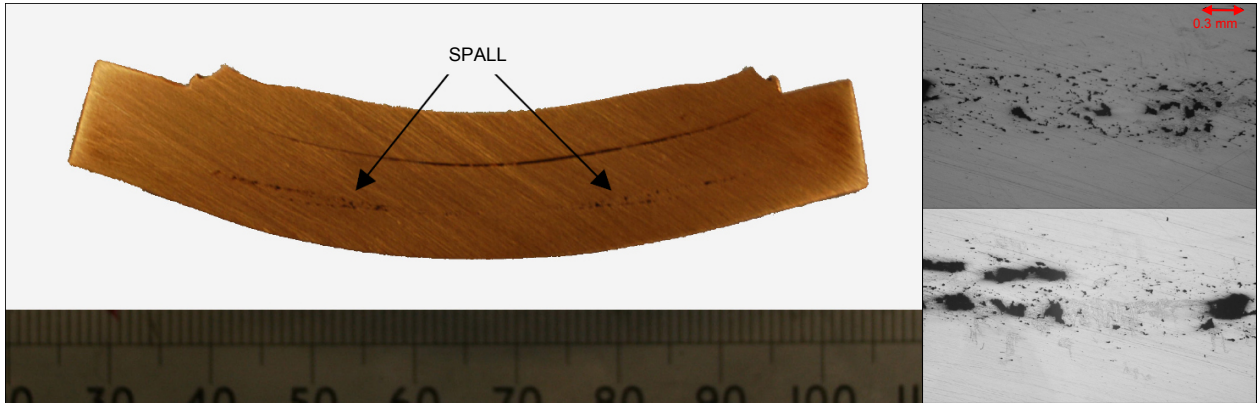
Figure 2.1-2: Cross section of a fly-plate impact experiment exhibiting spall fracture in OFHC copper and microphotographs of spall (inlaid) (Panov 2006).

Plate-impact spall is relatively simple to capture experimentally since it is deterministic in nature (Johnson 1981), i.e. the shape and size of the experiment can be selected such that the peak tensile stress and the spall position are known prior to the execution of the experiment. Spall associated with expanding ring-type or explosively loaded-type experiments is more difficult to capture experimentally since the spall position and tensile stresses are a function of the distribution of the size and location of flaws in the material, i.e. they are probabilistic in nature. Experiments dealing with these types of fracture usually characterise the failure via fragment-mass distributions.

Another phenomenon in fracture mechanics at high strain rate is 'adiabatic shearing.' This process is observed as bands of intense deformation in materials which have received impact loading; they often lead to fragmentation of a body. Zukas (1990) explains that adiabatic shear banding is caused by excess heat generated by high speed deformation. This heat causes the rate of thermal softening of the material to exceed the rate of work hardening. Deformation is then concentrated in narrow softened regions, which produces an adiabatic shear band.

Two experiments which are of particular interest to this research are the Slate, Billings and Fuller experiment (Slate *et al*. 1967) and the Mock Holt Experiment (Mock, Holt 1983) Descriptions of these two experiments are provided in the following two sub-sections since they both deal with the type of fracture that this research is interested in.

2.1.1.1 The Slate-Billings-Fuller Experiment

Slate, Billings and Fuller (SBF) (1967) investigated the rupture behaviour of metals at high strain rates by symmetrically expanding thin spherical shells of various metals via detonation of a sphere of explosive located centrally within the shell. They were able to determine the point of rupture for various rates of strain and various thicknesses of shell. Figure 2.1-3 displays the set-up of the experiment:
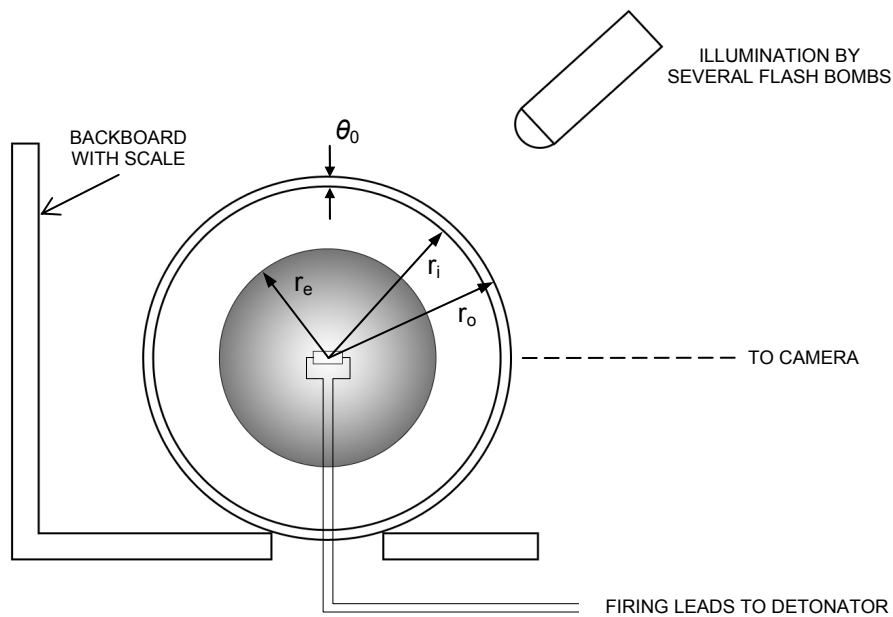
Figure 2.1-3: Diagram of spherical expansion assembly for experiments carried out by Slate, Billings and Fuller (1967).

In Figure 2.1-3 r_e is the explosive radius, r_i and r_o are the inner and outer radii of the metal shell respectively, and θ_0 is the metal shell thickness.

2.1.1.2   The Mock-Holt Experiment

Mock and Holt (1983) performed a series of exploding cylinder experiments using ARMCO iron and HF-1 steel to investigate their fragmentation behaviour at high strain rate. Figure 2.1-4 shows a schematic of the experiment set-up:
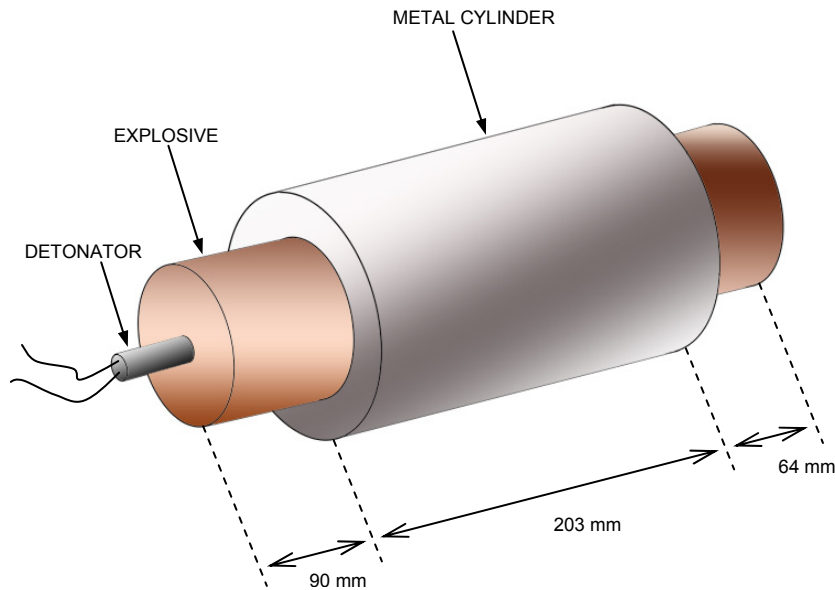
Figure 2.1-4: Diagram of explosive filled cylinder assembly for experiments carried out by Mock and Holt (1983).

To minimise end effects from the explosive interfering with the expansion of the metal cylinders, the explosive was extended 90 mm beyond the end of the cylinder on the detonator and 64 mm beyond the end of the cylinder on the opposite end. The diameter of the explosive was the same both inside and outside the cylinder.

After detonation, the fragments of the metal cylinders were captured in large volumes of sawdust and recovered via magnetic separation techniques. The mass distributions were determined by weighing and counting the fragments. From analysis of the fragments, Mock and Holt identified eight types of fragment that were generated from the experiments. These fragment types are well categorised in their paper. They identify a possible formation of the fragment types across the metal cross-section. A reproduction of their diagram is contained in Figure 2.1-5.
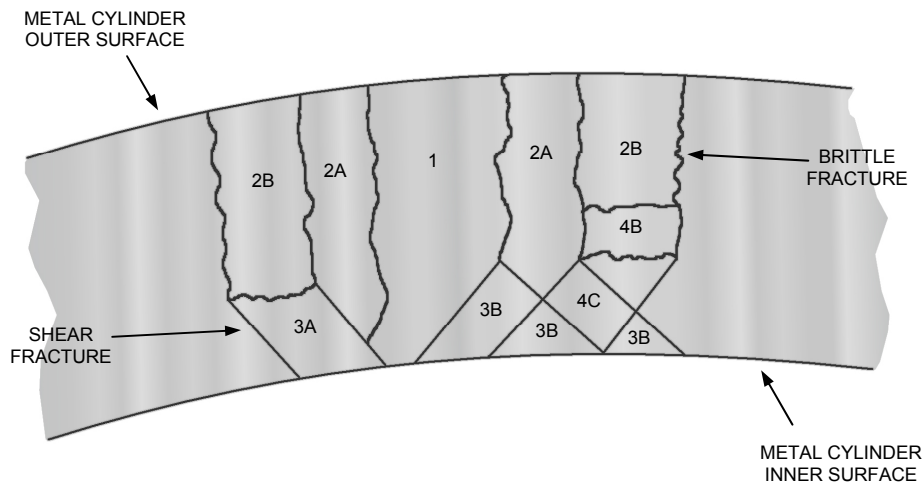
Figure 2.1-5: Schematic diagram through cylinder wall showing the possible formation of the eight fragment types defined by Mock and Holt (1983) from the fragments collected after the experiments had been conducted.

2.1.2 The Basic 'Ingredients' of a Numerical Model to Describe Fracture at High Strain Rate in Metals

From the information contained in Section 2.1 it is possible to identify a set of criteria for a numerical model to be able to capture the type of fracture associated with this research. These are as follows:

- A stable numerical method suitable for modelling rapid deformation in tension (i.e. an explicit hydrocode).
  - The ability to form new free surfaces (in this thesis this is termed a 'fracture model.' Refer to Chapter 7 for detailed information of the method developed for this research).
  - The ability to capture complex fracture patterns (i.e. a mesh resolution that is applicable to the scale of the damage).
- An elastic/plastic material model that is dependent on rate and temperature dependent flow (refer to Chapter 4 for detailed information on the constitutive models that were selected/used for this research).
  - A model to handle the initiation, growth and coalescence of voids. (In this thesis this is termed a 'damage growth model.' Refer to Chapter 8

for detailed information on the damage-growth models used in this research).

For this research the SPH meshless numerical method is utilised (See Chapter 3 for a detailed explanation of the SPH formulations used for this research and Chapter 5 for a detailed explanation of the research into stabilising the Eulerian SPH formulation in tension). The research will utilise an existing damage growth model from the literature and therefore a new model is not developed for this research. However it is understood that modifications to the existing model are likely to be required to integrate it into the SPH method.

2.2 A History of Failure Modelling in Hydrocodes

Numerical methods for modelling fracture mechanics in hydrocodes can be broken down into two main sub-sections, i.e. meshed and meshless methods, and will be considered separately. In this thesis a 'meshed' method refers to numerical simulation techniques that require a structured (or unstructured) computational mesh to provide connectivity between nodes (material points) and to approximate the value of a function or its derivative at a node. A 'meshless' (or 'meshfree') method refers to numerical simulation techniques which use the geometry of the simulated object, defined by a series of discrete nodes/particles (material points) directly for approximations to a function or its derivative. The key difference is that a meshless method does not require a fixed spatial grid (mesh) to provide connectivity between the nodes/particles. It should be noted however that in meshless numerical methods the particle geometry is often referred to as a 'mesh' when discussing the quantity of particles (e.g. 'mesh' density) or the specific packing arrangement (e.g. a square-packed 'mesh'). This terminology is not to be confused with the meshed numerical techniques.

Both meshed and meshless numerical techniques may employ the use of a Lagrangian or Eulerian (or both) specification of the flow field. A Lagrangian specification is a method of monitoring fluid motion where the data capture of a parcel of fluid is performed at the parcel of fluid as it moves through space and time. This may be visualised by an astronaut sitting in a space-craft in orbit about the earth, taking constant

measurements of the spacecraft motion. An Eulerian specification is a method of monitoring fluid motion where the data capture is performed at a series of fixed points in space through which the fluid flows. This may be visualised by a tracking satellite that constantly measures the motion of an orbiting space-craft relative to itself.

This review is split into three sections; firstly techniques for modelling fracture which utilise meshed discretisation methods are discussed in Section 2.2.1. Section 2.2.2 discusses techniques which use various meshless discretisation methods and finally Section 2.2.3 provides generalised conclusions to the ideas outlined in the previous two sections and through discussion of the evidence provided by the literature, a method is selected for development in this research. This section does not contain a review of literature that does not correspond to different methods of reproducing fracture; instead the other literature that has been studied, reviewed or incorporated as a part of this research is outlined or discussed in the chapters where it is relevant.

2.2.1 Representing Cracks within Meshed Methods

Numerous techniques for representing evolving cracks within meshed discretisation methods have been proposed. There are two main classes that these models can fall into: 'Inter-Element Separation Models' and 'Arbitrary Crack-Path Models' (these methods are summarised graphically in Figure 2.2-1).
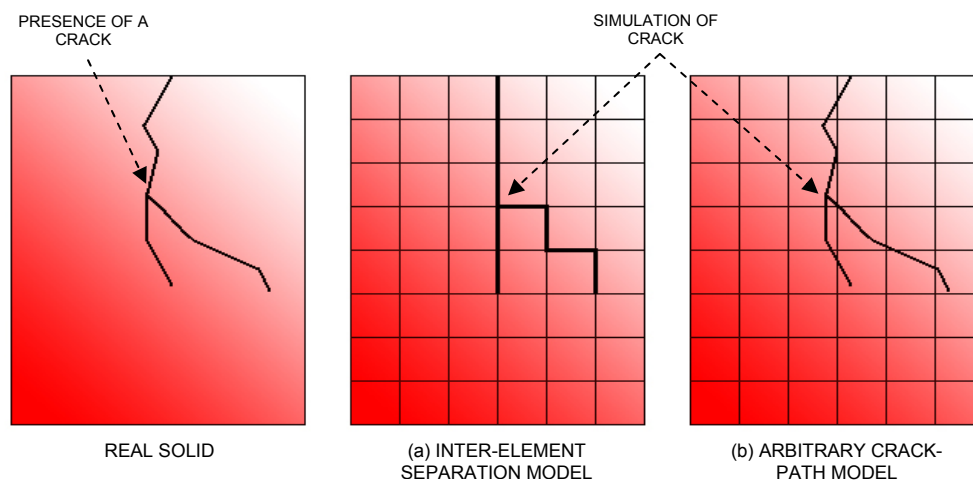
Figure 2.2-1: Visual representation of a crack in (a) an Inter-Element Separation Model and (b) an Arbitrary Crack-Path Model.

2.2.1.1   Inter-Element Separation Models

In an 'Inter-Element Separation Model' cracks are developed along element boundaries, i.e. they follow the lines of the mesh (see (a) in Figure 2.2-1 for a visual representation of this approach). The method is relatively simple compared to others; however when the crack-paths are not in line with the element edges, the fracture energy is often overestimated. Also the solution is sometimes dependent on a well refined mesh (Rabczuk, Belytschko 2004).

An example of this type of model is presented by Xu and Needleman (1994) who carried out simulations of dynamic crack growth in isotropic elastic solids using the finite element method. Their model makes use of a cohesive surface decohesion constitutive law developed by Needleman (1987), which adjusts traction across separating interfaces until it reaches a maximum, it then is allowed to decrease and finally vanish so that decohesion occurs. Cohesive surfaces border volume finite elements and are spread throughout the material to allow for a range of different crack-paths to develop. This approach is widely used to model delamination in composite materials when the specific crack path is known.

2.2.1.2   Arbitrary Crack-Path Models

In contrast to the method highlighted in the previous sub-section several researchers have presented models that are able to handle arbitrary crack paths (see (b) in Figure 2.2-1 for a visual representation of this approach).

The first method highlighted in this sub-section is the eXtended Finite Element Method (X-FEM) presented by Moës *et al*. (1999). Unlike traditional finite element systems, X-FEM allows for a crack to pass arbitrarily through elements by incorporating local enrichment functions to the stress and displacement fields which are discontinuous across a crack. Therefore no re-meshing is required throughout its evolution. The enrichment functions extend the space over which the approximations in the region of the crack are made; this enables the method to naturally reproduce the discontinuity.

The displacement approximation for a function $f(x)$ is given by Stolaska *et al.* (2001) as follows:

$$f^h(x,t) = \sum_i u_i N_i(x) + \sum_j b_j(t) N_j(x) H(\psi(x,t)) + \sum_k N_k(x) \left(\sum_{l=1}^{4} a_k^l(t) B_l(r,\theta) \right),$$

where $N_i(x)$ is the shape function associated with node i at time t. The first sum term in the above equation represents the standard FEM approximation, the other two sum terms are introduced in the X-FEM to alter the integration performed either side of a discontinuity. The suffixes j and k denote the sets of nodes which bisect a crack and contain a crack tip respectively. This is represented diagrammatically in Figure 2.2-2, where the j nodes are represented by circles and the k nodes by squares.
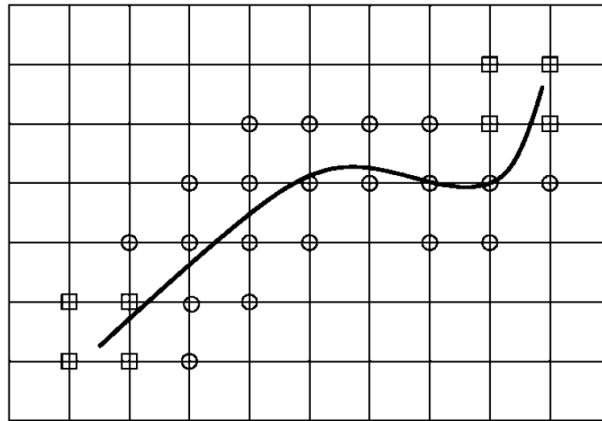
Figure 2.2-2: A set of X-FEM nodes containing a crack. Circles denote nodes enriched by the Heaviside enrichment function and squares denote nodes enriched by the branch function (Stolaska et al. 2001).

u_i, b_j and a_k denote the degrees of freedom of the displacement. $H(\psi(x,t))$ is a Heaviside function and $B_l(r,\theta)$ is termed the branch function, which are the enrichment functions (see Stolaska *et al.* (2001) for more information).

To model a discontinuity the X-FEM method also requires a method to track the crack location thus enabling identification of which nodes require enrichment. In their paper Stolarska *et al.* (2001) also describe the coupling of the Level Set Method (LSM) with

the eXtended Finite Element Method (X-FEM) to achieve this in two dimensions. The LSM method is a numerical scheme used to model the motion of interfaces. It represents the geometry of an interface as the zero contour, termed the zero level set, of a mapping function which is defined everywhere in the region of interest. With the X-FEM method the zero level set can then be used to describe the geometry of a crack, in 2D this would be a line. In practice three mapping functions are used to describe the crack geometry by Stolaska *et al.* (2001), one function defines the geometry of the crack and the other two each define the location of a crack tip (only one crack tip mapping function is required for edge cracks). This can be viewed diagrammatically in Figure 2.2-3, $\phi_1(x,t)$ and $\phi_2(x,t)$ are the mapping functions of the crack tips and $\psi(x,t)$ is the mapping function for the crack geometry.
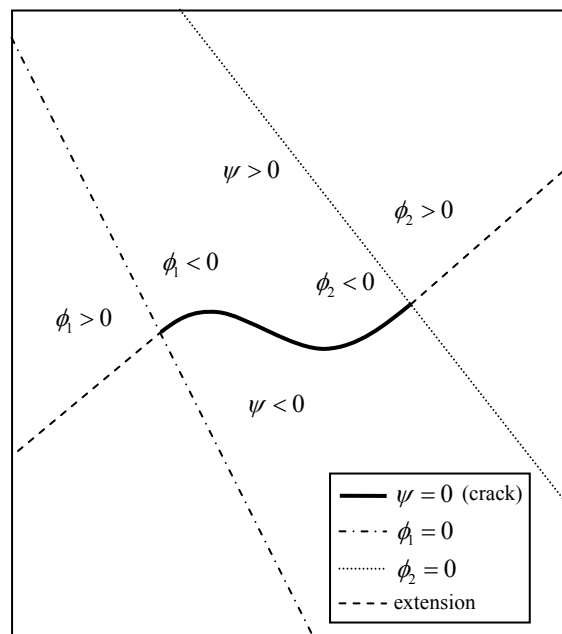
Figure 2.2-3: Diagrammatic representation of a crack described by three level-set mapping functions to define, the location of the crack and the location of the crack tips in 2D space (Stolaska et al., 2001).

The level set functions are held at the individual nodes allowing standard FE interpolation to be used to determine the values of ϕ_1, ϕ_2 and ψ at each node. This has the additional advantage that only local values of the mapping function must be updated in order for the crack geometry to evolve. If the conditions for intersection of either the crack or the crack tip are satisfied in a set of nodes defining an element, then these

nodes are flagged for enrichment by the enrichment functions described earlier. Once ψ has been defined, a crack exists within an element, it remains fixed and so the crack cannot change shape once it has been formed. The location of the crack tips are evolved by an evolution model which is relevant to the specific problem. Once the crack tip locations have been updated, any nodes which were previously flagged for enrichment by the branch function and now do not define an element containing a crack tip are switched to receive enrichment by the Heaviside function.

Moës *et al*. (2002) and Gravouil *et al*. (2002) present a methodology using X-FEM and level sets that updates the process to handle arbitrary cracks in 3D for elastic fracture. Their model is based on work conducted by Sukumar *et al*. (2002, 2003 cited in Moës *et al*. 2002) on 3D planar crack growth problems. The authors claim that for fatigue-crack growth problems, discontinuities can be modelled quite accurately using this approach. However, a new level-set description must be introduced for each individual crack, making it more difficult to model crack branching. This also makes the approach less appropriate for crack formation and growth at high strain rate. To model the fragmentation of a material under explosive loading (where several hundreds of small cracks form, propagate, branch and connect with each other) would require a vast level of complexity with hundreds of level-set descriptions continuously updating at each time-step. The computation would thus be very expensive and therefore it is assessed that using this approach would be inappropriate and would also add a level of complexity outside the scope of this research.

Cullis *et al*. (2003) outline some fracture model developments in the QinetiQ Eulerian Hydrocode – GRIM. The damage growth model they have employed is defined as follows: damage is initiated conventionally via the constitutive model, once a stress criterion has been met in an Eulerian cell. At this point a free surface is not created but instead a number of cells within a specific region will accumulate damage (applied to the flow stress). This will have an effect on the wave propagation which will affect the global behaviour of the material. A third order advection algorithm is used to propagate the damage across Eulerian cell boundaries accurately and failure occurs when the damage in a cell has reached a critical value. The fracture model is described as follows:

when a cell fails in tension the density is reset to the ambient density thus creating a volume deficit in the cell, which is filled with either a gas or a void. At the same instant the stress deviators and the strain are reduced in relation to the quantity of void/gas introduced into the cell. The change in energy of the failed material is assigned to the inserted gas; thus preventing conservation of mass problems. A failed cell is then a mixed material cell and boundary conditions are treated via an internal interface within the cell. However this process is approximate since the velocity nodes on either side of the interface are still connected. The paper explains that this could be improved by introduction of a surface separation algorithm which is still under development and therefore not discussed in detail. One of the test examples is a 2D simulation of an explosively loaded fracture cylinder. It is not clear how the model performed in these tests, but the idea of using a gas to simulate voids which coalesce to form a fracture is an interesting approach. The essence of this approach also forms the make up of the basic element by element spall models used in both LS-DYNA and LLNL-DYNA3D (Lin 2004, Hallquist 1998); i.e. by changing the material/element properties, the effect of a fracture within a material can be achieved (e.g. in the case of spall; a pull back signal is observed). The method employed by Cullis *et al.* (2003) changes more parameters than the DYNA models, but it is essentially still an element by element failure model. A process similar to this was adopted to test early developments of the fracture model developed from this research project.

Another approach to modelling fracture in meshed methods is to use remeshing at various points throughout the simulation to allow a crack to propagate in a direction which is not restricted by the mesh. These methods tend to be very computationally expensive, since in the case of dynamic crack growth, the amount of remeshing required is often substantial. Belytschko and Black (1999) outline a method for crack growth in a finite element environment using enrichment functions so that minimal remeshing of the material is required. The remeshing is done around the crack root so that complexities with the crack tip can be avoided; this means that in most cases the crack does not need to conform to the mesh at its tip. The test and trial functions are discontinuous and the discontinuity is placed along the crack. Thus, it is not treated by the standard FE shape functions. A partition of unity is generated by adding the discontinuous functions to the test and trial functions using original finite elements. Whilst their approach would be

difficult to modify for the purposes of this research their crack growth law is of interest. Crack growth is governed by a maximum circumferential (hoop) stress criterion which states that a crack will propagate from its tip in a direction that corresponds to the maximum circumferential stress. Detailed explanation of the model is given. The above approach has been used in the X-FEM framework by Daux, Moës *et al.* (2000) to extend the method to encompass multiple crack branching, voids and cracks emanating from holes.

Pandolfi, Krysl *et al.* (1999) perform simulations of expanding ring tests via a cohesive law embedded into cohesive finite elements and use continuous adaptive remeshing. The cohesive elements are introduced into the mesh when a critical traction between element interfaces is reached and then proceed to govern the aspects of separation. The authors compare this approach to experimental data and conclude that it agrees well. Due to the differences between the SPH method and the remeshing approach outlined here, it is assessed that it will not be possible to make use of the model for this research (see also Ortiz and Pandolfi (1999) and Zhou and Molinari (2004) for more information and other examples).

Another example of a cohesive segments approach is found in Remmers *et al.* (2003), the approach outlined in this paper is the motivation for Rabczuk and Belytschko (2004) outlined in the forthcoming EFG meshless approach and so is not discussed here.

2.2.2   Representing Cracks within Meshless Methods

A meshless hydrocode does not require the use of a fixed spatial grid to provide connectivity between nodes and therefore eliminates the issues of mesh entanglement under large deformations. Furthermore meshless methods also allow for a more arbitrary crack path to be formed since there are no mesh lines and hence pre-defined directions for a crack to follow. This makes them particularly suited to the application of fracture mechanics.

Many researchers have proposed methods for representing evolving cracks using one of the several different meshless discretisation techniques currently in operation; the following sub-sections describe several of these methods. The SPH framework which has been adopted for this research falls into the meshless method category, and therefore the following techniques are of particular interest to this work.

2.2.2.1   Element Free Galerkin Method (EFG)

The Element Free Galerkin (EFG) method has been applied to the problem of modelling dynamic fracture and is well documented in literature. For a detailed explanation of the EFG method and a discussion of its similarities and differences to the SPH method, refer to Section 3.5.1.

Belytschko and Tabbara (1996) describe some applications of the EFG method to elastodynamic non-rectilinear dynamic crack growth in 2D. Their model assumes the presence of a crack at the start of a simulation and therefore no fracture model is present, however a propagation criterion is employed to allow the crack to grow. A crack is modelled by removing nodes either side of a crack from their respective domains of influence, effectively introducing surfaces either side of the crack. Separated nodes are made traction-free to observe the correct boundary condition. The direction of a crack is determined by a crack growth criterion as follows; the crack is assumed to extend at a constant velocity and has a circumferential stress component implemented near its tip called the hoop stress. The hoop stress is dependent on two stress intensity factors and a crack propagates in a direction perpendicular to the maximum hoop stress, whilst it is less than a critical value. The critical hoop stress is dependent on a material specific dynamic fracture toughness parameter. Stress intensity factors are computed using domain integrals. A crack is therefore modelled as a series of straight-line segments. In their paper Belytschko and Tabbara (1996) explain the validity of their model via comparison of computational studies to test data. They conclude that their model agrees reasonably well with experiment, the main problem being that the crack path is not updated at every time-step. This is due to the fact that a critical length of the crack must be reached in each new direction to accurately evaluate the stress intensity

factors. (See also (Belytschko, Lu *et al*. 1995b, Lu, Belytschko *et al*. 1995, Belytschko, Lu *et al*. 1995a) for more information and background on the approach.)

This approach is potentially quite attractive for this research since it should be relatively straightforward to implement an 'exclude neighbour list' into the SPH code, which could be checked each time-step to allow for the discontinuity between particles.

Krysl and Belytschko (1999) propose a method for modelling arbitrarily propagating cracks in 3D using the EFG method. They use physical and representational models to simulate multiple cracks but their technique cannot model crack branching or interactions between different cracks. EFG nodes are only used in a region close to an expected path of a crack; the remainder of the continuum is modelled using finite elements. This is done because the EFG method is more computationally expensive than FE methods. Figure 2.2-4 displays a visual representation of the set-up of the simulation.
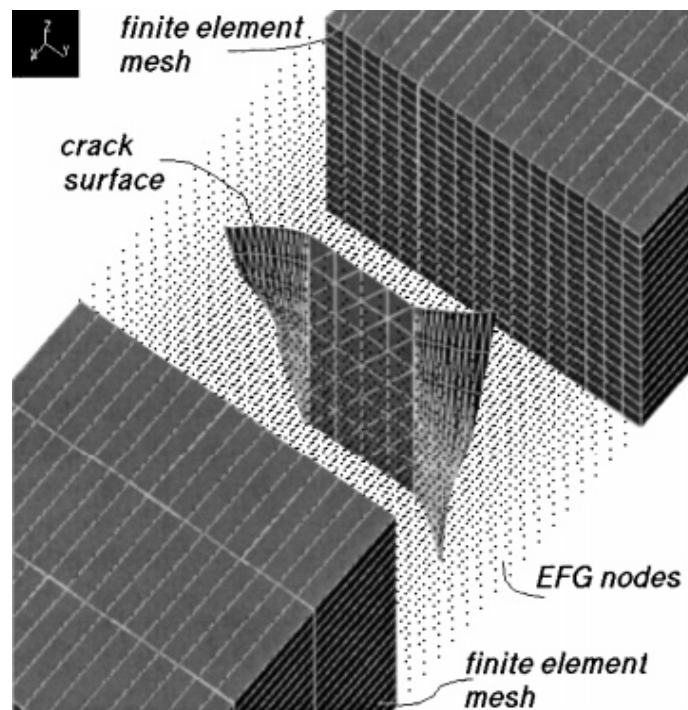
Figure 2.2-4: Visual representation of the crack surface developing within a continuum represented by EFG nodes (Krysl, Belytschko, 1999).

The crack surface is represented by a collection of flat triangles which are updated at each time-step via a propagation model. This is also based on stress intensity factors. When the crack passes between nodes a visibility criterion is enforced to prevent the nodes from interacting further. If the crack path will pass directly through (or very close to) a node, the node is split and the new nodes are placed either side of the crack path. The new nodes have the same kinematic quantities for the state variables but share the mass between them.

The crack representation propagation methods presented above are unsuitable for the applications concerned with this research as crack branching and crack interaction are an integral part of fracture at high strain rate loading. However the visibility criterion could be used to prevent SPH particles from interacting with each other, given that a crack is present in their vicinity.

A 2D approach to modelling cracks in the EFG method using the level set method (described in Section 2.2.1.2) is introduced by Ventura *et al*. (2002). In their approach the description of a crack consists of a vector level set function and the crack tip position. The vector level set function computes and stores the components of the vector which is the closest point projection to a crack; it is evaluated at points between the EFG nodes. For computational efficiency the mapping between the nodal values and the crack projection is only defined in a narrow region of the crack, thus this technique is appropriate for fatigue-type crack problems, since the location of the crack must be known *a priori*. The crack is considered to be fixed once it has formed, however the location of the crack tip is updated by simple algebraic equations based on the crack geometry and therefore no specific evolution algorithm is defined. This is in contrast to the method explained in Section 2.2.1.2 developed in for use with the X-FEM. The authors state that the as the mesh density increases, the geometric update provides more accurate representations of known crack phenomena for both uniform and irregular node formations. The EFG nodes either side of the crack are enriched using a similar technique to the one described in Section 2.2.1.2 for the X-FEM. Since their examples contain the existence of a crack prior to the start of each simulation no fracture model to describe the onset of failure is present. The authors conclude that the method can easily be extended to branching and intersecting cracks. It may be possible to implement this method into Cranfield's SPH code but it is likely to require significant work to couple

this approach with a fracture model and modify the technique to include three dimensions and crack branching.

In their paper Rabczuk and Belytschko (2004) describe a method for representing cracks by introducing discontinuities at individual particles. A continuous crack then consists of a set of contiguous cracked particles (see Figure 2.2-5). Their method is called EFG-P, which is a particle method within the EFG framework.
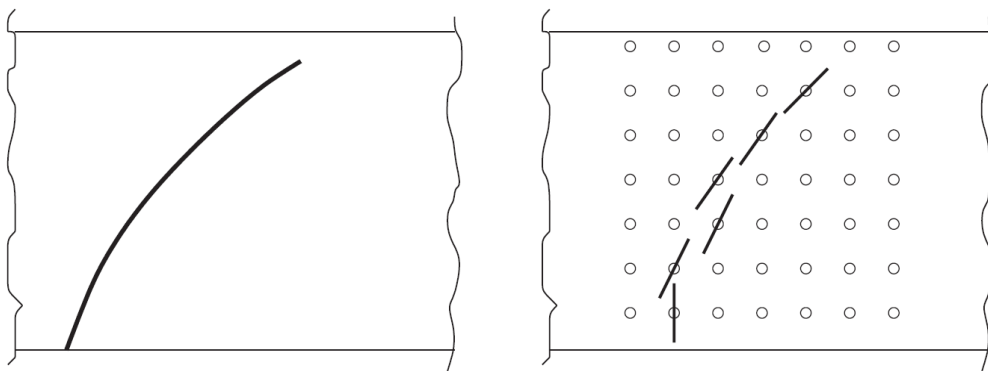
Figure 2.2-5: Crack model on the right (generated by a contiguous set of cracked particles) represents the crack on the left (Rabczuk, Belytschko, 2004).

A crack is initiated when a criterion is met at an individual particle; traction across the crack is then governed by a cohesive law. Two criteria exist; one is dependent on the principle tensile stress reaching a maximum, in this instance the crack direction is perpendicular to the direction of the principle tensile stress. The second is due to loss of hyperbolicity of the partial differential equation at a particle, again the direction of crack depends on the direction of the principle tensile stress. Models are tested on a number of two-dimensional problems and the paper reports encouraging results. A simulation to observe crack branching and another to observe fragmentation are included in the analysis and both phenomena appear to be captured quite well.

In a follow-up paper Rabczuk and Belytschko (2007) extend the method to three-dimensional problems. In contrast to previous work a mixed Lagrangian-Eulerian kernel formulation is introduced. The Lagrangian kernel is used in the early stages of the

simulation to ensure that the cracks are formed due to physical conditions. The switch to the Eulerian kernel is made around the fracture when the deformation begins to get large and thus enables the simulation to continue for extremely large deformations. Again very promising results are reported.

The method is of particular interest to this research since the principle of storing damage information at the particles on a local level exploits the unique environment that is available with a meshless technique.

2.2.2.2 Material Point Method (MPM)

This section describes an application of the material point method to model fracture, For a detailed explanation of the MPM method and a discussion of its similarities and differences to the SPH method, refer to Section 3.5.2.

Banerjee (2004) performed 3D simulations of explosively driven steel containers using the material point method. A hypoelastic-plastic constitutive model that decomposes the rate of deformation tensor into elastic and plastic parts has been used. The Johnson-Cook damage growth model (Johnson, Cook 1985) is used to compute damage. The container is assumed to have an initial uniform porosity. Failure occurs if one or all of the following conditions are satisfied:

- The temperature of a material point exceeds the melting point of its material at the applied pressure,
- The porosity of the material point exceeds a specified limit (i.e. a fracture condition is satisfied (Johnson, Addessio 1988 cited in Banerjee 2004)),
- If either the Drucker stability postulate (Drucker 1959 cited in Banerjee 2004) or the loss of hyperbolicity condition (Rudnicki, Rice 1975 cited in Banerjee 2004) are satisfied (these stability criteria account for bifurcation (crack-branching)).

Upon failure the material point (particle) is removed from the computation by setting the stress to zero. Banerjee (2004) concludes that due to the temperature dependence of

the plasticity models there is an issue of mesh dependence on the results. He also highlights that energy is not conserved in the simulations due to the removal of particles upon failure. He suggests that this could be circumvented by converting the particles into a material with a different velocity field, but a detailed explanation of the process is not available. A similar set of simulations are summarised in Banerjee (2005d) using different constitutive relations but the conclusions are the same. This is of interest for this research since it is relatively simple to remove a particle from the computation once it has failed. It is assessed to be a useful baseline to validate a fracture model implementation and/or as a baseline approach to the problem. The method is synonymous with the method adopted by Cullis *et al*. (2003) (discussed in Section 2.2.1.2) and adapted for use in the development phases of the fracture model designed in this research.

2.2.2.3   Smoothed Particle Hydrodynamics (SPH)

The chosen framework for this research is the Smoothed Particle Hydrodynamics (SPH) method. The code available is called Meshless Continuum Mechanics (MCM); this is an SPH code developed at Cranfield University (Please refer to Chapter 3 for a detailed explanation of the SPH method). This section outlines several attempts to model fracture mechanics within the chosen framework and is of particular interest since it is likely that any models presented here will be fairly simple to implement into MCM. A variety of different situations are considered. This is mainly because SPH can handle extremely large deformations (Liu, Liu 2003) and therefore is well suited to this type of application. There are two ways that fracture can be represented within an SPH framework.

The first exploits a breakdown in the code for tensile loads, which results in numerical fracture within the simulation (detailed explanation of this phenomenon can be found in Section 3.3.1). This approach is not mathematically rigorous and it is not clear if it can be properly validated.

An evaluation of fracture modelling within SPH that utilises this phenomenon has been conducted by Randles, Libersky *et al*. (1996). They used the SPH method to make qualitative comparisons of the method's capability of modelling fracture with experimental data. They chose to use the MK82 bomb as the test model because it is well characterised. They performed an experiment where the detonation and subsequent fragmentation was captured using a continuous framing camera. Using a Mie-Gruneisen equation of state with a linear elasticity and Johnson-Cook plasticity constitutive model, an evolving scalar damage growth model was employed. This responded to both shear and tensile damage to predict the break-up of the MK82 steel casing. The tensile damage growth model employs a damage parameter which accumulates when the stress in tension exceeds a threshold. The shear damage growth model is the Johnson-Cook damage growth model (Johnson, Cook 1985), which also uses a damage parameter. Both of the parameters are used to weaken material properties (bulk modulus and yield stress in the tensile case). The fracture method used here is simple; in an SPH simulation particles are no longer connected when the inter-particle distance becomes greater than the smoothing length. In this model the smoothing length of the damaged particle is reduced to a maximum of 80% of the original value using a scale factor dependent on the damage parameters, i.e. the damaged particles will have less neighbours than undamaged particles and consequently will separate from the continuum when their inter-particle distances have increased beyond their corresponding smoothing lengths. This is basically how numerical fracture occurs within an SPH framework; numerical fracture is discussed in Section 3.3.1.

They compared their simulations with the experimental results for a number of different times and identified that the main difference was more bulging of the tail-end region and overall expansion of the simulation results. They concluded that the SPH method predicts many of the features of fracture and fragmentation and that with more computational power and better constitutive models the process was expected to be greatly improved.

The above conclusions were also drawn by Randles, Carney *et al*. (1994) when they used the SPH method to simulate the fracture of tungsten cubes onto a plate at an

oblique angle. Their constitutive relations involved elasticity, perfect plasticity and a similar scalar tensile damage growth model.

In a private communication, Larry Libersky highlighted some of his results from modelling an RDEC 40 mm bullet made of 4340 steel (2006). His code made use of Monaghan's repulsive term to correct the tension instability problem inherent with any standard SPH method (Monaghan 2000) (See Section 5.1 for more information). He used the Johnson-Cook plasticity model (Johnson, Cook 1983) and allowed the simulation to run until numerical fracture occurred and fragmentation was observed in the result. He found that the fracture and fragmentation of the bullet matched well with experimental data. However this method relies on numerical fracture occurring where particle properties are degraded through the damage growth model, rather than through a physical fracture model.

The SPH approaches to modelling fracture that have been outlined thus far will not be incorporated into the model that will be developed for this research, since it is the purpose of this project to develop a mathematically rigorous approach to fracture mechanics. However, since an Eulerian SPH implementation which is prone to the tension instability and numerical fracture problems already exists (see Section 3.1.2), it would be interesting to make a comparison between any developed fracture model and a simulation without the fracture model active that has been left to run until numerical fracture naturally occurs. Evidence of this method is partially obtained in the development of a suitable validation simulation for this research (see Section 6.2 for more information).

The second method of handling fracture mechanics within SPH is the same as with the previous discretisation methods; a fracture model to handle the application of damage and a damage growth model to develop damage over time are both present in the code.

Benz and Asphaug (1995) developed a fracture model for brittle elastic solids which is based on randomly positioned incipient flaws in the material. The flaws are assigned into the SPH particles in the initialisation of the problem via crack activation threshold strains; i.e. when the strain of a particle exceeds its activation threshold(s), fracture

begins. Once fracture is present, damage accumulates in the particle by reducing the stress through a damage parameter, this causes a stress differential at the crack tip and an algorithm computes the translation of the damage to the surrounding particles; these are referred to as 'unresolved cracks'. When the damage parameter reaches a maximum (unity) the material is fully damaged and said to be *"a cohesionless fluid"* (Benz, Asphaug 1995), capable of carrying no tensile or shear stress. The inter-particle bonds are severed and a new free surface is opened; these are termed 'resolved cracks.' Once the material has stopped fracturing a search is performed for regions of undamaged particles surrounded by fully damaged ones. These regions are called fragments. Benz and Asphaug claim a high level of accuracy in their simulations to experimental data, which involved a tensile basalt rod problem, impact on basalt spheres and laboratory rock cratering experiments. This technique introduces the idea of severing the connection between particles, i.e. by affecting the inter-particle bonds. This is of interest because a fundamental principle of the SPH method is the interaction of particles within a neighbourhood and therefore manipulation of the number of particles contained within the neighbourhoods could be achieved by making modifications to the SPH code database structure.

2.2.2.4   Reproducing Kernel Particle Method (RKPM)

The Reproducing Kernel Particle Method (RKPM) is a development to the Smoothed Particle Hydrodynamics (SPH) technique discussed in the previous sub-section. For a detailed explanation of the RKPM method and a discussion of its similarities and differences to the SPH method, refer to Section 3.5.3.

Simonsen and Li (2004) propose a method for crack propagation in ductile metals using RKPM. Their model is 2-dimensional and assumes that a crack is already present in the material, i.e. no crack initiation model exists. Their propagation procedure is defined as follows:

- Lines connecting arbitrary points in space to various particles are defined as the initial cracks (termed as 'crack-lines'),

- An initial search routine that considers the position of the crack-line(s) is executed to determine particle connectivity,
- The time integration loop is begun and material damage is evaluated in the particles that surround the crack tip(s),
- If damage in a particle in front of the crack has reached a critical value, then:
    - The crack tip position is updated to this particle and the crack-line is adjusted accordingly.
    - A new particle is created at the old position of the crack tip and assumes all of the particle history for the existing particle at this point. The mass of the existing particle is distributed evenly between the two particles, and finally the particles are separated by a distance which is less than inter-particle distance for the continuum.

The 'critical value' that decides material failure is determined by the presence of voids in the material via the Gurson-Tvergaard-Needleman constitutive model (Tvergaard 1981, Tvergaard 1982, Tvergaard, Needleman 1984 cited in Simonsen and Li 2004). The discontinuity is implemented via a 'visibility-criterion' which suggests that a crack is opaque, i.e. particles either side of a crack-line cannot 'see' each other and therefore are not connected. This is enforced by modifying the domains in the integration equations for internal and external force and mass and by also updating the shape-functions. The correct boundary condition is observed because particles which lie on the edge of the crack are traction-free.

Simonsen and Li (2004) conclude that the 'visibility criterion' is of questionable validity because it may introduce discontinuities around the crack tip(s). The level of complexity required for this approach is assessed to be beyond the scope of this research. The damage growth model has also been designed to function with a specific constitutive model and so it may not be possible to adapt the technique to be used with others. However the concept of particle creation upon failure is synonymous with the particle split method outlined previously and is therefore of interest, since this aspect of the model could be incorporated in the available SPH code.

2.2.3   Summary of the Fracture Modelling Survey

To date there have been numerous attempts to accurately predict the break-up of materials using computational mechanics; the literature discussed earlier in this chapter provides several different approaches to the problem and each of these have their own merits; however, many of the published approaches for fracture modelling are not practical for use within an SPH framework and therefore for this research. Whilst it is likely that with enough time almost all of the approaches outlined in this chapter could be implemented in one way or another, many of them will not utilise the advantages of the meshless nature of the method or provide an appropriate framework to model the type of fracture that is required.

Several of the fracture models developed for meshed methods require a separate surface model for the crack (i.e. the Level Set Method); a model which contains branching and joining cracks requires a new set of surface data to be created for each crack and be able to interact with all the other surfaces present. This is generally regarded as a very complex procedure and is thus computationally expensive. For the purpose of modelling fatigue type crack phenomena, i.e. slow-developing discontinuities within a loaded material, this type of method can be very rewarding. However the type of simulations modelled in this research, i.e. explosively driven or hyper-velocity impact experiments would require simultaneous generation of multiple individual surfaces and their subsequent interaction with each other. This would inevitably lead to many hundreds, if not thousands, of surface interactions throughout the computation. It can therefore be assumed that methods requiring a separate set of data for the cracks are not practical for this research.

Other approaches that were identified from the meshed methods required the damage to be coincident with the element edges (i.e. along the mesh lines) or otherwise required a level of remeshing to be introduced as a crack was propagated through the continuum. Both of these approaches are unsuitable for use within an SPH framework since SPH is a meshless numerical method and therefore it is not compatible.

Primarily one of the main advantages of the SPH method is that data are stored individually at each particle and therefore damage (cracks) can be represented on a particle level without significant modifications to the method. Coupled to this, specific crack phenomena that are of interest in high strain rate modelling, i.e. crack branching and joining can be observed simply by natural progression of multiple failing particles and their interactions within the neighbourhoods. It was therefore assessed that the modelling of multiple discontinuities would be better handled on a local level, i.e. damage growth stored at the individual particles themselves or at the interactions between them.

Ordinarily any local method within SPH would require damage to be held at the individual particles, since data are updated and stored for SPH particles at each time-step. This is also the most popular approach in the literature that deals with meshless methods. Such an approach lends itself to a simple damage description, as the framework is already in place (in SPH) to store the required data, but the problem occurs once critical damage has been reached and a crack is ready to be formed. The most logical response to a fully damaged particle is to split it into two parts, where the original particle mass is distributed between the two particles thus adhering to the conservation laws. Algorithms based on this approach were implemented by Rabczuk and Belytschko (2007) and Simonsen and Li (2004). In practice this would also require modifications to the neighbour lists and material contact algorithm. This type of fracture representation is again assessed to be better suited to fatigue-crack type problems since such modifications become complicated when dealing with multiple failed particles within a single neighbourhood at a single time-step, i.e. the type of fracture associated with high strain rate loading. An implementation of a particle split method for this research is therefore considered to be too complex and therefore outside the scope of this project. It is also worth noting that if this method were to be implemented, it would likely be very computationally expensive and require significant code optimisation development to be useful for large simulations.

This decision poses a problem for the concept of individual particle failure since there are no other practical responses to dealing with a fully damaged particle. Several examples in the literature simply delete particles/elements from the computation or

prevent them from influencing surrounding particles/elements by making them traction-free. Whilst this type of approach is relatively simple to implement, the problem is that that the conservation laws are either broken or the correct boundary conditions are not applied. This approach is useful as a baseline to fracture model development and has been utilised in this research to facilitate the design of the damage initiation algorithm outlined later in this thesis (see Section 8.2), but it is not mathematically robust and therefore cannot be incorporated into the final fracture model for this research.

If failure on an individual particle level is to be ruled out, then consideration of applying the failure between the SPH particles (inter-particle failure) is the next logical step. In a standard (Eulerian) SPH code (see Section 3.1.2 for a description of the Eulerian SPH method), a new neighbour search is completed at each time-step; thus a particle may gain or lose neighbours throughout the computation as deformation takes place. This prevents the practical treatment of inter-particle damage which is a material history parameter and therefore must be integrated in time. However this research can utilise the Total Lagrangian formulation of SPH developed for Cranfield University by Reveles (2007) (see Section 3.4 for more information), which uses a fixed set of particle neighbourhoods and thus a fixed set of inter-particle bonds defined in the reference state throughout the computation. This means that the individual particle bonds remain unchanged during the calculation, can be tracked and manipulated as material history variables and be damaged via a damage parameter at every time-step. In this way the cracks can be allowed to grow between particles and can propagate in any direction.

Throughout the literature various levels of complexity have been introduced for the specific mechanics involved with fracture. In general the methods that deal with fatigue-type crack problems tend to incorporate a larger level of complexity than those that deal with multiple fractures. This is likely to be due to requirements to make the code as inexpensive as possible in terms of run time. A further attraction to the local approach for modelling cracks is that the need for complex fracture mechanics can be avoided because the location of the start of a crack will be driven by the parameters of the damage initiation criterion and the damage growth model that considers information at every particle interaction. Development of a crack is then simply in the direction(s) of the particle interactions which meet the damage criteria. The occurrence of branching

and joining can be captured naturally because any particle interaction in a neighbourhood can be broken in any direction.

It can therefore be concluded that damage within the SPH framework is best represented on a local level and more specifically, an inter-particle approach to modelling discontinuities is best suited for this research. A method for manipulating inter-particle bonds within the Total Lagrangian SPH framework has therefore been developed. See chapters 7 and 8 for specific information on the fracture model that has been developed for this research.

It is necessary to note that whilst the literature survey supports the decision to use an inter-particle approach, the decision was also influenced by the outcome of an investigation into SPH stabilisation methods (see Section 3.3.1 and Section 5 for more information). At the start of this research it was unclear which version of the SPH code would be most beneficial to the project and so an investigation into stabilisation methods was conducted alongside the literature survey. Had it been possible to develop a stable version of the Eulerian SPH method, it is likely that this would have been utilised (due to simplicity and the ability to model large deformations) and a particle based approach would also have been selected; this is because particle bonds are not constant throughout a computation and therefore cannot be tracked as material history parameters (as discussed previously). As such the decision to use the Total Lagrangian formulation of the SPH method was necessary because problems associated with the Eulerian SPH technique (which could not be solved) introduce issues when dealing with fracture (discussed later in the thesis).

3   The SPH Method (and Other Meshless Numerical Schemes)

This research makes use of the meshless numerical scheme known as Smoothed Particle Hydrodynamics (SPH). The SPH method is an interpolation technique which uses kernel estimates to approximate a function by calculating its values at an arbitrary set of discrete points (referred to as particles). The standard form of the SPH method uses an interpolation kernel that is a function of the spatial (Eulerian) co-ordinates in the current configuration. Belytschko *et al*. (2000) refer to this approach as the Eulerian SPH method and this is the terminology that has been adopted for this research. The main purpose of this Chapter is to introduce the SPH method and identify the fundamentals of the approach.

Cranfield University has developed an Eulerian formulation of the SPH method which was available for this research (De Vuyst, 2003, Vignjevic *et al*. 2002, De Vuyst *et al*. 2002); the modelling framework is called Meshless Continuum Mechanics (MCM). This chapter begins by introducing the concept of kernel approximation of a continuous function and identifies the specific kernel function adopted for the MCM code along with some examples of other kernel functions (see Section 3.1). Section 3.2 introduces the SPH method in more detail and presents an overview of the fundamentals of Eulerian SPH with derivations of the underlying formulae where applicable. Section 3.3 describes a few shortcomings of the approach which are relevant to this research.

Previous research at Cranfield University has been conducted by Reveles (2007) (see also Vignjevic *et al*. 2006), to develop an enhancement to the Eulerian SPH formulation implemented in MCM; i.e. a Total Lagrangian description of the SPH method that is not subject to some of the identified shortcomings. This enhancement to the SPH method was also available for use within this research. The underlying theory for the Total Lagrangian formulation and a corresponding stability analysis is outlined in Section 3.4.

Since the SPH method is not the only meshless numerical scheme currently being developed, Section 3.5 presents brief explanations of three other meshless numerical methods that were introduced in Chapter 2 as part of the literature survey into existing

techniques to model fracture using computational mechanics. In each case comparisons are made to the SPH method. Finally Section 3.6 explains how the MCM code (comprising both the Eulerian and Total Lagrangian formulations of SPH) has been used in the development of this project.

3.1 Kernel Approximation

3.1.1 Integral Representation of an Arbitrary Function

The interpolation procedure within the SPH method is derived from the exact representation of an arbitrary function in an integral form via the following identity:

$$f(x) \equiv \int_{\Phi} f(x') \delta(x - x') dx',$$

where f is an arbitrary function of the vector x, which is characterised in the integration domain Φ (i.e. Φ is equal to the volume of the integral that contains x). $\delta(x - x')$ is the Dirac delta function (see Kreyszig (1993) for more information), which is defined as follows:

$$\delta(x - x') = \lim_{\alpha \to 0} \begin{cases} 0 & , & \text{if } (x - x') < -\alpha/2 \\ 1/\alpha & , & \text{if } -\alpha/2 \leq (x - x') \leq \alpha/2 \\ 0 & , & \text{if } (x - x') > \alpha/2 \end{cases}.$$

This is shown diagrammatically in Figure 3.1-1.
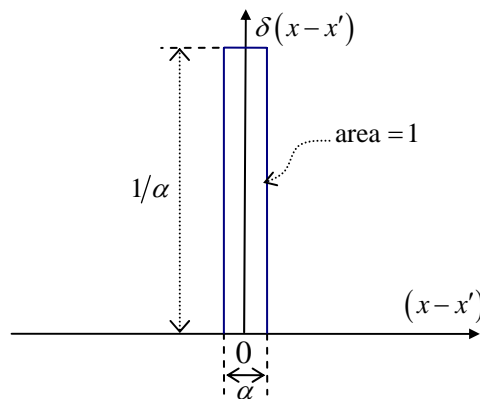
Figure 3.1-1: Diagrammatic representation of the Dirac delta function.

The Dirac delta function produces an exact representation of $f(x)$ because it demonstrates the following attribute:

$$\int_{-\infty}^{\infty} \delta(x - x') dx' = \int_{-\alpha/2}^{\alpha/2} \frac{1}{\alpha} dx' = 1.$$

In computational mechanics the Dirac delta function cannot be used because it is infinitesimally narrow, which means that the integration domain, Φ would not overlap with other material points. The Dirac delta function is therefore replaced with a kernel function (or interpolating kernel, or smoothing function) $W(x - x', h)$, where h is a smoothing length. The kernel approximation to $f(x)$ is therefore defined as follows:

$$\langle f(x) \rangle = \int_{\Phi} f(x') W(x - x', h) dx'.$$

The arbitrary function, $f(x)$ in the above equation is now marked with angle brackets $\langle \ \rangle$ to indicate that it is an approximation and no longer an exact representation. Many different kernel functions exist in literature (four in particular are identified in the following sub-sections), but all kernels must tend to the Dirac delta function (Delta Function Condition) as the smoothing length tends to zero, i.e.

$$W(x - x', 0) = \delta(x - x').$$

The Delta Function Condition is ensured by placing two further conditions on the kernel function. Since the integral of the Dirac delta function over its limits is equal to unity, the integral of the kernel across its support domain must also be equal to unity (often termed the Unity Condition), thus:

$$\int_{\Phi} W(x - x', h) = 1.$$

Also the kernel should exhibit compact support (Compact Condition):

$$W(x - x', h) = 0 \text{ for } x - x' > \varsigma h,$$

i.e. the kernel should have a finite range and ς is usually equal to 2. Whilst the Compact Condition is desirable, it is not mandatory provided that the kernel satisfies the Delta Function and Unity Conditions. However, in solid mechanics applications of the SPH method it is often necessary to develop code which is not computationally expensive. Therefore compact kernels are usually selected in place of non-compact examples. Also when the compact condition is applied to the kernel function, integration over the entire problem domain is localised as integration over the support domain of the kernel function. Therefore the integration domain, Φ can be considered to be the same as the support domain of the kernel function.

The following sub-sections highlight the formulae of four kernels identified in literature, these are briefly summarised in the following paragraphs.

The quadratic kernel (Johnson 1996) (see Section 3.1.1.1 for formulae and graphical representation) has the advantage of being low-order and is therefore computationally cheap. However its simplicity makes it unsuitable for capturing important effects of more complicated physical phenomena (Rogers 2010).

The Gaussian kernel function (Gingold, Monaghan 1977) (see Section 3.1.1.2 for formulae and graphical representation) is an example of a non-compact kernel since it extends past ςh to ∞. This makes it computationally expensive to implement and therefore it is rarely used in SPH codes intended for solving solid mechanics problems, even though it provides a very good approximation to the Dirac delta function.

The cubic B-spline (Monaghan, Lattanzio 1985) and quintic (Wendland 1995) kernel functions (see Section 3.1.1.3 and Section 3.1.1.4 respectively for formulae and graphical representations) are examples of kernels which provide good approximations to the Gaussian kernel. The quintic kernel has improved accuracy and is able to capture high-order material phenomena (Rogers 2010) although it is also computationally expensive to implement. The cubic B-spline strikes a good balance between the low-order quadratic kernel and the high-order quintic kernel and is therefore the most commonly used kernel for computational mechanics applications. It is the kernel function of choice in MCM.

3.1.1.1 Quadratic Kernel Function

The quadratic kernel function (Johnson 1996) has the following form:

$$W\left(x-x',h\right)=\frac{b}{h^{N}}\begin{cases}\dfrac{3}{16}q^{2}-\dfrac{3}{4}q+\dfrac{3}{4} & , \quad \text{if } 0\le q\le 2\\[2mm] 0 & , \quad \text{otherwise}\end{cases}$$

where $q=\left(x-x'\right)/h$, N is the number of spatial dimensions and b is a normalisation constant which has the values 1, $\dfrac{2}{\pi}$, $\dfrac{5}{4\pi}$ in 1, 2 and 3 dimensions respectively. The quadratic kernel is represented graphically in Figure 3.1-2.
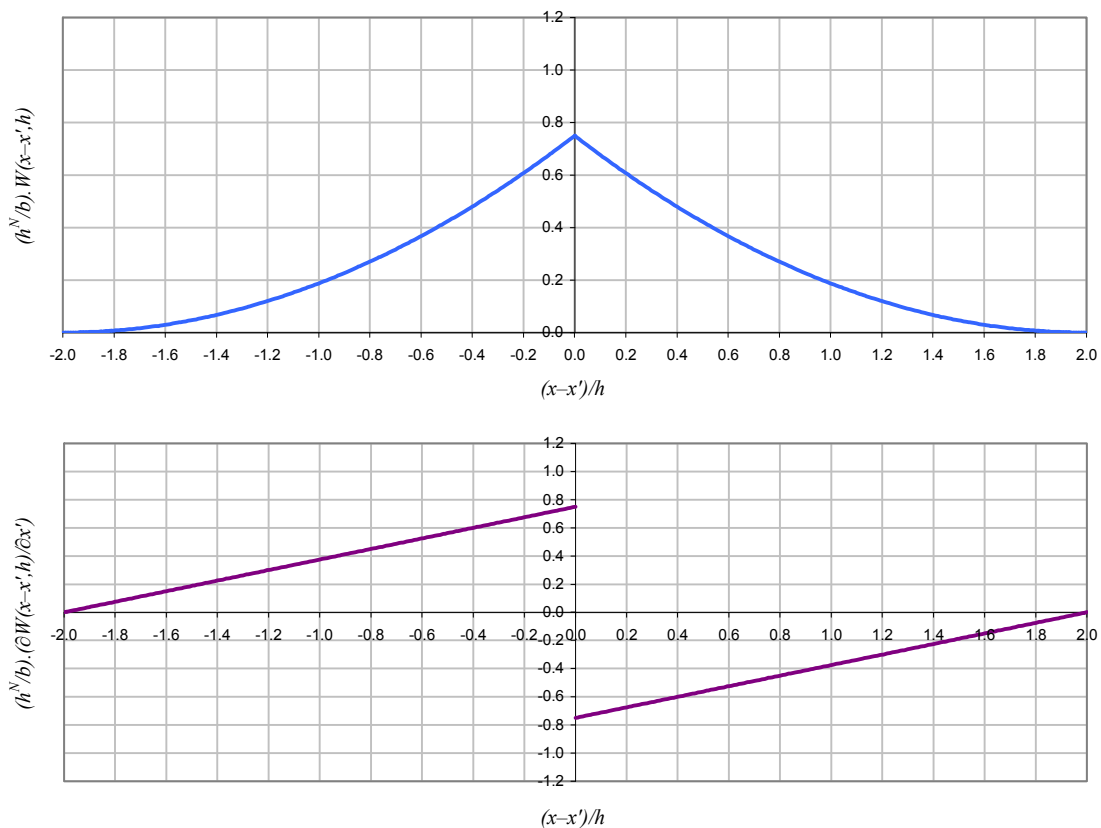
Figure 3.1-2: Quadratic kernel function and its derivative.

3.1.1.2 Gaussian Kernel Function

The Gaussian kernel function (Gingold, Monaghan 1977) has the following form:

$$W\left(x-x',h\right)=\frac{b}{h^{N}}\begin{cases} e^{-q^{2}} & , \quad \text{if } 0 \le q \le 2 \\ 0 & , \quad \text{otherwise} \end{cases}$$

where $q=\left(x-x'\right)/h$, N is the number of spatial dimensions and b is a normalisation constant which has the values $\dfrac{1}{\pi^{1/2}}$, $\dfrac{1}{\pi}$, $\dfrac{1}{\pi^{3/2}}$ in 1, 2 and 3 dimensions respectively. The Gaussian kernel is represented graphically in Figure 3.1-3.
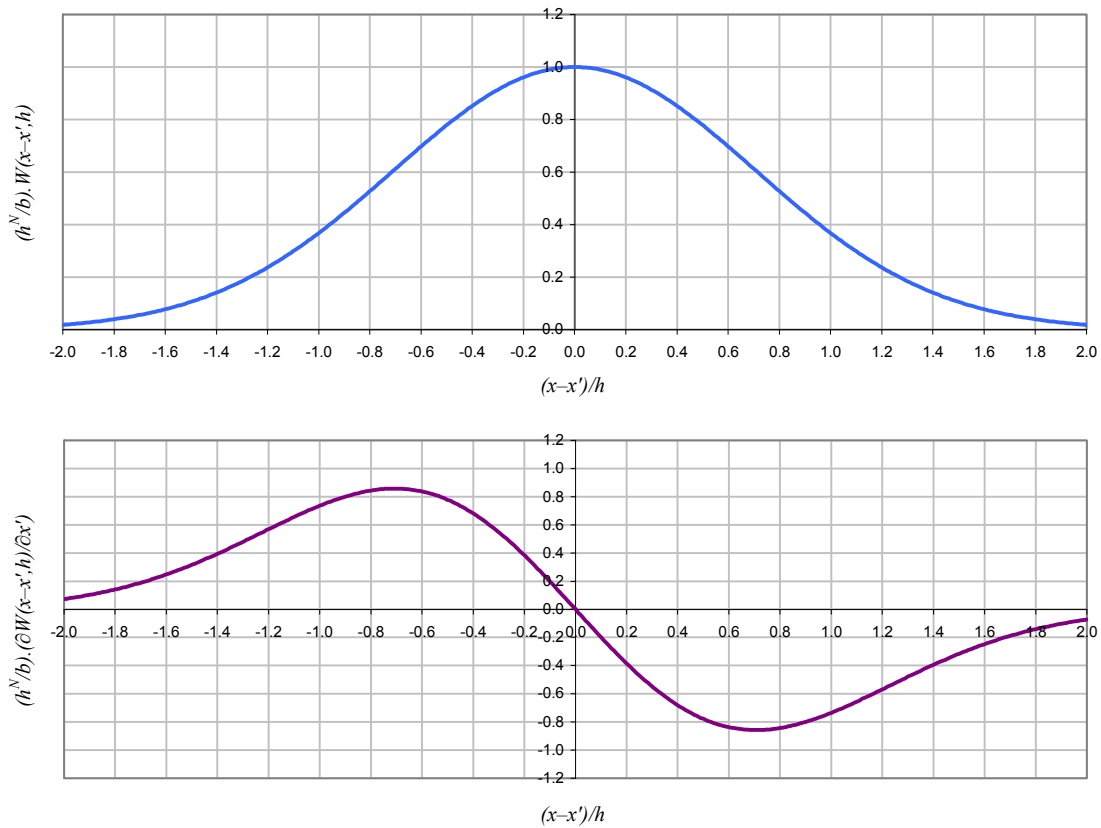
Figure 3.1-3: Gaussian kernel function and its derivative.

3.1.1.3 Cubic B-Spline Kernel Function

The Cubic B-Spline kernel function (Monaghan, Lattanzio 1985) is the kernel function implemented into MCM and has the following form:

$$W(x-x',h) = \frac{b}{h^N} \begin{cases} 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3 & , \quad \text{if } 0 \le q \le 1 \\ \frac{1}{4}(2-q)^3 & , \quad \text{if } 1 \le q \le 2 \\ 0 & , \quad \text{otherwise} \end{cases}$$

where $q = (x-x')/h$, N is the number of spatial dimensions and b is a normalisation constant which has the values $\frac{2}{3}$, $\frac{10}{7\pi}$, $\frac{1}{\pi}$ in 1, 2 and 3 dimensions respectively. The kernel is represented graphically in Figure 3.1-4.
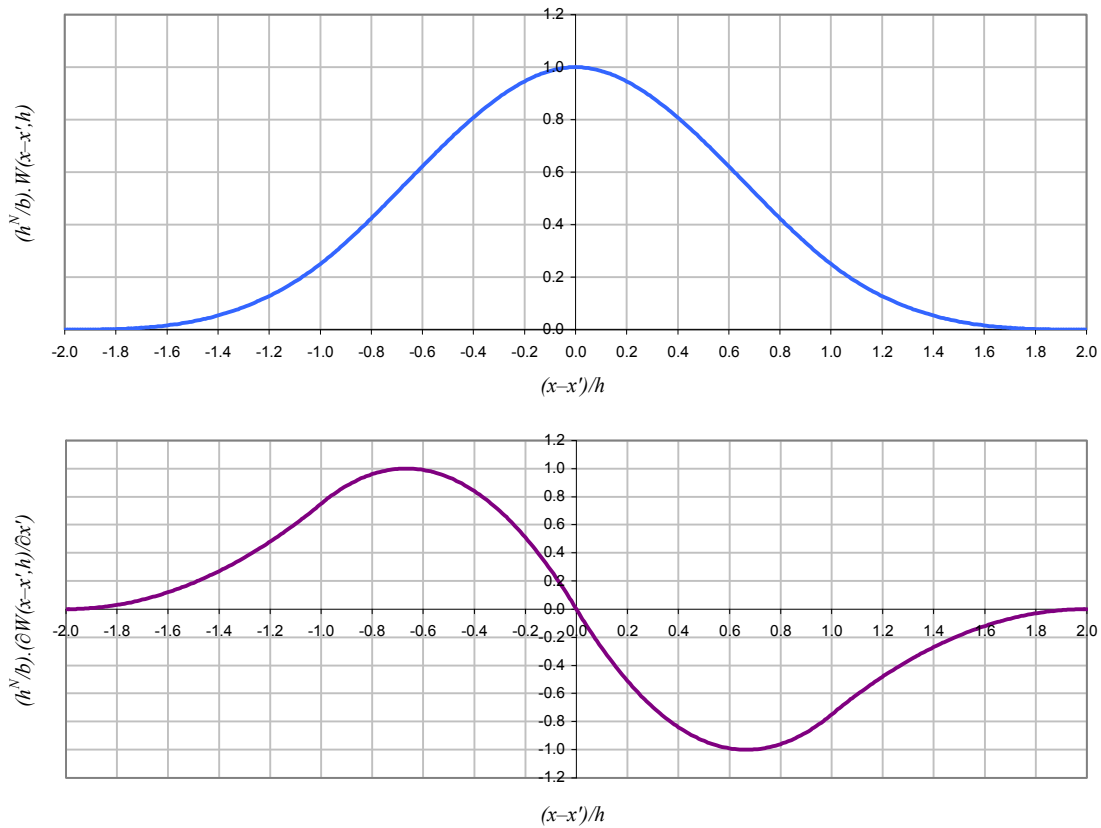
Figure 3.1-4: Cubic B-Spline kernel function and its derivative.

3.1.1.4 Quintic Kernel Function

The quintic kernel function (Wendland 1995) has the following form:

$$W\left(x-x',h\right)=\frac{b}{h^{N}}\left\{\begin{array}{cc}\left(1-\dfrac{q}{2}\right)^{4}\left(2q+1\right) & , \quad \text{if } 0\leq q\leq 2\\[2ex] 0 & , \quad \text{otherwise}\end{array}\right\}$$

where $q=\left(x-x'\right)/h$, N is the number of spatial dimensions and b is a normalisation constant which has the values $\dfrac{7}{4\pi}$, $\dfrac{21}{16\pi}$ in 2 and 3 dimensions respectively. The quintic kernel is represented graphically in Figure 3.1-5.
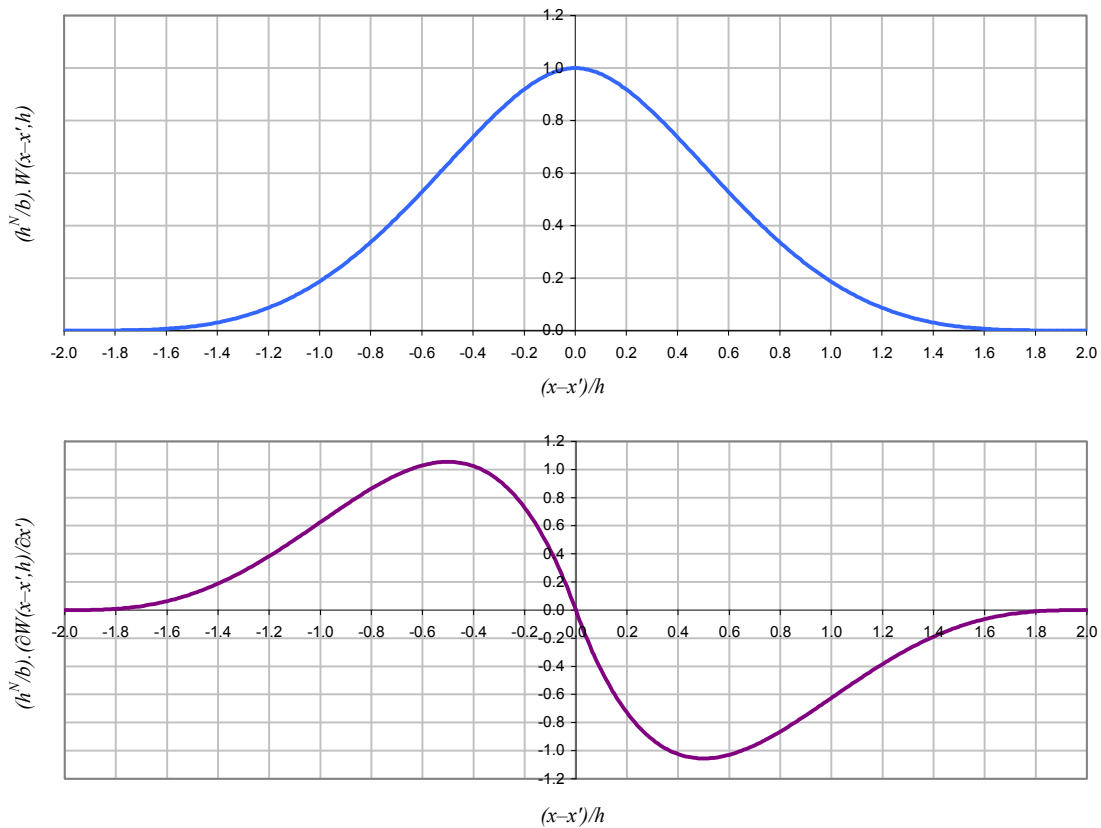
Figure 3.1-5: Quintic kernel function and its derivative.

3.1.2   Integral Representation of the Spatial Derivative of an Arbitrary Function

Starting from the kernel approximation of an arbitrary function derived in Section 3.1.1,

$$\langle f(x) \rangle = \int_{\Phi} f(x') W(x-x',h) dx' \; ;$$

the spatial derivative of the function, $\partial f(x)/\partial x$ is approximated by replacing $f(x)$ in the above equation. Thus:

$$\left\langle \frac{\partial f(x)}{\partial x} \right\rangle = \int_{\Phi} \frac{\partial f(x')}{\partial x} W(x-x',h) dx' \; .$$

From the rule of integration by parts, it is known that:

$$\int_{a}^{b} A'(x) B(x) dx \equiv \left[A(x) B(x) \right]_{a}^{b} - \int_{a}^{b} A(x) B'(x) dx \; .$$

Therefore the approximation of $\partial f(x)/\partial x$ may be rewritten as follows:

$$\left\langle \frac{\partial f(x)}{\partial x} \right\rangle = \left[f(x') W(x-x',h) \right]_{\Phi} - \int_{\Phi} f(x') \frac{\partial W(x-x',h)}{\partial x'} dx'$$

If the kernel function is defined to have compact support (as suggested previously), the square-bracketed term in the above equation is equal to zero when the support domain, Φ is located fully within the problem domain, P (see Figure 3.1-6). This is because the kernel is symmetrical and $W(\varsigma h) - W(-\varsigma h) = 0$. If however the support domain overlaps the problem domain, i.e. near to or at the problem boundary, the kernel is asymmetrical and the square bracketed term is no longer zero (see Figure 3.1-7).
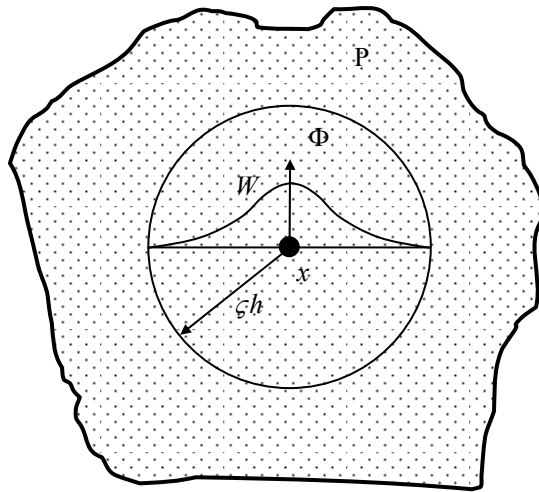
Figure 3.1-6: Diagrammatic representation of the support domain, Φ of the kernel function and the problem domain, P. The kernel is symmetrical since Φ lies completely within P.
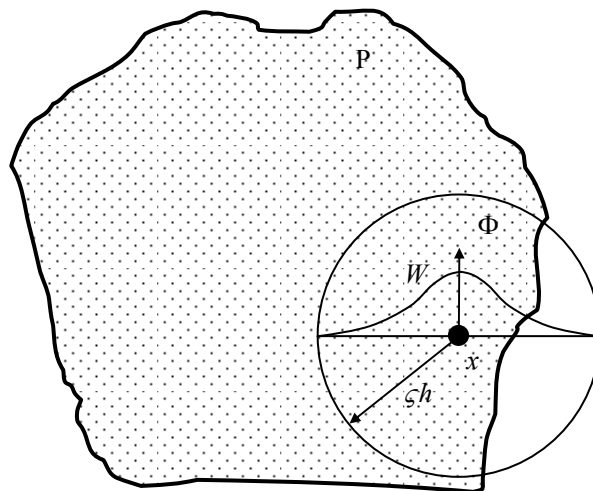
Figure 3.1-7: Diagrammatic representation of the support domain, Φ of the kernel function and the problem domain, P. The kernel is asymmetrical since Φ does not lie completely within P.

In practice the square-bracketed term is assumed to always be equal to zero and therefore a standard SPH formulation is subject to inconsistency at the boundaries (see Section 3.3.3 for further discussion). The approximation of the spatial derivative is therefore simplified as follows:

$$\left\langle \frac{\partial f(x)}{\partial x} \right\rangle = -\int_{\Phi} f(x') \frac{\partial W(x-x',h)}{\partial x'} dx',$$

and by replacing $\partial f(x)/\partial x$ with $\nabla_x f(x)$ (to be consistent with the notation used throughout this thesis), the equation becomes:

$$\langle \nabla_x f(x) \rangle = -\int_\Phi f(x')\nabla_{x'}W(x-x',h)dx'.$$

The above equation prevents the need to approximate the spatial derivative of an arbitrary function from the (unknown) derivatives of the function itself. Instead it produces stable solutions for partial differential equations by computing the spatial derivative of a function from the function itself and the gradient of the kernel function.

3.2 Eulerian SPH Formulation

The Smoothed Particle Hydrodynamics (SPH) method represents a continuum by a set of discrete nodes (particles), which each possess individual material properties and move according to the governing conservation equations of classical Newtonian hydrodynamics. SPH is a meshless numerical method as it does not require the use of a fixed spatial grid to provide connectivity between nodes; this also eliminates the problems associated with large deformations in Lagrangian methods such as mesh tangling. The SPH method can therefore handle extremely large deformations and thus is particularly attractive for applications such as explosion modelling, hypervelocity impact and fracture modelling. Furthermore it facilitates the propagation of an arbitrary crack path when dealing with fracture since there are no mesh lines joining the nodes/particles which could prevent growth in certain directions.

SPH was invented to solve astrophysical problems in three-dimensional open space (Gingold, Monaghan 1977, Lucy 1977) and has evolved to be used in many areas including computational fluid and solid mechanics (Libersky, Petschek 1991). To date the method has been applied to a wide spectrum of applications such as hyper-velocity impact (Libersky, Petschek *et al*. 1993). SPH aims to provide a numerical solution to a hydrodynamics problem which is in the form of partial differential equations (PDE); i.e. an approximation to the values of the field variables (velocity, density, energy, etc.) and

their derivatives at any point. Liu and Liu (2003) state the following key ideas that are employed in the SPH method to achieve this task:

1. The problem domain is represented by a set of arbitrarily distributed particles (meshfree).

2. The integral representation method is used for field approximation (*kernel approximation*, introduced previously in Section 3.1).

3. The kernel approximation is then further approximated using particles (*SPH particle approximation*, introduced in the following sub-section).

4. The SPH particle approximation is performed at every time-step (adaptive).

5. The SPH particle approximations produce a set of ordinary differential equations (ODE) in discretised form with respect to time only (Lagrangian).

6. The ODEs are evolved using an explicit integration algorithm to achieve fast time-stepping and time history for all the field variables and particles.

The following sub-sections outline the fundamentals of the Eulerian SPH formulation implemented in Cranfield University's SPH code, Meshless Continuum Mechanics (MCM). Section 3.2.1 introduces the concept of SPH particle approximation to provide a discrete form of the kernel approximation of an arbitrary function identified in the previous section. Section 3.2.2 presents the Navier-Stokes forms of the conservation equations and provides a derivation of the SPH forms of the conservation of mass and momentum. Section 3.2.3 discusses the numerical implementation of the conservation equations in the MCM code.

3.2.1   SPH Particle Approximation

In Section 3.1 the concept of kernel approximation was introduced in the form of a continuous integral representation of a function and its spatial derivative. The kernel approximation of the spatial derivative was identified as:

$$\left\langle \nabla_x f(x) \right\rangle = -\int_\Phi f(x') \nabla_{x'} W(x-x',h) dx'.$$

In the SPH method the arbitrary function $f(x)$ is calculated at a discrete point known as a particle. Thus the continuous representation of the derivative of the function at a particle, i can be converted into a discrete form, $\langle \nabla_i f(x_i) \rangle$ by summing the contributions from the particle and the corresponding set of neighbour particles, j_n located within the particle's support domain, Φ. This is shown diagrammatically in Figure 3.2-1 and the process is explained below.
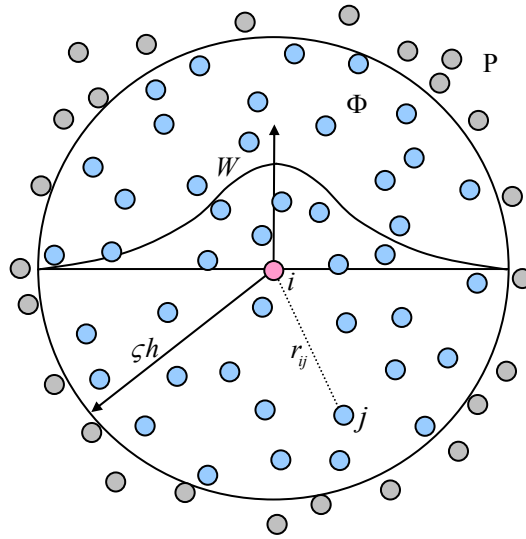
Figure 3.2-1: Particle approximation of an arbitrary function at an SPH particle, i (pink particle) with a circular support domain, Φ of radius ςh (neighbour particles are highlighted in blue).

The first step in the SPH particle approximation approach is to convert the infinitesimal volume, dx' in the continuous integral approximation into a finite volume, V_j of a neighbour particle, j (i.e. a particle contained within the support domain, Φ of a particle, i). The continuous integral is therefore transformed into a summation over discrete particles as follows:

$$\langle \nabla_i f(x_i) \rangle = -\sum_j f(x_j) \nabla_j W(x_i - x_j, h) V_j .$$

The volume, V_j is related to the particle mass via

$$m_j = V_j \rho_j ,$$

where ρ_j is the density of a neighbour particle, j. $W\left(x_i - x_j, h\right)$ may be replaced simply with W_{ij}. The approximation is now given as follows:

$$\left\langle \nabla_i f\left(x_i\right)\right\rangle = -\sum_j \frac{m_j}{\rho_j} f\left(x_j\right)\nabla_j W_{ij} .$$

Since the kernel function is symmetrical, its derivative is therefore asymmetrical and the following relationship is observed:

$$\frac{\partial W\left(x_i - x_j, h\right)}{\partial x_j} = -\frac{\partial W\left(x_i - x_j, h\right)}{\partial x_i} ,$$

or

$$\nabla_j W_{ij} = -\nabla_i W_{ij} .$$

Therefore the SPH particle approximation of the derivative of a function may finally be written as:

$$\left\langle \nabla_i f\left(x_i\right)\right\rangle = \sum_j \frac{m_j}{\rho_j} f\left(x_j\right)\nabla_i W_{ij} .$$

3.2.2 SPH Forms of the Conservation Equations

The governing equations of continuum mechanics are based on the three fundamental physical laws of conservation, i.e. the conservation of mass, momentum and energy. The equations for these laws can be written as a set of partial differential equations known as the Navier-Stokes equations. These are represented as follows:

Conservation of mass: $\qquad \dfrac{d\rho}{dt} = -\rho\dfrac{\partial v}{\partial x} ,$

Conservation of momentum: $\qquad \dfrac{dv}{dt} = \dfrac{1}{\rho}\dfrac{\partial \boldsymbol{\sigma}}{\partial x} ,$

Conservation of energy:
$$\frac{dE}{dt} = \frac{\boldsymbol{\sigma}}{\rho}\frac{\partial v}{\partial x},$$

where ρ is the density, v is velocity and $\boldsymbol{\sigma}$ is the full stress tensor.

The conservation equations evaluated by the SPH method are derived from the Navier-Stokes equations and have the following forms in the MCM code:

Conservation of mass:
$$\left\langle \frac{d\rho_i}{dt} \right\rangle = \rho_i \sum_j \frac{m_j}{\rho_j} \left[v_i - v_j \right] \nabla_i W_{ij},$$

Conservation of momentum:
$$\left\langle \frac{dv_i}{dt} \right\rangle = \sum_j m_j \left[\frac{\boldsymbol{\sigma}_i}{\rho_i^2} + \frac{\boldsymbol{\sigma}_j}{\rho_j^2} \right] \nabla_i W_{ij},$$

Conservation of energy:
$$\left\langle \frac{dE_i}{dt} \right\rangle = -\frac{\boldsymbol{\sigma}_i}{\rho_i^2} \sum_j m_j \left[v_i - v_j \right] \nabla_i W_{ij},$$

where the summation is over all neighbour particles, j and m_j is the mass of the j particle. W_{ij} is the SPH kernel function (see Section 3.1.1.3) and $\nabla_i W_{ij}$ denotes the gradient of the kernel function, taken with respect to the i particle.

The main assumption in arriving at these equations was introduced in Section 3.1.1. It was shown that the first term on the RHS of kernel approximations resulting from integrating by parts is always neglected due to kernel symmetry for neighbourhoods which are fully enclosed within the boundaries of the problem. In astrophysical problems this assumption is justified for the whole problem since the boundaries of the problem are taken to be at infinity. However for solid and fluid dynamics problems where the boundaries are finite, this assumption is not valid since the kernel is not symmetrical near to or at the boundary (depicted diagrammatically in Figure 3.1-7). However the assumption is still usually made for fluid and solid dynamics problems since it allows significant simplifications to the equations to be made (for further discussion on SPH consistency refer to Section 3.3.3).

The following two sub-sections provide a derivation of the mass and momentum equations; for a derivation of the energy equation refer to Campbell (1998):

3.2.2.1   Derivation of the SPH form of the Mass (Continuity) Equation

The derivation of the conservation of mass equation or continuity equation (to which it is often referred) starts with the Navier-Stokes partial differential form:

$$\frac{d\rho}{dt} = -\rho\frac{\partial v}{\partial x}.$$

First the above equation is converted into a kernel approximation as introduced in Section 3.1.1. Thus:

$$\left\langle\frac{d\rho}{dt}\right\rangle = -\int_\Phi \rho'\frac{\partial v'}{\partial x'}W(x-x',h)\,dx'\,.$$

The following part of the derivation contained within the dashed lines is generic to both the mass and momentum equation derivations and therefore it is only stated once.

The integral above contains the generic form:

$$\int_\Phi\left[A(x')\frac{\partial B(x')}{\partial x'}\right]W(x-x',h)\,dx'\,,$$

where $A(x')$ and $B(x')$ are arbitrary functions. The square-bracketed term in the above equation can be represented by the following Taylor expansion about $x=x'$ as follows:

$$\int_\Phi\left\{A(x')\frac{\partial B(x')}{\partial x'}\right\}W(x-x',h)\,dx' = \int_\Phi\left\{A(x)\frac{\partial B(x)}{\partial x}+(x-x')\frac{d}{dx}\left(A(x)\frac{\partial B(x)}{\partial x}\right)+...\right\}W(x-x',h)\,dx'$$

Since the kernel function is symmetrical and satisfies the Unity Condition (explained in Section 3.1.1), the odd powers of $(x-x')$ are equal to zero. Thus:

$$\int_{\Phi}\left[A(x')\frac{\partial B(x')}{\partial x'}\right]W(x-x',h)dx' = \left[A(x')\frac{\partial B(x')}{\partial x'}\right]_{(x=x')}.$$

By substituting $\frac{\partial}{\partial x'}\langle B(x)\rangle$ for $\frac{\partial}{\partial x'}B(x')$ in the RHS of the above equation provides the following rearrangement:

$$\int_{\Phi}\left[A(x')\frac{\partial B(x')}{\partial x'}\right]W(x-x',h)dx' = A(x)\int_{\Phi}\frac{\partial B(x')}{\partial x'}W(x-x',h)dx'.$$

Similarly:

$$\left\langle\frac{d\rho}{dt}\right\rangle = -\int_{\Phi}\rho'\frac{\partial v'}{\partial x'}W(x-x',h)dx' = -\rho\int_{\Phi}\frac{\partial v'}{\partial x'}W(x-x',h)dx'.$$

From the rule of integration by parts, it is known that:

$$\int_a^b A'(x)B(x)dx \equiv \left[A(x)B(x)\right]_a^b - \int_a^b A(x)B'(x)dx.$$

Therefore the approximation $\langle d\rho/dt\rangle$ may be rewritten as follows:

$$\left\langle\frac{\partial\rho}{\partial t}\right\rangle = -\rho\left[\left[v'W(x-x',h)\right]_{\Phi} - \int_{\Phi}v'\frac{\partial W(x-x',h)}{\partial x'}dx'\right]$$

As discussed previously in Section 3.1.2, the term $\left[v'W(x-x',h)\right]_{\Phi}$ is considered to be zero since the kernel is symmetrical and $W(\varsigma h) - W(-\varsigma h) = 0$. In practice this assumption is only true when the support domain, Φ (or particle neighbourhood) lies fully within the problem domain, P (see Section 3.3.3 for further discussion). The approximation in simplified form therefore becomes:

$$\left\langle \frac{\partial \rho}{\partial t} \right\rangle = \rho \int_\Phi v' \frac{\partial W(x-x',h)}{\partial x'} dx' \ .$$

Using the derivation of particle approximation explained in Section 3.2.1, the above continuous representation is discretised as follows:

$$\left\langle \frac{\partial \rho}{\partial t} \right\rangle = \rho \int_\Phi v' \frac{\partial W(x-x',h)}{\partial x'} dx'$$

$$\left\langle \frac{\partial \rho_i}{\partial t} \right\rangle = \rho_i \sum_j v_j \frac{\partial W(x_i - x_j, h)}{\partial x_j} V_j$$

$$= \rho_i \sum_j \frac{m_j}{\rho_j} v_j \frac{\partial W(x_i - x_j, h)}{\partial x_j}$$

$$= \rho_i \sum_j \frac{m_j}{\rho_j} v_j \left(-\frac{\partial W(x_i - x_j, h)}{\partial x_i} \right)$$

$$= -\rho_i \sum_j \frac{m_j}{\rho_j} v_j \frac{\partial W(x_i - x_j, h)}{\partial x_i}$$

Therefore the approximation can finally be written as:

$$\boxed{\left\langle \frac{\partial \rho_i}{\partial t} \right\rangle = -\rho_i \sum_j \frac{m_j}{\rho_j} v_j \nabla_i W_{ij}} \ .$$

By applying different transformations to the RHS of the above equation, different forms of the continuity equation such as the following can be obtained:

$$\boxed{\begin{aligned} \left\langle \frac{\partial \rho_i}{\partial t} \right\rangle &= \rho_i \sum_j \frac{m_j}{\rho_j} \left(v_i - v_j \right) \nabla_i W_{ij} \\ &= \sum_j m_j \left(v_i - v_j \right) \nabla_i W_{ij} \end{aligned}} \ .$$

The above two forms are often termed symmetrised forms because of the presence of the i particle in the equations and are attractive because they result in exact mass conservation (Monaghan 1982).

3.2.2.2 Derivation of the SPH form of the Momentum Equation

The derivation of the conservation of momentum equation starts with the Navier-Stokes partial differential form:

$$\frac{dv}{dt} = \frac{1}{\rho}\frac{\partial \boldsymbol{\sigma}}{\partial x} \ .$$

If the above equation is converted into a kernel approximation as introduced in Section 3.1.1, the following equation is obtained:

$$\left\langle \frac{dv}{dt} \right\rangle = \int_{\Phi} \frac{1}{\rho'}\frac{\partial \boldsymbol{\sigma}'}{\partial x'} W\left(x - x', h\right) dx' \ .$$

Using the same process from the dashed (---) line in the derivation of the continuity equation (see previous sub-section). The following particle approximation is obtained:

$$\left\langle \frac{\partial v_i}{\partial t} \right\rangle = \frac{1}{\rho_i}\sum_j \frac{m_j}{\rho_j}\boldsymbol{\sigma}_j \nabla_i W_{ij} \ .$$

The above form of the momentum equation however is not typically the form which is used by SPH codes. As with the continuity equation a transform can be applied to represent the RHS in the above approximation in a symmetrised form, i.e.:

$$\boxed{\left\langle \frac{\partial v_i}{\partial t} \right\rangle = \sum_j m_j \frac{\left(\boldsymbol{\sigma}_i + \boldsymbol{\sigma}_j\right)}{\rho_i \rho_j}\nabla_i W_{ij}}$$

A more widely used symmetrised form of the SPH momentum equation can be obtained by rearranging the Navier-Stokes equation into a different form. The derivation begins by observing the quotient rule of differentiation:

$$\left(\frac{A(x)}{B(x)}\right)' \equiv \frac{A'(x)B(x) - A(x)B'(x)}{\left(B(x)\right)^2} \ .$$

From the above, the following expression in terms of $\boldsymbol{\sigma}$ and ρ can be obtained:

$$\frac{\partial}{\partial x}\left(\frac{\boldsymbol{\sigma}}{\rho}\right) \equiv \frac{\frac{\partial \boldsymbol{\sigma}}{\partial x}\rho - \boldsymbol{\sigma}\frac{\partial \rho}{\partial x}}{\rho^2}.$$

Rearranging the above to contain the Navier-Stokes form of the momentum equation gives:

$$\frac{\partial}{\partial x}\left(\frac{\boldsymbol{\sigma}}{\rho}\right) = \boxed{\frac{1}{\rho}\frac{\partial \boldsymbol{\sigma}}{\partial x}} - \frac{\boldsymbol{\sigma}}{\rho^2}\frac{\partial \rho}{\partial x}.$$

Thus the Navier-Stokes form can be expressed as follows:

$$\frac{dv}{dt} = \frac{1}{\rho}\frac{\partial \boldsymbol{\sigma}}{\partial x} = \frac{\partial}{\partial x}\left(\frac{\boldsymbol{\sigma}}{\rho}\right) + \frac{\boldsymbol{\sigma}}{\rho^2}\frac{\partial \rho}{\partial x}.$$

Converting the new form into a kernel approximation produces the subsequent continuous integral form:

$$\left\langle\frac{dv}{dt}\right\rangle = \int_\Phi \frac{\partial}{\partial x'}\left(\frac{\boldsymbol{\sigma}'}{\rho'}\right)W(x-x',h)\,dx' + \int_\Phi \frac{\boldsymbol{\sigma}'}{\rho'^2}\frac{\partial \rho'}{\partial x'}W(x-x',h)\,dx'.$$

The above equation may be simplified using the same process from the dashed (---) line in the derivation of the continuity equation (see previous sub-section) to give:

$$\left\langle\frac{dv}{dt}\right\rangle = \left(\int_\Phi \frac{\partial}{\partial x'}\left(\frac{\boldsymbol{\sigma}'}{\rho'}\right)W(x-x',h)\,dx'\right) + \frac{\boldsymbol{\sigma}}{\rho^2}\left(\int_\Phi \frac{\partial \rho'}{\partial x'}W(x-x',h)\,dx'\right).$$

Integrating each of the curved-bracketed ( ) integrals in the above by parts produces the following:

$$\left\langle\frac{dv}{dt}\right\rangle = \left(\left[\frac{\boldsymbol{\sigma}'}{\rho'}W(x-x',h)\,dx'\right]_\Phi - \int_\Phi \frac{\boldsymbol{\sigma}'}{\rho'}\frac{\partial W(x-x',h)}{\partial x'}\,dx'\right) + \frac{\boldsymbol{\sigma}}{\rho^2}\left(\left[\rho'W(x-x',h)\right]_\Phi - \int_\Phi \rho'\frac{\partial W(x-x',h)}{\partial x'}\,dx'\right).$$

Using the same argument used to simplify the continuity equation which was introduced in Section 3.1.2 and explained further at the end of Section 3.2.2, the above can be simplified thus:

$$\left\langle \frac{dv}{dt} \right\rangle = -\int_\Phi \frac{\boldsymbol{\sigma}'}{\rho'} \frac{\partial W(x-x',h)}{\partial x'} dx' - \frac{\boldsymbol{\sigma}}{\rho^2} \int_\Phi \rho' \frac{\partial W(x-x',h)}{\partial x'} dx' .$$

Using the derivation of SPH particle approximation explained in Section 3.2.1, the above continuous representation is discretised as follows:

$$\left\langle \frac{dv}{dt} \right\rangle = -\int_\Phi \frac{\boldsymbol{\sigma}'}{\rho'} \frac{\partial W(x-x',h)}{\partial x'} dx' - \frac{\boldsymbol{\sigma}}{\rho^2} \int_\Phi \rho' \frac{\partial W(x-x',h)}{\partial x'} dx'$$

$$\left\langle \frac{dv_i}{dt} \right\rangle = -\sum_j \frac{\boldsymbol{\sigma}_j}{\rho_j} \frac{\partial W(x_i-x_j,h)}{\partial x_j} V_j - \frac{\boldsymbol{\sigma}_i}{\rho_i^2} \sum_j \rho_j \frac{\partial W(x_i-x_j,h)}{\partial x_j} V_j$$

$$= -\sum_j \frac{m_j}{\rho_j} \frac{\boldsymbol{\sigma}_j}{\rho_j} \frac{\partial W(x_i-x_j,h)}{\partial x_j} - \frac{\boldsymbol{\sigma}_i}{\rho_i^2} \sum_j \frac{m_j}{\rho_j} \rho_j \frac{\partial W(x_i-x_j,h)}{\partial x_j}$$

$$= -\sum_j \frac{m_j \boldsymbol{\sigma}_j}{\rho_j^2} \left(-\frac{\partial W(x_i-x_j,h)}{\partial x_i} \right) - \frac{\boldsymbol{\sigma}_i}{\rho_i^2} \sum_j m_j \left(-\frac{\partial W(x_i-x_j,h)}{\partial x_i} \right)$$

$$= \sum_j \frac{m_j \boldsymbol{\sigma}_j}{\rho_j^2} \frac{\partial W(x_i-x_j,h)}{\partial x_i} + \frac{\boldsymbol{\sigma}_i}{\rho_i^2} \sum_j m_j \frac{\partial W(x_i-x_j,h)}{\partial x_i}$$

$$= \sum_j \frac{m_j \boldsymbol{\sigma}_j}{\rho_j^2} \frac{\partial W(x_i-x_j,h)}{\partial x_i} + \sum_j \frac{m_j \boldsymbol{\sigma}_i}{\rho_i^2} \frac{\partial W(x_i-x_j,h)}{\partial x_i}$$

$$= m_j \left[\sum_j \frac{\boldsymbol{\sigma}_j}{\rho_j^2} \frac{\partial W(x_i-x_j,h)}{\partial x_i} + \sum_j \frac{\boldsymbol{\sigma}_i}{\rho_i^2} \frac{\partial W(x_i-x_j,h)}{\partial x_i} \right]$$

$$= m_j \sum_j \left(\frac{\boldsymbol{\sigma}_i}{\rho_i^2} + \frac{\boldsymbol{\sigma}_j}{\rho_j^2} \right) \frac{\partial W(x_i-x_j,h)}{\partial x_i}$$

Therefore the approximation can finally be written as:

$$\boxed{\left\langle \frac{dv_i}{dt} \right\rangle = m_j \sum_j \left(\frac{\boldsymbol{\sigma}_i}{\rho_i^2} + \frac{\boldsymbol{\sigma}_j}{\rho_j^2} \right) \nabla_i W_{ij}} .$$

As with the mass equations, the symmetrised forms are attractive because they result in exact momentum conservation (Monaghan 1982).

3.2.3   Numerical Implementation

The SPH particle method which has been identified in the previous sections of this chapter approximates the derivatives of a function by summing the contributions from a set of neighbour particles located within a domain of size ςh which is weighted by a kernel function, where ς is a scale-factor. It has already been highlighted that the SPH method does not exhibit any fixed connectivity between the particles and therefore in order to perform the approximations, a set of neighbours must be computed for each particle at every time-step. One of the main considerations for the development of an SPH code is thus the method used to compute the neighbours which contribute to the approximations; since this aspect is the main contributor to the efficiency of the code and its computational cost.

Clearly the simplest method of generating a list of contributing neighbours is to calculate the distance between a particle and every other particle in a problem Then check to see if these distances are less than ςh, storing the ones which satisfy the criteria in a list. The method must be completed for every particle in the problem so that a list of neighbours exists for all particles. This is an extremely inefficient method and as such would be prohibitive to implement for problems with high numbers of particles. This method scales with n^2, where n is the number of SPH particles.

The Cranfield MCM code makes used of a method termed the 'linked-list' method or approach (Hockney, Eastwood 1981), which is commonly used in SPH codes. It is a numerically effective data structure for storing the results of a 'bucket sort', where a bucket sort is an algorithm which partitions an array into a number of individual 'buckets'. In SPH the buckets are regular grids of side ςh and are set up throughout the problem. Lists of the particles contained within each bucket are then stored so that a neighbour search need only consider the particles in the relevant buckets to determine all of the neighbours. Therefore in 1D, 2D and 3D, three, nine and twenty-one buckets only need to be checked for each particle respectively. This is a significantly less computationally expensive approach to considering all the individual inter-particle distances; in contrast to the 'every particle connection' approach highlighted above the linked-list method scales with $n\log n$. In MCM the scale-factor, ς is hard-coded to $\varsigma = 2$ to be consistent with the support of the kernel function.

3.3 SPH Drawbacks

The SPH method has some known problems which are well documented in literature. Since it is the purpose of this research project to develop a fracture model within an SPH framework, it is necessary to understand some of these inherent problems, such that they can be avoided, corrected or accepted. The principle drawbacks which are discussed in this section are the tensile instability (see Section 3.3.1), saw tooth/zero energy modes (see Section 3.3.2) and SPH accuracy and consistency (see Section 3.3.3).

3.3.1 SPH 'Tension' Instability and Numerical Fracture

It was quickly discovered when using SPH for solid mechanics problems that there was an instability in the method when the particles were under tension, i.e. when the particles are under tensile stress[1] their motion becomes unstable. The problem is therefore termed the 'tension' instability and can be visualised as an unphysical clumping of the particles which in some cases leads to early termination of a simulation. Analysis of the problem has determined that the instability depends on the sign of the product of the stress and the second derivative of the SPH kernel function (Swegle, Attaway *et al*. 1995). This condition can be represented mathematically as follows:

$$W'' \sigma > 0 \,,$$

where W'' is the spatial second derivative of the SPH kernel function and σ is the stress. This condition tells us that as long as the above function is positive the method is unconditionally unstable. In practice this instability manifests itself as a clumping of particles that can be clearly seen in simulations. Figure 3.3-1 shows three stages of a 2D plain-strain simulation in which two steel plates travelling at a velocity of 200 ms[-1] in opposite directions impact one another; the constitutive model used is bi-linear elastic-plastic model with isotropic hardening, but no failure criterion and therefore no dislocation or fragmentation should be present. The figure clearly shows the clumping of particles, causing voids to appear in areas of high tensile stress, which was identified above as the manifestation of the problem.

---

[1] For large smoothing lengths this instability is also present in simulations dealing with compressive stress, however under normal operating conditions the instability is only evident in tension and therefore has become known as the 'tension' instability.
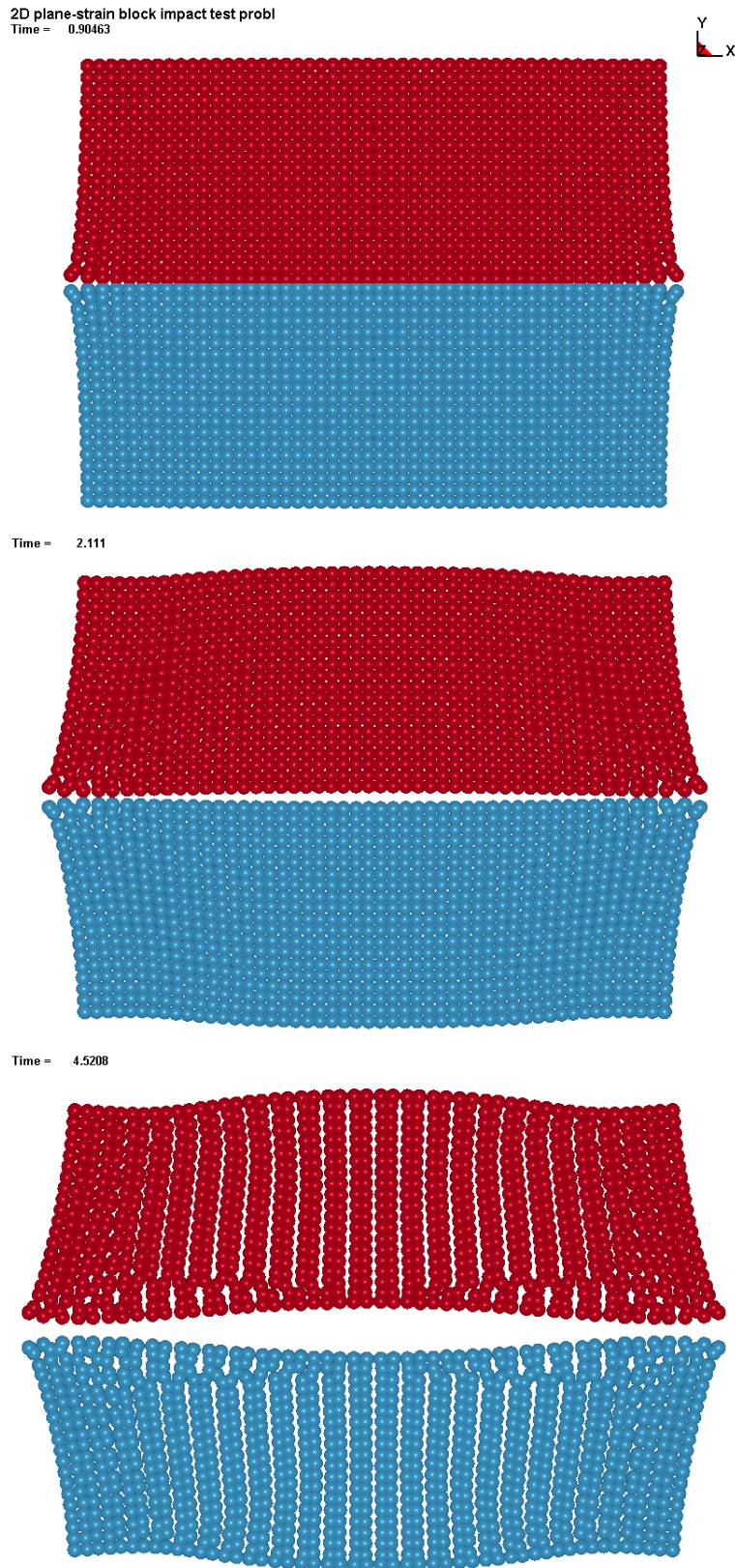
Figure 3.3-1: Visualisation of SPH 'Tension' Instability in a 2D block impact problem. Tension instability manifests as non-physical clumping of particles in the final image (time is in μs).

For in-depth explanations of the SPH tension instability please refer to Belytschko (2000), Swegle *et al*. (1995) and Gourma (2003).

In addition to the tension instability problem there is another relevant problem with an Eulerian SPH formulation known as numerical fracture. Two SPH particles are considered to be neighbours if the distance between them is less than twice the smoothing length (as described previously). Thus all particles within a spherical domain of radius $2h$ are neighbours of the particle at the centre of that domain (See Figure 3.3-2(a)). This domain remains a sphere throughout the calculation, with either a fixed or variable radius. In the case of anisotropic deformation two particles that should remain neighbours can move far enough apart that they no longer influence each other (See Figure 3.3-2(b)). The result in the simulation is fracture that should not be present in the calculation.
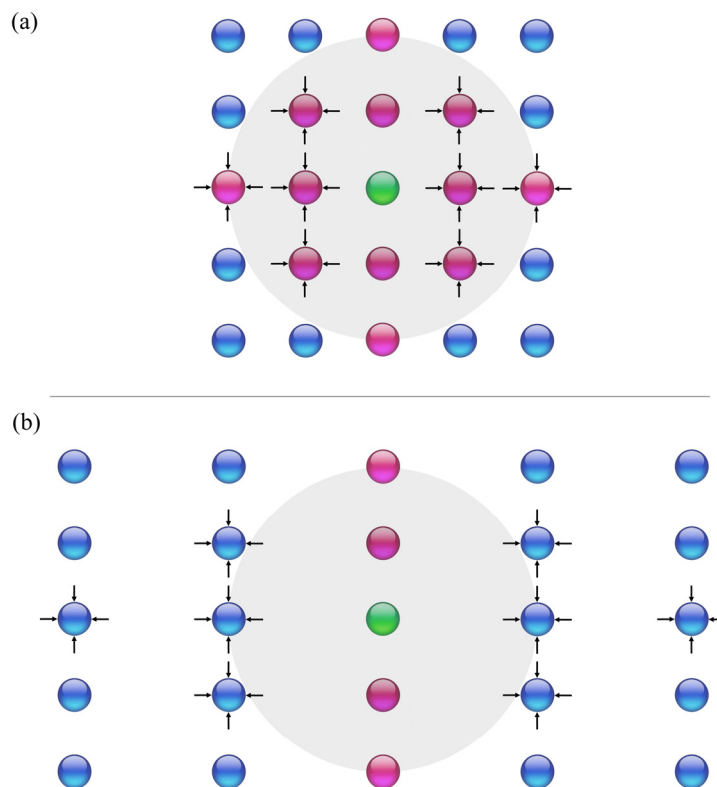
Figure 3.3-2: Diagrammatic representation of the occurrence of numerical fracture in Eulerian SPH simulations. The arrowed particles in (a) are part of the neighbourhood (pink particles) of the central particle (green particle), but in (b) they have moved outside the neighbourhood (blue particles) due to anisotropic deformation, resulting in numerical fracture.

To display this problem, a Taylor impact simulation of a 3.2 mm radius, 32 mm long, copper specimen impacting at 227 ms^{-1}, containing SPH particles in a Hexagonal Close Packed (HCP) formation was run using an Eulerian SPH formulation in MCM; the constitutive model is a bi-linear elastic-plastic law and no fracture model was present in the numerical model; hence the analysis should not predict any fracture. Figure 3.3-3 shows the model at 100 μs; clearly numerical fracture is present where the 'footprint' has split, which has also lead to an overestimate of the footprint diameter.
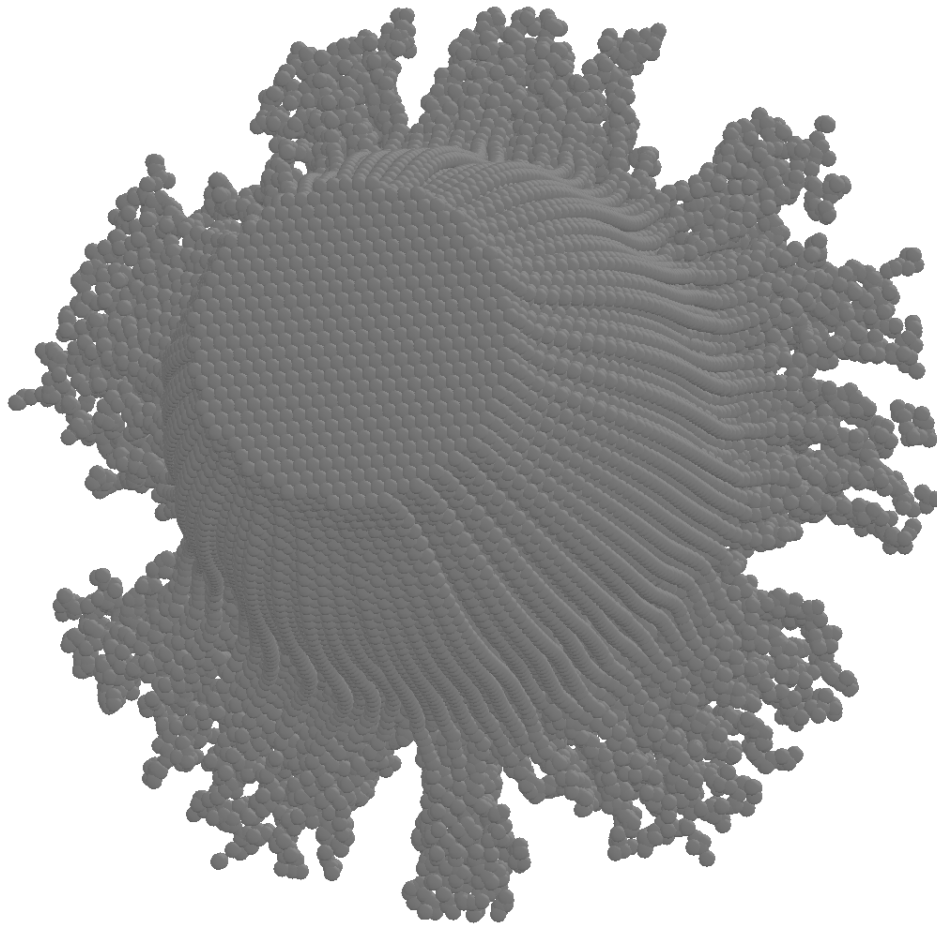
Figure 3.3-3: Visualisation of SPH numerical fracture in a Taylor Cylinder problem.

Without the presence of the tension instability this limitation bears no real problem to a simulation dealing with high strain rate loading. This is because it only becomes apparent when the deformation has become extremely large and therefore one can conclude that fracture and subsequent fragmentation due to a defined fracture model would occur before numerical fracture has begun to manifest itself in the problem. However, if the tension instability is present in the solution, the unphysical clumping of

the particles can lead to early manifestation of numerical fracture. The early presence of numerical fracture in simulations is therefore unacceptable for this research as it will not be possible to distinguish between numerical fracture and fracture calculated by a fracture model.

Therefore as a result of these two limitations on the Eulerian SPH method it is necessary for this research that a solution is made available (ideally an Eulerian SPH approach that is not prone to the tensile instability and rapid onset of numerical fracture) before attempting to accurately describe damage through a fracture model.

3.3.2   Saw Tooth and Zero Energy Modes

Another well known problem with the SPH method, but also occasionally encountered in standard Finite Element methods is the zero energy mode (Liu, Liu 2003). The occurrence of the SPH zero energy mode (sometimes referred to as the saw tooth mode) is the consequence of calculating variables and their derivatives at the same locations (i.e. at an SPH particle). In SPH the zero energy mode manifests as particle displacement which is non-physical, i.e. motion which is not resisted by internal stresses. In the case of an alternating velocity field with a period of twice the inter-particle spacing, a constant stress field is produced at the SPH particles (see Figure 3.3-4). This leads to the calculation of zero accelerations at the particles and thus the motion is not resisted.
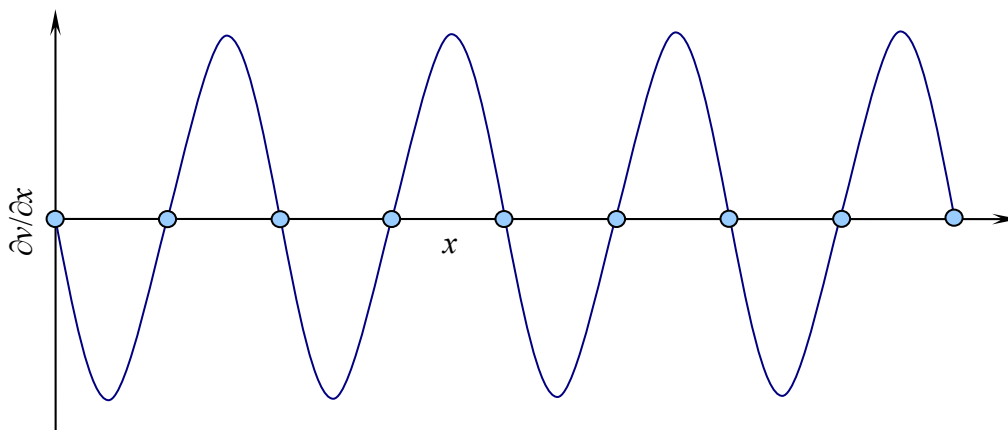
Figure 3.3-4: 1D Velocity gradient field for alternating particle velocities.

This can be demonstrated simply by considering the output of the momentum equation for a simple three particle problem where the centre particle, i is the particle under consideration and the other two are neighbour particles, $+j$ and $-j$ respectively. In the case where the stress at the particles is constant, the acceleration at particle i calculated by the momentum equation is equal to zero. This is because the gradient of the kernel at particle i is zero and asymmetric across the support, i.e.:

$$\nabla_i W\left(x_i - x_{-j}, h\right) = -\nabla_i W\left(x_i - x_{+j}, h\right).$$

Vignjevic *et al.* (2000) explain that the zero energy mode can be avoided by incorporating stress points into the SPH method. In this scheme, variables such as position, velocity and acceleration are still calculated at the particles, but stress is calculated at the centre point between particle-pairs. This produces non-zero accelerations for the case presented above and thus the unphysical motion is resisted. The disadvantage is that such an implementation adds an undesirable level of complexity to the otherwise simple SPH approach.

3.3.3   SPH Consistency

Through derivation of the SPH particle approximation approach (see Section 3.2.1) and the Eulerian SPH forms of the conservation equation (see Section 3.2.2), it was identified that the consistency of the SPH method is affected at the problem domain boundaries.

Figure 3.1-6 and Figure 3.1-7 presented in Section 3.1.2 show that when a particle and its support domain, Φ lies fully within the problem domain, P, the kernel is symmetrical across the support and thus the effect of neighbouring particles is felt by the particle on all sides. In truth, inaccuracies are introduced in problems constructed from non-linearly distributed sets of particles or in problems which have undergone deformation such that the particle spacing is no longer uniform. This is because an uneven number of particles are located either side of the particle in all planes and

therefore the kernel support is not truly symmetrical. This effect is usually considered to be small if sufficient particle density is used to construct the problem domain.

In the case of a particle which lies on or close to a problem boundary, the kernel is asymmetrical and the contribution from the neighbour particles is weighted heavily to one side of a plane only. Figure 3.3-5 shows SPH interpolations of a linear function in 1D using equally spaced SPH particles. In the centre of the body, good interpolation is seen, but near the edge the kernel deficiency is clear.
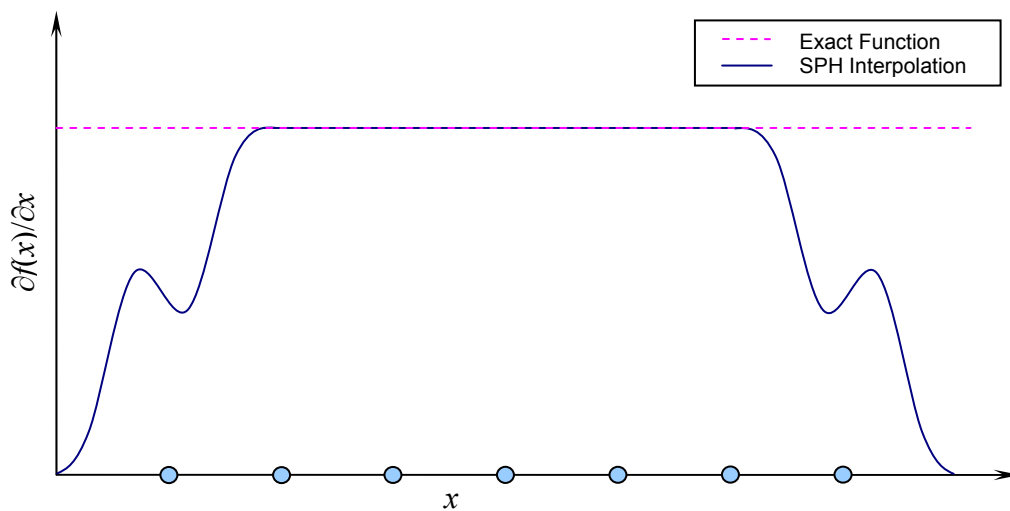
Figure 3.3-5: SPH kernel estimate of the gradient of a linear function (Campbell 1998).

The consequence of this behaviour is that stresses and velocities are approximately zero at the boundaries and thus the free surface boundary condition is approximated naturally by the Eulerian SPH method. The disadvantage is that other parameters such as material density also drop to zero at the boundaries which is unphysical. The results obtained at the boundaries using Eulerian SPH are therefore further approximated and problems tend to show deficiencies in interpolated values at the boundaries.

The particle deficiency can be compensated for by using normalised kernel functions although this introduces significant complexity since rigorous implementation of the boundary conditions must also be employed due to the fact that they are no longer naturally approximated by the method.

3.4   Total Lagrangian SPH Formulation

Belytschko (2000) showed that the tension instability and numerical fracture problems associated with the SPH method (see Section 3.3.1) can be eliminated completely by use of a kernel that is a function of the material (Lagrangian) co-ordinates in the initial (or reference) configuration. This method has been implemented into MCM by Reveles (2007) at Cranfield University and was available for this research. Reveles refers to this method as the 'Total Lagrangian Method' and this is the terminology that has been adopted for this research. The Total Lagrangian approach is briefly described in this chapter. For a more in depth explanation refer to Rabczuk, Belytschko *et al.* (2004) and Reveles (2007).

In the Total Lagrangian approach the spatial co-ordinates in the initial state of the domain (i.e. prior to any deformation taking place) are regarded as the reference state (material co-ordinates). Consequently the solution of the conservation equations are expressed in terms of the material co-ordinates (X), instead of the spatial co-ordinates, (x) as described for the Eulerian SPH method. The relationship between the co-ordinate systems which describes the motion is given as follows:

$$x = \phi(X,t),$$

where ϕ is a mapping function. The reference state is defined at time, $t = 0$:

$$x = \phi(X,0) = X .$$

The displacement, u of a Total Lagrangian SPH particle is therefore given as the difference between its current position and its reference position, i.e.:

$$u = x - X .$$

The deformation gradient tensor, $\mathbf{F}$ is the Jacobian of the matrix of the motion of the particles and has the following form:

$$\mathbf{F} = \frac{\partial x}{\partial X} = \frac{\partial(u + X)}{\partial X},$$

and

$$J = \det \mathbf{F}.$$

The SPH conservation laws are taken from Reveles (2007) as follows (for a derivation of these expressions within the Total Lagrangian SPH framework refer to the cited work):

Conservation of mass:

$$J_0 \rho_0 = J \rho,$$

Conservation of momentum:

$$\left\langle \frac{dv_i}{dt} \right\rangle = \sum_j m_j \left[\frac{\mathbf{P}_i}{\rho_i^2} + \frac{\mathbf{P}_j}{\rho_j^2} \right] \nabla_i W\left(X_i - X_j, h_0 \right),$$

Conservation of energy:

$$\left\langle \frac{dE_i}{dt} \right\rangle = \frac{\mathbf{P}_i}{\rho_i^2} \sum_j m_j \left(v_j - v_i \right) \nabla_i W\left(X_i - X_j, h_0 \right) V_{0j},$$

where ρ_0 and h_0 are the density and the SPH smoothing length in the reference configuration, $\mathbf{P}$ is the nominal stress tensor ($\mathbf{P} = \left(J\mathbf{F}^{-1}\boldsymbol{\sigma} \right)$), v is the velocity and V_{0j} is the initial volume of the j th neighbour. Note that the gradient of the kernel function, W is given in terms of the material co-ordinates at the reference configuration.

Since all calculations are performed in the reference state the neighbourhood therefore does not change, but rather remains fixed throughout the computation. This method is valid for as long as the mapping exists between the domains (i.e. J is positive and real), when large shear deformations are present this cannot be achieved and thus the limitation of the method is defined.

Figure 3.4-1 on the following page shows similar stages of the 2D plain-strain simulation explained in Section 3.3.1, for the simulation run with the Total Lagrangian Formulation. No clumping of particles associated with the tension instability is present.

Total Lagrangian Formulation also eliminates the problem of numerical fracture described at the end of Section 3.3.1, because the neighbourhood is fixed and therefore particles which move far apart from one-another do not stop influencing each other.

It has therefore been shown by Belytschko (2000) and Reveles (2007) that the Total Lagrangian correction to the SPH method is valid for removing the tension instability and numerical fracture problems inherent with Eulerian SPH.
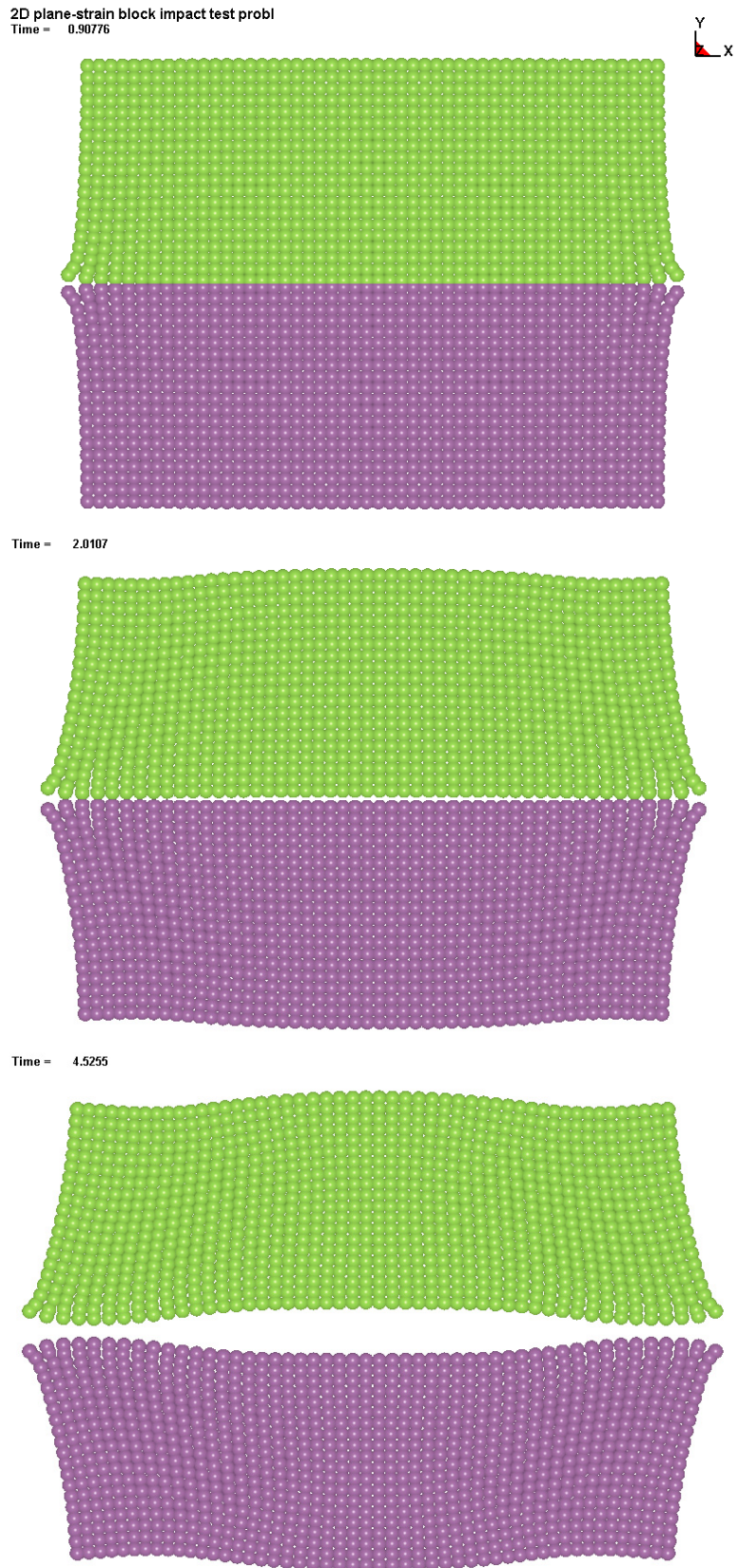
Figure 3.4-1: 2D block impact problem using Total Lagrangian Formulation. The material remains stable in tension (time is in µs).

3.4.1 Stability Analysis of Total Lagrangian Particle Equations

It was stated previously that the Total-Lagrangian SPH formulation is not subject to the tensile instability which is present in the Eulerian SPH particle equations (see Section 3.3.1 for more information). Reveles (2007) presents a stability analysis which identifies the stability criteria of the Total Lagrangian SPH particle equations. A summary of this analysis is presented below to provide a basis for discussing the impact of the proposed fracture model developed for this research on the properties of the SPH method. For a full derivation of the stability analysis presented herein, and also of the Eulerian SPH particle equations as well as continuum equations in general, refer to Reveles (2007).

The purpose of the stability analysis is to determine if a perturbation will decay (stable) or grow (unstable) as time progresses when introduced into the Total Lagrangian SPH particle equations. The perturbation is defined as follows:

$$\tilde{u} = u e^{\mathbf{i}(\varpi t + \kappa X)},$$

where u is displacement and $\tilde{u}$ represents the addition of the perturbation, although both u and $\tilde{u}$ could be replaced with any physical field variable. κ is the perturbation wave number and ϖ is the complex frequency of the perturbation, which is a function of real and imaginary components, i.e.:

$$\varpi = \omega_{\Re} + \mathbf{i}\omega_{\Im},$$

where $\mathbf{i}$ is the imaginary unit. Thus:

$$\tilde{u} = u e^{\mathbf{i}\left((\omega_{\Re} + \mathbf{i}\omega_{\Im})t + \kappa X\right)}$$
$$= u e^{\left(\mathbf{i}\omega_{\Re}t + \mathbf{i}^2\omega_{\Im}t + \kappa X\right)}$$
$$= u e^{\left(\mathbf{i}(\omega_{\Re}t + \kappa X) + (-1)\mathbf{i}\omega_{\Im}t\right)}$$

therefore:

$$\tilde{u} = u e^{-\omega_{\Im}t} e^{\mathbf{i}(\omega_{\Re}t + \kappa X)}.$$

The analysis is performed with the SPH momentum equation since the process is considered to be isothermal and adiabatic and thus the energy equation is irrelevant, also the continuity equation has a simple algebraic form. In Section 3.2.2.2 the Eulerian SPH momentum equation was derived from the following Navier-Stokes continuum equation:

$$\frac{dv}{dt} = \frac{d^2u}{dt^2} = \frac{1}{\rho}\frac{\partial \boldsymbol{\sigma}}{\partial x} \ .$$

Following a similar process, the Total Lagrangian form in terms of a force (via Newton's Second Law) with external and body forces neglected can be expressed by the following discrete particle equation:

$$m_i\left(\frac{d^2u_i}{dt^2}\right) = -\sum_j \frac{m_j}{\rho_{0j}}\mathbf{P}_j\frac{\partial W\left(X_i - X_j, h_0\right)}{\partial X_j} = -\sum_j \frac{m_j}{\rho_{0j}}\mathbf{P}_j\nabla_j W_{ij} \ .$$

By replacing the displacement, u with the perturbed displacement, $\tilde{u}$ the above becomes:

$$m_i\left(\frac{d^2\tilde{u}_i}{dt^2}\right) = -\sum_j \frac{m_j}{\rho_{0j}}\tilde{\mathbf{P}}_j\nabla_j W_{ij} \ ,$$

where $\tilde{\mathbf{P}}$ is the corresponding perturbed stress. By substituting $\tilde{u} = ue^{-\omega_\Im t}e^{\mathbf{i}(\omega_\Re t + \kappa X)}$ in the above equation and observing that in 1D:

$$\mathbf{P} = \mathbf{CF}\tilde{\mathbf{F}}\mathbf{F} + \mathbf{S}\tilde{\mathbf{F}} = \left(\mathbf{CF}^2 + \mathbf{S}\right)\tilde{\mathbf{F}} = \overline{\mathbf{C}}\tilde{\mathbf{F}} \ ,$$

where $\mathbf{F}$ is the deformation gradient tensor, $\tilde{\mathbf{F}}$ represents the perturbed state, $\mathbf{S}$ is the deviatoric stress tensor and $\mathbf{C}$ is the material tangent modulus; it is possible to derive an expression for the frequency of the perturbation to determine the conditions for stability.

If the imaginary part of the perturbation, $e^{-\omega_\Im t}$ is considered to be zero, the perturbation frequency may be expressed as:

$$\omega_\Re^2 = \frac{1}{m_i} \sum_j \frac{m_j}{\rho_{0j}} \nabla_j W_{ij} \overline{\mathbf{C}} \sum_k \left(e^{\mathbf{i}\kappa(X_k - X_i)} \right) \nabla_k W_{jk} \,,$$

where k denotes the neighbourhood of the j particle.

Through Euler's formula:

$$e^{\mathbf{i}\kappa(X_k - X_i)} = \cos\left(\kappa\left(X_k - X_i\right)\right) + \mathbf{i}\sin\left(\kappa\left(X_k - X_i\right)\right),$$

which provides the following expression for the real part of the perturbation frequency:

$$\omega_\Re^2 = \frac{1}{m_i} \sum_j \frac{m_j}{\rho_{0j}} \nabla_j W_{ij} \overline{\mathbf{C}} \sum_k \cos\left(\kappa\left(X_k - X_i\right)\right) \nabla_k W_{jk} \,.$$

From the above, it can be concluded that a spurious singular mode is present when:

$$\kappa = \frac{\pi}{2\left(X_k - X_i\right)} \quad \text{i.e.} \quad \cos\left(\kappa\left(X_k - X_i\right)\right) = \cos\left(\frac{\pi}{2}\right) = 0,$$

which is representative of the cut-off wavelength for regularly spaced particles by $\left(X_k - X_i\right)$ and occurs irrespective of the stress contained in $\overline{\mathbf{C}}$.

However, when the imaginary part of the perturbation exists, the amplitude of the perturbation is bounded by the $e^{-\omega_\Im t}$ term.

It can therefore be concluded that the imaginary part of the complex frequency drives the stability criteria.

Thus the stability of the Total Lagrangian SPH particle equations can therefore be summarised as follows:

- When $\omega_3 > 0$ the particle equations remain stable since $e^{-\omega_3 t}$ decays as t increases.

- When $\omega_3 < 0$ the particle equations are un-stable since $e^{-\omega_3 t}$ grows as t increases.

- When $\omega_3 = 0$ the particle equations have the potential for instability via a spurious singular mode identified above.

The first two bullet points are also characteristics of the continuum equations and therefore it can be stated that the stability of the Total Lagrangian SPH particle equations is closely related to the stability of the continuum itself; the difference being the occurrence of the spurious singular mode corresponding to the cut-off wavelength term present in particle equations.

Reveles (2007) also produces an equivalent stability analysis of the Eulerian SPH particle equations and identifies a condition which is the SPH tensile instability as identified by Swegle, Attaway *et al.* (1995), discussed in Section 3.3.1. This condition is not present in the Total-Lagrangian stability analysis summarised above and therefore it can be concluded that it is not subject to the Eulerian SPH tensile instability.

3.5   Other Meshless Numerical Schemes

The SPH meshless method was pre-selected for this research since an existing code developed at Cranfield University was available for immediate use. It is therefore not the purpose of this thesis to provide a comprehensive record of all the emerged meshless numerical schemes, but rather to present a brief explanation of the meshless numerical methods that were introduced in Chapter 2 as part of the literature survey into existing techniques to model fracture using computational mechanics. The following sub-sections explain the fundamentals of the Element Free Galerkin (EFG) method, the

Material Point Method (MPM) and the Reproducing Kernel Particle Method (RKPM) and highlight some communalities and differences with the SPH method in each case.

3.5.1 The Element Free Galerkin Method (EFG)

The EFG method is based on the Galerkin technique employed in Finite Element (FE) codes; its main feature is that it only requires a description of the problem geometry (i.e. relevant boundary conditions) and a set of nodes to construct the discrete equations. The main difference to the FE method is that the interpolation within an element is replaced by polynomial equations which are fit to the nodal values by a moving-least squares approximation (Belytschko *et al.* 1994, 1995a).

The approximation $u^h(x)$ to an arbitrary function $u(x)$ is expressed as follows:

$$u^h(x) = p^T(x)a(x),$$

where $p^T(x)$ is a polynomial of arbitrary order defined in the spatial coordinates $x^T = [x, y]$ and $a(x)$ are the corresponding coefficients. The coefficients are obtained by discretising the above over n points belonging to a 'neighbourhood' (domain of influence), Ω_I of x, using a weight function $w_I(x)$, where

$$w_I(x) = w(x - x_I) \neq 0.$$

Thus:

$$S = \sum_I^n w_I(x) \left[p^T(x_I) a(x_I) - u_I \right]^2,$$

where u_I is the nodal value at $x = x_I$. It should be noted that $u^h(x) \neq u(x)$ and therefore the boundary conditions are not directly satisfied, hence weak-form Lagrange multipliers are used to enforce them (Viana *et al.* 2007).

The derivative of the arbitrary function $u(x)$ is defined as follows:

$$\frac{du^h(x)}{dx} = \frac{dp^T(x)}{dx}a(x) + p^T(x)\frac{da(x)}{dx} \; .$$

In order to perform the numerical integration, the EFG method uses an auxiliary cell structure to enable the quadrature points to be defined. This is depicted diagrammatically in Figure 3.5-1.
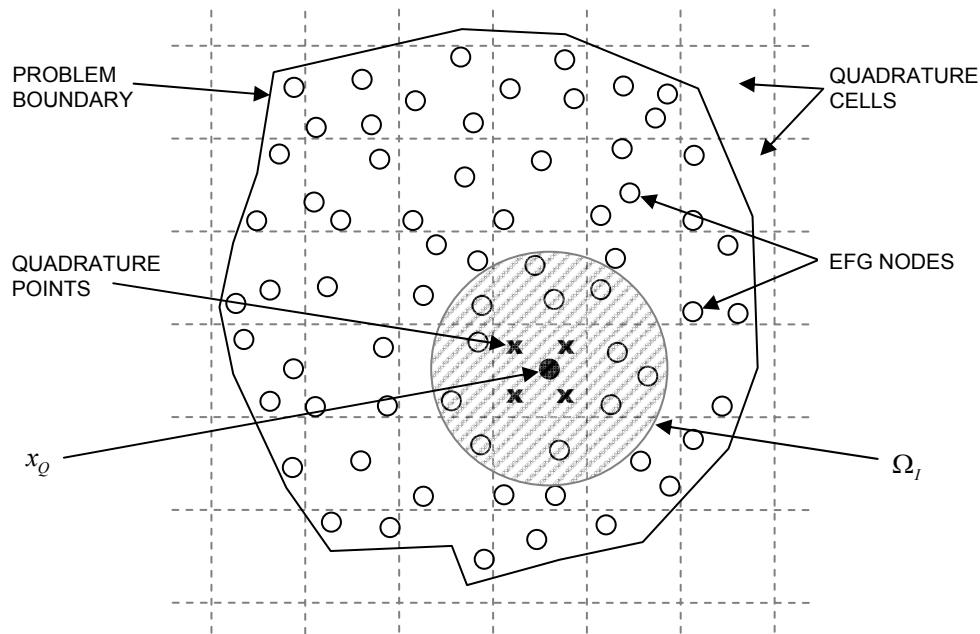
Figure 3.5-1: Diagrammatic representation of an EFG problem, showing the background cell structure and the domain of influence, Ω_I for an arbitrary point x_Q.

The EFG method is reported to achieve high accuracy although it exhibits increased computational costs (Belytschko *et al.* 1994). A version of the EFG method is available in the LS-DYNA commercial software package. The method is similar to the SPH method in that the integration is performed over a 'neighbourhood' of nodes using a weighted function (similar to the SPH kernel), although the SPH method does not require the use of a background cell in the calculations. The use of quadrature cells has caused many authors to state that the EFG method is not truly meshless (Viana *et al.* 2007).

3.5.2   The Material Point Method (MPM)

The Material Point Method is an extension of the Particle-In-Cell (PIC) method developed by Harlow (1964). For a similar reason to the EFG method (discussed in Section 3.5.1), the MPM is often disputed as being a true meshless method since it uses an Eulerian background grid to move the material points (or 'particles'). The mixed use of an Eulerian mesh and Lagrangian material points puts the MPM into the category of Arbitrary Lagrangian Eulerian (ALE) methods.

In the PIC method a series of Lagrangian particles which carry properties such as position and velocity are used to describe the motion and an Eulerian mesh is used to interpolate the information between the particles and Eulerian nodes, i.e. the density and velocity for a mesh cell can be obtained by summing the contributions from the particles within the cell. The MPM differs from the PIC in that the governing equations are formulated in the weak form (synonymous with other Galerkin methods) and provision is provided for constitutive models to be formulated on the individual material points. The governing equations of motion for the MPM as described by Sulsky *et al.* (1994) are given as follows:

Momentum: $\displaystyle\int_{\Omega} \rho \left[\mathbf{w}\frac{\partial v}{\partial t} + \boldsymbol{\sigma}^s \frac{\partial \mathbf{w}}{\partial x} \right] dV = \int_{\Omega} \rho \left[\mathbf{w}\,b \right] dV + \int_{\partial\Omega} \left[\mathbf{w}\,\tau \right] dS$,

Mass: $\displaystyle\int_{\Omega} \rho \mathbf{W} \left[\frac{\partial \boldsymbol{\sigma}^s}{\partial t} - \mathbf{T}^s \frac{\partial \mathbf{e}}{\partial t} \right] dV = 0$,

Energy: $\displaystyle\int_{\Omega} \rho \mathbf{W}^* \left[\frac{\partial \mathbf{e}}{\partial t} - \frac{1}{2}\left(\frac{\partial v}{\partial x} + \left(\frac{\partial v}{\partial x} \right)^{\mathrm{T}} \right) \right] dV = 0$,

where the integrals are taken over the current configuration, Ω. ρ is the density, $\boldsymbol{\sigma}^s$ is the specific stress tensor, b is the body force per unit mass, τ is the prescribed part of the traction on the surface $\partial\Omega$, $\mathbf{T}^s$ is the specific tangent modulus tensor, $\mathbf{e}$ is the strain

tensor and v is the velocity. $\mathbf{w}$, $\rho\mathbf{W}$ and $\rho\mathbf{W}^*$ are weighting functions and dV and dS denote the differentials of the volume and the surface respectively. The momentum equation is solved on the Eulerian grid, whereas the mass and energy equations are solved at the material points. Figure 3.5-2 shows a diagrammatic representation of the MPM Eulerian mesh and Lagrangian particles.
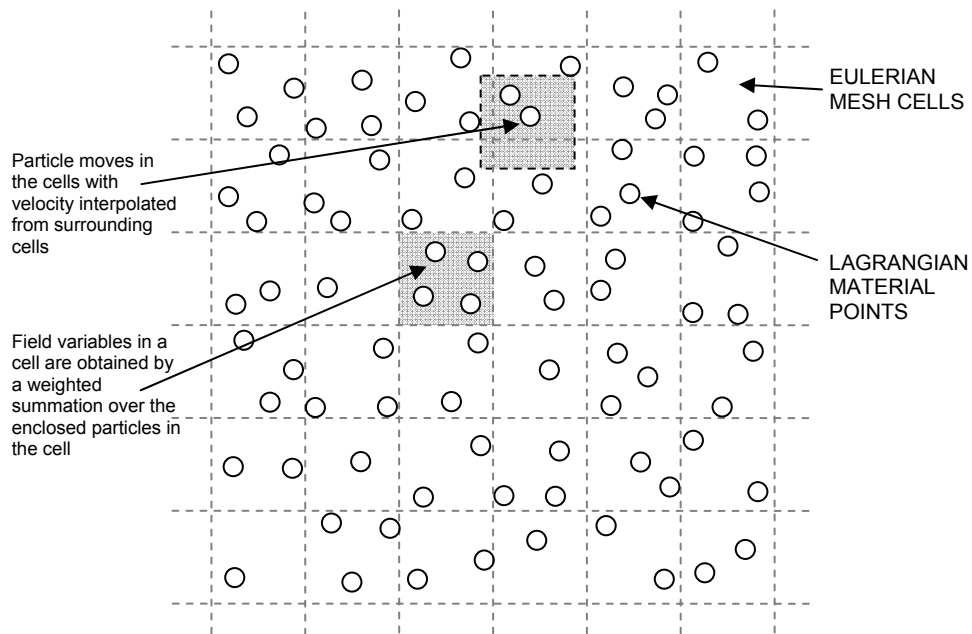
Figure 3.5-2: Diagrammatic representation of a typical MPM grid and material points (Liu, Liu 2003).

The MPM is similar to the SPH method in that it uses individual particles to hold material data, although the MPM particles are not used to approximate the field variables. The particle information is interpolated from or to the Eulerian mesh cells whereas in the SPH method no mesh is required since particle information is obtained directly by solving the governing equations. Fundamentally in the MPM method there is no interaction between the individual material points which is key to the operation of the SPH method.

3.5.3   The Reproducing Kernel Particle Method (RKPM)

As was shown with the SPH method in Section 3.1.1 an arbitrary function, $f(x)$ in a domain of interest, Φ may be approximated by use of a kernel function, W as follows:

$$\langle f(x)\rangle = \int_{\Phi} f(x')W(x-x')dx',$$

where $\langle f(x)\rangle$ depends only on values of the function at a set of discrete material points ('particles') which lie in the sub-domain for which $W(x-x')$ has non-zero values (termed the kernel support).

Also discussed earlier in this chapter were the shortcomings of the SPH method; one such shortcoming was the consistency at the problem boundaries (see Section 3.3.3 for more information). Investigation into a solution to this problem led to the formulation of the Reproducing Kernel Particle Method (RKPM) developed by Liu *et al*. (1995b, see also 1995a, 1996 and 1997). In the RKPM the kernel is modified by introducing a correction function, C into the formulation which has the purpose of enhancing the accuracy of the solution near to or on the problem boundaries. The reproduced kernel estimate of $f(x)$ can therefore be rewritten as follows:

$$\langle f(x)\rangle = \int_{\Phi} f(x')C(x,x-x')W(x-x')dx'.$$

The correction function restores first order linear completeness of the kernel function (Rabczuk, Belytschko *et al*. 2004), i.e. the kernel obtains the consistency conditions across the entire domain. Liu *et al*. (1995a) reports that the addition of the correction function also completely eliminates the presence of the SPH tensile instability (see Section 3.3.1 for more information).

Clearly the similarity of the RKPM to the SPH method is that the SPH formulation is used in its entirety and the difference is the addition of the correction function and the

structure of the kernel function. The RKPM can therefore be considered to be an extension (or improvement) to the SPH method. The main disadvantage to the RKPM is that the correction function adds extra computational cost and so a trade-off exists between the shortcomings in the SPH method providing a less accurate solution and the extra time required for simulations in the RKPM.

3.6   Use of SPH within the Research

This thesis makes use of both of the SPH formulations that have been identified and discussed in this chapter.

In the early design phases of the fracture model developed for this research, it was unclear which SPH formulation would be used. Initial investigations pointed to the Eulerian SPH method since simulations of explosively driven experiments outlined in Chapter 6 were being conducted as potential validation cases for the fracture model. Since they require the ability to model extremely large deformations, the total Lagrangian formulation was unsuitable. However Section 3.3.1 in this chapter has identified an instability in the standard Eulerian approach which manifests when a problem is under tension. This needs to be corrected if a robust fracture model is to be implemented into the code since is not desirable to have an element of fracture present in the solution which is due to numerical instabilities, as well as fracture due to a computational model. This is because one will inevitably interfere with the other and it will also be difficult to quantify the exact response of the fracture model.

A significant study was performed into different approaches that have been suggested to correct the Eulerian SPH draw-backs which is presented and discussed in Chapter 5. Unfortunately the study did not yield a stable Eulerian code and so the Total Lagrangian formulation was used for the fracture modelling research. This has meant that the explosively driven experiments have not been modelled with the fracture model discussed in Chapters 7 and 8.

There are examples in literature which make use of mixed Eulerian-Lagrangian formulations. One such implementation is detailed by Rabczuk and Belytschko (2007)

for use with their particle method within an EFG framework. They chose to represent fracture at the individual particles and split a particle into two once it had reached a defined limit. A Lagrangian kernel is used for the initial stages of a fracture simulation, i.e. while the damage is growing. This prevents the tensile instability from causing unphysical numerical fracture and skewing the effect of the damage model. Once a particle has reached a material limit indicating fracture, the kernel is switched to an Eulerian formulation, to enable the stable modelling of large separations. Particles which have not fractured continue to use a Lagrangian kernel, but exclude particles on the opposite side of the crack from the neighbourhood.

A mixed kernel implementation that enables either kernel to be selected for individual materials is required in the Cranfield code to allow the fracture model to be tested on the explosively driven experiments. However such an implementation is outside the scope of this research project and is therefore left as a recommendation for future work.

4   Constitutive and Explosion Modelling

The accuracy of any fracture model is critically dependent on the underlying constitutive model and therefore one of the first steps in this project was to identify the constitutive model(s) to be used in the research. Four candidate models were considered, the Johnson-Cook constitutive model (Johnson, Cook 1983), the Steinberg-Guinan model (Steinberg, Cochran *et al*. 1980), the Mechanical Threshold Stress (MTS) model (Follansbee, Kocks 1988) and the more recent Preston-Tonks-Wallace (PTW) model (Preston, Tonks *et al*. 2003). At the start of this research project, only the Johnson-Cook model was already implemented in MCM; this research requires the ability to model high strain rate loading and therefore the Johnson-Cook model is not well suited since it is only valid up to ~1000 s^{-1}. The Steinberg-Guinan model was considered as it is the constitutive model commonly used by the AWE; it was decided to implement the model to allow for possible future comparison with current AWE codes. The MTS model has also been implemented into MCM, after it was decided to be the more sensible option of the other two, following an evaluation of comparison between them by Banerjee (2005a).

This chapter begins with a brief explanation of J_2 plasticity theory since this is the underlying motivation for a constitutive model (see Section 4.1). An explanation of the three chosen constitutive models is then provided and the implementation of the Steinberg-Guinan and MTS models into the MCM code is detailed with baseline verification in each case (see Sections 4.2, 4.3 and 4.4). Section 4.5 briefly explains the comparison that was made between the MTS and PTW models for the selection process. For completeness an explanation of the PTW model is included in Appendix C.

This research has required the facility to simulate explosive detonation products and therefore an explosion model and the J.W.L. equation of state were also implemented into MCM as part of this research, this is explained in Section 4.7 along with the procedure that was used to verify the implementation. The process of implementing the models and equation of state outlined above provided capability for the fracture model development, but also aided an understanding into the set-up, manipulation and use of

the software and facilitated in learning the FORTRAN programming language. Section 4.6 covers this topic in more detail. This research has also made use of the Gruneisen equation of state and an explanation of this is also provided in Section 4.8.

4.1 J_2 Plasticity Theory

In order to solve solid mechanics problems concerning the behaviour of a material under external actions the relationships between stresses and strains within a material need to be understood and approximated appropriately. This leads to the concept of a yield surface to describe the stress state inside a material. The yield surface is defined by Philips and Sierakowski (1965) as "*the boundary of a region in stress space in which both loading and reloading produce elastic strains only*". Therefore when the stress lies on the yield surface, the material has reached its yield point and further loading produces plastic strains. In solid mechanics codes a yield function is used to represent the evolution of the yield surface, it is an equation in terms of the yield stress in tension or flow stress.

Malvern (1969) explains that in solid mechanics the yield point of a material can be described by the J_2 Plasticity (or J_2 Flow) Theory (also sometimes referred to as the von Mises yield criterion). It states that plastic deformation begins in a material once the second invariant of the deviatoric stress tensor, J_2 reaches the yield stress in pure shear, k, thus:

$$J_2 = k^2 .$$

In a state of uniaxial stress the yield stress in tension (flow stress), σ_y is equal to $\sqrt{3}$ times the yield stress in pure shear, i.e. $\sigma_y = \sqrt{3}k$. This leads to the derivation of the formula for the yield stress as follows:

$$\sigma_y = \sqrt{3}\sqrt{J_2} = \sqrt{3J_2} .$$

The above equation represents a common form of the yield condition. Thus for an elastic-plastic material the following condition can be defined for the effective stress ($\sqrt{3J_2}$):

$$\sigma_y \geq \sqrt{3J_2}$$

A typical elastic-plastic constitutive model (e.g. JC, SG, MTS, etc., see following sections) calculates a value of the current yield, or plastic flow stress based on the current conditions and load history of the material, then compares this value with the current effective stress and, if required, enforces the inequality through a remap algorithm such as the radial return algorithm.

Early developments to the J_2 plasticity theory which were the precursor of modern constitutive relationships are also explained in Malvern (1969). The Levy-Mises Perfect Plasticity Model assumes that a material has an infinite elastic stress and that any strain is due to plastic deformation only. It also introduced the assumption that plasticity only changes the shape of a material, i.e. it is incompressible under plastic strains and that changes in volume are due to elastic strains only. This is known as the incompressibility condition. Prandtl and Reuss provided a development to the Levy-Mises model with their elastic, plastic equations which decompose the total strain into elastic and plastic components; their model maintains the assumption of plastic incompressibility. All modern constitutive models assume Prandtl-Reuss Plasticity.

Since the introduction of J_2 plasticity theory, many developments to constitutive relationships have been made, such as the introduction of viscoplasticity (or rate-dependence) in the calculation of the plastic flow stress, as well as other parameters such as temperature. These more complicated models often attempt to incorporate the effects of macroscopic behaviour onto the continuum level to compute the flow stress. Numerous constitutive models have been developed for a wide variety of materials and have been used in many different hydrocodes to date. The following sections describe the various constitutive models which have been utilised for this research.

4.2   The Johnson-Cook Constitutive Model

A form of the Johnson-Cook model described by Johnson and Cook (1983) is implemented in the LLNL-DYNA3D finite element hydrocode (Material Model 15) (Lin 2004). At the start of this research the Johnson-Cook model had already been implemented into the Cranfield SPH code (MCM) from the LLNL-DYNA3D code and was available for immediate use.

The Johnson and Cook model expresses the yield stress (equivalent von Mises flow stress) as

$$\sigma_y = \left[A + B\left(\bar{\varepsilon}^p \right)^n \right]\left[1 + C\ln\left(\dot{\varepsilon}^* \right) \right]\left[1 - \left(T^* \right)^m \right],$$

where A, B, C, m and n are input constants. The flow stress is therefore a function of the effective plastic strain, $\bar{\varepsilon}^p$ when $\dot{\varepsilon}^* = 1.0$, and $T^* = 0$. The other two expressions in square brackets add dependency on the strain rate, $\dot{\varepsilon}^*$ and temperature, T^*. The model is valid for strain rates that lie between $10^{-3} \leq \dot{\varepsilon} \leq 10^3$ $\left(\text{s}^{-1} \right)$ (Panov 2006).

The strain rate, $\dot{\varepsilon}^*$ is non-dimensional and calculated from

$$\dot{\varepsilon}^* = \frac{\dot{\bar{\varepsilon}}^p}{\dot{\varepsilon}_0},$$

where $\dot{\varepsilon}_0$ is a reference strain rate and $\dot{\bar{\varepsilon}}^p$ is the effective plastic strain rate. The homologous temperature, T^* is calculated as follows:

$$T^* = \frac{\left(T - T_r \right)}{\left(T_m - T_r \right)},$$

where T is the current temperature, T_r is the room temperature and T_m is the melt temperature.

4.3   The Steinberg-Guinan Constitutive Model

A form of the Steinberg-Guinan model described by Steinberg, Cochran *et al*. (1980) is implemented in the LLNL-DYNA3D finite element hydrocode (Material Model 11). Cranfield University has the full source for the 2004 release of this program and the program structure of the Cranfield University SPH MCM solver is based on this software. The advantage to this is twofold, as it simplifies the process of numerical implementation and allows validation of the implementation through comparison with LLNL-DYNA3D results. This section outlines the LLNL-DYNA3D implementation, as it is slightly modified from the original Steinberg-Guinan model and the equation parameters are expressed slightly differently.

The Steinberg-Guinan model is essentially a modified elastic-perfectly plastic constitutive model that accounts for the effects of pressure, p temperature, T and equivalent plastic strain, $\bar{\varepsilon}^p$ on the yield strength, σ_y and shear modulus, G of a material. The model is valid for strain rates in excess of $10^5 \, \text{s}^{-1}$, i.e. $\dot{\varepsilon} \geq 10^5$. The constitutive relationships are given as follows:

$$G = G_0\left[1 + bpV^{\frac{1}{3}} + h(T - 300)\right]e^{\frac{fE_i}{E_m - E_i}}$$

$$\sigma_y = \sigma_0'\left[1 + b'pV^{\frac{1}{3}} + h(T - 300)\right]e^{\frac{fE_i}{E_m - E_i}}$$

where

$$\sigma_0' = \sigma_0\left[1 + \beta\left(\bar{\varepsilon}^p + \gamma_i\right)\right]^n,$$

subject to the limitation that $\sigma_0' < \sigma_m$, otherwise $\sigma_0' = \sigma_m$ and

$$\frac{E_i - E_c}{3R'} = T, \quad (3)R' = \frac{(3)R\rho}{A} = C, \quad h = \frac{G_T'}{G_0}, \quad b = \frac{G_p'}{G_0}, \quad b' = \frac{Y_p'}{Y_0}, \quad V = \frac{1}{\eta}.$$

Where $\bar{\varepsilon}^p$ is the compression, defined as the initial specific volume, v_0 divided by the specific volume, v. β and n are work-hardening parameters, and γ_i is the initial

plastic strain, which is normally equal to zero. Parameters with the subscript 0 are the reference state of that parameter ($T = 300\,\text{K}$, $P = 0$, $\varepsilon = 0$) and parameters which are primed and have either the subscript P or T, are derivatives of that parameter with respect to pressure or temperature at the reference state. ρ is the material density, Y is the yield strength, E_i is the internal energy, E_c is the cold compression energy, E_m is the melting energy, R is the universal gas constant, C is the specific heat, A is the atomic weight and η is a compression variable. σ_m is a material specific yield stress work hardening limit.

The difference from the original model is that the LLNL-DYNA3D implementation contains an extra $e^{\frac{fE_i}{E_m - E_i}}$ term at the end of both constitutive relations, where f is an energy exponential coefficient. This term is easily removed to make the model identical to the model by Steinberg, Cochran *et al*. (1980); simply by making $f = 0$ in the input file. This term has been determined as an improvement to the model's temperature response (Ball 2006). The purpose of this term is to reduce the yield strength and shear modulus smoothly to zero at the melt temperature, the usual value for f is -0.001. The term his minimal effect until the temperature is within a few degrees of the melt temperature.

LLNL-DYNA3D defines the cold compression energy as follows:

$$E_c(x) = \int_0^x p.dx$$

Where

$$x = 1 - V = 1 - \frac{1}{\eta}$$

The equation is integrated using the initial energy, E_0 and pressure, P_0 conditions that correspond to zero Kelvin. These parameters are given by the following equations:

$$E_0 = -3R'T_{room} = CT_{room}$$

and

$$P_0 = \gamma_0 E_0 \text{Debye}\left[\frac{\theta}{T_{room}}\right],$$

where T_{room} is room temperature, $\text{Debye}[\ ]$ is the Debye correction factor, γ_0 is the thermodynamic constant at the reference state and θ is the Debye coefficient, ($\text{Debye} = 1$ when $\theta = 0$). The Debye correction is also used to improve the model's temperature response. It can be ignored if $T_{room} = 300\,\text{K}$ and $\theta = 0$, (specified in the input).

The calculation of the cold compression energy curve in LLNL-DYNA3D is performed numerically using the equation of state selected for the material. Due to the expense of this calculation, it is performed only once at the start of an analysis and the curve is approximated by a polynomial of up to order 10. Alternatively the coefficients of the polynomial can be specified by the analyst.

Steinberg, Cochran *et al*. (1980) explain that it is necessary to have some means of turning off the material strength once it has melted and so they also include a melt model to describe this behaviour, they use a simplified GRAY model (Royce, LLNL 1971). In LLNL-DYNA3D this melt model is expressed as follows:

$$E_m(x) = E_c(x) + 3R'T_m(x),$$

where

$$T_m(x) = \frac{T_{m0}e^{2ax}}{V^{2(\gamma_0 - a - \frac{1}{3})}},$$

where T_{m0} is the melt temperature at the initial density, ρ_0 and a is a thermodynamic constant.

In LLNL-DYNA3D, the onset of damage is described by a choice of three spall models implemented at the end of the Steinberg-Guinan subroutine. Since it is the purpose of

this research to develop a fracture model, a description of the function of these models is not given here. The models have however, been implemented into the MCM code to in order to facilitate the validation of the model in MCM by comparing results from two similar problems run in both MCM and LLNL-DYNA3D. For more information on these models please refer to the relevant part of the Material Definition section in the LLNL-DYNA3D user manual (Lin 2004).

4.3.1 Steinberg-Guinan Model Implementation into MCM

The MCM implementation of Steinberg-Guinan is essentially a line for line copy of the LLNL-DYNA3D version of the model allowing for changes due to a significantly different database structure in each code. The implementation was verified using a simple 2D plain-strain simulation in which two steel plates travelling at a velocity of 200m/s in opposite directions impact one another. A series of particles/elements within the model were selected as time history particles/elements and values for y-velocity, effective plastic strain, pressure and x-displacement were plotted. Figure 4.3-1 below shows the locations of the particles/elements within the model.
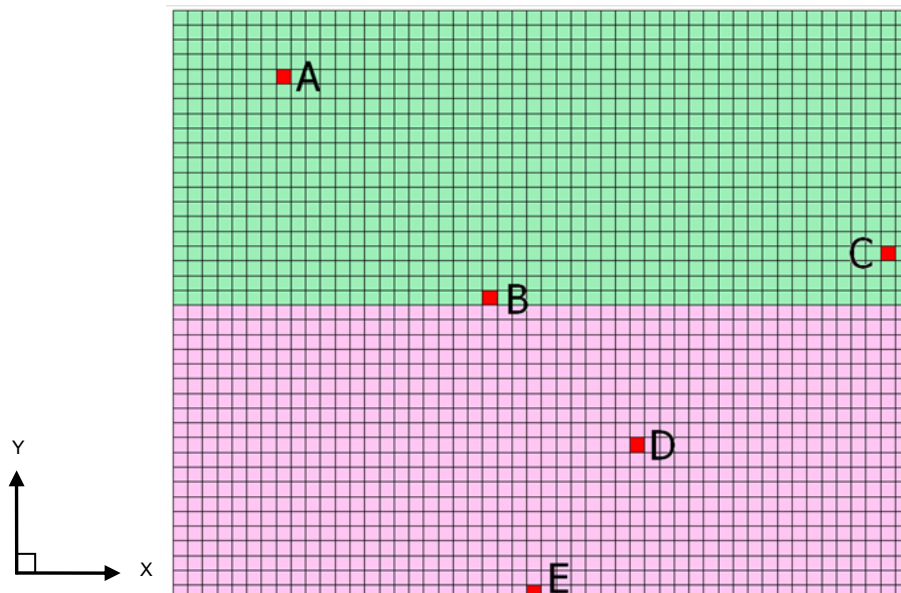
Figure 4.3-1: Location of the particles/elements in the 2D block problem used to verify the Steinberg-Guinan and MTS constitutive model implementations in MCM.

A good correlation between the graphs was observed and any error present was determined to be due to the difference between the methods and not due to errors within the code. Figure 4.3-2 shows the model at 4.392 μs for both simulations.
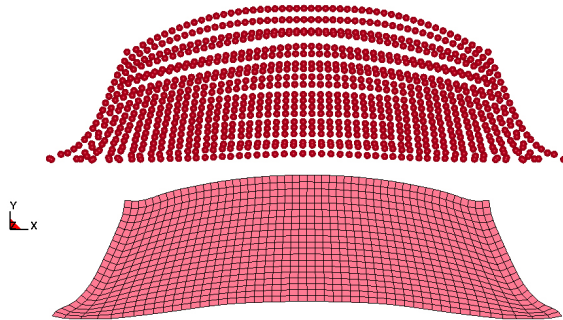
Figure 4.3-2: 2D block impact problem simulated using the Steinberg-Guinan constitutive model at 4.392 μs in MCM and LLNL-DYNA3D respectively.

The position of the boundary particles at the corners of the impact edge is overestimated in the SPH simulation; this is due to kernel deficiency at the boundaries in the SPH method (see Section 3.3.3 for a detailed discussion).

Figure 4.3-3 shows a plot containing x-displacement against time for particles/elements a, b and c for the Steinberg-Guinan model. Only three of the particles under consideration are included for clarity. The SPH response follows a similar trend to the LLNL-DYNA3D equivalent in each case; which further supports the conclusion that the implementation has been conducted correctly. An exact match is not expected since the numerical methods perform different numerical analyses to derive the motion of the particles/elements. Since the differences between the outputs in this case are small, they are assessed to be consistent with differences between the numerical techniques and not due to an error in the implementation.
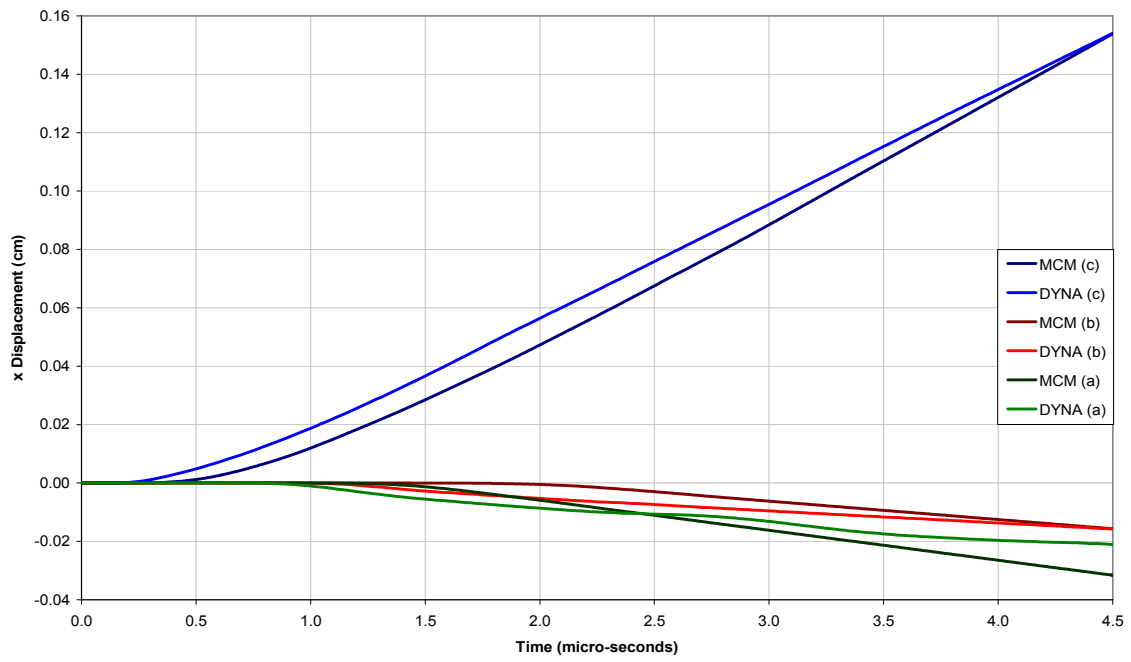
Figure 4.3-3: Steinberg-Guinan Test: x Displacement for Elements/Particles a, b and c.

4.4   The Mechanical Threshold Stress Model

The MTS model is a constitutive model that is applicable to metals at high strain rate (up to 10^5 s^{-1}). This sub-section briefly explains the model and its underlying equations, the implementation that is described here is taken from Lin (2004), which is material model 29 in LLNL-DYNA3D. This model is an implementation of the MTS model described by Maudlin, Davidson *et al.* (1999). The original model was developed by Follansbee and Kocks (1988).

Follansbee and Kocks (1988) describe their original model as follows: *"We consider plastic deformation in a poly-crystalline aggregate to occur by the accumulation and motion of dislocations and the rate controlling deformation mechanism to be the interactions of dislocations with "defects", i.e. grain boundaries, forest dislocations, solute atoms, etc."* The flow stress is specified as a function of a reference stress (or a micro-structure evolution variable), called the Mechanical Threshold Stress, $\hat{\sigma}$, which is the flow stress at 0 K (in the absence of any drag or inertial influences). LLNL-DYNA3D specifies the flow stress as follows:

$$\sigma = \frac{G}{G_0}\left(\hat{\sigma}_a + s_{th}\hat{\sigma} + s_{thi}\hat{\sigma}_i + s_{ths}\hat{\sigma}_s^*\right)$$

Where $\hat{\sigma}_a$, $\hat{\sigma}_i$ and $\hat{\sigma}_s^*$ are dislocation interactions for long-range barriers, interstitial atoms and solute atoms respectively, s_{th}, s_{thi} and s_{ths} are constant-structure deformation variables, G is the shear modulus and G_0 is the initial shear modulus.

A differential hardening law which represents dislocation-dislocation interaction is used to evolve the mechanical threshold stress as follows:

$$\frac{\partial\hat{\sigma}}{\partial\varepsilon_p} = \Theta_0\left[1 - \frac{\hat{\sigma}}{\hat{\sigma}_s}\right]$$

Θ_0 represents hardening due to dislocation generation, and the ratio of stresses represents softening due to dislocation recovery. In this ratio $\widehat{\sigma}_s$ is the MTS at zero strain-hardening, it is called the saturation threshold stress. The relationships for Θ_0 and $\widehat{\sigma}_s$ are as follows:

$$\Theta_0 = a_0 + a_1 \ln\left(\dot{\varepsilon}^p\right) + a_2 \sqrt{\dot{\varepsilon}^p}$$

and

$$\widehat{\sigma}_s = \widehat{\sigma}_{s0} \left(\frac{\dot{\varepsilon}^p}{\dot{\varepsilon}_{s0}}\right)^{\frac{kT}{GAb^3}}$$

Where a_0, a_1, a_2 and A are constants, k is Boltzmann's constant, b is the magnitude of Burger's vector, $\dot{\varepsilon}^p$ is the plastic strain rate, $\dot{\varepsilon}_{s0}$ is a reference strain rate , $\widehat{\sigma}_{s0}$ is the saturation threshold stress at zero Kelvin and T is the absolute temperature.

The shear modulus is a function of temperature and is defined as follows:

$$G = b_0 - \frac{b_1}{e^{b_2/T} - 1}$$

Where b_0 is the shear modulus at zero Kelvin and b_1 and b_2 are constants.

The constant-structure deformation variables; s_{th}, s_{thi} and s_{ths} control deformation by thermal-activation and have Arrhenius rate equations which are functions of absolute temperature, and plastic strain rate:

$$s_{th} = \left[1 - \left(\frac{kT \ln\left(\dot{\varepsilon}_0 / \dot{\varepsilon}^p\right)}{Gb^3 g_0}\right)^{1/q}\right]^{1/p},$$

$$s_{thi} = \left[1 - \left(\frac{kT \ln\left(\dot{\varepsilon}_{0i} / \dot{\varepsilon}^p\right)}{Gb^3 g_{0i}}\right)^{1/q_i}\right]^{1/p_i}$$

and

$$s_{ths} = \left[1 - \left(\frac{kT \ln \left(\dot{\varepsilon}_{0s} / \dot{\varepsilon}^p \right)}{Gb^3 g_{0s}} \right)^{1/q_s} \right]^{1/p_s} .$$

Where $\dot{\varepsilon}_0$, $\dot{\varepsilon}_{0i}$ and $\dot{\varepsilon}_{0s}$ are reference strain rates, g_0, g_{0i} and g_{0s} are normalised activation energies for a dislocation/dislocation, dislocation/interstitial and dislocation/solute interaction respectively, p, p_i, p_s, q, q_i and q_s are material constants. Assuming adiabatic conditions, the temperature of the material is integrated as follows:

$$T = T_r + \frac{1}{\rho C_p} \int_0^{\varepsilon_f} \sigma d\varepsilon ,$$

where T_r is the room temperature or a reference temperature, C_p is the specific heat capacity at constant pressure and ρ is the material density.

For each different material used, a different subset of the above equations may be used.

4.4.1   MTS Model Implementation into MCM

The MTS implementation into MCM is essentially a line for line copy of the LLNL-DYNA3D version of the model allowing for changes due to a significantly different database structure in each code. The implementation was verified using the same simple 2D block impact problem described in Section 4.3.1. The same selection of particles/elements within the model was used to output time-history data and values for y-velocity, effective plastic strain, pressure and x-displacement were plotted (See Figure 4.3-1 in the previous section). A good correlation between the graphs was observed, indicating that the implementation is correct. Figure 4.4-1 shows the model at 4.497 µs for both simulations.
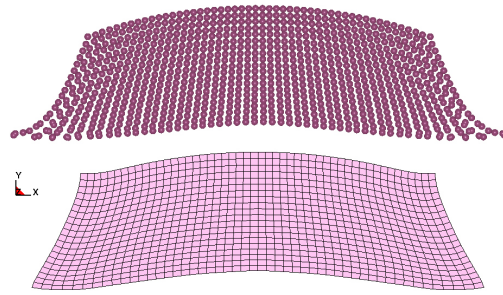
Figure 4.4-1: 2D block impact problem simulated using the MTS constitutive model at 4.497 μs in MCM and LLNL-DYNA3D respectively.

Figure 4.4-2 shows a plot containing x-displacement against time for time history particles/elements a, b and c for the MTS model (see Figure 4.3-1 in Section 4.3.1 for locations of time history particles/elements). Again the position of the boundary particles at the corners of the impact edge is slightly overestimated, which can be attributed to kernel deficiency (see Section 3.3.3). Otherwise the SPH response follows a similar trend to the LLNL-DYNA3D equivalent, which further supports the conclusion that the model has been implemented correctly. As discussed previously, an exact match is not expected since the numerical methods perform different numerical analyses to derive the motion of the particles/elements. Since the differences between the outputs in this case are small; they are assessed to be consistent with differences between the numerical techniques and not due to an error in the implementation.
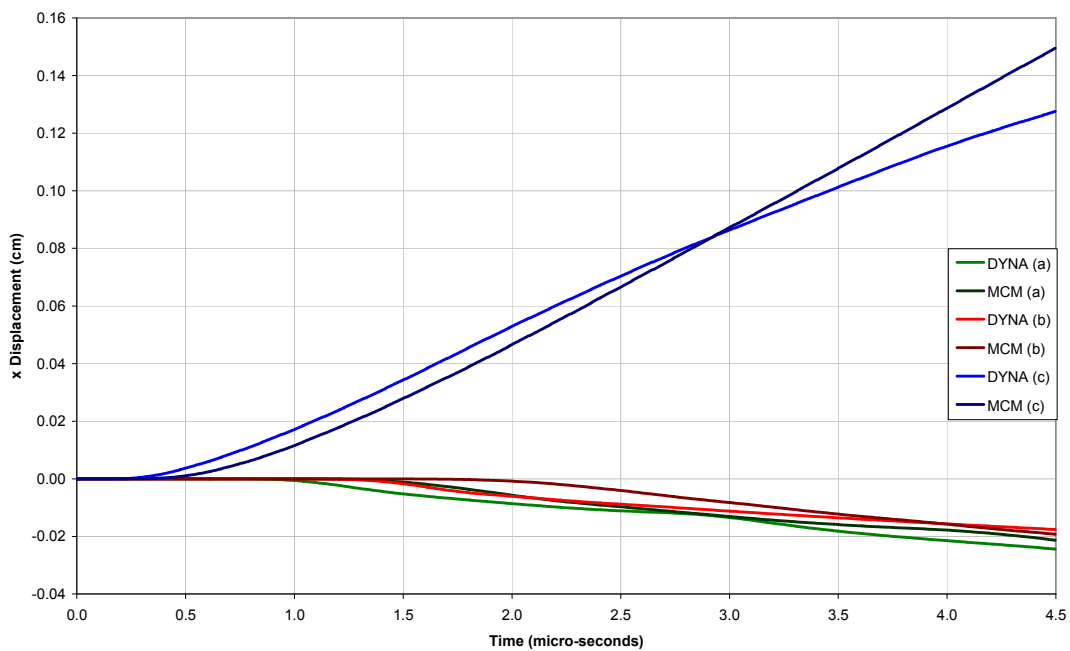
Figure 4.4-2: MTS Test: x Displacement for Elements/Particles a, b and c.

4.5   A Brief Summary of the MTS Model Selection

At the start of this research project, only the Johnson-Cook model was available in MCM. Implementing all three of the other models was not possible within the scope of this research. Therefore it was decided that two of the other models highlighted at the start of this chapter would be implemented for possible use within the research. The Steinberg-Guinan model was originally suggested by the AWE since it is the constitutive model that they most commonly use; it was decided to implement the model to allow for better comparison with current AWE codes.

This left only the MTS and PTW models; the absence of any hydrocode with both models available meant that a research-tailored comparison could not be performed. However, a comprehensive and detailed evaluation of five constitutive models has been conducted by Banerjee (2005a), in which the two models in question are included All information used to make the decision between the models was taken from this reference. For a comprehensive explanation of the results included herein please refer to the original paper.

Banerjee (2005a) conducted two experiments to evaluate each constitutive model. These were one-dimensional tension and compression tests, and a comparison of the deformed profiles of Taylor Cylinders from experiments with profiles that were obtained from running simulations using the same parameters on the constitutive models (for more information on the methods used to conduct these tests, refer to Banerjee (2005b).

Figure 4.5-1 shows the data Banerjee collected for the one-dimensional tension and compression tests for the three candidate constitutive models. The flow stresses predicted by the three models (solid lines) are plotted with those obtained from experimental data (symbol plots), for varying strain rates and varying temperature.

(a) Various strain-rates and temperatures.

(b) Various temperatures at 4000/s strain-rate.
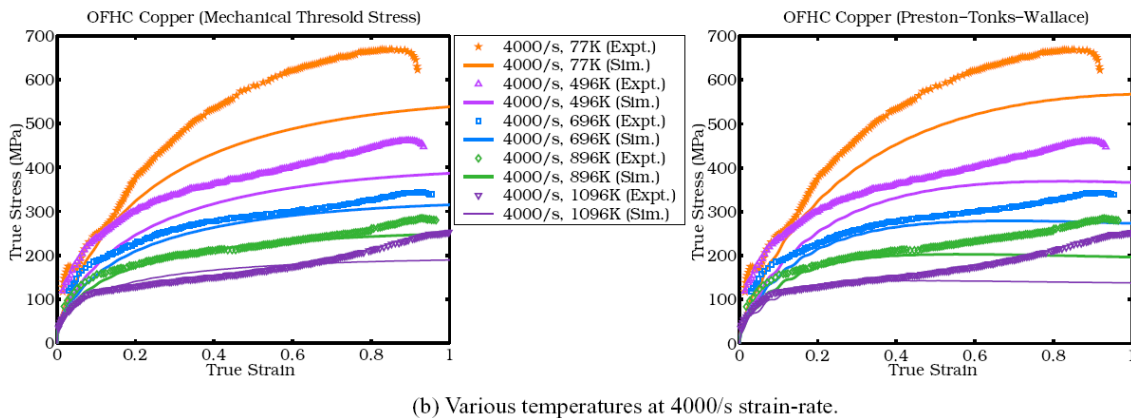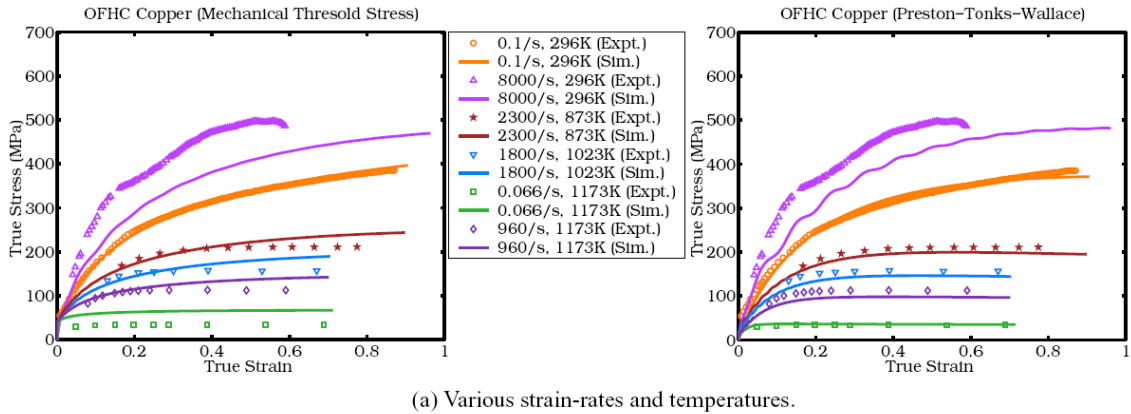
Figure 4.5-1: Predicted values of yield stress from the MTS and PTW constitutive models (Banerjee 2005a).

Figure 4.5-2 shows the Taylor cylinder profiles for the 3 candidate models displayed against the experimental data for one of the test cases explored by Banerjee.

(a) Mechanical Threshold Stress.   (b) Preston-Tonks-Wallace.
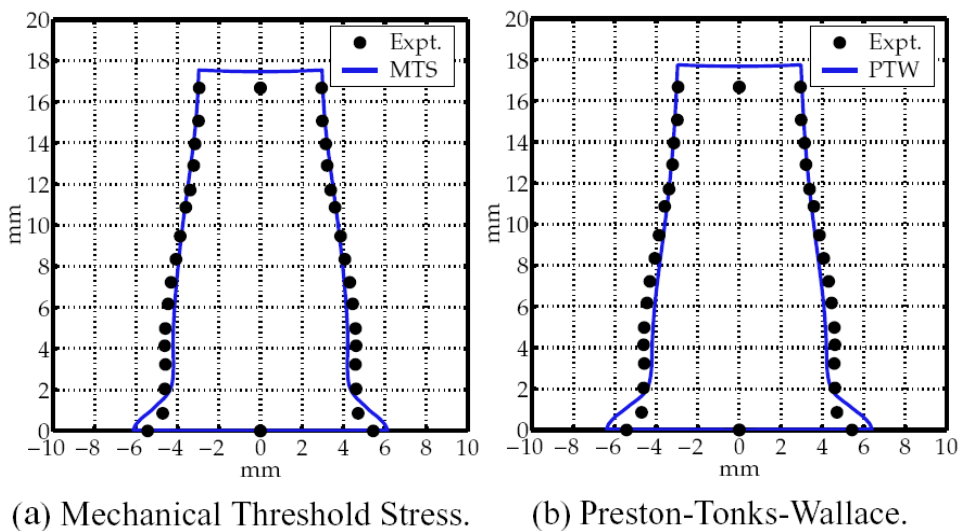
Figure 4.5-2: Computed versus experimental profiles for Taylor test Cu-2 (Banerjee 2005a).

At the end of his report Banerjee concludes that based on his findings: *"no model can accurately predict the yield stress for the full range of conditions considered,"* He also suggests that the model should be chosen based on which one fits the specific requirements for the particular application.

Therefore in order to draw some conclusions to which constitutive model is best suited to this project, it was first necessary to define some requirements that a conclusion can be based upon:

- The fracture model that will be developed will be intended for use at high strain rates,
- It is assumed that the onset of damage will be due to explosive loading and therefore the model will be required to perform well in tension,
- and also at high temperature.

With these three requirements it was possible to filter the data that were collected by Banerjee from his experiments. Table 4.5-1 displays these results (the 'All Tests' error is calculated as the mean overall error of the three requirements):

Condition	Average MA Error (%)	
	MTS	PTW
All Tests	19	17
Tension Tests	14	18
High Strain Rate ($\geq 100 \text{ s}^{-1}$)	15	18
High Temperature ($\geq 800 \text{ K}$)	27	16

Table 4.5-1: Comparison of average "maximum" absolute (MA) errors in yield stresses predicted by the two constitutive models for project specific requirements

Using the data in Table 4.5-1 and coupling it with the fact that the PTW model performed best at the highest strain rate modelled in the Taylor test simulations, it was concluded that overall the PTW model is best suited to the type of modelling required by this research project.

However, since the PTW model offers only marginally better performance for the set of requirements defined above, it was decided that an implementation of the PTW model, (which would have to be generated from scratch along with derivation of any material constants, since there is no implementation of the PTW model available) would be too costly in terms of the time that was available for this research project. An implementation of the MTS model was available in LLNL-DYNA3D for which the code is available; this could be simply modified and copied into MCM, thus saving valuable project time. Therefore, it was decided that the MTS model would be written into MCM.

4.6   Development of Learning through Constitutive Model Implementation

Prior to the start of this research the author had no experience in the FORTRAN programming language and only a limited understanding of finite element modelling through use of the LS-DYNA commercial software package. The implementations of the constitutive models outlined in the previous sub-sections also served as a medium through which grounding in both constitutive modelling and, more broadly, (meshless) hydrocode modelling could be achieved.

The process of implementation involved understanding of the various subroutines required by a constitutive model in both LLNL-DYNA3D and later MCM. This facilitated in gaining a top-level understanding of the discretisation process and the overall structure of a hydrocode model. The experience that was gained from this exercise aided in the more in-depth modifications to the MCM code that were required to implement a successful explosion model (detailed in the next sub-section).

4.7 Explosion Modelling in MCM

An effective method for simulating explosions was required for this research. The simulations that were conducted with the intention of validating the developed fracture model contain explosive materials which detonate and transfer energy to encasing metals (see Chapter 6 for more information). In order to accurately depict the explosive detonation products present in these simulations, a detonation model and equation of state designed for this purpose was implemented into the SPH code.

One commonly used approach, which is also available in the LLNL-DYNA3D package, is the Prescribed Burn Detonation Model with detonation points (material model 8) coupled with the J.W.L. Equation of State (EOS form 2). The Prescribed Burn Detonation Model has the advantage of a simple design where others are inherently more complicated; however due to its simplicity the model is not capable of capturing the correct detonation sequence of complex geometries (e.g. corners) and cannot accurately predict if an explosive will detonate following an impact or other loading (shock to detonation). An equation of state that attempts to capture these requirements is also available in LLNL-DYNA3D as EOS form 7 (Lin 2004). However the explosive modelling that has been conducted for this research uses cylindrical or spherical geometries and is detonated at the start of all the simulations; the Prescribed Burn Detonation Model is thus entirely adequate for this work.

The above approach was therefore implemented into MCM; the model, implementation and verification process are explained below.

4.7.1 The Prescribed Burn Detonation Model

This model is based on work described in Giroux (1971), and is used to model the burning of explosives. The detonation velocity, D is the velocity of a detonation or burn front. The Chapman-Jouguet pressure, P_{CJ} is the maximum pressure realizable in a constant volume adiabatic burn.

During MCM initialization, the lighting time of each particle is computed using a simple programmed burn algorithm. The lighting time, t_L for a particle is computed based on the distance from the particle to the nearest detonation point, l_d, the detonation velocity, and the lighting time of that detonation point, t_{det}, using

$$t_L = t_{det} + \frac{l_d}{D}.$$

Burn fractions are computed to control the release of chemical energy for simulating high explosive detonations. For this work the 'programmed burn' version of the model is used since a detonation time is defined for each particle, although a 'beta burn' version exists that is useful to detect initiation due to shock compression of high explosives (HE). The burn fraction, F is computed from

$$F = \max(F_1, F_2),$$

where

$$F_1 = \frac{(t - t_L)D}{1.5h}$$

and

$$F_2 = \beta(1-V), \text{ where } \beta = \frac{\rho_0 D^2}{P_{CJ}}.$$

h is a characteristic dimension of the particle under consideration, or the SPH smoothing length (in the case of MCM), V is the current relative volume and ρ_0 is the initial density. Before the lighting time has been reached, i.e. $t < t_L$, $F_1 = 0$ and $F = F_2$, which will be equal to zero for simulations which do not have shock loading since the relative volume will remain at unity. Therefore the burn fraction, F is not computed (initialised to zero) until the lighting time has been reached or the relative volume decreases significantly.

If the above equations produce a burn fraction that is greater than one, then it is reset to one. The burn front propagates by multiplying the pressure computed from an equation of state by the current burn fraction,

$$p = F p_{EOS}(V, E),$$

where p_{EOS} is the pressure computed from the EOS at the current relative volume, V and energy, E. High Explosives (HE) typically have large initial internal energies, E_0, which yield large pressures as $F \longrightarrow 1$.

The above explanation of the prescribed burn model and detonation points is taken from Lin (2004).

4.7.2 The JWL Equation of State

The Jones Wilkins Lee (JWL) equation-of-state has been used to describe accurately the pressure-volume-energy behavior of the detonation products of explosives in applications involving metal acceleration. All values are valid only for large charges. The equation for pressure, p is:

$$p = A\left(1 - \frac{\omega}{R_1 V}\right) e^{-R_1 V} + B\left(1 - \frac{\omega}{R_2 V}\right) e^{-R_2 V} + \frac{\omega E}{V}.$$

The pressure as a function of volume at constant entropy, p_s (i.e. the isentrope), is:

$$p_s = A e^{-R_1 V} + B e^{-R_2 V} + C V^{-(\omega+1)},$$

where A, B, and C are linear coefficients in Mbar (GPa), R_1, R_2, and ω are nonlinear coefficients, V is the volume of detonation products per volume of undetonated HE (relative volume) and E is the detonation energy per unit volume in $(\text{Mbar cm}^3)/\text{cm}^3$ [$(\text{GPa m}^3)/\text{m}^3$].

The above explanation of the JWL EOS is taken from Dobratz and Crawford (1985).

4.7.3 Verification of the Explosion Model

Once all of the relevant explosion model routines had been written into MCM the implementation had to be verified. The verification model was a simple cube made up of 1x1x1 cm particles/elements of side 22cm (giving a total of 10,648 particles) with a detonation point at the centre. The material was chosen to be TNT with the following input parameters (Table 4.7-1) taken from Dobratz and Crawford (1985).

Explosive	Density, ρ_0 (g / cm^3)	C-J Pressure, P_{CJ} (Mbar)	Detonation Velocity, D	Initial Energy, E_0 (Mbar)	A (Mbar)	B (Mbar)	R_1	R_2	ω
TNT	1.630	0.210	0.693	0.07	3.712	0.03231	4.15	0.95	0.3

Table 4.7-1: Material parameters for TNT (Dobratz and Crawford 1985).

The model was run in LLNL-DYNA3D and MCM. The lighting times of each particle/element were recorded along with the pressure for seven different particles/elements at each time-step. Figure 4.7-1 shows the simulation at times 0.0, 26.8 and 55.6 µs.
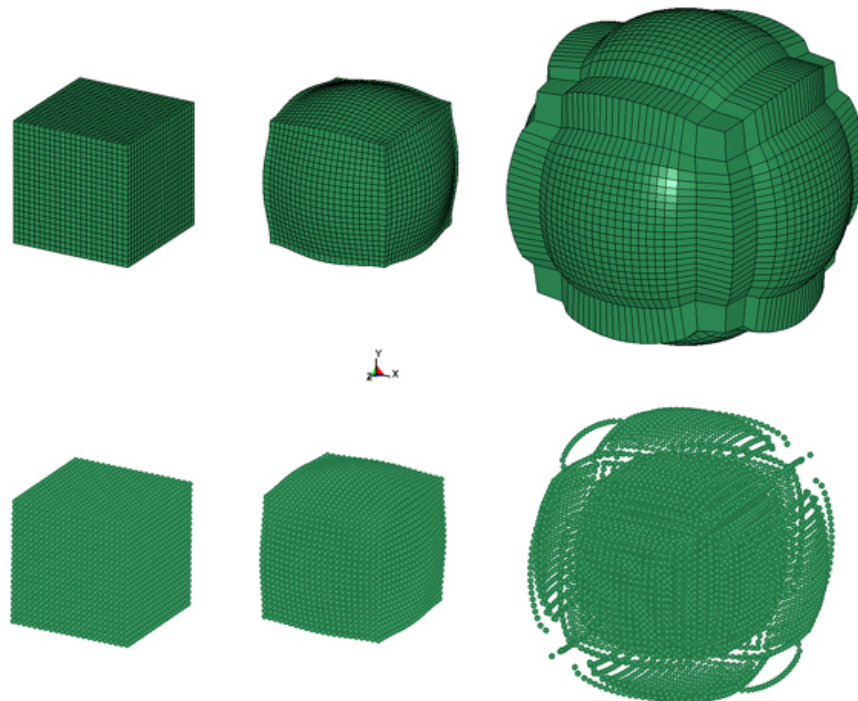
Figure 4.7-1: Explosion model validation cases at times 0.0, 26.8 and 55.6 µs (LLNL-DYNA3D and MCM).

The lighting times which were calculated for the elements/particles were plotted on the same axes and were found to be in agreement within 100% (see Annex 1). The pressure responses for seven different elements/particles distributed throughout the model were also plotted on the same axes and compared, (graphs containing the pressure response comparisons are included in Annex 1). Figure 4.3.2 shows the location of all the particles/elements under consideration:
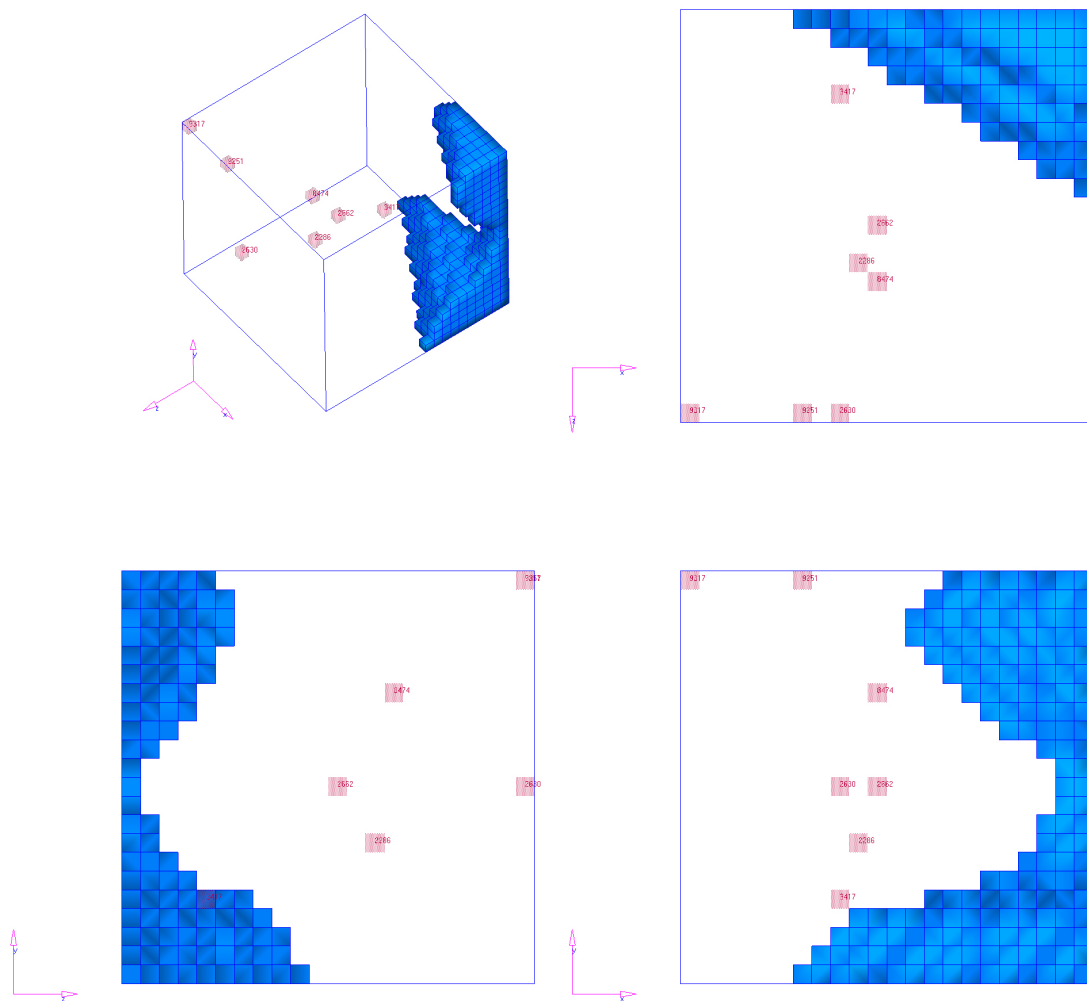
NOTE: Particles/elements in blue illustrate the make-up of the complete model and are included to aid in the understanding of which faces of the 3D diagram are depicted via the three side projections.

Figure 4.3.2: Location of seven particles/elements selected to plot out pressure/time curves for validation of the explosion model in MCM.

Particle/Element 2662 is in the centre of the model (closest to the detonation point); particles/elements 8474, 2286 and 3417 are randomly selected from inside the model,

both away from the boundaries and the centre of the model. Particle/element 2860 is on one face of the model (i.e. it lies on one boundary), particle/element 9251 is on an edge of the model (i.e. it lies on two boundaries), and particle/element 9317 is on the corner of the model (i.e. it lies on three boundaries).

The correlations between MCM and LLNL-DYNA3D for the particles/elements inside the model (8474, 2286 and 3417) are very good and any error is considered to be due to the differences in the discretisation methods; rather than an error in the code for the explosion model.

The correlations for the boundary particles/elements (2860, 9251 and 9317) are progressively worse as the number of boundaries a particle/element lies on increases (see Figures AN1-5 to AN1-7). This is expected because of the kernel deficiency observed at the boundaries in the SPH method (see Section 3.3.3 for more information).

The only correlation which has some unexpected error is the particle/element that lies in the centre of the model nearest to the detonation point (2662). It is observed that whilst the pressure response of in MCM follows a similar path to that in LLNL-DYNA3D, it is greatly exaggerated between about 10 and 40 µs. There is no explanation for this anomaly but since the rest of the particles/elements under consideration have given the correct pressure response; it is unlikely to be due to an error in the code and therefore has also been accepted as a difference between the discretisation methods.

4.8   The Gruneisen Equation of State

At various locations in this thesis references are made to the Gruneisen equation of state, which has been utilised for several of the simulations performed as a part of this research. The Gruneisen equation of state is available in the Cranfield MCM code and is a direct copy of the LLNL-DYNA3D implementation (EOS form 4); this section identifies the formulae associated with the equation of state and is taken from the LLNL-DYNA 3D user manual (Lin 2004).

The Gruneisen equation of state provides two equations for pressure: for materials which are under compression $(\mu > 0)$, pressure is defined as:

$$p = \frac{\rho_0 C^2 \mu \left[1 + \left(1 - \frac{\gamma_0}{2} \right) \mu - \frac{a}{2} \mu^2 \right]}{\left[1 - (S_1 - 1)\mu - S_2 \frac{\mu^2}{\mu + 1} - S_3 \frac{\mu^3}{(\mu + 1)^2} \right]^2} + (\gamma_0 + a\mu) E \,,$$

and for materials which are under tension $(\mu < 0)$, pressure is defined as:

$$p = \rho_0 C^2 \mu + (\gamma_0 + a\mu) E \,,$$

where C is the y-axis intercept of a curve displaying shock velocity vs. particle velocity $(v_s - v_p)$ for the material, S_1, S_2, and S_3 are the coefficients of the slope of the $v_s - v_p$ curve, γ_0 is a material constant known as the Gruneisen gamma, and a is the first order volume correction to γ_0. μ is the excess compression and is defined by

$$\mu \equiv \frac{\rho}{\rho_0} - 1 \,,$$

where ρ is the current density and ρ_0 is the initial density.

This page is intentionally blank

5   An Investigation into SPH Stabilisation Methods

In Chapter 3 the presence of a tensile instability was identified in the Eulerian SPH formulation. To date there have been several attempts to produce a version of the SPH method that is not subject to this numerical problem. One successful example of this is the Total Lagrangian (TL) SPH formulation that was implemented by Reveles (2007) (see Section 3.4 for an explanation of this method). Initially it was thought that a corrected Eulerian SPH formulation would be more beneficial to this research. Such a formulation would simplify the approach to dealing with fracture by eliminating the need for the more complicated TL formulation and would allow for the validation simulations (see Chapter 6) to be modelled successfully[2]. Fundamentally a corrected Eulerian SPH code would ensure that any fracture present in a simulation is only due to a physical fracture model, thus maintaining mathematical rigour in the fracture model design. It was also initially thought that the outcome of the literature survey (see Section 2.2) would be to adopt an individual particle-based description of the damage which could utilise either of the available SPH techniques (as opposed to the inter-particle approach that was eventually decided and is only possible with TL SPH). Investigation into methods for stabilising the Eulerian SPH formulation were therefore of particular interest and was conducted along side the literature review.

Two methods were examined; the first was proposed by Monaghan (2000) and makes use of a repulsive force [MRF] in the SPH momentum equations that acts to reduce the clumping of particles due to the tensile instability. The second aims to reduce the effects of the instability by applying a partition of unity to two calculations of an SPH approximation; the first calculation is made with a localised domain and the second provides an influence from a non-local (or extended) domain. The implementations of the two approaches are described in the following sub-sections along with the relevant verification and validation in each case. The MRF implementation is discussed in Section 5.1 and the Partition of Unity approach is discussed in Section 5.2.

---

[2] TL SPH cannot deal with the exceptionally large deformations associated with the explosive materials used in these simulations.

5.1 SPH with a Repulsive Force Implementation

The clumping of SPH particles due to the tension instability (explained in Section 3.3.1) is unphysical because in a real solid it would be prevented by repulsive forces between the atoms. If the solid is stretched; attractive forces between the atoms resist the stretching and if the solid is compressed; repulsive forces between the atoms resist the compression. Monaghan (2000) proposes the addition of a small repulsive term between the SPH particles to completely remove the tension instability, whist retaining all of the functionality of the SPH method. He also states that the change to a standard SPH code is minor, unlike the Total Lagrangian Formulation outlined in Section 3.4. Therefore the Monaghan Repulsive Force (MRF) was implemented into MCM and a baseline stability analysis was performed to assess the validity of the approach.

5.1.1 Governing Equations

Monaghan (2000) expresses the SPH momentum equation (acceleration of a particle i) as follows:

$$\overline{a}_i = \frac{d\overline{v}_i}{dt} = -\sum_j m_j \left[\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + \Pi_{ij} \right] \nabla_i W_{ij} ,$$

where the summation is over all the neighbours (j) that lie in the domain of influence (neighbourhood) for particle i (normally only near neighbours have a significant contribution because the kernel, W has a finite range). P is pressure, ρ is density and m is mass. Π_{ij} provides artificial bond viscosity (see Section 5.1.3.5 for further information). W_{ij} is the SPH kernel function (see Section 3.1.1.3 for more information).

Since the MRF is repulsive it must increase as the distance between the SPH particles decreases. To achieve this Monaghan chooses a function to scale the repulsive force in terms of the kernel as follows:

$$f_{ij} = \frac{W(r_{ij})}{W(\Delta p)}$$

Where Δp denotes the average particle spacing in the neighbourhood of particle i [3]. Monaghan suggests a typical relationship between Δp and the SPH smoothing length, h is $h = 1.3\Delta p$.

The SPH momentum equation is then rewritten as follows:

$$\overline{a}_i = \frac{d\overline{v}_i}{dt} = -\sum_j m_j \left[\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + Rf_{ij}^n + \Pi_{ij} \right] \nabla_i W_{ij},$$

where $n > 0$ ($n = 4$ for fluid dynamical simulations) and R depends on the pressure and density. The repulsive term can therefore be considered to be an artificial pressure. The factor R is given as follows:

$$R = R_i + R_j,$$

where R_i is determined by the rule, if $P_i < 0$:

$$R_i = \frac{\varepsilon |P_i|}{\rho_i^2}$$

Otherwise R_i is zero, i.e. the MRF is only active in tension [4]. R_j takes the same form with the indices changed to j. An appropriate value of ε is investigated in detail in the proceeding stability analysis. Monaghan suggests that a typical value is 0.2 and that it depends on the value of h, n and N, the number of spatial dimensions.

---

[3] The actual value of Δp is investigated more closely in the proceeding stability analysis. It has also been taken as the initial inter-particle distance (for a uniform distribution of particles); see Section 5.1.3.1 for more information.

[4] Section 3.3.1 highlighted that the SPH 'tension' instability is also present for simulations experiencing compression, but only for cases outside the normal operating conditions of the basic SPH method, i.e. for large smoothing lengths.

In MCM the SPH momentum equation is expressed in terms of the full stress tensor as follows (in this case the term **q** provides an artificial viscosity):

$$\bar{a}_i = \frac{d\bar{v}_i}{dt} = \sum_j m_j \left[\frac{(\boldsymbol{\sigma}_i - \mathbf{q}_i)}{\rho_i^2} + \frac{(\boldsymbol{\sigma}_j - \mathbf{q}_j)}{\rho_j^2} - Rf_{ij}^n \right] \nabla_i W_{ij} .$$

The MRF is therefore an artificial stress and takes the following form:

$$Rf_{ij}^n = (R_i + R_j)(f_{ij})^n = \varepsilon \left(\frac{\operatorname{tr}\boldsymbol{\sigma}_i}{\rho_i^2} + \frac{\operatorname{tr}\boldsymbol{\sigma}_j}{\rho_j^2} \right) \left(\frac{W(r_{ij})}{W(\Delta p)} \right)^n .$$

5.1.2   Selection of a Suitable Test Problem

In order to validate and verify the MRF a suitable test problem that displays the SPH 'tension' instability visually and graphically was required. Swegle (1995) provides such a test case which is extremely simple to generate.

The problem may be generated in 1, 2 or 3 dimensions, but the instability is best illustrated with the 2D case. A body is subjected to a uniform initial stress, either compressive or tensile. The particles that lie close to the boundary are fixed in all directions; this prevents wave propagation from relieving the initial stress. All viscosities are switched off in order to separate the properties of the Eulerian SPH method from any effects due to artificial viscosity. A small perturbation velocity (10^{-9} km s^{-1}) is applied to the centre particle.

The material is represented by a fluid constitutive model with a linear polynomial equation of state. For an initial relative volume, V_{rel} of 1.05, Figure 5.1-1 shows the initial state of the problem and Figure 5.1-2 shows the problem after just 12 μs. The effect of the tension instability is clear; the particles have undergone unphysical motion and clumped together forming large voids in the material. When the problem is run in compression no change in the particle positions is observed (the smoothing length for these simulations was taken to be 1.3d , where d is the initial inter-particle spacing).
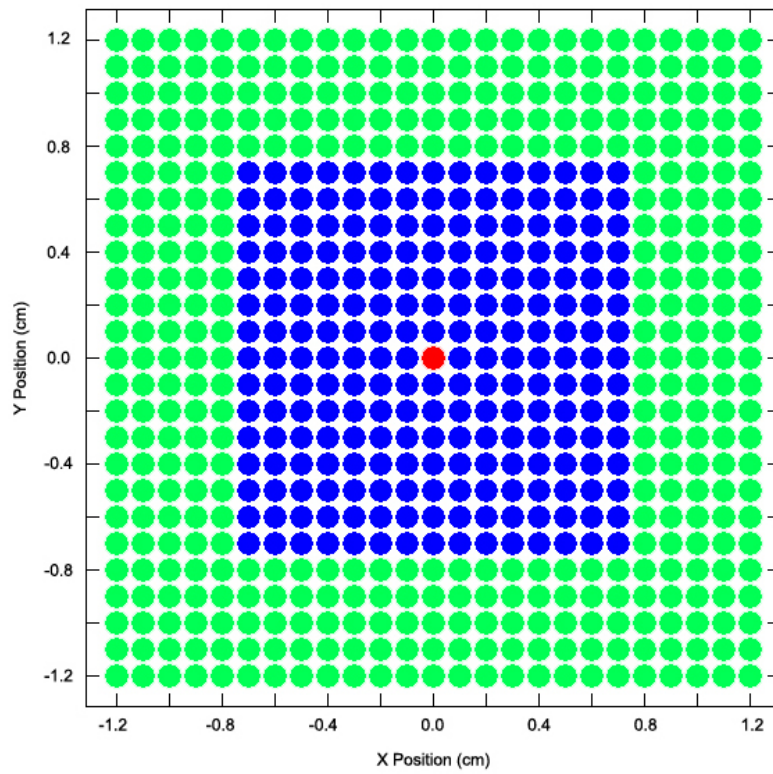
Figure 5.1-1: Initial particle positions in Swegle 2D stability test (green particles are fixed in 2D space and the red particle is given an initial x-velocity of 1x10^{-9} km s^{-1}).
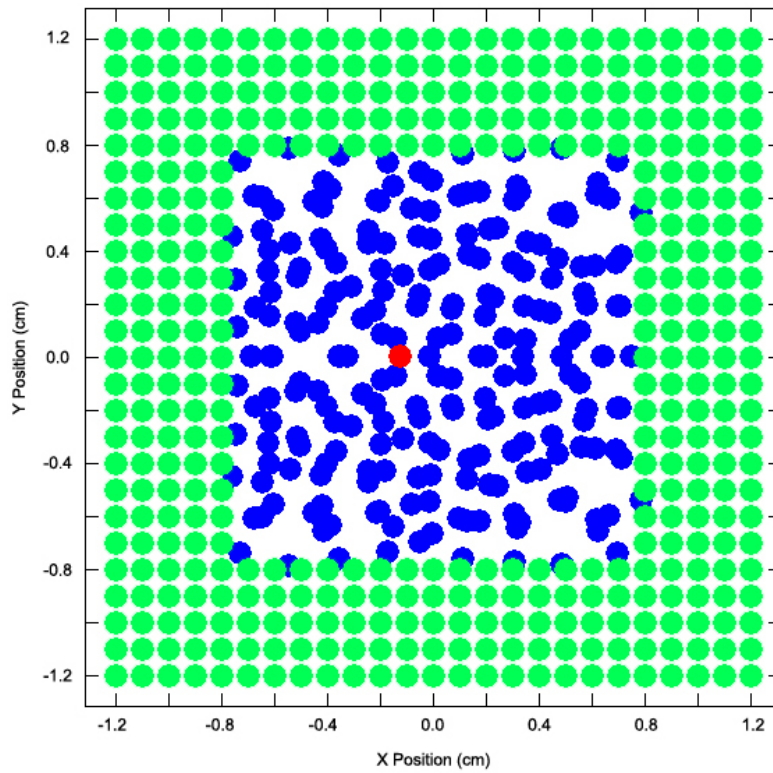
Figure 5.1-2: Particle positions at 12 μs for the 2D Swegle stability test, showing the effect of the tension instability.

A parameter denoted as 'total velocity' is plotted with time to better display the growth of this instability. The 'total velocity,' v_T is calculated from:

$$v_T = \sqrt{\frac{2E_K}{m}} ,$$

where m is the total mass and E_K is the total kinetic energy of the system. Figure 5.1-3 shows plots for varying levels of tension. It should also be noted that for a compressive stress the system is stable ($V_{rel} = 0.999$). A plot displaying the data for a simulation using total Lagrangian SPH method with $V_{rel} = 1.05$ is also included for clarity and is also stable.
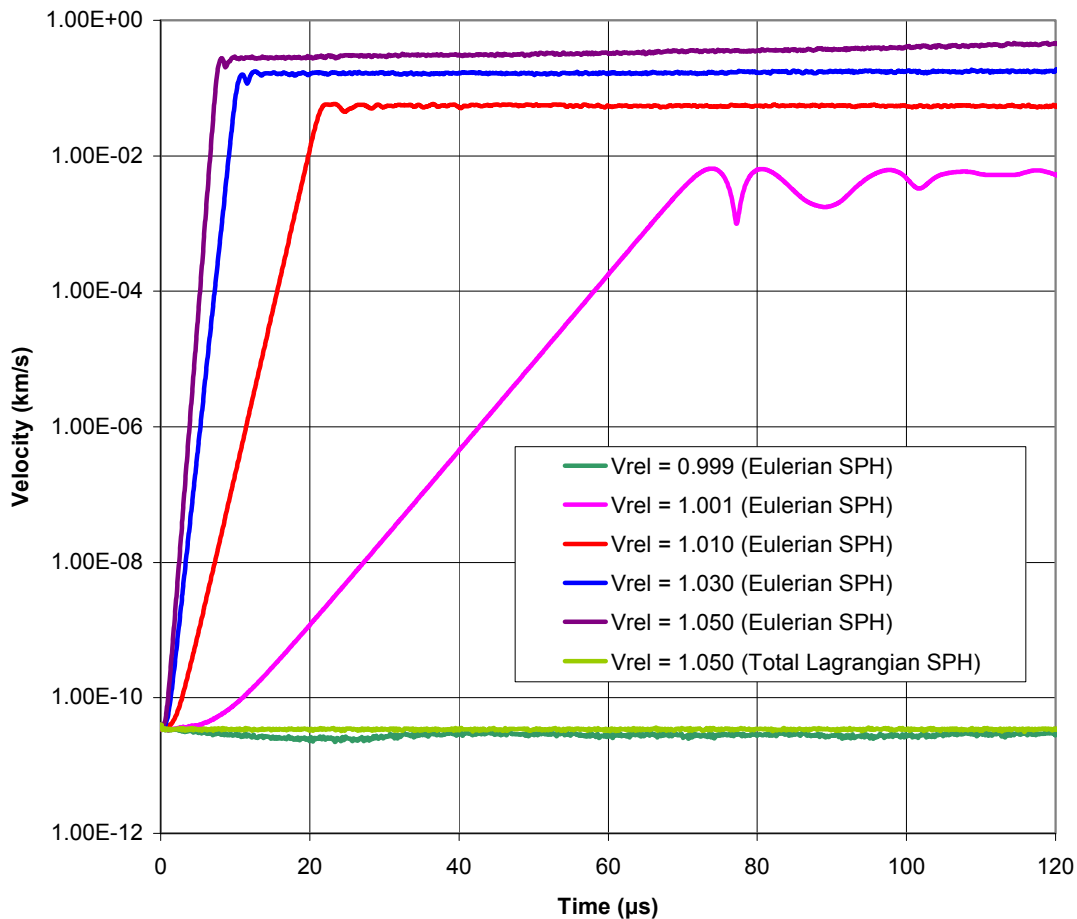
Figure 5.1-3: Perturbation amplitude history for the whole system, i.e. the total velocity of the system as function of time. Generated using an Eulerian SPH formulation for various levels of tension initially present in the model.

5.1.3 Stability Analysis

The MRF as presented in Section 5.1.1 was implemented into MCM and tested using the Swegle 2D problem introduced in Section 5.1.2. The initial tension was arbitrarily chosen to be $V_{rel} = 1.01$ for all simulations and $n = 4$, in accordance with Monaghan (2000).

5.1.3.1 Appropriate Selection of Δp

Monaghan states that *"Δp denotes the average particle spacing in the neighbourhood of particle i."* However he also states that typically $h = 1.3\Delta p$. These two statements give rise to some ambiguity in the method of calculating of Δp.

If Δp is calculated as the average distance from the i particle (particle under consideration) to each of its neighbouring particles as depicted by Δp_1 for 1D in Figure 5.1-4, then the value of Δp increases almost proportionally to the number of neighbours present (i.e. the smoothing length), but the constant multiplier is not equal to 1.3, as for 4 neighbours $h = 0.87\Delta p$ and for 6 neighbours $h = 0.85\Delta p$. This method can be expressed in the following form:

$$\Delta p = \frac{1}{N_{nbr}} \sum_{n=1}^{N_{nbr}} r_{ij_n} \, ,$$

where N_{nbr} is the number of neighbours present in the neighbourhood of particle i.

This suggests that Monaghan is referring to the average spacing between each pair of particles, and not the average distance from a neighbour to the i particle. The Swegle problem uses a uniformly distributed mesh and for a stable code there will be no position changes of the particles; therefore Δp can simply be taken to be equal to the initial particle spacing <u>for this instance only</u> (See Δp_2 for 1D in Figure 5.1-4). However for a real problem this method of calculating Δp will be complicated as it is not trivial to

calculate distances between neighbour particles in the neighbourhood of particle i in the current configuration of the MCM SPH code. Figure 5.1-5 depicts a typical situation that might be encountered and highlights the complexity of the problem. Due to time constraints, such an implementation is outside the scope of this research and is left for further work into better understanding of the MRF.

$$\Delta p_1 = (r_{ij_1} + r_{ij_2} + r_{ij_3} + r_{ij_4})/4$$
$$\Delta p_2 = r_{ij_1} = r_{ij_2} = r_{j_1j_3} = r_{j_2j_4}$$
$$\Delta p_1 \neq \Delta p_2$$
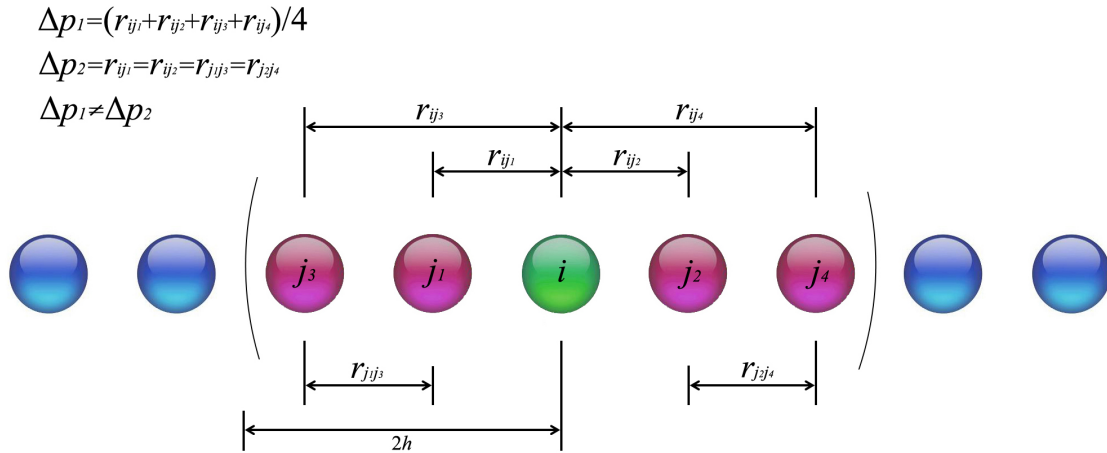
Figure 5.1-4: Two opposing methods of calculating Δp for a 1D uniform arrangement of particles.

$$\Delta p = \frac{\sum r}{N_{pairs}}$$
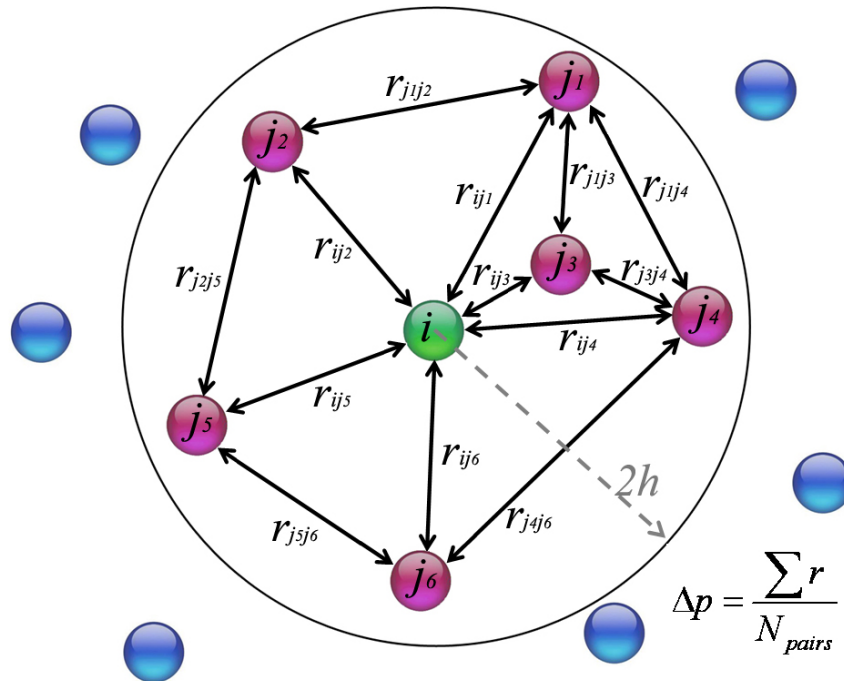
Figure 5.1-5: Diagrammatic representation of the complexity involved in obtaining Δp as the average of each inter-particle distance for a 2D arrangement of particles, if the particle positions in the neighbourhood are not uniformly distributed.

In the following analysis of the MRF, both of the methods for obtaining Δp (as discussed above) are examined to address the significance of this value in obtaining a stable code.

5.1.3.2 Determining a stable value of ε by trial and error

Once the implementation of the MRF was complete, the value of ε was varied in an attempt to obtain a value or set of values that would stabilise the code for the Swegle 2D problem outlined in Section 5.1.2.

It is worth noting that the MRF has been implemented by Larry Libersky of Los Alamos National Laboratory (Libersky 2006). He indicated that he had the method working successfully and provided the following relationship:

$$\varepsilon = 10^{(N-1)} \times \alpha \,,$$

where N is the number of spatial dimensions and $0.2 \leq \alpha \leq 0.4$. Therefore 'typical' values are 0.3, 3.0 and 30.0 in 1, 2 and 3-dimensions respectively. Monaghan suggests that a 'typical' value of ε is 0.2 (Monaghan 2000).

ε was varied between 1 and 10 and the plots of total velocity of the system (a component of the total kinetic energy) with time were compared with the simulation that was run using Eulerian SPH. It was discovered that whilst the MRF reduced the onset of the tension instability, it was not able to completely stabilise the code. For this test the value of Δp was taken as the average inter-particle distance (See the previous sub-section). The most stable value of ε that was obtained was 1.3; any value less or greater produced a more unstable code. Plots for several different values of ε are displayed in
Figure 5.1-6.
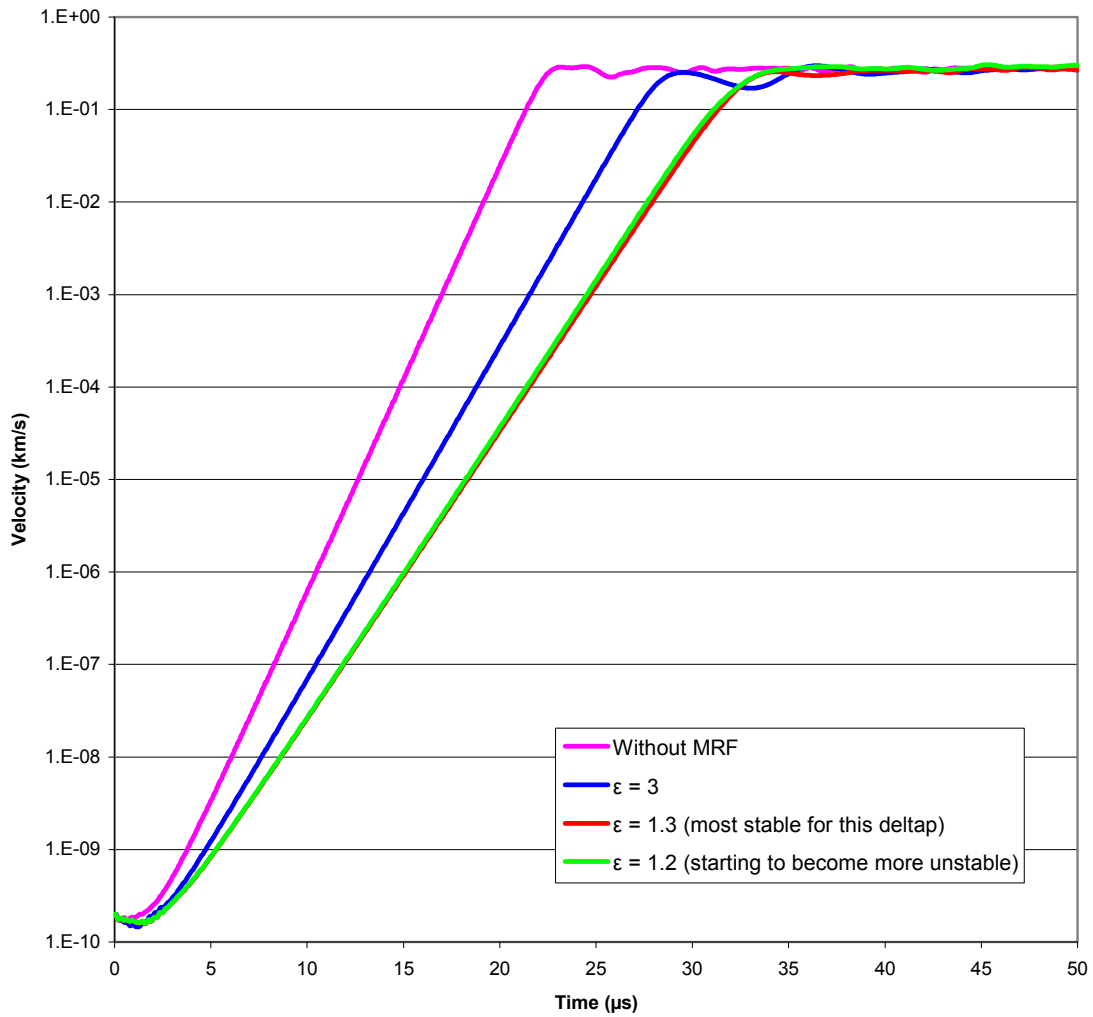
Figure 5.1-6: Varying values of ε for the Swegle 2D problem (Δp is taken to be the average inter-particle spacing).

The above test was repeated with Δp as the initial inter-particle distance but a completely stable code was still unobtainable.

In an attempt to simplify the problem, a 1D version of the Swegle problem was generated and the test was repeated for both methods of calculating Δp. This time the MRF was capable of stabilising the code for ε values of 1.045 and 1.119 for the average Δp calculation method and initial inter-particle distance Δp method respectively. It is also worth noting that the code was stable for any value of ε greater than those stated above until approximately when $\varepsilon = 90$, where the code began to

become unstable again. Plots for several different values of ε are displayed in Figure 5.1-7.
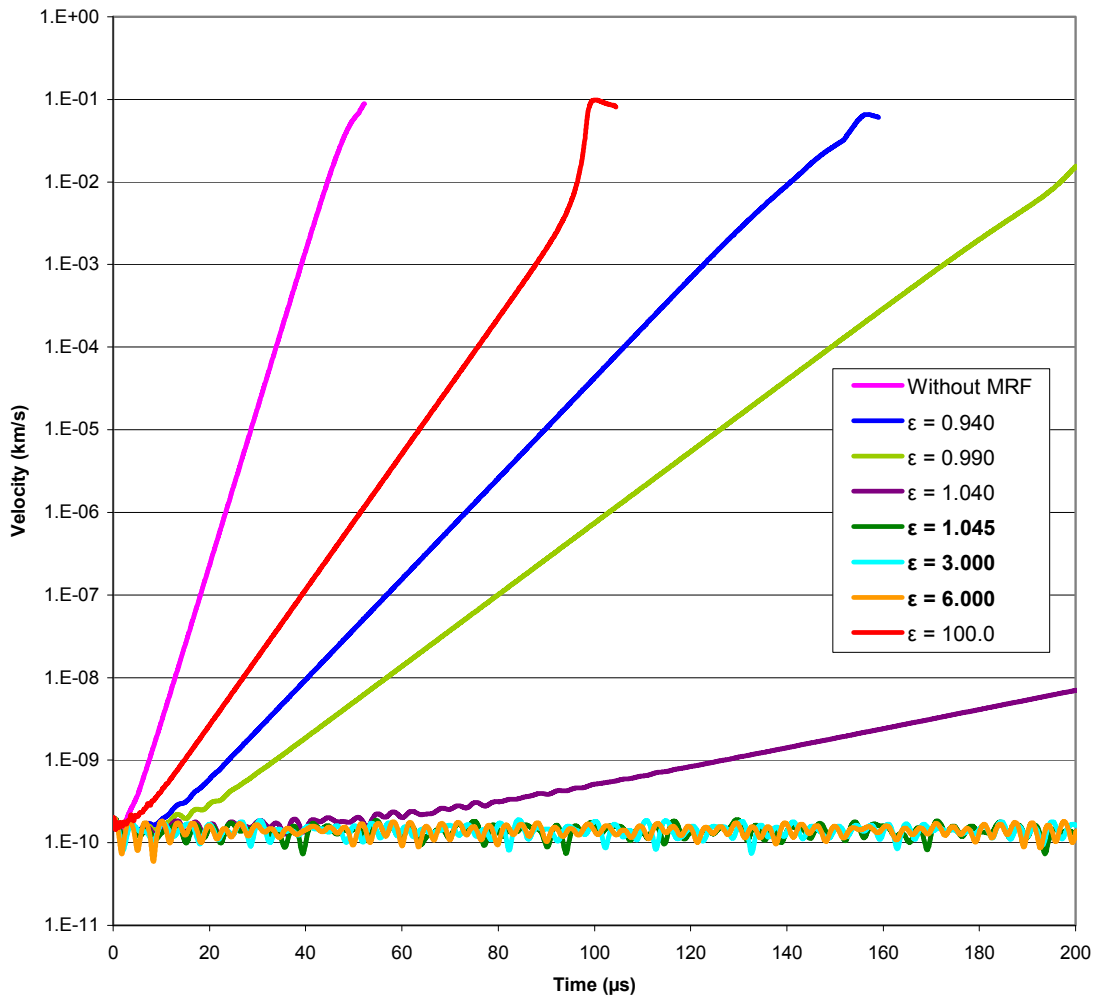
Figure 5.1-7: Varying values of ε for the Swegle 1D problem (Δp is taken to be the average inter-particle spacing).

5.1.3.3   Sensitivity analysis of ε

Having established a stable value of ε for the 1D Swegle problem, a sensitivity analysis was performed in an attempt to provide an insight into how to stabilise the 2D model.

- Primarily the 1D model was rerun with $\varepsilon = 0$; this was to check that the MRF implementation had not introduced coding errors in the standard running of the

SPH method. The results were identical to those obtained from the Eulerian SPH simulation, indicating correct operation of the code.

- Secondly, the time-step size was reduced for a stable value of ε (1.119) in a 1D simulation. For both time-step sizes the results were identical, indicating that the time-step size has no effect on the MRF.

- The effect of Δp on ε was investigated by changing Δp from 0.1 (initial inter-particle distance) to 0.2 and 0.3 for both the stable 1D model ($\varepsilon = 1.119$) and the most stable version of the 2D model ($\varepsilon = 1.300$). It was discovered that as Δp increases, the corresponding value of ε required to stabilise the 1D model decreases slightly. The same effect was observed for the 2D model but still no stable version was obtained.

- A test was performed to highlight a possible difference between axes in the 2D solver. First the particles in the 2D Swegle problem were fixed to only allow movement in the x-direction and then the y-direction. For both simulations $\varepsilon = 0$ and the total velocity was plotted with time. Both plots were identical, indicating that the solver is functioning correctly.

5.1.3.4 Work Done by the Monaghan Repulsive Force

It was thought that MRF could be adding unphysical strength to the material which might explain the reason for not being able to stabilise the SPH code in 2D. In order to investigate this theory the work done by the MRF was calculated as follows:

The acceleration of a particle due only to the MRF may be expressed as

$$a_i = \sum_j m_j R f_{ij}^n \nabla_i W_{ij} \,,$$

from Newton's second law:

$$F_i = m_i a_i \,.$$

The work done by the MRF is then simply the force multiplied by the change in distance:

$$WD_{MRF} = \sum_i F_i \sqrt{\left(x_{(n-1)} - x_n\right)^2 + \left(y_{(n-1)} - y_n\right)^2 + \left(z_{(n-1)} - z_n\right)^2} \; .$$

The work done was plotted with time for a stable and unstable 1D model and it was apparent that the MRF only has a significant effect on the energy of the system when the code is unstable, but for the stable simulation its effect is negligible ($\sim 10^{-17}$ J).

5.1.3.5 Effect of Monaghan's artificial bond viscosity on the repulsive force implementation

In Section 5.1.1, the term Π_{ij} was identified as an artificial bond viscosity (ABV) in the SPH momentum equation (Monaghan, Gingold 1983). Since the only difference between the MRF equations outlined in Monaghan (2000) and the implementation in MCM is the presence of this artificial viscosity, it was assessed that it would be beneficial to partially implement Π_{ij} for the 1D Swegle test problem. If the value of ε required to stabilise the code with the ABV active is radically different from the previous values obtained, then it might explain the reason why the MRF is not stabilising the 2D solver. This would therefore merit further investigation into the MRF coupled with the ABV. The governing equations for the artificial viscosity in 1D are taken from Johnson (1996) as follows:

$$\Pi_{ij} = \left\{ \begin{array}{ll} \left(\dfrac{C_L c_{ij} \left| \mu_{ij} \right| + C_Q \mu_{ij}^2}{\rho_{ij}} \right) & , \quad \dot{u}_{ij}.x_{ij} < 0 \\[4mm] 0 & , \quad \dot{u}_{ij}.x_{ij} > 0 \end{array} \right\},$$

where C_L and C_Q are linear and quadratic coefficients and have the values 1.0 and 2.0 respectively. c_{ij} is the average speed of sound and ρ_{ij} is the average density between each pair of particles. μ_{ij} is calculated via the following formula:

$$\mu_{ij} = \frac{h_{ij}\left(\dot{u}_{ij}x_{ij}\right)}{r_{ij}^2 + \varepsilon_0 h_{ij}^2},$$

where h_{ij} is the average smoothing length, x_{ij} is the x-coordinate difference and $\dot{u}_{ij}$ is the x-velocity difference between each pair of particles. $\varepsilon_0 = 0.01$ and $r_{ij} = \left|x_{ij}\right|$.
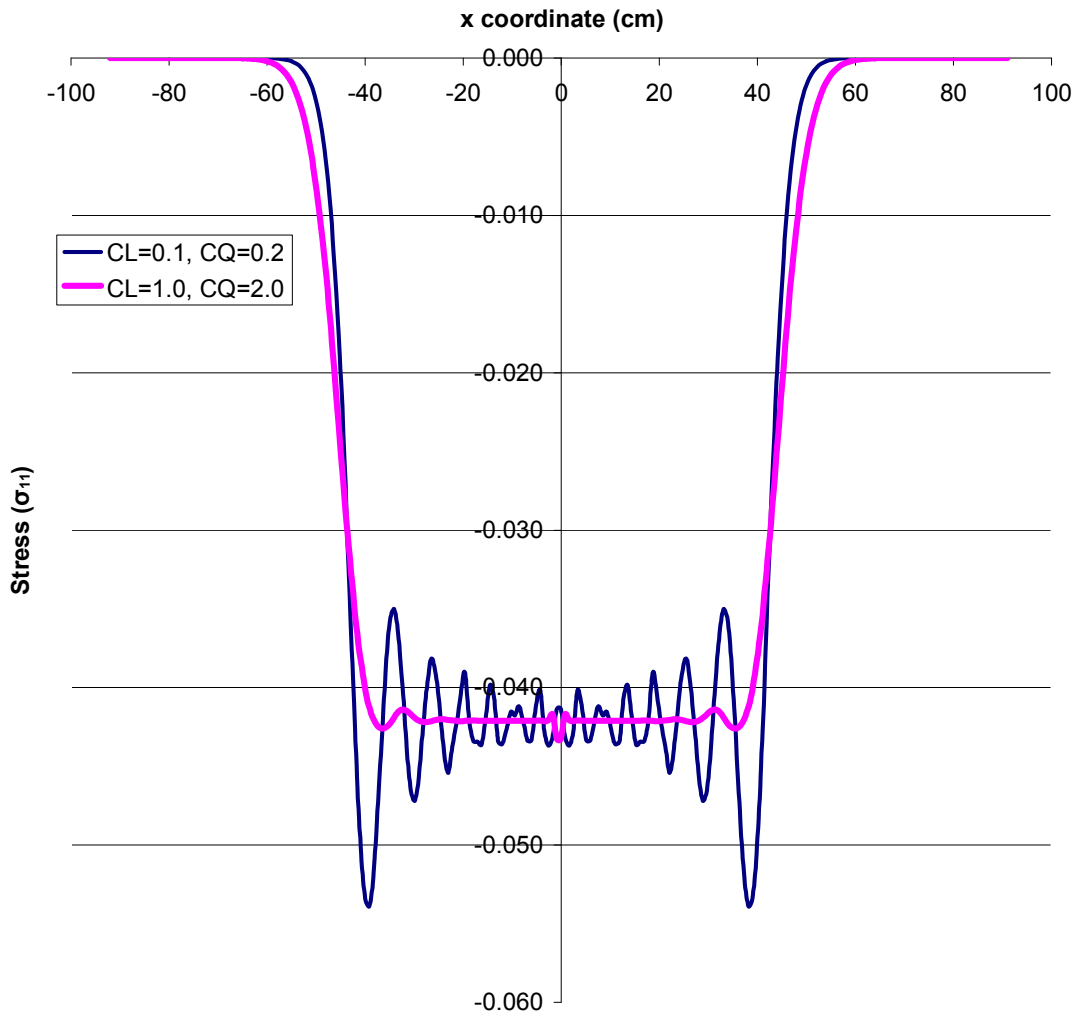
Figure 5.1-8: Plots of xx-Stress with x-coordinate for different intensities of Monaghan's bond viscosity at 400 μs in a 1D rod impact validation problem.

Once it had been hardwired into the 1D solver, a test problem was conceived to validate the implementation. This involved two 1D rods of length 100 cm (100 particles) impacting each other at a speed of 200 ms^{-1}. The xx-stress was plotted against *x* position at 400 μs for values of $C_L = 0.1, C_Q = 0.2$ and $C_L = 1.0, C_Q = 2.0$ respectively. The plots showed a smoothing of the curve for the higher values of the coefficients, thus indicating that the implementation was correct (See Figure 5.1-8).
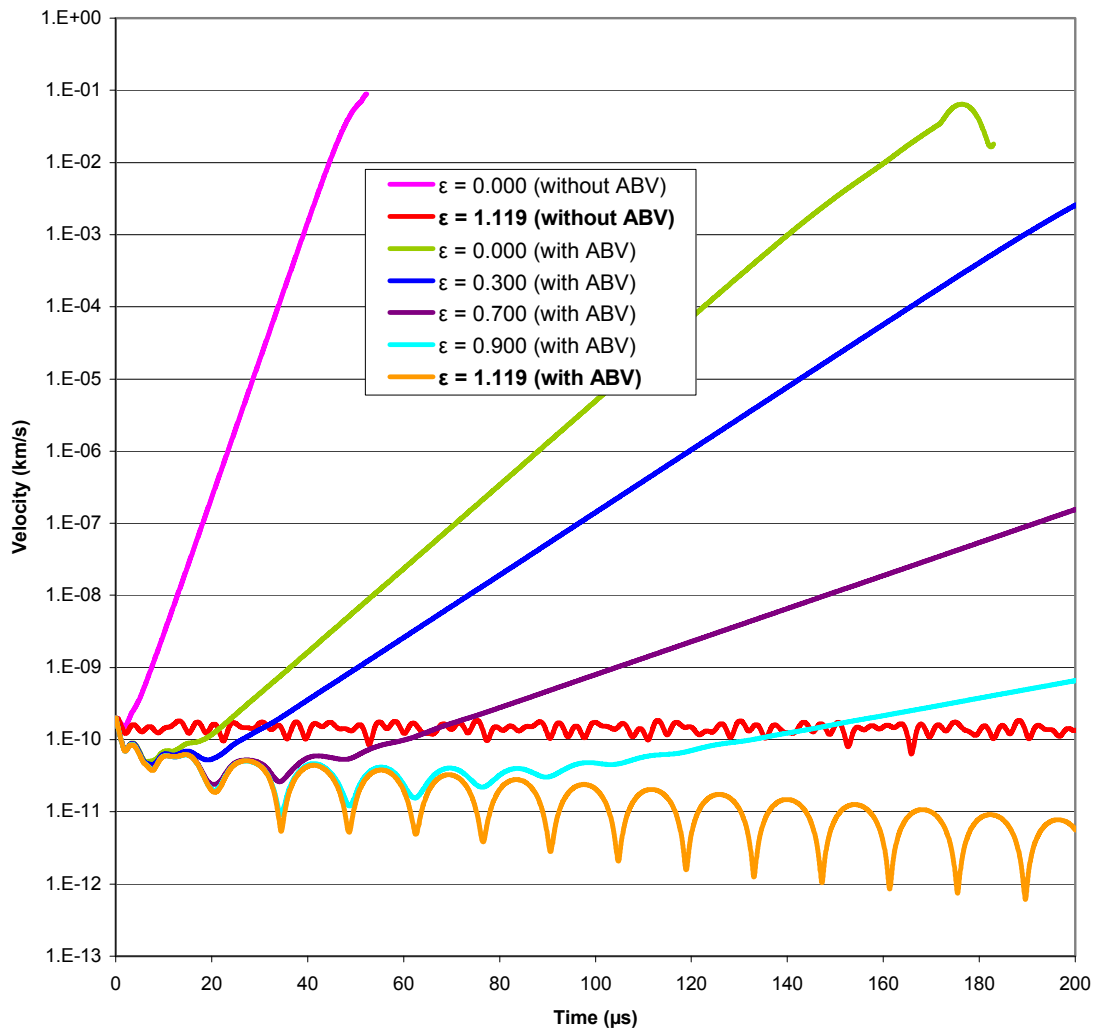
Figure 5.1-9: Varying values of ε for the Swegle 1D problem with Monaghan's artificial bond viscosity present (Δp is taken to be the initial inter-particle spacing).

The 1D Swegle problem was then run with the artificial bond viscosity active and the same minimum value of ε that stabilised the Swegle problem without any artificial viscosity present was required to stabilise the code (See Figure 5.1-9). The artificial

bond viscosity is dissipative as is apparent from Figure 5.1-9, i.e. when the code is stable the total kinetic energy of the system reduces in an oscillatory fashion over time. It is assessed that further investigation into the bond viscosity or its behaviour is not of any further value in connection with the MRF because the purpose of its implementation was only to determine if it had an effect on the stabilising value of ε. Since an effect on the stability is not observed, it can therefore be concluded that the artificial viscosity is not responsible for the problems experienced with the MRF and the 2D-solver.

5.1.3.6   Effect of SPH smoothing length on the value of ε

Since analysis of all the previous parameters had been unsuccessful in identifying the reason for not being able to stabilise the 2D code using the MRF, it was decided that an analysis of the effect of the smoothing length on the MRF was required. Until now the smoothing length had been taken as $h = 1.3\Delta p$ in accordance with Monaghan (2000), where Δp is the initial inter-particle distance.

In order to gauge the effect of varying this ratio, the value of the smoothing length was varied to adjust the number of neighbours that would be present in the neighbourhood of particle i for the 1D Swegle problem. When $h = 1.3\Delta p$, the centre particle has 4 neighbours; h was therefore adjusted to $0.7\Delta p$ and $1.8\Delta p$, to give 2 and 6 neighbours respectively.

For each new value of h, an attempt was made to find the corresponding minimum value of ε required to stabilise the code. For the 2-neighbour problem it was not possible to select any value of ε to stabilise the code. However for the 6-neighbour problem the minimum stable value of ε dropped from 1.119 (for 4 neighbours) to 0.600. This clearly demonstrates that the choice of smoothing length has a significant impact on the stability of the MRF. Figure 5.1-10 shows the results graphically.
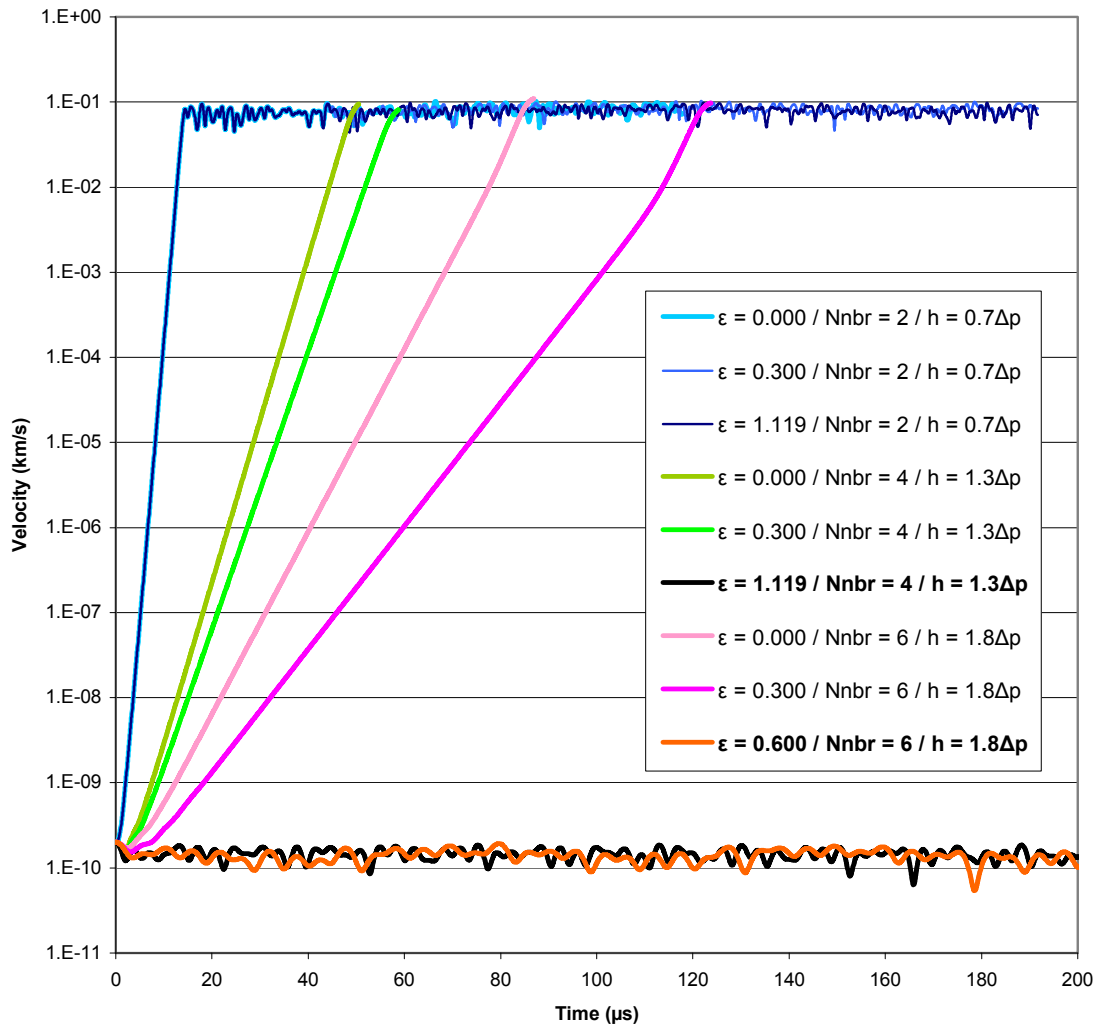
Figure 5.1-10: Varying values of ε for different ratios of h to Δp for the Swegle 1D problem (Δp is taken to be the initial inter-particle spacing).

The same analysis was then repeated for the Swegle 2D problem. In this case when $h = 1.3\Delta p$, the centre particle has 20 neighbours; h was therefore adjusted to $1.52\Delta p$ and $1.8\Delta p$, to give 28 and 36 neighbours respectively. For the 28-neighbour problem it was still not possible to select any value of ε to stabilise the code, however for the 36-neighbour problem stability was achieved with an ε of 0.950. The results are displayed in Figure 5.1-11.

It is worth noting that the tension instability in the Eulerian SPH simulation is significantly less prominent due to the enlarged neighbourhood. It is also worth noting

the code becomes unstable again with a value of ε as low as 10. This is unlike the 1D case where the upper value of ε at which the code is still stable is in excess of 90.

In addition, the use of the enlarged neighbourhood has moved outside the normal operating conditions of the SPH method and as such, the instability is also present when the model is initially in compression (green curve in Figure 5.1-11). In order for the MRF to completely stabilise the code under these conditions, it will need to be able to correct for both tension and compression instabilities with the same value of ε. The MRF was modified to be active in compression but no value of ε could be obtained to stabilise the compression instability.
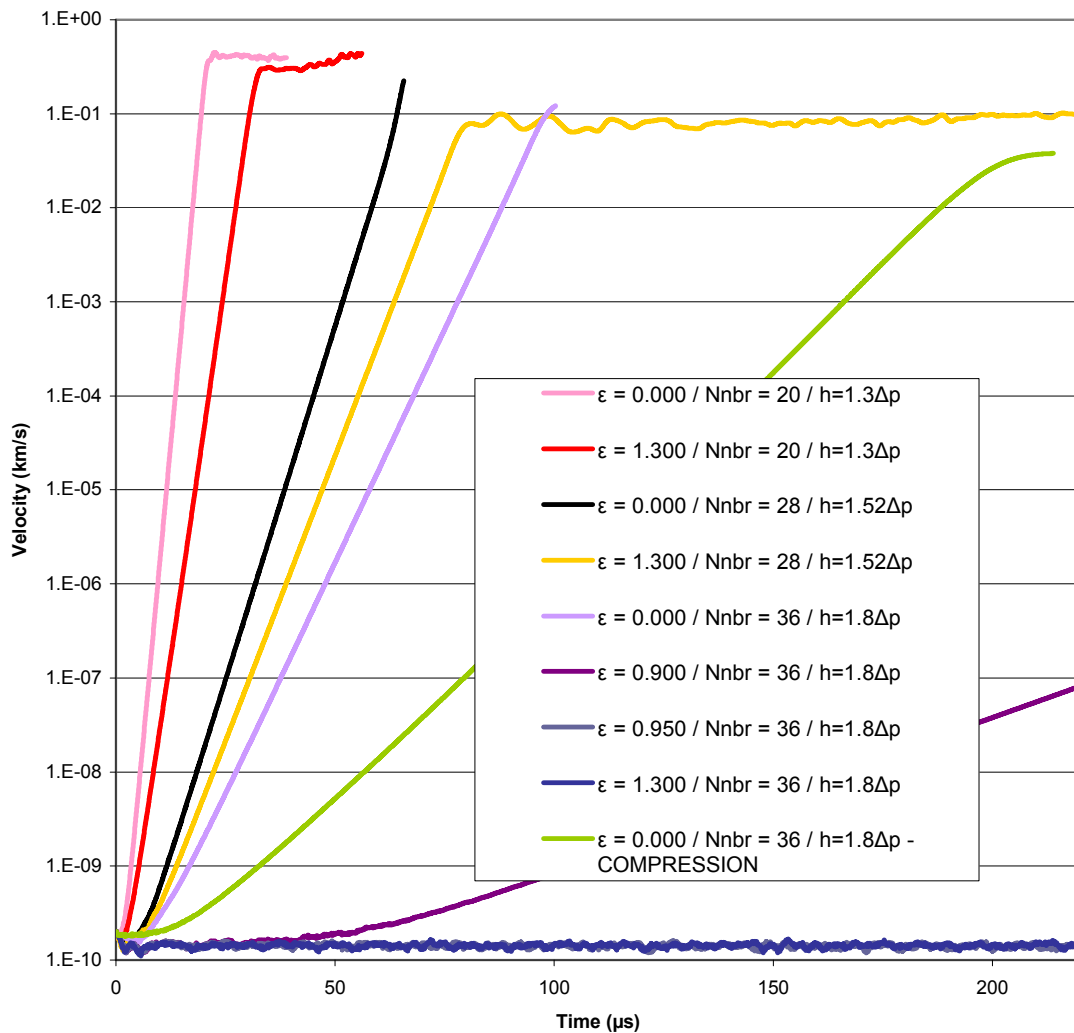
Figure 5.1-11: Varying values of ε for different ratios of h to Δp for the Swegle 2D problem (Δp is taken to be the initial inter-particle spacing).

5.1.3.7   Effect of adjusting the level of pre-tension in the model on the value of ε

Figure 5.1-3 on page 108 shows the growth of the tension instability for several different levels of pre-tension in the Swegle 2D problem for a smoothing length of $h = 1.3\Delta p$ (where Δp is the initial inter-particle distance). It is clear that the instability growth rate and intensity varies with the level of tension present in the model. For the analysis of the MRF outlined in the previous sub-sections, the initial tension was arbitrarily chosen to be $V_{rel} = 1.01$ for all simulations.

In the MRF the parameter R is a function of pressure and the density and should therefore account for the level of tension present in the material at each particle. This suggests that a value of ε which produces a stable code for a model with pre-tension set via $V_{rel} = 1.01$ ($0.95 \leq \varepsilon \leq 10.0$) should also produce a stable code for the same model containing other levels of pre-tension. This is of vital importance, since a real model will contain a range of different tensions and compressions depending on the loading conditions.

To test this theory, the stable 2D Swegle model ($h = 1.8\Delta p$) was adjusted to contain pre-tensions of $V_{rel} = 1.001$, $V_{rel} = 1.03$ and $V_{rel} = 1.05$. Once again the value of ε was varied within the range $0.95 \leq \varepsilon \leq 10.0$, in attempt to find a value that would be stable for all cases. Figure 5.1-12 shows several of the results.

For the pre-tension set via $V_{rel} = 1.001$ it was not possible to stabilise the code; the most stable value of ε was 0.900 (which is outside the range for the original tension case ($V_{rel} = 1.01$)). For the other pre-tensions (including the original model) all the codes were stable when $\varepsilon = 1.3$. It is worth noting that as the level of pre-tension is increased, the corresponding minimum value of ε required to stabilise the code also increases. This trend is not infinite because it has been shown previously that the range values of ε in 2D that produce a stable code is very narrow.

A 3D analysis of the MRF has not been conducted for this research, but it is likely that the corresponding range of ε will be even smaller, thus making a suitable value of ε difficult (and extremely time consuming) to obtain for real models.
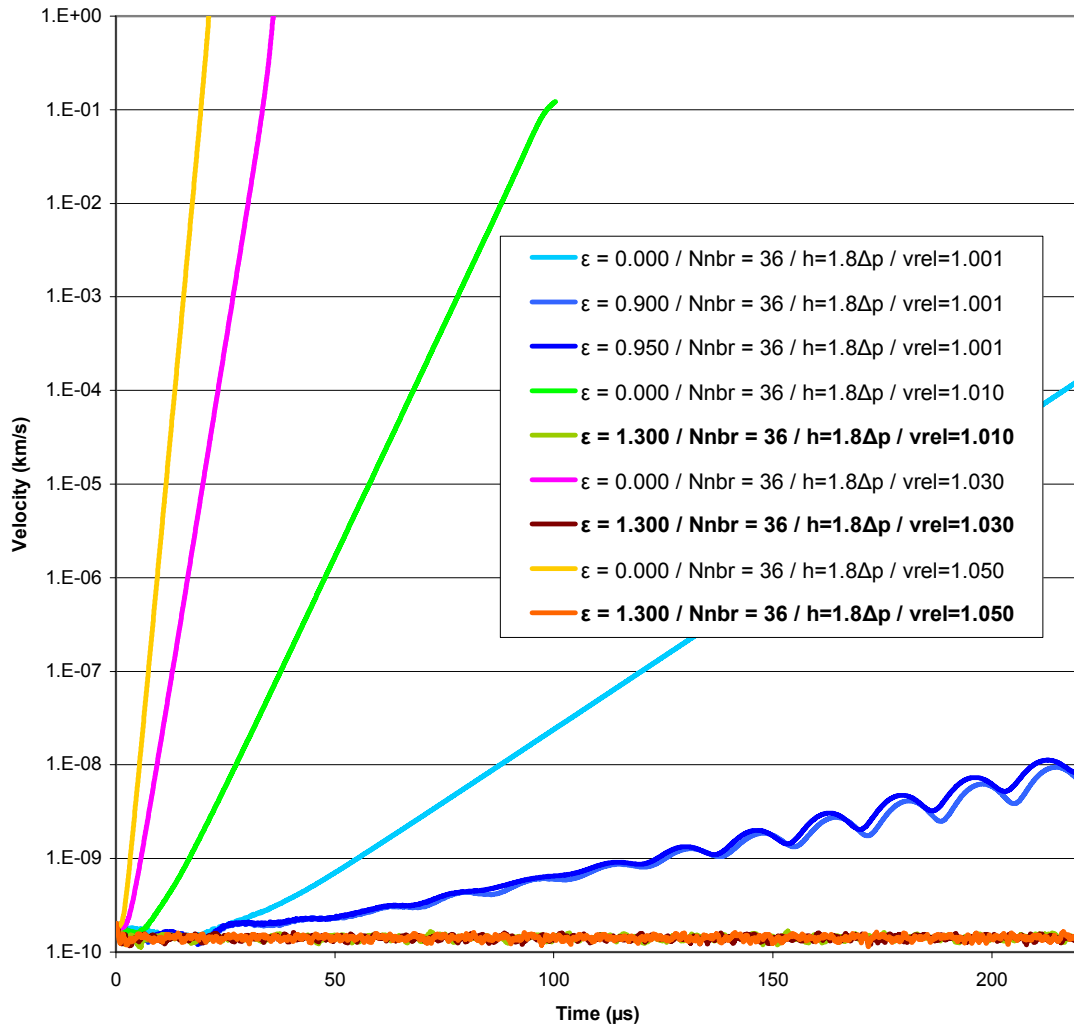
Figure 5.1-12: Varying values of ε for different levels of pre-tension for $h = 1.8\Delta p$ for the Swegle 2D problem (Δp is taken to be the initial inter-particle spacing).

5.1.4 Conclusions to the effectiveness of the repulsive force implementation

It has been shown in the previous sub-sections that Monaghan's Repulsive force is capable of eliminating the SPH tension instability but it has not been possible to obtain

a set of parameters for which the code is unconditionally stable. This sensitivity study has highlighted that the stability criteria of the MRF varies with:

- Inter-particle spacing (appropriate selection of Δp),

- The number of spatial dimensions, and

- The level of tension present in the model;

- These parameters are also likely to vary from simulation to simulation.

This level of sensitivity makes MRF in its current state unsuitable for use for this research. The results that have been obtained in the previous analysis are somewhat confusing since there are many literary sources that indicate the MRF has been successfully implemented into an SPH framework, in all dimensions.

It is possible that the choice of test problem used for this analysis is a too specific representation of the tension instability, and perhaps the MRF would work better on a real problem.

The uncertainty in the reliability of the method coupled with the time constraints of this research project have prevented the method from being investigated further. It could be possible for the MRF to be modified so that ε is calculated individually for each particle as a function of the number of neighbours present in the neighbourhood and the amount of tension between them. This idea however is outside the scope of this research project and will be left open for further investigation by others.

Further investigation is also required into the calculation of Δp. The method outlined at the end of Section 5.1.3.1 and highlighted in Figure 5.1-5 is of particular interest, (i.e. obtaining Δp as the average of each inter-particle distance within a neighbourhood). Such an analysis will require a test problem that involves deformation, in order to produce varying particle positions and neighbourhoods.

5.2 SPH with a Partition of Unity (Local / Non-Local) Approach to Approximate Derivatives

It was hypothesised that the clumping of SPH particles due to the tension instability (explained in Section 3.3.1) could potentially be eliminated by applying a partition of unity weighting to the approximation of the derivatives, such that a kernel estimate (or function, f) is split into two terms: one which approximates the function using a standard (or local) domain of influence, and another which approximates the function using a non-local (or extended) domain of influence. A similar approach is reported by Bažant (1984) to model strain-softening regions within finite elements using a non-local continuum called the imbricate continuum.

Since the implementation of the Monaghan Repulsive Force (outlined in the previous sub-section) was unsuccessful in removing the tensile instability sufficiently to provide an unconditionally stable version of the Eulerian SPH code, an investigation into a partition of unity (local/non-local) approach to SPH approximation was conducted in a final attempt to stabilise the method.

5.2.1 Theory

In the SPH method, any function, F is approximated for a particle, i via the following approximation:

$$\langle F_i \rangle = \sum_j \frac{m_j}{\rho_j} (F_i) W_{ij},$$

where the summation is over all the neighbours, j that lie in the domain of influence (neighbourhood) for particle i (neighbours are particles that lie within $2h$ of the i particle, where h is the SPH smoothing length, defined in the input). ρ is density and m is mass. W_{ij} is the SPH kernel function where $W_{ij} = f\left(r_{ij}, 2h\right)$ where r_{ij} is the length of the position vector between a pair of particles (see Section 3.1 for more information).

Since SPH approximation yields $\langle F_i \rangle \approx (F_i)$, then the term $\sum_j \dfrac{m_j}{\rho_j} W_{ij}$ can be considered to be the SPH estimate of unity, $\in$.

A partition of unity (local/non-local) weighting can be applied to a kernel estimate (or function, F) such that it is split into two terms; one which approximates the function using a local (or standard) domain of influence, and another which approximates the function using a non-local (or extended) domain of influence. The SPH estimate of unity outlined above can be used to apply the partition of unity between the two approximations such that a single expression for the function can be achieved:

$$\nabla F = \left[\nabla F_L \in \right] + \left[\nabla F_{NL} \left(1 - \in \right) \right].$$

In SPH there are two methods in which this partition of unity theory can be applied. First a derivative can be approximated by using different values for the SPH smoothing length to determine the kernel; a smaller value (local) and a larger value (non-local). The two approximations are combined via a partition of unity weighting to produce a single approximation. The neighbourhood is determined by the local smoothing length for both approximations. The second method utilises an augmented neighbourhood for the non-local approximation, the non-local smoothing length determines the new neighbourhood and the local neighbourhood is subtracted from it to leave a ring of particles. Again both approximations are combined into a single calculation via a partition of unity weighting. In both cases the SPH estimate of unity defined above is used to provide the partition of unity weighting between the approximations. For this research, the first method is termed 'the Smoothing Length Criterion,' and the second method is termed 'the Neighbourhood Criterion.' These two approaches are outlined in the following two sub-sections (see Sections 5.2.2 and 5.2.3 respectively); an explanation of the specific applications of the technique and the results that were observed is provided in each case. Section 5.2.4 provides some generalised conclusions to the method.

5.2.2 The Smoothing Length Criterion

The Smoothing Length Criterion is the simpler of the two partition of unity methods that have been implemented since it requires only a small modification to the code. In this method an approximation is calculated twice over the same domain, once with an SPH kernel that is calculated for the local domain and once with an SPH kernel that is calculated over a non-local domain, the two are then combined with a partition of the SPH estimate of unity to form a single approximation. It was decided that this method would be applied to the SPH momentum equation since the Monaghan Repulsive Force correction (outlined earlier in this chapter) is also applied in this location. However to test the theory completely, it was also applied to the strain rate equation as well (although this was tested independently).

If the SPH estimate of unity is calculated using the local domain from:

$$\in = \sum_j \frac{m_j}{\rho_j} W_{ij}^L$$

The momentum and strain rate equations are rewritten as follows:

$$a_i = \sum_j \left[\in \left(m_j \xi \nabla_i W_{ij}^L \right) + (1-\in) \left(m_j \xi \nabla_i W_{ij}^{NL} \right) \right],$$

$$\dot{\varepsilon}_i = \sum_j \left[\in \left(\frac{m_j}{\rho_j} \zeta \nabla_i W_{ij}^L \right) + (1-\in) \left(\frac{m_j}{\rho_j} \zeta \nabla_i W_{ij}^{NL} \right) \right],$$

where

$$\xi = \left[\frac{(\boldsymbol{\sigma}_i - \mathbf{q}_i)}{\rho_i^2} + \frac{(\boldsymbol{\sigma}_j - \mathbf{q}_j)}{\rho_j^2} \right], \text{ and } \zeta = \left[v_j - v_i \right].$$

The local SPH kernel function, W_{ij}^L is defined as $W_{ij}^L = f\left(r_{ij}, 2h^L\right)$ and the non-local SPH kernel function, W_{ij}^{NL} is defined as $W_{ij}^{NL} = f\left(r_{ij}, 2h^{NL}\right)$, where h^L and h^{NL} are the local and non-local SPH smoothing lengths respectively and are related via $h^{NL} = \lambda h^L$, where λ is a scale factor which is defined in the input.

5.2.2.1 Results

The Smoothing Length Criterion method of the Partition of Unity approach to approximating derivatives was implemented into the MCM code such that it could be active on either the momentum or strain rate equations, but not at the same time. In order to test the capability to correct the tension instability, the same 2D Swegle (1995) test problem which was used in the investigation of the Monaghan Repulsive Force (MRF) was adopted (see Section 5.1.2 for a detailed explanation of this test problem).

The initial relative volume in the Swegle simulations was set as $V_{rel} = 1.01$) to be consistent with the analysis performed on the MRF. The implementation was first verified via a control simulation for both the momentum and strain rate equation versions. The verification consisted of a comparison of the results of a simulation run on the Eulerian SPH code without the Smoothing Length Criterion implementation and the two versions of the implementation with λ set to 1.0. Setting λ to 1.0 produces the same equations as the Eulerian SPH method since the non-local approximation is then identical to the local approximation, i.e. the partition of unity generates a single expression made up of two partitions of the same calculation. All three simulations provided identical results and so the implementation was assessed to be functioning correctly.

The value of λ was then used to scale the local smoothing length to four larger values for the non-local components. Each value of λ was individually simulated with the Swegle test case on both the momentum and strain rate variants of the Smoothing Length Criterion method. Table 5.2-1 summarises the tests which were performed.

Test Number	Local smoothing length, h^L (cm)	Smoothing length scale factor, λ	Non-local smoothing length, h^{NL} (cm)	Number of neighbours	Correction applied to acceleration equation	Correction applied to strain equation
Control A1		1.00	0.12			
1		1.25	0.15			
2		1.50	0.18		✓	✗
3		1.75	0.21			
4	0.12	2.00	0.24	20		
Control S1		1.00	0.12			
5		1.25	0.15			
6		1.50	0.18		✗	✓
7		1.75	0.21			
8		2.00	0.24			

Table 5.2-1: Description of tests carried out for the Smoothing Length Criterion.

In Section 5.1.2 the growth of the tension instability in the Swegle problem was displayed by a logarithmic plot of the total velocity of the system with time. It was also shown that stable versions of the simulation were represented by horizontal line plots at a velocity of approximately 5.0E-11 km/s. The total velocity, v_T is calculated from the total mass, m and kinetic energy, E_K of the system via $E_K = mv_T^2/2$. The same approach has been adopted to display the results of the tests outlined in Table 5.2-1 and are displayed below in Figure 5.2-1.

Figure 5.2-1: Perturbation amplitude history for the whole system of the Swegle problem, i.e. the total velocity of the system as function of time. Results are displayed for the Smoothing Length Criterion applied to the momentum equation (left) and strain equation (right) for varying sizes of non-local smoothing length (see Table 5.2-1).

It is clear that none of the tests applied to either of the momentum or strain rate equations has had any visible effect on the stability of the simulations, since there are no differences between the control simulations and the other tests. This is not entirely unexpected since the only difference in the non-local approximation in this approach is the calculation of the SPH kernel. The value of the SPH kernel for the range of non-local smoothing lengths selected does not change significantly and therefore has little effect in the calculations. Further investigation was conducted with even higher values of λ, but no change was observed with the results. The method was therefore assessed to be inadequate for SPH stability and no further analysis into the Smoothing Length Criterion was performed.

5.2.3   The Neighbourhood Criterion

The Neighbourhood Criterion is the more complex of the two partition of unity methods that have been implemented, since it requires significant modification to the neighbour search routine. It is also a more rigorous application of the partition of unity approach since the approximation is based on two entirely different domains.

The neighbour search is augmented to find all local neighbour particles, j^L of a particle, i which lie in a standard domain ($2h^L$) and to find all non-local neighbour particles, j^{NL} of a particle, i which lie in an extended domain ($2h^{NL}$). h^L and h^{NL} are the local and non-local SPH smoothing lengths respectively. Figure 5.2-2 displays the Neighbourhood Criterion diagrammatically for an arbitrarily spaced uniform distribution of particles.
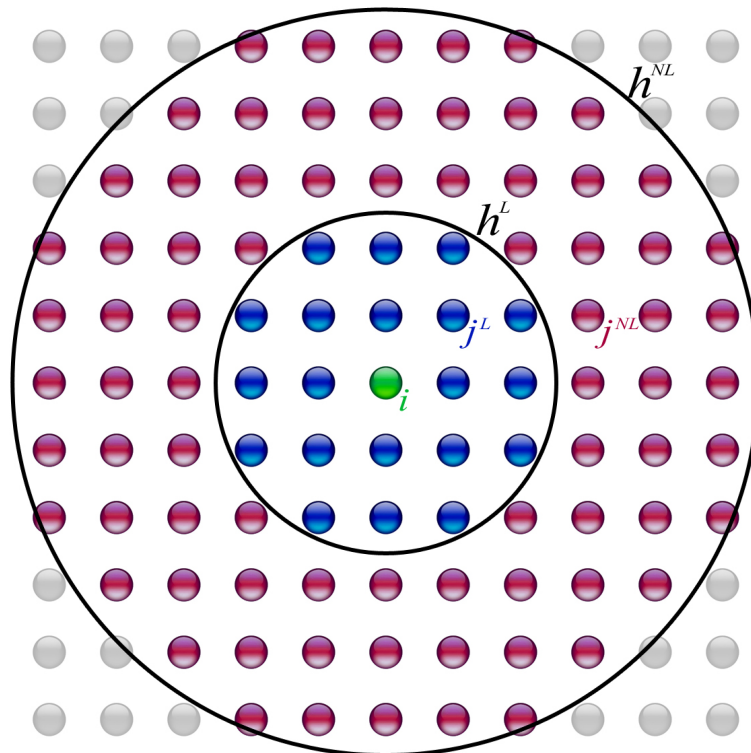
Figure 5.2-2: Diagrammatic representation of the Neighbourhood Criterion method within the Partition of Unity approach to approximate derivatives. The local (standard) neighbourhood containing all j^L neighbours (blue particles) and the non-local (extended) neighbourhood containing all j^{NL} neighbours (red particles) for a particle, i (green particle) at the centre of both domains is highlighted for an arbitrarily spaced uniform distribution of particles.

As with the Smoothing Length Criterion, the method was individually applied to both the SPH momentum and strain rate equations.

If the SPH estimate of unity is calculated from the local domain:

$$\in^L = \sum_{j^L} \frac{m_{j^L}}{\rho_{j^L}} W_{ij}^L$$

The momentum and strain rate equations are rewritten as follows:

$$a_i = \sum_{j^L} \in^L \left[m_{j^L} \xi^L \nabla_i W_{ij}^L \right] + \sum_{j^{NL}} \left(1 - \in^L\right) \left[m_{j^{NL}} \xi^{NL} \nabla_i W_{ij}^{NL} \right],$$

$$\dot{\varepsilon}_i = \sum_{j^L} \in^L \left[\frac{m_{j^L}}{\rho_{j^L}} \zeta^L \nabla_i W_{ij}^L \right] + \sum_{j^{NL}} \left(1 - \in^L\right) \left[\frac{m_{j^{NL}}}{\rho_{j^{NL}}} \zeta^{NL} \nabla_i W_{ij}^{NL} \right],$$

where

$$\xi = \left[\frac{(\boldsymbol{\sigma}_i - \mathbf{q}_i)}{\rho_i^2} + \frac{(\boldsymbol{\sigma}_j - \mathbf{q}_j)}{\rho_j^2} \right], \text{ and } \zeta = \left[v_j - v_i \right].$$

The local SPH kernel function, W_{ij}^L is defined as $W_{ij}^L = f\left(r_{ij}, 2h^L\right)$ and the non-local SPH kernel function, W_{ij}^{NL} is defined as $W_{ij}^{NL} = f\left(r_{ij}, 2h^{NL}\right)$. The local and non-local smoothing lengths are related via the following expression:

$$h^{NL} = \lambda h^L,$$

where λ is a scale factor which is defined in the input.

Provision was also made to test the above equations with an SPH estimate of unity that was calculated with the non-local neighbourhood and W_{ij}^{NL} as follows:

$$\in^{NL} = \sum_{j^{NL}} \frac{m_{j^{NL}}}{\rho_{j^{NL}}} W_{ij}^{NL}.$$

5.2.3.1   Results

The Neighbourhood Criterion method of the Partition of Unity approach to approximating derivatives was implemented into the MCM code such that it could be active on either the momentum or strain rate equations, but not at the same time. An option to use the SPH estimate of unity calculated from the local or non-local neighbourhood was also implemented. Again the 2D Swegle (1995) test problem outlined in Section 5.1.2 was adopted to test the ability of the approach to correct the tension instability.

For consistency the initial relative volume in the Swegle simulations was again set as $V_{rel} = 1.01$. The implementation was first verified via control simulations for both the momentum and strain rate equation versions, with both calculations of the SPH estimate of unity. The verification consisted of a comparison of the results of a simulation run on the Eulerian SPH code without the Neighbourhood Criterion implementation and the four versions of the implementation with λ set to 1.0. Setting λ to 1.0 produces the same equations as the Eulerian SPH method since the non-local neighbourhood is then identical to the local neighbourhood, i.e. the partition of unity generates a single expression made up of two partitions of the same calculation. All five simulations provided identical results and so the implementation was assessed to be functioning correctly.

The value of λ was then used to scale the local smoothing length to the same four larger values for the non-local components that were used in the analysis of the Smoothing Length Criterion method. Each value of λ was individually simulated with the Swegle test case on both the momentum and strain rate variants of the Neighbourhood Criterion method with both versions of the SPH estimate of unity. Table 5.2-1 summarises the tests which were performed and Figure 5.2-3 displays a graphical representation of the augmented neighbourhoods for the centre particle in the Swegle problem in each of the four cases.

Test Number	Local smoothing length, h^L (cm)	Smoothing length scale factor, λ	Non-local smoothing length, h^{NL} (cm)	Number of local neighbours	Number of non-local neighbours	Correction applied to acceleration equation	Correction applied to strain equation	SPH estimate of unity calculated with local neighbourhood	SPH estimate of unity calculated with non-local neighbourhood
Control A2		1.00	0.12		0			✓	✗
Control A3								✗	✓
9		1.25	0.15		8			✓	✗
13								✗	✓
10		1.50	0.18		16	✓	✗	✓	✗
14								✗	✓
11		1.75	0.21		36			✓	✗
15								✗	✓
12		2.00	0.24		48			✓	✗
16	0.12			20				✗	✓
Control S2		1.00	0.12		0			✓	✗
Control S3								✗	✓
17		1.25	0.15		8			✓	✗
21								✗	✓
18		1.50	0.18		16	✗	✓	✓	✗
22								✗	✓
19		1.75	0.21		36			✓	✗
23								✗	✓
20		2.00	0.24		48			✓	✗
24								✗	✓

Table 5.2-2: Description of tests carried out for the Neighbourhood Criterion.
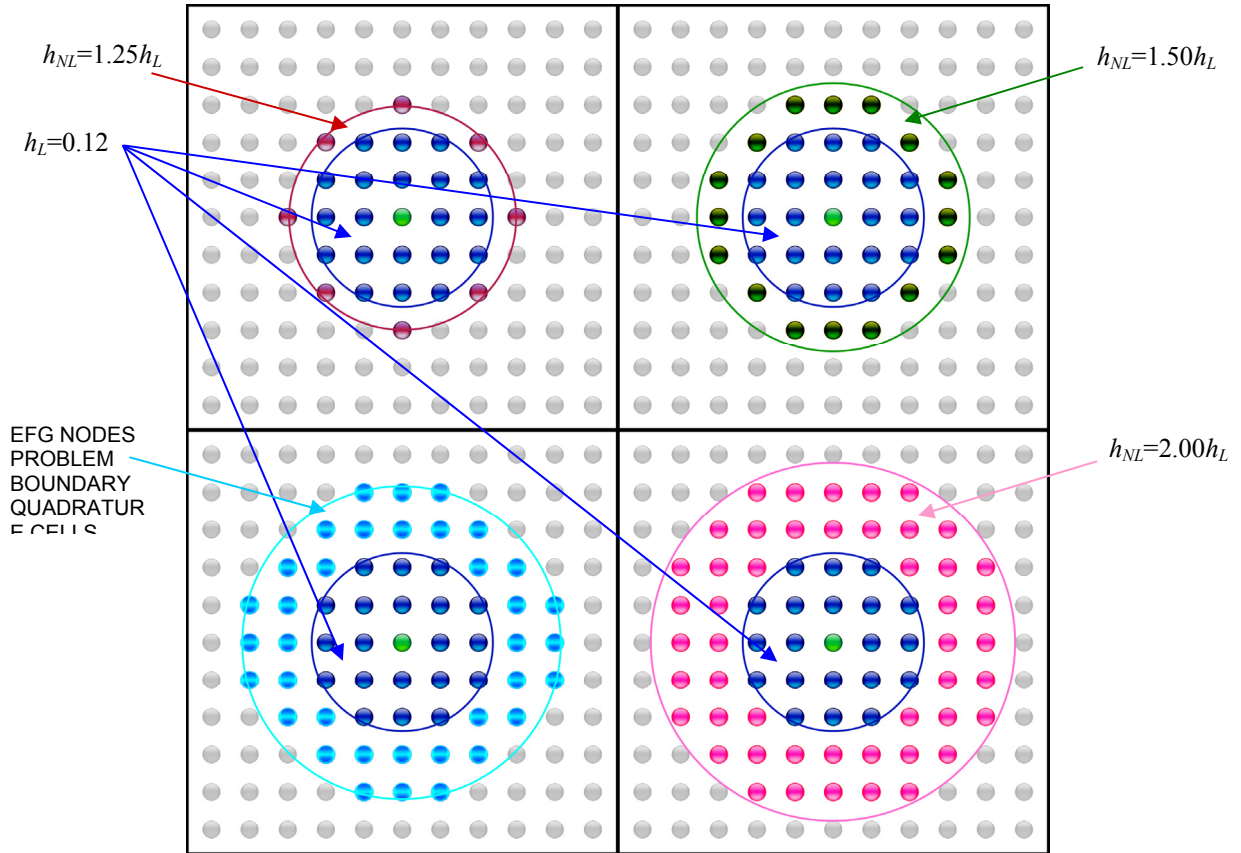
Figure 5.2-3: Diagrammatic representation of the number of neighbours in the non-local domain for different non-local smoothing lengths for the centre particle in the Swegle Problem. The local domain contains a fixed 20 neighbours (dark blue particles).

The same logarithmic plots of the total velocity of the system with time that were produced for the Smoothing Length Criterion were reproduced to display the results of the Neighbourhood Criterion tests outlined in Table 5.2-2 and are displayed below in Figure 5.2-4.
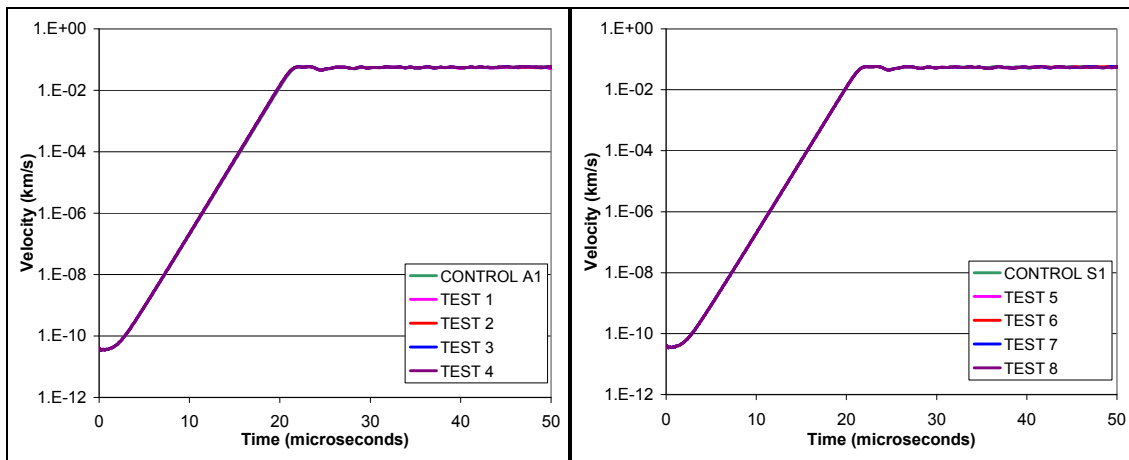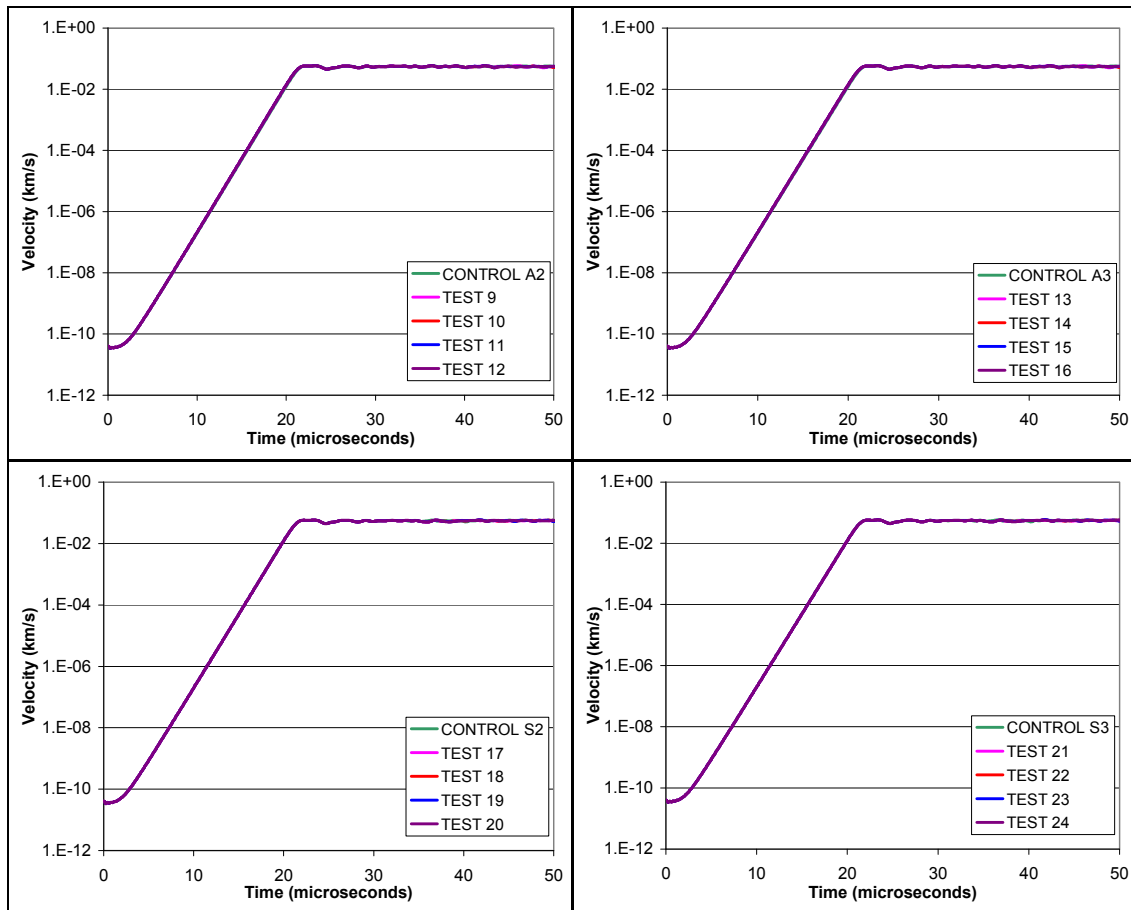

Figure 5.2-4: Perturbation amplitude history for the whole system of the Swegle problem, i.e. the total velocity of the system as function of time. Results are displayed for the Neighbourhood Criterion applied to the momentum equation (top) and strain equation (bottom) for varying sizes of non-local smoothing length. Tests were performed with a local SPH estimate of unity (left) and a non-local SPH estimate of unity (right).

Once again it is clear that none of the tests applied to either of the momentum or strain equations with either the local or non-local calculation of the SPH estimate of unity has had any visible effect on the stability of the simulations since there are no differences between the control simulations and the other tests.

The implementation was thoroughly debugged to check if a coding error could be affecting the results. An error was not found, but the investigation highlighted the cause of the problem. The local SPH estimate of unity, $\in^L$ has a value very close to unity, i.e. the non-local partition of unity used in the simulations has a value that lies in the range $0 < \left(1 - \in^L\right) <<< 1$. A representative value is 0.9979, for a particle close to the centre of the test problem in the control test case. This means that the non-local approximation has a negligible contribution to the combined approximation. It is also worth noting that the calculation of the non-local estimate of unity, $\in^{NL}$ was discovered to often yield a value that is greater than 1 and so the partition of unity weighting is invalid in this approach, i.e. all non-boundary particles at the start of the simulation have an estimate of unity of 1.00076, which fluctuates either side of 1 for the whole simulation.

A single simulation was run with the local partition of unity weighting applied to the momentum equation with a scale factor added to the non-local approximation. The scale factor was set to 10 to see if any response could be observed in the total velocity of the system. The result was the same as the previous experiments. The Neighbourhood Criterion method was therefore also assessed to be inadequate for SPH stability and the investigation into The Partition of Unity approach to approximating derivatives was consequently abandoned.

5.2.4   Conclusions to the Partition of Unity approach

Analysis was conducted for two methods of applying a partition of unity weighting to either the SPH momentum equation or strain rate equation using the SPH estimate of unity as the weighted variable. The first method was termed the Smoothing Length Criterion. The partition of unity was applied to a local and non-local approximation of the equation by using a larger SPH smoothing length to calculate the SPH kernel in the non-local contribution. The second method was termed the Neighbourhood Criterion and was the same as the Smoothing Length Criterion, except that the non-local approximation was also calculated over a larger neighbourhood.

Unfortunately neither of the approaches were able to correct the SPH tensile instability and also did not provide any indication that the partition of unity approach was having even a minor effect on the stability.

Investigation into the implementation of the methods determined that the SPH estimate of unity was very close to unity for the test problem that was being used, and so the non-local contribution was negligible in the calculation. It is therefore unclear if the choice of test problem was too uniform to produce estimates of unity that are representative of normal values, or if the approach itself cannot be applied in this manner. These questions are left for further work since it was assessed that further investigation would not likely yield a working method in the remaining time available that could be used with the development of the damage/fracture model for this research.

The test simulation that was selected for this work was specifically designed to quantify the tensile instability so that the level of instability could be easily monitored. However it would be interesting to test the code on different simulation models that exhibit the effects of the tension instability to identify if a better partition of unity weighting is present and if a response is achieved.

It was observed that the Monaghan Repulsive Force (MRF) implementation applied to the SPH momentum equation in Section 5.1 can also be considered to be a form of local/non-local approach. This is evident when the SPH momentum equation is expressed with a fully expanded MRF term as follows:

$$\overline{a}_i = \frac{d\overline{v}_i}{dt} = -\sum_j m_j \left[\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} - \left[\varepsilon \left(\frac{|P_i|}{\rho_i^2} + \frac{|P_j|}{\rho_j^2} \right) \left(\frac{W(r_{ij})}{W(\Delta p)} \right)^n \right] + \Pi_{ij} \right] \nabla_i W_{ij} ,$$

NON-LOCAL   LOCAL

(see Section 5.1 for an explanation of the terms in the above equation). The MRF in the above equation provides effects from pairs of particles to a neighbourhood. The MRF could be considered to be a local component and the neighbourhood could be considered to be a non-local component. Using this assumption it can be argued that the

partition of unity weighting applied in this research could be switched around so that the $\in$ term is applied to the non-local contribution and the $(1- \in)$ term is applied to the local contribution. Investigation into this assumption is also left for further work.

5.3 Conclusion to the Implementation of Eulerian SPH Stability Solutions

At the start of this research project two versions of the SPH code were available; a standard Eulerian implementation of the SPH method and a Total Lagrangian (TL) formulation of the method. The advantage of the Eulerian method is that it can be used to model the explosively driven experiments that were investigated for this research (see Chapter 6) and were of particular interest to provide a validation framework for the fracture model that would be developed. Its disadvantage is that it contains a numerical problem which is apparent in tension (see Section 3.3.1) which would hinder the development of a fracture model. The advantage of TL formulation is that it does not contain the tensile instability, although it is not capable of modelling very large deformations (such as those produced by an explosion) and is therefore not able to model the explosively driven experiments.

Prior to the completion of the literature survey (which was being conducted to develop ideas about the direction that would be taken to develop a fracture model), it was unclear which version of the SPH code would be most beneficial to the project. Therefore an investigation into stabilisation methods was conducted alongside the literature survey in an attempt to produce a stable Eulerian SPH formulation. If a stable version of the Eulerian SPH method had been obtained, it is quite likely that it would have been utilised (due to simplicity and the ability to model large deformations) and a very different approach to modelling the damage outlined later in this thesis would have been taken.

Two different approaches were identified for solving the tensile instability, the Monaghan Repulsive Force (MRF) and the Partition of Unity method for approximating SPH derivatives. These two methods have been described in detail in this chapter and

tests have been performed to assess the ability of each method to correct the tensile instability.

The MRF is capable of eliminating the tensile instability but it has not been possible to obtain a set of parameters for which the code is unconditionally stable in all dimensions. The partition of unity method applied in this research did not produce a stable code. Both of the methods require further investigation before either can be completely ruled out; however any further analyses into these techniques is outside the scope of this research.

It should be noted that a significant amount of the project time was taken up with the implementations included in this chapter. The failure of either approach to provide a stable Eulerian SPH framework led to the decision to utilise the TL formulation of the SPH method to develop the fracture model discussed later in this thesis. This decision has meant that it has not been possible to use the simulations of the explosively driven experiments (contained in the next chapter) to validate the developed fracture model, since the existing TL formulation that was available for this research was not able to model them successfully.

This page is intentionally blank

6   Modelling of Explosively Driven Experiments

This chapter aims to explain the modelling that has been performed on two specific explosively driven experiments taken from literature. Research into this area was conducted with the intention of validating the developed fracture model (discussed in the next two chapters) through its capability to reproduce the results of these experiments. Whilst this has not been possible (for reasons that were discussed at the end of the previous chapter); a significant amount of the project work went into modelling these experiments and it is therefore necessary to include an explanation of the modelling development.

The specific experiments that were chosen for this research were discussed and agreed with the AWE early into the project. In the past there have been many studies conducted to observe and categorise fragmentation; however, there are only a few which apply to this research, since this research is only interested in metals subject to high strain rate loading.

Candidate experiments are those that accurately describe the experiment and display the results clearly, thus allowing a representative simulation to be constructed. The type of experiments that have been researched are explosively driven cylinders or shells, since these deal directly with the type of fracture and fragmentation at high strain rate that is within the scope of this research. The first candidate experiment was carried out by Slate, Billings and Fuller (SBF) (1967) and the second was conducted by Mock and Holt (1983). The methodology and set-up of both of these experiments were discussed in Section 2.1 and therefore are not repeated here. This Chapter deals with the numerical simulations of these experiments that were conducted for this research.

6.1   Baseline Simulation of the Slate Billings and Fuller Experiment

The Slate Billings and Fuller experiment (SBF) was identified and explained in Section 2.1.1.1; it was also identified by the AWE as a test problem that is used to validate their existing fracture model. It was initially regarded as a potential experiment of interest

and a baseline investigation into modelling the experiment was conducted using LLNL-DYNA3D. Unfortunately the results that are contained in the original paper are quite general and it does not provide data that can be used to discriminate between models. Since only the published paper was available, it was decided to abandon the modelling of this experiment in favour of the more detailed Mock-Holt experiment (See Section 2.1.1.2). The results of the limited investigation into the SBF experiment are detailed below.

An initial investigation into the SBF experiment was conducted with LLNL-DYNA3D to begin development of the modelled metal shell (leading to SPH particle packing arrangement). A simple 2D representation of the model was constructed using a quarter of the explosive and shell, with symmetry planes on each edge. Two versions of the model were tested; a coarse mesh (80 elements in the metal shell, with 4 through the thickness), and a fine mesh (1000 elements in the metal shell, with 10 through the thickness). A butterfly mesh arrangement was used to construct the explosive (see Figure 6.1-1). The dimensions of the experiment are taken from a typical set-up in Slate, Billings and Fuller (1967) and are detailed in Table 6.1-1.
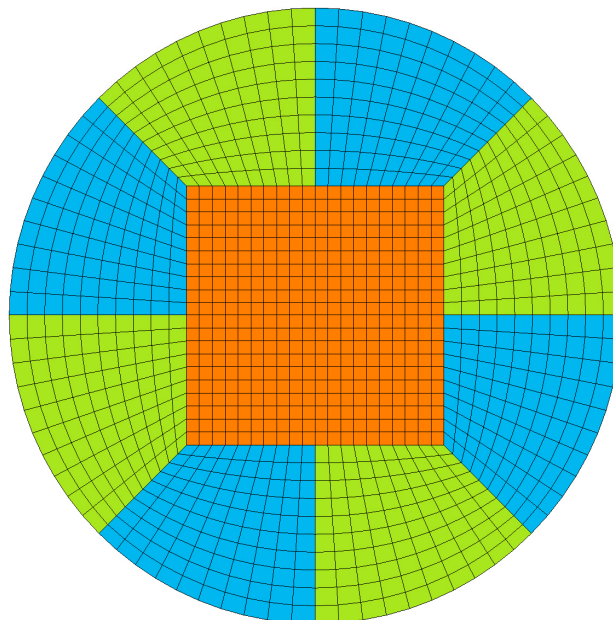
Figure 6.1-1: 2D butterfly mesh arrangement for representing a circle.

The Steinberg-Guinan constitutive model (see Section 4.3), with a Gruneisen equation of state (see Section 4.8 for more information) was used to describe the metal shell, which was chosen to be aluminium. The material parameters were taken from Steinberg (1996) and are detailed in Table 6.1-2 The explosive material was chosen to be TNT in accordance with the SBF experiment (material parameters are contained in Table 4.7-1).

r_e (cm)	r_i (cm)	r_o (cm)	θ_0 (cm)
4.75	6.00	6.50	0.5

Table 6.1-1: Model parameters for SBF simulation.

Material Parameters (Steinberg, 1996)										
Density (g/cm^3)		Shear Modulus, G_0 (100 GPa)			Yield Stress, σ_y (100 GPa)			Melting Temperature, T_{m0} (K)		
2.804		0.267			0.452E-05			1220		
Constitutive Constants (Steinberg (1996), *Wikipedia, †AWE)										
A^* (g mol^{-1})	β	n	b	b'	h	$f^†$		γ_0	a	p_{cut}
26.9815	965	0.10	65	0.0	-0.62	-0.001		2.2	1.7	-3.8
Equation of State Parameters (Steinberg, 1996)										
C	S$_1$	S$_2$	S$_3$	γ_0	a		E$_0$		V$_0$	
0.524	1.4	0.0	0.0	1.97	0.48		0.0		1.0	

Table 6.1-2: Material parameters for aluminium shell in SBF simulation.

Diagrams displaying the two simulations at 0 µs and 25 µs respectively are contained in Figure 6.1-2 (coarse mesh simulation) and Figure 6.1-3 (fine mesh simulation).

In the coarse mesh simulation, the explosive expanded up to the aluminium shell which subsequently started to stretch, this behaviour was observed until the simulation end time (100 µs). This was expected since no fracture model is present in the simulation and the explosive will continue to expand indefinitely. This simulation was essentially just an inexpensive test of the model set-up; the mesh is too coarse to be of any use for detailed analysis. However it is worth noting that since the mesh lines of the explosive are not coincident with those of the aluminium shell, the mesh becomes distorted with increasing expansion. This effect is magnified in the fine mesh example: the mesh becomes entangled and causes the simulation to terminate early. This would not have posed a problem for an SPH simulation of the SBF experiment, since the method is meshless and therefore not subject to mesh entanglement.
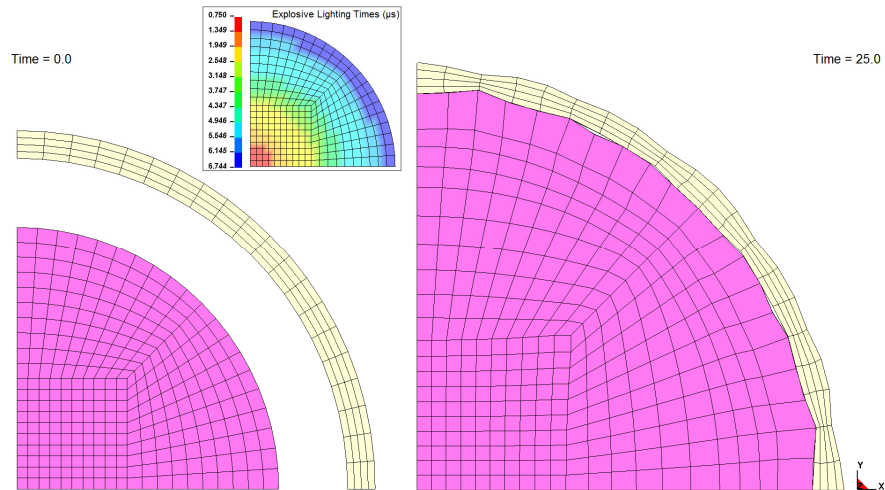
Figure 6.1-2: 2D coarse mesh LLNL-DYNA3D SBF simulation at 0 μs and 25 μs.
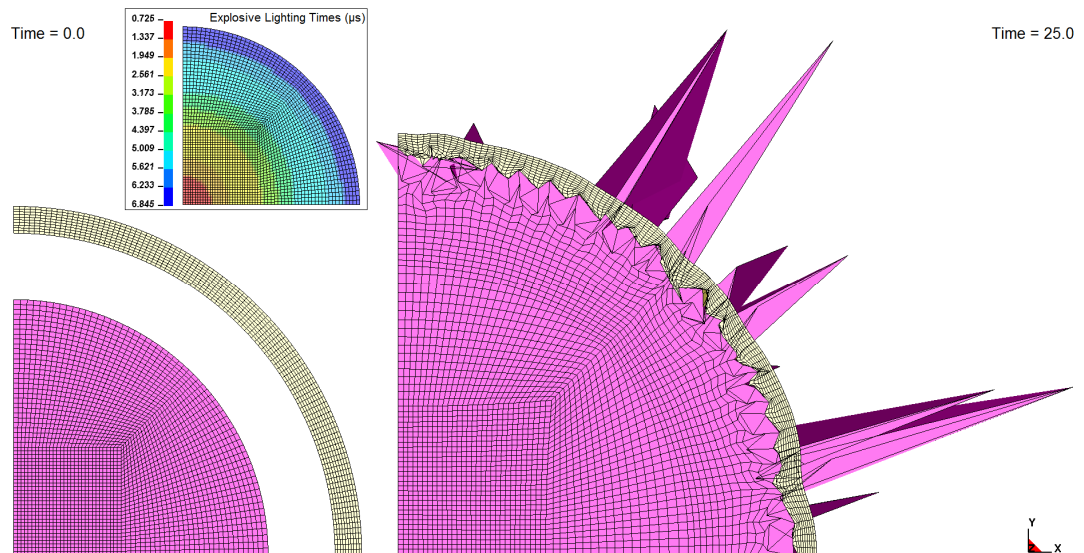
Figure 6.1-3: 2D fine mesh LLNL-DYNA3D SBF simulation at 0 μs and 25 μs.

Further investigation into this experiment would have included extending the model to the SPH method and attempting simulations with different particle packing arrangements such as; radial packing and the generation of a 3D model. It was decided after the above simulations were performed to abandon the modelling of the SBF experiment in favour of the more detailed Mock-Holt experiment (See next sub-section). This was due to the SBF experiment results being quite general and difficult to deduce specific behaviour for specific experiments, i.e. the analyses concentrate on strain to rupture and the change in radii with time. Both of these provide a limited ability to discriminate between the results accurately.

6.2   Simulations of the Mock-Holt Experiment

The literature survey conducted for this research was extended to search for other candidate explosively driven experiments along side the initial modelling of the Slate, Billings and Fuller (SBF) experiment (1967) detailed in the previous sub-section. One experiment that came out of this survey was of particular interest, since it contains more detailed information than the SBF experiment; it is called the Mock-Holt (MH) experiment (Mock, Holt 1983). A description of the experiment is provided in Section 2.1.1.2.

The MH experiment is of interest since it captures the type of fracture that this research is attempting to model. It is also of interest because Randles and Libersky (1996) conducted a simulation of the Mock-Holt experiment (ARMCO iron cylinder) using an SPH code to assess the SPH method's capability in modelling fracture. They used an Eulerian SPH formulation, with a Johnson-Cook constitutive model and a Mie-Gruneisen equation of state for the ARMCO iron and the J.W.L. equation of state for the Composition B explosive. They employed a scalar shear and tensile damage model to predict the break-up of the iron cylinder. They concluded that the SPH method gives very promising results. Figure 6.2-1 shows a diagrammatic representation of the Randles and Libersky simulation model at 0 µs and a sectional view at 40 µs. Figure 6.2-2 shows the experimental data for fragment mass distribution compared to the fragments generated by their SPH simulation. It is clear that the simulation matches exceptionally well with the experiment.

The evolving scalar damage model used by Randles and Libersky to facilitate the break-up of the Mock-Holt cylinder represents damage as a field variable which is stored for every particle and is used to soften some of the material properties. Specifically the damage reduces the tensile bulk modulus, shear modulus, flow stress and the SPH smoothing length. It is the reduction of SPH smoothing length which causes the material to fracture by limiting the communication between particles, i.e. as SPH neighbourhoods become smaller, particles stop communicating with each other and numerical fracture occurs (see Section 3.2 for a description of numerical fracture). Although this approach yields successful results, it exploits a break-down in the SPH

code to generate fracture rather than relying on a mathematical model to break-bonds or split particles at specific locations. Since the approach cannot be verified mathematically, it is therefore not a robust method for modelling the fragmentation.
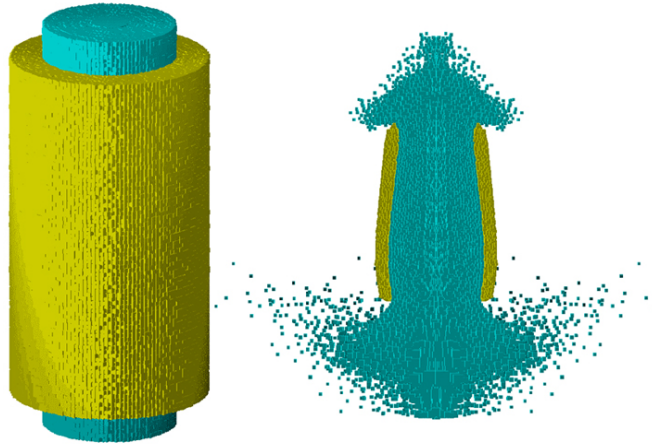
Figure 6.2-1: Mock-Holt Experiment conducted by Larry Libersky of LANL shown at 0.0 and 40 μs (Randles, Libersky 1996).
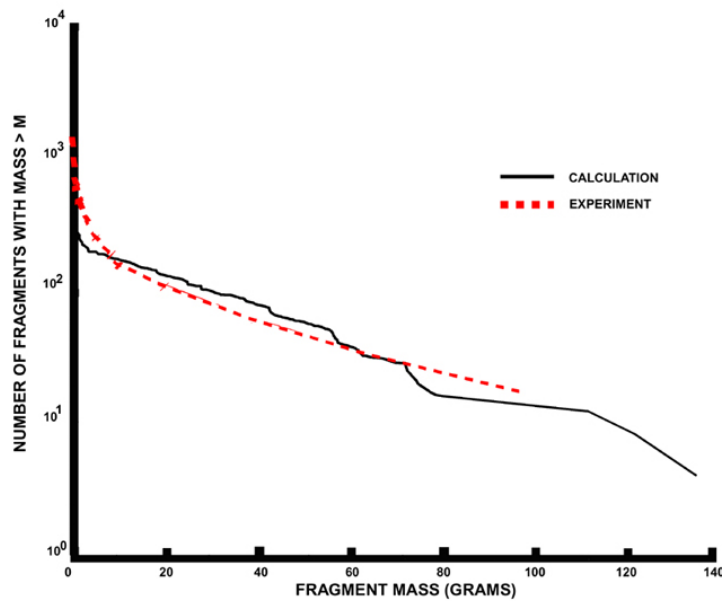
Figure 6.2-2: Larry Libersky's Mock-Holt Simulation results against experimental data (Randles, Libersky 1996).

It was decided that some detailed simulations of the MH Experiment would be conducted as a part of this research to allow for comparison to Randles and Libersky's findings, but also to provide a test case for the final fracture model. Unfortunately the

latter was not possible for this research since the developed fracture model ended up utilising the Total Lagrangian SPH formulation which is not compatible with the simulation of the explosive detonation product used in the Mock-Holt experiment[5].

The following sections outline the set-up of the simulation models and explain the investigation that was performed into different particle-packing arrangements in order to achieve stable simulations.

6.2.1 Mock-Holt Simulation Model Dimensions and Material Parameters

All Mock-Holt simulations contained within this research are based on Experiment No. 1 contained within the original paper (Mock, Holt 1983).

The explosive for this experiment is Composition Type B High Explosive (HE). Material parameters for the JWL EOS are taken from Dobratz and Crawford (1985) and are contained in Table 6.2-1.

Explosive	Density, ρ_0 (g / cm^3)	C-J Pressure, P_{CJ} (Mbar)	Detonation Velocity, D	Initial Energy, E_0 (Mbar)	A (Mbar)	B (Mbar)	R_1	R_2	ω
Comp B	1.717	0.295	0.798	0.0850	5.242	0.07678	4.20	1.10	0.34

Table 6.2-1: Material Parameters for Composition Type B HE (Dobratz and Crawford 1985).

The dimensions for the explosive are taken from Mock and Holt (1983) and are contained in Table 6.2-2.

Length (cm)	Diameter (cm)	Mass (g)
35.7	7.62	2750

Table 6.2-2: Dimensions for Composition Type B HE (Mock and Holt 1983).

The metal cylinder is made from (ARMCO) iron; unfortunately no one data source provided all the required input parameters for this material and so the data were collected from a number of resources and are contained in Table 6.2-3.

---

[5] Developments into a coupled Eulerian / Lagrangian formulation of the SPH method that can use different kernels for different materials have been made at Cranfield University since the end of this research. Mock-Holt simulations have been performed that could eventually be used to test the developed fracture model (see Chapters 7 and 8), although such advances are outside the scope of this research.

Material Parameters ([a]Johnson and Cook (1983), [b]Wikipedia)			
Density[a] (g/cm^3)	Shear Modulus[b], G (100 GPa)	Specific Heat[a], C_v (cm^2 μs^{-2})	Melting Temperature[a], T_m
7.890	0.82	0.452E-05	1181
Constitutive Constants (Johnson and Cook 1983)			
A (100 GPa)	B (100 GPa)	C	n	m
1.75E-03	3.80E-03	0.060	0.32	0.55
Equation of State Parameters ([a]Steinberg (1996), [b]Kinslow (1970), [c]Szichman and Krumbein (1986))			
C[a]	S$_1$[b]	γ_0[c]	a[a]
0.4096	1.5244	2.15	0.5

Table 6.2-3: Material parameters for ARMCO Iron.

The dimensions for the cylinder are also taken from Mock and Holt (1983) and are contained in Table 6.2-4.

Length (cm)	Inside Dia. (cm)	Outside Dia. (cm)	Location along Explosive* (cm)	Wall Thickness (cm)	Mass (g)
20.3	7.62	11.43	9.0	1.91	9130

(*From the detonator end of the explosive)

Table 6.2-4: Dimensions for ARMCO Iron (Mock and Holt 1983).

6.2.2   2D Initial Simulations

In order to determine the best set-up for the Mock-Holt experiment, some initial simulations were performed in 2D using the Eulerian SPH formulation with no fracture model present. A coarse square-packed mesh of particles was defined to enable short run times, so that changes could be tested quickly. The model was a 2D slice of the centre of the Mock-Holt experiment. The mesh contained 3,808 particles, 3,016 in the explosive and 792 in the iron. The iron was separated into two materials of 396 particles since the 2D model contained unconnected iron material either side of the explosive. Each iron slice contained 6-particles through the thickness. The inter-particle spacing was 0.3 cm and the SPH smoothing length was greater than this value by a factor of 1.3.

The initial simulation was performed without a specific contact algorithm and kernel contact between the SPH particles; then several tests were performed with a repulsive

force contact algorithm which was available in the SPH code[6]. The kernel contact simulation ran successfully, although evidence of the SPH tensile instability was present in the results; i.e. rows of particles in the iron clumped together and numerical fracture occurred at the top of the iron (see Section 3.3.1 for more information). In the repulsive force contact simulations with typical values of the input parameters ($k = 0.1$ and $n = 4.0$), the first layer of explosive particles passed through the iron. Increasing the input parameters to values outside the typical values ($0.3 \le k \le 0.7$ and $5.0 \le n \le 8.0$) did not solve the problem. Only when the input values were increased to values far outside the typical values ($k = 0.8$ and $n = 10.0$), did the explosive remain within the iron shell.

It was concluded that the repulsive force contact algorithm was likely to be unsuitable for providing contact between the explosive and the iron in 3D simulations although some tests would be conducted to verify if this was the case. Figure 6.2-3 below shows several of the simulations at 40 µs. The differences between the simulations are clear and it is apparent that the simulation with contact input values of $k = 0.8$ and $N = 10.0$, provide the best agreement with Libersky's simulation (Figure 6.2-1).
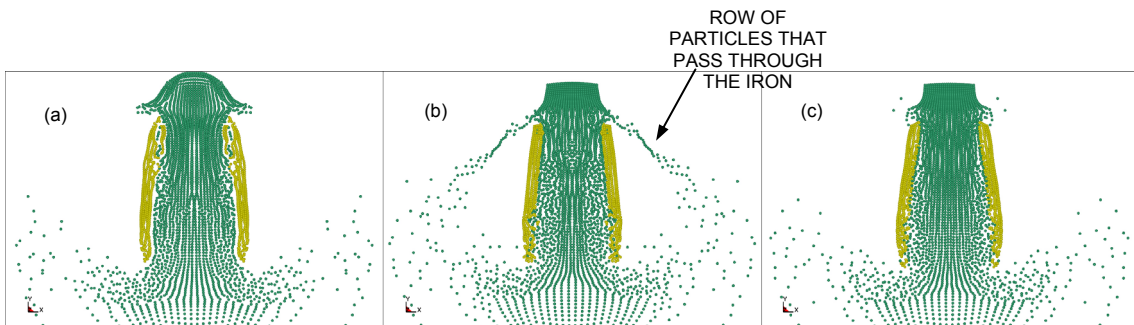
Figure 6.2-3: 2D Mock-Holt simulations at 40 µs for (a) kernel contact and repulsive force contact with (b) $k = 0.1$ and $n = 4.0$ and (c) $k = 0.8$ and $n = 10.0$.

6.2.3   3D Square-Packed Simulations

The Mock-Holt experiment was extended to 3D with the same material parameters that were used for the 2D simulations. Initially a square-packed arrangement of the particles

---

[6] See Annex 2 for a description of the contact descriptions used in this research. Section An2.1 describes the kernel contact description and Section An2.2 describes the repulsive force contact description.

was used to represent the configuration of the Mock-Holt experiment. The initial model was coarse to minimise the computation time. It was made up of 101,168 particles; 42,240 particles in the ARMCO iron and 58,928 in the explosive. The inter-particle distance, Δp was 0.301 cm and the smoothing length was set to $1.3\Delta p$. The Eulerian formulation of the SPH method without any fracture model present was used in the simulations. Figure 6.2-4 shows a diagrammatic representation of the simulation model.

Figure 6.2-4: Cross Section and 3D View of the Mock-Holt Simulation Model using square-packed SPH particles.

The first simulation was performed with the repulsive force contact algorithm active and the input parameters were set to the values determined from the 2D simulations (see previous section). The simulation terminated with complex speeds of sound in an explosive particle as soon as the explosive material expanded and came into contact with the iron. Other input values were tested but none could be made to work successfully and therefore the contact algorithm was discarded and further analysis was performed using kernel contact. The kernel contact simulation reached a time of 84 µs at which point the time-step became very small and thus unable to progress the simulation further, this was possibly due to large deformations in the ARMCO iron.

Figure 6.2-5 shows a longitudinal sectional of the model at 40 µs as a direct comparison with Randles and Libersky's simulation (shown in Figure 6.2-1). It can be seen that the response is similar but there are differences mainly in the spread of the explosive

material. It should be noted that the model created for this research was generated using the dimensions specified in Mock and Holt (1983) and that Randles and Libersky's simulation uses less explosive material either side of the ARMCO iron cylinder. As a first iteration of the simulation the result was very promising.
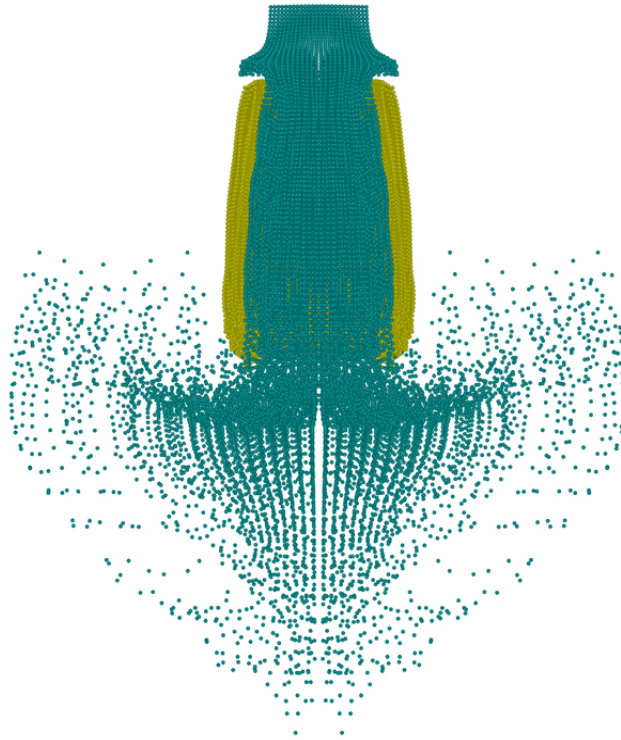
Figure 6.2-5: Cut-away, side projection of the Mock-Holt simulation using square-packed SPH particles at 40 μs.

The model was extended to use a much finer mesh, but the onset of the drop in time-step size occurred earlier in this simulation and thus it did not progress far enough to produce any useful results. A closer analysis identified that the simulation failed soon after the explosive detonated in the region near the base of the iron. The particles began clumping together and explosive material moved between the gaps that were formed. This occurred on the planar edges of the cylinder which were a feature of the square-packing arrangement. It can therefore be concluded that a square-packing arrangement is not the best way to represent the cylindrical shape as it prevents a linear contact between explosive and iron and leads to rapid development of the tensile instability (discussed in detail in Section 3.3.1). The model was therefore extended to utilise different mesh types and these are discussed in the following sections.

6.2.4   3D Butterfly Mesh Simulation

As discussed previously the square-packed particle arrangement does not accurately define the circular shape of the Mock-Holt experiment. An octagonal butterfly mesh was generated using the Hypermesh software package for FE modelling, and was converted to SPH particles using a tool that was developed at Cranfield University which places an SPH particle in the centre of every element and assigns it the element mass. The element volume is used to compute the SPH smoothing length. Figure 6.2-6 shows a cross-section of the Mock-holt simulation with the octagonal butterfly arrangement.
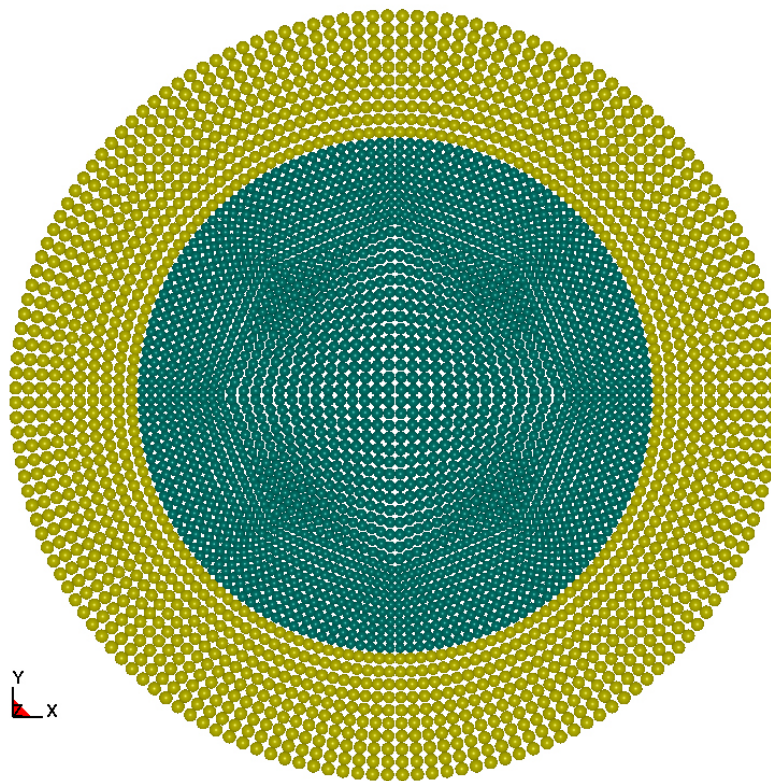
Figure 6.2-6: Cross-section of the octagonal butterfly SPH particle mesh to describe the Mock-Holt experiment.

Initial simulations were configured exactly like the square-packed simulations with the only differences being the size and shape of the mesh. All modelling attempts using this packing-arrangement were unsuccessful because several of the explosive particles achieved complex speeds of sound when the detonation of the explosive reached the

edge of the iron cylinder. This is thought to be due to the fact that the packing is very irregular and the volumes of each particle vary vastly across the cross-section. This leads to irregular neighbourhood sizes. In some cases particles have very small neighbourhood sizes and the large deformation of the explosive leads to rapid onset of numerical fracture, such particles become more energetic than others which leads to explosive particles that travel freely through the other explosive material and/or the iron. This packing arrangement was discarded and further analysis was conducted with a radial-packed particle mesh.

6.2.5   3D Radial-Packed Simulation

It was assessed that the best way to allow for an even contact between SPH particles in an expanding cylinder was to utilise a radial-packing arrangement. In a radial particle arrangement, the particle representation of the outer surface of the explosive is evenly distributed around the particle representation of the inner surface of the iron cylinder, thus allowing a smooth transfer of energy between the two materials. A FORTRAN program was developed specifically to generate the Mock-Holt shape. It places a centre particle and then populates concentric rings of a defined radius with an evenly spaced number of particles. The minimum particle spacing and the number of concentric rings within the mock-Holt experiment dimensions can also be defined in the input. The source code for this program is included in Appendix D which explains the program in more detail.

Several meshes were generated but the best results were observed with a configuration that contained 426,247 particles, 174,727 in the iron and 251,520 in the explosive. The iron had 10 particles through the thickness and the explosive had 20 rings of particles and the centre particle. On each face there were 1,603 particles in the iron and 1,310 particles in the explosive. The iron had 109 rows through its length and the explosive had 192. The average inter-particle spacing was 1.88 mm. The mesh is represented diagrammatically in Figure 6.2-7, but a larger, clearer version is also included in Appendix D.
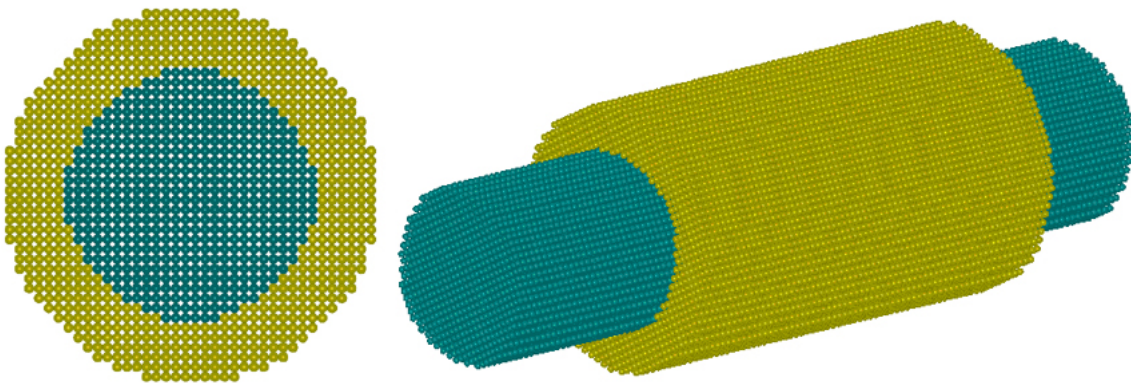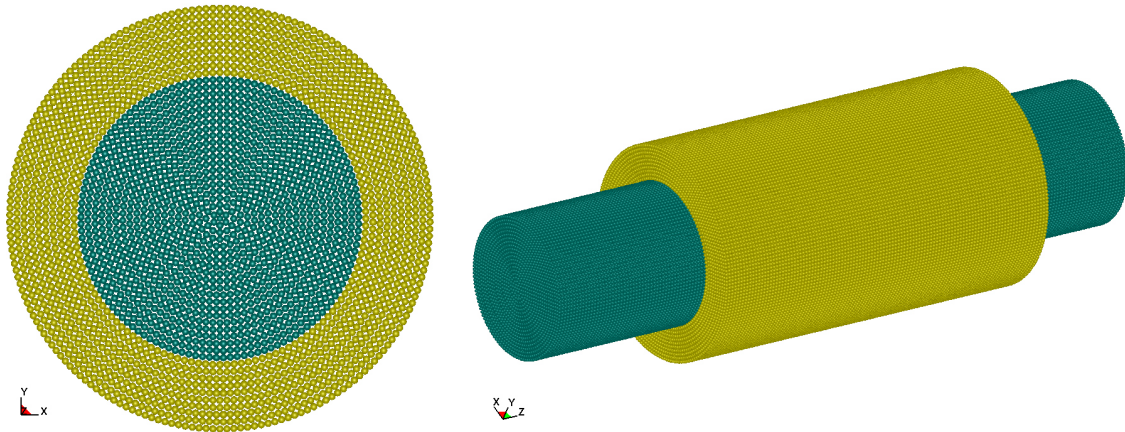
Figure 6.2-7: Cross Section and 3D View of the Mock-Holt Simulation Model using radial-packed SPH particles.

Initial simulations were configured exactly like the square-packed simulations with the only differences being the size and shape of the mesh. The simulation ran to 153 µs when termination occurred again due to complex speeds of sound in several particles. This length of time was considered to be adequate since fracture would already have occurred in the real experiment by this time and therefore further studies with the simulation were stopped. Figure 6.2-8 shows a side projection of the iron cylinder with the explosive material hidden from view at 39, 60, 81, 120 and 153 µs respectively. Figure 6.2-9 shows a longitudinal sectional of the model at 40 µs as a direct comparison with Libersky's simulation (shown in Figure 6.2-1). It can be seen that the response is similar but again there are differences, mainly in the spread of the explosive material. Figure 6.2-10 shows the same side projection at 153 µs, just prior to the termination of the simulation.

It is clear that numerical fracture occurs and the iron fragments into many pieces. Since this research is interested in robust mathematical modelling of the failure state of metals at high strain rate, it was assessed to be unnecessary to perform any post-processing of the data generated from this simulation. It would be possible to determine the fragment mass distributions and compare them directly to Libersky findings presented in Figure 6.2-2, but there is little to be gained from this exercise.
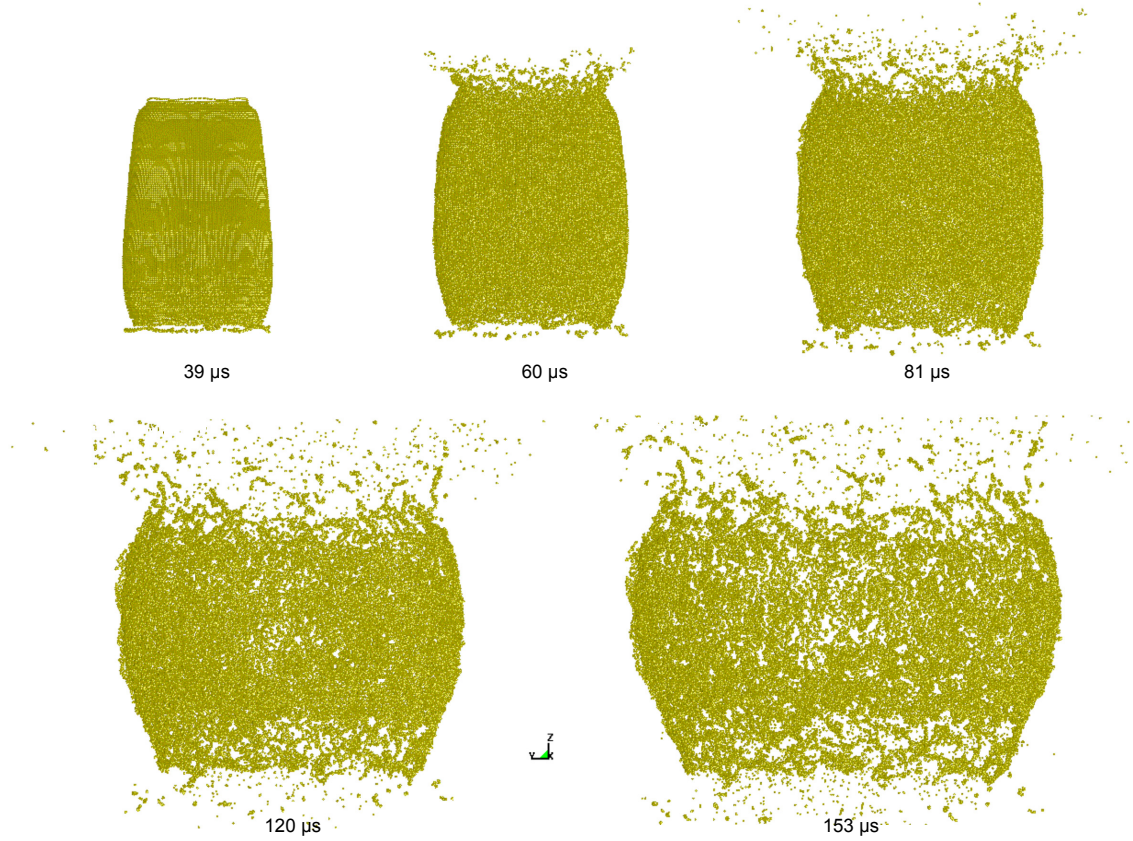
Figure 6.2-8: Side elevation of the iron cylinder at various times throughout the simulations of the Mock-Holt experiment using radial-packed SPH particles.
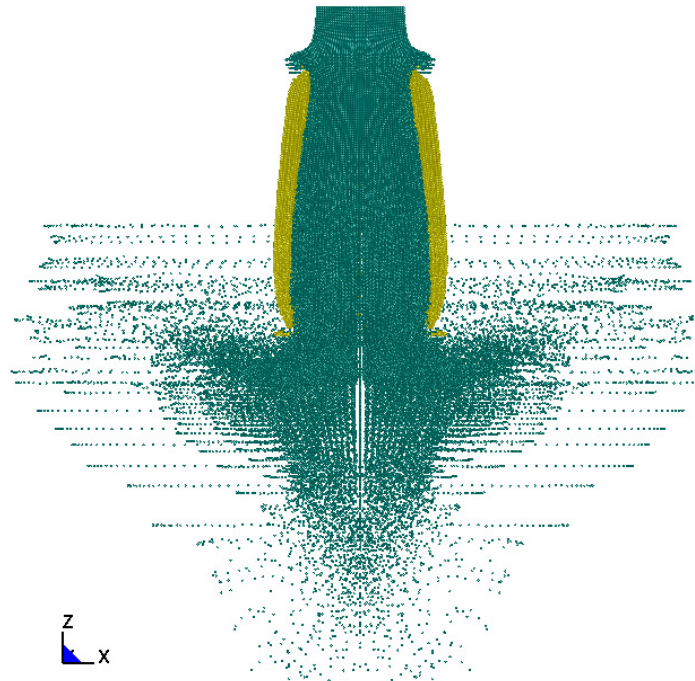
Figure 6.2-9: Cut-away, side projection of the Mock-Holt simulation using radial-packed SPH particles at 40 µs.
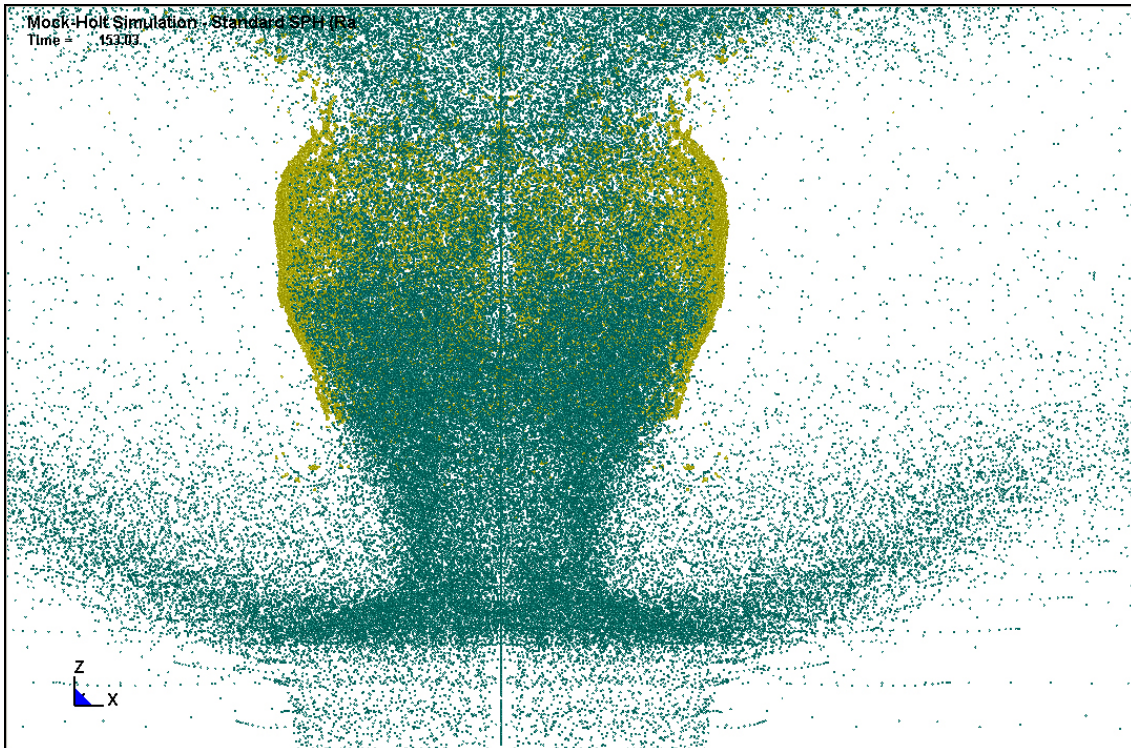
Figure 6.2-10: Cut-away, side projection of the Mock-Holt simulation using square-packed SPH particles at 153 µs.

This simulation highlights the importance of having a stable SPH and provides justification for the investigation into various Eulerian SPH stabilisation techniques outlined in Chapter 5. Since that investigation proved to be unsuccessful, it has not been possible to generate a Mock-Holt simulation that is not prone to the numerical fracture[7].

---

[7] Work was conducted at Cranfield University after the completion of this research into a combined Eulerian / Lagrangian kernel. The simulation model with the radial-packing arrangement outlined in this section has been modelled with a Lagrangian description for the iron and the simulation has produced stable expansion beyond the point where a fracture model would predict the break-up of the material. Unfortunately such advances in the SPH code were outside the scope of this research and therefore the results of these tests are not present herein.

7   A Particle-Particle Interaction Fracture Model

The main objective of this research was to develop a methodology for modelling the initiation and growth of damage and subsequent failure of metals subjected to high strain rate loading within an SPH framework. From the literature survey (see Section 2.2), it is possible to identify the principle features/actions that any competent fracture model should contain, these criteria are listed below:

1) **A method of applying and tracking damage within the numerical method.**

   This requires a damage parameter to be introduced into the continuum; this can be done either globally (i.e. crack data can be held separately from the particle data) or locally (i.e. as a function of the SPH particles themselves) within the SPH code. In this research a local approach is implemented, i.e. on a particle-particle level; the concept of an effective area on which the stress acts is defined between pairs of SPH particles and is damaged accordingly (for justification of the decision to use this approach please refer to Section 2.2.3 in the literature review). The theory behind this approach is identified and discussed within this chapter.

2) **Initiation of damage at a location within a continuum.**

   A method for identifying when a location within a continuum has failed is required to allow a crack to begin to form. For this research the Cochran-Banner spall model is adapted to represent the initiation of damage. This model initiates damage when direct stress between particle pairs reaches a material defined level. The methodology of implementing the selected criterion is outlined in Section 8.2.

3) **Growth of damage.**

   Given that damage has initiated at a location within a continuum, the continuum's strength at that location is reduced via a damage growth model; when damage has reached a critical value a discontinuity has formed. The model used in this work was based on the 1D Cochran-Banner spall model. This research explored several 3D extensions of this model coupled to the fracture

model; these models are detailed in the next chapter (Chapter 8) and the results that were obtained from testing the fracture model with each damage growth model are analysed and reviewed.

4) Crack propagation.

Once the level of damage has reached the critical level for macroscopic fracture at particular locations in a continuum (i.e. damage has become relevant on the length scale of the simulation model), locations that are coincident with one another must coalesce in some form to produce a crack. A failed location may also produce or lead to more failures in a given direction so that the failure can propagate through the mesh. Crack propagation is handled in this research by the selection of an appropriate initiation criterion, such that damage will develop in regions of tension within a simulation model. Damage orientation is not defined or tracked, although the damage direction is perpendicular to the particle-particle vector; damage naturally propagates through the material as the criterion is met via the specific loading conditions of the problem under test.

5) Crack branching and crack joining, leading to fragmentation.

Once damage is present in a material, the local failure might be such that multiple crack directions have been formed, this results in the crack bifurcating and continuing along separate paths. It is also possible for two individual cracks to become coincident. The multiple bifurcations and connections occurring within a material will inevitably result in fragmentation. These phenomena are also not specifically treated in the damage description, but are left to be observed naturally as a result of multiple bond failures around a particle or set of particles. It should be noted that this approach places a limit on the minimum numerical resolution required, i.e. a suitably fine particle mesh is required to capture the damage at the correct length scale.

This chapter describes the fracture model that has been developed for this research. The novelty in this work arises from the coupling of an existing concept in the SPH method and the classical understanding of continuum damage mechanics (Kachanov 1958, Lemaitre 1985). The justification for using a particle-particle interaction approach to the

damage description was provided in conclusion to the literature review in Section 2.2.3 and so it is not repeated here. This chapter focuses on the description of the novel approach to applying the damage within the SPH method, i.e. point 1 in the above list. Initiation of damage (point 2) and the various different methods for growing the damage parameter over time (points 3-5) along with the results that were observed with these models are outlined in the next chapter.

7.1 Classical Continuum Damage Mechanics

A standard concept in continuum damage mechanics is that a damage variable, D represents an effective surface density of microscopic cracks or voids within a continuum. This leads to the concept of an effective stress, first introduced by Kachanov (1958) and has since been used by many authors including Lemaitre (1985).

If $\mathbf{F}$ is the load acting on the overall section area S of a volume element, the traction vector is given by $\mathbf{T} = \mathbf{F}/S$ which leads to the Cauchy stress tensor $\boldsymbol{\sigma}$ ($\mathbf{T} = \boldsymbol{\sigma} \cdot \mathbf{n}$, where $\mathbf{n}$ is the normal). In order to account for the presence of micro-cracks and cavities in the material an effective area, $\tilde{S}$ is defined with respect to the total area of micro-cracks and voids, S_D as follows:

$$\tilde{S} < S - S_D = S\left(1 - D\right).$$

This concept of effective area is visualised in Figure 7.1-1 which provides a diagrammatic representation of a damaged volume element within a damaged material.
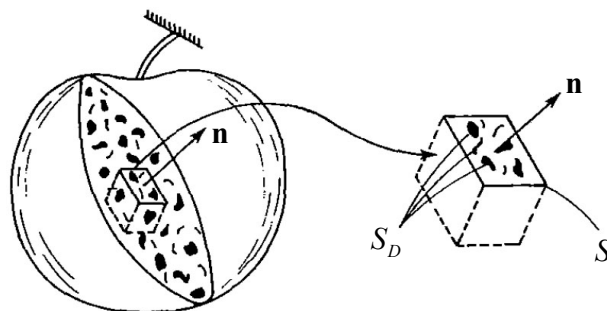
Figure 7.1-1: Diagrammatic representation of a damaged volume element within a damaged material (Lemaitre 1985).

In a finite element code it is not possible to augment the individual element areas to account for micro-crack and void growth and therefore the damage variable is used to scale the stress which allows a numerical representation of the damage to be created. This scaled stress is referred to as the 'effective stress' and replaces the element stress tensor in the time integration loop. Using the terminology defined above the effective stress is defined as follows:

$$\tilde{\mathbf{T}} = \frac{\mathbf{F}}{\tilde{S}} = \frac{\mathbf{T}}{1-D},$$

where $\tilde{\mathbf{T}}$ is the effective traction vector. D is a scalar and so the effective stress tensor may be written as $\tilde{\boldsymbol{\sigma}}$ via $\tilde{\mathbf{T}} = \tilde{\boldsymbol{\sigma}} \cdot \mathbf{n}$, where $\tilde{\boldsymbol{\sigma}}$ is given by:

$$\tilde{\boldsymbol{\sigma}} = \frac{\boldsymbol{\sigma}}{1 - D}$$

In the case of isotropic damage, two examples of the use of effective stress in finite element codes are given for elastic and plastic strain respectively as follows (Lemaitre 1985):

$$\varepsilon_e = \frac{\tilde{\sigma}}{E} = \frac{\sigma}{(1-D)E} \ , \ \varepsilon_p = \left(\frac{\tilde{\sigma}}{K}\right)^M = \left(\frac{\sigma}{(1-D)K}\right)^M,$$

where E is the Young's Modulus and K and M are material coefficients.

In the SPH method the concept of effective stress is translated to the individual particle stress in the same fashion, i.e. damage is calculated on a particle level. The treatment of a fully damaged particle, i.e. once $\tilde{\boldsymbol{\sigma}}$ has reached a critical level, is not trivial in the SPH method, since it often leads to the splitting of the particle. This requires some book-keeping to ensure that mass and momentum are conserved for the two new particles and also requires augmentation of the SPH neighbour lists and contact descriptions.

The following two sections explain how an existing concept in SPH can be coupled to the fundamentals of continuum damage mechanics without the need for an effective stress tensor and thus the complex treatment of split particles.

7.2   Swegle's Interaction Area

Swegle (2000) introduced the concept of area vectors within the SPH method for the purpose of discussing the tensile instability inherent in any Eulerian SPH description (See Section 3.3.1 for a detailed explanation of the problem). He noted that the fundamental definition of the stress tensor shows that a force exerted on a surface due to stress is given by

$$\mathbf{F} = \boldsymbol{\sigma} \cdot \mathbf{A} .$$

He then showed that the SPH momentum equation could be rewritten in terms of an interaction area in accordance with the above equation.

In MCM the SPH momentum equation is given as follows:

$$a_i = \frac{dv_i}{dt} = \sum_j m_j \xi_{ij} \nabla_i W_{ij} ,$$

where

$$\xi_{ij} = \left[\frac{(\boldsymbol{\sigma}_i - \mathbf{q}_i)}{\rho_i^2} + \frac{(\boldsymbol{\sigma}_j - \mathbf{q}_j)}{\rho_j^2} \right] .$$

From Newton's Second Law:

$$F_i = m_i a_i ,$$

and may therefore rewrite the SPH momentum equation in the following form:

$$F_i = m_i a_i = \sum_j m_i m_j \xi_{ij} \nabla_i W_{ij} .$$

By rearranging ξ_{ij} in the above equation it is possible to group the terms such that the expression yields $\mathbf{F} = \boldsymbol{\sigma} \cdot \mathbf{A}$, i.e.

if

$$\xi_{ij} = \frac{1}{\rho_i \rho_j} \left[\left(\boldsymbol{\sigma}_i - \mathbf{q}_i \right) \frac{\rho_j}{\rho_i} + \left(\boldsymbol{\sigma}_j - \mathbf{q}_j \right) \frac{\rho_i}{\rho_j} \right] = \frac{1}{\rho_i \rho_j} \Psi_{ij} ,$$

then:

$$F_i = \sum_j \frac{m_i m_j}{\rho_i \rho_j} \Psi_{ij} \nabla_i W_{ij} = \sum_j \Psi_{ij} A_{ij} .$$

Where A_{ij} is the area vector which has a direction normal to the surface and a magnitude equal to the area of the surface and has the following form:

$$A_{ij} = V_i V_j \nabla_i W_{ij} ,$$

where V_i and V_j are the volumes of the i and j particles respectively. A diagrammatic representation of A_{ij} is shown in Figure 7.2-1 below.
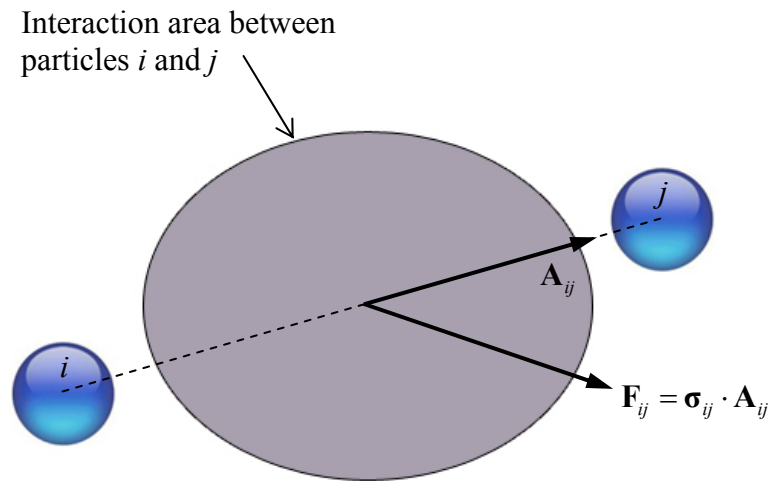
Figure 7.2-1: Area vector for particle-particle interactions.

Thus, the gradient of the kernel function can be thought of as defining the area on which the particle stresses act to produce a force between the particles.

7.3   Particle-Particle Interaction Area Fracture Model

The concept of effective area outlined in the previous section provides a unique method of measuring the surface area within a continuum which does not have a direct equivalent in finite element codes. The effective area can therefore be applied to the principles of continuum damage in a novel manner.

Damage can be evolved as an inter-particle parameter, D_{ij} and acts to reduce the interaction area, A_{ij} (For explanation of damage growth models used in this research see Chapter 8). Using this concept the SPH momentum equation (defined as a force in the previous section) may be rewritten in the following form:

$$F_i = \sum_j \Psi_{ij} A_{ij} \left(1 - D_{ij}\right),$$

thus the Particle-Particle Interaction Fracture Model is defined.

Critical damage assumes the material to have failed and the interaction area is set to zero causing the particles to cease being neighbours. To prevent unphysical interactions between fully damaged particles, all interpolated values are also set to zero via $\nabla_i W_{ij} = 0$, once critical damage has been reached. In the SPH code (MCM) this ensures that two particles either side of a fully damaged bond cannot interact with each other. This is possible in the Total Lagrangian formulation of the code since the kernel is stored in memory.

This approach does not require the modification of values which are held at a particle since damage is applied at the inter-particle level. This means that no assumptions are required for the treatment of the mass and momentum represented by a particle and the need for a complicated routine to spit a particle or some other work to represent the formation of a new free surface can also be avoided.

The main limitation to this approach is that it requires the use of the Total-Lagrangian formulation since it is necessary to be able to track inter-particle bonds (i.e. a fixed neighbourhood is required), which is not possible with the Eulerian SPH method. This has meant that it was not possible to test the explosively driven experiments that have been modelled as a part of this research with the developed fracture model since the Total Lagrangian formulation cannot handle the large deformations associated with the explosive materials. It is also worth noting that the mesh density of particles is required to be consistent with the scale of the damage; otherwise the direction of the damage growth (crack propagation) will be partially defined by the shape of the mesh. Continuum damage mechanics deals with the effects of damage when the physical phenomena are occurring below the resolution length of the model. The crack appears once the new free surface has meaning on the mesh resolution that is being used.

7.4   Implementation of the Fracture Model into MCM

This section describes the implementation into the Cranfield SPH code (MCM) of the particle-particle interaction area fracture model discussed in the previous section. As discussed previously, the fracture model requires consistent inter-particle bonds between particles and their neighbours; the Total Lagrangian SPH formulation is therefore utilised. All calculations are performed in the reference state and so the particle neighbourhoods remain fixed throughout the computation (see Section 3.4 for further explanation of TL SPH).

First the inter-particle damage parameter is defined in the MCM database and allocated the correct amount of memory in the initialisation routine for all the inter-particle bonds in the computation. A failure flag is also defined for each bond to facilitate in activating damage growth on a particle-pair. In this method each bond is actually stored twice, since for every particle pair the bond can exist under two circumstances, i.e. once when the first particle is the current particle and the second is a neighbour particle and again when the second particle is the current particle and the first is a neighbour particle. A decision was made that no treatment of these multiple bonds is required since the damage initiation criterion is based on direct stress between the particles which is the

same regardless of which direction the calculation is made (see Section 8.2 for a full explanation of the damage initiation criterion used for this research). In this way both bonds will be flagged for failure in the same time-step, consequently grow damage at the same rate and fail at the same time.

It is understood that this double calculation is inefficient and will result in longer simulation times; however for this work it is assessed to be an acceptable consequence of inter-particle failure modelling. A search could be performed for identical bonds in the initialisation and then the damage parameter calculation routine could be tidied up by only calculating it for the first occurrence of each double bond. It could then be applied to both bonds in the momentum equation update. Such a modification is not complicated, but requires significant book-keeping to be applied rigorously. Since it is the purpose of this research to develop a fracture model and demonstrate its validity; optimisation of the code is left for further work.

The damage parameter is calculated via a series of routines that are called before the constitutive model is updated in the time integration loop. This is because damage is calculated using the data from the previous time-step and any parameters that are required for the damage growth model (see the next chapter), must be based on data from the previous step and not the current. Table 7.4-1 identifies the time integration algorithm for MCM and highlights the position of the damage routine within the time integration loop (in red). Table 7.4-2 highlights the layout of the damage routine.

The damage parameter is applied to the SPH momentum equation via the Swegle Interaction Area (SIA) discussed in the previous two sections. The Total Lagrangian momentum equation update routine is modified to calculate the SIA which is then multiplied by $\left(1 - D_{ij}\right)$ to apply the damage. The momentum equation update is rearranged in terms of a force as per the equation given in the previous section, this new form contains the damaged SIA. Finally the acceleration of the current particle is calculated by dividing through by the mass of the current particle.

This implementation was verified by running two test simulations in parallel with the standard form of the Total Lagrangian momentum routine and the new momentum

routine with $D_{ij} = 0$. Both simulations provided identical results. The above implementation of the fracture model was demonstrated and validated by testing several different configurations of the damage parameter growth routines, the results and discussion of this process are provided in the next chapter.

Start of time-step *n*   *(Know $x^n, v^{n-\frac{1}{2}}, t^n, \Delta t^{n-\frac{1}{2}}, \sigma^{n-1}, \rho^{n-1}, P^{n-1}, E^{n-1}$)*

    Store $\Delta t^{n-\frac{1}{2}}$

    Calculate strain rate

    Update inter-particle damage parameter   (See Table 7.4-2)

    ⎡ Update density

      Update strength model   Constitutive

      Calculate speed of sound   Routine

    ⎣ Calculate bulk viscosity terms

    Calculate new critical time step

    Update time

    Solve momentum equation (and apply damage)

    Update pressure and internal energy

    Check t^n to determine if plot files should
    be written or for termination

    Update velocity

    Update position

End of time-step *n*   *(Know $x^{n+1}, v^{n+\frac{1}{2}}, t^{n+1}, \Delta t^{n+\frac{1}{2}}, \sigma^n, \rho^n, P^n, E^n$)*

Table 7.4-1: Central Difference Time Integration Algorithm in MCM.

Loop over all particles

    Loop over all neighbour particles

        Calculate direct stress between particle pair

        Check damage initiation criterion (flag failed inter-particle bonds for damage growth)   (See Section 8.2)

        Calculate the damage increment for the current time-step using a damage growth model   (See Sections 8.3 and 8.4)

        Sum the damage from the previous step and the increment of damage to achieve the new damage parameter

        If maximum damage has been reached (i.e. $D_{ij} = 1$) then $\nabla_i W_{ij} = 0$

Table 7.4-2: Routine to update inter-particle damage parameter.

7.5   Effect of the Fracture Model on the Basic Properties of SPH

A stability analysis performed by Reveles (2007) is summarised in Section 3.4.1. The conclusion to this analysis is that the Total Lagrangian SPH particle equations exhibit the same stability as the continuum equations when a perturbation is introduced, i.e. in most cases the perturbation decreases with time and also that it is not subject to the tensile instability associated with the Eulerian SPH particle equations (see Section 3.3.1 for more information). The stability analysis begins by representing the SPH momentum equation in the following form:

$$m_i \left(\frac{d^2 \tilde{u}_i}{dt^2} \right) = -\sum_j \frac{m_j}{\rho_{0j}} \tilde{\mathbf{P}}_j \nabla_j W_{ij} \ .$$

The novel fracture model identified in Section 7.3 is formulated from the SPH momentum equation represented as a force via Newton's second law and the addition of a scalar damage parameter. The fracture model does not alter the strain rate or any other field variable. The fracture model form can therefore be applied to the above equation to

obtain an assessment of the effect of the fracture model on the stability of the Total Lagrangian SPH formulation. Thus:

$$m_i \left(\frac{d^2 \tilde{u}_i}{dt^2} \right) = - \sum_j \frac{m_j}{\rho_{0j}} \tilde{\mathbf{P}}_j \left(1 - D_{ij} \right) \nabla_j W_{ij} \ .$$

Following the same derivation procedure in Reveles (2007) on the above equation, the real part of the complex perturbation frequency is represented in the following form:

$$\omega_{\Re}^2 = \frac{1}{m_i} \sum_j \frac{m_j}{\rho_{0j}} \left(1 - D_{ij} \right) \nabla_j W_{ij} \overline{\mathbf{C}} \sum_k \cos \left(\kappa \left(X_k - X_i \right) \right) \nabla_k W_{jk} \ .$$

For a description of the terms in the above equation refer to Section 3.4.1. It can be concluded that the addition of the damage term does not alter the outcome of the stability analysis, since $\left(1 - D_{ij} \right)$ is a scalar with a value between 0 and 1. The sign of the real part of the complex perturbation frequency remains unchanged and therefore the original conclusions to the stability analysis remain valid. Clearly when $D_{ij} = 1$, the potential for instability is present since this replicates the effect of the spurious singular mode identified when $\kappa = \pi / 2 \left(X_k - X_i \right)$ (see Section 3.4.1 for more information). However this represents a condition relating to complete material failure where all interpolated values are set to zero and therefore this does not alter the conditions for Total Lagrangian SPH stability.

It is also worth noting that since $\left(1 - D_{ij} \right)$ is applied to a particle pair, i.e. the damage parameter calculated from i to j is the same as the damage parameter calculated from j to i. Therefore damage is applied evenly across the bond and so local conservation of momentum is maintained.

8 Demonstration and Validation of the Particle-Particle Interaction Fracture Model

The objective of the work presented in this chapter was to assess, demonstrate and validate the Particle-Particle Interaction Fracture model outlined in the previous chapter. A damage growth model is used to compute the damage parameter in the fracture model over time. It should be noted that whilst significant development of the damage growth model was required for this research (mainly to update the selected model(s) for inter-particle damage), the growth model exists to test the Swegle Interaction Area approach for applying damage. Two existing growth models were selected for use with this research, these were:

- A simple linear damage growth model that grows damage in a linear fashion over a number of model time-steps. This concept is simple and was used primarily to identify that the description of damage outlined in the previous chapter was a valid method for describing damage in metals at high strain rate.

- A modified Cochran-Banner damage growth model that incorporates the original 1D model developed by Cochran and Banner (1977) and has been updated for the 3D solver with modifications to utilise the Swegle Interaction Area. This method was explored in detail and represents the bulk of the work conducted into fracture modelling (and damage growth) for this research.

This chapter is laid out as follows; Section 8.1 describes the test simulation that was used throughout the fracture model development. Section 8.2 identifies the damage criterion that has been chosen to flag the start of damage growth within a particle pair. Section 8.3 outlines the linear damage growth model with simple results that was used for concept demonstration. Section 8.4 explains the modified Cochran-Banner damage growth model that forms the core of this research and identifies the successful results that were achieved in 1D. This model was also partially extended to 2 and 3 dimensions and the outcome of this exercise is also discussed in this section. Finally Section 8.5 provides the conclusions to the investigations discussed in this Chapter.

8.1   A Simple Problem for Fracture Model Test and Development

In order to demonstrate the capability of the fracture model outlined in the previous Chapter, a simple test problem was required; at the time of development it was not possible to simulate a Mock-Holt experiment (see Section 2.1.1.2 for a description) in the Total Lagrangian SPH framework that is utilised for this approach to modelling damage[8]. The Mock-Holt experiment is also not ideal for development of the fracture model since it has a complicated failure state which is difficult to quantify for comparison to experimental data on a level that can be easily interpreted as correct representation of the damage. Fundamentally the Mock-Holt experiment requires a large mesh to accurately describe the material, which ultimately leads to simulations with high computation times; it also is not suited to straightforward discrimination between different fracture/damage growth model options and /or tests.

Ideally a suitable test problem is an experiment which is simple to represent in the model, has clear and concise results to indicate damage/failure and can be represented by a mesh that is not computationally expensive. Also a simple model facilitates rapid re-simulation with different damage growth model parameters, or updates. One such experiment which meets these criteria is a simple plate impact test.

Simulation results throughout the fracture model development were therefore compared to experimental data for a simple 1D copper plate impact test which was obtained from Panov (2006). In the experiment a 10 mm thick OFHC copper sample was impacted by an OFHC copper plate 5 mm in thickness at a velocity of 304 m/s. Both target and flyer were machined into circular plates with diameters of 70 mm and 50 mm respectively. To record longitudinal stress data, a Manganin stress gauge was supported on the rear surface of the target with a 12 mm block of Poly(methyl methacrylate) (PMMA). The geometry of the flyer and the target were chosen such that the reflected stress wave off the free ends would interact in the centre of the target plate. Furthermore the experiment was precisely controlled to yield a uniaxial state of deformation inside the target plate. Figure 8.1-1 shows a diagrammatic representation of the experiment.

---

[8] See end of Section 5.3 and Section 6.2.5 for explanations.
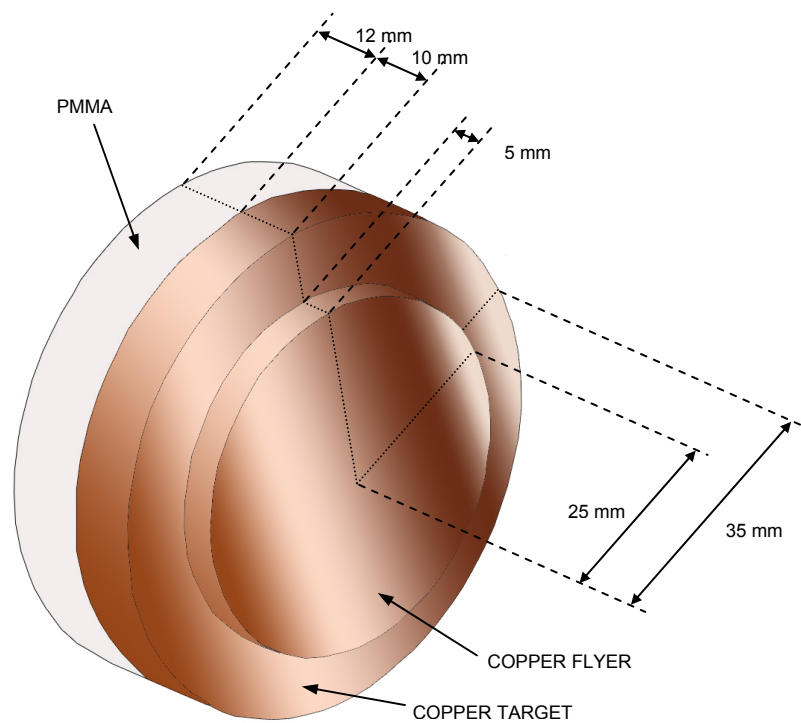
Figure 8.1-1: Diagrammatic Representation of the Copper Plate Impact Experiment.

The results of the experiment are displayed graphically via a plot of longitudinal stress with time in Figure 8.1-2. It is observed that the release waves from the free surfaces propagate into the material and cause high-tensile stress in the centre of the target plate. When this tension exceeds the spall strength of the material, the material fails and micro-voids begin to form and coalesce in this region, ultimately leading to the generation of a new free surface. This free surface reduces the tensile stress in the material to zero and results in the reflection of the remainder of the release wave as a compressive wave.

It is worth noting that this research is only concerned with the initial pulse and reload signal for this experiment (i.e. 0-5µs in Figure 8.1-2), since the behaviour after the reload signal is dependent on the specific set-up of the experiment for which specific data are not available. Assessments of the fracture modelling are therefore made on the consistency of the modelling predictions and the level of correlation within region of interest only.
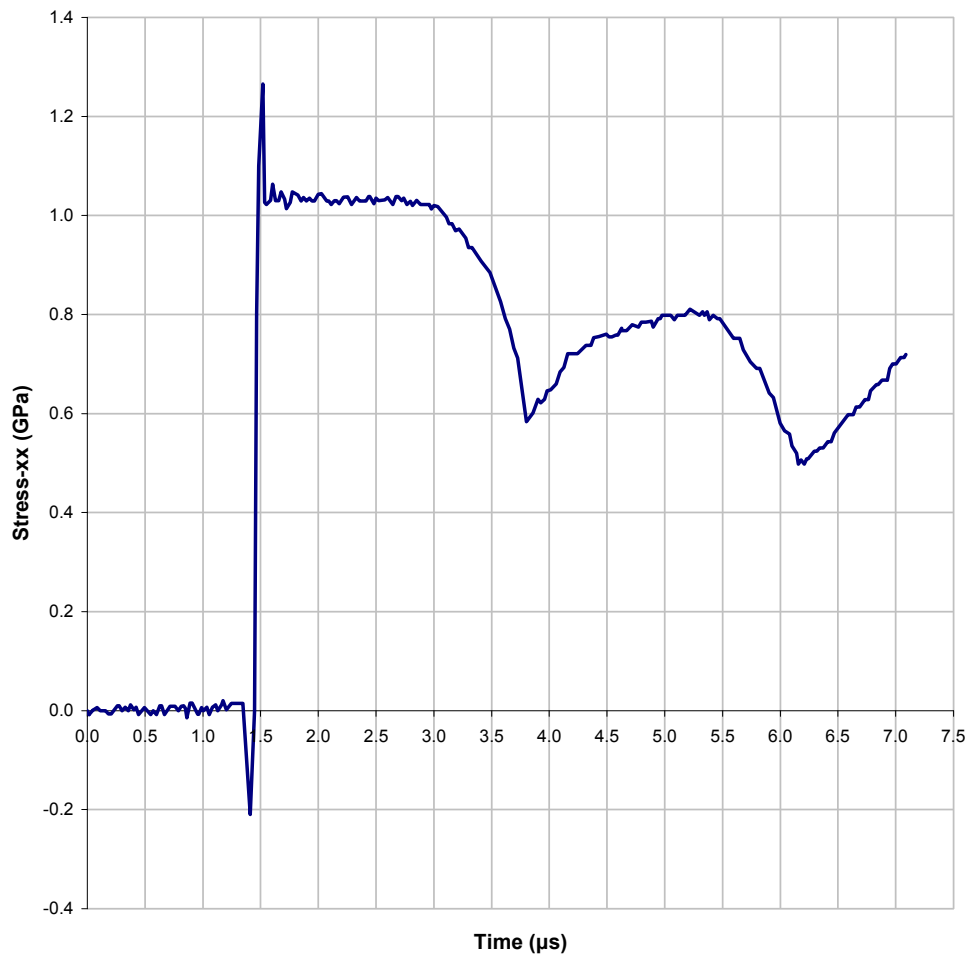
Figure 8.1-2: Experimental Results for the Copper Plate Impact Experiment
(longitudinal stress against time at the rear of the target plate) (Panov 2006).

A 1D simulation of the experiment was generated initially which could be extended to the other dimensions as required. The simulation made use of an Isotropic-Elastic-Plastic-Hydrodynamic material model with a Gruneisen equation of state (see Section 4.8 for more information) for the OFHC Copper and the PMMA. Parameters for both materials can be found in Table 8.1-1 and Table 8.1-2 respectively and a diagrammatic representation of the model is given in Figure 8.1-3. The model consisted of 540 particles; 100 in the flyer, 200 in the target and 240 in the PMMA. This was considered to be a sufficient number to be able to display the effects of the damage, whilst maintain a short run-time. A gap of 0.2 mm was placed between the flyer and target (a gap of 17.5 μm was also placed between the target and the PMMA to facilitate in minimising noise from the contact algorithm). The SPH smoothing length was taken to be greater than the initial inter-particle spacing by a factor of 1.3. The spall strength for OFHC

copper used to activate the damage growth models outlined in the following sections (see Section 8.2) was taken from Steinberg (1996) as 1.2 GPa.
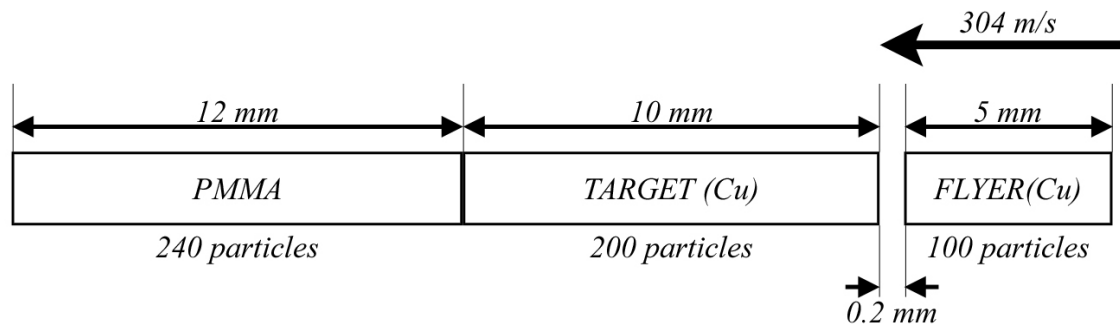
Figure 8.1-3: Diagrammatic Representation of the Damage Test Problem in 1D.

Material Parameters (Panov, 2006)						
Density (kg/m³)		Shear Modulus, G_0 (GPa)		Yield Stress, σ_y (MPa)		Plastic Modulus, E_p (MPa)
8924		48.4		150		500
Equation of State Parameters (Panov, 2006)						
C (m/s)	S_1	S_2	S_3	γ_0	a	E_0	V_0
3940	1.4985	0.0	0.0	2.02	0.47	0.0	1.0

Table 8.1-1: Material parameters for OFHC Copper.

Material Parameters (Panov, 2006)						
Density (kg/m³)		Shear Modulus, G_0 (GPa)		Yield Stress, σ_y (MPa)		Plastic Modulus, E_p (MPa)
1180		2.32		350		300
Equation of State Parameters (Panov, 2006)						
C (m/s)	S_1	S_2	S_3	γ_0	a	E_0	V_0
2180	2.0880	-1.1240	0.0	0.85	0.0	0.0	1.0

Table 8.1-2: Material parameters for PMMA.

Initially the model was run with the Total Lagrangian SPH solver and no fracture model present to establish the tensile behaviour of the material and for clear comparison to the first part of the experimental data (i.e. the material response up to the reload signal). Figure 8.1-4 shows a comparison of the 1D simulation to the experiment (stress data was plotted for the 3rd particle into the PMMA to avoid a skew in the results due to any noise from the material contact algorithm). It is evident that the simulation slightly underestimates the peak stress, but the shape of signal is represented well. This simulation forms a good baseline from which the effectiveness of any fracture model will be immediately apparent.
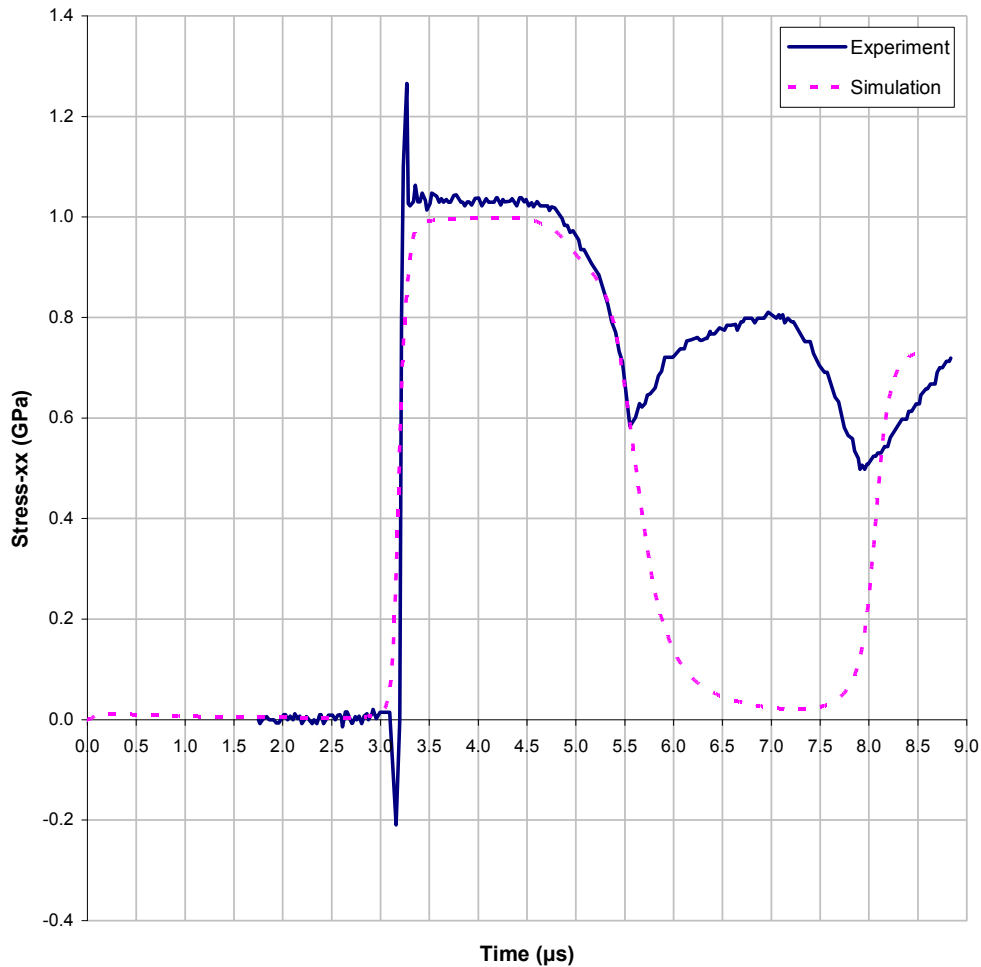
Figure 8.1-4: Comparison of 1D simulation (with no fracture model present) to experiment for the Copper Plate Impact Experiment (longitudinal stress against time at the rear of the target plate).

It is worth noting that the experimental data has been time aligned to the simulation, this is because it is unclear at what point the recording of experimental data was started and also what gap was placed between target and flyer at the start of the experiment. Therefore there will invariably be a difference in the timings which needs to be corrected if a comparison is to be made.

It is also relevant that due to the usage of the Total Lagrangian SPH solver, the model handles tension without a problem, i.e. it is not subject to the tensile instability identified earlier in this thesis (See Section 3.3.1 for more information). The model is therefore validated as a suitable test problem for the Particle-Particle (Swegle Interaction Area) fracture model.

8.2   Damage Growth Initiation Criterion

The interaction area fracture model that is identified and discussed in Section 7.3 is initialised when a criterion is met in the simulation. Since the fracture model applies damage between particle pairs (or in particle bonds), the initialisation criterion used in this research is satisfied when a tracked inter-particle parameter reaches a specific (material defined) level. The criterion that is used in this model is a basic spall criterion (Lin 2004, Hallquist 1998); this is essentially a stress cut-off, which suggests that when material is subjected to stresses greater than the spall stress, micro-cracks and voids will begin to form in the material (which are subsequently accounted for computationally via a damage growth model, see the following descriptions for discussion of the damage growth models that have been used for this research).

For this research damage occurs when critical stress is reached between a pair of particles i and j in tension (Hydrostatic Tension Spall). In order to check for this criterion, it is necessary to track the direct stress (traction) between the particle pair.

Since SPH data are stored at the individual particles; the stress in the $(i \rightarrow j)$ direction, $\mathbf{T}_i$ and the $(i \leftarrow j)$ direction, $\mathbf{T}_j$ must first be obtained. From these stresses a composite value can be obtained for the traction between the particle pair ($\mathbf{T}_{ij}$). $\mathbf{T}_i$ and $\mathbf{T}_j$ are therefore the first component of the respective stress tensor in the direction of $(i \rightarrow j)$ and $(i \leftarrow j)$ respectively. It is possible to obtain these values by rotating the full stress tensor to the new co-ordinate system for both particles; however this is computationally expensive and will yield the full stress tensor where only the component of direct stress along the inter-particle direction is required. Therefore the following method is used to obtain $\mathbf{T}_i$ and $\mathbf{T}_j$ directly.

First the magnitude of the vector aligned in the $(i \rightarrow j)$ and $(i \leftarrow j)$ direction is obtained from:

$$M = \sqrt{\left(x_j - x_i\right)^2 + \left(y_j - y_i\right)^2 + \left(z_j - z_i\right)^2} \ .$$

Using this magnitude, the unit vectors which project onto the new direction can be calculated as follows:

$$
\mathbf{n}_i = \begin{bmatrix} \mathbf{n}_{i_x} \\ \mathbf{n}_{i_y} \\ \mathbf{n}_{i_z} \end{bmatrix} = \begin{bmatrix} \dfrac{(x_j - x_i)}{M} \\ \dfrac{(y_j - y_i)}{M} \\ \dfrac{(z_j - z_i)}{M} \end{bmatrix} \; ; \; \mathbf{n}_j = \begin{bmatrix} \mathbf{n}_{j_x} \\ \mathbf{n}_{j_y} \\ \mathbf{n}_{j_z} \end{bmatrix} = \begin{bmatrix} \dfrac{(x_i - x_j)}{M} \\ \dfrac{(y_i - y_j)}{M} \\ \dfrac{(z_i - z_j)}{M} \end{bmatrix} .
$$

Now the traction vector is simply the dot product of the unit vector and the stress tensor (Malvern 1969) as follows:

$$
\mathbf{T}_i = \mathbf{n}_i \cdot \boldsymbol{\sigma}_i = \begin{bmatrix} \mathbf{n}_{i_x} \\ \mathbf{n}_{i_y} \\ \mathbf{n}_{i_z} \end{bmatrix} \cdot \begin{bmatrix} \sigma_{i_{11}} & \sigma_{i_{12}} & \sigma_{i_{13}} \\ \sigma_{i_{21}} & \sigma_{i_{22}} & \sigma_{i_{23}} \\ \sigma_{i_{31}} & \sigma_{i_{32}} & \sigma_{i_{33}} \end{bmatrix} = \begin{bmatrix} \left(\mathbf{n}_{i_x} \sigma_{i_{11}} + \mathbf{n}_{i_y} \sigma_{i_{12}} + \mathbf{n}_{i_z} \sigma_{i_{13}} \right) \\ \left(\mathbf{n}_{i_x} \sigma_{i_{21}} + \mathbf{n}_{i_y} \sigma_{i_{22}} + \mathbf{n}_{i_z} \sigma_{i_{23}} \right) \\ \left(\mathbf{n}_{i_x} \sigma_{i_{31}} + \mathbf{n}_{i_y} \sigma_{i_{32}} + \mathbf{n}_{i_z} \sigma_{i_{33}} \right) \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{i_x} \\ \mathbf{T}_{i_y} \\ \mathbf{T}_{i_z} \end{bmatrix} .
$$

The magnitude of the traction vector is simply:

$$
\left| \mathbf{T}_i \right| = \sqrt{ \mathbf{T}_{i_x}^{\,2} + \mathbf{T}_{i_y}^{\,2} + \mathbf{T}_{i_z}^{\,2} } \; .
$$

The sign of $\mathbf{T}_i$ (to indicate compression or tension) can be determined from the sign of $\mathbf{T}_i \cdot \mathbf{n}_i$; a negative value indicates compression and a positive value indicates tension ($\mathbf{T}_j$ is obtained by replacing the suffix i with j in the traction equations above).

The traction between the particle pair, $\mathbf{T}_{ij}$ is then assumed to be the average value of the traction in the individual particles:

$$\mathbf{T}_{ij} = \frac{\left(\mathbf{T}_i + \mathbf{T}_j\right)}{2}.$$

This is a common assumption in the SPH method; it is used for ensuring symmetric behaviour between particles (Monaghan 2005).

If $\mathbf{T}_{ij}$ exceeds a user defined spall criterion, Σ (which is a material specific parameter) damage will begin to grow for that particle pair (according to a specific damage growth model identified in the following sections). Thus the damage parameter is subject to the following limitation:

$$D_{ij} = \begin{cases} 0 & , \quad \text{if } \mathbf{T}_{ij} < \Sigma \\ 0 \leq D_{ij} \leq 1 & , \quad \text{if } \mathbf{T}_{ij} \geq \Sigma \end{cases}.$$

8.3   Linear Damage Growth Model

The first step in the fracture model development was to determine if the concept of the Swegle Interaction Area coupled with inter-particle damage outlined in Chapter 7 was a feasible approach to modelling fracture at high strain rate. It was important to assess the method's potential via a simple implementation; therefore a linear damage growth model was implemented purely as an aid to concept demonstration. The inter-particle damage parameter is incremented linearly over an arbitrary number of discrete time-steps, allowing the user to define the time to failure for the material.

$$D_{ij_n} = D_{ij_{(n-1)}} + \Delta D$$

Total failure of the material is reached when $D_{ij} = D_{crit}$, where D_{crit} represents critical damage (also a user defined variable, but for this research it was considered to be equal to 1.0). Therefore the total damage of the material at time, t is given as follows:

$$D_{ij} = \text{MIN}\left[D_{ij}, D_{crit}\right]$$

In the initial stages of the research, it was unclear if the stress in the particle pairs would need to be set to zero upon absolute failure of the inter-particle bond, i.e. analogous to other existing damage growth models highlighted in the literature survey (see Section 2.2). It was hypothesised that setting the stress to zero would be unnecessary since the bond would be broken (via $\nabla_i W_{ij} = 0$) and the particles would no longer interact with each other. In order to test this, a user defined flag was added to the model (in the input file) that controlled if the zero stress upon failure option was active in the analysis. Initial tests were conducted with this flag both active and inactive to assess its validity.

8.3.1   1D Modelling Results

This portion of the research was conducted in the early phases of the fracture model development and the therefore the test simulation that was used with the Linear Damage

Growth Model was a simplified version of the scenario described in Section 8.1; essentially the model was the same test problem. However it utilised a fluid material model to describe the copper plates in order to minimise the complexity of the analysis and allow the effect of the fracture model to show clearly in the results. This model was not intended to be rigorously representative of a 'real-world' situation, however it can be considered to be a reasonable approximation, since the behaviour is dominated by the wave propagation in the region of interest via the equation of state. The intention was to demonstrate the concept successfully before taking it on to more complicated analyses with detailed material models and simulations. For this reason the demonstration was also conducted with the 1-dimensional SPH solver and would be extended to the other dimensions with the more complicated damage growth and simulation models after the approach was tested (see the proceeding sections for more information).

The results contained in this section were presented at an International Conference on the 17-19th June 2008 at the NTNU in Trondheim, Norway. The extended abstract and the presentation slides have been included in Appendix A.

The SIA modification was used in the SPH solver (see Section 7.3), with the damage initiation criterion outlined in Section 8.2 set to check every particle pair in the model. Once the damage initiation criterion had been met the damage was incremented from 0 to 1 over a user defined period of time-steps. Three time-to-failure-modes were tested, i.e. damage was incremented over 20, 50 and 70 time-steps; for clarity, these are equivalent to times of approximately 0.07, 0.18 and 0.25 μs respectively.

For the concept to be valid in 1D, a set of detached particles would need to be present in the centre of the target plate, which would be visualised by large gaps forming between them. Graphically a plot of longitudinal (XX) stress at the rear surface of the target plate should display a reload signal (reflected wave off of a new free surface) similar to the response obtained experimentally in Figure 8.1-4. The plots for the 50 and 70 time-steps to failure with both the 'stress to zero' flag active and inactive are displayed in Figure 8.3-1. A plot of the 'no damage' case has also been included for comparison (the XX stress is displayed for the 3rd particle into the PMMA to avoid noise from the material contact algorithm interfering with the results).
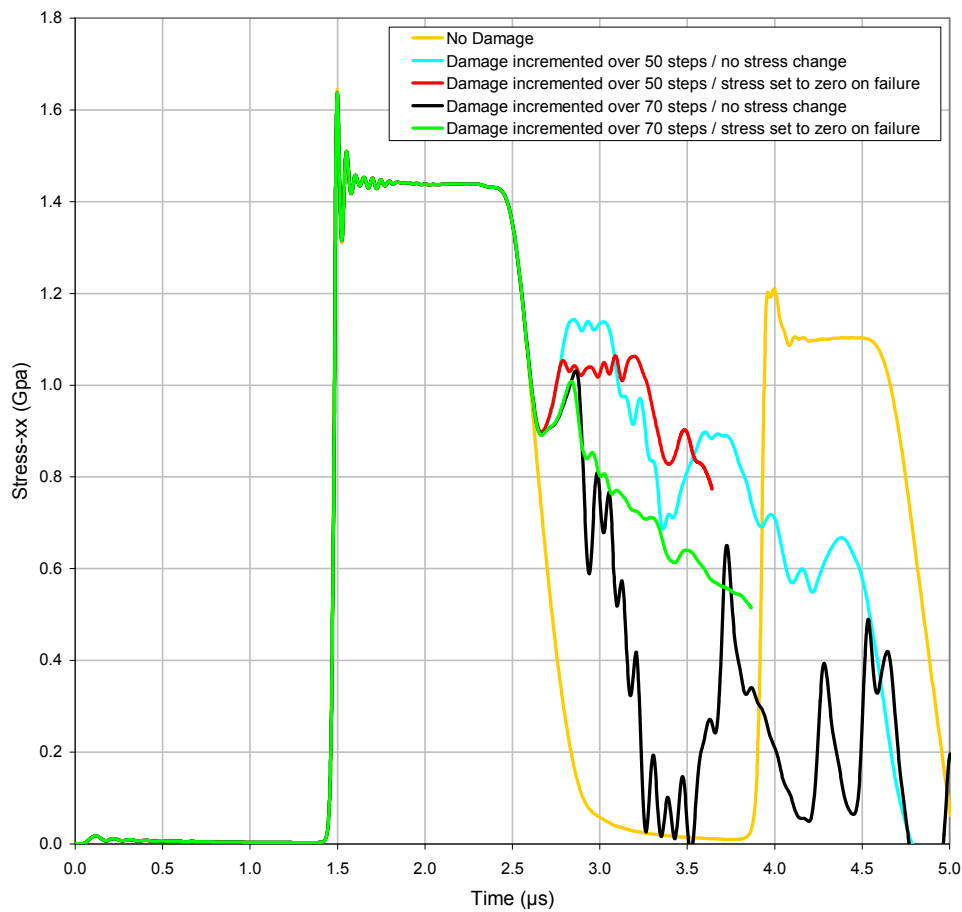
Figure 8.3-1: Results for a 1D plate impact test problem with the SIA activated and coupled with a linear damage growth model (longitudinal stress against time at the rear of the target plate). All particles are considered in the damage computation.

It is clear from the above figure that a reload signal has been generated and that when damage is present in the computations, the response is different to the 'no damage' case. It is not clear from these results if the 'stress to zero' approach is having a positive or negative effect on the damage description. After the reload signal the results become noisy and unclear; in several of the cases the simulation terminated early with errors in the computation. Despite these initial problems the outcome is positive because these results prove that the SIA damage concept does generate the expected response.

After closer analysis of the results, it was identified that damage was activated over a large portion of the target plate and was not localised to the region of tensile stress. It was decided to see if the signal could be cleaned up to demonstrate the reload signal properly. The damage growth model was limited to particles that lay in the expected

damage region, i.e. locally either side of the centre of the target plate or about the point of the initiation of tensile stress. This is a reasonable choice since it is based on the physical observations of the target plate after the experiment, which showed a spall plane had opened in this region. The 'stress to zero' flag was left activated (it was decided to analyse this effect on a more valid test problem) and plots for 20 and 50 time-steps to failure were generated. It was anticipated that since the damage region was limited to the area where it is expected to see damage form; smooth reload signals would be observed on the rear surface of the target. Figure 8.3-2 shows the results.
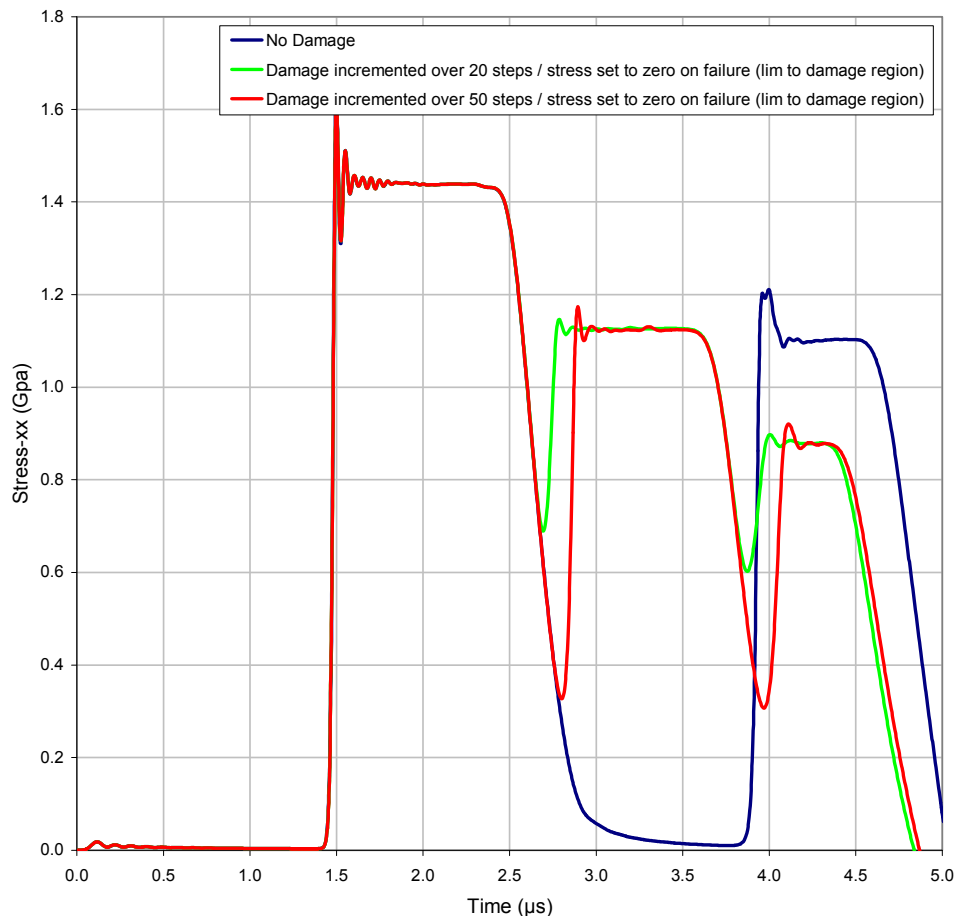
Figure 8.3-2: Results for a 1D plate impact test problem with the SIA activated and coupled with a linear damage growth model (longitudinal stress against time at the rear of the target plate). Only particles in the expected damage region are considered in the damage computation.

It is observed that the longer the time to failure (i.e. the more time-steps over which the damage is incremented), the later the reload signal is formed. These results confirmed the expected outcome and provide an initial validation of the concept of the Swegle Interaction Area coupled with inter-particle failure.

Following basic confirmation of the validity of the particle-particle (Swegle Interaction Area) fracture approach using the Linear Damage Growth Model, the next step was to implement a more physically reasonable damage growth model. A discussion of the theory associated with this implementation and the presentation of the results are outlined in the following sub-sections.

8.4   Modified Cochran-Banner Damage Growth Model

The Cochran-Banner damage growth model (Cochran, Banner 1977) is a one-dimensional model that assumes all changes in volume past the material spall strength is due to micro-crack and void growth, i.e. the total volume of a material after it has reached a material specific limit is assumed to be made up of the material and a quantity micro-cracks and voids. The amount of micro-cracks and voids is accounted for by a damage parameter via the following relationship:

$$D(x,t) = \int_0^t dV/A, \quad dV > 0.$$

A three dimensional version of this model implemented within an FE code is given by Mirkovic (2004) and is the main model that has been adapted for use with the Swegle interaction area outlined in Section 7.2. In the cited work the damage was implemented in a conventional manner, i.e. to generate an 'effective stress,' consistent with Kachanov's original approach (1958) (see Section 7.1 for more information), however the model is utilised differently for this research since it makes us of the Swegle interaction area fracture model outlined in Section 7.2. The Cochran-Banner damage growth model was selected because it has been shown to be a simple model that can effectively represent the type of damage that occurs in spall.

The current change in volume for a particle, dV may be calculated from the trace of the rate of deformation tensor, $\mathrm{tr}\,\dot{\boldsymbol{\varepsilon}}$ and the current time-step, dt. The incompressibility condition ensures that the volumetric plastic strain rate tensor is traceless (i.e. $\mathrm{tr}\,\dot{\boldsymbol{\varepsilon}}^{pl} = 0$, see Section 4.1 for more information) and therefore $\mathrm{tr}\,\dot{\boldsymbol{\varepsilon}} = \mathrm{tr}\,\dot{\boldsymbol{\varepsilon}}^{el}$, Since inter-particle parameters are of interest, the change in volume for a particle pair under consideration is calculated as the mean of the change in volume for the individual particles, thus:

$$dV_{ij} = \frac{dt}{2}\left(V_i\,\mathrm{tr}\,\dot{\boldsymbol{\varepsilon}}_i + V_j\,\mathrm{tr}\,\dot{\boldsymbol{\varepsilon}}_j\right).$$

The cross-sectional area which is reduced by the damage is taken to be the magnitude of the Swegle interaction area, and so the Cochran-Banner damage parameter is simply:

$$D_{CB} = dV_{ij}\Big/\big|A_{ij}\big|.$$

The inter-particle damage is applied via the following equation:

$$D_{ij_n} = D_{ij_{(n-1)}} + \left(D_{CB} / D_{crit} \right)^{2/3},$$

where D_{crit} represents critical damage (material parameter)[9]. Total failure of the material is reached when $D_{ij} = 1.0$. Therefore the total damage of the material at time, t is given as follows:

$$D_{ij} = \text{MIN} \left[D_{ij}, 1.0 \right].$$

8.4.1 1D Modelling Results

The Cochran-Banner damage growth model outlined above was coupled with Swegle Interaction Area (SIA) approach explained in Chapter 7. This configuration was the most successful for this research and the majority of the work on fracture modelling was conducted using this approach or similar modifications. The following sub-sections highlight the work that was first conducted in 1D to test the validity of the approach and make adjustments so that the various sensitivities of the model could be better understood before moving the model to the more complicated two and three dimensions.

8.4.1.1 Preliminary Tests

The 1D copper plate test simulation explained in Section 8.1 was used to test the Cochran-Banner damage growth model coupled with the SIA approach for applying the damage. Initially the simulation was run with the 'stress to zero' concept outlined in Section 8.2 active, however the simulation terminated early at the point where the stress was set to zero for the first particle bond. Attempts to get the code running were unsuccessful and so the 'stress to zero' concept was abandoned for the remainder of the

---

[9] The value of D_{crit} for OFHC copper used in the plate impact experiment (see Section 8.1) was taken from Cochran and Banner (1977) to be 0.007 cm.

research. This decision was considered to have a positive effect, since the approach is an unphysical representation of the conditions during failure. After opening a new free surface the stress should naturally reduce to zero after the fracture has occurred. This is captured naturally without this criterion and so it is therefore not required.

The simulation was re-run without the 'stress to zero' flag active and it ran successfully. Figure 8.4-1 shows a comparison of the longitudinal stress against time for the experimental and simulation data (simulation data is plotted for the 3rd particle in the PMMA as previously stated). A plot of the simulation data with no fracture model present has also been included to establish the tensile behaviour of the material and for comparison.

Figure 8.4-1: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 1D simulation data is shown for the Cochran-Banner damage growth model and is compared to the experimental data.

It is clear from the results shown in Figure 8.4-1 that the fracture model causes the particle bonds in the region of the expected spall plane to fail and a release wave is formed. However the exact shape of the reload signal has not been captured. The results were encouraging since this was the first simulation that was completed and unlike the tests performed with the linear damage growth model discussed previously, the Cochran-Banner damage growth model was not limited to the particles in the expected region of the damage. All particles in the simulation were subjected to the damage routines provided that the direct-stress in a particle bond had satisfied the damage initiation criterion. This is an initial indication that the approach is valid since the damage has grown in the correct region.

8.4.1.2 Additional manipulation of inter-particle bonds

After the success of the initial tests performed with the Cochran-Banner damage growth model, further tests were performed to see if the model could be made to be more responsive and better represent the reload signal in the experimental data.

The fracture model developed for this research makes use of the fixed particle neighbourhoods in the Total Lagrangian SPH formulation by tracking each particle bond as a material history parameter. Each particle bond is discrete and subjected to damage growth once the damage initiation criterion has been met. In a real solid when a connection between two parts of the material has been broken (i.e. a new free surface has opened), there is no interaction other than material contact across this boundary. In the SPH method an issue arises with the discrete bond approach since particles have overlapping domains of influence and multiple neighbours within their own neighbourhoods. This means that one failed bond does not prevent the interaction across the new boundary as other bonds may still be active. This is especially evident in 1D where all the particles are in the same plane. Unfortunately the exact representation of an inter-particle bond in a real solid is not understood and therefore the exact damage treatment of multiple bonds is unclear. It was assessed that an investigation into the effect of failing multiple bonds given that one bond has failed due to maximum damage growth was necessary to assess the effect on the material response. Tests were therefore

conducted using various multiple bond-break criteria added to the damage growth model.

The copper-plate test simulation is configured such that the inter-particle spacing, Δp is related to the SPH smoothing length, h via $h = 1.3\Delta p$. This produces a maximum of 4 neighbour particles for a particle that does not lie on or next to a boundary, i.e. 2 neighbour particles on either side. This means that every non-boundary particle has 4 bonds and thus 4 damage parameters associated with it; this is depicted in Figure 8.4-2. The layout described above allows manipulation of bonds to be fairly simple to implement in 1D, although it is worth noting that any multiple bond-break criterion becomes significantly more complicated to apply in the other dimensions.
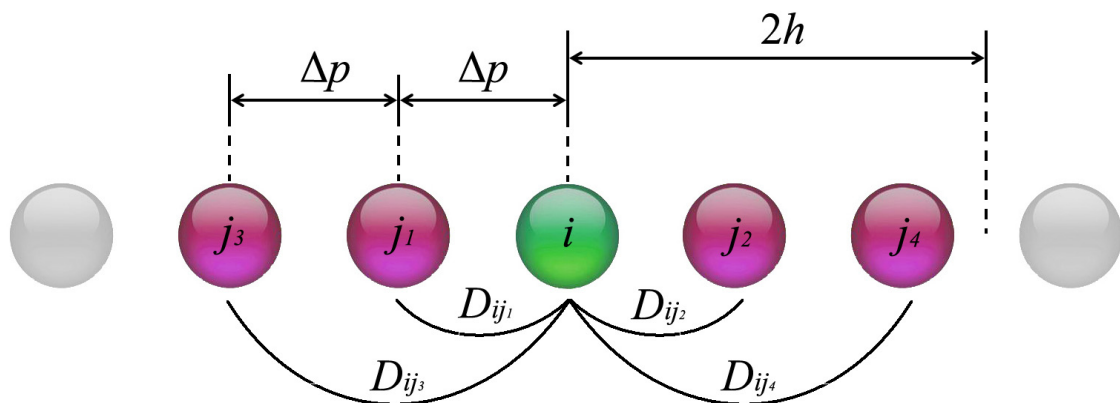
Figure 8.4-2: Number of neighbours in 1D for a single particle, i that does not lie on or near a boundary in the standard configuration of the copper plate impact simulation. The damage parameter for each bond is also displayed.

Maximum Damage Criterion

The first criterion that was tested calculates the damage parameter, D_{ij} for all particle bonds of a particle, i in the usual manner with the Cochran-Banner model. However, the value of D_{ij} on each side of the i particle is then recalculated to be the maximum value of D_{ij} in both bonds on each side, i.e. the values of D_{ij} for the bonds associated with the i particle can be written as a matrix of Left-Hand-Side (LHS) damage parameters and Right-Hand-Side (RHS) damage parameters as follows:

$$\mathbf{D}_{ij_n} = \begin{bmatrix} D_{ij_1} & D_{ij_2} \\ D_{ij_3} & D_{ij_4} \end{bmatrix},$$

where n is the neighbour number. This damage matrix is then reset via:

$$D_{LHS} = \mathbf{max}\left(D_{ij_1}, D_{ij_3}\right) \text{ and } D_{RHS} = \mathbf{max}\left(D_{ij_2}, D_{ij_4}\right),$$

such that

$$\mathbf{D}_{ij_n} = \begin{bmatrix} D_{ij_1} & D_{ij_2} \\ D_{ij_3} & D_{ij_4} \end{bmatrix} = \begin{bmatrix} D_{LHS} & D_{RHS} \\ D_{LHS} & D_{RHS} \end{bmatrix}.$$

In this way the damage on one side of the i particle grows at the fastest possible rate and both bonds will reach maximum damage at the same time. This bond manipulation criterion has been termed the 'Maximum Damage Criterion.'

Figure 8.4-3 shows the comparison of the longitudinal stress against time for the experimental and simulation data.
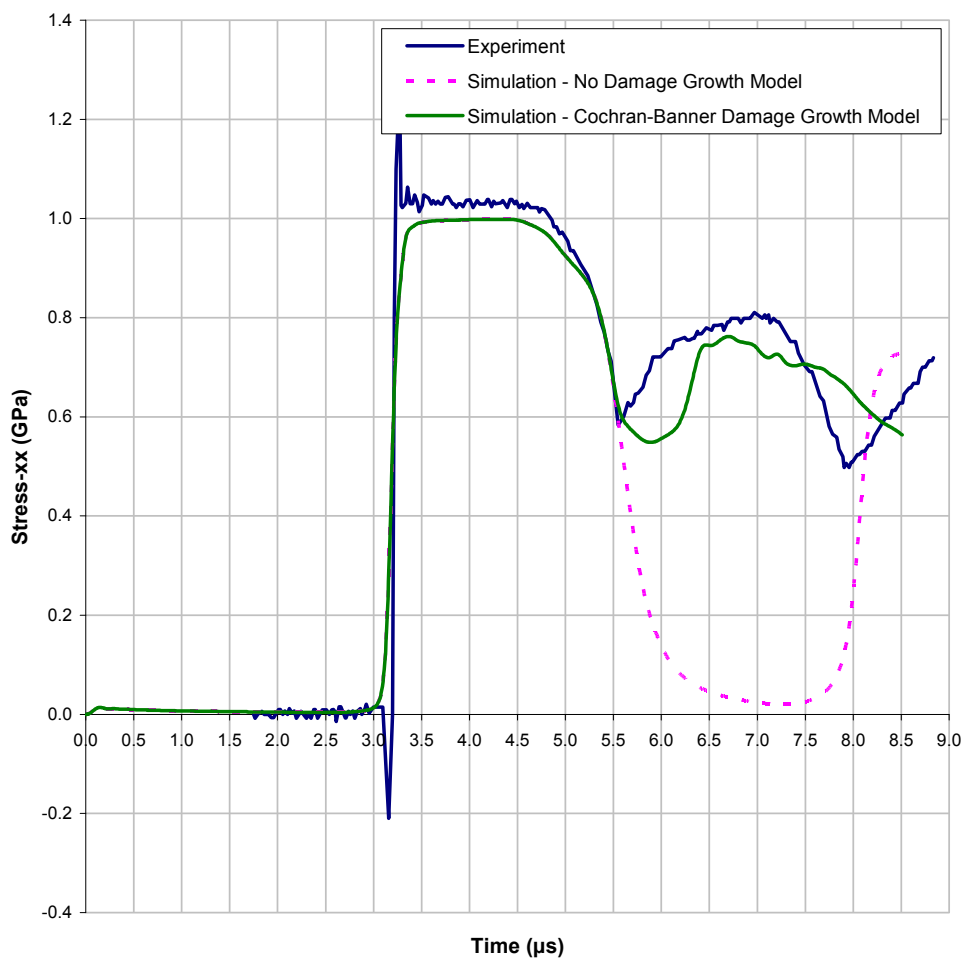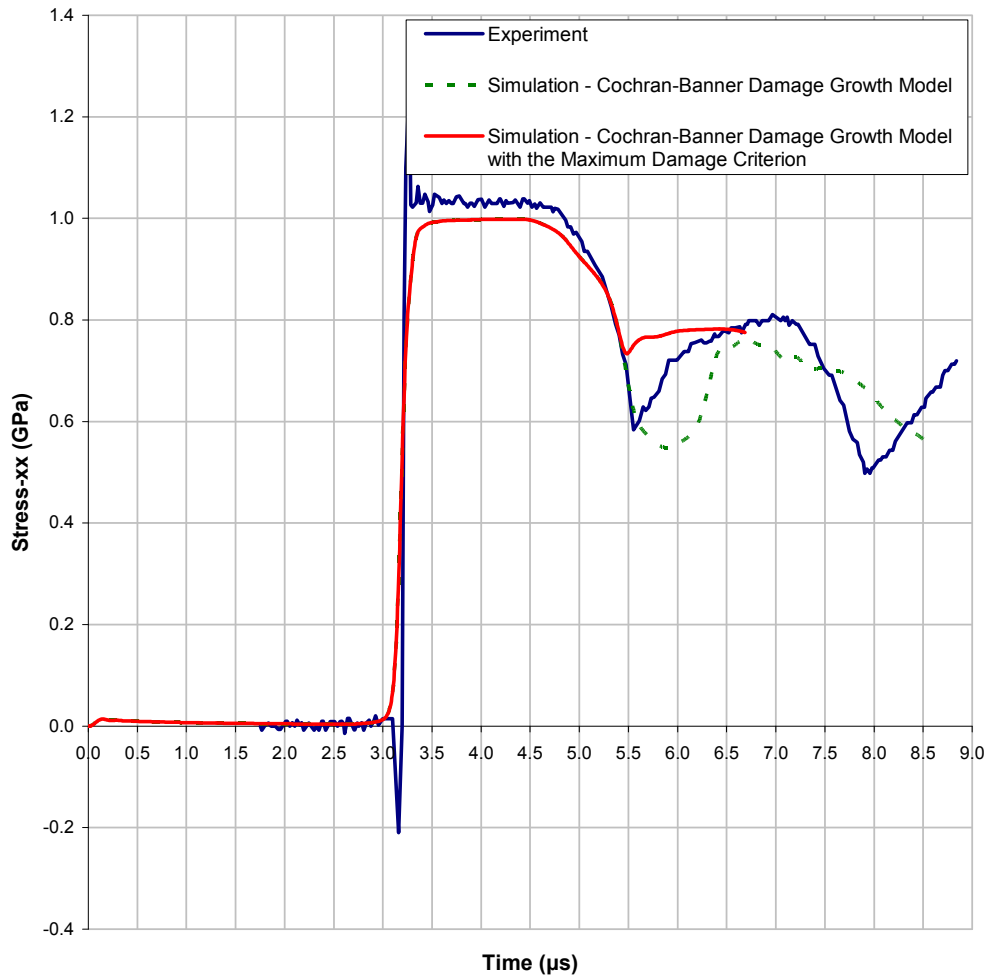
Figure 8.4-3: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 1D simulation data is shown for the Cochran-Banner damage growth model with the Maximum Damage Criterion active.

From the above plot it is apparent that this criterion has overcompensated for the slower damage growth observed previously and is now too rapid. It was therefore assessed that this criterion was unsuitable for further analysis.

<u>Bond Break Criteria</u>

The second bond manipulation criterion that was tested fails one bond given that another has already failed. This criterion was split into three separate tests as follows:

- 'Bond Break Criterion 1' states that if the 1st neighbour on one side of a particle, i has failed (i.e. $D_{ij} = 1.0$), then the 2nd neighbour is also failed.

$$\text{If } D_{ij_1} = 1.0 \text{, then } D_{ij_3} = 1.0 \text{.}$$

$$\text{If } D_{ij_2} = 1.0 \text{, then } D_{ij_4} = 1.0 \text{.}$$

- 'Bond Break Criterion 2' states that if the 2nd neighbour on one side of a particle, i has failed, then the 1st neighbour is also failed.

$$\text{If } D_{ij_3} = 1.0 \text{, then } D_{ij_1} = 1.0 \text{.}$$

$$\text{If } D_{ij_4} = 1.0 \text{, then } D_{ij_2} = 1.0 \text{.}$$

- 'Bond Break Criterion 3' states that if either the 1st or 2nd neighbour on one side of a particle, i has failed, then the other neighbour is also failed.

$$\text{If } D_{ij_3} = 1.0 \text{ or } D_{ij_1} = 1.0 \text{, then } D_{ij_1} = D_{ij_3} = 1.0 \text{.}$$

$$\text{If } D_{ij_4} = 1.0 \text{ or } D_{ij_2} = 1.0 \text{, then } D_{ij_2} = D_{ij_4} = 1.0 \text{.}$$

Figure 8.4-4 shows the comparison of the longitudinal stress against time for the experimental and simulation data with each of the above criteria active.

Figure 8.4-4: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 1D simulation data is shown for the Cochran-Banner damage growth model with the each of the Bond Break Criterions.

From the plot it is observed that Bond Break Criterion 1 has no effect on the simulation at all since the results are identical to the simulation without any criterion active. Bond Break Criteria 2 and 3 are also identical and both agree well with the experimental data. Further analysis of the results highlighted that damage grows in the bonds between the i particle and the furthest neighbours faster than in the bonds between the i particle and the nearest neighbours. This explains why Criterion 1 produces identical results to the standard case and why Criterion 2 produces identical results to Criterion 3. It was identified that this occurs because the Cochran-Banner damage parameter grows more rapidly in the furthest neighbours. It is difficult to assess if this is a correct representation of real behaviour since there is not a direct comparison of what nearest and furthest bonds represent in a real solid.

It is clear however that when the nearest neighbour bonds fail at the same rate as the furthest, the correct behaviour is observed. This suggests that some form of bond breaking criterion may be necessary to capture the correct material response, although the 1D case is too simplified to make accurate assumptions about what method to apply.

These results further reinforce that the concept of applying damage via the SIA is a valid method for representing damage since it is apparent that accurate results could be obtained; given that a damage growth model that can accurately define the damage parameter for inter-particle damage can be found. The next two sub-sections describe various changes that were made to the Cochran-Banner model in an attempt to make it more accurate.

An explanation of the SIA approach to applying damage and the Cochran-Banner damage growth model were made the subject of a journal paper submitted to the International Journal of Impact Engineering in December 2008. The paper included the results from the preliminary test depicted in Figure 8.4-1 and the results from Bond Break Criterion 2/3 depicted in Figure 8.4-4. A copy of the paper has been included in Appendix B.

8.4.1.3 Modification to the calculation of the inter-particle damage parameter in the Cochran-Banner Damage Growth Model

At the start of this section the inter-particle damage for the Cochran-Banner Damage growth model was identified as the following expression:

$$D_{ij_n} = D_{ij_{(n-1)}} + \left[\left(\frac{dV_{ij}}{|A_{ij}|} \right) \Big/ D_{crit} \right]^{2/3}.$$

The round-bracketed term in the above equation is the increment of damage applied at each time-step. This format of the equation was interpreted from work by Mirkovic (2004) and the original paper that first reported the use of the model (Cochran, Banner 1977); in these works the formula is given as follows:

$$F_{dam} = 1 - \left(\frac{D}{D_{crit}} \right)^{2/3} ,$$

where F_{dam} is the factor that reduces the area over which the stress acts ($F_{dam} \equiv 1 - D_{ij}$ in comparison to this research). Neither paper specifies whether the damage term, D in the above formula is the total damage or the damage increment. In the original format of the equation used for this research the value of D was assumed to be the damage increment. During a review of the Cochran-Banner damage growth model that was implemented in this research it was observed that the value of D could be interpreted as the total damage (i.e. the current damage plus the damage increment). In this instance the equation would therefore be written as:

$$D_{ij_n} = \left[\left(D_{ij(n-1)} + \frac{dV_{ij}}{|A_{ij}|} \right) \bigg/ D_{crit} \right]^{2/3} .$$

Since it was unclear which version of the equation was correct, the new configuration of the equation was also implemented into the MCM code and tested on the copper plate impact experiment. Figure 8.4-5 shows the comparison of the longitudinal stress against time for the experimental and simulation data with both versions of the equation used in the damage calculation.
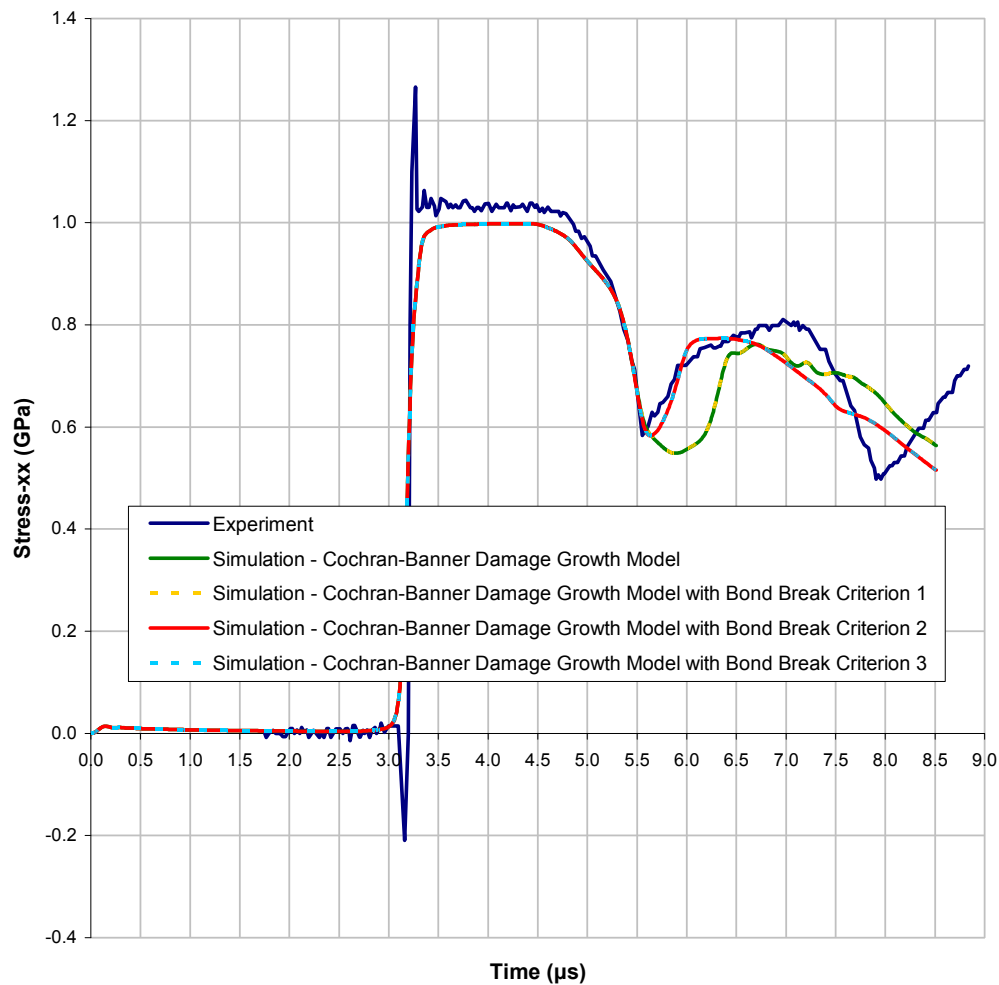
Figure 8.4-5: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 1D simulation data is shown for the Cochran-Banner damage growth model before and after the damage equation reconfiguration.

It is evident from Figure 8.4-5 that the new configuration of the equation does not represent the growth of the damage as accurately as the original configuration. Since it is unclear from the published works which version of the equation has been used, further analysis performed using the Cochran-Banner damage growth model reverted to using the original method.

8.4.1.4 Modifications to the calculation of the Cochran-Banner damage parameter to use an inter-particle length, l_{ij}

The Cochran-Banner damage growth model defined at the start of this section computes the increment of damage in a particle bond at each time-step (after it has been activated) as follows:

$$\Delta D_{ij} = \left(\frac{D_{CB}}{D_{crit}} \right)^{2/3} = \left(\frac{dV_{ij}}{|A_{ij}|} \bigg/ D_{crit} \right)^{2/3},$$

where the change in inter-particle volume, dV_{ij} is a function of the volumetric strain (from the rate of deformation tensor). Whilst this is consistent with the approach adopted by Mirkovic (2004), the volumetric strain is calculated at the individual particles and averaged between each particle pair to obtain the inter-particle property. It was assessed that since the units of D_{CB} are the units of length, the model could be adapted to compute damage in a particle bond as a function of the direct-strain or a length-scale between the two particles, thus achieving a damage growth model that is solely written in terms of inter-particle parameters. Figure 8.4-6 identifies a parameter l_{ij} (inter-particle distance) for this purpose. Where l_{ij} is defined at an arbitrary time q and at a previous time p, where p is related to q via $p = q - \eta$ and η is the current time-step.
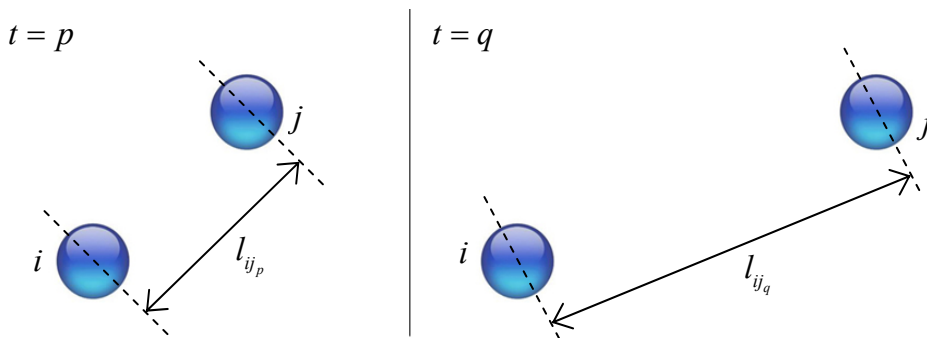
Figure 8.4-6: Diagrammatic representation of the inter-particle distance, l_{ij} at an arbitrary time, q and the previous time p.

Two possible modifications using l_{ij} that keep the original dimensionality of the damage parameter were investigated, these are summarised below.

Cochran-Banner Modification A

In this modification the change in volume (dV_{ij}) term in the damage increment equation stated at the start of this sub-section is replaced with the following:

$$dV_{ij} = \frac{\varepsilon_{ij}}{2}\left(V_i + V_j\right), \text{ where } \varepsilon_{ij} = \begin{cases} \left(\dfrac{l_{ij_q} - l_{ij_p}}{l_{ij_q}}\right) & , & l_{ij_q} > l_{ij_p} \\ 0 & , & l_{ij_q} \leq l_{ij_p} \end{cases}.$$

ε_{ij} is the current inter-particle strain at an arbitrary time q. ε_{ij} is only computed for positive changes in inter-particle distance (i.e. tension).

It should be noted that since ε_{ij} is non-dimensional, in this modification the units of D_{CB} has the units of length and so the original dimensionality has been maintained.

Cochran-Banner Modification B

In this modification the increment of damage equation (ΔD_{ij}) stated at the start of this sub-section is replaced with the following:

$$\Delta D_{ij} = \left(\Delta l_{ij} / D_{crit}\right)^{\frac{2}{3}}, \text{ where } \Delta l_{ij} = \begin{cases} \left(l_{ij_q} - l_{ij_p}\right) & , & l_{ij_q} > l_{ij_p} \\ 0 & , & l_{ij_q} \leq l_{ij_p} \end{cases}.$$

Δl_{ij} is the current change in inter-particle distance at an arbitrary time q. Δl_{ij} is only computed for positive changes in inter-particle distance (i.e. tension).

It should be noted that since Δl_{ij} has the units of length, in this modification the units of D_{CB} also has the units of length and so the original dimensionality has been maintained.

<u>Preliminary Results</u>

Both of the modifications stated above were implemented into the Cochran-Banner Damage Growth Model such that each modification could be selected independently of the original format and the other modification. Simulations were run using the copper plate impact test on each modification to see if any provided a sensible response. Figure 8.4-7 shows the comparison of the longitudinal stress against time for the experimental and simulation data with the original format of the Cochran-Banner Damage Growth Model and the two modifications (A and B).
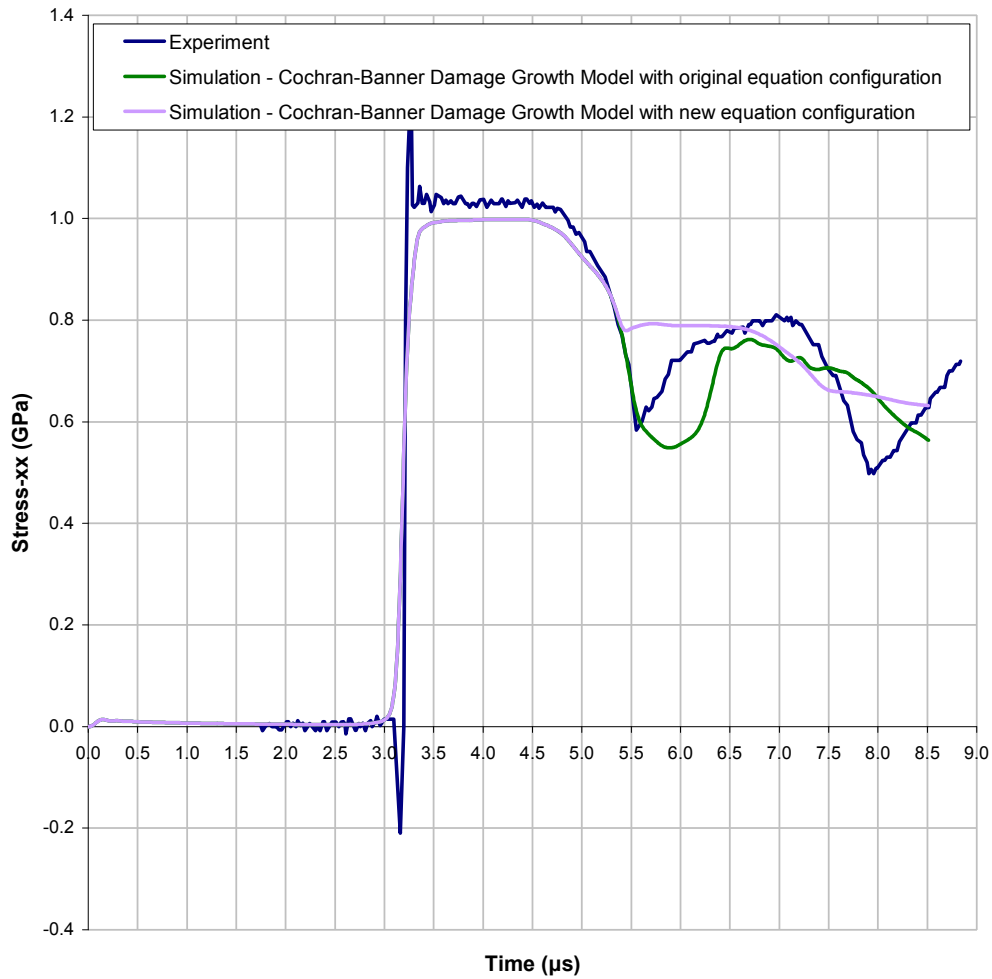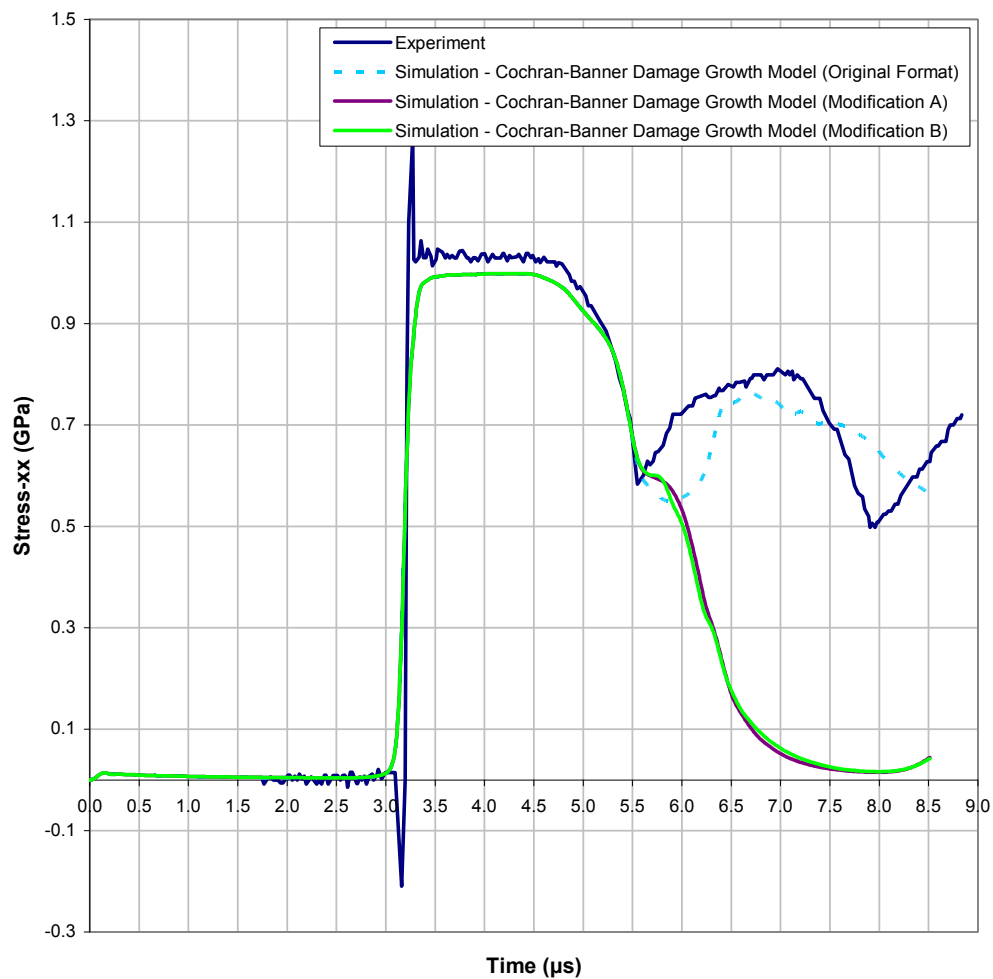

Figure 8.4-7: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 1D simulation data is shown for the Cochran-Banner damage growth model with the each of the inter-particle length, l_{ij} Modifications (A and B).

It was anticipated that the best response would be provided by Modification B, since the only difference with this method from the original format is the replacement of volumetric strain with direct strain.

However, both modifications had very similar responses, i.e. they show a response in the correct region, but both failed to produce a reload signal and ultimately return to follow the same response as a simulation with no damage present. This indicates that whilst damage is growing in the model, the formation of a new free surface to reflect the stress is not fully achieved. The values of Δl_{ij} and ε_{ij} do not change significantly enough to drive the damage growth at the correct pace.

It is possible that a better response could be achieved by examining different values of D_{crit} to 'tune' the damage growth model. The issue with this approach is that the model is then no longer compatible with the material data supplied in the original Cochran-Banner paper (1977) and would therefore require to be 'tuned' for different materials as well. This is assessed as not being a very robust approach to modelling the damage because in this case the fracture model would be made to fit the specific experiment and it will be difficult to justify the findings if a match to the experimental data can be achieved. It is therefore concluded that no benefit is gained by investigating these Modifications any further and that the original format still provides the best response.

8.4.1.5   Sensitivity Study of the SPH smoothing length on 1D Fracture Modelling

One final investigation was conducted with the fracture model in 1D before testing the code in other dimensions. A brief sensitivity study into the effect of larger and smaller neighbourhoods, i.e. the number of inter-particle bonds in the damage computation was investigated. This was achieved by testing different values of the SPH smoothing length, h. In the modelling conducted previously the SPH smoothing length was related to the initial particle spacing, Δp via:

$$h = 1.3\Delta p \, ,$$

which provided each particle (that does not lie on or next to a boundary) with four neighbours; two on either side (for the 1D simulations). It should be noted that for all the fracture modelling performed for this research, the value of h was fixed. A series of tests were performed with different ratios of Δp. Three values were chosen to reduce the total number of neighbours of a non-boundary particle to two (one either side) and also two values were chosen that still provided two neighbours either side of the i particle. It was assessed that exploring neighbourhoods that were larger than four particles would be impractical since this would add an extra level of complexity to the damage calculation and likely only increase the time to failure, i.e. the larger the neighbourhood, the more bonds between particles that need to be broken. Table 8.4-1 lists the five tests that were carried out and Figure 8.4-8 identifies them diagrammatically ($\Delta p = 0.005$ cm).

Test	Smoothing length, h (Δp)	Smoothing length, h (cm)	$2h$ (cm)	Total Number of neighbours for a particle i
W	$0.80\Delta p$	0.00400	0.00800	2
X	$0.90\Delta p$	0.00450	0.00900	2
Y	$0.99\Delta p$	0.00495	0.00990	2
Z1	$1.20\Delta p$	0.00600	0.01200	4
(Original)	$1.30\Delta p$	0.00650	0.01300	4
Z2	$1.40\Delta p$	0.00700	0.01400	4

Table 8.4-1: Description of tests carried out to test the smoothing length sensitivity for the Cochran-Banner Damage Growth Model.
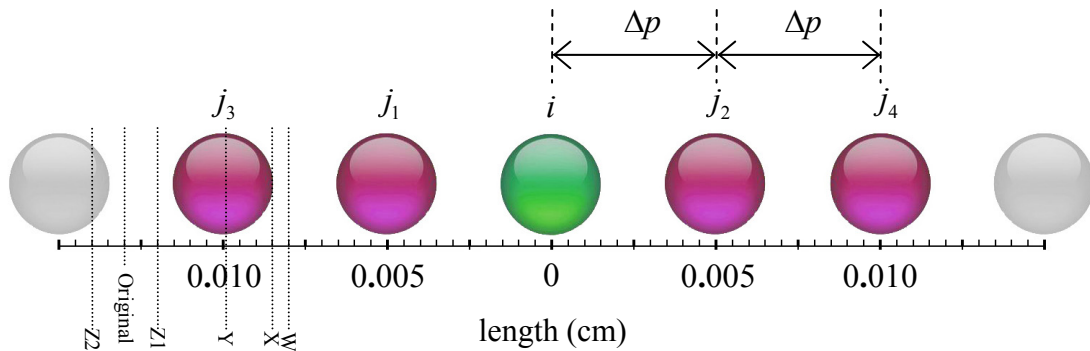
Figure 8.4-8: Diagrammatic representation of the smoothing length sensitivity tests (W-Z) for the Cochran-Banner Damage Growth Model.

The first tests that were conducted were Tests Z1 and Z2, i.e. tests with four neighbours and a larger and smaller smoothing length than used previously. Since there are 4 neighbours in both the computations, the bond breaking criteria tested in Section 8.4.1.2 (see Figure 8.4-4) could also be tested. Simulations were run on the 1D copper plate impact experiment used throughout the fracture model analysis and graphs of longitudinal stress with time were plotted to assess the response. In all cases the behaviour was identical to the original tests ($h = 1.3\Delta p$) and so it can be concluded that the smoothing length does not effect or drive the damage calculation when four inter-particle bonds are present in the damage calculation for each non-boundary particle, i.e. the normal operating conditions for the SPH method.

The remaining three tests (W-Y) cannot be tested with a bond breaking criterion since there is only one bond present either side of a particle. It was assumed that the effect of reducing the number of neighbours down to two would have an effect on both the damage growth and the response of the material prior to damage; this is because the SPH approximation is less accurate with such a small neighbourhood and therefore the model's ability to represent the experiment accurately is also reduced. Figure 8.4-9 shows the comparison of the longitudinal stress against time for the experimental and simulation data with the original smoothing length and the three, two-neighbour smoothing lengths (W-Y).

Figure 8.4-9: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 1D simulation data is shown for the Cochran-Banner damage growth model with the original smoothing length and the three tests to reduce the number of neighbours to two (Tests W-Y).

As expected the two-neighbour simulations all failed to reproduce the correct response and in each case the initial loading signal is not captured as well as the original case. The onset of damage is more rapid in these simulations since there are less bonds to fail to create a 'free particle' This is observed in the above plot just after the initial loading signal where the stress plateaus momentarily before continuing to reduce. A reload signal is then generated later than the original model but it is less severe. The main conclusion to these tests is that for the two-neighbour case, the smoothing length has an impact on the ability of the simulation to represent the experiment which also affects the exact shape of the damage. It is therefore assessed that the fracture model will not function correctly for smoothing lengths less than one times the inter-particle distance.

8.4.2   2D Modelling Results

It was decided that a sufficient investigation into the Cochran-Banner damage growth model coupled with the SIA approach for applying the damage (fracture model) had been conducted in 1D to attempt to extend the model to 2D. The initial configuration of the Cochran-Banner Damage Growth Model outlined at the start of Section 8.4 was assessed to be the best candidate for this purpose because it provided the best results when compared to the experimental data.

The 1D copper plate test simulation explained in Section 8.1 was therefore upgraded to 2D by extending the entire 1D model by 6 particles in the y-direction. Symmetry planes were added to the top and bottom of the rods so that effects from reflected waves off the boundaries would not skew the results (in the experiment a uniaxial stress state is maintained by having a sufficiently wide target plate such that waves will not reflect off of the free surfaces and reach the damage region before the damage has occurred). A full 2D slice of the actual experimental layout was assessed to be unnecessary since a uniaxial stress state is achieved with symmetry planes. The 6-particle thickness model was assessed to be adequate for initial investigations into 2D fracture modelling. Also it would require a considerable number of particles to maintain the same resolution as the 1D model and would significantly increase the computation time for each simulation.

Initial simulations were conducted without the fracture model active to determine that the same longitudinal stress signal used in the 1D modelling to display the presence of damage is achieved. Figure 8.4-10 shows the longitudinal stress for a particle that lies just behind the boundary between the PMMA and the copper target plate (three particles inside the PMMA) for a 2D simulation with an identical set-up to the 1D 'no-damage' simulation. It is apparent that a large amount of noise has been introduced into the signal.
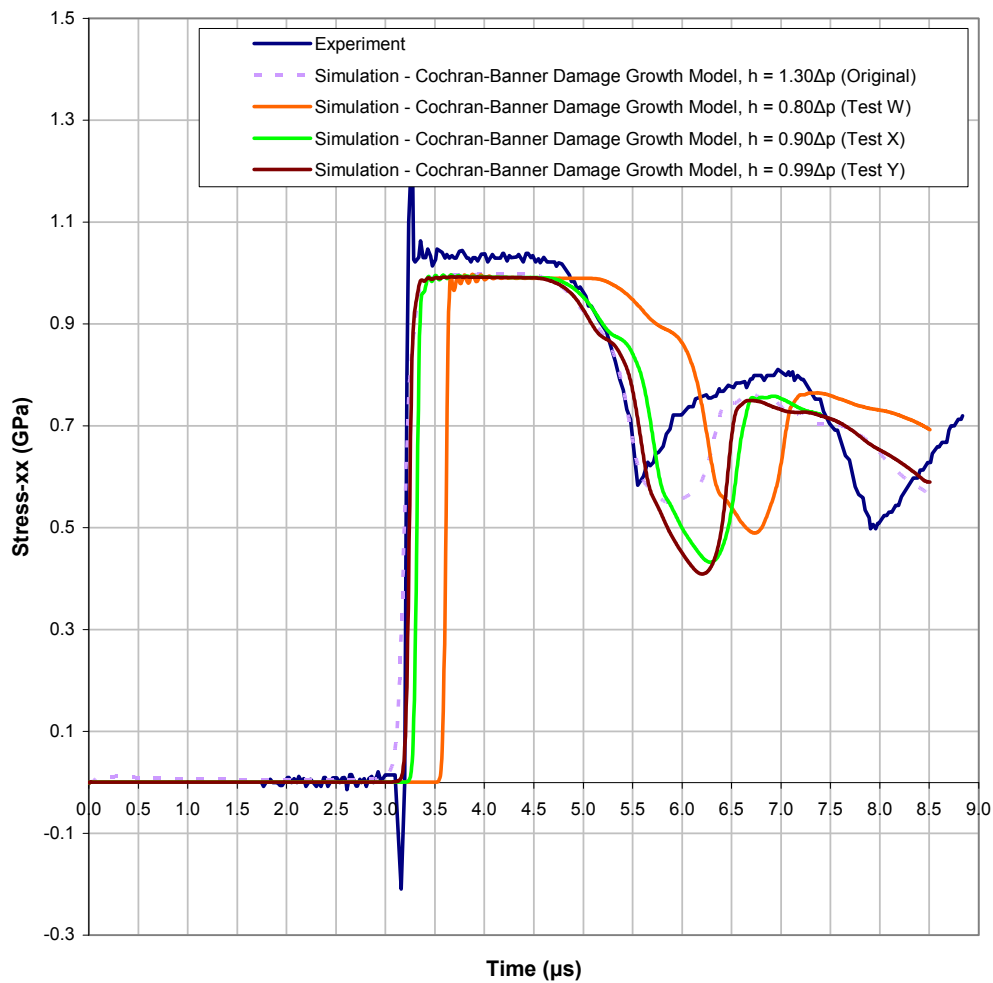
Figure 8.4-10: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 2D simulation data is shown for the 'no-damage' case.

The noise has been introduced by the contact algorithm used in the MCM code. The effect is caused by erratic motion of the particles on the contact boundaries. This can be seen in Figure 8.4-11 where a magnified view of the contact boundaries after 8.51µs is displayed.
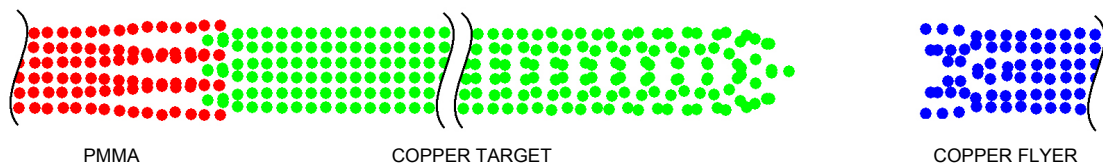
Figure 8.4-11: Magnified view of the contact boundaries in the 2D Copper Plate Impact Simulation with unconstrained SPH particles after 8.51 µs.

The material contact description in MCM is a repulsive force contact algorithm and requires values for the input parameters to be determined through test simulations. Dimensionality is known to have an effect on this particular contact description and therefore it was not unexpected that different values to the parameters used in the 1D simulations would be required to stabilise the contact in 2D. However after an extensive iteration process of testing different combinations of the input values, a stable 2D model could not be found. This was rectified by constraining the motion of the SPH particles in the y-direction, i.e. effectively constructing a 1D simulation within the 2D solver. The result of the 'no-damage' simulation with the constrained particles is also displayed in Figure 8.4-10 above and Figure 8.4-12 shows a magnified view of the contact boundaries at 8.51μs.

| PMMA | COPPER TARGET | COPPER FLYER |
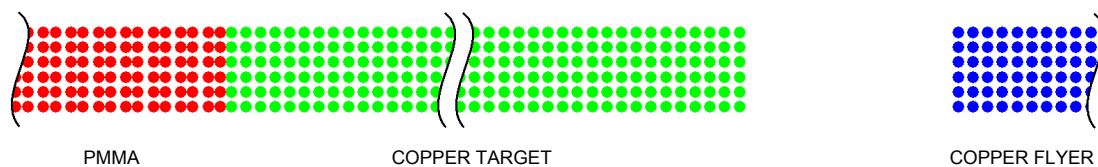
Figure 8.4-12: Magnified view of the contact boundaries in the 2D Copper Plate Impact Simulation with constrained SPH particles in the y-direction after 8.51 μs.

The problem with this approach is that model is not a true 2D representation of the experiment. Ideally an investigation into a different contact algorithm that is less sensitive to the input parameters or a correction to the SPH solver would be conducted to allow this simulation to be modelled without constraints. However the 2D analysis was performed towards the end of this research and therefore such investigations were outside the scope of the project. It is understood that the 2D fracture modelling conducted with the constrained model does not provide a true representation of the capability of the fracture model and therefore can only be considered to give an indication of performance.

8.4.2.1 Preliminary Test

An initial damage simulation was conducted using the Cochran-Banner Damage growth model as described at the start of this section (Section 8.4) as per the conclusions of the 1D modelling. The simulation model utilised the constrained version of the 2D model discussed previously. Figure 8.4-13 shows a comparison of the longitudinal stress against time for the experimental and simulation data (simulation data is plotted for the 3rd particle in the PMMA as previously stated). A plot of the simulation data with no fracture model present has also been included for comparison.
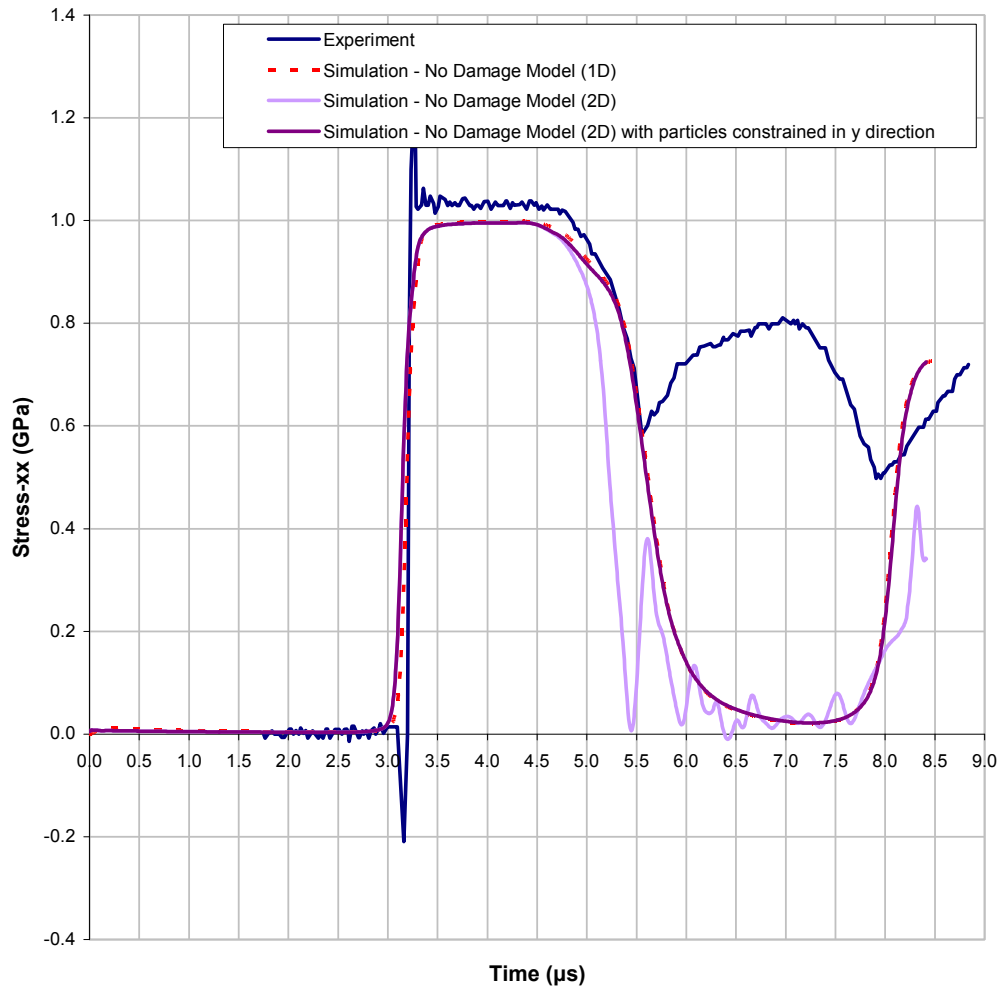

Figure 8.4-13: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 2D simulation data is shown for the Cochran-Banner damage growth model and is compared to the experimental data.

It is clear from the results shown in Figure 8.4-13 that the fracture model causes the particle bonds in the region of the expected spall plane to fail and a release wave is formed. Unlike the initial 1D simulation the shape of the reload signal has been captured very well. This provides further evidence to suggest that the SIA approach used to apply the damage is very effective.

8.4.2.2   Damage Growth Model Normalisation using 1D and 2D values of A_{ij}

The initial 1D damage simulation presented in Section 8.4.1.1 did not represent the experimental data correctly; however the initial 2D damage simulation provided a good representation of the experiment. Since the 2D simulation provides a 1D state in the 2D framework (because of the constrained particles), it was thought that perhaps the Cochran-Banner damage growth model could be normalised to provide the same response in both 1D and 2D simulations. The Cochran Banner damage parameter is computed by the following formula:

$$D_{CB} = dV_{ij} \Big/ \left| A_{ij} \right|,$$

where $\left| A_{ij} \right|$ is the magnitude of the Swegle Interaction Area (SIA). The SIA is dependent on the SPH kernel and therefore varies in each dimension. It was hypothesised that this parameter was creating a dependence on the dimensionality of the problem for the damage growth and a single constant value in place of $\left| A_{ij} \right|$ might therefore be able to normalise the results between 1D and 2D. It should be noted that the area normalisation conducted here is limited to the Cochran Banner damage growth model, i.e. the Swegle Interaction Area component of the SPH momentum equation remains unchanged and is reduced by the inter-particle damage parameter as outlined in Chapter 7.

An investigation was conducted into the variation of $\left| A_{ij} \right|$ in the undamaged material within the 1D and 2D neighbourhoods. The results of the investigation are displayed in Figure 8.4-14 which shows the neighbourhoods for a particle, i which does not lie on or next to a boundary in both the 1D and 2D simulations. The figure also displays the corresponding values of $\left| A_{ij} \right|$ within each of the inter-i particle bonds.

1D NEIGHBOURHOOD

<u>EACH i PARTICLE HAS 2 SETS OF UNIQUE BONDS (4 BONDS TOTAL)</u>

$\left|A_{ij}\right|_{1D-1} = 0.3851 \text{ cm}^2$ } DARK BLUE NEIGHBOUR PARTICLES FORM BONDS $\left|A_{ij}\right|_{1D-1}$ WITH PARTICLE, i

$\left|A_{ij}\right|_{1D-2} = 0.0630 \text{ cm}^2$ } PINK NEIGHBOUR PARTICLES FORM BONDS $\left|A_{ij}\right|_{1D-2}$ WITH PARTICLE, i

2D NEIGHBOURHOOD

<u>EACH i PARTICLE HAS 4 SETS OF UNIQUE BONDS (20 BONDS TOTAL)</u>

$\left|A_{ij}\right|_{2D-1} = 0.0010104 \text{ cm}^2$ | DARK BLUE NEIGHBOUR PARTICLES FORM BONDS $\left|A_{ij}\right|_{2D-1}$ WITH PARTICLE, i

$\left|A_{ij}\right|_{2D-2} = 0.0006458 \text{ cm}^2$ | LIGHT BLUE NEIGHBOUR PARTICLES FORM BONDS $\left|A_{ij}\right|_{2D-2}$ WITH PARTICLE, i

$\left|A_{ij}\right|_{2D-3} = 0.0001653 \text{ cm}^2$ | PINK NEIGHBOUR PARTICLES FORM BONDS $\left|A_{ij}\right|_{2D-3}$ WITH PARTICLE, i

$\left|A_{ij}\right|_{2D-4} = 0.0000608 \text{ cm}^2$ | DARK GREEN NEIGHBOUR PARTICLES FORM BONDS $\left|A_{ij}\right|_{2D-4}$ WITH PARTICLE, i

Figure 8.4-14: Diagrammatic representation of the number of neighbours (and inter-particle bonds) for a particle, i and the corresponding values of the magnitude of the SIA for each bond in the undamaged material.

It is clear from the data on the previous page that the value of $\left|A_{ij}\right|$ varies by two orders of magnitude between 1D and 2D simulations. This explains why the growth of damage is different in each case. The average value of $\left|A_{ij}\right|$ in the neighbourhood for the 1D simulation is 0.075cm^2 and 0.00038772cm^2 for the 2D simulation. Test simulations were performed with constant values of $\left|A_{ij}\right|$ in the damage increment formula in both the 1D and 2D simulations. The values that were chosen were 0.00075cm^2, 0.0075cm^2, 0.075cm^2 and 0.75cm^2 to ensure that all the orders of magnitude of $\left|A_{ij}\right|$ from the 1D and 2D simulations were covered. Figure 8.4-15 and Figure 8.4-16 show the results from the 1D and 2D tests respectively.
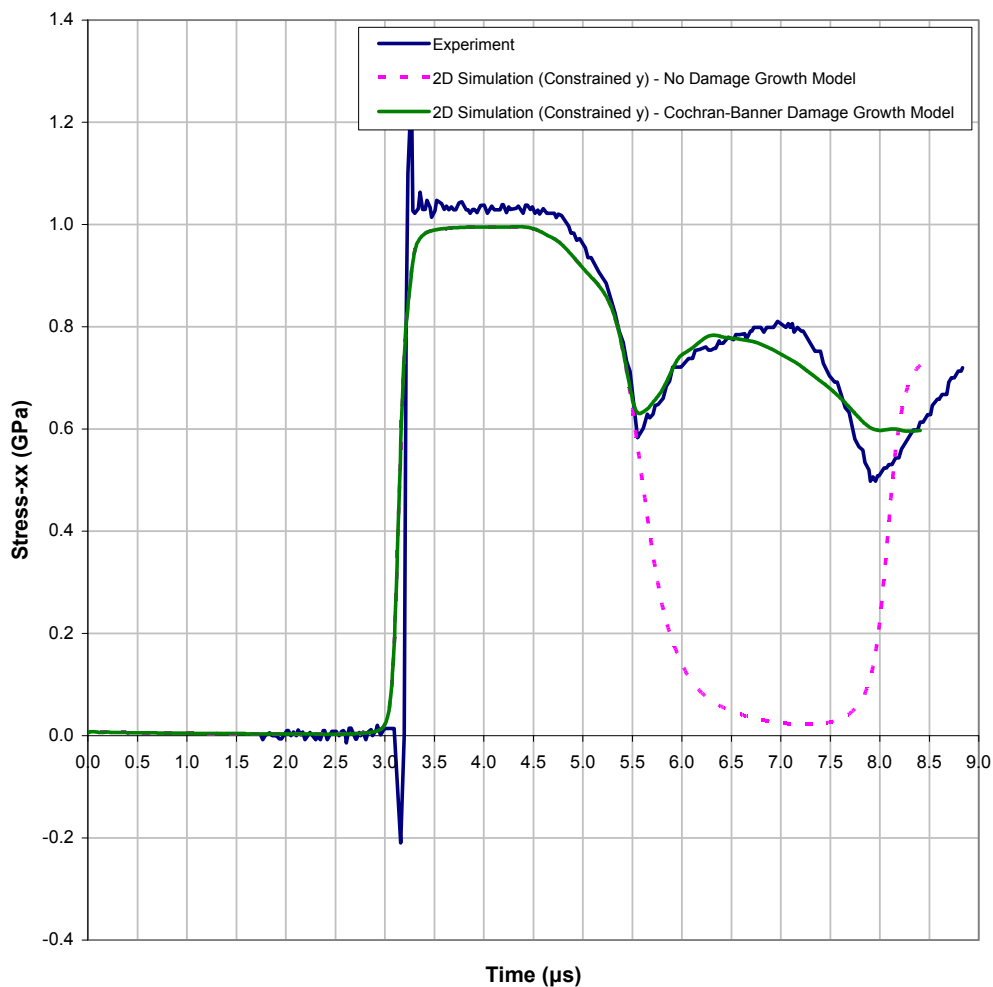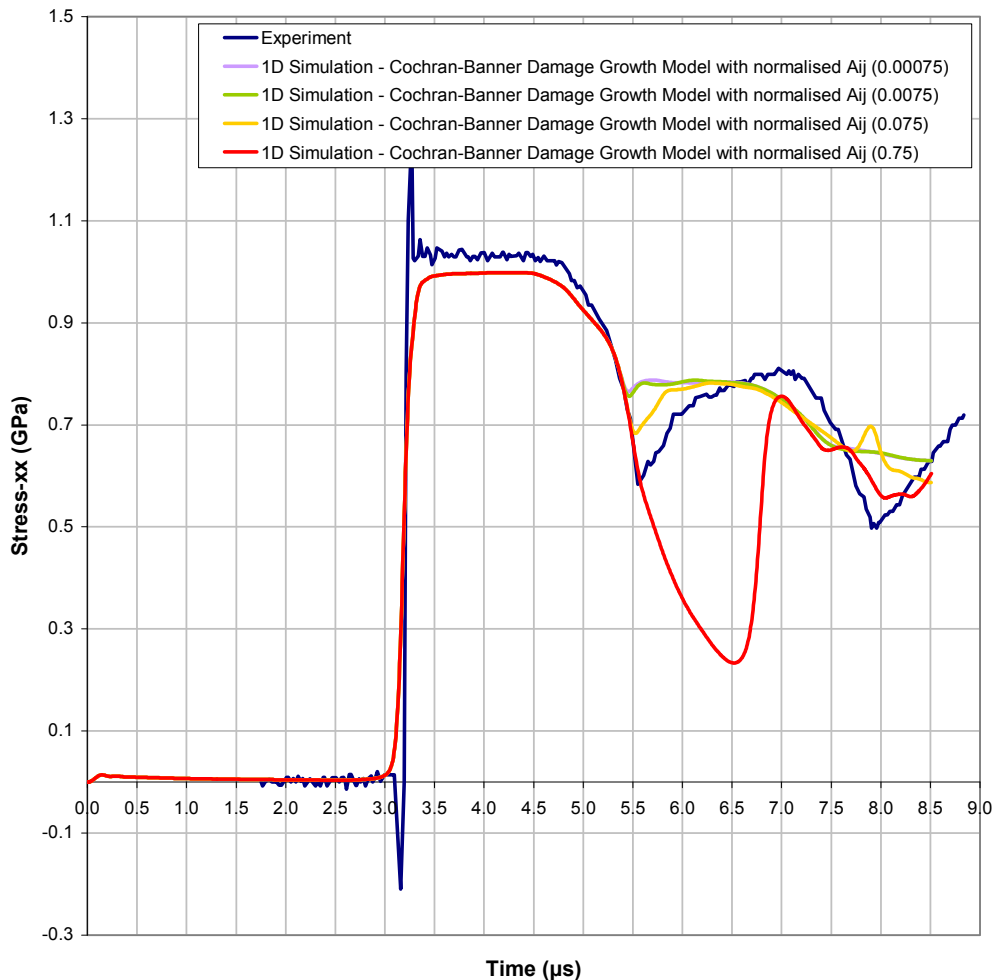

Figure 8.4-15: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 1D simulation data is shown for the Cochran-Banner damage growth model with normalised values of $\left|A_{ij}\right|$ used in the calculation of the damage increment.
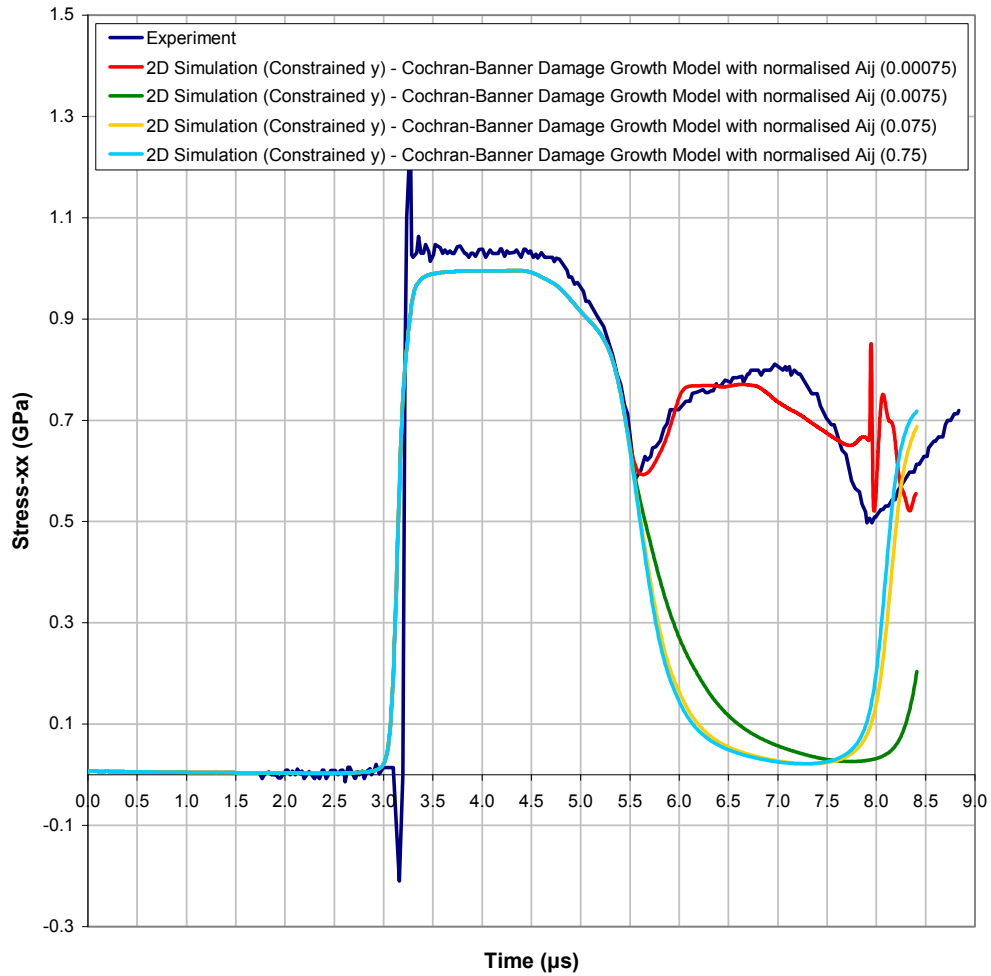
Figure 8.4-16: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 2D simulation data is shown for the Cochran-Banner damage growth model with normalised values of $\left|A_{ij}\right|$ used in the calculation of the damage increment.

From the two figures above it is apparent that none of the normalised values tested in 1D have produced a response that broadly matches the experimental data. However the 0.00075cm^2 value tested in a 2D simulation has produced a good representation. It is worth noting that in both cases the best results are achieved with normalised values that are closest to the respective average values of $\left|A_{ij}\right|$ in the undamaged cases.

A series of 1D simulations were conducted to iterate towards a normalised value of $\left|A_{ij}\right|$ that also produces the correct damage response. The value was determined to be 0.15cm^2 and the longitudinal stress with time is displayed for both the 1D and 2D cases in Figure 8.4-17 below.
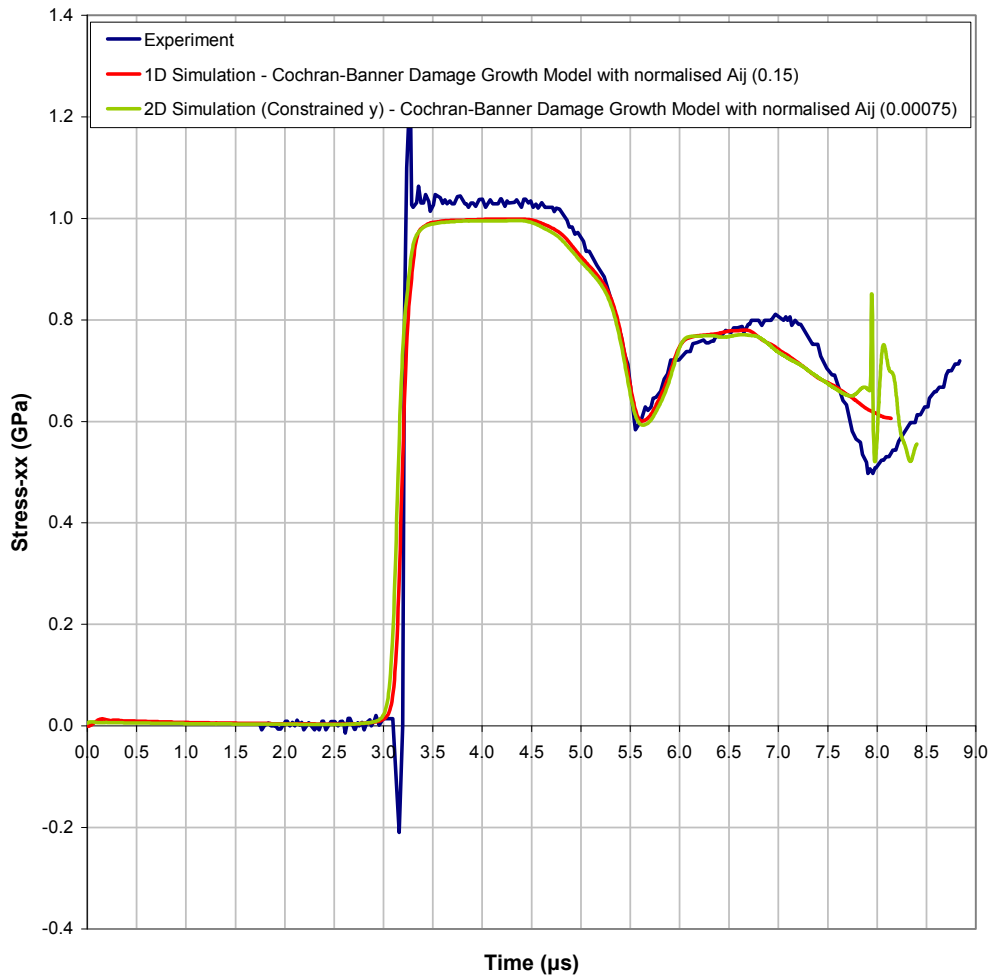
Figure 8.4-17: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 1D and 2D simulation data is shown for the Cochran-Banner damage growth model with normalised values of $\left|A_{ij}\right|$ used in the calculation of the damage increment that produce the same results.

Although it was possible to find normalised values of $\left|A_{ij}\right|$ that produce the correct damage response in both the 1D and 2D simulations, the values were more than two orders of magnitude different from each other.

It can therefore be concluded that the Cochran-Banner Damage Growth Model that has been investigated is insufficient to describe the growth of damage in all dimensions since it is inconsistent across the 1D and 2D SPH solvers when the 2D solver is set-up to behave like 1D solver (via constraining the nodes in the y-direction). It has been highlighted that the increment of damage is influenced by the dimensionality of the problem via the $\left|A_{ij}\right|$ term which is dependent on the SPH kernel function. It is therefore

assessed that this dependency should be removed before attempting to test the fracture model on more complicated 2D and 3D simulations. Since several investigations into other ways of describing damage within the Cochran-Banner Damage Growth Model have already been attempted as part of the initial 1D fracture modelling; it is assessed that further modifications to the model are outside the scope of this research and are left for further work. Therefore the assumption for the remainder of the work is that there are 1D, 2D and 3D versions of the damage growth model.

8.4.2.3   Additional manipulation of inter-particle bonds

The best results that were observed with the 1D fracture modelling were achieved by the addition of a multiple bond breaking criterion coupled to the Cochran-Banner Damage Growth Model. It is possible that a criterion could be used to reduce the dependency of the damage growth model on the dimensionality of the simulation (as discussed in the previous sub-section). This is because a multiple bond-breaking criterion adds a second level of failure to the model that bypasses the damage growth model. As with the 1D simulations however, the concern with this approach is that it is difficult to design a coherent and robust bond breaking criterion in multiple dimensions and therefore difficult to justify such a criterion with respect to a real scenario. With these concerns in mind only a simple criterion was tested that could be used in both 2D and 3D simulations; more complicated criteria were left for possible future investigations. It is also worth noting that before any criterion was implemented it was thought that it would be unlikely to have a positive effect since the standard case (discussed in Section 8.4.2.1) provided such accurate results.

The criterion that was used breaks all the bonds in a neighbourhood that lie within an angle, θ_D of a bond (position vector between two particles) which has reached maximum damage. This is shown diagrammatically for 2D in Figure 8.4-18. In a 3D simulation the damage zone is a cone with the angle θ_D, i.e. all bonds are broken in a neighbourhood that lie within a cone whose tip is at the i particle and is centred along the vector $(i \rightarrow j)$ where the bond between i and j has reached maximum damage.
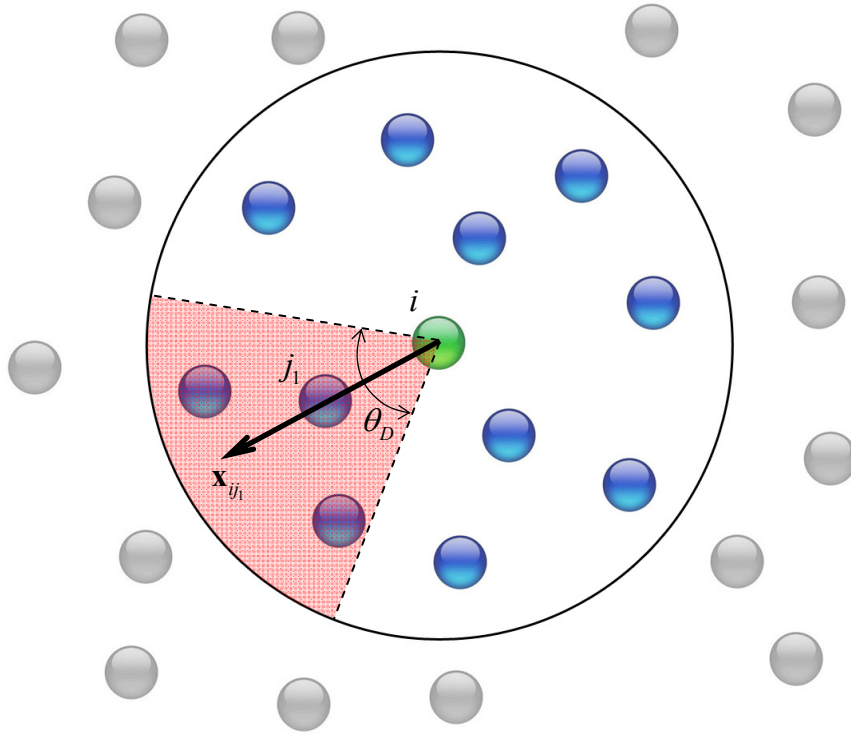
Figure 8.4-18: Diagrammatic representation of the 2D Angular Bond Break Criterion coupled to the Cochran-Banner Damage Growth model.

The Angular Bond Breaking Criterion is implemented into the code as follows:

When the bond between a particle i and one of its neighbours j_1 has failed ($D_{ij_1} = 1.0$), the position vector, $\mathbf{x}_{ij_1}$ between particle i and j_1 and its magnitude, $\left| \mathbf{x}_{ij_1} \right|$ are calculated via:

$$\mathbf{x}_{ij_1} = \begin{bmatrix} \left(x_{j_1} - x_i \right) \\ \left(y_{j_1} - y_i \right) \\ \left(z_{j_1} - z_i \right) \end{bmatrix} = \begin{bmatrix} x_{ij_1} \\ y_{ij_1} \\ z_{ij_1} \end{bmatrix} \text{ and } \left| \mathbf{x}_{ij_1} \right| = \sqrt{x_{ij_1}^2 + y_{ij_1}^2 + z_{ij_1}^2} \; .$$

The angle between vector $\mathbf{x}_{ij_1}$ and $\mathbf{x}_{ij_n}$ is then calculated using the following relationship:

$$\cos \phi = \frac{\left(\mathbf{x}_{ij_1} \cdot \mathbf{x}_{ij_n} \right)}{\left| \mathbf{x}_{ij_1} \right| \left| \mathbf{x}_{ij_n} \right|},$$

where $\mathbf{x}_{ij_n}$ represents the position vectors between the i particle and all the remaining neighbours in the neighbourhood of i. If $\cos\phi > \cos(\theta_D/2)$ then the bond between i and j_n is also broken by setting $D_{ij_n} = 1.0$.

Test simulations were conducted with damage angles of 40° and 180°. The longitudinal stress against time at the rear end of the target plate is displayed in Figure 8.4-19 below.

Figure 8.4-19: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 2D simulation data is shown for the Cochran-Banner damage growth model with an Angular Bond Break Criterion, (the damage angle is set to 40° and 180°).

The addition of the Angular Bond Breaking Criterion in 2D causes the reload signal to occur too early because it acts to increase the onset of the damage. The original case without any criterion active provided a close match to the experimental data and therefore this behaviour is expected.

8.4.3   3D Modelling Results

A 3D demonstration of the model is necessary to show that the model can be practically applied to the applications of interest. The original objective of the project was to demonstrate the model on the Mock-Holt explosively driven experiment simulation that was developed as a part of this project (see Section 2.1.1.2). Unfortunately at the end of the research the Mock-Holt simulations could only be performed with the Eulerian SPH formulation. The concept and application of inter-particle failure is fundamental to the fracture modelling approach and thus the requirement to track inter-particle bonds as material history parameters limits the fracture modelling to the Total Lagrangian SPH formulation. The available Mock-Holt model is therefore unsuitable for use with the SIA fracture model and Cochran-Banner Damage Growth Model.

There was insufficient time remaining to research and develop any other suitable experiments that could be extended to a 3D simulation model and therefore a 3D version of the 1D copper plate test simulation explained in Section 8.1 was developed by extending the entire 1D model by 6-particles in both the y and z-directions (to be consistent with the 2D model). Symmetry planes were added to the four sides to maintain a uniaxial stress state.

Initial simulations were conducted without the fracture model active to determine that the same longitudinal stress signal used in the 1D modelling to display the presence of damage is achieved. Figure 8.4-20 shows the longitudinal stress for a particle that lies just behind the boundary between the PMMA and the copper target plate (three particles inside the PMMA) for a 3D simulation with an identical set-up to the 1D 'no-damage' simulation. From the figure it is apparent that the same contact issue that was present in the 2D model is introducing noise into the signal.
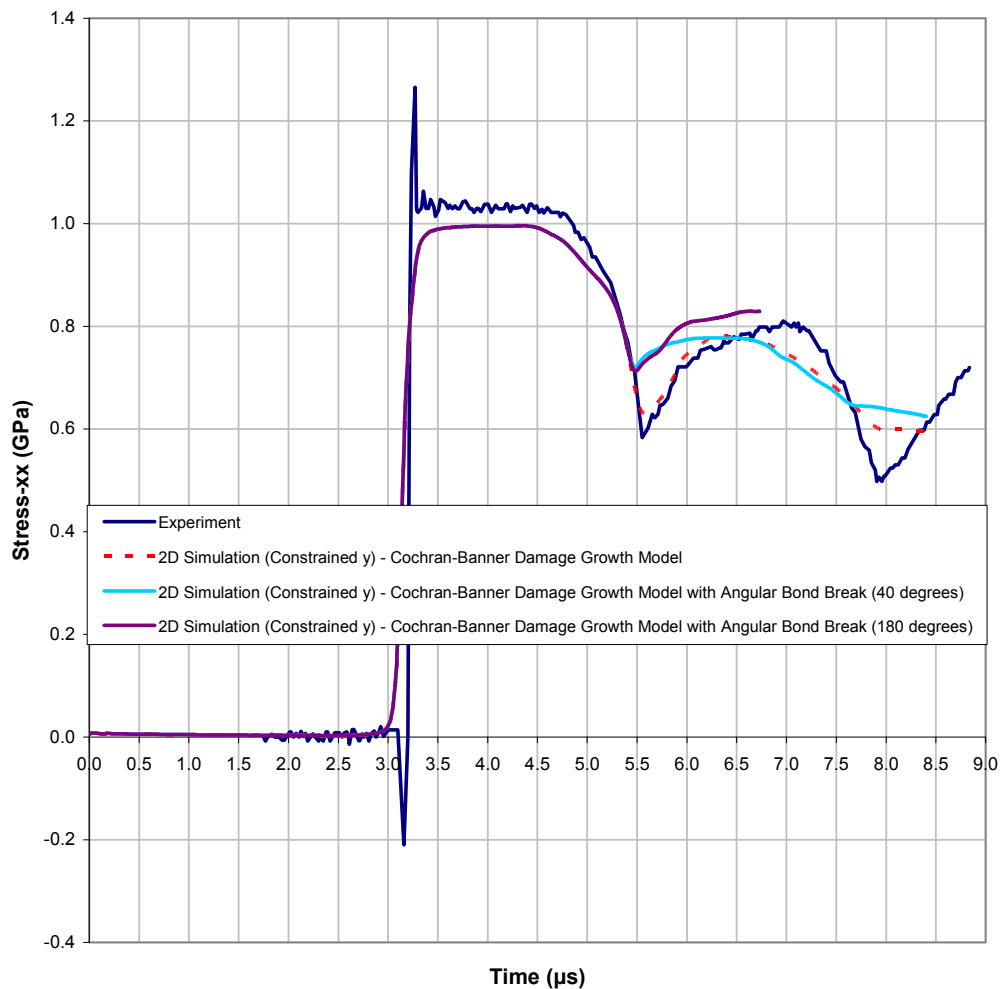
Figure 8.4-20: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 3D simulation data is shown for the 'no-damage' case.

Figure 8.4-21 displays a magnified view of the contact boundaries after 8.51 μs, the same erratic behaviour that was observed with the 2D model is present in the simulation. As with the 2D model a series of simulations were performed with varying values of the contact algorithm input parameters, but a stable simulation could not be achieved. This prevents an unconstrained version of the model from being used. Figure 8.4-22 shows magnified view of the contact boundaries after 8.51μs for a simulation that has been constrained in both the y and z-directions; the simulation is stable. The longitudinal stress with time for this simulation is also displayed in Figure 8.4-20 above and it is clear that the behaviour is consistent with the 1D model.
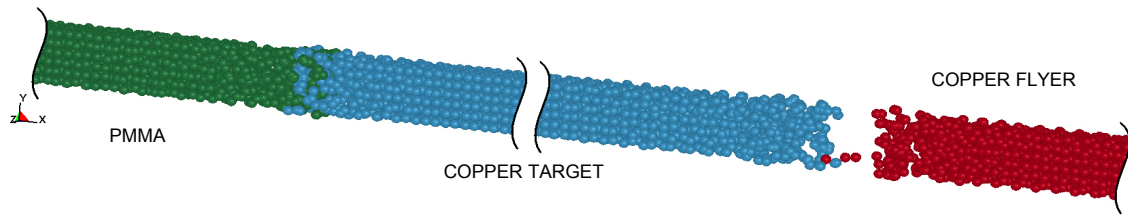
Figure 8.4-21: Magnified view of the contact boundaries in the 3D Copper Plate Impact Simulation with unconstrained SPH particles after 8.51 µs.
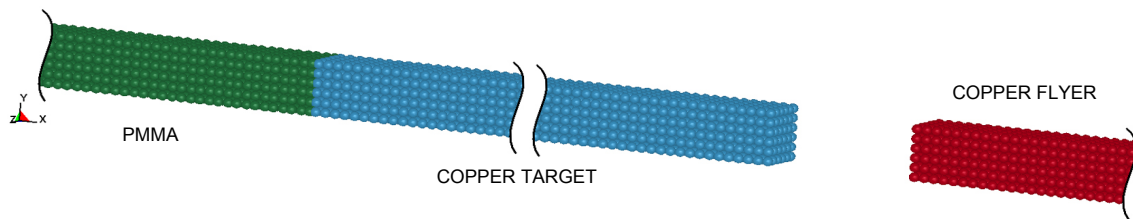
Figure 8.4-22: Magnified view of the contact boundaries in the 3D Copper Plate Impact Simulation with constrained SPH particles in the y and z-directions after 8.51 µs.

Again the problem with this approach is that model is not a true 3D representation of the experiment, it is effectively a 1D simulation within the 3D solver. It is understood that the 3D fracture modelling conducted with the constrained model does not provide a true representation of the capability of the fracture model and therefore can only be considered to give an indication of performance.

8.4.3.1   Preliminary Test

Again the initial configuration of the Cochran-Banner Damage Growth Model outlined at the start of Section 8.4 was assessed to be the best candidate for the 3D modelling test because it provided the best results when compared to the experimental data in both the 1D and 2D analysis. The simulation model utilised the constrained version of the 3D model discussed previously. Figure 8.4-23 shows a comparison of the longitudinal stress against time for the experimental and simulation data (simulation data is plotted for the 3rd particle in the PMMA as previously stated). A plot of the simulation data with no fracture model present has also been included for comparison.
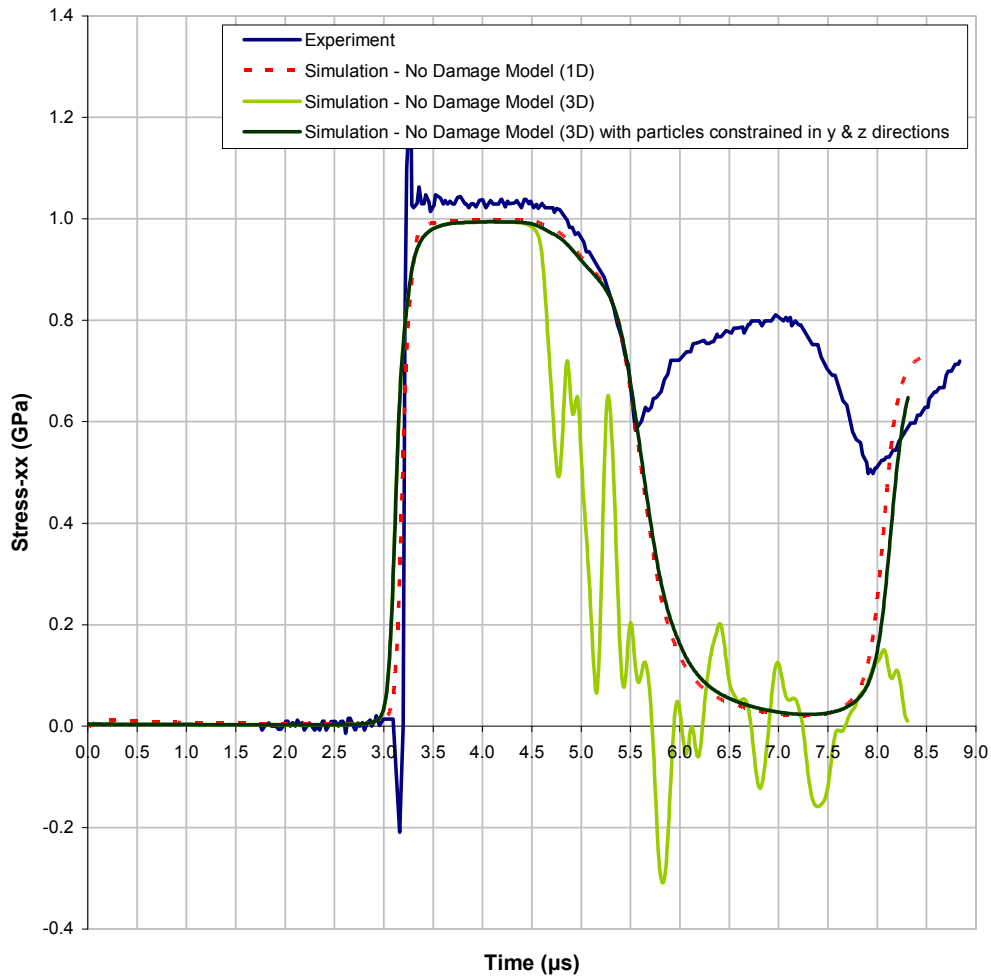
Figure 8.4-23: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 3D simulation data is shown for the Cochran-Banner damage growth model and is compared to the experimental data.

From the results shown in Figure 8.4-23 it is evident that the damage growth model causes the damage to grow too quickly. The version of the Cochran-Banner Damage Growth Model that was used has the magnitude of the SIA term, $\left| A_{ij} \right|$ present in the calculation for the increment of damage, which has been shown previously to have a dependency on the kernel, i.e. a dependency on the dimensionality of the problem. The 2D simulation showed a more rapid onset of damage than the 1D simulation and therefore a more rapid response to the 2D simulation was anticipated in 3D. As discussed previously this suggests that the dimensionality of the growth model needs to be rectified so that a consistent response is generated in all three solvers. At the time that the 3D work was conducted further modifications to the code were impractical with

the time remaining and therefore the development of the fracture model was halted at this point.

An important thing to note is that whilst the damage response in the 3D model was not captured entirely correctly, the simulation achieved maximum damage in several complete planes of the target rod and new free surfaces were generated. These can be seen in Figure 8.4-24 below.
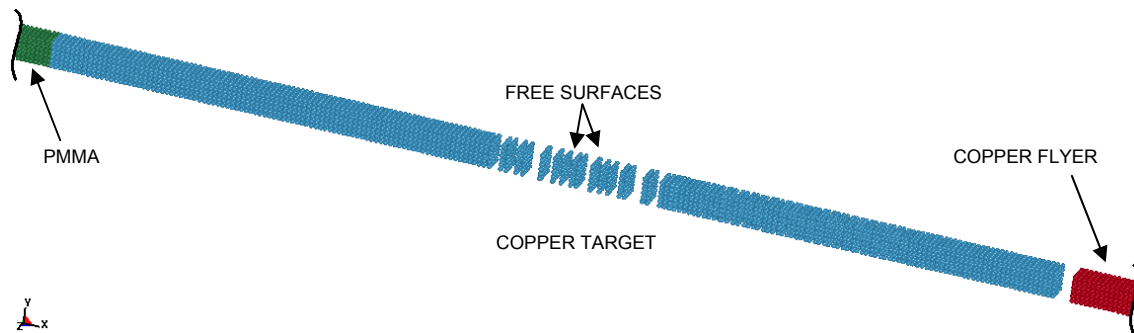
Figure 8.4-24: Magnified view of the centre of the target rod (plate) at 8.51 µs depicting the generation of new free surfaces in the 3D fracture modelling copper plate impact test with the SIA and the Cochran-Banner Damage Growth Model.

8.4.3.2 Additional manipulation of inter-particle bonds

One final test was conducted with the 3D model, which was to run a simulation with the Angular Bond Breaking Criterion explained in Section 8.4.2.3 coupled to the damage growth model. Since in all previous cases the addition of a bond breaking criterion has acted to increase the onset of failure, it was anticipated the 3D damage response would only get worse when the angular criterion was activated. An arbitrary damage angle of 40° was selected and the results were as expected. The damage occurs more rapidly than the case without the criterion active. The longitudinal stress with time is displayed in Figure 8.4-25 below.
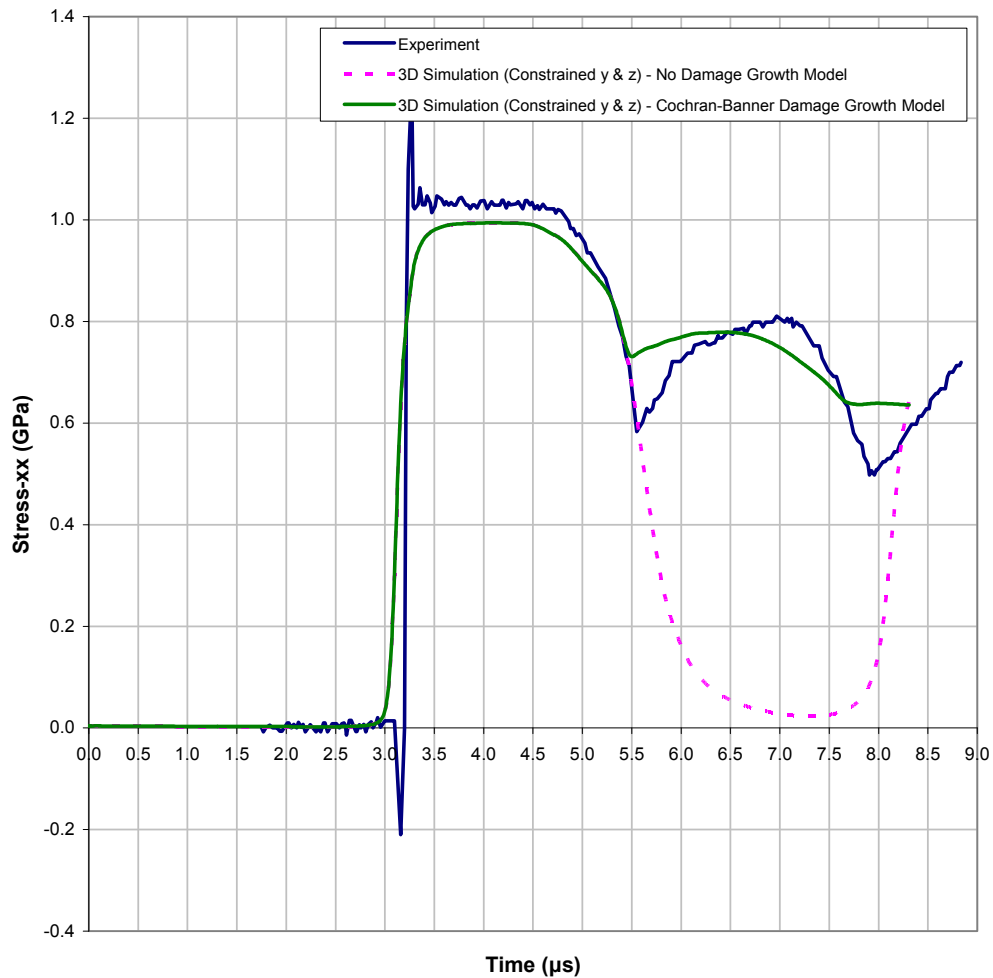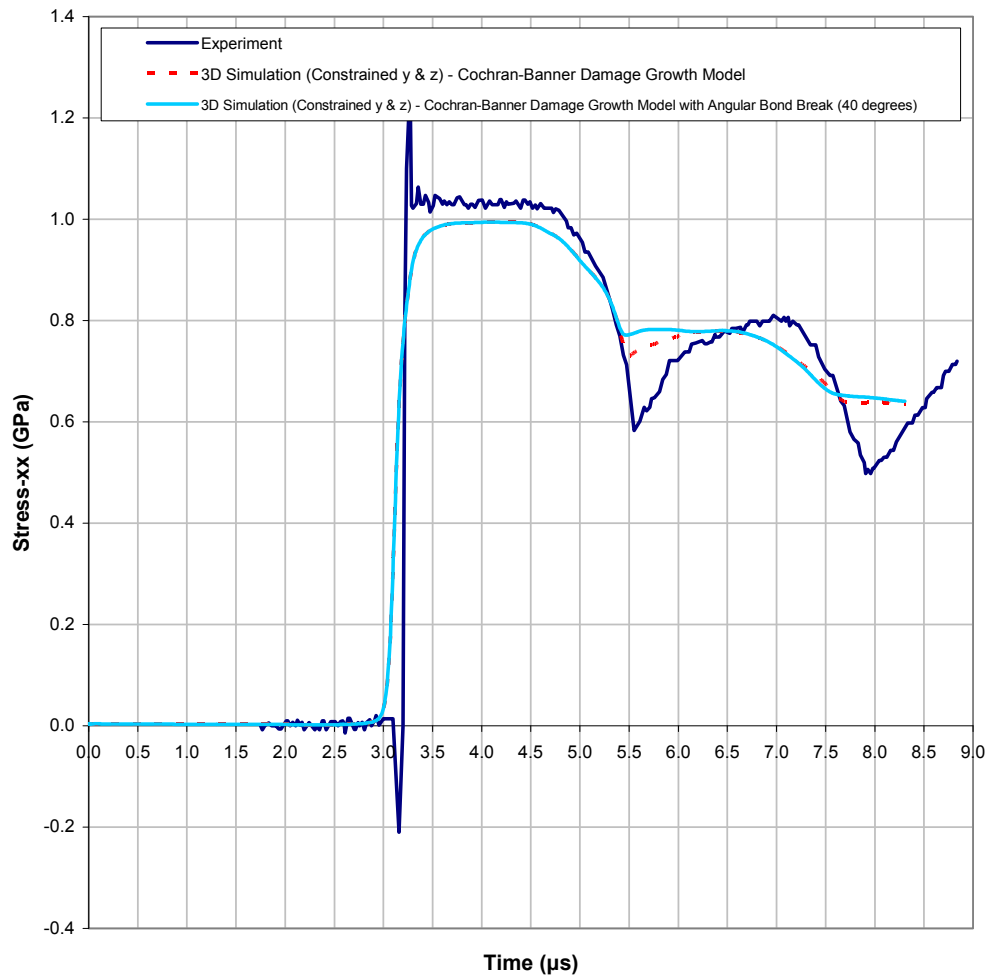
Figure 8.4-25: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. 3D simulation data is shown for the Cochran-Banner damage growth model with an Angular Bond Break Criterion, (the damage angle is set to 40°).

8.5   Conclusions to Fracture modelling

This chapter has outlined the demonstration and validation of the Swegle Interaction Area (SIA) approach for applying damage (fracture model) in 1D (see Chapter 7 for a detailed explanation). The purpose of conducting 1D modelling was to primarily identify if the approach was capable of representing the level of damage in an experiment successfully, but also to identify a suitable model for developing the damage parameter over time. Several different methods for growing the inter-particle damage parameter have been established and investigated with mixed results.

Initially the validity of the approach was tested with a simple linear damage growth model that incremented the damage parameter from zero to one over a fixed number of time-steps (see Section 8.2 for a detailed explanation). Simulations of a simplified 1D copper plate impact experiment were conducted and plots of longitudinal stress at the rear of the target plate with time were generated to ascertain if damage was present in the model. The plots showed evidence of a reload signal which indicated that a new free surface was being opened in the correct region. This work concluded that the SIA approach was adequate for applying the concept of inter-particle damage.

After the success of the initial tests a version of the Cochran-Banner damage growth model (1977) that was developed for research by Mirkovic (2004) was selected and adapted for use within the SPH method and for inter-particle damage. The model computes the damage growth as a function of the change in inter-particle volume per Swegle Interaction Area, i.e. as a length-scale between two particles (see Section 8.4 for a detailed explanation). An Initial 1D simulation was conducted and compared to actual experimental data for a 1D copper plate impact test described by Panov (2006). The simulation showed promising results but the onset of damage was not rapid enough to fully capture the experimental material response. This led to an analysis of various inter-particle bond breaking criteria that failed bonds given that other bonds had also failed. Excellent results were observed for a particular criterion, i.e. a close match to the experimental data was observed. The concern was that the 1D simulations were too simplified to identify the true effect of failing multiple bonds within a neighbourhood.

An investigation was therefore conducted to attempt to improve the response of the Cochran-Banner damage growth model without any bond-breaking criteria active. Several modifications were considered from changes to the position of terms within the damage equations to a rewrite of the model to use direct strain or a length scale between a particle pair. None of the modifications were able to provide a response that was as good as the initial test or the successful bond breaking criterion. Finally a brief investigation into the sensitivity of the damage growth to the SPH smoothing length was conducted. This showed that the damage was unaffected by increasing smoothing length within a range of standard values (i.e. 1.2 to 1.4 times the initial inter-particle spacing).

It was decided that a sufficient investigation into the Cochran-Banner model had been conducted in 1D to attempt to extend the model to multiple dimensions. The initial configuration of the model outlined at the start of Section 8.4 was assessed to be the best candidate for this purpose.

The 1D copper plate simulation model was extended to 2D but the sensitivity of the existing material contact algorithm in the SPH code caused the motion of the particles at the boundaries to become erratic. An investigation into the contact parameters defined in the input was conducted, but a stable version of the simulation could not be obtained. Finally a stable simulation was achieved by constraining the particles in the y-direction. Initial tests of the standard Cochran-Banner damage growth model, i.e. without any bond-breaking criteria or modifications provided an excellent representation of the experimental data. Attempts were made to normalise the damage growth model so that the same response could be observed between the 1D and 2D simulations. The magnitude of the SIA was observed for multiple positions within the 1D and 2D neighbourhoods and it was concluded that the dependency of this parameter on the SPH kernel was causing the differences between the different dimensions. Typical values of the SIA with varying orders of magnitude were tested as constant values in the 1D and 2D simulations, but a single value that worked for both cases was not found. However the same response was achieved with different values for each dimension. Further tests were conducted with an Angular Bond Breaking criterion that compared the angle between the position vectors of a pair of particles whose bond had failed and the other neighbours in a neighbourhood to a damage angle. All the bonds between neighbours

that lie within the damage angle were broken. As expected this increased the onset of damage from the case without the criterion active, but its validity was unclear because the 2D case had already provided good results.

A 3D model of the copper plate impact experiment experienced the same contact problems as the 2D case and a stable simulation was only achieved by constraining the SPH particles in both the y and z-directions. Tests were conducted with both the natural evolution of the damage as described by the Cochran-Banner damage growth model and the manipulated damage growth via the Angular Bond Breaking criterion. Both cases did not achieve the correct reload signal, i.e. the onset of damage was too rapid. Despite this the 3D model displayed the opening of new free surfaces in the centre of the target plate which demonstrates the applicability of the method for applying damage.

Figure 8.5-1 shows all three of the of the responses achieved with the standard layout of the Cochran-Banner Damage Growth model coupled to the SIA approach for applying the damage (fracture model) in all three dimensions. Figure 8.5-2 shows the best representations of the experimental data that were achieved by the fracture modelling, which are the use of Bond Break Criterion 3 in 1D and the normalised Cochran-Banner damage growth model with $\left| A_{ij} \right|$ set to 0.00075cm^2 in 2D.
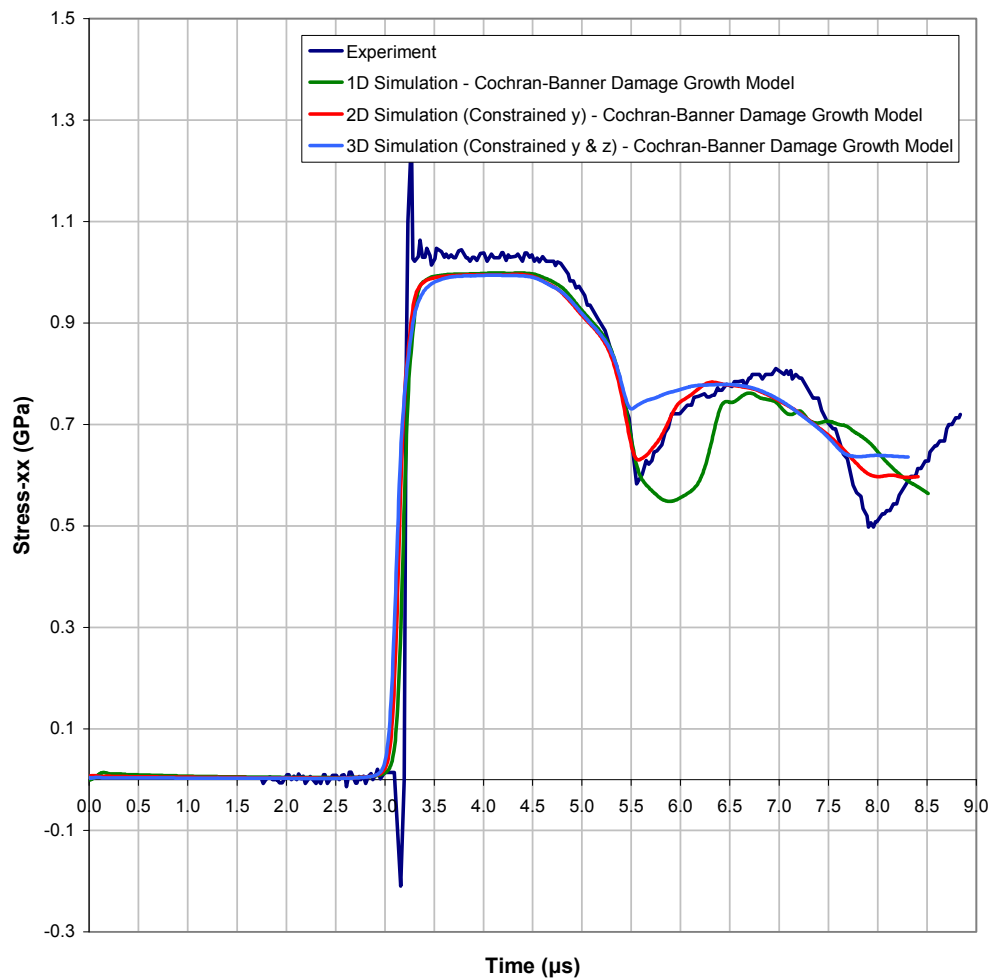
Figure 8.5-1: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. Simulation data is shown for the Cochran-Banner damage growth model applied to 1D, 2D and 3D models and is compared to the experimental data.

It can therefore be concluded that the SIA approach for applying the damage to inter-particle bonds, tracked in a Total Lagrangian SPH formulation is capable of describing the break-up of materials at high strain rate. The selection of an appropriate damage growth model is necessary to accurately develop the damage parameter over time and the Cochran-Banner damage growth model investigated for this work shows promising results. Some further work is required to remove the model's dependency on the dimensionality of the problem so that it can be made consistent in all dimensions. It should be noted that the method of applying damage is not limited to this specific damage growth model and therefore other models that are capable of using inter-particle

parameters to develop the damage could also be used. Further tests in multiple dimensions are required to fully identify the validity of the approach, although this will likely require some modifications to the current SPH contact algorithm so that unconstrained versions of 2D and 3D simulations of the Copper Plate impact experiment can be performed.
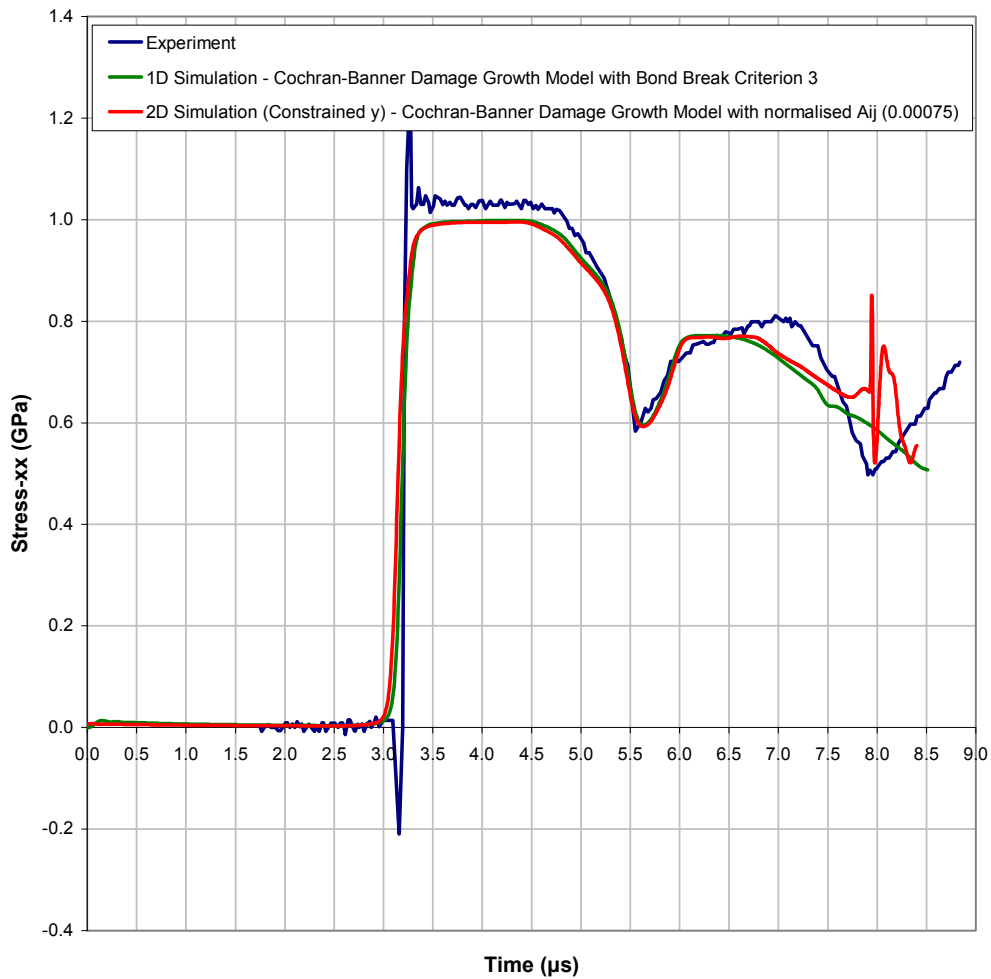
Figure 8.5-2: Longitudinal stress against time at the rear of the target plate for the Copper Plate Impact Experiment. Simulation data is shown for the best representations of the damage using versions of the Cochran-Banner damage growth model: 1D data is displayed for Bond Break Criterion 3 and 2D data is displayed for the magnitude of the SIA normalised to 0.00075cm³ in the damage increment calculation.

Since the completion of the research a material based mixed Eulerian/Lagrangian SPH description has been implemented into the MCM code at Cranfield University. Initial

Mock-Holt simulations with an Eulerian kernel used to describe the explosive and a Lagrangian kernel to describe the iron cylinder have shown promising results, although further work is required to correctly define the material contact. Once this is working correctly the damage description developed for this research can be tested, although such further developments are outside the scope of this research and therefore are not reported in this thesis.

This page is intentionally blank

9   Conclusions

The main objective of this research was to develop a fracture model for the initiation, growth and subsequent failure of metals at high strain rate within the SPH modelling framework. The research commenced with a description of the physics of the problem, but also provided a survey into the existing approaches documented in literature to identify the possible routes which could be explored in this body of research. The survey concluded that a 'local' description of the damage was more favourable for use with the SPH method as opposed to the 'global' methods, which make use of geometries to store crack data and are then interfaced to the continuum via some numerical mapping. The 'local' approach is interpreted in two ways: either SPH particles themselves are damaged directly and ultimately split apart on failure, or the interfaces between particles are manipulated and broken when damage reaches a critical value. The choice of the exact approach to take was dependent on a number of influencing factors.

Firstly, two formulations of the SPH method were available at the start of this research; these were the standard Eulerian SPH formulation and the Total Lagrangian SPH formulation. The Eulerian SPH method was the initial logical choice since the literature study (see Section 2.2) had also identified two explosively driven experiments for which Eulerian SPH captured the correct type of fragmentation behaviour. Modelling of these experiments required the ability to model extreme deformations and thus the Total Lagrangian formulation was unsuitable. Unfortunately the Eulerian SPH approach is subject to some numerical problems which often lead to unphysical fracture in the solution. Development of a fracture model requires a robust framework that is stable without the application of damage so that the specific effects of the failure model can be identified and quantified. This led to an investigation into stabilisation techniques for the Eulerian SPH method, thus satisfying the objective to explore further developments or enhancements to the existing Cranfield SPH code to facilitate the development and implementation of the fracture model.

Significant project time was spent investigating the Monaghan Repulsive Force correction and a technique which applied a partition of unity weighting to two SPH

approximations; one calculated with a local (or standard) neighbourhood and another that was calculated with a non-local (or extended) neighbourhood. Whilst the Monaghan Repulsive force was able to stabilise the code in 1D, attempts to make it work successfully for all cases in multiple dimensions were unsuccessful. The partition of unity approach could also not be made to work successfully.

Without a stable Eulerian approach, the decision was made to develop the fracture model with the Total Lagrangian SPH formulation. This led to the conclusion that an inter-particle approach to modelling failure was the best choice for the fracture model, because the Total Lagrangian method requires a fixed set of particle neighbourhoods and thus each inter-particle bond can be set as a material history parameter and damaged accordingly. This decision was backed up in the literature since most of the particle based approaches were used for fatigue type crack problems, which were able to include higher levels of complexity due to the slow nature of the crack development. The rapid development of multiple cracks leading to rapid fragmentation is complex to describe with splitting particles and updating neighbourhoods. The inter-particle approach allows for a more natural development of phenomena such as crack-joining and branching, since these occur when adjacent bonds are failed.

Along side the literature survey various additions were made to the SPH code to include two extra constitutive models (Steinberg-Guinan and Mechanical Threshold Stress) that were ultimately implemented to allow for detailed material modelling once the fracture model was completed, but also facilitated in the learning process of the SPH code structure and writing of the code itself. The explosively driven experiments required a method for modelling explosive detonation products and so the Prescribed Burn detonation model and the JWL equation of state were also implemented into the SPH code.

Since modelling of the explosively driven experiments had already commenced with the Eulerian approach prior to the decision to use the Total Lagrangian formulation for modelling damage, this work was continued and successful simulations were run with the Mock-Holt experiment up to 153 μs. This came after an investigation into mesh sensitivity which concluded that the best approach to modelling this experiment in SPH is to utilise a radial packing arrangement for the particles. 153 μs is considered to be

long enough for the material to have undergone fracture and therefore the model is in a condition where it can be easily adapted when future research extends the Cranfield SPH code to allow for coupled kernel formulations, i.e. the modelling of the iron cylinder with the Total Lagrangian formulation and modelling of the explosive with the Eulerian formulation.

The fracture model was developed by applying an existing concept in continuum damage mechanics in a novel manner. Classical continuum damage theory identifies an effective stress to apply damage, which is a function of the actual stress and some level of computed damage. It is accepted that in a real system, it is the cross-sectional area over which the stress acts that is reduced via the presence of micro-cracks and voids in the material. From the literature an effective area is defined by multiplying the SPH momentum equation by mass and then rearranging the terms such that it yields Newton's second law. This effective area is defined for each individual particle pair and the damage parameter is applied to this area directly. The area is reduced over time until total failure results in the area being set to zero. At this point the two particles are no longer numerically connected and a free surface is generated between them by also setting the gradient of the SPH kernel function to zero. Damage is tripped in the model when direct stress reaches a critical level and the damage parameter is computed via a damage growth model.

To demonstrate and validate the fracture model, two damage growth models were investigated and tests were performed in 1D using a copper plate impact simulation of an experiment for which data were readily available. A linear damage growth model was used to test the initial area-damage concept and the tests proved successful because evidence of a reload signal was generated in the results. The bulk of the investigation was performed with a modified version of the Cochran-Banner damage growth model which was updated to be relevant for inter-particle, multi-dimensional damage. The initial state of the model in 1D provided good results but did not accurately match the experimental data. Investigations into various enhancements provided mixed results, although the addition of a multiple bond-breaking criterion, which failed multiple bonds given that a particular bond had failed, was able to accurately represent the experimental data.

Limited fracture modelling assessments were performed with the 2D and 3D solvers, but the simulations had to be constrained to only allow movement along the x-axis because interference from the material contact algorithm prevented the copper plate simulation from running correctly. Unfortunately this meant that the multi-dimensional modelling was not a true representation and therefore it can only be considered to give an indication of performance of the fracture/damage growth model. The 2D modelling produced excellent results but the 3D model results showed that the onset of damage was too rapid, although it did show evidence of new free surfaces opening in the region of expected failure. Several attempts to normalise the damage growth model to be consistent in all three dimensions were unsuccessful and therefore further tests with unconstrained multi-dimensional models are required before firm assumptions can be made about the growth model's ability to develop damage in all dimensions.

The main conclusion of this research is that the interaction area approach for applying damage to inter-particle bonds, tracked in a Total Lagrangian framework is valid for describing the break-up of materials in the SPH method, which fulfils the main objective of this research. Further developments to the Cochran-Banner damage growth model are required to get this model working consistently in multi-dimensional problems. It should be noted however that the interaction area approach is not limited to this growth model since any model that can be written in terms of inter-particle variables could be used.

Ideally the fracture model needs to be tested with the Mock-Holt experiment, once enhancements to permit modelling of this experiment within the Total Lagrangian framework have been made. This will also require the development of a post-processing tool that can calculate the fragment mass distributions of the failed Mock-Holt metal cylinder so that comparisons to the experimental data can be made. It is also necessary to test the fracture model with more complex constitutive models such as Steinberg-Guinan and MTS (which were implemented as a part of this research) to see if the response is improved when more accurate methods for determining the material behaviour are used.

10 Reference List and Bibliography

10.1 Reference List

BALL, G., 2006. An explanation and value of the energy exponential coefficient in the Steinberg-Guinan constitutive model. UK: Atomic Weapons Establishment. (Private Communication).

BANERJEE, B., 2005a. An Evaluation of Plastic Flow Stress Models for the Simulation of High-Temperature and High-Strain-Rate Deformation of Metals. High-Strain-Rate Material Modeling edn. Department of Mechanical Engineering, University of Utah.

BANERJEE, B., 2005b. Taylor Impact Tests: Detailed Report. C-SAFE-CD-IR-05-0001. C-SAFE Internal Report.

BANERJEE, B., 2004. Material Point Method Simulations of Fragmenting Cylinders, *17th ASCE Engineering Mechanics Conference,* June 13-16th, 2004, http://www.eng.utah.edu/~banerjee/Papers/EM2004.pdf.

BANERJEE, B., GUILKEY, J.E., HARMAN, T.B., SCHMIDT, J.A. and MCMURTRY, P.A., 2005d. Simulation of Impact and Fragmentation with the Material Point Method, *11th International Conference on Fracture,* March 20-25th, 2005d, http://www.icf11.com.

BAŽANT, Z.P., 1984. Imbricate Continuum and its Variational Derivation. *Journal of Engineering Mechanics,* **110**(12), 1693-1712.

BELYTSCHKO, T., GUO, Y., LIU, W. K. and XIAO, S. P., 2000. A unified stability analysis of meshless particle methods. *International Journal for Numerical Methods in Engineering,* **48**(9), 1359-1400.

BELYTSCHKO, T. and BLACK, T., 1999. Elastic crack growth in finite elements with minimal remeshing. *International Journal for Numerical Methods in Engineering,* **45**(5), 601-620.

BELYTSCHKO, T., LU, Y.Y. and GU, L., 1994. Element-free galerkin methods. *International Journal for Numerical Methods in Engineering,* **37**, 229-256.

BELYTSCHKO, T., LU, Y.Y. and GU, L., 1995a. Crack propagation by element-free Galerkin methods. *Engineering Fracture Mechanics (USA),* **51**(2), 295-315.

BELYTSCHKO, T., LU, Y.Y., GU, L. and TABBARA, M., 1995b. Element-free galerkin methods for static and dynamic fracture. *International Journal of Solids and Structures,* **32**(17-18), 2547-2570.

BELYTSCHKO, T. and TABBARA, M., 1996. Dynamic fracture using element-free Galerkin methods. *International Journal for Numerical Methods in Engineering,* **39**(6), 923-938.

BENZ, W. and ASPHAUG, E., 1995. Simulations of brittle solids using smooth particle hydrodynamics. *Computer Physics Communications,* **87**(1-2), 253-265.

CAMPBELL, J., 1998. Lagrangian Hydrocode Modelling of Hypervelocity Impact on Spacecraft, Cranfield University - College of Aeronautics.

CAMPBELL, J., VIGNJEVIC, R. and LIBERSKY, L., 2000. A contact algorithm for smoothed particle hydrodynamics. *Computer Methods in Applied Mechanics and Engineering,* **184**(1), 49-65.

COCHRAN, S. and BANNER, D., 1977. Spall Studies in Uranium. *Journal of Applied Physics,* **48**(7), 2729-2737.

CULLIS, I., CHURCH, P., GREENWOOD, P., HUNTINGTON-THRESHER, W. and REYNOLDS, M., 2003. Advances in fracture algorithm development in GRIM. *Journal De Physique. IV : JP*, **110**, 359-364.

DAUX, C., MOËS, N., DOLBOW, J., SUKUMAR, N. and BELYTSCHKO, T., 2000. Arbitrary branched and intersecting cracks with the extended finite element method. *International Journal for Numerical Methods in Engineering,* **48**(12), 1741-1760.

DE VUYST, T.A.I., 2003. Hydrocode Modelling of Water Impact, Cranfield University - School of Engineering.

DE VUYST, T., VIGNJEVIC, R., BOURNE, N., CAMPBELL, J., 2002. Evaluation of the SPH Method for the Modelling of Spall in Anisotropic Alloys, *5th International Conference on Dynamics and Control of Systems and Structures in Space,* July 14-18th, 2002, Cambridge, UK.

DOBRATZ, B.M. and CRAWFORD, P.C., 1985. LLNL Explosives Handbook Properties of Chemical Explosives and Explosive Stimulants. UCRL-52997 / DE85 015961. University of California, Livermore, California, 94550: Lawrence Livermore National Laboratory.

DRUCKER, D.C., 1959. A Definition of Stable Inelastic Material. *Journal of Applied Mechanics,* **26**(1), 101-106.

FOLLANSBEE, P.S. and KOCKS, U.F., 1988. Constitutive description of the deformation of copper based on the use of the mechanical threshold stress as an internal state variable. *Acta Metallurgica,* **36**(1), 81-93.

FRENCH, I.E. and WEINRICH, P.F., 1977. The shear mode of ductile fracture in commercial copper. *Scripta Metallurgica,* **11**, 965-968.

GINGOLD, R.A. and MONAGHAN, J.J., 1977. Smoothed particle hydrodynamics - Theory and application to non-spherical stars. *Royal Astronomical Society, Monthly Notices,* **181**, 375-389.

GIROUX, E.D., 1971. HEMP User's Manual. UCRL-51079. University of California, Livermore, California, 94550: Lawrence Livermore National Laboratory.

GOURMA, M., 2003. Towards Better Understanding of the SPH (Full) Method, Cranfield University - School of Engineering.

GRAVOUIL, A., MOËS, N. and BELYTSCHKO, T., 2002. Non-planar 3D crack growth by the extended finite element and level sets - Part II: Level set update. *International Journal for Numerical Methods in Engineering,* **53**(11), 2569-2586.

HALLQUIST, J.O., 1998. LS-DYNA Theoretical Manual. United States of America: Livermore Software Technology Corporation.

HANSON, K.M., 2004. Introduction to PTW. Los Alamos Report LA-UR-04-0305.

HARLOW, F.H., 1964. The Particle-in-cell computing method for fluid dynamics. In: ALDER, B., FERNBACH, S. and ROTENBERG, M. eds, Methods for Computational Physics, Volume 3, New York: Academic Press, pp. 319-343.

HOCKNEY, R.W. and EASTWOOD, J.W., 1981. Computer Simulation Using Particles. USA: McGraw-Hill.

JOHNSON, J.N., 1981. Dynamic Fracture and Spallation in Ductile solids. *Journal of Applied Physics,* **52**(4), 2812-2825.

JOHNSON, J.N. and ADDESSIO, F.L., 1988. Tensile plasticity and ductile fracture. *Journal of Applied Physics,* **64**(12), 6699-6712.

JOHNSON, G.R., 1996. Artificial viscosity effects for SPH impact computations. *International Journal of Impact Engineering,* **18**(5), 477-488.

JOHNSON, G.R. and BEISSEL, S.R., 1996. Normalised smoothing functions for SPH impact computations. *International Journal for Numerical Methods in Engineering,* **39**, 2725-2741.

JOHNSON, G.R. and COOK, W.H., 1983. A Constitutive Model and Data for Metals Subjected to Large Strains, High Strain Rates and High Temperatures, *7th International Symposium on Ballistics,* 1983, pp541-547.

JOHNSON, G.R. and COOK, W.H., 1985. Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures. *Engineering Fracture Mechanics,* **21**(1), 31-48.

KACHANOV, L.M., 1958. Time of the rupture process under creep conditions. *Izv. Akad. Nauk., S.S.R., Otd Tech Nauk,* **8**(8), 26-31.

KINSLOW, R., 1970. High-Velocity Impact Phenomena. United Kingdom edn. London: Academic Press, Inc. (London) Ltd.

KREYSZIG, E., 1993. Advanced Engineering Mathematics, 7th Edition. USA: John Wiley and Sons, Inc.

KRYSL, P. and BELYTSCHKO, T., 1999. The element free Galerkin method for dynamic propagation of arbitrary 3-D cracks. *International Journal for Numerical Methods in Engineering,* **44**(6), 767-800.

LEMAITRE, J., 1985. A continuous damage mechanics model for ductile fracture. *Journal of Engineering Materials and Technology, Transactions of the ASME,* **107**(1), 83-89.

LIBERSKY, L., 2006. SPH modelling of an RDEC 40mm bullet. USA: Los Alamos National Laboratory. (Private Communication).

LIBERSKY, L.D. and PETSCHEK, A.G., 1991. Smooth particle hydrodynamics with strength of materials. In: H.E. TREASE, M.J. FRITTS and W.P. CROWLEY, eds, Advances in the Free-Lagrange Method Including Contributions on Adaptive Gridding and the Smooth Particle Hydrodynamics Method. Berlin / Heidelberg: Springer, pp. 248-257.

LIBERSKY, L.D., PETSCHEK, A.G., CARNEY, T.C., HIPP, J.R. and ALLAHDADI, F.A., 1993. High strain lagrangian hydrodynamics a three-dimensional SPH code for dynamic material response. *Journal of Computational Physics,* **109**(1), 67-75.

LIN, J.I., 2004. DYNA3D: A Nonlinear, Explicit, Three-Dimensional Finite Element Code for Solid and Structural Mechanics - User Manual. United States of America: Laurence Livermore National Laboratory.

LIU, W.K., CHEN Y., JUN, S., CHEN, J.S., BELYTSCHKO, T., PAN, C., URAS, R.A. and CHANG, C.T., 1996. Overview and applications of the reproducing kernel particle methods. *Archives of Computational Methods in Engineering,* **3**(1), 3-80.

LIU, W.K., JUN, S., LI, L., ADEE, J. and BELYTSCHKO, T., 1995a. Reproducing kernel particle methods for structural dynamics. *International Journal for Numerical Methods in Engineering,* **38**, 1655-1679.

LIU, W.K., JUN, S. and ZHANG, Y.F., 1995b. Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids,* **20**(8-9), 1081-1106.

LIU, W., LI, S. and BELYTSCHKO, T., 1997. Moving least-square reproducing kernel methods (I) methodology and convergence. *Computer Methods in Applied Mechanics and Engineering,* **143**(1-2), 113-154.

LIU, G.R. and LIU, M.B., 2003. Smoothed Particle Hydrodynamics A Meshfree Particle Method. Singapore: World Scientific Publishing Co. Pte. Ltd.

LU, Y.Y., BELYTSCHKO, T. and TABBARA, M., 1995. Element-free Galerkin method for wave propagation and dynamic fracture. *Computer Methods in Applied Mechanics and Engineering,* **126**(1-2), 131-153.

LUCY, L.B., 1977. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal,* **82**, 1013-1024. Dec.

MALVERN, L.E., 1969. Introduction to the Mechanics of a Continuous Medium. USA: Prentice-Hall Inc.

MAUDLIN, P.J., GRAY III, G.T., CADY, C.M. and KASCHNER, G.C., 1999. High-Rate Material Modelling and Validation Using the Taylor Cylinder Impact Test. *The Royal Society,* **357**, 1707-1729.

MIRKOVIC, J., 2004. Modelling of Nonlinear Behaviour of Metallic Structure Components, Cranfield University, School of Engineering.

MOCK, W.JR., and HOLT, W.H., 1983. Fragmentation Behavior of Armco Iron and HF-1 Steel Explosive-Filled Cylinders. *Journal of Applied Physics,* **54**(5), 2344-2351.

MOËS, N., DOLBOW, J. and BELYTSCHKO, T., 1999. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering,* **46**(1), 131-150.

MOËS, N., GRAVOUIL, A. and BELYTSCHKO, T., 2002. Non-planar 3D crack growth by the extended finite element and level sets - Part I: Mechanical model. *International Journal for Numerical Methods in Engineering,* **53**(11), 2549-2568.

MONAGHAN, J.J., 2005. Smoothed particle hydrodynamics. *Reports on Progress in Physics,* **68**(8), 1703-1759.

MONAGHAN, J.J., 2000. SPH without a Tensile Instability. *Journal of Computational Physics,* **159**, 290-311.

MONAGHAN, J.J., 1982. Why particle methods work. *Society for Industrial and Applied Mathematics Journal on Scientific and Statistical Computing,* **3**(4), 422-433.

MONAGHAN, J.J. and GINGOLD, R.A., 1983. Shock simulation by the particle method SPH. *Journal of Computational Physics,* **52**, 374-389.

MONAGHAN, J.J. and LATTANZIO, J.C., 1985. A refined particle method for astrophysical problems. *Astronomy and Astrophysics,* **149**(1), 135-143.

NEEDLEMAN, A., 1987. Continuum model for void nucleation by inclusion debonding. *Journal of Applied Mechanics, Transactions ASME,* **54**(3), 525-531.

ORTIZ, M. and PANDOLFI, A., 1999. Finite-deformation irreversible cohesive elements for three-dimensional crack-propagation analysis. *International Journal for Numerical Methods in Engineering,* **44**(9), 1267-1282.

PANDOLFI, A., KRYSL, P. and ORTIZ, M., 1999. Finite element simulation of ring expansion and fragmentation: The capturing of length and time scales through cohesive models of fracture. *International Journal of Fracture,* **95**(1-4), 279-297.

PANOV, V., 2006. Modelling of behaviour of metals at high strain rate, Cranfield University, School of Engineering.

PHILIPS, A. and SIERAKOWSKI, R.L., 1965. On the concept of the yield surface. *Acta Mechanica*, **1**, 29-35.

PRESTON, D.L., TONKS, D.L. and WALLACE, D.C., 2003. Model of plastic deformation for extreme loading conditions. *Journal of Applied Physics,* **93**(1), 211-220.

RABCZUK, T. and BELYTSCHKO, T., 2007. A three-dimensional large deformation meshfree method for arbitrary evolving cracks. *Computer Methods in Applied Mechanics and Engineering,* **196**(29-30), 2777-2799.

RABCZUK, T. and BELYTSCHKO, T., 2004. Cracking particles: A simplified meshfree method for arbitrary evolving cracks. *International Journal for Numerical Methods in Engineering,* **61**(13), 2316-2343.

RABCZUK, T., BELYTSCHKO, T. and XIAO, S.P., 2004. Stable particle methods based on Lagrangian kernels. *Computer Methods in Applied Mechanics and Engineering,* **193**(12-14), 1035-1063.

RANDLES, P.W., CARNEY, T.C., LIBERSKY, L.D., RENICK, J.D. and PETSCHEK, A.G., 1994. Calculation of oblique impact and fracture of tungsten cubes using smoothed particle hydrodynamics. *International Journal of Impact Engineering (UK),* **17**(4-6), 661-672.

RANDLES, P.W. and LIBERSKY, L.D., 1996. Smoothed particle hydrodynamics: Some recent improvements and applications. *Computer Methods in Applied Mechanics and Engineering,* **139**(1-4), 375-408.

RANDLES, P.W., LIBERSKY, L.D., CARNEY, T.C. and SANDSTROM, F.W., 1996. SPH Simulation of Fragmentation in the MK82 Bomb. *American Institute of Physics,* , 331-334.

REMMERS, J.J.C., DE BORST, R. and NEEDLEMAN, A., 2003. A cohesive segments method for the simulation of crack growth. *Computational Mechanics,* **31**(1-2 SPEC.), 69-77.

REVELES, J.R., 2007. Development of a Total Lagrangian SPH code for the simulation of solids under impact loading, Cranfield University, School of Engineering.

ROGERS, B.D., 2010. Introduction to Smoothed Particle Hydrodynamics (SPH). *Lecture notes for Advanced CFD Course, School of MACE, University of Manchester.*

ROYCE, E.B. and LAWRENCE LIVERMORE NATIONAL LAB., CA, 1971. Gray: A three phase equation of state for metals (Three-phase equation of state for metals). UCRL-51121; Pagination 49P.

RUDNICKI, J.W. and RICE, J.R., 1975. Conditions for the localization of deformation in pressure-sensitive dilatant materials. *Journal of the Mechanics and Physics of Solids,* **23**(6), 371-394.

SIMONSEN, B.C. and LI, S., 2004. Mesh-free simulation of ductile fracture. *International Journal for Numerical Methods in Engineering,* **60**(8), 1425-1450.

SLATE, P.M.B., BILLINGS, M.J.W. and FULLER, P.J.A., 1967. The Rupture Behaviour of Metals at High Strain Rates. *Journal of the Institute of Metals,* **95**(8), 244-251.

STEINBERG, D.J., 1996. *Equation of State and Strength Properties of Selected Materials.* UCRL-MA-106439. USA: Laurence Livermore National Laboratory.

STEINBERG, D.J., COCHRAN, S.G. and GUINAN, M.W., 1980. A Constitutive Model for Metals Applicable at High-Strain Rate. *Journal of Applied Physics,* **51**(3), 1498-1504.

STOLARSKA, M., CHOPP, D.L., MOËS, N. and BELYTSCHKO, T., 2001. Modelling crack growth by level sets in the extended finite element method. *International Journal for Numerical Methods in Engineering,* **51**(8), 943-960.

SUKUMAR, N., CHOPP, D.L. and MORAN, B., 2003. Extended finite element method and fast marching method for three-dimensional fatigue crack propagation. *Engineering Fracture Mechanics,* **70**(1), 29-48.

SUKUMAR, N., MOES, N., MORAN, B. and BELYTSCHKO, T., 2000. Extended finite element method for three-dimensional crack modelling. *International Journal for Numerical Methods in Engineering,* **48**(11), 1549-1570.

SULSKY, D., CHEN, Z. and SCHREYER, H.L., 1994. A particle method for history dependent materials. *Computer Methods in Applied Mechanics and Engineering*, **118**, 179-196.

SWEGLE, J.W., 2000. Conservation of momentum and tensile instability in particle methods. SAND2000-1223. USA: Sandia National Laboratories.

SWEGLE, J.W., ATTAWAY, S.W., HEINSTEIN, M.W., MELLO, F.J. and HICKS, D.L., 1994. An analysis of smoothed particle hydrodynamics. SAND93-2513. USA: Sandia National Laboratories.

SWEGLE, J.W., ATTAWAY, S.W. and HICKS, D.L.W., 1995. Smoothed particle hydrodynamics stability analysis. *Journal of Computational Physics,* **116**(1), 123-134.

SZICHMAN, H. and KRUMBEIN, A.D., 1986. Semiempirical calculations of the zero-degree-Kelvin isotherm in metals. *Physical Review A-General Physics, 3rd Series,* **33**, 706-714. Jan.

TVERGAARD, V. and NEEDLEMAN, A., 1984. Analysis of the cup-cone fracture in a round tensile bar. *Acta Metallurgica,* **32**(1), 157-169.

TVERGAARD, V., 1982. On localization in ductile materials containing spherical voids. *International Journal of Fracture,* **18**(4), 237-252.

TVERGAARD, V., 1981. Influence of voids on shear band instabilities under plane strain conditions. *International Journal of Fracture,* **17**(4), 389-407.

VENTURA, G., XU, J.X. and BELYTSCHKO, T., 2002. A vector level set method and new discontinuity approximations for crack growth by EFG. *International Journal for Numerical Methods in Engineering,* **54**(6), 923-944.

VIANA, S.A., RODGER, D. and LAI, H.C., 2007. Overview of meshless methods. *International Compumag Society Newsletter,* **14**(2).

VIGNJEVIC, R., CAMPBELL, J. and LIBERSKY, L., 2000. A treatment of zero-energy modes in the smoothed particle hydrodynamics method. *Computer Methods in Applied Mechanics and Engineering,* **184**(1), 67-85.

VIGNJEVIC, R., DE VUYST, T., CAMPBELL, J., 2002. Modelling of a Hydrodynamic Ram using Smoothed Particle Hydrodynamics, *5th International Conference on Dynamics and Control of Systems and Structures in Space,* July 14-18th, 2002, Cambridge, UK.

VIGNJEVIC, R., REVELES, J.R. and CAMPBELL, J., 2006. SPH in a total lagrangian formalism. *CMES - Computer Modeling in Engineering and Sciences,* **14**(3), 181-198.

WALLACE, D.C., 1981. Irreversible thermodynamics of overdriven shocks in solids. *Physical Review B (Condensed Matter),* **24**(10), 5597-5606.

WENDLAND, H., 1995. Piecewiese polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mechanics,* **4**(1), 389-396.

ZUKAS, J.A. (ed.), 1990. High Velocity Impact Dynamics. USA: John Wiley and Sons, Inc.

10.2 Bibliography

ANDERSON JR., C.E., CHOCRON, I.S. and NICHOLLS, A.E., 2006. Damage modeling for Taylor impact simulations, *8th International Conference on Mechanical and Physical Behaviour of Materials under Dynamic Loading,* 11 September 2006 through 15 September 2006, pp331-337.

ANDERSON JR., C.E., PREDEBON, W.W. and KARPP, R.R., 1985. Computational modeling of explosive-filled cylinders.

BELYTSCHKO, T. and XIAO, S., 2001. Stability analysis of particle methods with corrected derivatives. *USACM: Sixth US National Congress on Computational Mechanics Abstracts*.

BLAZYNSKI, T.Z., ed, 1987. Materials at High Strain Rates. Great Britain: Elsevier Applied Science Publishers.

BONET, J. and RODRÍGUEZ-PAZ, M.X., 2005. Hamiltonian formulation of the variable-h SPH equations. *Journal of Computational Physics,* **209**(2), 541-558.

BORESI, A.P. and SCHMIDT, R.J., 2003. Advanced Mechanics of Materials. 6 edn. United States of America: John Wiley and Sons, Inc.

BRACKBILL, J.U., 2005. Particle methods. *International Journal for Numerical Methods in Fluids,* **47**(8-9), 693-705.

BROBERG, K.B., 1999. Cracks and Fracture. Great Britain: Academic Press.

BUCHANAN, G.R., 1988. Mechanics of Materials. United States of America: Saunders College Publishing.

CHATELAIN, P., COTTET, G. and KOUMOUTSAKOS, P., 2007. Particle mesh hydrodynamics for astrophysics simulations. *International Journal of Modern Physics C,* **18**(4), 610-618.

CHEN, J.S., PAN, C., WU, C.T. and LIU, W.K., 1996. Reproducing kernel particle methods for large deformation analysis of non-linear structures. *Computer Methods in Applied Mechanics and Engineering,* **139**, 195-227.

CHEN, S.R., MAUDLIN, P.J. and GRAY III, G.T., 1999. Constitutive Behaviour of Model FCC, BCC, and HCP Metals: Experiments, Modeling and Validation. New Mexico: Los Alamos National Laboratory.

CHHABILDAS, L.C., THORNHILL, T.F., REINHART, W.D., KIPP, M.E., REEDAL, D.R., WILSON, L.T. and GRADY, D.E., 2001. Fracture resistant properties of AerMet steels. *International Journal of Impact Engineering (UK),* **26**(1-10), 77-91.

CHING, E.S.C., 2000. Multifractality of mass distribution in fragmentation. *Physica A: Statistical Mechanics and its Applications,* **288**(1-4), 402-408.

CHING, E.S.C., YIU, Y.Y. and LO, K.F., 1999. Energy dependence of mass distributions in fragmentation. *Physica A: Statistical Mechanics and its Applications,* **265**(1-2), 119-128.

CHURCH, P.D., MACMAHON, J., SOFTLEY, I. and CAMERON, C., 2000. Simulation of explosive fracture cylinder test using DYNA. *Journal De Physique. IV: JP,* **10**(9), 391-396.

DANIAN, C., YUYING, Y., ZHIHUA, Y., HUANRAN, W., GUOQING, L. and SHUGANG, X., 2005. A modified Cochran–Banner spall model. *International Journal of Impact Engineering,* **31**, 1106-1118.

DE VUYST, T., VIGNJEVIC, R. and CAMPBELL, J.C., 2005. Coupling between meshless and finite element methods. *International Journal of Impact Engineering,* **31**(8), 1054-1064.

EDWARDS, M., 2006. Properties of metals at high rates of strain. *Materials Science and Technology,* **22**(4), 453-462.

ELLERO, M., SERRANO, M. and ESPAÑOL, P., 2007. Incompressible smoothed particle hydrodynamics. *Journal of Computational Physics,* **226**(2), 1731-1752.

ELLIS, T.M.R., PHILIPS, I.R. and LAHEY, T.M., 1994. Fortran 90 Programming. England: Addison-Wesley Publishers Ltd.

GINGOLD, R.A. and MONAGHAN, J.J., 1982. Kernel estimates as a basis for general partice methods in hydrodynamics. *Journal of Computational Physics,* **46**, 429-453.

GOLD, V.M. and BAKER, E.L., 2008. A model for fracture of explosively driven metal shells. *Engineering Fracture Mechanics,* **75**, 275-289.

GRADY, D.E., 1988. The Spall Strength of Condensed Matter. *Journal Mech. Phys. Solids,* **36**(3), 353-384.

GRAY, J.P. and MONAGHAN, J.J., 2004. Numerical modelling of stress fields and fracture around magma chambers. *Journal of Volcanology and Geothermal Research,* **135**(3), 259-283.

GRAY, J.P., MONAGHAN, J.J. and SWIFT, R.P., 2001. SPH elastic dynamics. *Computer Methods in Applied Mechanics and Engineering,* **190**(49-50), 6641-6662.

HOGGATT, C.R. and RECHT, R.F., 1969. Stress-Strain Data obtained at High Rates using an Expanding Ring. *Exp Mech,* **9**(10), 441-448.

HOGGATT, C.R. and RECHT, R.F., 1968. Fracture behavior of tubular bombs.

HOLT, W.H., MOCK, J., W., ZERILLI, F.J. and CLARKE, J.B., 1994. Experimental and Computational Study of the Impact Deformation of Titanium Taylor Cylinder Specimens. *Mechanics of Materials,* **17**, 195-201.

HOOVER, W.G., HOOVER, C.G. and MERRITT, E.C., 2004. Smooth-particle applied mechanics: Conservation of angular momentum with tensile stability and velocity averaging. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics,* **69**(1 2), 167021-1670210.

JOHNSON, J.N., GRAY III, G.T. and BOURNE, N.K., 1999. Effect of Pulse Duration and Strain Rate on Incipient Spall Fracture in Copper. *Journal of Applied Physics,* **86**(9), 4892-4901.

JUANICOTENA, A. and LLORCA, F., 1997. Expanding ring test for the study of elastoplastic behaviour in the fast deformation range. Application to copper and steel Mars 190, *5th International Conference on Mechanical and Physical Behaviour of Materials Under Dynamic Loading, 22-26 Sept. 1997,* 08/ 1997, Editions de Physique pp541-6.

KATSURAGI, H., IHARA, S. and HONJO, H., 2005. Explosive fragmentation of a thin ceramic tube using pulsed power. *Physical Review Letters,* **95**(9), 1-8.

KAWASHIMA, Y., SAKAI, Y. and YAMAGATA, N., 2006. Large deformation analysis by smoothed particle hydrodynamics, *ASME PVP2006/ICPVT-11 Conference,* 23 July 2006 through 27 July 2006.

LI, S. and LIU, W.K., 2002. Meshfree and particle methods and their applications. *Applied Mechanics Reviews,* **55**(1), 1-34.

LIU, W.K., CHEN, Y., URAS, R.A. and CHANG, C.T., 1996. Generalized multiple scale reproducing kernel particle methods. *Computer Methods in Applied Mechanics and Engineering,* **139**, 91-157.

LIBERSKY, L.D., RANDLES, P.W., CARNEY, T.C. and DICKINSON, D.L., 1997. Recent improvements in SPH modeling of hypervelocity impact. *International Journal of Impact Engineering (UK),* **20**(6-10), 525-532.

LIU, Z.S., SWADDIWUDHIPONG, S. and KOH, C.G., 2004. HIGH velocity impact dynamic response of structures using SPH method. *International Journal of Computational Engineering Science,* **5**(2), 315-326.

LIVERMORE SOFTWARE TECHNOLOGY CORPORATION, 2004. LS-DYNA Keyword User's Manual (Update) Version 970 / Rev 5434. United States of America: Livermore Software Technology Corporation.

LIVERMORE SOFTWARE TECHNOLOGY CORPORATION, 2003. LS-DYNA Keyword User's Manual Version 970. United States of America: Livermore Software Technology Corporation.

MAUDLIN, P.J., DAVIDSON, R.F. and HENNINGER, R.J., 1990. Implementation and assessment of the mechanical-threshold-stress model using the EPIC2 and PINON computer codes. DE90-017855; LA-11895-MS.

MELEÁN, Y., SIGALOTTI, L.D.G. and HASMY, A., 2004. On the SPH tensile instability in forming viscous liquid drops. *Computer Physics Communications,* **157**(3), 191-200.

MONAGHAN, J.J., 1992. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics,* **30**(1), 543-574.

MONAGHAN, J.J., 1988. An introduction to SPH. *Computer Physics Communications,* **48**, 89-96.

NEMES, J.A. and RANDLES, P.W., 1994. Constitutive modeling of high strain-rate deformation and spall fracture of graphite/peek composites. *Mechanics of Materials,* **19**(1), 1-14.

OGER, G., DORING, M., ALESSANDRINI, B. and FERRANT, P., 2007. An improved SPH method: Towards higher order convergence. *Journal of Computational Physics,* **225**(2), 1472-1492.

OLIVE, F., NICAUD, A., MARILLEAU, J. and LOICHOT, R., 1979. Rupture behaviour of metals in explosive expansion.

PARSHIKOV, A.N. and MEDIN, S.A., 2002. Smoothed particle hydrodynamics using interparticle contact algorithms. *Journal of Computational Physics,* **180**(1), 358-382.

PERZYNA, P., 1998. Constitutive Modelling of Dissipative Solids: CISM Courses and Lectures No. 386. In: P. PERZYNA, ed, Localization and fracture phenomena in inelastic solids. New York: Springer Wien, pp. 99-241.

PHILLIPS, E.M. and PUGH, D.S., 2005. How to Get a PhD. 4 edn. England: Open University Press.

POPOV, E.P., 1976. Mechanics of Materials. United States of America: Prentice-Hall, Inc.

QINGDONG, D., BAYI, H., CHANGSHENG, H. and HAIBO, H., 1994. Expansion and fracture of AISI 1045 steel explosive-filled cylinders. *Journal De Physique. IV: JP,* **4**(8).

RABCZUK, T. and BELYTSCHKO, T., 2006. Application of particle methods to static fracture of reinforced concrete structures. *International Journal of Fracture,* **137**(1-4), 19-49.

RABCZUK, T. and EIBL, J., 2003. Simulation of high velocity concrete fragmentation using SPH/MLSPH. *International Journal for Numerical Methods in Engineering,* **56**(10), 1421-1444.

RABCZUK, T. and ZI, G., 2007. A meshfree method based on the local partition of unity for cohesive cracks. *Computational Mechanics,* **39**(6), 743-760.

RAFTENBERG, M.N., 1989. Response of Steel Disks to Explosive Loading. *Computers and Structures,* **33**(3), 637-653.

RAJENDRAN, A.M., DIETENBERGER, M.A. and GROVE, D.J., 1989. A Void Growth-Based Failure Model to Describe Spallation. *Journal of Applied Physics,* **65**(4), 1521-1527.

RANDLES, P.W., CARNEY, T.C. and LIBERSKY, L.D., 1995. Continuum dynamical simulations of bomb fragmentation. *Proc. 15th Int. Ballistics. Symp.,* **2**, 289-296.

RANDLES, P.W. and LIBERSKY, L.D., 2000. Normalized SPH with stress points. *International Journal for Numerical Methods in Engineering,* **48**(10), 1445-1462.

ROSENBERG, Z., LUTTWAK, G., YESHURUN, Y. and PARTOM, Y., 1983. Spall Studies of Differently Treated 2024A1 Specimens. *Journal of Applied Physics,* **54**(5), 2147-2152.

SCHEY, H.M., 2005. Div Grad Curl and All That. 4 edn. United States of America: W. W. Norton and Company.

SIGALOTTI, L.D.G. and LÓPEZ, H., 2008. Adaptive kernel estimation and SPH tensile instability. *Computers and Mathematics with Applications,* **55**(1), 23-50.

SIGALOTTI, L.D.G., LÓPEZ, H., DONOSO, A., SIRA, E. and KLAPP, J., 2006. A shock-capturing SPH scheme based on adaptive kernel estimation. *Journal of Computational Physics,* **212**(1), 124-149.

SIL'VESTROV, V.V., 2004. Application of the gilvarry distribution to the statistical description of fragmentation of solids under dynamic loading. *Combustion, Explosion and Shock Waves,* **40**(2), 225-237.

SIL'VESTROV, V.V., 2003. Statistics of impact fragmentation of a steel sphere. *International Journal of Impact Engineering,* **29**(1-10), 659-670.

STEINBERG, D.J. and LUND, C.M., 1989. A constitutive model for strain rates from 10^-4 to 10^6 s^-1. *Journal of Applied Physics,* **65**(4), 1528-1533.

STRONGE, W.J., XIAOQING, M. and LANTING, Z., 1989. Fragmentation of explosively expanded steel cylinders. *International Journal of Mechanical Sciences,* **31**(11/12), 811-823

SUKUMAR, N., SROLOVITZ, D.J., BAKER, T.J. and PRÉVOST, J., 2003. Brittle fracture in polycrystalline microstructures with the extended finite element method. *International Journal for Numerical Methods in Engineering,* **56**(14), 2015-2037.

SUNWOO, A.J., BECKER, R., GOTO, D.M., ORZECHOWSKI, T.J., SPRINGER, H.K., SYN, C.K. and ZHOU, J., 2006. Adiabatic shear band formation in explosively driven Fe-Ni-Co alloy cylinders. *Scripta Materialia,* **55**(3), 247-250.

TAYLOR, G.I., 1963. The fragmentation of tubular bombs.

VIGNJEVIC, R., BOURNE, N.K., MILLETT, J.C.F. and DE VUYST, T., 2002. Effects of Orientation on the Strength of the Aluminium Alloy 7010-T6 During Shock Loading: Experiment and Simulation. *Journal of Applied Physics,* **92**(8), 1-7.

VIGNJEVIC, R., DE VUYST, T. and CAMPBELL, J.C., 2006. A frictionless contact algorithm for meshless methods. *CMES - Computer Modeling in Engineering and Sciences,* **13**(1), 35-47.

WINTER, R.E., 1979. Measurement of Fracture Strain at High Strain Rates, *The Institute of Physics Conference Ser. No 47: Chapter 1,* 1979, pp81-89.

XIAOYU, J., YONGCHI, L., XICHENG, H. and YIHUI, Y., 2007. Smoothed Particle Hydrodynamics (SPH) method for stress wave propagation problems. *International Journal of Computational Methods in Engineering Science and Mechanics,* **8**(4), 195-200.

YANG, Z.Y., YANG, D.M. and WANG, F.C., 2001. Deformation and fragmentation of explosively loaded steel tubes. *Cailiao Kexue yu Gongyi/Material Science and Technology,* **9**(1), 42-46.

YELLUP, J.M., 1984. The Computer Simulation of an Explosive Test Rig to Determine the Spall Strength of Metals. *International Journal of Impact Engineering,* **2**(2), 151-167.

ZHANG, L., JIN, X. and HE, H., 1999. Prediction of fragment number and size distribution in dynamic fracture. *Journal of Physics D: Applied Physics,* **32**(5), 612-615.

ZHOU, F., MOLINARI, J.-. and RAMESH, K.T., 2006. Characteristic fragment size distributions in dynamic fragmentation. *Applied Physics Letters,* **88**(26),.

ZI, G., RABCZUK, T. and WALL, W., 2007. Extended meshfree methods without branch enrichment for cohesive cracks. *Computational Mechanics,* **40**(2), 367-382.

ZI, G., SONG, J., BUDYN, E., LEE, S. and BELYTSCHKO, T., 2004. A method for growing multiple cracks without remeshing and its application to fatigue crack growth. *Modeling and Simulation in Materials Science and Engineering,* **12**(5), 901-915.

Annex 1 – Explosion Model Verification: Supporting Graphs

This annex contains two graphs which support the verification of the explosion model which was implemented into MCM as a part of this research.
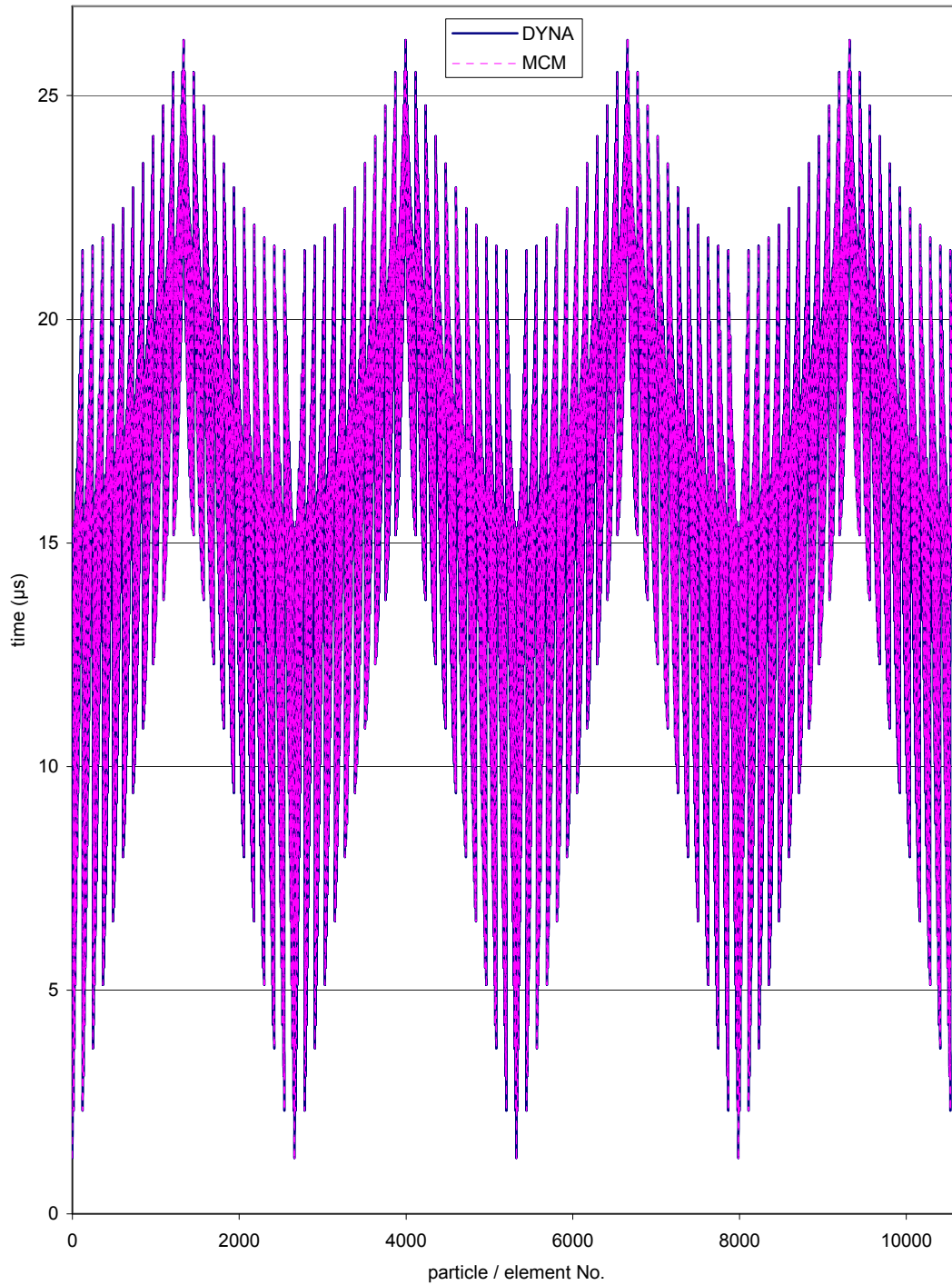
Figure AN1-1: Lighting times for particles/elements in MCM and LLNL-DYNA3D for the explosion model verification simulation.
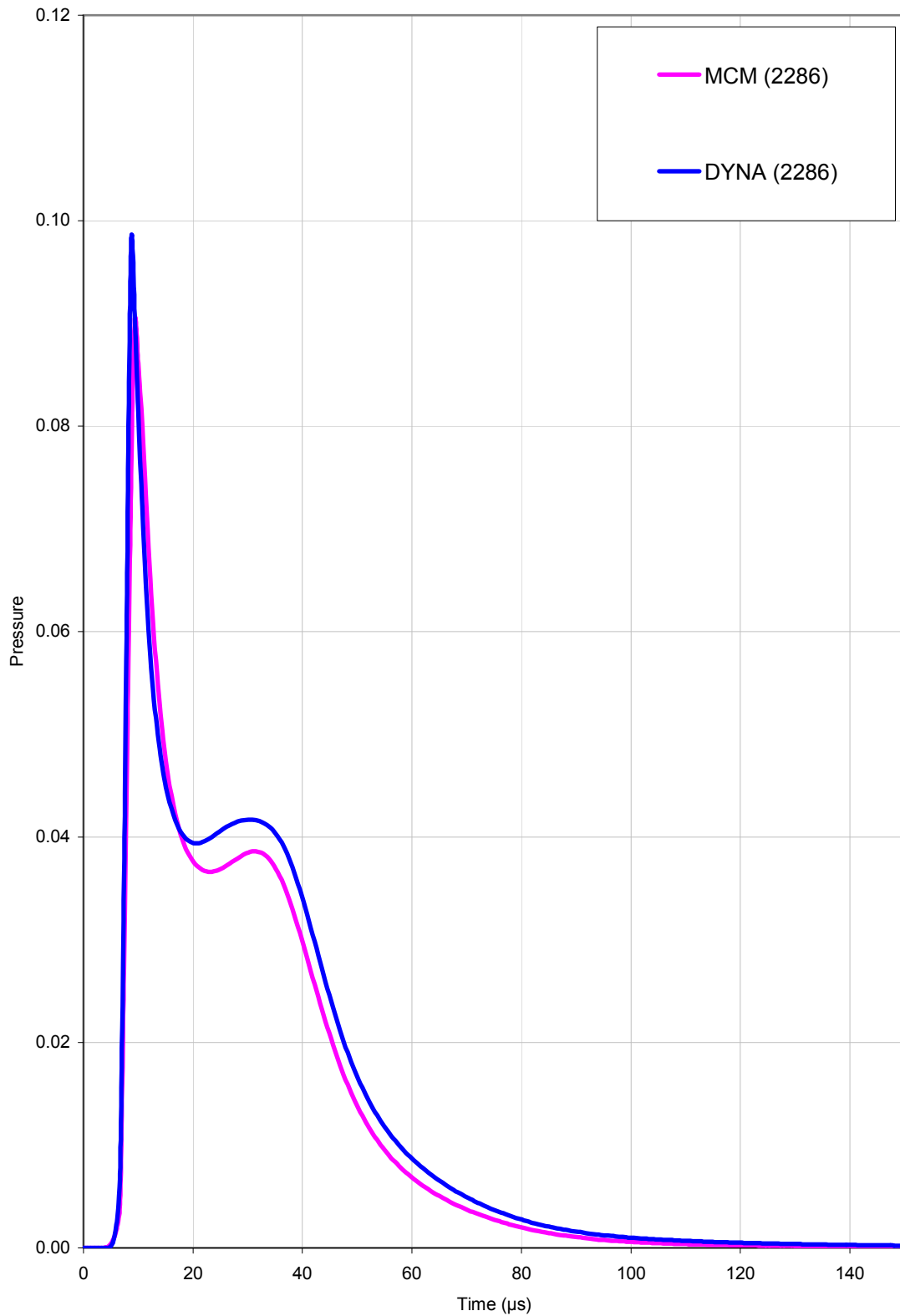
Figure AN1-2: Pressure/time curves for element/particle 2286 in MCM and LLNL-DYNA3D for the explosion model verification simulation.

Figure AN1-3: Pressure/time curves for element/particle 8474 in MCM and LLNL-DYNA3D for the explosion model verification simulation.

Figure AN1-4: Pressure/time curves for element/particle 3417 in MCM and LLNL-DYNA3D for the explosion model verification simulation.

Figure AN1-5: Pressure/time curves for element/particle 2680 (on 1 boundary) in MCM and LLNL-DYNA3D for the explosion model verification simulation.

Figure AN1-6: Pressure/time curves for element/particle 9251 (on 2 boundaries) in MCM and LLNL-DYNA3D for the explosion model verification simulation.
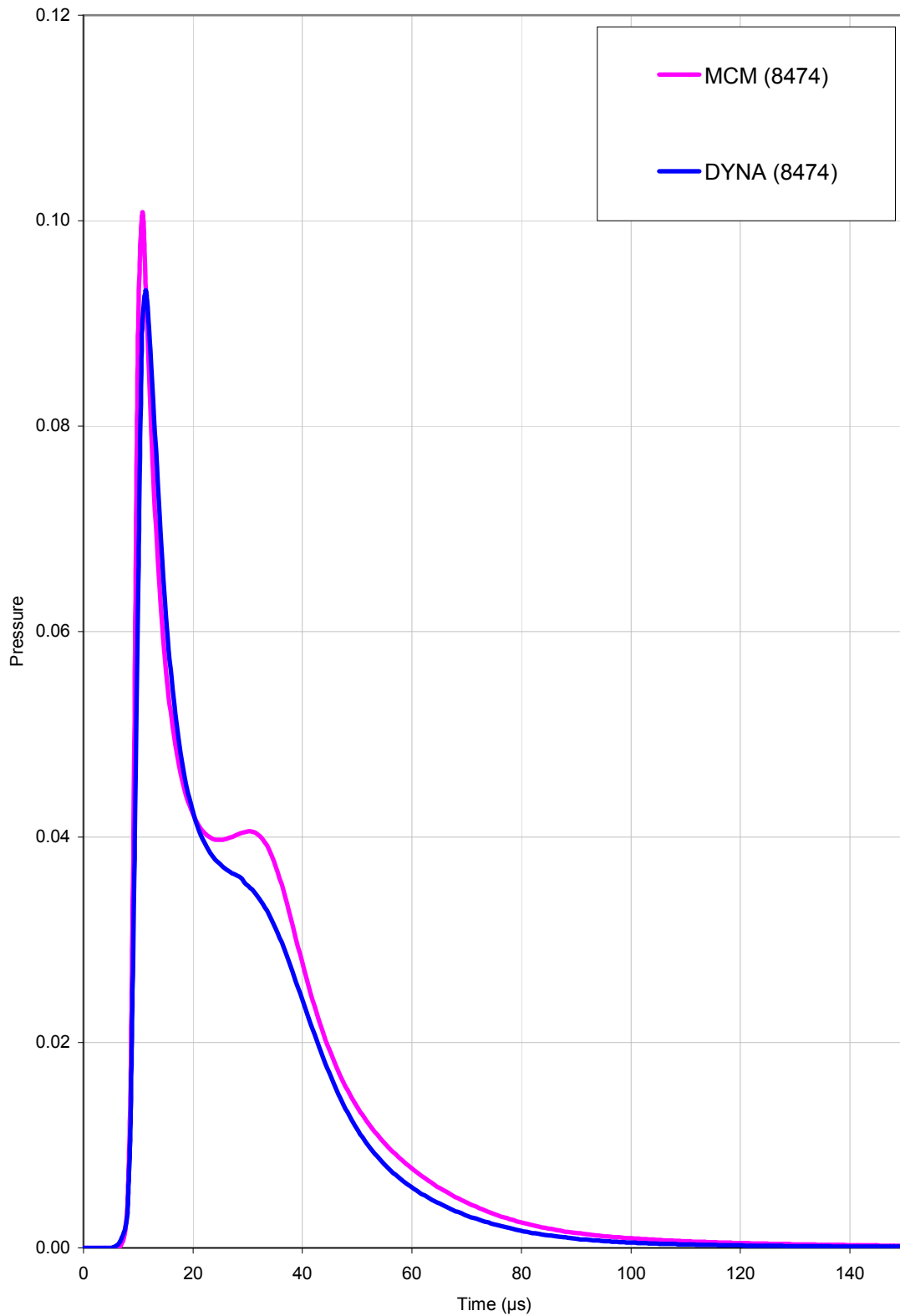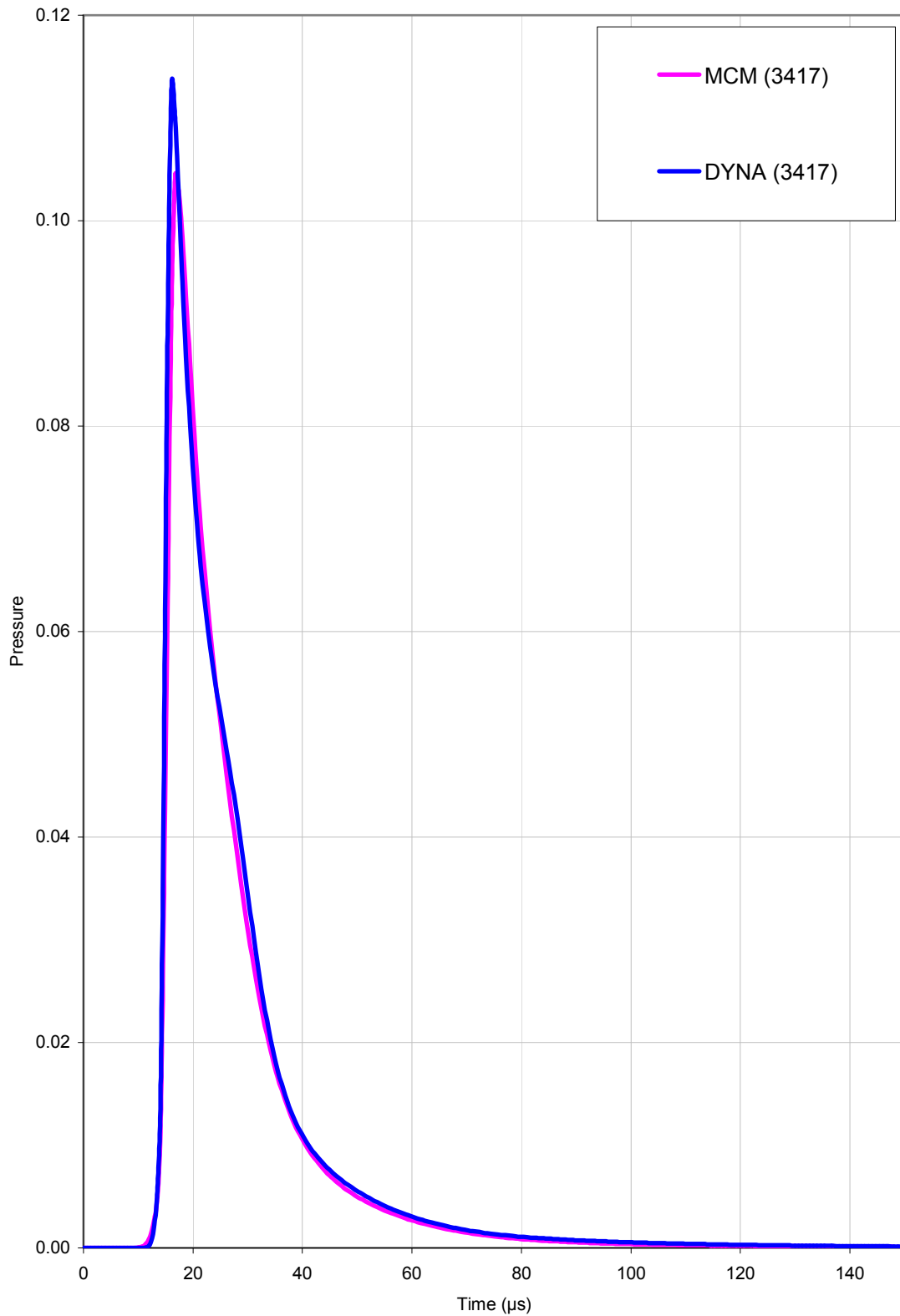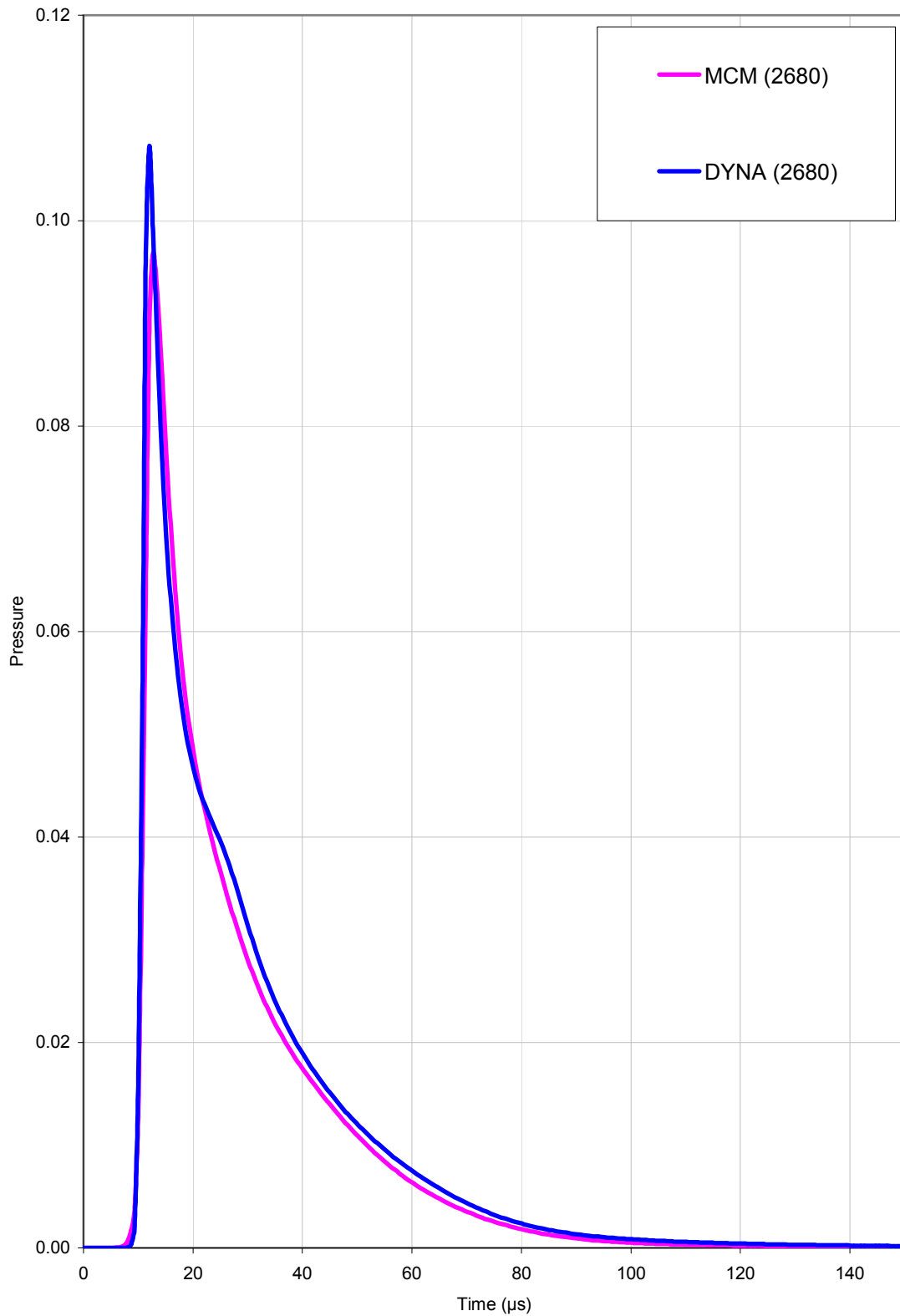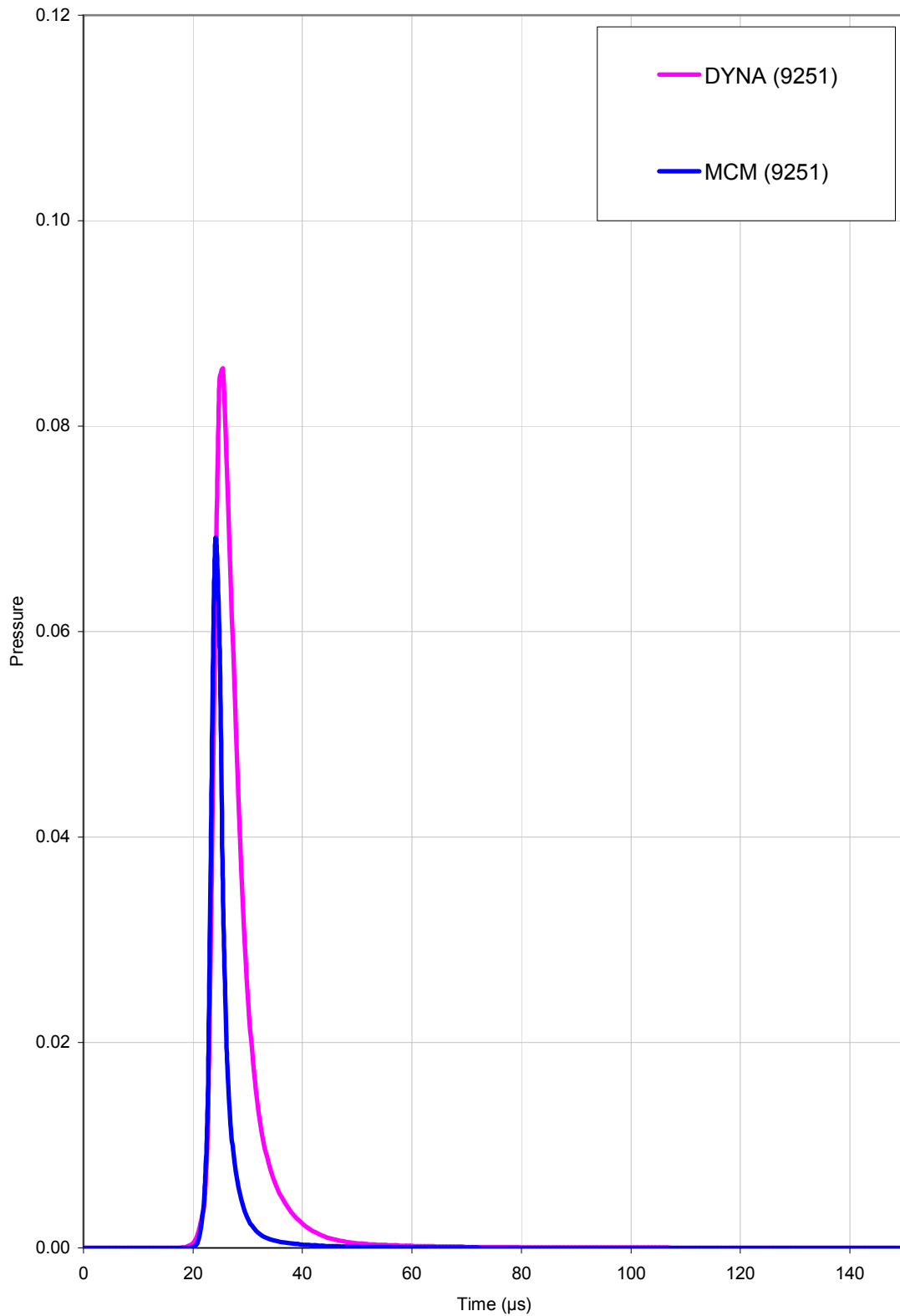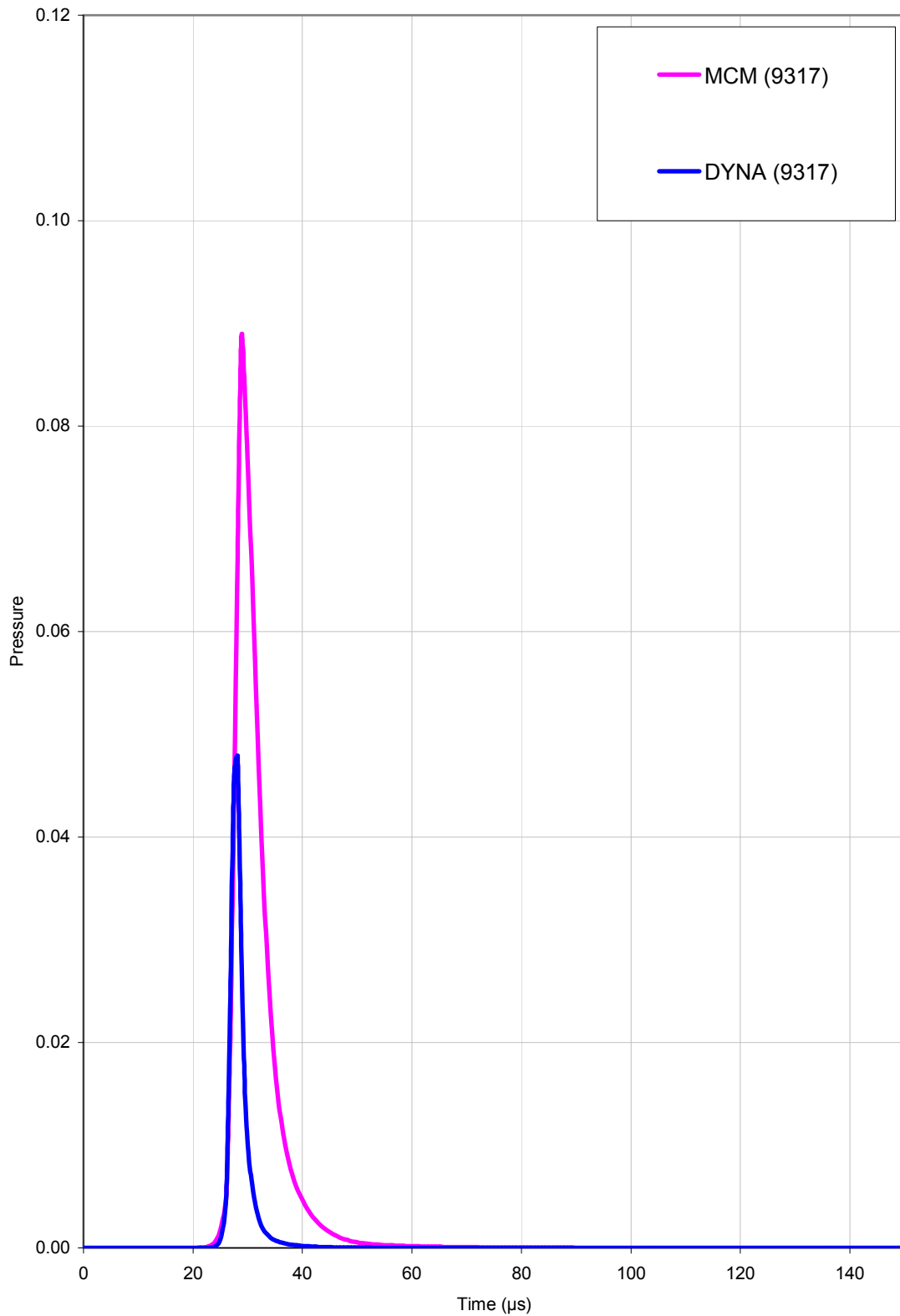
Figure AN1-7: Pressure/time curves for element/particle 9317 (on 3 boundaries) in MCM and LLNL-DYNA3D for the explosion model verification simulation.

This page is intentionally blank

Annex 2 – MCM Contact Descriptions

This annex contains a brief description of the contact descriptions implemented into MCM and therefore used in this research. The first is termed 'kernel contact' and is described in Section An2.1, the second is termed the 'repulsive force contact' and is described in Section An2.2.

An2.1 Kernel Contact

A standard method for representing contact between two bodies in SPH is to ignore the free surface boundary condition and allow mathematical interaction between boundary particles to occur naturally as two bodies move together. In other words contact is handled through the conservation equations, with no restriction on particles from one body being treated as neighbour particles of a particle in another body. They then interact through the sums over neighbour particles.

A limitation to this approach is that a degree of penetration and mixing occurs at the contact surface. Typically the first one to two particle rows of each body pass each other.

This explanation is taken from Campbell *et al.* (2000).

An2.2 Repulsive Force Contact

The repulsive force contact algorithm is similar to the Monaghan repulsive force (MRF) which is intended to remove the tensile instability inherent in a standard Eulerian SPH code (see Section 3.3.1 for a description of the SPH tensile instability and Section 5.1 for a thorough description of the MRF). The MRF works by placing a term in the SPH momentum equation that activates when two particles move together during the clumping of particles. The repulsive force contact algorithm uses the same approach to provide a contact force between boundary particles of different materials.

A repulsive force parameter, $f_{c_{ij}}$ is defined for each particle in the computation and initialised to zero for all particles. For each boundary particle, i with boundary neighbours, j from a separate material the repulsive force term is scaled in terms of the SPH kernel in the same manner as the MRF as follows:

$$f_{c_{ij}} = \frac{W(r_{ij})}{W(h_j)},$$

where r_{ij} is the magnitude of the position vector between i and j and h_j is the smoothing length of the j particle. The repulsive force contact assumes the use of the Cubic B-Spline kernel in the above formula (see Section 3.1.1.3).
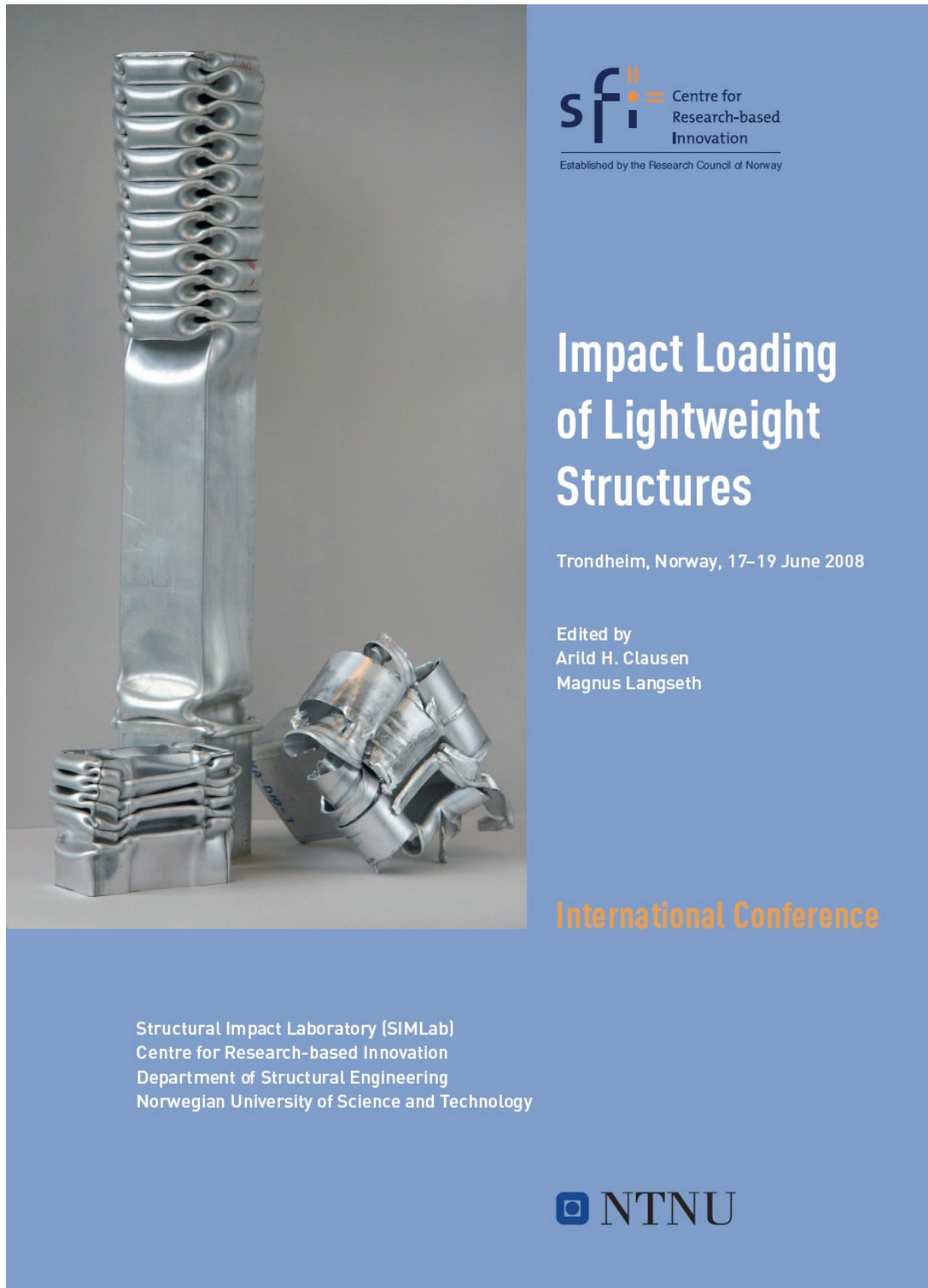
The repulsive term is applied to the SPH momentum equation as follows:

$$\bar{a}_i = \frac{d\bar{v}_i}{dt} = \sum_j m_j \left[\frac{(\boldsymbol{\sigma}_i - \mathbf{q}_i)}{\rho_i^2} + \frac{(\boldsymbol{\sigma}_j - \mathbf{q}_j)}{\rho_j^2} - k f_{c_{ij}}^n \right] \nabla_i W_{ij},$$

where k and n are user defined input parameters. Typical values for k and n are 0.1 and 4.0 respectively, although in practice these input parameters are used to 'tune' the contact algorithm for each individual application.

Appendix A – International Conference: Impact Loading of Lightweight Structures, SIMLab, sfi, NTNU, 17-19 June 2008

This appendix contains the extended abstract which was published in the conference proceedings and the PowerPoint slides that were presented on the 19th June 2008.

<u>Extended Abstract</u>

SPH Modelling for Failure in Metals

J.C. Campbell, Russell Strand, R. Vignjevic

Department of Impact, Structures and Mechanics Dynamics, School of Engineering,
Cranfield University, Cranfield, Bedfordshire,
MK43 0AL, United Kingdom
E-mail: J.Campbell@cranfield.ac.uk, R.K.Strand.2004@cranfield.ac.uk, V.Rade@cranfield.ac.uk

It is desirable to be able to accurately predict the initiation and growth of fracture at high strain rate in metals using computational mechanics for a wide range of applications. This is generally regarded as a difficult task and many researchers have taken different paths in attempt to achieve an accurate simulation of the process, with mixed results.

Meshless methods, such as Smoothed Particle Hydrodynamics (SPH), do not require the use of a fixed spatial grid to provide connectivity between nodes and therefore eliminates the issues of mesh entanglement under large deformations and also allows damage to grow in an arbitrary direction, reducing the influence on the spatial discretisation of the solution. This makes them particularly suited to the application of fracture mechanics.

This paper describes research towards developing a model to predict the growth of failure in explosively driven metals within a meshless framework. The model will have to be able to initiate damage via an existing damage model, propagate the damage to the surrounding neighbourhood in accordance with a growth criterion and allow for bifurcation of cracks which will lead to fragmentation. It is also desirable to allow for the explosive material to be able to flow between the new free surfaces and drive the discontinuity to open further. Such a model will require improvement of the treatment of boundary conditions to adequately create new free surfaces.

A basic method of modelling fracture is the 'stress to zero' approach, analogous to simple spall models in finite element codes, where once material failure is detected the stress in the particle is set to zero so that tensile loads cannot be supported. This approach has been implemented as a precursor for a more complex model that will identify failure on a local particle level. Two specific techniques are of interest for this research; the first proposes the identification of a particle-particle interaction area which is affected by a damage criterion, critical damage results in the interaction area being set to zero [1]. The second represents a crack by introducing discontinuities at individual particles, a continuous crack then consists of a set of contiguous cracked particles. Failure occurs when a damage parameter reaches a critical level, the failed particle is then split into two particles with the interaction between these new particles is governed by a contact description [2,3].

In this paper the progress of several of the approaches will be presented and discussed. The developed models are demonstrated on flyer-plate impacts and the Mock-Holt experiment [4]. In addition to these findings the accuracy of the computation will be discussed with reference to a local/non local implementation which considers a partition of unity weighting for calculating various kernel estimates, the validity of this approach will be discussed and potential applications within the failure model will be highlighted.

References:

[1] Swegle, J.W., 2000. *Conservation of momentum and tensile instability in particle methods.* SAND2000-1223. USA: Sandia National Laboratories.

[2] Rabczuk, T. and Belytschko, T., 2004. Cracking particles: A simplified meshfree method for arbitrary evolving cracks. *International Journal for Numerical Methods in Engineering,* **61**(13), pp. 2316-2343.

[3] Rabczuk, T. and Belytschko, T., 2007. A three-dimensional large deformation meshfree method for arbitrary evolving cracks. *Computer Methods in Applied Mechanics and Engineering,* **196**(29-30), pp. 2777-2799.

[4] Mock, W., Holt, W. H., 1983. Fragmentation Behavior of Armco Iron and HF-1 Steel Explosive-Filled Cylinders. *Journal of Applied Physics,* **54**(5), pp. 2344-2351.

Presentation Slides

Overview of the Presentation

Cranfield
UNIVERSITY

- *Brief overview of the research*

- *The Mock-Holt problem*

- *Representing fracture in a meshless method*

- *Inter-particle failure*
  - Swegle Interaction area
  - Demonstration on a plate-impact problem

www.cranfield.ac.uk

Overview of the Project

Cranfield
UNIVERSITY

- It is desirable to model the initiation and growth of fracture in metals at high strain-rates for many applications.

- The aim of the research is to develop a method for modelling material failure within the SPH meshless method. The absence of a fixed spatial grid reduces the influence of the spatial discretisation on the damage growth.

- Implementation of the model in Cranfield's SPH code

- Demonstration of the developed model through comparison with experimental and theoretical results.
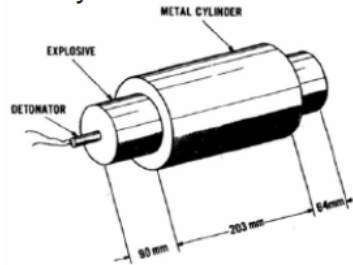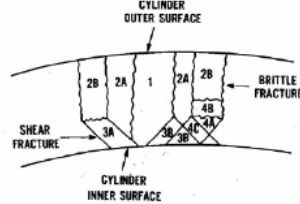
www.cranfield.ac.uk

The Mock Holt Experiment validation case continued

Cranfield UNIVERSITY

Mock-Holt Simulation - Standard SPH (Re
Time =   0

Model at 150us
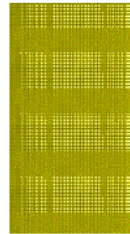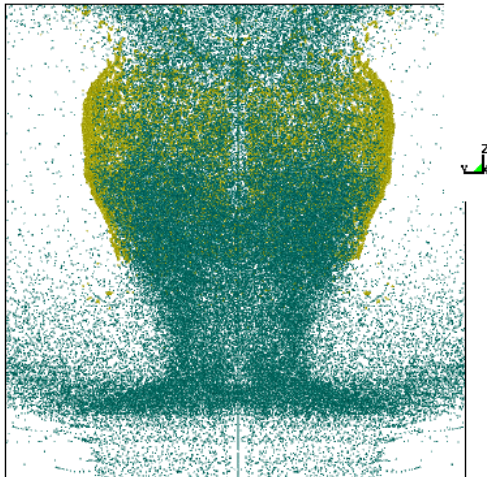
www.cranfield.ac.uk

Representing fracture within a meshless method

Cranfield UNIVERSITY

The fracture model employed for this research will have to cope with the following phenomena:
- Initiation and growth of damage, via a damage model.
- Crack formation and propagation.
- Crack branching and crack joining, leading to fragmentation.

1. 'Stress to zero concept' – Upon failure a particle will be prevented from transferring tensile loads by setting its stress to zero. Equivalent to basic spall models implemented in FE codes

Time =   9.6055
Contours of Z-stress
min=-0.0054991, at elem# 3355
max=0.00554996, at elem# 8255

Fringe Levels
5.550e-03
4.445e-03
3.340e-03
2.235e-03
1.130e-03
2.543e-05
-1.079e-03
-2.184e-03
-3.289e-03
-4.394e-03
-5.499e-03

Target Plate

PMMA

A

Spall fracture

Flyer plate
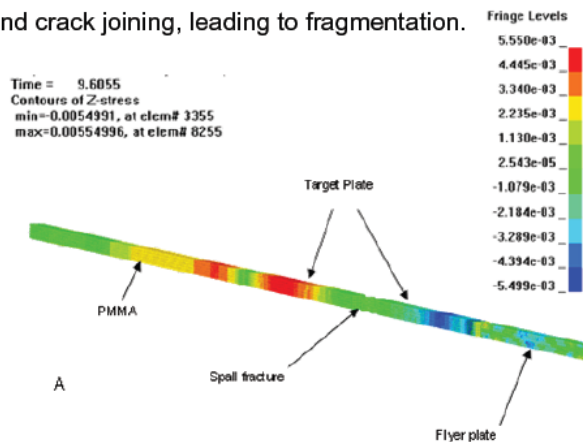
www.cranfield.ac.uk

How to represent a Crack – conclusions to literature survey

Cranfield UNIVERSITY

2. 'Cracked/failed' particle concept - A failed particle is split into two particles, this approach has been demonstrated by Rabczuk and Belytschko (2007).
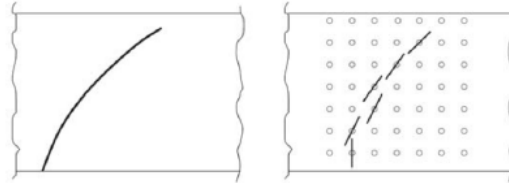
3. Inter-particle failure – here damage is evaluated for pairs of neighbour particles. Upon fracture the particles cease being neighbours.

   We use concept of particle-particle interaction area (Swegle, 2000),
   - damage affects interaction area
   - failure sets area to zero
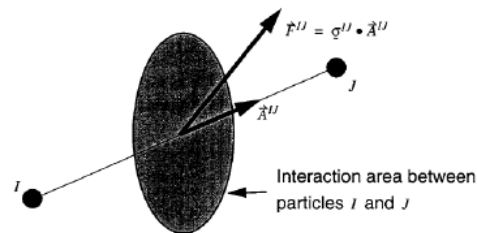
Interaction area between particles *i* and *j*

Cranfield UNIVERSITY

Swegle Interaction Area

- Swegle (2000) introduced the concept of area vectors within SPH for the purpose of discussing the tensile instability.
- Noting the fundamental definition of the stress tensor shows that the force exerted on a surface due to a stress is given by

$$\mathbf{F} = \mathbf{\sigma} \bullet \mathbf{A}$$

- He showed that the SPH momentum equation could be rewritten in terms of an interaction area:

$$a_i = \frac{dv_i}{dt} = \sum_j m_j \left[\frac{\sigma_i}{\rho_i^2} + \frac{\sigma_j}{\rho_j^2} \right] \nabla_i W_{ij}$$

$$m_i a_i = \sum_j vol_i vol_j \nabla_i W_{ij} \left[(\sigma_i)\frac{\rho_j}{\rho_i} + (\sigma_j)\frac{\rho_i}{\rho_j} \right] = F_i$$

$$= \sum_j A_{ij} \left[(\sigma_i)\frac{\rho_j}{\rho_i} + (\sigma_j)\frac{\rho_i}{\rho_j} \right]$$

Continuum Damage Mechanics

Cranfield UNIVERSITY

- A standard concept in continuum damage mechanics is that the damage variable represents an effective surface density of cracks or voids within the material.

- This leads to the concept of effective stress first introduced by Kachanov in 1958 and used by many authors including Lemaitre.

- The effective stress tensor can be defined as:

$$\tilde{\sigma} = \frac{\sigma}{1 - D}$$

- The concept of the interaction area within SPH lets up apply this concept in a different manner

Inter-particle failure

Cranfield UNIVERSITY

- Damage is evolved as an inter-particle value, D_{ij}, and acts to reduce the interaction area:

$$F_i = \sum_j \left[(\sigma_i)\frac{\rho_j}{\rho_i} + (\sigma_j)\frac{\rho_i}{\rho_j} \right] A_{ij}\left(1 - D_{ij}\right)$$

- When damage reaches a critical value the material is assumed to have failed and the particles cease to be neighbours.
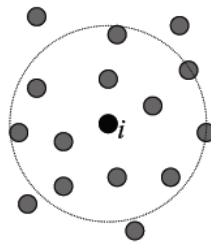
Cranfield
UNIVERSITY

- A10 -
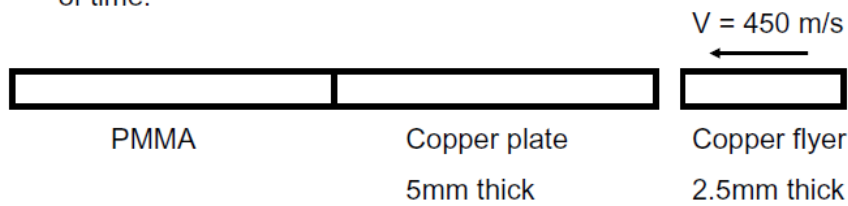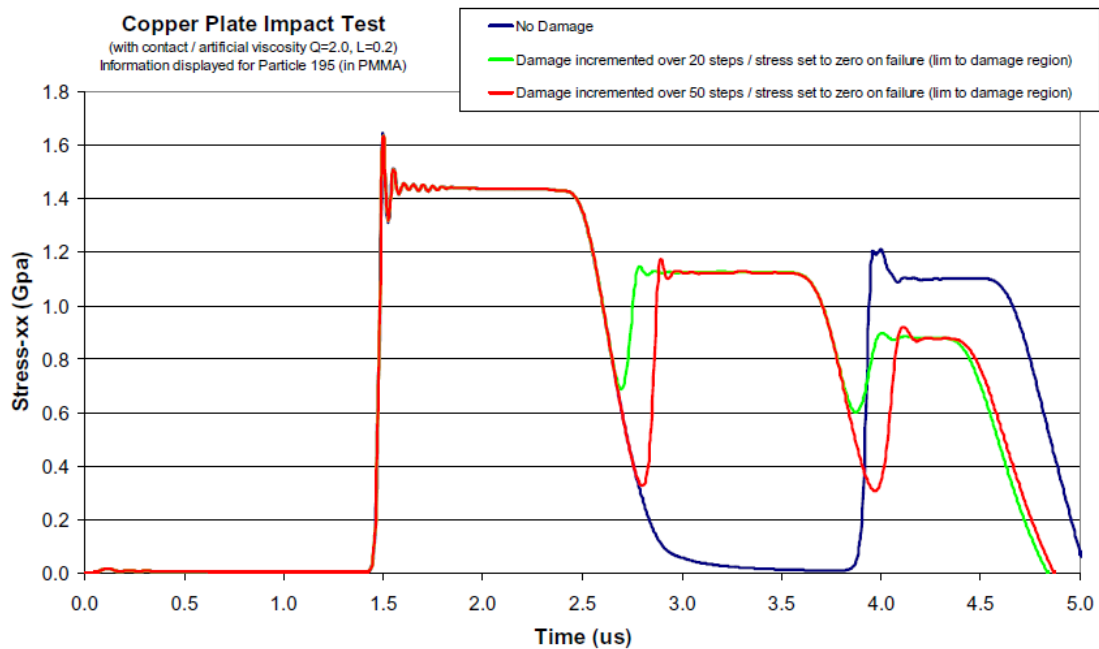
Future Work

Tasks

1. A Cochran-Banner damage growth model has been implemented and will be tested shortly.   The use of other damage models will be investigated.

2. Develop an approach to treat the influence of inter-damage on the surrounding neighbours.

3. Test model capability on experimental data (plate impact test data and the Mock-Holt Experiment)

www.cranfield.ac.uk

Cranfield
UNIVERSITY

Questions?

www.cranfield.ac.uk

Appendix B – International Journal of Impact Engineering: Paper Submitted Dec 2008

A particle-particle interaction-area approach for modelling material damage within the SPH method

R.K. Strand, J.C. Campbell[*], R. Vignjevic

Crashworthiness, Impact and Structural Mechanics Group, School of Engineering,
Cranfield University, Cranfield, Bedfordshire, MK43 0AL, UK

Abstract

It is generally regarded to be a difficult task to model multiple fractures leading to fragmentation in metals subjected to high-strain rates using continuum mechanics. Meshless methods such as Smooth Particle Hydrodynamics (SPH) are well suited to the application of fracture mechanics, since they are not prone to the problems associated with mesh tangling. The SPH momentum equation can be rearranged in terms of a particle-particle interaction area. Damage acts to reduce this area and is ultimately set to zero, indicating material fracture. The model makes use of a simplified Cochran-Banner description for evolving the damage parameter and incorporates a multiple bond break criterion for each neighbourhood of particles. The model was tested on a 1D flyer plate impact test and the results were compared to experimental data. The test showed that the model can recreate the phenomena associated with uniaxial spall to a high degree of accuracy.

Keywords: Fracture Modelling; Continuum Damage Mechanics; Smooth Particle Hydrodynamics; Interaction Area; Spall

1. Introduction

It is desirable to be able to accurately predict the initiation and growth of multiple fractures at high strain rate in metals using computational mechanics to represent the continuum for a wide range of applications. This is generally regarded as a difficult task and researchers have taken different paths in attempt to achieve an accurate simulation of the process, with mixed results.

A large portion of crack-growth research is conducted with the use of 'meshed' methods such as the finite element technique. This is not surprising since these methods are well established and understood. There are two main types of crack representation model that are adopted for this approach, these are; inter-element separation models and arbitrary crack-path models.

---

[*] Corresponding author. Fax: +44 1234 752 149.
   E-mail address: J.Campbell@cranfield.ac.uk (J. C. Campbell).

In an inter-element separation model cracks are developed along element boundaries, i.e. they follow the lines of the mesh [1]. The method is relatively simple compared to others; however when the crack-paths are not in line with the element edges, the fracture energy is often overestimated. Also the solution is often dependent on a well refined mesh. [2]. It is not desirable to have to define the mesh around the crack path, as this assumes that one can anticipate the failure mechanism of the material. This problem can be circumvented by the use of re-meshing techniques [3-6], although these methods tend to be very computationally expensive since in the case of dynamic crack growth, the amount of re-meshing required is often substantial.

Arbitrary crack path models are more realistic in their approach to dealing with fracture, but prove to be quite complicated within a 'meshed' framework. One such example is the extended finite element method [7]. Crack problems have been modelled in two and three dimensions [8-10] by coupling the technique with a level set method [8], i.e. modelling the 'crack-geometry' separately from the model. For fatigue-crack growth problems, discontinuities can be modelled quite accurately using this approach, however, a new level-set needs to be introduced for each individual crack, which would become very expensive for problems dealing with fragmentation. A re-meshing approach has been adopted by Pandolfi et al [4] to extend the method to encompass crack branching.

'Meshless' methods, such as Smoothed Particle Hydrodynamics (SPH), do not require the use of a fixed spatial grid to provide connectivity between nodes and therefore eliminates the issues of mesh entanglement under large deformations, and also allows damage to grow in an arbitrary direction, reducing the influence on the spatial discretisation of the solution. This makes them particularly suited to the application of fracture mechanics.

Extensive work has been conducted using the Element Free Galerkin (EFG) method, for several different failure modes. Single discontinuities have been modelled in 2D [11] and 3D [12] and a method that makes use of level sets to describe multiple cracks is proposed by Ventura et al [13]. An interesting approach to dealing with multiple fractures in 2D is given by Rabczuk and Belytschko

[2], and they have since extended the method to 3D [14]; the method is called EFG-P, which is a particle method within the EFG framework. Cracks are represented by introducing discontinuities at individual particles. A continuous crack then consists of a set of contiguous cracked particles.

From the literature it is clear that the technique adopted to model fracture is dependent on the exact failure mode that is being modelled. Cracks that develop slowly, such as those found in fatigue crack problems are more suited to the more global 'crack-geometry' approach; and problems which contain multiple cracks lend themselves to a more 'local' description of the damage.

This paper describes research towards developing a model to predict the growth of failure in explosively driven metals within the meshless SPH framework. The model will have to be able to initiate damage via an existing damage model, propagate the damage to the surrounding neighbourhood in accordance with a growth criterion and allow for bifurcation of cracks which will lead to fragmentation. It is also desirable to allow for the explosive material to be able to flow between the new free surfaces and drive the discontinuity to open further.

The SPH momentum equation can be rearranged to include an 'interaction-area' between particles; damage acts to reduce this area until, upon failure, the area is set to zero. This may be coupled with a criterion that breaks multiple bonds in a given direction, thus increasing the onset of failure. In this paper a 1D proof-of-concept is outlined and tested on a simple plate impact test. The results are compared to experimental data to validate the effectiveness of the model.

The article is arranged as follows: A brief overview of the SPH method can be found in Section 2. The governing equations and relevant theory surrounding the particle-particle interaction area are given in Section 3. The damage growth model that has been developed for this research is outlined in Section 4. In Section 5, the results of the 1D plate impact test are presented and compared to the experimental data and the validity of the model is discussed. The conclusions to the research are given in Section 6.

2. The SPH Method

In the SPH method, the continuum is represented by a discrete set of nodes (particles), that posses individual material properties and move according to the conservation laws (See Equations 2.1-3). The method was originally developed for problems in astrophysics [15,16] and was updated to include solid mechanics problems by Libersky [17]. To date, the method has been applied to a wide spectrum of applications such as hyper-velocity impact [18].

A common set of equations that are solved by an SPH code are the conservation laws, given as follows:

Conservation of mass:
$$\frac{d\rho_i}{dt} = \rho_i \sum_j \frac{m_j}{\rho_j} \left[v_i - v_j \right] \nabla_i W_{ij}, \tag{2.1}$$

Conservation of momentum:
$$\frac{dv_i}{dt} = -\sum_j m_j \left[\frac{\boldsymbol{\sigma}_i}{\rho_i^2} + \frac{\boldsymbol{\sigma}_j}{\rho_j^2} \right] \nabla_i W_{ij}, \tag{2.2}$$

Conservation of energy:
$$\frac{de_i}{dt} = \frac{\boldsymbol{\sigma}_i}{\rho_i^2} \sum_j m_j \left[v_i - v_j \right] \nabla_i W_{ij}, \tag{2.3}$$

where the summation is over all neighbour particles, *j*. m_j is the mass of the *j* particle, $\boldsymbol{\sigma}$ is the full stress tensor, *v* is velocity and ρ is the density. W_{ij} is SPH kernel function and ∇_i denotes the gradient of the kernel function, taken with respect to the *i* particle.

The kernel function used in this research to approximate the derivatives is the cubic spline kernel [16],

$$W(r,h) = \frac{b}{\pi h^a} \begin{cases} 1 - \frac{3}{2}q^2 + \frac{3}{4}q^3 & \text{if } 0 \le q \le 1 \\ \frac{1}{4}(2-q)^3 & \text{if } 1 \le q \le 2 \\ 0 & \text{otherwise} \end{cases} \tag{2.4}$$

where a is the number of spatial dimensions, b is a normalised constant which has the values; $\frac{2}{3}$,

$\frac{10}{7\pi}$, $\frac{1}{\pi}$ in 1, 2 and 3 dimensions respectively. $q = r/h$, r is the position vector between a pair of

particles and h is the finite range of the kernel (the SPH smoothing length).

There are several problems within the SPH method outlined by Swegle et al [19], such as the

tensile instability, numerical fracture [20] and zero energy modes [21]. The tensile instability is a

numerical problem that manifests in the discretisation of the conservation equations, i.e. when the

particles are under tensile stress their motion becomes unstable. The problem can be visualised as an

unphysical clumping of the particles and in some cases leads to early termination of a simulation.

Analysis of the problem has determined that the instability depends on the sign of the product of the

stress times the second derivative of the kernel function [20], this condition can be represented

mathematically as follows:

$$W'' \boldsymbol{\sigma} > 0 , \tag{2.5}$$

where W'' is the spatial second derivative of the gradient of the kernel function and $\boldsymbol{\sigma}$ is the stress.

This condition tells us that as long as Equation 2.5 is positive the method is unconditionally unstable.

In practice this instability manifests itself as a clumping of particles that can be clearly seen in

simulations.

In addition, severe manifestations of the tensile instability lead to another relevant problem with

a basic SPH formulation known as numerical fracture. Two SPH particles are considered to be

neighbours if the distance between them is less than twice the smoothing length. Thus all particles

within a spherical domain are neighbours of the particle at the centre of that domain. This domain

remains a sphere throughout the calculation, with either a fixed or variable radius. In the case of

anisotropic deformation two particles that should remain neighbours can move far enough apart that

they no longer influence each other. The result in the simulation is fracture that should not be present

in the calculation.

The tension instability and numerical fracture can be eliminated completely by use of a Lagrangian kernel function [22-25]. In the Total Lagrangian approach the initial state of the domain is regarded as the reference state, consequently the solution of the conservation equations are expressed in terms of the material co-ordinates, $\mathbf{X}$, instead of the spatial co-ordinates, $\mathbf{x}$. The relation between these co-ordinates is given as follows:

$$\overline{\mathbf{x}} = \phi\left(\overline{\mathbf{X}}, t\right), \tag{2.6}$$

where ϕ is a mapping function. It is usual that at time, $t = 0$:

$$\tilde{\mathbf{x}} = \phi\left(\overline{\mathbf{X}}, 0\right) = \tilde{\mathbf{X}}. \tag{2.7}$$

The deformation gradient is given as follows:

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}}, \tag{2.8}$$

and the conservation of mass is:

$$J_0 \rho_0 = J \rho, \tag{2.9}$$

where

$$J = \det \mathbf{F}. \tag{2.10}$$

All calculations are performed in the reference state and therefore the neighbourhood does not change. This method is valid for as long as the mapping exists between the domains; when large shear deformations are present this cannot be achieved and thus the limitation of the method is defined.

3. A Particle-Particle Interaction Area Approach

The concept of area vectors within the SPH method was outlined by Swegle [26] for the purpose of discussing the tensile instability inherent in any basic SPH description (see previous section for more information). The paper noted that the fundamental definition of the stress tensor shows that a force exerted on a surface due to stress is given by:

$$\mathbf{F} = \mathbf{\sigma} \bullet \mathbf{A} . \tag{3.1}$$

He then showed that the SPH momentum equation could be rewritten in terms of an interaction area in accordance with the above equation.

From Newton's Second Law we have:

$$F_i = m_i a_i , \tag{3.2}$$

and may therefore rewrite Equation 2.2 in the following form:

$$F_i = m_i a_i = -\sum_j m_i m_j \left[\frac{\mathbf{\sigma}_i}{\rho_i^2} + \frac{\mathbf{\sigma}_j}{\rho_j^2} \right] \nabla_i W_{ij} . \tag{3.3}$$

By rearranging the above equation it is possible to group the terms such that the expression yields equation 3.1:

$$F_i = -\sum_j A_{ij} \left[\left(\mathbf{\sigma}_i \right) \frac{\rho_j}{\rho_i} + \left(\mathbf{\sigma}_j \right) \frac{\rho_i}{\rho_j} \right] , \tag{3.4}$$

$$A_{ij} = V_i V_j \nabla_i W_{ij} . \tag{3.5}$$

Where V is the volume of the particle and A_{ij} is the area vector which has a direction normal to the surface and a magnitude equal to the area of the surface as shown in Figure 3.1 below.
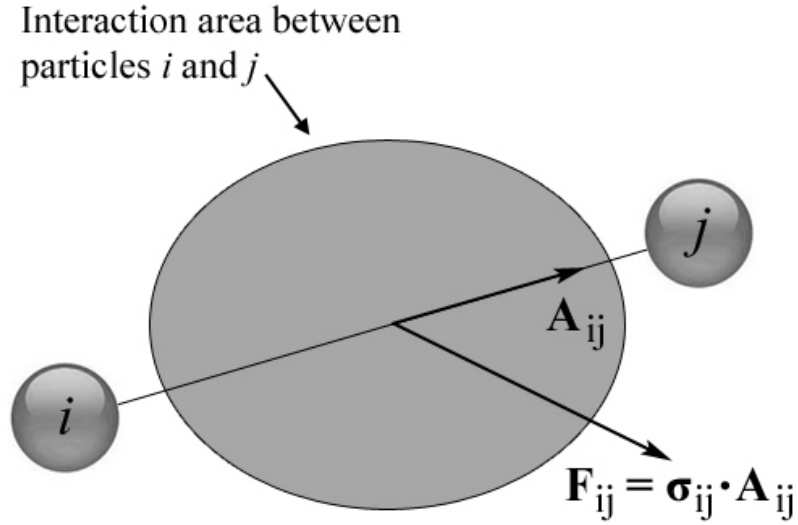
Figure 3.1 - Area vector for particle interactions.

Thus, the gradient of the kernel function can be thought of as defining the area on which stress acts to produce a force between the particles.

A standard concept in continuum damage mechanics is that the damage variable, D represents an effective surface density of microscopic cracks or voids within the material. This leads to the concept of an effective stress, first introduced by Kachanov [27] and has since been used by many authors including Lemaitre [28].

If $\mathbf{F}$ is the load acting on the overall section area S of an element, the usual stress tensor is given by:

$$\mathbf{T} = \mathbf{F}/S \,, \tag{3.6}$$

which leads to the Cauchy stress tensor $\boldsymbol{\sigma}$ ($\mathbf{T} = \boldsymbol{\sigma} \bullet \mathbf{n}$, where $\mathbf{n}$ is the normal). In order to account for the presence of micro-cracks and cavities in the material an effective area is defined by the quantity

$$\tilde{S} = S(1-D) \,, \tag{3.7}$$

which effectively carries the load, **F**. By combining equations 3.6 and 3.7 we have:

$$\tilde{\mathbf{T}} = \frac{\mathbf{F}}{\tilde{S}} = \frac{\mathbf{T}}{1-D}, \tag{3.8}$$

where $\tilde{\mathbf{T}}$ is the effective stress vector. D is a scalar and so the effective stress tensor may be written as $\tilde{\mathbf{T}} = \tilde{\boldsymbol{\sigma}} \bullet \mathbf{n}$, where $\tilde{\boldsymbol{\sigma}}$ is given by:

$$\tilde{\boldsymbol{\sigma}} = \frac{\boldsymbol{\sigma}}{1-D}. \tag{3.9}$$

The SPH method coupled with the concept of Swegle's interaction area offers an alternative approach to the common effective stress method of applying damage in computational mechanics as shown in Equation 3.9. The interaction area, A_{ij} can be thought of as being synonymous with the section area, S defined in Equation 3.6. Damage can be evolved as an inter-particle parameter, D_{ij} and acts to reduce A_{ij} by applying the concept of continuum damage from Equation 3.7 directly. By applying damage in this manner no modification to the stress is required, thus Equation 3.4 becomes:

$$F_i = -\sum_j \left[(\boldsymbol{\sigma}_i)\frac{\rho_j}{\rho_i} + (\boldsymbol{\sigma}_j)\frac{\rho_i}{\rho_j} \right] A_{ij} \left(1 - D_{ij}\right). \tag{3.10}$$

Critical damage ($D_{ij} = 1.0$) assumes the material to have failed and the interaction area is set to zero, causing the particles to cease being neighbours. To prevent unphysical interactions between fully damaged particles, all interpolated values are also set to zero ($\nabla_i W_{ij} = 0$), once critical damage has been reached.

In a standard Eulerian SPH code, a new neighbour search is competed at each time-step; thus a particle may gain or lose neighbours throughout the computation as deformation takes place. This prevents the practical treatment of inter-particle damage, which is a material history parameter and therefore must be integrated in time. For this reason we have chosen to use a Total Lagrangian description for the SPH interpolation; i.e. the neighbourhood, and thus the particle-particle bonds

defined in the reference state remain fixed throughout the computation (see previous section for a description of the Total Lagrangian method). The interaction areas between particles can therefore be stored as material history variables and be damaged via the damage parameter at every time-step.

4. Implementation

A damage growth model is required to develop the damage parameter, D_{ij}. Any model is valid provided that it represents damage as a percentage of damaged material (i.e. micro-cracks/voids) within a particle-particle pair, based on local particle parameters. The failure criterion to initiate damage growth should also be dependent on an inter-particle variable (such as inter-particle traction), in order to correctly 'grow' a crack in the material. This method assumes that phenomena such as crack branching/bifurcation and crack joining are incorporated as a function of multiple bond failures in a localised region; and so careful selection of the failure criterion is required. For the work presented here, damage is initiated via a critical stress between particles and a simplified Cochran-Banner model [29] is used for damage growth.

Damage occurs when critical stress is reached between a pair of particles i and j in tension (hydrostatic tension spall). In order to check for this criterion, it is necessary to track the direct stress (traction) between the particle pair. Since SPH data are stored at the individual particles; the stress in the $(i \rightarrow j)$ direction, T_i and the $(i \leftarrow j)$ direction, T_j must first be obtained. From these stresses a composite value can be calculated for the traction between the particle pair (T_{ij}). T_i and T_j are therefore the first component of the respective stress tensor in the direction of $(i \rightarrow j)$ and $(i \leftarrow j)$ respectively. It is possible to obtain these values by rotating the full stress tensor to the new co-ordinate system for both particles; however this is computationally expensive and will yield the full stress tensors, where only the (1,1) values are required. From Malvern [30] we observe that the traction vector, $\hat{T}_i$ is simply the dot product of the unit vector which projects onto the $(i \rightarrow j)$ direction, $\hat{n}_i$ and the stress tensor:

$$\hat{T}_i = \hat{n}_i \bullet \boldsymbol{\sigma}_i . \tag{4.1}$$

Thus the direct stress in the i particle, in the $(i \rightarrow j)$ direction is simply the magnitude of the traction vector, $T_i = \left| \hat{T_i} \right|$. The sign of T_i (to indicate compression or tension) can be determined from the sign of $T_i \square \hat{n}_i$, a negative value indicates compression and a positive value indicates tension (T_j is obtained by replacing the suffix i with j in the traction equations above). The traction between the particle pair, T_{ij} is then assumed to be the average value of the traction in the individual particles:

$$T_{ij} = \frac{\left(T_i + T_j \right)}{2} .$$

(4.2)

If T_{ij} exceeds a user defined spall criterion, Σ (based on material data); damage will begin to grow for that i-j pair.

The Cochran-Banner damage growth model [29] is a one dimensional model that assumes all changes in volume past the material spall strength is due to micro-crack and void growth, i.e.

$$D\left(x,t \right) = \int_0^t dV / A, \; dV > 0 ,$$

(4.3)

A three dimensional version of this model is given by Mirkovic [31] and it is this model that has been adapted for use with the interaction area outlined in the previous section.

The current change in volume for a particle may be calculated from the strain rate and the current time-step. We are interested in the change in volume for the area between the particle pair under consideration and so we take the mean value of the change in volume for the two particles, thus:

$$dV_{ij} = \frac{dt}{2} \left[V_i \sum_{n=1}^{3} \dot{\varepsilon}_{i_n}^{pl} + V_j \sum_{n=1}^{3} \dot{\varepsilon}_{j_n}^{pl} \right],$$

(4.4)

where $\dot{\varepsilon}^{pl}$ is the effective plastic strain-rate. The cross-sectional area which the damage acts to reduce is taken to be the magnitude of the interaction area, A_{ij} (Equation 3.5) and so the Cochran-Banner damage parameter is simply:

$$D_{CB} = \frac{dV_{ij}}{|A_{ij}|} \qquad (4.5)$$

The sign of dV_{ij} will depend on whether the force between the particles is compressive or tensile. Therefore damage is only grown for positive changes in volume, i.e. void growth. It is clear that in a real continuum, compressive forces will act to close the voids and the material will regain some strength, but subsequent void re-growth under tension will be more rapid. However, in this version of the model, we simply hold the damage parameter constant (at its previous value) for the times when dV_{ij} is negative.

The inter-particle damage is applied via the following equation:

$$D_{ij_n} = D_{ij_{(n-1)}} + \left(D_{CB}/D_{crit}\right)^{2/3}, \qquad (4.6)$$

where D_{crit} represents critical damage (material parameter). Total failure of the material is reached when $D_{ij} = 1.0$. Therefore the total damage of the material at time, t is given as follows:

$$D_{ij} = \text{MIN}\left[D_{ij}, 1.0\right]. \qquad (4.7)$$

In a real solid, localised damage would have a small influence on the surrounding material, i.e. the damage of the model on a global scale. In an attempt to capture this phenomenon, a multiple bond-break criterion has also been added to the damage model.

5. Numerical Demonstration

In order to demonstrate the capability of the damage model outlined in the previous section, simulation results were compared to experimental data for a simple 1D copper plate impact test. Experimental data was obtained from Panov [32]. In the experiment a 10 mm thick OFHC copper sample was impacted by an OFHC copper plate 5 mm in thickness at a velocity of 304 m/s. Both target and flyer were machined into circular plates with diameters of 70 mm and 50 mm respectively. To record longitudinal stress data, a Manganin stress gauge was supported on the rear surface of the target with a 12 mm block of Polymethylmethacrylate (PMMA). The geometry of the flyer and the target were chosen such that the reflected stress wave off the free ends would interact in the centre of the target plate. This type of experiment is extremely useful for damage model validation purposes in 1D, since the experiment can be precisely controlled to yield a uniaxial state of deformation inside the target plate.

The simulation of the experiment made use of an Isotropic-Elastic-Plastic-Hydrodynamic material model with a Gruneisen equation of state for the OFHC Copper and the PMMA. Parameters for both materials can be found in Tables 5.1-2 and 5.3-4 respectively and a diagrammatic representation of the model is given in Figure 5.1. The model consisted of 540 particles; 100 in the flyer, 200 in the target and 240 in the PMMA, a gap of 0.2 mm was placed between the flyer and target. The SPH smoothing length was taken to be 1.3 times greater than the initial inter-particle spacing. The spall strength for OFHC copper was taken from Steinberg [33] as 1.2 GPa and the value of critical damage, D_{crit} required for the Cochran-Banner damage model was taken from Cochran and Banner [29] to be 0.007.
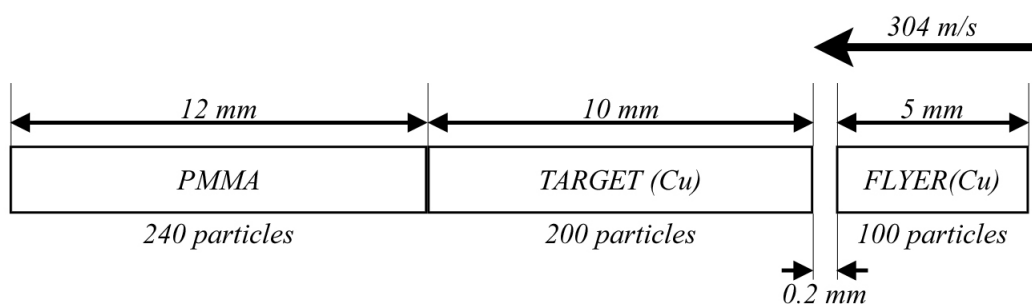
Figure 5.1 – Diagrammatic representation of the Simulation model.

Table 5.1 – Material model parameters for OFHC copper used in the numerical simulation

Parameter	Description	Nominal Value
G	Shear Modulus	48.4 GPa
Σ_0	Yield Stress	150 MPa
E_p	Plastic Modulus	500 MPa
P	Density	8.924 g/cm^3

Table 5.2 – Gruneisen equation of state constants for OFHC copper used in the numerical simulation

Parameter	Description	Nominal Value
C	Speed of Sound	0.394 cm/μs
S_1	1st Hugoniot slope coefficient	1.4985
S_2	2nd Hugoniot slope coefficient	0.0
S_3	3rd Hugoniot slope coefficient	0.0
γ_0	Gruneisen coefficient	2.02
a	1st order volume correction coefficient	0.47

Table 5.3 – Material model parameters for PMMA used in the numerical simulation

Parameter	Description	Nominal Value
G	Shear Modulus	2.32 GPa
σ_0	Yield Stress	350 MPa
E_p	Plastic Modulus	300 MPa
P	Density	1.180 g/cm^3

Table 5.4 – Gruneisen equation of state constants for PMMA used in the numerical simulation

Parameter	Description	Nominal Value
C	Speed of Sound	0.218 cm/μs
S_1	1st Hugoniot slope coefficient	2.088
S_2	2nd Hugoniot slope coefficient	-1.124
S_3	3rd Hugoniot slope coefficient	0.0
γ_0	Gruneisen coefficient	0.85
a	1st order volume correction coefficient	0.0

From the experimental data we observe that the release waves from the free surfaces propagate into the material and cause high-tensile stress in the centre of the target plate. When this tension exceeds the spall strength of the material, the material fails and micro-voids begin to form and coalesce in this region, ultimately leading to the generation of a new free surface. This free surface reduces the tensile stress in the material to zero and results in the reflection of the remainder of the release wave as a compressive wave. In the 1D simulation, we expect to observe the same phenomena since a new free surface will be generated when the bonds between a particle and all its neighbours have been broken. Initially the damage growth in particle-particle bonds was developed

independently of any other bonds in the neighbourhood of the *i*-particle. Figure 5.2 shows longitudinal stress against time for the experimental and simulation data. A plot of the simulation data with no damage model present has also been included to establish the tensile behaviour of the material and for comparison. In the simulation the stress data was plotted for the 3rd particle into the PMMA to avoid a skew in the results due to any noise from the material contact algorithm.
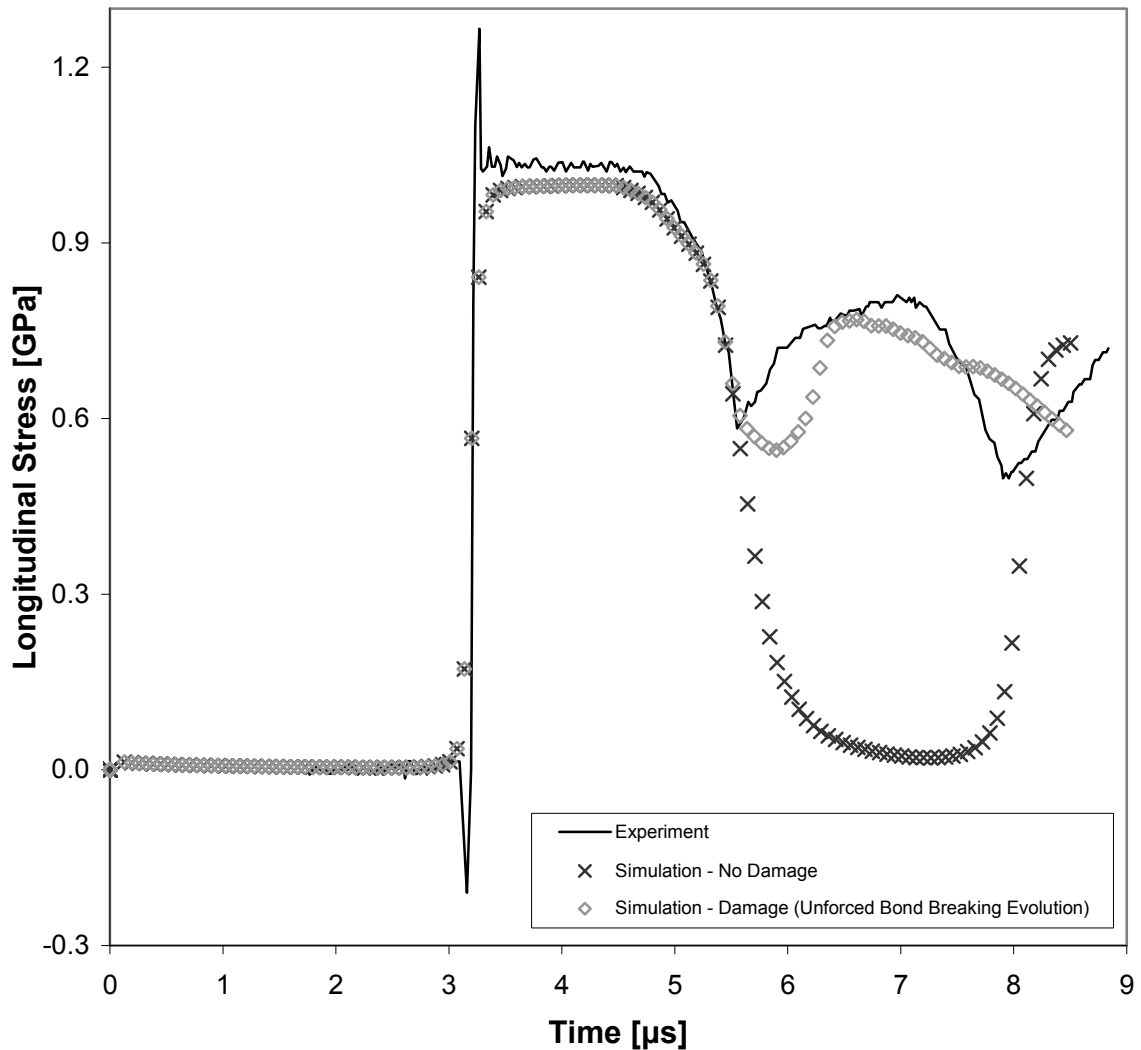
Figure 5.2 – Experiment and simulation results for 1D OFHC Copper plate impact test.

It is clear from the results shown in Figure 5.2 that the damage model causes the particle bonds in the region of the expected spall plane to fail, and a release wave is formed. However the exact shape of the reload signal has not been captured. As mentioned at the end of the previous section,

local damage is likely to have an effect on the global behaviour of the material. This can be represented within this model via a bond-break visibility criterion. Such a criterion is simple to implement in 1D since any particle that is at least *2h* from a free surface, will have an equal number of neighbours either side of it. The criterion states that given that either of the bonds between the *i*-particle and its first neighbour or the *i*-particle and its second neighbour (on one side) have failed ($D_{ij} = 1.0$), then whichever bond is still active is also failed. To simplify; in this simulation case, a particle may have a maximum of 4 neighbours, 2 on each side. Therefore if one bond fails, the other on that side is also failed. Figure 5.3 displays the results of the model with the bond-break criterion active and compares them to the experimental data.
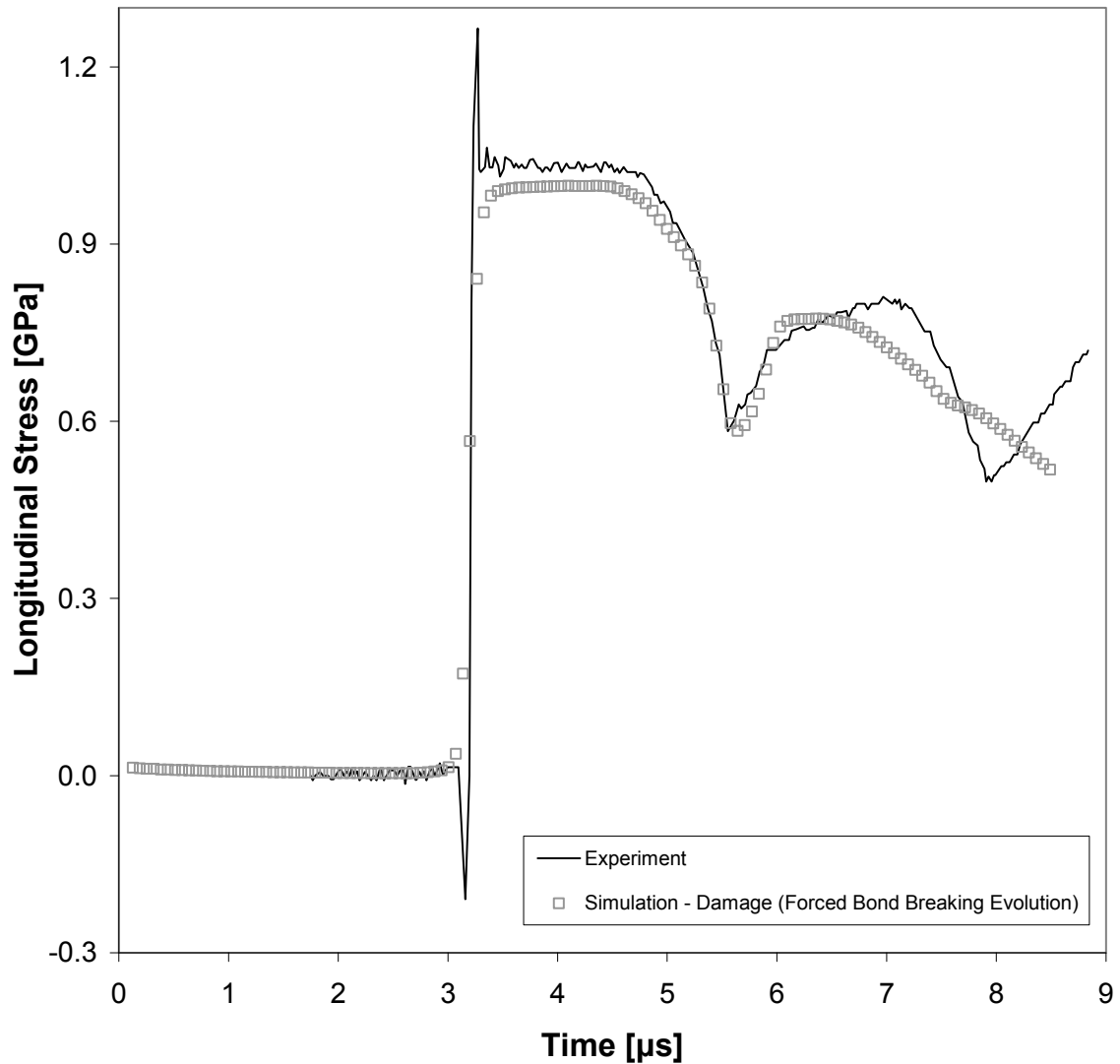
Figure 5.3 – Experiment and simulation results for 1D OFHC Copper plate impact test with a local bond break criterion added to the damage model.

From Figure 5.3 we observe that the reload signal now agrees well with the experimental data. We are not concerned with the rest of the signal for this experiment, since the behaviour after the reload is dependent on the specific set-up of the experiment; for which, specific data are not available. This set-up may not be represented in the simulation model and therefore it is unclear if a correlation should be observed.

6. Conclusions

A damage model for metals that have undergone high strain-rate has been developed. The method is consistent with classical continuum damage mechanics techniques, but unlike the majority of other models, does not require the use of an effective stress to apply the damage. The SPH momentum equation can be rearranged such that it contains a particle-particle interaction area; damage acts to reduce this area, until upon failure (critical damage), the area is set to zero. Direct stress between particle pairs is compared to a spall criterion and a bond is flagged as damaged when this is exceeded. A simplified Cochran-Banner damage growth model has been implemented to evolve the damage parameter and this has been coupled with a criterion that fails multiple neighbour particle bonds when one has reached critical damage. Simulation results are compared to a uniaxial copper plate impact test and the data compares well with the experiment. The model will now be extended to 3D and to a wider range of problems.

Acknowledgements

References

[1]   Xu XP and Needleman A. Numerical simulations of fast crack growth in brittle solids. J Mech Phys Solids 1994;42(9):1397-1434.

[2]   Rabczuk T and Belytschko T. Cracking particles: A simplified meshfree method for arbitrary evolving cracks. Int J Numer Methods Eng 2004;61(13):2316-2343.

[3]   Belytschko T and Black T. Elastic crack growth in finite elements with minimal remeshing. Int J Numer Methods Eng 1999;45(5):601-620.

[4]   Pandolfi A, Krysl P and Ortiz M. Finite element simulation of ring expansion and fragmentation: The capturing of length and time scales through cohesive models of fracture. Int J Fract 1999;95(1-4):279-297.

[5]   Ortiz M and Pandolfi A. Finite-deformation irreversible cohesive elements for three-dimensional crack-propagation analysis. Int J Numer Methods Eng 1999;44(9):1267-1282.

[6]   Zhou F and Molinari JF. Dynamic crack propagation with cohesive elements: A methodology to address mesh dependency. Int J Numer Methods Eng 2004;59(1):1-24.

[7]   Moës N, Dolbow J and Belytschko T. A finite element method for crack growth without remeshing. Int J Numer Methods Eng 1999;46(1):131-150.

[8]   Stolarska M, Chopp DL, Moës N and Belytschko T. Modelling crack growth by level sets in the extended finite element method. Int J Numer Methods Eng 2001;51(8):943-960.

[9]   Moës N, Gravouil A and Belytschko T. Non-planar 3D crack growth by the extended finite element and level sets - Part I: Mechanical model. Int J Numer Methods Eng 2002;53(11):2549-2568.

[10]   Gravouil A, Moës N and Belytschko T. Non-planar 3D crack growth by the extended finite element and level sets - Part II: Level set update. Int J Numer Methods Eng 2002;53(11):2569-2586.

[11]   Belytschko T and Tabbara M. Dynamic fracture using element-free Galerkin methods. Int J Numer Methods Eng 1996;39(6):923-938.

[12]   Krysl P and Belytschko T. The element free Galerkin method for dynamic propagation of arbitrary 3-D cracks. Int J Numer Methods Eng 1999;44(6):767-800.

[13]   Ventura G, Xu JX and Belytschko T. A vector level set method and new discontinuity approximations for crack growth by EFG. Int J Numer Methods Eng 2002;54(6):923-944.

[14]   Rabczuk T and Belytschko T. A three-dimensional large deformation meshfree method for arbitrary evolving cracks. Comput Methods Appl Mech Eng 2007;196(29-30):2777-2799.

[15]   Gingold RA and Monaghan JJ. Smoothed particle hydrodynamics - Theory and application to non-spherical stars. Mon Not R Astron Soc 1977;181(-):375-389.

[16]   Lucy LB. A numerical approach to the testing of the fission hypothesis. Astron J 1977;821013-1024. Dec.

[17]   Libersky LD and Petschek AG. Smooth particle hydrodynamics with strength of materials. In: Trease HE, Fritts MJ and Crowley WP, editors. *Advances in the Free-Lagrange Method Including Contributions on Adaptive Gridding and the Smooth Particle Hydrodynamics Method.* Berlin / Heidelberg: Springer, 1991. p. 248-257.

[18]   Libersky LD, Petschek AG, Carney TC, Hipp JR and Allahdadi FA. High strain lagrangian hydrodynamics a three-dimensional SPH code for dynamic material response. J Comput Phys 1993;109(1):67-75.

[19]   Swegle JW, Attaway SW, Heinstein MW, Mello FJ and Hicks DL. An analysis of smoothed particle hydrodynamics. USA: Sandia National Laboratories, 1994. SAND93-2513.

[20]   Swegle JW, Attaway SW and Hicks DLW. Smoothed particle hydrodynamics stability analysis. J Comput Phys 1995;116(1):123-134.

[21]   Vignjevic R, Campbell J and Libersky L. A treatment of zero-energy modes in the smoothed particle hydrodynamics method. Comput Methods Appl Mech Eng 2000;184(1):67-85.

[22]   Belytschko T. A unified stability analysis of meshless particle methods. Int J Numer Methods Eng 2000;48(9):1359-1400.

[23]   Rabczuk T, Belytschko T and Xiao SP. Stable particle methods based on Lagrangian kernels. Comput Methods Appl Mech Eng 2004;193(12-14):1035-1063.

[24]   Reveles JR. Development of a Total Lagrangian SPH code for the simulation of solids under impact loading. UK: Cranfield University, School of Engineering, PhD, 2007.

[25]   Vignjevic R, Reveles JR and Campbell J. SPH in a total lagrangian formalism. CMES-Comp Model Eng 2006;14(3):181-198.

[26]   Swegle JW. Conservation of momentum and tensile instability in particle methods. USA: Sandia National Laboratories, 2000. SAND2000-1223.

[27]   Kachanov LM. Time of the rupture process under creep conditions. Izv. Akad. Nauk., S.S.R., Otd Tech Nauk 1958;8(8):26-31.

[28]   Lemaitre J. A continuous damage mechanics model for ductile fracture. J Eng Mater Technol Trans ASME 1985;107(1):83-89.

[29]   Cochran S and Banner D. Spall Studies in Uranium. J Appl Phys 1977;48(7):2729-2737.

[30]   Malvern LE. Introduction to the Mechanics of a Continuous Medium. USA: Prentice-Hall Inc, 1969.

[31]   Mirkovic J. Modelling of Nonlinear Behaviour of Metallic Structure Components. UK: Cranfield University, School of Engineering, PhD, 2004.

[32]   Panov V. Modelling of behaviour of metals at high strain rate. UK: Cranfield University, School of Engineering, PhD, 2006.

[33]   Steinberg DJ. Equation of State and Strength Properties of Selected Materials. USA: Laurence Livermore National Laboratory, 1996. UCRL-MA-106439.

This page is intentionally blank

Appendix C – The Preston Tonks Wallace Model

This appendix provides a brief summary of the Preston-Tonks-Wallace model, for a complete explanation please refer to the original paper (Preston, Tonks *et al*. 2003). This summary was also written with the aid of Hanson (2004). A comparison of this model to the MTS model considered for this research can be found in Section 4.5.

The PTW model is a very recent model (2003) that describes the plastic deformation of metals at very high strain rates (up to hypervelocity impact; $10^{-3} \leq \dot{\psi} \leq 10^{12}$) in terms of the dependence of plastic stress on plastic strain. The plastic stress in a material is a function of equivalent plastic strain, ψ, the strain rate, $\dot{\psi}$, the temperature, T and the density, ρ, it is assumed to be independent of the history of the material. The plastic flow is assumed to be isotropic.

There are three constitutive relationships, which are all scaled dimensionless variables; the scaled dimensionless stress variable, temperature variable and strain rate variable.

The dimensionless stress variable, $\hat{\tau}$ is as follows:

$$\hat{\tau} = \frac{\tau}{G(\rho, T)}$$

and

$$\tau = \frac{\sigma}{2},$$

where τ is the flow stress, G is the shear modulus and σ is the von Mises equivalent deviatoric stress.

The dimensionless temperature variable, $\hat{T}$ is a function of material temperature and the melting temperature, T_m, which is a function of the density.

For plastic flow $\hat{T} < 1$, it is given as follows:

$$\hat{T} = \frac{T}{T_m(\rho)}.$$

The shear modulus, G which is a function of the density, and temperature, can now be shown as follows:

$$G(\rho, T) = G_0(\rho)\left(1 - \alpha\hat{T}\right),$$

where $G_0(\rho)$ is the shear modulus at zero Kelvin and α is a material parameter.

The dimensionless strain rate variable is $\dot{\psi}/\dot{\xi}$, where $\dot{\xi}$ is the reciprocal of the time for a transverse sound wave to cross an atom and is given as follows:

$$\dot{\xi} = \frac{1}{2}\left(\frac{4\pi\rho}{3M}\right)^{\frac{1}{3}}\left(\frac{G}{\rho}\right)^{\frac{1}{2}}$$

and

$$M = \frac{A}{A_{vo}},$$

where M is the atomic mass of the metal, A is the atomic weight and A_{vo} is Avogadro's constant, $A_{vo} = 6.025 \times 10^{23}$ g mol^{-1}.

When strain rate and temperature are held constant, the dimensionless stress constant ranges between two limits, the yield stress, $\hat{\tau}_y$ and the saturation stress, $\hat{\tau}_s$, therefore functional form for $\hat{\tau}$ depends on the strain thus:

$$\hat{\tau} = \hat{\tau}_s + \frac{1}{p}\left(s_0 - \hat{\tau}_y\right)\ln\left[1 - \left[1 - \exp\left(-p\frac{\hat{\tau}_s - \hat{\tau}_y}{s_0 - \hat{\tau}_y}\right)\right]\exp\left(-\frac{p\theta\psi}{\left(s_0 - \hat{\tau}_y\right)\left[\exp\left(p\frac{\hat{\tau}_s - \hat{\tau}_y}{s_0 - \hat{\tau}_y}\right) - 1\right]}\right)\right],$$

where p and θ are material parameters specific to each material, the parameter s_0 is explained below.

The values of $\hat{\tau}_y$ and $\hat{\tau}_s$ depend on the strain rate applied to the material. For low strain rates, the plastic deformation process is controlled by thermal activation and the values of $\hat{\tau}_y$ and $\hat{\tau}_s$ are given as follows:

$$\hat{\tau}_y^L = y_0 - \left(y_0 - y_\infty\right)\mathrm{erf}\left[\kappa\hat{T}\ln\left(\frac{\gamma\dot{\xi}}{\dot{\psi}}\right)\right],$$

$$\hat{\tau}_s^L = s_0 - \left(s_0 - s_\infty\right)\mathrm{erf}\left[\kappa\hat{T}\ln\left(\frac{\gamma\dot{\xi}}{\dot{\psi}}\right)\right],$$

where κ and γ are dimensionless material parameters, the superscript L indicates the low strain rate criteria. The parameters y_0 and y_∞ are the values of $\hat{\tau}_y$ at zero temperature and very high temperature respectively. s_0 and s_∞ have analogous meanings for $\hat{\tau}_s$.

At very high strain rate, i.e. $\dot{\psi} > 10^8\ \mathrm{s^{-1}}$ the process of plastic deformation is described by Wallace (Wallace 1981), the values of $\hat{\tau}_y$ and $\hat{\tau}_s$ are given as follows (the superscript H indicates the high strain rate criteria):

$$\hat{\tau}_s^H = s_0\left(\frac{\dot{\psi}}{\gamma\dot{\xi}}\right)^\beta,$$

$$\hat{\tau}_y^H = \hat{\tau}_s^H.$$

$\hat{\tau}_y$ requires a transition at intermediate strain rates and therefore (the superscript M indicates the intermediate strain rate criteria):

$$\hat{\tau}_y^M = y_1\left(\frac{\dot{\psi}}{\gamma\dot{\xi}}\right)^{y_2},$$

where β, y_1 and y_2 are transition parameters.

The PTW model implements these transitions by the following relations:

$$\hat{\tau}_s = \mathbf{max}\left(\hat{\tau}_s^L, \hat{\tau}_s^H\right), \text{ and } \hat{\tau}_y = \mathbf{max}\left(\hat{\tau}_y^L, \mathbf{min}\left(\hat{\tau}_y^M, \hat{\tau}_y^H\right)\right).$$

Appendix D – Source Code for Radial SPH Mesh Generation

This appendix provides the source code for a program that was written to generate a radial SPH mesh of particles that adhere to the size and shape of the Mock-Holt experiment. The code was used to develop the simulations discussed in Section 6.2.5.

The code evenly distributes 3D SPH particles around the circumference of concentric rings with expanding radii. The final radius and the number of concentric rings within the outer ring can all be specified via the set-up parameters at the start of the code. The completed cross-section is extruded to form a cylinder by duplicating the cross-section and increasing the particles' z-positions. The number of rows and final length of the cylinder are also all defined at the start of the code.

Since the explosive cylinder in the Mock-Holt Experiment extrudes from both the top and bottom of the iron cylinder, the code starts generating extra rings of particles to a different final radius after an input defined offset and stop after another offset, these particles are entered in a separate material.

The SPH smoothing length for each particle is calculated as the cube root of the volume of each particle multiplied by the h-scale factor (currently 1.3). The figure on the next page shows the exact output of the code included in this appendix diagrammatically. The output is written to a file 'OUTPUT' that contains the MCM input format for 3D SPH particles.

Mock-Holt
Experiment No. 1

Iron Cylinder

Explosive

Iron Radius: 5.72 cm
Explosive Radius: 3.81 cm
Iron thickness: 1.91 cm

64 mm

203 mm

90 mm
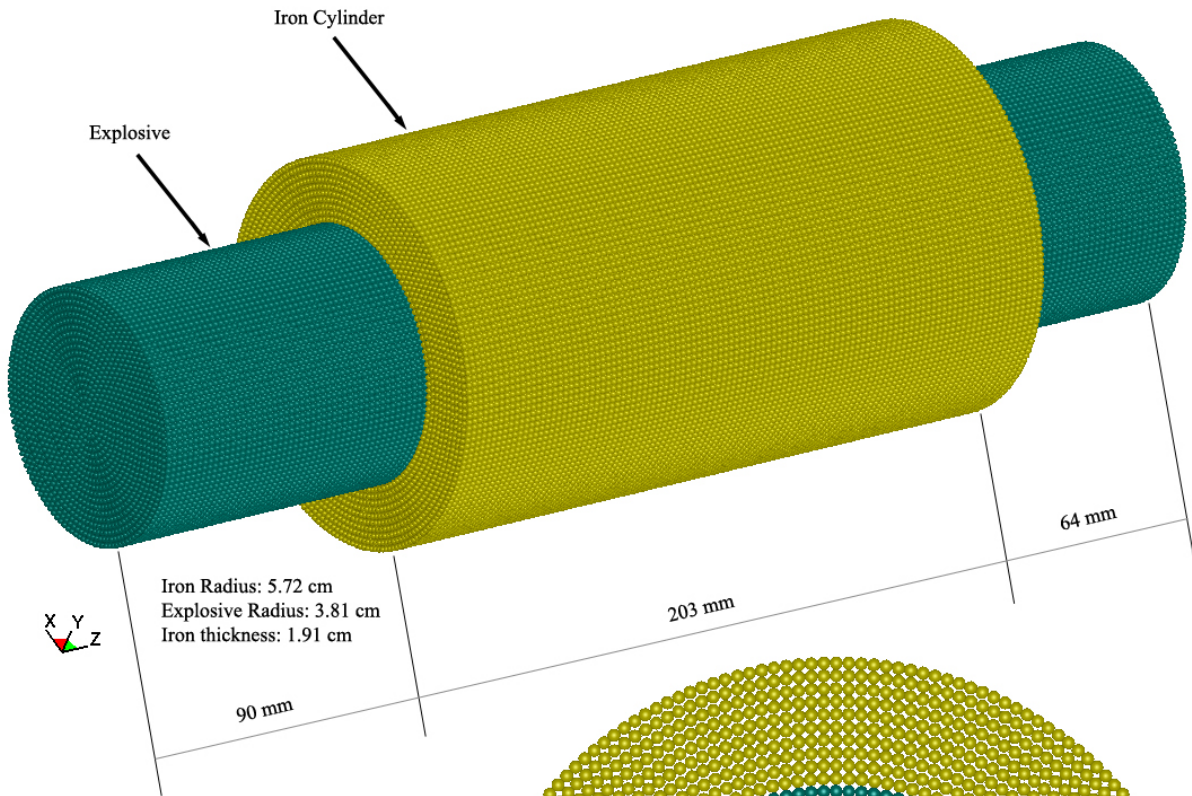
Mock-Holt Experiment
(Radial Mesh)

Number of Particles

Total: 426,247
Iron: 174,727
Explosive: 251,520

x/y Direction
Iron Face: 1,603
Iron Rings: 10
Explosive Face: 1,310
Explosive Rings: 20 + Center Particle

z Direction
Iron Rows: 109
Explosive Rows: 192

Average Inter-Particle Spacing: 1.88 mm
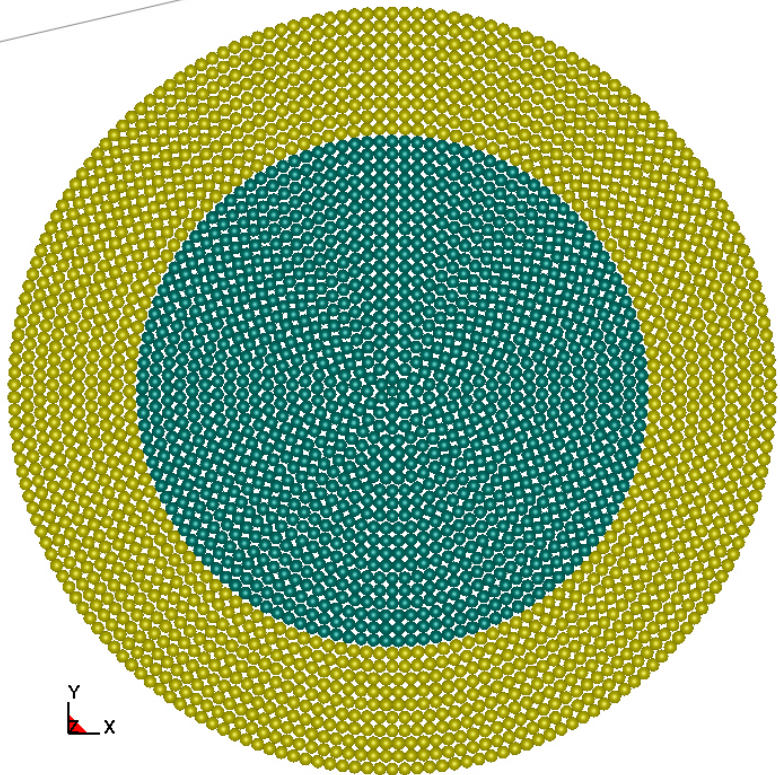
```fortran
PROGRAM rad_mesh
!******************************************************************************
!                                                                            *
!     Raidal Mesh Generator                                                  *
!     A 3D cylndrical mesh generator                                         *
!                                                                            *
!     Author: Russell K. Strand                                              *
!                                                                            *
!     Version: 1.0                                                           *
!                                                                            *
!     Date of last modification: 01-04-2008                                  *
!                                                                            *
!     School of Engineering                                                  *
!     Cranfield University                                                   *
!     Cranfield                                                              *
!     Bedfordshire                                                           *
!     MK43 0AL                                                               *
!                                                                            *
!     Copyright 2008-2009 Cranfield University                              *
!     All rights reserved.                                                   *
!                                                                            *
!     This program is intended as mesh generating tool for generating        *
!     3D SPH particles for use with Cranfield University's MCM code.         *
!                                                                            *
!     Assumes cm / g / us units                                             *
!                                                                            *
!******************************************************************************
!
IMPLICIT NONE
!
! Define minimum floating point accuracy
INTEGER, PARAMETER :: real_acc = SELECTED_REAL_KIND(P=15,R=50)
INTEGER, PARAMETER :: d=real_acc
INTEGER :: lcount
CHARACTER (LEN=80) :: txts
!
INTEGER :: nid, n, i, j, k, npar, totpar, zpar, radpar2, radpar1
REAL(KIND=real_acc) :: zmax, pz, radmax2, pxy2, rho2, radmax1, rho1,     &
                       pi, r, C, theta, inc_theta, hscale, rold, pxy1,  &
                       r2, r1, vring, vi, newz, vtot1, masstot1, third, &
                       vtot2, masstot2
CHARACTER (LEN=16) :: output
REAL(KIND=real_acc), DIMENSION(3,1000000) :: x          ! Limited to 1 million nodes
REAL(KIND=real_acc), DIMENSION(1000000) :: mass, h, V ! Limited to 1 million nodes
INTEGER, DIMENSION(1000000) :: mid                      ! Limited to 1 million nodes
!
! Define output file name
output='radgrid.mcm'
!
OPEN(UNIT=13, FILE=output, STATUS='new', FORM='formatted')
!
!-------------------------------------------------------------------
! Set-up cylinder parameters
!-------------------------------------------------------------------
!
nid = 1                     ! Start point for nodes
newz = 0.0_d                ! Initial z co-ordinate (currently this program assumes
                            ! cylinder to extrude in z direction)
zmax = 35.7_d               ! Length of cylinder (currently length of Mock-Holt explosive)
zpar = 192                  ! Number of particles through cylinder length (z-direction)
pz = zmax / zpar            ! Inter-particle spacing in z-direction (currently 0.1859)
!
radmax2 = 3.81_d            ! Radius of explosive (currently radius of Mock-Holt explosive)
radpar2 = 20               ! Number of rows of particles through explosive radius
pxy2 = radmax2 / (radpar2 + 0.5_d) ! Inter-particle spacing in x/y direction for
                                    ! explosive (should be similar to pz - currently 0.1856)

radmax1 = 1.91_d           ! thickness of next cylinder (currently thickness of Mock-Holt iron)
radpar1 = 10               ! Number of rows of particles through iron radius
pxy1 = (radmax1 + (pxy2 / 2.0_d) ) / (radpar1 + 0.5_d) ! Inter-particle spacing in x/y direction
                                                        ! for iron (should be similar to pz -
                                                        ! currently 0.1908)
```

```fortran
rho1 = 7.890_d              ! Material density of iron
rho2 = 1.717_d              ! Material density of explosive
hscale = 1.3_d              ! Multiplier for smoothing length
pi = 3.14159265358979_d     ! PI
!
third = 1.0_d / 3.0_d       ! 1/3
!
!-------------------------------------------------------------------
! Calculate particle positions
!-------------------------------------------------------------------
!
DO i=1,zpar                    ! Loop over number of rows in z-direction
   !
   r = 0.0_d                   ! Initialise radius for new z increment
   rold = 0.0_d                ! Initialise previous radius to centre of cylinder
   !
   DO j=1,radpar2              ! Loop over number of rings from centre to outer edge of cylinder
      !
      IF (j .EQ. 1) THEN
         Vring = pi * pz * (pxy2 / 2.0_d)**2 ! Special case of volume for centre particle
         Vi = Vring ! In this case the volume of the ring is the volume of the particle
         !
         x(1,nid) = 0.0_d          ! Set x co-ordinate for centre particle in array
         x(2,nid) = 0.0_d          ! Set y co-ordinate for centre particle in array
         x(3,nid) = newz           ! Set z co-ordinate to current z value in array
         mass(nid) = rho2 * Vi     ! Calculate mass for particle and add to array
         h(nid) = hscale * Vi**third ! Calculate h for particle and add to array
         V(nid) = Vi               ! Add volume to volume array
         mid(nid) = 2              ! Material number is 2 (for explosive)
         !
         nid = nid + 1             ! Increment node number
      ENDIF
      !
      r = r + pxy2               ! Increment Radius
      C = 2 * pi * r             ! Calculate circle circumference
      npar = INT(C / pxy2)       ! Calculate no. of particles around circumference
      theta = (2 * pi) / npar    ! Calculate angle between particles in radians
      inc_theta = theta          ! Calculate angle increment for each particle around row
      !
      r2 = r + (pxy2 / 2.0_d)        ! Radius of outer edge of particle plane
      r1 = rold + (pxy2 / 2.0_d)     ! Radius of inner edge of particle plane
      Vring = pi * pz * ( r2**2 - r1**2 ) ! Volume of current ring of particles
      Vi = vring / npar              ! Calculate individual particle volume from ring volume
      !
      DO k=1,npar                ! Loop over number particles in row
         !
         x(1,nid) = r * SIN(theta)   ! Calculate x co-ordinate and add to array
         x(2,nid) = r * COS(theta)   ! Calculate y co-ordinate and add to array
         x(3,nid) = newz             ! Set z co-ordinate to current z value
         mass(nid) = rho2 * Vi       ! Calculate mass for particle and add to array
         h(nid) = hscale * Vi**third ! Calculate h for particle and add to array
         V(nid) = Vi                 ! Add volume to volume array
         mid(nid) = 2                ! Material number is 2 (for explosive)
         !
         theta = theta + inc_theta   ! Increment theta to next position around circle
         nid = nid + 1               ! Increment node number
         !
      ENDDO
      !
      rold = r                       ! Set current radius to old radius for next loop
      !
   ENDDO
   !
   ! Include Iron Ring if correct z distance from ends
   IF (newz .GE. 9.0_d .AND. newz .LE. 29.3_d) THEN ! positioning of mock-holt iron along explosive
      DO j=1,radpar1           ! Loop over number of rings from centre to outer edge of cylinder
         !
         IF (j .EQ. 1) THEN
            r = r + pxy1                 ! Increment Radius
            C = 2 * pi * r               ! Calculate circle circumference
            npar = INT(C / pxy1)         ! Calculate no. of particles around circumference
            theta = (2 * pi) / npar      ! Calculate angle between particles in radians
            inc_theta = theta            ! Calculate angle increment for each particle around row
            !
```

```fortran
              r2 = r + (pxy1 / 2.0_d)         ! Radius of outer edge of particle plane
              r1 = 3.81_d                     ! Radius of outer edge of explosive
              Vring = pi * pz * ( r2**2 - r1**2 ) ! Volume of current ring of particles
              Vi = vring / npar               ! Calculate individual particle volume from ring volume
              !
              DO k=1,npar                     ! Loop over number particles in row
                 !
                 x(1,nid) = r * SIN(theta)    ! Calculate x co-ordinate and add to array
                 x(2,nid) = r * COS(theta)    ! Calculate y co-ordinate and add to array
                 x(3,nid) = newz              ! Set z co-ordinate to current z value
                 mass(nid) = rho1 * Vi        ! Calculate mass for particle and add to array
                 h(nid) = hscale * Vi**third  ! Calculate h for particle and add to array
                 V(nid) = Vi                  ! Add volume to volume array
                 mid(nid) = 1                 ! Material number is 2 (for explosive)
                 !
                 theta = theta + inc_theta    ! Increment theta to next position around circle
                 nid = nid + 1                ! Increment node number
                 !
              ENDDO
              !
           ELSE
              !
              r = r + pxy1                    ! Increment Radius
              C = 2 * pi * r                  ! Calculate circle circumference
              npar = INT(C / pxy2)            ! Calculate no. of particles around circumference
              theta = (2 * pi) / npar         ! Calculate angle between particles in radians
              inc_theta = theta               ! Calculate angle increment for each particle around row
              !
              r2 = r + (pxy1 / 2.0_d)         ! Radius of outer edge of particle plane
              r1 = rold + (pxy1 / 2.0_d)      ! Radius of inner edge of particle plane
              Vring = pi * pz * ( r2**2 - r1**2 ) ! Volume of current ring of particles
              Vi = vring / npar               ! Calculate individual particle volume from ring volume
              !
              DO k=1,npar                     ! Loop over number particles in row
                 !
                 x(1,nid) = r * SIN(theta)    ! Calculate x co-ordinate and add to array
                 x(2,nid) = r * COS(theta)    ! Calculate y co-ordinate and add to array
                 x(3,nid) = newz              ! Set z co-ordinate to current z value
                 mass(nid) = rho1 * Vi        ! Calculate mass for particle and add to array
                 h(nid) = hscale * Vi**third  ! Calculate h for particle and add to array
                 V(nid) = Vi                  ! Add volume to volume array
                 mid(nid) = 1                 ! Material number is 2 (for explosive)
                 !
                 theta = theta + inc_theta    ! Increment theta to next position around circle
                 nid = nid + 1                ! Increment node number
                 !
              ENDDO
              !
           ENDIF
           !
           rold = r                          ! Set current radius to old radius for next loop
           !
        ENDDO
        !
     ENDIF
     !
     newz = newz + pz                  ! Increment z co-ordinate and repeat loop
     !
  ENDDO
  !
  !-------------------------------------------------------------------
  ! Write data to output
  !-------------------------------------------------------------------
  !
  totpar = nid - 1                    ! Remove extra 1 added at end of last loop
  Vtot1 = 0.0_d                       ! Initialise total volume for material 1
  Vtot2 = 0.0_d                       ! Initialise total volume for material 2
  masstot1 = 0.0_d                    ! Initialise total mass for material 1
  masstot2 = 0.0_d                    ! Initialise total mass for material 2
  !
  DO i=1,totpar
     IF (mid(i) .EQ. 1) THEN
        Vtot1 = Vtot1 + V(i)          ! Sum all particle volumes in material 1
        masstot1 = masstot1 + mass(i) ! Summ all particle masses in material 1
```

```fortran
    ELSE
       Vtot2 = Vtot2 + V(i)              ! Sum all particle volumes in material 2
       masstot2 = masstot2 + mass(i)   ! Summ all particle masses in material 2
    ENDIF
ENDDO
!
WRITE(*,30) vtot1, masstot1, vtot2, masstot2 ! Print total volume and mass to screen
!
! Write out individual particle coordinates to output
DO i=1,totpar
   WRITE(13,10) i, 0, x(1,i), x(2,i), x(3,i), mid(i) ! Write particle data to output
ENDDO
!
! Write out individual particle masses and smoothing lengths to output
DO i=1,totpar
   WRITE(13,20) i, mass(i), h(i)
ENDDO
!
10 FORMAT(I8,I5,3E20.12,I7)
20 FORMAT(I8,2E20.12)
30 FORMAT('Total volume of material 1: ',E14.7,/  &
          'Total mass of material 1:   ',E14.7,// &
          'Total volume of material 2: ',E14.7,/  &
          'Total mass of material 2:   ',E14.7)
!
END PROGRAM rad_mesh
```