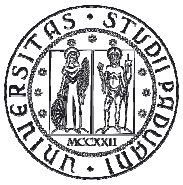

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Dipartimento di Ingegneria Civile, Edile e Ambientale

SCUOLA DI DOTTORATO DI RICERCA IN: Scienze dell'Ingegneria Civile e Ambientale
CICLO XXIV

RADIATION DAMAGE ASSESSMENT FOR THE CONCRETE SHIELDING OF SPES, A NUCLEAR FACILITY FOR THE SELECTIVE PRODUCTION OF EXOTIC SPECIES

Direttore della Scuola: Ch.mo Prof. Stefano Lanzoni

Supervisori: Ch.mo Prof. Carmelo Majorana
Dott. Fabiana Gramegna
Dott. Ing. Valentina Salomoni

Dottoranda: Beatrice Pomaro

Dicembre 2011

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Dipartimento di ____Ingegneria Civile, Edile e Ambientale____

SCUOLA DI DOTTORATO DI RICERCA IN: Scienze dell'Ingegneria Civile e Ambientale
CICLO XXIV

RADIATION DAMAGE ASSESSMENT FOR THE CONCRETE SHIELDING OF SPES, A NUCLEAR FACILITY FOR THE SELECTIVE PRODUCTION OF EXOTIC SPECIES

Direttore della Scuola: Ch.mo Prof. Stefano Lanzoni

Supervisori: Ch.mo Prof. Carmelo Majorana
Dott. Fabiana Gramegna
Dott. Ing. Valentina Salomoni

Dottoranda: Beatrice Pomaro

Aim of this Thesis

Our effort in this thesis was paid, first, at the definition and validation of a law for radiation damage, inclusive of the phenomenological aspects envisaged in experimental campaigns relative to irradiated concretes.

Secondarily, at the simulation of the work scenario for a specific case study: a facility under design for the Selective Production of Exotic Species (SPES) at the Legnaro National Laboratory (Padova), thus pointing out the limiting factors for the durability of the concrete structure, which was the main issue of this work lead in conjunction with the Department of Structural and Transportations Engineering and the National Institute of Nuclear Physics (INFN) of Legnaro (Padova).

Sì, come il ferro s'arruginisce sanza esercizio
così lo 'ngegno sanza esercizio si guasta.

— Leonardo da Vinci

Desidero ringraziare i miei Supervisori, per l'argomento di ricerca accordatomi e, con esso, l'opportunità di mettermi in gioco con temi di studio così all'avanguardia. Grazie al Prof. Majorana per avermi guidato con tanta preparazione e passione nel lavoro di ricerca e per l'incessante incoraggiamento alla diffusione dei nostri risultati. Grazie alla Dott. Ing. Salomoni per il supporto costante, gli innumerevoli suggerimenti, la stima e la fiducia sempre dimostratami. Grazie alla Dott. Gramegna per la fondamentale condivisione delle conoscenze in campo fisico, l'entusiasmo prestato nel seguire il lavoro e le tante proposte formative rese possibili.

Un sentito grazie al Prof. Massimo Cuomo per aver letto il mio lavoro e aver contribuito alla sua stesura con consigli e osservazioni.

Grazie al Dott. Mastinu, all'Ing. Sarchiapone, al Dott. Zafiropoulos per l'ampia disponibilità e collaborazione.

Ringrazio nella forma de 'il forum FLUKA' gran parte degli sviluppatori del codice e degli utenti esperti, per la chiarezza e la tempestività nella risoluzione dei miei dubbi e per la dedizione dimostrata in occasione della preziosa esperienza di workshop per l'utilizzo del codice.

Grazie ai miei amici colleghi, che hanno arricchito non poco questo periodo di Dottorato, per gli scambi di idee e la loro contagiosa allegria.

Dedico questo lavoro ai miei familiari, ai quali vanno immense dosi di affetto e gratitudine.

Contents

List of Tables

List of Figures

Riassunto

L'attività di ricerca si inserisce nell'ambito di una collaborazione del Dipartimento di Costruzioni e Trasporti dell'Università di Padova con l'Istituto Nazionale di Fisica Nucleare (INFN), Laboratori Nazionali di Legnaro (Padova), finalizzata alla messa a punto del progetto SPES (Selective Production of Exotic Species), per la costruzione nei suddetti laboratori di un impianto nucleare di ultima generazione per la produzione di speciali fasci di ioni radioattivi, detti specie esotiche, a scopi di ricerca in campo fisico, astrofisico e auspicabile applicazione in campo medico.

L'argomento di studio è il rivestimento in calcestruzzo per un impianto di questo tipo e le problematiche connesse all'irraggiamento da neutroni a seguito delle reazioni di fissione nucleare generate dalla collisione di un fascio primario di protoni su un elemento bersaglio in uranio-carbonio.

Lo studio può suddividersi idealmente in tre fasi:

a) una prima fase di ricerca nella letteratura scientifica di settore (articoli e manuali);
b) una successiva fase di implementazione numerica e validazione in apposito codice agli elementi finiti di una legge di danno da radiazione sul calcestruzzo, basata su sperimentazione;
c) una terza fase di applicazione del modello numerico al caso di studio, il progetto SPES, con la necessaria definizione delle condizioni al contorno per il calcestruzzo esposto, dovute alle condizioni di progetto del macchinario. Ciò ha richiesto un uso congiunto della tecnica Monte Carlo e del modello FEM.

L'iniziale indagine bibliografica ha coinvolto la definizione dello stato dell'arte sui materiali di schermatura impiegati in campo nucleare e la raccolta di dati sperimentali di irraggiamento neutronico su campioni in calcestruzzo.

Il primo punto ha permesso di considerare, a fianco del calcestruzzo ordinario, l'impiego di altri impasti migliorati per la presenza o di aggregati idrati, in grado di ritenere il loro contenuto d'acqua anche ad alte temperature, o di aggregati pesanti (di natura ferritica o baritica soprattutto); la prima caratteristica garantisce una buona capacità schermante nei confronti dei neutroni, essendo in grado l'idrogeno contenuto nelle molecole d'acqua di assorbire dopo pochi eventi di scattering una grande aliquota dell'energia incidente di un neutrone; la

seconda caratteristica è indice di una buona prestazione schermante nei confronti dei raggi gamma, indesiderato prodotto secondario delle reazioni atomi-neutroni.
Il secondo punto ha condotto alla comprensione dei meccanismi di deterioramento del calcestruzzo sotto un ambiente irraggiato e alla quantificazione della soglia di flusso neutronico oltre la quale si hanno le prime manifestazioni macroscopiche di perdita di resistenza del materiale, valutata in termini di resistenza a compressione, a trazione e modulo elastico.

La legge di danno ricercata è stata definita come la curva di inviluppo del decadimento del modulo elastico di calcestruzzo esposto, rispetto al materiale vergine, proveniente da diversi tests, in diverse condizioni sperimentali, che, tuttavia, hanno permesso di identificare un trend univoco, in funzione del flusso di neutroni, di questo parametro macroscopico (il modulo elastico), scelto in accordo alla teoria dello stress effettivo di Kachanov.
La legge è stata implementata in un preesistente codice FEM che numericamente risolve il problema termo-igro-meccanico accoppiato per i mezzi porosi multifase, come si configura il calcestruzzo. Il materiale è qui modellato nel suo comportamento visco-elastico danneggiato, in cui le forme di danno possibili provengono dal carico meccanico, dal carico termico e, grazie all'upgrade prefissato, dal campo di radiazione nucleare.
La validazione è stata fatta sulla base di una prova di irraggiamento reperita in letteratura per calcestruzzo serpentinitico, sottoposto a due diversi flussi neutronici; nel complesso la risposta del materiale irraggiato è ben colta dal modello numerico, in termini di legame tensioni-deformazioni.
A questa ha seguito un'analisi comparativa sulla bontà di schermatura del calcestruzzo ordinario, rispetto a provini fatti di impasti migliorati individuati anch'essi da letteratura, nell'ipotesi che il campo di radiazione spazialmente segua un modello semplificato monodimensionale, nella fattispecie noto in letteratura come modello diffusivo per i neutroni a bassa energia o termici; un modello analogo è definibile per i neutroni ad alta energia o veloci, sulla base della cosiddetta teoria dei due gruppi, che assume la suddivisione dello spettro neutronico reale in due soli livelli energetici, in una logica di semplificazione computazionale della teoria generale del trasporto della radiazione nella materia.
L'evoluzione monodimensionale è accettabile se si analizza l'attenuazione della radiazione lungo lo spessore di una parete uniformemente investita, geometria che non si discosta dal

caso di studio, il quale considera una porzione di parete orientata in direzione del fascio, per la quale la sezione trasversale è la faccia esposta e l'attenuazione avviene lungo lo spessore.
A parità di flusso incidente, l'analisi comparativa ha messo in luce valori di danno superiori in presenza di neutroni veloci, rispetto a quelli dati da neutroni termici e ha permesso di quantificare le migliori prestazioni degli impasti speciali, rispetto al calcestruzzo ordinario.

Successivamente si è preso in considerazione l'elemento sensibile dell'impianto in progetto per i Laboratori Nazionali di Legnaro, ovvero il vano ospitante l'elemento bersaglio e sede delle reazioni di fissione. La geometria dell'impianto è stata ricreata in un codice Monte Carlo di ricerca, un brevetto CERN-INFN di Milano in grado di effettuare calcoli 3D di trasporto della radiazione, allo scopo di ricavare i campi di radiazione e temperatura attesi per SPES nelle condizioni di lavoro di progetto per il macchinario.
L'uso congiunto della tecnica Monte Carlo con il codice FEM, modificato per tenere in conto gli effetti del campo di radiazione nel materiale, ha consentito di definire l'aspetto termico, ossia lo sviluppo di calore interno al materiale per effetto del deposito di energia da radiazione, il fattore limitante per descrivere uno scenario di lavoro compatibile con la durabilità del calcestruzzo.
Le simulazioni hanno condotto alla definizione di un profilo di irraggiamento ammissibile di cicli continuativi al più di 6-7 mesi all'anno, per un quinquennio, alle specifiche di progetto del sistema di accelerazione e del fascio primario sull'elemento fissile, in assenza di ulteriori provvedimenti o dispositivi di attenuazione del fronte termico, come ad esempio predisposizione di liners metallici all'intradosso delle pareti direttamente investite della camera di fissione o impianti di raffreddamento annegati in parete.

Summary

The research activity moves from a partnership between the Department of Structural and Transportation Engineering of the University of Padova and the National Institute of Nuclear Physics (INFN), National Laboratories of Legnaro (Padova), aimed at the design of the SPES project (Selective Production of Exotic Species), for the construction at the INFN laboratories of Legnaro of a next-generation nuclear facility, dedicated to the production of a special kind of radioactive ion beams called exotic species. Their study is expected to provide advances in physics and astrophysics and a possible medical application of these beams is foreseen, as well.

The topic of our study is the concrete shielding surrounding such a facility and the deeper understanding of the physical aspects involved in the radiation exposure of concrete to nuclear radiation generated by the fission reactions due to a primary proton beam impinging a uranium-carbide target.

The study can be ideally subdivided into three phases:

a) a bibliographic research within the scientific production in the field of nuclear physics and concrete behaviour (articles and manuals);
b) the numerical implementation and validation within an ad hoc finite element code of a damage law, experimentally based, for concrete under neutron radiation;
c) the application of the numerical model developed so far to the SPES project, provided the boundary conditions for the exposed concrete have been defined, in accordance with the design work specifics of the facility. This phase has required the combined use of the Monte Carlo technique and the FEM model.

The previous bibliographic investigation has lead mainly to define the status of the art of the shielding materials employed in nuclear engineering and collect experimental data of irradiated concrete samples.

As for the first point, the use of some improved mixtures has been envisaged to be of interest in the common practice so that some special concretes have been considered, in addition to ordinary Portland concrete; they differ basically for the presence of hydrous aggregates in the mixture, which are able to retain their water content even at high temperature, or heavy aggregates (iron-based or barytes aggregates, mainly). The former are known to assure good

shielding properties against neutrons, since the atoms of hydrogen in the water molecules can easily absorb a huge amount of the incident kinetic energy of neutrons in a few scattering events; the latter provide, generally, good shielding properties against gamma rays, unsought by-product of atom-neutron reactions.

The second point of the study allowed us to better understand the degradation mechanisms for concrete under an irradiated environment and quantify the threshold value for neutron fluence, marking the beginning of the macroscopic decay in the mechanical properties, in terms of compression and tension strength and Young modulus.

The required damage law has been defined based on the enveloping curve of several empirical data describing the behaviour of the Young modulus of exposed concrete, with respect to that proper of the virgin material, in function of the neutron fluence; this macroscopic parameter for defining radiation damage has been chosen in agreement with the effective stress theory by Kachanov.

The damage law here introduced has been implemented in a research FEM code able to solve the coupled hygro-thermo-mechanical problem in multiphase porous systems like concrete. Concrete is modeled by the code in its visco-elastic behaviour, taking into account damage effects due to mechanical loads, thermal loads and, thanks to the upgrade, a surrounding radiation field.

The validation of the code has been accomplished by reproducing a particular irradiation test found in the scientific literature and involving serpentinitic concrete samples subject to two different neutron fluences; the overall stress-strain trend numerically simulated is found in good agreement with the data of the empirical investigation.

A comparative analysis has followed the validation phase, aimed at studying the shielding performance of ordinary concrete, with respect to that of the improved mixtures, for a given radiation field, which is assumed to follow the simplified one-dimensional model known as diffusive model for thermal neutrons; a similar model is defined for high energy neutrons, i.e. fast neutrons, based on the so-called two-group theory. The one-dimensional simplification above mentioned is meant to be acceptable if the attenuation of radiation is considered to happen along the thickness of a wall, which is the geometry under analysis in the following: as explained later on, a portion of the directly impinged wall has been modeled with the FEM code and the attenuation process has been studied along the beam direction.

The incident flux being equal, the comparative analysis has shown higher damage values for concrete specimens under fast neutrons than thermal and, as expected, better shielding characteristics have been found for the special concretes than ordinary Portland.

The third phase of our work has focused the target room, the most critical area of the facility under design for the National Laboratories of Legnaro, dedicated to the fission reactions; it has been modeled in a Mote Carlo environment through a special software developed by CERN and INFN of Milan; the statistical tool is able to handle 3D radiation transport calculations and it has been exploited for our purposes to define the radiation and the temperature field from the design specifics of the SPES facility.

The combined use of the Monte Carlo technique and the FEM code, upgraded to take into account the radiation exposure effects on concrete, has allowed us to identify in the thermal aspect, i.e. the temperature rise in the shielding due to radiation energy deposition, the most severe factor for prescribing a work scenario consistent with concrete durability.

The numerical have allowed us to quantify an admissible irradiation profile of up to 6-7 months per year, for five years, under the design characteristics of the accelerating system and of the primary beam, not considering any protective device, such as outer metallic liners working as coats for the biological shield or the presence of a cooling systems inside the walls.

Introduction

The study performed during this work of thesis has required the employment of a numerical code, called NEWCON3D, assessing the behaviour of concrete when exposed to specific hygro-thermal and mechanical conditions; the presence of a radiation source accounts for the upgraded version suggested in this thesis.

The original code, which does not consider radiation effects, is based on a test campaign lead by G.A. Khoury (2006), at the Imperial College, London, investigating the strain behaviour, but also the heat and mass transfer of three concrete mixtures of three different advanced gas-cooled nuclear reactors (AGR).

The samples, in unsealed conditions, have been there exposed to 14-day two thermal cycles (thermal transient 1°C/min), including a period at constant temperature after both the first and the second heating, for test temperatures up to 600°C, with and without applied load.

The adopted temperature field is supposed to simulate different conditions that may be experienced by a concrete vessel:

d) normal operating conditions of heat cycling (start-up and shut-down of the reactor), and restart of the reactor (which is meant to be simulated by the second-time heating).

e) a hypothetical thermal transient, e.g. due to the loss of cooling water or a possible bad performance of the thermal insulation of the reactor.

The work is enlightening for several aspects:

a) it has allowed Khoury to define the different behaviour, in terms of strain, between virgin heating (i.e. heating at first thermal cycle) and the subsequent thermal transient: the first heating is characterized by the development of so-called “load induced thermal strain” (LITS). These newly identified component of strain are defined as the difference between thermal strain of the *unloaded* specimen and the strain measured *under load* for the same specimen, minus the initial elastic strain. Also shrinkage due to moisture loss or chemical dissociations is another kind of strain paying an important role during first heating. On the other hand, significant “delayed” LITS and shrinkage strain components characterize the early few hours at constant temperature following the first thermal transient; they slowly dissipate during the 5 days period at constant

temperature. The second thermal transient is characterized by the absence of LITS and by the development of expansive crack-inducing and chemical rehydration forces particularly during cooling without load.

b) Khoury has identified many strain components and has classified them into: load-free or load-induced strain; he has evaluated the superposition of effects for predicting the *total* strain, in terms of which he has given a mathematical formulation of concrete creep in non-isothermal conditions.

c) He has investigated for the first time the effect of constant temperature creep, i.e. the strain field after concrete is heated *under load*, retained more representative of reality than the well-known investigated basic creep, which occurs when specimens are heated *without* load to the required constant test temperature, and then loaded at that temperature (i.e. after any physico-chemical reaction has ended).

d) He has correlated phenomenologically the behaviour at high temperature of the overall concrete mixture with the behaviour of its constituent parts (cement paste and aggregates). In particular, the degree of damage is found to depend primarily upon the aggregate type (basalt aggregate being more stable than limestone) and, secondarily, upon the type of cement paste; the detrimental effects are found to be the most significant when heating/cooling without load to temperatures above 300-400°C.

e) He has determined concrete-specific “critical temperatures”, never addressed before, as the value of temperature up to which a second thermal cycle would not cause significant additional damage.

Temperatures, weight loss and strains have been measured for the three different concretes, during heating and cooling, with and without compressive load over a period of several days.

The experimental data have been then integrated in NEWCON3D, a fully coupled thermo-hydro-mechanical numerical model of heated concrete, where aggregates are distinguished from the cement paste, in order to study the behaviour of concrete as a composite material. As regards the mechanical field, NEWCON3D couples thermal, creep and shrinkage effects with chemo-thermo-mechanical damage and elasticity, in non isothermal conditions.

The majority of commercial software modeling the effect of temperature on concrete can be classified as thermo-mechanical; they consist of two separate analyses: thermal and

mechanical, that are interfaced in a way that the temperature field, output of the first, becomes the input for the second, at every time step, to produce the resulting strains and stresses in concrete due to both thermal and mechanical loads.

In NEWCON3D the thermal and mechanical analyses are coupled; in other words, if the structure develops cracks that influence the heat flow, then that is supposed to affect the thermal analysis. Therefore such a program is expected to predict more accurately than commercial codes the overall behaviour of concrete structures in non isothermal conditions, in terms of stress and deflections.

On the other hand one has to consider that there is no commercial software coupling the thermo-mechanical field with the hydraulic field, so that they are unable to predict phenomena related to moisture flow during heating.

And there are several scenarios in which a coupled thermo-hygro-mechanical study is to be preferred to a common thermo-mechanical analysis, e.g:

a) explosive spalling of concrete structures during a fire
b) radiation heat on biological shields of nuclear facilities.

Spalling occurs in concrete when it is heated at high temperatures within minutes, e.g. by a fire. In this case capillary water close to the surface is heated first, and as the surface temperature reaches 100°C, it starts to boil. Initially, the produced water vapour can easily escape from the surface. As the heating continues, water starts to boil further away from the surface and the vapour has to be transported a certain distance through the concrete. The resistance to the flow is directly related to the permeability of concrete. Therefore, a low permeability acts like a “closing valve” for water vapour, leading to high pressures inside the concrete, necessary to accommodate the vapour transport away from the boiling front.

Pressures close to the tensile strength of the material finally cause cracking and the subsequent decrease in its mechanical properties. Concrete subject to fire typically is shown to lose, time after time, the most external surface in a sort of “onion effect” that can result in a significant loss of the section, leading to a pronounced reduction in the load-bearing capacity of exposed concrete elements and final explosion.

The second scenario is the one of interest for this thesis: it is common practice in nuclear engineering to use concrete as shielding material for reactors, thanks to its versatility, cheapness but also its capacity to absorb radiation, due to its intrinsic water content, namely hydrogen.

The presence of this light-atomic weight element in the mixture of concrete is responsible for slowing down neutrons emitted by nuclear facilities, that is why, in the nuclear field, concrete is referred to as a “biological” barrier.

For gamma-rays radiation heavy elements have good shielding properties, on the contrary, so ordinary concrete mixtures are often enriched with heavy aggregates (magnetite, barytes, limonite) to ensure a good absorption of secondary radiation, too.

Therefore the knowledge of the moisture content evolution with the work history of a nuclear facility is fundamental to ensure the maintenance of its biological shielding.

This fact justifies why the phenomenon is here treated with a model where the hydraulic field is coupled to thermo-mechanics.

[1] Khoury G.A., Strain of heated concrete during two thermal cycles. Part 1: strain over two cycles, during first heating and at subsequent constant temperature, Magazine of Concrete Research 58(6), 367-385, 2006.

[2] Khoury G.A., Strain of heated concrete during two thermal cycles. Part 2: strain during first cooling and subsequent thermal cycle, Magazine of Concrete Research 58(6), 387-400, 2006.

[3] Khoury G.A., Strain of heated concrete during two thermal cycles. Part 3: isolation of strain components and strain model development, Magazine of Concrete Research 58(7), 421-435, 2006.

Outline

We have explained in the Introduction the importance to deal with thermo-hydro-mechanical modeling for nuclear shielding problems, to predict the behaviour of concrete at high temperature as well as the evolution of its moisture content.
In this thesis the employment of an upgraded version of NEWCON3D is justified by both these reasons, together with the necessity to guarantee the shielding properties of concrete as "neutron absorber" for the nuclear facility.

We will start in Chapter 1 with an outlook of the case study: SPES, an on-going project of a next generation facility for the production of special radioactive ion beams (RIB) for nuclear, medical and astrophysical research purposes.

An overview of the basic physical principles of radiation interaction with matter is given in Chapter 2, with a special interest on neutron radiation, which is the main form of radiation recognized for SPES.

The characterization of the physical properties of concrete, which justify its wide employment as shielding material, is made in Chapter 3. In the same chapter we introduce the thermal aspect of heat production conveyed by radiation absorption.

Chapter 4 contains the main aspects defining radiation damage on concrete, useful at the numerical definition of radiation damage, that will be focused on in Chapter 6.

An overview of Monte Carlo codes for transport calculations of radiation into media is provided in Chapter 5, also describing the main features of Fluka (theory and setup), a CERN-INFN patent applied in conjunction with NEWCON3D, in order to describe the main physical quantities characterizing the SPES issue.

Chapter 6 develops from a numerical definition of radiation damage, based on the tests collected in literature and provided in Chapter 4, and contains the description of NEWCON3D, with its main laws (balances and constitutive laws), starting from a thermodynamical framework defined in the field of irreversible thermodynamics.

Finally, in Chapter 7, the numerical analyses are reported; they are ideally divided into two sets: the first is a trial series of tests aimed at the calibration and validation of radiation damage, and the comparison of the behaviour of ordinary concrete with other special mixtures; the second set of analyses refers to the specific work scenario of SPES and includes

the modeling of the thermal aspect on concrete linked with radiation exposure, both treated as continuous and cyclic in time. The main findings are here discussed, with respect to the numerical results, and final considerations are made in the Conclusions.

Appendices A, B, C, D are recalled throughout Chapter 7 and contain the way in which the highlighted points are numerically implemented into NEWCON3D.

Chapter 1

1.1 A facility for the Selective Production of Exotic Species: SPES Project at Legnaro

The last few decades of research on heavy ion collisions have seen a great advance in radioactive ion beam (RIB) physics worldwide, thus giving new input to nuclear structure physics (Blumenfeld 2008; Moscatello 2008) far from the stability valley.

In order to explore unknown species classified in the nuclear chart, so-called exotic species, towards the limits of stability of nuclei, European nuclear physicists have built several large-scale facilities in various countries. Today they are collaborating in planning next-generation RIB facilities which will permit the investigation of hitherto unreachable regions of the nuclear chart (see in Fig.1.1 the dotted line).

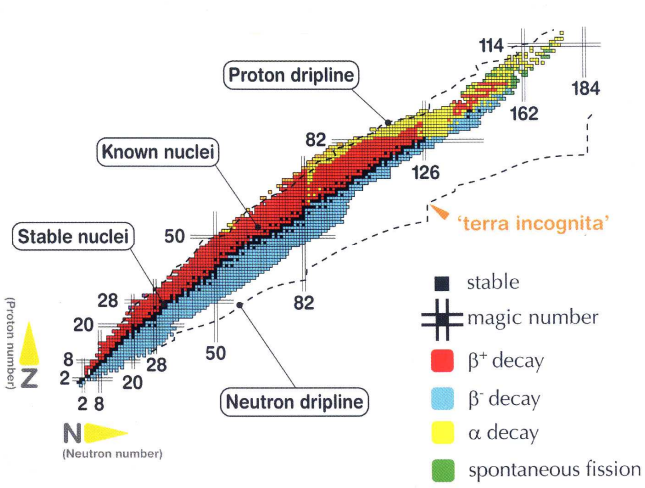

Fig.1.1 – The nuclear landscape: in colors are the known regions; within the dotted line the nuclear species under research.

In particular, the European roadmap calls for the construction of two next generation facilities for the production of RIBs by two different techniques: FAIR, making use of projectile fragmentation and EURISOL based on the Isotope Separation On-Line (ISOL) technique (http://www.nupecc.org/lrp2010/Documents/lrp2010_final_hires.pdf).

FAIR is already under construction at GSI, Darmstadt (Germany) (http://www.gsi.de/portrait/fair_e.html); with fragmentation the RIBs are produced by fragmentation of a projectile on a thin target. The radioactive nuclei created are separated in flight. The secondary beam of interest has high energy and high selectivity but low intensity and worse optical quality than the one produced by the ISOL technique.

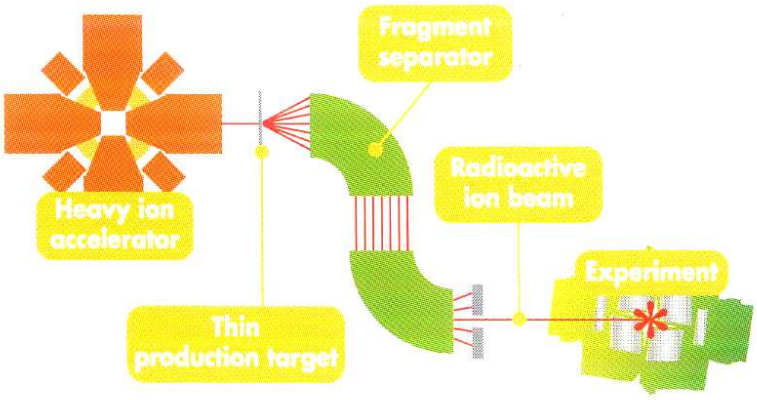

Fig.1.2 – Schematic view of the fragmentation technique for the production of RIBs.

As for the ISOL facility named EURISOL (http://www.eurisol.org/site02/index.php), there are too many technical challenges preventing an immediate construction, that is why the path towards EURISOL encompasses the construction of “intermediate” generation facilities, which will be themselves a testing ground for many technical solutions that will be carried over to final EURISOL : SPIRAL2 at GANIL, Cedex (France) (http://www.ganil-spiral2.eu/spiral2), HIE ISOLDE at CERN, Geneva (Switzerland) (http://hie-isolde.web.cern.ch/hie-isolde/), and SPES at the INFN National Laboratories of Legnaro (LNL), Padova (Italy) (http://www.lnl.infn.it/~spesweb/).

With the ISOL technique the radioactive nuclides are produced by spallation, fission or fragmentation reactions of a projectile with a thick target. The products of these reactions diffuse out of the target, undergo an ionization step, are separated on-line and eventually reaccelerated. The resulting secondary beams can be very intensive, depending both on the

initial driver/target characteristics and on the production target and post-accelerator efficiencies.

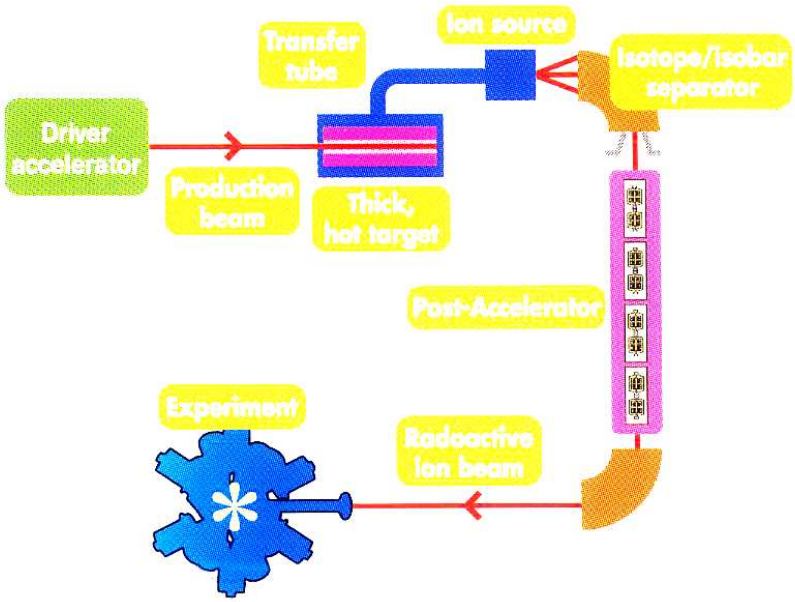

Fig.1.3 – Schematic view of the ISOL technique for the production of RIBs.

The main issues of the EURISOL research program comprise advances in:

a) nuclear structure:
   - A full understanding of the structure of halo nuclei. A halo structure occurs when one or more neutrons orbit around the core of the atom, leading to very large spatial extension atoms; this is the case of the most neutron-rich among the light-weight nuclei (e.g. ^{11}Li, ^{14}Be, ^{22}C), which the production of more intense and heavier beams allows to better detect.
   - The study of the behaviour of the shell structure of exotic nuclei, i.e. neutron-rich nuclei. Changes of nuclear structure has already been envisaged for high-neutron-number nuclei far from the stability region (in the drip-line), but a better understanding of more exotic nuclei is needed. In the medium and heavy mass regions the n-drip line is far away from the stability valley: the interest in studying nuclei with a large neutron excess in not only devoted to the localization of the drip line but also to the investigation of the density dependence of the effective interaction between the nucleons for exotic N/Z (isospin) ratios.

- The discovery and characterization of Super Heavy Nuclei (SHE): the complete definition of the Mendeleiev chart by the discovery of heavier elements and their isotopes, possibly leading to the identification of their physical and chemical properties is a very important challenge. The existence of SHE is based on nuclear structure effects: shell stabilization creates a barrier against spontaneous fission, which could otherwise terminate the periodic table just above Z=100.
- The complete systematic investigation of the radioactive decay of exotic species, possibly defining new forms of radioactivity:

b) nuclear astrophysics:
   - Neutron capture processes followed by β-decay are very important and are responsible for the production of most of the elements heavier than Iron. In particular the study of the r-process remains a major challenge both from the experimental point of view (measurements of reaction rates) and in the theoretical framework.
   - The better understanding of neutron stars, especially at the level of their crust, composed of neutron-rich nuclei immersed in a neutron gas, can be important, to hopefully explain the cooling process and the observed irregularities of these kind of stars, which are the most compact stellar objects after black holes, able to generate the most intense magnetic field of the Universe.
   - Another important item is the study of X-ray bursts generated on the surface of neutron stars as a consequence of their accretion process. Many open question and observables are there which indicate the needs of nuclear physics with stable and unstable beams.

c) fundamental interactions:
   - The search for new forms of interaction in nuclear decay modes, by precision measurements in nuclear beta-decay to test the Standard Model (SM) of electroweak interactions or by the improvement of physics at low energies, e.g.
   - A deep understanding of neutrinos, that can be produced by the radioactive beta-decay of massive amounts of unstable nuclei, e.g. ^{6}He and ^{18}Ne, accelerated at very high speed, close to the speed of light.

A 4-year study for EURISOL was initiated in 2005, which was devoted to the various technological aspects of the project, in particular the technological challenge of the driver

accelerator, the high power production targets, the physics program and the related instrumentation, the safety and radiation protection framework. The main item was to evaluate possible sites for the construction of the facility and to quantify the activation of the environment (soil and ground water) surrounding such a high-power experimental area, especially the target and linac buildings (Schlögl 2007).

The EURISOL Design Study was financially supported by the European Community under the Framework Program (FP) 6 "Research Infrastructure Action" (Documentation of the Final EURISOL Design Study Town Meeting 2009).

The EURISOL concept consists in a superconducting linear accelerator, capable of generating protons with an energy of 1GeV and an impressive power of 5MW, but also able of accelerating deuterons, ^{3}He and ions up to mass 40.

The proton beams will collide simultaneously with two types of targets, either directly or after conversion of the impinging protons into neutrons through a loop containing mercury as refrigerating material surrounded by fissile material.

The resulting unstable nuclei will be scattered out of the target, ionized and selected, and may then be directly used at a lower energy or reaccelerated by another linear accelerator up to energies of 150MeV per nucleon, thereby inducing nuclear reactions.

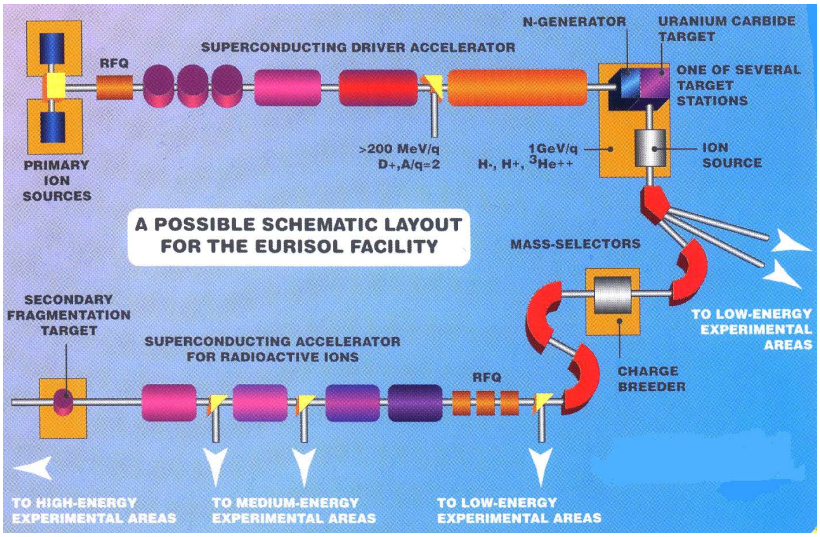

Fig.1.4 – EURISOL layout.

The SPES project, as part of a INFN Road Map for the Nuclear Physics development in Italy, is supported by the whole Italian Nuclear Physics community and mainly by LNL and LNS (National Laboratories of South) (SPES Executive Summary 2008).

Up to now, ISOLDE at CERN and HRIBF at Oak Ridge National Laboratory (ORNL) in Tennessee are the only facilities able to produce reaccelerated neutron rich beams. In Table 1.1 the main characteristics of the intermediate facilities under construction for the European project are listed: the specifics of the primary beam, the power delivered on the target, the kind of target (direct or conversion target, i.e. after conversion of protons into neutrons), the rate of fission reactions to produce the exotic species, the reaccelerator system, the ^{132}Sn rate, being it a measure of the production of the expected beam on experimental targets.

Table 1.1 – Characteristics of the three European intermediate facilities towards EURISOL (SPES Executive Summary 2008).

	Primary beam	*Power on target*	*Target*	*Fission rate [fissions/s]*	*Reaccelerator*	*^{132}Sn rate [pps]*
HIE ISOLDE (upgrade)	-	-	Direct and conversion	$4x10^{12}$	Superconductive linac	$2x10^{8}$
SPIRAL2	40MeV 5mA	200kW	Conversion	10^{14}	Cyclotron	$2x10^{9}$
SPES	40MeV 200mA	8kW	Direct	10^{13}	Superconductive linac	$3x10^{8}$

The main aspect of SPES is to provide high-intensity and high-quality radioactive beams of neutron-rich nuclei.

The conceptual design is based on a high-intensity 70MeV proton driver for the production beam, on a high power direct target (8KW) for the production target and on the availability of the heavy-ion post-accelerator ALPI, the presence of which at Legnaro INFN Laboratories justifies this site as the favorite for the construction of the Italian intermediate facility towards the European project EURISOL.

SPES will be mainly devoted to the production by means of the ISOL technique of intense neutron-rich RIBs of high quality, in the range of masses between 80 and 160 and, at the same time, to perform forefront research in interdisciplinary fields like medical, biological and material science.

A primary proton beam will be delivered by a cyclotron working at a variable energy (15-70MeV) able to supply at least a 40MeV and 0.2mA proton beam on a direct target made of a compound of uranium and carbon (UCx). SPES is planned to provide two target positions

located into two twin properly shielded dedicated areas, independent from each other, with the aim to operate them in alternate way, for safety reasons.

The exotic species (the secondary beam, or RIBs, of interest) will be generated as a consequence of the induced fission by the protons on Uranium at an expected rate of 10^{13} fissions/s.

The radionuclides will be extracted at 20kV from the target and ionized to the 1+charge state, where a first low resolution separation will be performed. The exotic products will then be sent either directly to laboratories for low-energy experiments, or to the secondary-beam accelerator. On the transport line towards the post-acceleration, a high resolution mass separator is foreseen with a nominal resolution of $\Delta m/M \approx 1/40000$. Moreover to optimize the re-acceleration a Charge Breeder will be developed in order to increase the charge state to N^+, necessary for the injection in ALPI and for the determination of the final energies of the produced accelerated beams.

To fit the ALPI linac with these very low energy, the secondary beam requires a new accelerating stage (pre-accelerator); the pre-accelerator consists in the existing PIAVE superconductive RFQ, or, as an alternative in a new conceived nor, mal conductive RFQ, which is still under study. Then the beam is ready for the injection into the ALPI linac and final delivery to the high-energy experimental areas.

An overall view of the facility is reported in Fig.1.5 where the blue zone are the environments to be built next to the already existing laboratories; the new spaces will host the cyclotron proton driver, the twin target rooms and new experimental areas.

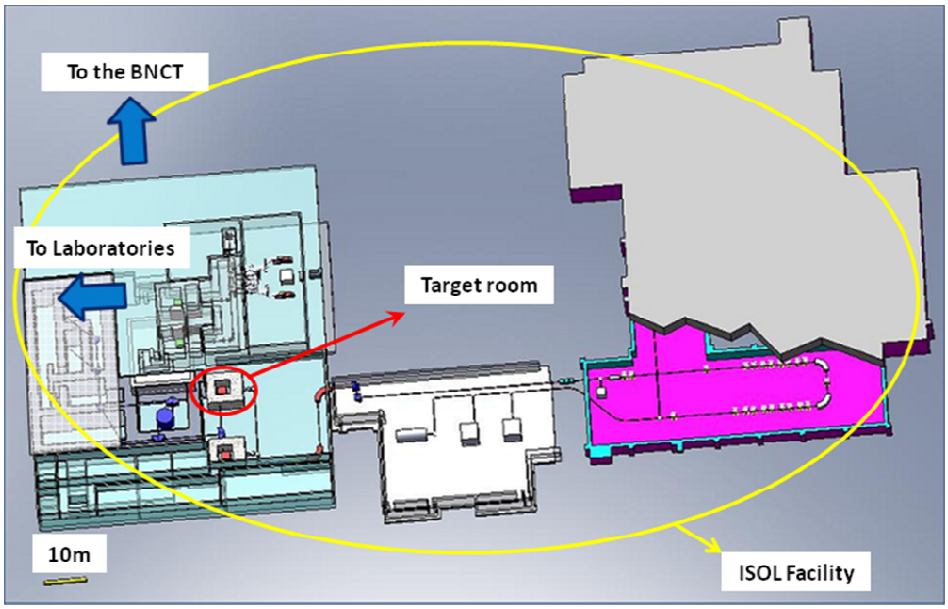

Fig.1.5 – Layout of the SPES facility at LNL.

The most critical issue is the design of the direct target, for which many studies have been already carried out (Radivojevic et al. 2002; Andrighetto et al. 2003, 2008, 2010; Alyakrinskiy et al. 2007; Barbui et al. 2008; Meneghetti et al. 2011). The target, up to day, consists in 7 UCx disks of nearly 1mm thickness and a few centimeters of diameter, for a total mass of about 30g. The disks are separated from each other in order to increase the total thermal exchange surface and optimize the cooling of the system by thermal radiation, taking advantage of the high operative temperature of the order of 2000°C (Meneghetti et al. 2011).

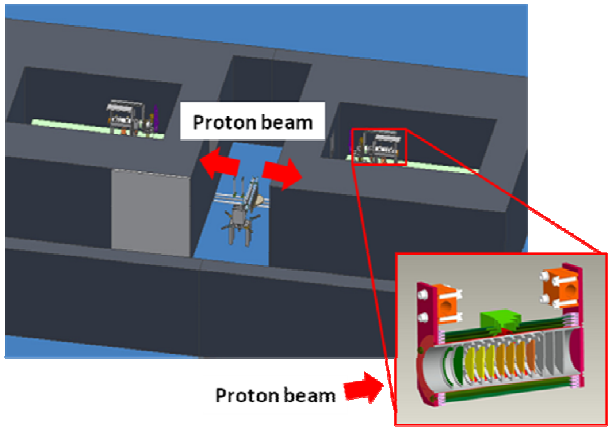

Fig.1.6 – The multi-foil direct target and its location inside the two twin target caves.

From the point of view of radioprotection the two target bunkers are again the most critical point. A preliminary study for the design of the shielding concrete thickness necessary to fulfill

radioprotection issues (annual ambient dose equivalent less than 0.5mSv/y, according to the Italian radiation protection law (Decree Law n.230, 17 March 1995)) are of the order of 3m in all directions, even 3.7m in direction of the impinging beam (SPES Executive Summary 2008), as reported in Fig.1.7, when an intense work scenario is envisaged.

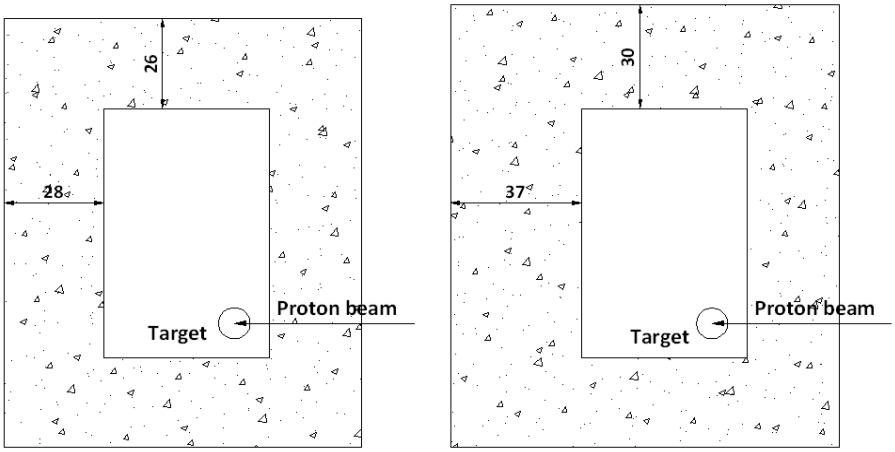

Fig.1.7 – Concrete shielding thickness for the most serious work scenario (70MeV, 350μA) after preliminary radioprotection studies [dm].

The shielding for the direct target is determined mainly by the neutron production rate and their energy spectrum. Moreover the target caves are planned to be located below ground level to assist in the prompt neutron radiation shielding.

To prevent air activation by scattered neutrons the target rooms are to be equipped with a dedicated nuclear ventilation system capable to process the air of the target room to avoid contamination by inlet in the surrounding spaces.

Also the activity induced in the target, after the decay of the fissile material, requires handling system to dismount it with remote operations. It has been evaluated by LNL Laboratories that after 14 days of cooling down of the target, it can be removed in safety with an automatic system or, with a minimum shield of 2cm of lead and at an operation distance of 2m, potentially with hands.

Due to the possibility given by the chosen driver, that is a cyclotron which can provide simultaneously two proton beams at two exit ports, which are independent in terms of intensities and energies, a study called LARAMED (LAboratorio di RAdionuclidi per la MEDicina) is foreseen for the research and development of new radionuclides, which are of interest for the medical diagnostics and therapy. Within this framework the study of

production cross section of important radionuclides like *Cu-64*, *Cu-67*, *Sr-82/Rb-82 e Ge-68/Ga-68* and the development of power targets which must be able to support high intensities provided by modern accelerators are foreseen.

As for research in radiobiology and nuclear medicine, the expected usefulness of SPES is the better understanding of the biological effects induced by ionizing radiation in cultured (normal and tumor) mammalian cells, also in relation to the medical application of radioactive ions in hadron-therapy, that is always requiring improvements of the quality of the selected radioisotope doses to treat the patient with.

References to Chapter 1

[1] Alyakrinskiy O., Antoshin A., Avilov M., Bolkhovityanov D., Fadeev S., Golikov V., Gubin K., Lebedev N., Logatchev P., Popov V., Shiyankov S., Tecchio L.B., Thermal experimental results on the prototype for high-power neutron converter for low-energy proton/deuteron beams, Nuclear Instruments and Methods in Physics Research A 578, 357-369, 2007.

[2] Andrighetto A., Biasetto L., Manzolaro M., Barbui M., Bisoffi G., Carturan S., Cinausero M., Gramegna F., Prete G., Rizzi V., Antonucci C., Cevolani S., Petrovich C., Colombo P., Meneghetti G., Di Bernardo P., Zanonato P., Cristofolini I., Fontanari V., Monelli B., Oboe R., The SPES multi-foil direct target, Nuclear Instruments and Methods in Physics Research B 266, 4257-4260, 2008.

[3] Andrighetto A., Biasetto L., Manzolaro M., Scarpa D., Montano J., Stanescu J., Benetti P., Cristofolini I., Carturan M.S., Colombo P., di Bernardo P., Guerzoni M., Meneghetti G., Monelli B., Prete G., Puglierin G., Tomaselli A., Zanonato P., Production of high-intensity RIB at SPES, Nuclear Physics A 834, 754c-757c, 2010.

[4] Andrighetto A., Li J., Petrovich C., You Q., Study of isotopes production for the SPES project, Nuclear Instruments and Methods in Physics Research B 204, 205-210, 2003.

[5] Barbui M., Andrighetto A., Antonucci C., Biasetto L., Carturan S., Cervellera F., Cevolani S., Cinausero M., Colombo P., Dainelli A., Di Bernardo P., Giacchini M., Gramegna F., Lollo M., Maggioni G., Manzolaro M., Meneghetti G., Petrovich C., Piga L., Prete G., Re M., Rizzi V., Stracener D.W., Tonezzer M., Zafiropoulos D., Zanonato P., Calculations and first results obtained with a SiC prototype of the SPES direct target, Nuclear Instruments and Methods in Physics Research B 266, 4289-4293, 2008.

[6] Blumenfeld Y., Radioactive ion beam facilities in Europe, Nuclear Instruments and Methods in Physics Research B 266, 4074-4079, 2008.

[7] Esposito J., Ceballos C., Concin M., Fabris C., Friso E., Moro D., Colautti P., Jori G., Rosi G., Nava E., Feasibility of BNCT radiobiological experiments at the HYTHOR facility, Nuclear Instruments and Methods in Physics Research B 266, 2587-2593, 2008.

[8] EURISOL, Documentation of the Final EURISOL Design Study Town Meeting, Pisa (Italy), 30 March – 1 April 2009.

[9] Meneghetti G., Manzolaro M., Andrighetto A., Thermal-electric numerical simulation of a target for the production of radioactive ion beams, Finite Elements in Analysis and Design 47, 559-570, 2011.

[10] Moscatello M.H., Safety and radioprotection issues with post-accelerated high-intensity ribs, Nuclear Instruments and Methods in Physics Research B 266, 4682-4686, 2008.

[11] National Standard "Decree Law n.230", 17 March 1995: Implementation of guidelines: 89/618/Euratom, 90/641/Euratom, 92/3/Euratom and 96/29/Euratom on ionizing radiations.

[12] Radivojevic Z., Andrighetto A., Brandolini F., Dendooven P., Lyapin V., Stroe L., Tecchio L., Trzaska W.H., Vakhtin D., Neutron yields from a thick ^{13}C target irradiated by 30MeV protons, Nuclear Instruments and Methods in Physics Research B 194, 251-256, 2002.

[13] Schlögl B., Master Thesis, EURISOL DS/Task5/TN-07-01: Calculation of the activity inventory of transportable radionuclides in soil and groundwater for large neutron sources – Contribution to Eurisol Task 5 (safety), WP2.4: activity transport with groundwater, 2007.

[14] INFN-LNL-224, SPES Selective Production of Exotic Species: Executive Summary. A. Covello, G. Prete Editors, Laboratori Nazionali di Legnaro, 2008.

[15] http://www.eurisol.org/site02/index.php

[16] http://www.ganil-spiral2.eu/spiral2

[17] http://www.gsi.de/portrait/fair_e.html

[18] http://hie-isolde.web.cern.ch/hie-isolde/

[19] http://www.lnl.infn.it/~spesweb/

[20] http://www.nupecc.org/lrp2010/Documents/lrp2010_final_hires.pdf

Chapter 2

2.1 Interaction of radiation with matter

2.1.1 Some useful definitions

"Particle fluence" Φ is defined as the number of particles (photons or neutrons) entering a specified volume:

$$\Phi = \frac{\Delta N}{\Delta A} \tag{1}$$

where ΔN is the number of particles entering a cross-sectional area ΔA. It has units [photons or neutrons/cm^2].

"Particle flux" or "particle flux density" ϕ is defined as:

$$\phi = \frac{\Delta \Phi}{\Delta t} \tag{2}$$

where ΔΦ is the increment of particle fluence in time Δt. It is expresses as [photons or neutrons/(cm^2 s)]. Notice that, when the increment ΔΦ is constant in time, Φ is obtained by multiplying ϕ for the time span in which ΔΦ lasts.

The same quantities are defined, similarly, in terms of energy: "energy fluence" of particles F is defined as:

$$F = \frac{\Delta E_F}{\Delta A} \tag{3}$$

where ΔE_F is the sum of the energies of all particles entering a cross-sectional area ΔA. It has units [MeV/cm^2].

The "energy flux density" or "current" I is:

$$I = \frac{\Delta F}{\Delta t} \tag{4}$$

and is expressed, usually, in [MeV/(cm^2 s)].

2.1.2 Interaction processes with target materials

The interaction of radiation with matter is stochastic in nature. The mode of interaction depends on the type of radiation, its energy and the material through which it travels, as it will better explained later on.

When one form of radiation interacts with the atoms of the absorber medium, an emission of energy occurs; then the particles continue propagating through space or throughout the material, releasing other energy until rest or final capture of the initial radiation.

In a travel within the target medium, radiation may undergo:

a) absorption

b) scattering

c) transmission

The first two phenomena allow radiation to be removed from the incident beam, therefore they both are referred to as “attenuation processes”. Absorption is the taking up of the energy from the impinging beam by the irradiated material. Scattering refers to a change in the direction of the colliding radiation with subsequent possible loss of the initial energy. The amount of radiation which is not subject to these processes is transmitted.

We can distinguish between two types of radiation:

a) electromagnetic radiation

b) particulate radiation.

The first includes x-rays and gamma-rays. The former are produced when high-speed electrons collide with outer electrons, the latter are generated by *intranuclear* disintegration and bring, generally, more energy than x-rays, typically of the order of 0.1MeV. It is common practice to define gamma-ray any electromagnetic radiation which possesses an energy equal or greater than this value, which characterizes radiation originating from a nucleus of an atom. At these energy electromagnetic radiation behaves like a particle, more than a wave, and the particles or quanta of electromagnetic radiation are called “photons”.

Particulate radiation refers to the energy propagated by traveling corpuscles, with defined rest mass, momentum and position at any instant. It includes atomic particles (electrons, protons and neutrons) and subatomic particles (positrons, neutrinos, mesons).

Radiation from uncharged particles (photons and neutrons) are of most concern with regards to nuclear facilities, because, differently from charged particles, interacting strongly with the medium and readily absorbed (so-called “directly” ionizing radiation), they are more

penetrating and produce charged particles as by-products of first collisions (for this reason they are classified as “indirectly” ionizing radiation).

The removal mechanisms for photons and neutrons will be illustrated in the following.

2.1.2.1 Interaction of photons

Photons are known to interact with the *electrons* in the atoms of the absorber medium. With respect to radiation shielding three kinds of interactions are considered to be the most important: photoelectric effect, pair production and Compton scattering.

In the photoelectric effect, the incident photon impinges on an orbital electron of the shielding material, and transfers all its energy to the electron by ejecting it from the atom. It is the inner orbital electrons which possess the closest binding energy to the incident gamma-ray, so the interaction occurs at the level of the inner, strongly bounded electrons; all of the energy of the impinging photon in excess of the binding energy turns into the kinetic energy of the ejected photoelectron.

The photoelectric effect is meant to be a pure absorption process, since it results in the complete absorption of photons, except for the negligible emission of relatively soft secondary radiation.

The importance of this process increases as the atomic number of the target medium increases and as the energy of the photon decreases, so that it is most likely for shields made of heavy elements and at low gamma-rays energies (below 0.5MeV). In fact it is known to be inversely proportional to the photon energy E_γ with E_γ^3 and approximately proportional to Z^4, Z being the atomic number of the absorber medium.

Pair production is a phenomenon occurring at gamma-energies greater than 1.02MeV (twice the rest mass of an electron) and consists in an interaction in which the photon completely disappears generating a pair of electrons, one positive (positron) and one negative (negatron), in the region of an atomic nucleus.

All the photon energy is absorbed and goes into the production of the pair and their kinetic energy, apart from low-energy emissions due to annihlation of the positrons from the pair production, which is considered negligible, thus classifying this phenomenon, too, as a true absorption process in radiation shielding calculations.

The importance of this form of interaction increases as the atomic number of the absorber (proportionally to Z^2) and the photon energy increase (the increase with E_γ is slow until 4MeV,

than it becomes proportional to $\log(E_\gamma)$), becoming the dominant interaction process of photons with matter at high energies.

Compton scattering is also known as modified, incoherent or inelastic scattering and it was first elucidated by Arthur H. Compton in 1923; after this study the Compton effect was accepted as the final proof for the dual nature of light and established the legitimacy of the quantum theory.

In this kind of interaction a photon collides with a free electron and hands over part of its energy to it. As a result, the incident photon is deflected from its original path so that the Compton effect results in both attenuation and absorption. Successive collisions due to several Compton scatterings can slow down the photon to an energy comparable for absorption by the photoelectric effect. However, the ineffectiveness in the loss of energy, which is a function of the scattering angle, makes this process not a guarantee for the sought absorption of photons.

The probability for Compton scattering to occur is proportional to Z and, over a considerable range of photon energies, it is higher than the other two mechanisms, for all except the heaviest elements.

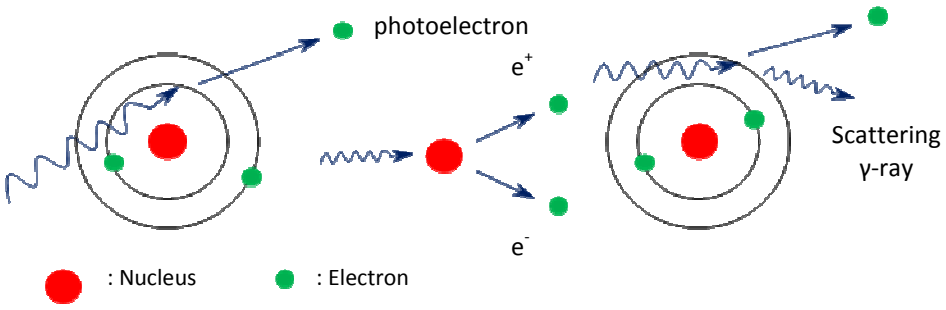

Fig.2.1 – Schematic representation of principal interaction processes for gamma-rays in radiation shielding: photoelectric effect, pair production, Compton scattering.

2.1.2.1.1 Attenuation of gamma radiation in matter

Under "good geometry" conditions, i.e. analyzing a narrow, perfectly collimated mono-energetic beam, the decrease of the initial intensity of the photon flux ϕ_0 of the incident beam, in crossing a shield slab of given composition, is known to be exponential through the thickness x:

$$\phi(x) = \phi_0(x)\, e^{-\mu x} \tag{5}$$

where $\phi(x)$ is the gamma-ray flux at point x in the slab, ϕ_0 the incident flux, μ is the linear gamma-ray attenuation coefficient.

This one-dimensional simplified model is also known as the Lambert's law for the attenuation of photons in matter (Jaeger 1970).

The linear attenuation coefficient μ is a measure of a photon interacting in a particular way with a given material, per unit path-length. It has units [cm^{-1}] and depends on the energy of radiation, the type of interaction occurring, the density and composition of the absorber. In fact μ is known to depend upon the number of atoms per unit volume N of the shielding medium:

$$\mu = \sigma N \tag{6}$$

where $N = N_0\, \rho/A$, with ρ density, A atomic weight of the material and N_0 the Avogadro's number ($N_0 = 0.6025 \times 10^{24}$ atoms/mole); σ is the proportionality constant called "microscopic cross-section"; it is conceived as an area and represents the probability of an interaction of a photon with one target nucleus or electron, per unit volume of the material.

The microscopic cross-section is assumed to be the sum of the cross-sections of the three described processes of attenuation, other interaction processes having negligible effect (Harrison 1958; Kaplan 1989):

$$\sigma = \sigma_{Compt} + \sigma_{ph} + \sigma_{pp} \tag{7}$$

where the three quantities σ_{Compt}, σ_{ph} and σ_{pp} are the microscopic interaction cross-section for Compton scattering, photoelectric absorption and pair production, respectively, all per atom.

Their relative importance varies with photon energy and the atomic number of the absorber. The total cross-section σ for a light element (Al) and a heavy element (Pb) are shown in Fig.2.2, over the energy range 0.1-10MeV, usual for nuclear reactors. For aluminum the Compton process is dominant in a wider range of energy. For both the photoelectric effect predominates at low energies and pair production at high energies. For light elements a minimum in the total behaviour of σ would be envisaged but at higher energies (Harrison 1958).

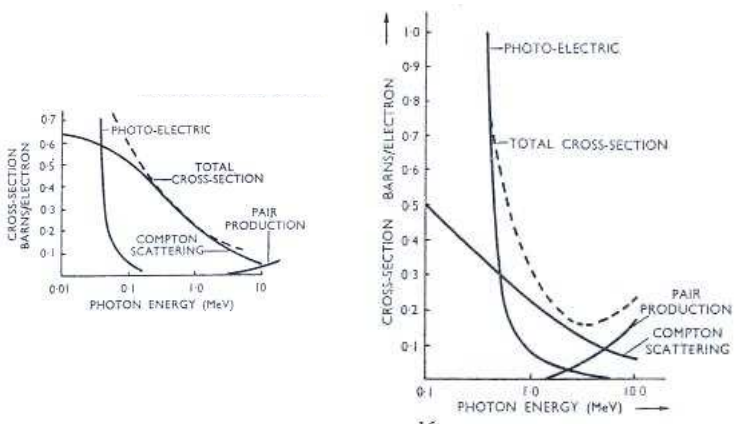

Fig.2.2 – Relative importance of the interaction processes for photons with aluminum (light element) and lead (heavy element) in the range 0.1 to 10MeV of photon energy (Harrison 1958).

To obviate the dependence of the linear attenuation coefficient on the density, it is common practice to deal with "mass attenuation coefficients", i.e. μ/ρ, a measure of the shielding effectiveness per unit mass of the material [cm^2/g].

Attenuation coefficients (both linear and mass specific) concern the number of photons which are affected by photon interactions with matter, but related *energy-absorption* coefficients, concerning with the loss or deposition of energy, are significant in connection with the determination of the effects of radiation.

The "mass energy absorption coefficient" is a measure of the escape of all secondary photons produced in a small volume element. Different definitions seem to hold for this term, with respect to the forms of escape included: the "mass energy absorption coefficient" μ_{en} comprises the photons escapes by:

a) fluorescence,
b) annihilation,
c) bremsstrahlung,
d) Compton scatterig.

Fluorescence occurs when the absorbed electromagnetic radiation is intense, so that it is possible for one electron to absorb two photons; this two-photon absorption can lead to emission of radiation having a shorter wavelength than the absorbed radiation.

Positron annhilation is an interaction with a negative electron resulting in the production of two gamma-rays each of 0.5MeV.

Bremsstrahlung is the production of gamma-radiation in the process of rapid deceleration of high-energy beta-particles.

They all have small effects and can be generally neglected for reactor shielding.

The "mass energy transfer coefficient" μ_k limits the escape of secondary photons to fluorescence, annihilation radiation and Compton-scattered photons; the "mass absorption coefficient" μ_a is more restrictive and considers only Compton-scattered photons, i.e.:

$$\mu_a = \mu - \mu_{Compt} . \tag{8}$$

In general the values of μ_a are the largest, with μ_k values being slight smaller by a few percent, and μ_{en} values the smallest. Moreover, the spread between the values of these three coefficients is just a few percent for light elements but may reach a factor of 3 for heavy elements at very high energies, in fact, at high energies (beyond 1MeV) and higher atomic weights, the effect of annihilation radiation resulting from pair production, bremsstrahlung or fluorescence becomes significant; it is only at lower energies, for light elements, that the photoelectric effect is dominant and the numerical values of the cross-sections for the different processes are the same for all three quantities.

As already stated at the very beginning, the Lambert's law predicts the intensity of the undeflected beam or the direct component of the flux at point x, but neglects energy changes and scattering of the radiation in the medium, treating all scattering phenomena as absorptions, while, more realistically, some of the radiation scattered out of the beam is scattered back to the detection point x and decays in energy, so that the total flux in x is the sum of direct and scattered fluxes.

These components can be calculated, for simple geometries, by solving the differential expression for the transport of photons in the material, so-called Boltzmann transport equation, which needs integration over space, angle and energy of the incident photon, as it will be illustrated later on, or by stochastic Monte Carlo calculations, for more complex geometries.

However, to help in shielding design, it is a common practice to deal with a modified Lambert's law, by the introduction of the buildup factor B(μx), which is expected to correct the one-dimensional model (see Eq. (5)), under broad beam conditions, according to:

$$\phi(x) = \phi_0(x)\, B(\mu x)\, e^{-\mu x} \tag{9}$$

where the buildup factor accounts for the computation of the scattered component of the flux.

Buidup factors depend on many parameters: the energy of the photon source, the angle of incidence of the photons, the composition and thickness of the shield, the geometry of the problem or position of the source with respect to the target point.

The buildup factor is defined as the ratio between the total photon flux and the direct, undeflected photon flux at the same detection point:

$$B(\mu x) = \frac{\phi_\gamma(x)_{TOT}}{\phi_\gamma(x)_{dir}} = 1 + \frac{\phi_\gamma(x)_{scat}}{\phi_\gamma(x)_{dir}} \qquad \phi_\gamma(x)_{TOT} = \phi_\gamma(x)_{dir} + \phi_\gamma(x)_{scat} \tag{10}$$

Buildup factors are calculated after calculation of $\phi_\gamma(x)_{TOT}$ by solving the Boltzmann transport equation and of $\phi_\gamma(x)_{dir}$ by the Lambert's law. The results for a given energy are then plotted and fitted to simple mathematical expressions. The following analytical approximations, due to different authors, are found in literature, useful at defining buildup factors for different materials and different photon energies:

a) exponential fit, due to Taylor:

$$B(\mu x) = Ae^{-\alpha_1 \mu x} + (1 - A)e^{-\alpha_2 \mu x} \tag{11}$$

b) polynomial fit, due to Hungerford:

$$B(\mu t) = A + B(\mu t) + C(\mu t)^2 + \ldots \tag{12}$$

c) Berger's formulation:

$$B(\mu x) = 1 + A\ \mu x\ e^{B\ u\ x} \tag{13}$$

d) Capo's formulation:

$$B(\mu x) = \sum_{i=1}^{3} \sum_{j=1}^{4} C_{ij} \left(\frac{1}{E}\right)^j (\mu x)^i \tag{14}$$

2.1.2.2 Interaction of neutrons

Neutrons interact mainly with the *nuclei* of the atoms of the absorber medium, depending on their kinetic energy, which allows to classify neutrons, at increasing energy as:

a) thermal
b) epithermal
c) slow
d) resonance
e) intermediate energy
f) fast

g) very fast.

The slow component is known to keep at energies below 1 or 10eV, though sometimes the declared threshold energy is 1keV (Kaplan 1989); fast neutrons possess energy greater then about 0.5MeV until 10MeV. Very fast neutrons reach energies greater than about 20MeV.

The main neutron interactions with matter are: elastic scattering, inelastic scattering and neutron capture.

The elastic scattering happens when, after a collision with a nucleus, both the neutron and the residual nucleus are left unexcited. The kinetic energy is totally transferred to the scattering nucleus. The phenomenon is important for light nuclei which can absorb a large fraction of the energy of the incident neutron, just in one collision.

Inelastic scattering occurs when the target nucleus is left in an excited state, so that the excitation energy is emitted in the form of one or more gamma-rays. The importance of the process goes with the mass number of the target nucleus: it is higher for materials made of heavy elements.

Neutron capture happens when a neutron is captured by a nucleus and the binding energy, plus the kinetic energy of the absorbed neutron, are released as gamma-radiation by radioactive decay of the new-formed unstable nucleus (so-called “radiative capture” phenomenon). The total energy emitted depends on the capturing nucleus, and it is known to vary smoothly over the periodic table, except for low-atomic-weight nuclei and the so-called “magic number nuclei” (i.e. those containing either 16, 50, 82 or 126 neutrons or protons. The released energy is about 8MeV for medium-weight nuclei and drops to about 6MeV for heavy nuclei).

It is the slow component of the neutron spectrum that most likely undergoes neutron capture, i.e., basically, after the fast component has been slowed down to thermal energies by elastic or inelastic scattering, the thermalized neutrons can be easily absorbed by neutron capture, with subsequent production of secondary radiation.

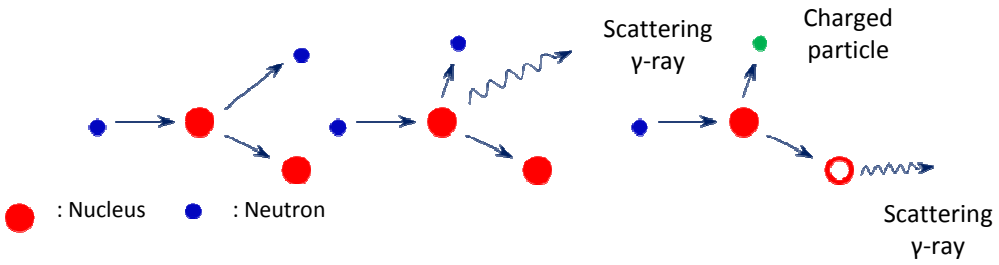

Fig.2.3 – Schematic representation of principal interaction processes for neutrons in radiation shielding: elastic scattering, inelastic scattering, neutron capture.

2.1.2.2.1 Attenuation of neutrons in matter

The theoretical methods defining the statistical process of interaction of neutrons with matter are conveniently described by the general “transport theory”, with its main result being the Boltzmann transport equation, which applies for both photons and neutrons. From a statistical point of view the problem, both for photons and neutrons, can be faced by Monte Carlo technique, which becomes useful especially when complicated geometry are to be analyzed.

The attenuation process for neutrons is more complex than for photons, because of a significant dependence of the neutron attenuation coefficients on the neutron energy and on the atomic weight of the target material (Price et al. 1957).

However, as for photons, some simpler methods have been developed to define the slowing down of fast neutrons through matter and subsequent capture of the thermal component of the neutron spectrum.

In the following, two approaches of the more rigorous transport theory are illustrated; they apply, respectively, for neutrons which are already at thermal energy (though thermalization from the fast group occurs chronologically after), i.e. the so-called “diffusion theory”, and for fast neutrons, i.e. the so-called “two-group theory” (Price et al. 1957; Shultis and Faw 1996).

The two simplified theories develop from the above mentioned Boltzmann equation, which is obtained by considering the inflow and outflow balance of particles through the surface of an arbitrary closed volume in a steady state condition for the radiation field, as it will better explained in Chapter 5.

These approaches are simplified in the sense that they renounce to solve the problems associated to primary neutrons attenuation, such as the production of photons during inelastic

scattering, the capture of thermal neutrons leading to capture gamma photons or the production of secondary neutrons as a result of fission reactions; nevertheless they offer, in their simplified way, acceptable estimates of the expected flux density field (Harrison 1958; Kaplan 1989; Shultis and Faw 1996).

The basic hypothesis of the diffusion theory is that the neutron current **I** is isotropic and is given by:

$$\mathbf{I} = -D\nabla\phi \tag{15}$$

thus leading to the following conservation equation:

$$-D\nabla^2\phi + \Sigma_a\phi - S = 0 \tag{16}$$

where ϕ is the flux density (for instance in [n/(cm^2 s)], S is the source term, Σ_a is the macroscopic absorption cross-section for thermal neutrons and D the thermal diffusion coefficient, which is defined in a simple diffusion theory as:

$$D = \frac{\lambda_t}{3} \tag{17}$$

λ_t being the transport mean free path: the average distance which a neutron travels in a medium before an interaction with the atoms of the medium is followed by another one.
The one-dimensional solution of a plane source in an infinite medium can be derived from this approximated theory by imposing the proper boundary conditions:

$$\phi(x) = \frac{SL}{2D}e^{-\frac{x}{L}}, \qquad L = \sqrt{\frac{D}{\Sigma_a}} \tag{18}$$

where L is known as the diffusion length: the length in which the intensity or density of neutrons is reduced by a factor 1/e=0.368.
This theory is valid deep in the medium, where the flux ϕ is often given by multiplied scattered particles and hence it can be expected to be nearly equal in all directions, whereas near a free surface or a source the flux density is quite anisotropic and the diffusion approximation may be poor. Moreover the same hypothesis makes the theory suitable for thermal neutrons rather than fast ones (Harrison 1958).

The second approach allows for getting a simplified expression for the fast group equation in a semi-infinite plane geometry.
According to the two-group theory, neutrons in the shield are divided into neutrons at thermal energies and neutrons above this energy. The diffusion equation (16) is applied to each group, the source term in each group being the neutrons slowed down in the immediately higher energy group. This leads to two equations, one for each energetic group:

$$\begin{aligned} -D_f \nabla^2 \phi_f + \Sigma_f \phi_f &= 0 \\ -D_t \nabla^2 \phi_t + \Sigma_t \phi_t - \Sigma_f \phi_f &= 0 \end{aligned} \tag{19}$$

for the fast and the thermal group, respectively. Notice that the fast source term in the shield is zero, whereas the thermal source term equals the fast neutron component scattered into thermal energies ($\Sigma_f\phi_f$) and, in a similar way to the diffusion equation (see Eq. (18)), the solution of the fast group equation in a semi-infinite plane geometry is found:

$$\phi_f(x) = \phi_0(x)e^{-\Sigma_R x}, \qquad \Sigma_R = \frac{1}{L_s} \tag{20}$$

where Σ_R is the macroscopic removal cross-section, which is roughly equal to the reciprocal of the average relaxation length for fast neutrons in the shielding material L_S, and ϕ_0 is the fast neutron flux at x=0.

The theory in this case is valid until the attenuation of fast neutrons is dominated by a removal process, i.e. if concrete contains sufficient moderating material (mainly, hydrogen of its water content) (Kaplan 1989), as it will better explained in Chapter 3, relative to the use of concrete for shielding.

One can see that for both the thermal and the fast component an exponential decay for neutron attenuation occurs, which is similar to what shown for photon radiation; in the case of neutrons the material-specific quantity is the macroscopic cross-section (Σ_a or absorption cross-section for thermal neutrons, Σ_R or removal cross-section for fast neutrons), which is also strictly dependent on the energy of the incident neutrons.

Similarly to what stated for the linear attenuation coefficient for photons, the macroscopic cross-section is a measure of the probability that a neutron of a given energy interacts in some way with the shielding medium.

The macroscopic cross-section Σ equals Nσ and has units [cm^{-1}], where N is the number of atoms/cm^3, equal to N_0 ρ/A (ρ density, A atomic weight of the material, N_0 the Avogadro's number); σ is the microscopic cross-section for neutrons, which has units [cm^2] or [barns] (1barn = $10^{-24}cm^2$]. It defines the number of absorptions/(cm^3 s) as the product Nnvσ, where n is the volume density of neutrons [neutrons/cm^3] and v their velocity [cm/s], so that nv is the neutron flux ϕ [neutrons/(cm^2 s)].

As in the case of photons, the total neutron cross-section σ is the sum of the cross-sections for all the neutron-interaction processes.

References to Chapter 2

[1] Harrison J.R, Nuclear reactor shielding, Temple Press, London, 1958.

[2] Jaeger R.G., Engineering compendium on radiation shielding, vol. 2, Springer, Berlin, 1970.

[3] Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[4] Price B.T., Horton C.C., Spinney K.T., Radiation shielding, Pergamon Press, London, 1957.

[5] Shultis J.K., Faw R.E., Radiation shielding, Prentice Hall PTR, 1996.

Chapter 3

3.1 Reasons for the civil application of concrete in nuclear facilities

The durability assessment of existing reactor vessels, the growing interest in nuclear research facilities like SPES, the use of high-current, low- and medium- energy proton accelerators for medical and industrial use require ad hoc radioprotection studies and the design of adequate shields is an important issue for guaranteeing safety service lives of these structures.

For all these structures, only neutrons and photons are sufficiently penetrating and biologically damaging to constitute a shielding problem: in nuclear reactors and, similarly, in the core of nuclear research facilities, such as the target room in SPES, neutrons are the primary products of fission reactions and photons are the by-products due to neutron capture; in proton accelerators neutron are produced by the interaction of the proton beam with the structures of the accelerator, of the beam transfer lines and, in the case of medical facilities, of the beam delivery system used to irradiate the patient (such as collimators and field-shaping devices) and with the patient himself, who the beam is ultimately delivered to; also in this case gamma-rays are produced as a result of the binding energy released when neutrons are absorbed by the nuclei of the target materials, during the acceleration process of protons.

Looking at the attenuation process for high-energy neutrons colliding with the nuclei of a shield, they can be either scattered or absorbed in the shielding material, the first process being much more likely. The scattering event may be elastic or inelastic, as already explained in the previous Chapter, dealing with radiation interaction with matter. From the shielding point of view, inelastic scattering occurs in heavier nuclei, capable of a significant decrease of

energy of the incident fast neutrons. Elastic scattering is, on the other hand, significant for light nuclei, able to provide a large energy loss in a single collision with the colliding fast neutrons.
In a concrete shield it is hydrogen which plays an important role in the slowing down process of fast neutrons, due to elastic scattering; the availability of hydrogen as the intrinsic water content of concrete makes it the most popular material for nuclear shielding.

Hydrogen occurs in concrete mainly in the form of fixed or bound water, present in its constituents (aggregates, cement paste) or some organic impurities, and in the form of the capillary water partially filling the pores of concrete.
Ordinary concretes used for neutron shields should have a minimum water content of at least 5 to 7% by weight; special shielding concretes may contain as much as 15% by weight water (Jaeger 1970). Part of this amount will be held as water of hydration, or fixed, bound water, above 100°C; the rest will evaporate at these temperatures.
It is, therefore, desirable to increase the amount of fixed water in concrete, by the employment of special mixtures, to exploit the moderating properties of hydrogen, which can slow down neutrons and make them easily available for their final capture.

Low-energy or thermal neutrons are readily absorbed and are, for this reason, of less, though not negligible, concern in the shield design. In fact the final absorption of thermal neutrons, by the nuclei of the shielding material is known to manifest through the emission of gamma-rays or photons, as products of the binding energy release after the absorption process. The energy release is associated with heat generation in the shield, an important collateral effect of radiation exposure. Both the form and the magnitude of the energy release depend on the absorber nucleus, in particular it is higher for heavier nuclei, so that also the development of radiation heat depends strongly in the composition of the shielding.

The attenuation process of gamma radiation increases as the density of the material increases, at least in nuclear reactors, due to the fact that, at the characteristic energies of photons for such structures, the Compton effect is the primary removal process.
Thanks to the nature of the Compton scattering mechanism, i.e. its effectiveness is directly proportional to the electron density of the collided atoms, which goes in most cases with the density of the absorber material, it is common belief that higher the density of the shield, the higher the removal of gamma-rays.

Therefore, the density of concrete, too, plays an important role in its nuclear performance, because density is a measure of the atomic concentration of the constituent elements of a

medium and the greater the density, the greater the number of nuclear interactions which lead to the absorption of radiation.

The specific weight of concrete is of much concern in the effectiveness of stopping both gamma rays and neutrons and it can be improved by using heavier aggregates in the composite mixture, such as iron ore-aggregates.

3.2 Nuclear properties of concrete

Provided we consider dominating the removal processes of elastic scattering, for neutrons, and Compton scattering, for photons, preliminary estimates of the attenuation of these forms of radiation in the target materials can be held by means of the simplified exponential expressions reported in Chapter 2 (Eq. (9) for photons and Eqs. (18) and (20) for neutrons) (Harrison 1958; Jaeger 1970; Kaplan 1989; Price et al. 1957), which strictly depend on the properties of the shield, through the relaxation length, that is to say through the linear attenuation coefficient, for photons, or the macroscopic removal cross-section and the diffusion length, for neutrons, respectively.

The relaxation length (or diffusion length) of any material is that thickness of material in which the incident radiation decreases to a factor 1/e (0.368) of its original intensity. It is defined for neutrons, as well as for photons, and, under the conditions of a collimated beam, as already seen, it represents the average distance which a neutron or a photon travels in a medium before an interaction occurs.

Notice, in fact, from Eq. (9) that it coincides with the reciprocal of the linear attenuation coefficient, for photons and, from Eqs. (20) and (18), with the reciprocal of the macroscopic removal cross-section and the diffusion length for fast and thermal neutrons, respectively.

Measured fast neutrons relaxation lengths in various concretes are given in Table 3.1.

Table 3.1 – Measured neutron relaxation lengths in concrete after Jaeger (1970).

Concrete	Density [g/cm³]	Neutron energy	Range of measurement [cm]	Relaxation length [cm]	Ref.
Boron cement	2.0	Fast		7.6	E, J
Ordinary (Portland)	2.3	Fast		12.2	C
Ordinary (Portland)	2.33	Fast		11.1	G
Ordinary (Portland)	2.4	Fast		10.5	G
Ordinary (Portland)	2.4	>1.5 MeV	20– 80	11.0	J
Ordinary (Portland)	2.4	>1.5 MeV	80–120	16.0	J
Ordinary (Portland)	2.5	Fast		11.1	C
Limonite	2.7	>1.5 MeV	20– 80	9.0	J
Limonite	2.7	>1.5 MeV	80–120	13.0	J
Barytes	3.1	Fast		10.0	K
Barytes	3.46	Fast		8.0	H
Barytes	3.5	>1 MeV	30– 60	8.0 ± 0.3	B
Barytes	3.5	Fast		9.5	H
Magnetite	3.29	Fast		8.75	I
W-1	3.6	Fast		6.6	C, E
Ilmenite	3.52	Fast		13.4	A
Brookhaven	4.3	Fast		6.3	H
Brookhaven	4.5	Fast		6.3	H
MO	5.6	Fast		6.1	L
Ml	4.7	Fast		5.9	L
Ferrophosphorus	4.84	Fast		7.19	F
Limonite + steel	4.24	Fast		7.7	D
Limonite + steel	4.15	Fast		7.9	D
Steel shot	5.9	Fast		6.0	H

References:

A. Argone National Laboratory, Reactor Physics Constants, ANL-5800, U.S. Government Printing Office, July 1963.
B. B. T. Price, C. C. Horton, and K. T. Spinney: Radiation Shielding, New Nork: Pergamon Press, 1957.
C. B. T. Price: The Attenuation of Gamma Radiation and Neutrons in the Shield of BEPO, British Report AERE R/R-872, Feb. 28, 1952 (Classified).
D. W. L. Bunch: Attenuation Properties of High-Density Portland Cement Concretes as a Function of Temperature, USAEC Report HW54656, January 22, 1958.
E. H. P. Sleeper, Jr.: A Critical Review of ORNL Shield Measurements: Neutron Attenuation, USAEC Report ORNL-436 (Decl.) Oak Ridge National Laboratory, Dec. 21, 1949.
F. E. G. Peterson: Shielding Properties of Ferrophosphorus Concrete as a Function of Temperature, USAEC Report HW64774, July 15, 1960.
G. E. G. Peterson: Shielding Properties of Ordinary Concrete as a Function of Temperature, USAEC Report HW 65572, August 2, 1960.
H. C. R. Tipton, Jr. (Ed.): Reactor Handbook, 2nd Ed., Vol. I "Materials" Chapter 51, Interscience 1960.
I. D. E. Wood: The Effect of Temperature on the Neutron Attenuation of Magnetite Concrete, USAEC Report HW58497, December 1958.
J. V. S. DiKarev et al.: The Shielding Properties of Concrete. J. Nucl. Energy *5*, 1, 166 (1957).
K. T. V. Blosser et al.: A Study of the Nuclear and Physical Properties of the ORNL Graphite Reactor Shield, USAEC Report ORNL-2195, Sept. 8, 1958.
L. A. S. Kitzes, Th. Rockwell, and R. B. Gallaher: Summaries of Studies on Oxychloride Cement and Concrete, USAEC Report ORNL-709, Oak Ridge National Laboratory (Unpublished).

The removal cross-section of a material is a measure of the probability that a fast neutron will reach a given detection point without being absorbed or scattered out of its original path. For concrete, consisting of a mixture of several compounds, the macroscopic removal cross-section can be obtained from a weighted average of its constituent elements:

$$\left(\frac{\Sigma_R}{\rho}\right)_{mix} = \sum_{i=1}^{N} \frac{N_0}{A_i} f_{wi}\, \sigma_i \,, \tag{21}$$

where N is the total number of elements in the chemical composition of the material, N_0 is the Avogadro's number [atoms/mole], A_i is the atomic weight of element i [grams/mole], f_{wi} its fractional weight, σ_i its microscopic removal cross-section [cm^2/atom] and ρ is the density of the material [g/cm^3].

This expression for Σ_R is independent of the energy of the incident neutron and holds, actually, for very high-energy neutrons from fission. Actually the cross-section is energy dependent and,

in the case of concrete, if the thickness is relatively thin (of the order of three relaxation lengths or less), such that there is not enough hydrogen in the concrete to thermalize (and subsequently capture) the majority of delayed neutrons in the given thickness, the total fast flux may be higher than that predicted by Eq. (20), due to a prevented slow down of a significant amount of fast neutrons. In Table 3.2 are some typical values of the macroscopic cross-section for different concrete mixtures reported in Table 3.3. The quantities included in Table 3.2 are: the measured macroscopic removal cross-section Σ_r, the reciprocal of the relaxation length λ, the total macroscopic cross-section Σ_t, the macroscopic absorption cross-sections for thermal neutrons $\Sigma_{a(the)}$, the thermal diffusion coefficient D_{th}, and the thermal attenuation coefficient K_{th}, equal to the reciprocal of the diffusion length L (Kaplan 1989):

$$L = \sqrt{\frac{D}{\Sigma_{a(the)}}} . \qquad (22)$$

Table 3.2 – Neutron constants for concretes (Walker and Grotenhuis 1961).

Concrete[a)]	Density ϱ [g/cm^3]	Σ_R [cm^{-1}] (calc)	Σ_r [cm^{-1}] (exptl.)	$1/\lambda$ [cm^{-1}] (exptl.)	Σ_t [cm^{-1}] (calc)	$\Sigma_a^{(the)}$ [cm^{-1}] (calc)	D_{th} [cm]	K_{th} [cm^{-1}]
01	2.33	0.0740	0.083[b)]	0.090	0.149	0.0059	0.968	0.0778
02-a	2.30	0.0849	—	—	0.174	0.0094	0.484	0.139
02-b	2.20	0.0748	—	—	0.154	0.0074	0.749	0.0993
03	2.39	0.0837	0.086	—	0.168	0.0097	0.512	0.137
04	2.35	0.0793	—	—	0.162	0.0086	0.688	0.112
05	2.50	0.0833	—	—	0.179	0.0187	0.485	0.196
06	1.30	0.0611	—	—	0.126	0.380	0.406	0.968
07	2.09	0.0836	—	—	0.168	0.0850	0.378	0.474
0-HW1	2.33	0.0781	0.078	0.083	0.156	0.0129	0.629	0.143
0-HW2	2.26	0.0712	0.0735	0.078	0.142	0.0115	0.926	0.112
FP-a	4.68	0.125	0.15[b)]	—	0.216	0.0924	0.344	0.518
FP-b	4.57	0.115	—	—	0.195	0.0903	0.463	0.442
FP-HW1	4.82	0.125	0.131	0.139	0.216	0.0902	0.358	0.502
FP-HW3	4.67	0.111	0.128	0.136	0.187	0.0872	0.585	0.386
BA-a	3.50	0.0926	—	0.125	0.188	0.0197	0.440	0.212
BA-b	3.39	0.0817	—	—	0.165	0.0176	0.667	0.162
BA-H	2.575	0.0643	—	—	0.125	0.0220	0.912	0.155
BAHA	2.35	0.0770	—	—	0.155	0.0128	0.421	0.174
BAHA-d	2.28	0.0773	—	0.098	0.158	0.0111	0.412	0.164
BA-OR	3.30	0.0985	0.0993	—	0.199	0.0224	0.334	0.259
M-a	3.55	0.106	—	—	0.198	0.0553	0.393	0.375
M-b	3.45	0.0966	—	—	0.179	0.0534	0.540	0.314
M-c	3.62	0.133	—	—	0.212	0.0566	0.330	0.414
M-HW1	3.29	0.0984	—	0.114	0.186	0.0402	0.482	0.289
M-HW2	3.27	0.0965	—	0.105	0.182	0.0398	0.517	0.277
I-1a	3.50	0.107	—	—	0.209	0.0848	0.419	0.450
I-1b	3.40	0.098	—	—	0.190	0.0829	0.592	0.374
I-2a	3.76	0.111	—	—	0.213	0.0975	0.401	0.493
I-2b	3.66	0.107	—	—	0.193	0.0956	0.557	0.414
I-NRU	3.49	0.0990	—	—	0.188	0.0699	0.536	0.361
I-NRUe	3.44	0.0983	—	0.075	0.187	0.0766	0.579	0.364
MS-a	4.70	0.124	—	—	0.213	0.0953	0.337	0.532
MS-b	4.60	0.114	—	—	0.193	0.0933	0.445	0.458
MS-c	4.73	0.127	—	—	0.220	0.0959	0.313	0.554
LS-a	4.54	0.121	0.114[b)]	—	0.207	0.0888	0.340	0.511
LS-b	4.44	0.111	—	—	0.187	0.0868	0.445	0.437
LS-c	4.65	0.132	—	—	0.230	0.0910	0.268	0.583
LS-BRa	4.16	0.110	—	—	0.191	0.0800	0.382	0.458
LS-BRb	4.08	0.102	—	—	0.173	0.0783	0.502	0.395
LS-BRc	4.28	0.122	—	0.159	0.214	0.0823	0.290	0.532
LS-HW1	4.23	0.118	0.128	—	0.208	0.0800	0.316	0.503
LS-HW2	4.14	0.110	0.119	—	0.190	0.0782	0.396	0.445

Table 3.3 – Summary of concrete composition for reference mixtures in Table 3.2 (Walker and Grotenhuis 1961).

Concrete	Symbol[a]	Density (ϱ) g/cm³	Density (ϱ) lb/cu.ft	Composition [lb/cu.yd] Water	Cement	Aggregate	Steel punchings	Total	Ref.
Ordinary									
1	01	2.33	145						16
2a	02a	2.30	144	260	318	3300		3878	16
2b	02b	2.20	137			(sand and gravel)			16
3 ORNL	03	2.39	149						4
4 NBS	04	2.35	147						8
5 Harwell	05	2.50	156						2
6 APDA[f]	06	1.30	80	0.231[h]	0.256[h]	0.513[f, h]			11
7 APDA[g]	07	2.09	130	373	525	2032 950		3887	11
0 HW1[j]	0-HW1	2.33	145			(serpentine) (sand)			14
0 HW2[j]	0-HW2	2.26	141						14
Ferrophosphorus									
1a	FP-a	4.68	292	383	730	4070 2710		7893	10
1b	FP-b	4.57	285			(coarse) (fine)			10
HW1 Hanford[j]	FP-HW1	4.82	301						13
HW3 Hanford[j]	FP-HW3	4.67	292						13
Barytes									
1a	BA-a	3.50	219						16
1b	BA-b	3.39	212						16
H Harwell	BA-H	2.575	160						2
Haydite ORNL, X-10	BAHA	2.35	147	10.0[i]	16.3[i]	46.4[i] 27.3[i]			5
Haydite ORNL, X-10	BAHA-d	2.28	142			(barytes) (haydite)			5
OR ORNL	BA-OR	3.30	206	383	468	4711		5562	18
Magnetite									
1a EBWR	M-a	3.55	222	330	875	2623 2160		5988	7, 10
1b ANL, EBWR	M-b	3.45	215			(coarse) (fine)			7, 10
1c ANL, EBWR	M-c	3.62	226	330	875	2700 2200		6105	7, 10
HW1 Hanford[j]	M-HW1	3.29	205						17
HW2 Hanford[j]	M-HW2	3.27	204						17
Ilmenite									
1a New York ore	I-1a	3.50	219	330	875	4695		5988	12
1b New York ore	I-1b	3.40	212						12
2a Swedish ore	I-2a	3.76	235	330	875	5140		6345	12
2b Swedish ore	I-2b	3.66	228						12
NRU Chalk River	I-NRU	3.49	218						15
NRUe Chalk River	I-NRUe	3.44	215						15
Magnetite and steel									
a ANL; EBWR	MS-a	4.70	293	340	940	1846	4800	7926	7, 10
b ANL; EBWR	MS-b	4.60	287						7, 10
c ANL; EBWR	MS-c	4.73	296	340	940	1900	4800	7980	7, 10
Limonite and steel									
1a ANL, CP-5	LS-a	4.54	284	347	980	1661	4680	7668	9
1b ANL, CP-5	LS-b	4.44	277						9
1c ANL, CP-5	LS-c	4.65	290	347	980	1825	4680	7832	1
2a BNL	LS-BRa	4.16	260	296	940	1684	4100	7020	3
2b BNL	LS-BRb	4.08	255						3
2c BNL	LS-BRc	4.28	267	296	940	1880	4100	7216	3
HW1 Hanford[j]	LS-HW1	4.23	264						6
HW2 Hanford[j]	LS-HW2	4.14	258						6

[a] Indicates 100% water retention. Footnotes [a] through [e] refer to the small letters in the symbols in this column.
[b] Indicates 50% water retention.
[c] Indicates 100% water retention including free moisture in mix.
[d] Average composition of four sample cores.
[e] Composition after two years.
[f] Diatomoceous earth aggregate used in concrete.
[g] Serpentine concrete.
[h] Volume fractions.
[i] Weight percent.
[j] HW1 as cured concrete. HW2 heated to 100 °C. HW3 heated to 320 °C.

Fig.3.1 plots the total macroscopic cross-section of ordinary concrete as a function of neutron energy, starting from the microscopic quantities of its compounding elements, for energies above 10keV.

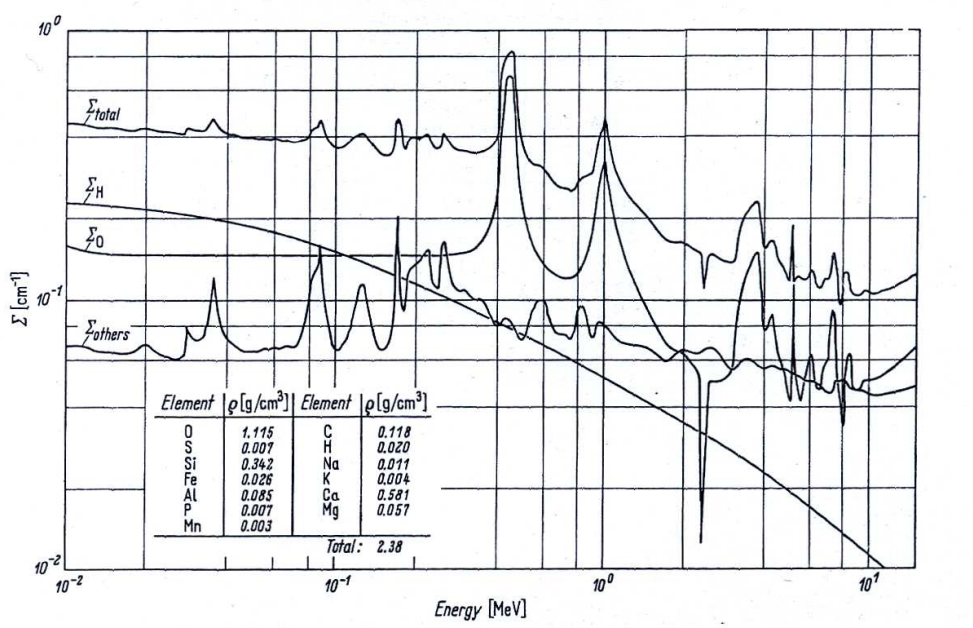

Fig.3.1 – Total macroscopic neutron cross-section for ordinary concrete after Jaeger (1970).

Neutron capture by nuclei of a material always produces secondary gamma-rays, referred to as “capture” gamma-rays. The yield of these emissions is higher if radiative capture is involved, which is meant to be the primary source of internally generated gamma rays from thermal neutrons in concrete (Jaeger 1970).

The yield of capture gamma-rays, or the number of gamma-rays emitted per neutron capture, from 1Mev and 10MeV, is given in Table 3.4 for the different concretes defined in Table 3.3.

Table 3.4 – gamma-rays spectra from thermal neutron capture in several mixtures (Walker and Grotenhuis 1961).

Concrete[a]	Σ_a [cm^{-1}] (calc.)	Photons per neutron capture						
		1 MeV	2 MeV	3 MeV	4 MeV	6 MeV	8 MeV	10 MeV
01	0.0059	0.355	1.29	0.676	0.715	0.408	0.0332	0.0003
02-a	0.0094	0.546	0.533	0.739	0.447	0.143	0.0542	0.0004
02-b	0.0074	0.694	0.678	0.668	0.569	0.182	0.0689	0.0005
03	0.0097	0.374	0.911	0.751	0.483	0.284	0.0514	0.0003
04	0.0086	0.675	0.765	0.694	0.595	0.218	0.0618	0.0004
05	0.0187	0.781	0.796	0.611	0.435	0.277	0.127	0.0006
06	0.380	0.0047	0.0032	0.0171	1.08	0.275	0.0593	0.0078
07	0.0850	0.0477	0.0475	0.0916	1.00	0.261	0.0017	0.0071
0-HW 1	0.0129	0.636	0.813	0.584	0.420	0.270	0.151	0.0010
0-HW 2	0.0115	0.713	0.912	0.533	0.471	0.302	0.169	0.0011
FP-a	0.0924	0.871	0.655	0.373	0.302	0.258	0.312	0.0016
FP-b	0.0903	0.890	0.670	0.359	0.308	0.264	0.319	0.0016
FP-HW 1	0.0902	0.873	0.669	0.309	0.283	0.284	0.320	0.0036
FP-HW 3	0.0872	0.903	0.692	0.286	0.293	0.294	0.331	0.0037
BA-a	0.0197	0.279	0.274	0.426	0.483	0.204	0.0920	0.0008
BA-b	0.0176	0.313	0.307	0.357	0.541	0.229	0.103	0.0009
BA-H	0.0220	0.576	0.511	0.323	0.372	0.224	0.257	0.0016
BAHA	0.0128	0.244	0.391	0.635	0.442	0.176	0.0278	0.0004
BAHAd	0.0111	0.195	0.344	0.618	0.374	0.139	0.0413	0.0005
BA-OR	0.0224	0.330	0.298	0.477	0.383	0.184	0.126	0.0009
M-a	0.0553	0.676	0.721	0.325	0.240	0.304	0.296	0.0020
M-b	0.0534	0.700	0.747	0.301	0.249	0.315	0.307	0.0020
M-c	0.0566	0.660	0.704	0.341	0.235	0.297	0.289	0.0019
M-HW 1	0.0402	0.698	0.615	0.338	0.247	0.244	0.337	0.0019
M-HW 2	0.0398	0.705	0.621	0.332	0.249	0.246	0.340	0.0019
I-1a	0.0848	0.568	1.34	0.242	0.250	0.636	0.0783	0.0018
I-1b	0.0829	0.581	1.37	0.225	0.256	0.650	0.0801	0.0019
I-2a	0.0975	0.599	1.25	0.248	0.248	0.588	0.116	0.0018
I-2b	0.0956	0.611	1.27	0.234	0.253	0.600	0.118	0.0019
I-NRU	0.0699	0.632	1 07	0.246	0.240	0.496	0.189	0.0020
I-NRUe	0.0766	0.618	1.15	0.229	0.239	0.540	0.165	0.0020
MS-a	0.0953	0.705	0.647	0.304	0.234	0.271	0.339	0.0020
MS-b	0.0933	0.723	0.661	0.290	0.239	0.277	0.346	0.0020
MS-c	0.0959	0.701	0.643	0.309	0.232	0.269	0.337	0.0020
LS-a	0.0888	0.713	0.600	0.315	0.237	0.245	0.352	0.0019
LS-b	0.0868	0.729	0.614	0.299	0.243	0.251	0.360	0.0020
LS-c	0.0910	0.696	0.585	0.332	0.231	0.239	0.344	0.0019
LS-BRa	0.0800	0.712	0.603	0.313	0.236	0.246	0.354	0.0020
LS-BRb	0.0783	0.728	0.616	0.298	0.241	0.252	0.362	0.0020
LS-BRc	0.0823	0.693	0.586	0.332	0.229	0.240	0.345	0.0019
LS-HW 1	0.0800	0.695	0.589	0.324	0.230	0.241	0.339	0.0019
LS-HW 2	0.0782	0.711	0.602	0.309	0.235	0.247	0.357	0.0020

The linear attenuation coefficient is the probability of a photon interacting in a particular way with a given material, per unit path length. Since the specific weight of a material depends on its physical state (in the case of concrete, its moisture content), the linear attenuation coefficient is, for reference purposes, expressed as a mass attenuation coefficient, i.e. the linear attenuation coefficient per unit mass of the material, to obviate the effects of variations in the density of materials (Kaplan 1989).

Values of the mass attenuation coefficients μ for various concretes at various gamma energies are tabulated in Table 3.5 for the same reference mixtures reported in Table 3.3.

The mass attenuation coefficient, similarly to the macroscopic cross-section for fast neutrons, is related to the sum of the gamma-ray interaction cross-sections for a mixture, like concrete, by:

$$\left(\frac{\mu}{\rho}\right)_{mix} = \sum_{i=1}^{N}\left(\frac{\mu}{\rho}\right)_i f_{wi}\,, \tag{23}$$

where the previous meaning holds for f_{wi} and:

$$\left(\frac{\mu}{\rho}\right)_i = \frac{N_0}{A_i}\left(\sigma_{Compt} + \sigma_{ph} + \sigma_{pp}\right)_i\,. \tag{24}$$

The quantities σ_{Compt}, σ_{ph} and σ_{pp} are the microscopic interaction cross-section for Compton scattering, photoelectric absorption and pair production, respectively.

Table 3.5 – Total gamma-ray linear attenuation coefficients [cm^{-1}] of several concretes (Walker and Grotenhuis 1961).

Concrete[a)]	Density [g/cm³]	Photon energy								
		0.5 MeV	1 MeV	2 MeV	3 MeV	4 MeV	5 MeV	6 MeV	8 MeV	10 MeV
01	2.33	0.2033	0.1482	0.1042	0.0851	0.0743	0.0675	0.0630	0.0570	0.0539
02-a	2.30	0.2017	0.1473	0.1034	0.0841	0.0732	0.0663	0.0615	0.0554	0.0520
02-b	2.20	0.1917	0.1400	0.0983	0.0801	0.0697	0.0632	0.0587	0.0529	0.0497
03	2.39	0.2095	0.1527	0.1074	0.0877	0.0766	0.0696	0.0649	0.0588	0.0556
04	2.35	0.2040	0.1489	0.1046	0.0852	0.0744	0.0674	0.0627	0.0566	0.0533
05	2.50	0.2188	0.1596	0.1122	0.0916	0.0801	0.0728	0.0679	0.0615	0.0581
06	1.30	0.1154	0.0843	0.0591	0.0479	0.0415	0.0374	0.0346	0.0309	0.0288
07	2.09	0.1841	0.1343	0.0943	0.0768	0.0668	0.0605	0.0561	0.0506	0.0474
0-HW1	2.33	0.2025	0.1476	0.1038	0.0849	0.0744	0.0677	0.0632	0.0575	0.0545
0-HW2	2.26	0.1957	0.1426	0.1004	0.0821	0.0720	0.0656	0.0613	0.0588	0.0529
FP-a	4.68	0.3946	0.2851	0.2024	0.1697	0.1529	0.1430	0.1371	0.1304	0.1280
FP-b	4.57	0.3839	0.2773	0.1969	0.1653	0.1491	0.1397	0.1341	0.1277	0.1256
FP-HW1	4.82	0.4058	0.2930	0.2080	0.1747	0.1576	0.1477	0.1418	0.1351	0.1329
FP-HW3	4.67	0.3915	0.2825	0.2007	0.1688	0.1526	0.1432	0.1377	0.1316	0.1297
BA-a	3.50	0.3168	0.2142	0.1517	0.1295	0.1187	0.1130	0.1097	0.1066	0.1064
BA-b	3.39	0.3054	0.2058	0.1459	0.1248	0.1147	0.1094	0.1065	0.1038	0.1038
BA-H	2.57	0.2265	0.1569	0.1112	0.0941	0.0856	0.0809	0.0781	0.0752	0.0745
BAHA	2.35	0.2108	0.1470	0.1038	0.0869	0.0782	0.0731	0.0698	0.0661	0.0647
BAHA-d	2.28	0.2038	0.1425	0.1005	0.0838	0.0751	0.0699	0.0666	0.0627	0.0610
BA-OR	3.30	0.2964	0.2030	0.1436	0.1217	0.1107	0.1047	0.1010	0.0972	0.0963
M-a	3.55	0.3027	0.2192	0.1550	0.1290	0.1150	0.1066	0.1014	0.0950	0.0922
M-b	3.45	0.2932	0.2132	0.1502	0.1251	0.1117	0.1036	0.0987	0.0927	0.0900
M-c	3.62	0.3093	0.2240	0.1684	0.1317	0.1174	0.1087	0.1033	0.0967	0.0937
M-HW1	3.29	0.2809	0.2037	0.1439	0.1192	0.1059	0.0977	0.0926	0.0863	0.0833
M-HW2	3.27	0.2789	0.2022	0.1429	0.1184	0.1052	0.0971	0.0921	0.0958	0.0828
I-1a	3.50	0.2997	0.2173	0.1535	0.1271	0.1129	0.1042	0.0987	0.0919	0.0886
I-1b	3.40	0.2902	0.2103	0.1486	0.1233	0.1096	0.1012	0.0960	0.0895	0.0864
I-2a	3.76	0.3193	0.2312	0.1635	0.1361	0.1214	0.1126	0.1071	0.1005	0.0975
I-2b	3.66	0.3090	0.2236	0.1582	0.1319	0.1178	0.1093	0.1042	0.0979	0.0951
I-NRU	3.49	0.2965	0.2147	0.1518	0.1263	0.1126	0.1043	0.0992	0.0930	0.0902
I-NRUe	3.44	0.2909	0.2106	0.1490	0.1241	0.1108	0.1028	0.0978	0.0919	0.0892
MS-a	4.70	0.3957	0.2855	0.2027	0.1706	0.1542	0.1446	0.1391	0.1328	0.1308
MS-b	4.60	0.3860	0.2783	0.1977	0.1666	0.1507	0.1416	0.1363	0.1304	0.1286
MS-c	4.73	0.3989	0.2878	0.2043	0.1719	0.1553	0.1456	0.1400	0.1336	0.1315
LS-a	4.54	0.3827	0.2761	0.1960	0.1650	0.1490	0.1398	0.1344	0.1283	0.1263
LS-b	4.44	0.3727	0.2688	0.1909	0.1609	0.1455	0.1367	0.1316	0.1258	0.1241
LS-c	4.65	0.3935	0.2840	0.2015	0.1694	0.1528	0.1432	0.1375	0.1310	0.1288
LS-BRa	4.16	0.3511	0.2533	0.1798	0.1513	0.1365	0.1280	0.1230	0.1173	0.1154
LS-BRb	4.08	0.3426	0.2471	0.1755	0.1478	0.1336	0.1253	0.1206	0.1152	0.1135
LS-BRc	4.28	0.3623	0.2615	0.1855	0.1558	0.1405	0.1315	0.1262	0.1201	0.1179
LS-HW1	4.23	0.3583	0.2583	0.1833	0.1539	0.1387	0.1298	0.1246	0.1185	0.1164
LS-HW2	4.14	0.3492	0.2519	0.1788	0.1503	0.1357	0.1271	0.1221	0.1164	0.1144

In Table 3.6 the energy absorption linear attenuation μ_a for some concretes, as already defined in §2.1.2.1.1, are reported.

Table 3.6 – Gamma-ray energy absorption linear attenuation coefficients [cm^{-1}] of several concretes (Walker and Grotenhuis 1961).

Concrete[a)]	Density [g/cm³]	Photon energy								
		0.5 MeV	1 MeV	2 MeV	3 MeV	4 MeV	5 MeV	6 MeV	8 MeV	10 MeV
01	2.33	0.0694	0.0650	0.0558	0.0507	0.0477	0.0456	0.0444	0.0428	0.0423
02-a	2.30	0.0684	0.0644	0.0551	0.0499	0.0466	0.0443	0.0429	0.0410	0.0403
02-b	2.20	0.0650	0.0612	0.0524	0.0475	0.0444	0.0423	0.0410	0.0392	0.0386
03	2.39	0.0716	0.0670	0.0575	0.0523	0.0492	0.0470	0.0458	0.0441	0.0437
04	2.35	0.0693	0.0651	0.0558	0.0506	0.0475	0.0452	0.0439	0.0422	0.0415
05	2.50	0.0746	0.0699	0.0600	0.0546	0.0513	0.0491	0.0478	0.0461	0.0456
06	1.30	0.0391	0.0369	0.0315	0.0283	0.0263	0.0249	0.0239	0.0227	0.0222
07	2.09	0.0627	0.0589	0.0504	0.0456	0.0426	0.0406	0.0393	0.0376	0.0370
0-HW1	2.33	0.0691	0.0647	0.0556	0.0507	0.0478	0.0458	0.0447	0.0432	0.0429
0-HW2	2.26	0.0668	0.0625	0.0537	0.0490	0.0463	0.0444	0.0434	0.0420	0.0417
FP-a	4.68	0.1382	0.1251	0.1096	0.1040	0.1020	0.1015	0.1020	0.1035	0.1067
FP-b	4.75	0.1345	0.1217	0.1068	0.1014	0.0996	0.0993	0.0999	0.1015	0.1048
FP-HW1	4.82	0.1424	0.1286	0.1127	0.1072	0.1054	0.1051	0.1057	0.1075	0.1110
FP-HW3	4.67	0.1375	0.1240	0.1088	0.1038	0.1022	0.1022	0.1029	0.1049	0.1085
BA-a	3.50	0.1263	0.0978	0.0840	0.0813	0.0815	0.0825	0.0839	0.0869	0.0905
BA-b	3.39	0.1224	0.0941	0.0809	0.0786	0.0790	0.0802	0.0817	0.0848	0.0885
BA-H	2.57	0.0862	0.0705	0.0610	0.0585	0.0580	0.0583	0.0590	0.0605	0.0626
BAHA	2.35	0.0790	0.0659	0.0566	0.0534	0.0522	0.0517	0.0518	0.0522	0.0535
BAHA-d	2.28	0.0762	0.0639	0.0547	0.0514	0.0499	0.0493	0.0491	0.0492	0.0502
BA-OR	3.30	0.1153	0.0920	0.0791	0.0758	0.0752	0.0756	0.0764	0.0783	0.0811
M-a	3.55	0.1055	0.0962	0.0837	0.0784	0.0759	0.0746	0.0743	0.0742	0.0756
M-b	3.45	0.1022	0.0932	0.0811	0.0761	0.0738	0.0727	0.0724	0.0725	0.0740
M-c	3.62	0.1077	0.0984	0.0855	0.0800	0.0773	0.0760	0.0756	0.0754	0.0768
M-HW1	3.29	0.0976	0.0894	0.0776	0.0722	0.0695	0.0680	0.0674	0.0669	0.0678
M-HW2	3.27	0.0969	0.0888	0.0771	0.0717	0.0691	0.0676	0.0670	0.0665	0.0675
I-1a	3.50	0.1041	0.0955	0.0828	0.0770	0.0741	0.0725	0.0718	0.0712	0.0722
I-1b	3.40	0.1009	0.0924	0.0802	0.0747	0.0720	0.0705	0.0700	0.0695	0.0705
I-2a	3.76	0.1114	0.1016	0.0884	0.0828	0.0801	0.0789	0.0786	0.0785	0.0801
I-2b	3.66	0.1079	0.0982	0.0855	0.0803	0.0779	0.0768	0.0766	0.0767	0.0783
I-NRU	3.49	0.1033	0.0943	0.0820	0.0767	0.0743	0.0731	0.0727	0.0726	0.0740
I-NRUe	3.44	0.1015	0.0925	0.0805	0.0755	0.0732	0.0721	0.0718	0.0719	0.0733
MS-a	4.70	0.1392	0.1253	0.1100	0.1048	0.1032	0.1032	0.1039	0.1058	0.1095
MS-b	4.60	0.1359	0.1221	0.1073	0.1025	0.1011	0.1012	0.1020	0.1041	0.1078
MS-c	4.73	0.1403	0.1263	0.1108	0.1056	0.1039	0.1038	0.1045	0.1064	0.1100
LS-a	4.54	0.1345	0.1211	0.1063	0.1013	0.0997	0.0996	0.1004	0.1022	0.1057
LS-b	4.44	0.1311	0.1179	0.1036	0.0989	0.0975	0.0976	0.0984	0.1004	0.1039
LS-c	4.65	0.1382	0.1246	0.1092	0.1039	0.1021	0.1018	0.1024	0.1041	0.1075
LS-BRa	4.16	0.1234	0.1112	0.0975	0.0928	0.0913	0.0912	0.0918	0.0933	0.0964
LS-BRb	4.08	0.1205	0.1085	0.0952	0.0908	0.0895	0.0894	0.0901	0.0918	0.0950
LS-BRc	4.28	0.1272	0.1148	0.1006	0.0956	0.0938	0.0935	0.0940	0.0953	0.0984
LS-HW1	4.23	0.1257	0.1134	0.0993	0.0944	0.0926	0.0923	0.0927	0.0941	0.0971
LS-HW2	4.14	0.1227	0.1106	0.0970	0.0923	0.0907	0.0905	0.0910	0.0925	0.0956

As already mentioned, the simplest model of attenuation of photons through a slab (a mono-energetic, mono-directional beam of gamma-rays, i.e. narrow beam conditions) is expressed by the Lambert's law reported in Eq. (5), where μ is the linear gamma-ray mass attenuation coefficient in [cm^{-1}].

To account for scattering processes, the introduction of buildup factors is needed.

3.3 Nuclear heat

Elevated temperatures in the shields can result from the attenuation or absorption of radiation in the shielding medium, because of the radiation heat production.

Due to the low thermal conductivity proper of concrete, this heat does not dissipate readily so that an overall temperature rise is observed within. This rise is dependent on the energy of the incident flux but independent of the thickness of the shield; it is the temperature gradient, however, that is sensitive to the thickness.

Under nuclear heating concrete tends to expand, compromising the durability of the structure, as a consequence of the formation of cracks due to its low tensile strength and subsequent localization of internal stresses caused by the temperature buildup.

Concrete protecting nuclear facilities as a so-called “biological barrier” should, therefore, be prevented from excessive radiation, that is why in nuclear reactors, usually, a first steel barrier is designed for thermal loads and cooling systems are provided to reduce the temperature rise. Also wide employment is made of prestressed concrete for reactor vessels, in order to guarantee no tensile stress in the material, under normal use conditions, and withstand internal design tensile pressures, during accident situations.

After experimental data on heating concrete, Jaeger (1970) indicates that the maximum allowable internal heating rate for concrete is about 1mW/cm^3, which is meant to produce a temperature rise of the order of 6°C within the material; Price et al. (1957) mention that an overall temperature rise of 30°C in a monolithic concrete shield is admissible; this value is declared to correspond to an incident energy flux density of 10^{11}MeV/(cm^2 s), approximately 16mW/cm^2 (1meV/(cm^2 s) = 1.6x10^{-13} W/cm^2).

Kaplan (1989) quotes the American National Standard ANSI/ANS-6.4-1985, when he states that the temperature buildup in concrete is negligible for incident energy flux densities less than 10^{10}MeV/(cm^2 s) and that, if concrete temperatures are maintained below 65°C, there is even no need to considerate temperature effects on concrete. On the contrary, when the incident flux densities are more than that prescribed or the operating temperatures higher than 65°C, radiation damage on the physico-mechanical properties of concrete must be taken into account.

The general requirements of the ACI Standard ACI 349-85 state that, under normal operation, the temperature of the concrete shall not exceed 66°C, except for local areas, where the recommended value is 93°C, higher temperatures being allowed, if justified by ad hoc test campaigns.

Damage to concrete caused by nuclear heating is known to be more important than the effects of radiation damage, because the flux density required to cause damage from nuclear heating is very much less than that required to cause direct radiation damage, basically explained by atomic displacements in the lattice.

E.g., a fast neutron exposure of 10^{21} n/cm^2 appears to lead to the reduction of the rapture strength of concrete of a factor of 2 (Kaplan 1989; Price et al. 1957). At a constant (in time) flux of 10^{12} n/(cm^2 s) it would take 30 years, to reach this limit, but such a flux would produce in a few hours, only, a temperature rise of nearly 4 times the recommended limit for internal nuclear heating (6°C).

3.3.1 Volumetric heating rate

An expression for calculating the volumetric heating rate along the thickness of a concrete slab is suggested by (Jaeger 1970; Kaplan 1989) for neutrons:

$$H_0(x) = 1.6 \cdot 10^{-13} \, \Sigma_C(E) \, \varphi(x) \, E_B \tag{25}$$

where:

$H_0(x)$ = volumetric heating rate at point x in the slab [W/cm^3]

$\Sigma_C(E)$ = macroscopic capture cross-section for neutrons of energy E [cm^{-1}]; for mixtures, like concrete:

$$\left(\frac{\Sigma_C}{\rho}\right)_{mix} = \sum_{i=1}^{N} \frac{N_0}{A_i} f_{wi} \, \sigma_{ci} \quad [\text{cm}^2/\text{g}] \tag{26}$$

with: σ_{ci} = microscopic capture cross-section for the ith element in concrete [cm^2/atom]

$\phi(x)$ = neutron flux of energy E, at point x [n/(cm^2 s)]

E_B = binding energy for the capture reaction (8MeV for most reactions) [MeV]

ρ = density of concrete [g/cm^3]

N_0 = Avogadro's number [atoms/mole]

A = atomic weight of the compound element

and for photons:

$$H_0(x) = 1.6 \cdot 10^{-13} \, \mu_{en} \, \varphi(E) \, E\gamma \tag{27}$$

where:

μ_{en} = energy absorption coefficient for gamma-rays of energy E_γ [cm^{-1}]; for mixtures, like concrete:

$$\left(\frac{\mu_{en}}{\rho}\right)_{mix} = \sum_{i=1}^{N}\left(\frac{\mu_{en}}{\rho}\right)_i f_{wi} \quad [cm^2/g] \tag{28}$$

$\phi(E)$ = gamma-rays flux at energy E_γ [photons/(cm^2 s)].

As for neutrons, the greatest amount of heat to the shield comes from thermal neutron capture, for which the energy release per capture is quite independent from the initial energy of the neutron, therefore $\phi(x)$ may be taken as the total flux $\phi(x)_{TOT}$.

While for photons, *each* gamma-rays has a specific energy, therefore *each* gamma-ray flux must be computed for each heat contribution. The total energy absorption is the sum of these contributions for the different gamma-rays produced.

A superposition holds, similarly, when neutrons and photons are irradiated at the same time.

3.3.2 Heating through a slab of given thickness

The approximate heating rate through a slab of given thickness results from integration in space of Eqs. (20) and (9) for neutrons and photons, respectively (Jaeger 1970), reminding that in both cases the attenuation in the material is exponential in nature:

$$H_1(h) = 1.6 \cdot 10^{-13}\ \lambda\ \Sigma_C(E)\ E_B\ \varphi_0 \left(1 - e^{-\frac{h}{\lambda}}\right) \tag{29}$$

$$H_1(h) = 1.6 \cdot 10^{-13}\ \frac{\mu_{en}}{\mu}\ E_\gamma\ \Gamma_0 \left(1 - e^{-\mu h}\right) \tag{30}$$

where:

$H_1(h)$ = heat through a slab of thickness h [W/cm^2]

h = thickness of the slab [cm]

ϕ_0 = incident neutron flux on the slab [n/(cm^2 s)]

λ = relaxation length for neutrons [cm]

Γ_0 = incident gamma-rays flux of energy E_γ on the slab [photons/(cm^2 s)]

μ = total linear mass absorption coefficient for energy E_γ [cm^{-1}].

3.3.3 Estimated temperature rise within a shield

The temperature distribution during steady-state conditions in a shielding slab exposed to a collimated beam of photons is provided, according to (Gill 1964; Hallidy 1954; Kaplan 1989], by:

$$k \frac{d^2T}{dx^2} = -\frac{E\, I_0}{\lambda} e^{-\frac{x}{\lambda}} = -\frac{q_0}{\lambda} e^{-\frac{x}{\lambda}} \quad (31)$$

where:

k = thermal conductivity of concrete [W/(cm °C)]

T = temperature [°C]

x = distance into the shield [cm]

E = photon energy [MeV]

I_0 = flux density [photons/(cm^2 s)]

λ = relaxation length [cm]

q_0 = heat or energy flux density [MeV/(cm^2 s)].

An approximate formula for estimating the temperature rise from nuclear heating, in function of the distance from the directly impinged surface, into a shield, is provided by Jaeger (1970):

$$\Delta T(x) = \frac{H_0\, \lambda^2}{k} \left(1 - e^{-\frac{x}{\lambda}} \right) \quad (32)$$

where:

ΔT(x) = temperature rise at distance x into the shield [°C]

x = distance into the shield [cm]

H_0 = heating rate at point x in the shield [W/cm^3]

λ = relaxation length (may be the reciprocal of the removal cross-section Σ_R for neutrons or the reciprocal of the gamma-ray linear absorption coefficient μ for photons) [cm]

k = thermal conductivity of concrete [W/(cm °C)]

The temperature distribution in a concrete shield due to the generation of heat from nuclear radiation is known to have a typical parabolic trend (Gill 1964; Kaplan 1989) and to be a superposition of effects between direct radiation (linear gradient of thermal conduction law

by Fourier) and nuclear heating, as reported in Fig.3.2 below, provided a gradient t1>t2 is present at the two faces of the slab.

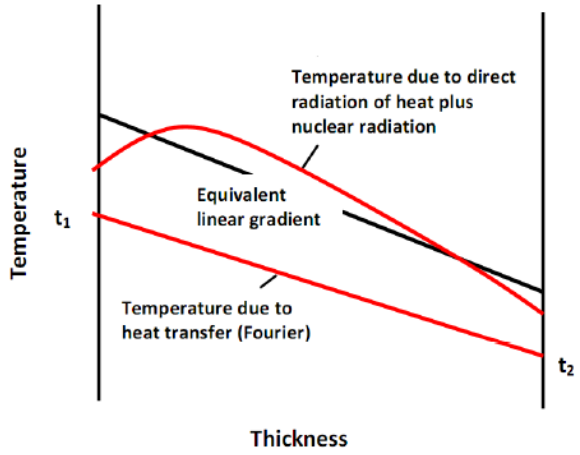

Fig.3.2 – Temperature distribution in a concrete shield after (Gill 1964; Kaplan 1989).

Komarovskii (1965) gives an analytical expression for estimating the temperature distribution in a plane concrete slab shield:

$$T(x) = t_1 + \frac{c\,E\,I_0\,\lambda}{k}\left(1 - \frac{x}{h} - e^{-\frac{x}{\lambda}}\right) - \frac{x}{h}\left(t_1 - t_2\right). \qquad (33)$$

He also provides the distance x_{max} between the inner face of the shield and the point of maximum temperature:

$$x_{max} = -\lambda \ln\left[\frac{\lambda}{h} + \frac{k}{c\,E\,I_0\,h}\left(t_1 - t_2\right)\right] \qquad (34)$$

where:

t_1 and t_2 = temperature at inner and outer faces of the shield [°C]

k = thermal conductivity of concrete [W/(cm °C)]

c = conversion factor (1.6×10^{-13} W s/MeV)

x = distance from the inner face and a point inside the shield [cm]

h = shield thickness [cm]

E = average energy of gamma-rays or neutrons [MeV]

I_0 = flux density [photons or neutrons/(cm^2 s)]

λ = relaxation length [cm]

3.3.4 Estimated stress state for nuclear heat

The average stress σ_0, developed in a concrete slab of given thickness and given heating rate at one surface, is provided by Jaeger (1970):

$$\sigma_0 = P_0 - \frac{E\, H_0\, \lambda^2\, \alpha}{k\,(1-\nu)}\left[1-\frac{2\lambda}{h}-\left(1-\frac{2\lambda}{h}\right)e^{-\frac{h}{\lambda}}\right] \tag{35}$$

where:

σ_0 = average stress in a slab from internal radiation absorption [lb/in^2]

h = thickness of the slab [ft]

H_0 = heating rate at the surface of the slab [Btu/(h ft^3)]

P_0 = average slab loading [psi]

E = elastic modulus of concrete [psi]

λ = relaxation length [cm]

α = linear heat expansion coefficient of concrete [°C^{-1}]

k = thermal conductivity of concrete [Btu/(h ft °F)]

ν = Poisson's modulus of concrete.

and the thermal stress in one-dimensional geometry is given by:

$$\sigma(x) = \sigma_0 + \frac{E\,\Delta T(x)\,\alpha}{1-\nu} \tag{36}$$

where:

ΔT(x) = temperature rise at distance x into the shield [°F] as given in Eq. (32) for radiation from one side (similar expression is in (Jaeger 1970) for radiation from both sides of the shield).

Positive values for σ_0 or σ(x) are for compression, negative values for tension.

3.4 Use of special concretes for radiation shielding

Concrete has been accepted worldwide as an efficient and convenient shielding material (Baumeister et al. 1957; Browne 1967; Mazzoleni 1956; Price et al. 1957; Rockwell 1956; Sani 1972; Jaeger 1970; Kaplan 1989; Komarovskii 1961). A special attention should be paid in the choice of aggregates, employed in the so-called "improved concretes", in order to optimize the shielding properties with respect to ordinary concrete.

These special concretes act as better shielding material for one of these two reasons, namely:

a) the use of high-density aggregates enhances gamma-rays attenuation. This is the case of barytes concrete, ferro-phosphorus concrete, magnetite concrete, hematite concrete, e.g.
b) The presence of minerals capable of increasing the fixed-water content of concrete or retaining their water of crystallization at high temperatures justifies the good performance of such concretes against fast neutron radiation. This is the case of serpentine concrete, limonite concrete.
c) The presence of certain elements in the minerals, such as boron, enhances the thermal neutron attenuation properties of concrete and cuts down secondary gamma-rays production.

Barytes (barium sulphate – $BaSO_4$) is used both as coarse and as fine aggregate for concrete. The mean volumetric weight of barytes concrete is about 50% higher than that of ordinary concrete and, therefore, it is more efficient than the latter against gamma-radiation. Additionally, this high-specific-weight aggregate has almost null reactivity with alkalis in the cement, which is understood to be one of the main causes of degradation of concrete during irradiation (Ichikawa and Kimura 2007; Ichikawa and Koizumi 2002; Naus 2007). Moreover, the high neutron capture cross-section of barium ensures good shielding properties even against neutrons, though also limonite or other hydrous aggregates can be used in conjunction to improve it (Komarovskii 1961).

However the cohesion of barytes aggregates decreases as the temperature increases. It would appear impossible to operate a barytes concrete beyond 180°C, since the coefficient of linear exansion becomes very high beyond that temperature (Jaeger 1970).

Ferro-phosphorus aggregates help reaching high densities on the mortars, thus ensuring a good efficiency against photons. A volume increase at temperatures above 350°C has been observed (Kaplan 1989), which has been interpreted as due to accelerated rates of oxidation of the aggregate. It has also been reported that ferro-phosphorus concretes may lead to the production of inflammable gases that could develop high pressures if confined and that they experience delay in setting and hardening.

Magnetite (Fe_3O_4) is used as heavy aggregate for special concretes, due to its higher iron content than ordinary concrete. Its water content is not as high as in limonite, another iron ore-aggregate, and, consequently, it has lower shielding properties.

Serpentine is a hydrous magnesium iron phyllosilicate ($(Mg, Fe)_3Si_2O_5(OH)_4$) mineral. It is used as aggregate in concrete because it can retain most of its water of crystallisation at temperatures up to about 500°C, thus keeping its efficiency against neutrons also in the extreme conditions of nuclear reactors.

Limonite ($FeO(OH)\cdot nH_2O$) ores are hydrous iron ores used in concrete primarily because of their high content of chemically combined water (up to twice the concentration of water of an ordinary concrete), thus improving the effectiveness of shielding against neutron flux. Limonite retains its water of crystallization up to 200°C. Fine limonite aggregate, which gives sufficient plasticity to the concrete mix for casting, is used for heavy concretes with the addition of metal scrap (steel punching, sheared bars, steel shots).

As for boron, its employment in concrete mixtures is explained not only by the very large neutron cross-section of boron (in the form of isotope ^{10}B), but also by its capacity of capturing thermal neutrons without the production of penetrating gamma-rays, but only soft rays readily absorbed within the shield. Boron in the isotope of interest is present in several minerals: colemanite, priceite (pandermite), calcium borates, ulexite, gerstley borates. The drawback of the presence of these minerals in concrete mixtures is the observed delaying in the setting of Portland cements, particularly when used in the form of fine particles, which is the preferred solution in order to increase the probability of a homogeneous neutron capture. The employment of alumina cement removes this drawback but brings another one, linked with the reactivity with water of such cements, which may eventually lead to a loss in strength, provided that the water/cement ratio is too high, so that a compromise should be sought in order to satisfy the two opposing requirements: an optimum boron content and a satisfactory concrete. Generally, less than 1% by weight of concrete is the recommended quantity of boron to prevent setting and hardening problems (Kaplan 1989).

Also, due to the presence of boron substances, having very low apparent densities, great care must be taken during the placing of the concrete, in order to avoid segregation of the aggregates.

3.5 Recent advances in the use of concrete for nuclear facilities

Many studies have been carried out, recently, to compare ordinary Portland performances with improved mixtures' shielding properties, which are a proof of an on-going interest for this topic.

A Turkish work has analyzed the effects of the addition of trommel sieve waste (TSW), occurring during manufacture of borax from tincal to clinker (Boncukcuoğlu et al. 2005), which is justified by the huge boron deposits possessed by Turkey.

Admixtures like fly ash (FA) and blast furnace slag (BFS) are studied by Demirboğa et al. as replacement for cement to improve the mechanical properties, decrease the rate of hydration, decrease the alkali aggregate reactivity and decrease the permeability of Portland concrete (Demirboğa et al. 2004, 2007; Türkmen 2008a).

The effect of the autoclaving process and the addition of silica fume on Portland cement, or the use of hydrated Portland cement in conjunction with granulated lead have been studied by El-Faramawy and El-Hosiny (El-Faramawy and El-Hosiny 1998; El-Hosiny and El-Faramawy 2000) for shielding towards gamma-radiation.

Concrete with different amounts of barytes aggregates is compared to concrete with normal-weight aggregates by Akkurt et al. (2006) and barytes concretes with varying water content by Topçu (2003).

The behaviour of hematite-serpentine, ilmenite-limonite, ilmenite concrete and ilmenite concrete, included iron punching additions, is investigated in (Makarious et al. 1996; Bashter et al. 1996) for shielding against fast neutrons and gamma-rays.

The shielding properties of many buildings materials (the linear attenuation coefficient for shielding against gamma-rays and the removal cross-section against neutrons) are still being studied, in relation to wide energy ranges (a few keV until hundreds of GeV). The linear attenuation coefficient is a physical quantity that represents the probability of a photon interacting in a particular way with a given material, per unit path length. It is strictly dependent on the shielding material and strongly determines its degree of absorption of gamma-rays.

The removal cross-section is a similar physical quantity for neutron radiation: it is the probability that a fast or fission energy neutron undergoes a first collision which removes it from the group of penetrating, uncollided neutrons and, if concrete contains sufficient

moderating material (i.e. sufficient water content), the attenuation of neutrons is dominated by the removal process (Kaplan 1989).

Bashter (1997) has calculated these quantities for seven types of concrete: ordinary, hematite-serpentine, ilmenite-limonite, basalt-magnetite, ilemnite, steel-scrap and steel-magnetite concrete, at photon energies from 10keV to 1GeV.

Türkmen et al. (2008b) calculated the mass attenuation coefficient of Portland cement, zeolite, blast furnace slag, silica fume for photon energies ranging from 1keV to 2MeV.

For several building materials (glass, concrete, marble, flyash, cement and lime), the same quantity has been provided by (Singh et al. 2004) over a wide energy range, from 10keV to 100GeV.

Flux and dose transmission through concrete of neutrons generated by proton induced reactions on different targets in the energy range of 25-200MeV are analyzed in (Maiti et al. 2004) and empirical formulae for their calculation are here suggested.

The attenuation of neutrons and secondary radiation through an iron shielding and composite iron/concrete shielding, for buidings housing proton accelerators, has been calculated by means of Monte Carlo simulations in (Agosteo et al. 2008).

References to Chapter 3

[1] Agosteo S., Magistris M., Mereghetti A., Silari M., Zajacova Z., Shielding data for 100-250MeV proton accelerators: attenuation of secondary radiation in thick iron and concrete/iron shields, Nuclear Instruments and Methods in Physics Research B 266, 3406-3416, 2008.

[2] American Concrete Institute, Code requirements for nuclear safety related concrete structures, ACI 49-85; cited in: Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[3] American National Standard for guidelines on the nuclear analysis and design of concrete radiation shielding for nuclear power plants, ANSI/ANS-6.4-1985, American Nuclear Society, Illinois; cited in: Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[4] Akkurt I., Basygit C., Kilincarslan S., Mavi B., Akkurt A., Radiation shielding of concetes containing different aggregates, Cement & Concrete Composites 28, 153-157, 2006.

[5] Bashter I.I., Calculation of radiation attenuation coefficients for shielding concretes, Annals of Nuclear Energy 24(17), 1389-1401, 1997.

[6] Bashter I.I., Makarious A.S., El-Sayed Abdo A., Investigation of hematite-serpentine and ilmenite-limonite concretes for reactor radiation shielding, Annals of Nuclear Energy 23(1), 65-71, 1996.

[7] Baumeister T., Bonilla C.F., Dunning J.R., Failla G., Freudenthal A.M., Havens Jr. W.W., Hoopes Jr. J.W., Wayne Huston R., Kehl G.L., Lndis J.W., Palfrey J.G., Peaslee D.C., Wu C.S., Nuclear engineering, McGraw-Hill, 1957.

[8] Boncukcuoğlu R., İçelli O., Erzeneoğlu S., Muhtar Kocakerim M., Comparison of radioactive transmission and mechanical properties of Portland cement and modified cement with trommel sieve waste, Cement and Concrete Research 35, 1082-1087, 2005.

[9] Browne R.D., Properties of concrete in reactor vessels, Paper for Conference on Prestressed Concrete Pressure Vessels, Group C, paper 13, LC.E., London, 1967.

[10] Demirboğa R., Türkmen İ., Karakoç M.B., Relationship between ultrasonic velocity and compressive strength for high-volume mineral-admixtured concrete, Cement and Concrete Research 34, 2329-2336, 2004.

[11] Demirboğa R., Türkmen İ., Karakoç M.B., Thermo-mechanical properties of concrete containing high-volume mineral admixtures, Building and Environment 42, 349-354, 2007.

[12] El-Faramawy N.A., El-Hosiny F.I., The effect of the autoclaving process and addition of silica fume on Portland cement in shielding gamma radiation, Radiation Measurements 29(6), 619-623, 1998.

[13] El-Hoisiny F.I., El-Faramawy N.A., Shielding of gamma radiation by hydrates Portland cement-lead pastes, Radiation Measurements 32, 93-99, 2000.

[14] Gill S., Structures for nuclear power, C.R. Books, London, 1964; cited in: Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[15] Halliday D.B., Heat release in concrete reactor shields, Report AERE R/R, UK Atomic Energy Authority, 1954; cited in: Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[16] Harrison J.R, Nuclear reactor shielding, Temple Press, London, 1958.

[17] (Compiled by) Hungerford H.E., from: Rockwell III T. (ed.), Reactor shielding design manual, USAEC Report TID-7004, U.S. Government Printing Office, 1956 and: Argonne National Laboratory, Reactor physics constants, USAEC Report ANL-5800, U.S. Government Printing Office, 1963; cited in: Jaeger R.G., Engineering compendium on radiation shielding, vol. 2, Springer, Berlin, 1970.

[18]Ichikawa T., Kimura T., Effect of Nuclear Radiation on Alkali-Silica Reaction of Concrete, Journal of Nuclear Science and Technology, 44(10), 1281-1284, 2007.

[19] Ichikawa T., Koizumi H., Possibility of radiation-induced degradation of concrete by alkali-silica reaction of aggregates, Journal of Nuclear Science and Technology, 39(8), 880-884, 2002.

[20] Jaeger R.G., Engineering compendium on radiation shielding, vol. 2, Springer, Berlin, 1970.

[21] Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[22] Komarovskii A.N., Design of nuclear plants, Atomizdat, Moscow 1965; cited in: Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[23] Komarovskii A.N., Shielding materials for nuclear reactors, Pergamon Press, London, 1961.

[24] Maiti M., Nandy M., Roy S.N., Sarkar P.K., Flux and dose transmission through concrete of neutrons from proton induced reactions on various target elements, Nuclear Instruments and Methods in Physics Research B 226, 585-594, 2004.

[25] Makarious A.S., Bashter I.I., El-Sayed Abdo A., Samir Abdel Azim M., Kansouh W.A., On the utilization of heavy concrete for radiation shielding, Annals of Nuclear Energy 23(3), 195-206, 1996.

[26] Mazzoleni F., Nuclear engineering (in Italian), Ulrico Hoepli, Milan, 1956.

[27]Naus D.J., Primer on durability of nuclear power plant reinforced concrete structures: a review of pertinent factors, Oak Ridge National Laboratory, U.S. Nuclear Regulatory Commission Office of Nuclear Regulatory Research Washington, DC, 2007.

[28] Price B.T., Horton C.C., Spinney K.T., Radiation shielding, Pergamon Press, London, 1957.

[29] Rockwell T., Reactor shielding design manual, McGraw-Hill, New York, 1956.

[30] Sani L., Electronuclear power stations, Part II, course lessons (in Italian), Academic Year 1972-1973, University Sapienza, Rome.

[31] Singh C., Singh T., Kumar A., Mudahar G.S., Energy and chemical composition dependence of mass attenuation coefficients of building materials, Annals of Nuclear Energy 31, 1199-1205, 2004.

[32] Topçu İ.B., Properties of heavyweight concrete produced with barite, Cement and Concrete Research 33, 815-822, 2003.

[33] Türkmen İ., Gül R., Çelik C., A Taguchi approach for investigation of some physical properties of concrete produce from mineral admixtures, Building and Environment 43, 1127-1137, 2008a.

[34] Türkmen İ., Özdemir Y., Kurudirek M., Demir F., Önder S., Calculation of radiation attenuation coefficients in Portland cements mixed with silica fume, blast furnace slag and natural zeolite, Annals of Nuclear Energy 35, 1937-1943, 2008b.

[35] Walker R., Grotenhuis M., A summary of shielding constants for concretes, USAEC Report ANL-6443, Argonne National Laboratory, 1961; cited in: Jaeger R.G., Engineering compendium on radiation shielding, vol. 2, Springer, Berlin, 1970.

Chapter 4

4.1 Radiation damage on concrete: phenomenological aspects

Radiation on shielding materials is known to affect them in their chemical structure. In crystalline materials radiation leads to the displacements of atoms from their equilibrium lattice sites, causing lattice defects, which are responsible for an increase of hardness but also embrittlement, i.e. loss of ductility, of the material (Billington 1956; Faris 1956; Konobeevsky 1956). In polymers the same result is due to the formation of additional cross-links produced by the surplus energy brought by radiation (Holmes-Siedle and Adams 2002). In geomaterials, like concrete, nuclear radiation leads to the break down of atomic bounds, which is supposed to explain the decay in the mechanical properties envisaged in exposed concrete (Price et al. 1957; Jaeger 1970).

Several authors (Hilsdorf et al. 1978; Kaplan 1989) have collected published experimental data on the effects of nuclear radiation on the properties of concrete. It stands out that up to integrated neutron fluences of the order of 10^{19}n/cm^2 the effects of the irradiation are relatively small (Elleuch et al. 1972; Granata and Montagnini 1972; Hilsdorf et al. 1978; Kaplan 1989; Naus 2007), while higher fluences may have detrimental effects on concrete strength and modulus of elasticity.

Thermal expansion coefficient, thermal conductivity and shielding properties appear to be little affected by radiation.

Radiation damage in concrete is basically explained by a volume increase of its aggregates, due to enhanced degradation reactions, like alkali-silica reactions, leading to the formation of cracks and subsequent loss in strength.

Different aggregates show different radiation resistance so that the selection of suitable aggregates appears to be the most important parameter in the definition of a long lasting concrete shielding.

4.2 Effect of neutron radiation on concrete

4.2.1 Concrete compressive strength

In Fig.4.1a the compressive strength of concrete samples f_{cu} from various test series as a fraction of the compressive strength of companion specimens f_{cuo}, which are neither irradiated nor temperature exposed is reported, while in Fig.4.1b the same strength values are presented but here the concrete compressive strength is related to the strength of companion specimens, f_{cuT}, which are not irradiated but temperature exposed, in order to uncouple the effect due to radiation absorption from high temperature on the samples (in many cases the samples are subject to thermal loads, in conjunction with radiation exposure, to simulate the condition of concrete in reactor vessels).

Looking at the enveloping curves of the published experimental results, some concrete mixtures are envisaged to resist to neutron radiation above $5x10^{19}n/cm^2$, while others exhibit a strength loss at a considerably smaller radiation dose. However, as an average, a neutron fluence of the order of $10^{19}n/cm^2$ is meant to mark the beginning of a decrease of the compressive strength of concrete.

A comparison of the two plots indicates that the observed strength loss is primarily due to neutron radiation though some detrimental effects seem to be due to temperature.

The experimental data vary over a certain range for a given neutron fluence: for a neutron fluence of $5x10^{19}n/cm^2$ the strength ratios range from 0.72 to 1.05 and from 0.65 to 1.05 for f_{cu}/f_{cuo} and f_{cu}/f_{cuT} , respectively. Fast neutrons seem to induce the greater decrease.

The scattering emerging from the comparison of different test series is explained by the large variety of test conditions: concrete making materials, mix proportions of mortars, specimens size, cooling and drying conditions, impinging of fast or slow neutrons, test procedures, any simultaneous temperature exposure of specimens.

As a guide-line, filled symbols apply for radiation on samples by thermal neutrons, empty symbols for radiation by fast neutrons; not for each author a distinction was possible, because of lack of information.

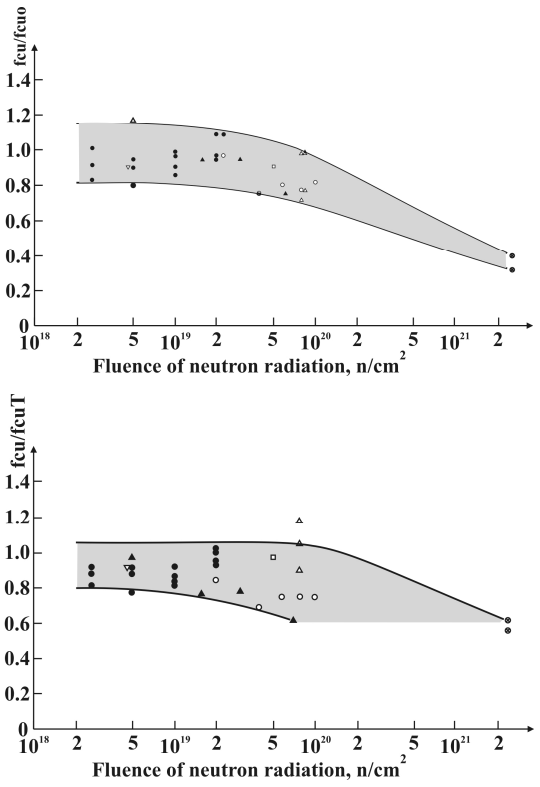

Fig.4.1 – (**a**) Compressive strength of concrete exposed to neutron radiation f_{cu} related to strength of untreated concrete f_{cuo}; (**b**) Compressive strength of concrete exposed to neutron radiation f_{cu} related to strength of temperature exposed concrete f_{cuT} (Hilsdorf et al. 1978).

There is also evidence that concrete hardens under irradiation (Granata and Montagnini 1972; Nawakowski 1972): the interference of penetrating ionizing radiation with the process of setting of cement paste has been found in the form of increased compressive and tensile strength, especially at the very beginning of the hardening period.

4.2.2 Concrete tensile strength

The effect of neutron radiation on the tensile strength f_{ru} of concrete samples is shown in Fig.4.2a and Fig.4.2b. The first gives the tensile strength of concrete samples after neutron radiation as a fraction of the tensile strength of companion specimens f_{ruo}, which are neither

irradiated nor temperature exposed, whereas in 2b the tensile strength is related to the strength of non irradiated but temperature exposed specimens, f_{ruT}, as similary done for compressive strength.

According to Fig.4.2a, a neutron fluence of more than 10^{19}n/cm^2 may lead to a significant decrease of concrete tensile strength (reductions greater than that for compressive strength, at the same neutron fluence); comparing the two figures, it stands out that temperature exposure is responsible for the strength loss but neutron radiation, rather, has caused the most considerable part of the observed strength reduction.

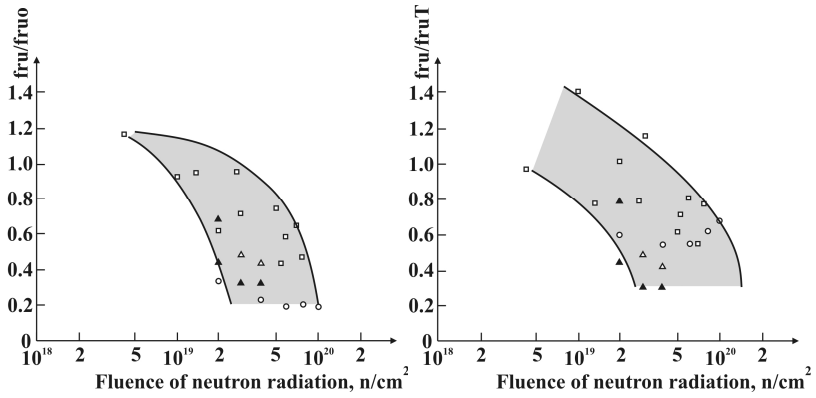

Fig.4.2 – (**a**) Tensile strength of concrete exposed to neutron radiation fru related to strength of untreated concrete fruo; (**b**) Tensile strength of concrete exposed to neutron radiation fru related to strength of temperature exposed concrete fruT (Hilsdorf et al. 1978).
Also for the tensile strength the values vary over a wide range: for a neutron fluence of 5x10^{19}n/cm^2 the observed strength ratios range between 0.2 and 0.82 and between 0.33 and 0.98 for f_{ru}/f_{ruo} and f_{ru}/f_{ruT} , respectively.

4.2.3 Modulus of elasticity

Fig.4.3 shows the effect of neutron radiation on the modulus of elasticity of concrete; the modulus of irradiated concrete E_c is given as a fraction of the modulus of companion specimens E_{co} neither irradiated nor temperature exposed.

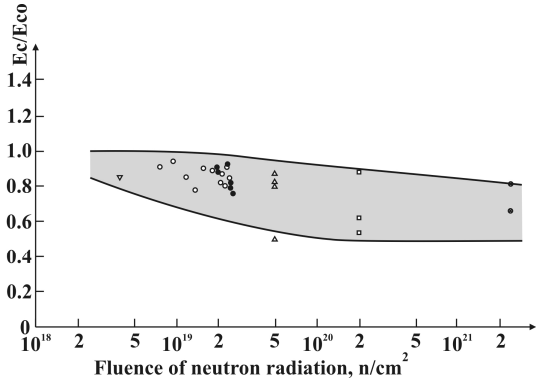

Fig.4.3 – Modulus of elasticity of concrete after neutron radiation E_c related to modulus of elasticity of untreated concrete E_{co} (Hilsdorf et al. 1978).

Up to a neutron fluence less than 10^{19}n/cm^2, again, a slight decrease of the modulus of elasticity, compared to the modulus of untreated companion specimens is envisaged. At increasing neutron fluence, the modulus of concrete decreases more significantly.

Distinguishing between fast or slow neutrons, Gray (1972) found that for fast neutron fluences between 7x10^{18} and 3x10^{19}n/cm^2 the modulus of irradiated concrete was between 10 and 20% less than that of unirradiated unheated concrete. Alexander (1963) reported similar reductions in values of E for slow neutron fluences of about 2x10^{19}n/cm^2.

For the Young modulus a comparison with data affected by temperature, only, was not provided, however it seems reasonable, in analogy with the previous mechanical properties, that neutron radiation would be envisaged as the predominant source of damage.

4.2.4 Thermal conductivity and thermal expansion

In the plot below after (Hilsdorf et al. 1978) results are shown of thermal conductivity K_c of irradiated samples, as a fraction of the thermal conductivity of non irradiated companion specimens K_{co}, in function of neutron fluence. Tests relative to flint aggregate (circles), limestone aggregate (squares) and light weight aggregate (rhombi) are taken into consideration.

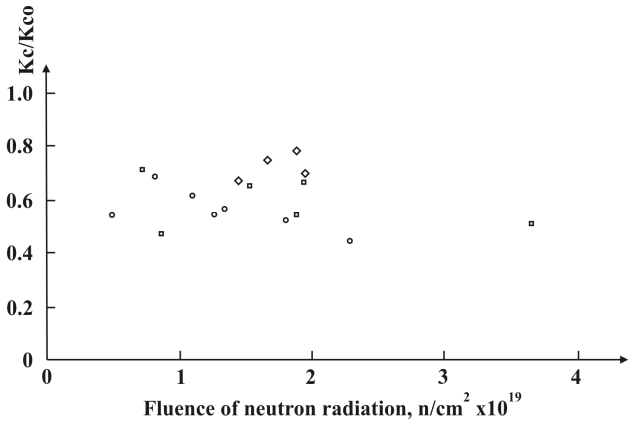

Fig.4.4 – Thermal conductivity of concrete after neutron radiation K_c related to thermal conductivity of untreated concrete K_{co} (Hilsdorf et al. 1978).

Thermal conductivity is shown to decrease between 20 and 50% than that of companion unheated unirradiated specimens, as a consequence of the exposure.

As for the coefficient of thermal expansion, Hilsdorf et al. (1978) and Granata and Montagnini (1972) indicate that for neutron fluences less than $5x10^{19}n/cm^2$ there is no significant difference between the coefficient of thermal expansion of neutron-irradiated concrete samples and the coefficient of unirradiated samples which have been exposed to high temperature.

4.3 Parameters affecting the resistance of concrete against nuclear radiation

It has been suggested that the energy spectrum pays an important role in defining the consequences of radiation on the properties of the material, in particular, as a guide-line, fast neutrons with an energy of about 1MeV are generally required to bring about a sufficiently large number of lattice defects in crystalline solids to result in changes in physical and mechanical properties (Kaplan 1989).

Figs.4.1, 4.2, 4.3, previously analyzed, report the experimental data distinguishing between irradiation from slow (filled symbols) or fast (empty symbols) neutrons (sometimes no identification was possible). It seems here not possible to define whether fast or thermal neutrons are more detrimental: in particular, though it is generally expected that fast neutron

radiation lead to a more pronounced radiation damage, this is not univocally confirmed by the plots.

Many authors (Campbell-Allen et al. 1966; Gray 1972; Hilsdorf et al. 1978; Kaplan 1989; Naus 2007) state that different types of aggregates lead to concrete with different resistance against neutron radiation. Gray (1972) reports that different types of aggregate lead to different volume changes of concrete, depending on the exposure duration.

In Fig.4.5 the relation between volume change and neutron fluences for concrete made of limestone aggregate and of flint aggregate is shown:

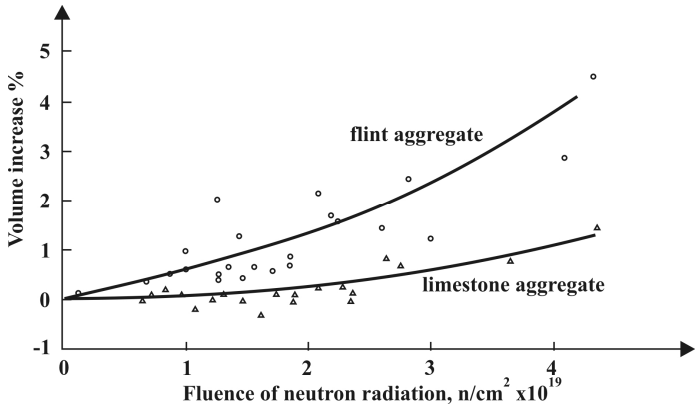

Fig.4.5 – Volume change of concrete specimens exposed to fast neutrons (Hilsdorf et al. 1978).

Concrete made with flint aggregate shows considerably larger volume changes than concrete made with limestone aggregates; the phenomenon has been explained in (Kaplan 1989; Naus 2007) in terms of the chemical bounding characterizing the compounds in the aggregates: quartz aggregates are charazterized by covalent bondings, which are weaker than ionic bondings present in limestone aggregates, resulting in being more affected by radiation than calcareous aggregates.

With the volume growth, tensile strength, too, varies (on average decaying of the 62 and 47% respectively for flint and limestone aggregates in the range of $2x10^{19}$-$4x10^{19}n/cm^2$) (Kaplan 1989).

According to (Granata and Montagnini 1972) a neutron fluence of less than $10^{19}n/cm^2$ does not lead to a volume increase of the irradiated samples, rather in this range the volume change of irradiated samples is the expected shrinkage due to temperature exposure of the specimens.

Further, there is evidence (Ichikawa and Kimura 2007; Ichikawa and Koizumi 2002; Naus 2007) that nuclear radiation significantly increases the reactivity of silica-rich aggregates to alkali (see Fig.4.6); the decrease of the resistance to nuclear radiation with increasing the content of SiO_2 in aggregates strongly indicates that the deterioration is due to the acceleration of this kind of reaction in concrete and it explains, at least in part, the degradation of the mechanical properties of concrete shields.

An alkali-silica reaction is understood to proceeds as follows: OH^- ions of the alkaline solution in the micropores of concrete react with SiO_2 in aggregates to perform the scission of the Si–O bonding and the subsequent expansion of the aggregates by hydration of SiO_2. The consumption of OH^- ions due to hydrolysis leads to the dissolution of Ca^{2+} ions into the solution. The Ca^{2+} ions then react with hydrated SiO_2 gels to generate calcium silicate. Rigid calcium silicate shells are therefore formed on the surfaces of the aggregates by succeeding reactions with OH^- and Ca^{2+} ions. The alkaline solution is possible to penetrate into the aggregates through the calcium silicate shells and to dissolve SiO_2. Since the rigid shells prevent the deformation of the aggregates, the expansion pressure generated by the penetration of the solution is accumulated in the aggregates, thus leading to cracks and their final expansion.

The alkali-silica reaction of the aggregates may be accelerated both by lattice defects in SiO_2 minerals from fast neutrons irradiation or the pre-existence of cracks in the aggregates.

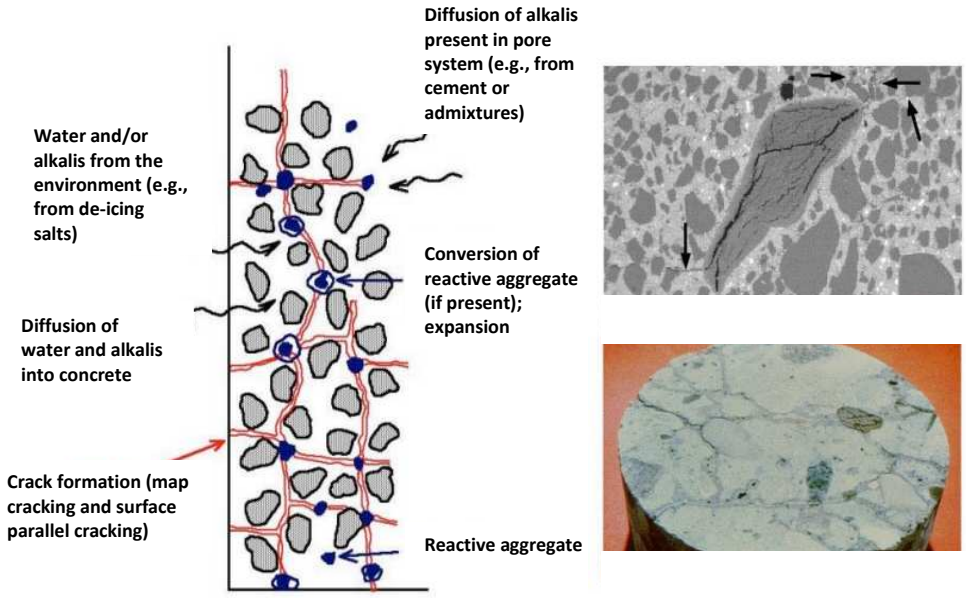

Fig.4.6 – Alkali-silica reaction on concrete.

4.3.1 Effect of gamma-radiation on concrete strength

During the tests in which the effect of neutron radiation was studied the concrete samples were also exposed to secondary gamma radiation, produced by nuclear reactions.

Not many tests have been reported on the effects of gamma radiation alone without the simultaneous exposure to neutron radiation. Alexander (1963) reports that for gamma radiation doses of about 10^{10}rad there is no reduction in the compressive strength of concrete, when compared with the strength of companion specimens which have neither been irradiated nor heated; there is, however, evidence of reductions of between 25 and 60% in compressive strength for doses exceeding about 10^{11}rad (Sommers 1969).

In this last test campaign the specimens were immersed in demineralized water in order to shield them against neutrons and study the effect of gamma-radiation alone, but demineralized water was found to contribute in deteriorating concrete and after several years of exposure the surface of the irradiated samples were partially destroyed.

Experimental data (Vodák et al. 2005) on concrete under gamma irradiation also suggest that the interaction with the shielding material leads to lowering both its strength and its porosity. The mechanism is explained to happen as a series of chemical reactions within the material, starting from the radiolysis of water and ending with the formation of calcite ($CaCO_3$); crystallites of calcite grow into pores, decreasing the size of pores and destroying tobermorite gel, a calcium silicate hydrate mineral responsible for concrete strength, by crystallization pressure.

References to Chapter 4

[1] Alexander S.C., Effects of irradiation on concrete: final results, Atomic Energy Research Establishment, Harwell, United Kingdom Atomic Energy Authority, 1963.

[2] Billington D.S., Radiation damage in reactor materials, in: Proceedings of the international conference on the peaceful uses of atomic energy, Geneva, 8-20 August, 1955, Nuclear chemistry and effects of irradiation, vol. 7, 421-432, United Nations, New York, 1956.

[3] Campbell-Allen D., Low E.W.E., Roper H., An investigation on the effect of elevated temperatures on concrete for reactor vessels, in: Nuclear Structural Engineering, vol. 2, 382-388, North-Holland Publishing Comp., Amsterdam, 1966.

[4] Elleuch M.F., Dubois F., Rappeneau J., Effects of neutron radiation on special concretes and their components. Amer Concr Inst Special Publication SP-34: Concrete for nuclear reactors, Paper SP34-51:1071-1108, 1972.

[5] F.E. Faris, The effect of irradiation on structural materials, in: Proceedings of the international conference on the peaceful uses of atomic energy, Geneva, 8-20 August, 1955, Nuclear chemistry and effects of irradiation, vol. 7, 484-489, United Nations, New York, 1956.

[6] Granata S., Montagnini A., Studies on behavior of concretes under irradiation, in: Concrete for nuclear reactors (Special Publication SP-34, vol. 2) American Concrete Institute, 1163-1172, 1972.

[7] Gray B.S., The effect of reactor radiation on cements and concrete, in: Conference on Prestressed Concrete Pressure Vessels, Commission of European Communities, Luxembourg, 17-39, 1972.

[8] Hilsdorf H.K., Kropp J. and Koch H.J., The effects of nuclear radiation on the mechanical properties of concrete, in: Douglas McHenry International Symposium on Concrete and Concrete Structures (ACI SP 55-10, American Concrete Institute, Detroit, Michigan, 1978), Mexico City, October 1976, 223-251, 1978.

[9] Holmes-Siedle A., Adams L., Handbook of radiation effects, Oxford University Press, 2002.

[10] Ichikawa T., Kimura T., Effect of Nuclear Radiation on Alkali-Silica Reaction of Concrete, Journal of Nuclear Science and Technology, 44(10), 1281-1284, 2007.

[11] Ichikawa T., Koizumi H., Possibility of radiation-induced degradation of concrete by alkali-silica reaction of aggregates, Journal of Nuclear Science and Technology, 39(8), 880-884, 2002.

[12] Jaeger R.G., Engineering compendium on radiation shielding, vol. 2, Springer, Berlin, 1970.

[13] Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[14] Konobeevsky S.T., Pravdyuk N.F., Kutaitsev V.I., The effect of irradiation on the structure and properties of structural materials, in: Proceedings of the international conference on the peaceful uses of atomic energy, Geneva, 8-20 August, 1955, Nuclear chemistry and effects of irradiation, vol. 7, 479-483, United Nations, New York, 1956.

[15] Naus D.J., Primer on durability of nuclear power plant reinforced concrete structures: a review of pertinent factors, Oak Ridge National Laboratory, U.S. Nuclear Regulatory Commission Office of Nuclear Regulatory Research Washington, DC, 2007.

[16] Nawakowski B., Influence of penetrating ionizing radiation on the curing of grouts and cement mortars, Building Science, 7, 271-275, 1972.

[17] Price B.T., Horton C.C., Spinney K.T., Radiation shielding, Pergamon Press, London, 1957.

[18] Sommers J.F., Gamma radiation damage of structural concrete immersed in water, in: Health physics (vol. 16), 503-508, 1969.

[19] Vodák F., Trtík K., Sopko V., Kapičková O., Demo P., Effect of γ-irradiation on strength of concrete for nuclear-safety structures, Cement and Concrete Research, 35, 1447-1451, 2005.

Chapter 5

5.1 Particle transport simulation

The description of all the possible interactions between a radiation particle and the absorber medium is the topic of the so-called “transport theory”, a special branch of statistical mechanics.

Accurate calculations of particle transport are needed, nowadays more than in the past, due to many reasons:

a) The necessity to develop theoretical models of radiation transport also for particle energies above 20MeV: attenuation coefficients, such as neutron cross-sections, need to be evaluated for such energies, while complete libraries are available for a representative set of nuclides up to 20MeV (Vaz 2009). Neutrons are known to be easily transported over many energy decades, from the hundreds of MeV down to the meV range, therefore the knowledge of the cross-section over the whole energy range concerned is required.

   Many new facilities, working at high energy, are going to be built to perform experiments, in this sense, to calibrate and validate predictive theoretical models; this is one of the aims of SPES Project, also.

b) The need to accurately describe the proton- and neutron-induced interaction mechanisms: the fact arises due to the spread of a number of new applications, in the last fifteen years, involved in the production of intense neutron fluxes by protons impinging on a thick target of a heavy element (mercury, lead, uranium, …) and producing a cascade of nuclear reactions, known as “spallation reactions”, of the

protons themselves, neutrons, nuclear fragments (alpha-particles, tritium, deuterium) and secondary gamma-rays.

c) The assistance in the design of accelerator driven systems (ADS), which have emerged in recent years as a solution to the problem of the transmutation, as a possible mechanism for reducing the volume and hazard of the radioactive waste (spent fuel) in nuclear power reactors: ADS consist in a reactor coupled to a high-intensity proton accelerator impinging on a high-density target. The spallation reactions taking place in the target result in high neutron fluxes here and in the surrounding core, which require accurate evaluation, for shielding design purposes, as well as for the radiological protection assessment (thermo-mechanical analyses, study of the activation of components and materials, e.g.).

d) Similarly, the assistance in the design of other nuclear facilities, such as new generation reactors (generation IV): the concept with these next generation reactors is to use different coolants (sodium-, gas- or liquid-metal cooling system) and to consider recycling of the fuel, in order to both reach higher power (improved efficiency) and minimize the production of waste.

   For these systems a radiation damage assessment of the materials subject to the intense neutron fluxes (fuel claddings, vessel, cooling system) and the radioprotection studies for fulfilling safety requirements are mandatory (Workshop AIM-IENI 2011).

e) The scientific knowledge of the biological effects of radiation, to study particle transport for biomedical purposes. Monte Carlo simulations of neutron transport in human tissues has been conducted in recent years, due to the difficulty in carrying out experiments, in order to determine the tissue-specific weighting factors at different energies, which are useful for the assessment of the effective dose of exposed subjects.

   Also the effective dose rate assessment for aircrew and crews of space flights, which are understood to be seriously exposed to cosmic radiations, is expected to be refined in the future.

Two approaches are possible to define the radiation flux density throughout a certain domain: the deterministic approach given by the solution of the Boltzmann transport equation and the statistical approach given by Monte Carlo simulations.

5.1.1 Deterministic transport theory

The Boltzmann transport equation is a linear integro-differential and time-dependent equation for the quantity $\phi(\mathbf{r}, E, \mathbf{\Omega}, t)$, i.e. a particle flux density, which, in general, depends on space **r**, energy E, angular direction **Ω** and time t.

Therefore $\phi(\mathbf{r}, E, \mathbf{\Omega}, t)$ physically is the total path length travelled in a unit time by particles, in a unit volume, at **r**, that have energies in unit energy E and that are travelling in a unit solid angle about the direction **Ω**, at time t, so that, if v is the particle speed (for photons v is the speed of light c, while, for neutrons with mass m, v equals $(2E/m)^{1/2}$) and $n(\mathbf{r}, E, \mathbf{\Omega}, t)$ is the expected number of particles per unit volume at **r**, with energies in unit energy E and moving in a unit solid angle dΩ about **Ω**, at time t (the particle number density distribution), then:

$$\phi(\mathbf{r}, E, \mathbf{\Omega}, t) = v\, n(\mathbf{r}, E, \mathbf{\Omega}, t) \tag{37}$$

The equation for $\phi(\mathbf{r}, E, \mathbf{\Omega}, t)$ is obtained considering the particle balance in an elemental volume V within a closed surface S, for those particles with energies in dE about E and moving in dΩ about the angular direction **Ω**.

The time variation of the particle density ∂p/∂t inside the elemental volume and interval dE and angular cone dΩ, must equal the net flow rate in which particles are lost (either by leaving or entering the volume V or by changing their energy or direction of travel after interactions; they are both "destruction" components, that is why they are preceded by a minus sign) plus the rate secondary or source particles are introduced in V with the same energy and direction of motion (both are "production" terms, therefore they are preceded by a plus sign):

$$\begin{aligned}\int_V \frac{1}{v}\frac{\partial}{\partial t}\phi(\mathbf{r}, E, \mathbf{\Omega}, t)\, dV = &-\int_S \mathbf{n}\cdot\mathbf{\Omega}\ \phi(\mathbf{r}, E, \mathbf{\Omega}, t)\, dE\, d\Omega\, dS - \int_V \Sigma(\mathbf{r}, E)\ \phi(\mathbf{r}, E, \mathbf{\Omega}, t)\, dE\, d\Omega\, dV + \\ &+ \int_V \int_0^\infty \int_{4\pi} \Sigma_s(\mathbf{r}, E' \to E, \mathbf{\Omega'} \to \mathbf{\Omega})\ \phi(\mathbf{r}, E', \mathbf{\Omega'}, t)\, d\Omega'\, dE'\, dV + \int_V S(\mathbf{r}, E, \mathbf{\Omega}, t)\, dV\end{aligned} \tag{38}$$

where **n** is the normal of the surface S in any point, Σ_R is the total interaction coefficient, i.e. the macroscopic removal cross-section Σ_R, if the particles are neutrons, the linear attenuation coefficient μ, if photons, and $\Sigma_s(\mathbf{r}, E' \to E, \mathbf{\Omega'} \to \mathbf{\Omega})$ is the scattering cross-section for the analyzed particles, so that $\Sigma_s(\mathbf{r}, E' \to E, \mathbf{\Omega'} \to \mathbf{\Omega})\ dE\ d\Omega$ is the probable number of secondary particles at point **r**, with energies in dE about E, in direction dΩ about **Ω**, produced by an incident particle of energy E' traveling in direction **Ω'** per unit path length of the incident particle.

The integrals above, therefore, express, in function of the differential flux density of the particles $\phi(\mathbf{r}, E, \mathbf{\Omega}, t)$ (in the order in which they are written in Eq. (38)):

a) the variation in time of the particle density $\partial p/\partial t$ inside the elemental volume, energy interval dE and angular cone $d\Omega$;
b) the net number of particles flowing out of V across S in a unit time;
c) the total number of particles undergoing interactions in a unit time, i.e. being attenuated for scattering or absorption processes;
d) the number of secondary particles in dE about E and $d\Omega$ about $\mathbf{\Omega}$, produced by any particle interaction with matter, in V, in a unit time. If the particles are neutrons, this "production" term takes into account fission reactions and any other kind of nuclear interaction, provided it is known the specific cross-section for each of them.
e) the external, non radiation-induced source of particles with energy in dE about E and moving in $d\Omega$ about $\mathbf{\Omega}$, introduced in unit time in V.

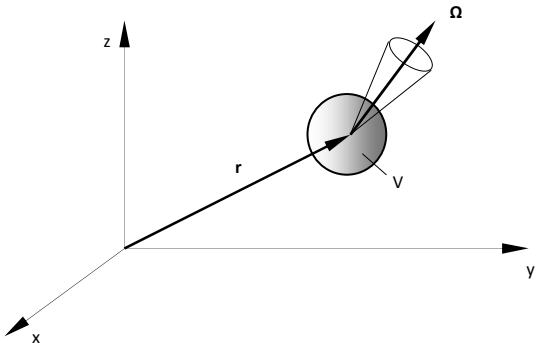

Fig.5.1 – Reference system for the Boltzmann transport equation, establishing the particle balance inside the elemental volume V, at position $\mathbf{r}$ and within the angular cone $d\Omega$ around direction $\mathbf{\Omega}$.

After applying the Gauss-Green integral theorem (or divergence theorem) to the second surface integral, and considering that: i) the divergence operator affects only the spatial variable $\mathbf{r}$, not E, t or $\mathbf{\Omega}$, and ii) the relationship holds for an arbitrary volume V, then the balance can be rewritten in its most common form:

$$\frac{1}{v}\frac{\partial}{\partial t}\phi(\mathbf{r},E,\mathbf{\Omega},t) = -\mathbf{\Omega}\cdot\nabla\phi(\mathbf{r},E,\mathbf{\Omega},t) - \Sigma(\mathbf{r},E)\,\phi(\mathbf{r},E,\mathbf{\Omega},t) + \\ + \int_0^\infty \int_{4\pi} \Sigma_s(\mathbf{r},E'\to E,\mathbf{\Omega'}\to\mathbf{\Omega})\,\phi(\mathbf{r},E',\mathbf{\Omega'},t)\,d\Omega'\,dE' + S(\mathbf{r},E,\mathbf{\Omega},t) \tag{39}$$

For shielding applications it is sufficient to solve the steady-state problem, i.e. the only spatial, energy and directional flux density distribution $\phi(\mathbf{r}, E, \mathbf{\Omega})$, in fact the time dependence of the

radiation field is usually assumed to be that of the primary radiation source and the time delays caused by the finite speed of the particles can be ignored in shielding application (Shultis and Faw 1996).

Therefore the overall balance in steady-state conditions is the following:

$$\mathbf{\Omega} \cdot \nabla\phi(\mathbf{r},E,\mathbf{\Omega}) + \Sigma(\mathbf{r},E)\ \phi(\mathbf{r},E,\mathbf{\Omega}) = \int_0^{\infty} \int_{4\pi} \Sigma_S(\mathbf{r},E' \rightarrow E, \mathbf{\Omega}' \rightarrow \mathbf{\Omega})\ \phi(\mathbf{r},E',\mathbf{\Omega}')\ d\Omega'\ dE' + S(\mathbf{r},E,\mathbf{\Omega}) \quad (40)$$

which is known as the “Boltzmann transport equation”, or “master equation”, named after Ludwig Boltzmann, who first derived it over a century ago.

It is linear because neutron-neutron interactions (or photon-photon) are neglected.

Exact solutions to this analytical expression is possible only in a few cases, and, what is more important, in realistic shielding problems it is not, consequently much effort has been paid in the development of approximations to the transport equation, both numerical and analytical. In this second case the diffusion theory and the two-group theory, previously illustrated for neutrons, are worth mentioning.

Numerically speaking, the solution of the Boltzmann equation requires:

a) energy discretization: the multigroup formalism is useful in this sense, in that it consists in dividing the whole energy range of the particles in a finite number of energy intervals and considering that for each group the cross-sections are constant (average values over the flux).
b) Angular discretization: the most widely used method to do that is the so-called “S_N discrete ordinate method”, which consists in a selection at the point (**r**, E) of a finite set of angular directions, with assigned associated weights and fluxes, which the total flux is a linear combination of.
c) Spatial discretization: the discretization in space implies the solution to be calculated at discrete points of a mesh superimposed to the real geometry of the problem, according to the Finite Element Method (FEM).

5.1.2 Monte Carlo technique

The concept of the Monte Carlo technique in solving the particle transport problem is to use a large number of particle tracks or histories, random in nature, to estimate an average particle behaviour.

Provided that we have n variables x, y, z, …, distributed according to the distributions (normalized) f'(x, y, z, …), g'(x, y, z, …), h'(x, y, z, …), where:

$$f'(x) = \frac{f(x)}{\int_{x\,min}^{x\,max} f(x)\,dx} \tag{41}$$

then the mean or average of a function of those variables A(x, y, z, …), over an n-dimensional domain D, is:

$$\overline{A} = \int_x \int_y \int_z \ldots \int A(x, y, z, \ldots)\, f'(x, y, z, \ldots)\, g'(x, y, z, \ldots)\, h'(x, y, z, \ldots)\, dx\, dy\, dz \ldots . \tag{42}$$

To overcome integration, it is possible to sample N values of A, by sampling N sets of variables x_i, y_i, z_i, … with probability $f' \cdot g' \cdot h' \ldots$ and divide the sum by N:

$$S_N = \frac{\sum_{i=1}^{N} A(x_i, y_i, z_i, \ldots)}{N} \tag{43}$$

The central limit theorem guarantees that, for large values of N, the normalized sum of N independent and identically distributed random variables tends to a Gaussian distribution with mean $\overline{A}$ and variance σ_A^2/N:

$$\lim_{N\to\infty} S_N = \lim_{N\to\infty} \frac{\sum_{i=1}^{N} A(x_i, y_i, z_i, \ldots)}{N} = \overline{A}\,. \tag{44}$$

Monte Carlo technique results in both:

a) an integration method that allows to solve multi-dimensional integrals, describing statistically an average function $\overline{A}$, by sampling many values of A according to the probability distributions of the random variables that A depends on.

b) A computer simulation of a physical process, in that the function A is the particle density and the samples are the single experiences of each primary particle from the time it leaves its source until it is absorbed or passes out of the domain, i.e. particle "histories", which are generated by simulating the random nature of the particle interaction with the medium. The transport of a single particle, i.e. the particle tracks, in fact, can be described as a sequence of collisions occurring at discrete points in space and subsequent displacement of the particle from one collision point to the next; the choice of an interaction type at each collision point, the choice of a new energy and new direction, if scattering occurs, or the possible production of secondary

particles, are all stochastic variables of the process, with their probability distribution, which is randomly selected at each step of the particle path.

The main assumptions for a particle transport Monte Carlo analysis are (Fluka Workshop 2010):

a) the absorber media are static, homogeneous, isotropic and amorphous;
b) the particle transport is a Markovian process, i.e. the fate of a particle depends only upon its actual properties, not on previous events;
c) particles do not interact with each other but only with individual atoms, nuclei or molecules;

The accuracy and reliability of a Monte Carlo estimator depends:

a) on the models or data on which the probability density functions of the statistical variables are based;
b) on the number N of samples or histories; the degree of accuracy goes with $N^{-1/2}$.

A way to reduce the error of the statistical estimate of the solution, i.e. the variance of the sample mean, is to increase the number of histories, with subsequent increase of the computer time required, or to use special techniques that allow both to reduce the variance σ^2 at the expense of the CPU time t, or viceversa or, sometimes, to reduce the product σ^2 x t (that is to reduce σ^2 at a faster rate than t increases or viceversa). The methods to do that are known as the "biasing" technique.

Biased Monte Carlo analyses are also referred to as "non-analog" analyses, to distinguish them from "analog" ones. An analog Monte Carlo implies that each particle has the same importance with respect to the others, while biased Monte Carlo applies weights to the particles, that are importance values affecting the probability of survival of the particle, during its path to reach a certain point.

An analog analysis can predict all statistical moments of any order, in particular, not only the mean of the contributions converges to the mean of the real distribution, but also the variance and all moments of higher order converge as well, and fluctuation and correlations are preserved. While a non-analog analysis can predict the average quantity, minimizes the second moment, but can not predict higher moments and can not reproduce the real correlations and fluctuations.

Moreover, the total score of an analog Monte Carlo transport calculations, i.e. the statistical estimate of the flow or fluence or other related quantity results (e.g. absorbed dose, deposited

energy, …), is the sum of the contributions from each history, included the zero contributions of those histories that do not stop within the domain. Due to the use of weights, the final scoring in a non-analog Monte Carlo simulation is not just the sum of the particles' scores but becomes the sum of the particles' weighted scores.

5.1.2.1 Fluka

The Fluka code is a statistical code officially developed from 2003 by a joint-venture between CERN and INFN of Milan, for calculations of particle transport and their interactions with matter.

The history of Fluka spans more than 40 years: Fluka was born in 1962, from an idea by Johannes Ranft (Leipzig University) and Hans Geibel (CERN) aimed at developing Monte Carlo for high-energy proton beams. The name "Fluka" (from "FLUktuierende Kaskade") has been given to the code in 1970 after the study of event-by-event fluctuations in a calorimeter and it has been maintained for historical reasons.

However, it is possible to distinguish between three generations of Fluka (Fassò et al. 2005): Fluka of the '70s (whose main authors have been J. Ranft and J. Routti (Helsinki University)), Fluka of the '80s (P. Aarnio (Helsinki University), A. Fassò (CERN), H.-J. Mőhring (Leipzig University), J. Ranft, G.R. Stevenson (CERN)) and Fluka of today (A. Fassò, A. Ferrari (INFN Milan), J. Ranft and P.R. Sala (INFN Milan)).

From '90s the code has been renewed in many aspects (introduction of new physical models, improvements in: code structure, tracking strategy and scoring, implementation of more biasing methods). If up to Fluka86-87 the code was essentially specialized in studying high-energy proton accelerators, the present version allows for dealing with many radiation sources. In particular the implementation of a model for low-energy neutrons (<20MeV) makes it possible to study electronics and the introduction of a model for heavy ion interactions to exploit Fluka for hadrontherapy applications.

Nowadays Fluka is open to a wide range of applications:

a) cosmic ray physics
b) neutrino physics
c) accelerator design
d) particle physics: calorimetry, tracking and detector simulation
e) ADS design and waste transmutation

f) shielding design
g) dosimetry and radioprotection
h) hadrontherapy
i) neutronics.

Fluka is based on microscopical models, when possible, and consistency among all the reaction steps or reaction types, as well as conservation laws at each step, are ensured. Every model is validated against experimental data.

Fluka can model the interaction and propagation in matter of many particles (Fassò et al. 2005):

a) photons and electrons from 1keV to 10000TeV
b) neutrinos
c) muons
d) hadrons of energies up to 20TeV
e) neutrons (also low-energy from 0 to 20MeV)
f) heavy ions
g) polarized photons
h) optical photons

also studying the time evolution of phenomena. Some of the physical issues handled by Fluka are:

a) hadron-hadron and hadron-nucleus interactions from 0 to 10000TeV
b) elastic scattering
c) nucleus-nucleus interactions from 100MeV/n to 10000TeV/n
d) transport of charged hadrons and muons
e) transport in magnetic field.

Fluka is compiled in Fortran 77 and works in Linux platforms. The whole program, including numerical constants, is coded in double precision, except for the low-energy neutron cross-sections, which are stored in single precision. In the code the use of dynamical memory allocation is implemented, mostly.

It is endowed of the user interface Flair, useful at several purposes: creating the input file (geometry, materials), linking source routines and output files, running the code, merging data, post-processing.

In general, for a problem to be fully determined, the following steps are mandatory:

a) definition of the geometry of the problem;
b) assignment of a unique material to all regions of the domain;
c) request of the desired scorings;
d) setting of the parameters, accuracy, conditions on how holding calculations.

Especially the first point is quite onerous: the geometry of the problem must be created by means of elemental volumes and Boolean operations between them, through the use of the Fluka Combinatorial Geometry (CG) package. The visualization and debugging step-by-step are mandatory in order to verify the created geometry, because particles' tracks and destiny depends on the crossed regions, which are volumes with assigned material, and need to be uniquely defined.

The code provides itself the chemical composition of the most common elements and compounds (air, water, concrete, biological tissues, …) in stored libraries, but new materials can be defined.

Some possible scorings with Fluka are the following:

a) particle flux, fluence, current, track length
b) particle/energy spectrum
c) energy deposition
d) radioactive decay of residual nuclei, referred to user-defined irradiation and cooling time profiles
e) absorbed dose.

References to Chapter 5

[1] Fassò A., Ferrari A., Ranft J., Sala P.R., FLUKA: a multi-particle transport code. CERN-2005-10, INFN/TC_05/11, SLAC-R-773, 2005.

[2] Proceedings of the 1st Fluka Advanced Course/User Workshop, Ericeira, Portugal, 4-8 October, 2010.

[3] Proceedings of the Workshop: IV generation nuclear reactors: which issues for metals? (in Italian) organized by AIM (Italian Association of Metallurgy) with Ansaldo Nucleare and IENI (National Council for Research, Institute for Energy and Interphases), Geneva, 17 February, 2011.

[4] P. Vaz, Neutron transport simulation (selected topics), Radiation Physics and Chemistry 78, 829-842, 2009.

[5] Shultis J.K., Faw R.E., Radiation shielding, Prentice Hall PTR, 1996.

Chapter 6

6.1 The mathematical model

The FEM model adopted to perform radiation-induced damage analyses is the 3D research code NEWCON3D (Lewis and Schrefler 1987; Schrefler et al. 1989; Baggio et al. 1995; Majorana et al. 1995-1997-1998-2004; Salomoni et al. 2007-2008-2009; Mazars 1984-1989; Mazars and Pijaudier-Cabot 1989-1996; Kachanov 1958; Gerard et al. 1998; Nechnech et al. 2001). Concrete is here treated as a multiphase system where the micropores of the skeleton are partially filled with liquid water, both in the form of bound or absorbed water and free or capillary water, and partially filled with a gas mixture composed of dry air (non-condensable constituent) and water vapour (condensable), supposed to behave like an ideal gas (Schrefler et al. 1989; Baggio et al. 1995; Majorana et al. 1998-2004).

When higher than standard temperatures are taken into account several phenomena are considered within the model, dealing with concrete as a porous medium: heat conduction, vapour diffusion and liquid water flow in the voids.

As regards the mechanical field, the model couples shrinkage, creep and damage; the constitutive law of the material will be developed in §6.1.1 starting from a thermodynamic approach.

6.1.1 Thermodynamic formulation

In order to define the thermodynamic framework for an elastic multiphase medium system, accounting for damage and creep, as concrete is modeled into NEWCON3D, we consider a solid body Ω occupying a material volume V bounded by a surface Σ, whose outer normal is **n**.

Let the solid, of mass density ρ, absorb vapor through its boundary and let m denote the vapor/fluid mass per unit volume of the solid (i.e. moisture concentration). Also, let **x** be the position of a solid mass particle in the deformed configuration that corresponds to the place **X** in the undeformed state, and let **f** and **q** denote fluxes of mass vapor and heat, respectively, and **v** the velocity of the solid particles.

In addition, let e and s be the internal energy and entropy densities of the solid/vapor mixture per unit solid mass, and let σ_{ij} and T denote the components of the Cauchy stress due to mechanically applied loads and temperature, respectively.

We consider the solid/fluid mixture as an open system, with the fluid phase at pressure $\bar{p}$ and density $\bar{\rho}$; we define with $\bar{e}$ and $\bar{s}$ the internal energy and entropy densities of the fluid phase, respectively.

Conservation of the solid and fluid masses gives (the dot denotes total derivative with respect to time, Δ denotes divergence):

$$\dot{\rho} + \rho \Delta \mathbf{v} = 0 \tag{45}$$

$$\dot{m} = -\Delta \mathbf{f} \tag{46}$$

Conservation of energy over the volume Ω is given by the first principle of thermodynamics:

$$\frac{d\left(E+K\right)}{dt} = P_{(e)} + Q + F \tag{47}$$

where E and K are the internal energy and the kinetic energy of the domain Ω, $P_{(e)}$ is the mechanical power due to external forces, Q is the heat power received by Ω, F is the power due to the fluid flow.

If we denote by **t** the volumetric force density and by **T** the surface force density, then $P_{(e)}$ is given by the Cauchy axiom (repeated indices imply summation, according to tensorial algebra formalism):

$$P_{(e)} = \int_V t_i \, v_i \, dV + \int_\Sigma T_i \, v_i \, dA \,. \tag{48}$$

On the other hand, Q consists in two components: the heat generated in Ω due to actions external to Ω, defined as r in the following, and the heat received for conduction through the surface Σ:

$$Q = \int_V r \, dV - \int_\Sigma q_i \, n_i \, dA \,. \tag{49}$$

F is given by two contributions: the mechanical power due to the vapor flux (notice that $f_i/\bar{\rho}$ is the velocity of the fluid phase) and the rate of internal energy due to the fluid flux (Weitsman 1990):

$$F = -\int_{\Sigma} \bar{p} \frac{f_i}{\bar{\rho}} n_i \, dA - \int_{\Sigma} \bar{e} \, f_i \, n_i \, dA \,. \tag{50}$$

Reminding that (Lemaitre and Chaboche 1988):

$$P_{(i)} + P_{(e)} = \frac{dK}{dt} \,, \tag{51}$$

the extended formulation of the conservation law that gives the first principle becomes:

$$\frac{d}{dt}\int_{V} \rho e \, dV = \int_{V} \sigma_{ij} D_{ij} \, dV + \int_{V} r \, dV - \int_{\Sigma} q_i \, n_i \, dA - \int_{\Sigma} \bar{p} \frac{f_i}{\bar{\rho}} n_i \, dA - \int_{\Sigma} \bar{e} \, f_i \, n_i \, dA \tag{52}$$

where:

$$P_{(i)} = -\int_{V} \sigma_{ij} \, D_{ij} \, dV \,, \tag{53}$$

D_{ij} being the strain rate tensor:

$$D_{ij} = \frac{1}{2}\left(v_{i,j} + v_{j,i}\right) \tag{54}$$

which is linked with the strain tensor **ε**, in the hypothesis of infinitesimal strain theory, by:

$$\mathbf{D} = \dot{\boldsymbol{\varepsilon}} \tag{55}$$

or, being **u** the displacements vector:

$$\varepsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right) \qquad v_i = \dot{u}_i \,. \tag{56}$$

Eq. (52) must hold for any arbitrary domain Ω so that, after applying the Green's theorem and employing (46), the conservation law in the field of small deformations yields:

$$\rho \dot{e} = \sigma_{ij} \, \dot{\varepsilon}_{ij} + r - q_{i,i} - \bar{h}_{,i} \, f_i + \bar{h} \dot{m} \tag{57}$$

where $\bar{h}$ is the enthalpy of the fluid phase, function of its pressure $\bar{p}$, its mass density $\bar{\rho}$ and its internal energy density $\bar{e}$:

$$\bar{h} = \frac{\bar{p}}{\bar{\rho}} + \bar{e} \tag{58}$$

The Clausius-Duhem inequality, or second principle, requires:

$$\frac{d}{dt}\int_{V} \rho s \, dV \geq \int_{V} \frac{r}{T} dV - \int_{\Sigma} \frac{q_i}{T} n_i \, dA - \int_{\Sigma} \bar{s} f_i \, n_i \, dA - \int_{\Sigma} \frac{\gamma_i}{T} n_i \, dA \tag{59}$$

where the fourth integral expresses the fluid phase contribution for entropy and the last one, depending on a general quantity γ, is the overall dissipation due to the processes concrete undergo: damage, viscosity and, in general autogenous effects from internal chemical reactions or (though not implemented in NEWCON3D), plasticity.

Again for arbitrary Ω, by applying the Green's theorem to (59) and employing (46), the thermodynamic inequality states:

$$\rho\,\dot{s}+\left(\frac{q_i}{T}\right)_{,i}+\left(\frac{\gamma_i}{T}\right)_{,i}-\frac{r}{T}+\bar{s}_{,i}\;f_i-\bar{s}\,\dot{m}\geq 0 \tag{60}$$

We now derive r from the first principle (see (57)) and multiply by T > 0; notice also that:

$$\left(\frac{q_i}{T}\right)_{,i}=\frac{q_{i,i}}{T}-\frac{q_i}{T^2}T_{,i}\,; \tag{61}$$

so we obtain:

$$\rho\left(T\,\dot{s}-\dot{e}\right)+\sigma_{ij}\,\dot{\varepsilon}_{ij}-\frac{q_i}{T}T_{,i}+\gamma_{i,i}-\bar{h}_{,i}\;f_i+\bar{h}\,\dot{m}+T\bar{s}_{,i}\;f_i-T\bar{s}\,\dot{m}\geq 0\,. \tag{62}$$

Now if we define the Helmholtz free energy:

$$\psi=e-T\,s \tag{63}$$

and the chemical potential of the fluid phase according to Weitsman (1990):

$$\bar{\mu}=\bar{h}-T\,\bar{s}\,, \tag{64}$$

then the second principle of thermodynamics for a multiphase system, accounting for damage, plasticity, autogenous effects and creep, in the hypothesis of infinitesimal strains yields:

$$-\rho\left(\dot{\psi}+s\dot{T}\right)+\sigma_{ij}\,\dot{\varepsilon}_{ij}-\frac{q_i+\gamma_i}{T}T_{,i}+\gamma_{i,i}+\bar{\mu}\,\dot{m}-f_i\,\bar{\mu}_{,i}-T_{,i}\,\bar{s}_{,i}\;f_i\geq 0\,. \tag{65}$$

It is worth being underlined that the formulation we have here, for a multiphase system able to undergo autogenous effects, elasto-plasticity, damage and creep, turns out to the classical expression of the Clausius-Duhem inequality:

$$-\rho\left(\dot{\psi}+s\dot{T}\right)+\sigma_{ij}\,\dot{\varepsilon}_{ij}-\frac{q_i}{T}T_{,i}\geq 0 \tag{66}$$

but whith some terms generated in addition, due to the newly introduced sources of dissipation, which is in agreement with the approach suggested in (Ireman et al. 2003).

6.1.2 Local state approach

With the two main results of thermodynamics being reached, we use now the local state approach to formulate the constitutive equations for concrete as an elastic-plastic material, with damage, autogenous and viscous effects included as well, in the assumption of infinitesimal deformations and coupling between thermal, hygral and mechanical fields.

Within the local state approach the thermodynamics of a solid continuum, at a point and a given instant of time, is defined as long as the value of a certain number of state variables is known, depending on the space coordinates only and not time-varying (Lemaitre and Chaboche 1988). This assumption implies that the evolution of the thermodynamic state can be considered as a sequence of equilibrium states. We distinguish between observable variables and internal variables.

The physical phenomenon here analyzed, i.e. the considered constitutive law for the material under thermo-hygro-mechanical loads, defines itself the first group of variables: $\boldsymbol{\varepsilon}$ the total strain tensor, T temperature and m the moisture mass per unit volume of the solid (moisture concentration), for which the conservation law (46) holds; these quantities are, in fact, strictly necessary to describe the reversible part of the process. The other variables are required when dissipative phenomena occur and, as a consequence, the present state depends also from the history of the problem, which is described by the value in a specific instant of time by the internal variables.

For the plastic free energy, the necessary internal variables are the following: the cumulative plastic strain p, defined as (Lemaitre and Chaboche 1988):

$$p = \int_0^t \left[\frac{2}{3} \dot{\boldsymbol{\varepsilon}}^p(\tau) : \dot{\boldsymbol{\varepsilon}}^p(\tau) \right]^{1/2} d\tau \tag{67}$$

(the symbol : denoting tensor contraction); p is a scalar measure of the plastic deformation and, as an always increasing quantity, it is associated to a specific instant in a monotone loading history, and it is not sufficient, alone, to describe more complicated cycling loads, for which one or more kinematic tensor variables are needed. Another useful internal variable is a hardening parameter $\boldsymbol{\alpha}$, often coinciding with the plastic deformation ($\boldsymbol{\alpha} = \boldsymbol{\varepsilon}^p$) as in the Prager model for plasticity (linear kinematic hardening).

For the elasto-damaged component of the total free energy, the necessary internal variable is represented by a scalar variable taking into account damage effects due to irreversible changes on the internal structure D, and a scalar variable β providing the current state of

damage -introduced to prescribe the effects of current damage state on the further development of damage, in agreement with (Hayakawa et al. 1998) in which thermodynamics for elastic-plastic materials with damage is investigated.

For the other two components we assume that the amount of total strain accounting for autogenous effects $\boldsymbol{\varepsilon}^{a}$ and the one accounting for relaxation (creep) effects $\boldsymbol{\varepsilon}^{c}$ are themselves the internal variables for the two phenomena, in analogy to plasticity, this approach not preventing the generality of this topic.

The physical process is named to be thermodynamically admissible if it satisfies the Clausius-Duhem inequality at each instant of the evolution in time of the process.

The state variables being identified, the Helmholtz free energy of the material per unit volume ψ needs to be defined. We remind that, in the assumption of infinitesimal strains, the total strain tensor $\boldsymbol{\varepsilon}$ can be split into the components given by the elastic strain, supposed to be affected by damage $\boldsymbol{\varepsilon}^{ed}$, the plastic strain $\boldsymbol{\varepsilon}^{p}$, the autogenous strain $\boldsymbol{\varepsilon}^{a}$ and the viscous strain $\boldsymbol{\varepsilon}^{c}$. Under this hypothesis ψ can be expressed as a superposition of effects: the fraction coming from the elastic behaviour, affected by damage $\psi_e(\boldsymbol{\varepsilon}^{ed}, D, m, T)$, the one due to the development of microscopic defects $\psi_d(\beta)$, the one given by plastic deformation, i.e. hardening process $\psi_p(p, \boldsymbol{\alpha}, m, T)$, the one due to autogenous effects (e.g. given by chemical reactions due to carbonation or to the action of chlorides or sulfates) $\psi_a(\boldsymbol{\varepsilon}^{a}, m, T)$, the one linked to relaxation effects, i.e. creep (at low to medium temperatures), which is supposed to evolve into load-induced thermal effects (at high temperatures) $\psi_c(\boldsymbol{\varepsilon}^{c}, m, T)$:

$$\rho\psi = \rho\psi_e\left(\boldsymbol{\varepsilon}^{e}, D, m, T\right) + \rho\psi_d\left(\beta\right) + \rho\psi_p\left(p, \boldsymbol{\alpha}, m, T\right) + \rho\psi_a\left(\boldsymbol{\varepsilon}^{a}, m, T\right) + \rho\psi_c\left(\boldsymbol{\varepsilon}^{c}, m, T\right) \quad (68)$$

It is worth specifying that the internal structural changes due to damage are supposed to influence the elastic deformation only, under the assumption that the free energy due to plastic, autogenous and viscous deformation is small in comparison with the elastic one, so that the effects of damage on plasticity, autogenous fields and viscosity can be neglected.

6.1.2.1 The elastic free energy

As regards the free energy in the elastic field ψ_e, where damage is supposed to occur, the requirement of its convexity with respect to all the state variables (in order to verify *a priori* the second law inequality) leads to an expression for ψ_e -in presence of damage- still quadratic in $\boldsymbol{\varepsilon}^{e}$ and T, as when damage is not accounted for (Lemaitre and Chaboche 1988). Generalizing

the thermo-elastic approach provided by Lemaitre and Chaboche (1988) to coupled thermo-hygro-mechanics, the following expression for the elastic contribution to the total free energy is obtained:

$$\rho\psi_e = \frac{1}{2}(1-D)\,\mathbf{D_T} : \left(\boldsymbol{\varepsilon}^e - \mathbf{k}\,dT - \boldsymbol{\mu}\,dm\right) : \left(\boldsymbol{\varepsilon}^e - \mathbf{k}\,dT - \boldsymbol{\mu}\,dm\right) + C\,dT^2 + M\,dm^2 \tag{69}$$

in which $\boldsymbol{\varepsilon}^e$ is the elastic strain tensor of the undamaged material, $\mathbf{D_T}$ the tangent stiffness tensor, $\mathbf{k}$ the tensor containing thermal expansion coefficients, $\boldsymbol{\mu}$ the tensor containing the coefficients of hygrometric expansion, so that $\mathbf{k}dT$ and $\boldsymbol{\mu}dm$ represent the thermal and hygral dilation, respectively.

We derive here the state laws starting from the specific free energy ψ. Provided that the following decomposition of the total strain tensor holds:

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^e + \boldsymbol{\varepsilon}^p + \boldsymbol{\varepsilon}^a + \boldsymbol{\varepsilon}^c\,, \tag{70}$$

the potential expressed in Eq. (68) can be written, in a more general formulation, as dependent on the following quantities:

$$\psi = \psi\left(\left[\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^p - \boldsymbol{\varepsilon}^a - \boldsymbol{\varepsilon}^c\right], m, T, V_k\right) = \psi\left(\boldsymbol{\varepsilon}^e, m, T, V_k\right) \tag{71}$$

where we denote by V_k the internal variables defined in §6.1.2.

For the derivative of ψ, then:

$$\frac{\partial\psi}{\partial\boldsymbol{\varepsilon}^e} = \frac{\partial\psi}{\partial\boldsymbol{\varepsilon}} = -\frac{\partial\psi}{\partial\boldsymbol{\varepsilon}^p} = -\frac{\partial\psi}{\partial\boldsymbol{\varepsilon}^a} = -\frac{\partial\psi}{\partial\boldsymbol{\varepsilon}^c}\,. \tag{72}$$

By differentiating ψ with respect to time:

$$\dot{\psi} = \frac{\partial\psi}{\partial\boldsymbol{\varepsilon}^e} : \dot{\boldsymbol{\varepsilon}}^e + \frac{\partial\psi}{\partial m}\dot{m} + \frac{\partial\psi}{\partial T}\dot{T} + \frac{\partial\psi}{\partial V_k}\dot{V}_k \tag{73}$$

and substituting this expression in the second principle (Eq. (65)), one gets in tensorial notation (∇ denotes the gradient and Δ, again, the divergence):

$$\begin{aligned}\left(\boldsymbol{\sigma} - \rho\frac{\partial\psi}{\partial\boldsymbol{\varepsilon}^e}\right) : \dot{\boldsymbol{\varepsilon}}^e + \boldsymbol{\sigma} : \dot{\boldsymbol{\varepsilon}}^p + \boldsymbol{\sigma} : \dot{\boldsymbol{\varepsilon}}^a + \boldsymbol{\sigma} : \dot{\boldsymbol{\varepsilon}}^c - \rho\left(s + \frac{\partial\psi}{\partial T}\right)\dot{T} - \rho\frac{\partial\psi}{\partial V_k}\dot{V}_k + \\ - \frac{\mathbf{q}+\boldsymbol{\gamma}}{T}\cdot\nabla T + \Delta\boldsymbol{\gamma} + \left(\bar{\mu} - \rho\frac{\partial\psi}{\partial m}\right)\dot{m} - \mathbf{f}\cdot\nabla\bar{\mu} - \nabla T\cdot\nabla\bar{s}\cdot\mathbf{f} \geq 0\end{aligned} \tag{74}$$

We make use of the classical hypothesis of thermo-elasticity to reset some terms to zero at the same time. In addition, if also the terms $\Delta\boldsymbol{\gamma}$, and $\mathbf{f}\cdot\nabla\bar{\mu}$ tend to zero, the inequality holding for any $\dot{\boldsymbol{\varepsilon}}^e$, $\dot{m}$, $\dot{T}$, gives the state laws in thermo-elasticity:

$$\boldsymbol{\sigma} = \rho\frac{\partial\psi}{\partial\boldsymbol{\varepsilon}^e} = (1-D)\,\mathbf{D_T} : \left(\boldsymbol{\varepsilon}^e - \mathbf{k}\,dT - \boldsymbol{\mu}\,dm\right) = \overline{\mathbf{D_T}} : \left(\boldsymbol{\varepsilon}^e - \mathbf{k}\,dT - \boldsymbol{\mu}\,dm\right) \tag{75}$$

and:

$$s = -\frac{\partial \psi}{\partial T} \qquad \bar{\mu} = \rho \frac{\partial \psi}{\partial m} \tag{76}$$

Eq. (75) shows that the free energy is a stress potential or, in other words, that the stress tensor **σ** is the conjugate thermodynamic variable for the elastic strain tensor. Moreover, according to (75), the stress tensor **σ** can be considered as a modified stress tensor $\bar{\boldsymbol{\sigma}}$ in presence of damage (Lemaitre and Chaboche 1988):

$$\bar{\boldsymbol{\sigma}} = \frac{\boldsymbol{\sigma}}{1-D} = \mathbf{D_T} : \left(\boldsymbol{\varepsilon}^e - \mathbf{k}\, dT - \boldsymbol{\mu}\, dm \right) \tag{77}$$

Eq. (76), on the other hand, defines the specific entropy and the chemical potential as the conjugate variables to temperature and moisture concentration, respectively.

6.1.2.2 The free energy due to creep

Acknowledged analytic expressions for ψ_c are available in literature. Its definition implies to analyze the class of thermodynamically admissible creep and relaxation functions, **c** and **r** respectively, in the classic visco-elastic constitutive equations, relating the strain and stress tensor **ε** and **σ**:

$$\boldsymbol{\varepsilon} = \mathbf{c} \circ \boldsymbol{\sigma} \qquad\qquad \boldsymbol{\sigma} = \mathbf{r} \circ \boldsymbol{\varepsilon} \tag{78}$$

where the operator $\circ$ is the Stieltjès convolution. It is in fact already known that the field of thermodynamically admissible creep-relaxation functions is larger than the traditional set of monotone positive definite functions used in practice (exponential or power functions), including several non-monotone and even locally negative functions (Hazanov 1995-1997). Particularly, the use of exponentials is prior to power functions due mainly to historical reasons: linear viscosity was born in connection with the study of behaviour of polymers and, only later on, with creep on porous materials, where the power integral kernels *K*, required by the heredity theory were introduced in the Volterra integral form of the constitutive law, expressed in one-dimensional domain by:

$$\sigma(t) = \int_{-\infty}^{t} K(0, t-v)\, d\varepsilon(v) \qquad\qquad K(0,t) = r(t) \tag{79}$$

in which *r(t)* is the relaxation function and the integral is the Stieltjes integral.

One of the expressions for the density of the Helmholtz free energy, still most frequently applied, comes from the Staverman-Schwarzl formulation (Staverman and Schwarzl 1952;

Staverman 1954), established for exponential relaxation functions for polymers in isothermal conditions:

$$\psi_c(t) = \frac{1}{2}\int_{0^-}^{t}\int_{0^-}^{t} r(2t-u-v)\, d\varepsilon(v)\, d\varepsilon(u) \tag{80}$$

where the lower limits of the integrals are justified when loading starts at time *t=0* with a jump; Eq. (80) represents a particular case of the more general expression:

$$\psi_c(t) = \frac{1}{2}\int_{-\infty}^{t}\int_{-\infty}^{t} K(t-u, t-v)\, d\varepsilon(v)\, d\varepsilon(u) \tag{81}$$

where *K* is such a kernel that the functional ψ_c in Eq. (81) is non-negative definite.
Notice, in fact, that (80) comes from (81) when the following relationship holds:

$$K(x, y) = K(x+y) \tag{82}$$

and in (80) *r(t)* is a function of the relaxation spectrum *ρ(μ)*:

$$r(t) = \int_0^{\infty} \rho(\mu)\, e^{-\frac{t}{\mu}}\, d\mu + r_\infty \tag{83}$$

r_∞ being the long-term modulus.

Non isothermal conditions were successively taken into account by Staverman assuming small changes of temperature, leading to a similar expression, though in incremental form (Staverman and Schwarzl 1952).

As shown by Eq. (80), the Staverman-Schwarzl's expression is a non-aging-like model for visco-elastic materials, since only heredity is taken into account i.e. the relaxation function, within the Volterra integral, is dependent just on the time lag *t-u-v* (duration of unit constant stress), and not separately on *t*, *u*, *v*. This second scenario would account for the description of the typical time-hardening process exhibited by concrete under solidification, mainly due to the mass growth of cement hydration products per unit volume.

A thermodynamic formulation for ageing visco-elasticity has been more recently suggested by Bazant and Huet (Bazant and Huet 1999), by generalization of the Staverman-Schwarzl's formula; after considerations on the integration domain, an expression for ψ_c is given that involves double Stieltjes integration, over a square domain for the strain history, of a quadratic expression depending only on a symmetrized form of the relaxation function, according to the following:

$$\psi_c(t) = \frac{1}{2}\int_{0}^{t}\int_{0}^{t} f(t, u, v)\, d\varepsilon(v)\, d\varepsilon(u) \tag{84}$$

where *f(t, u, v)* is found to be symmetric with respect to *u* and *v*:

$$f(t,u,v) = f(t,v,u) \tag{85}$$

and:

$$f(t,u,v) = \min\left[R(2t-v,u),\, R(2t-u,v)\right] \tag{86}$$

with *R(t, t')* relaxation function for the aging material.

Expression (84) suggested by Bazant and Huet for the free energy is shown to be a potential for the stress, in fact after differentiation of (84) one gets:

$$\begin{aligned}\frac{\partial \psi_c(t)}{\partial \boldsymbol{\varepsilon}(t)} &= \frac{1}{2}\int_0^t \min\left[R(2t-u,t), R(2t-t,u)\right] d\varepsilon(u) \\ &\quad + \frac{1}{2}\int_0^t \min\left[R(2t-t,v), R(2t-v,t)\right] d\varepsilon(v).\end{aligned} \tag{87}$$

If we substitute t = u + Δ, with Δ > 0, and we suppose the material to harden, rather than to soften, as it ages, that is:

$$R(t+\Delta, t'+\Delta) \geq R(t,t'), \tag{88}$$

then: i) in the first integral of Eq. (87) the second of the two relaxation functions is the minimum and ii) in the second integral the first relaxation function is the minimum, so that:

$$\begin{aligned}\frac{\partial \psi_c(t)}{\partial \boldsymbol{\varepsilon}(t)} &= \frac{1}{2}\int_0^t R(t,u)\, d\varepsilon(u) + \frac{1}{2}\int_0^t R(t,v)\, d\varepsilon(v) \\ &= \int_0^t R(t,t')\, d\varepsilon(t') = \boldsymbol{\sigma}^c(t)\end{aligned} \tag{89}$$

which proves that the free energy per unit volume of the material is a potential for the stress and allows to obtain a constitutive equation for ageing visco-elasticity consistent with continuum thermodynamics. Relationship (89) holds for viscoelastic ageing materials exhibiting instantaneous elasticity, however in absence of instantaneous elasticity it is sufficient to add the (instantaneous) purely viscous stress to the stress obtained from the potential, to have the correct stress, so that the main relationship in Eq. (89) still holds.

6.1.2.3 The other state laws and the complementary laws

In analogy to what stated for the stress tensor, temperature and moisture concentration in §6.1.2.1, the conjugate forces corresponding to the internal state variables D, p, **α**, β, $\boldsymbol{\varepsilon}^a$ and $\boldsymbol{\varepsilon}^c$,

that are Y, R, **A**, B, $\boldsymbol{\sigma}^a$, $\boldsymbol{\sigma}^c$ respectively, can be found by taking into account the different components of the total free energy ρψ:

$$\begin{gathered} Y = \rho\frac{\partial\psi_e}{\partial D} = -\frac{1}{2}\mathbf{D_T} : \left(\boldsymbol{\varepsilon}^e - \mathbf{k}\, dT - \boldsymbol{\mu}\, dm\right) : \left(\boldsymbol{\varepsilon}^e - \mathbf{k}\, dT - \boldsymbol{\mu}\, dm\right) \\ R = \rho\frac{\partial\psi_p}{\partial p} \qquad \mathbf{A} = \rho\frac{\partial\psi_p}{\partial \boldsymbol{\alpha}} \qquad B = \rho\frac{\partial\psi_d}{\partial \beta} \qquad \boldsymbol{\sigma}^a = -\rho\frac{\partial\psi_a}{\partial \boldsymbol{\varepsilon}^a} \qquad \boldsymbol{\sigma}^c = -\frac{\partial\psi_c}{\partial \boldsymbol{\varepsilon}^c} \end{gathered} \tag{90}$$

In order to describe the inelastic dissipative process, as well as the evolution of the internal state variables, it is necessary to refer to the dissipation potential, i.e. a function continuous and convex with respect to the flux variables (i.e. time derivatives of the internal state variables and heat flux). By means of such a function, the relationships between flux variables and dual variables given by the conjugate forces, i.e. the complementary laws, can be found.
If we define the dissipation potential φ and its dual potential φ*, which is function of the conjugate forces:

$$\varphi = \varphi\left(\dot{\boldsymbol{\varepsilon}}^p, \dot{D}, \dot{p}, \dot{\boldsymbol{\alpha}}, \dot{\beta}, \boldsymbol{\varepsilon}^a, \boldsymbol{\varepsilon}^c, \frac{\mathbf{q}}{T}, \mathbf{f}\right) \qquad \varphi^* = \varphi^*\left(\boldsymbol{\sigma}, Y, R, \mathbf{A}, B, \mathbf{l}, \mathbf{i}\right) \tag{91}$$

where **l** and **i** are the thermodynamic conjugate forces related to heat flux **q** and moisture flux **f**, respectively:

$$\mathbf{l} = \nabla T \qquad \mathbf{i} = \nabla\bar{h}\,, \tag{92}$$

then, the complementary laws are the following:

$$\begin{gathered} \dot{\boldsymbol{\varepsilon}}^p = \frac{\partial\varphi^*}{\partial\boldsymbol{\sigma}} \qquad \dot{D} = -\frac{\partial\varphi^*}{\partial Y} \qquad \dot{p} = -\frac{\partial\varphi^*}{\partial R} \qquad \dot{\boldsymbol{\alpha}} = -\frac{\partial\varphi^*}{\partial\mathbf{A}} \\ \dot{\beta} = -\frac{\partial\varphi^*}{\partial B} \qquad \boldsymbol{\varepsilon}^a = \frac{\partial\varphi^*}{\partial\boldsymbol{\sigma}} \qquad \boldsymbol{\varepsilon}^c = \frac{\partial\varphi^*}{\partial\boldsymbol{\sigma}} \qquad \frac{\mathbf{q}}{T} = -\frac{\partial\varphi^*}{\partial\mathbf{l}} \qquad \mathbf{f} = -\frac{\partial\varphi^*}{\partial\mathbf{i}}. \end{gathered} \tag{93}$$

It is worth underlining that the first law leads to the plasticity law whereas the following four relationships represent the evolution laws for the internal state variables, particularly the second is the so-called property of normality for the damage variable D.

Now, if the flux vector **J** is defined as:

$$\mathbf{J} = \rho\left\{\dot{\boldsymbol{\varepsilon}}^p, \dot{D}, \dot{p}, \dot{\boldsymbol{\alpha}}, \dot{\beta}, \boldsymbol{\varepsilon}^a, \boldsymbol{\varepsilon}^c, \mathbf{q}, \mathbf{f}\right\}^T \tag{94}$$

and its thermodynamic conjugate force vector **X** as follows:

$$\mathbf{X} = \left\{\boldsymbol{\sigma}, Y, R, \mathbf{A}, B, \frac{\mathbf{l}}{T}, \mathbf{i}\right\}, \tag{95}$$

then the rate of entropy production can be expressed as the product between the two, and the second principle requires it is non-negative:

$$\mathbf{X}\cdot\mathbf{J}=\boldsymbol{\sigma}:\dot{\boldsymbol{\varepsilon}}^{p}-Y\dot{D}-R\dot{p}-\mathbf{A}:\dot{\boldsymbol{\alpha}}-B\dot{\beta}+\boldsymbol{\sigma}:\boldsymbol{\varepsilon}^{a}+\boldsymbol{\sigma}:\boldsymbol{\varepsilon}^{c}-\frac{1}{T}\mathbf{l}\cdot\mathbf{q}-\mathbf{i}\cdot\mathbf{f}\geq 0\,. \tag{96}$$

If the uncoupling among intrinsic, thermal and hygral dissipation is conceived (which does not imply the uncoupling of effects) and the hardening effect is supposed to happen without damage and, viceversa, damage occurs without a macroscopic plastic deformation, the following inequalities hold separately:

$$\boldsymbol{\sigma}:\dot{\boldsymbol{\varepsilon}}^{p}-R\dot{p}-\mathbf{A}:\dot{\boldsymbol{\alpha}}-B\dot{\beta}+\boldsymbol{\sigma}:\boldsymbol{\varepsilon}^{a}+\boldsymbol{\sigma}:\boldsymbol{\varepsilon}^{c}\geq 0 \qquad -Y\dot{D}\geq 0 \tag{97}$$

which, Y being quadratic, definite positive according to Eq. (90), states the increase of damage in time, in order to fulfill the second principle of thermodynamics.

Eq. (96) represents another expression for the Clausius-Duhem inequality (65), where the extra entropy flux γ is here shown to include as dissipative processes -in addition to heat dissipation and fluid flow dissipation- the mechanical dissipative processes connected with damage, autogenous effects and (irreversible) creep, which appear in (96).

6.1.3 The coupled thermo-hygro-mechanical problem

The NEWCON3D model consists in a series of balance equations, i.e. a mass balance equation of water (both liquid and vapour, taking into account phase changes and hydration/dehydration processes), an enthalpy balance equation of the whole multiphase medium (considering the latent heat of phase change and the hydration/dehydration process), a linear momentum balance equation of the fluid phases (Darcy's equation) and a linear momentum balance equation of the whole medium.

Appropriate constitutive equations and some thermodynamic relationships are included as well (Lewis and Schrefler 1987; Schrefler et al. 1989; Baggio et al. 1995; Majorana et al. 1995-1997-1998-2004; Salomoni et al. 2007-2008-2009; Mazars 1984-1989; Mazars and Pijaudier-Cabot 1989-1996; Kachanov 1958; Gerard et al. 1998; Nechnech et al. 2001).

The field equations of the model are briefly recalled below: the continuity equation for non-isothermal flow is expressed in terms of relative humidity as:

$$\frac{\partial h}{\partial t}-\nabla^{T}\mathbf{C}\nabla h-\frac{\partial h_{s}}{\partial t}-k\frac{\partial T}{\partial t}+\chi\mathbf{m}^{T}\frac{\partial\boldsymbol{\varepsilon}}{\partial t}=0 \tag{98}$$

where h is the relative humidity, T is temperature, $k=\left(\frac{\partial h}{\partial T}\right)\Big|_{m,\boldsymbol{\varepsilon}}$ is the hygrothermic coefficient, i.e. the change in h due to one-degree change of T at constant m, $\boldsymbol{\varepsilon}$ and a fixed

degree of saturation, dh_s is the self-desiccation, $\chi = \left(\frac{\partial h}{\partial \boldsymbol{\varepsilon}_v} \right)\Big|_{T,m}$ and represents the change in h due to unit change of volumetric strain $\boldsymbol{\varepsilon}_v$ at constant m, T and a given degree of saturation, **C** is the (relative humidity) diffusivity diagonal matrix, $\mathbf{m}^T$ is the vector [1 1 1 0 0 0]. The last term in Eq. (98), in particular, accounts for the coupling of the hygro-thermal and the mechanical response (Schrefler et al 1989).

The relative humidity is expressed as the ratio between the equilibrium water vapour pressure p_{gw} and the saturation pressure p_{gwsat} through the Kelvin equation (Bazant and Najjar 1972):

$$h = \frac{p_{gw}}{p_{gw\,sat}(T)} \tag{99}$$

and it is directly connected to the moisture concentration m, at variable temperature, by the sorption-desorption isotherms:

$$dh = Kdm + kdT + dh_s - \chi \mathbf{m}^T d\boldsymbol{\varepsilon} \tag{100}$$

where $K = \left(\frac{\partial h}{\partial m} \right)\Big|_T$ is the so-called inverse slope of desorption isotherm.

Heat balance requires:

$$\rho C_q \frac{\partial T}{\partial t} - C_a \frac{\partial m}{\partial t} - C_w \mathbf{f} \cdot \nabla T = -\Delta \mathbf{q}\,, \tag{101}$$

where ρ is the mass density of concrete, C_q the isobaric heat capacity of concrete (per kilogram of concrete) including chemically bound water but excluding free water, C_a is the heat of sorption of free water (per kilogram of free water); C_w is the isobaric heat capacity of liquid water, $C_w \mathbf{f} \cdot \nabla T$ is the rate of heat supply due to convection by moving water, **f** is the flux of humidity, depending on the gradient of h by means of the permeability c:

$$\mathbf{f} = -c\, \nabla h\,, \tag{102}$$

q is the heat flux which can be due to temperature gradient or moisture concentration gradient. Combining the two cases, the constitutive law defining the heat flux comes from the Fourier's law (heat flux due to temperature gradient) and the Dufour's flux (heat flux due to moisture concentration gradient):

$$\mathbf{q} = -a_{TW}\, \nabla m - a_{TT}\, \nabla T \tag{103}$$

where the coefficients a_{Tw} and a_{TT} depend on m and T.

Finally, considering the macroscopic linear momentum balance equation for the whole medium we have:

$$div\boldsymbol{\sigma} + \rho\mathbf{g} = 0 \tag{104}$$

where ρ is the density of the multiphase medium (concrete plus water species) and **g** the acceleration related to gravity.

As regards the mechanical field, the constitutive relationship for the solid skeleton in incremental form can be written as:

$$d\boldsymbol{\sigma}' = (1-D)\,\mathbf{D}_T \;:\left(d\boldsymbol{\varepsilon} - d\boldsymbol{\varepsilon}^T - d\boldsymbol{\varepsilon}^c - d\boldsymbol{\varepsilon}^{lits} - d\boldsymbol{\varepsilon}^p - d\boldsymbol{\varepsilon}^{sh} - d\boldsymbol{\varepsilon}^a\right) \tag{105}$$

where $\boldsymbol{\sigma}'$ is the effective stress tensor according to Terzaghi ($\boldsymbol{\sigma}' = \boldsymbol{\sigma} + p_{gw}\,\mathbf{I}$, **I** being the identity matrix and p_{gw} is the pore water pressure); D is the upgraded scalar radio-chemo-thermo-mechanical damage (see below), $\mathbf{D}_T$ is the tangent stiffness matrix, $d\boldsymbol{\varepsilon}^T$ is the strain rate caused by thermo-elastic expansion, $d\boldsymbol{\varepsilon}^c$ the strain rate accounting for creep, $d\boldsymbol{\varepsilon}^{lits}$ the load induced thermal strain rate, $d\boldsymbol{\varepsilon}^p$ the plastic strain rate, $d\boldsymbol{\varepsilon}^{sh}$ is due to shrinkage and $d\boldsymbol{\varepsilon}^a$ represents the autogenous strain increments already defined in §6.1.2.

Particularly, when only mechanical forces are taken into account, D is the damage scalar variable following the scalar isotropic model by Mazars.

According to this theory, the classical effective stress concept (Kachanov 1958) is modified to take into account damage, measuring a reduction in the resistant area due to cracks beginning and spreading:

$$\overline{\boldsymbol{\sigma}}' = \boldsymbol{\sigma}'\frac{S}{\overline{S}} = \frac{\boldsymbol{\sigma}'}{1-D}, \tag{106}$$

where S and $\overline{S}$ are the resistant area of the uncracked and cracked material, respectively.

Since the damaging mechanisms are different in uniaxial tension and compression experiments, the damage parameter D_m (the subscript stands for mechanical contributions only) is decomposed into two parts, d_t for tension and d_c for compression, which are function of the equivalent strain:

$$\tilde{\varepsilon} = \sqrt{\sum_{i=1}^{3}\left(\langle\varepsilon_i\rangle_+\right)^2} \qquad \left(\langle\varepsilon_i\rangle_+ = \frac{|\varepsilon_i| + \varepsilon_i}{2}\right), \tag{107}$$

ε_i being the principal strains. Hence:

$$D_m = \alpha_t d_t + \alpha_c d_c \tag{108}$$

where α_t and α_c are weighting coefficients defined in (Mazars 1989).

Thermo-chemical damage D_{tc} has been introduced for the first time in (Majorana et al. 1995); thermo-chemical effects are here combined with mechanical effects in a multiplicative way, as proposed also by Gerard et al. (1998) and Nechnech et al. (2001). D_{tc} is expected to describe thermo-chemical material degradation at elevated temperatures (mainly due to micro-cracking and cement dehydration) resulting in reduction of the material strength properties, and, combining it with mechanical damage, Eq. (106) becomes:

$$\overline{\boldsymbol{\sigma}}' = \frac{\boldsymbol{\sigma}'}{(1-D_m)(1-D_{tc})} \tag{109}$$

and, under multiplicativity, the total damage D is defined by:

$$D = 1-(1-D_m)\,(1-D_{tc})\,. \tag{110}$$

The upgrade of the model consists in the introduction of radiation damage in this framework, assuming that nuclear radiation can activate a damage process which combines with the mechanical and thermo-chemical ones so that the above multiplicative relation is maintained and the total damage is:

$$D = 1-(1-D_m)\,(1-D_{tc})\,(1-D_r) \tag{111}$$

in which D_r accounts for radiation damage. Its definition and evaluation will be better illustrated in the following Chapter.

The application, within the numerical code NEWCON3D, of a standard Finite Element discretization in space of Eqs. (98), (101), (104) results in a system of the kind:

$$\begin{bmatrix} \mathbf{K} & \mathbf{HU} & \mathbf{TU} \\ \mathbf{L}^T & \mathbf{I} & \mathbf{TP} \\ \mathbf{0} & \mathbf{TH} & \mathbf{TS} \end{bmatrix} \begin{Bmatrix} \dot{\overline{\mathbf{u}}} \\ \dot{\overline{\mathbf{h}}} \\ \dot{\overline{\mathbf{T}}} \end{Bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{TR} \end{bmatrix} \begin{Bmatrix} \overline{\mathbf{u}} \\ \overline{\mathbf{h}} \\ \overline{\mathbf{T}} \end{Bmatrix} = \begin{Bmatrix} \dot{\mathbf{f}} + \mathbf{c} \\ \dot{\mathbf{HG}} \\ \dot{\mathbf{TG}} \end{Bmatrix} \tag{112}$$

in which $\overline{\mathbf{u}}$, $\overline{\mathbf{h}}$ and $\overline{\mathbf{T}}$ are the nodal values of the basic variables: displacements, relative humidity and temperature, **HU** and **TU** account for shrinkage and thermal dilation effects, respectively; $\mathbf{L}^T$ and **TP** are the coupling matrices describing the influence of the mechanical and thermal field on the hygral one, respectively; **Q** is the diffusivity matrix accounting for sorption-desorption isotherms; **TH** is the coupling matrix between the hygral and the thermal

field in terms of capacity ; **TS** is the matrix of heat capacity; **TR** is the matrix of thermal transmission including the convective term; **c** is the matrix accounting for creep; **HG** is the matrix of humidity variation due to drying and **TG** accounts for heat fluxes.

References to Chapter 6

[1] Baggio P., Majorana C.E., Schrefler B.A., Thermo-hygro-mechanical analysis of concrete, Int J Num Meth Fluids 20, 573-595, 1995.

[2] Bazant Z.P., Huet C., Thermodynamic functions for ageing viscoelasticity: integral form without internal variables, Int J Solids Structures 36, 3993-4016, 1999.

[3] Bazant Z.P., Najjar L.J., Nonlinear water diffusion in nonsaturated concrete, Matériaux et constructions 25(5), 3-20, 1972.

[4] Gerard B., Pijaudier-Cabot G., Laborderie C., Coupled diffusion-damage modeling and the implications on failure due to strain localization, Int J Solids Structures 35(31-32), 4107-20, 1998.

[5] Hayakawa K., Murakami S., Liu Y., An irreversible thermodynamics theory for elastic-plastic-damage materials. Eur J Mech, A/Solids 17(1), 13-32, 1998.

[6] Hazanov S., New class of creep-relaxation functions, Int J Solids Structures 32(2), 165-172, 1995.

[7] Hazanov S., On separation of energies in viscoelasticity, Mech Res Commun 24(2), 167-177, 1997.

[8] Ichikawa T., Koizumi H., Possibility of radiation-induced degradation of concrete by alkali-silica reaction of aggregates, J of Nucl Sci Technol 39(8):880-884, 2002.

[9] Ireman P., Klarbring A., Stromberg N., A model of damage coupled to wear. Int J Sol Str 40, 2957-2974, 2003.

[10] Kachanov M.D., Time of Rupture Process under Creep Conditions. Izvestia Akademii Nauk 8, 26-31 (in Russian), 1958.

[11] Lemaitre J., Chaboche J.-L., Mechanics of solid materials, Dunod, Paris, (in French), 1988.

[12] Lewis R.W., Schrefler B.A., The Finite Element Method in the Deformation and Consolidation of Porous Media. J. Wiley & Sons Ltd., New York, (Chapter 9 with C. Majorana, “Two-dimensional, non-linear, thermo-elasto-plastic consolidation program PLASCON”, 207-333), 1987.

[13] Majorana C.E., Saetta A., Scotta R., Vitaliani R., Mechanical and durability models for lifespan analysis of bridges. IABSE Sym.: Extending the lifespan of structures, San Francisco, CA, USA, aug. 23-25, 1253-1258, 1995.

[14] Majorana C.E., Salomoni V.A., Parametric analyses of diffusion of activated sources in disposal forms. J of Hazard Mat A113, 45-56, 2004.

[15] Majorana C.E., Salomoni V.A., Secchi S., Effects of mass growing on mechanical and hygrothermic response of three-dimensional bodies, J Mat Process Technol PROO64/1-3, 277-286, 1997.

[16] Majorana C.E., Salomoni V.A., Schrefler B.A., Hygrothermal and mechanical model of concrete at high temperature. Mat Str 31, 378-386, 1998.

[17] Mazars J., Application de la mecanique de l'endommagement au comportament non lineaire et la rupture du beton de structure. Ph.D. dissertation, L.M.T., Université de Paris, France, 1984.

[18] Mazars J., Description of the behaviour of composite concretes under complex loadings through continuum damage mechanics. In: Proc. 10th U.S. National Congress of Applied Mech, ASME. Lamb J.P. Ed., 1989.

[19] Mazars J., Pijaudier–Cabot G., Continuum Damage Theory – Application to Concrete, J Engrg Mech ASCE 115, 345-365, 1989.

[20] Mazars J., Pijaudier-Cabot G., From damage to fracture mechanics and conversely: a combined approach, Int J Solids Structures 33, 3327-42, 1996.

[21] Nechnech W., Reynouard J.M., Meftah F., On modeling of thermo-mechanical concrete for the finite element analysis of structures submitted to elevated temperatures. In: Proc. Fracture Mech of Concrete Structures, R. De Borst, J. Mazars, G. Pijaudier-Cabot, J. van Mier eds., 271-278, 2001.

[22] Salomoni V.A., Majorana C.E., Giannuzzi G.M., Miliozzi A., Thermal-fluid flow within innovative heat storage concrete systems for solar power plants, Int J Num Meth for Heat and Fluid Flow (Special Issue) 18(7/8), 969-999, 2008.

[23] Salomoni V.A., Majorana C.E., Khoury G.A., Stress-strain experimental-based modeling of concrete under high temperature conditions. In: Civil Engineering Computations: Tools and Techniques, Ch.14. B.H.V. Topping ed., Saxe-Coburg Publications, 319-346, 2007a.

[24] Salomoni V.A., Mazzucco G., Majorana C.E., Mechanical and Durability Behavior of Growing Concrete Structures, Engrg Comput 24(5), 536-561, 2007b.

[25] Salomoni V.A., Majorana C.E., Mazzucco G., Xotta G., Khoury G.A., Multiscale modelling of Concrete as a Fully Coupled Porous Medium. In: Concrete Materials: properties, performance and applications, Ch.3, J.T. Sentowski ed., Nova Science Publishers Inc, 2009.

[26] Schrefler B.A., Simoni L., Majorana C.E., A general model for the mechanics of saturated-unsaturated porous materials, Mat Str 22, 323-334, 1989.

[27] Staverman A.J., Schwarzl F., Thermodynamics of viscoelastic behaviour. In: Proc Acad Sci Amst, The Netherlands, B55, 474-490, 1952.

[28] Staverman A.J., Thermodynamics of linear viscoelastic behavior. In: Proc Second International Congress on Rheology, Academic Press, inc., New York, 134-138, 1954.

[29] Weitsman Y., A continuum diffusion model for viscoelastic materials, J Phys Chem 94, 961-968, 1990.

Chapter 7

7.1 Numerical analyses

First numerical analyses have been held to simulate the response of a concrete structure for a time span of about 50 years, under a trial neutron flux density of 10^{12}n/(cm^2 s), in order to get the validation of the newly introduced radiation damage law and characterize the different behaviour of ordinary concrete (OPC), compared to other improved concretes.
At a second stage, the specific work scenario for the SPES facility has been simulated and the FEM analysis has been held in conjunction with Monte Carlo analyses of radiation transport into the concrete shield, to model also the thermal aspect of the phenomenon, which is found to be not negligible.

7.1 Radiation damage

Radiation damage has been introduced next to the pre-existing parameters of damage (mechanical and thermo-chemical), based on the experimental evidences reported in Chapter 4. In particluar the decay of the elastic moduls in function of the neutron fluence has been considered to be the phenomenological index for radiation damage, in accordance with the effective stress theory.

As reported by available experimental data (see Fig.4.3), a decay in the elastic modulus of concrete is to be expected due to nuclear radiation; radiation damage has been defined as the ratio $\overline{E}/E$ (elastic modulus after radiation related to the elastic modulus of the virgin material), with respect to the lower envelopping curve, which the expression is provided of, in Fig.7.1. The scattering emerging from the comparison of different test series is explained by the large

variety of test conditions: concrete making materials, mix proportions of mortars, specimens size, cooling and drying conditions, impinging of fast or slow neutrons, possible simultaneous temperature exposure of specimens.

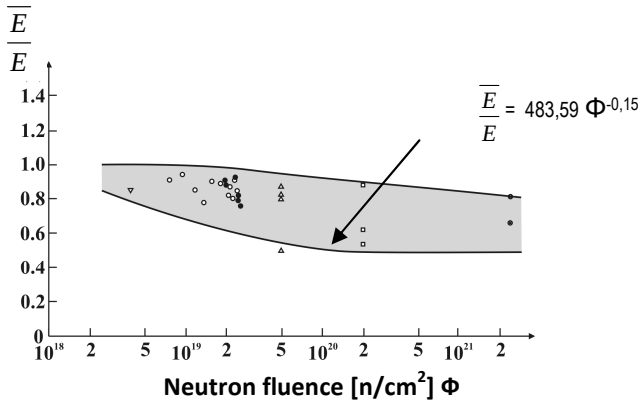

Fig.7.1 – Modulus of elasticity of concrete after neutron radiation $\overline{E}$ related to modulus of elasticity of untreated concrete *E* (expression for radiation damage in NEWCON3D).

The upper behaviour shows that a neutron fluence less than 10^{19}n/cm^2 leads to a slight decrease in the in the $\overline{E}$ /*E* ratio, then the decrease reaches also 50% at high neutron fluences. Hence it has been supposed in the numerical code that under neutron fluences of 10^{18}n/cm^2 the radiation damage is zero and above 10^{20}n/cm^2 it keeps equal to 50% (see Appendix A for the numerical implementation within the code).

The available data are not sufficient to separate the effects of radiation and of heating, which in many cases the samples undergo in nuclear vessels conditions, however it seems reasonable to observe that the strength loss is primarily due to neutron radiation, in analogy with similar graphs for the compressive and the tensile strength, for which a separation of the two effects was possible (see Figs.4.1 and 4.2).

7.2 Geometry

The case study takes its origin from the SPES Project that is currently being developed at the National Laboratories of Legnaro (Padova, Italy). The facility will be directed to the production of special radioactive heavy ion beams from a primary proton beam impinging on a target made of fissionable material, where fission reactions are expected to take place, thus representing ideally a point-source of neutrons in the specific problem.

In a first step the attenuation process of neutrons in concrete is considered to be one-dimensional through the thickness of the shielding slab. This assumption is not restrictive if one considers the geometry of the problem, i.e. the diffusion of neutrons from a neutron source through a wall; at a second stage, a more realistic simulation of the radiation field has been sought with the aid of the Monte Carlo technique.

In both cases the FEM analyses with NEWCON3D have investigated a $1m^2$ square cross-sectional area portion of the wall of the target bunker building, just opposite the source; the thickness of the walls and the disposition of the source inside the structure is in agreement with the design drawnings provided by INFN, after first radio-protection studies involving also the target cave, that is to be kept under 0.25mSv/h ambient dose equivalent (Spes Executive Summary 2008) to fulfill dose limits.

The geometry relative to two different work scenarios (radioactive ion beam of 40MeV, 200μA and of 70MeV, 350μA) is reported in Fig.7.2, while in Fig.7.3 the reference sub-model investigated with the FEM analyses is reported, with a schematic view of its boundary conditions.

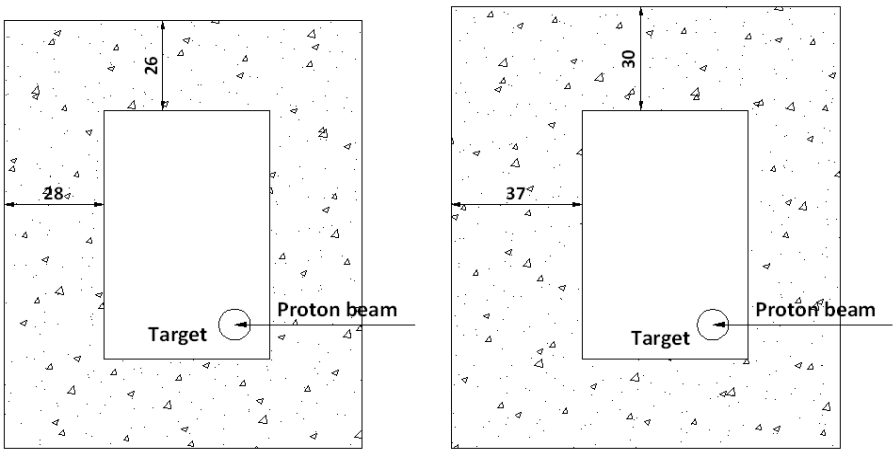

Fig.7.2 – Cross-section of the target cave with the required shielding thickness [dm] relative to two scenarios for the proton beam.

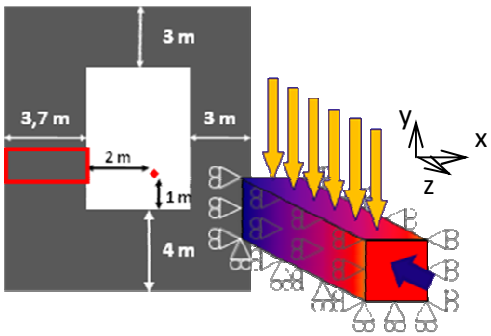

Fig.7.3 – Cross-section of the target cave and investigated volume for the numerical model.

The reference slice has been discretized by 20-nodes brick elements, with constraints on the base surface and laterally: dilations/contractions are allowded along the wall thickness only; self-weight has been accounted for as well.

7.3 First step analyses: radiation damage effect

In the first level analyses a neutron flux density of 10^{12}n/(cm^2 s) has been assumed to irradiate the target room, non-stop for 50 years duration span.

No humidity and temperature variations on the FEM model have been imposed at this stage, to investigate the mechanical response due to radiation damage only.

Neutron fluence (obtained by integration of the neutron flux density in time) is supposed to proceed outwards from the room through the shielding medium, mono-dimensionally.

Two extreme situations have been considered for neutron energies: the case in which the whole flux is given by fast neutrons and the one in which it comes totally from thermal neutrons, in order to predict the minimum and maximum threshold for radiation damage under a realistic variegated neutron spectrum, which should represent an intermediate solution. Neutron attenuation for both fast and thermal neutrons has been described applying the two simplified theories illustrated in §2.1.2.2.1.

Several concrete mixtures have been compared; their application on nuclear facilities was discussed in Chapter 4; the mechanical and shielding properties used for the analyses are taken from Kaplan (1989) and here summarized (see Tables 7.1 and 7.2).

When not provided in literature, the Young's modulus has been derived from the empirical relation proposed by the national standard (Ministerial Decree and Circular to Ministerial Decree 2008) as a function of the compressive cylinder strength:

$$E_{lcm} = 22000 \left(\frac{f_{lcm}}{10} \right)^{0,3} \eta_{E,} \qquad \eta_E = \left(\frac{\rho}{2200} \right)^2, \tag{113}$$

where E_{lcm} is the average value of the secant elastic modulus after 28 days curing [Mpa], f_{lcm} is the average compressive cylinder strength [Mpa], ρ is the density of concrete [kg/cm^3]. Concrete is supposed to behave elastically under its self-weight.

Table 7.1 – Elemental composition of the reference mixture for serpentine (**a**), barytes (**b**), limonite-steel (**c**), magnetite (**d**) concrete, in agreement with Ohgishi et al. (1972), Gallaher and Kitzes (1953), Davis et al. (1956).

a)

Element	*% by weight [%]*
MgO	33.4
SiO_2	37.59
Al_2O^3	3.42
Fe_2O_3	5.15
FeO	2.04
CaO	4.82
CO	1.65
Na_2O	1.13
H_2O	10.26

b)

Element	*per g/cm^3 of concrete*
H in water	0.0243
O in water	0.195
in ore	0.872
in cement	0.118
Mg in cement	0.00385
Al in cement	0.0137
Si in cement	0.0352
S	0.364
Ca in ore	0.0203
in cement	0.147
Fe in ore	0.151
in cement	0.0091
Ba	1.551

c)

Element	*per g/cm^3 of concrete*
H	0.028
O	0.806
Mg, Al	0.039
Si	0.078
Ca	0.25
Ti	-
Mn	-
Fe	3.03

d)

Element	*% by weight [%]*
Fe	55.9
O	34.4
Ca	6.6
Si	1.6
H	0.7
Al	0.6
Mg	0.2

Table 7.2 – Assumed values for mechanical and shielding properties of the analyzed concretes, in agreement with Kaplan (1989), Ohgishi et al. (1972), Gallaher and Kitzes (1953), Davis et al. (1956).

Type of concrete	*Density ρ [kg/m^3]*	*f_{ck} [Mpa]*	*E [MPa]*	*Σ_R [cm^{-1}]*	*L [cm]*
Ordinary	2.33	25	32.484	0.0787	8.0434
Serpentine	2.09	19.3	24.069	0.0836	2.11
Barytes	3.5	24.8	31.157	0.0819	5.4928
Ferro-phosphorus	4.65	30.4	28.200	0.119	2.2045
Limonite-steel	4.27	38.4	47.900	0.1158	2.1016
Magnetite	3.41	41.8	57.800	0.1061	3.0367

Other mechanical parameters of interest, implemented in the numerical code, are reported in Table 7.3:

Table 7.3 – Material data for numerical analyses.

Poisson's ratio	0.2
Permeability/g (isotropic)	40 mm^2/days
Thermal expansion coefficient	10^{-6} °C^{-1}
Specific heat	880 J/(kg °C)
Heat conductivity (isotropic)	$0.13 \cdot 10^{-2}$ J/(mm °C s)
Hygrothermic coefficient k	0.005 °C^{-1}
Coefficient χ for h=0	0.004
Damage parameters according to Mazars model (1984-1989)	
Triggering of damage K_0	$0.1 \cdot 10^{-3}$
A_t	1
B_t	2000
A_c	1.4
B_c	1545
Double creep law parameters (Bazant and Osman 1976)(time in days)	
Number of Maxwell units	8
E_0	70836 MPa
φ_1	4.5
M	0.296
A	0.076
N	0.181

In particular the parameters for damage refer to the Mazars model for concrete (Mazars 1984-1989); the parameters for creep refer to the Maxwell chain model for the relaxation function R(t, t') of Eq. (89), in the form of the double power law by Bazant and Osman (1976).

Results are reported in Fig.7.4 in terms of displacements u_y, principal stress σ_y (with y loading direction) and total damage *D* as functions of the irradiation time on OPC; the graphs refer to a node belonging to the directly impinged concrete surface of the sub-model.
Damage is shown to reach its maximum (50%) after 5-10 years and, correspondingly, stresses reach their equilibrium value; differently, a delay in the occurrence of the asymptotic (steady) value for displacements is shown, accompanied by the occurrence of a moderate expansion

after the peak value: such a behaviour is referable to concrete creep. The stress state is strongly affected by radiation, generating the inability of concrete for sustaining even low compression values at the expense of larger deformations.

In Fig.7.5 the stress-strain curve is plotted for OPC, after a 50 years exposure: a 50% decay in the elastic modulus is envisaged, due to radiation damage, as it was expected from Fig.7.1, for the neutron fluence corresponding to 50 years duration span at 10^{12}n/(cm^2 s).

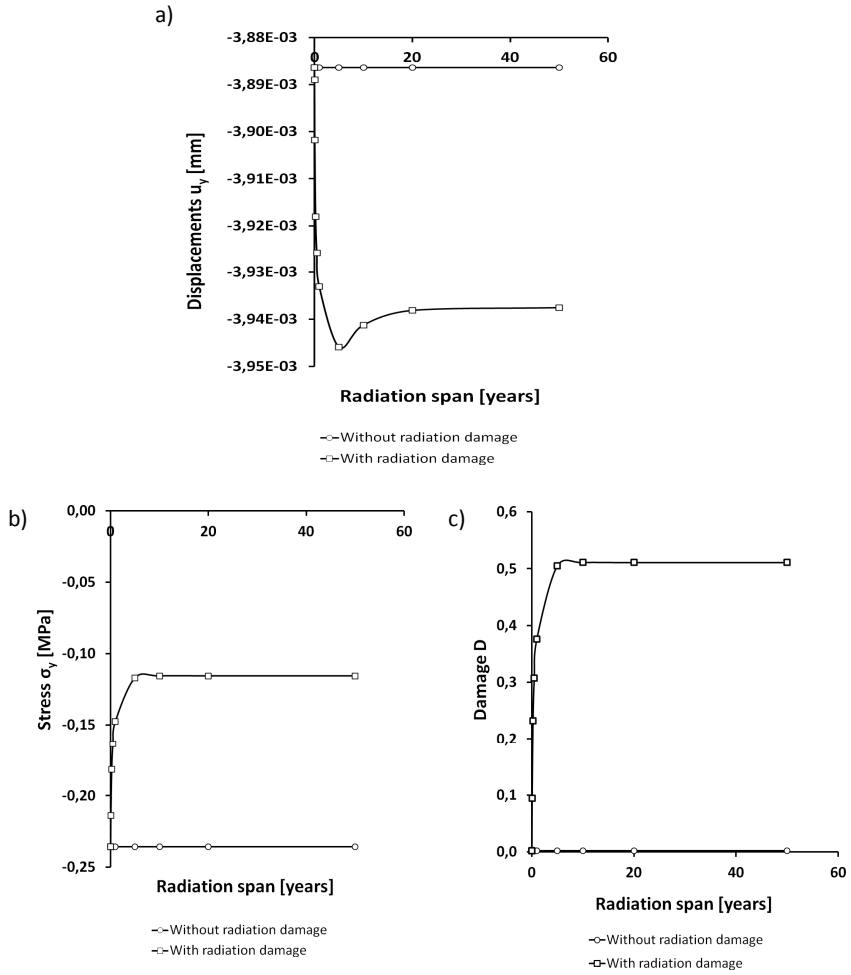

Fig.7.4 – Displacements u_y (**a**), principal stress σ_y (**b**) and total damage D (**c**) as functions of the radiation span on OPC, close to the directly exposed surface, both in case that radiation damage is or not activated.

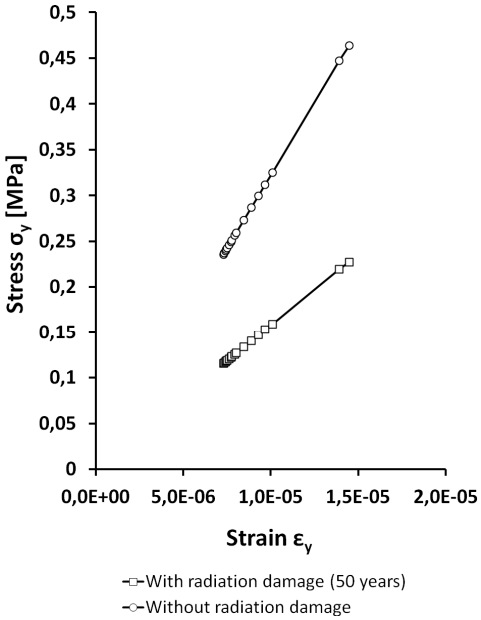

Fig.7.5 – Stress-strain relation along direction *y*, loading direction, for OPC, near the most exposed surface, after a 50 years long radiation span.

The behaviour of damage in function of depth for OPC has been reported in Fig.7.6, which shows the influence of thermal and fast neutrons on the development of damage vs. distance from the directly impinged surface: when fast neutrons are impinging, the damaged thickness is more pronounced; the amount of the damaged thickness induced by fast neutrons is shown to exceed the one induced by thermal neutrons of 80%, after 5 years radiation, and of 70%, at 50 years radiation. Due to the negligible effect of compression from self-weight load, damage can be considered as totally due to radiation, and not also to mechanical effects.

Special concretes (i.e. improved mixtures) have been taken in comparison with OPC, both for fast and thermal neutrons, for one year exposure, in Fig.7.7: all the mixtures show better shielding properties than OPC, particularly magnetite, limonite-steel and ferro-phosphorus for fast neutrons and serpentine, limonite-steel and ferro-phosphorus for thermal neutrons.

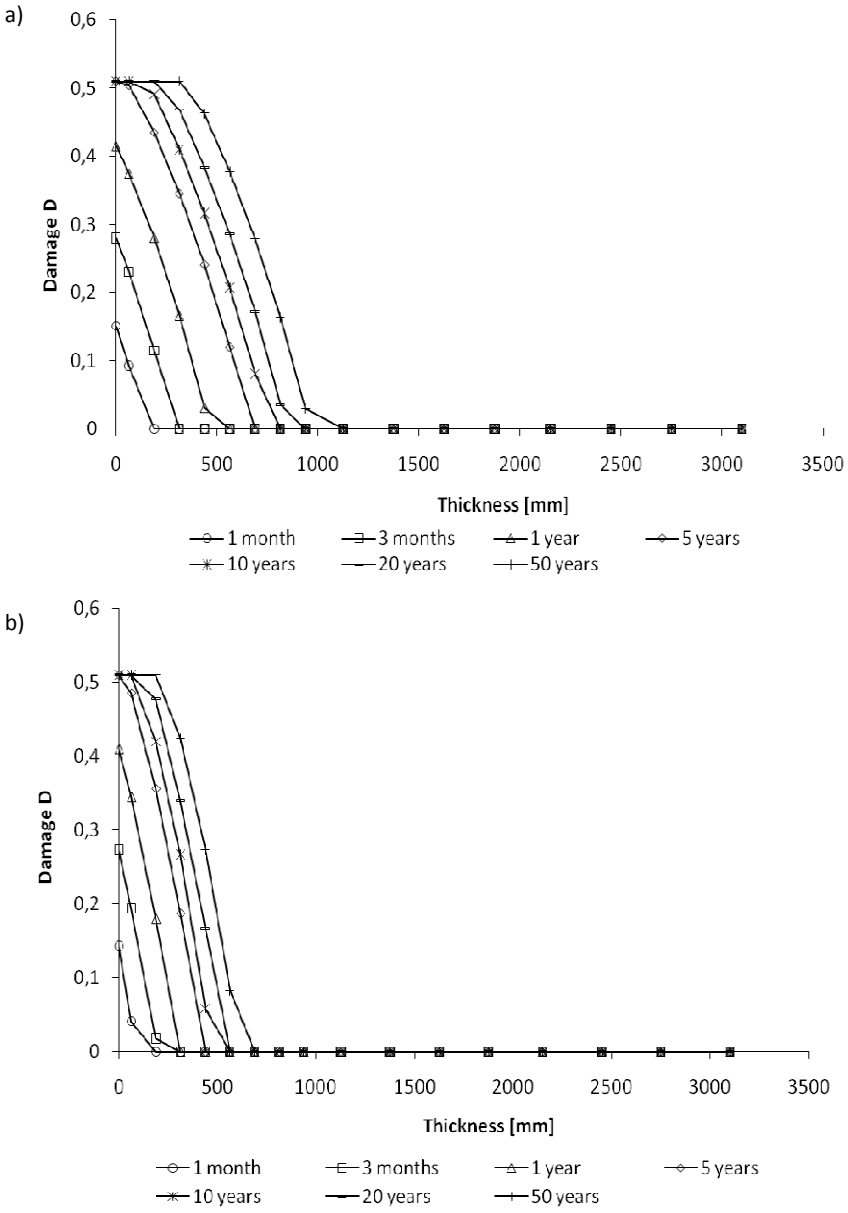

Fig.7.6 – Radiation damage progression with depth of the sample under radiation fluence up to 50 years, from fast neutrons (**a**) and thermal neutrons (**b**) on OPC.

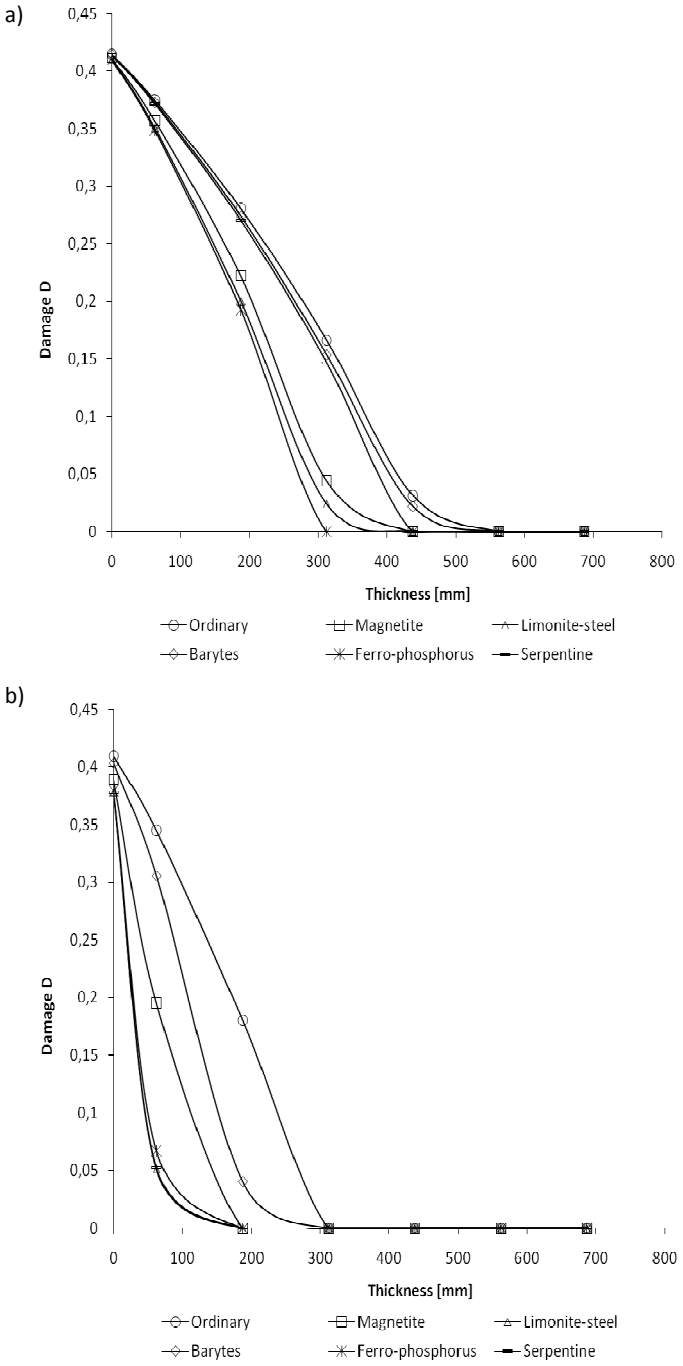

Fig.7.7 – Radiation damage progression with depth of the sample under the radiation fluence of 1 year, from fast neutrons (**a**) and thermal neutrons (**b**) on ordinary and special shielding concretes.

7.3.1 Validation

For validation purposes the results have been additionally compared with the experimental evidences provided by Elleuch et al. (1972) who have studied the behaviour under irradiation

of special shielding concretes; specifically, changes in properties of a serpentine concrete irradiated at fast-neutron fluences of $3 \cdot 10^{19}$ and 10^{20}n/cm^2 have been accounted for. The values of the elastic moduli at different radiation exposures have been estimated in laboratory indirectly, by measuring the propagation times of an ultra-sonic wave longitudinally transmitted on each sample, in order to avoid the definition of the Poisson's coefficient for the irradiated samples.

The authors have found decays in the Young's moduli of about 40% and 45%, for the two fluences respectively, if compared to that of not irradiated control specimens, cured at 20°C and 60% relative humidity.

The test has been numerically reproduced by taking into account a sample made with serpentine concrete, assigning the same neutron fluences at one face, supposing an exponential decay within the thickness, according to Eqs. (18) and (20). The decay in the elastic modulus related to the unirradiated condition is reported in Fig.7.8, referring to a point close to the directly exposed face. The predicted results are in agreement with the experimental test, confirming that the upgrade of NEWCON3D is able to describe the ultimate strength of concrete, when affected by radiation.

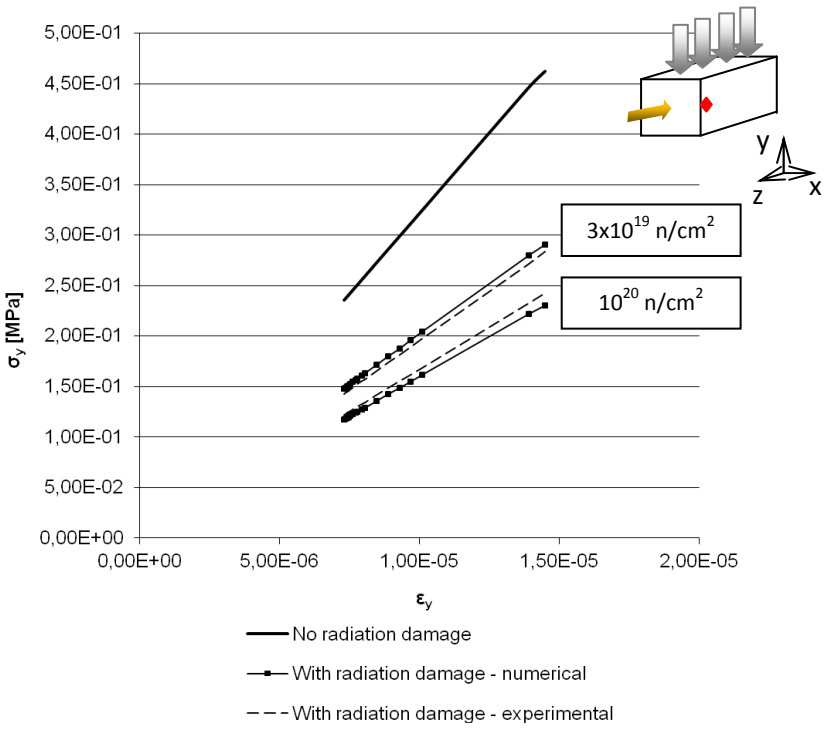

Fig.7.8 – Validation of the numerical model with experimental results by Elleuch et al. (1972).

7.4 Second step analyses: the effect of radiation heat: a combined Monte Carlo – FEM approach

7.4.1 6-month analysis: radiation transport analyses with Fluka

Additional analyses have been held in conjunction with a Monte Carlo code developed by CERN and INFN of Milan, Fluka (Fassò et al. 2005), used to describe the radiation field (neutron fluence and deposited energy) which the mechanical field is dependent on. The objective has been now to take into account the collateral effect represented by the development of heat within the shielding, consequent to absorbed radiation and evaluate the coupled hygro-thermal response of concrete.

The same geometry of Fig.7.3 has been implemented in Fluka, in order to obtain an estimate of the main physical quantities affecting the problem: neutron fluence and energy deposition. The former, which the decay in the Young modulus is function of, is directly responsible for the quantification of the radiation damage; the latter allows for determining the boundary conditions in terms of temperature, at the exposed surface of the wall, in the FEM sub-model.

The geometry in Fluka is assigned by means of elemental volumes (planes, spheres, parallelepipeds, regular prisms, …) and Boolean operations between them (addition, subtraction, union, …) applied to describe “regions” made of different materials. Each portion of space needs to be assigned to one region only.

The concrete room is supposed to be surrounded by “air” which becomes the vehicle for radiation; in defining the geometry in Fluka the presence of an overall surrounding black sphere, which is assigned by default the so called material “blackhole”, is necessary, in order to “close” the mathematical problem of mass balance of particles. The blackhole material, contained in the library of Fluka, is defined as an extremely highly absorbing material, responsible for entrapping all the scattered particles escaping the investigated control volume.

The material “air” is with known chemical composition and it is produced by the already implemented library in the code

As regards concrete, the chemical composition of an ordinary concrete in percentage by weight provided in (Kaplan 1989) has been adopted (Table 7.4).

Table 7.4 – Assumed chemical composition for the ordinary concrete implemented in Fluka (Kaplan 1989).

Element	*% by weight [%]*	*Element*	*% by weight [%]*
Hydrogen	0.64	Phosphorus	0.09
Oxygen	45.36	Sulphur	0.09
Carbon	-	Potassium	0.64
Sodium	1.76	Calcium	12.66
Magnesium	3.66	Titanium	0.47
Aluminium	5.88	Iron	0.13
Silicon	20.90	Nickel	7.64
Concrete density [g/cm^3]		2.33	

The target of fissionable material has been modelled as seven disks (of uranium and carbon compounds) for a total mass of 30g, a radius of 2cm and thickness of about 1.3mm, which makes it an ideal point-source of neutrons.

The impinging beam starts at an arbitrary point inside a vacuum pipe; thanks to the vacuum environment, it is therefore supposed to be delivered not attenuated from the target to the front wall. The characteristics of the primary proton beam come from the most serious exercise scenario designed by INFN for the SPES facility, as reported in Fig.7.2: a beam of 70MeV energy, and 300μA current, so that the proton flux *p* is the following:

$$p = \frac{300 \cdot 10^{-6}}{1.602 \cdot 10^{-19}} = 1.87 \cdot 10^{15} \text{ p/s.} \qquad (114)$$

The stochastic simulation with Fluka required 3 cycles per run, with 10^6 primary particles launched at each run, in order to reach acceptable statistics.

Fluka results are provided on a regular parallelepiped grid, at the centre of mass for each element, the dimensions of which are determined by the required results resolution. A regular grid of small elements 5cm wide along each direction has been adopted for our study. A subsequent interface program has been additionally used to pass results from the centre of mass for each volume of the results' grid to the FEM mesh adopted in NEWCON3D, based on the algorithm of the minimum distance.

The code makes a 3D analysis, though the visualization is two-dimensional.

In Fig.7.9 the map of the neutron flux density for the control volume corresponding to the sub-model analyzed in NEWCON3D is evidenced; the first meter of wall, only, seems to be affected by neutron absorption, with the surplus thickness being necessary for dose transmission issues. These values of neutron flux density have been transferred to the corresponding nodes of the sub-model employed in NEWCON3D (the program is reported in Appendix B).

In Fig.7.10 the results in terms of deposited energy are shown, for a wider investigated area of the shielding. Results refer to prompt radiation, i.e. the constant component of the total amount of deposited energy; they are here intended for primary incident particle (i.e. per proton) and need to be normalized by the factor of Eq. (114), in order to reach total values of power density.

For a better visualization, Fig.7.11 depicts the same values of neutron flux and power density (here normalized), for the sub-model directly exposed at the cutting plane shown in Figs.7.9 and 7.10. The order of magnitude of the maximum neutron flux density is proved to be 10^{10} n/(cm^2 s); the maximum power density is 10^{10} GeV/(cm^3 s).

Neutron flux density, integrated over the time of radiation exposure (thus getting neutron fluence), is intended to enter NEWCON3D as the parameter directly affecting radiation damage; power density has been used to estimate the temperature field in concrete due to heat production by radiation, thus obtaining the proper boundary conditions for temperature to be imposed at the outer face of the sub-model investigated in NEWCON3D (see Appendix B for how the boundary conditions were assigned, numerically). Therefore the temperature rise has been quantified via a transient thermal analysis and then applied as boundary conditions variable in time to the FEM model.

In Fig.7.12 the contour map of temperature for a neutron source working continuously for 6 months is shown.

Fig.7.12 evidences that the maximum temperature is encountered at the corner close to the source, where after 6 months nearly 70°C are reached; in the last picture a time history for the same critical node is reported, which shows a parabolic increase of temperature in time.

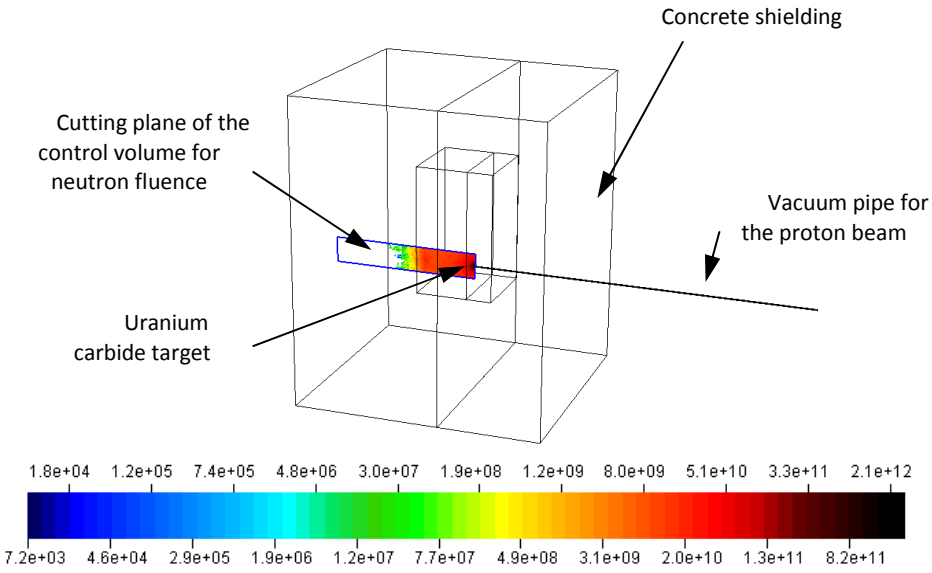

Fig.7.9 – Neutron flux density for the control volume corresponding to the FEM mesh implemented in NEWCON3D. Results from Fluka on the cutting plane located in front of the source [n/(cm^{2}s)].

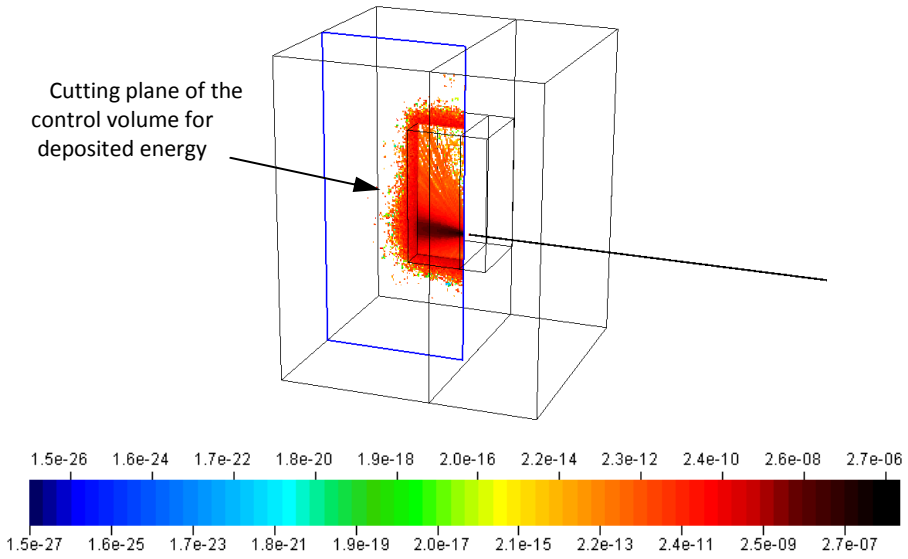

Fig.7.10 – Energy deposition on the directly exposed area of the concrete shielding. Results from Fluka on the cutting plane located in front of the source [GeV/(cm^3 primary)].

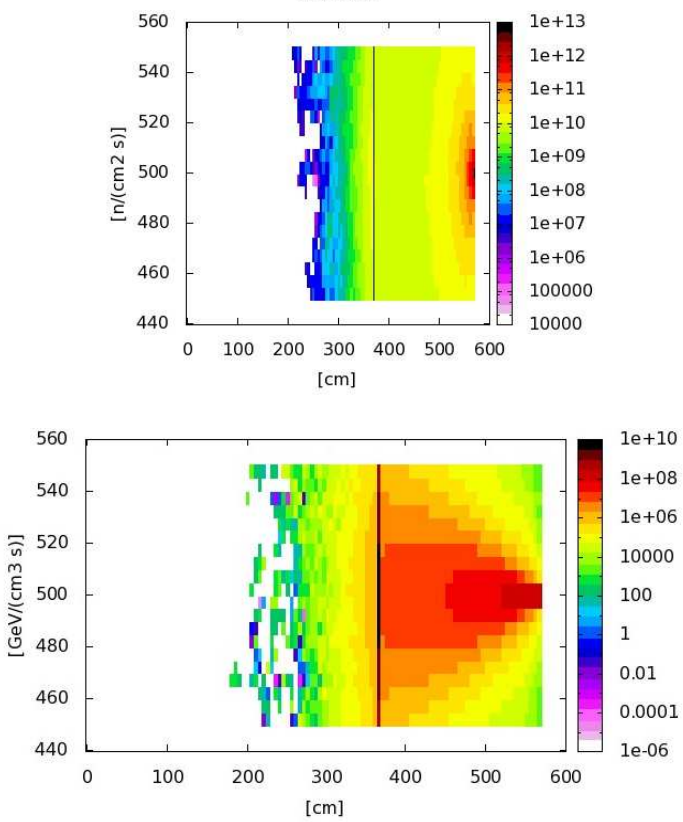

Fig.7.11 – Neutron flux density [n/(cm^2 s)] and power density [GeV/(cm^3 s)] on the cutting plane of the directly exposed control volume.

The effect is shown to be not negligible, which is in agreement with the requirements of ANSI/ANS-6.4-1985, according to which radiation heat and subsequent thermal effects are to be taken into account for energy flux densities (power densities) above 10^{10}MeV/(cm^2 s).

It is to be noticed that Fig.7.11 gives maximum values of the order of magnitude of 10^{10}GeV/(cm^3 s), where cm^3 are to be intended per volume of the small elements defined by the resolution of the result grid (side of the cubes: 5 cm), therefore one gets $5 \cdot 10^{10}$ GeV/(cm^2 s), which is quite above the prescribed limit required to neglect thermal effects.

So thermal effects are expected to be the main responsible for the stress state on concrete and this will be better illustrated in the results obtained from the thermo-hygro-mechanical analysis, in fact.

The thermal analysis justifies also the working period of the facility assumed in the study, i.e. 6 months, nearly 4500 hours per year, longer durations resulting unacceptable for the material, due to high temperature.

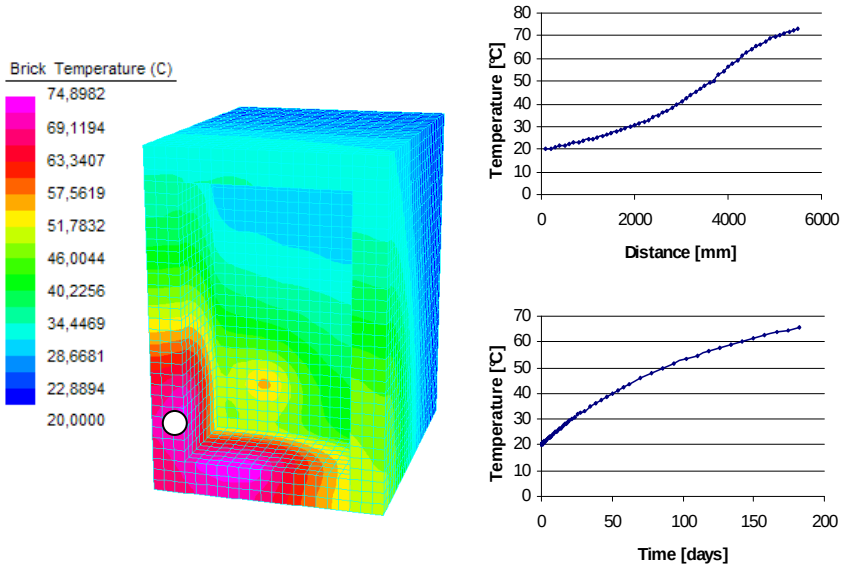

Fig.7.12 – Temperatures on the directly impinged zone after 6 months of work for the facility.

Back-analyses with Fluka have been made in order to study the influence of radiation heat in the variation of water content of concrete: the model implemented in Fluka has been modified taking into account the change in the water content and density of concrete, due to temperature; the change of properties has been made by exploiting the “voxel geometry” in Fluka, a utility which allow to classify a defined region as a “map of density”, an approach that is particularly useful at modelling organ tissues under radiation, from CT scans.

In our case concrete has been supposed to change in density and water content according to the experimental curves of Fig.7.13, provided by Kaplan (1989) for the same adopted concrete mixture reported in Table 7.4.

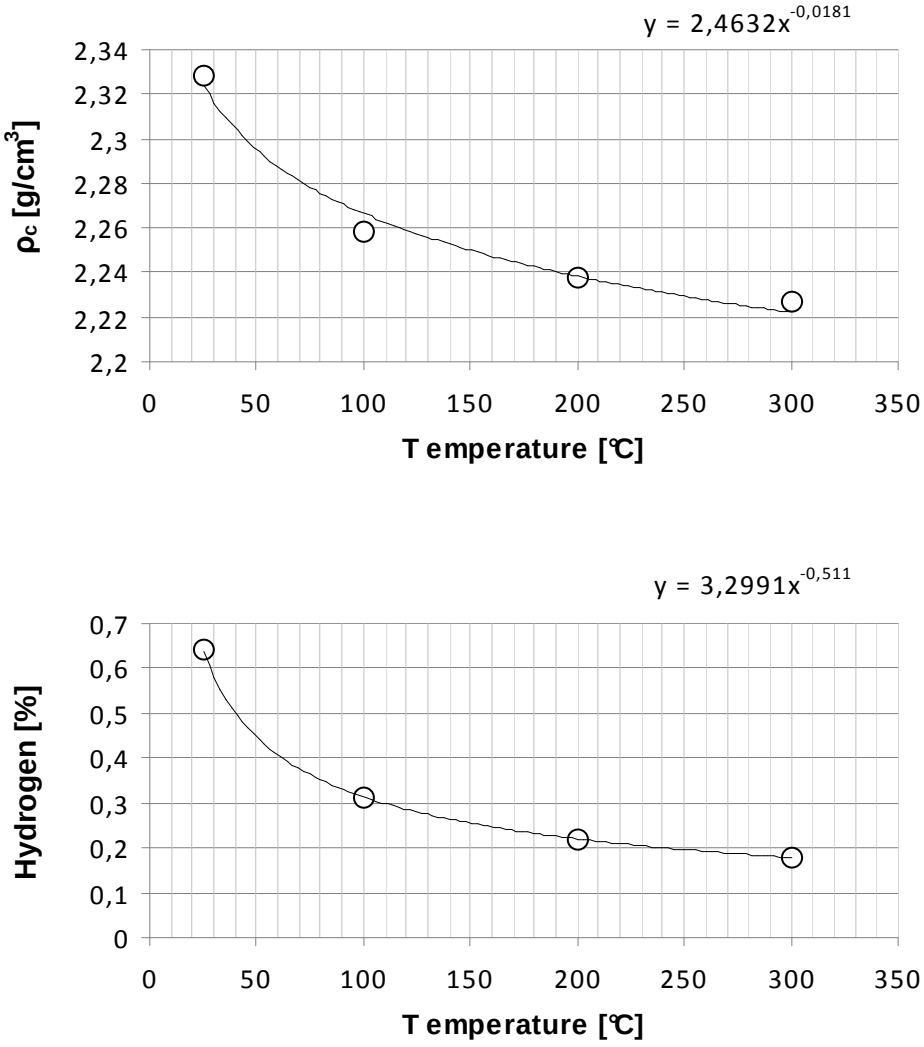

Fig. 7.13 – Experimental curves for density loss and hydrogen content in concrete (Table 7.4) at varying temperature.

A section view of the new Fluka geometry is reported, for several temperature conditions in time, in Fig.7.14: the shielding is now described as a map of regions still made of concrete, but differentiated for its specific weight and chemical composition. The three types of concrete have been defined as new materials in the code, in function of the reached temperatures: they are supposed to have the same chemical composition of the original mixture (see Table 7.4) except for hydrogen content and specific weight, assigned as reported in Table 7.5.

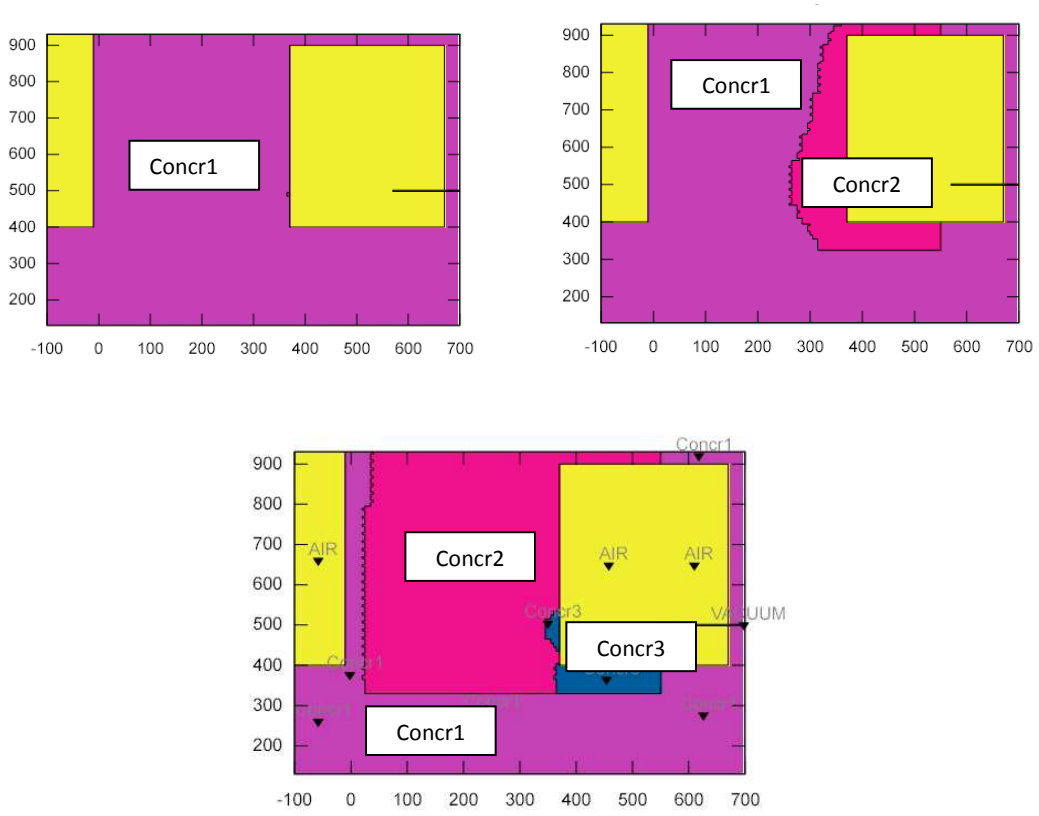

Fig.7.14 – New geometry for the target cave obtained by differentiating concrete properties according to temperature rise: concr1 (violet), concr2 (pink), concr3 (blue).

Table 7.5 – Definition of modified concretes for updating the radiation field.

Concrete	*Temperature range [°C]*	*ρ_c [g/cm^3]*	*H [%]*
Concr1	<20	2.33	0.64
Concr2	20-50	2.31	0.54
Concr3	50-75	2.29	0.40

Slight variations in concrete properties, however, are proved not to bring significant changes in the radiation field, for the loss in hydrogen not justifying a relevant alteration in the shielding properties of concrete, at least for the investigated period of time. As a check, in Fig.7.15 the neutron flux after 6 months and with the new geometry is compared to the same quantity for the original problem, in which no distinction was made on concrete in function of the temperature: the critical value keeps of the order of 10^{10} n/(cm^2 s).

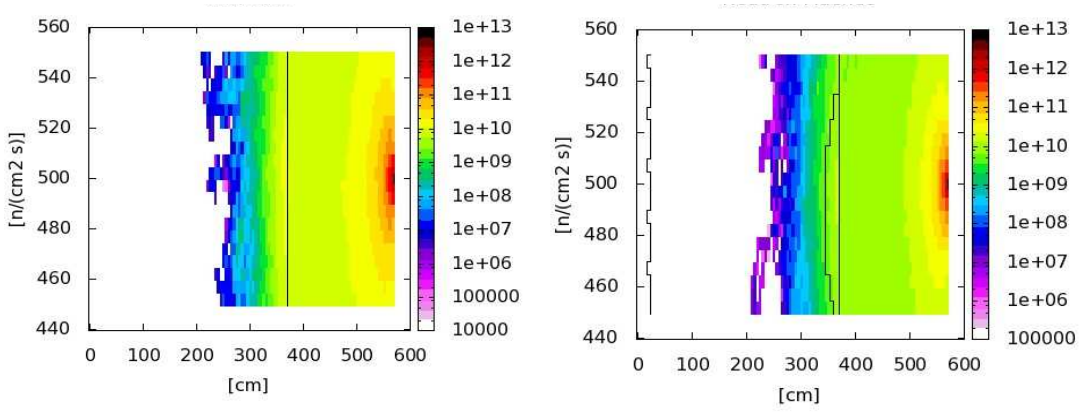

Fig.7.15 – Neutron flux density compared to the one obtained by considering the new geometry, after 6 months.

One can conclude that the resultant deposited energies does not provide significant losses in the shielding properties of concrete; i.e. the slight variations in water content, due to radiation heat, have not shown to affect the moderating capacity of concrete towards neutrons, namely due to the intrinsic hydrogen content given by its water content.

7.4.2 6-month analysis: implementation into NEWCON3D

As already evidenced, the attenuation process for the incident radiation on concrete is a phenomenon that evolves in time: neutron fluence [n/cm^2] is a flux density [n/(cm^2 s)] integrated over the time duration of radiation; instead power density [GeV/(cm^3 s)] varies with the irradiation profile of the facility and, in general, is due to a instantaneous component, so called “prompt radiation”, instantaneously provided when the facility works (zero otherwise), and a “delayed” component represented by the decay of radioactive particles, which is the time-variable amount.

Consequently, the problem is numerically treated as a time dependent process where radiation damage and temperatures at the boundary needs to be calculated for several time steps.

The sub-model analyzed in NEWCON3D has been descritezed through 800 elements (4037 nodes); 20-node isoparametric brick elements have been applied (Figure 4).

Boundary conditions in terms of temperature have interested the front free surface.

Initial conditions are homogeneous and consider an internal relative humidity of 60% and a temperature of 20°C.

The coupling between the hygral, the thermal and the mechanical field provided by NEWCON3D has allowed to catch the variation in time for several variables (Figs.7.16-7.18; only a portion of the model is here shown): relative humidity seems not to be much affected by the 6-month prolonged radiation, whereas the temperature rise is understood to be of interest, leading to thermal gradients up to 50°C (at the impinged area); for this reason the irradiation profile for the SPES facility should not exceed 4500 hours/year, i.e. approximately 6 months of continuous service at 70MeV, 300μA.

As regards damage, it is intended to be due only to thermo-chemical effects, since the neutron fluence, for the investigated time span, always stays under the critical value of 10^{19}n/cm^2, which is expected to mark the beginning of first evidences of concrete damage by radiation; therefore the thermal aspect of the problem, once more, represents the most restrictive condition to prescribe a safe service period for the facility.

In Fig.7.19 relative humidity, temperature and longitudinal displacements are plotted, along the prism central axis and a parallel line at its border.

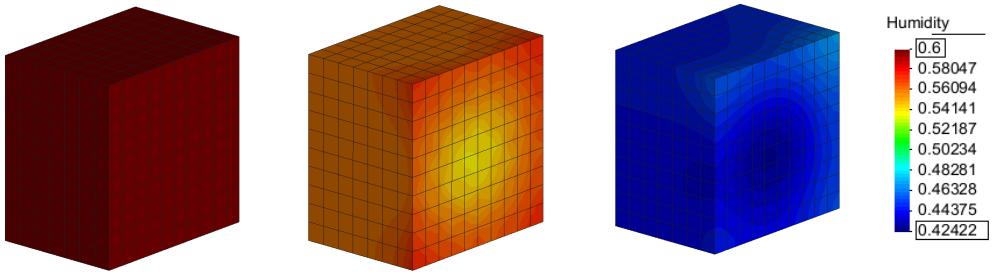

Fig.7.16 – Contour maps of relative humidity after 1 hour, 1 week, 6 months [-].

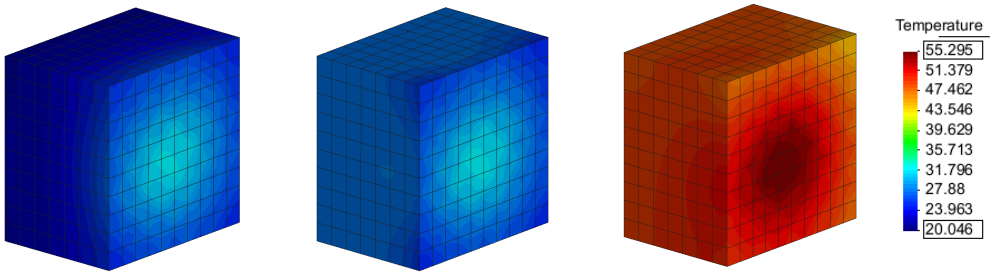

Fig.7.17 – Contour maps of temperature after 1 hour, 1 week, 6 months [°C].

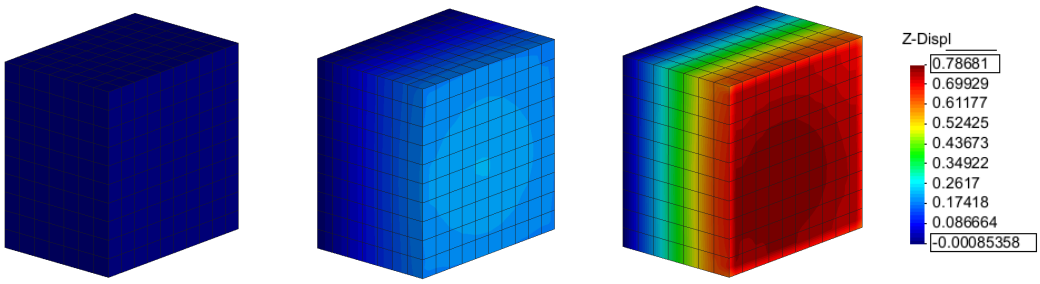

Fig7.18 – Contour maps of displacements along the prism length after 1 hour, 1 week, 6 months [mm].

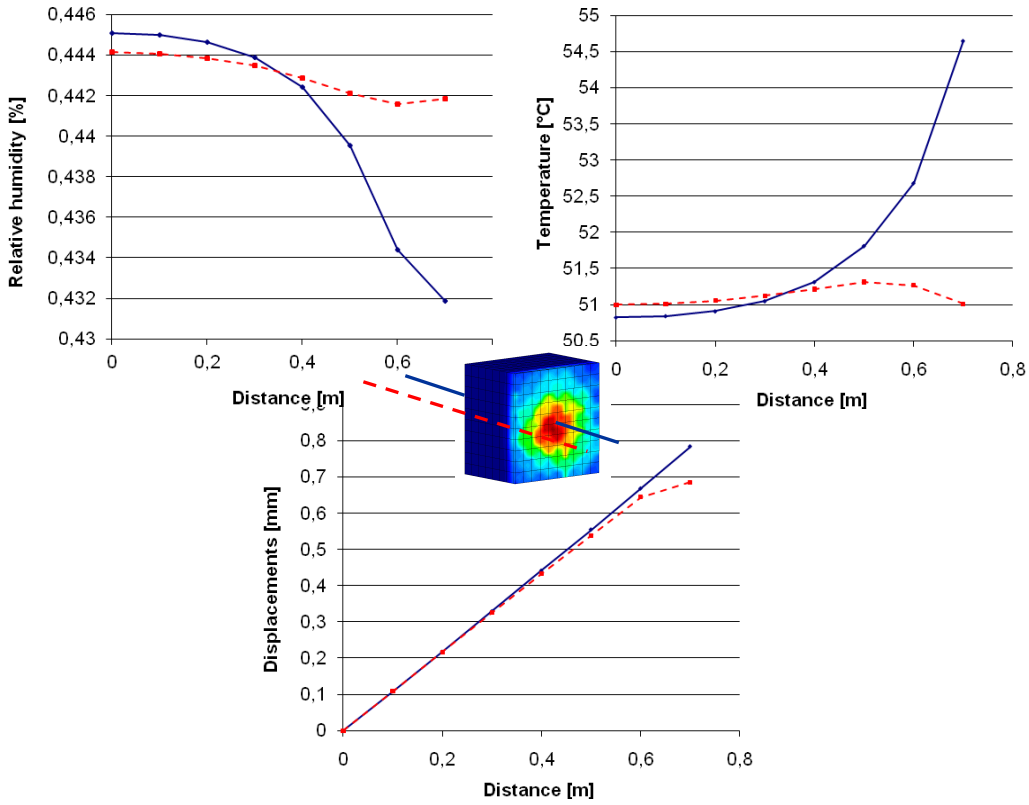

Fig.7.19 – Distribution of the main variables after 6 months along the first 0.7m wall thickness (reference lines: prism central axis and parallel axis at the border).

7.4.3 5-year analysis

After the temperature effect has been envisaged not to be negligible and a trial continuous working period of 4500hours/year has been assessed to lead to a temperature rise in concrete of some interest, a cyclic irradiation profile has been assumed for the facility.

Each irradiation cycle is made of a continuous working period of 5000hours/year (7 months), almost equal to the one previously investigated; in this period both prompt radiation and delayed radiation due to radioactive decay are to produce radiation heat in concrete; the remaining hours are supposed to be of rest for the facility so that the only source responsible for heat development within the shielding is radioactive decay of the activated species in concrete.

The annual cycle just described is repeated for a five-year period.

The first step was to consider how the temperature field evolves on the overall structure of the target cave, due to the irradiation profile. At this purpose Fluka results in terms of deposited energy have been elaborated with a proper program developed in Fortran language (see Appendix D), in order to get boundary conditions assigned in terms of:

- thermal fluxes at the inner faces of the target room
- heat sources for the internal elements

for a thermal transient analysis, which has been held with the FEM commercial code Straus7.

The program provided in Appendix D basically takes the values from Fluka, assigns them to the nodes of the FEM mesh of the target bunker and averages them on the different exposed faces indicated in Fig.7.20 (2b, 3b, 3b1, 3b2, 4b, 5b, 6b for heat fluxes) or inner elements (s1, s2 for heat sources). In particular the front face has been differentiated into 3 zones: 3b, 3b1, 3b2 to better catch the “hot spot” opposite to the target; similarly source elements have been averaged on two zones: s1 and s2.

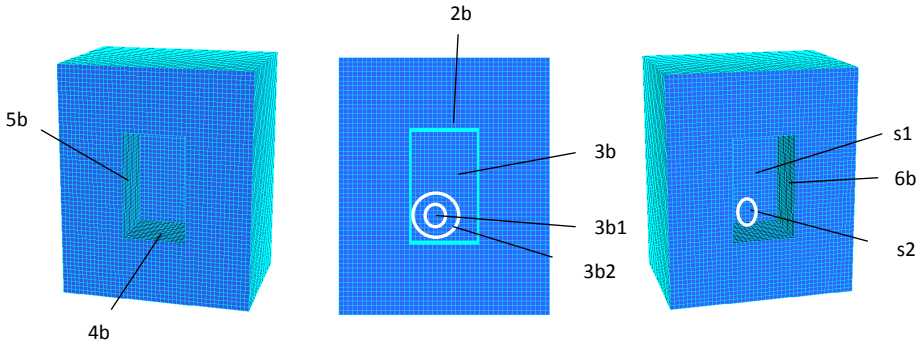

Fig.7.20 – Averaging areas for heat fluxes and heat sources calculations.

The assigned profiles in time for fluxes and sources, after the averaging process over the previously defined zones, are reported in Fig.7.21.

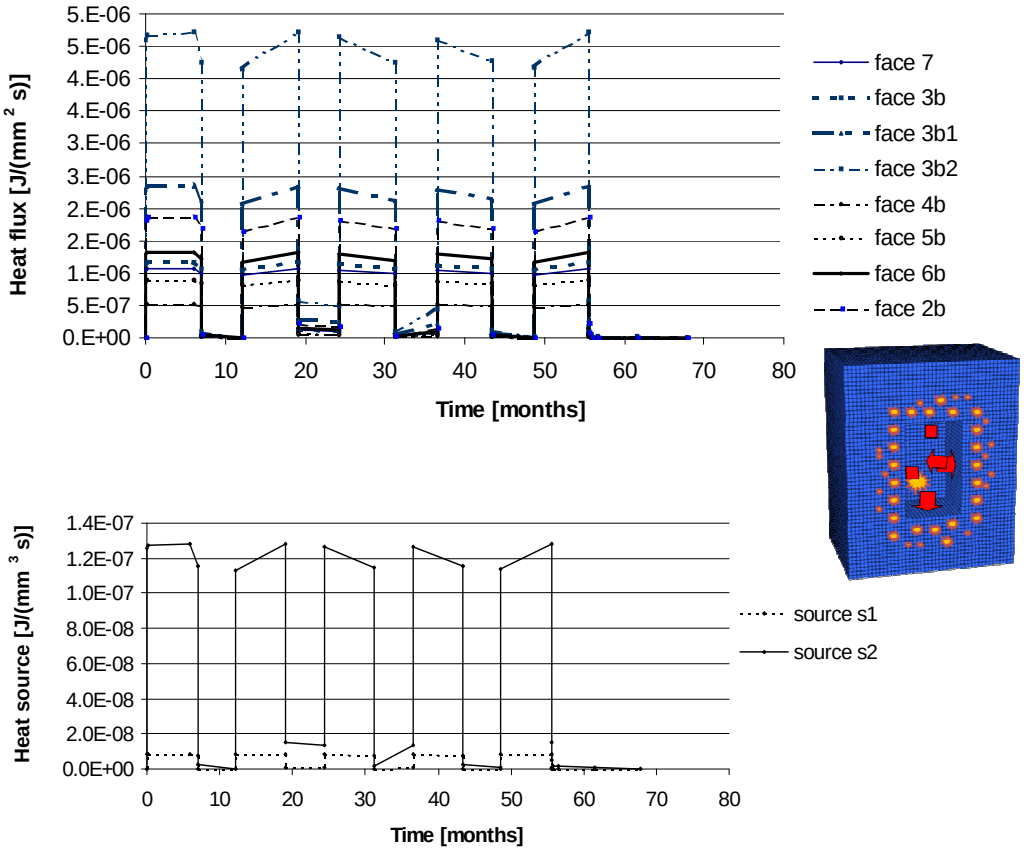

Fig.7.21 – Boundary conditions for the thermal transient analysis from Fluka results for a 5-year irradiation profile.

In Fig.7.21 it is remarkable that, even after the averaging process of Fluka deposited energy values, the behaviour in time for fluxes and sources mirrors the cyclic working of the facility, considering that the total deposited energy is the sum of energy due to prompt radiation (constant in time, zero when the facility is off) and energy due to decay radiation (variable in time, always present); the latter depends on the implemented irradiation profile in Fluka and is clearly of a lower order of magnitude than the former. In Appendix C is the program performing the superposition of effects for the total deposited energy values.

The results in terms of temperature due to radiation heat provided by the thermal analysis for a 5-year cyclic working period are reported in Fig.7.22 for three detection points on the structure.

The system is found to stabilize at 50°C after 5 irradiation cycles of 5000hours/year each.

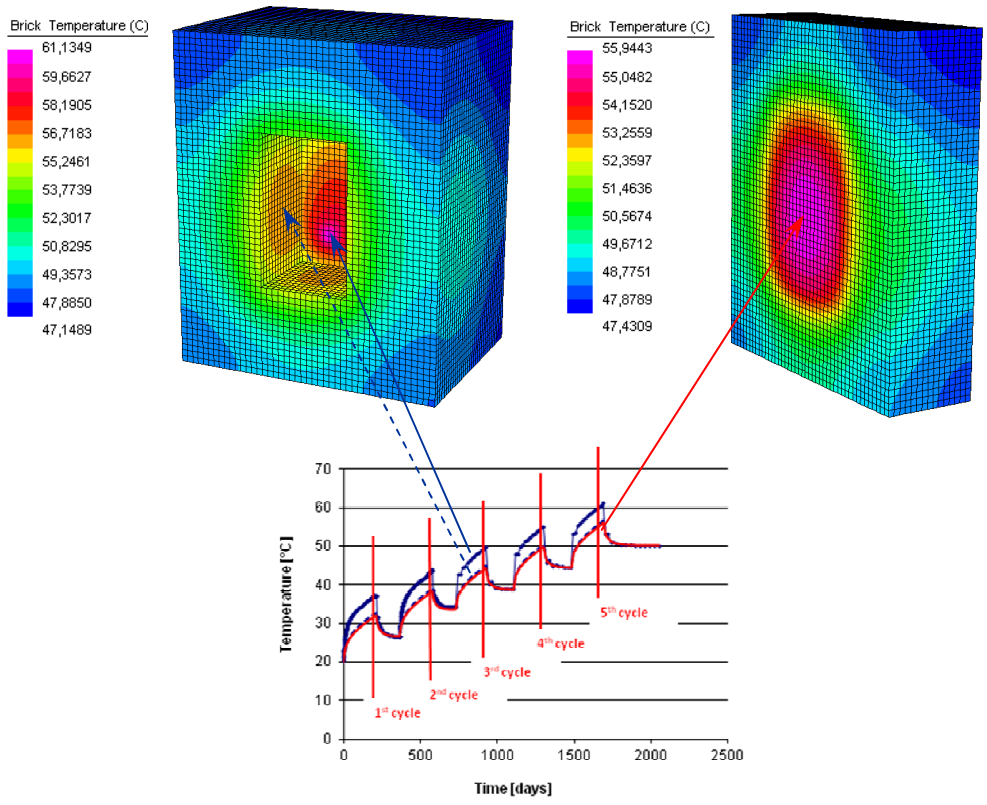

Fig.7.22 – Contours after the fifth irradiation cycle and evolution of temperature at three points of the target bunker.

The hygro-thermo-mechanical coupling effects have been investigated with NEWCON3D, as done in the previous simulations, for the first 1.5m slice of wall opposite to the target source (1m^2 of cross-sectional area, again) and the results for the main variables are in Figs.7.23, 7.24 and 7.25.

The temperature is shown to be maximum at the boundary surface, where it reaches 60°C after the fifth cycle. Humidity, coherently, is maximum at the same point, where 60% was assigned as boundary condition.

The variation through the investigated thickness is negligible.

For displacements a similar cyclic behaviour is envisaged. The free face shows the maximum displacement; all other faces are endowed with translational restraints accounting for symmetry.

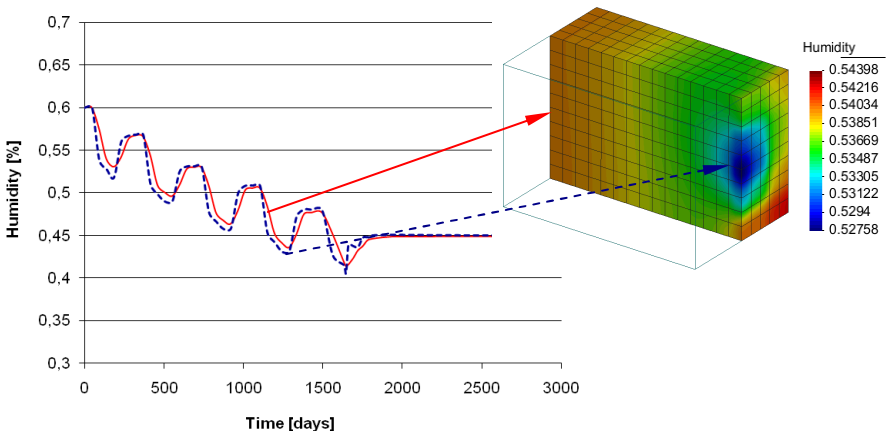

Fig.7.23 – Distribution of humidity in a slice of wall opposite to the target.

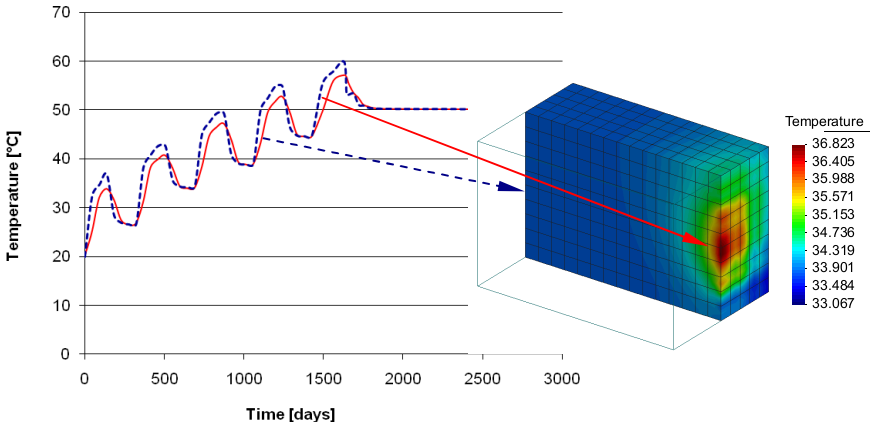

Fig.7.24 – Distribution of temperature in a slice of wall opposite to the target.

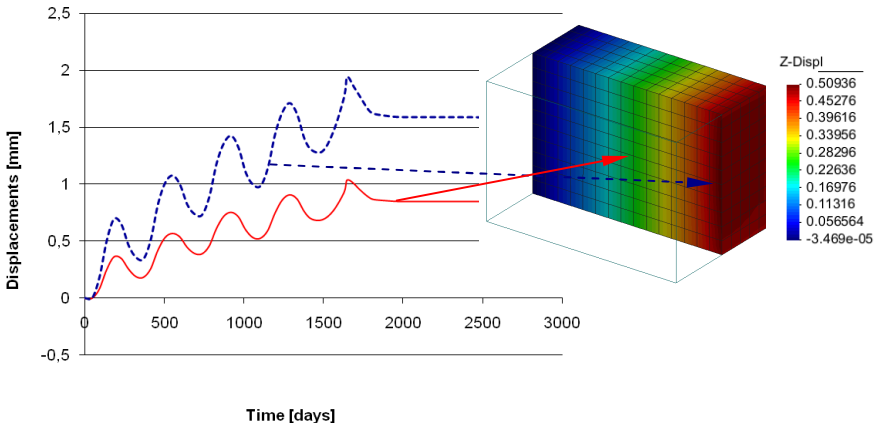

Fig.7.25 – Distribution of longitudinal displacements in a slice of wall opposite to the target.

References to Chapter 7

[1] American National Standard for guidelines on the nuclear analysis and design of concrete radiation shielding for nuclear power plants, ANSI/ANS-6.4-1985, American Nuclear Society, Illinois; cited in: Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design, and construction, John Wiley & Sons, New York, 1989.

[2] Bazant Z.P., Osman E., Double power law for basic creep of concrete, Matériaux et constructions (RILEM, Paris) 49(9), 3-11, 1976.

[3] Circular 02-02-2009, n. 617 – Instructions for the application of the "New technical standard for constructions" – Ministerial Decree 14-01-2008, 2009 (in Italian).

[4] Davis H.S., Browne F.L., Witter H.C., Properties of high-density concrete made with iron aggregate, J Amer Concr Inst 52(3), 705-726, 1956.

[5] Elleuch M.F., Dubois F., Rappeneau J., Effects of neutron radiation on special concretes and their components, Amer Concr Inst Special Publication SP-34: Concrete for nuclear reactors, Paper SP34-51, 1071-1108, 1972.

[6] Fassò A., Ferrari A., Ranft J., Sala P.R. FLUKA: a multi-particle transport code. CERN-2005-10, INFN/TC_05/11, SLAC-R-773, 2005.

[7] Gallaher R.B., Kitzes A.S., Summary report on Portland cement concretes for shielding. USAEC Report ORNL-1414, 1953.

[8] INFN-LNL-224, SPES Selective Production of Exotic Species: Executive Summary. A. Covello, G. Prete Editors, Laboratori Nazionali di Legnaro, 2008.

[9] Kaplan M.F., Concrete radiation shielding: nuclear physics, concrete properties, design and construction. John Wiley & Sons, New York, 1989.

[10] Mazars J., Application de la mecanique de l'endommagement au comportament non lineaire et la rupture du beton de structure. Ph.D. dissertation, L.M.T., Université de Paris, France, 1984.

[11] Mazars J., Description of the behaviour of composite concretes under complex loadings through continuum damage mechanics. In: Proc. 10th U.S. National Congress of Applied Mech, ASME. Lamb J.P. Ed., 1989.

[12] Ministerial Decree 14-01-2008 – "New technical standard for constructions" (in Italian), 2008.

[13] Ohgishi S., Miyasaka S., Chida J., On properties of magnetite and serpentine concrete at elevated temperatures for nuclear reactor shielding, Amer Concr Inst Special Publication SP-34: Concrete for nuclear reactors, Paper SP34-57, 1243-53, 1972.

Conclusions

In this study a combined approach Monte Carlo – Finite Element is suggested to analyze the physical phenomenon of radiation interaction with concrete as shielding material for nuclear facilities. Collateral thermal effects and subsequent water loss, due to heat production by absorbed radiation, can be of interest, thus requiring a fully coupled hygro-thermo-mechanical approach for this kind of problems, to better catch the response of the shield to a radiation exposure.

After a collection of the most relevant experimental results on neutron irradiated concrete, a formulation for radiation damage in the context of damage mechanics for concrete materials was possible, within a FEM research code assessing the coupled hygro-thermo-mechanical behaviour of concrete. Therefore total damage becomes, here, the result of a multiplicative relationship, accounting for radiation, together with thermo-chemo-mechanical damage.

The numerical analyses have been directed to plan a safe work scenario for the SPES Project, an ongoing research project at INFN Laboratories in Legnaro (Padova, Italy) dedicated at the production of a special type of neutron-rich nuclei, with possible future applications in medicine.

The machine-specific characteristics of the facility have been implemented in a Monte Carlo code performing radiation transport calculations, in order to define the radiation field for the concrete shield, under both a continuous exposure and a cyclic work of the facility.

The combined Monte Carlo – FEM approach allowed us to identify in 4500-5000 hours per year, i.e. 6-7 months, the maximum time span for a continuous work of SPES, in order to stay within admissible thermal gradients for the material, provided no form of primary barrier or cooling is present.

Thermal effects due to radiation are found to be the most critical aspect to prescribe the duration of the exercise life of the facility, in fact, rather than the expected neutron flux.

Appendix A

```
C   MODIFICHE POMARO '09
        XCORD(MASH) = 0.D0
        YCORD(MASH) = 0.D0
        ZCORD(MASH) = 0.D0
        FLUG(MASH) = 0.D0
        DANFLUGE(MASH) = 0.D0
        DANFLUGC(MASH) = 0.D0
        DANFLUGT(MASH) = 0.D0
             DO 999  I = 1,NCN
                J     =  NOP(NEL,I)
               XCORD(MASH) =  XCORD(MASH) + PDUM(I,IG) * CORD(J,1)
                YCORD(MASH) =  YCORD(MASH) + PDUM(I,IG) * CORD(J,2)
                ZCORD(MASH) =  ZCORD(MASH) + PDUM(I,IG) * CORD(J,3)
                FLUG(MASH) = FLUG(MASH) + PDUM(I,IG) * FLU(J)

             IF (FLUG(MASH) .LE. 1.D18)  THEN
                DANFLUGE(MASH) = 0.D0
                ELSE
                DANFLUGE(MASH) = 1.D0 - 483.59D0 * FLUG(MASH)**(-0.15D0)
                END IF
             IF (FLUG(MASH) .GE. 1.D20)  DANFLUGE(MASH) = 1.D0 - 0.49D0
  999        CONTINUE
C   FINE MODIFICHE POMARO
C
C                  *********  CALCOLO  DEL DANNO  ****
C
C ***   DANNO ALLA MAZARS
C
 5000            TDAN     =  0.D0
                 CDAN     =  0.D0
                 DAN      =  0.D0
C   MODIFICHE POMARO '09
                 DANR     =  0.D0
C   FINE MODIFICHE POMARO
             TDAN = FDAN(XK0,AT,BT,CEPSEQ)
             CDAN = FDAN(XK0,AC,BC,CEPSEQ)
C
C ***   DANNO ALLA KACHANOV
C
         TDAN = TDAN + DANKT(MASH)
         CDAN = CDAN + DANKC(MASH)
             IF (TDAN .GT. 1.D0)     TDAN = 1.D0
             IF (CDAN .GT. 1.D0)     CDAN = 1.D0
C
C ***     DANNO COMPLESSIVO
C
C   MODIFICHE POMARO '09
        IF (KDANF. EQ. 1) THEN
             DANR  =  DANFLUGE(MASH)
        ELSE
             DANR  =  0.D0
        ENDIF
C   FINE MODIFICHE POMARO

             DAN  =  (TALFA(MASH) * TDAN + CALFA(MASH) * CDAN)
             DUM = 1.D0 - 1.D-30
             IF (DAN .GE. DUM)  DAN = DUM
C   MODIFICHE POMARO '09
             IF (KDANF. EQ. 1. AND. DANR .GE. DUM)  DANR = DUM
C   FINE MODIFICHE POMARO
             IF (DAN .LT. DVET(MASH))     DAN = DVET(MASH)
             ENER  =  ENER - ESUM(MASH) * (DAN - DVET(MASH))
             TENER    =  TENER + ESUM(MASH)
C   MODIFICHE POMARO '09
C    IF(KDANF. EQ. 1. AND.DANR .LT. DVETR(MASH)) DANR = DVETR(MASH)
C             ENER     =  ENER - ESUM(MASH) * (DANR - DVETR(MASH))
C             TENER    =  TENER + ESUM(MASH)
C   FINE MODIFICHE POMARO
C   MODIFICHE POMARO '09
      IF (KDANF. EQ. 1) THEN
         DCOEF = (1.D0 - DVET(MASH)) * (1.D0 - DVETR(MASH))
      ELSE
         DCOEF = 1.D0 - DVET(MASH)
```

```
      ENDIF
C   FINE MODIFICHE POMARO

              ST(IG,1) = STD1(MASH) * DCOEF
              ST(IG,2) = STD2(MASH) * DCOEF
              ST(IG,3) = STD3(MASH) * DCOEF
              ST(IG,4) = STD4(MASH) * DCOEF
              ST(IG,5) = STD5(MASH) * DCOEF
              ST(IG,6) = STD6(MASH) * DCOEF
              ST(IG,7) = STD7(MASH) * DCOEF
              ST(IG,8) = STD8(MASH) * DCOEF
              ST(IG,9) = STD9(MASH) * DCOEF
            STD1(MASH) = CEPSEQK
            DV(MASH) = DCOEF
              DVET(MASH) = DAN
C   MODIFICHE POMARO '09
              IF (KDANF .EQ. 1)  DVETR(MASH) = DANR
C   FINE MODIFICHE POMARO
```

Appendix B

```
program FlukaStraus

! Prende le coordinate nodali dal file di Straus e,tramite la subroutine readData: a) le formatta,
    b) crea e formatta le coordinate baricentriche di Fluka e "vede" solo i valori di fluence da
    USRBIN al "concrete";
! tramite la subroutine trasporto riporta ai nodi di Straus i valori di fluence ai baricentri della
    mesh di Fluka.


implicit none
integer::  unit1, unit2, unit3,unit4,unit5, unit6, unit7, unit8, unit9, unit10


! totI = Numero di volumetti che danno le dimensioni della griglia di USRBIN nelle direzioni
    cartesiane
! totIHole = Numero di volumetti che danno le dimensioni del vano interno allo shielding
! x,y,z = Indice di ordine del volumetto inizio dell'hole
! nnodi = Numero di nodi della mesh Straus
! nvol = Numero di volumetti della griglia USRBIN di solo cls
! dati(i) = Fluence ai baricentri USRBIN di solo "concrete" (da value(i) in da Fluka.dat)
! iB(i) = Coordinate baricentri griglia USRBIN
! A(i,j) = Coordinate nodi Straus
! D = Dimensione lato griglia USRBIN in cm
! L = Dimensione lato mesh Straus in mm
! dist = Distanza nodo Straus-baricentro griglia USRBIN
! fluence(i) = Fluence ai nodi di Straus al solo "concrete"


integer                             ::  totX,totY,totZ, totXd,totYd,totZd
integer                             ::  totXHole,totYHole,totZHole, totXHoled,totYHoled,totZHoled
integer                             ::  x,y,z,xd,yd,zd

!---------------------------------
integer, parameter                 ::  nnodi = 4037
integer, parameter                 ::  nnodid = 64194
integer, parameter                 ::  nvol = 29600
integer, parameter                 ::  nvoldep = 4752000
!---------------------------------
integer,dimension(:),allocatable   ::  ro
real*8,dimension(:),allocatable    ::  dati
real*8,dimension(:),allocatable    ::  xB,yB,zB
real*8,dimension(:),allocatable    ::  xBd,yBd,zBd
real*8,dimension(nnodi,3)          ::  A
real*8,dimension(nnodid,3)         ::  Ad
real*8,dimension(nnodid)           ::  tempMODEL
real*8,dimension(nvoldep)          ::  tempmod

real*8                             ::  D
real*8                             ::  L

real*8                             ::  dist
real*8,dimension(nnodi)            ::  fluence

integer                            ::  i,j

    !-------------------
    unit1        = 9
    unit2        =10
    unit3        =11
    unit4        =12
    unit5        =13
    unit6        =14
    unit7        =15
    unit8        =16
    unit9        =17
    unit10       =18

!   VALORI PER GRIGLIA USRBIN CON D=5cm E DIMENSIONI REALI (3,7m) DA FlukaB.ris
!
    totX     = 20
    totY     = 20
    totZ     = 114

    totXHole = 20
    totYHole = 20
    totZHole = 40

    x        = 0
    y        = 1
    z        = 75

    totXd     = 180
    totYd     = 240
```

```
    totZd     = 110

    totXHoled = 60
    totYHoled = 100
    totZHoled = 36

    xd        = 60
    yd        = 81
    zd        = 75

    !-------------------

! Dimensione dati(i), iB(i) pari a numero volumetti di solo "concrete"

    allocate(dati(totX*totY*totZ-totXHole*totYHole*totZHole))
    allocate(xB(totX*totY*totZ-totXHole*totYHole*totZHole))
    allocate(yB(totX*totY*totZ-totXHole*totYHole*totZHole))
    allocate(zB(totX*totY*totZ-totXHole*totYHole*totZHole))
    allocate(ro(totXd*totYd*totZd))
    allocate(xBd(nvoldep))
    allocate(yBd(nvoldep))
    allocate(zBd(nvoldep))


    dati=0.d0

    open(unit3, file='FlukaBprova.dat',status='unknown')
    open(unit4, file='StrausNprova.dat',status='unknown')
    open(unit5, file='fluence.ris',status='unknown')
    open(unit6, file='ascii_shi',status='unknown')
    open(unit8, file='FlukaBdprova.dat',status='unknown')

! Richiamo subroutine readData: a) le formatta, b) crea e formatta le coordinate baricentriche di
    Fluka e "vede" solo i valori
! di fluence da USRBIN al "concrete"

    call readData(unit1, unit2,               &
                  totX      , totY   , totZ   ,&
                  totXHole  ,totYHole,totZHole,&
                  x         ,y       ,z       ,&
                  size(dati),dati, xB,yB,zB,A)

    do i=1,size(dati)
        write(unit3,999)i,xB(i),yB(i),zB(i),dati(i)
    end do

    do i=1,nnodi
        write (unit4,99) i,(A(i,j), j=1,3)
    end do

! Richiamo tramite la subroutine trasporto riporta ai nodi di Straus i valori di fluence ai
    baricentri della mesh di Fluka


    call trasporto (unit3,unit4,size(dati),dati,xB,yB,zB,L,A,dist, fluence)

    do i=1,nnodi
        write(unit5,9) fluence(i)
    end do


    call density(unit7,  unit4, unit9, unit10,    &
                 totXd    , totYd   , totZd    ,           &
                 size(ro), ro,xBd,yBd,zBd,L,Ad,tempmod)


    do i=1,nvoldep
        write(unit8,999)i,xBd(i),yBd(i),zBd(i),tempmod(i)
    end do

! Stampa coordinate nodali per NEWCON3D

    write(unit5,997)
    do i=1,nnodi
        write(unit5,997) i,(A(i,j),j=1,3)
    end do

! Stampa condizioni iniziali per NEWCON3D
    write(unit5,997)
    do i=1,nnodi
        write(unit5,998) i
    end do

! Stampa file ascii per fluka con voxel

    do i=1,nvoldep
        write(unit6,995) ro(i)
    end do
    write(unit6,995) (ro(i), i=1,nvoldep)
    close(unit3)
    close(unit4)
    close(unit5)
    close(unit6)
```

```
pause
stop

999 format(i10,4(E15.4))
 99 format(i10,3(E15.4))

  9 format(E15.4)
995 FORMAT(4752000I2)
996 format(i10,4(E15.4))
997 FORMAT  (i10,f12.2,2f10.2)
998 FORMAT  (i10,'    0.0000    0.0000    0.0000    0.1000   20.0000')

end program  FlukaStraus

-------------------------------------------------------------------------------------

subroutine readData(unit1, unit2,       &
                    totX    , totY   , totZ   ,&
                    totXHole,totYHole,totZHole,&
                    x       ,y       ,z       ,&
                    nvol ,dati,xB,yB,zB, A)

implicit none
integer                 ::  unit1, unit2

integer                 ::  totX,totY,totZ
integer                 ::  totXHole,totYHole,totZHole
integer                 ::  x,y,z
integer                 ::  nvol

!---------------------------------
integer, parameter      ::  nnodi=4037

! c deriva da moltiplicare i risultati Fluka in n/(cm2 primary) per i protoni/s dati dal macchinario
    a 70MeV, 300microA;
! se voglio n/cm2 moltiplico per i s: 60*60*5000*50 se lavora a 5000hours/y per 50anni
  real*8, parameter     ::  c=1.87E+15*3600.0d0*24.0d0*30.d0*6.d0

!---------------------------------

real*8,dimension(nvol)   ::  dati
real*8,dimension(nvol)   ::  xB,yB,zB
real*8                   ::  xstart,ystart,zstart, D
real*8,dimension(nnodi,3)::  A

! dimRow = Dimensione plottaggio fluence da Fluka: 10 righe
! value (i) = Fluence da USRBIN, dimensione 10
! limitI = Indice di ordine dei volumetti limite per l'hole

integer, parameter      ::  dimRow = 10
real*8,dimension(dimRow) ::  value
integer,dimension(2)    ::  limitX, limitY, limitZ
integer                 ::  posX,posY,posZ

integer                 ::  i,j,k
integer                 ::  cont

    open(unit1, file='Copia di da Fluka.dat',status='unknown',err=998)
    open(unit2, file='Copia di da Straus txt.dat',status='old')

! Definizione limiti dell'hole

    limitX(1) = x
    limitX(2) = x+totXHole
    limitY(1) = y
    limitY(2) = y+totYHole-1
    limitZ(1) = z
    limitZ(2) = z+totZHole-1

    write(*,900)limitX(1),limitX(2),              &
                limitY(1),limitY(2),              &
                limitZ(1),limitZ(2)

! Inizializzazione variabili contatore

    cont      = 0
    posX      = 0
    posY      = 1
    posZ      = 1
    k         = 1

!----------------------
!   VALORI PER GRIGLIA USRBIN CON D=5cm E DIMENSIONI REALI (3,7m) DA FlukaB.ris
!
    xstart    =2.5d0
    ystart    =2.5d0
    zstart    =2.5d0
    D = 5.d0

!-----------------------

    xB = 0.d0
    yB = 0.d0
```

```
    zB = 0.d0

! Lettura nodi mesh Straus da da Straus.dat per formattazione nel main

      do i=1,nnodi
      read(unit2,995) (A(i,j), j=1,3)
      end do

! Lettura value(i) da da Fluka.dat per estrarre i soli valori di fluence del "concrete" e rispettive
    coordinate baricentri volumetti griglia USRBIN

    readFileLoop: do while (.not.EOF(unit1))

      value = 0.d0

!       Lettura per righe da da Fluka.dat

      read(unit1,*,end=100) (value(i),i=1,dimRow)

!       Aggiornamento variabile contatore cont per identificare la posizione nel file delle fluences
    in 10 righe

100   continue
      do j=1,dimRow
        cont = cont+1

!       Se cont minore di 10 ma maggiore del numero di volumetti esco dal ciclo do

        if (cont>totX*totY*totZ) exit
        if (k>nvol) exit

!       Avanzo (secondo plottaggio fluences da Fluka) in X fino a completamento di X

        posX = posX+1

        xB(k) = (xstart+D*(posX-1))*10.d0
        yB(k) = (ystart+D*(posY-1))*10.d0
        zB(k) = (zstart+D*(posZ-1))*10.d0

!       Avanzo (secondo plottaggio fluences da Fluka) in Y da completamento di X fino a
    completamento di Y

        if (posX==totX+1) then
                posX = 1
                posY = posY+1

        xB(k) = (xstart+D*(posX-1))*10.d0
        yB(k) = (ystart+D*(posY-1))*10.d0
        zB(k) = (zstart+D*(posZ-1))*10.d0

!       Avanzo (secondo plottaggio fluences da Fluka) in Z da completamento di Y fino a
    completamento di Z

                if(posY==totY+1) then
                        posY = 1
                        posZ = posZ+1

        xB(k) = (xstart+D*(posX-1))*10.d0
        yB(k) = (ystart+D*(posY-1))*10.d0
        zB(k) = (zstart+D*(posZ-1))*10.d0

                end if
        end if

!       Se sono nel fuori dallo shielding per X e Z ma non per Y allora prendo il valore di fluence
    per il vettore dati(i)

        if((posZ>=limitZ(1)).and.(posZ<=limitZ(2))) then
                if((posX>limitX(1)).and.(posX<=limitX(2))) then
                  if ((posY<limitY(1)).or.(posY>limitY(2))) then
                        dati(k) = value(j)*c
                k=k+1
                  end if
                else
                        dati(k) = value(j)*c
                k=k+1
                end if
        else
          dati(k) = value(j)*c
          k=k+1
        end if
      end do



    end do readFileLoop

    close(unit1)

return
900 format('start x = ',i2,'end x = ',i2,/ &
           'start y = ',i2,'end y = ',i2,/ &
           'start z = ',i2,'end z = ',i2)
```

```
995 format(29x,3(E25.14))

998 continue
    write(*,*)'open file ERROR'
    pause
    stop
end subroutine readData

------------------------------------------------------------------------------------------

subroutine trasporto (unit3,unit4,nvol,dati,xB,yB,zB,L,A,dist,fluence)


implicit none
integer                   ::  unit3, unit4, unit5

integer                   ::  nvol

!---------------------------------
integer, parameter        ::  nnodi = 4037
!---------------------------------

real*8,dimension(nvol)    ::  dati
real*8,dimension(nvol)    ::  xB,yB,zB

! L = Dimensione lato mesh Straus
! A(i,j) = Coordinate nodi Straus
! dist = Distanza nodo Straus-baricentro griglia USRBIN
! ok = Distanza minima
! fluence(i) = Fluence ai nodi di Straus al solo "concrete"

real*8                                    ::  L
real*8,dimension(nnodi,3) ::  A
real*8                                    ::  dist, ok
real*8,dimension(nnodi)   ::  fluence

integer                                   ::  i,j,k,m

    open(unit3, file='FlukaBprova.dat',status='old')
    open(unit4, file='StrausNprova.dat',status='old')

! Lettura da file coordinate formattate sia di Fluka che di Straus

    k = 0
    do while (.not.EOF(unit3))
        k= k+1
        read (unit3,996) i, xB(k),yB(k),zB(k),dati(k)
    end do
    close(unit3)
    write(*,*)'lettura completata unita: unita3'

    k = 0
    do while (.not.EOF(unit4))
        k=k+1
         read(unit4,997) i,A(k,1),A(k,2),A(k,3)
    end do
    close(unit4)
    write(*,*)'lettura completata unita: unita4'

! Inizializzazione variabili

      m=1

      dist=0.d0
      fluence=0.d0

!----------------------
      L = 100.d0
!----------------------

! Per ogni nodo Straus, se la distanza col baricentro del volumetto si trova entro la dimensione
    della mesh Straus in ogni direzione cartesiana, calcolo distanza 3D

      do j=1,nnodi
        do k=1,nvol

        if (k>nvol) exit


        if (Abs(A(j,1)-xB(k)).LE.L) then
             if (Abs(A(j,2)-yB(k)).LE.L) then
                        if (Abs(A(j,3)-zB(k)).LE.L) then

        dist=((A(j,1)-xB(k))**2+(A(j,2)-yB(k))**2+(A(j,3)-zB(k))**2)**0.5d0

!       Se la distanza 3D è minore di L/2 allora prendo il valore corrispondente del baricentro per
    fluence(i)
!       e aggiorno la distanza come distanza minima (ok) (caso a parte per iterazione 0: ok=0.d0)

        if (dist.LE.L/2) then
                if (ok.EQ.0.d0) then
```

```
                        ok=dist
                        fluence(m)=dati(k)
                end if
                if (dist<ok) then
                        ok=dist
                        fluence(m)=dati(k)
                end if
                if (dist.EQ.ok) then
                        ok=dist
                        fluence(m)=dati(k)
                end if

        else
                go to 100
        end if

100     continue
                        end if
                end if
        end if


        end do

        m=m+1
        ok=0.d0

      end do

return

996 format(i10,4(E15.4))
997 format(i10,3(E15.4))

pause
stop

end subroutine trasporto

-------------------------------------------------------------------------------------

subroutine density (unit7, unit4, unit9, unit10,          &
                    totXd    , totYd   , totZd   ,        &
                    nvoldep, ro,xBd,yBd,zBd,L,Ad,tempmod)

implicit none
integer                 ::  unit7, unit4,unit9, unit10

integer                 ::  totXd,totYd,totZd
integer                 ::  totXHoled,totYHoled,totZHoled
integer                 ::  xd,yd,zd

! dimRow = Dimensione plottaggio fluence da Fluka: 10 righe
! value (i) = Fluence da USRBIN, dimensione 10
! limitI = Indice di ordine dei volumetti limite per l'hole
! a,b,c,d,e = Valori limite di fluence per gli intervalli di differenziazione regione in Fluka per
    densità (in funzione di fluence)
! reg1, reg2, reg3, reg4, reg5, reg6 = Indici interi di regione diversa in Fluka per densità (in
    funzione di fluence)
! ro(i) = Componenti matrice ascii per voxel, dimensione pari al numero totale di volumetti,
    componenti intere
! nB = Numero totale volumetti

integer, parameter      ::  dimRow = 10
real*8,dimension(dimRow) ::  valued

integer                 ::  posX,posY,posZ
real*8                  ::  xstart,ystart,zstart

integer                 ::  i,j,k, m
integer                 ::  cont

real*8                  ::  q, r, D
real*8                  ::  rega, regb, regc, rege, regf

integer                 ::  reg1, reg2, reg3, reg4, reg5, reg6,regaria
real*8                  ::  h,b,b2,c,c2,e,e2,f,f2,g,g2


integer                 ::  nvoldep
integer, parameter      ::  nnodi = 4037
integer, parameter      ::  nnodid = 64194


integer,dimension(nvoldep)::  ro
real*8,dimension(nvoldep) ::  xBd, yBd, zBd

real*8                  ::  L
real*8,dimension(nnodid,3)::  Ad
real*8,dimension(nnodid)  ::  tempMODEL
real*8,dimension(nvoldep) ::  tempmod
real*8                    ::  dist, ok
```

```
    open(unit7, file='Flukadep_6m.dat',status='unknown',err=998)
    open(unit4, file='StrausNprova.dat',status='old')
    open(unit9, file='StrausNprova2.dat',status='old')
    open(unit10, file='tempMODEL_6m.dat',status='old')

! Inizializzazione variabili contatore

    cont      = 0
    posX      = 0
    posY      = 1
    posZ      = 1
    k         = 1

    xBd = 0.d0
    yBd = 0.d0
    zBd = 0.d0

    xstart = 2.5d0
    ystart = 2.5d0
    zstart = 2.5d0

!----------------------

! Inizializzazione variabili

      m=1

      dist=0.d0
      tempmod=0.d0
      ro=0.d0

!----------------------
      L = 200.d0
!----------------------
    q=1.87E+15
    r=1.6E-10
    D = 5.d0

    rega= 20.45d0
    regb= 50.d0
    regc= 75.d0
    rege= 100.d0
    regf= 125.d0   !fino 150°C
    reg1= 1
    reg2= 2
    reg3= 3
    reg4= 4
    reg5= 5
    reg6= 6
    regaria= 0

    h = 25.0d0
    b = 11975.0d0
    b2 = 9025.0d0
    c = 5475.0d0
    c2 = 3675.0d0
    e = 25.0d0
    e2 = 3975.0d0
    f = 25.0d0
    f2 = 2975.0d0
    g = 8975.0d0
    g2= 6025.0d0

!---------------------------------


! Lettura ordinate nodali mesh grande

    k = 0
    do while (.not.EOF(unit9))
        k=k+1
        if (k>nnodid) exit
         read(unit9,997) i,Ad(k,1),Ad(k,2),Ad(k,3)
    end do
    close(unit9)

! Lettura temperature mesh grande

      do i=1,nnodid
      read(unit10,*) tempMODEL(i)
      end do


! Lettura value(i) da da Fluka.dat per estrarre i soli valori di fluence del "concrete" e rispettive
! coordinate baricentri volumetti griglia USRBIN

    k = 1

    readFileLoop: do while (.not.EOF(unit7))

        valued = 0.d0
```

```
!       Lettura per righe da da Flukadep.dat

      read(unit7,*,end=100) (valued(i),i=1,dimRow)

!       Aggiornamento variabile contatore cont per identificare la posizione nel file delle fluences
    in 10 righe

100   continue
      do j=1,dimRow
        cont = cont+1

!       Se cont minore di 10 ma maggiore del numero di volumetti esco dal ciclo do

        if (cont>totXd*totYd*totZd) exit
        if (k>nvoldep) exit

!       Avanzo (secondo plottaggio fluences da Fluka) in X fino a completamento di X

        posX = posX+1

        xBd(k) = (xstart+D*(posX-1))*10.d0
        yBd(k) = (ystart+D*(posY-1))*10.d0
        zBd(k) = (zstart+D*(posZ-1))*10.d0


!       Avanzo (secondo plottaggio fluences da Fluka) in Y da completamento di X fino a
    completamento di Y

        if (posX==totXd+1) then
                posX = 1
                posY = posY+1

        xBd(k) = (xstart+D*(posX-1))*10.d0
        yBd(k) = (ystart+D*(posY-1))*10.d0
        zBd(k) = (zstart+D*(posZ-1))*10.d0

!       Avanzo (secondo plottaggio fluences da Fluka) in Z da completamento di Y fino a
    completamento di Z

                if(posY==totYd+1) then
                        posY = 1
                        posZ = posZ+1

        xBd(k) = (xstart+D*(posX-1))*10.d0
        yBd(k) = (ystart+D*(posY-1))*10.d0
        zBd(k) = (zstart+D*(posZ-1))*10.d0

                end if
        end if
        k=k+1
      end do
   end do readFileLoop

    close(unit7)

! Per ogni nodo Straus, se la distanza col baricentro del volumetto si trova entro la dimensione
    della mesh
! Straus in ogni direzione cartesiana, calcolo distanza 3D

        do j=1,nvoldep
                do k=1,nnodid
        if (k>nnodid) exit

        if (Abs(Ad(k,1)-xBd(j)).LE.L) then
             if (Abs(Ad(k,2)-yBd(j)).LE.L) then
                        if (Abs(Ad(k,3)-zBd(j)).LE.L) then

        dist=((Ad(k,1)-xBd(j))**2+(Ad(k,2)-yBd(j))**2+(Ad(k,3)-zBd(j))**2)**0.5d0

!       Se la distanza 3D è minore di L/2 allora prendo il valore corrispondente del baricentro per
    fluence(i)
!       e aggiorno la distanza come distanza minima (ok) (caso a parte per iterazione 0: ok=0.d0)

        if (dist.LE.L/2) then
                if (ok.EQ.0.d0) then
                        ok=dist
                        tempmod(m)=tempMODEL(k)
                end if
                if (dist<ok) then
                        ok=dist
                        tempmod(m)=tempMODEL(k)
                end if
                if (dist.EQ.ok) then
                        ok=dist
                        tempmod(m)=tempMODEL(k)
                end if

        else
                go to 102
        end if

102     continue
```

```
                        end if
                end if
        end if


        end do

        m=m+1
        ok=0.d0

      end do

      k = 1

!       Se sono fuori dallo shielding per X e Z ma non per Y allora prendo il valore di
    regione/densità voluto per il vettore ro(i), altrimenti (cioè
!       se sono in effetti fuori dallo shielding ovvero sono nel fuori dallo shielding anche per Y)
    prendo come valore di regione/densità quello corrispondente alla
!       regione "aria"

        do j=1,nvoldep
        if((zBd(j)>c2).and.(zBd(j)<=c)) then
                if((xBd(j)>f2).and.(xBd(j)<g2)) then
                  if ((yBd(j)<=e2).or.(yBd(j)>=b2))then



                if ((tempmod(j)<=rega)) then
                        ro(k)=reg1
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>rega).and.(tempmod(j)<=regb)) then
                        ro(k)=reg2
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>regb).and.(tempmod(j)<=regc)) then
                        ro(k)=reg3
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>regc).and.(tempmod(j)<=rege)) then
                        ro(k)=reg4
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>rege).and.(tempmod(j)<=regf)) then
                        ro(k)=reg5
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>rege)) then
                        ro(k)=reg6
                        k=k+1
                        go to 101
                end if

                  else
                  ro(k)=regaria
                  k=k+1
                  go to 101
                  end if


                else


                if ((tempmod(j)<=rega)) then
                        ro(k)=reg1
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>rega).and.(tempmod(j)<=regb)) then
                        ro(k)=reg2
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>regb).and.(tempmod(j)<=regc)) then
                        ro(k)=reg3
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>regc).and.(tempmod(j)<=rege)) then
                        ro(k)=reg4
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>rege).and.(tempmod(j)<=regf)) then
                        ro(k)=reg5
                        k=k+1
                        go to 101
                end if
```

```
                if ((tempmod(j)>rege)) then
                        ro(k)=reg6
                        k=k+1
                        go to 101
                end if

                end if
        else


                if ((tempmod(j)<=rega)) then
                        ro(k)=reg1
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>rega).and.(tempmod(j)<=regb)) then
                        ro(k)=reg2
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>regb).and.(tempmod(j)<=regc)) then
                        ro(k)=reg3
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>regc).and.(tempmod(j)<=rege)) then
                        ro(k)=reg4
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>rege).and.(tempmod(j)<=regf)) then
                        ro(k)=reg5
                        k=k+1
                        go to 101
                end if
                if ((tempmod(j)>rege)) then
                        ro(k)=reg6
                        k=k+1
                        go to 101
                end if

101     end if
      end do

return
900 format('start x = ',i2,'end x = ',i2,/ &
           'start y = ',i2,'end y = ',i2,/ &
           'start z = ',i2,'end z = ',i2)
997 format(i10,3(E15.4))
998 continue
    write(*,*)'open file ERROR'
    pause
    stop
end subroutine density
```

Appendix C

```
program sommadep

implicit none
integer                   ::  unit1, unit2, unit3
integer                   :: i


integer, parameter        ::  dimRow = 10
real*8,dimension(dimRow)  ::  prompt
real*8,dimension(dimRow)  ::  dec
real*8,dimension(dimRow)  ::  dati


unit1 = 10
unit2 = 11
unit3 = 12


call readData(unit1, unit2, unit3, prompt, dec, dati)


pause
stop

end program sommadep

------------------------------------------------------------------------------------------

subroutine readData(unit1, unit2, unit3,prompt, dec, dati)


implicit none
integer                  ::  unit1, unit2, unit3

integer, parameter       ::  dimRow = 10
real*8,dimension(dimRow) ::  prompt
real*8,dimension(dimRow) ::  dec
real*8,dimension(dimRow) ::  dati

integer                  ::  i,j,k
real*8                   ::  q, r

    q=1.87E+15
    r=1.6E-10

    open(unit1, file='31endep_prompt.dat',status='unknown',err=998)

    open(unit2, file='38dec_o.dat',status='unknown',err=998)

    open(unit3, file='Flukadep_o_promptedec.dat',status='unknown')


! Lettura prompt(i) e dec(i) da da Fluka.dat per sommarli come J/(mm3 s): prompt: da GeV/(cm3 p);
    dec da GeV/(cm3 s)

    readFileLoop1: do while (.not.EOF(unit1))
    readFileLoop2: do while (.not.EOF(unit2))

      prompt = 0.d0
      dec = 0.d0

!       Lettura per righe da da Fluka.dat

      read(unit1,*,end=100) (prompt(i),i=1,dimRow)
      read(unit2,*,end=101) (dec(i),i=1,dimRow)

!       Aggiornamento variabile contatore cont per identificare la posizione nel file delle fluences
    in 10 righe

100   continue
101   continue

      do j=1,dimRow
          dati(j) = prompt(j)*q*r/1000.0d0+dec(j)*r/1000.0d0  ! dec è 0 per somma con promp per
    misure dopo 10 anni e dopo ogni ciclo
      end do

    write(unit3,995) (dati(i), i=1,dimRow)

    end do readFileLoop2
    end do readFileLoop1
```

```
    close(unit1)
    close(unit2)

return

995 format  (1(5x,1p,10(1x,e11.4)))

998 continue
    write(*,*)'open file ERROR'
    pause
    stop
end subroutine readData
```

Appendix D

```
program FlukaStraus

!---------------------------------
integer, parameter                              ::  nnodi = 7733
integer, parameter                              ::  nnodid = 64194
integer, parameter                              ::  nvol = 29600
integer, parameter                              ::  nvoldep = 4752000
!---------------------------------

    totX     = 20
    totY     = 20
    totZ     = 114

    totXHole = 20
    totYHole = 20
    totZHole = 40

    x        = 0
    y        = 1
    z        = 75

    totXd     = 180
    totYd     = 240
    totZd     = 110

    totXHoled = 60
    totYHoled = 100
    totZHoled = 36

    xd        = 60
    yd        = 81
    zd        = 75

-------------------------------------------------------------------------------------

subroutine readData(unit1, unit2,       &
                    totX    , totY   , totZ   ,&
                    totXHole,totYHole,totZHole,&
                    x       ,y       ,z       ,&
                    nvol ,dati,xB,yB,zB, A)

!---------------------------------
integer, parameter       ::  nnodi=7733

! c deriva da moltiplicare i risultati Fluka in n/(cm2 primary) per i protoni/s dati dal macchinario
    a 70MeV, 300microA;
! se voglio n/cm2 moltiplico per i s: 60*60*5000*50 se lavora a 5000hours/y per 50anni

  real*8, parameter        ::  c=1.87E+15

!---------------------------------

!----------------------
!   VALORI PER GRIGLIA USRBIN CON D=5cm E DIMENSIONI REALI (3,7m) DA FlukaB.ris

    xstart    =2.5d0
    ystart    =2.5d0
    zstart    =2.5d0
    D = 5.d0

!----------------------

-------------------------------------------------------------------------------------

program Medieflussi

! Prende i valori di prompt e decay radiation adeguatamente sommati da sommadep e li passa alla mesh
    Straus del vano bunker mediati per le condizioni al contorno dell'analisi termica transitoria.

implicit none
integer                          ::  unit1, unit2, unit3, unit4, unit5, unit6, unit7

! totI = Numero di volumetti che danno le dimensioni della griglia di USRBIN nelle direzioni
    cartesiane
! totIHole = Numero di volumetti che danno le dimensioni del vano interno allo shielding
! x,y,z = Indice di ordine del volumetto inizio dell'hole
! nnodi = Numero di nodi della mesh Straus
! nvol = Numero di volumetti della griglia USRBIN di solo cls
! dati(i) = Fluence ai baricentri USRBIN di solo "concrete" (da value(i) in da Fluka.dat)
! iB(i) = Coordinate baricentri griglia USRBIN
! A(i,j) = Coordinate nodi Straus
! D = Dimensione lato griglia USRBIN in cm
! L = Dimensione lato mesh Straus in mm
```

```
! dist = Distanza nodo Straus-baricentro griglia USRBIN
! fluence(i) = Fluence ai nodi di Straus al solo "concrete"


integer                           ::  totX,totY,totZ
integer                           ::  totXHole,totYHole,totZHole
integer                           ::  x,y,z

!---------------------------------

integer, parameter                ::  nvol = 4536000  ! è il valore di sola parte di cls cioé nvoldep
    - hole= 4752000 - 216000!

integer, parameter                ::  nvols  = 14202

integer, parameter                ::  nmax = 17100

integer, parameter                ::  nnodi = 64194

!---------------------------------

real*8,dimension(:),allocatable   ::  dati
real*8,dimension(:),allocatable   ::  xB,yB,zB
real*8,dimension(nvols)           ::  xBSl,yBSl,zBSl,xBS,yBS,zBS
real*8,dimension(nnodi,3)         ::  A
integer,dimension(nvols,20)       ::  nop


real*8                            ::  D

real*8                            ::  dist
real*8,dimension(nvols)           ::  energy
real*8,dimension(nmax)            ::  energyf


integer                           ::  i,j

    !-------------------
    unit1        = 9
    unit2        =10
    unit3        =11
    unit4        =12
    unit5        =13
    unit6        =14
    unit7        =15

!   VALORI PER GRIGLIA USRBIN CON D=5cm E DIMENSIONI REALI (3,7m) DA FlukaB.ris

    totX         = 180
    totY         = 240
    totZ         = 110

    totXHole     = 60
    totYHole     = 100
    totZHole     = 36

    x            = 60
    y            = 81
    z            = 75

    !-------------------

! Dimensione dati(i), iB(i) pari a numero volumetti di solo "concrete"

    allocate(dati(totX*totY*totZ-totXHole*totYHole*totZHole))
    allocate(xB(totX*totY*totZ-totXHole*totYHole*totZHole))
    allocate(yB(totX*totY*totZ-totXHole*totYHole*totZHole))
    allocate(zB(totX*totY*totZ-totXHole*totYHole*totZHole))

    dati=0.d0

    open(unit3, file='FlukaBprova2.dat',status='unknown')
    open(unit4, file='StrausNprova2.dat',status='unknown')
    open(unit5, file='fluence.ris',status='unknown')
    open(unit6, file='Straus txt_elementi2.txt',status='old')

! Richiamo subroutine readData: a) le formatta, b) crea e formatta le coordinate baricentriche di
    Fluka e "vede" solo i valori di fluence da USRBIN al "concrete"

    do i=1,nvols
      read(unit6,*) (nop(i,j), j=1,20)
    end do

    call readData(unit1, unit2,                &
                  totX      , totY   , totZ   ,&
                  totXHole  ,totYHole,totZHole,&
                  x         ,y       ,z       ,&
                  size(dati),dati, xB,yB,zB,A)

    do i=1,size(dati)
        write(unit3,999) i,xB(i),yB(i),zB(i),dati(i)
    end do
```

```
    do i=1,nnodi
        write (unit4,99) i,(A(i,j), j=1,3)
    end do

! Richiamo tramite la subroutine trasporto riporta ai nodi di Straus i valori di fluence ai
    baricentri della mesh di Fluka

    write(*,*) 'ok'

    call trasportobaric (unit3, unit4, unit7,                                &
        size(dati),dati,xB,yB,zB,xBSl,yBSl,zBSl,xBS,yBS,zBS,                 &
        A,nop,dist,energy,energyf)

! Stampa coordinate baricentriche per ogni elemento di Straus

    do i=1,nvols
        write(unit5,99)i,xBS(i),yBS(i),zBS(i)
    end do

! Stampa per ogni volumetto il valore sorgente [J/s/mm3]

    do i=1,nvols
        write(unit5,9) energy(i)
    end do

! Stampa per i soli volumetti di contorno (e non quello a T imposta) il valore di flusso [J/s/mm2]

    write(unit5,*)

    do i=1,nmax
        write(unit5,9) energyf(i)
    end do

! Stampa coordinate nodali per NEWCON3D

    write(unit5,997)
    do i=1,nnodi
        write(unit5,997) i,(A(i,j),j=1,3)
    end do

    close(unit3)
    close(unit4)
    close(unit5)

pause
stop

999 format(i10,4(E15.4))
 99 format(i10,3(E15.4))

 91 format(i10,E15.4)
  9 format(E15.4)


996 format(i10,4(E15.4))

997 FORMAT  (i10,f12.2,2f10.2)

end program  Medieflussi

-----------------------------------------------------------------------------------------

subroutine readData(unit1, unit2,             &
                    totX    , totY   , totZ   ,&
                    totXHole,totYHole,totZHole,&
                    x       ,y       ,z       ,&
                    nvol ,dati,xB,yB,zB, A)

implicit none
integer                  ::  unit1, unit2

integer                  ::  totX,totY,totZ
integer                  ::  totXHole,totYHole,totZHole
integer                  ::  x,y,z
integer                  ::  nvol


!---------------------------------

  integer, parameter     ::  nnodi=64194

!---------------------------------

real*8,dimension(nvol)   ::  dati
real*8,dimension(nvol)   ::  xB,yB,zB
real*8                   ::  xstart,ystart,zstart, D
real*8,dimension(nnodi,3)::  A

! dimRow = Dimensione plottaggio fluence da Fluka: 10 righe
! value (i) = Fluence da USRBIN, dimensione 10
! limitI = Indice di ordine dei volumetti limite per l'hole
```

```
integer, parameter       ::  dimRow = 10
real*8,dimension(dimRow) ::  value
integer,dimension(2)     ::  limitX, limitY, limitZ
integer                  ::  posX,posY,posZ

integer                  ::  i,j,k
integer                  ::  cont

    open(unit1, file='Flukadep_d_solodec.dat',status='unknown',err=998)
    open(unit2, file='Straus txt_nodi2.txt',status='unknown',err=998)

! Definizione limiti dell'hole

    limitX(1) = x
    limitX(2) = x+totXHole
    limitY(1) = y
    limitY(2) = y+totYHole-1
    limitZ(1) = z
    limitZ(2) = z+totZHole-1

    write(*,900)limitX(1),limitX(2),              &
                limitY(1),limitY(2),              &
                limitZ(1),limitZ(2)

! Inizializzazione variabili contatore

    cont      = 0
    posX      = 0
    posY      = 1
    posZ      = 1
    k         = 1

!----------------------
!   VALORI PER GRIGLIA USRBIN CON D=5cm E DIMENSIONI REALI (3,7m) DA FlukaB.ris
!
    xstart    =2.5d0
    ystart    =2.5d0
    zstart    =2.5d0
    D = 5.0d0

!-----------------------

    xB = 0.d0
    yB = 0.d0
    zB = 0.d0

! Lettura nodi mesh Straus da da Straus.dat per formattazione nel main

      do i=1,nnodi
      read(unit2,995) (A(i,j), j=1,3)
      end do

! Lettura value(i) da da Fluka.dat per estrarre i soli valori di fluence del "concrete" e rispettive
    coordinate baricentri volumetti griglia USRBIN

    readFileLoop: do while (.not.EOF(unit1))

      value = 0.d0

!       Lettura per righe da da Fluka.dat

      read(unit1,*,end=100) (value(i),i=1,dimRow)

!       Aggiornamento variabile contatore cont per identificare la posizione nel file delle fluences
    in 10 righe

100   continue
      do j=1,dimRow
        cont = cont+1

!       Se cont minore di 10 ma maggiore del numero di volumetti esco dal ciclo do

        if (cont>totX*totY*totZ) exit
        if (k>nvol) exit

!       Avanzo (secondo plottaggio fluences da Fluka) in X fino a completamento di X

        posX = posX+1

        xB(k) = (xstart+D*(posX-1))*10.d0
        yB(k) = (ystart+D*(posY-1))*10.d0
        zB(k) = (zstart+D*(posZ-1))*10.d0

!       Avanzo (secondo plottaggio fluences da Fluka) in Y da completamento di X fino a
    completamento di Y

        if (posX==totX+1) then
                posX = 1
                posY = posY+1

        xB(k) = (xstart+D*(posX-1))*10.d0
        yB(k) = (ystart+D*(posY-1))*10.d0
        zB(k) = (zstart+D*(posZ-1))*10.d0
```

```
!       Avanzo (secondo plottaggio fluences da Fluka) in Z da completamento di Y fino a
    completamento di Z

                if(posY==totY+1) then
                        posY = 1
                        posZ = posZ+1

        xB(k) = (xstart+D*(posX-1))*10.d0
        yB(k) = (ystart+D*(posY-1))*10.d0
        zB(k) = (zstart+D*(posZ-1))*10.d0

                end if
        end if

!       Se sono fuori dallo shielding per X e Z ma non per Y allora prendo il valore di fluence per
    il vettore dati(i)

        if((posZ>=limitZ(1)).and.(posZ<=limitZ(2))) then
                if((posX>limitX(1)).and.(posX<=limitX(2))) then
                  if ((posY<limitY(1)).or.(posY>limitY(2))) then
                        dati(k) = value(j)
                k=k+1
                  end if
                else
                        dati(k) = value(j)
                k=k+1
                end if
        else
          dati(k) = value(j)
          k=k+1
        end if
      end do


    end do readFileLoop

    close(unit1)

return
900 format('start x = ',i2,'end x = ',i2,/ &
           'start y = ',i2,'end y = ',i2,/ &
           'start z = ',i2,'end z = ',i2)


995 format(29x,3(E25.14))

998 continue
    write(*,*)'open file ERROR'
    pause
    stop
end subroutine readData

-------------------------------------------------------------------------------------------

subroutine trasportobaric (unit3,unit4,unit7,nvol,dati,xB,yB,zB,   &
                           xBSl,yBSl,zBSl,xBS,yBS,zBS,             &
                           A, nop, dist, energy,energyf)

! Prende i valori di prompt e decay radiation adeguatamente sommati da sommadep e valuta i valori
    mediati dei flussi di calore alle facce 2b, 3b, 3b1, 3b2, 4b, 5b, 6b e delle sorgenti di calore
    alle zone s1 e s2.

implicit none
integer                             ::  unit3, unit4, unit7

integer                             ::  nvol
integer, parameter                  ::  nvols = 14202
integer, parameter                  ::  nnodi = 64194
real*8                              ::  h,b,b2,c,c2,e,e2,f,f2,g,g2
real*8                              ::  c22
real*8                              ::  f21, g21, f22, g22
real*8                              ::  e21, b21, e22, b22
real*8                              ::  f23, g23
real*8                              ::  e23, b23

integer, parameter                  ::  nmax=17100  ! è il numero di facce con condizioni al contorno,
    non lo so a priori, o lo vedo dalle cond al cont messe in Straus

real*8,dimension(nvol)              ::  dati
real*8,dimension(nvol)              ::  xB,yB,zB
real*8,dimension(nvols)             ::  xBSl,yBSl,zBSl,xBS,yBS,zBS

! L = Dimensione lato mesh Straus
! A(i,j) = Coordinate nodi Straus
! dist = Distanza nodo Straus-baricentro griglia USRBIN
! ok = Distanza minima
! fluence(i) = Fluence ai nodi di Straus al solo "concrete"

real*8                              ::  L
real*8,dimension(nnodi,3)           ::  A
```

```
integer                             ::  alfa
real*8                              ::  dist, ok, q, r, D, sommas1, sommas2, sommas1cont, sommas2cont
real*8                              ::  sommaf2,sommaf3,sommaf4, sommaf5, sommaf6
real*8                              ::  sommaf2b,sommaf3b,sommaf3b1, sommaf3b2
real*8                              ::  sommaf4b, sommaf5b, sommaf6b

real*8                              ::  sommaf2cont, sommaf3cont, sommaf4cont,sommaf5cont,sommaf6cont
real*8                              ::  sommaf2bcont, sommaf3bcont, sommaf3b1cont,sommaf3b2cont
real*8                              ::  sommaf4bcont, sommaf5bcont, sommaf6bcont

real*8                              ::  medias1, medias2, mediaf2,mediaf3
real*8                              ::  mediaf4, mediaf5, mediaf6,mediaf2b,mediaf3b
real*8                              ::  mediaf3b1, mediaf3b2
real*8                              ::  mediaf4b, mediaf5b, mediaf6b

real*8,dimension(nvols)             ::  energy
real*8,dimension(nmax)              ::  energyf

integer,dimension(nvols,20)         ::  nop

integer                             ::  i,j,k,m,m1, m2,n,k1,k2,k3,k4,k5,k6,k7,k8, cont
integer                             ::  contm1, contm2
integer                             ::  cont2, cont3, cont4, cont5, cont6
integer                             ::  cont2b, cont3b, cont3b1, cont3b2, cont4b
integer                             ::  cont5b, cont6b

integer                             ::  n2, n3, n4, n5, n6, n2b, n3b, n3b1, n3b2
integer                             ::  n4b, n5b, n6b

!---------------------------------


! q,r,D derivano da moltiplicare i risultati Fluka in GeV/(cm3 primary) per i protoni/s dati dal
    macchinario a 70MeV, 300microA; e ho energy in J/s/mm3 per Straus

    q=1000.0d0
    r=1.0d0

    D = 5.0d0         ! [cm]
    L= 200.0d0         ! [mm]

! Con riferimento alla numerazione Straus:
! h = zBS di faccia 1 del brick [mm] = 500.0d0  voglio T imposta  -> niente flusso
! b = yBS di faccia 2 del brick [mm] = 2500.0d0   voglio flusso uscente
! c = zBS di faccia 3 del brick [mm] = 2500.0d0  voglio flusso ENTRANTE (negativo)
! e = yBS di faccia 4 del brick [mm] = 500.0d0  voglio flusso uscente
! f = xBS di faccia 5 del brick [mm] = 500.0d0  voglio flusso uscente
! g = xBS di faccia 6 del brick [mm] = 2500.0d0  voglio flusso uscente


    h = 200.0d0
    b = 9500.0d0
    b2 = 9100.0d0
    c = 5400.0d0
    c2 = 3600.0d0
    e = 3500.0d0
    e2 = 3900.0d0
    f = 2500.0d0
    f2 = 2900.0d0
    g = 6500.0d0
    g2= 6100.0d0

    c22 = 3600.d0

    f21 = 3700.0d0
    g21 = 4300.0d0
    f22= 3300.0d0
    g22 = 4500.0d0
    e21 = 4700.0d0
    b21 = 5300.0d0
    e22= 4500.0d0
    b22 = 5500.0d0

    f23 = 3900.0d0
    g23 = 4100.0d0

    e23 = 4900.0d0
    b23 = 5100.0d0

!---------------------------------

    open(unit3, file='FlukaBprova2.dat',status='old')
    open(unit4, file='StrausNprova2.dat',status='old')
    open(unit7, file='medie.dat',status='unknown')


! Lettura da file coordinate formattate sia di Fluka che di Straus + lettura incidenze Straus

    k = 0
    do while (.not.EOF(unit3))
        k= k+1
        read (unit3,996) i, xB(k),yB(k),zB(k),dati(k)
```

```
    end do
    close(unit3)
    write(*,*)'lettura completata unita: unita3'

    k = 0
    do while (.not.EOF(unit4))
        k=k+1
         read(unit4,997) i,A(k,1),A(k,2),A(k,3)
    end do
    close(unit4)
    write(*,*)'lettura completata unita: unita4'


! Inizializzazione variabili
          m=1
          n=1                                                sommaf2=0.d0
                                                             sommaf3=0.d0
          m1=1                                               sommaf4=0.d0
          m2=1                                               sommaf5=0.d0
          n2=1                                               sommaf6=0.d0
          n3=1                                               sommaf2b=0.d0
          n4=1                                               sommaf3b=0.d0
          n5=1                                               sommaf3b1=0.d0
          n6=1                                               sommaf3b2=0.d0
          n2b=1                                              sommaf4b=0.d0
          n3b=1                                              sommaf5b=0.d0
          n3b1=1                                             sommaf6b=0.d0
          n3b2=1
          n4b=1                                              sommaf2cont=0.d0
          n5b=1                                              sommaf3cont=0.d0
          n6b=1                                              sommaf4cont=0.d0
                                                             sommaf5cont=0.d0
          cont=1                                             sommaf6cont=0.d0
                                                             sommaf2bcont=0.d0
          contm1=1                                           sommaf3bcont=0.d0
          contm2=1                                           sommaf3b1cont=0.d0
          cont2=1                                            sommaf3b2cont=0.d0
          cont3=1                                            sommaf4bcont=0.d0
          cont4=1                                            sommaf5bcont=0.d0
          cont5=1                                            sommaf6bcont=0.d0
          cont6=1
          cont2b=1
          cont3b=1                                           mediaf2=0.d0
          cont3b1=1                                          mediaf3=0.d0
          cont3b2=1                                          mediaf4=0.d0
          cont4b=1                                           mediaf5=0.d0
          cont5b=1                                           mediaf6=0.d0
          cont6b=1                                           mediaf2b=0.d0
                                                             mediaf3b=0.d0
          dist=0.d0                                          mediaf3b1=0.d0
          energy=0.d0                                        mediaf3b2=0.d0
          energyf=0.d0                                       mediaf4b=0.d0
                                                             mediaf5b=0.d0
          sommas1=0.d0                                       mediaf6b=0.d0
          sommas1cont=0.d0
          medias1=0.d0                                       xBSl = 0.d0
          sommas2=0.d0                                       yBSl = 0.d0
          sommas2cont=0.d0                                   zBSl = 0.d0
          medias2=0.d0
                                                             xBS = 0.d0
                                                             yBS = 0.d0
                                                             zBS = 0.d0

! Per ogni nodo Straus, se la distanza col baricentro del volumetto si trova entro la dimensione
    della mesh Straus in ogni direzione cartesiana, calcolo distanza 3D

      do j=1,nvols

         k1=nop(j,1)
         k2=nop(j,2)
         k3=nop(j,3)
         k4=nop(j,4)
         k5=nop(j,5)
         k6=nop(j,6)
         k7=nop(j,7)
         k8=nop(j,8)
         xBSl(j)=((A(k1,1)+A(k2,1)+A(k5,1)+A(k6,1))/4-(A(k3,1)+A(k4,1)+A(k7,1)+A(k8,1))/4)/2
         xBS(j)=A(k4,1)+xBSl(j)
         yBSl(j)=((A(k2,2)+A(k3,2)+A(k6,2)+A(k7,2))/4-(A(k1,2)+A(k4,2)+A(k5,2)+A(k8,2))/4)/2
         yBS(j)=A(k4,2)+yBSl(j)
         zBSl(j)=((A(k5,3)+A(k6,3)+A(k7,3)+A(k8,3))/4-(A(k1,3)+A(k2,3)+A(k3,3)+A(k4,3))/4)/2
         zBS(j)=A(k4,3)+zBSl(j)

       end do


        do j=1,nvols

           do k=1,nvol
        if (k>nvol) exit

        if (Abs(xBS(j)-xB(k)).LE.L) then
             if (Abs(yBS(j)-yB(k)).LE.L) then
```

```
                        if (Abs(zBS(j)-zB(k)).LE.L) then

        dist=((xBS(j)-xB(k))**2.0d0+(yBS(j)-yB(k))**2.0d0+(zBS(j)-zB(k))**2.0d0)**0.5d0

!       Se la distanza 3D è minore di L/2 allora prendo il valore corrispondente del baricentro per
    fluence(i)
!       e aggiorno la distanza come distanza minima (ok) (caso a parte per iterazione 0: ok=0.d0)

        if (dist.LE.((L/2.0d0)**2+(L/2.0d0)**2)**0.5d0) then
                if (ok.EQ.0.d0) then
                        ok=dist
                        energy(m)=dati(k)*q*r/1000.0d0


! Inserimento zona calda per SORGENTI: s2 e meno calda s1, il resto è s

if (dati(k).NE.0.d0) then
    if((xBS(j).GE.f23).and.(xBS(j).LE.g23)) then
                                                        if((yBS(j).GE.e23).and.(yBS(j).LE.b23)) then

    if((zBS(j).GE.c22).and.(zBS(j).LE.c2)) then

                                                                                        sommas2cont =
        sommas2 + energy(m)
        contm2=m2+1

        else

        sommas1cont = sommas1 + energy(m)
        contm1=m1+1
        end if

        else
        sommas1cont = sommas1 + energy(m)
        contm1=m1+1
        end if

        else
        sommas1cont = sommas1 + energy(m)
        contm1=m1+1
        end if
end if

        alfa=k

! Fine inserimento di 3 zone calde per SORGENTI

!       FACCIA 2
                        if(yBS(j).EQ.b) then
                            if(xBS(j).EQ.f) then
                                if(zBS(j).EQ.h) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 2
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 5

                                        if (dati(k).NE.0.d0) then
                                        sommaf2cont = sommaf2 + energyf(n)
                                        sommaf5cont = sommaf5 + energyf(n)
                                        cont2=n2+1
                                        cont5=n5+1
                                        end if
                                        go to 2
                                        end if
                                    if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 2
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0         ! faccia 5

                                        if (dati(k).NE.0.d0) then
                                        sommaf2cont = sommaf2 + energyf(n)
                                        sommaf5cont = sommaf5 + energyf(n)
                                        cont2=n2+1
                                        cont5=n5+1
                                        end if
                                        go to 2
                                end if
                                        if(zBS(j).EQ.c) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 2
                        energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3
                        energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 5

                                        if (dati(k).NE.0.d0) then
                                        sommaf2cont = sommaf2 + energyf(n)
                                        sommaf3cont = sommaf3 + energyf(n)
                                        sommaf5cont = sommaf5 + energyf(n)
                                        cont2=n2+1
                                        cont3=n3+1
                                        cont5=n5+1
                                        end if
                                        go to 3
                                        end if
                                end if


                                if((xBS(j).GT.f).and.(xBS(j).LT.g)) then
```

```
                                if(zBS(j).EQ.h) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 2

                                    if (dati(k).NE.0.d0) then
                                    sommaf2cont = sommaf2 + energyf(n)
                                    cont2=n2+1
                                    end if
                                    go to 1
                                    end if
                                if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
          energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 2

                                    if (dati(k).NE.0.d0) then
                                    sommaf2cont = sommaf2 + energyf(n)
                                    cont2=n2+1
                                    end if
                                    go to 1
                            end if
                                if(zBS(j).EQ.c) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0             ! faccia 2
                        energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3

                                    if (dati(k).NE.0.d0) then
                                    sommaf2cont = sommaf2 + energyf(n)
                                    sommaf3cont = sommaf3 + energyf(n)
                                    cont2=n2+1
                                    cont3=n3+1
                                    end if
                                    go to 2
                                    end if
                            end if


                            if(xBS(j).EQ.g) then
                                if(zBS(j).EQ.h) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 2
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                                    if (dati(k).NE.0.d0) then
                                    sommaf2cont = sommaf2 + energyf(n)
                                    sommaf6cont = sommaf6 + energyf(n)
                                    cont2=n2+1
                                    cont6=n6+1
                                    end if
                                    go to 2
                                    end if
                                if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 2
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 6

                                    if (dati(k).NE.0.d0) then
                                    sommaf2cont = sommaf2 + energyf(n)
                                    sommaf6cont = sommaf6 + energyf(n)
                                    cont2=n2+1
                                    cont6=n6+1
                                    end if
                                    go to 2
                            end if
                                if(zBS(j).EQ.c) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0        ! faccia 2
                        energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3
                        energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                                    if (dati(k).NE.0.d0) then
                                    sommaf2cont = sommaf2 + energyf(n)
                                    sommaf3cont = sommaf3 + energyf(n)
                                    sommaf6cont = sommaf6 + energyf(n)
                                    cont2=n2+1
                                    cont3=n3+1
                                    cont6=n6+1
                                    end if
                                    go to 3
                                    end if
                            end if
                        end if

! FACCIA 3

                        if(zBS(j).EQ.c) then
                            if(xBS(j).EQ.f) then
                                if(yBS(j).EQ.e) then
                        energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0      ! faccia 3
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 4
                        energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 5

                                    if (dati(k).NE.0.d0) then
                                    sommaf3cont = sommaf3 - energyf(n)
                                    sommaf4cont = sommaf4 - energyf(n)
                                    sommaf5cont = sommaf5 - energyf(n)
                                    cont3=n3+1
                                    cont4=n4+1
                                    cont5=n5+1
                                    end if
```

```
                    go to 3
                    end if
                if((yBS(j).GT.e).and.(yBS(j).LT.b)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0  ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0 ! faccia 5

                    if (dati(k).NE.0.d0) then
                    sommaf3cont = sommaf3 - energyf(n)
                    sommaf5cont = sommaf5 - energyf(n)
                    cont3=n3+1
                    cont5=n5+1
                    end if
                    go to 2
            end if
                    if(yBS(j).EQ.b) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 2
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3
energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 5

                    if (dati(k).NE.0.d0) then
                    sommaf2cont = sommaf2 + energyf(n)
                    sommaf3cont = sommaf3 + energyf(n)
                    sommaf5cont = sommaf5 + energyf(n)
                    cont2=n2+1
                    cont3=n3+1
                    cont5=n5+1
                    end if
                    go to 3
                    end if
            end if

! PER HOLE IN FACCIA 3

            if((xBS(j).GT.f).and.(xBS(j).LT.f2)) then
                if(yBS(j).EQ.e) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0         ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0        ! faccia 4

                    if (dati(k).NE.0.d0) then
                    sommaf3cont = sommaf3 - energyf(n)
                    sommaf4cont = sommaf4 - energyf(n)
                    cont3=n3+1
                    cont4=n4+1
                    end if
                    go to 2
                    end if
                if((yBS(j).GT.e).and.(yBS(j).LT.b)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0  ! faccia 3

                    if (dati(k).NE.0.d0) then
                    sommaf3cont = sommaf3 - energyf(n)
                    cont3=n3+1
                    end if
                    go to 1
            end if
                if(yBS(j).EQ.b) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0             ! faccia 2
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3

                    if (dati(k).NE.0.d0) then
                    sommaf2cont = sommaf2 + energyf(n)
                    sommaf3cont = sommaf3 + energyf(n)
                    cont2=n2+1
                    cont3=n3+1
                    end if
                    go to 2
                    end if
            end if


            if(xBS(j).EQ.f2) then
                if(yBS(j).EQ.e) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 4

                    if (dati(k).NE.0.d0) then
                    sommaf3cont = sommaf3 - energyf(n)
                    sommaf4cont = sommaf4 - energyf(n)
                    cont3=n3+1
                    cont4=n4+1
                    end if
                    go to 2
                    end if
                if((yBS(j).GT.e).and.(yBS(j).LE.e2)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3

                    if (dati(k).NE.0.d0) then
                    sommaf3cont = sommaf3 - energyf(n)
                    cont3=n3+1
                    end if
                    go to 1
                    end if
                if((yBS(j).GT.e2).and.(yBS(j).LT.b2)) then
```

```
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 3
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0        ! faccia 5'

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf5bcont = sommaf5b - energyf(n)
                cont3=n3+1
                cont5b=n5b+1
                end if
                go to 2
                end if
            if((yBS(j).GE.b2).and.(yBS(j).LT.b)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                cont3=n3+1
                end if
                go to 1
                end if
            if(yBS(j).EQ.b) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0             ! faccia 2
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf2cont = sommaf2 + energyf(n)
                sommaf3cont = sommaf3 + energyf(n)
                cont2=n2+1
                cont3=n3+1
                end if
                go to 2
                end if
        end if


        if((xBS(j).GT.f2).and.(xBS(j).LT.g2)) then
            if(yBS(j).EQ.e) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 4

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf4cont = sommaf4 - energyf(n)
                cont3=n3+1
                cont4=n4+1
                end if
                go to 2
                end if
            if((yBS(j).GT.e).and.(yBS(j).LT.e2)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                cont3=n3+1
                end if
                go to 1
                end if
            if(yBS(j).EQ.e2) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 4'

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf4bcont = sommaf4b - energyf(n)
                cont3=n3+1
                cont4b=n4b+1
                end if
                go to 2
                end if
            if(yBS(j).EQ.b2) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 2'
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf2bcont = sommaf2b - energyf(n)
                sommaf3cont = sommaf3 - energyf(n)
                cont2b=n2b+1
                cont3=n3+1
                end if
                go to 2
                end if
            if((yBS(j).GT.b2).and.(yBS(j).LT.b)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                cont3=n3+1
                end if
                go to 1
                end if
            if(yBS(j).EQ.b) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0             ! faccia 2
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3
```

```
                if (dati(k).NE.0.d0) then
                sommaf2cont = sommaf2 + energyf(n)
                sommaf3cont = sommaf3 + energyf(n)
                cont2=n2+1
                cont3=n3+1
                end if
                go to 2
                end if
        end if

        if(xBS(j).EQ.g2) then
            if(yBS(j).EQ.e) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 4

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf4cont = sommaf4 - energyf(n)
                cont3=n3+1
                cont4=n4+1
                end if
                go to 2
                end if
            if((yBS(j).GT.e).and.(yBS(j).LE.e2)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                cont3=n3+1
                end if
                go to 1
                end if
            if((yBS(j).GT.e2).and.(yBS(j).LT.b2)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0         ! faccia 6'

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf6bcont = sommaf6b - energyf(n)
                cont3=n3+1
                cont6b=n6b+1
                end if
                go to 2
                end if
            if((yBS(j).GE.b2).and.(yBS(j).LT.b)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                cont3=n3+1
                end if
                go to 1
                end if
            if(yBS(j).EQ.b) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0             ! faccia 2
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf2cont = sommaf2 + energyf(n)
                sommaf3cont = sommaf3 + energyf(n)
                cont2=n2+1
                cont3=n3+1
                end if
                go to 2
                end if
        end if


        if((xBS(j).GT.g2).and.(xBS(j).LT.g)) then
            if(yBS(j).EQ.e) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 4

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf4cont = sommaf4 - energyf(n)
                cont3=n3+1
                cont4=n4+1
                end if
                go to 2
                end if
            if((yBS(j).GT.e).and.(yBS(j).LT.b)) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 3

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                cont3=n3+1
                end if
                go to 1
        end if
                if(yBS(j).EQ.b) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0             ! faccia 2
```

```
                        energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3

                                    if (dati(k).NE.0.d0) then
                                    sommaf2cont = sommaf2 + energyf(n)
                                    sommaf3cont = sommaf3 + energyf(n)
                                    cont2=n2+1
                                    cont3=n3+1
                                    end if
                                    go to 2
                                    end if
                            end if


! FINE PER HOLE IN FACCIA 3

                            if(xBS(j).EQ.g) then
                                if(yBS(j).EQ.e) then
                        energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 3
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 4
                        energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 6

                                    if (dati(k).NE.0.d0) then
                                    sommaf3cont = sommaf3 - energyf(n)
                                    sommaf4cont = sommaf4 - energyf(n)
                                    sommaf6cont = sommaf6 - energyf(n)
                                    cont3=n3+1
                                    cont4=n4+1
                                    cont6=n6+1
                                    end if
                                    go to 3
                                    end if
                                if((yBS(j).GT.e).and.(yBS(j).LT.b)) then
                        energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0  ! faccia 3
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 6

                                    if (dati(k).NE.0.d0) then
                                    sommaf3cont = sommaf3 - energyf(n)
                                    sommaf6cont = sommaf6 - energyf(n)
                                    cont3=n3+1
                                    cont6=n6+1
                                    end if
                                    go to 2
                            end if
                                if(yBS(j).EQ.b) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0       ! faccia 2
                        energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3
                        energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                                    if (dati(k).NE.0.d0) then
                                    sommaf2cont = sommaf2 + energyf(n)
                                    sommaf3cont = sommaf3 + energyf(n)
                                    sommaf6cont = sommaf6 + energyf(n)
                                    cont2=n2+1
                                    cont3=n3+1
                                    cont6=n6+1
                                    end if
                                    go to 3
                                    end if
                            end if
                        end if


! FACCIA 4

                        if(yBS(j).EQ.e) then
                            if(xBS(j).EQ.f) then
                                    if(zBS(j).EQ.h) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0       ! faccia 4
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 5

                                    if (dati(k).NE.0.d0) then
                                    sommaf4cont = sommaf4 + energyf(n)
                                    sommaf5cont = sommaf5 + energyf(n)
                                    cont4=n4+1
                                    cont5=n5+1
                                    end if
                                    go to 2
                                    end if
                                if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
                        energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 4
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0 ! faccia 5

                                    if (dati(k).NE.0.d0) then
                                    sommaf4cont = sommaf4 + energyf(n)
                                    sommaf5cont = sommaf5 + energyf(n)
                                    cont4=n4+1
                                    cont5=n5+1
                                    end if
                                    go to 2
                            end if
                                if(zBS(j).EQ.c) then
                        energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 3
                        energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 4
```

```
energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 5

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf4cont = sommaf4 - energyf(n)
                sommaf5cont = sommaf5 - energyf(n)
                cont3=n3+1
                cont4=n4+1
                cont5=n5+1
                end if
                go to 3
                end if
        end if

        if((xBS(j).GT.f).and.(xBS(j).LT.g)) then
            if(zBS(j).EQ.h) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0        ! faccia 4

                if (dati(k).NE.0.d0) then
                sommaf4cont = sommaf4 + energyf(n)
                cont4=n4+1
                end if
                go to 1
                end if
                if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 4

                if (dati(k).NE.0.d0) then
                sommaf4cont = sommaf4 + energyf(n)
                cont4=n4+1
                end if
                go to 1
        end if
            if(zBS(j).EQ.c) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 4

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf4cont = sommaf4 - energyf(n)
                cont3=n3+1
                cont4=n4+1
                end if
                go to 2
                end if
        end if

        if(xBS(j).EQ.g) then
            if(zBS(j).EQ.h) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0       ! faccia 4
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                if (dati(k).NE.0.d0) then
                sommaf4cont = sommaf4 + energyf(n)
                sommaf6cont = sommaf6 + energyf(n)
                cont4=n4+1
                cont6=n6+1
                end if
                go to 2
                end if
            if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 4
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0 ! faccia 6

                if (dati(k).NE.0.d0) then
                sommaf4cont = sommaf4 + energyf(n)
                sommaf6cont = sommaf6 + energyf(n)
                cont4=n4+1
                cont6=n6+1
                end if
                go to 2
        end if
            if(zBS(j).EQ.c) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 4
energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf4cont = sommaf4 - energyf(n)
                sommaf6cont = sommaf6 - energyf(n)
                cont3=n3+1
                cont4=n4+1
                cont6=n6+1
                end if
                go to 3
                end if
        end if
end if

! FACCIA 5

if(xBS(j).EQ.f) then
```

```
    if(yBS(j).EQ.e) then
            if(zBS(j).EQ.h) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0       ! faccia 4
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 5

                if (dati(k).NE.0.d0) then
                sommaf4cont = sommaf4 + energyf(n)
                sommaf5cont = sommaf5 + energyf(n)
                cont4=n4+1
                cont5=n5+1
                end if
                go to 2
                end if
            if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 4
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0 ! faccia 5

                if (dati(k).NE.0.d0) then
                sommaf4cont = sommaf4 + energyf(n)
                sommaf5cont = sommaf5 + energyf(n)
                cont4=n4+1
                cont5=n5+1
                end if
                go to 2
        end if
            if(zBS(j).EQ.c) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 4
energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 5

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf4cont = sommaf4 - energyf(n)
                sommaf5cont = sommaf5 - energyf(n)
                cont3=n3+1
                cont4=n4+1
                cont5=n5+1
                end if
                go to 3
                end if
        end if

        if((yBS(j).GT.e).and.(yBS(j).LT.b)) then
            if(zBS(j).EQ.h) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 5

                if (dati(k).NE.0.d0) then
                sommaf5cont = sommaf5 + energyf(n)
                cont5=n5+1
                end if
                go to 1
                end if
            if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 5

                if (dati(k).NE.0.d0) then
                sommaf5cont = sommaf5 + energyf(n)
                cont5=n5+1
                end if
                go to 1
        end if
            if(zBS(j).EQ.c) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 5

                if (dati(k).NE.0.d0) then
                sommaf3cont = sommaf3 - energyf(n)
                sommaf5cont = sommaf5 - energyf(n)
                cont3=n3+1
                cont5=n5+1
                end if
                go to 2
                end if
        end if

        if(yBS(j).EQ.b) then
            if(zBS(j).EQ.h) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0      ! faccia 2
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 5

                if (dati(k).NE.0.d0) then
                sommaf2cont = sommaf2 + energyf(n)
                sommaf5cont = sommaf5 + energyf(n)
                cont2=n2+1
                cont5=n5+1
                end if
                go to 2
                end if
            if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 2
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0 ! faccia 5

                if (dati(k).NE.0.d0) then
```

```
                    sommaf2cont = sommaf2 + energyf(n)
                    sommaf5cont = sommaf5 + energyf(n)
                    cont2=n2+1
                    cont5=n5+1
                    end if
                    go to 2
            end if
                if(zBS(j).EQ.c) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 2
energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3
energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 5

                    if (dati(k).NE.0.d0) then
                    sommaf2cont = sommaf2 + energyf(n)
                    sommaf3cont = sommaf3 + energyf(n)
                    sommaf5cont = sommaf5 + energyf(n)
                    cont2=n2+1
                    cont3=n3+1
                    cont5=n5+1
                    end if
                    go to 3
                    end if
            end if
end if

! FACCIA 6

if(xBS(j).EQ.g) then
    if(yBS(j).EQ.e) then
                if(zBS(j).EQ.h) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 4
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                    if (dati(k).NE.0.d0) then
                    sommaf4cont = sommaf4 + energyf(n)
                    sommaf6cont = sommaf6 + energyf(n)
                    cont4=n4+1
                    cont6=n6+1
                    end if
                    go to 2
                    end if
                if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 4
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0 ! faccia 6

                    if (dati(k).NE.0.d0) then
                    sommaf4cont = sommaf4 + energyf(n)
                    sommaf6cont = sommaf6 + energyf(n)
                    cont4=n4+1
                    cont6=n6+1
                    end if
                    go to 2
            end if
                if(zBS(j).EQ.c) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0        ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 4
energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                    if (dati(k).NE.0.d0) then
                    sommaf3cont = sommaf3 - energyf(n)
                    sommaf4cont = sommaf4 - energyf(n)
                    sommaf6cont = sommaf6 - energyf(n)
                    cont3=n3+1
                    cont4=n4+1
                    cont6=n6+1
                    end if
                    go to 3
                    end if
            end if

            if((yBS(j).GT.e).and.(yBS(j).LT.b)) then
                if(zBS(j).EQ.h) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0             ! faccia 6

                    if (dati(k).NE.0.d0) then
                    sommaf6cont = sommaf6 + energyf(n)
                    cont6=n6+1
                    end if
                    go to 1
                    end if
                if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 6

                    if (dati(k).NE.0.d0) then
                    sommaf6cont = sommaf6 + energyf(n)
                    cont6=n6+1
                    end if
                    go to 1
            end if
                if(zBS(j).EQ.c) then
energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0            ! faccia 3
energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6
```

```
                                        if (dati(k).NE.0.d0) then
                                        sommaf3cont = sommaf3 - energyf(n)
                                        sommaf6cont = sommaf6 - energyf(n)
                                        cont3=n3+1
                                        cont6=n6+1
                                        end if
                                        go to 2
                                        end if
                                end if

                                if(yBS(j).EQ.b) then
                                    if(zBS(j).EQ.h) then
                            energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0       ! faccia 2
                            energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                                        if (dati(k).NE.0.d0) then
                                        sommaf2cont = sommaf2 + energyf(n)
                                        sommaf6cont = sommaf6 + energyf(n)
                                        cont2=n2+1
                                        cont6=n6+1
                                        end if
                                        go to 2
                                        end if
                                    if((zBS(j).GT.h).and.(zBS(j).LT.c)) then
                            energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0   ! faccia 2
                            energyf(n+1)=dati(k)*q*r/1000.0d0*D*10.d0 ! faccia 6

                                        if (dati(k).NE.0.d0) then
                                        sommaf2cont = sommaf2 + energyf(n)
                                        sommaf6cont = sommaf6 + energyf(n)
                                        cont2=n2+1
                                        cont6=n6+1
                                        end if
                                        go to 2
                                end if
                                        if(zBS(j).EQ.c) then
                            energyf(n)=dati(k)*q*r/1000.0d0*D*10.d0           ! faccia 2
                            energyf(n+1)=-dati(k)*q*r/1000.0d0*D*10.d0    ! faccia 3
                            energyf(n+2)=dati(k)*q*r/1000.0d0*D*10.d0     ! faccia 6

                                        if (dati(k).NE.0.d0) then
                                        sommaf2cont = sommaf2 + energyf(n)
                                        sommaf3cont = sommaf3 + energyf(n)
                                        sommaf6cont = sommaf6 + energyf(n)
                                        cont2=n2+1
                                        cont3=n3+1
                                        cont6=n6+1
                                        end if
                                        go to 3
                                        end if
                                end if
                            end if

! FACCIA 2'

                            if(yBS(j).EQ.b2) then
                                if((xBS(j).GT.f2).and.(xBS(j).LT.g2)) then
                                        if((zBS(j).GT.c2).and.(zBS(j).LT.c)) then
                            energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0  ! faccia 2'

                                        if (dati(k).NE.0.d0) then
                                        sommaf2bcont = sommaf2b - energyf(n)
                                        cont2b=n2b+1
                                        end if
                                        go to 1
                                end if
                                end if
                            end if

! FACCIA 3'

                            if(zBS(j).EQ.c2) then
                                    if((xBS(j).GT.f2).and.(xBS(j).LT.g2)) then
                                        if((yBS(j).GT.e2).and.(yBS(j).LT.b2)) then
                            energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 3'

! Inserimento zona colda 3b2 e meno calda 3b1, il resto è 3b

        if (dati(k).NE.0.d0) then
                                                        if((xBS(j).GE.f22).and.(xBS(j).LE.g22))
    then

        if((yBS(j).GE.e22).and.(yBS(j).LE.b22)) then

        if((xBS(j).GE.f21).and.(xBS(j).LE.g21)) then

        if((yBS(j).GE.e21).and.(yBS(j).LE.b21)) then

                                                                        sommaf3b2cont =
    sommaf3b2 - energyf(n)
                                                                        cont3b2=n3b2+1

    else
```

```
                                                                                sommaf3b1cont =
    sommaf3b1 - energyf(n)
                                                                                cont3b1=n3b1+1
    end if
    else
    sommaf3b1cont = sommaf3b1 - energyf(n)
    cont3b1=n3b1+1
    end if
    else
    sommaf3bcont = sommaf3b - energyf(n)
    cont3b=n3b+1
    end if
    else
    sommaf3bcont = sommaf3b - energyf(n)
    cont3b=n3b+1
    end if
    end if

! Fine inserimento di 3 zone calde in faccia 3'


                                go to 1
                                end if
                                end if
                        end if

! FACCIA 4'

                        if(yBS(j).EQ.e2) then
                                if((xBS(j).GT.f2).and.(xBS(j).LT.g2)) then
                                    if((zBS(j).GT.c2).and.(zBS(j).LT.c)) then
                        energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 4'

                                    if (dati(k).NE.0.d0) then
                                    sommaf4bcont = sommaf4b - energyf(n)
                                    cont4b=n4b+1
                                    end if
                                    go to 1
                                end if
                                end if
                        end if

! FACCIA 5'

                        if(xBS(j).EQ.f2) then
                                if((yBS(j).GT.e2).and.(yBS(j).LT.b2)) then
                                    if((zBS(j).GT.c2).and.(zBS(j).LT.c)) then
                        energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 5'

                                    if (dati(k).NE.0.d0) then
                                    sommaf5bcont = sommaf5b - energyf(n)
                                    cont5b=n5b+1
                                    end if
                                    go to 1
                                end if
                                end if
                        end if

! FACCIA 6'

                        if(xBS(j).EQ.g2) then
                                if((yBS(j).GT.e2).and.(yBS(j).LT.b2)) then
                                    if((zBS(j).GT.c2).and.(zBS(j).LT.c)) then
                        energyf(n)=-dati(k)*q*r/1000.0d0*D*10.d0          ! faccia 6'

                                    if (dati(k).NE.0.d0) then
                                    sommaf6bcont = sommaf6b - energyf(n)
                                    cont6b=n6b+1
                                    end if
                                    go to 1
                                end if
                                end if
                        end if


                        else
    go to 4
3   cont=n+3

    go to 4
2   cont=n+2

    go to 4
1   cont=n+1

    go to 4
4   continue


                end if

                if (dist<ok) then
```

```
                        ok=dist
                        energy(m)=dati(k)*q*r/1000.0d0

                alfa = k

[…]

                        else
    go to 8
7   cont=n+3

    go to 8
6   cont=n+2

    go to 8
5   cont=n+1

    go to 8
8   continue



                end if
                if (dist.EQ.ok) then
                        ok=dist
                        energy(m)=dati(k)*q*r/1000.0d0

                alfa = k

[…]

                        else
    go to 12
11  cont=n+3

    go to 12
10   cont=n+2

    go to 12
9   cont=n+1

    go to 12
12   continue

                end if

        else
                go to 100
        end if

100     continue

                        end if
                end if
        end if

        end do

        m=m+1
        ok=0.d0
        n=cont

        m1=contm1
        m2=contm2
        n2=cont2
        n3=cont3
        n4=cont4
        n5=cont5
        n6=cont6
        n2b=cont2b
        n3b=cont3b
        n3b1=cont3b1
        n3b2=cont3b2
        n4b=cont4b
        n5b=cont5b
        n6b=cont6b


        sommas1=sommas1cont
        sommas2=sommas2cont

        sommaf2=sommaf2cont
        sommaf3=sommaf3cont
        sommaf4=sommaf4cont
        sommaf5=sommaf5cont
        sommaf6=sommaf6cont
        sommaf2b=sommaf2bcont
        sommaf3b=sommaf3bcont
        sommaf3b1=sommaf3b1cont
        sommaf3b2=sommaf3b2cont
        sommaf4b=sommaf4bcont
        sommaf5b=sommaf5bcont
```

```
         sommaf6b=sommaf6bcont

         end do

         medias1 = sommas1cont/(contm1-1)
         medias2 = sommas2cont/(contm2-1)
         mediaf2 = sommaf2cont/(cont2-1)
         mediaf3 = sommaf3cont/(cont3-1)
         mediaf4 = sommaf4cont/(cont4-1)
         mediaf5 = sommaf5cont/(cont5-1)
         mediaf6 = sommaf6cont/(cont6-1)
         mediaf2b = sommaf2bcont/(cont2b-1)
         mediaf3b = sommaf3bcont/(cont3b-1)
         mediaf3b1 = sommaf3b1cont/(cont3b1-1)
         mediaf3b2 = sommaf3b2cont/(cont3b2-1)
         mediaf4b = sommaf4bcont/(cont4b-1)
         mediaf5b = sommaf5bcont/(cont5b-1)
         mediaf6b = sommaf6bcont/(cont6b-1)


write (7,1001) medias1, contm1-1
write (7,*)
write (7,1014) medias2, contm2-1
write (7,*)
write (7,1002) mediaf2, cont2-1
write (7,*)
write (7,1003) mediaf3, cont3-1
write (7,*)
write (7,1004) mediaf4, cont4-1
write (7,*)
write (7,1005) mediaf5, cont5-1
write (7,*)
write (7,1006) mediaf6, cont6-1
write (7,*)
write (7,1007) mediaf2b, cont2b-1
write (7,*)
write (7,1008) mediaf3b, cont3b-1
write (7,*)
write (7,1009) mediaf3b1, cont3b1-1
write (7,*)
write (7,1010) mediaf3b2, cont3b2-1
write (7,*)
write (7,1011) mediaf4b, cont4b-1
write (7,*)
write (7,1012) mediaf5b, cont5b-1
write (7,*)
write (7,1013) mediaf6b, cont6b-1


return

996 format(i10,4(E15.4))
!996 format(2(1x,i10,4(E15.4)))
997 format(i10,3(E15.4))
1001 format ('medias1=', E10.4,i10)
1014 format ('medias2=', E10.4,i10)
1002 format ('mediaf2=', E10.4,i5)
1003 format ('mediaf3=', E10.4,i5)
1004 format ('mediaf4=', E10.4,i5)
1005 format ('mediaf5=', E10.4,i5)
1006 format ('mediaf6=', E10.4,i5)
1007 format ('mediaf2b=', E10.4,i5)
1008 format ('mediaf3b=', E10.4,i5)
1009 format ('mediaf3b1=', E10.4,i5)
1010 format ('mediaf3b2=', E10.4,i5)
1011 format ('mediaf4b=', E10.4,i5)
1012 format ('mediaf5b=', E10.4,i5)
1013 format ('mediaf6b=', E10.4,i5)
pause
stop

end subroutine trasportobaric

-----------------------------------------------------------------------------------------

program temp

! Prende il campo delle temperature dall'analisi termica transitoria in Straus e li riporta alla
    mesh analizzata in NEWCON3D.


implicit none
integer                          ::  unit1, unit2, unit3,unit4


integer, parameter               ::  nnodi = 7733
integer, parameter               ::  nnodid = 121520
integer, parameter               ::  nmax = 400

!---------------------------------

real*8,dimension(nnodi,3)        ::  A
```

```
real*8,dimension(nnodid,3)        ::  Ad
real*8,dimension(nnodid)          ::  tempMODEL
real*8,dimension(nmax)            ::  tempmod

real*8                            ::  L, q, r

real*8                            ::  dist, ok

integer                           ::  i,j,k,m

    !-------------------
    unit1       = 9
    unit2       =10
    unit3       =11
    unit4       =12

    open(unit1, file='TempMODEL_132_e4m.dat',status='old')
    open(unit2, file='Straus txt_nodi2.txt',status='old')
    open(unit3, file='Copia di da Straus txt.dat',status='old')
    open(unit4, file='Tempmod_132_e4m.dat',status='unknown')

! Lettura temperature mesh Straus

      do i=1,nnodid
      read(unit1,*) tempMODEL(i)
      end do


! Lettura nodi mesh Straus da da Straus.dat per formattazione nel main

      do i=1,nnodid
      read(unit2,995) (Ad(i,j), j=1,3)
      end do


      do i=1,nnodi
      read(unit3,995) (A(i,j), j=1,3)
      end do


! Inizializzazione variabili

      m=1

      dist=0.d0
      tempmod=0.d0

!----------------------
      L = 200.d0
      q = 3720.d0           ! coordinata X degli elementi ambiente
      r= 20.d0
!----------------------

! Per ogni nodo Straus, se la distanza col baricentro del volumetto si trova entro la dimensione
    della mesh Straus in ogni direzione cartesiana, calcolo distanza 3D

      do j=1,nnodi

      if (A(j,3).EQ.q) then

        do k=1,nnodid

        if (k>nnodid) exit

           if (Abs(A(j,3)-Ad(k,3)).LE.r) then
                 if (Abs(A(j,2)+4500.0d0-Ad(k,2)).LE.L) then         ! y centrata in 5m da terra
    (per esse ho fluences da Fluka)
                       if (Abs(A(j,1)+3500.0d0-Ad(k,1)).LE.L) then   ! x centrata in 4m da bordo
    libero


        dist=(((A(j,1)+3500.0d0)-Ad(k,1))**2+((A(j,2)+4500.0d0)-Ad(k,2))**2+(A(j,3)-
    Ad(k,3))**2)**0.5d0


!       Se la distanza 3D è minore di L/2 allora prendo il valore corrispondente del baricentro per
    fluence(i) e aggiorno la distanza come distanza minima (ok) (caso a parte per iterazione 0:
    ok=0.d0)

        if (dist.LE.L) then
                if (ok.EQ.0.d0) then
                        ok=dist
                        tempmod(m)=tempMODEL(k)
                end if
                if (dist<ok) then
                        ok=dist
                        tempmod(m)=tempMODEL(k)
                end if
                if (dist.EQ.ok) then
                        ok=dist
                        tempmod(m)=tempMODEL(k)
                end if
```

```
        else
                go to 100
        end if

100     continue

                        end if
                end if
        end if

        end do

        m=m+1
        ok=0.d0

      end if
      end do

! Stampa condizioni al contorno come T per modello NEWCON3D

    do i=1,nmax
        write(unit4,9) tempmod(i)
    end do

    close(unit1)
    close(unit2)
    close(unit3)
    close(unit4)
pause
stop

995 format(29x,3(E25.14))
  9 format(F9.4)

end program  temp.
```