University of Montana

ScholarWorks at University of Montana

Graduate Student Theses, Dissertations, & Professional Papers

Graduate School

2004

Ignition of crown fuels above a spreading surface fire

Miguel Gomes da Cruz
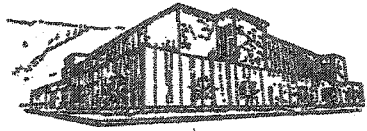*The University of Montana*

Follow this and additional works at: https://scholarworks.umt.edu/etd

Let us know how access to this document benefits you.

The University of

Montana

Permission is granted by the author to reproduce this material in its entirety, provided that this material is used for scholarly purposes and is properly cited in published works and reports.

Please check "Yes" or "No" and provide signature

Yes, I grant permission   _X_

No, I do not grant permission   _____

Author's Signature: _Mibun 607-5 4 cwz_

Date: _July 9, 2004_

Any copying for commercial purposes or financial gain may be undertaken only with the author's explicit consent.

8/98

IGNITION OF CROWN FUELS ABOVE A SPREADING SURFACE FIRE

Submitted by

Miguel Gomes da Cruz

B.S. Forestry, Coimbra Polytechnic Institute, Portugal, 1994

M.Sc. Forestry, University of Montana, 1999

Presented in partial fulfillment of the requirements

for the degree of

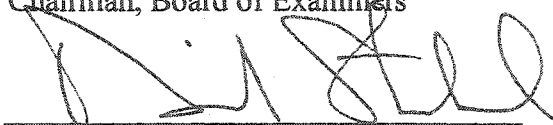Doctor of Philosophy

University of Montana

Missoula, Montana

2004

Approved by:

_____
Chairman, Board of Examiners

_____
Dean, Graduate School

7-8-04
_____
Date

UMI Number: 3134701

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.
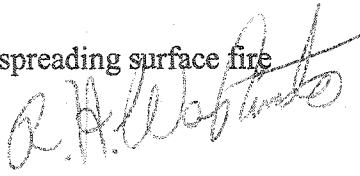
UMI®

UMI Microform 3134701

Copyright 2004 by ProQuest Information and Learning Company.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

Gomes da Cruz, Miguel, Ph.D., June 2004   Forestry

Ignition of crown fuels above a spreading surface fire

Director: Ronald H. Wakimoto

This thesis describes a two-part study aimed at understanding the physical processes leading to the ignition of forest canopy fuels above a spreading surface fire. The study comprises a theoretical component focusing on the development of a crown fuel ignition model based on heat transfer principles (Chapter 1) and an experimental component aimed at comprehending how fuels, weather and fire behavior variables determine upward convective and radiative heat fluxes (Chapter 2).

The crown fuel ignition model integrates the properties of the heat source as defined by the surface fire flame front and crown fuel characteristics, which determine the heat requirements for crown ignition. Fuel particle temperature increase was determined through a simplified energy balance equation relating heat absorption to fuel particle temperature. The final model output is the temperature of the crown fuel particles, which upon reaching ignition temperature are assumed to ignite. Model results indicate that the primary factors influencing crown fuel ignition are those determining the depth of the surface fire burning zone and the vertical distance between the ground and surface fuel strata in the fuel complex and the lower base of the crown fuel layer. The coupling of the crown fuel ignition model with models determining the spread of crown fires allows for the prediction of the potential of sustained crowning.

A number of laboratory and outdoor experimental fires were instrumented to measure upward radiative and convective heat fluxes. No evidence was found to a preponderance of one heat flux process over the other. The fire behavior characteristics that were most related with the upward heat fluxes were reaction time and predicted flame height. No significant relationships were found between fire intensity measures, such as fireline and reaction intensity and various measures quantifying upward heat flux, namely peak and cumulative heat fluxes. The use of models to predict upward radiative heat flux and buoyant plume behavior showed no evidence of bias, although predictions showed some degree of variability. Analysis of the observed heat fluxes and model predictions indicates that the heat flux partitioning into convective and radiative processes is highly dynamic in time and space, and determined by fuel complex characteristics and burning conditions.

Keywords: fire behavior, modeling, heat transfer, crown fire initiation.

Acknowledgements

Table of Contents

List of Tables

CHAPTER 1:

1

Development of a model describing the ignition of crown fuels above a spreading surface fire.

Abstract:

A model was developed to predict the ignition of forest canopy fuels above a surface fire based on simple heat transfer theory. The crown fuel ignition model (hereafter referred to as CFIM) is based on first principles, integrating: (1) the characteristics of the heat source as defined by surface fire flame front properties; (2) buoyant plume dynamics; (3) heat sink as described by the crown fuel particles characteristics; and (4) heat transfer (gain and losses) to the crown fuels. Fuel particle temperature increase is determined through an energy balance relating heat absorption to fuel particle temperature. The final model output is the temperature of the crown fuel particles which upon reaching ignition temperature are assumed to ignite. Model results indicate that the primary factors influencing crown fuel ignition are those determining the depth of the surface fire burning zone and the vertical distance between the ground/surface fuel strata and the lower boundary of the crown fuel layer in the fuel complex. The CFIM does not discriminate the on-set of crown fire spread *per se*. The coupling of the CFIM with models determining the spread rate of propagation of crown fires allows for the prediction of the potential of sustained crowning. Evaluations carried out against high intensity experimental fire data and predictions from other models gave encouraging results relative to the validity of the model system. The CFIM has the potential to be used in a number of fire management decision support systems.

Keywords: fire behavior, modeling, heat transfer, crown fire initiation.

1.1. Introduction

Knowledge of fire behavior is an important component in fire management decision-making. Fire behavior modeling allows managers to predict fire potential with a certain level of confidence. Fire behavior information is used in fire management activities such as prescribed fire planning, fuel hazard assessment and development of fire suppression strategies on wildfires. Of the various fire behavior descriptors used by fire managers in their operational planning, the onset of crowning assumes particular importance. The onset of crowning marks the transition between a surface fire and a fire involving all strata of the fuel complex. After crowning, fires have been observed to increase their rate of spread, intensity and spotting activity (e.g., Wade and Ward 1973, Simard et al. 1983, Albini 1999). Crown fires are virtually impossible to control by direct action (Albini and Stocks 1986, Alexander 2000) and are responsible for a large proportion of the overall area burned in large fires (e.g., Anderson 1968, Albini 1984, Graham 2003, Simard et al. 1983). The importance of the onset of crowning in determining overall fire potential has made it a common target variable when assessing the effectiveness of fuel management treatments in reducing fire potential (e.g., Scott 1999, Graham et al. 1999, Stephens 1998, Scott and Reinhardt 2001, Fulé et al. 2001, 2002, Keyes and O'Hara 2002).

Fire behavior modeling can be empirical or physical (or theoretical) or combination of both (Catchpole and de Mestre 1986, Pastor et al. 2003). Empirical models are based on the relationship between the response variable and explanatory variables without explicitly considering the controlling physical processes (e.g. Byram 1959, Cheney et al. 1998, Fernandes et al. 2000, Cruz et al. 2004). Physical models are

3

formulated as expressions of physical processes but rely on empirical data to some extent. For example, the physical fire spread models consider heat transfer processes (e.g., Pagni and Peterson 1973, Konev and Sukhinin 1977, Albini 1985, 1996) but derive combustion phenomena and its results, such as flame properties, from empirical or simplified relationships.

The increase in available inexpensive computing power made it possible to apply modern numerical methods to the theoretical analysis of the chemical and physical processes involved in a wildland fire. This approach attempts to describe fire phenomenology by numerically solving a set of equations describing the local conservation of mass, momentum, energy and species for the system. These equations are mathematical statements of the basic laws of physics and when applied to fire phenomenology allow for the incorporation of combustion, fluid dynamics, and heat and mass transport in both solid and gaseous phases. Examples of these fundamentally based models are Grishin and Perminov (1991), Grishin (1997), Linn (1997) and Morvan and Dupuy (2001). By their comprehensiveness, these models should be able to predict most of fire behavior phenomena and their interaction with small-scale meteorological conditions. This allows the modeling of fire as a complete, closed system. Nevertheless these systems can not be considered as pure physical models. These models are based on the derivation of the conservation equations with varying degrees of rigor, with their essential differences being in the way they treat combustion and heat transfer processes in the solid and gas phases, and consequently the demands upon empiricism. The current-state-of-knowledge in fire related processes lead to the use of numerous empirically based sub-models or constants to describe several phenomena where our knowledge of

4

the processes is still incomplete. Examples of these are the convective heat transfer to the unburned fuel particles, their thermal decomposition, flame structure and the formation and oxidation of soot. These general assumptions may suffice in an academic exercise but might not be realistic when applied to a real world wildland fire environment. These models are best seen as providing better insight into the processes involved than as models developed for predicting outcomes.

A further drawback of these fundamentally based models is that they have seldom been subjected to any evaluation against independent data. This might arise from their inherent complexity and computing requirements, which make them difficult to analyze except by the developer or a few researchers. Thus it is difficult to know whether the model accurately describes the processes it claims to represent. Given this, the review of previous models developed to predict the onset of crowning will disregard these theoretical models and focus on the analysis of the so called empirical based models.

Through the analysis of heat balances involved in surface and crown fires in pine stands, Molchanov (1957) estimated the amounts of surface fire heat output required to ignite crown fuels and described the influence of the effect of the amount of fuel in the crown, foliar moisture content, and foliage chemical composition on the onset of crowning. Fahnestock (1970) through his Crowning Key, produced one of the first tools to allow fire managers to assess crown fire potential in forest stands in the US. He identified several fuel complex characteristics that lead to the onset of crowning, namely, canopy cover density, existence of ladder fuels, and foliage state, and ranked their possible combinations into a scale describing the potential crown fire occurrence. Kilgore and Sando (1975) assessed crown fire potential in giant sequoia stands through

5

the knowledge of crown base height and crown volume ratio as a measure of canopy density. These quantitative descriptions of the crown fuel strata properties was not accompanied by any quantitative fire behavior models that could use their data in support of fire management decisions.

The semi-empirical approach to crown fire initiation modeling has lead to models suitable for operational implementation (e.g., Van Wagner 1977, Xanthopoulos 1990, Alexander 1998). Van Wagner (1977), through a combination of physical theory and empirical observation, defined quantitative criteria to predict the onset of crowning. His analysis was based on plume theory developed by Yih (1953, 1969, p. 413) that linked an idealized linear heat source with the maximum temperature attained at a certain height in the buoyant plume above. This relationship, based on dimensional analysis, was rearranged by Van Wagner (1977) to allow for the determination of a critical surface fireline intensity (as per Byram 1959) needed to induce crown combustion, as a function of canopy base height, heat required for ignition (as determined by the moisture content of the available canopy fuel), and a proportionality constant, "best regarded as an empirical constant of complex dimensions" (Van Wagner 1977). The proportionality constant was estimated by Van Wagner (1977) to be 0.01 based largely on a single experimental fire conducted in a red pine (*Pinus resinosa*) plantation stand (Alexander 1998).

Although Van Wagner's (1977) formulation is based on convective theory, the proportionality constant was derived from fireline intensity estimated from the total amount of fuel consumed as opposed to just the quantity involved in the active flame front. This measure of fireline intensity reflects the heat or energy release associated

6

with both flaming and smoldering combustion (Rothermel 1994). Van Wagner's (1977) model is presently used in whole or in part for assessing crown fire initiation in several North American fire behavior prediction systems (Van Wagner 1989, Forestry Canada Fire Danger Group 1992, Finney 1998, Scott and Reinhart 2001) and the basis for several field guides and aids (Rothermel 1983, Alexander 1988).

Xanthopoulos (1990) approached the development of a crown fire initiation model by deriving separate equations to: (1) predict time-temperature profiles at different heights in the convection plume above a fire; and (2) predict the time to ignition for foliage of three different conifer species (Xanthopoulos and Wakimoto 1993). The coupling of these equations with the output from the surface fire spread model of Rothermel (1972) with Albini's (1976) refinements as embodied in the BEHAVE system (Andrews 1986, Andrews and Chase 1989) would according to Xanthopoulos (1990), presumably overcome some of the limitations in the original Van Wagner (1977) model. Nevertheless, scale effects from the experimental laboratory set-up (i.e., small fire front width, no free convection, and low wind velocities) limits model application to real-world crown fires (Alexander 1998).

By combining and refining elements of the approaches taken by previous fire behavior modelers, coupled with new insights, Alexander (1998) was able to develop a simple algorithm to predict the onset of crowning. His model integrates the ignition requirements as defined by Xanthopoulos and Wakimoto's (1993) time-to-ignition equations with the convection plume thermal structure which is in turn deemed a function of fireline intensity, plume angle (as dictated by fireline intensity and wind speed), and

7

the surface fire reaction time. A proportionality constant was also used to apply to specific fuel complexes.

Cruz et al. (2004) modeled the likelihood of crown fire occurrence based on logistic regression analysis applied to an experimental fire behavior dataset. Their logistic model predicted the likelihood of crown fire occurrence based on three fire environment variables, namely the 10-m open wind speed, fuel strata gap (equivalent to live crown base height in some stands), estimated moisture content of fine dead fuels, and one fire behavior descriptor, namely surface fuel consumption. In contrast to the models developed by Van Wagner (1977), Xanthopoulos (1990) and Alexander (1998) that attempt to characterize and quantify the main processes involve in crown fire initiation, this logistic model does not directly incorporate any physical reasoning relative to the heat transfer processes taking place during a forest fire. Nevertheless, the analysis from the experimental fire dataset and model results provided qualitative information on the effects of several fire environment variables presumed to influence the onset of crowning. Foliar moisture content was not found to be a significant variable determining the occurrence of crown fires. Conversely, surface fuel consumption was found to be significant in determining the occurrence of crown fire behavior.

The objective of the present work was to describe a model developed to predict the onset of crowning based on fundamental heat transfer theory. The physical structure of the model should provide general applicability to diverse fuel complexes and allow the investigation of the role of fuel complex configuration in the heat transfer processes determining the initiation of crown fires. A list of symbols/abbreviations used throughout this study, along with their units is given on page 107.

1.2. Model idealization

The crown fuel ignition model developed in this study is based on a simplified fundamental modeling approach. By considering a surface fire spreading at a steady state, the model attempts to describe its upward radiative and convective heat source terms, determines heat transport to the fuels at the base of the crown and the change in the surface temperature of these fuels. The surface fire front is characterized by its: (1) rate of spread (2) reaction time; (3) flame depth; (4) flame height; (5) flame temperature-time profile above the fuelbed; and (6) the average gas temperature and vertical velocity at the tip of the flame. These characteristics define the initial conditions to solve the radiative heat transfer and buoyant plume models (Fig. 1.1). The canopy[1] fuel layer is assumed as a homogeneous layer of a certain depth composed by randomly distributed thermally thin cylindrical particles characterized by their surface area to volume ratio (σ), density (ρ_f), specific heat (c_f) and foliar moisture content (*FMC*). The heat transfer calculations are solved for the fuels at the base of the canopy fuel stratum. A nominal fuel strata gap (*FSG*) defines where vertical fire propagation occurs after ignition of crown fuels at that height. Fuel particle temperature was determined from a simplified energy balance equation integrating both radiative and convective heating and cooling terms. The temperature of the fuel particle being subject to the impinging convective and radiative heat fluxes is the final model output. A fuel ignition temperature of 600 K is assumed (de Mestre et al. 1989, Albini 1996). At this temperature it is presumed that piloted ignition of the fuel volatiles being released by the fuel particles occur, and fire

---

[1] In the present study the term "crown" is applied to describe aerial fuels at the tree level and "canopy" at the stand level.

9

propagates vertically into the crown. The sources of pilot ignition can be embers and firebrands carried in the buoyant plume, occasional flame flashes extending above the flame envelope, torching of understory vegetation (small trees and tall shrubs) and flame attachment and vertical spread in the lee side of tree trunks (Alexander 1998). In its present form the model does not consider the effect of lower ladder fuels and short range spotting in changing the geometry of the heat source and its power output. Fig. 1.2 describes the model system structure and the various sub-model linkages.
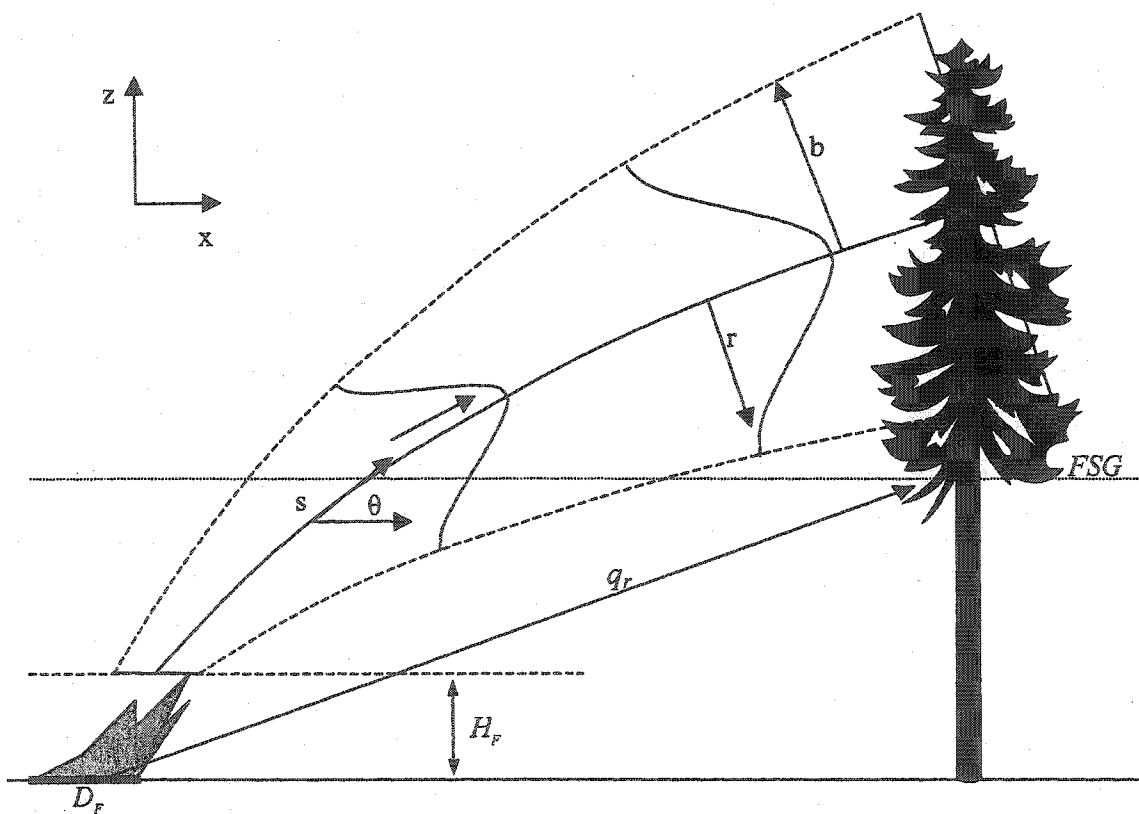
Figure 1.1. Diagram representing the two-dimensional implementation of the crown fuel ignition model (CFIM). Emphasis given to buoyant plume and radiative heat source dimensions and location. Description of the various symbols and abbreviations is given in page 107.

Input variables:
U_{io}, MC, FSG,
FMC, w_o, T_a,
SH, SD,
(surface fuelbed struc.)*

Wind profile
(eq. [1.10] and [1.11])

ROS
(any model)

τ_r
Nelson (2003b)

T_{Pmax}
(eq. [1.5])

I_B

H_P

D_P

T_{pi}
b_{pi}
U_{pi}

Plume model
(eq. [A11] and [A27])

T_p, b_p, U_p

I_{12}
(eq. [1.25])

h_c
(eq. [1.19])

FMC, σ

Fuel particle heating
(eq. [1.31])

Is T_f
> 600 K?

Crown fire initiation is possible.
Check for potential for crown fire
spread

Ignition of crown fuels
is unlikely
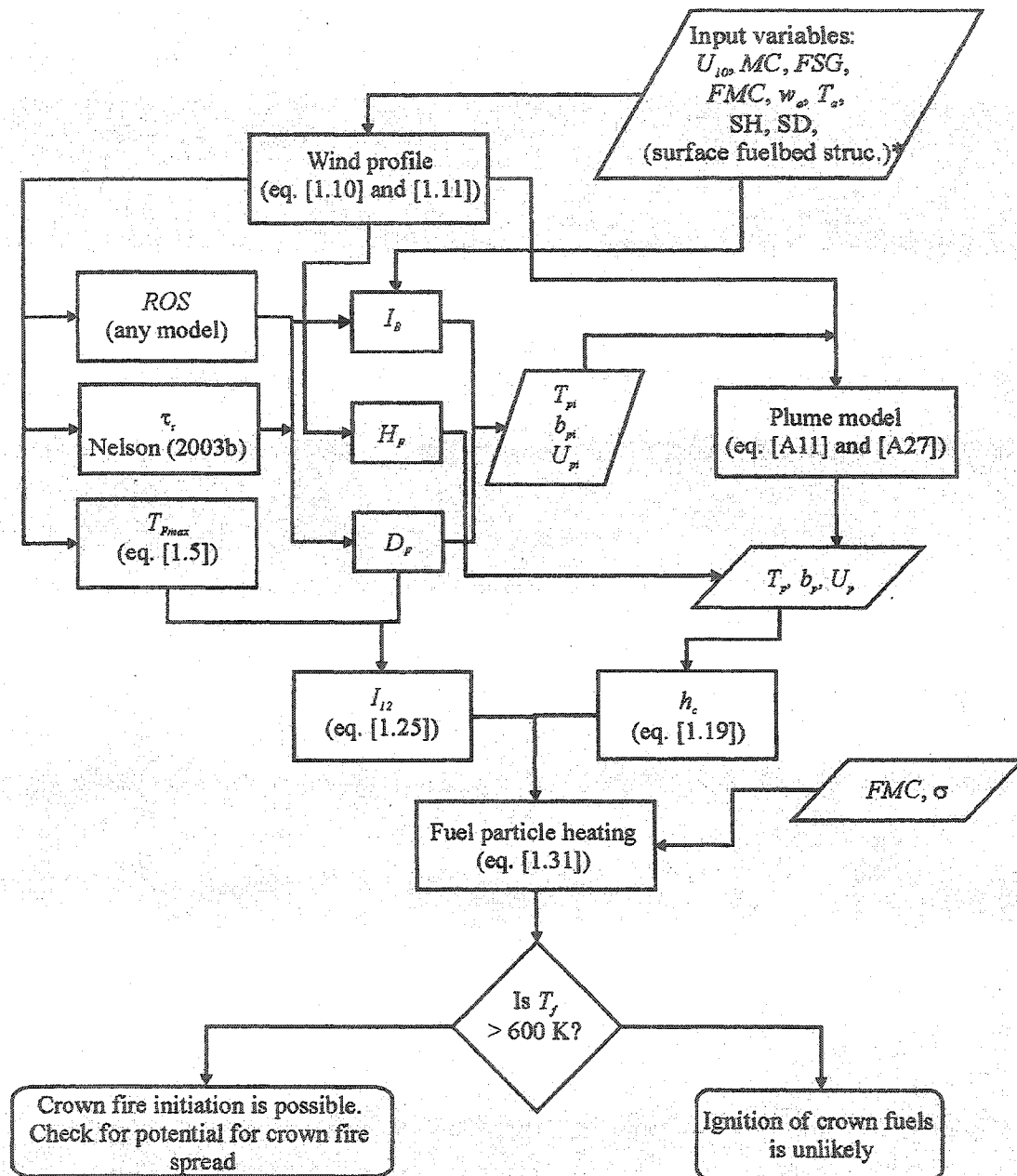
Figure 1.2. Flow diagram describing crown fuel ignition model (CFIM) and the various sub-model linkages. Description of the various symbols and abbreviations is given in page 107. * - The required surface fuelbed structure parameters depends on the model used for prediction of surface fire behavior.

11

1.3. Model structure

1.3.1. The Heat Source

The final output of the model system described in Fig. 1.2 is the temperature of the fuels situated at the base of the canopy fuel layer. Being this fuel temperature determined from the heat absorbed by the fuel particles, it is required to characterize the surface fire as a heat source so that heat transport through the sub-canopy space and heat absorption calculations can be made. The modeling approach sought requires the specification of the radiative and convective heat sources separately.

1.3.1.1. The radiative heat source

For the radiative heat transfer calculations the flame front is idealized as a horizontal planar radiative surface at the top of the surface fuelbed with dimensions of flame front width by flame depth (Fig. 1.1) and radiating as a black body, i.e., with a diffuse directional distribution of emissions and an emissivity of 1.0. The radiant heat flux (E) leaving the radiating surface is given by integrating the radiative intensity, obtained from the Stefan-Boltzmann equation, over the flame surface dimensions:

$$[1.1] \quad E = \int_0^{D_F} \int_0^{W_F} \varepsilon \cdot \sigma_{SB} \cdot T_F^{\,4} \, dx \, dy$$

where ε is the flame emissivity; σ_{SB} is the Stefan-Boltzmann constant, $5.67 \cdot 10^{-8}$ W m^{-2} K^{-4}; and T_F is the flame temperature in K. Flame depth is estimated from the product of the surface fire rate of spread and reaction time. Given the assumption of $\varepsilon = 1.0$, flame temperature is considered to be the flame radiometric temperature. The flame radiometric temperature is the effective blackbody temperature, i.e., the equivalent

12

radiating temperature of a region of space assuming an emissivity of 1.0 (Sullivan et al. (2003). Flame radiosity is a function of flame radiometric temperature, which is not constant along the reaction zone (e.g., Mendes-Lopes et al. 2003, Morvan and Dupuy 2001). The combustion processes in the surface fuelbed dictate that the gas temperature at the top of the fuelbed can be characterized by three distinct phases (Fendell and Wolff 2001): (1) a preheating phase described by a slow rise in temperature; (2) the onset of flaming characterized by a rapid increase in temperature until the maximum temperature is attained; and (3) a gradual decrease in temperature associated with the residual release of volatiles and radiation from the partially combusted fuels. This profile of gas temperature on the top of the fuelbed is dependent on several processes. Factors such as the amount and rate of fuel volatiles release, the amount of oxygen available to react with the volatiles and radiation emitted by the incandescent soot particles formed due to the inefficient partial oxidation of fuel will determine local flame temperature (Saito 2001). Assuming steady state fire propagation, the temperature along the depth of the flame front, $T_F(x)$, is a function of time, the so-called temperature-time (T-T) profile.

A simple model for the T-T profile was developed to integrate with eq. [1.1]. As we are interested in modeling radiation solely from flaming combustions the model considers only the last two of the three phases identified by Fendell and Wolff (2001). The T-T model is composed of three distinct parts: (1) a model describing the nondimensional shape of the T-T profile; (2) a model to predict the maximum temperature, T_{Fmax}; and (3) a numerical method to find the shape of the rising and decay components of the temperature curve.

13

The Gaussian shape form of the nondimensional temperature rise was described by (Weber et al. 1995):

$$[1.2] \quad \Delta T_F(t) \uparrow = \exp\left(\frac{-(t + \lambda_F)^2}{\beta_F^2}\right) .$$

where t is time (sec), β_F is an entrainment constant, and λ_F is a time adjustment variable used to locate the reaction zone interface in the non-dimensional curve at $t = 0$. For the temperature fall section, Newton law of cooling states:

$$[1.3] \quad \frac{dT_F}{dt} = -\gamma_F (T - T_a)$$

which upon integration yields (Weber et al. 1995):

$$[1.4] \quad \Delta T_F(t) \downarrow = C \cdot \exp(-\gamma_F \cdot t)$$

where γ_F is a proportionality constant determining the cooling rate and C the constant of integration. Both eqs. [1.2] and [1.4] are nondimensional and an estimate of the maximum temperature in the flame and reaction time are required to find the solution of the two-equation system.

An empirical model for maximum flame temperature on the top of the fuelbed was developed through non-linear regression analysis of an experimental fire database including both laboratory and outdoor experimental fires. The laboratory fires were conducted in the wind tunnel of the Fire Sciences Laboratory, Missoula MT (Catchpole et al. 1993, 1998). The outdoor fire data were obtained from experimental fires in shrublands and operational prescribed fires. Details regarding these fires are given in

14

Chapter 2. The model was idealized as dependent on certain fire environment properties. The best model fit was:

[1.5] $T_{F \max} = 300 + \beta_1 \cdot w_a + \beta_2 \cdot U_s^{\beta_3} + \beta_4 \cdot MC^{\beta_5}$

where the regression coefficients are (with standard errors in parenthesis): $\beta_1 = 300.684$ (72.03); $\beta_2 = 136.791$ (72.80); $\beta_3 = 0.506$ (66.63); $\beta_4 = 100.448$ (0.38); and $\beta_5 = -0.531$ (0.21). The model explained 50% of the variability in maximum flame temperature in the dataset.

The temperature - time distribution model is applicable only for temperatures above 600 K and is implemented through the following procedure: (1) determine maximum flame temperature following eq. [1.5]; (2) given the maximum flame temperature, determine the nondimensional temperature associated to the arrival of the reaction zone (600 K) from:

[1.6] $AT_{600} = \dfrac{\left(T_{ig} - T_a\right)}{\left(T_{F \max} - T_a\right)}$

(3) from T_{Fmax} and τ_r the location of AT_{600} in the T-T curve is found by solving eq. [1.2] for β_F based on Newton's method. This method assumes a fixed rate of increase of gas temperature from the arrival of the ignition interface to attainment of the T_{Fmax}. From the analysis of experimental fire T-T curves (Chapter 2), the average rate of temperature increase after the arrival of the ignition interface was assumed to be 60 K sec^{-1}; (4) iteratively find γ_F in eq. [1.4] based on Newton's method so that the decay curve would match the 600 K temperature at $t = \tau_r$, the reaction time. The T-T profile model was then given by coupling the two equations:

15

$$[1.7] \quad T_F(t) = \begin{cases} T_a + (T_{F\,max} - T_a) \cdot \exp\left(\dfrac{-(t + \lambda_F)^2}{\beta_F^2}\right), & \rightarrow \Delta T \uparrow \\[2ex] T_a + (T_{F\,max} - T_a) \cdot \exp(-\gamma_F \cdot (t + \lambda_F)), & \rightarrow \Delta T \downarrow \end{cases}$$

To define both the radiating surface dimensions and the T-T profile an estimate of the surface fire reaction time is required. Reaction time is estimated through Nelson's (2003b) model, which calculates the fuelbed reaction time of a surface fire from the flameout time of a single particle and fuel bed structural properties. This model is based on a simplified description of essential processes determining the thermochemical properties of flame gases, heat transfer and combustion rates in the surface fuelbed (Albini 1980). Reaction time is defined in the present study as the time required for the flame front to pass a certain point at the top of the fuelbed. The ignition temperature, 600 K, is used here as the lower threshold value to indicate the presence of flame. It is worth noting that the concept of reaction and residence time has been subjected to distinct conceptual interpretations and measurement methodologies (Anderson 1969; Rothermel and Deeming 1980, Catchpole et al. 1993; Alexander 1998; Catchpole et al. 1998; Nelson 2003b). Nelson's (2003b) reaction time model is most applicable to uniform fuelbeds and its use in wildland situations characterized by heterogeneous fuelbed structures and moisture contents is open to question. The existence of gradients throughout the fuelbed, namely in terms of fuelbed compactness and moisture content determine the amount of fuel available to be consumed in flaming combustion, and consequently the reaction time. To my knowledge, no quantitative information exists on the limitations of the application of Nelson's (2003b) reaction time model to natural surface fuelbeds. The mathematical formulation of this model and the iteration procedure used to solve it is described in the Appendix A.

16

Table 1.1. Conservation equations for the plume model (after Mercer and Weber 1994)

Conservation of mass	$$\frac{d(\rho_p \cdot b \cdot U_p)}{ds} = \rho_a \cdot v_e$$
Conservation of s-momentum	$$\frac{d(\rho_p \cdot b \cdot U_p^2)}{ds} = \rho_a \cdot v_e \cdot U_z \cdot \cos\theta - b(\rho_p - \rho_a) \cdot g \cdot \sin\theta$$
Conservation of r-momentum	$$\frac{d\theta}{ds} = \frac{\rho_a \cdot v_e \cdot U_z \cdot \sin\theta + b(\rho_p - \rho_a) \cdot g \cdot \cos\theta}{\rho_p \cdot b \cdot U_p^2}$$
Conservation of thermal energy	$$\frac{d(\rho_p \cdot b \cdot U_p \cdot T_p)}{ds} = \rho_a \cdot v_e \cdot T_a$$

The equations for conservation of mass, s-momentum (along the plume centerline), r-momentum (normal to the plume centerline), temperature (by rearranging the mass and thermal energy equations) along the plume centerline and its trajectory in two dimensions (Fig. 1.1) form a system of six coupled ordinary differential equations that are solved simultaneously with other three algebraic equations: (1) an equation of state, which for the gaseous plume can be represented by the ideal gas law:

[1.8] $$\rho_p = \rho_a \cdot \frac{T_a}{T_p}$$

(2) an entrainment assumption, represented by the entrainment velocity. This quantity is implemented considering two entrainment constants α_p and χ_p:

[1.9] $$v_e = \alpha_p \cdot (U_p - U_s \cdot \cos\theta) + \chi_p \cdot U_s \cdot \sin\theta$$

And (3) an equation describing the vertical ambient wind profile within the stand (Cionco 1965; Amiro 1990):

$$[1.10] \quad U_s = U_{SH} \cdot \exp\left(\alpha_U \cdot \left(\frac{z}{SH} - 1 \right) \right)$$

The steady state solution does not consider the interaction of the fire generated buoyancy in the ambient cross flow. The exponential decay function to describe the wind profile within the stand is valid for $0.6 < z/h < 1$ (Cionco 1965). Below this level the wind velocity is assumed constant. For the velocity profile above the canopy, between the measuring height and the canopy top height, a logarithmic wind profile was applied (Albini 1983):

$$[1.11] \quad U_s = U_{10} \cdot \left(\frac{\ln\left(\dfrac{z - 0.64 \cdot SH}{0.13 \cdot SH} \right)}{\ln\left(\dfrac{10 - 0.64 \cdot SH}{0.13 \cdot SH} \right)} \right)$$

The two coupled equations that describe the vertical wind speed profile provide a rough approximation of the wind flow within and above the forest canopy. This simple approach contrasts with more complex models solving the Navier-Stokes equations taking into account canopy architecture and generated turbulence (e.g., Shen and Leclerc 1997, Sanz 2003). The limitations of using eqs. [1.10] and [1.11] to describe the wind profile is acknowledged and its use justified by the focus of the present model system on the process of crown fire initiation, while simplifying the description of other characteristics associated with he fire environment, such as wind flow within and above the canopy.

The system describing the buoyant plume is solved using a Fehlberg order 4-5 Runge-Kutta method (Wolfram 1999). The required initial conditions are the initial plume half width (b_i), initial vertical velocity in the plume (U_{pi}) and initial plume

19

temperature (T_{pi}). The initial plume temperature is the flame tip temperature, assumed as 800 K (Draper point). This is the temperature at which red light first becomes visible (Siegel and Howel 1992), and can be used to loosely define the limit of the flame. This theoretical value is within experimentally determined flame tip temperatures, such as 773 K (Thomas 1963) and 823 K (Cox and Chitty 1980). The initial vertical velocity in the plume is given by (from Nelson 2003a):

$$[1.12] \quad U_{pi} = \left(\frac{2 \cdot g \cdot I_B}{\rho_a \cdot c_p \cdot T_a} \right)^{1/3}$$

The initial half width of the plume is assumed to equal half flame depth. The plume model only applies to the buoyant plume, and its base (Fig. 1.1) should correspond with the height where exothermic reactions due to oxidation of pyrolized fuels have ceased. This is assumed to coincide with what we perceive as the flame height. Nelson and Adkins (1986) parameterized Albini's (1981) flame height model that uses fireline intensity and average horizontal wind speed incident on the flame:

$$[1.13] \quad H_F = \frac{k_F \cdot I_B}{U_s}$$

With fireline intensity, I_B, following Byram's (1959) formulation -- i.e. the product of rate of spread by the available fuel for flaming combustion and the heat content of fuel particles after correction for fuel moisture content. The constant of proportionality k_F was estimated as 0.0028 $m^3 \ kJ^{-1}$ from the analysis of experimental fire data (Nelson and Adkins 1986). Flame height is measured from the top of the fuelbed (e.g. Thomas and Scott 1963, Albini 1981, Alexander 1982, Albini and Stocks 1986, Simard et al. 1989, Dupuy et al. 1998). This arises from the theory that it is above the fuelbed that mixing of

20

the fuel volatiles with air will approach an optimum, and the rate of thermal energy release is higher.

The model solution applies to the plume centerline, considering a top-hat profile for average temperature and vertical velocity. The distribution of these two quantities along the radial dimension of the plume was obtained by fitting a Gaussian distribution to the top-hat results (Davison 1986a; Zukoski 1995). Assuming self-similarity, the temperature profile along the r-direction:

$$[1.14] \quad \frac{(T - T_a) \cdot \kappa_p}{N_p \cdot (T_P - T_a)} = \exp\left(-\frac{r^2}{\kappa_p^2 \cdot b^2}\right)$$

The spreading ratio, κ_p, accounts for the turbulent diffusities of mass and heat versus momentum (Davidson 1986a). With a spreading ratio of 1.0 (Mercer and Weber 1994) the edge criterion, N_p, was estimated as 1.35 by finding the root of the difference of the integrals of the top-hat and Gaussian solutions. Distinct solutions can be found for these quantities. An approximation of the plume velocity along the r-direction was given by:

$$[1.15] \quad \frac{(U - U_s) \cdot \kappa_p}{N_p \cdot (U_P - U_s)} = \exp\left(-\frac{r^2}{\kappa_p^2 \cdot b^2}\right)$$

1.3.2. Heat transfer to canopy fuel particles

The present model system aims to predict the temperature of the fuel particles located in the lower layers of the canopy fuel strata. The canopy fuel strata can be discretized into layers of uniform bulk density (Scott and Reinhardt 2002, Alexander et al. 2004). In each layer the canopy fuels are composed of randomly distributed thermally

21

thin cylindrical particles -- i.e., there is no temperature gradient along their radial dimension, characterized by their σ, c_f, ρ_f, and *FMC*. The definition of the lower boundary of the canopy fuel layer is based on the fuel strata gap (*FSG*) concept (Cruz 1999, Cruz et al. 2004). *FSG* is defined as the distance from the top of the surface fuelbed to the lower limit of the aerial fuel stratum constituted by the ladder and live crown fuels that can sustain vertical fire propagation. This definition is distinct from previous descriptions of crown or canopy base height (Kilgore and Sando 1975, Van Wagner 1977, McAlpine and Hobbs 1994, Cruz et al. 2003b, 2003c) in which the vertical fuel gap in the fuel complex is equated to the live canopy base height (*CBH*). Surface fuelbed height was defined as done by Brown et al. (1982) and Burgan and Rothermel (1984). The lower limit of the aerial fuel stratum is where its ignition is presumed to result in sustained vertical fire spread (i.e. crown combustion).

From the assumption that flame development requires the attainment of a critical air-fuel ratio, above which the mixture fails to ignite, the formation of flame is dependent on the amount of volatiles being released, and consequently on the rate of heating and the amount and surface area of fuel per unit volume present. To my knowledge there are no published experimental results defining this fuel surface area or quantity required to sustain vertical fire spread. One possible approach in defining a critical crown structure property that allows vertical fire propagation is to consider a value of crown bulk density. Previous research defining critical canopy bulk densities have concentrated in the mechanisms of horizontal fire spread. Sando and Wick (1972) arbitrarily defined this quantity as the canopy lower vertical 0.3 m (1.0 ft.) section with a weight greater than 112.4 kg ha^{-1} (i.e., 100 lbs acre^{-1}), which equates to 0.033 kg m^{-3}. Williams (1977)

22

reports a personal communication from R.W. Sando in which this researcher considers the above threshold as too low, and suggests a value of 0.067 kg m^{-3}. Scott and Reinhardt (2001) used the same concept to describe *CBH* but defined sufficient crown fuel for vertical fire propagation as 0.011 kg m^{-3}. From the analysis of a large experimental fire dataset Cruz et al. (*in review*) found evidence of crown fire activity, in the form of passive crown fire spread, for stands with canopy bulk density values larger than 0.04 kg m^{-3}. In the present study critical crown bulk density for vertical fire propagation is defined as 0.05 kg m^{-3}. Nevertheless, the limitations of this definition need to be recognized. Aspects such as leaf morphology, the spatial distribution of leaves and the existence of fuels with high surface area to volume ratios, such as lichens, play an important role in the development of sustained flaming combustion in a tree crown.

The temperature change in the fuel particle is obtained from the conservation of energy principle, in which the amount of thermal energy transferred to the fuel particle is equated to its internal energy, and consequently its temperature:

[1.16] $\rho_f \cdot V \cdot c_f \dfrac{\partial T}{\partial t} = q''$

The model treats convective and radiative heat transfer separately. The net heat flux to the fuel particle is given by:

[1.17] $q'' = q_c + q_r - q_l$

where q_c is the convective heat flux, q_r is the radiant heat flux and q_l is the fuel particle radiant cooling to the surroundings.

The convective heat transfer implemented at the moment considers heat transfer to a single fuel particle. Heat transfer to a control volume was analyzed as well. Better results were achieved by considering a single fuel particle. Convective heat transfer to a fuel particle is given by:

[1.18] $\quad q_c = h_c \cdot A_f \cdot (T_g - T_f)$

If gas temperature (T_g), which can be plume or ambient temperature, is lower than the fuel particle surface temperature, q_c is a cooling term. When considering heat transfer to a control volume A_f is substituted by $\sigma \cdot \beta$, which expresses the surface area of fuel in a unit volume of fuel bed.

The average convective heat transfer coefficient, h_c, is a proportionality factor dependent on the convection regime, the thermodynamic properties of the fluid, namely, density, viscosity, specific heat, thermal conductivity and velocity, and the efficiency of heat absorption by the solid phase. The approach often followed in convective heat transfer problems is to determine the convection coefficient from its relationship with the dimensionless Nusselt number (N_u):

[1.19] $\quad h_c = \dfrac{N_u k}{d}$

Depending on the convective regime present, N_u is usually estimated from correlations relating it with several dimensionless variables, namely, the Reynolds (R_e) and Prandtl (P_r) numbers. No correlations are known to have been established between N_u and (R_e, P_r) for fuel particles common in a forest fire for forced convection under the R_e values characteristic of a buoyant plume. Fire behavior models incorporating convective heat transfer have relied on correlations established for cylinders or banks of tubs (e.g., Izbicki

24

and Keane 1989, Linn 1997, Dupuy and Larini 1999). Nevertheless, the applicability of these relationships is open to question, namely due to differences in the configuration of interest and surface roughness. Estimates of N_u were obtained from:

[1.20] $\quad N_u = 0.1417 \cdot R_e^{0.6053}$

This equation was proposed by Mendes-Lopes et al. (2002) for convective heat transfer to arrays of pine needles under forced flow for a R_e range between 50 and 400. The convective heat transfer coefficient is evaluated at film temperature. Fluid properties were estimated from interpolation functions derived from values given in Incropera and DeWitt (2002) for air.

The radiative heat transfer process considers the heat transfer between two flat surfaces, the surface fire idealized as a radiating plane, and the base of the canopy fuel layer. The radiating plane was defined with dimensions of flame depth by a fixed fireline width. The view factor represents the fraction of the uniform diffuse radiant energy leaving a surface that is incident upon another surface. The general expression for the view factor between arbitrarily oriented surfaces is (Modest 1993):

[1.21] $\quad F_{12} = \dfrac{1}{A_1} \displaystyle\int_{A_1} \int_{A_2} \dfrac{\cos\theta_1 \cos\theta_2}{\pi \cdot S^2} dA_2 dA_1$

where A_1 and A_2 are the surface areas of the emitter and of receptor respectively, S is the distance between the infitessimal surface elements dA_1 and dA_2, and θ_1 and θ_2 are the angles formed between S and the surface normals n_1 and n_2 (Fig. 1.3).

25
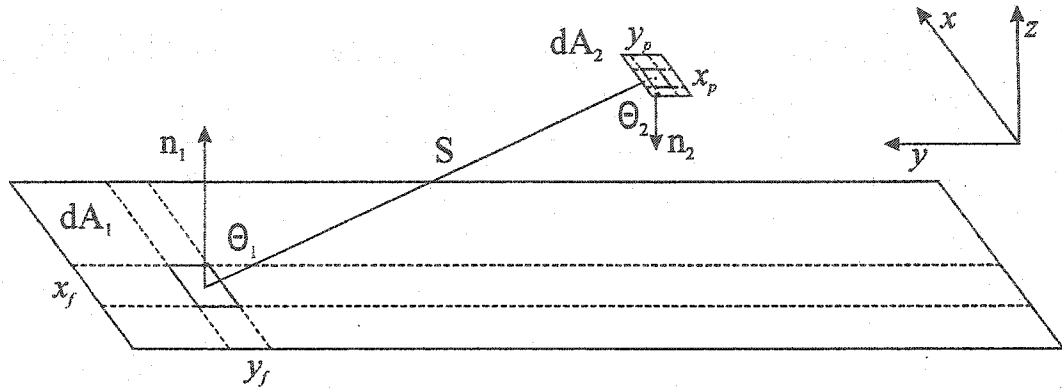
Figure 1.3. View factor geometry for the radiation interchange between the radiative heat source (A_1) and the lower surface of the canopy fuel layer (A_2).

For the geometric configuration of interest, described by two finite areas with unequal dimensions, the view factor integration needs to be solved with the integrands known in terms of the local coordinate system. The solution is simplified by the assumption that the surfaces are parallel, making $\theta_1 = \theta_2$. The view factor is given by the solution of the fourth-order integral:

$$[1.22] \quad F_{12} = \frac{1}{A_1} \int_{-\frac{L_p}{2}}^{\frac{L_p}{2}} \int_{D_x}^{D_x+D_F} \int_{-\frac{W_p}{2}}^{\frac{W_p}{2}} \int_{-\frac{W_f}{2}}^{\frac{W_f}{2}} \frac{\left(\frac{z}{S}\right)^2}{\pi \cdot S^2} dy_f \, dy_p \, dx_f \, dx_p$$

being

$$[1.23] \quad S = \sqrt{\left(y_p - y_f\right)^2 + z^2 + \left(x_f - x_p\right)^2}$$

To account for the attenuation of the intensity of radiant energy while the radiation travels a distance S through the absorbing porous medium consisting of the subcanopy-space, an exponential decay function incorporating the radiation absorption coefficient is

26

introduced. The absorption coefficient is calculated for the sub-canopy space by considering the contribution of the tree boles in reducing the mean path length. The radiation absorption coefficient is estimated from (Committee on Fire Research 1961):

$$[1.24] \quad \alpha = \frac{\sigma_S \cdot \beta_S}{4}$$

where σ_S is an surface area to volume ratio of the tree trunk with average dbh, and β_S is the fraction of the subcanopy space filled with tree matter. β_S is estimated from the ratio of stand averaged unit mass by unit volume in the sub-canopy space and the oven-dry mass density of wood, namely 510 kg m^{-3}. The average unit mass is estimated from stand density and average dbh. Stem space limbwood and tree trunk taper are not considered in this approximation of the radiation absorption coefficient. Upward radiative transport is also affected by the presence of water vapor and carbon particles in and above the flame. This radiation attenuation is not considered in the present modeling exercise. For the purpose of estimating radiant heat transport to the fuel particles, it is assumed that they behave as a blackbody -- i.e., they have an emissivity and absorbity of 1.0.

To take into account the non-constant radiosity of the surface fire and the attenuation of radiation within the sub-canopy space Eqs. [1.1] and [1.24] are incorporated into Eq. [1.22] to calculate the radiative transfer to the surface of the canopy fuel particles:

$$[1.25] \quad I_{12} = \int_{\frac{L_p}{2}}^{\frac{L_p}{2}} \int_{D_x}^{D_x + D_F} \int_{\frac{W_p}{2}}^{\frac{W_p}{2}} \int_{\frac{W_F}{2}}^{\frac{W_F}{2}} \frac{\varepsilon \cdot \sigma_{SB} \cdot \left(T_F^4 (x_F) - T_f^4 \right) \cdot e^{-\alpha S} \cdot \left(\frac{z}{S} \right)^2}{\pi \cdot S^2} dy_F dy_p dx_F dx_p$$

where T_F is the radiometric temperature of the flame in K, T_f is the fuel particle surface temperature and α is the radiation absorption coefficient. Eq. [1.25] was numerically integrated through the non-adaptive Halton-Hammersley-Wozniakowski algorithm (Wolfram 1999).

The calculation of the radiative energy absorbed by the fuel particle considers only half the area of the fuel particle:

[1.26] $q_r = I_{12} \cdot \dfrac{A_f}{2}$

The radiative cooling of the particle is described by the net radiative heat transfer between the fuel particle and the surroundings.

[1.27] $q_l = F_{23} \cdot A_f \cdot \sigma_{SB} \cdot \varepsilon \cdot \left(T_f^4 - T_a^4 \right)$

The view factor between the fuel particle and the surroundings, F_{23}, is calculated from the application of the reciprocity relation and the summation rule (Incropera and DeWitt 2002):

[1.28] $F_{23} = 1 - \dfrac{A_p}{A_F} F_{12}$

1.3.3. Fuel particle heating

Fuel heating assumes that the net energy gained or lost by the fuel particle equates to its internal energy, and consequently its temperature (eq. [1.16]). The two fuel variables of the canopy fuels determining the increase in temperature are its σ and FMC. The quantity σ determines the surface area available for heat transfer between the

28

gaseous and the solid phase per fuel particle unit volume. The moisture content increases the energy required to increase fuel temperature due to the high specific heat of water and the latent heat of vaporization. By integrating the three step heating model of Albini (1985) that takes into account the latent heat of the water present in the fuel and the specific heats of water and fuel as the fuel particle is taken from ambient temperature to ignition temperature, an average specific heat value can be calculated (Catchpole and Catchpole 2000):

[1.29] $c* \cdot (T_{ig} - T_a) = (c_f + m \cdot c_w) \cdot (373 - T_a) + (m \cdot L) + (c_f) \cdot (T_{ig} - 373)$

This model assumes that: (1) moisture is continually being evaporated from ambient to ignition temperature; and (2) all moisture must be driven out before ignition takes place. The application of the average specific heat model produces a better agreement with experimental data than the three- (Albini 1985) or two- (de Mestre et al. 1989) step heating models (Section 1.4.7).

By combining Eqs. [1.16], [1.18], [1.21], and [1.29] the governing equation for heat transfer to a single fuel particle was stated as:

[1.30] $\rho_f \cdot V \cdot c_f * \dfrac{dT_f}{dt} = h_c \cdot A_f \cdot (T_g - T_s) + I_{12} \cdot \dfrac{A_f}{2} - F_{23} \cdot A_f \cdot \sigma_{SB} \cdot (T_s^4 - T_a^4)$

This differential equation for the temperature of a fuel particle can be integrated as the fire front approaches and passes the fuel particle location. By integrating eq. [1.30] over dt we obtain:

[1.31] $\rho_f \cdot V \cdot c_f * T_f = \int_{x_0}^{\infty} h_c \cdot A_f \cdot (T_g - T_s)dt + \int_{x_0}^{\infty} I_{12} \cdot \dfrac{A_f}{2} dt - \int_{x_0}^{\infty} F_{23} \cdot A_f \cdot \sigma_{SB} \cdot (T_s^4 - T_a^4)dt$

29

The model is implemented in a Cartesian coordinate system with origin (x = 0, y = 0) coinciding with the canopy fuel particle location. The model of eq. [1.31] is iterated until ignition temperature (600 K) is attained or the back of the flaming front passes the fuel particle location. Ignition temperature is expected to vary within a range around 600 K as a function of species and moisture content. Sussot (1982) characterized the rate of volatilization of fuels as function of temperature, and demonstrated it to be species and fuel type dependent. Catchpole et al. (2002) pointed out the effect of vaporizing water vapor in diluting volatile pyrolysis products, and hypothetically raising the temperature required to obtain a ignitable mixture. Given the variability in fuel chemical composition and surface fire heat flux rates, which determine the rate of volatile production of the unburned fuels, the use of a fixed ignition temperature of 600 K (Rothermel 1972, Albini 1996) is a practical assumption required to implement the model.

1.3.4. Model implementation

The CFIM can be considered as a model system coupling several empirical and simple physical models that allow tracking the energy being absorbed by lower canopy fuels, and consequently their temperature. The model is implemented as time and space explicit, with sub-models being solved in 1-D, 2-D vertical plane (buoyant plume structure) and 3-D (radiative heat transfer). The model system can be run with a relative small number of input variables (Table 1.2). Most of the variables in Table 1.2 are commonly used in predicting fire behavior to support decision making in fire management related issues (NWCG 1993, Canadian Interagency Forest Fire Centre 1996, Andrews and Bevins 1999). The variables describing the surface fuels follow the fuel

model concept (Rothermel 1972, Burgan and Rothermel 1984). There is a vast number of studies that provide quantitative description of surface fuelbeds for North American fuel complexes (e.g., Anderson 1982, Ottmar et al. 1998, 2000, 2002). Canopy fuel stratum structure, i.e., crown and canopy bulk density and fuel strata gap, have not been subject to an extensive description as for the surface fuelbeds. These quantities can be estimated directly from common stand inventory data (e.g., McAlpine and Hobbs 1994, Cruz et al 2003b) or inferred by other means (Keane et al. 1998, Riano et al. 2003). Stand density and average *dbh* are required to estimate the radiation opacity coefficient. This coefficient can also be entered directly, avoiding the need to have a detailed stand description to run CFIM.

Table 1.3 provides a list of models (and sources) integrating CFIM. The functional relationships between the various models are presented in Fig. 1.2. The modular structure of CFIM allows changing any of these intermediate models if others are considered to have higher explanatory power or are more adequate in particular situations. This is essentially applicable to the models with a higher empirical component, such as the model describing the vertical wind profile (eqs. [1.10] and [1.11]), the flame height model (eq. [1.13]) and the surface fire maximum flame temperature (eq. [1.7]). A discussion on the behavior of each of these models and its effect on CFIM behavior are given in Section 1.4.

Table 1.2. List of main fuel and weather input variables required to run CFIM

Variable	Symbol	Units	Period of change	Use
Fuel complex				
Dead fuel Moisture content	MC	fraction	Very short	ROS, τ_r, T_{Fmax}
Foliar moisture content	FMC	fraction	Medium	$c*$
Surface area to volume ratio	σ	m^{-1}	Long	ROS, τ_r
Available surface fuel load	w_a	$kg\ m^{-2}$	Long	ROS, τ_r, T_{Fmax}
Surface fuel layer depth	δ	m	Long	τ_r
Fuel strata gap	FSG	$kg\ m^{-3}$	Long	F_{12}, I_{12}, T_p, T_f
Stand height	SH	m	Long	U_z
Stand density	SD	$trees\ ha^{-1}$	Long	A
Average dbh	dbh	m	Long	A
Heat content	H_c	$kJ\ kg^{-1}$	Long	I_B
Fire weather				
Wind velocity	U	$m\ sec^{-1}$	Very short	ROS, T_p, v_p, b_p
Air temperature	T_a	K	Very short	T_p, T_f

Table 1.3. List of intermediate models, their input variables and their use in CFIM

Model	Output symbol	Input variables	Output	Use to estimate
Vertical wind profile (Cionco 1965, Albini 1983)	U_z	U_{10}	Wind velocity vertical profile.	ROS, H_F, τ_r, T_{Fmax}, T_p, U_p, b_p
Rate of forward fire spread (Rothermel 1972)	ROS	β, σ, δ, U, MC, w_a,	Rate of movement of the fire front.	I_B, F_{12}, x_F.
Fireline intensity (Byram 1959)	I_B	ROS, w_a, Hc	Integrated rate of energy released per unit time per unit length of the fire front	F_h, U_{pi},
Flame height (Albini 1981)	H_F	I_B, U	Height of the surface fire flame.	Tp, vp, bp
Reaction time (Nelson 2003b)	τ_r	β, σ, δ, U, MC, w_a,	Duration of flaming combustion at a fixed point in the fuelbed.	I_{12}, bi
Maximum gas temperature (surface fire) (eq. 1.5)	T_{Fmax}	w_a, σ, U,	Maximum temperature attained in the surface fire flame.	I_{12}
Initial buoyant velocity (Nelson 2003a)	U_{pi}	I_B, T_a, ρ_a,	Initial buoyant velocity above the flame (z = H_F).	Vp
Plume dynamics model (Mercer and Weber 1994)	T_p, U_p, b_p	vi, U_z,	Fluid temperature, fluid velocity and half width of the buoyant plume.	h_c
Flame temperature – time distribution (eq. 1.7)	$T_F(t)$	T_{Fmax}, τ_r,	Temperature – time history of the surface fire flame.	I_{12}
Radiative heat source (eq. 1.25)	I_{12}	$T_F(t)$, ε	Radiative energy leaving heat source.	q_r
Radiative heat flux (eq. 1.26)	q_r	I_{12}, q_r	Radiative heat flux to fuel particle in canopy.	T_f
Convective heat flux (eq 1.18)	q_c	σ, U, R_e, h_c, k	Convective heat flux to fuel particle in canopy.	T_f

33

CFIM is implemented through the following calculation procedure:

1. Specification of initial conditions:

   a. Fuel complex properties (NFFL (Northern Forest Fire Laboratory) fuel model (Anderson 1982) or w_a and δ, FSG; SH;

   b. Ambient temperature;

   c. Fuel moisture content (MC and FMC);

   d. 10-m open wind velocity;

   e. Estimation of wind profile (eq. [1.10] and [1.11]);

   f. x- position;

   g. Estimation of the surface fire rate of spread, ROS;

   h. Estimation of reaction time, τ_r (Nelson 2003b);

   i. Estimation of flame depth, $D_F = ROS \cdot \tau_r$;

   j. Estimation of fireline intensity, I_B (Byram 1959);

2. Estimation of plume properties Mercer and Weber (1994):

   a. Within stand U from wind profile;

   b. Initial plume half-width as ½ flame depth;

   c. Initial plume velocity from (Nelson 2003a).

3. Estimation of time- temperature profile above surface fire (eq. [1.7]).

4. Flame height, H_F is estimated from Nelson and Adkins (1986) model (eq. [1.13]). This height sets the base of the plume model.

34

5. A solution for the plume model at a $z = FSG - H_F$ is calculated and air temperature and velocity results are stored for the convective heat transfer calculations (eqs. [1.A11] – [1.A27]).

6. The fire front approaches the fuel particle and heat transfer calculations are executed. This is accomplished through a time loop:

    a. Initial T_f (or previous iteration T_f) is defined as current T_f;

    b. From D_F, the tT-T profile in the flame front and the position of the fire front relative to the fuel particle the radiant heat flux is estimated (eq. 25).

    c. From the solution of the plume model, T_p and U_p at the location of the fuel particle are used to estimate the convective heat transfer coefficient. All air thermodynamic properties are estimated at T_{film}.

    d. Heat transferred to the fuel particle is estimated and new T_f is defined.

    e. If $T_f < 600$ K, T_f is used in the next time loop. If $T_f > 600$ K, it is assumed that crown ignition has taken place, and the process is stopped. If $T_f < 600$ K and the back of the fire front passed the fuel particle x-location, the process is interrupted also.

1.4. Analysis of model components

The phenomenon of the onset of crowning is a complex one, with a multitude of independent variables and intermediate processes that vary over a broad range and in turn affect the outcome in distinct ways. Any modeling approach aimed at increasing our understanding of these processes requires a fundamental approach that incorporates both combustion and heat transfer components. In the present study, simplified models describing the various processes determining the onset of crowning were integrated in a manner that allow determination of the conditions that lead to the ignition of canopy fuels. This approach requires a substantial amount of sub-modeling. The overall model structure was described in Section 1.3. The present section describes the behavior of the various sub-models used and justifies the selection of particular sub-models when appropriate.

1.4.1. Wind profile

The structure of wind flow above and within forest canopies is critical to modeling fire behavior and its effects. Fire modeling studies have normally required wind as a scalar quantity measured or estimated at a given height (Rothermel 1972, Albini 1996, Forestry Canada Fire Danger Group 1992, Finney 1998, Nelson 2002). The present modeling system requires estimates of wind speed at various heights within a forest canopy. This vertical structure of wind within the stand is used to estimate several surface fire behavior quantities, such as rate of spread (e.g., Rothermel 1972, Albini 1976), reaction time (Nelson 2003b), maximum flame temperature (eq. [1.7]), and to define air entrainment at various heights in the buoyant plume.

The model used here to describe wind flow above and within a forest stand canopy is composed of two coupled equations (i.e., eqs. [1.10] and [1.11]). These provide a simple description of the vertical wind profile based solely on three input variables, namely the 10-m open wind speed, stand height and a wind attenuation coefficient. This approach contrasts with more complex models taking into account canopy architecture and aerodynamics in absorbing momentum (e.g. Raupach and Thom 1981, Shen and Leclerc 1997, Sanz 2003, Cescatti and Marcolla 2004). The choice in favoring eqs. [1.10] and [1.11] in lieu of more complete models is justified by the focus of the present study on the process of crown fire initiation, while describing other characteristics of the fire environment through simple models. It is recognized that the implementation of a model that better describes the wind flow and its interaction with the fire may provide a better description and understanding of some of the processes determining the initiation of crown fires. The eventual implementation of CFIM into a fire behavior prediction system or its use as a research tool can be accompanied by a better model for wind flow if necessary. Still avoiding the complexity of solving the conservation equations above and within the canopy, other models could be applied (e.g., Amiro and Davis 1988, Kinnersley et al. 1994, Lalic et al. 2003).

It should be noticed that the current model used to predict the vertical wind profile has certain limitations when applied to predict the wind above a surface fire. Besides the fact that the model does not account for the fire-wind interactions, the model applies to mechanically produced turbulence and it is appropriate solely to adiabatic conditions (Campbell and Norman 1998). The strong heating caused by the surface fire increases turbulence and mixing, technically violating those assumptions. However, this limitation

is inherent to most wind modeling studies, that describe ideal situations, such as, neutral to slightly unstable conditions, horizontally extensive and uniform canopy, level terrain (Raupach and Thom 1981, Lee 2000), and probably do not detract from the practical purposes of CFIM.

The wind attenuation coefficient, a quantity dependent on canopy structure and stand occupancy, is the main unknown of eq. [1.10]. Several studies have determined the attenuation coefficient for different forest types using leaf area index (LAI) (Cionco 1972, Amiro 1990). Amiro (1990) determined attenuation coefficient values between 2.6 and 4.8 for stands of jack pine (*Pinus banksiana*) (LAI = 2) and black spruce (*Picea mariana*) (LAI = 7) respectively. Analysis of the wind data during an experimental crown fire in a red pine (*Pinus resinosa*) plantation (Van Wagner 1968, Alexander 1998) yielded an attenuation coefficient of 1.2. This value might be affected by edge effects. The effect of the wind attenuation coefficient in determining wind speed at any given height below the canopy is illustrated in Fig. 1.4.

A wind attenuation coefficient of 1.0 would be representative of open stands, although for such stands wind flow might not be approximated by eq. [1.10]. In sparse stands the effect of individual plants is discernible and the horizontal flow is no longer uniform (Lee 2000). The higher values of the wind attenuation coefficient are appropriate for excessively dense stands. Albini and Baughman (1979) provide tabular information describing the reduction of wind speed within forest stands that can be used to derive generalistic estimates of the attenuation coefficient for forest stands.

38
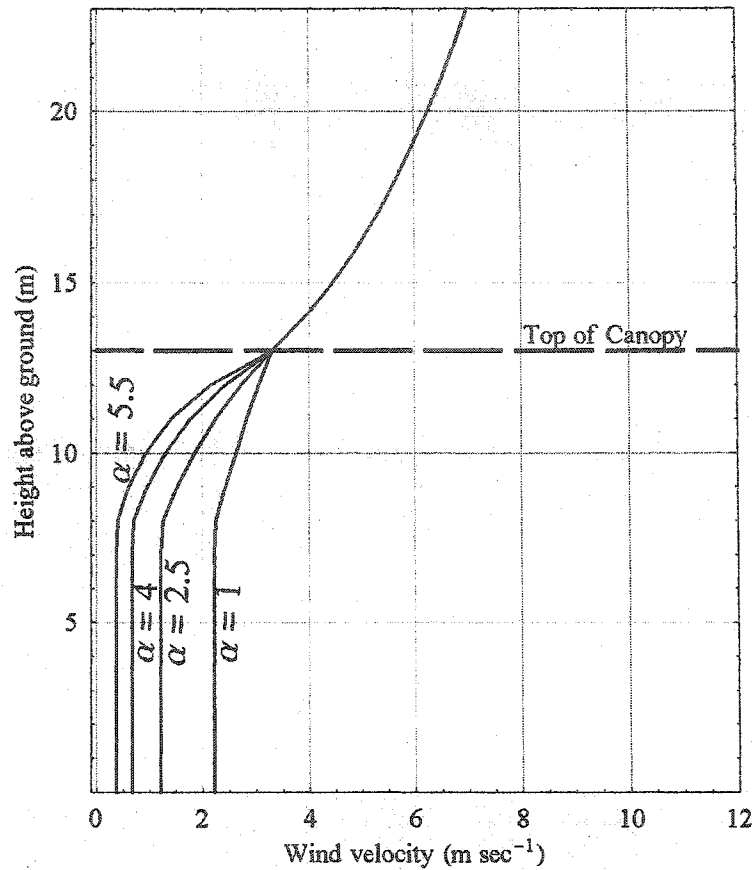
Figure 1.4. Simulated vertical wind profiles for four distinct wind attenuation coefficients.

1.4.2. Temperature-time profile model

In the present study the radiant heat flux was modeled based on the Stefan-Boltzmann equation, assuming a fixed emissivity and leaving the flame radiometric temperature as the main unknown. The spatial and temporal variations in this temperature properties was modeled through a temperature-time (T-T) profile model (eq. [1.7]) that predicts flame temperature at the top of the surface fuelbed. The temperature-time profile is of critical importance to the understanding of several fire behavior and

39

effects processes (Gill and Knight 1991, Moore et al. 1995). Fire researchers have characterized these profiles above surface fires with the objective of calculating damage to plant tissues (Byram and Nelson 1952, Byram 1958, Johnson and Miyanishi 1995) and soil characteristics (e.g., Sackett and Haase 1992, Bailey and Anderson 1980, Campbell et al 1994).

A small number of models aimed at characterizing the T-T profile above a fire were found in the literature. Weber et al. (1995) formulated a two-step model to characterize the temperature above a surface fire based on two free parameters that needed to be found empirically from field measurements. Fire behavior models incorporating simple combustion principles approximated by Arrhenius laws and solving the conservation equations for the solid and gaseous phases are able to produce estimates of the temperature-time profile (e.g. Dupuy and Larini 1999, Porterie et al. 2000, Morvan and Dupuy 2001).

The T-T model developed in this study is dependent on two surface fire characteristics, namely maximum flame temperature and reaction time. These two flame front properties are not easily defined and have been interpreted differently by fire researchers.

The maximum theoretical temperature that a combustion system can attain is the so-called adiabatic flame temperature. This temperature could only be attained if no heat losses occurred and can be calculated from thermo-chemical principles. Turbulent diffuse flames associated with wildfires are characterized by finite-rate chemical reactions accompanied by (mostly) radiative and convective heat losses (Saito 2001). Thus the temperatures attained in the fire will be a function of the rate of the chemical

reactions and flame size. For wildland fires, thermocouples and infrared thermography are two commonly used methods that allow one to obtain spatially and temporally resolved temperature estimates. Both these methods are associated with significant uncertainty and measurement errors. The temperatures recorded by a thermocouple are not true flame temperatures, but the result of a heat balance attained in the thermocouple bead (Martin et al. 1969). Thermocouple physical characteristics will affect the temperature reading due to thermocouple heat losses and response time, function of material thermal conductivity, density and size (Walker and Stocks 1968, Shaddix 1998, Saito 2001, Sullivan et al. 2003). These errors are minimized with the use of fine wire thermocouples and corrections for heat loses (Shaddix 1998, Saito 2001, Shannon and Butler 2003). The use of infrared thermography provides a non-intrusive technique that yields large spatial and temporal resolution. Nevertheless, flame emissivity variability and radiation absorption due to smoke, water vapor and CO_2 will influence the temperature reading (Saito 2001). Infrared thermography also allows indirect estimation of fluid velocities within the flame, albeit with considerable uncertainty (Clark et al. 1999).

Maximum temperatures reported in fire research studies should be interpreted with caution, as measured values are a function of the measurement method and technique used. In laboratory fires characterized by small, low opacity flames, Walker and Stocks (1968) found significant differences between the temperatures of bare wire thermocouples with different wire diameters. For these flames, the average temperature recorded varied between 1038 and 1296 K for thermocouples with a diameter varying from 0.8 to 0.13 mm, respectively.

41

In the present study we relied on the use of small diameter type K thermocouples ($\approx$ 0.127 mm wire diameter and 0.25 mm bead diameter – Omega Engineering, Stamford, Connecticut, USA). This wire diameter was a compromise between the response time of the thermocouple and its resistance to the chemical and mechanical combustion environment. This wire/bead diameter combination minimizes radiative and convective losses from the thermocouple. The time constant of these thermocouples is around 0.1 s. The temperature measured by the thermocouples was assumed as flame temperature. The model developed for the prediction of the flame maximum temperature was based on an experimental fire dataset covering a restricted range of fuelbed characteristics. Details relative to the experimental setup and the fire behavior dataset assembled are given in Chapter 2.

Another important variable for the definition of the T-T profile is the surface fire reaction time. A fire front reaction time can be defined as the duration of flaming combustion at a fixed point in the fuelbed (Nelson 2003b). The difficulty in defining the rear boundary of the flaming combustion zone makes it difficult to measure this quantity in both laboratory and outdoor fires. As pointed out by Alexander (1998) and Nelson (2003b) no standard exists to deduce reaction time from observed fire behavior. Consequently various definitions of reaction time (or flame front residence time as called by several authors) appeared in the literature (Anderson 1964, Rothermel and Deeming 1980, Nelson 2003b). Reaction time can be estimated from the ratio of flame depth and rate of spread (Alexander 1982), from temperature-time curves (Rothermel and Deeming 1980, Bidwell and Engle 1991) or from visual estimates of the time the flaming front takes to pass a particular location in the fuelbed.

The T-T model (Eq. [1.7]) is a simplification of the T-T curve as it only considers the temperatures above 600 K. The heating phase associated with surface fuel preheating and the post frontal residual combustion characterized by scattered flamelets was ignored in the model. Both these phases radiate at relatively low temperatures, which was considered negligible for the radiative heat transfer calculations to the canopy fuels. The T-T model is dependent on the definition of three points, the maximum temperature, the location of the ignition interface and the location where active flaming combustion ceases. Therefore, the predicted T-T profile is strongly dependent on the results of the predicted T_{Fmax} and τ_r. The implemented T_{Fmax} model can be considered one of the weakest components of the model system. The T_{Fmax} model was developed from a fire behavior dataset for which the majority of the data were originated from laboratory fires with limited variability in some fire environment variables (see Chapter 2). Consequently, the implemented T_{Fmax} model should be viewed as an interim model, which can be replaced by a better model when one becomes available.

Fig. 1.5 presents the predicted and observed T-T profiles for four experimental fire situations. Fires FiSL30 and FiSL31 were laboratory fires using coarse excelsior conducted in the USDA Forest Service Fire Sciences Laboratory (FiSL), Missoula, Montana. LEIF11 was also a laboratory experimental fire conducted at the University of Coimbra Fire Research Laboratory (LEIF) wind tunnel. Gestosa517 was a high intensity outdoor experimental fire conducted in a shrubland fuel complex in Portugal (Viegas et al. 2002, Cruz et al. 2003d). Details regarding these fires are given in Chapter 2. The plots show the general behavior of the T-T model and its dependence in accurate estimates of T_{Fmax} and τ_r. From the standpoint of estimating the radiative heat transport

to the canopy fuels, the most significant errors are the ones introduced by differences between observed and predicted area integral above ≈ 1000 K.
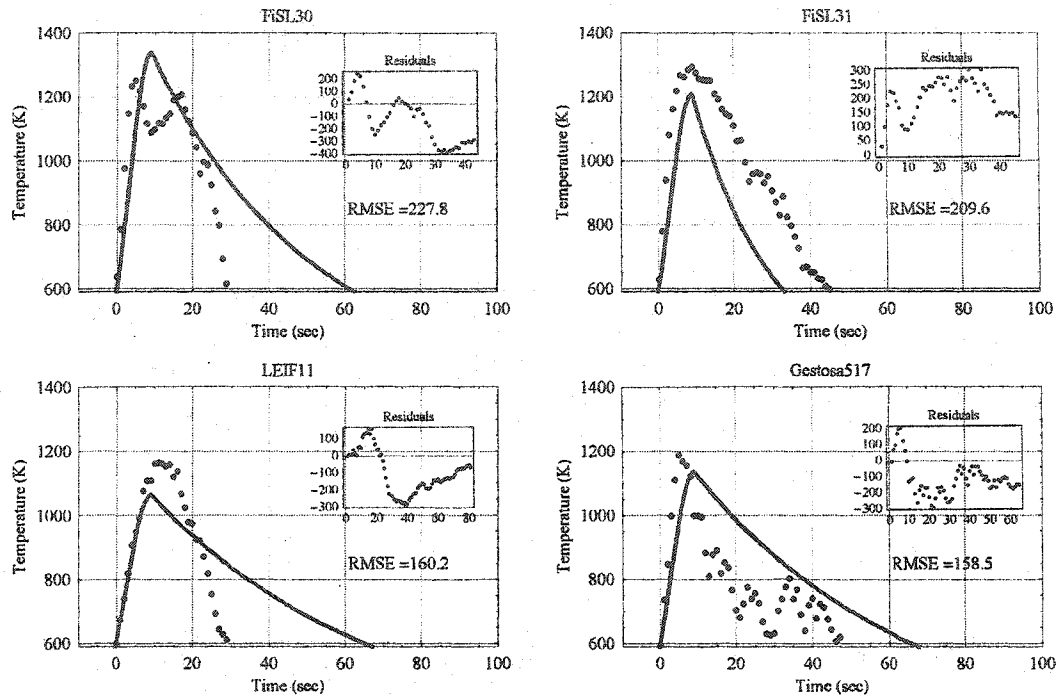
Figure 1.5. Predicted (line) and observed (dots) T-T profiles for various experimental fires. FiSL and LEIF were laboratory fires; Gestosa 517 was a high intensity experimental fire carried out in a shrubland fuel complex. RMSE is the root mean square error of the T-T profile.

1.4.3. Geometrical view factor

The view factor describes the fraction of radiative energy leaving the emitter that reaches a receiving surface or volume. Given the transient nature of wildfire flames and the complex orientation of fuels, an accurate estimation of view factors is extremely difficult. Typically, in forest fire research, both the emitter and the receiver are normally idealized shapes with well-defined orientation and boundaries. Several distinct methods can be used to estimate the view factors between surfaces. For the present study both

44

analytical solutions (Howell 1982) and numerical integration methods (Modest 1993, Butler and Cohen 1998) were evaluated to test their adequacy to the idealized configuration (Fig. 1.3). Both the numerical integration (eq. [1.22]) and analytical method (Howell 1982, equation C-13) produced identical results. Figs. 1.6a – 1.6c illustrates the view factor two-dimensional distribution around the flame front. For this simulation the view factor was estimated assuming a source of dimensions D_F (depth) x 20 m (width) and a receiving element as a 0.1 m side square. From this figure it can be seen that the fraction of the total radiant heat released by the surface fire that reaches the surface of interest is rather small, with the view factor not changing substantially with the increase in flame depth. While the fire front is somewhat distant from the fuel particles for which the heat transfer calculations are done, fuels located at higher z coordinates "see more flame", and consequently are subjected to higher incident radiative heat fluxes. As the flame front approaches the fuel particle, the lower the location of the fuels, the more flame they will see, and the calculated view factor will be higher.
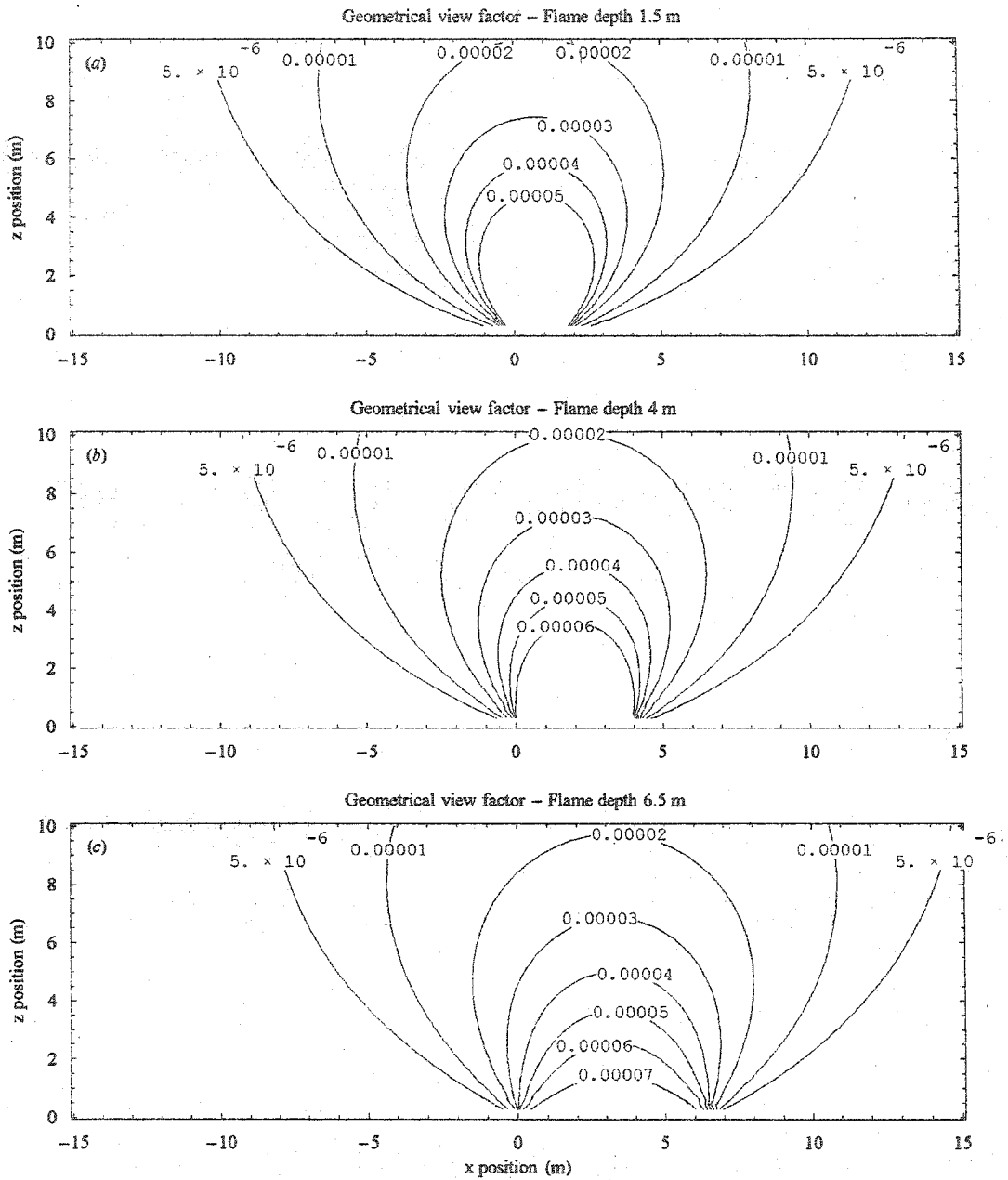
Figure 1.6. Distribution of geometrical view factors between a surface emitter with variable flame depth and a receiver as idealized in the present study. (a) flame depth of 1.5 m; (b) flame depth of 4 m; (c) flame depth of 6.5 m. Flame width is 20 m. and receiver has dimensions of 0.1 x 0.1 m.

46

1.4.4. Incident radiative heat flux

The calculation of the incident radiative heat flux to the canopy fuel particles involves integrating models describing: (1) the surface fire T-T profile; (2) the geometrical view factor between the idealized surface fire flat dimensions and the canopy fuels; and (3) the attenuation of the radiant energy intensity while radiation is transported through the sub-canopy space.

Both the flame T-T profile and view factor model characteristics have been already described. Eq. [1.24] has been commonly used by fire researchers to estimate radiation absorption in surface fuelbeds (e.g., de Mestre et al. 1989, Butler 1994, Wotton and Renaud 1996) and within the canopy layer and the sub-canopy space (e.g., Albini and Stocks 1986, Albini 1996, Call 1997). Measurements of radiant heat fluxes in laboratory fires by Butler (1994) corroborated the predictions of eq. [1.24] for surface fuelbeds. Measurements of radiative heat flux profiles in a jack pine-black spruce fuel complex during the International Crown Fire Modeling Experiment (ICFME) (Alexander et al. 2004, Stocks et al. 2004) suggest a sub-canopy space mean path length between 20 and 60 m (Butler 2003). Predictions of eq. [1.24] for the average ICFME stand conditions, *dbh* 0.084 m and stand density of 4115 trees ha^{-1} (Stocks et al. 2004), yields a mean path length of $\approx$37 m. Fig. 1.7 illustrates how both the radiation mean path length and the radiation absorption coefficient varies with stand occupancy, represented by a combination of average *dbh* and stand density.

47
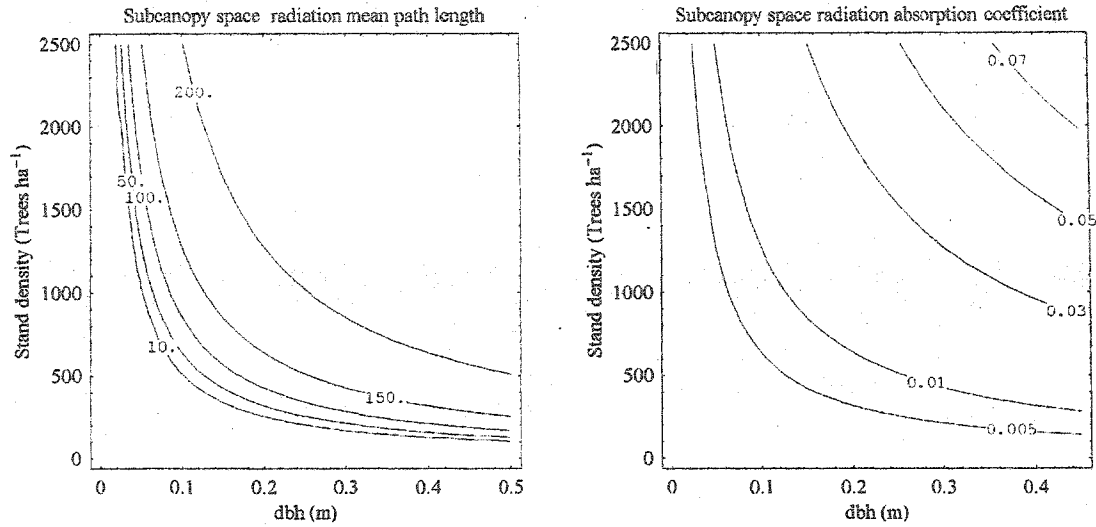
Figure 1.7. Estimated radiation mean path lengths (m) and absorption coefficients (m^{-1}) for subcanopy space function of average *dbh* (m) and stand density (trees ha^{-1}).

From eq. [1.21] it is possible to analyze the dependence of incident radiative heat flux to canopy fuels on the dimensions of the idealized surface fire emitter and the location of the fuels relative to the emitter. Fig. 1.8a – 1.8c plots isolines of incident radiative heat fluxes to the lower boundary of the canopy fuel layer function of spatial location {x,z} and flame depth. The results of Fig. 1.8a – 1.8c show the strong effect that the increase in flame depth has in the radiation being received by canopy fuels. For fuel particles located above the most intense zone of the surface fire, broadly defined as the first third of the surface fire reaction zone, the flame depth increase from 1.5 to 6.5 m results in a proportional increase in the incident radiative heat flux. In the model system, an increase in flame depth of this order would be the result of an increase in *ROS* and τ_r. The input variables with the most effect on these two intermediate fire behavior properties are respectively wind speed and fuel available for flaming combustion. In Section 1.5.1 the effect of these variables on the various model components were analyzed in detail. The distribution of the incident radiative heat flux isolines in the two-

48

dimensional space of Fig. 1.8 are mostly a function of the geometric relation between the emitter and the receiver, as shown in Fig. 1.6a – 1.6c. The analysis of Fig. 1.8b and 1.8c also demonstrate the effect of considering the variation in the radiative intensity along flame depth. Radiation intensity along flame depth is a function of the T-T profile. The higher radiation intensity in the forward section of the flame front, as determined by the T-T profile, originates a non-symmetrical distribution of the incident radiative heat flux (Fig. 1.8c). This is in contrast with what would be predicted if an average radiometric temperature would be assumed (e.g. Albini and Stocks 1986, Sullivan et al. 2003). A solution based on an average radiometric temperature would result in a symmetrical distribution of incident radiative heat flux, as for the view factor case. The plots of Fig. 1.8a – 1.8c assume a fixed T-T profile. Given the dependence of the radiative intensity on fourth power of temperature, changes in the T-T profile, namely an increase in the predicted maximum temperature will generate larger asymmetries in the incident radiative heat flux distribution.

The use of the variable radiometric temperature instead of an average value also affects the model system result by generating simultaneous convective and radiative heating under high heat flux. With the exception of a no-wind, no-slope scenario, both convective and radiative heat flux distributions are offset in time and space. In the presence of wind, plume tilting subjects the canopy fuel particles first to high convective heat fluxes, followed by a peak in the incident radiative heat flux (Byram 1948 in Alexander 1998). Although the peak intensities of both convective and radiative heat transfer to the fuel particle occur at different times, the use of the variable radiometric temperature solution results in an approximation of the two peaks, and a slight overlap of

the two distributions. This is because for the same wind speed and consequently plume tilt angle, peak incident radiative heat flux will occur when the fuel particle of interest is still immersed in the buoyant plume, resulting in an increase in the predicted maximum fuel temperature. With the average radiometric temperature solution, the distribution of incident radiant heat flux shifts to the center of the flame zone, reducing overlap with the incident convective heat flux distribution. The result is that while the canopy fuel particle is being subject to the maximum radiative heat flux, the convective heating component is lower, thereby inducing convective cooling.

50
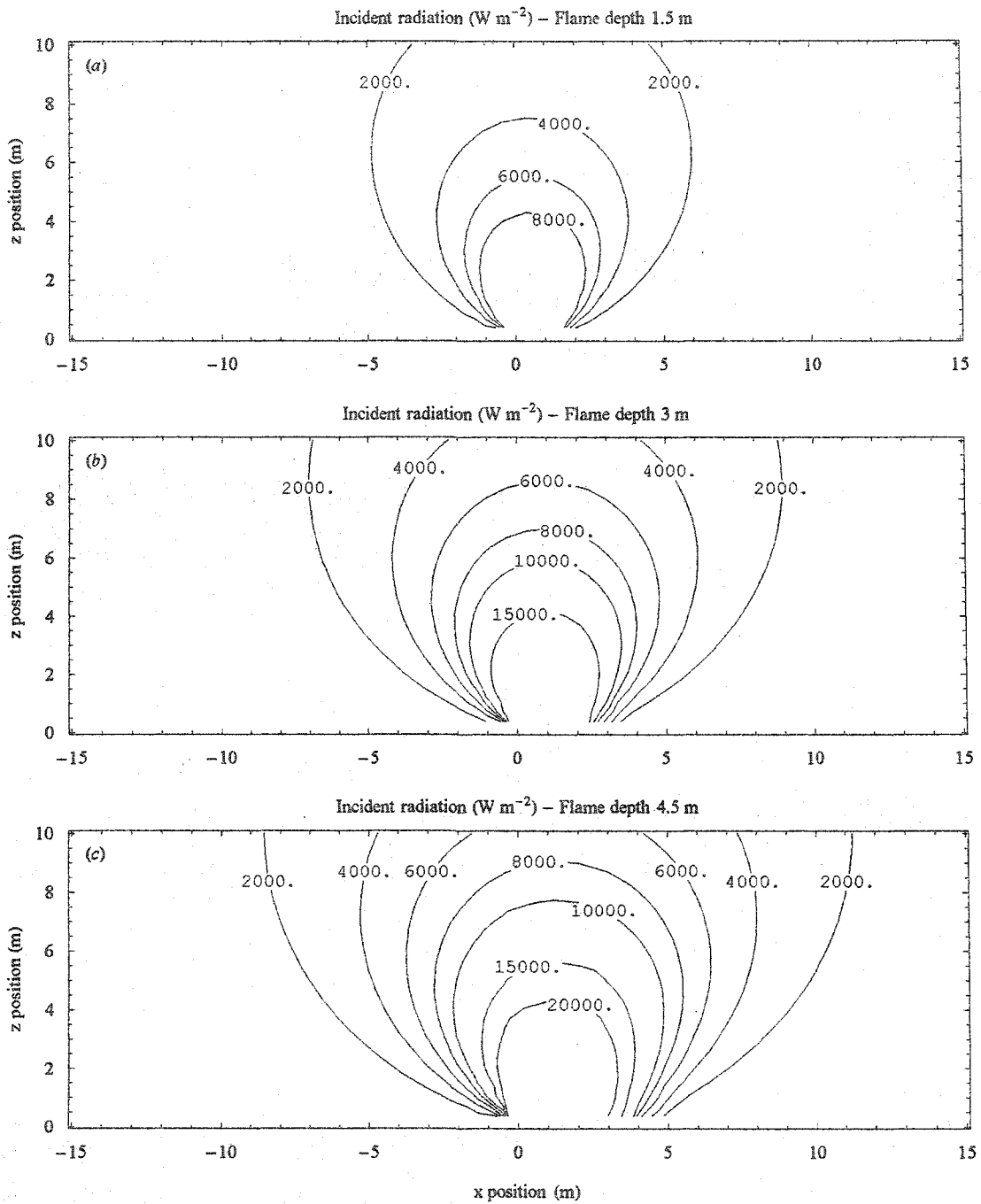
Figure 1.8. Distribution of incident radiative heat fluxes with various flame depths. (a) flame depth of 1.5 m; (b) flame depth of 4 m; (c) flame depth of 6.5 m. Flame front width is 20 m.

51

1.4.5. Flame height model

Flames in wildland fires can be characterized as turbulent diffusion flames (Sullivan et al. 2003). The release of energy within the flame, and consequently its temperature, is controlled by the rate at which air mixes with the volatilized fuels (Nelson 1980). What we perceive as the visible flame is the radiation emitted by the incandescent soot and ash particles in the flame. Hypothetically, a model incorporating the physical and chemical processes occurring in the flame by solving the conservation equations could produce estimates of flame structure by describing composition, temperature and velocity fields within the flame. The present model system structure describes the flame through relatively simple models of the various flame characteristics of interest, namely, flame depth and flame height. Flame height is an important intermediate input in CFIM as it will be used to locate the base of the buoyant plume in the reference coordinate system. The vague definition of what is perceived as a wildland fire free flame and its transient nature (Ryan 1981, Johnson 1982, Beer 1991, Mendes-Lopes et al. 2003) make it difficult to compare data from distinct studies and evaluate models describing flame geometry. Unlike flame length, a measure of flame size related to the integrated energy release by the fire which has been extensively analyzed in fire research (e.g., Byram 1959, Nelson and Adkins 1986, Fernandes et al. 2000), flame height has received little attention. A few studies attempted to quantify flame height from easily measured fire environment and/or fire behavior quantities. Albini (1981) developed a model to describe the structure of an idealized wind-blown flame above a free-burning line fire. Although the complexity of this model precludes its application in a predictive sense, Albini proposed an approximation for flame height as being proportional to the ratio of

52

fireline intensity and ambient wind speed (eq. [1.13]). The dimensional constant in this model should have some dependence on fuelbed characteristics (Albini and Stocks 1986) and it was estimated by these authors for experimental crown fires in immature jack pine stands as 0.005 m^3 kJ^{-1} (Albini 1996). From the analysis of various laboratory fires Nelson and Adkins (1986) suggested a value for k_F of 0.0028 m^3 kJ^{-1}. Catchpole and Catchpole (2000) evaluated this relationship against a large number of experimental laboratory fires conducted in the FiSL wind tunnel and proposed two interim empirically derived models to describe flame height.

The three formulations for the prediction of flame height from the ratio of fireline intensity and wind speed were compared (Fig. 1.9). The approximation of Albini (1981) phenomenological flame model is not applicable to low wind velocities. The use of a proportionality constant of 0.005 in low wind flow conditions, characteristic of forest stands, results in the over-prediction of flame height. For within-stand wind velocities lower than 2 m s^{-1}, the flame height prediction is larger than flame length as predicted by Byram's (1959) model (Fig. 1.9). Albini and Stocks (1986) parameterization predicted the highest flame heights. Nelson and Adkins (1986) model predicted lower flame heights for low intensity fires than Catchpole and Catchpole (2000) model, while for the upper range of fireline intensities tested it tended to produce higher flame heights.

To my knowledge no comprehensive evaluation of these models with independent data collected in outdoor fires has been carried out. From the results presented in Fig 1.9, Nelson and Adkins (1986) parameterization appeared to give the most realistic results over a broad range of conditions and this was implemented in the system to locate the

53

height of the buoyant plume base. As with other sub-models in the model system, a different model for flame height can be used in the future if proven adequate.
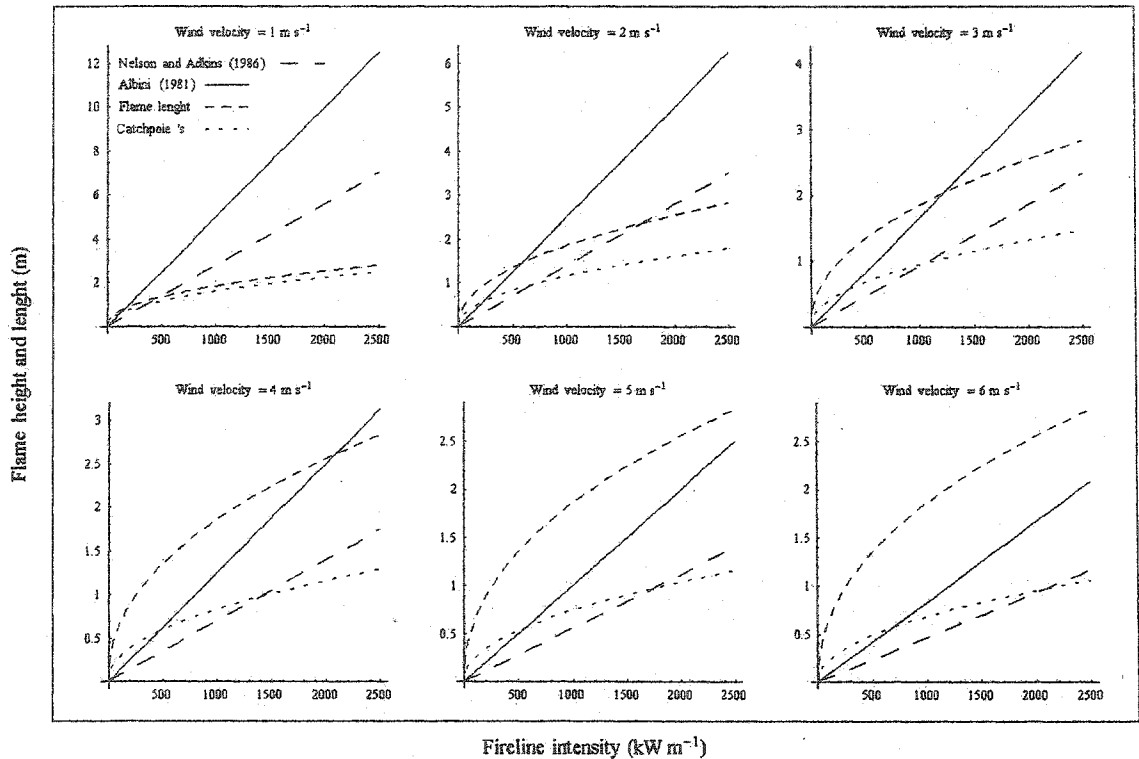
Figure 1.9. Comparison between flame height models based on fireline intensity and wind speed as inputs. Predictions from Byram's (1959) flame length model are also given for comparative purposes.

1.4.6. Buoyant plume model

The buoyant plume model is a critical sub-model in the model system due to the importance of convective heat transfer in crown fire initiation. The Mercer and Weber (1994) plume model is a simplified mathematical description of the plume as it solves the conservation equations for the plume centerline assuming a top hat approach. This formulation does not take into account the interaction between the fire generated buoyant

54

flows and the surrounding ambient wind field. There have been a number of models solving Navier-Stokes equations that attempt to describe this interaction (e.g., Lopes et al. 1995, Clark et al. 1996, Porterie et al. 1999). Although the advantages of such approach, namely a more accurate description of the flow field around and above a surface fire, are theoretically appealing, none of these models have been quantitatively evaluated against data collected on outdoor experimental fires. This arises from the inherent difficulty in defining what the buoyant plume is from field measurements. For implementation in the present study model system, an obvious advantage of the Mercer and Weber (1994) formulation, or others such Morton et al. (1956) and Gould et al. (1997), over the solution of the Navier-Stokes equations is the quick solution of the system of equations describing the plume. The modular approach followed in the present study would allow the exchange of the Mercer and Weber (1994) model with another model if any obvious advantages existed.

Although no quantitative evaluation of the plume model has been carried out, qualitative evaluation of the model results indicates reasonable behavior. The model responds well to changes in fire strength, namely the changes in initial plume diameter and initial vertical velocity (eq. [1.12]). This is illustrated in Figs. 1.10a – 1.10i with a plot of the predicted plume temperature field for nine distinct combinations of within stand wind speed and plume initial vertical velocity. In this idealized situation, plume half-width is kept constant. An increase in wind velocity induces (1) the transport of the plume downwind; (2) decreases plume diameter due to increased entrainment; and (3) a decrease in temperature reached at a given height. This decrease in temperature at a given height result from the dilution of the plume and the increase in the distance that any

55

given point is from the source due to plume tilting. The opposite occurs with an increase in the initial vertical velocity in the plume. An increase in the plume's initial vertical velocity will increase the vertical momentum and strength of the plume, limiting plume dilution and consequently maintaining higher air temperatures for any given height.

The case of Figs. 1.10a – 1.10i is illustrative of plume behavior, but not a faithful representation of the dynamics occurring in wildland fires. The increase in wind velocity will induce an increase in rate of spread and consequently in the rate of burning and flame front depth. These two factors determine plume dynamics by increasing, respectively, the vertical momentum in the plume and the plume diameter. By integrating the various model components necessary to simulate plume behavior we are able to better understand how wind affects the distribution of air temperatures above a surface fire. Figs. 1.11a – 1.11i illustrates the predicted plume temperature fields for distinct idealized surface fuelbeds. Three different surface fuel models, NFFL 2 and 5 (Anderson 1982) and a custom fuel model for red pine plantations, RPFM,[2] (Cruz et al. 2004) were used to describe surface fire properties through the use of Rothermel (1972) surface fire rate of spread and Nelson (2003b) reaction time models for various within stand wind speeds. The distinct predictions for fireline intensity and flame depth result in the differences in plume behavior as presented in Figs. 1.11a – 11i. This illustrates the sensitivity of the plume model to the variation in burning conditions.

---

[2] Based on Van Wagner (1968). The fuel model description for RPFM was: 1-h TL fuel load - 0.3 kg m^{-2}; 10-h TL fuel load 0.3 kg m^{-2}; surface area-to-volume ratio - 5500 m^{-1}; fuelbed depth - 0.25 m; moisture of extinction - 55%; H_c - 19 600 kJ kg^{-1}.

56
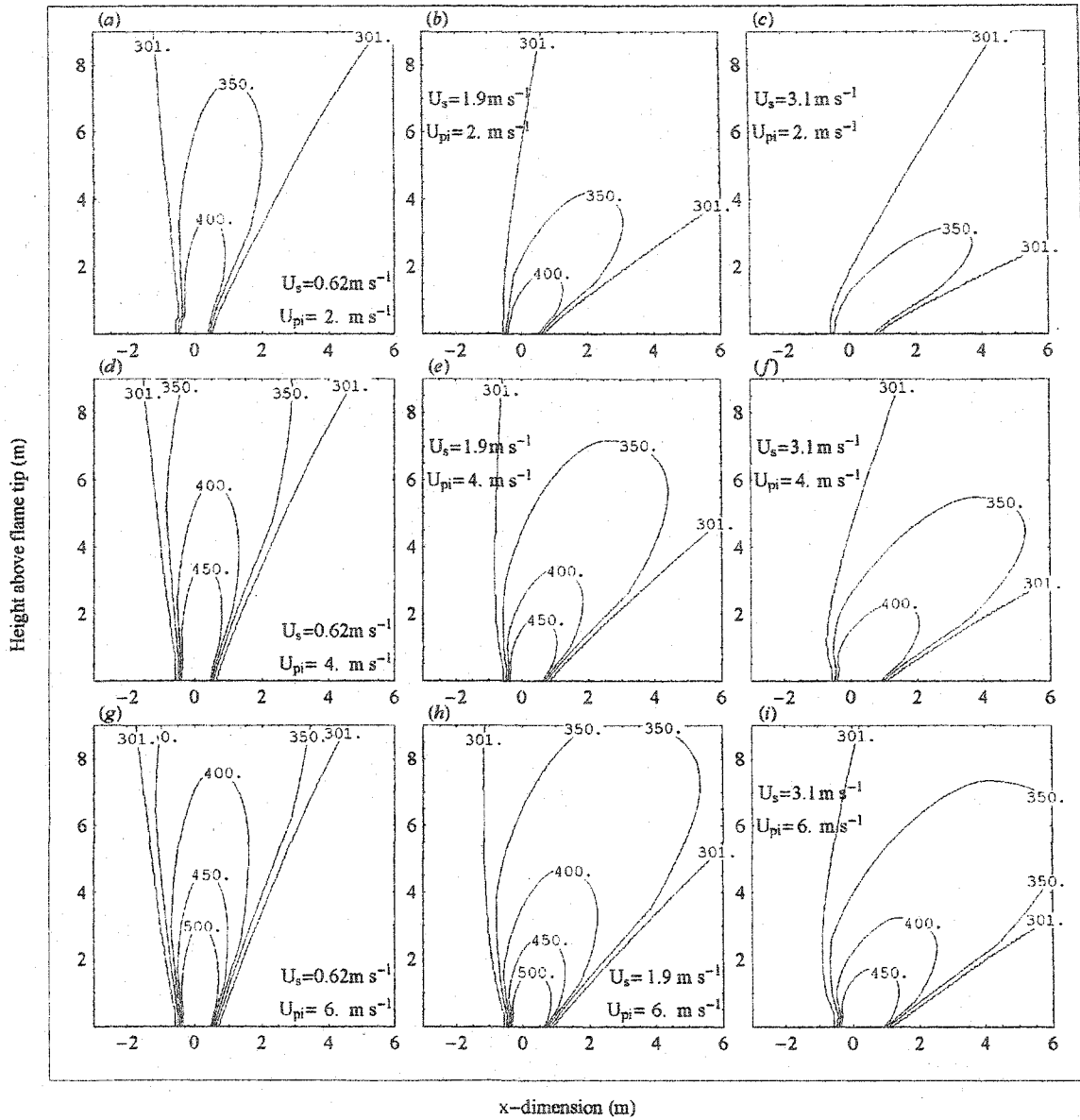
Figure 1.10. Buoyant plume steady state trajectory, dimensions and temperature distribution (K) for variable horizontal within stand wind velocity (U_s) and initial vertical plume velocity (U_{pi}). Simulations are based on a fixed initial plume half-width of 1.0 m.

57
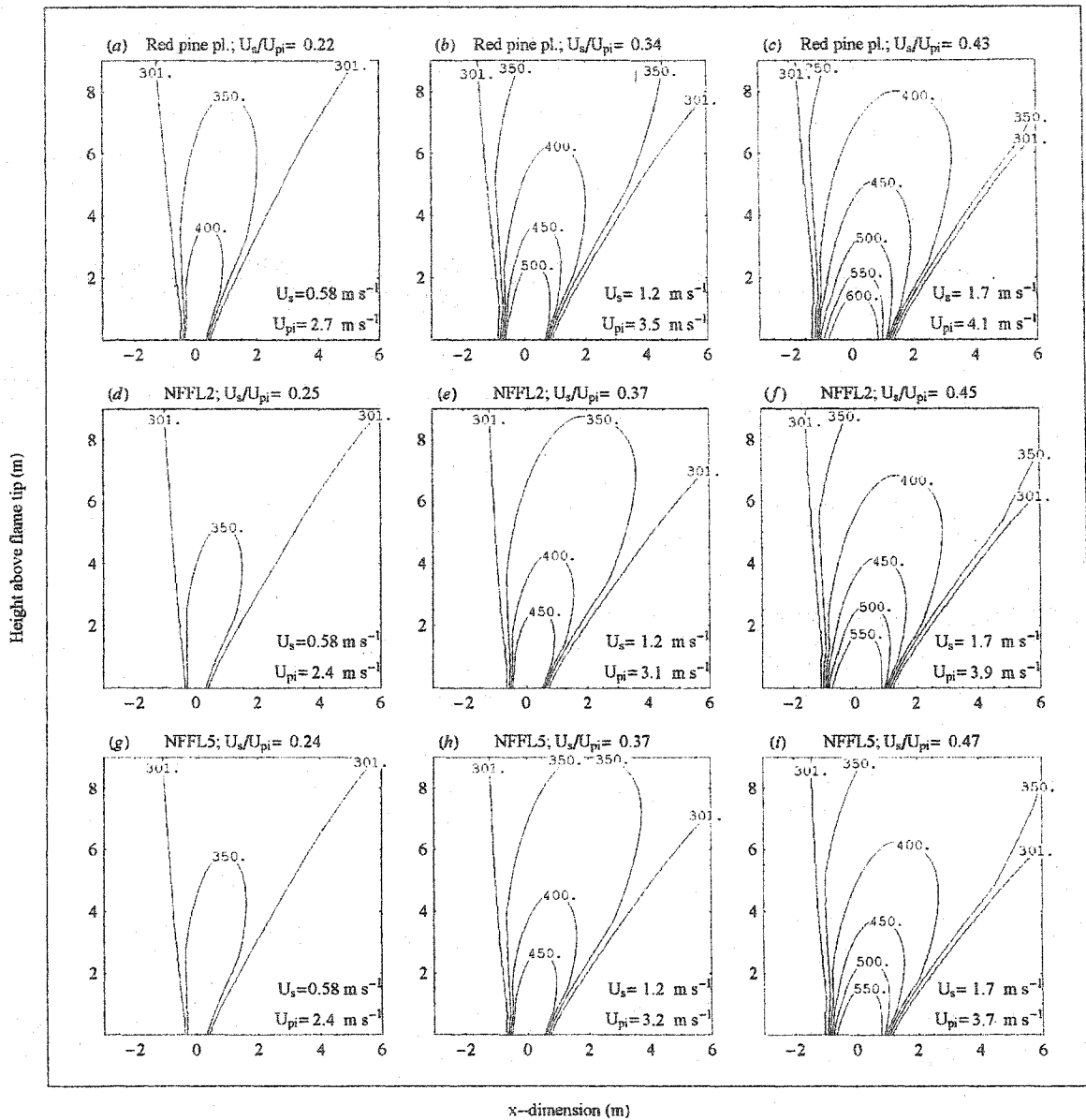
Figure 1.11. Buoyant plume steady state trajectory, dimensions and temperature distribution (K) for three distinct fuel models under variable horizontal within stand wind velocity (U_s).

Another important input parameter in the plume model is the initial plume temperature. No consistent value for flame tip temperature has been found in the literature. Besides the two experimentally derived values already referenced from the studies of Thomas (1963) and Cox and Chitty (1980), other values can be found in the

58

literature ranging from 573 K (Sullivan et al. 2003) from field measurements to 900 K (Mercer and Weber 1994, Porterie et al. 1999). As described previously, air or fluid temperatures above and within the flame as reported in the literature are subject to large experimental errors due to radiation losses and the physical properties of the thermocouples. Ultimately the reported values are thermocouple temperatures (Alexander 1998). Given this, the initial plume temperature was defined from physical reasoning. Siegel and Howell (1992) indicate the Draper point, ≈800 K, as the temperature at which red light first becomes visible. This temperature could then be used to loosely define the flame tip temperature. This value is consistent with the flame height model that was parameterized from observations of flames in the visible wavelengths. Nevertheless, the temperature as defined above does not assure that exothermic reactions in the fire plume have ceased.

As implemented, the plume model does not integrate the heat release by the smoldering combustion of duff and large size dead fuels such as downed woody fuels (Rothermel 1994). Although the combustion of duff and large size fuels might have only a reduced contribution to the horizontal spread of the fire (Forestry Canada Fire Danger Group 1992, Rothermel 1994), the heat released by these fuels behind the fire front may contribute to other fire characteristics, such as scorching and ignition of canopy foliage, the power of the fire (Byram 1959, Nelson 2003a), and suppression difficulties (Sullivan et al. 2002). The contribution of the combustion of the large fuels is responsible for the occasional torching of individual or group of trees tens of meters behind the leading fire edge. The effect of smoldering combustion in the buoyant plume could be implemented by considering the combustion rate of duff and large woody fuels (Cheney 1981, Albini

59

and Reinhardt 1995) and the amount of large fuels available. This would nevertheless require the solution of a more complex plume model, in which both air temperature and velocity would vary with time and location.

1.4.7. Specific heat formulation

The simplified heat balance equation (eq. [1.16]) describes how the crown fuel particles increase their internal energy while being subjected to a certain heat flux. For any given species foliar moisture content is the fuel particle variable determining the net energy required to increase fuel particles to their ignition temperature (Xanthopoulos and Wakimoto 1993). Previous fire models incorporating the effect of moisture content on the heating of fuel particles have done so assuming that all moisture needs to be driven out before ignition occurs (Alexander 1998). By considering the existence of moisture in the fuel particles several authors have modeled the fuel particle temperature increase through a 3-step model (Albini 1985, 1996, Wotton and Renaud 1996). This model considers: (1) the energy required to take the fuel particle with MC fractional moisture content from ambient to 373 K; (2) the energy required to evaporate the water; and (3) the energy require to take the dry fuel particle from 373 K to ignition temperature, 600 K. By assuming that water is continuously evaporated from the fuel particles until 373 K is reached, de Mestre et al. (1989) suggest a two step model, where the first step integrates both the effect of the specific and latent heat of water in raising the heat requirements as the fuel particle temperature increases up to 373, and the second step assumes that the fuel particle is dry. By assuming that moisture is continuously being evaporated from the fuel particle from ambient temperature to ignition temperature, the specific heat term in

eq. [1.16] is based on the average specific heat integrating both the fuel and water specific heats and the latent heat of water (eq. [1.27]).

Due mainly to convective heating, which depends on the temperature of the absorbing surface, these models produce a distinct fuel temperature – time profiles. Fig. 1.12 displays such profiles for the 3-step and eq. [1.27] models under a certain convective heat flux as determined by the plotted air temperature. As both net convective and radiative heat transfer to the fuel particle are a function of the fuel particle surface temperature (eqs. [1.18] and [1.21]), the time required to heat a fuel particle to ignition temperature will vary depending on the fuel temperature profile. For the same heating conditions, the 3-step model will always attain ignition temperature earlier than that produced by eq. [1.27]. This is the result of the latent heat effect that produces the flat section on the fuel particle temperature profile. While in the flat portion, the differences between the air and fuel temperatures are higher than what is predicted by eq. [1.27], resulting in higher net heat transfer rates, and consequently an earlier attainment of ignition temperature. The smooth curve generated by eq. [1.27] seems more realistic in light of recent research (e.g., Catchpole et al. 2002, Smith et al. 2003).
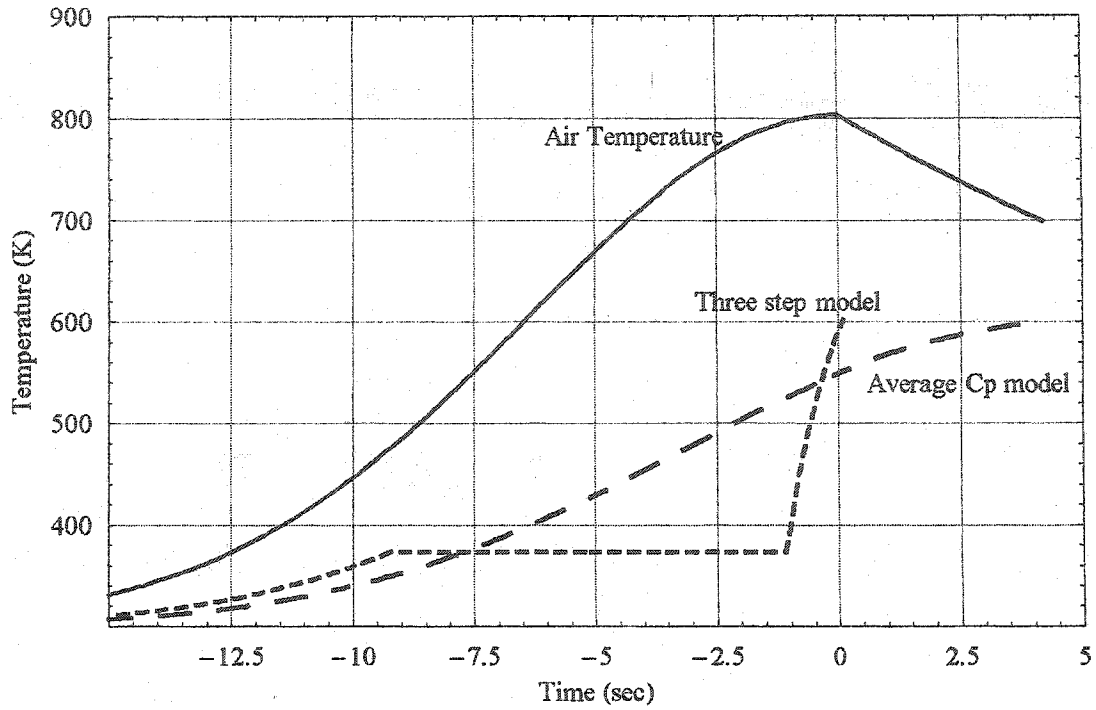
Figure 1.12. Estimated fuel temperature profile from the 3-step and average specific heat fuel heating models under a convective heat flux defined by air temperature and a fixed air velocity of 10 m s^{-1}. Simulations based on a fuel particle with a characteristic σ of 5900 m^{-1} and FMC of 1.5.

1.5. Model behavior

Models can contribute to the understanding of natural processes through the conceptualization of the process structure and quantification of the effect of input parameters and intermediate processes. The CFIM response to changes in input and intermediate output parameters was analyzed to better understand the effect of the input parameters and intermediate models on the overall model system behavior. The effect of individual inputs and sub-models assumes particular importance in a complex model system like the one described in this study. The combined sub-models, e.g., rate of spread, reaction time and flame height, can lead to error propagation problems due to the contribution of individual parameter and model errors to final output uncertainty. This aspect was analyzed by Cruz et al. (2004) when characterizing error propagation in systems to predict crown fire initiation based on Rothermel (1972) surface fire spread and Van Wagner (1977) crown fire initiation models.

Due to the problem of error propagation, the number of sub-models integrated in the model system can be seen as a substantial disadvantage to the present modeling approach. A reduction in the number of sub-models would require a better understanding of the chemical and physical processes occurring in a fire front. An example would be the inclusion of a combustion model based on the conservation equations that would produce estimates of reaction times and flame temperature. Nevertheless, based on the current state-of-knowledge concerning combustion, fluid dynamics and heat transfer processes occurring in a heterogeneous medium as found on a wildland fire environment, a pure fundamental modeling approach seems unrealistic and impractical for field use. This approach would require formulating and solving the equations of state and

63

continuity at a molecular level and without any kind of empiricism, in order to track the fuel state and their interaction as they were heated, change state, release their bond energy and energy is transferred to unignited fuels.

1.5.1. Effect of individual inputs variables on intermediate model outputs

Of the various weather and fuel complex variables that affect fire behavior, six variables were selected for analysis because of their dominant effect on the process of ignition of crown fuels. They were: wind speed, surface fuel available for flaming combustion, fuel strata gap, moisture content of fine dead surface fuels, foliar moisture content and surface area to volume ratio of crown fuel particles. Some of these input variables affect various sub-models. In order to better understand the effect of intermediate model outputs on the final model behavior, the effect of fire rate of spread, reaction time, maximum flame temperature, and wind profile models were analyzed independently. The effect of these input/intermediate output variables in CFIM was analyzed through the impact on the predicted crown fuel temperature history, the final model output, and variables determining the convective and radiative heat sources and transfer processes, namely the surface fire T-T profile, the convective heat transfer coefficient, and the incident convective and radiative heat fluxes.

The effect of the wind profile model was analyzed by changing the wind attenuation coefficient (eq. [1.10]). The baseline values (in bold) for the various simulations and the variability in the parameter being varied are presented in Table 1.4. The parameters were varied within a range expected to be found in both prescribed and wild fires.

64

Table 1.4. Baseline values for input/intermediate models outputs parameters (in bold at center) and variability used in simulations to analyze model behavior.

	Parameter variability
Input variables	
U_{10} (m s^{-1})	2, 4, **6**, 8, 10
MC (fraction)	0.03, 0.045, **0.06**, 0.075, 0.09
w_a (kg m^{-2})	0.5, 0.7, **0.9**, 1.1, 1.3
FSG (m)	3, 4, **5**, 6, 7,
FMC (fraction)	0.8, 1.0, **1.2**, 1.4, 1.6
σ (m^{-1})	3000, 4000, **5000**, 6000, 7000
Intermediate model outputs	
ROS (m s^{-1})	0.02, 0.05, **0.08**, 0.11, 0.14
τ_r (s)	20, 35, **50**, 65, 80
T_{Fmax} (K)	900, 1000, **1100**, 1200, 1300
α_U	1, 1.5, **2**, 2.5, 3

Fig. 1.13a – 1.13d displays how the final model output – i.e., the canopy fuel particle temperature profile, change with the variation in the perturbed intermediate output parameters. The 0 in the x-location indicate that the ignition interface of the surface fire is directly beneath the crown fuel particle being heated. The initial steep increase in fuel particle temperature happens before the flame front arrives. At this point the fuel particle is inside the buoyant plume while at the same time being subjected to substantial radiative heating. After reaching its maximum temperature, the fuel particle temperature decreases at a rate that is a function of the incident radiative heat flux while at the same time it is being subjected to convective cooling. The model simulation stops when the fuel particle reaches ignition temperature (e.g., Fig. 1.13b, simulations for a reaction time of 80 s).

For the simulation of the effect of rate of spread (Fig. 1.13a) this variable was varied over a range that would represent a moderate to a very-high intensity surface fire. Reaction time was also varied over a broad range covering the reaction time expected to

occur in lightly uncompacted to heavily compacted surface fuel beds (Fig. 1.13b). Surface fire maximum flame temperature was varied from 900 to 1300 K, values characteristic of thin and deep flames observed respectively in low intensity and high intensity wildland fires (Chapter 2). The wind attenuation coefficient was varied within the bounds found in field studies (Amiro 1990). *ROS* showed the most effect on the variability of the predicted temperature of the canopy fuel particles. By determining fireline intensity and flame depth, *ROS* predictions affect three important intermediate components of the model system, flame height, the size of the radiative surface and the initial air velocity in the buoyant plume. This explains the effect of the surface fire rate of spread on both the convective and radiative heat fluxes (Fig. 1.15c and 1.15d). A twofold increase in the predicted *ROS* will double the size of the radiating surface, and the width, and consequently the integrity, of the buoyant plume. It should be noted that if this increase in *ROS* is caused by an increase in wind speed, the plume would also be subjected to higher entrainment. Similarly, flame height will be impacted in distinct ways (eq. [1.13]). The effect of the increase in fireline intensity in increasing flame height will be counteracted by the effect of the increase in wind speed in tilting the flame front.

Reaction time affects mostly the size of the radiating surface and the buoyant plume initial half-width. As with *ROS*, these result in an increase in the incident radiant heat flux to the lower canopy fuel particles and the increase in the integrity of the buoyant plume, and consequently conservation of its thermal energy and momentum. For the range of *ROS* and τ_r simulated, the effect of these intermediate outputs on the increase in

the incident radiative heat flux is comparable (Fig. 1.15d and 1.16d) while the effect of

ROS in the convective heat flux is much larger (Fig. 15c and 1.16c).
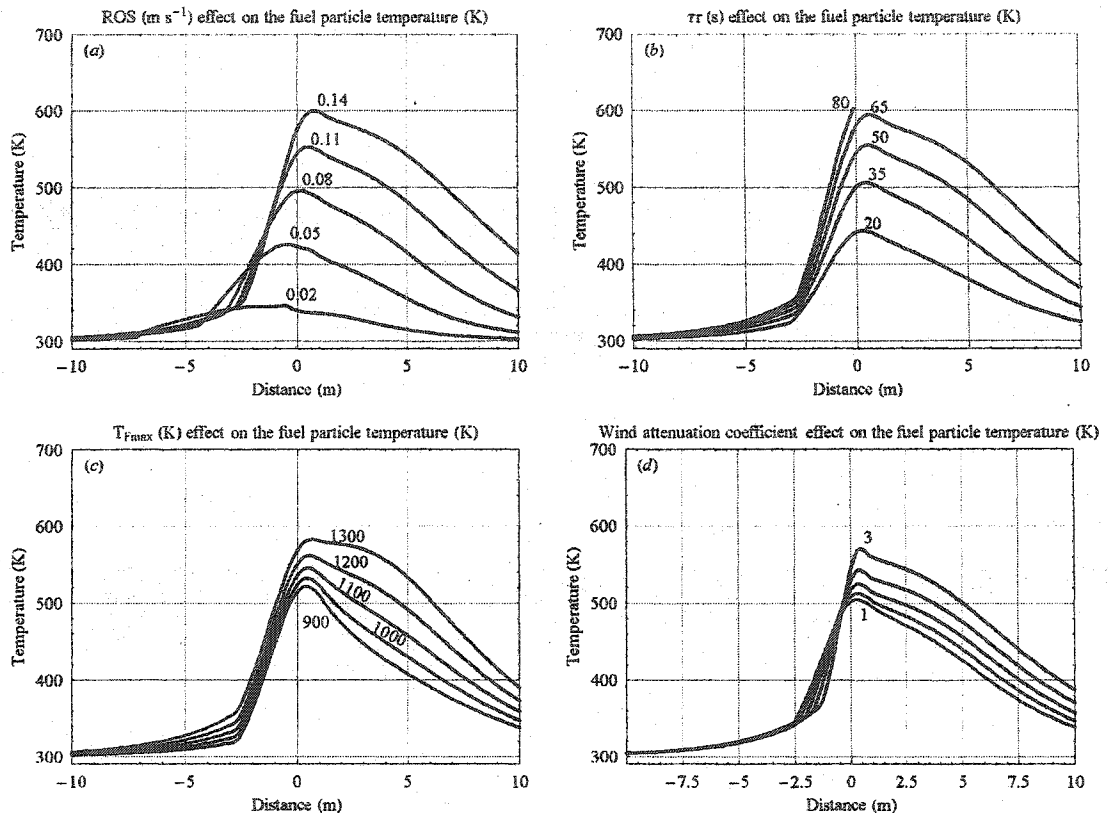
Figure 1.13. Predicted temperature of lower canopy fuel particles above a spreading surface fire as a function of intermediate model outputs: (a) rate of spread (m s^{-1}); (b) reaction time (s); (c) maximum flame temperature (K); and (d) wind attenuation coefficient. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. Baseline values for simulations are given in Table 1.4. Space did not permit all of the intermediate wind attenuation coefficient values (1.5, 2, 2.5) to be plotted in (d).

Both T_{Fmax} and α_U have less effect in determining the predicted fuel particle temperature than *ROS* and τ_r, but are still noteworthy (Fig. 1.13c and 13d). Changes in T_{Fmax} affect the T-T profile, and therefore the incident radiative heat flux. The flame radiometric temperature is raised to the fourth power in the Stefan-Boltzamn equation

(eq. [1.1]), making the radiant heat flux very sensitive to changes in flame temperature at the higher range of the T-T profile. Although the changes in the incident radiative heat flux are similar to the ones simulated for ROS and τ_r, the effect on the fuel particle temperature profile are much smaller. T_{Fmax} affects only the radiant heat transfer process, whereas ROS and τ_r, affect both the incident radiant and convective heat fluxes by increasing in plume width and air temperature at any given height.

The wind attenuation coefficient determines the reduction in wind speed within the stand relative to the wind speed at the canopy top. Consequently its effect is rather restricted as the changes in within-stand wind speed over the range of a_U values tested are small (Fig. 1.1.4).

Of the various input variables under analysis, the 10-m open wind speed (Fig. 1.14a) and surface fuel consumed during the flaming combustion phase (Fig. 1.14c) showed the most effect on the process of heating the canopy fuels. The strong effect that these two variables exert on the model predictions is due to their influence on fireline intensity and flame depth. These two input parameters will determine, respectively, flame height and the depth of the radiating surface. U_{10} affects the surface fire rate of spread, fireline intensity and therefore flame height. Although an increase in U_{10} has a strong effect on the overall incident heat flux to the canopy fuels, this variable affects heat transfer processes in different ways. The increase in U_{10} will increase both convective cooling prior and after the passage of the buoyant plume and air entrainment in the plume.

68
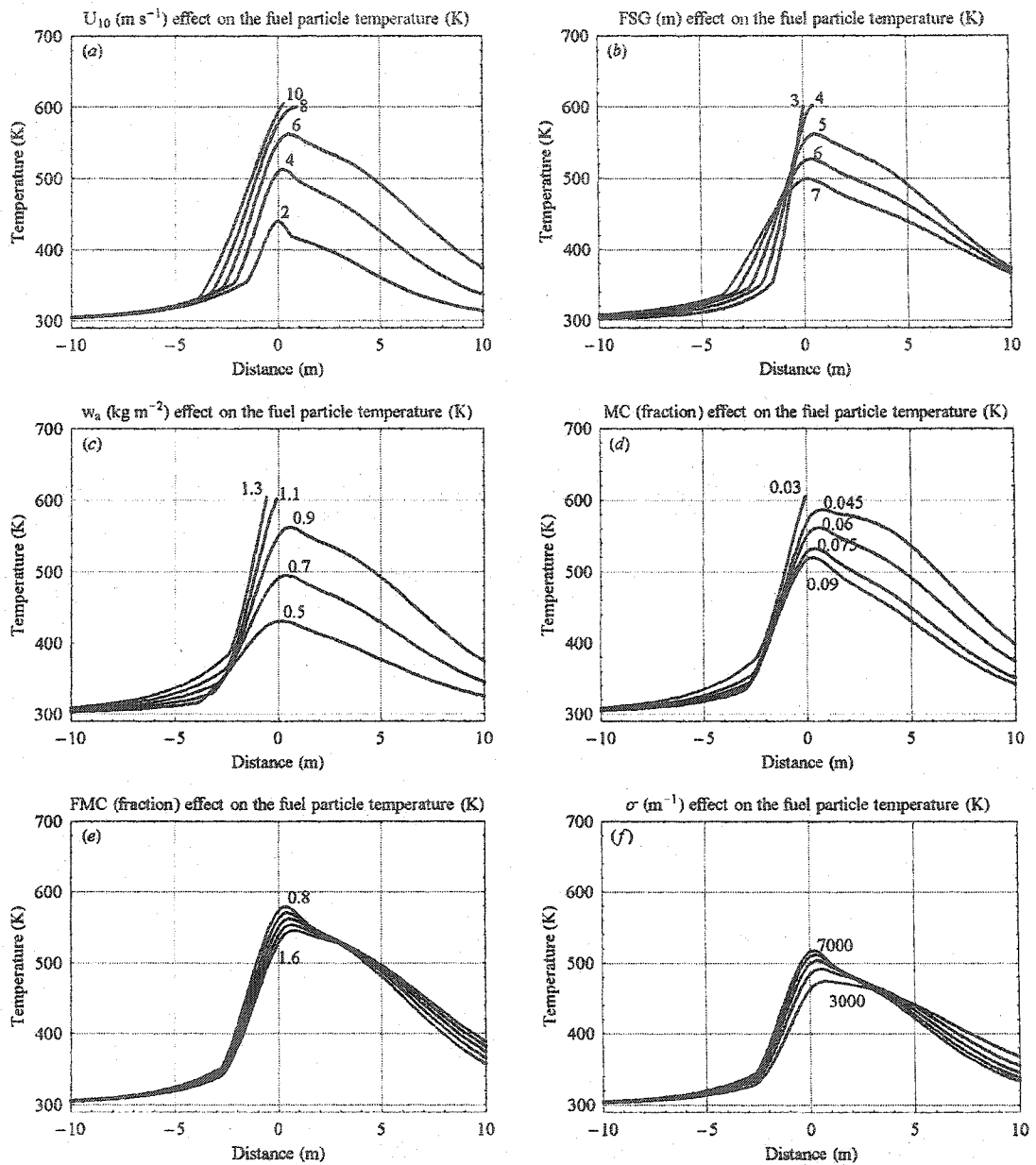
Figure 1.14. Predicted temperature of lower canopy fuel particles above a spreading surface fire as a function of various input parameters: (a) wind speed (m s^{-1}); (b) fuel strata gap (m); (c) surface fuel available for flaming combustion (kg m^{-2}); (d) surface fuel moisture content (fraction); (e) foliar moisture content (fraction); and (f) crown fuel particles surface area to volume ratio (m^{-1}). Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. In some cases, space did not permit all of the intermediate input values for *FMC* and surface area to volume ratio to be plotted. Baseline values for simulations are given in Table 1.4.

69
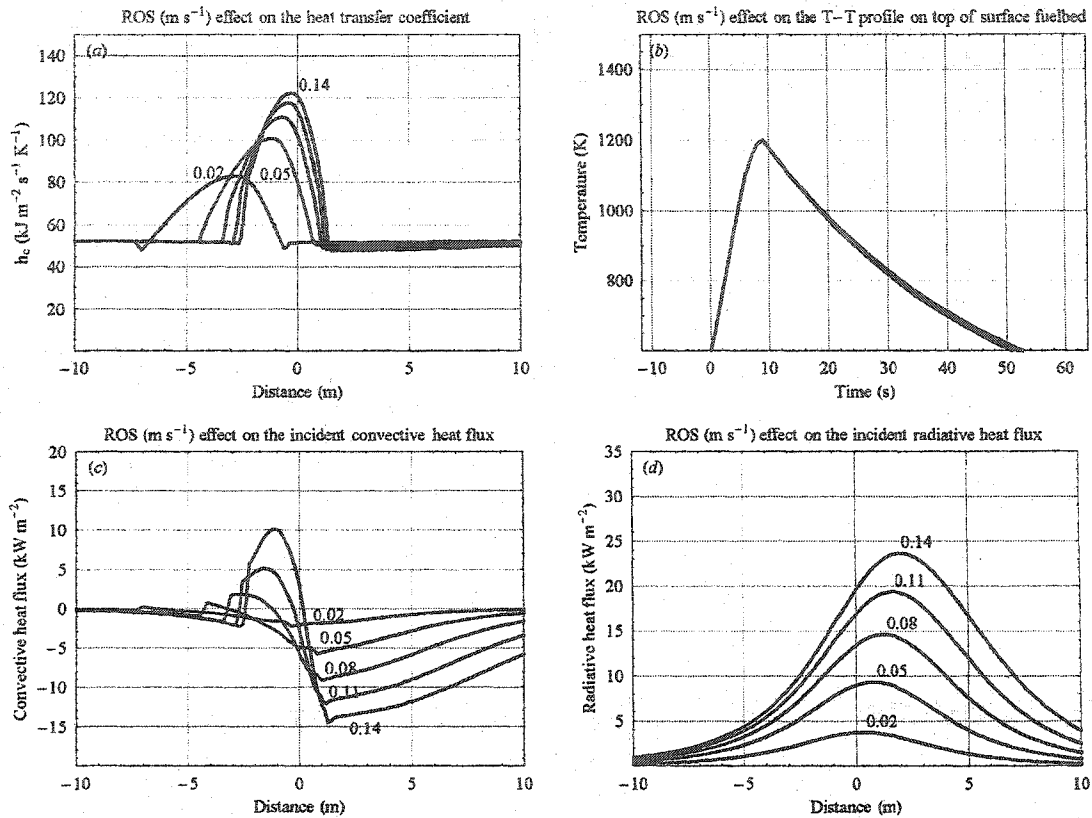
Figure 1.15. Effect of surface fire rate of spread on the intermediate outputs determining and characterizing the convective and radiative heat flux reaching the base of the canopy fuels: (a) effect on the convective heat transfer coefficient; (b) effect on the surface fire T-T curve; (c) effect on the incident convective heat flux; and (d) effect on the incident radiative heat flux. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. On some cases, space did not permit all intermediate rate of fire spread values to be plotted. Baseline values for simulations are given in Table 1.4.

The w_a determines fireline intensity, flame height and reaction time. Flame height assumes particular importance as this variable determines the z-location of the base of the buoyant plume. For the same FSG, the higher flames result in less plume degradation, which causes an increase in the convective heating of the fuel particles (Fig. 1.20). Contrary to the influence of U_{10}, which has positive and negative effect in the heat

70

transfer processes, the increase in w_a results in proportional increases in both the radiative

and convective heat fluxes to the canopy fuels.
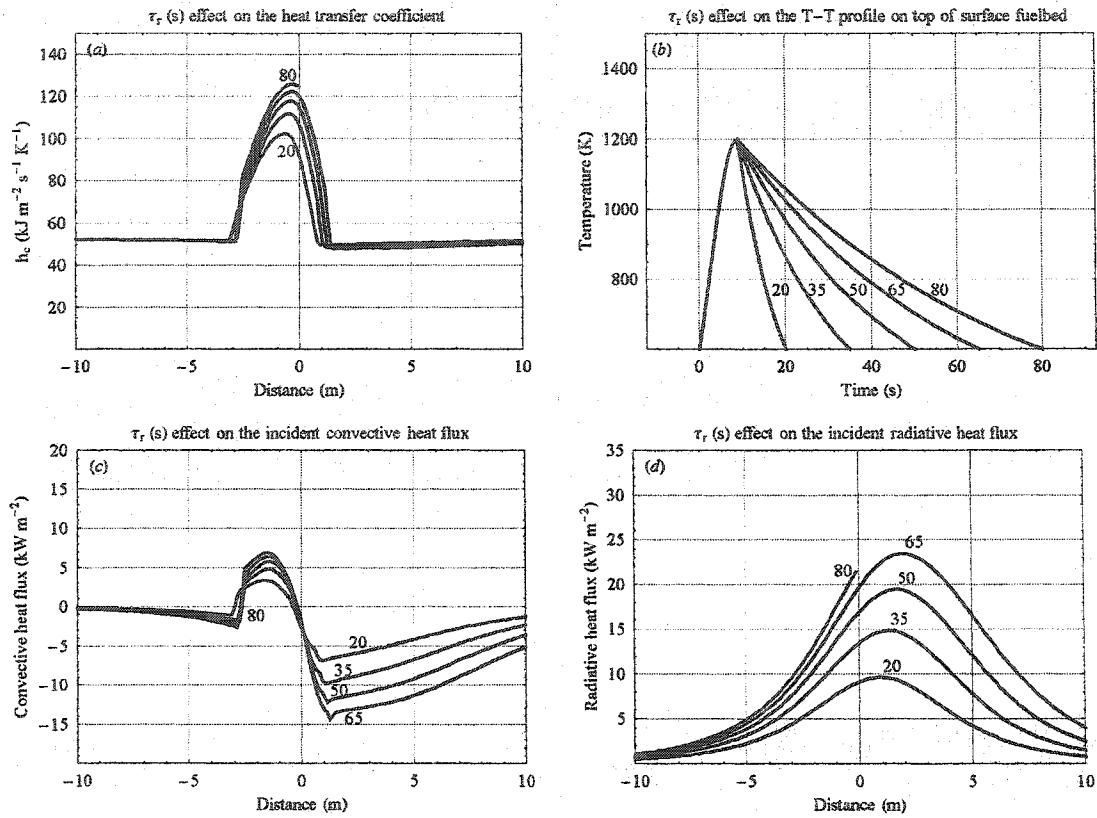
Figure 1.16. Effect of surface fire reaction time on intermediate outputs determining and characterizing the convective and radiative heat flux reaching the base of the canopy fuels: (a) effect on the convective heat transfer coefficient; (b) effect on the surface fire T-T curve; (c) effect on the incident convective heat flux; and (d) effect on the incident radiative heat flux. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. Space did not permit all intermediate reaction time values to be plotted in (a). Baseline values for simulations are given in Table 1.4.

Fuel strata gap and moisture content of the fine dead fuels (*MC*) in the surface

fuel layer also showed a strong effect on the model output, albeit lower than U_{10} and w_a

(Fig. 1.14b and 1.14d). The effect of *FSG* in the model system is solely related to

71

changes in the incident radiative (due to the reduction in the view factor with height) and convective heat (due to air entrainment and consequent cooling of the plume) flux at any height.

The effect of MC on model output is restricted to ROS and T_{Fmax}, and the subsequent effect on heat transfer processes as described before for these two intermediate fire behavior properties. The effect of MC in the system is unidirectional -- i.e., the increase in surface fine fuels dryness result in a directly proportional increase in ROS, and consequently in fireline intensity, flame height, flame depth and buoyant plume strength. Conversely, a reduction in MC results in a higher T_{Fmax} value and consequently an increase in the incident radiative heat flux to the canopy (Fig 1.22b and 1.22d).
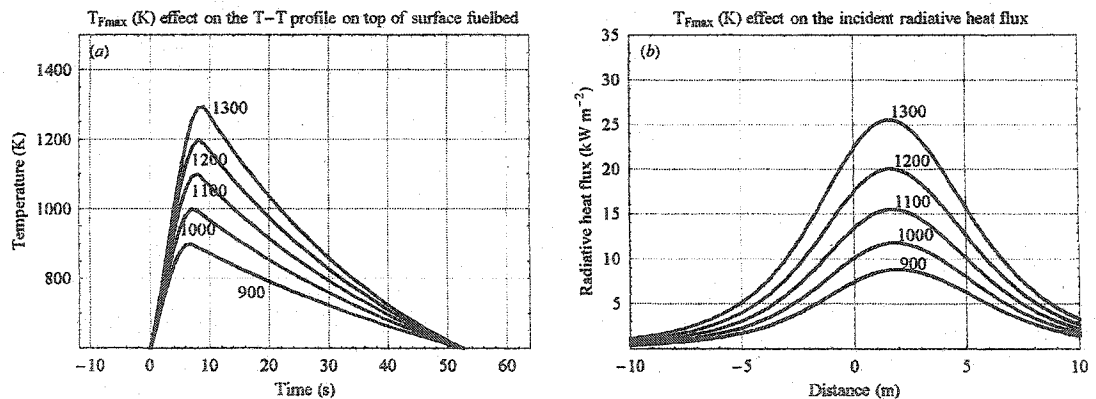
Figure 1.17. Effect of surface fire maximum flame temperature on: (a) the surface fire T-T curve; and (b) the incident radiative heat flux. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. Baseline values for simulations are given in Table 1.4.
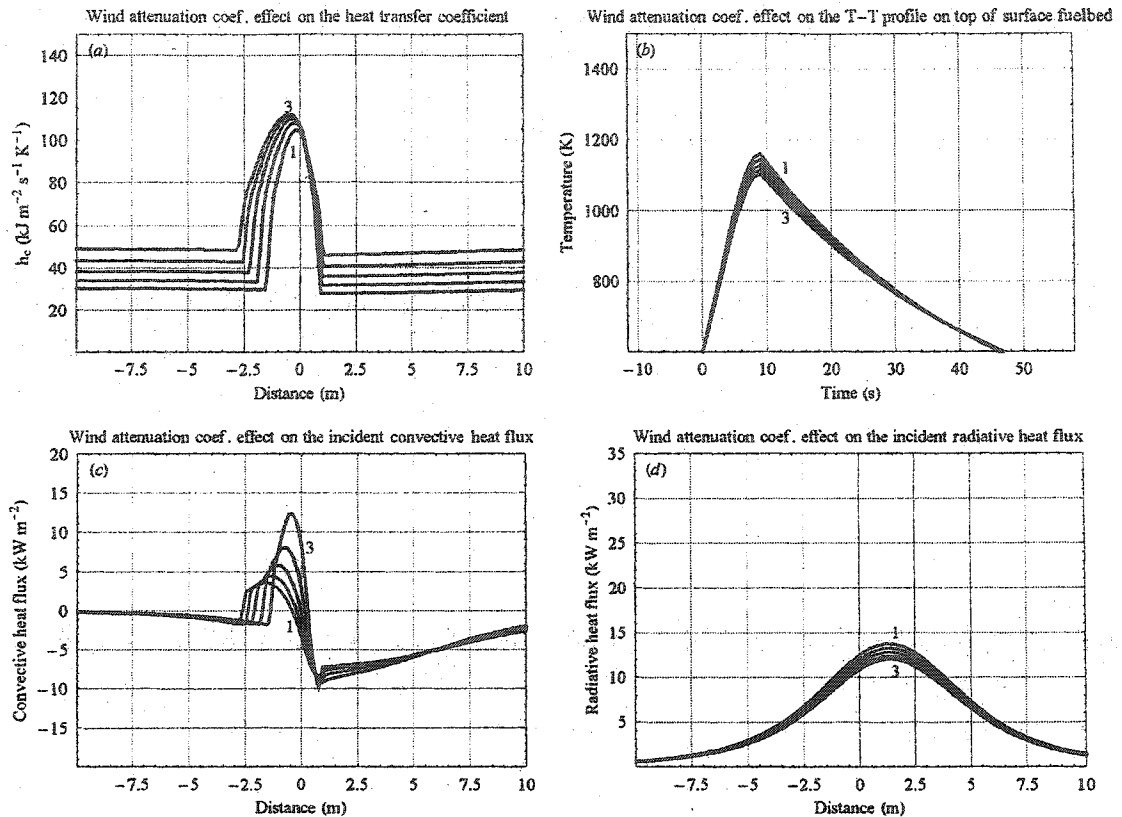
Figure 1.18. Effect of the wind attenuation coefficient on intermediate outputs determining and characterizing the convective and radiative heat flux reaching the base of the canopy fuels: (a) effect on the convective heat transfer coefficient; (b) effect on the surface fire T-T curve; (c) effect on the incident convective heat flux; and (d) effect on the incident radiative heat flux. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. In all cases, space did not permit all of the wind attenuation coefficient values to be plotted. Baseline values for simulations are given in Table 1.4.

The two variables determining the heat sink in the canopy fuel layer, *FMC* and σ showed the least effect on the profiles of fuel particle temperature (Fig. 1.14e and 1.14f). *FMC* determines the heat required to take the fuel particle to ignition by increasing the canopy fuel average specific heat (eq. [1.27]). The small effect of *FMC* in the simulated canopy fuel temperature profile might be explained by the relative small effect that the increase in the average specific heat has when compared to the magnitude of the heat fluxes above a vigorous, high intensity surface fire. Fuel particles in the canopy are

73

subjected to continuous heating of variable duration while the surface fire approaches and passes under their location (Alexander 1998). The increase in the heat requirements for fuel particle ignition due to the increase in *FMC* are comparatively small compared to the cumulative heat flux absorbed by the fuel particles. This theoretical result corroborates the analysis of Cruz et al. (2004). Through the analysis of a field experimental dataset the authors failed to find a statistically significant effect of FMC on the likelihood of crown fire occurrence. The reduced effect that σ has in the heat transferred to the fuel particle hints that, under the range tested, the effect of the variation of this variable in the system is also small compared to the effect of the surface fire heat flux.

The simulations in Fig. 1.19 show the effect of U_{10} on the intermediate outputs that determine the convective and radiative heat fluxes to the crown fuel particles. The simulations of the effect of U_{10} on the convective heat transfer coefficient suggests that an increase in U_{10} results in an increase in the convective heat transfer coefficient and on the length of time the fuel particle is being subjected to the hot gases in the plume (Fig. 1.19a and 1.19c). This arises from the effect of U_{10} on surface fire behavior, namely an increase in rate of spread and in turn fireline intensity. The increase in the rate of spread induces a deeper flame depth and consequently a wider buoyant plume. The increase in fireline intensity also induces a higher flame height and higher air velocities inside the plume. Although the wind effect on the T-T curve seems relatively small (Fig. 1.19b), its impact on the radiative heat flux is substantial (Fig. 1.19d). This arises from the already noted increase in flame depth and the increase of duration the fire exhibit high temperatures.

74
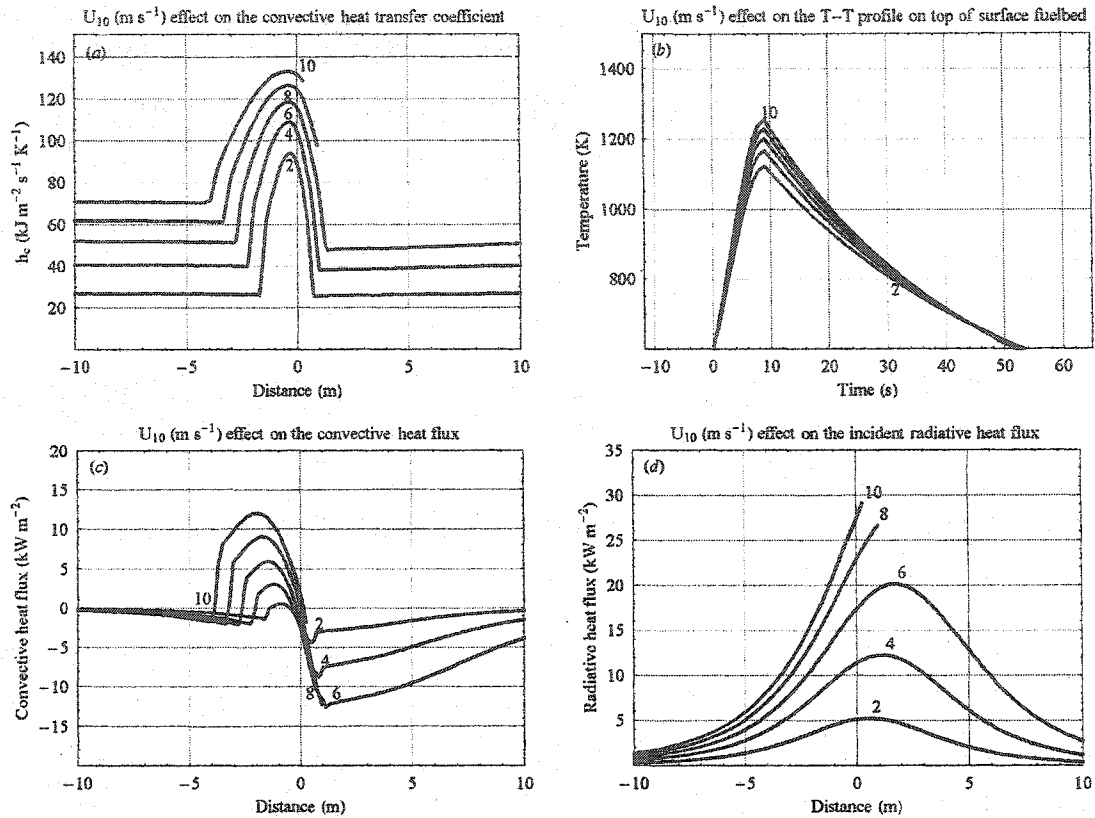
Figure 1.19. Effect of 10-m open wind speed on intermediate outputs determining and characterizing the convective and radiative heat flux reaching the base of the canopy fuels: (a) effect on the convective heat transfer coefficient; (b) effect on the surface fire T-T curve; (c) effect on the incident convective heat flux; and (d) effect on the incident radiative heat flux. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at $x = 0$. Space did not permit all of the intermediate wind speed values to be plotted in (a). Baseline values for simulations are given in Table 1.4.
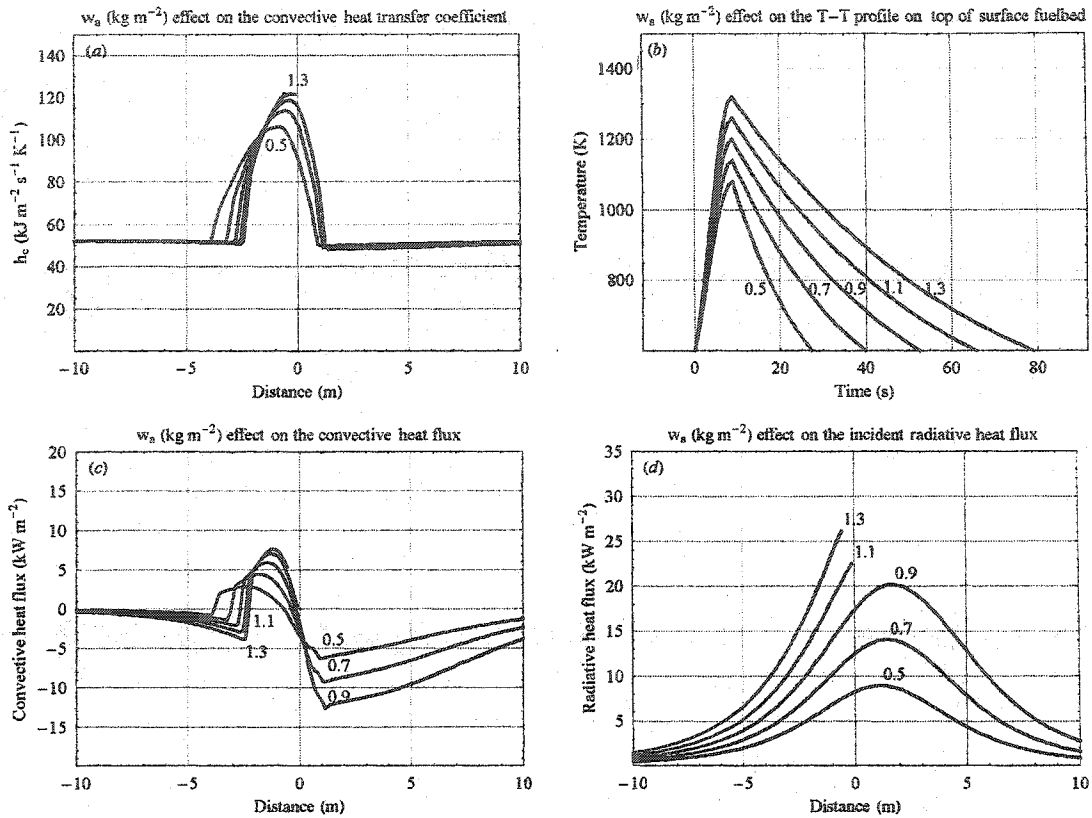
Figure 1.20. Effect of surface fuel available for flaming combustion on intermediate outputs determining and characterizing the convective and radiative heat flux reaching the base of the canopy fuels: (a) effect on the convective heat transfer coefficient; (b) effect on the surface fire T-T curve; (c) effect on the incident convective heat flux; and (d) effect on the incident radiative heat flux. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. Space did not permit all of the surface fuel available for flaming combustion values to be plotted in (a). Baseline values for simulations are given in Table 1.4.

76
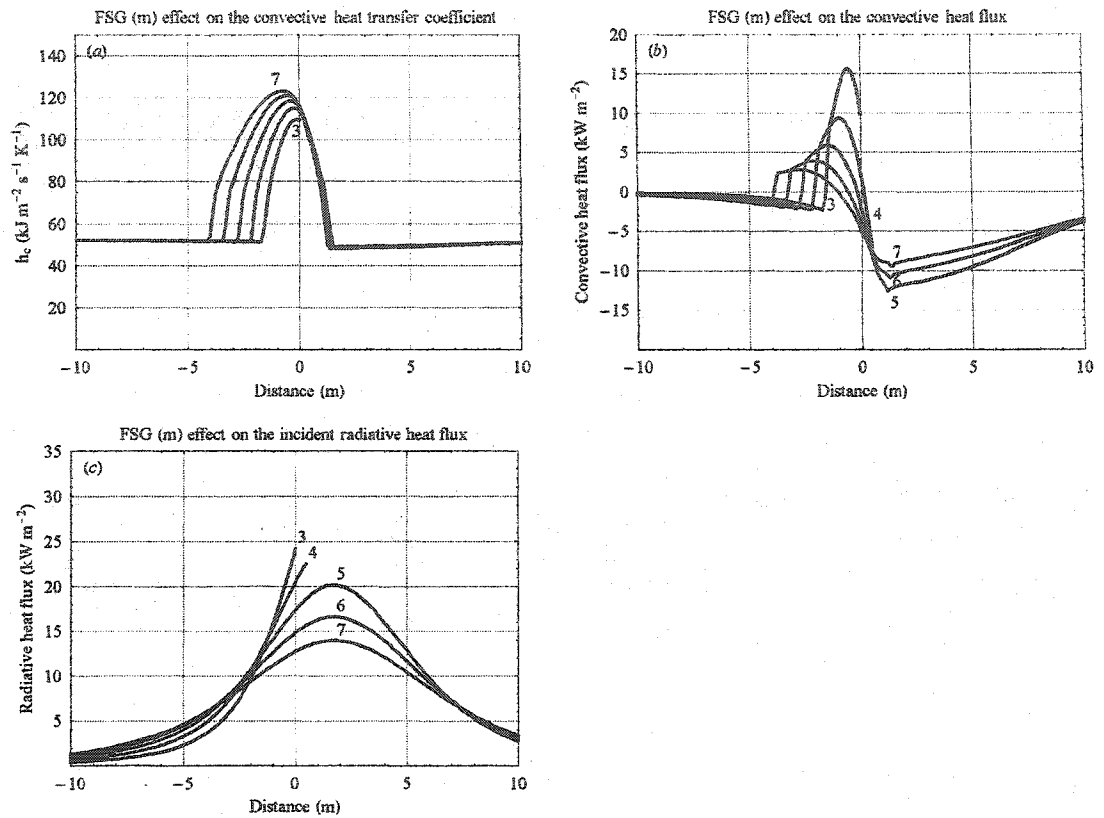
Figure 1.21. Effect of fuel strata gap (m) on determining the convective and radiative heat flux reaching the base of the canopy fuels: (a) variation on the estimated convective heat transfer coefficient; (b) variation on the estimated incident convective heat flux; and (c) variation on the estimated incident radiative heat flux. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. Space did not permit all of the intermediate fuel strata gap values to be plotted in (a). Baseline values for simulations are given in Table 1.4.
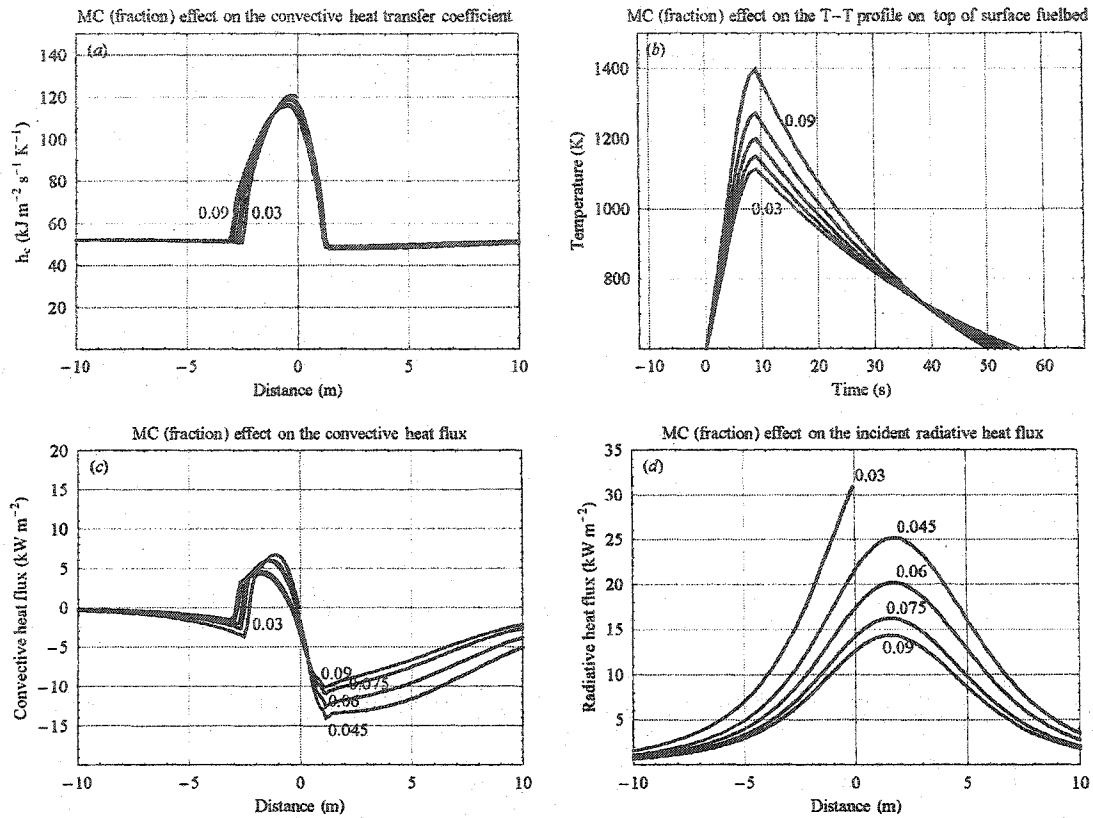
Figure 1.22. Effect of surface fine fuels moisture content (fraction) on intermediate outputs determining and characterizing the convective and radiative heat flux reaching the base of the canopy fuels: (a) effect on the convective heat transfer coefficient; (b) effect on the surface fire T-T curve; (c) effect on the incident convective heat flux; and (d) effect on the incident radiative heat flux. Plots can be interpreted as a snapshot in time while surface fire ignition interface is at x = 0. Space did not permit all of the intermediate moisture content values to be plotted for (a) and (b). Baseline values for simulations are given in Table 1.4.

Surface fuel consumed in flaming combustion impacts model predictions mostly through its effect in the radiative heat flux (Fig. 1.20d). The increase in w_a induces higher maximum flame temperatures and reaction times (Fig. 1.20b). Because w_a is a primary component of fireline intensity it also affects other sub-models such as flame height and the initial plume air velocity, which will influence the amount of convective heat transferred to the fuel particle (Fig. 1.20c).

78

In the CFIM formulation, *FSG* does not influence any fire behavior property, but does directly determine the incident convective and radiative heat fluxes reaching the base of the canopy fuel layer. Both temperature and velocity of air in the buoyant plume decrease with height due to ambient air entrainment. Being closer to the source, the decrease of air temperature in the plume with height (K m^{-1}) is larger. The increase in plume diameter with height results in a longer convective pre-heating of fuel particles (Fig 1.21a and 1.21b). Figure 1.21b shows how the fuel particles situated higher in the canopy are subjected to the convective heat flux earlier, but under cooler conditions, whereas the fuel particles at lower *FSG* are subjected to a shorter but more intense convective pulse. A similar trend describes the incident radiative heat flux to the canopy fuels (Fig. 1.21c). Canopy fuels situated at higher *FSG* values see the fire earlier, and are subjected to higher radiative heat fluxes while the fire is farther from the fuel particle of interest. Depending on other burning conditions, this radiative pre-heating phase might be accompanied by convective cooling until the fuel particles are involved in the buoyant plume. With the approaching of the fire front the lower canopy fuels will "see more fire" as a larger fraction of their surrounding is filled with the emitting surface. The larger view factors calculated for the fuels located closer to the radiating surface while the fire front is directly beneath the fuels result in higher incident radiating energy for lower *FSG* values (Fig. 1.21c and 1.8a – 1.8c). The variation in the view factors with {x,z} location was discussed in Section 1.4.3. The decrease in incident heat flux with height is nonlinear.

Moisture content of surface fine dead fuels affects model output by directly influencing *ROS*, w_a and T_{Fmax} (Fig. 1.22b). This mostly influences the radiative heat

output of the surface fire due to the increase in flame depth and the changing the T-T profile (Fig. 1.22b). The increase in fireline intensity derived from the increase in rate of fire spread will also increase the flame height and the initial air velocity in the plume. Both these intermediate outputs will result in an increase in the convective heat flux to the fuel particles (Fig. 1.22c), albeit a smaller one when compared to the effect of MC on the radiative heat flux.

1.5.2. Sensitivity analysis

Sensitivity analysis was applied to fuel temperature (the maximum attained in the simulation) and three intermediate model outputs determining heat transfer to the fuel particles, air temperature (maximum), convective heat transfer coefficient (maximum) and surface fire radiative power. An index of sensitivity was calculated to quantify the percent change in the outputs to changes in input variables, (i.e., U_{10}, w_a, MC, FMC, FSG, σ) and intermediate fire behavior outputs, (i.e., ROS, I_B and τ_r). The index of sensitivity was defined as (Bartlink 1998, Cruz et al. 2003a):

$$[1.32] \quad RS = \frac{V_{+10\%} - V_{-10\%}}{V_{def}\,0.2}$$

where, $V_{+10\%}$ and $V_{-10\%}$ are the resulting value of the critical parameter when the value of the parameter under analysis is changed by 10%; V_{def} is the resulting value of the critical parameter under default conditions, the value 0.2 is the relative range of the parameter to be analyzed. The 10% intervals were arbitrarily assigned. A RS score indicates the proportional response of the model to the changes in the perturbed input parameter. A sensitivity scale can be drawn from the results. RS scores less than one indicate

insensitive (<0.5) or slightly sensitive (0.5 − 1.0) model responses to inputs; and *RS*

scores larger than one indicate model sensitivity, which can be divided into moderate (1.0

− 2.0) and high (> 2.0).

Table 1.5. Baseline values used in sensitivity analysis.

	"*Normal* summer" conditions	"Summer drought" conditions
U_{10} (m s^{-1})	4, 6	4, 6
1 hr. TL *MC* (fraction)	0.06	0.04
10 hr. TL *MC* (fraction)[1]	0.07	0.05
100 hr. TL *MC* (fraction)	0.08	0.06
FMC (fraction)[2]	1.2	0.85
FSG (m)[3]	4, 6	5, 6
w_a (kg m^{-2})	0.8	1.1

[1] − The 10 and 100 hr TL (timelag) fuel moisture contents were assigned values of plus one and two percent points of the value of the 1-hr timelag fuels as per Rothermel (1983).

[2] − After Philpot and Mutch (1971) for ponderosa pine (*Pinus ponderosa*).

[3] − The lowest value of *FSG* was different for the normal summer (4 m) and drought summer (5 m) to ensure that crown fuel ignition would not occur.

Due to the nonlinear nature of the model system the relative, effect of the

parameters being tested were varied with burning conditions and fuel complex structure.

The sensitivity analysis was applied under two broadly defined burning conditions,

namely "Normal summer" and "Summer drought" conditions (Table 1.5) as suggested by

Rothermel (1991) for the Northern Rockies. Given the strong effect of U_{10} a5nd *FSG* on

model behavior, the sensitivity analysis score for these two variables was based on the

average sensitivity score from two baseline values given in Table 1.5 for each burning

condition scenarios. The lowest value of *FSG* was different for the normal summer and

drought summer burning conditions. This ensured that during the simulation crown fuel

ignition would not occur. This was required so that the simulation would proceed until

the end, instead of being stopped due to the crossing of the 600 K ignition temperature threshold (which would then be the maximum temperature attained in the simulation). The results from this analysis might not be true for all burning conditions. Some variability around the sensitivity scores obtained should be expected.

The computed sensitivity scores are plotted in Figs. 1.23a – 1.23d. None of the parameters under analysis had a *RS* larger than 1, meaning that the changes introduced by the variation of the input and intermediate output variables are proportionally smaller. This suggests a balanced model, without any variable having a disproportionate effect on the heat transfer processes and the final model output. *FSG* and w_a are parameters with notable effect on the maximum fuel temperature attained (Fig. 1.23a). *FSG* has a large negative effect due to the dissipation of thermal energy in the plume with height and the reduction on the radiative energy emitted by the flaming surface reaching the canopy fuels. In turn, w_a affects the model output by increasing fireline intensity and reaction time. The increase in fireline intensity will increase the vertical velocity at the tip of the flame and also increase flame height. Increasing reaction time will increase the depth of the flame zone radiating surface and the diameter of the buoyant plume, limiting plume entrainment and consequent dilution of the plume temperature and velocity, variables which determine convective heat transfer.

As found in the previous section, the sensitivity analysis results suggest model insensitivity to *FMC* and σ, the two crown fuel variables (Fig. 1.23a). This might be explained by the magnitude of the heat fluxes produced by the surface fire. The effect of *FMC* and σ in modifying, respectively, the heat requirements for ignition and the surface

82

area available to heat transfer may be insignificant when compared with the amount of energy reaching the fuel particles as a result of being exposed to intense surface fires.
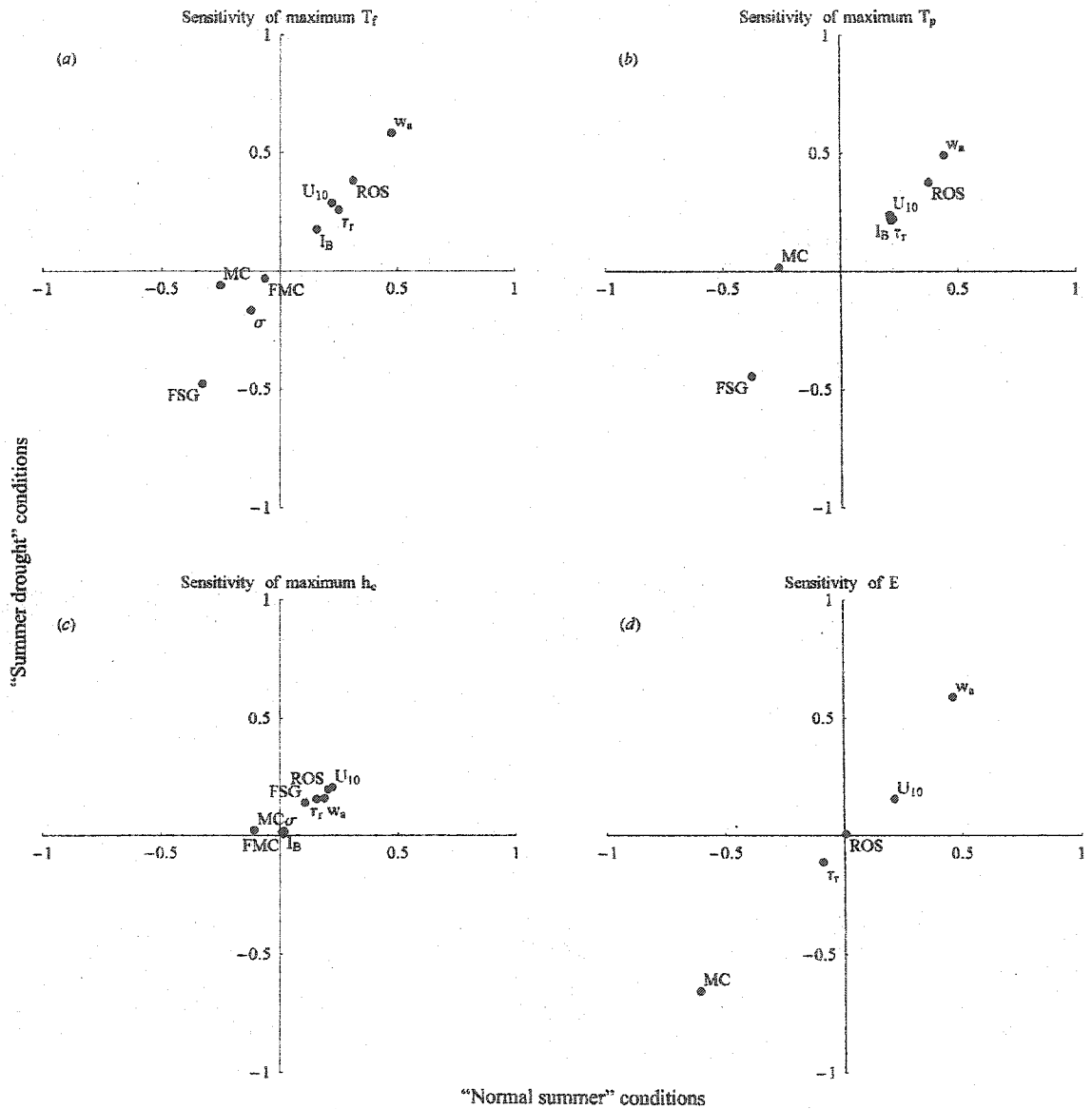
Figure 1.23. Sensitivity analysis of (a) maximum canopy fuel particle temperature (T_f); (b) maximum plume temperature (T_p); (c) maximum convective heat transfer coefficient (h_c); and (d) flame surface emissive power (E), to variability in input/intermediate output parameters under "normal summer" and "summer drought" burning conditions.

The dispersion of most of the sensitivity analysis results along an imaginary 45 degree line in Figs. 1.23a – 1.23d suggests a relatively stable response of the output parameters to changes in the burning conditions. Nevertheless, analysis of Figs. 1.23b and 1.23d confirms that some variables have divergent effects in determining convective and radiative heat fluxes. Changes in reaction time result in an increase in the maximum attained plume temperature and the maximum convective heat transfer coefficient, but a slight decrease in the energy leaving the radiating surface. The effect of MC in the model system is complex. MC showed distinct effects between "normal summer" and "summer drought" conditions. In the "summer drought" conditions the maximum plume temperature is relatively insensitive to MC, whereas in the normal summer conditions MC has a moderately negative effect on the maximum air temperature in the plume (Fig 23b). The effect of an increase in MC on the emissive power of the surface fire is negative, due to its effect in decreasing ROS and T_{Fmax}. Of the output parameters under analysis, the maximum convective heat transfer coefficient (Fig. 1.23c) showed the least sensitivity to changes in the input and intermediate output variables.

1.6. Model evaluation

1.6.1. Comparison with other models

The comparison between models describing the same event provides insight into possible model deficiencies and limits of applicability. In order to better understand the behavior of the CFIM as developed, in this study its behavior was compared with the predictions of other crown fire initiation models, namely those of Van Wagner (1977), Alexander (1998) and Cruz et al. (2004). The nature of the different modeling

84

approaches taken in the development of these models, each with their own distinct input requirements and output form, places constrains on type of comparative analysis that can be undertaken. The output being sought was the threshold for crowning, which is the attainment of: (1) a critical fireline intensity in the Van Wagner (1977) and Alexander (1998) models; (2) a probability of crown fire occurrence of 0.5 or higher in the Cruz et al. (2004) model; and (3) a canopy fuel particle temperature of 600 K in the CFIM formulation. Model comparison was based on the determination of these critical outputs as a function of the two input variables common to all models, U_{10} and *FSG*. Both the Van Wagner (1977) and Alexander (1998) models are based on convective theory. Both these authors do not specifically consider the effect of radiation in heating canopy fuels, but acknowledge that this heat transport mechanism has an effect on the process of crown fire initiation. By the very nature of their model formulation, it is possible that the effect of radiation is integrated into the proportionality constants used in the models. To better understand the effect of convective heat transfer on crown fire initiation, the CFIM was also applied in this model comparison exercise by blocking the radiative heat transfer component.

The simulations were conducted under the baseline conditions specified in Table 1.5 for the sensitivity analysis. The surface fire rate of spread, which is required to run the present model system and in calculating fireline intensity, was estimated by the BEHAVE system using the previously described red pine plantation custom fuel model (RPFM). An additional variable that was required to know to proceed with the model comparison was total surface fuel consumption (*SFC*). The logistic model requires the categorical description of this variable through its classification into three broad classes

85

(SFC $\leq$ 1.0 kg m^{-2}; 1.0 < SFC < 2.0 kg m^{-2}; SFC $\geq$ 3.0 kg m^{-2}). For the model

comparison exercise SFC was assumed to be between 1.0 and 2.0 kg m^{-2}.
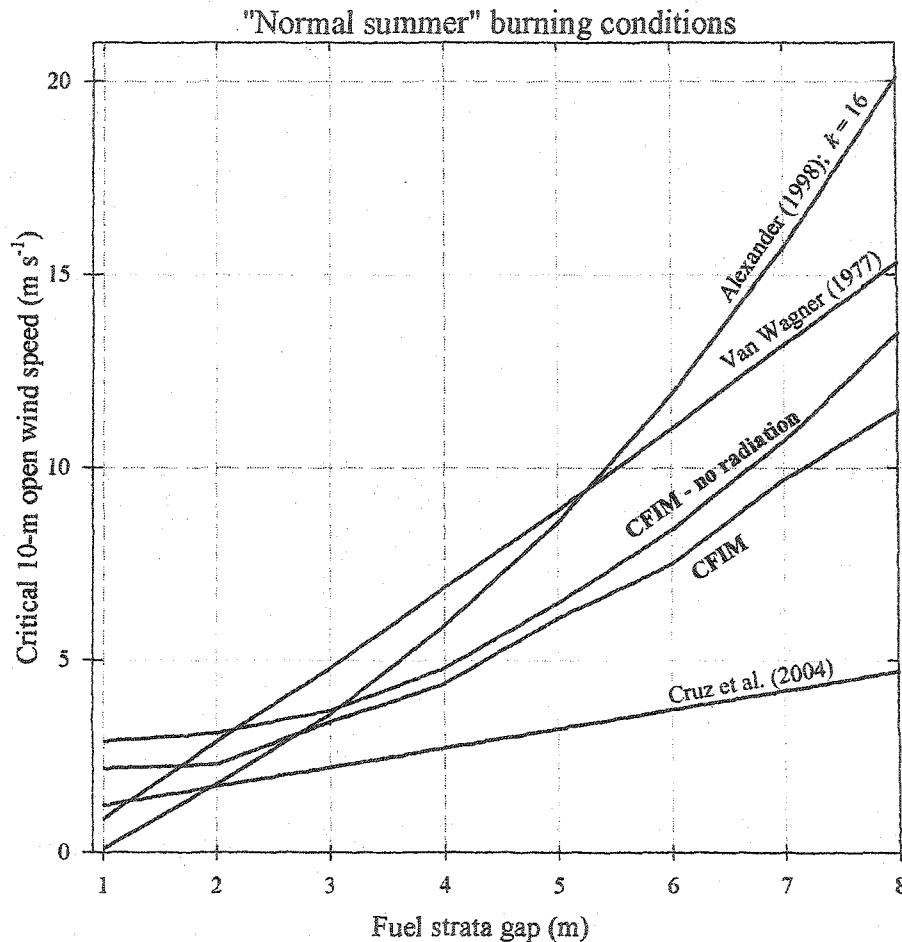
Figure 1.24. Critical 10-m open wind speed for crown fire initiation as a function of fuel
strata gap for Van Wagner (1977), Alexander (1998), Cruz et al. (2004) and CFIM.
Fixed environment conditions are as follows: MC - 0.06; FMC - 1.1; w_a - 0.8 kg m^{-2};
SH - 13 m.

The various models compared showed distinct behavior (Fig. 1.24 and 1.25). For

both "normal summer" and "summer drought" burning conditions the CFIM yield results

between the Cruz et al. (2004) model and the Van Wagner (1977) and Alexander (1998)

models for moderate to high *FSG* values. For low *FSG* values (i.e., < 3 m) the CFIM had slightly higher wind requirements for crowning. The use of the CFIM without the radiation component did not result in large changes in the predicted thresholds for crown fire initiation compared to the complete CFIM for the conditions simulated. The largest differences were attained at the higher *FSG* values where the buoyant plume temperatures are lower, and under burning conditions that lead to the development of deep flame fronts. The difference between the CFIM and CFIM – no radiation curves suggest that, when considering the energy requirements to lead a canopy fuel particle to ignition temperature, the convective component dominates over the radiative.

The Alexander (1998) model tends to be less conservative than the Van Wagner (1977) model for low *FSG* values (Fig. 1.24 and 1.25), while for larger *FSG*s the Alexander's is the most conservative, i.e., requires more severe burning conditions to attain crowning. The logistic model of Cruz et al. (2004) exhibits the least wind requirements for the occurrence of crown fire activity.

The changes in model responses between the "normal summer" and "summer drought" burning conditions were substantial. For the "summer drought" conditions all models required lower 10-m open wind speeds to attain the requirements for crown fire initiation (Fig. 1.25). The model that showed the most changes was the CFIM model followed by the Van Wagner (1977) and Alexander (1998) models. The Cruz et al. (2004) model again showed the lowest 10-m open wind speed requirements for crown fire occurrence, albeit its changes between the "normal summer" and "summer drought" situations were comparatively small in comparison to the other three models.
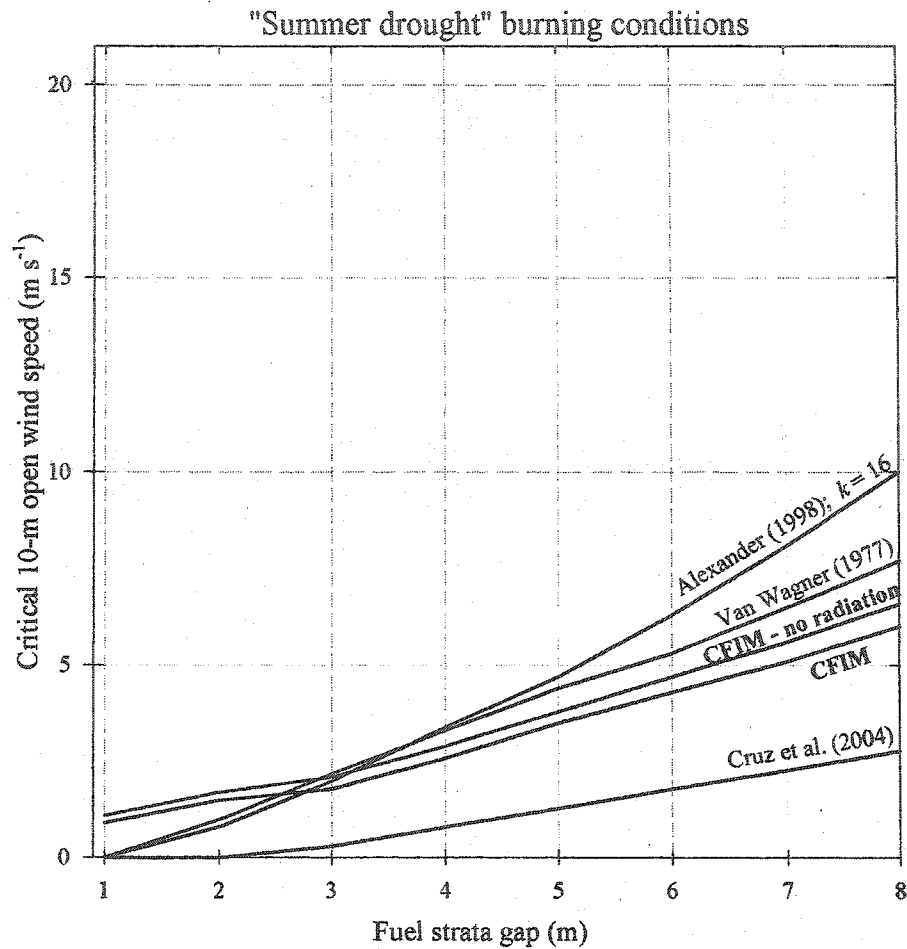
Figure 1.25. Critical open wind speed for crown fire initiation as a function of fuel strata gap for Van Wagner (1977), Alexander (1998), Cruz et al. (2004) and CFIM. Fixed environment conditions are as follows: MC - 0.03; FMC - 0.85; w_a - 1.1 kg m^{-2}; SH - 13 m.

The lower MC characteristic of the "summer drought" resulted in higher predicted surface fire rates of spread, which in turn affected the Van Wagner (1977) and Alexander (1998) model predictions through the attainment of the critical fireline intensity level required for crowning. The increase in the predicted surface fire rate of spread also affects the CFIM model results by increasing in the depth of the flaming zone. Nevertheless, the large differences in the critical 10-m open wind speed between "normal summer" and "summer drought" conditions found for the CFIM model seems to arise

88

from the influence that the amount of fuel available for flaming combustion has in determining both the convective and radiative heat fluxes in the model. The increase in fuel available for flaming combustion, which is expected to occur throughout the burning season in response to gradual drying (Williams and Rothermel 1992, Rothermel 1994), affects the CFIM by increasing the dimensions of the flaming zone, which in turn positively affects the radiative heat transfer to the canopy and the depth and integrity of the buoyant plume.

It should be noted that although the model comparison exercise presented here gives some idea of general model behavior it was not intended to be a comprehensive comparison nor to provide absolute differences in model behavior. Given the distinct model forms and sensitivities to input parameters, model results could compare quite differently for different burning conditions. There are innumerable combinations of input variables that would result in distinct differences between model outcomes. For example, both the Van Wagner (1977) and Alexander (1998) models are sensitive to variability in FMC, whereas the CFIM and Cruz et al. (2004) model are not. Similarly, the application of CFIM to dense stands characterized by strong drag (higher wind attenuation coefficients) and higher radiation opacity coefficients, would induce higher open wind requirements for crowning, whereas the response of the Cruz et al. (2004) model would remain unaltered.

Another model characteristic that is interesting to compare is the sensitivity to input parameters as done in Section 1.5.2. Cruz et al. (2003a) applied the same sensitivity analysis test to the Van Wagner (1977), Alexander (1998) and Cruz et al. (2004) models. The computed RS scores were invariably higher for these models than

89

the ones found in the present study. This might be the result of the lower number of input variables determining the model outputs. This creates a situation where the changes in the output are concentrated in a few input variables. The Cruz et al. (2004) model showed the largest magnitude and variation in the computed RS values. The logistic model form in which the response probability density function is characterized by a large slope around the middle of the probability scale (i.e., 0.5) is the cause of this high sensitivity to input variables. When predicting outcomes near the inflection point, small changes in the input result in large changes in the predicted probability of crown fire occurrence. Conversely, if the predictions are made for the asymptotic component of the probability density function, changes in the input variable invariably produce marginal changes in the outcome. Regardless of the input variable in question, RS scores for Cruz et al. (2004) varied from around 2.5 near the inflection point to 0.1 in the asymptotic region of the probability density function curve. The Alexander (1998) model being the most complex of the three, had the lowest computed RS values. As for common input variables, both Van Wagner (1977) and Alexander (1998) showed substantial higher sensitivity to changes in the FSG and FMC than the ones computed for CFIM. The Van Wagner (1977) model produced a RS of 1.5 and 1.3 for FSG and FMC respectively. The application of the sensitivity test to Alexander (1998) resulted in RS values of: 1.5 for FSG, between 1 and 0.4 for FMC, and between -0.5 and -1.1 for τ_r.

90

1.6.2. Evaluation against experimental fire data

The CFIM is best viewed as a model system that integrates several sub-models to predict the fire behavior characteristics that determine the surface fire upward heat fluxes and consequently the likelihood of crown fuel ignition. The effect of the most relevant sub-models was analyzed in the previous sections. Given the complexity of the model system, with a number of endogenous variables being estimated within the system, an obvious question is how well the model predicts crowning potential of real world fires. An evaluation of the buoyant plume model results and the radiative heat flux model are presented in Chapter 2. For the evaluation of the model system as a whole, data obtained from outdoor experimental fires was used. The selected fires had a complete description of the fuel complex and associated burning conditions. This allowed the application of CFIM without the need to estimate important input variables, which would have only increased the uncertainty in the results. The experimental fires selected were moderate-to high-intensity surface fires, with some of them exhibiting a limited degree of candling or torching. No low-intensity surface fires or crown fires burning under extreme fire weather conditions were selected. This condition was imposed to avoid redundancy in the analysis.

Table 1.6 lists the various fires used in the present analysis and their sources. All fires were from pine stands with a well defined gap between the surface and canopy fuel layers. In the absence of a reliable method to estimate *ROS*, for all but the red pine plantation fires (Van Wagner 1968), the observed rate of spread was used as the surface fire rate of spread in the model system. This ensured that the rate of spread prediction would not introduce error into the analysis. The rate of spread for the red pine plantation

91

fires was predicted through the BEHAVE system with a calibrated custom fuel model (Cruz et al. 2004). All the remaining fires with some degree of crowning would be described as passive crown fires as per Van Wagner (1977). This suggests that the surface phase was controlling the fire's rate of spread, and that the use of the observed rate of spread would not introduce any substantial errors in the output for the model system. Within this modeling exercise, the wind adjustment factor used in the calculations was one that would fit the observed wind profile within each stand. This was done by solving a system incorporating eqs. [1.10] and [1.11] for α_U from the knowledge of the 10-m open and within stand wind speeds.

Table 1.6. The experimental fire data used in the evaluation of the CFIM.

Fire name	Stand type	T_a (°C)	RH (%)	U_{10} (m s^{-1})	U_s (m s^{-1})	MC (%)	FMC (%)	w_a (kg m^{-2})	SH (m)	δ (m)	FSG (m)	ROS (m s^{-1})	I_B (kW m^{-1})	Crowning activity (Y/N)	Source[a]
VW67_R3	Red pine	--	--	3.61	1.65	9	92	0.9	13	0.08	6.92	0.1	2456	N	[1]
VW67_R4	Red pine	--	--	3.06	0.89	13	100	0.9	13	0.08	6.92	0.025	457	N	[1]
VW67_R5	Red pine	--	--	1.67	0.76	4	108	0.9	13	0.08	6.92	0.034	899	N	[1]
VW67_R1	Red pine	24	26	4.16	1.38	10	100	0.9	13	0.08	6.92	0.18	7300	Y	[1]
BW&B_P1	Maritime pine	21	37	5.56	0.89	9[b]	120	1.2	14	--	2.4	0.05	1104	Y	[2]
BW&B_P2	Maritime pine	23	33	6.11	0.94	9[b]	120	1.21	14	--	2.4	0.0556	1237	Y	[2]
BW&B_P3	Maritime pine	25	30	6.67	0.81	9[b]	120	1.18	14	--	2.4	0.0439	953	Y	[2]
McA66	Radiata pine	21	33	4.5	2.2	12.6	145	1.5	18	--	10	0.066	3875	N	[3], [4]
VL&L_A2	Slash pine	30.5	50	3.75	1.03	10.2	109	0.87	6.5	--	1.8	0.038	1104	Y	[5]
VL&L_A4	Slash pine	28	71	0.97	0.31	27.8	109	0.76	6.5	--	1.8	0.017	1237	N	[5]
VL&L_C2	Slash pine	30	64	3.75	1.03	17.2	109	0.53	7.7	--	1.8	0.015	953	N	[5]
PF&al_Un	Maritime pine	29	25	5.7	1.89	5	116	1.63	9.1	0.52	4.2	0.060	4925	Y	[6]
PF&al_RX13	Maritime pine	29	25	3.2	1.06	5	116	1.54	8.5	0.50	3.5	0.032	1520	Y	[6]
PF&al_RX3	Maritime pine	29	25	4.5	1.47	5	116	0.69	10.1	0.31	5.1	0.043	931	N	[6]

[a] [1] - Van Wagner (1968); [2] - Burrows et al. (1988); [3] - McArthur (1966); [4] - Nicholls and Cheney (1974); [5] - Van Loon and Love (1973); [6] – Fernandes et al. (*in press*).
[b] estimated as per Rothermel (1983).

93

Table 1.7. Results of CFIM for intermediate model outputs and canopy base fuel temperature for the experimental fires used in the evaluation exercise.

Fire name	H_F (m)	FSG - H_F (m)	D_F (m)	t_r (s)	b_{pi} (m)	U_{pi} (m s^{-1})	Max. T_f (K)	CFIM (Y/N)	Crowning activity CEVW (1977) (Y/N)	MEA (1998) (Y/N)
VW67_R3	2.3	4.7	3.4	66	1.7	3.3	547	N	N	N
VW67_R4	1.1	5.9	2.3	65	1.1	2.8	392	N	N	N
VW67_R5	1.3	5.7	2.9	51	1.4	3.1	415	N	N	N
VW67_R1	2.9	4.1	6.2	57	3.1	3.9	>600	Y	N*	N*
BW&B_P1	1.8	0.6	3.8	77	1.9	3.4	>600	Y	Y	Y
BW&B_P2	1.9	0.5	4.3	77	2.2	3.5	>600	Y	Y	Y
BW&B_P3	1.7	0.7	3.3	75	1.6	3.2	>600	Y	Y	Y
McA66	1.9	8.1	6.9	104	3.4	4	490	N	N	N
VL&L_A2	1.4	0.4	2.1	55	1	2.8	>600	Y	Y	Y
VL&L_A4	0.9	0.9	0.88	52	0.44	2	492	N	Y*	Y*
VL&L_C2	0.3	1.5	0.53	53	0.26	1.7	398	N	Y*	Y*
PF&al_UN	2.1	2.6	5.9	98	2.9	4	>600	Y	Y	Y
PF&al_RX13	1.5	2.5	2.9	92	1.5	3.1	600	Y	N*	N*
PF&al_RX3	0.83	4.6	1.7	39	0.84	2.7	416	N	N	N

* indicates fires incorrectly predicted.

Van Wagner (1968) published the results of 9 experimental fires in red pine (*Pinus resinosa*) plantations. Four of those fires were used in the present analysis (Table 1.6), three of them spreading as surface fires and the fourth spread as a crown fire for less than a minute. Based on the surface fuel layer description given by Van Wagner (1968), w_a was assumed as 0.9 kg m^{-2}. This value integrates the litter layer and a fraction (15 %) of the duff layer that was assumed to burn within the flaming phase of the fire front. The four red pine fires were correctly predicted by the CFIM system (Table 1.7). For the three surface fires the CFIM predicted maximum canopy fuel temperatures between 392 K for fire R4 and 547 K for fire R3.

Alexander (1998) assembled the data of a series of publications describing various prescribed and experimental fires in maritime pine (*Pinus pinaster*) (Burrows et al 1988), radiata pine (*Pinus radiata*) (McArthur 1966, Nicholls and Cheney 1974) and slash pine (*Pinus elliotii*) (Van Loon and Love 1973) plantations. These fires were comprehensively described and are an excellent source of data to evaluate a model such as the CFIM. The three operational prescribed fires reported in Burrows et al. (1988) were described as having "...short bursts of crown fire activity..." and being "...just below the threshold for sustaining crown fires" (Alexander 1998, page 142). The CFIM predicted that ignition of the canopy fuels would occur for the three firs (Table 1.7). Van Loon and Love (1973) describe the fire behavior associated with eight prescribed fires in a young slash pine plantation, three of which spread as head fires (Table 1.6). Plot A2 was described as exhibiting localized crown fire activity, whereas the other fires spread as moderate-intensity surface fires. CFIM predicted the ignition of canopy fuels for fire

A2 (Table 1.7). Simulations for plots A4 and C2 predict canopy fuel temperatures of 492 and 398 K respectively.

Fernandes et al. (*in press*) report on an experimental fire in a 28-year old *Pinus pinaster* block consisting on four distinct fuel complex situations: a plot prescribed burned 13 years before the experiment (RX13), an untreated plot (UN), and two plots prescribed burned 3 and 2 years before (RX3 and RX2). The experimental fire was accomplished by igniting one side of the block and let the fire burning successively through the RX13, UN, RX3 and RX2 portions of the block. Both plots RX13 and UN exhibit crowning activity, with 37 and 100% of canopy fuel consumption, respectively. Both those fires were described as burning as passive crown fires, with the ignition of canopy fuels occurring some meters behind the leading edge of the surface fire flame front. RX2 was a low intensity surface fire and was not used in the analysis.

The CFIM predicted the ignition of canopy fuels for plot UN and RX13, with the maximum T_f predicted for plot RX13 being 599.9 K. This result was interesting from the standpoint that the CFIM barely estimated the ignition of canopy fuels on a fire characterized by the consumption of one third of the canopy fuel stratum. The fuel temperature prediction trace (Fig.1.26.a) for this fire qualitatively describes the observed behavior, with crown ignition occurring when the plume temperatures are already decreasing. Fig 1.26.b details the predicted convective and radiative heat transfer to the canopy fuels for fire RX13. The canopy fuels are radiatively preheated before the arrival of the buoyant plume. A rapid increase in fuel particle temperature occurs while the fuel particle is immersed in the plume (Fig. 1.26.a) after which radiative heating continues to

96

raise fuel temperature while the convective component is negative (Fig. 1.26.b). The

CFIM predicted a maximum canopy fuel temperature of 416 K for fire RX3.
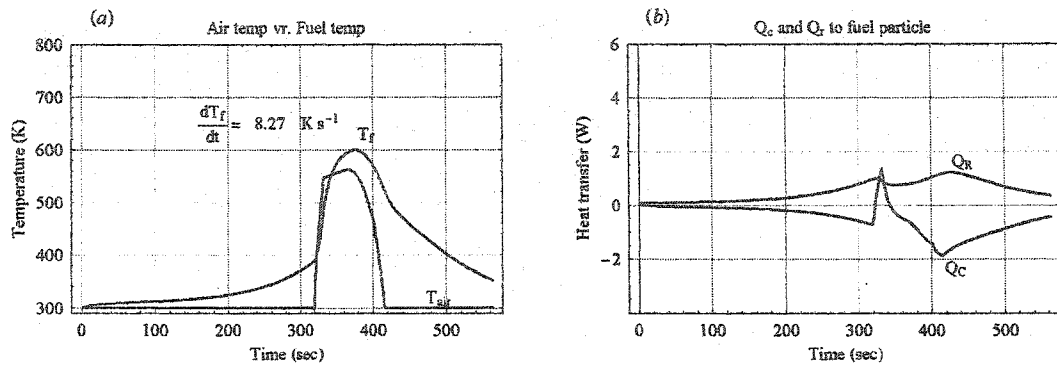
Figure 1.26. CFIM predictions for Fernandes et al. (*in press*) experimental fire RX13: (a) air and canopy fuel temperature profiles; and (b) convective (Q_c) and radiative (Q_r) heat transfer to a fuel particle.

Table 1.8. Classification table comparing observed and predicted type of fire through the application of CFIM, Van Wagner (1977) and Alexander (1998) models to experimental fires detailed in Table 1.6.

		Predicted		Correctly predicted (%)
		Surface fire	Crown fire	
		CFIM		
Observed	Surface fire	7	0	100
	Crown fire	0	7	100
		Van Wagner (1977)		
Observed	Surface fire	5	2	71
	Crown fire	2	5	71
		Alexander (1998)		
Observed	Surface fire	5	2	71
	Crown fire	2	5	71

97

Overall, the CFIM system correctly predicted all 14 experimental fires selected for evaluation. For comparison purposes both the Van Wagner (1977) and Alexander (1998) crown fire initiation models were also applied to the experimental fire dataset. For the application of both models, fireline intensity was estimated from total surface fuel consumption (Table 1.6) as the proportionality constants used in those two models were derived from this quantity. This can be seen as both a theoretical and practical limitation of the models. By using the total surface fuel consumed we are considering fuels that were consumed in residual combustion, not contributing to the main heat pulse associated with flaming combustion. From a practical point of view, there will be also a need to estimate the total fuel consumed by the fire, which can introduce further uncertainty in the final result. The application of Alexander (1998) crown fire initiation model requires the use of a proportionality constant dependent on surface fuelbed structure, and eventually burning conditions. Following the analysis of Alexander (1998), a proportionality constant of 9 (presence of dense understory vegetation) was used for the Fernandes et al. (in review) fires, and 16 (needle fuelbed) for the remaining fires.

The Van Wagner (1977) and Alexander (1998) crown fire initiation models predicted similar outcomes for the dataset. Both models predicted crowning in two of the surface fires and failed to indicate crowning in two fires where the ignition of canopy fuels occurred (Tables 1.7 and 1.8).

1.7. Concluding remarks and future work

Either at a personal or institutional level, we can find several reasons to devote resources to forest fire modeling. Ultimately, the main purpose is to develop tools that can be used to help land managers to achieve their management objectives (e.g., fight fire safely, mitigate fire hazard at a stand and landscape levels, incorporate fire processes in ecosystem dynamics). The present study described the reasoning behind the structure of a model aimed at predicting the ignition of crown fuels above a spreading surface fire. The model was developed with the objective of providing a better understanding of the variables and processes controlling the initiation of crown fires. In addition, model development was carried out with the initiative that its design allow its use as both a research and fire management decision support tool.

The crown fuel ignition model (CFIM) quantifies the upward heat fluxes originating from a spreading surface fire and in turn calculates both the convective and radiative heat transfer to fuel particles located at the base of the canopy fuel layer. CFIM can be characterized as a hybrid model that combines fundamental heat transfer processes with empirically derived parameters. CFIM simplifies the description of certain sub-model components, falling short of describing important fire phenomenology, such as reaction zone processes and flame dynamics. Important flame front parameters need as intermediate outputs such as reaction zone temperature-time profile and flame height were obtained from simple models. The detailed description of such processes falls in the realm of more sophisticated models solving the conservation equations for the solid and gas phases.

99

The CFIM as a model system and its individual components were subjected to a thorough evaluation. Model results suggest that the onset of crowning is dependent more heavily on the mechanisms that determine the surface fire characteristics, namely reaction time, flame depth and rate of energy release rather than on the physical characteristics of the canopy layer. Sensitivity analysis results suggest that the CFIM is well balanced. Over the natural range of their variability, no variable was found to have an overwhelming effect on sub-model components or on the final model output. The CFIM was compared with other models used to predict crown fire initiation. A comparison between existing crown fire initiation models and the CFIM was based on the determination of critical open wind speed – fuel strata gap thresholds for crowning under two broadly defined burning conditions. For the burning conditions simulated, the CFIM tended to predict crowning under less severe burning conditions than the Van Wagner (1977) and Alexander (1998) crown fire initiation models. When compared with the logistic crown fire occurrence model developed by Cruz et al. (2004), the CFIM required higher wind speeds/lower fuel strata gaps in order to predict ignition of crown fuels for the same burning conditions.

Model evaluation against an independent dataset from experimental fires provided encouraging results and gave insight into some limitations of the model system, namely the difficulty of correctly estimating some input variables. The CFIM was applied to 14 experimental fire situations that had a good description of the fuel complex, fire weather conditions and fire behavior characteristics. The model correctly predicted all fires, seven surface and seven passive crown fires, with respect to the ignition of crown fuels.

The main difficulties in application of the CFIM against the independent experimental fire dataset resulted from the need to accurately estimate the available surface fuel for flaming combustion and to adequately describe the vertical wind profile. Surface fuelbeds are a complex array of live and dead fuels of differing size classes, displaying innumerable possible arrangements determined by compactness and relative proportions of the individual fuel particles. The physical structure of the surface fuelbed and associated burning conditions will largely determine the amount of fuel consumed in the flaming combustion phase. No objective method to estimate this quantity currently exists (Rothermel 1994). In the application of the CFIM to the experimental fire situations it was assumed for most of the fires that only the fine fuels, either live or dead, were consumed during the flaming combustion stage. Evidence from outdoor experimental fires (e.g., Van Wagner 1968) and laboratory fires (Chapter 2) suggests that such assumption is not necessarily true, and substantial errors can be introduced as a result.

It was not possible to model the vertical wind profile on *a priori* basis for the experimental fires selected for evaluation purposes, and the simulations were based on fitting the wind profile model to the wind measurement pairs existent for each fire (i.e., U_{10} and U_S). The difficulty in accurately estimating the wind profile could result not only from the distinct stand structures and the inherent variability in the decay parameter quantifying within stand wind flow, but also from the existence of cleared fireguards around some of the experimental plots, which undoubtedly affected wind flow within the experimental plot.

In the present study we did not apply CFIM under sloping terrain conditions. It is expected that under low to moderate slope conditions the current model implementation, although not perfect from a theoretical point of view, should provide reasonable results. In steep slopes several factors, such as, plume interaction with the ground and geometrical considerations of the implementation of the CFIM in a coordinate system, might induce substantial differences between the predicted behavior and reality. To our knowledge no data is available to analyze such a hypothesis. Possible insight into these effects related to plume behavior on steep slopes can nevertheless be obtained through computational fluid dynamics. The implementation of the CFIM under slope conditions will also require an appropriate adjustment of fuel strata gap (Alexander 1998).

The evaluation of model behavior carried out in the present study hints at the adequacy of the CFIM to be used as a viable tool in answering several fire management questions. It is believed that the overall CFIM structure, incorporating important flame front phenomena and their interactions allows a better description of the processes determining crown fire initiation than found in previous developed empirical-based crown fire initiation models. This gives the CFIM the potential to be used to answer not only important fire management questions but also to be applied as a fire research tool. The balance between empiricism and fundamental heat transfer formulations allows one to gain insights into the influence of certain fire environment variables and heat transfer processes on crown fire initiation.

The CFIM system could be applied to problems related to the implications of fuel treatments and silvicultural operations in determining the resultant fire behavior potential. By integrating a large number of processes, the model is likely to individualize the effect

of management options in altering outcomes, either as single processes or as a whole. An example of this would be the analysis of stand treatments (e.g., distinctly different thinning methods) and the consequent changes in fuel complex characteristics and micrometeorology processes, on the susceptibility of the stand to initiate crown fire activity. By its structure, the CFIM is expected to take into account many of the changes in the fire environment and in turn the resultant fire behavior induced by the treatment (e.g., higher within stand wind speed, increase or decrease in surface fire reaction time, changes in the buoyant plume characteristics) and thereby provide for an adequate description of the post-treatment potential for crown fire activity.

Although the results from the evaluation exercise are encouraging, the possible use of the model to predict fire behavior to support operational fire suppression activities should be preceded by additional evaluation of the model system and familiarity of users to model's structure, main underlying assumptions, and limitations. Additional evaluation should focus on the applicability of the model system to specific fuel types. For example, an important question to answer is how particular surface fuel beds and burning conditions, as determined by fuel moisture variation by fuel particle type or layer, determine the surface fuel available for combustion in the active flame front, and consequently the reaction time. The range of possible surface fuelbed structures–burning conditions is broad, and decisions relative to the best estimates of surface fuel available for flaming combustion should be complemented by the expert opinion of knowledgeable users with extensive operational experience in the particular fuel type of interest.

The CFIM does not predict the occurrence of crown fires *per se*. The model is solely aimed at predicting the ignition of crown fuels. The forward propagation of crown

103

fires is controlled in part by mechanisms other than the ones considered in the CFIM, such as forward radiation from the reaction and free flame zones (Van Wagner 1968, 1977, Albini and Stocks 1986). The coupling of the CFIM with models for crown fire spread (e.g., Albini 1996, Butler et al. 2004, Cruz et al. *in review*) could provide information regarding the potential spread of crown fires, namely for passive crown fires which depend to a certain extent on the surface fire heat source (Van Wagner 1977).

The model system described in the present work relies on the integration of several previously published models to describe some of its components. This model system is consequently the result of previous knowledge, assembled by the work of innumerous researchers and technicians, and should be seen as an ongoing effort to best understand the effects of fire environment variables and fire behavior processes in influencing the onset of crowning. In no sense should this modeling effort be considered complete. With the objective of better understanding the onset of crowning as a physical process, two main areas of future research can be individualized. One dealing with the improvement of model components, leading to a better description of the physics of fire. The other area of research is concerned with the application of CFIM to specific situations to advance our understanding of the physical processes determining fire behavior.

The modular structure of the model system allows changes of individual sub-models if found advantageous. There are several aspects of CFIM that would greatly benefit from the implementation of more robust sub-models. An obvious improvement would be the implementation of a superior model for the estimation of flame height. This

model could possibly be derived from solving a simplified formulation of the conservation equations for solid and gaseous fuels. Such an approach could also determine the amount of fuel available for flaming combustion from the knowledge of the surface fuelbed structure. These two aspects are felt to be the greatest weakness in the current form of the CFIM. Nevertheless, it is not known if the substantial increases in computation requirements due to the implementation of such models would provide a significant improvement in the accuracy of the model output. Such a modeling approach would also provide a better description of the residual burning after the passage of the flaming combustion phase, and consequently the characterization of a secondary heat pulse following the passage of the active flaming front. The coupling of such a detailed characterization of the heat source with a solution of the conservation equations for the buoyant plume would allow a more realistic calculation of the plume properties, namely the description of complex fire-atmosphere interactions.

Incorporate improvements in the individual sub-models would also enable the CFIM to serve as a better tool to conduct research into processes controlling crown fire initiation in relation to changes in the fire environment and in turn fire behavior. An interesting research application of the CFIM would be to investigate the implications of simulating the onset of crowning as a transient process instead of the steady state simulation as applied in the present study. The constant variation in wind flow, characterized by the occurrence of peaks and lulls, and the non-homogeneity of the spatial distribution of surface fuels results in a cyclic pulsing pattern in the active combustion zone (Albini 1982, Alexander 1998). This unsteady behavior results in a complex pattern of upward heat fluxes, where increases in flame depth, due to short-term

105

increases in rate of spread or reaction time, result in convective pulses (Gould et al. 1997) in combination with peaks in upward radiative heat fluxes. These two processes might or not be coincident. From the stand point of the ignition of crown fuels, where the interest is focused on the occurrence, or not, of the event instead of its average behavior, the modeling of such non-transient phenomena should be more appropriate than the current steady-state approach. A further refinement of the processes controlling crown fire initiation would be to integrate such transient formulation of the CFIM with a statistical description of fuel strata gap. This method could possibly be used to analyze the results through a probabilistic approach.

List of symbols, quantities and units used in equations and text

A_f	Fuel particle area (m^2)
A_F	Free flame tilt angle from vertical (°)
AT_{600}	Nondimensional parameter describing location of ignition interface
σ_{SB}	Stefan-Boltzmann constant, $5.67 \cdot 10^{-8}$ W m^{-2} K^{-4}
b	Plume half width (m)
c^*	Average specific heat of fuel (kJ kg^{-1} K^{-1})
CBH	Canopy base height (m)
c_f	Specific heat of fuel particles (kJ kg^{-1} K^{-1})
c_p	Specific heat of air (kJ kg^{-1} K^{-1})
c_v	Mean specific heat of volatiles (kJ kg^{-1} K^{-1})
c_w	Specific heat of water (kJ kg^{-1} K^{-1})
d	Fuel element diameter (m)
D_F	Flame depth (m)
dbh	Tree diameter at breast height (m)
E	Radiant heat flux (kW m^{-2})
$EFFM$	Estimated fine fuel moisture (% ovendry weight)
F_{12}	Geometrical view factor
FMC	Foliar moisture content (fraction of ovendry weight)
FSG	Fuel strata gap (m)
g	Acceleration of gravity (m s^{-2})
h_c	Fuel particle convective heat transfer coefficient (kJ m^{-2} s^{-1} K^{-1})
H_c	Fuel low heat of combustion (kJ kg^{-1})
H_F	Flame height (m)
H_N	Heat to desiccate and decompose unit fuel mass (kJ kg^{-1})
h_{ef}	Fuel particle effective heat transfer coefficient (kJ m^{-2} s^{-1} K^{-1})
h_r	Fuel particle radiative heat transfer coefficient (kJ m^{-2} s^{-1} K^{-1})
I_B	Fireline intensity (kW m^{-1})
k_F	Constant of proportionality for flame height model (m^3 kJ^{-1})
k	thermal conductivity of the gas at film temperature (kJ m^{-2} s^{-1} K^{-1})
L	Latent heat of vaporization of water (kJ kg^{-1} K^{-1})
MC	Moisture content of fine dead fuel particles (fraction of ovendry weight)
mf	Mineral fraction content
N_p	Plume edge criterion
N_u	Nusselt number
N_v	Stoichiometric air/fuel mass ratio of volatiles
P_r	Prandtl number
Q	Internal energy of fuel particle (kJ m^{-3})
q''	heat transferred to the fuel particle (kJ m^{-2} s^{-1})
q_c	Convective heat flux to fuel particle (kJ m^{-2} s^{-1})
Q_f	Heat to raise dry fuel temperature from ambient to sublimation temp (kJ kg^{-1})
q_l	Radiative heat losses from fuel particle (kJ m^{-2} s^{-1})
Q_w	Heat to raise temperature of water in the fuel particle from ambient to

107

	373 K (kJ kg^{-1})
q_r	Radiative heat flux to fuel particle (kJ m^{-2} s^{-1})
r	Coordinate in plume normal to s
r_f	radius of idealized cylindrical fuel particle (m)
ROS	Surface fire rate of spread (m min^{-1})
R_e	Reynolds number
s	Distance along plume centerline (m)
SH	Stand height (m)
SD	Stand density (trees ha^{-1})
T	Temperature (K)
t	Time (sec)
T_a	Ambient temperature (K)
T_F	Flame temperature (K)
T_g	Gas temperature (K)
T_{ig}	Ignition temperature (600 K)
Tm	Reacting mixture mass-average temperature (K)
T_{Fmax}	Flame maximum temperature (K)
Ts	Sublimation temperature (K)
T_s	Fuel particle surface temperature (K)
T_x	Temperature at which production of volatiles ceases (K)
U_F	Free flame velocity (m s^{-1})
U_{10}	10-m open wind speed (m s^{-1})
U_{SH}	Wind speed at the top of the canopy (m s^{-1})
U_z	Wind speed at height z (m s^{-1})
U_s	Within stand wind speed (m s^{-1})
U_{Fh}	Horizontal component of U (m s^{-1})
U_p	Plume velocity (m s^{-1})
V	Mixture velocity in reaction zone (m s^{-1})
v_e	Entrainment speed (m s^{-1})
W_F	Flame front width (m)
w_a	Surface fuel available for flaming combustion (kg m^{-2})
x	Horizontal distance (m)
X_l	Fraction of volatiles that burn
Z	non-reacting air entering the reaction zone (kg kg^{-1})
z	Vertical distance (m)
α_U	Wind attenuation coefficient
γ_F	Flame temperature cooling parameter
λ_F	Flame temperature time adjustment variable
κ_p	Plume spreading ratio
σ	Fuel particle surface/volume ratio (m^{-1})
γ_c	Fuel bed char fraction
β_F	Flame entrainment parameter
β_i	Regression coefficients
β	Fuel bed packing ratio
δ	Fuel bed depth (m)

E	Flame emissivity
θ	Plume trajectory angle from vertical (°)
ν	Kinematic viscosity of reacting mixture ($m^2\ s^{-1}$)
ρ_a	Ambient air mass density ($kg\ m^{-3}$)
ρ_f	fuel mass density ($kg\ m^{-3}$)
ρ_m	Reacting mixture mass density ($kg\ m^{-3}$)
ρ_p	Plume mass density ($kg\ m^{-3}$)
a_U	Wind attenuation coefficient
α_p	entrainment constant
χ_p	entrainment constant
τ_r	Reaction time

List of subscripts

a	ambient
c	convective
F	flame
f	fuel particle
i	initial
p	plume
r	radiative
v	volatiles
w	water

Bibliographic references:

Albini, F.A. 1976. Estimating wildfire behavior and effects. USDA For. Serv. Gen. Tech. Rep. INT-30, Intermountain For. and Range Exp. Stn., Ogden Utah. 92 p.

Albini, F.A. 1980. Thermochemical properties of flame gases from fine wildland fuels. Intermountain Forest and Range Experimental station, USDA For. Ser. Res. Pap. INT-243 Odgen,UT. 42 p.

Albini, F.A. 1981. A model for wind-blown flame from a line fire. Combustion and Flame 43: 155-174.

Albini, F.A. 1982. Response of free burning fires to nonsteady wind. Combust. Sci. and Tech., 29:225-241.

Albini, F.A. 1983. Potential spotting distanced from wind-driven surface fires. USDA For. Serv. Gen. Res. Pap. INT-309, Intermountain For. and Range Exp. Stn., Ogden Utah, 27p.

Albini, F.A. 1984. Wildland fires. American Scientist 72:590-597.

Albini, F.A. 1985. A model for fire spread in wildland fuels by radiation. Combust. Sci. and Tech., 42:229-258.

Albini, F.A. 1996. Iterative solution of the radiation transport equations governing spread of fire in wildland fuel. Fizika Goreniya i Vzriva 32(5):71-82. [in english] translated journal is printed in english as "Physics of Combustion, Explosion and Shock Waves".

Albini, F.A. 1999. Maximum spotting distance from an active crown fire. In the Proceedings of the National Interagency Fire Behavior Workshop. May 1999, Phoenix. 22 p

Albini, F.A., Baughman, R.G. 1979. Estimating windspeeds for predicting wildland fire behavior. USDA For. Serv. Res. Pap. INT-221, Intermountain For. and Range Exp. Stn., Ogden Utah, 12p.

Albini, F.A., Stocks, B. J. 1986. Predicted and observed rates of spread of crown fires in immature jack pine. Combustion Science and Technology 48: 65–76.

110

Albini, F.A., Reinhardt, E.D. 1995. Modeling ignition and burning rate of large woody natural fuels. International journal of Wildland Fire 5(2):81-91.

Alexander, M.E. 1982. Calculating and interpreting forest fire intensities. Can. J. Bot. 60:349-357.

Alexander, M.E. 1988. Help with making crown fire hazard assessments. P. 147-156 *in* Protecting people and homes from wildfire in the Interior West: proc. symp. and workshop. W.C. Fischer and S.F. Arno (comps.). USDA For. Serv. Gen. Tech. Rep. INT-251.

Alexander, M.E. 1998. Crown fire thresholds in exotic pine plantations of Australasia. Ph.D. Thesis, Australian National University, Canberra, Australia. 228 p.

Alexander, M.E. 2000. Fire behaviour as a factor in forest and rural fire suppression. For. Res., Rotorua in association with New Zealand Fire Serv. Comm. and Natl. Rural Fire Authority, Wellington, NZ. For. Res. Bull. No. 197, For. Rural Fire Sci. Technol. Ser. Rep. No. 5. 28 p.

Alexander, M.E., C.N. Stefner, J.A. Mason, B.J. Stocks, G.R. Hartley, M.E. Maffey, B.M. Wotton, S.W. Taylor, N. Lavoie, and G.N. Dalrymple. 2004. Characterizing the jack pine-black spruce fuel complex of the International Crown Fire Modelling Experiment (ICFME). Can. For. Serv. Inf. Rep. NOR-X-[*in prep. for publ.*].

Amiro, B.D. 1990. Comparison of turbulence statistics within three boreal forest canopies. Boundary-Layer Meteorology 51:99-121.

Amiro, B.D., Davis, P.A. 1988. Statistics of atmospheric turbulence within a natural black spruce forest canopy. Boundary-Layer Meteorology 44:267-283.

Anderson, H.E. 1964. Mechanisms of fire spread research progress report No1. USDA Forest Service Research Paper INT-8, Intermountain For. and Range Exp. Station, Ogden, Utah, 20 p.

Anderson, H.E. 1968. Sundance fire: an analysis of fire phenomena. USDA Forest Service Research Paper INT-56, Intermountain For. and Range Exp. Station, Ogden, Utah, 39 p.

Anderson, H.E. 1969. Heat transfer and fire spread. USDA Forest Service Research Paper INT-69, Intermountain For. and Range Exp. Station, Ogden, Utah, 20 p.

Anderson, H.E. 1982. Aids to determining fuel models for estimating fire behavior. USDA For. Serv. Res. Pap. INT-122, Intermountain For. and Range Exp. Stn., Ogden Utah, 22 p.

Andrews, P.L. 1986. BEHAVE: fire behavior prediction and fuel modeling system— BURN subsystem, part 1. Gen. Tech. Rep. INT-260. Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Research Station. 130 p.

Andrews, P.L. Chase, C.H. 1989. BEHAVE: fire behavior prediction and fuel modeling system - BURN subsystem, part 2. USDA For. Serv. Gen. Tech. Rep. INT-194, Intermountain For. and Range Exp. Stn., Ogden Utah. 93 p.

Andrews, P.L., Bevins, C.D. 1999. BEHAVE fire modeling system: Redesign and expansion. Fire Manage. Notes 59(2): 16-19.

Bailey, A.W., Anderson, M.L. 1980. Fire temperatures in grass, shrub and aspen forest communities of Central Alberta. J. Range Management 33(1):37-40.

Bartlink, H.H. 1998. A model of dry matter partitioning in trees. Tree Physiology. 18: 91–101.

Beer, T. 1991. The interaction of wind and fire. Boundary-Layer Meteorol 54:287-308.

Bidwell, T.G., Engle, D.M. 1991. Behavior of headfires and backfires on tallgrass prairie. Pages 344-350 in S.C. Nodvin and T.A. Waldrop (eds.). Proceedings of the international symposium on Fire and the Environment: Ecological and Cultutral perspectives. USDA, Forest Service GTR-SE-69. Southeast Forest and Range Exp. Station, Ashvile, SC. 13 p.

Brown, J.K., Oberheu, R.D., Johnston, C.M. 1982. Handbook for inventorying surface fuels and biomass in the interior west. USDA For. Serv. Res. Pap. INT-129, Intermountain For. and Range Exp. Stn., Ogden Utah, 48 p.

112

Burgan, R.E., Rothermel, R.C. 1984. BEHAVE: Fire behavior prediction and fuel modeling system -- FUEL subsystem. USDA For. Serv. Res. Pap. INT-167, Intermountain For. and Range Exp. Stn., Ogden Utah, 126 p.

Burrows, N.D., Smith, R.H., Robinson, A.D. 1988. Prescribed burning slash fuels in *Pinus radiata* plantations in Western Australia. West. Australia Dep. Conser. Land Manag., Perth, Western Australia, Tech. Rep. 20. 12 p.

Butler, B.W. 1994. Experimental measurements of radiant heat fluxes from simulated wildfire flames. Pages 104-111 in Proceedings of 12th Conference on Fire and Forest Meteorology, Jekyll Island, Georgia. Soc. Am. For., Bethesda, Maryland. SAF Publ. 94-02.

Butler, B.W. 2003. Field measurements of radiant energy transfer in fuel scale wind driven crown fires. 6th ASME-JSME Thermal Engineering Joint Conference, March 16-20.

Butler, B.W. Cohen, J.D. 1998. Firefighter safety zones: a theoretical model based on radiative heating. Int. J. Wildland Fire 8(2):73-77.

Butler, B.W., Finney, M.A., Andrews, P.L., Albini, F.A. 2004. Characterization of a Radiation-Driven Model for Crown Fire Spread. Can J. For. Res. In review.

Byram, G.M. 1948. Vegetation temperature and fire damage in the southern pines. Fire Control Notes 9(4):34-36.

Byram, G.M. 1958. Some basic thermal processes controlling the effects of fire on living vegetation. USDA For. Serv. , Southeastern For. Exp. Stn., Asheville, North Carolina, Res. Note 114. 2p.

Byram, G.M. 1959. Combustion of forest fuels. In: Davis, K. P., ed. Forest fire: control and use. New York: McGraw Hill Book Co.: 61–89.

Byram, G.M., Nelson, R.M. 1952. Lethal temperatures and fire injury. USDA For. Serv. , Southeastern For. Exp. Stn., Asheville, North Carolina, Res. Note 1. 2 p.

Call, P.T. 1997. Run-time optimization of a radiation driven crown fire model. Montana State University, Bozeman, M.Sc. Thesis 86 p.

113

Campbell, G.S., Norman, J.M. 1998. An introduction to environment biophysics. Second Edition. Springer. 286 pp.

Campbell, G.S., Jungbauer, J.D., Bidlake, W.R., Hungerford, R.D., 1994. Predicting the effect of temperature on soil thermal conductivity. Soil Science 158:307-313.

Canadian Interagency Forest Fire Centre 1996. Wildland Fire Behavior Specialist Course, Hinton, Alberta, Canada.

Catchpole, E.A., de Mestre, N. 1986. Physical models for a spreading line fire. Aust. For. 49(2):102-111.

Catchpole, W.R., Catchpole E.A. 2000 The second generation U.S. fire spread model. US Forest Service – ADFA Joint Research Venture #RMRS-94962-RJVA final report.

Catchpole, E.A., Catchpole, W.R., Rothermel, R.C., 1993. Fire behavior experiments in mixed fuel Complexes. Int. J. Wildland Fire 3(1):45-57.

Catchpole, W.R., Catchpole, E.A., Butler, B.W., Rothermel, R.C., Morris, G.A., Latham, D.J. 1998. Rate of spread of free-burning fires in woody fuels in a wind tunnel. Comb. Sci. and Tech. 131:1-37.

Catchpole, W.R., Catchpole, E.A., Tate, A.G., Butler, B.W., Rothermel, R.C. 2002. A model for the steady spread of fire through a homogeneous fuel bed. In Viegas, D.X. (ed.) Proceedings of 4th International Conference on Forest Fire Research, 2002 Wildland Fire Safety Summit, Luso - Coimbra, Portugal – 18/23 November 2002. Millpress, Rotherdam. 11 p.

Cescatti, A., Marcolla, B. 2004. Drag coefficient and turbulence intensity in conifer canopies. Agricultural and Forest Meteorology 121:197-206.

Cheney, N.P. 1981. Fire behaviour. Pages 151-175 in Fire and the Australian Biota, edited by Gill, A.M., Groves, R.H., Noble, I.R., Australian Academy of Sciences, Canberra.

Cheney, N.P., Gould, J.S., Catchpole, W.R. 1998. Prediction of fire spread in grasslands. Int. J. Wildland Fire 8(1):1-13.

Cionco, R.M. 1965. A mathematical model for air flow in a vegetative canopy. J. App. Meteorol. 4:517-522.

Cionco, R.M. 1972. A wind profile index for canopy flow. Boundary-Layer Meteorology 3:255-263.

Clark, T.L., Jenkins, M.A., Coen, J., Packam, D. 1996. A couple atmosphere-fire model: Convective Feedback on fire-line dynamics. J. Appl. Meteo. 35:875-901.

Clark, T.L., Radke, L., Coen, J., Middleton, D. 1999. Analysis of small-scale convective dynamics in a crown fire using infrared video camera imagery. J. Applied Meteo. 38:1401-1420.

Committee on Fire Research. 1961. U.S. Nat. Acad. Sci. Res. Counc. Publ. 949.

Cox, G., Chitty, R. 1980. A study of the deterministic properties of unbounded fire plumes. Combustion and Flame 39:191-209.

Cruz, M.G. 1999. Modeling the initiation and spread of crown fires. Master of Science thesis. University of Montana, Missoula. 162 pp.

Cruz, M.G., Alexander, M.E., and Wakimoto, R.H. 2003a. Definition of a fire behavior model evaluation protocol: a case study application to crown fire behavior models. *In* Fire, Fuel Treatments, and Ecological Restoration: Conference Proceedings. *Technical edited by* P.N. Omi and L.A. Joyce. USDA For. Serv., Rocky Mt. Res. Stn., Fort Collins, CO. Proc. RMRS-P-29. pp. 49-67.

Cruz, M.G., Alexander, M.E., and Wakimoto, R.H. 2003b. Assessing canopy fuel stratum characteristics in crown fire prone fuel types of western North America. Int. J. Wildland Fire 12:39-50.

Cruz, M.G., Alexander, M.E., and Wakimoto, R.H.. 2003c. Assessing the probability of crown fire initiation based on fire danger indices. For. Chron. 79:976-983.

Cruz, M.G., Butler B.W. and D. X. Viegas. 2003d. Characterization of flame radiant heat fluxes in shrubland fires. Poster presented at the 2nd International Fire Ecology and Fire management Congress, November 2003, Orlando, FL.

Cruz, M.G., Alexander, M.E., and Wakimoto, R.H. 2004. Modeling the probability of crown fire initiation in conifer stands. For. Sci. 50: *In press.*

Cruz, M.G., Alexander, M.E., and Wakimoto, R.H. XXXX. Modeling the rate of spread of crown fires in conifer forest stands. Can. J. For. Res. XX: *in review.*

Davidson, G.A. 1986a. A discussion of Schatzmann's integral plume model from a control volume viewpoint. Journal of Climate and Apliied Meteorology 25:858-867.

Davidson, G.A. 1986b. Gaussian versus top-hat profile assumptions in integral plume models. Atmospheric Environment 20:471-478.

de Mestre, N.J., Catchpole, E.A., Anderson, D.H., Rothermel, R.C. 1989. Uniform propagation of a planar fire front without wind. Combustion Sci. Tech 65:231-244.

Dickinson, M.B., Johnson, E.A. 2001. Fire Effects on Trees. In Johnson, E.A. and Miyanishi, K. (eds.) Forest Fires, Behavior ands Ecological Effects. Academic Press, San Diego, CA. pp. 477-525 p

Dupuy, J.L., Larini, M. 1999. Fire spread through a porous forest fuel bed: A radiative and convective model including fire induced flow effects. Int. J. Wildand Fire 9(3):155-172.

Dupuy, J.L., Marechal, J., Bouvier, L., Lois, N. 1998. Measurement of temperatures and radiant heat fluxes during static fires in a porous fuel. Pages 843-858 in the Proceedings of 3rd International Conference on Forest Fire Research – 14th Conference on Fire and Forest Meteorology, Luso - Coimbra, Portugal - 16/20 November 1998.

Fahnestock, G.R. 1970. Two keys for appraising forest fire fuels. USDA Forest Service Research Paper PNW-99, Pacific Northwest For. and Range Exp. Station, Portland, Or.

Fendell, F.E. and Wolff, M.F. 2001. Wind-aided fire spread. In Johnson, E.A. and Miyanishi, K. (eds.) Forest Fires, Behavior ands Ecological Effects. Academic Press, San Diego, CA. pp. 171-223.

Fernandes, P.M., Catchpole, W.R., Rego, F.C. 2000. Shrubland fire behavior modelling with microplot data. Can. J. For. Res. 30:889-899.

Fernandes, P.A.M., Loureiro, C.A., Botelho, H.S. Fire behavior and severity in a maritime pine stand under differing fuel conditions. Ann. For. Sci. *In press*.

Finney, M.A. 1998. FARSITE: Fire area simulator—model development and evaluation. Res. Pap. RMRS-RP-4. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station. 47 p.

Forestry Canada Fire Danger Group. 1992, Development and structure of the Canadian forest fire behavior prediction system. Inf. Rep. ST-X-3. Ottawa, ON: Forestry Canada, Science and Sustainable Development Directorate. 65 p.

Fulé, P.Z.M., McHugh, C.; Heinlein, T.A.; M.; Covington, W.W. 2001. Potential fire behavior is reduced following forest restoration treatments. Pages 28-35 In: Vance, R.K. Edminster, C.B., Convigton, W. W., Blake, J.A. comps. 2001. Ponderosa pine ecosystems restoration and conservation: steps toward stewardship; 2000 April 25-27; Flagstaff, AZ. Proceedings RMRS-P-22. Ogden UT: USDA Forest Service, Rocky Mountain Research Station.

Fulé, P.Z.M., Covington, W. W., Smith, H.B., Springer, J.D., Heinlein, T. A., Huisinga, K.D., Moore, M.M. 2002. Comparing ecological restoration alternatives: Grand Canyon, Arizona. For. Ecol. and Man. 170:19-41.

Gill, A.M., Knight, I.K. 1991. Fire measurement. In: Cheney, N. P; Gill, A. M., eds. Proceedings of conference on bushfire modeling and fire danger rating systems; 1988 July 11–12; Canberra. Yarralumla, ACT: CSIRO Division of Forestry: 137–146.

Gould, J.S., Knight, I, Sullivan, A.L. 1997. Physical modelling of leaf scorch height from prescribed fires in young *Eucalyptus sieberi* regrowth forests in South-eastern Australia. Int. J. Wildland Fire 7(1):7-20.

Graham, R.T. (Tech. Editor) 2003. Hayman fire case study. Gen. Tech. Rep. RMRS-GTR-114. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station. 396 p.

117

Graham, R.T., Harvey, A.E., Jain, T.B., Tonn, J.R. 1999. The effects of thinning and similar stand treatments on fire behavior in Western forests. U.S.D.A. Forest Service General Technical Report PNW-GTR-463, Pacific Northwest Research Station, Portland, Or. 27 p.

Grishin, A.M. 1997. Mathematical modeling of forest fires and new methods of fighting them. Tomsk State University, Tomsk, Russia. 390 p.

Grishin, A.M., Perminov, V.A. 1991. Transition of the forest ground fire to crown fire. In Combustion Explosions and Shock Waves. Plenum Publishing Corporation.

Howell, J.R. 1982. A catalog of radiation configuration factors. 220 p. [http://www.me.utexas.edu/~howell].

Incropera, F.P., DeWitt, D.P. 2002. Fundamentals of Heat and Mass Transfer. Fifth Edition, John Wiley and Sons, Inc., 980 pp.

Izbicki, S., Keane, R. 1989. An Attempt at developing a crown fire ignition model. USDA For. Serv., Intermountain Fire Sciences Lab., Missoula, Montana. Coop. Agreement INT-88352-COA Final Rep. 26 p.

Johnson, E.A. and Miyanishi, K. The need for consideration of fire behavior and effects in prescribed burning. Restoration Ecology. 3(4):271-278.

Johnson, V.J. 1982. The dilemma of flame length and intensity. Fire Management Notes 43(4):3-7.

Keane, R.E., Garner, J.L., Schmidt, K.M., Long, D.G., Menakis, J.P., and Finney, M.A. 1998. Development of input layers for the FARSITE fire growth model for the Selway-Bitterroot Wilderness complex, USA. USDA For. Serv. Res. Pap. RMRS-RP-3.

Keyes, C.R., O'Hara, K.L. 2002. Quantifying stand targets for silvicultural prevention of crown fires. West. J. Appl. For. 17: 101-109.

Kilgore, B.M., Sando, R.W. 1975. Crown-fire potential in a sequoia forest after prescribed burning. Forest Sci. 21:83-87.

118

Kinnersley, R.P., Farrington-Smith, J.G., Shaw, G., Minski, M.J. 1994. Aerodynamic characterization of model tree canopies in a wind tunnel. The Science of the Total Environment 157:29-33.

Konev, E.V., Sukhinin, A.I. 1977. The analysis of flame spread through forest fuel. Combust. Flame 28:217-223.

Lalic, B., Mihailovic, D.T., Rajkovic, B., Arsenic, I.D., Radlovic, D. 2003. Wind profile within the forest canopy and in the transition layer above it. Environmental Modelling and Software 18:943-950.

Latham, D. 1998. High speed photography of fires. Pp 24-26 in preprint volume of the Second Symposium on Fire and Forest meteorology, 11-16 January 1998, Phoenix, AZ, by the American Meteorological Society, Boston, MA.

Lee, X. 2000. Air motion within and above forest vegetation in non-ideal conditions. Forest Ecology and Management. 135:3-18.

Linn, R.R. 1997. Transport model for prediction of wildfire behavior. Los Alamos National Laboratory Scientific Report LA13334-T. 195 pp.

Lopes, A.M.G., Sousa, A.C.M., Viegas, D.X. 1995. Numerical simulation of turbulent flow and fire propagation in complex topography. Numerical Heat transfer, Part A, 27:229-253.

Martin, R.E., Cushwa, C.T., Miller, R.L. 1969. Fire as a physical factor in wildland fire management. Proc. Annu. Tall Timbers Fire Ecol. Conf. 9:271-288.

McAlpine, R.S. Hobbs, M.W. 1994. Predicting the height to live crown base in plantations of four boreal forest species. Int. J. Wildland Fire 4(2):103-106.

McArthur, A.G. 1966. The application of a drought index system to Australian fire control. Commonw. Aust., Forest and Timber Bureau, For. Res. Inst., Camberra, 18 p.

Mendes-Lopes, J.M.C., Ventura, J.M.P., Rodrigues, J.A.M. 2002. Determination of heat transfer coefficient through a matrix of *Pinus pinaster* needles. In Forest Fire Research & Wildland Fire Safety. Proceedings of the IV International Conference

on Forest Fire Research/2002 Wildland Fire Safety Summit. Edited by D.X. Viegas. Millpress Scientific Publications, Rotterdam, The Netherlands. CD-ROM. 10 p.

Mendes-Lopes, J.M.C., Ventura, J.M.P. Amaral, J.M.P. 2003. Flame characteristics, temperature-time curves, and rate of spread in fires propagating in a bed of *Pinus pinaster* needles. Int. J. Wildland Fire 12:67-84.

Mercer, G.N., Weber, R.O. 1994. Plumes above line fires in a cross wind. Int. J. Wildland Fire 4(4):201-207.

Modest, M.F. 1993. Radiative heat transfer. McGraw-Hill, New York, 832 p.

Molchanov, V.P. 1957. Conditions for the spread of crown fire in pine forest. Translated from Lesnoe Khozydystvo 10:50-63

Moore, P.H., Gill, A.M., Kohnert, R. 1995. Quantifying bushfires for ecology using two electronic devices and biological indicators. CALM Science Supplement 4:83-88.

Morton, B.R., Taylor, G.I., and Turner, J.S. 1956. Turbulent gravitational convection from maintained and instantaneous sources. Proc. R. Soc. Lond. 234A:1-23.

Morvan, D., Dupuy, J.L, 2001. Modeling of fire spread through a forest fuel bed using a multiphase formulation. Combustion and Flame 127:1981-1994.

Nelson, R.M., Jr. 1980. Flame characteristics for fires in southern fuels. USDA For. Serv., Southeastern Forest Exp. Stn., Res. Pap. SE-205. Asheville, NC: 26 p.

Nelson, R.M., Jr. 2002. An effective wind speed for models of fire spread. Int. J. Wildland Fire 11:153-161.

Nelson, R.M., Jr. 2003a. Power of the fire – a thermodynamic analysis. Int. J. Wildland Fire 12:51-65.

Nelson, R.M., Jr. 2003b. Reaction times and burning rates for wind tunnel headfires. Int. J. Wildland Fire 12:195-211.

Nelson, R.M. Jr., Adkins, C.W. 1986. Flame characteristics of wind driven fires. Can. J. For. Res. 16:1293-1300.

Nicholls, J.W.P., Cheney, N.P. 1974. Effect of experimental and wild fires in pine plantations on wood characteristics. Aust. For. 36:164-177.

NWCG 1993. S-490 Advanced wildland fire behavior calculations. National Wildfire Coordinating Group. National Interagency Fire Center. NFES 2285. Boise, Idaho.

Ottmar, R.D., Vihnanek, R.E., Wright, C.S. 1998. Stereo photo series for quantifying natural fuels. Vol. I: Mixed-conifer with mortality, Western Juniper, Sagebrush, and Grassland types in the Interior Pacific Northwest. PMS 830. NFES 2580. Boise, Idaho: National Wildfire Coordinating Group. National Interagency Fire Center. 73 p.

Ottmar, R.D., R.E. Vihnanek, and C.S. Wright. 2000. Stereo photo series for quantifying natural fuels. Volume III: Lodgepole pine, quaking aspen, and gambel oak types in the Rocky Mountains. Natl Wildfire Coord. Group, Natl Interagency Fire Cent., Natl Fire Equip. Syst. [NFES], Boise, ID. Publ. NFES 2629. 85 p.

Ottmar, R.D., R.E. Vihnanek, and C.S. Wright. 2002. Stereo photo series for quantifying natural fuels. Volume Va: Jack pine in the Lakes States. Natl Wildfire Coord. Group, Natl Interagency Fire Cent., Natl Fire Equip. Syst. [NFES], Boise, ID. Publ. NFES 2669. 49 p.

Pagni, P.J., Peterson, T.G. 1973. Flame spread through porous fuels. 14th Symposium (international) on Combustion. 1099-1107.

Pastor, E., Zarate, L., Planas, E., and Arnaldos, J. 2003. Mathematical models and calculation systems for the study of wildland fire behaviour. Prog. Energy Combust. Sci. 29:139-153.

Philpot, C.W., Mutch, R.W. 1971. The seasonal trends in moisture content, ether extractives, and energy of ponderosa pine and Douglas-fir needles. Intermountain Forest and Range Experimental station, USDA For. Ser. Res. Pap. INT-102 Odgen,UT. 21 p.

Porterie, B., Loraud, J.C., Morvan, M., Larini, M. 1999. A numerical study of buoyant plumes in cross-flow conditions. Int. J. Wildland Fire 9:101-108.

121

Porterie, B., Morvan, D., Loraud, J.C., Larini, M. 2000. Firespread through fuel beds: modeling of wind-aided fires and induced hydrodynamics. Physics of Fluids 12(7):1762-1782.

Raupach, M.R., Thom, A.S. 1981. Turbulence in and above plant canopies. Ann. Rev. Fluid Mech. 13:97-129.

Riano, D.; Meier, E.; Allgower, B.; Chuvieco, E.; Ustin, S.L. 2003. Modeling airborne laser scanning data for the spatial generation of critical forest parameters in fire behavior modeling. Remote Sensing Environ. 86:177-186.

Rothermel, R.C. 1972. A mathematical model for predicting fire spread in wildland fuels. USDA For. Serv. Res. Pap. INT-115, Intermountain For. and Range Exp. Stn., Ogden Utah, 40 p.

Rothermel, R.C. 1983. How to predict the spread and intensity of forest and range fires. Gen. Tech. Rep. INT-143, Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station. 161 p.

Rothermel, R.C. 1991. Predicting behavior and size of crown fires in the Northern Rocky Mountains. Gen. Tech. Rep. INT-438. Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Research Station. 46 p.

Rothermel, R.C. 1994. Some fire behavior modeling concepts for fire management systems. Pages 164-171 in Proceedings of 12th Conference on Fire and Forest Meteorology, Jekyll Island, Georgia. Soc. Am. For., Bethesda, Maryland. SAF Publ. 94-02.

Rothermel, R.C., Deeming, J.E. 1980. Measuring and interpreting fire behavior for correlation with fire effects. USDA For. Serv. Res. Pap. INT-93, Intermountain For. and Range Exp. Stn., Odgen Utah, 4 p.

Ryan, K.C. 1981. Evaluation of passive flame height sensor to estimate forest fire intensity. USDA, Forest Service Res. Not. PNW-390. Pacific Northwest Forest and Range Exp. Station, Seattle, WA. 13 p.

Sackett, S.S., Haase, S.M., 1992. Measuring soil and tree temperatures during prescribed fires with thermocouple probes. USDA For. Serv. Gen. Tec. Rep. PSW-131, Pacific Southwest Forest and Range Experiment Station, Berkeley, CA, 15 p.

Saito, K. 2001. Flames. In Johnson, E.A. and Miyanishi, K. (eds.) Forest Fires, Behavior ands Ecological Effects. Academic Press, San Diego, CA. pp. 11-54.

Sando, R.W., Wick, C.H. 1972. A method for evaluating crown fuels in forest stands. USDA Forest Service Research Paper NC-84, North Central For. and Range Exp. Station, St. Paul, MN. 26 p.

Sanz, C. 2003. A note on κ - ε modeling of vegetation canopy air-flows. Boundary-Layer Meteorology 108:191-197.

Schatzmann, M. 1979. An integral model of plume rise. Atmospheric Environment 13:721-731.

Scott, J.H. 1999. NEXUS: A system for assessing crown fire hazard. Fire Management Notes 59(2):20-24.

Scott, J.H. Reinhardt, E.D. 2001. Assessing crown fire potential by linking models of surface and crown fire behavior. Res. Pap. RMRS-RP-29. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station. 59 p.

Scott, J.H. Reinhardt, E.D. 2002. Estimating canopy fuels in conifer forests. Fire Management Notes 62(4):45-50.

Shaddix, C.R. 1998. Practical aspects of correcting thermocouple measurements for radiation loss. Paper WSS/CI 98F-24. 1998 Fall meeting of Western States Section, The Combustion Institute, University of Washington, Seattle, WA. 18 p.

Shannon K.S., Butler, B.W. 2003. A review of error associated with thermocouple temperature measurement in fire environments. In preprint proceedings of the 2nd International Fire Ecology and Fire management Congress, November 2003, Orlando, FL. 3 p.

Shen, S., Leclerc, M.Y. 1997. Modelling the turbulence structure in the canopy layer. Agricultural and Forest Meteorology 87:3-25.

Siegel, R., Howell, J.R. 1992. Thermal radiation heat transfer. Third EditionTaylor and Francis, 1072 p.

Simard, A.J., Haines, D.A., Blank, R.W., Frost, J.S. 1983. The Mack lake fire. USDA For. Serv. Gen. Tec. Rep. NC-83, North Central Forest Experiment Station St. Paul, Minnesota. 36 p.

Simard, A.J., Blank, R.W., Hobrla, S.H. 1989. Measuring and interpreting flame height in wildland fires. Fire Tech. 25:114-133.

Smith, S.G., Engstrom, J.D., Butler, J.K., Baxter, L.L., Fletcher, T.H., Weise, D.R. 2003. Ignition behavior of live California chaparral leaves. In preprint proceedings of the 2nd International Fire Ecology and Fire management Congress, November 2003, Orlando, FL. 7 p.

Stephens, S.L. 1998. Evaluation of the effects of silvicultural and fuels treatments on potential fire behavior in Sierra Nevada mixed conifer forests. Forest Ecology and Management 105:21-35.

Stocks, B.J., M.E. Alexander, B.M. Wotton, C.N. Stefner, M.D. Flannigan, S.W. Taylor, N. Lavoie, J.A. Mason, G.R. Hartley, M.E. Maffey, G.N. Dalrymple, T.W. Blake, M.G. Cruz, and R.A. Lanoville. 2004. Crown fire behavior in northern jack pine/black spruce forests: the International Crown Fire Modelling Experiment. Can. J. For. Res. 34: *Accepted.*

Sullivan, A.L., Knight, I.K., Cheney, N.P. 2002. Predicting radiant heat flux from burning logs in a forest following a fire. Australian Forestry 65(1):59-67.

Sullivan, A.L., Ellis, P.F., Knight, I.K. 2003. A review of radiant heat flux models used in bushfire applications. Int. J. Wildland Fire 12:101-110.

Sussot, R.A. 1982. Characterization of the thermal properties of forest fuels by combustible gas analysis. Forest Sci. 28:404-420.

Thomas, P.H. 1963. The size of flames from natural fires. In: Proceedings ninth symposium on combustion. Pages 844-859. Ithaca, NY.

Thomas, P.H., Scott, R. 1963. Research on forest fires. Report on Forest Research 1962, 116-119.

Van Loon, A.P., Love, A.P. 1973. A prescribed burning experiment on young slash pine. For. Comm. N.S.W., Sydney, N.S.W. Res. Note no. 25. 53 p.

Van Wagner, C.E. 1968. Fire behaviour mechanisms in a red pine plantation: field and laboratory evidence. Can. Dep. For. Rural Develop. Publ. 1229. 30 p.

Van Wagner, C.E. 1973. Height of crown scorch in forest fires. Can. J. For. Res. 3:373-378.

Van Wagner, C.E. 1977. Conditions for the start and spread of crown fire. Canadian Journal of Forest Research 7:23-34.

Van Wagner, C.E. 1989. Prediction of crown fire behavior in conifer stands. Pages 207-212 in Maciver, D.C., Auld, H., Whitewood, R. (eds) Proceedings of the 10th conference on fire and forest meteorology; April 17-21, 1989, Ottawa, ON, Canada, Forestry Canada and Environment, Canada.

Viegas, D.X., Cruz, M.G., Ribeiro, L.M., Silva, A.J., Oleero, A., Arrue, B., Dios, R., Gomez-Rodriguez, F., Merino, L., Miranda, A.I., Santos, P. 2002. Gestosa fire spread experiments. In Forest Fire Research & Wildland Fire Safety. Proceedings of the IV International Conference on Forest Fire Research/2002 Wildland Fire Safety Summit. Edited by D.X. Viegas. Millpress Scientific Publications, Rotterdam, The Netherlands. CD-ROM. 13 p.

Wade, D.D., Ward, D.E. 1973. An analysis of the Air Force Bomb Range Fire. USDA For Serv. Southeast For. Range Exp. Stn., Asheville, North Carolina. Res. Pap. SE-105.38 p.

Walker, J.D., Stocks, B.J. 1968. Thermocouple errors in forest fire research. Fire Technology 4(1):59-62.

Weber, R.O. 1991. Modelling fire spread through fuel beds. Proc. Energy Combust. Sci. 17:67-82.

Weber, R.O., Gill, A.M., Lyons, P.R.A., Mercer, G.N. 1995. Time dependence of temperature above wildland fires. CALMScience Supplement 4:17-22.

Williams, D.F. 1977. Influence of fuel quantity, distribution and moisture content on fire management of radiate pine plantations. Univ. Melbourne, Victoria, M.Sc. Thesis 188 p.

Williams, J.T., Rothermel, R.C. 1992. Fire dynamics in Northern rocky mountain stand types. USDA For. Serv. Res. Note INT-405, Intermountain For. and Range Exp. Stn., Ogden Utah, 4p.

Wolfram, S. 1999. The Mathematica book. Wolfram Media/Cambridge University Press, 4th Ed. 1470 pp.

Wotton, B.M., Renaud, J. 1996. A simple physical fire growth model. Pages 175-182 in Weber, R.O. (Chair) Proceeding of the 13th Conference on fire and forest meteorology, Oct. 1996, Lorne, Victoria, Australia. International Association of Wildland Fire, Moran WY.

Xanthopoulos, G. 1990. Development of a wildland crown fire initiation model. Ph.D. Thesis, University of Montana, Missoula. 152 p.

Xanthopoulos, G., Wakimoto, R.H. 1993. A time to ignition – temperature – moisture relationship for branches of three western conifers. Can. J. For. Res. 23:253-258.

Yih, C.S. 1953. Free convection due to boundary sources. Pages 117 – 133 in the Proceeding of the First symposium on the use of models in geophysics. U.S. Government Printing Office.

Yih, C.S. 1969. Fluid Mechanics, A concise introduction to the theory. McGraw-Hill Book Company, New York. 622 p.

Zukoski, E.E. 1995. Properties of fire plumes. Pages 101-219 in Cox, G. (Editor) Combustion fundamentals of fire. Academic Press Limited. 476 pp.

CHAPTER 2:

Upward heat fluxes from spreading surface fires: observation and prediction

Abstract:

How energy released by wildland fires is partitioned into radiative and convective heat transfer components is a complicated process, determined by such factors as fuelbed structure, weather variables, fuel moisture content, fire energy release rates and water vapor in the fire and buoyant plumes. The understanding of how upward radiative and convective heat fluxes above spreading surface fires are partitioned could result in better models describing heat transfer processes controlling fire spread and ignition of canopy fuels.

A limited number of laboratory and outdoor experimental fires were instrumented to measure upward radiative and convective heat fluxes. For the laboratory fires, peak incident radiative heat fluxes measured 1.1 m above the fuelbed ranged from 0.6 to 24 kW m^{-2}. Measured convective heat fluxes ranged from 2.2 to 35 kW m^{-2}. No evidence was found for a preponderance of one heat flux process over the other. The fire behavior quantities that were most related with the upward heat fluxes were reaction time and predicted flame height. No significant relationships were found between fire intensity measures, such as fireline intensity or reaction intensity and various measures quantifying upward heat flux, namely peak and cumulative heat fluxes. Measured heat fluxes above the outdoor experimental fires were substantially higher than those observed in the laboratory setting.

The use of models to predict upward radiative heat flux and buoyant plume behavior showed no evidence of bias, although predictions showed some degree of variability. Analysis of the observed heat fluxes and model predictions indicates that the heat flux partitioning into convective and radiative processes is highly dynamic in time and space, and determined by fuel complex characteristics and burning conditions.

Keywords: fire behavior, modeling, heat transfer, crown fire initiation.

2.1. Introduction

Fire behavior modeling is aimed at developing tools that can be used to support fire management decision making and advance our understanding of various fire phenomena. The increase in available inexpensive computing power has made possible the application of modern numerical methods to the theoretical analysis of the combustion and heat transfer processes involved in wildland fires. This approach attempts to describe fire phenomenology by solving a set of equations describing the local conservation of mass, momentum, energy and species for the system (Porterie et al. 2000, Morvan and Dupuy 2001, Sero-Guillaume and Margerit 2002). The fundamental description of fire and atmospheric dynamics processes presented by these models give them the potential to fully describe the behavior of free burning wildland fires. Nevertheless, our current-state-of-knowledge regarding fire-related processes has lead to the use of numerous empirically based sub-models and constants to describe fire phenomena where our knowledge is still incomplete. Important fire phenomena driving combustion and heat transfer processes are characterized on the basis of fine-scale laboratory experiments (e.g., Churchill and Bernstein 1977 in Dupuy and Larini 1999; Kaplan et al. 1996 in Porterie et al. 2000) or based on simplified theories (e.g., Clark et al. 1996) without verification of their validity to the chemical and thermal environment that characterizes high-intensity wildland fires. Examples of these are the convective heat transfer to unburned fuel particles, their thermal decomposition, flame structure, and the formation, oxidation and radiative properties of soot. These general assumptions suffice as an academic exercise, but might not be realistic when applied to an actual wildland fire environment. A further difficulty in understanding how these physically

129

based fire behavior models actually perform is that they have seldom been subjected to evaluation against independent data. This might arise from their inherent complexity and computing requirements, which makes it difficult to analyze, and the lack of adequate data to use for comparison (Clark et al. 1999).

The successful application of these models require proper calibration and testing of sub-models describing the various physicochemical processes determining combustion and heat transfer processes (Dupuy et al. 2003). The violent environment produced by a wildland fire restricts the type of sensors that can be used to quantitatively describe these fire properties. This has limited the description of fundamental flame properties in spreading fires to a few parameters. Fluid temperatures are one of the most common fundamental parameters measured in fire research (e.g., Van Wagner 1970, Moore et al. 1995, Weber et al. 1995). In order to better understand the heat transfer mechanisms determining fire spread rate, several studies aimed at describing the vertical radiometric profile of the flame front in free-burning fires (Butler 1994, Wotton et al. 1998, Butler 2003, Cruz et al. 2003). Measurements of fluid velocities, both within and above the flame, have been reported for several laboratory (Latham 1998, Dupuy et al. 2003) and field studies (Palmer 1980, Gould et al. 1997, Clark et al. 1999).

To my knowledge only a few studies have report measurements of upward heat fluxes from spreading surface fires. Packham (1969) quantified upward heat transfer above a moderate intensity surface fire in a maritime pine (*Pinus pinaster*) plantation by measuring upward radiation and deriving convective heat from air temperature and velocity in the plume. This case study indicated a prevalence of convection over radiation. Gould et al. (1997) provided a complete description of the upward heat fluxes

130

by measuring the radiative heat flux and buoyant plume structure above prescribed fires. The Gould et al. (1997) analysis concentrated on the buoyant plume dynamics and its effect in determining lethal crown scorching. These authors bypassed the analysis of the radiative heat flux data, possibly because the measured radiative heat fluxes were found to be low and not relevant to the process of lethal crown scorching. Their study also illustrated the difficulties inherent in conducting outdoor experimental fire. Of the 14 fully instrumented prescribed fires only one produced data suitable for analysis of the buoyant plume.

The objectives of the present study were three-fold. One was to collect data on the fundamental quantities determining heat transfer to fuels above moving surface fires; the measured quantities were incident radiative and convective heat flux, and fluid velocities and temperature. Secondly, we examine how commonly measured fire behavior quantities, namely fireline intensity, flame front dimensions, reaction time and fuel consumption relate to the heat transfer quantities. Finally the heat flux data collected was used to evaluate the models described in Chapter 1.

2.2. Material and methods

2.2.1. Instrumentation

To capture the temporal and spatial dynamics of (1) upward incident radiative and convective heat fluxes, and (2) fluid velocities and temperature, various instruments were assembled into a measurement package that allowed the sampling of all quantities within a relatively small unit volume. Incident radiant and total heat fluxes were measured with

131

a Schmidt-Boelter type dual heat flux sensor (Model 64-20T-20R, Medtherm Corporation Huntsville, AL, USA). This sensor incorporates a hemispherical radiometer and a total heat flux sensor, ensuring that the measurements are made in the same surface. Assuming heat transfer by conduction as negligible within the flame and buoyant plume, convective heat flux to the sensor was derived from the difference of the measured total and radiative heat flux. For the range of heat fluxes measured in wildland fire flames, the sensor has a linear output response directly proportional to the incident transfer rate and a time constant of less than 100 ms. The radiative and total heat flux sensors were separately calibrated based on a known heat source until a heat flux of 200 kW m^{-2}.

Fluid temperatures were measured with small diameter type K thermocouples (chrome-alumel) with a wire and bead diameter of 0.125 mm and 0.25 mm, respectively. It is expected that the use of these fine thermocouples would limit any errors as a result of radiative heat transfer. Fire generated flow velocities, within the flame and buoyant plume, were measured with bi-directional Kiel-static probes (Rothermel 1967b, McCaffrey and Heskestad 1976, Newman 1987). These probes were used in pairs to sample both the horizontal and vertical flow components. The Probes were calibrated in a wind tunnel at a constant air temperature (295 K) and atmospheric pressure (1018 mb) for a wind speed range of 3 – 13 m s^{-1}. Data was acquired at a rate of one sample per second by an AM25T solid state multiplexer and stored in a Campbell CR10X datalogger (Campbell Scientific, Logan, UT, USA).

132

2.2.2. Laboratory fires

The laboratory experiments were carried out in the wind tunnel at the Fire Sciences Laboratory (FiSL) of the USDA Forest Service Rocky Mountain Research Station, Missoula, MT. The FiSL experiments attempted to characterize upward heat fluxes under different burning conditions as determined by fuel load and wind speed. The FiSL wind tunnel has a cross section of 3.0 x 3.0 m and an overall length of 12 m. The experiments were carried out in a 7.0 m long and 1.0 m wide fuelbed placed in the wind tunnel floor. Details relative to the wind tunnel characteristics can be found in Rothermel (1967) and Catchpole et al. (1993, 1998). Fuelbeds were built with excelsior. Excelsior is comprised of long wood strands commonly used in past laboratory fire research at FiSL (e.g., Rothermel 1972, McAlpine and Wakimoto 1991, Catchpole et al. 2002). The advantages of this fuel relative to others, such as pine needles, are the relatively uniform size and shape (for a surface-area-to-volume ratio of 3092 m^{-1}) and the ease in reproducing reasonably uniform fuelbeds. In order to affect combustion characteristics, fuelbed properties were varied: fuel load (0.3, 0.61 and 1.2 kg m^{-2}) and packing ratio (0.0094, 0.0152 and 0.031). The experiments were conducted at five different wind speeds (0.0, 0.9, 1.34, 1.8, and 2.6 m s^{-1}). Given the turbulent wind flow within the wind tunnel (Rothermel 1967a), the wind tunnel flow velocity was considered to be the mid-flame wind speed (Rothermel 1972). Fuel moisture was not experimentally controlled, but held relatively constant by controlling air temperature and relative humidity. Two fuel samples were collected prior to each burn for fuel moisture determination. Samples were dried for 24 h at ±100 °C. Fuelbed ignition was achieved through an electrically heated coil lying in excelsior soaked with an inflammable mixture

133

(85 % gasoline and 15 % diesel). The ensuing flame front was allowed to spread for 3.0 m in order to achieve a steady-state propagation, after which fire measurements were undertaken. Rate of fire spread was measured through the use of thermocouples placed at 0.5 m intervals along the fuelbed centerline. A thermocouple temperature reading of 600 K was used to indicate the arrival of the flame front. A weight slot (0.25 x 0.40 m) placed near the end of the fuelbed was used to measure fuel weight loss, a measure of the rate of gasification of fuel. Directly above the weight slot two dual heat flux sensors were placed at 0.1 and 1.1 m, respectively. Two other dual heat flux sensors were placed in the fuelbed to measure incident heat fluxes from the free flame into the top of the fuelbed. One of the sensors was oriented vertically and the other at a 45° angle towards the flame front.

Five fire behavior quantities were derived or estimated for each burn. Fireline intensity (I_B), as defined by Byram (1959) was determined by the following equation (Catchpole et al. 1993):

[2.1] $\quad I_B = ROS \cdot w_a \cdot Hc \cdot \eta$

with ROS being the fire rate of spread (m s^{-1}), w_a the fuel consumed in flaming combustion (kg m^{-2}), Hc the fuel particle heat content (kJ kg^{-1}) and η the combustion efficiency. Combustion efficiency is estimated from the original dry fuel converted to char and it is considered constant at 0.85 for the fuels used (Nelson 2003b). Average reaction intensity (I_R) was estimated as (Catchpole et al. 1993):

[2.2] $\quad I_R = \dfrac{w_a \cdot Hc \cdot \eta}{\tau_r}$

134

where τ_r is the fire's reaction time (s). Reaction time was derived from the time-temperature profiles measured in the experimental fires. Reaction time is assumed to be the time that temperature is above 600 K on the top of the fuelbed. Flame depth (D_F) was determined from the product of rate of spread by reaction time. Flame height and length were only measured on a small number of fires. Flame height (H_F) was then estimated from an equation developed by Catchpole and Catchpole (2000) in the same experimental setting using the following equation:

[2.3] $\quad H_F = 3.2743 \cdot w_a^{0.9405} \cdot U^{0.1129} \cdot \exp(-0.6323 \cdot w_a - 1.067 \cdot MC - 17.044 \cdot \beta)$

where U is the wind tunnel wind speed (m s^{-1}), MC the moisture content of fuels (% ovendry weight) and β is the fuelbed packing ratio.

2.2.3. Outdoor experimental fires

Measurements of upward radiant and convective heat fluxes, fire generated flow velocities and temperatures were made in eight outdoor fires. Measurements were made on six shrubland experimental fires in Portugal, Gestosa (4 plots) and Paredes (2 plots) and two prescribed fires in central Montana. Two other prescribed fires in Montana were instrumented but data was lost due to malfunctioning of the data acquisition system.

2.2.3.1. Shrubland experimental fires

The shrubland experimental fires were conducted in two distinct locations in Portugal, Gestosa and Paredes, with the objective to study fire behavior in shrubland fuels. The Gestosa site is located in central Portugal Lousã mountain range (40°15'N, 8°10'W) at an average altitude of 700 m. Average annual precipitation and temperature

135

are respectively 760 mm and 11°C. The vegetation cover in the experimental area consists of continuous shrublands dominated by the following species: *Erica umbellata, Erica australis* and *Chamaespartium tridentatum*. Detailed information regarding the experimental burning program, which was started in 1998 can be found in Viegas et al. 2002). The Paredes plots were located in the Alvão mountain range (41°21'N, 7°45'W) at an average elevation of 1000 m. Dominant species were *Erica australis* and *Chamaespartium tridentatum*. Average annual precipitation and temperature for the Paredes site are 1100 mm and 12°C, respectively Detailed information about fuel complex structure and associated fire behavior on these fuel types can be found in Cruz and Viegas (1998) and Fernandes et al. (2000).

2.2.3.2. Tenderfoot Creek Experimental Forest prescribed fires

Two prescribed fires were carried out in multi-aged lodgepole pine (*Pinus contorta*) stands in the Tenderfoot Creek Experimental Forest (TCEF), (46° 55' N, 110° 51' W, elev. 2166 m), central Montana. The forest type consists of lodgepole pine stands typical of moderate to high altitudes in the Northern Rocky Mountains. Climate records indicate an average annual precipitation of 880 mm, with approximately 70 percent falling from November to May (Schmidt and Friede 1996). The burns were carried out within the scope of a project evaluating the ecological and biological effects of silvicultural treatments and prescribed fire in manipulating stand structure. Unit 10/4 (hereafter TCEF10) had been thinned the previous summer, leaving the remaining lodgepole pine/Engelmann spruce (*Picea engelmannii*) stand with an average tree height of 21 m, canopy base height of 8.7 m and a basal area of 8.6 m^2 ha^{-1}. Unit 16/25

(hereafter TCEF16) was a pure lodgepole stand with a basal area of 20.5 m^2 ha^{-1}, average tree height of 18.9 m, and a canopy base height of 8.6 m.

2.2.3.3. Surface fuel sampling

For the Gestosa and Paredes burns, fuelbed structure was determined by double sampling techniques (Catchpole and Wheeler 1992). Destructive sampling of shrub fuels was aimed at determining fuel load and bulk density by fuel particle size class and state (live or dead). The linear transects method (Canfield 1941) was used to estimate species composition and percent coverage, height and volume by species in each burn plot. Ground and surface fuelbed sampling in the TCEF burn plots followed the planar intersect method (Brown et al. 1982).

2.2.3.4. Weather

Relevant weather conditions directly influencing fire behavior, namely wind speed, temperature and relative humidity were recorded differently at each experimental fire site. For the Gestosa experiments, a network of three automatic weather stations positioned within the experimental area was used, whereas for the Paredes fires only one weather station was used. The weather stations were located so that these measurements were not influenced by fire-generated wind flow. For the two TCEF prescribed fires, wind speed, temperature and relative humidity were recorded manually at eye-level using a handheld sensor.

137

2.2.3.5. *Fuel moisture sampling*

Fuel moisture samples were collected prior to ignition on all experimental fires. For the shrubland fires, live fuel moisture sampling comprised foliage and small twigs of the dominant species. For the two TCEF prescribed burns, fuel moisture sampling was restricted to the litter and duff layers. Fuel moisture samples were oven dried at $\pm 100°C$ for 24 hours.

2.2.3.6. *Fire behavior measurements*

Measurement packages, each consisting of one dual heat flux sensor, one thermocouple and one Kiel-static probe pair were mounted in a 4.5 m tall tower at 1.5, 3 and 4.5 m above ground with the objective of measuring upward incident radiant and total heat fluxes, and fire-generated flow velocities and temperatures at various heights above the experimental fires. A thermocouple array was vertically placed along the tower to collect temperature measurements at 0.5 m intervals. The tower was insulated with several layers of aluminum foil/fiberglass cloth and placed within the experimental plot in an area with homogeneous fuel distribution. The data acquisition system was buried and thermally insulated in the ground at a distance approximately of 3-5 meters from the tower.

In the Gestosa fire experiments rate of spread was measured through the use of oblique infrared images (Agema ThermoVision 550, Agema, Sweden) obtained from a low flying helicopter. For the Paredes experimental fires, rate of spread was obtained from the flame front arrival time to known reference points. In the TCEF prescribed fires rate of spread was estimated through the analysis of video images. For all the

138

experimental fires, flame front characteristics were estimated from still photo and video images.

Figure 2.1. Photographs of two experimental fires approaching the instrument tower. Left photograph was taken at TCEF16; right photograph was taken at Gestosa Plot 519 experimental fire. Height of tower is 4.5 m. Distance between sensor packages is 1.5 m.

139

2.2.4. Description of the sub-models being evaluated

The various heat flux and fluid temperatures and velocities measured in the laboratory and field experimental fires are appropriate to evaluate and calibrate the radiative transfer equation and the plume model described in Chapter 1. Some restrictions exist. It is not expected that the laboratory fires, conducted in a wind tunnel with a 3.0 x 3.0 m section, provide an adequate situation to evaluate the plume model, as the combination of forced flow in an enclosed space and low ceiling limit the development of buoyancy as expected to occur in outdoor experimental fires. Thus, comparisons between the plume model output and the observed plume characteristics were restricted to the outdoor experimental fires.

The basis of the radiative transfer equation being evaluated was described in Chapter 1. The equation describing the radiative heat transfer to a surface above and parallel to a horizontal radiant plane considering the variation in radiosity with flame depth is:

$$[2.4] \quad I_{12} = \int_{-\frac{L_p}{2}}^{\frac{L_p}{2}} \int_{D_x}^{D_x+D_F} \int_{-\frac{W_p}{2}}^{\frac{W_p}{2}} \int_{-\frac{W_F}{2}}^{\frac{W_F}{2}} \frac{\varepsilon \cdot \sigma_{SB} \cdot \left(T_F^{\,4}(x_F) - T_a^{\,4}\right) \cdot \left(\frac{z}{S}\right)^2}{\pi \cdot S^2} dy_F \, dy_p \, dx_F \, dx_p$$

This equation differs slightly from the one described in Chapter 1. The varying surface temperature of the absorbing surface is changed to a constant temperature (ambient) of the heat flux sensor. The sensor constant temperature during an experimental fire is assured by water-based cooling system for the FiSL laboratory fires and by setting the sensor within a high conductivity heat sink for the field fires. The exponential decay

140

function to account for the reduction in radiant energy while radiation travels in the absorbing porous medium consisting of the subcanopy space was removed in eq. 2.4.

The variation of flame gases temperature on the top of the fuelbed with flame depth is obtained by combining *ROS* with the time-temperature profile model developed in Chapter 1:

$$[2.5] \quad T_F(D_F) = \begin{cases} T_a + (T_{F\max} - T_a) \cdot \exp\left(\dfrac{-((t + \lambda_F) \cdot ROS)^2}{\beta_F^2} \right) & \rightarrow \Delta T \uparrow \\ T_a + (T_{F\max} - T_a) \cdot \exp(-\gamma_F \cdot ((t + \lambda_F) \cdot ROS)), & \rightarrow \Delta T \downarrow \end{cases}$$

With maximum gas temperature on top of the fuelbed (T_{Fmax}) obtained from:

$$[2.6] \quad T_{F\max} = 300 + 300.7 \cdot w_a + 136.8 \cdot U^{0.51} + 100.5 \cdot MC^{-0.53}$$

Parameter λ_F in equation [2.5] ensures that the ignition interface is at x = 0. The other two parameters, γ_F and β_F, in this equation are determined through Newton's method after determination of T_{Fmax} and τ_r.

Simulations of the plume characteristics above a surface fire were obtained using Mercer and Weber's (1994) buoyant plume model. This model consists of a system of six ordinary differential equations that solve a simplified version of the conservation equations for the plume centerline assuming a top-hat temperature and velocity distribution (see Section 1.3.1.2 and Appendix A).

141

2.3. Results and discussion

2.3.1. FiSL laboratory fires: analysis of upward heat fluxes

The laboratory fires were designed with the objective of quantifying upward heat fluxes above a spreading flame front under controlled conditions and analyzing the relationships between measured heat flux characteristics with commonly measured fire behavior descriptors. Before describing the results it is important to introduce a note relative to differences between radiative and convective heat transfer. From the standpoint of the physical phenomenon, convection and radiation are distinct processes. While considering radiative heat transfer, we can refer to incident (i.e., the amount of radiant energy incident on a surface and independent of surface properties) and absorbed (i.e., the amount of radiant energy absorbed by a surface, function of incident radiation and surface characteristics such as optical properties) heat transfer. Given the instrument calibration and assumption that sensor remained at ambient temperature for duration of fire, the measured quantity by the radiometer is essentially the incident radiative heat flux, which is primarily dependent on the source. Convective heat transfer refers to the amount of heat absorbed by a surface from a gas, and is dependent not only on the thermodynamic properties of the gas but also strongly dependent on the shape, size and temperature of the object being heated or cooled (Hornbaker and Rall 1968). The result is that based on certain assumptions (e.g., blackbody, fuel particle geometry and orientation) the measured incident radiation in the present study can be expected to be very similar to incident radiative heat flux to the fuel particles above a surface fire, while the measured convective heat flux is distinct from the convective heat flux to fuel particles. Nevertheless, the measurement of the radiative and convective heat flux by the

142

dual heat flux sensors increases out understanding of how energy released by the fire is partitioned into those components given certain fuel complex characteristics and burning conditions.

In order to smooth local fluctuations in the collected heat flux and fluid temperature data, the collected time series data was linearly transformed using a simple 5-s moving average. Fig. 2.2 presents the typical space and time dependent heat flux and temperature traces for a laboratory fire (FiSL experimental fire 20, see Table B.1 in Appendix B for experimental database details). For this experimental fire (w_a = 1.2 kg m^{-2}; U = 1.8 m s^{-1}; ROS = 0.033 m s^{-1}; D_F = 1.7 m) the upward convective heat flux was higher than the radiative heat flux at both 0.1 and 1.1 m above the fuelbed. Fluid temperature traces show that the heat flux sensors located 1.1 m above the fuelbed were immersed in the flame. The incident heat flux trace obtained from the 45° oriented heat flux sensors located in the fuelbed shows the importance of the radiative heat flux pre-heating for this particular fire followed by a predominance of convective heat transfer while the sensor is within the reaction zone.

143
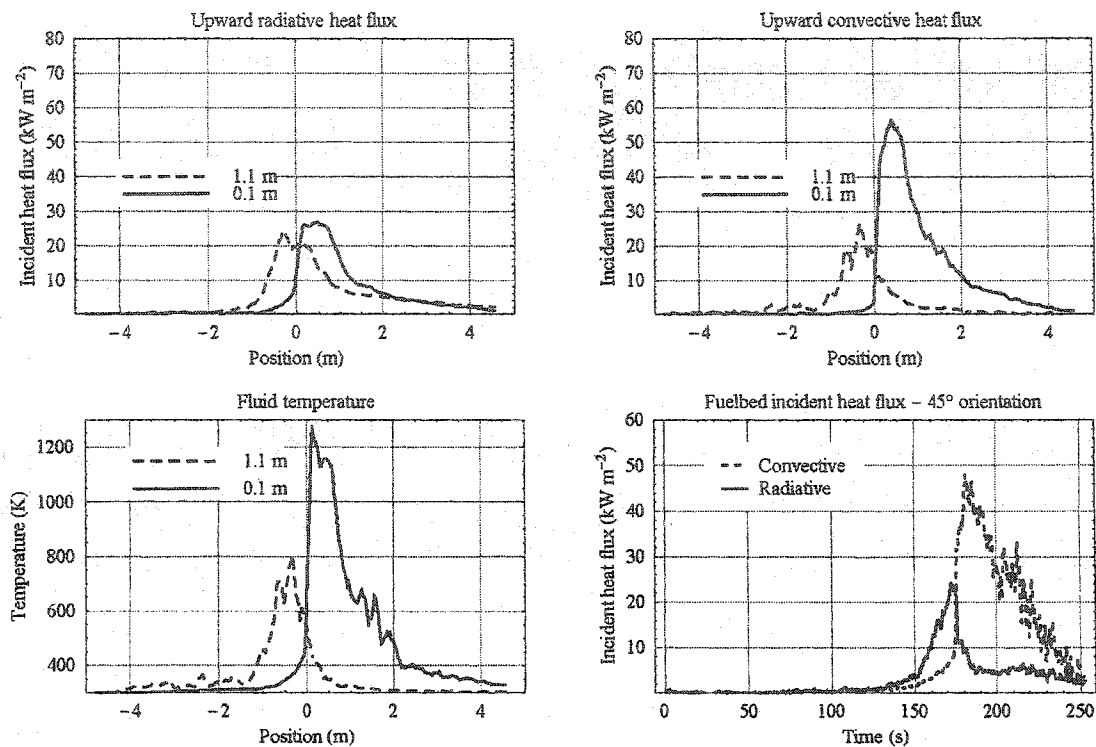
Figure 2.2. Upward and forward incident heat flux and temperature traces collected in FiSL experimental fire 20. The 0 m position in the x-axis indicates the location of the ignition interface. Specific fire characteristics for this experimental fire are given in Table A1.

Relationship between upward heat flux measures with fuel and environmental variables

From each upward heat flux trace various characteristics were derived for analysis, namely peak heat flux, cumulative heat flux until the peak heat flux is attained ($\Sigma_{peak}q$), total cumulative heat flux (Σq), maximum rate of change in heat flux (Max Δq) and average rate of change in heat flux (Av Δq) during the rising component of the trace. The two cumulative heat flux measures corresponded to the integration of the measured heat flux over time. Table 2.1 summarizes the descriptive statistics in the dataset. The relationship between radiative and convective heat flux descriptors and the fire behavior characteristics determined for each experimental fire were then analyzed for possible

144

relationships. For the purpose of the present study, the heat flux variables measured at a height of 1.1 m assume more relevance than the ones measured at 0.1 m. At 1.1 m, most of the energy in the volatiles from the pyrolized solid fuels has been released, while for the most intense fires only a fraction of the released volatiles have reacted with air at 0.1 m.

From the environmental and fuel variables analyzed, w_a showed the highest number of significant correlations (Table 2.2, * - $p < 0.05$; ** - $p < 0.01$) to the heat flux measures (13 of the 20 heat flux descriptors). Given lack of an objective method to estimate w_a, this quantity was assumed equal to the experimental fire fuel load. The relationship between w_a and the heat flux descriptors measured at 1.1 m was noteworthy. Nine out of ten of the heat flux descriptors measured at this height were significantly correlated with w_a. Wind speed was correlated significantly with 5 of the 20 heat flux properties analyzed. Contrary to the trend evidenced with other variables analyzed, wind speed showed a higher number of correlations with heat flux characteristics measured at 0.1 m.

Both fire behavior quantities describing fire intensity, I_B and I_R, produced a relatively small number of significant correlations with the heat flux measures, nine and four, respectively (Table 2.2). I_B, defined as the integrated heat released rate by the fire front, has been extensively used as a surrogate of flame length (Alexander 1998) and to predict air temperature above surface fires (Van Wagner 1975, Weber et al. 1995, Gould et al. 1997) and in explaining lethal crown scorch heights (Van Wagner 1973, Saveland et al. 1990, Alexander 1998). For the present dataset, I_B was not significantly related with the peak and the two cumulative measures of convective heat flux, variables that in theory should be indicative of the convective energy reaching the fuels above a surface fire (Fig

2.3). The small number of significant correlations between I_B and the heat flux measures might indicate that I_B is a crude fire descriptor when characterizing fire behavior from a fundamental standpoint. A possible explanation is that, as I_B does not discriminate how fuel consumption is partitioned between flaming and residual components, it fails to provide an adequate description of how heat is being released by the fire.

The average I_R was also weakly related with the heat flux descriptors (Table 2.2, Fig. 2.4). As with I_B, I_R estimation assumed the experimental fire fuel load as w_a. Even for the excelsior fuels used, with a relatively high surface-area-to-volume ratio, a proportion of fuel load will be not consumed during the flaming combustion stage. Observation of the flaming front trailing edge suggested that the experiments with higher fuel loads and higher packing ratio had substantial amounts of fuels being consumed in glowing combustion. Average I_R can also be estimated from the fuel weight loss trace as per Frandsen and Rothermel (1972). This average I_R based on the rate of fuel weight loss failed to show differences between experimental fires, possibly due to the small variability in some fuelbed structure characteristics, such as fuel particle size.

From the fire behavior descriptors analyzed, the predicted H_F and τ_r (measured) showed the most number of significant correlations, 15 and 16 respectively. The predicted H_F was significantly linearly correlated with the heat flux variables captured at 1.1 m above the fuelbed (Table 2.2, Fig.2.5). The relevance of the relationship between the heat flux variables and the predicted H_F is that the heat flux variables are possibly related with what we perceive as flame height. Conversely, the various measures of heat flux quantified in the present study could possibly be satisfactorily explained through non-linear regression analysis incorporating fuelbed and fire environment variables. Reaction time was

146

significantly correlated with all heat flux quantities but the ones related to the incident radiative heat flux measured 0.1 m above the fuelbed. Reaction time can best be interpreted as the duration of time the fuel volatiles are released at a rate that allow for the occurrence of flaming combustion. This explains why its duration was well related with the measured upward heat flux, while D_F was not.

Table 2.1. General statistics for environmental, fuel, fire behavior and heat flux data for FiSL experimental fire dataset.

Parameters	N	Minimum	Maximum	Mean	Std. Dev.
Wind velocity (m s^{-1})	35	0.00	2.80	1.38	0.78
Fuelbed depth (m)	35	0.05	0.18	0.10	0.03
Fuelbed load (kg m^{-2})	35	0.30	1.20	0.62	0.24
Packing ratio	35	0.01	0.02	0.01	0.00
Rate of spread (m s^{-1})	31	0.004	0.17	0.05	0.03
Fireline intensity (kW m^{-1})	32	23.56	1004.41	454.7	233.25
Reaction intensity (kW m^{-2})	32	137.86	911.08	372.5	178.99
Reaction time (s)	34	9.00	91.00	33.3	17.21
Predicted flame height (m)	35	0.37	1.37	0.9	0.29
Flame depth (m)	32	0.04	6.26	1.6	1.33
Peak $q_{r\,01}$ (kW m^{-2})	34	1.10	47.18	21.3	13.23
Peak $q_{c\,01}$ (kW m^{-2})	34	10.89	106.48	67.8	25.12
Peak $q_{r\,11}$ (kW m^{-2})	34	0.59	24.21	11.1	6.24
Peak $q_{c\,11}$ (kW m^{-2})	34	2.21	34.89	10.6	7.04
$\Sigma_{peak}q_{r\,01}$ (kW m^{-2})	34	12.05	539.10	202.5	138.00
$\Sigma_{peak}q_{c\,01}$ (kW m^{-2})	34	232.10	1093.96	607.6	215.47
$\Sigma_{peak}q_{r\,11}$ (kW m^{-2})	34	4.86	373.87	191.9	85.50
$\Sigma_{peak}q_{c\,11}$ (kW m^{-2})	34	32.26	460.36	153.4	91.36
$\Sigma q_{r\,01}$ (kW m^{-2})	34	32.02	1311.42	581.3	395.82
$\Sigma q_{c\,01}$ (kW m^{-2})	34	728.93	2979.10	1795.2	641.05
$\Sigma q_{r\,11}$ (kW m^{-2})	34	2.93	1350.30	565.9	308.08
$\Sigma q_{c\,11}$ (kW m^{-2})	34	119.58	988.36	378.3	184.11
Max $\Delta q_{r\,01}$ (kW m^{-2} s^{-1})	34	0.07	8.82	2.93	2.22
Max $\Delta q_{c\,01}$ (kW m^{-2} s^{-1})	34	2.17	15.48	9.24	3.33
Max $\Delta q_{r\,11}$ (kW m^{-2} s^{-1})	34	0.10	4.72	1.25	1.13
Max $\Delta q_{c\,11}$ (kW m^{-2} s^{-1})	34	0.28	5.32	2.01	1.46
Av. $\Delta q_{r\,01}$ (kW m^{-2} s^{-1})	34	0.04	3.36	1.00	0.78
Av. $\Delta q_{c\,01}$ (kW m^{-2} s^{-1})	34	0.10	5.85	2.21	1.28
Av. $\Delta q_{r\,11}$ (kW m^{-2} s^{-1})	34	0.00	0.91	0.27	0.21
Av. $\Delta q_{c\,11}$ (kW m^{-2} s^{-1})	34	0.03	0.83	0.21	0.16

Table 2.2. Pearson correlation coefficients (df = 20) between environmental and fire behavior variables and upward heat flux characteristics. (* - p < 0.05; ** - p < 0.01)

Parameters	U	w_a	R	I_B	I_R	τ_r	D_F	Pred. H_F
Peak $q_{r\,01}$	0.443*	0.208	0.517*	0.594**	-0.340	0.364	0.504*	0.151
Peak $q_{c\,01}$	0.337	0.492*	-0.023	0.329	-0.477*	0.639**	0.281	0.703*
Peak $q_{r\,11}$	0.446*	0.774**	0.110	0.556**	-0.368	0.665**	0.307	0.851**
Peak $q_{c\,11}$	0.115	0.908**	-0.133	0.335	-0.151	0.613**	0.126	0.639**
$\Sigma_{peak}\,q_{r\,01}$	0.042	0.344	0.065	0.185	-0.079	0.247	0.137	-0.199
$\Sigma_{peak}\,q_{c\,01}$	0.376	0.472*	-0.087	0.323	-0.350	0.571**	0.293	0.669**
$\Sigma_{peak}\,q_{r\,11}$	0.241	0.700**	-0.104	0.317	-0.307	0.570**	0.097	0.871**
$\Sigma_{peak}\,q_{c\,11}$	0.025	0.935**	-0.277	0.208	-0.010	0.587**	0.051	0.496*
$\Sigma\,q_{r\,01}$	0.150	0.387	0.131	0.346	-0.125	0.343	0.269	-0.056
$\Sigma\,q_{c\,01}$	0.276	0.703**	-0.179	0.337	-0.339	0.716**	0.257	0.776**
$\Sigma\,q_{r\,11}$	0.273	0.925**	-0.081	0.463*	-0.199	0.690**	0.223	0.794**
$\Sigma\,q_{c\,11}$	-0.009	0.967**	-0.296	0.196	-0.057	0.598**	0.015	0.530*
Max $\Delta q_{r\,01}$	0.489*	0.289	0.535*	0.658**	-0.383	0.429*	0.532*	0.325
Max $\Delta q_{c\,01}$	0.199	0.489*	-0.128	0.195	-0.332	0.495*	0.079	0.689**
Max $\Delta q_{r\,11}$	0.283	0.824**	0.038	0.490*	-0.271	0.625**	0.242	0.761**
Max $\Delta q_{c\,11}$	0.241	0.781**	0.014	0.471*	-0.268	0.608**	0.266	0.644**
Av. $\Delta q_{r\,01}$	0.712**	-0.181	0.83**	0.718**	-0.513*	0.264	0.690**	0.186
Av. $\Delta q_{c\,01}$	0.521*	0.264	0.141	0.432*	-0.621**	0.578**	0.383	0.776**
Av. $\Delta q_{r\,11}$	0.748	0.379	0.532*	0.778**	-0.509*	0.541**	0.583**	0.721**
Av. $\Delta q_{c\,11}$	0.240	0.599**	0.049	0.393	-0.350	0.554**	0.246	0.696**

Although D_F, the variable describing the size of both the radiating surface and convective heat source, could be expected to be well related with the upward heat flux descriptors, such did not happen. The lack of significant relationships between D_F and the measures of upward heat flux could result from two possible sources. D_F is essentially a function of ROS or/and τ_r, although it is expected that changes in these two variables affect upward heat fluxes differently. It can be expected that an increase in D_F due to higher τ_r results in higher upward heat flux, as pointed out above. Higher ROS might result in a substantial increase in D_F, although the overall energy released in

148

flaming combustion per unit area remains comparable. This might explain why D_F was significantly correlated with only four of the 20 heat flux measures analyzed. There might also be a bias introduced by the experimental setup. The forced flow occurring under the high wind speed fires, which cause the highest ROS and consequently D_F, induce forced flow conditions in the wind tunnel that cause the dissipation of convective energy at higher rates than what would occur under natural conditions.
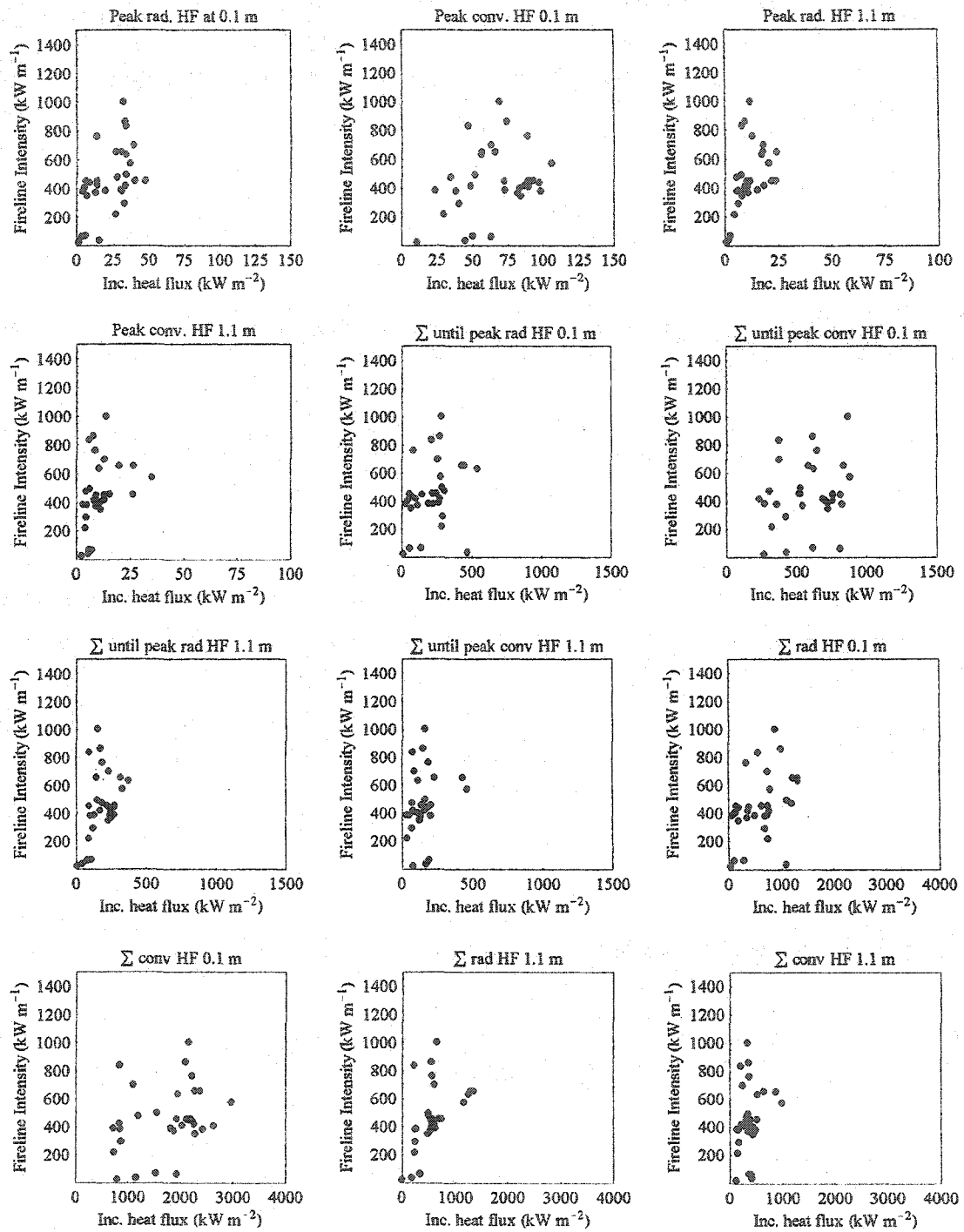
Figure 2.3. Relationship between fireline intensity and heat flux quantities measured in FiSL experimental fires
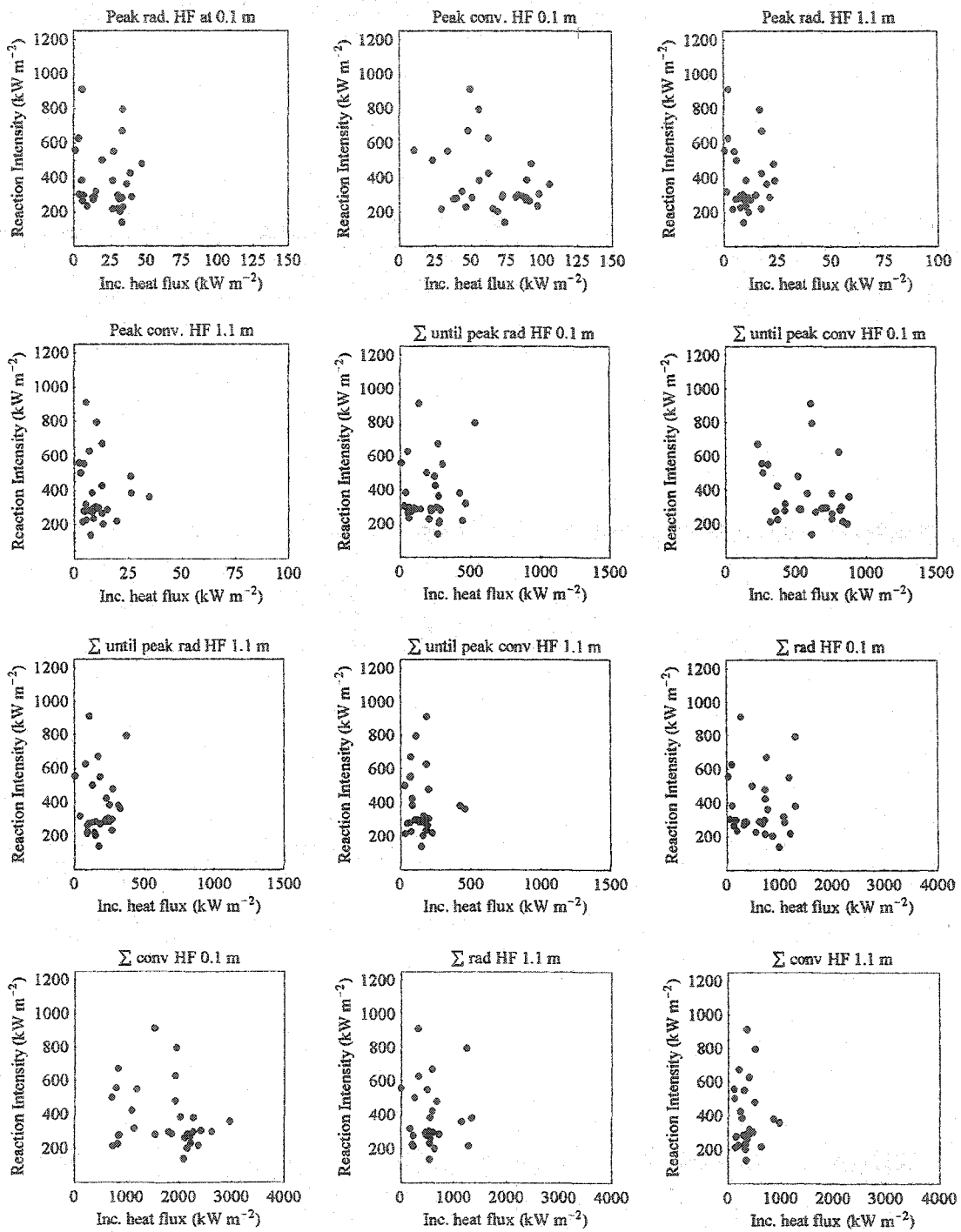
Figure 2.4. Relationship between reaction intensity and heat flux quantities measured in FiSL experimental fires

151
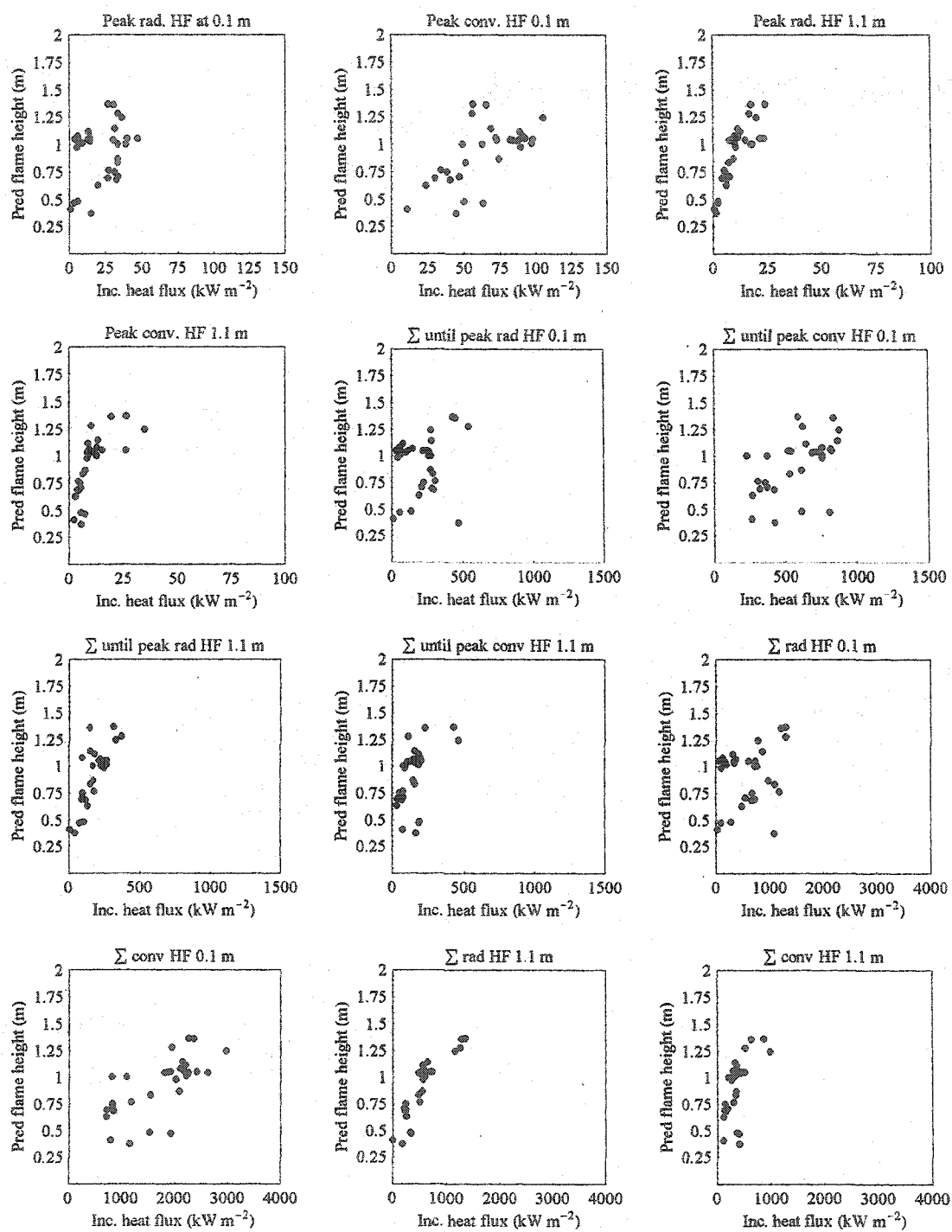
Figure 2.5. Relationship between predicted flame height and heat flux quantities measured in FiSL experimental fires.
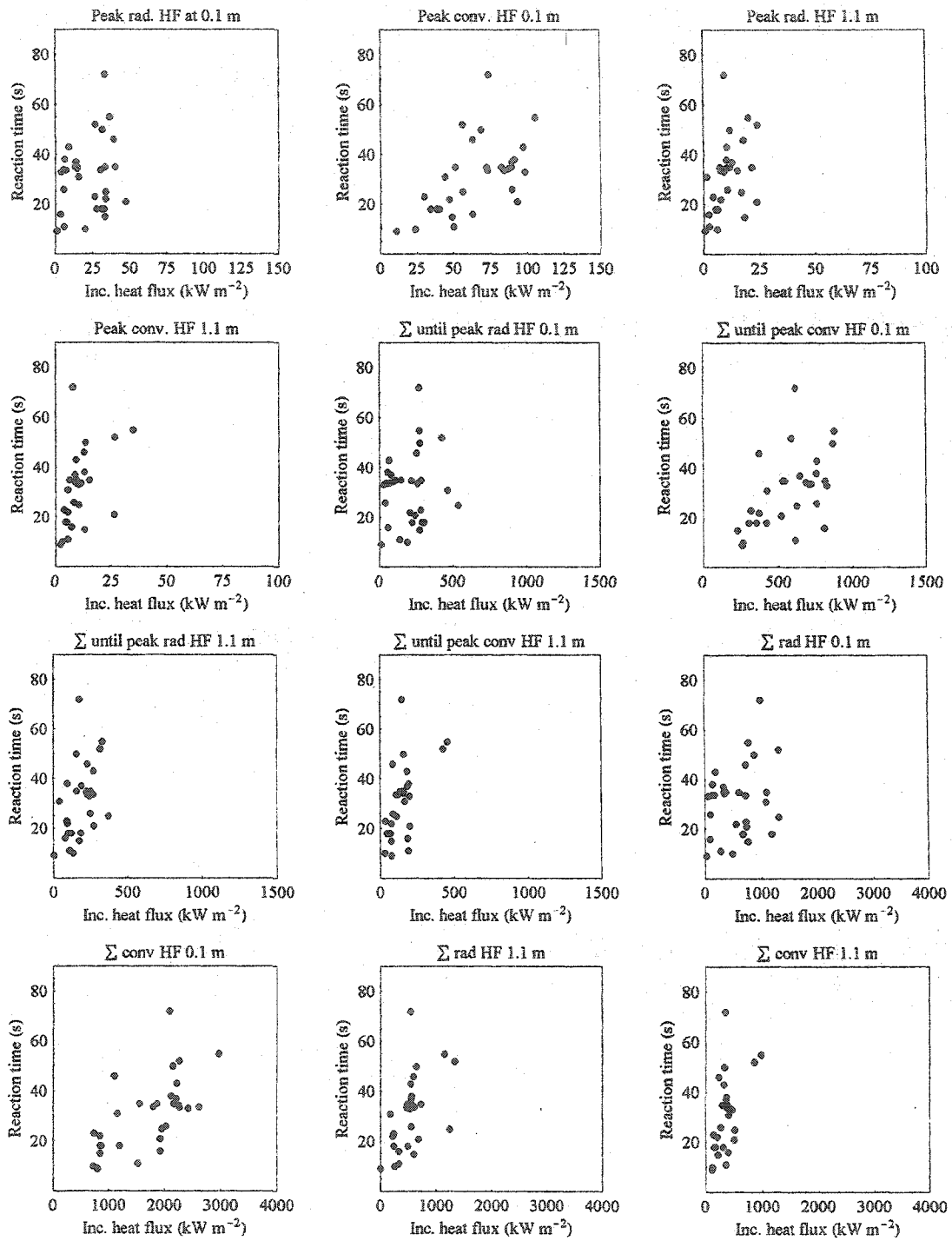
Figure 2.6. Relationship between reaction time and heat flux quantities measured in FiSL experimental fires
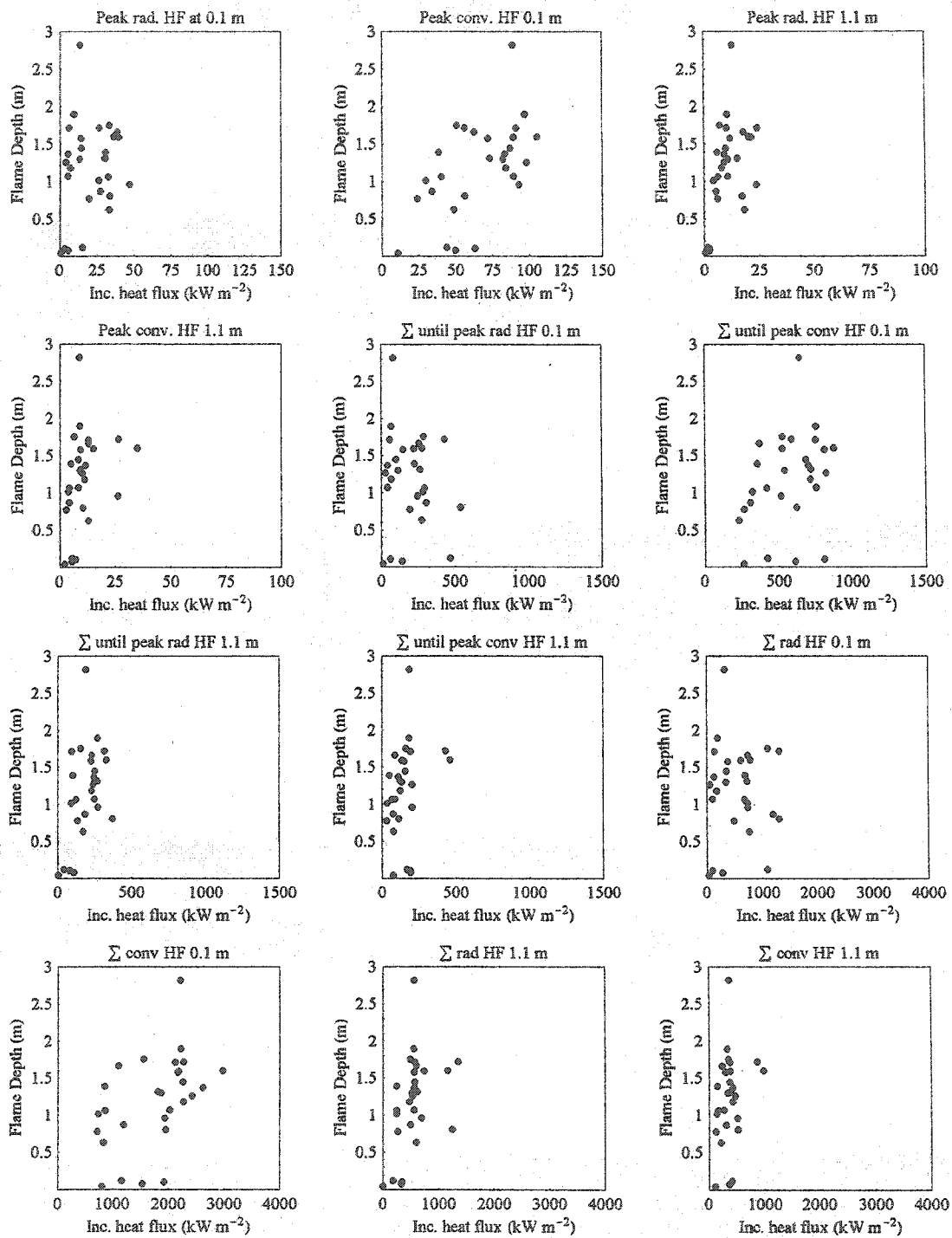
Figure 2.7. Relationship between flame depth and heat flux quantities measured in FiSL experimental fires

154

2.3.2. Outdoor fire analysis

Contrary to the laboratory fires, conducted under controlled conditions, the field fires were influenced by several factors that induced unsteady fire propagation, namely: (1) heterogeneous surface fuelbed structure; (2) wind speed and direction fluctuations (making the fire passing under the measuring tower as a flank fire); and (3) ignition pattern. These factors conditioned the observed incident radiative and convective heat flux measurements, resulting in time-heat flux profiles that should be analyzed in light of the spread and burning conditions. Fuelbed structure, fuel moisture content, wind speed and observed rate of spread and intensity for the outdoor experimental fires are given in Table 2.3 and 2.4. Measurement of the flow velocities within the flame and buoyant plume with the Kiel-static probes was restricted to a few fires. The complexity of the Kiel-static probes resulted in various problems while assembling the probes in the field, namely physical damage to probe components, incorrect assemblage of the probes, and other unknown causes. Only two measurements of the 24 made were considered satisfactory. Given the small number of collected airflow data it will not be analyzed here.

Gestosa fires

The combination of large available fuel loads and dry conditions resulted in high intensity fire propagation for the Gestosa experimental fires. Of the four experimental fires documented, plots 519 and 613, provided the best data for analysis as the fire passed directly under the sensor tower spreading as a head fire (Fig 2.8 and 2.11). The experimental fire associated with plot 517 (Fig. 2.9) was the result of two interacting

155

flame fronts, and although the main flame front passed the sensor tower as an head fire, flame and plume flow were affected by the additional flame front. Smoke precluded the determination of the rate of spread for this fire. For the Gestosa 517 and 519 plots, flame height occasionally exceeded the instrument tower height (e.g., Fig. 2.1). This limited analysis of the buoyant plume behavior as all sensors were within the fire plume. The experimental fire associated with plot 605 passed under the sensor tower spreading as a flank fire. Video evidence from this fire suggest a thin burning zone with a flame height lower than the height of the lower sensor. No reliable estimate of rate of spread was obtained for this fire. The low intensity characteristics of this fire made it very responsive to fuel and wind variations. The unsteady fire propagation created various peaks in the heat flux traces measured at 3.0 and 4.5 m (Fig. 2.10).

Paredes fires

The two Paredes fires provided the best fire propagation characteristics to evaluate the radiative heat flux and buoyant plume models. Both fires spread at a reasonably steady state with flame heights between 2-2.5 m. Given these characteristics, the measuring tower sampled both the flaming zone and the buoyant plume. Unfortunately, all heat flux sensors but one (radiative heat flux at 4.5 m) failed to capture data due to cable damage. Data from these two fires was then restricted to the above mentioned radiative heat flux and plume temperature at four heights (i.e., 1.0, 1.5, 3.0 and 4.5 m).

156

TCEF prescribed fires

The two TCEF prescribed fires were low intensity fires. Marginal burning conditions and fuelbed characteristics, namely a compacted litter layer, limited fire spread in TCEF10. This situation required an area fire ignition pattern, which created distinct heat flux patterns from what would be expected above a single moving flame front. For this prescribed fire, incident radiative heat flux was moderate and maintained through time (Fig 2.12). The measured convective heat fluxes were small, with the lower sensor registering a peak corresponding to active flaming combustion occurring directly beneath the instrument tower.

In TCEF16 the fire front passed the measuring tower as a head fire (Fig 2.1 and 2.13) displaying unsteady fire behavior characteristics. This was the result of flame front sensitivity to changes in wind speed. Convective heat fluxes (Fig. 2.13b) and temperature traces were characterize by several peaks following gusts in wind speed, while the radiative heat flux traces were mostly insensitive to changes in the flame front characteristics.

2.3.2.1. Analysis of upward heat fluxes

The measurement of incident heat flux data above outdoor surface fires had the objective of quantifying the radiative and convective heat flux parameters under conditions that could not be attained in laboratory fires, namely higher fuel loads, higher fireline intensities and deeper flame depths. The field data could then be used to analyze the representativeness of the laboratory data in replicating some of the features observed in the high-intensity outdoor experimental fires. As for the laboratory fire data, in order

157

to smooth local fluctuations in the collected heat flux and fluid temperature data, the collected time series data was linearly transformed using a simple 5-s moving average. The smoothing procedure when applied to the outdoor experimental fires did not result in smooth traces such as the ones observed for the laboratory fires. Because any further smoothing would limit comparison with the laboratory data, the decision was made to limit the smoothing of the heat flux data to the 5-s moving average. Fig. 2.8 through 2.13 present the typical time dependent radiative and convective heat flux traces for the outdoor experimental fires, excluding the Paredes fires.

The analysis of the upward incident radiative and convective heat flux traces do not hint at any general trends where the peak heat fluxes should occur or which heat transfer process dominates. The measured heat fluxes are the result of chemical and physical processes that determine fire characteristics such as the rate and amount of volatile production, how and where these volatiles react with oxygen which eventually determine the size of the flaming zone. The combination of these factors will determine the heat distribution around and above a fire front.

For the Gestosa 517 and 519 experimental fires, the highest incident radiative heat flux were measured in the instruments located at 4.5 m, whereas the remaining fires showed the inverse trend (i.e., decrease in measured incident heat flux with height above fire). The trend of experimental fires 517 and 519 arises possibly due to the combination of a large flaming zone and the relative position of the sensors in relation to the space occupied by the flame itself. The sensors located lower in the tower (for these two fires the lower sensors height coincide with the top of the fuelbed) have a limited field of view and thus, "see" less of the flaming zone. The instruments located higher in the tower will

158

"see" the fire earlier and will also "see" more of the fire. For the lower intensity fires, with a smaller flaming zone depth, all sensors would be seeing similar amounts of fire front regardless of their position. In this situation, there will be a decrease in the measured incident radiative heat flux with height. The same pattern was observed for some of the laboratory fires. As expected from the effect of air entrainment into the fire and buoyant plume, convective heat flux decreased with height for all fires.

The ratio between convective and radiative heat fluxes (C/R) was used to analyze the prevalence of one heat transfer mechanism (as measured by the sensors) over the other for the various outdoor experimental fires. No definitive trend was found for the C/R ratios based either on the peak (Table 2.5) or cumulative (Table 2.6) heat fluxes. The shape of the heat flux profile (Figs. 2.8 – 2.13), and consequently the C/R ratios, vary with burning conditions and fire behavior properties such as flame geometry (both height, tilt angle and depth) and buoyant plume characteristics, and the sensor location relative to the flame front. For some experimental fires (e.g., Gestosa 517, 519 and TCEF10) the C/R ratio of the peak heat flux was above 1.0 at the lowest measuring height, and decreased to values below 1.0 for the top instrument (Table 2.5). Data from the Gestosa 605 and 613 experimental fires showed an opposite trend. More consistent values were obtained from the analysis of the C/R ratios based on cumulative values. This could be expected as these ratios are integrating the heat released by the fire over an extended period, while the peak based values might be the result of a short fluctuation in fire dynamics and not be representative of the fire's general behavior. Overall, it seems that the farther the instrument was from the flame the lowest the C/R ratio. For the three fires with a flame height smaller then the lower instrument (i.e., Gestosa 605, TCEF10

159

and TCEF16) the C/R ratios based on cumulative values were always above 1.0 (Table 2.6). For the three fires in which the lower instruments were just above the top of the fuelbed (i.ei., Gestosa 517, 519 and 613) the lowest heat flux sensors indicated C/R ratios above 1.0. The way this ratio changed with height differed for the three fires, possibly as a results of flame and plume characteristics.

Given the restricted number of fires being analyzed we can only conjecture as to the dominance of one heat transfer mechanism when considering upward heat fluxes above spreading surface fires. The evidence suggests that the C/R ratios will depend on flame and plume geometry, and where the measurement is made relative to flame size and location.

Because the sensors in the two TCEF fires were located much higher relative to the top of the surface fuelbed than in the laboratory fires, comparison between the field and laboratory data is restricted to the shrubland fires. Both peak and cumulative incident radiative heat fluxes measured in the shrubland fires were higher then the values observed in the laboratory experiments. This could be expected from the reasoning that it is the size of the flaming zone that will determine these two quantities. Higher peak convective heat fluxes were measured in the laboratory fires. Convective heat transfer, being a function of local gas temperature and velocity, is dependent on factors determining these two quantities. The existence of larger amounts of water content in the live fuels, which upon release will lower the temperature of the gases in the flame could account for these differences. The measured cumulative convective heat flux until peak was higher for the shrubland fires then for the laboratory fires, possibly due to the higher amount of fuel being consumed in flaming combustion in the field fires.

160

Table 2.3. Pre-fire surface fuelbed structure for the Gestosa, Paredes and Tenderfoot experimental fire plots.

Fire name	Fuelbed height (m)	Fuel load (kg m^{-2})						
		Litter	1 hr	10-hr	100-hr	1000-hr	Duff	Live fuels
Gestosa 517	1.67	-	-	-	-	-	-	4.8[1]
Gestosa 519	1.44	-	-	-	-	-	-	4.9[1]
Gestosa 605	0.78	-	-	-	-	-	-	3.2[1]
Gestosa 613	1.12	-	-	-	-	-	-	4.3[1]
A4	0.48	-	0.87	0.01	-	-	-	1.8
Aextra	0.46	-	0.96	0.01	-	-	-	2.0
TCEF10	NM	1.55	0.16	1.21	0.41	4.1	7.45	-
TCEF16	NM	0.71	0.03	0.24	0	1.71	1.57	-

[1] Include both dead and live fine fuels; NM – not measured;

Table 2.4. Dates, burning times, weather conditions and fire behavior for the Gestosa, Paredes and Tenderfoot experimental fires.

Outdoor experimental fire	Date of burning	Time of burning (hhmm)	Slope (°)	Temp. (°C)	RH (%)	U_{10} (m s^{-1})	Fuel moisture (% oven dry weight)				ROS (m min^{-1})	I_B (kW m^{-1})
							Live	1-hr	10-hr	100-hr		
Gestosa 517	May 30, 2002	1126	13	20	27	1.7	86	9.1	-	-	NA	-
Gestosa 519	May 31, 2002	1432	12	24	44	2.8	71	7.5	-	-	8.6	14700
Gestosa 605	May 16, 2003	1515	15	20	47	4.5	93	10	-	-	NA	-
Gestosa 613	May 17 2003	1650	18	26	37	5.3	89	8.4	-	-	6.5	9500
A4	May 23, 2003	1619	11	29	22	4.4	101	5.4	-	-	9.9	6300
Aextra	May 23, 2003	1458	20	28	22	3.5	101	7.8	-	-	8.6	5900
TCEF10	Sep. 11, 2002	1830	0	16	40	0	-	13	14	20	NA	-
TCEF16	Sep. 12, 2002	1530	0	19	32	2/-	-	9	11	16	12	300
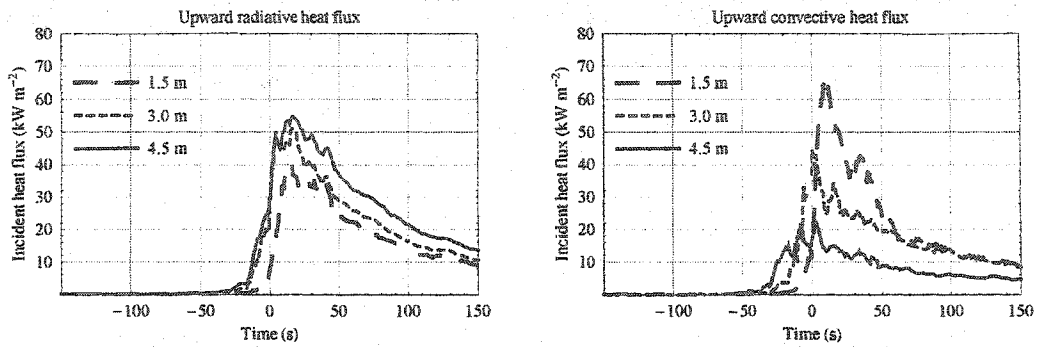
Figure 2.8. Upward incident radiative and convective heat fluxes for Gestosa plot 517 experimental fire.
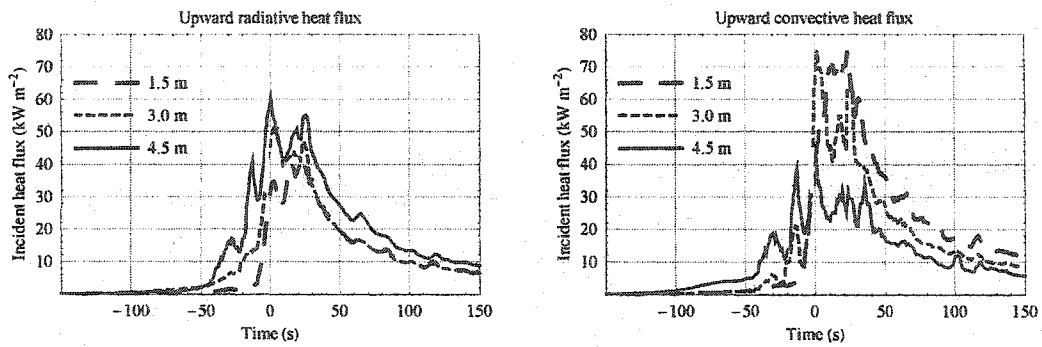
Figure 2.9. Upward incident radiative and convective heat fluxes for Gestosa plot 519 experimental fire.
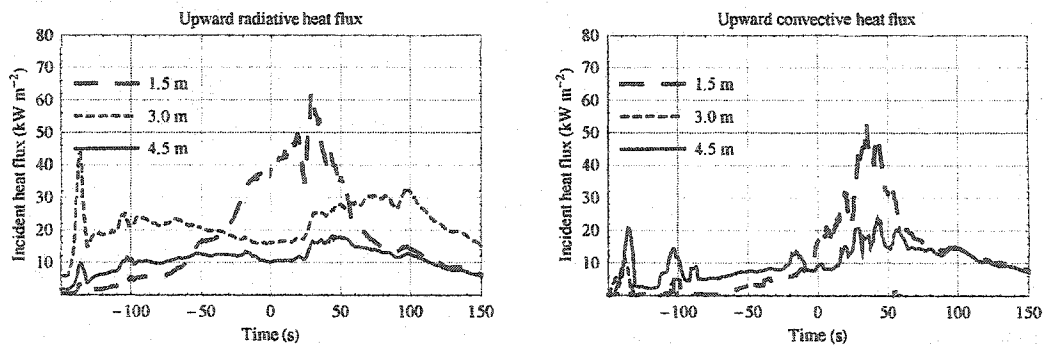
Figure 2.10. Upward incident radiative and convective heat fluxes for Gestosa plot 605 experimental fire.

162
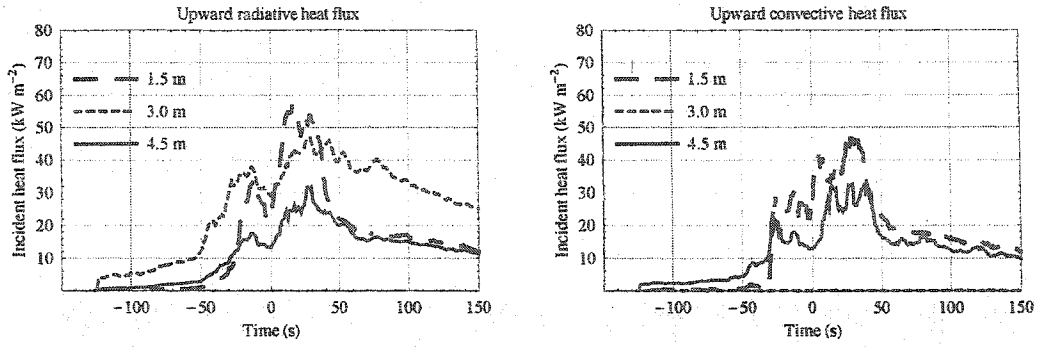
Figure 2.11. Upward incident radiative and convective heat fluxes for Gestosa plot 613 experimental fire.
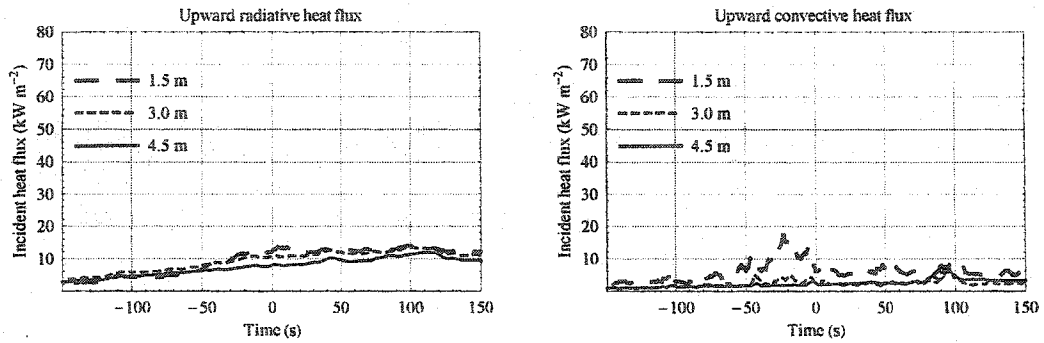
Figure 2.12. Upward incident radiative and convective heat fluxes for TCEF10 prescribed fire.
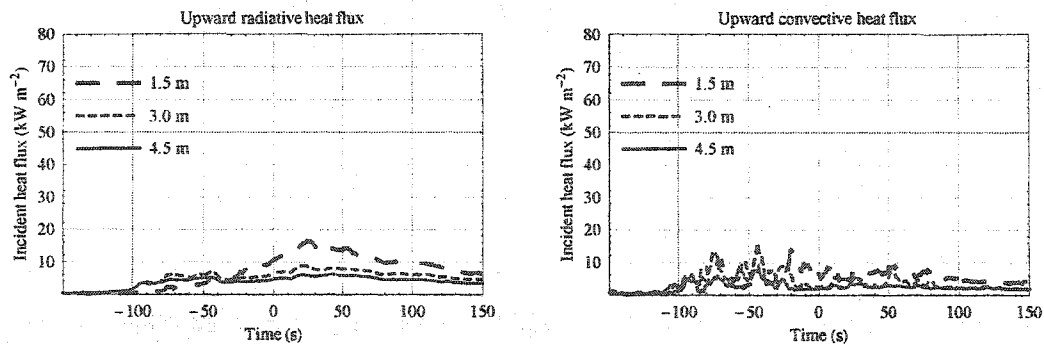
Figure 2.13. Upward incident radiative and convective heat fluxes for TCEF16 prescribed fire.

163

Table 2.5. Peak upward radiative and convective heat fluxes (kW m^{-2}) measured in the Gestosa, Paredes and TCEF outdoor experimental fires.

Parameter	Outdoor experimental fire							
	517	519	605	613	Aex	A4	U10	U16
Heat flux at 1.5 m (kW m^{-2})								
Radiative	39.6	42.2	62.3	57			14.0	16.4
Convective	65.4	74.3	53.1	46.6			17.4	13.9
Conv/rad	1.6	1.8	0.9	0.8			1.2	0.8
Heat flux at 3 m (kW m^{-2})								
Radiative	51.1	51.2	43.6	46.9			13.3	8.9
Convective	44.5	75					7.8	16
Conv/rad	0.9	1.5					0.6	1.8
Heat flux at 4.5 m (kW m^{-2})								
Radiative	54.7	60.1	18.2	31.8	30.8	25.1	11.8	6
Convective	21.8	43.7	23.7	32.5			5.8	8.5
Conv/rad	0.4	0.7	1.3	1.0			0.5	1.4

Table 2.6. Cumulative upward radiative and convective heat fluxes (kW m^{-2}) until peak measured in the Gestosa, Paredes and TCEF outdoor experimental fires.

Parameter	Outdoor experimental fire							
	517	519	605	613	Aex	A4	U10	U16
Heat flux at 1.5 m (kW m^{-2})								
Radiative	482	1010	3113	1522			2292	720
Convective	537	1802	1395	1842			748	387
Conv/rad	1.1	1.8	0.4	1.2			0.3	0.5
Heat flux at 3 m (kW m^{-2})								
Radiative	981	732	592	3029			2410	673
Convective	422	718					554	432
Conv/rad	0.4	1					0.2	0.6
Heat flux at 4.5 m (kW m^{-2})								
Radiative	1123	1096	2102	1268	1274	631	1988	528
Convective	500	1088	1684	1575			492	324
Conv/rad	0.4	1.0	0.8	1.2			0.2	0.6

2.3.3. Modeling results

2.3.3.1. Radiative heat transfer model

Simulations of the upward incident radiative heat flux were sensitive to the fuelbed and environmental variables determining the strength and size of the radiative heat source, namely U, MC, w_a and D_F. The simulations for evaluation of the radiative heat transfer model relied on the estimation of the required fire behavior input variables, with the exception of ROS. D_F was estimated from the product of τ_r (estimated from Nelson's (2003b) model) and ROS (observed for experimental fire). Fig. 2.14 shows the predicted and observed incident radiative heat fluxes above the fuelbed for four FiSL experimental fires. The model tended to overpredict the peak incident radiative heat flux (Fig. 2.15a). This seemed to occur mostly due the inability to correctly define w_a and a consequent overprediction trend in τ_r. The model overpredicted incident radiative heat flux for the fires with a fuel load (and consequently an assumed w_a) of 1.2 kg m^{-2}, and fires with compacted fuelbeds (packing ratio of 0.032). For each of these situations, restrictions to air flow within the fuelbed reduce the amount of total fine fuel that is consumed in flaming combustion. This was supported by observation in these fires of substantial glowing combustion after the passage of the flame front. Nevertheless, no objective method presently exists to estimate w_a, and the total surface fuel load was used to run the radiative model. This overestimation of w_a induced a higher τ_r and consequently a larger radiative heat source, leading to the overprediction of the peak radiative heat flux for these situations.

The model appears to adequately predict both measures of cumulative radiative heat transfer (Figs. 2.15b and 2.15c). Nevertheless, the model tended to slightly

165

overpredict the cumulative radiative heat flux until peak and underpredict the total cumulative radiative heat flux. The four large overpredictions evident in Fig 2.15b (between 600 and 800 kW m^{-2}) were for the experimental fires with w_a of 1.2 kg m^{-2}. Of the radiative heat flux quantities analyzed, the cumulative radiative heat flux until the peak is reached seems to be the most important. This quantity will determine the bulk heat transferred to the fuel particles in the canopy while (1) convective heat transfer is concurrent and (2) incident heat fluxes to the fuel particles are still at intensities that could result in the fuel particle reaching ignition temperature.
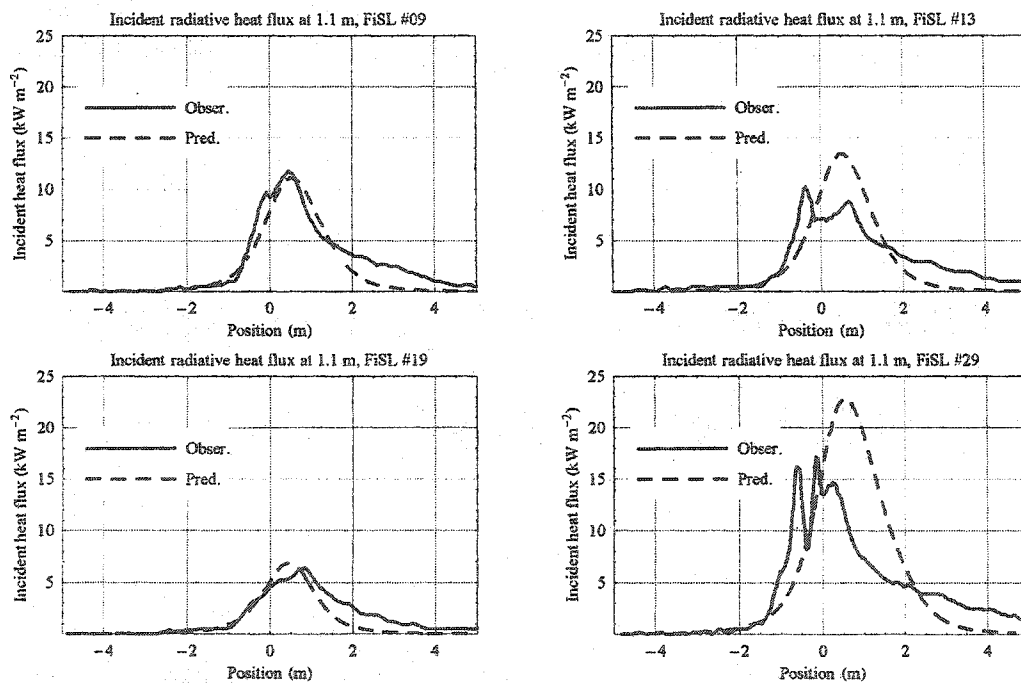
Figure 2.14. Comparison between observed and predicted upward incident radiative heat flux 1.1 m above the top of the fuelbed in laboratory experimental fires. The 0 m position in the x-axis indicates the location of the ignition interface. Specific fire characteristics are given in Table A.1.

166
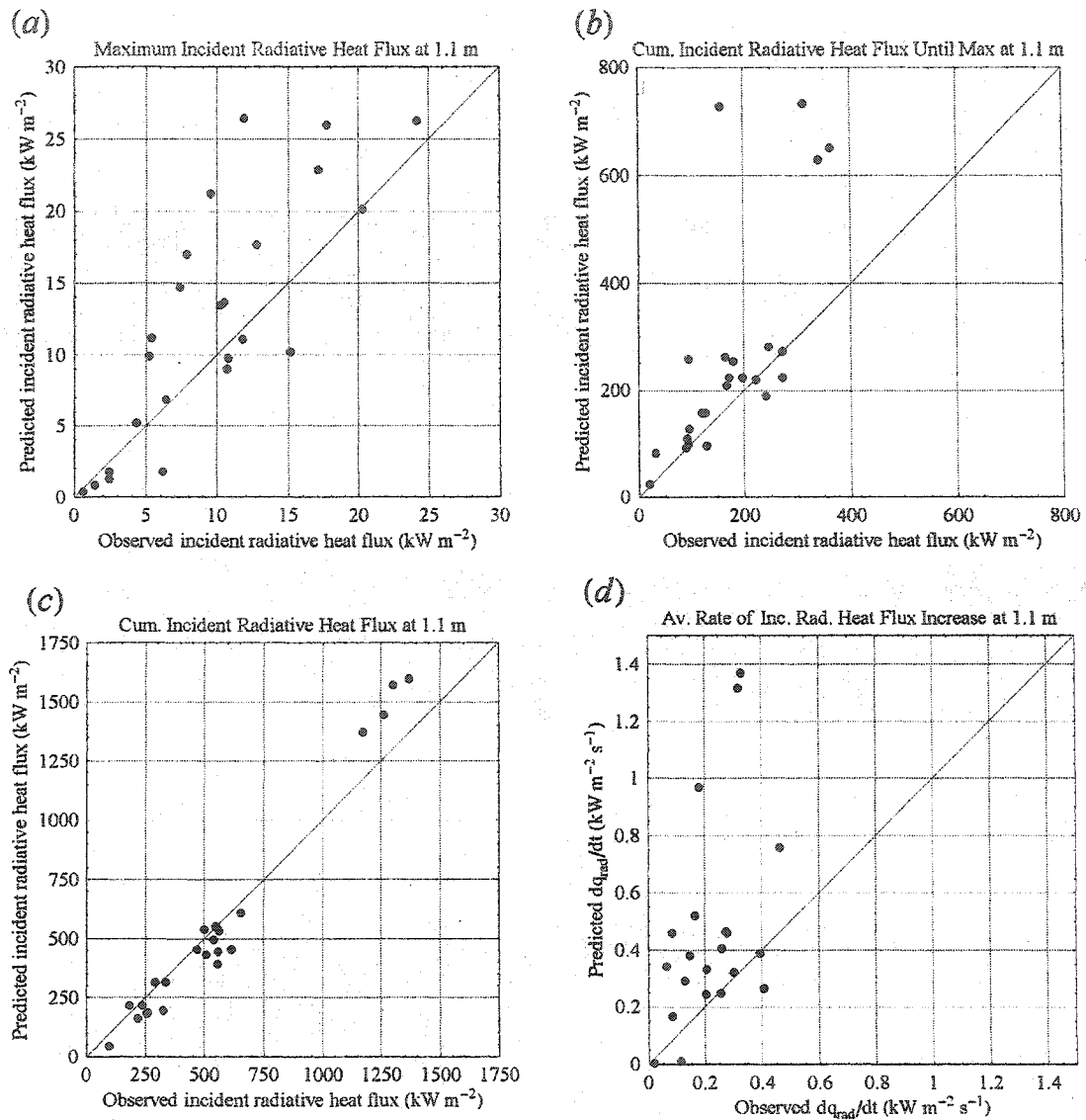
Figure 2.15. Relationship between observed and predicted upward incident radiative heat flux quantities. *(a)* maximum incident radiative heat flux; *(b)* cumulative incident radiative heat flux until point of maximum incident radiative heat flux; *(c)* cumulative incident radiative heat flux; and *(d)* average rate of increase in incident radiative heat flux.

The average derivative of the radiative heat flux curve was the quantity most poorly explained by the model (Fig. 2.15d). This might be derived from the turbulent nature of the flame. The average rate of change in radiative heating was found to be highly variable, probably dependent on small scale unsteady phenomena in the flame

167

front, such as evidenced by flame flickering. Such behavior was not expect to be explained by the radiative heat transfer model assumptions of steady-state propagation and time-constant irradiance properties.
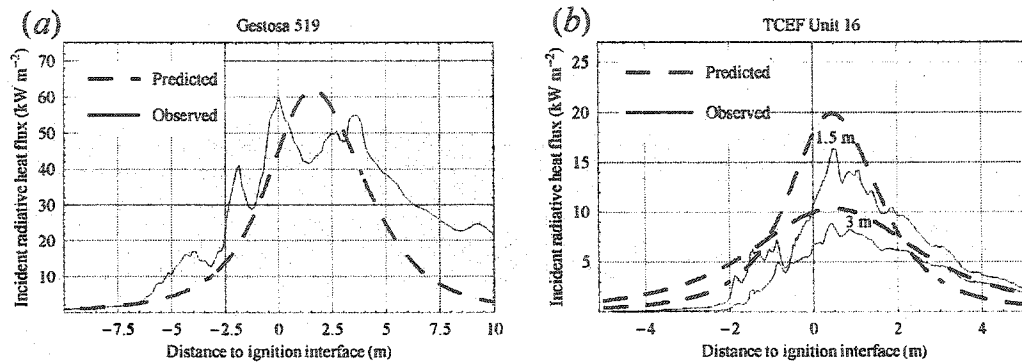
Figure 2.16. Comparison between observed and predicted upward incident radiative heat flux for two outdoor experimental fires. *(a)* Observed and predicted incident radiative heat flux 3 m above the top of fuelbed for Gestosa 519; and *(b)* Observed and predicted incident radiative heat flux 1.5 and 3.0 m above the top of fuelbed for TCEF16 prescribed fire.

The comparison of the observed and predicted radiative heat fluxes was restricted to the fires for which rate of spread information existed (Table 2.4). As a result of the nonsteady fire propagation conditions and pulses in flame activity the radiative heat flux traces from the field fires showed higher variability (Fig. 2.16a and 2.16b) than results obtained in the laboratory fires. The radiative heat flux model seems to reproduce adequately the timing of peak radiative heat flux, occurring some distance behind the ignition interface (i.e., the 0 m position on the x-coordinate of Fig. 2.16). In regards to the radiative heat transfer model adequacy to predict peak radiative heat flux and cumulative radiative heat flux until peak, the results from the application of the model to the outdoor experimental fires produced trends similar to what was found for the laboratory fires (Figs. 2.17a and 2.17b). The model adequately predicted the peak

168

radiative heat flux at the 1.5 and 3 m heights for TCEF16 and reveals larger overpreditions for the shrubland fires. Possible explanations for the overprediction trend in the shrubland fires were the difficulty in defining w_a and the effect of large quantities of water vapor being released prior to the arrival and within the flame. The water vapor is expected to cool flame temperatures and absorb radiation, phenomena that are not taken into account in Equation 2.4 model. In high-intensity fires, such as some of the shrubland fires used in the analysis, oxygen deficiencies within the flame result in the formation of large quantities of soot. Accumulation of soot in the radiometer window will change its optical properties and consequently the amount of radiation sensed, which will be lower than what an unobstructed window would measure.
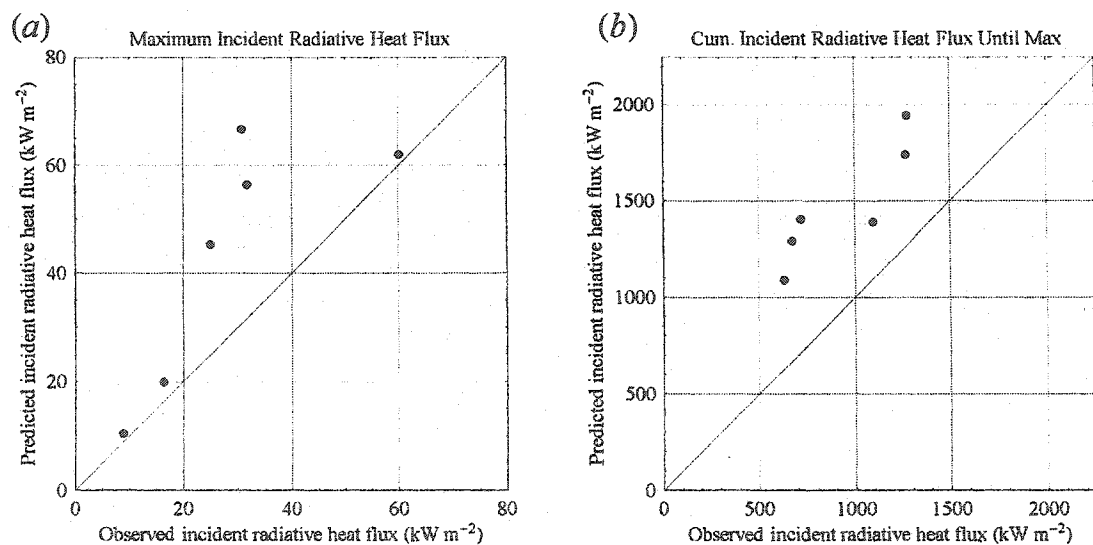
Figure 2.17. Relationship between observed and predicted upward incident radiative heat flux quantities for outdoor experimental fires. (a) maximum incident radiative heat flux; (b) cumulative incident radiative heat flux until point of maximum incident radiative heat flux.

169

2.3.3.2. Plume characteristics

The evaluation of Mercer and Weber (1994) buoyant plume model implementation against the fire behavior data collected in this study was limited due to the characteristics of the laboratory and outdoor data collected. The data collected in the FiSL wind tunnel fires was not adequate to analyze plume characteristics as ceiling height and forced flow limits the development of natural buoyancy (Catchpole et al. 1993). In regard to the outdoor experimental fires, the two limiting factors were the inherent variability in fire characteristics and instrumentation placement relative to the flame dimensions. Only three fires (i.e., TCEF16, Paredes A4 and Aextra) meet the criteria of a spreading line fire with sensors above the fire plume. Reconstruction of plume structure, namely to obtain a snapshot of the plume air temperature and velocity distribution in space at an instant in time, based on measurements of those quantities in a single tower is impractical due to nonsteady fire rate of spread and turbulent plume characteristics, namely its unsteady dimensions, inclination and temperature/velocity distribution. The two Paredes outdoor experimental fires were spreading at a reasonable steady state while passing the instrument tower, but the flame and buoyant plume turbulence associated with the intensity of the fire generate too much variability in the time-temperature traces for possible reconstruction of the plume structure. Low-intensity fires (such as TCEF16) were very responsive to changes in burning conditions inducing unsteady fire characteristics and limiting also reconstruction of plume structure from point source data. These limitations associated with the detailed analysis of plume characteristics above outdoor fires were already identified by Gould et al. (1997).

The evaluation of the plume model against field data was restricted to the Paredes outdoor experimental fires (Figure 2.8). One difficulty in evaluating the plume model was the determination of flame height from field data. For the Paredes fires, flame height (H_F in Table 2.7) was estimated from the analysis of a series of still images obtained while the fire passed under the instrument tower. The still images, obtained from a vantage point and taken with a high shutter speed revealed a large variability in flame height, leading to an estimation of this quantity that might introduce substantial errors in the estimation of plume characteristics. For the estimation of plume characteristics for the Paredes fires, three other quantities were required: initial half width of the plume (b_i) initial plume centerline vertical velocity (U_{pi}), and initial plume temperature (assumed 800 K). The initial half width of the plume was assumed as half of the flame depth and initial plume centerline vertical velocity was estimated from Nelson's (2003a) model (eq. 1.12 in Chapter A). A logarithmic vertical wind profile was applied (Albini 1983) to describe the variation in wind velocity with height above the vegetation cover.

It is worth nothing that the present evaluation exercise is based on a number of quantities estimated with a corresponding large uncertainty. The evaluation should be considered relative to the plume model implementation in CFIM as described in Chapter 1, and not the plume model *per se*. An evaluation of the plume model would require better data quality than the data available for this study, namely higher data resolution and higher certainty in the quantities being estimated. As measures of model adequacy we analyzed temperature conservation along the plume centerline and plume width as described by time above 500 K at two instrument heights, 3.0 and 4.5 m. Malfunction of

the Kiel-static probes for these fires limited the analysis of air velocity distribution within the plume.
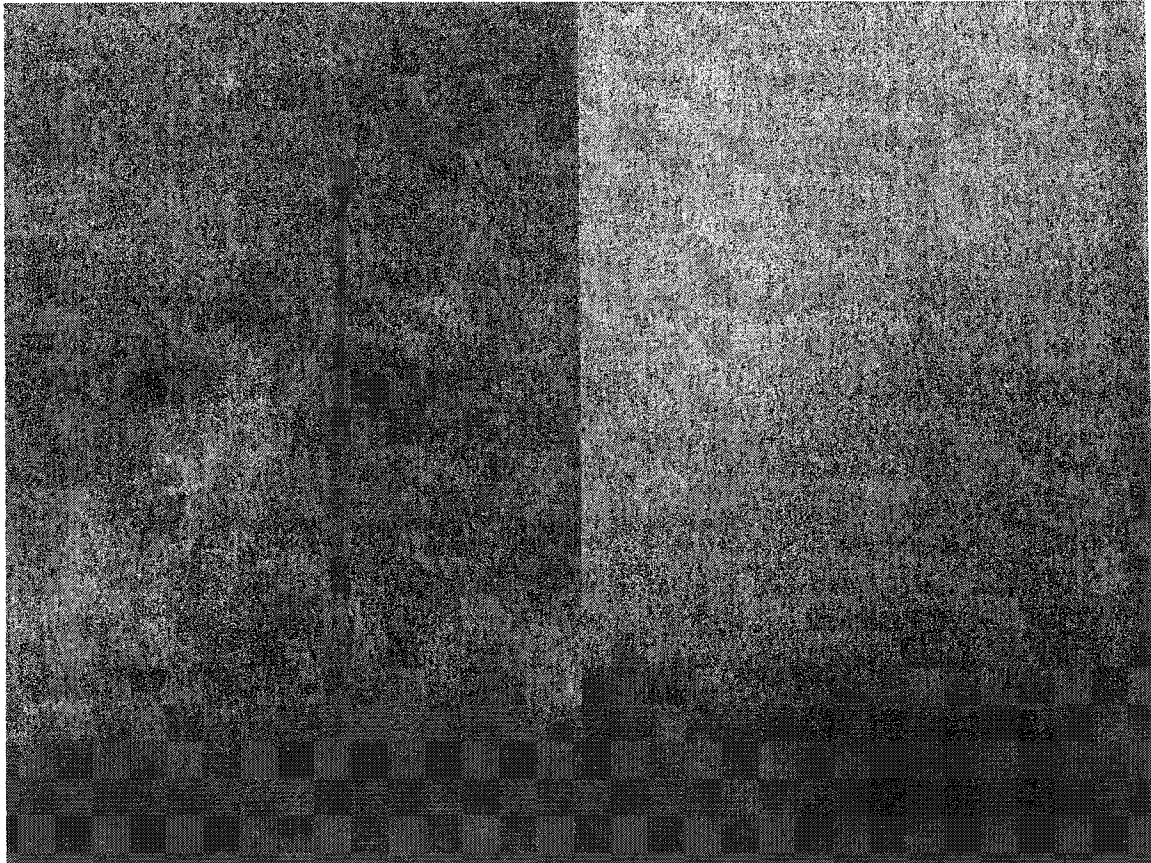
Figure 2.18. Photographs of flame front passing under instrument tower in the Paredes Aextra (left photograph) and Paredes A4 (right photograph) outdoor experimental fires.

For Paredes A4 experimental fire the plume model overestimated plume centerline temperatures at 3.0 and 4.5 m by respectively 112 and 79 K (Table 2.7). For Paredes Aextra the model underpredicted plume centerline temperature at 3 m by 7 K and overpredicted at 4.5 m by 44 K. The model overpredicted the temperature conserved in the plume (ΔT) for Paredes A4 fire by 22 K m^{-1}, and underpredicted for the Paredes Aextra fire by 34 K m^{-1}. The observed higher plume conservation for the Aextra fire compared to the A4 fire was expected because of the larger active burning zone in the

172

Aextra fire (9.1 versus 4.8 m). This should result in a larger plume width for the Aextra

fire and consequently higher conservation of temperature at the plume centerline. The

conservation of temperature between 3.0 and 4.5 m predicted by the model was similar

for both fires (Table 2.7), albeit the differences in plume initial half width (b_i).

Table 2.7. Buoyant plume input variables, observed and predicted plume characteristics at 3 and 4.5 m for Paredes A4 and Aextra outdoor experimental fires. Other fire behavior properties for these fires are given in Table 2.4.

Parameter	Paredes A4		Paredes Aextra	
H_F (m)	2		2.5	
U_{pi} (m s^{-1})	6.9		6.7	
b_i (m)	2.4		4.5	
Maximum plume centerline temperature (K)				
	Observed	Predicted	Observed	Predicted
3 m	596	708	773	766
4.5 m	527	606	624	668
ΔT (K m^{-1})	46	68	99	65
Time (s) above 500 K				
3 m	44	29	61	50
4.5 m	14	27	39	47

Regarding the time above 500 K, the plume model underpredicted this quantity at

3 m and overpredicted it at 4.5 m, for both of the Paredes fires (Table 2.7). As for the

plume centerline temperatures, the agreement between observed–predicted values was

better for Aextra fire. The underestimation for the observed values at 3.0 m could be

linked to the energy released in residual combustion. The high surface fuel loads of these

two plots (Table 2.3) should result in a substantial amount of fuels consumed in residual

(non-flaming) combustion. This was supported by the observe time-temperature profile

(not shown here) for these plots. By only considering the heat being release as a result of

flaming combustion, the plume model implementation will underestimate the time above

173

certain temperatures. This bias will be related to lower temperatures associated with the trailing edge of the plume. The overprediction of plume duration above 500 K at 4.5 m follows the overprediction trend at this height observed also for the plume centerline temperature (Table 2.7).

The application of plume model predictions to only two outdoor experimental fires limited the possible analysis of plume model components responsible for the differences between observed and predicted variables. Some error could be introduced while specifying the initial conditions, namely initial vertical velocity, plume half width and the assumed logarithmic vertical wind profile. The existence of large quantities of water in live fuels might also have influence in plume characteristics, namely the initial temperature of the plume and the thermodynamic properties of plume gases.

2.4. Conclusions

Upward radiative and convective heat fluxes were measured above spreading surface fires with the objective of better understanding the fuel, environment and fire behavior mechanisms determining them. The nature of the present study can be considered essentially exploratory. In addition to further our understanding of the relationships between commonly used fire behavior quantities (e.g., flame depth, reaction time, fireline intensity) and fundamental processes determining heat transfer to unburned fuels above spreading surface fires, the aim of the present study was to collect heat flux data that can be used to parameterize physical fire behavior models based on combustion and heat transfer processes.

174

The relationship between the various measures of upward heat flux with fireline intensity and reaction intensity were found to not be significant. The relevance of this finding comes from the fact that these two intensity measures are commonly used to describe the energy release rate of a fire. The inadequacy of fireline and reaction intensity in explaining upward heat flux components might come from how they are calculated and ultimately their meaning. The assumption that w_a was equal to the quantity of fine dead fuels used in the estimation of the fire intensity measures, without a separation of this quantity into how the fuel is consumed, and ultimately how heat is being generated, fail to give them the detail necessary to describe how energy is being released through time above a fire. Fireline intensity integrates the energy release rate per unit length of the fireline. It can be expected that the estimation of fireline intensity based on total fuel consumed produces a measure of fire intensity even less related to upward heat fluxes above the flaming zone. The present estimation of reaction intensity yields an average reaction intensity, while the rate of energy release per unit area in the fire front has been found to vary with time or location along the flame depth (Frandsen and Rothermel 1972). These results suggest that a more detailed measure of fire energy release rates is required to better understand and model relatively small-scale fire phenomena, such as the heating of canopy fuels. Reaction time and estimated flame height were the fire behavior variables most related with the upward heat fluxes.

The radiative heat transfer model slightly overpredicted peak and cumulative until peak incident radiative heat fluxes for both laboratory and outdoor experimental fires. One of the possible reasons for this overprediction trend might be due to the overestimation of what constitutes available fuel for flaming combustion. Given the

175

inexistence of a method to objectively estimate this quantity, the estimation of radiative heat fluxes was based on the assumption that all fine fuels were consumed in flaming combustion. This results in an overestimation of residence time, and consequently the size of the radiating surface. Considerable overprediction of incident radiative heat flux in the outdoor fires occurred only for the shrubland fires. Besides the possible effect of the overestimation of w_a, other possible factors contributing to the overprediction in the shrubland fires was the inability to account for the formation of larger quantities of soot in the fire plume and the existence of large quantities of water vapor in absorbing radiant heat. This distinction between the flame characteristics in full-scale outdoor experimental fires to the ones observed in the laboratory burns suggests care needs to be taken when extrapolating results from laboratory experiments to describe certain processes occurring in real world wildland fire situations. The scale of the laboratory fires might limit the development of certain phenomena determining certain combustion and heat transfer processes occurring in high-intensity wildland fires.

The characteristics of the instrumentation setup and the behavior of the outdoor experimental fires resulted in the evaluation of the plume model implementation to only two fires. The characteristics of the data, derived from point measurements from a single instrument tower limit's the inferences that can be made. Extrapolating point data to derive a snapshot in space and time of plume characteristics requires the assumption of steady plume conditions, which simply do not exist above free burning fires. Flame and buoyant plume dynamics are characterized by constant lulls and bursts of activity. The plume model results showed some differences relative to the observed plume temperatures, but no bias was evident. The plume model overpredicted temperature

176

conservation in one fire and underpredicted it in the other. Plume dimension, expressed by time above 500 K, was underpredicted for both fires closer to the flame tip, and overpredicted higher in the plume.

It is difficult to draw conclusions regarding the adequacy of the plume model implementation from the small quantity of field data analyzed. The differences in observed plume characteristics in the two fires that seemed to burn under very similar burning conditions lead to speculation about the effect of large amount of water vapor in the plume in changing its thermodynamic properties, and consequent structure.

The results obtained in the present study raise some future research needs for (1) the possible implementation of the crown fuel ignition model described in Chapter 1, and (2) improving our understanding of the fundamental processes determining fire behavior.

The correct estimation of the fuel available for flaming combustion, which to a large extent influences the prediction of reaction time and consequently the size of the radiative surface and the buoyant plume strength, was identified as critical element for the successful implementation of the crown fuel ignition model. Currently there are no objective methods to estimate the fuel available for flaming combustion. The assumption that fine fuels describe this quantity seems inadequate for fuelbeds characterized by high and/or compacted fine fuel loads. Conversely, during prolonged dry periods it is expected that the volatilization of certain surface fuels, such as medium-sized fuels or certain litter and duff layers, will contribute to flaming combustion. What seems to be required is a general flame front model that describes fundamental processes taking place in the pre-heating zone, combustion zone and the free flame. Such a model should consider heat transfer to fuels, heat transport within the fuels, and some simplified

177

description of combustion processes that would allow one to determine not only the available fuel for flaming combustion, but also describe fire phenomena such as flame front characteristics, reaction time, local flame temperature, residual combustion and heat fluxes to the mineral soil. Within the scope of the current study, what is needed is a model that while describing these processes at a fundamental level is simple enough to allow its use, either because its computation is relatively fast and/or the variables required to run the model can be reasonably estimated or determined in the field.

List of symbols, quantities and units used in equations and text

γ_F	Flame temperature cooling parameter
λ_F	Flame temperature time adjustment variable
β_F	Flame entrainment parameter
b	Plume half width (m)
C/R	Ratio beween convective to radiative heat flux
D_F	Flame depth (m)
Hc	Fuel low heat of combustion (kJ kg^{-1})
H_F	Flame height (m)
I_B	Fireline intensity (kW m^{-1})
I_R	Reaction intensity (kW m^{-2})
L_p	Length of receiver (m)
MC	Moisture content of fine dead fuel particles (fraction of ovendry weight)
q_c	Convective heat flux to fuel particle (kJ m^{-2} s^{-1})
q_r	Radiative heat flux to fuel particle (kJ m^{-2} s^{-1})
ROS	Surface fire rate of spread (m min^{-1})
S	Distance radiation travels from source to receiver (m)
T	Temperature (K)
t	Time (sec)
T_a	Ambient temperature (K)
T_F	Flame temperature (K)
T_{Fmax}	Flame maximum temperature (K)
U	Wind speed (m s^{-1})
U_{10}	10-m open wind speed (m s^{-1})
U_p	Plume velocity (m s^{-1})
w_a	Surface fuel available for flaming combustion (kg m^{-2})
W_F	Flame front width (m)
W_p	Width of receiver (m)
z	Vertical distance (m)
β	Fuel bed packing ratio
δ	Fuel bed depth (m)
Δ	Denotes rate of change
ε	Flame emissivity
η	Combustion efficiency
σ_{SB}	Stefan-Boltzmann constant, $5.67 \cdot 10^{-8}$ W m^{-2} K^{-4}
τ_r	Reaction time (s)

List of subscripts

a	ambient
c	convective
F	flame
i	initial
p	plume
r	radiative

Bibliographic references:

Albini, F.A. 1983. Potential spotting distanced from wind-driven surface fires. USDA For. Serv. Gen. Res. Pap.. INT-309, Intermountain For. and Range Exp. Stn., Ogden Utah, 27p.

Alexander, M.E. 1998. Crown fire thresholds in exotic pine plantations of Australasia. Ph.D. Thesis, Australian National University, Canberra, Australia. 228 p.

Brown, J.K., Oberheu, R.D., Johnston, C.M. 1982. Handbook for inventorying surface fuels and biomass in the interior west. USDA For. Serv. Res. Pap. INT-129, Intermountain For. and Range Exp. Stn., Ogden Utah, 48 p.

Butler, B.W. 1994. Experimental measurements of radiant heat fluxes from simulated wildfire flames. Pages 104-111 in Proceedings of 12th Conference on Fire and Forest Meteorology, Jekyll Island, Georgia. Soc. Am. For., Bethesda, Maryland. SAF Publ. 94-02.

Butler, B.W. 2003. Field measurements of radiant energy transfer in fuel scale wind driven crown fires. 6th ASME-JSME Thermal Engineering Joint Conference, March 16-20.

Byram, G.M. 1959. Combustion of forest fuels. In K.P. Davis (ed.). Forest Fire: Control and Use. McGraw-Hill Book Company, New York. pp. 61-89, 554-555.

Canfield, R.H. 1941. Application of the line intersect method in sampling range vegetation. J. Forestry 39:388-394.

180

Catchpole, E.A., Catchpole, W.R., Rothermel, R.C., 1993. Fire behavior experiments in mixed fuel Complexes. Int. J. Wildland Fire 3(1):45-57.

Catchpole, W.R., Wheeler, C.J. 1992. Estimating plant biomass: A review of techniques. Australian Journal of Ecology 17:121-131.

Catchpole, W.R., Catchpole E.A. 2000. The second generation U.S. fire spread model. US Forest Service – ADFA Joint Research Venture #RMRS-94962-RJVA final report.

Catchpole, W.R., Catchpole, E.A., Butler, B.W., Rothermel, R.C., Morris, G.A., Latham, D.J. 1998. Rate of spread of free-burning fires in woody fuels in a wind tunnel. Comb. Sci. and Tech. 131:1-37.

Catchpole, W.R., Catchpole, E.A., Tate, A.G., Butler, B.W., Rothermel, R.C. 2002. A model for the steady spread of fire through a homogeneous fuel bed. In Viegas, D.X. (ed.) Proceedings of 4th International Conference on Forest Fire Research, 2002 Wildland Fire Safety Summit, Luso - Coimbra, Portugal – 18/23 November 2002. Millpress, Rotherdam. 11 p.

Churchill, S.W., Bernstein, M 1977. A correlating equation for forced convection from gases and liquids to a circular cylinder in crossflow. J Heat Transfer 99:300.

Clark, T.L., Jenkins, M.A., Coen, J., Packam, D. 1996. A couple atmosphere-fire model: Convective Feedback on fire-line dynamics. J. Appl. Meteo. 35:875-901.

Clark, T.L., Radke, L., Coen, J., Middleton, D. 1999. Analysis of small-scale convective dynamics in a crown fire using infrared video camera imagery. J. Applied Meteo. 38:1401-1420.

Cruz, M.G., Viegas, D.X. 1998. Fire behavior in some common Central Portugal fuel complexes: Evaluation of fire behavior models performance. Pages 829-875 in the Proceedings of 3rd International Conference on Forest Fire Research – 14th Conference on Fire and Forest Meteorology, Luso - Coimbra, Portugal - 16/20 November 1998.

Cruz, M.G., Butler B.W. and D. X. Viegas. 2003. Characterization of flame radiant heat fluxes in shrubland fires. Poster presented at the 2nd International Fire Ecology and Fire management Congress, November 2003, Orlando, FL.

Dupuy, J.L., Larini, M. 1999. Fire spread through a porous forest fuel bed: a radiative and convective model including fire-induced flow effects. Int. J. Wildland Fire 9(3):155-172.

Dupuy, J.L., Marechal, J., Morvan, D. 2003. Fires from a cylindrical forest fuel burner: combustion dynamics and flame properties. Combustion and Flame 135:65-76.

Fernandes, P.M., Catchpole, W.R., Rego, F.C. 2000. Shrubland fire behavior modelling with microplot data. Can. J. For. Res. 30:889-899.

Frandsen, W.H., Rothermel, R.C. 1972. Measuring the energy-release rate of a spreading fire. Combustion and flame 19:17-24.

Gould, J.S., Knight, I, Sullivan, A.L. 1997. Physical modelling of leaf scorch height from prescribed fires in young Eucalyptus sieberi regrowth forests in South-eastern Australia. Int. J. Wildland Fire 7(1):7-20.

Hornbaker, D.R., Rall, D.L. 1968. Heat flux measurements: a practical guide. Instrumentation Technology. February 1968, pp. 51-56.

Kaplan, C.R., Shaddix, C.R., Smyth, K.C. 1996. Computations of enhaced soot production in time-varying CH_4/air diffusion flames. Combustion and Flame106: 392.

Latham, D.J. 1998. High Speed Photography of Fires. In: Second Symposium on Fire and Forest Meteorology. 11-16 January 1998, Phoenix, AZ. AMS. pp 24-26.

McAlpine, R.S., Wakimoto, R.H. 1991. The acceleration of fire from point source to equilibrium spread. For. Sci. 37(5):1314-1337.

McCafreey, B.J., Heskestad, G. 1976. A robust bi-directional low-velocity probe for flame and fire application. Combustion and Flame, 26:125.

Mercer, G.N., Weber, R.O. 1994. Plumes above line fires in a cross wind. Int. J. Wildland Fire 4(4):201-207.

Moore, P.H., Gill, A.M., Kohnert, R. 1995. Quantifying bushfires for ecology using two electronic devices and biological indicators. CALM Science Supplement 4:83-88.

Morvan, D., Dupuy, J.L, 2001. Modeling of fire spread through a forest fuel bed using a multiphase formulation. Combustion and Flame 127:1981-1994.

Nelson, R.M. Jr. 2003a. Power of the fire – a thermodynamic analysis. Int. J. Wildland Fire 12:51-63.

Nelson, R.M. Jr. 2003b. Reaction times and burning rates for wind tunnel headfires. Int. J. Wildland Fire 12:195-211.

Newman, J.S. 1987. Multi-directional flow probe assembly for fire application. Journal of Fire Sciences 5:50-56.

Packham, D.R. 1969. Heat transfer above a small ground fire. Aust. For. Res. 5(1):19-24.

Palmer, T.Y. 1980. Temperature and winds in chaparral fires. In. 6th Conference on Ifre and Forest Meteorology. Pp. 256-259. Society of American Foresters.

Porterie, B., Morvan, D., Loraud, J.C., Larini, M. 2000. Firespread through fuel beds: modeling of wind-aided fires and induced hydrodynamics. Physics of Fluids 12(7):1762-1782.

Rothermel, R.C. 1967a. Airflow characteristics - wind tunnels and combustion facilities, Northern Forest Fire Laboratory. USDA For. Sev., Intermountain For. And Range Exp. Stn., Missoula, MT. 32 p.

Rothermel, R. C. 1967b. Low airspeed differential pressure integrating system. Res. Note INT-37. Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station. 8 p.

Rothermel, R. C. 1972. A mathematical model for predicting fire spread in wildland fuels. Res. Pap. INT-115. Ogden, UT: U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station. 40 p.

Saveland, J.M., Bakken, S.R., Neuenschwander, J.F. 1990. Predicting mortality and scorch height form prescribed burning of ponderosa pine in northern Idaho. Univ.

183

Idaho, Coll. For. Wildlife Range Sci., For., Wildlife Range Exp. Stn., Moscow, Idaho. Bull. 53, 9 p.

Schmidt, W.C., Friede, J.L. 1996. Experimental forests, ranges, and watersheds in the Northern Rocky Mountains: A compendium of outdoor laboratories in Utah, Idaho, and Montana. USDA For. Serv. Gen. Tech. Rep. INT-GTR-334, Intermountain For. and Range Exp. Stn., Ogden Utah, 117 p.

Sero-Guillaume, O., Margerit, J. 2002. Modelling forest fires. Part I: a complete set of equations derived by extended irreversible thermodynamics. Int. J. of Heat and Mass Transfer 45:1705-1722.

Van Wagner, C.E. 1970. On the value of temperature data in forest research. Canadian Forestry Service, Internal Report PS-20, Ottawa, Canada.

Van Wagner, C.E. 1973. Height of crown scorch in forest fires. Can. J. For. Res. 3:373-378.

Van Wagner, C.E. 1975. Convection temperatures above low intensity forest fires. Bi-monthly. Res. Notes, Can. For. Serv. Bi-mon. Res. Notes 31:21,26.

Viegas, D.X., Cruz, M.G., Ribeiro, L.M., Silva, A.J., Oleero, A., Arrue, B., Dios, R., Gomez-Rodriguez, F., Merino, L., Miranda, A.I., Santos, P. 2002. Gestosa fire spread experiments. In Forest Fire Research & Wildland Fire Safety. Proceedings of the IV International Conference on Forest Fire Research/2002 Wildland Fire Safety Summit. Edited by D.X. Viegas. Millpress Scientific Publications, Rotterdam, The Netherlands. CD-ROM. 13 p.

Weber, R.O., Gill, A.M., Lyons, P.R.A., Mercer, G.N. 1995. Time dependence of temperature above wildland fires. CALMScience Supplement 4:17-22.

Wotton, M.B., Martin, T.L., Engel, K. 1998. Vertical flame intensity profile from a surface fire. Pages 175-182 in Weber, R. (Chair) Proceeding of the 13th Conference on fire and forest meteorology, Oct. 1996, Lorne, Victoria, Australia. International Association of Wildland Fire, Moran WY.

Appendix A

Description of reaction time and buoyant plume model

This appendix describes the models Nelson's (2003b) reaction time model and Mercer and Weber (1994) the buoyant plume model. The reaction time is used to predict the size of the radiating surface and the plume model is used to determine plume characteristics and subsequently the convective heat transfer coefficient from Reynolds and Prandtl numbers.

Nelson's (2003b) reaction time model.

The model predicts the fuelbed reaction time from:

$$[A1] \quad \tau_r = \frac{2 \cdot w_a \cdot \left(Q_f + Q_m \cdot MC\right)}{\left(h_{ef} \cdot \left(T_m - T_s\right)\right)}$$

with the particle effective heat transfer coefficient, h_{ef}, integrating both radiative and convective heat transfer within the fuelbed. The radiative heat transfer coefficient in the reaction zone is given by (see Albini and Reinhardt 1995):

$$[A2] \quad h_r = 0.5 \cdot B \cdot \left(T_m + T_s\right) \cdot \left(T_m^2 + T_s^2\right)$$

and the convective heat transfer coefficient in the reaction zone as:

$$[A3] \quad h_c = 0.344 \cdot \left(\frac{\sigma \cdot k_c}{4}\right) \cdot \left(\frac{4 \cdot V}{\sigma \cdot \nu}\right)^{0.56}$$

This convective heat transfer coefficient is distinct from the one used to estimate convective heat transfer to the canopy fuels (eq. [19]). The sublimation temperature, T_s, is assumed constant (673 K) and the mass average temperature of the reacting mixture in the fuelbed is given by:

185

$$[A4] \quad T_m = 500 + \frac{500 \cdot (2.09 \cdot MC + 1.05 \cdot X_1 \cdot (1 - \gamma_c) \cdot (1 + N_v))}{2.09 \cdot MC + 1.05 \cdot Z + 1.05 \cdot (1 - \gamma_c) \cdot (1 + X_1 \cdot N_v)}$$

The fraction of volatiles that burn is given by:

$$[A5] \quad X_1 = \frac{H_N + 1045 \cdot MC + 709 * Z}{(1 - \gamma) \cdot (\Delta H_v - 525 - 1024 \cdot N_V)}$$

The stoichiometric air/fuel mass ratio of volatiles is (Albini 1980):

$$[A6] \quad N_V = \frac{\dfrac{(1 - \varepsilon) \cdot H_c - (\gamma_c - mf) \cdot \Delta H_V}{1 - \gamma_c} + 1580}{3270}$$

The non-reacting air entering the reaction zone is:

$$[A7] \quad Z = \frac{\tau_r \cdot \cos A_F \cdot \rho_m \cdot V}{w_a} - MC - X_1 \cdot N_v$$

With A_F being the flame tilt angle from the vertical

$$[A8] \quad A_F = ArcTan\left(\frac{U_s}{U_{Fv}}\right)$$

The vertical component of free flame velocity is estimated from (Nelson 2003a):

$$[A9] \quad U_{Fv} = \left(\frac{2 \cdot g \cdot I_B}{\rho_a \cdot c_p \cdot (T_a)}\right)^{1/3}$$

The mixture velocity in the reaction zone, V, is obtained from:

$$[A10] \quad V = \frac{(H_N \cdot (1 - \gamma_c) \cdot \rho_f \cdot \beta \cdot \delta)}{(\tau \cdot Cos(A_F)) \cdot (\rho_m \cdot c_v \cdot (T_m - T_x))}$$

Within this system of equations there are two unknowns, the reaction time, τ_r, and the non-reacting air entering the reaction zone, Z. The dependence of the reaction time on the temperature of the reacting mixture and the dependence of this variable on non-reacting air entering the reaction zone makes it necessary to iterate the system until

186

converging of reaction time values. Initial values for Z and τ_r are respectively 0 and $75571*\beta*\delta$. The calculations in the iteration loop use the following order: X_1, T_m, A, V, Z, h_p, τ_r.

Mercer and Weber (1994) buoyant plume model

The plume model is implemented as a system of 6 coupled first order ordinary differential equations and three algebraic equations. The six differential equations are:

[A11] $\quad \dfrac{dy_1}{ds} = \rho_a \cdot V_e$

[A12] $\quad \dfrac{dy_2}{ds} = \rho_a \cdot V_e \cdot U_z \cdot \cos y_3 + \dfrac{y_1 \cdot y_4 \cdot g \cdot \cos y_3}{y_2 \cdot T_a}$

[A13] $\quad \dfrac{dy_3}{ds} = -\dfrac{\rho_a \cdot V_e \cdot U_z \cdot \sin y_3}{y_2} + \dfrac{y_1 \cdot y_4 \cdot g \cdot \cos y_3}{y_2^{\,2} \cdot T_a}$

[A14] $\quad \dfrac{dy_4}{ds} = 0$

[A15] $\quad \dfrac{dy_5}{ds} = \cos y_3$

[A16] $\quad \dfrac{dy_6}{ds} = \sin y_3$

The three algebraic equations are the equation of state (eq. [8]), the entrainment velocity (eq. [9]), and the ambient wind profile (eqs. [10] and [11]). The six initial conditions are stated as:

[A17] $\quad y_1 = \rho_p \cdot b \cdot U_p$;

[A18] $\quad y_2 = \rho_p \cdot b \cdot U_p^{\,2}$;

[A19] $y_3 = \theta$;

[A20] $y_4 = \rho_p \cdot b \cdot \Delta T$;

[A21] $y_5 = x$;

[A22] $y_6 = z$;

And the quantities of interest are obtain as function of position along the plume centerline:

[A23] $T_p = T_a + \dfrac{y_4}{y_1}$

[A24] $U_p = \dfrac{y_2}{y_1}$

[A25] $b = y_1 \cdot \dfrac{y_1}{\rho_p \cdot y_2}$

[A26] $x = y_5$

[A27] $z = y_6$

188

Appendix B

Crown Fuel Ignition Model source Code

Crown fuel ignition model main code

```
//cfim.cpp - source file with main function.

#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>
#include <conio.h>
#include <sys/timeb.h>

#define NRANSI

#include "nr.h"
#include "nrutilcpp.h"

#define LENGTH 256
#define SUCCESS 0
#define FAIL 1
#define PI 3.14159265358979324
#define SB 0.00000005669
#define Rhoa 1.177              //kg/m^3
#define Alpha 0.16              //coefficient used in plume model
#define Beta 0.5                //coefficient used in plume model
#define g 9.81                  //acceleration of gravity
#define Cpdry 2100.0            //canopy fuel specific heat (J/kg/K)
#define Cpwater 4187.0          //water specific heat (J/kg/K)
#define L 2254000.0             //water latent heat of vaporization
#define EPS 3.0e-11             //parameter in Gaussian quadrature


#define NVAR 6                  //number of differential equations solved in plume
model
#define NSTEP 10000             //number of steps when solving differential
equations (s-points)

#include "rkdumb.h"


typedef struct input {
    double u10;double slope;double Ta;double sh;double alpha;double
igtemp;double tstep;long iters;double xstart;
    long FuelModelNumber;double FuelMoisture[5];double rho_surf;double
sigma_surf;double hc;
    double sigma_can;double canbaseht;double diameter;double FMC;double
length;double rho_can;

}Input;
Input in;

double windprofile(Input IN,double Z);
double maxflametemp(double Us,Input IN);
double reaction_time(double R,double Ua,double beta,double gamma,Input IN);

double behave(double MidflameWindspeed,Input IN,double
*FirelineIntensity,double *FlameLength, double *HeatPerUnitArea);
double pow2(double base);
void SetStandardFuelModel(long number);
///void SetCustomFuelModel(.....);
void SetFuelMoistures(double ones, double tens, double hundreds, double liveh,
double livew);
double CalcSpreadRate(double *Fuel, double *Moisture, double Slope,
                      double WindSpeed, double *FlameLength,
```

190

```c
                              double *FirelineIntensity, double *HeatPerUnitArea);
void derivs(double x,double y[],double dydx[]);
void locate(double x2[],unsigned long n,double x,long *j);
double linterp(double X[],double Y[],long n,double xi);
double ptinterp(double x1,double x2,double y1,double y2,double xi);
void fillconvect0(void);
void fillconvect1(void);
double gettemp(double x);
double getspeed(double x);
void plumemodel(void);
void calc_xplumert(void);
void calc_xplumelt(void);
double fxn(double pt[],double wgt);
double getrad(double Dx,double Tp);
double radflametemp(double X);
double getconvectflux(double Dx,double Tp);
void gauleg(double x1, double x2, long n);
double qgaus(double (*func)(double,double,double), double a, double b,double
Dx,double Tp);
double xx2(double y);
double xx1(double y);
double f2(double x,double Dx,double Tp);
double f1(double y,double Dx,double Tp);
double integration_function(double x,double y,double Dx,double Tp);

FILE *fcustomin;
FILE *fout;

static double FuelModel[13];
static double fuelmoisture[5];
static double rhoa = 1.177;            //  kg/m^3
static double gamma = 0.15;            //  fuels unavailable for combustion
static double combust_efficiency= 0.85;    //fuels available for flaming
combustion in surface fire
static double ROS;
static double avail_surf_fuel;
static double iByram;                  //  byram's intensity
static double beta_surf;               //  packing ratio for surface fuel bed
char custominfile[LENGTH];
double FirelineIntensity, FlameLength, HeatPerUnitArea;
static double taur;                    //  residence time
static double flamedepth;
static double flameheight;
static double Ti=800.0;                //initial temperature of the plume(plume
model)
static double Cp=1.05;                 //specific heat of air(plume model)
static double ucantop,umid,uz,maxflmtemp;
static double Up;

extern double **y,*xx;     /* referencing declaration */
static double **finalplume,**rawplume;  //declare matrices to store plume info
static double s_canopyht,s_rtcanopyht,s_ltcanopyht;      //plume model
variables
static double xplumeright,xplumeleft,xplumemid;     //x value of the respective
place in the plume at the canopy height (global coord. sys.)
static double **convect,**convect2;              //matrix of x, air temp,
velocity in the plume at the canopy height
static long j;
static double zinternrt1,zinternrt2,zinternlt1,zinternlt2,zerror1,zerror2;
static double xinternrt,xinternlt;
static double x1=0.0, x2=4.0, px2;
static double NPTS;                //number of values used in convect array
static double radenergy,convectenergy;
```

```c
static double part_vol,part_surf_area,C,xpos,Tp,Qtotal,sink;
static double maxparttemp=0.0;

long idum;          /* for ranno */
int ndim=4;         /* for fxn */
double pemiss=1.0,femiss=1.0;      //emissivity of the particle and flame
double attenuation=0.02;\
double Lp=0.001,Wp=0.001,Wf=20.0;   //length and width of particle, width of
flame

static int radflag=0;        //if 0, quadruple integral Monte Carlo used;if 1,
double integral Gaussian Quadrature used;
static double Y1=-Wf/2.0, Y2=Wf/2.0;    //limits of integration for Gaussian
radiation model;
static long n=21;         //parameter for Gaussian quadrature integration
(controls how many "iterations" happen)
static double ysav,x[1408],w[1408];    //used in Gaussian quadrature; length
of vectors must be n+1

static int convectflag=0;    //if 0, gaussian profile of plume follows the "c"
line;if 1, gaussian profile follow canbaseht line;

//TEMPORARY!!!TEMPORARY!!!TEMPORARY!!!TEMPORARY!!!TEMPORARY!!!TEMPORARY!!!
 //double Tp=300.0;       //temp of particle
//double Dx=1.0;         //distance from flm leading edge to particle center
//TEMPORARY!!!TEMPORARY!!!TEMPORARY!!!TEMPORARY!!!TEMPORARY!!!TEMPORARY!!!

int main(void)
{
    FILE *fuserin;
    FILE *fresearcherin;

    char    userinfile[LENGTH] = "user_inputs.txt",
            researcherinfile[LENGTH] = "researcher_inputs.txt",
            outfile[LENGTH] = "outfile.xls",
            junk[LENGTH];
    long i;
    long printed_iters=5;
    int flag=0;

//     double FirelineIntensity, FlameLength, HeatPerUnitArea;




    printf("\n\nCRUZ CROWNFIRE MODEL\n");
//     printf("\n\nEnter user input filename (with extension):\n");
//     scanf("%s",&userinfile);
//     printf("\n\nEnter researcher input filename (with extension):\n");
//     scanf("%s",&researcherinfile);

    if((fuserin = fopen (userinfile, "r")) == NULL)
        {
            printf("A user input file cannot be found.  Exiting program...\n");
            exit(0);
        }

    if((fresearcherin = fopen (researcherinfile, "r")) == NULL)
        {
            printf("A researcher input file cannot be found.  Exiting
program...\n");
            exit(0);
        }
```

192

```c
    if((fout=fopen(outfile,"w+")) == NULL)
        {
            printf("There is a problem opening \"outfile.dat\".\nThe file may
currently be in use by another program.\n\n  Exiting program...\n");
            exit(0);
        }


    fscanf(fuserin,"%s %s %s %lf %s %lf %s %lf %s %lf %s %lf",
&junk,&junk,&junk,&in.u10,&junk,&in.slope,&junk,&in.Ta,&junk,&in.sh,&junk,&in.a
lpha);
    fscanf(fuserin,"%s %s %i %s %lf %s %lf %s %lf %s %lf %s %lf %s
%lf",&junk,&junk,&in.FuelModelNumber,&junk,&in.FuelMoisture[0],&junk,&in.FuelMo
isture[1],&junk,&in.FuelMoisture[2],&junk,&in.FuelMoisture[3],&junk,&in.FuelMoi
sture[4],&junk,&in.sigma_surf);
    fscanf(fuserin,"%s %s %lf %s %lf %s %lf %s
%lf",&junk,&junk,&in.sigma_can,&junk,&in.canbaseht,&junk,&in.diameter,&junk,&in
.FMC);

    fscanf(fresearcherin,"%s %s %s %lf %s %lf %s %ld %s %lf",
&junk,&junk,&junk,&in.igtemp,&junk,&in.tstep,&junk,&in.iters,&junk,&in.xstart);
    fscanf(fresearcherin,"%s %s %lf %s
%lf",&junk,&junk,&in.rho_surf,&junk,&in.hc);
    fscanf(fresearcherin,"%s %s %lf %s
%lf",&junk,&junk,&in.length,&junk,&in.rho_can);



    ucantop=windprofile(in,in.sh);          //windspeed at the canopy top
    umid=windprofile(in,0.1*in.sh);         //surface fire midflame windspeed

    ROS=behave(umid,in,&FirelineIntensity,&FlameLength,&HeatPerUnitArea);
    avail_surf_fuel=0.2242*FuelModel[0];       //0.2242 converts to correct
units
    maxflmtemp=maxflametemp(umid,in);
    FirelineIntensity*=3461.4693327428;              //convert to W/m
    FlameLength*=0.3048;                             //convert to m
    HeatPerUnitArea*=11356.526682227;               //convert to j/m^2
    iByram=combust_efficiency*ROS*avail_surf_fuel*in.hc;
    beta_surf=avail_surf_fuel/(FuelModel[11]*in.rho_surf*(1.0-gamma));
//packing ratio for surface fuel bed
    taur=reaction_time(ROS,umid,beta_surf,gamma,in);
    flamedepth=ROS*taur;
    flameheight=iByram/(385.0*umid);
    part_vol=PI*pow2(in.diameter)/4.0*in.length;
    part_surf_area=PI*in.diameter*in.length;
    Up=pow(((2.0*g*iByram)/(rhoa*Cp*(in.Ta))),(1.0/3.0));

    plumemodel();

    xpos=in.xstart-ROS*in.tstep;    //initialize x position to one step before
xstart
    Tp=in.Ta;                       //initialize the fuel particle temperature to
ambient temperature
    C=((Cpdry+in.FMC*Cpwater)*(373.0-in.Ta)+(in.FMC*L)+Cpdry*(in.igtemp-
373.0))/(in.igtemp-in.Ta);    //"effective" Cp of moist fuel

    printf("\n\nIterating...\n");
    for(i=1;i<=in.iters;i++)                //solving for particle temperature
        {
            xpos=xpos-ROS*in.tstep;     //position of the particle from the fire
leading edge coordinate system
```

```
            convectenergy=getconvectflux(xpos,Tp)*part_surf_area;     //get the
convective energy (W)
            radenergy=getrad(xpos,Tp);                //get the radiative energy (W)
            Qtotal=in.tstep*(radenergy+convectenergy);    //in Joules
            sink=in.rho_can*part_vol*C;                //sink is rho*V*C
            Tp=Tp+Qtotal/sink;                //calculate the new temperature
of the particle
//
fprintf(fout,"%lf\t%lf\t%lf\t%lf\t%lf\n",xpos,Tp,Qtotal,convectenergy,radenergy
);
            fprintf(fout,"%lf\t%lf\t%lf\t%lf\t%lf\n",(double)(i-
1)*in.tstep,Tp,Qtotal,convectenergy,radenergy);
            if((i%printed_iters)==0)
            {
                printf("\nIteration\t%ld",i);
                printf("\n\tx-position = %lf\tparticle temp = %lf",xpos,Tp);
            }
            if(Tp>maxparttemp)
                maxparttemp=Tp;
            if(Tp>=in.igtemp)
                flag=1;
        }

    printf("\n\nmaxflametemp=%lf\tiByram=%lf\n",maxflmtemp,iByram);
    printf("\nreact time=%lf\tROS=%lf\n",taur,ROS);

    printf("\nflamedepth %lf\tflameheight %lf\tflamelength
%lf\n",flamedepth,flameheight,FlameLength);
    printf("\ncanbaseht %lf\tumid %lf\t\tucantop
%lf\n",in.canbaseht,umid,ucantop);
    printf("\nxplumeleft %lf\txplumemid %lf\txplumeright
%lf\n",xplumeleft,xplumemid,xplumeright);
//    printf("\nsplumeleft %lf\tsplumemid %lf\tsplumeright
%lf\n",s_ltcanopyht,s_canopyht,s_rtcanopyht);

    printf("\nmax canopy fuel temperature %lf",maxparttemp);
    if(flag<=0)
    {
        printf("\n\n****The canopy DID NOT ignite.****\n");
    }else{
        printf("\n\n****The canopy DID ignite.****\n");
    }

    free_matrix(rawplume,1,6,1,NSTEP+1);
    free_vector(xx,1,NSTEP+1);
    free_matrix(convect,1,3,1,NPTS+1);
//-------------------------------------------------------------------------------
-
//-------------------------------------------------------------------------------
-
//-------------------------------------------------------------------------------
-

    printf("\n\nPress any key to continue\n");
    getch();
//    scanf("%s",&junk);
    return(0);
}

double windprofile(Input IN,double Z)
    {
      double u,ush;
```

194

```c
        ush=IN.u10*((log((IN.sh-0.64*IN.sh)/(0.13*IN.sh)))/(log(((10.0+IN.sh)-
0.64*IN.sh)/(0.13*IN.sh)))));
        if(Z>=0.6*IN.sh)
            {
            u=ush*exp(IN.alpha*(Z/IN.sh-1.0));
            }
        else
            {
            u=ush*exp(IN.alpha*(IN.sh*0.6/IN.sh-1.0));
            }

        return u;
        }

double maxflametemp(double Us,Input IN)
    {
        double beta1=300.684,beta2=136.791,beta3=0.506,beta4=100.448,beta5=-0.531;
        double maxtemp;


maxtemp=300.0+beta1*avail_surf_fuel+beta2*pow(Us,beta3)+beta4*pow(IN.FuelMoistu
re[0],beta5);
        return maxtemp;
        }
/*
double radflametemp(double maxflametemp, double Tab, double time,double
IgTemp,double Taur)
{
        double heatingrate=60.0,beta=8.0,k=1.0,Adimen600,rise,ig,decay,coef,res;

        Adimen600=(IgTemp-Tab)/(maxflametemp-Tab);
        rise=sqrt(-1.0*pow(beta,2.0)*(log(Adimen600)));
        ig=-rise;
        decay=(-1.0*(log(Adimen600)))/(Taur+ig);
        if((time+ig)<0.0)
        {
            coef=k*exp(-1.0*(pow((time+ig),2.0))/pow(beta,2.0));
        }else
        {
            coef=k*exp(-1.0*(decay*(time+ig)));
        }
        res=coef*(maxflametemp-Tab)+Tab;
        printf("\n\n%lf\n",rise);
        return(res);
}
*/
double getleadingxpos(double Time,double Ros,double Startx)
{
        double Xpos=Startx+Ros*Time;
        return (Xpos);
}


double reaction_time(double R,double Ua,double BETA,double gamma,Input IN)
    {
        double
Zini=0.0,taufbini=75571.0*BETA*FuelModel[11],taufbold=300.0,epsilon=0.0035,rhoa
=1.2,Tab=IN.Ta-273.0,Cp=1.05,Tx=500.0,Hiprime=504.0,
            deltaHc=31200.0,SBolt=5.67*pow(10,-11),Ts=400,kc=6.63*pow(10,-
5),Qf=711,Qm=2570,nu=1.13*pow(10,-4),
            taufb=taufbini,Z=Zini,IBreacT=0.85*IN.hc*(1.0-
gamma)*IN.rho_surf*BETA*FuelModel[11]*R,theta=IN.sigma_surf*BETA*FuelModel[11],
F=0.283+0.178*log(theta),
```

```c
            HI=(Hiprime+2570.0*IN.FuelMoisture[0])/F,Hp=207.0,HR=175.0*(1.0-
gamma),HS=836.0*IN.FuelMoisture[0],deltaHv=((1.0-epsilon)*IN.hc-(gamma-
epsilon)*deltaHc)/(1.0-gamma),

deltaHvhigh=deltaHv+1580.0,Nv=deltaHvhigh/3270.0,HN,X1,HA1,Tm,Uv,A,rhom,V,Z1,hr
,hc,hef,taufb1,taur;
//      double Hd=772.0,Cps=2.09;
        do
            {
            taufbold=taufb;
            Nv=deltaHvhigh/3270.0;
            HN=HI+Hp+HR+HS;
            X1=(HN+1045.0*IN.FuelMoisture[0]+1.05*(700.0-25.0)*Z)/((1.0-
gamma)*(deltaHv-525.0-1024.0*Nv));
//          HA1=X1*(1.0-gamma)*(deltaHv-525.0-1024.0*Nv)-
1045.0*IN.FuelMoisture[0]-709.0*Z;
            Tm=500.0+(500.0*(2.09*IN.FuelMoisture[0]+1.05*X1*(1.0-
gamma)*(1.0+Nv)))/(2.09*IN.FuelMoisture[0]+1.05*Z+1.05*(1.0-
gamma)*(1.0+X1*Nv));
            Uv=pow(((2.0*g*IBreacT)/(rhoa*Cp*(Tab+273.0))),(1.0/3.0));
            A=atan(Ua/Uv);
            rhom=rhoa*((Tab+273.0)/(Tm+273.0));
            V=(HN*BETA*(1.0-
gamma)*FuelModel[11]*IN.rho_surf*(1.0/(cos(A)))/(taufb*rhom*Cp*(Tm-Tx)));
            Z1=((taufb*cos(A)*rhom*V)/avail_surf_fuel)-IN.FuelMoisture[0]-
(1.0+X1*Nv);

hr=0.5*SBolt*((Tm+273.0)+(Ts+273.0))*(pow((Tm+273.0),2.0)+pow((Ts+273.0),2.0));

hc=0.344*((IN.sigma_surf*kc)/4.0)*pow(((4.0*V)/(IN.sigma_surf*nu)),0.56);
            hef=hr+hc;
//          taup=2.0*(1.0-
gamma)*IN.rho_surf*BETA*IN.sigma_surf*(Qf+Qm*IN.FuelMoisture[0])/(hef*(Tm-
Ts)*(1.0-BETA));
            taufb1=2.0*(1.0-
gamma)*IN.rho_surf*BETA*FuelModel[11]*(Qf+Qm*IN.FuelMoisture[0])/(hef*(Tm-
Ts)*(1.0-BETA));
            Z=Z1;
            taufb=taufb1;
//          printf("\n%lf %lf",A,taufb);
            }
        while(fabs(taufbold-taufb)>0.0001);
        taur=taufb;
        return taur;
        }


//////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////
double behave(double MidflameWindspeed,Input IN,double
*FirelineIntensity,double *FlameLength, double *HeatPerUnitArea)
        {
        double SpreadRate;
        long fuelmodelnumber=IN.FuelModelNumber;
        double slope=IN.slope;


        int i;
        for (i=0;i<5;i++)
        {
            fuelmoisture[i]=IN.FuelMoisture[i];
```

```c
        }

        MidflameWindspeed*=2.23694;                     //convert to mph
//      MidflameWindspeed=4.0;                          // miles per hour
//      Slope=30.0;                                     // percent

        // set fuel model parameters into FuelModel array to pass to spread func
        SetStandardFuelModel(fuelmodelnumber);

        // moistures for comparison of BEHAVE outputs
        //SetFuelMoistures(FuelMoisture[0], FuelMoisture[1], FuelMoisture[2],
FuelMoisture[3], FuelMoisture[4]);            // dry, Andrews 1986 p19
        //SetFuelMoistures(0.12, 0.13, 0.14, 1.70, 1.70);        // moderate,
Andrews 1986 p19

        // Call Spread function
        SpreadRate=CalcSpreadRate(FuelModel, fuelmoisture, MidflameWindspeed,
slope,
                        FlameLength, FirelineIntensity, HeatPerUnitArea);

        SpreadRate=SpreadRate*0.00508;          // convert to m/s
//      printf("%lf %lf %lf %lf\n", SpreadRate, FirelineIntensity, FlameLength,
HeatPerUnitArea);
        return SpreadRate;
        }


double pow2(double base)
{
        return base*base;
}

/*void SetFuelMoistures(double Ones, double Tens, double Hundreds, double
LiveH, double LiveW)
{
        FuelMoisture[0]=Ones;
        FuelMoisture[1]=Tens;
        FuelMoisture[2]=Hundreds;
        FuelMoisture[3]=LiveH;
        FuelMoisture[4]=LiveW;
}
*/


void SetStandardFuelModel(long number)
{
        char junk[LENGTH];
        if(number>=14)
        {
            sprintf(custominfile,"custom_%ld.txt",number);
            if((fcustomin = fopen (custominfile, "r")) == NULL)
            {
                printf("A custom fuel model input file cannot be found.  Exiting
program...\n");
                exit(0);
            }
            fscanf(fcustomin,"%s %lf %s %lf %s %lf %s %lf %s %lf %s %lf %s %lf %s
%lf %s %lf %s %lf %s %lf %s %lf %s %lf
%s",&junk,&FuelModel[0],&junk,&FuelModel[1],&junk,&FuelModel[2],&junk,&FuelMode
l[3],&junk,&FuelModel[4],&junk,&FuelModel[5],&junk,&FuelModel[6],&junk,&FuelMod
el[7],&junk,&FuelModel[8],&junk,&FuelModel[9],&junk,&FuelModel[10],&junk,&FuelM
odel[11],&junk,&FuelModel[12],&junk);
```

197

```c
    }else{
        FuelModel[3]=0.0;                                          // loading for live
herb
        FuelModel[6]=1800.0;                                       // surf for live herb
        FuelModel[8]=FuelModel[9]=FuelModel[10]=8000.0;   // heat contents all
the same
        switch(number)
        {       case 1: FuelModel[0]=0.74; FuelModel[1]=0.0; FuelModel[2]=0.0;
FuelModel[4]=0.0;FuelModel[5]=3500;FuelModel[7]=1500;FuelModel[11]=1.0;FuelMode
l[12]=.12;break;
                case 2: FuelModel[0]=2.0; FuelModel[1]=1.0; FuelModel[2]=0.5;
FuelModel[4]=.5;FuelModel[5]=3000.0;FuelModel[7]=1500.0;FuelModel[11]=1.0;FuelM
odel[12]=.15;break;
                case 3: FuelModel[0]=3.01; FuelModel[1]=0.0; FuelModel[2]=0.0;
FuelModel[4]=0.0;FuelModel[5]=1500.0;FuelModel[7]=1500.0;FuelModel[11]=2.5;Fuel
Model[12]=.25;break;
                case 4: FuelModel[0]=5.01; FuelModel[1]=4.01; FuelModel[2]=2.0;
FuelModel[4]=5.01;FuelModel[5]=2000.0;FuelModel[7]=1500.0;FuelModel[11]=6.0;Fue
lModel[12]=.20;break;
                case 5: FuelModel[0]=1.0; FuelModel[1]=0.5; FuelModel[2]=0.0;
FuelModel[4]=2.0;FuelModel[5]=2000.0;FuelModel[7]=1500.0;FuelModel[11]=2.0;Fuel
Model[12]=.20;break;
                case 6: FuelModel[0]=1.5; FuelModel[1]=2.5; FuelModel[2]=2.0;
FuelModel[4]=0.0;FuelModel[5]=1750.0;FuelModel[7]=1500.0;FuelModel[11]=2.5;Fuel
Model[12]=.25;break;
                case 7: FuelModel[0]=1.13; FuelModel[1]=1.87; FuelModel[2]=1.5;
FuelModel[4]=0.37;FuelModel[7]=1550.0;FuelModel[5]=1750.0;FuelModel[11]=2.5;Fue
lModel[12]=.40;break;
                case 8: FuelModel[0]=1.5; FuelModel[1]=1.0; FuelModel[2]=2.5;
FuelModel[4]=0.0;FuelModel[5]=2000.0;FuelModel[7]=1500.0;FuelModel[11]=0.2;Fuel
Model[12]=.30;break;
                case 9: FuelModel[0]=2.92; FuelModel[1]=0.41; FuelModel[2]=0.15;
FuelModel[4]=0.0;FuelModel[5]=2500.0;FuelModel[7]=1500.0;FuelModel[11]=0.2;Fuel
Model[12]=.25;break;
                case 10: FuelModel[0]=3.01; FuelModel[1]=2.0; FuelModel[2]=5.01;
FuelModel[4]=2.0;FuelModel[5]=2000.0;FuelModel[7]=1500.0;FuelModel[11]=1.0;Fuel
Model[12]=.25;break;
                case 11: FuelModel[0]=1.5; FuelModel[1]=4.51; FuelModel[2]=5.51;
FuelModel[4]=0.0;FuelModel[5]=1500.0;FuelModel[7]=1500.0;FuelModel[11]=1.0;Fuel
Model[12]=.15;break;
                case 12: FuelModel[0]=4.01; FuelModel[1]=14.03;
FuelModel[2]=16.53;
FuelModel[4]=0.0;FuelModel[5]=1500.0;FuelModel[7]=1500.0;FuelModel[11]=2.3;Fuel
Model[12]=.20;break;
                case 13: FuelModel[0]=7.01; FuelModel[1]=23.04;
FuelModel[2]=28.05;
FuelModel[4]=0.0;FuelModel[5]=1500.0;FuelModel[7]=1500.0;FuelModel[11]=3.0;Fuel
Model[12]=.25;break;
                }
    }
    if(fuelmoisture[0]>=FuelModel[12])
    {
        printf("\n\nSurface fire will not spread because you have \nreached the
moisture of extinction.\nExiting...\n");
        exit(0);
    }

    // constants for 13 standard models
}


/*
void SetCustomModel(......)
```

```
        // some arbitrary fuel model, with live woody and herbaceous
        FuelModel[0]=1.0;       // 1hr loading t/a
        FuelModel[1]=0.0;       // 10hr loading t/a
        FuelModel[2]=2.0;       // 100hr loading t/a
        FuelModel[3]=0.0;       // live herb loading t/a
        FuelModel[4]=0.5;       // live woody loading t/a
        FuelModel[5]=0.5;       // surf 1hr 1/ft
        FuelModel[6]=1000.0;    // surf live herb 1/ft
        FuelModel[7]=500.0;     // surf live woody 1/ft
        FuelModel[8]=8000.0;    // heat content dead btu/lb
        FuelModel[9]=9000.0;    // heat content live herb btu/lb
        FuelModel[10]=9000.0;   // heat content live herb btu/lb
        FuelModel[11]=0.6;      // depth (ft)
        FuelModel[12]=0.25;     // extinction mx (0-1)

        spreadrate
}
*/



double CalcSpreadRate(double *Fuel, double *Moisture, double WindSpeed,
                      double Slope, double *FlameLength,
                      double *FirelineIntensity, double *HeatPerUnitArea)
{// Rothermel spread equation based directly on BEHAVE source code
     long i, j, ndead=0, nlive=0;
     double seff[3][2]={{.01,.01},{.01,.01},{.01,0}};     //mineral content
     double wtfact, fined=0, finel=0, wmfd=0, fdmois=0, w=0, wo=0, beta;
     double rm, sigma=0, rhob=0, sum3=0, betaop=0, rat, aa, gammax=0, gamma=0,
wind=0;
     double xir, rbqig=0, xi=0, b, c, e, part1=0, slopex=0;
     double ewind, wlim, sum1=0, sum2=0, phis, phiw, phiew;
     double rateo, SpreadRate;



     double mois[3][2]=                            // fraction of oven-dry weight
         {       {Moisture[0], Moisture[3]},
                 {Moisture[1], Moisture[4]},
                 {Moisture[2], 0.0},
         };

     if(Fuel[0]) ndead++;
     if(Fuel[1]) ndead++;
     if(Fuel[2]) ndead++;
     if(Fuel[3]) nlive++;
     if(Fuel[4]) nlive++;

     if(nlive>0)
         nlive=2;                    // boost to max number
     if(ndead>0)
         ndead=3;

     double nclas[2]={ndead,nlive};  // # of dead & live fuel classes

     double load[3][2]=                            // tons per acre, later converted to
lb/ft2
     {       {Fuel[0], Fuel[3]},
             {Fuel[1], Fuel[4]},
             {Fuel[2], 0.0},
     };

     double sav[3][2]=                      // 1/ft
```

```
{       {Fuel[5], Fuel[6]},
        {109.0, Fuel[7]},
        {30.0, 0.0},
};

  double heat[3][2]=
{     {Fuel[8], Fuel[9]},
      {Fuel[8], Fuel[10]},
      {Fuel[8], 0.0},
};

  double depth=Fuel[11];

  double wn[3][2]={{0,0},{0,0},{0,0}};
  double qig[3][2]={{0,0},{0,0},{0,0}};
  double a[3][2]={{0,0},{0,0},{0,0}};
  double f[3][2]={{0,0},{0,0},{0,0}};
  // double g[3][2]={0,0,0,0,0,0,};
  double ai[2]={0,0};
  double fi[2]={0,0};
  double hi[2]={0,0};
  double se[2]={0,0};
  double xmf[2]={0,0};
  double si[2]={0,0};
  double wni[2]={0,0};
  double etam[2]={0,0};
  double etas[2]={0,0};
  double rir[2]={0,0};
  double xmext[2]={Fuel[12], 0};

  wind=WindSpeed*88.0;                          // ft/minute
  slopex=Slope/100.0; //tan((double) Slope/180.0*PI);   // convert from
degrees to tan

  // fuel weighting factors
  for(i=0; i<2; i++)
  {       for(j=0; j<nclas[i]; j++)
          {    a[j][i]=load[j][i]*sav[j][i]/32.0;
               ai[i]=ai[i]+a[j][i];
               wo=wo+0.04591*load[j][i];
          }
          if(nclas[i]!=0)
          {       for(j=0;j<nclas[i];j++)
                  {    if(ai[i]>0.0)
                          f[j][i]=a[j][i]/ai[i];
                  else
                          f[j][i]=0.0;
                  }
          }
  }
  fi[0]=ai[0]/(ai[0]+ai[1]);
  fi[1]=1.0-fi[0];

  /* no need for this, because extinction moistures are assigned */
  /* as on last page of Burgan and Rothermel 1984 */
  /*      rhob=(wo/depth);
          beta=rhob/32;
          xmext[0]=.12+4.*beta;
  */

  //moisture of extinction
  if(nclas[1]!=0)
  {       for(j=0; j<nclas[0]; j++)
```

```
{       wtfact=load[j][0]*exp(-138.0/sav[j][0]);
            fined=fined+wtfact;
            wmfd=wmfd+wtfact*mois[j][0];
}
        fdmois=wmfd/fined;
        for(j=0; j<nclas[1]; j++)
            finel=finel+load[j][1]*exp(-500.0/sav[j][1]);
        w=fined/finel;
        xmext[1]=2.9*w*(1.0-fdmois/xmext[0])-0.226;
        if(xmext[1]<xmext[0])
            xmext[1]=xmext[0];
}


// intermediate calculations, summing parameters by fuel component
for(i=0;i<=1;i++)
{       if(nclas[i]!=0)
        {       for(j=0;j<nclas[i];j++)
                {       wn[j][i]=0.04591*load[j][i]*(1-0.0555);
                        qig[j][i]=250.0+1116.0*mois[j][i];
                        hi[i]=hi[i]+f[j][i]*heat[j][i];
                        se[i]=se[i]+f[j][i]*seff[j][i];
                        xmf[i]=xmf[i]+f[j][i]*mois[j][i];
                        si[i]=si[i]+f[j][i]*sav[j][i];
                        sum1=sum1+0.04591*load[j][i];
                        sum2=sum2+0.04591*load[j][i]/32.0;
                        sum3=sum3+fi[i]*f[j][i]*qig[j][i]*exp(-
138.0/sav[j][i]);
                }
                for(j=0; j<nclas[i]; j++)
                    wni[i]=wni[i]+f[j][i]*wn[j][i];   /* g[j][i] should be
subst for f[j][i] in the wni[i] equation */
                                                          /* if the
above g-factors are calculated */
                rm=xmf[i]/xmext[i];
                etam[i]=1.0-2.59*rm+5.11*pow2(rm)-3.52*pow(rm,3.0);
                if(xmf[i] >= xmext[i])
                        etam[i]=0;
                etas[i]=0.174/(pow(se[i],0.19));
                if(etas[i]>1.0)
                        etas[i]=1.0;
                sigma=sigma+fi[i]*si[i];
                rir[i]=wni[i]*hi[i]*etas[i]*etam[i];
        }
}


/* final calculations */
rhob=sum1/depth;
beta=sum2/depth;
betaop=3.348/pow(sigma,0.8189);
rat=beta/betaop;
aa=133.0/pow(sigma,0.7913);
gammax=pow(sigma,1.5)/(495.0+0.0594*pow(sigma,1.5));
gamma=gammax*pow(rat,aa)*exp(aa*(1.0-rat));
xir=gamma*(rir[0]+rir[1]);
rbqig=rhob*sum3;
xi=exp((0.792+0.681*pow(sigma,0.5))*(beta+0.1))/(192.0+0.2595*sigma);
/*      flux=xi*xir;*/
rateo=xir*xi/rbqig;    /* this is in English units */

phis=5.275*pow(beta,-0.3)*pow2(slopex);
        c=7.47*exp(-0.133*pow(sigma,0.55));
        b=0.02526*pow(sigma,0.54);
        e=0.715*exp(-0.000359*sigma);
```

201

```
              part1=c*pow(rat,-e);
          phiw=pow(wind,b)*part1;

          wlim=0.9*xir;

          SpreadRate=(rateo*(1.0+phiw+phis));                    // ft/min
              *FirelineIntensity=384.0*xir*SpreadRate/(60.0*sigma);   // btu/ft/sec
                *FlameLength=0.45*pow(*FirelineIntensity,0.46);       // ft
                *HeatPerUnitArea=xir*384.0/sigma;                     // btu/ft2




/*      maximum windspeed effect on ros*/
          phiew=phiw+phis;
          ewind=pow(((phiew*pow(rat,e))/c),1.0/b);

          if(ewind>wlim)
          {       ewind=wlim;
                  phiew=c*pow(wlim,b)*pow(rat,-e);
                  SpreadRate=rateo*(phiew+1.0);
                  *FirelineIntensity=384.0*xir*SpreadRate/(60.0*sigma);
                  *FlameLength=0.45*pow(*FirelineIntensity, 0.46);
          }
//      react=xir*1.633;                                // convert btu/f2/s to kW/m2

          return SpreadRate;

}
///////////////////////////////////////////////////////////////////
///////////////////////////////////////////////////////////////////
///////////////////////////////////////////////////////////////////
///////////////////////////////////////////////////////////////////

void derivs(double x,double y[],double dydx[])
{
          dydx[1] = Rhoa*(Alpha*(y[2]/y[1]-
(windprofile(in,y[6]+flameheight))*cos(y[3]))+Beta*(windprofile(in,y[6]+flamehe
ight))*sin(y[3]));
          dydx[2]=Rhoa*(Alpha*(y[2]/y[1]-
(windprofile(in,y[6]+flameheight))*cos(y[3]))+Beta*(windprofile(in,y[6]+flamehe
ight))*sin(y[3]))*(windprofile(in,y[6]+flameheight))*cos(y[3])+((y[1]*y[4]*g*si
n(y[3]))/(y[2]*in.Ta));
          dydx[3]=-(Rhoa*(Alpha*(y[2]/y[1]-
(windprofile(in,y[6]+flameheight))*cos(y[3]))+Beta*(windprofile(in,y[6]+flamehe
ight))*sin(y[3]))*(windprofile(in,y[6]+flameheight))*sin(y[3])-
((y[1]*y[4]*g*cos(y[3]))/(y[2]*in.Ta)))/y[2];
          dydx[4]=0.0*y[1]*sin(y[3]);
      dydx[5]=cos(y[3]);
      dydx[6]=sin(y[3]);
}

void locate(double x2[],unsigned long n,double x,long *j)
{
      unsigned long ju,jm,jl;         //function locates the value j such that x
      int ascnd;                      //is between x2[j] and x2[j+1];x2 must be
                                      //monotonic (either always increasing or
      jl=0;                           //always decreasing)
      ju=n+1;
      ascnd=(x2[n]>=x2[1]);           //x2[] is a vector (can also pass a row or
      while(ju-jl>1)                  //column of a matrix)
      {
          jm=(ju+jl)>>1;
          if(x>=x2[jm]==ascnd)
```

```
                jl=jm;
            else
                ju=jm;
    }
    if(x==x2[1])*j=1;
    else if(x==x2[n])*j=n-1;
    else *j=jl;
}

double linterp(double X[],double Y[],long n,double xi)
{
    long J;
    double yi,slope,b;
    locate(X,n,xi,&J);
    slope=(Y[J+1]-Y[J])/(X[J+1]-X[J]);
    b=Y[J]-slope*X[J];
    yi=slope*xi+b;
    return yi;
}

double ptinterp(double x1,double x2,double y1,double y2,double xi)
{
    double yi,slope,b;
    slope=(y2-y1)/(x2-x1);
    b=y1-slope*x1;
    yi=slope*xi+b;
    return yi;
}

void fillconvect0(void)
    {
        long jleft,jright,i,k,split;
        double maxsize=0.02;              //maximum x-distance between values in
the convect array
        double dist,diff,size;
        double N1=1.35, lamda=1.0;       //constants controlling temp/vel
profiles
        locate(xx,NSTEP,s_ltcanopyht,&jleft);
        locate(xx,NSTEP,s_rtcanopyht,&jright);
        jleft++;
        jright++;
        NPTS=jright-jleft+1;
        convect=matrix(1,3,1,NPTS+1);
        convect[1][1]=xplumeleft;
        convect[2][1]=in.Ta;
        convect[3][1]=0.0;
        i=2;
        for(j=jleft;j<jright;j++)
        {
            convect[1][i]=(in.canbaseht-rawplume[5][j]-
rawplume[4][j]/(tan(rawplume[6][j])))*tan(rawplume[6][j])*(-1.0);
            dist=pow((pow2(rawplume[4][j]-convect[1][i])+pow2(in.canbaseht-
rawplume[5][j])),0.5);
            convect[2][i]=in.Ta+(N1/pow2(lamda))*(rawplume[1][j]-in.Ta)*exp(-
(pow2(dist)/(pow2(lamda)*pow2(rawplume[3][j]))));
            convect[3][i]=(N1/pow2(lamda))*(rawplume[2][j])*exp(-
(pow2(dist)/(pow2(lamda)*pow2(rawplume[3][j]))));
            diff=convect[1][i]-convect[1][i-1];
            if(diff>maxsize)
            {
                split=ceil(diff/maxsize);
                size=diff/(double)split;
                convect2=matrix(1,3,1,NPTS+1+split-1);
```

```
                    for(k=1;k<i;k++)
                    {
                        convect2[1][k]=convect[1][k];
                        convect2[2][k]=convect[2][k];
                        convect2[3][k]=convect[3][k];
                    }
                    free_matrix(convect,1,3,1,NPTS+1);
                    NPTS=NPTS+split-1;
                    convect=matrix(1,3,1,NPTS+1);
                    for(k=1;k<i;k++)
                    {
                        convect[1][k]=convect2[1][k];
                        convect[2][k]=convect2[2][k];
                        convect[3][k]=convect2[3][k];
                    }
                    free_matrix(convect2,1,3,1,NPTS+1+split-1);
                    convect[1][i+split-1]=(in.canbaseht-rawplume[5][j]-
rawplume[4][j]/(tan(rawplume[6][j])))*tan(rawplume[6][j])*(-1.0);
                    //                  convect[1][i+split-1]=(in.canbaseht-
(rawplume[5][j]+1.0/rawplume[6][j]*rawplume[4][j]))*tan(rawplume[6][j])*(-1.0);
                    dist=pow((pow2(rawplume[4][j]-convect[1][i+split-
1])+pow2(in.canbaseht-rawplume[5][j])),0.5);
                    convect[2][i+split-1]=in.Ta+(N1/pow2(lamda))*(rawplume[1][j]-
in.Ta)*exp(-(pow2(dist)/(pow2(lamda)*pow2(rawplume[3][j]))));
                    convect[3][i+split-1]=(N1/pow2(lamda))*(rawplume[2][j])*exp(-
(pow2(dist)/(pow2(lamda)*pow2(rawplume[3][j]))));
//                      printf("\ndiff>maxsize\ti=%ld",i);
                    for(k=i;k<i+split-1;k++)
                    {
//                      printf("\ninner-loop");
                        convect[1][k]=convect[1][k-1]+size;
                        convect[2][k]=ptinterp(convect[1][i-1],convect[1][i+split-
1],convect[2][i-1],convect[2][i+split-1],convect[1][k]);
                        convect[3][k]=ptinterp(convect[1][i-1],convect[1][i+split-
1],convect[3][i-1],convect[3][i+split-1],convect[1][k]);
                    }
                    i=i+split;
                    continue;
                }
                i++;
            }
        }

void fillconvect1(void)
{
    double cellsize=0.02;              //maximum x-distance between values in the
convect array
    double N1=1.35, lamda=1.0;        //constants controlling temp/vel profiles
    double leftdist,rightdist,dist;
    double Tplumemid,Uplumemid;        //temp/vel at plume centerline and canopy
height
    Tplumemid=linterp(xx,rawplume[1],NSTEP,s_canopyht);
    Uplumemid=linterp(xx,rawplume[2],NSTEP,s_canopyht);
    leftdist=xplumemid-xplumeleft;
    rightdist=xplumeright-xplumemid;
    long lnum,rnum,i;
    lnum=ceil(leftdist/cellsize);
    rnum=ceil(rightdist/cellsize);
    NPTS=lnum+rnum+1;
    convect=matrix(1,3,1,NPTS+1);
    for(i=1;i<=lnum+1;i++)
    {
        convect[1][i]=xplumemid-(lnum-i+1)*cellsize;
```

```
            dist=(lnum-i+1)*cellsize;
            convect[2][i]=in.Ta+(N1/pow2(lamda))*(Tplumemid-in.Ta)*exp(-
(pow2(dist)/(pow2(lamda)*pow2(lnum*cellsize))));
            convect[3][i]=(N1/pow2(lamda))*(Uplumemid)*exp(-
(pow2(dist)/(pow2(lamda)*pow2(lnum*cellsize))));
        }
//      convect[1][lnum+1]=xplumemid;
//      convect[2][lnum+1]=Tplumemid;
//      convect[3][lnum+1]=Uplumemid;
        for(i=lnum+2;i<=NPTS;i++)
        {
            convect[1][i]=xplumemid+(i-lnum-1)*cellsize;
            dist=(i-lnum-1)*cellsize;
            convect[2][i]=in.Ta+(N1/pow2(lamda))*(Tplumemid-in.Ta)*exp(-
(pow2(dist)/(pow2(lamda)*pow2(rnum*cellsize))));
            convect[3][i]=(N1/pow2(lamda))*(Uplumemid)*exp(-
(pow2(dist)/(pow2(lamda)*pow2(rnum*cellsize))));
        }
}

double gettemp(double x)
{
    double temp;
    if(x<=xplumeleft)
    {
        temp=in.Ta;
    }else if(x>=xplumeright)
    {
        temp=in.Ta;
    }else
    {
        temp=linterp(convect[1],convect[2],NPTS,x);
    }
    return temp;
}

double getspeed(double x)
{
    double speed;
    if(x<=xplumeleft)
    {
        speed=windprofile(in,in.canbaseht);
    }else if(x>=xplumeright)
    {
        speed=windprofile(in,in.canbaseht);
    }else
    {
        speed=linterp(convect[1],convect[3],NPTS,x);
    }
    return speed;
}


void plumemodel(void)
{
        double *vstart;


    do
    {
        vstart=vector(1,NVAR);
        /* Note: The arrays xx and y must have indices up to NSTEP+1 */
        xx=vector(1,NSTEP+1);
```

```
        y=matrix(1,NVAR,1,NSTEP+1);

vstart[1]=(rhoa*(in.Ta/Ti))*(flamedepth/2.0)*(pow(((2.0*g*iByram)/(rhoa*Cp*(in.
Ta))),(1.0/3.0)));

vstart[2]=(rhoa*(in.Ta/Ti))*(flamedepth/2.0)*(pow((pow(((2.0*g*iByram)/(rhoa*Cp
*(in.Ta))),(1.0/3.0))),2.0));
        vstart[3]=PI/2.0;

vstart[4]=(rhoa*(in.Ta/Ti))*(flamedepth/2.0)*(pow(((2.0*g*iByram)/(rhoa*Cp*(in.
Ta))),(1.0/3.0)))*(Ti-in.Ta);
        vstart[5]=0.0;
        vstart[6]=0.0;
        rkdumb(vstart,NVAR,x1,x2,NSTEP,derivs);
        free_vector(vstart,1,NVAR);
        rawplume=matrix(1,6,1,NSTEP+1);
        for(j=1;j<=NSTEP;j++)
        {

//          printf("\nTa %lf\ty4 %lf\ty1 %lf",in.Ta,y[4][j],y[1][j]);
        rawplume[1][j]=in.Ta+y[4][j]/y[1][j];              //Tp (plume
temperature)
        rawplume[2][j]=y[2][j]/y[1][j];                     //Up (plume velocity)
        rawplume[3][j]=y[1][j]*y[1][j]/(y[2][j]*(Rhoa*in.Ta/rawplume[1][j]));
//b (plume half-width)
        rawplume[4][j]=y[5][j]-flamedepth/2.0;           //x (horizontal
distance from the point of interest to the flame leading edge)
        rawplume[5][j]=y[6][j]+flameheight;              //z (vertical distance
from the point of interest to the ground)
        rawplume[6][j]=y[3][j];                           //theta (angle between
s line and horizontal)
        }
        free_matrix(y,1,NVAR,1,NSTEP+1);
        if(flameheight>=in.canbaseht)                    //exit program if
flames touch the canopy
        {
            printf("\n*************Crown fire initiation HAS
occured.*************\n**The flame height is greater than the canopy base
height.**\n");
            exit(1);
        }
        s_canopyht=linterp(rawplume[5],xx,NSTEP,in.canbaseht);        //get the s
value at the canopy base height
        xplumemid=linterp(xx,rawplume[4],NSTEP,s_canopyht);          //get the x
value where the plume centerline crosses the canopy base height

        calc_xplumert();

    }while(px2!=x2);

    calc_xplumelt();


    if(convectflag<=0){
        fillconvect0();
    }else{
        fillconvect1();
    }
//    for(j=1;j<=NSTEP;j++)
//    {
//          if(j%200==0)
```

```c
//
printf("\n%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t%lf\t\n",xx[j],rawplume[1][j],rawplume[
2][j],rawplume[3][j],rawplume[4][j],rawplume[5][j]-1.6,rawplume[6][j]);
//     }

}

void calc_xplumert(void)
{
  do{
    j=0;
    px2=x2;
    do
    {
        j++;
        if(j>NSTEP)
        {   x2+=2.0;
            printf("Trying x2=%lf\n", x2);
            break;
        }
        zinternrt2=rawplume[3][j]*cos(rawplume[6][j]);
//        printf("%ld\n", j);
    }while((rawplume[5][j]-zinternrt2)<in.canbaseht);
    if(px2!=x2)
        break;
    zerror2=(rawplume[5][j]-zinternrt2)-in.canbaseht;
//    printf("\nj is %i\n",j);
//    printf("\nzerror2 is %lf\n",zerror2);
    j--;
    zinternrt1=rawplume[3][j]*cos(rawplume[6][j]);
    zerror1=(rawplume[5][j]-zinternrt1)-in.canbaseht;
    s_rtcanopyht=ptinterp(zerror1,zerror2,xx[j],xx[j+1],0.0);          //s
value of the right(down wind) side of the plume at canopy height
//    printf("\nj is %i\n",j);
//    printf("\nzerror1 is %lf\n",zerror1);
//    printf("\ns_rtcanopyht is %lf\n",s_rtcanopyht);
//    double check;
//    check=linterp(xx,rawplume[5],NSTEP,s_rtcanopyht)-zinternrt1+flameheight;
//    printf("\ncheck is %lf\n",check);

xinternrt=linterp(xx,rawplume[3],NSTEP,s_rtcanopyht)*sin(linterp(xx,rawplume[6]
,NSTEP,s_rtcanopyht));
    xplumeright=xinternrt+linterp(xx,rawplume[4],NSTEP,s_rtcanopyht);        //x
value of the right side of the plume at the canopy base height
//    printf("\nxplumeright is %lf\n",xplumeright);
  }while(px2!=x2);
}

void calc_xplumelt(void)
{
    j=0;
    do
    {
        j++;
        zinternlt2=rawplume[3][j]*sin((PI/2.0)-rawplume[6][j]);
    }while((rawplume[5][j]+zinternlt2)<in.canbaseht);
    zerror2=(rawplume[5][j]+zinternlt2)-in.canbaseht;
//    printf("\nj is %i\n",j);
//    printf("\nzerror2 is %lf\n",zerror2);
    j--;
    zinternlt1=rawplume[3][j]*sin((PI/2.0)-rawplume[6][j]);
    zerror1=(rawplume[5][j]+zinternlt1)-in.canbaseht;
```

207

```
        s_ltcanopyht=ptinterp(zerror1,zerror2,xx[j],xx[j+1],0.0);              //s
value of the left (up wind) side of the plume at canopy height
//      printf("\nj is %i\n",j);
//      printf("\nzerror1 is %lf\n",zerror1);
//    printf("\ns_ltcanopyht is %lf\n",s_ltcanopyht);
//      check=linterp(xx,rawplume[5],NSTEP,s_ltcanopyht)+zinternlt1+flameheight;
//      printf("\ncheck is %lf\n",check);
      xinternlt=linterp(xx,rawplume[3],NSTEP,s_ltcanopyht)*cos((PI/2.0)-
(linterp(xx,rawplume[6],NSTEP,s_ltcanopyht)));
      xplumeleft=linterp(xx,rawplume[4],NSTEP,s_ltcanopyht)-xinternlt;        //x
value of the left side of the plume at the canopy base height
}


double fxn(double pt[],double wgt)   //function inside the quadruple integral
for radiation transfer
{
       double ans,Trad,S;

//     Trad=(maxflmtemp-in.igtemp)/flamedepth*pt[1]+maxflmtemp;
       Trad=radflametemp(-pt[1]);
       S=pow((pow(pt[4]-pt[2],2.0)+pow(in.canbaseht,2.0)+pow(pt[1]-
pt[3],2.0)),0.5);
      ans=(femiss*SB*pow(Trad,4.0)*exp(-
attenuation*S)*pow(in.canbaseht/S,2.0))/(PI*pow(S,2.0));
       return ans;
}


double getrad(double Dx,double Tp)
{
   double Ap;                //Ap is the area of the particle
   double rad,I12;           //I12 is the irradiation from flame to fuel particle
   if(radflag<=0)
   {
     int init,itmax,j,ncall,nprn=0;      //This function uses the VEGAS Monte
Carlo program from Numerical Recipes in C on
       double avgi,chi2a,sd,xoff;         //page 320.  Most of the variables
used here are described in the book.
       double *regn;                      //Dx is distance from flame leading
edge to particle center; Tp is particle temperature.
     double temp;
     long i;
     timeb t;            //from the header file <sys/timeb.h>

       regn=vector(1,8);
//     printf("IDUM=\n");
//     scanf("%ld",&idum);
     ftime(&t);                  //set seed (idum) for rand2 using time from the
system
       srand(t.time+t.millitm);
     idum=-((rand()%10000)+1);

       if (idum > 0) idum = -idum;

//             printf("ENTER NCALL,ITMAX,NPRN\n");
//             if (scanf("%d %d %d",&ncall,&itmax,&nprn) == EOF) break;
       avgi=sd=chi2a=0.0;
     regn[1]=-flamedepth;      //set the limits of integration
     regn[2]=-Wf/2.0;          // 1-4 are the "lower limits" and 5-8 are the
"upper limits"
     regn[3]=Dx-Lp/2.0;        //   1&5 = flame "x" limits
     regn[4]=-Wp/2.0;          //   2&6 = flame "y" limits
     regn[5]=0.0;              //   3&7 = particle "x" limits
     regn[6]=Wf/2.0;           //   4&8 = particle "y" limits
```

```c
    regn[7]=Dx+Lp/2.0;
    regn[8]=Wp/2.0;

//    maxflmtemp=maxflametemp(wind);

        init = 0;
    ncall=1000;                     //controls the number of iterations in the monte-
carlo algorithm for radiation
    itmax=5;
        vegas(regn,ndim,fxn,init,ncall,itmax,nprn,&avgi,&sd,&chi2a);
//      printf("Number of iterations performed: %d\n",itmax);
//      printf("Integral, Standard Dev., Chi-sq. = %12.6f %12.6f% 12.6f\n",
//             avgi,sd,chi2a);
        init = 1;
    ncall=10000;                    //controls the number of iterations in the monte-
carlo algorithm for radiation
    itmax=1;
        vegas(regn,ndim,fxn,init,ncall,itmax,nprn,&avgi,&sd,&chi2a);
//      printf("Additional iterations performed: %d \n",itmax);
//      printf("Integral, Standard Dev., Chi-sq. = %12.6f %12.6f% 12.6f\n",
//                  avgi,sd,chi2a);

//      for(i=0;i<=5;i++){
//      init = 2;
//      ncall=100000;
//      itmax=1;
//      vegas(regn,ndim,fxn,init,ncall,itmax,nprn,&avgi,&sd,&chi2a);
//      printf("Additional iterations performed: %d \n",itmax);
//      printf("Integral, Standard Dev., Chi-sq. = %12.6f %12.6f% 12.6f\n",
//                  avgi,sd,chi2a);
//      }
    Ap=Wp*Lp;
    I12=avgi/Ap;     //avgi is the total radiation to the particle; Ap is the
assumed rectangular particle area for radiation calculation
    rad=I12*pemiss*part_surf_area*0.5-part_surf_area*(pemiss*SB*pow(Tp,4.0)-
SB*pow(in.Ta,4.0));
                            //note that energy recieved by the particle from the
surroundings is not accounted for; assumed negligible
//      printf("\nI12=%lf  rad=%lf",I12,rad);
//      printf("\nradflux = %lf\n",rad);
//      for(i=0;i<=301;i++)
//      {
//          temp=(maxflmtemp-in.igtemp)/flamedepth*(double)i/(-100.0)+maxflmtemp;
//          printf("\nX = %lf\ttemperature = %lf",(double)(i)/(-100.0),temp);
//      }

        free_vector(regn,1,20);
//      printf("\n\nNormal completion\n");
        return rad;
    }else{
    gauleg(0.0,1.0,n);
    I12=qgaus(f1,Y1,Y2,Dx,Tp);
    rad=I12*pemiss*part_surf_area*0.5-part_surf_area*(pemiss*SB*pow(Tp,4.0)-
SB*pow(in.Ta,4.0));
                            //note that energy recieved by the particle from the
surroundings is not accounted for; assumed negligible
    printf("\nDx=%lf flame=%lf loss=%lf rad=%lf
I=%lf",Dx,I12*pemiss*part_surf_area*0.5,-part_surf_area*(pemiss*SB*pow(Tp,4.0)-
SB*pow(in.Ta,4.0)),rad,I12);

    return rad;
    }
}
```

```
double integration_function(double x,double y,double Dx,double Tp)
{                                   //function integrated by gaussian quadrature to
obtain radiant flux to fuel particle
     double ans,Trad,S;
     Trad=radflametemp(-x);
     S=pow((pow(0.0-y,2.0)+pow(in.canbaseht,2.0)+pow(x-Dx,2.0)),0.5);
     ans=(femiss*SB*pow(Trad,4.0)*exp(-
attenuation*S)*pow(in.canbaseht/S,2.0))/(PI*pow(S,2.0));
     return ans;
}


double f1(double y,double Dx,double Tp)
{                                   //used in gaussian quadrature integration to
obtain radiant flux to fuel particle
//     i++;
//     printf("\nf1 %ld",i);
     ysav=y;
     return qgaus(f2,xx1(y),xx2(y),Dx,Tp);
}


double f2(double x,double Dx,double Tp)
{                                   //used in gaussian quadrature integration to
obtain radiant flux to fuel particle
//     j++;
//     printf("\nf2 %ld",j);
     return integration_function(ysav,x,Dx,Tp);
}


double xx1(double y)
{                                   //used in gaussian quadrature integration to
obtain radiant flux to fuel particle
     return 0.0;
}


double xx2(double y)
{                                   //used in gaussian quadrature integration to
obtain radiant flux to fuel particle
     return in.canbaseht;
}

double qgaus(double (*func)(double,double,double), double a, double b,double
Dx,double Tp)
{                                   //gaussian quadrature function
      long j;
      double xr,xm,dx,s;
//      static double x[]={0.0,0.1488743389,0.4333953941,
//            0.6794095682,0.8650633666,0.9739065285};
//      static double w[]={0.0,0.2955242247,0.2692667193,
//            0.2190863625,0.1494513491,0.0666713443};

      xm=0.5*(b+a);
      xr=0.5*(b-a);
      s=0;
      for (j=1;j<=n;j++) {
            dx=xr*x[j];
            s += w[j]*((*func)(xm+dx,Dx,Tp)+(*func)(xm-dx,Dx,Tp));
      }
      return s *= xr;
}

void gauleg(double x1, double x2, long n)
```

210

```
{                                                       //function to give absicissas and weights for
the Gaus-Legendre case of Gaussian quadrature
        long m,j,i;
        double z1,z,xm,x1,pp,p3,p2,p1;

        m=(n+1)/2;
        xm=0.5*(x2+x1);
        x1=0.5*(x2-x1);
        for (i=1;i<=m;i++) {
                z=cos(3.141592654*(i-0.25)/(n+0.5));
                do {
                        p1=1.0;
                        p2=0.0;
                        for (j=1;j<=n;j++) {
                                p3=p2;
                                p2=p1;
                                p1=((2.0*j-1.0)*z*p2-(j-1.0)*p3)/j;
                        }
                        pp=n*(z*p1-p2)/(z*z-1.0);
                        z1=z;
                        z=z1-p1/pp;
                } while (fabs(z-z1) > EPS);
                x[i]=xm-x1*z;
                x[n+1-i]=xm+x1*z;
                w[i]=2.0*x1/((1.0-z*z)*pp*pp);
                w[n+1-i]=w[i];
        }
}


double radflametemp(double X)
{
    double beta=8.0,k=1.0,Adimen600,rise,ig,decay,coef,res,t;

    t=X/ROS;
    Adimen600=(in.igtemp-in.Ta)/(maxflmtemp-in.Ta);
    rise=sqrt(-1.0*pow(beta,2.0)*(log(Adimen600)));
    ig=-rise;
    decay=(-1.0*(log(Adimen600)))/(taur+ig);
    if((t+ig)<0.0)
    {
        res=k*exp(-1.0*(pow((t+ig),2.0))/pow(beta,2.0))*(maxflmtemp-
in.Ta)+in.Ta;
    }else
    {
        res=k*exp(-1.0*(decay*(t+ig)))*(maxflmtemp-in.Ta)+in.Ta;
    }
    return(res);
}


double getconvectflux(double Dx,double Tp)   //Dx is distance from flame leading
edge to particle center, Tp is particle temperature.
{
    double
Tfilm,Tair,Vair,rhoair,cpair,visco,kinemvis,thermcond,thermdif,Reynolds,Nusselt
,h,flux;

    Tair=gettemp(Dx);
    Vair=getspeed(Dx);
    Tfilm=(Tair+Tp)/2.0;
    rhoair=358.98*pow(Tfilm,-1.0046);
    cpair=-3.0*pow(10,-10.0)*pow(Tfilm,3.0)+7.0*pow(10.0,-7.0)*pow2(Tfilm)-
0.0003*Tfilm+1.0486;
```

```c
        visco=pow(10.0,-14.0)*pow(Tfilm,3.0)-4.0*pow(10.0,-
11.0)*pow2(Tfilm)+7.0*pow(10.0,-8.0)*Tfilm+8.0*pow(10.0,-7.0);
        kinemvis=visco/rhoair;
        thermcond=2.0*pow(10.0,-11.0)*pow(Tfilm,3.0)-6.0*pow(10.0,-
8.0)*pow2(Tfilm)+0.0001*Tfilm-0.0015;
        Reynolds=(Vair*in.diameter)/kinemvis;
        Nusselt=0.1417*pow(Reynolds,0.6053);
        h=Nusselt*thermcond/in.diameter;          //in W/m^2/K
//      printf("\nh= %lf\n",h);
//      printf("\nTair= %lf\n",Tair);
        fprintf(fout,"%lf\t",Tair);
        flux=h*(Tair-Tp);          //positive flux means energy goes INTO the fuel
particle
        return(flux);
}

#undef NRANSI
```

212

Random number generator function

```cpp
//ran2.cpp - Random number generator function

#define IM1 2147483563
#define IM2 2147483399
#define AM (1.0/IM1)
#define IMM1 (IM1-1)
#define IA1 40014
#define IA2 40692
#define IQ1 53668
#define IQ2 52774
#define IR1 12211
#define IR2 3791
#define NTAB 32
#define NDIV (1+IMM1/NTAB)
#define EPS 1.2e-7
#define RNMX (1.0-EPS)

double ran2(long *idum)
{
        int j;
        long k;
        static long idum2=123456789;
        static long iy=0;
        static long iv[NTAB];
        double temp;

        if (*idum <= 0) {
                if (-(*idum) < 1) *idum=1;
                else *idum = -(*idum);
                idum2=(*idum);
                for (j=NTAB+7;j>=0;j--) {
                        k=(*idum)/IQ1;
                        *idum=IA1*(*idum-k*IQ1)-k*IR1;
                        if (*idum < 0) *idum += IM1;
                        if (j < NTAB) iv[j] = *idum;
                }
                iy=iv[0];
        }
        k=(*idum)/IQ1;
        *idum=IA1*(*idum-k*IQ1)-k*IR1;
        if (*idum < 0) *idum += IM1;
        k=idum2/IQ2;
        idum2=IA2*(idum2-k*IQ2)-k*IR2;
        if (idum2 < 0) idum2 += IM2;
        j=iy/NDIV;
        iy=iv[j]-idum2;
        iv[j] = *idum;
        if (iy < 1) iy += IMM1;
        if ((temp=AM*iy) > RNMX) return RNMX;
        else return temp;
}
#undef IM1
#undef IM2
#undef AM
#undef IMM1
#undef IA1
#undef IA2
#undef IQ1
#undef IQ2
#undef IR1
#undef IR2
```

```
#undef NTAB
#undef NDIV
#undef EPS
#undef RNMX
```

214

rebin function for MonteCarlo integration

```
//rebin.cpp - Function used on MonteCarlo integration

void rebin(double rc, int nd, double r[], double xin[], double xi[])
{
        int i,k=0;
        double dr=0.0,xn=0.0,xo=0.0;

        for (i=1;i<nd;i++) {
                while (rc > dr)
                        dr += r[++k];
                if (k > 1) xo=xi[k-1];
                xn=xi[k];
                dr -= rc;
                xin[i]=xn-(xn-xo)*dr/r[k];
        }
        for (i=1;i<nd;i++) xi[i]=xin[i];
        xi[nd]=1.0;
}
```

rk4 function for Runge-Kutta solution

```cpp
//rk4.cpp - Function used in Runge-Kutta solution

#define NRANSI
#include "nrutilcpp.h"

/*typedef struct input {
     double u10;double slope;double Ta;double sh;double alpha;double igtemp;
     long FuelModelNumber;double FuelMoisture[5];double rho_surf;double
delta;double Wa;double sigma_surf;double hc;
     double sigma_can;

}Input;
Input in;
*/
void rk4(double y[], double dydx[], int n, double x, double h, double yout[],
        void (*derivs)(double, double [], double []))
{
        int i;
        double xh,hh,h6,*dym,*dyt,*yt;

        dym=vector(1,n);
        dyt=vector(1,n);
        yt=vector(1,n);
        hh=h*0.5;
        h6=h/6.0;
        xh=x+hh;
        for (i=1;i<=n;i++) yt[i]=y[i]+hh*dydx[i];
        (*derivs)(xh,yt,dyt);
        for (i=1;i<=n;i++) yt[i]=y[i]+hh*dyt[i];
        (*derivs)(xh,yt,dym);
        for (i=1;i<=n;i++) {
                yt[i]=y[i]+h*dym[i];
                dym[i] += dyt[i];
        }
        (*derivs)(x+h,yt,dyt);
        for (i=1;i<=n;i++)
                yout[i]=y[i]+h6*(dydx[i]+dyt[i]+2.0*dym[i]);
        free_vector(yt,1,n);
        free_vector(dyt,1,n);
        free_vector(dym,1,n);
}
#undef NRANSI
```

rkdumb function for Runge-Kutta solution

```cpp
//rkdumb.cpp - Function used in Runge-Kutta solution

#define NRANSI
#include "nrutilcpp.h"

double **y,*xx;

/*typedef struct input {
    double u10;double slope;double Ta;double sh;double alpha;double igtemp;
    long FuelModelNumber;double FuelMoisture[5];double rho_surf;double
delta;double Wa;double sigma_surf;double hc;
    double sigma_can;

}Input;
Input in;
*/

void rkdumb(double vstart[], int nvar, double x1, double x2, int nstep,
        void (*derivs)(double, double [], double []))
{
        void rk4(double y[], double dydx[], int n, double x, double h, double
yout[],
                void (*derivs)(double, double [], double []));
        int i,k;
        double x,h;
        double *v,*vout,*dv;

        v=vector(1,nvar);
        vout=vector(1,nvar);
        dv=vector(1,nvar);
        for (i=1;i<=nvar;i++) {
                v[i]=vstart[i];
                y[i][1]=v[i];
        }
        xx[1]=x1;
        x=x1;
        h=(x2-x1)/nstep;
        for (k=1;k<=nstep;k++) {
                (*derivs)(x,v,dv);
                rk4(v,dv,nvar,x,h,vout,derivs);
                if ((double)(x+h) == x) nrerror("Step size too small in routine
rkdumb");
                x += h;
                xx[k+1]=x;
                for (i=1;i<=nvar;i++) {
                        v[i]=vout[i];
                        y[i][k+1]=v[i];
                }
        }
        free_vector(dv,1,nvar);
        free_vector(vout,1,nvar);
        free_vector(v,1,nvar);
}
#undef NRANSI
```

MonteCarlo integration

```cpp
//Vegas.cpp - MonteCarlo integration function

#include <stdio.h>
#include <math.h>
#include "nrutilcpp.h"
#define ALPH 1.5
#define NDMX 50
#define MXDIM 10
#define TINY 1.0e-30

extern long idum;

void vegas(double regn[], int ndim, double (*fxn)(double [], double), int init,
        unsigned long ncall, int itmx, int nprn, double *tgral, double *sd,
        double *chi2a)
{
        double ran2(long *idum);
        void rebin(double rc, int nd, double r[], double xin[], double xi[]);
        static int i,it,j,k,mds,nd,ndo,ng,npg,ia[MXDIM+1],kg[MXDIM+1];
        static double calls,dv2g,dxg,f,f2,f2b,fb,rc,ti,tsi,wgt,xjac,xn,xnd,xo;
        static double d[NDMX+1][MXDIM+1],di[NDMX+1][MXDIM+1],dt[MXDIM+1],
                dx[MXDIM+1], r[NDMX+1],x[MXDIM+1],xi[MXDIM+1][NDMX+1],xin[NDMX+1];
        static double schi,si,swgt;

        if (init <= 0) {
                mds=ndo=1;
                for (j=1;j<=ndim;j++) xi[j][1]=1.0;
        }
        if (init <= 1) si=swgt=schi=0.0;
        if (init <= 2) {
                nd=NDMX;
                ng=1;
                if (mds) {
                        ng=(int)pow(ncall/2.0+0.25,1.0/ndim);
                        mds=1;
                        if ((2*ng-NDMX) >= 0) {
                                mds = -1;
                                npg=ng/NDMX+1;
                                nd=ng/npg;
                                ng=npg*nd;
                        }
                }
                for (k=1,i=1;i<=ndim;i++) k *= ng;
                npg=IMAX(ncall/k,2);
                calls=(double)npg * (double)k;
                dxg=1.0/ng;
                for (dv2g=1,i=1;i<=ndim;i++) dv2g *= dxg;
                dv2g=SQR(calls*dv2g)/npg/npg/(npg-1.0);
                xnd=nd;
                dxg *= xnd;
                xjac=1.0/calls;
                for (j=1;j<=ndim;j++) {
                        dx[j]=regn[j+ndim]-regn[j];
                        xjac *= dx[j];
                }
                if (nd != ndo) {
                        for (i=1;i<=IMAX(nd,ndo);i++) r[i]=1.0;
                        for (j=1;j<=ndim;j++) rebin(ndo/xnd,nd,r,xin,xi[j]);
                        ndo=nd;
                }
                if (nprn >= 0) {
```

```
//                    printf("%s:  ndim= %3d  ncall= %8.0f\n",
//                            " Input parameters for vegas",ndim,calls);
//                    printf("%28s  it=%5d  itmx=%5d\n"," ",it,itmx);
//                    printf("%28s  nprn=%3d  ALPH=%5.2f\n"," ",nprn,ALPH);
//                    printf("%28s  mds=%3d  nd=%4d\n"," ",mds,nd);
//                    for (j=1;j<=ndim;j++) {
//                            printf("%30s xl[%2d]= %11.4g xu[%2d]= %11.4g\n",
//                                    " ",j,regn[j],j,regn[j+ndim]);
//                    }
//            }
    }
    for (it=1;it<=itmx;it++) {
            ti=tsi=0.0;
            for (j=1;j<=ndim;j++) {
                    kg[j]=1;
                    for (i=1;i<=nd;i++) d[i][j]=di[i][j]=0.0;
            }
            for (;;) {
                    fb=f2b=0.0;
                    for (k=1;k<=npg;k++) {
                            wgt=xjac;
                            for (j=1;j<=ndim;j++) {
                                    xn=(kg[j]-ran2(&idum))*dxg+1.0;
                                    ia[j]=IMAX(IMIN((int)(xn),NDMX),1);
                                    if (ia[j] > 1) {
                                            xo=xi[j][ia[j]]-xi[j][ia[j]-1];
                                            rc=xi[j][ia[j]-1]+(xn-ia[j])*xo;
                                    } else {
                                            xo=xi[j][ia[j]];
                                            rc=(xn-ia[j])*xo;
                                    }
                                    x[j]=regn[j]+rc*dx[j];
                                    wgt *= xo*xnd;
                            }
                            f=wgt*(*fxn)(x,wgt);
                            f2=f*f;
                            fb += f;
                            f2b += f2;
                            for (j=1;j<=ndim;j++) {
                                    di[ia[j]][j] += f;
                                    if (mds >= 0) d[ia[j]][j] += f2;
                            }
                    }
                    f2b=sqrt(f2b*npg);
                    f2b=(f2b-fb)*(f2b+fb);
                    if (f2b <= 0.0) f2b=TINY;
                    ti += fb;
                    tsi += f2b;
                    if (mds < 0) {
                            for (j=1;j<=ndim;j++) d[ia[j]][j] += f2b;
                    }
                    for (k=ndim;k>=1;k--) {
                            kg[k] %= ng;
                            if (++kg[k] != 1) break;
                    }
                    if (k < 1) break;
            }
            tsi *= dv2g;
            wgt=1.0/tsi;
            si += wgt*ti;
            schi += wgt*ti*ti;
            swgt += wgt;
            *tgral=si/swgt;
```

219

```
                        *chi2a=(schi-si*(*tgral))/(it-0.9999);
                        if (*chi2a < 0.0) *chi2a = 0.0;
                        *sd=sqrt(1.0/swgt);
                        tsi=sqrt(tsi);
                        if (nprn >= 0) {
//                              printf("%s %3d : integral = %14.7g +/-  %9.2g\n",
//                                      " iteration no.",it,ti,tsi);
//                              printf("%s integral =%14.7g+/-%9.2g chi**2/IT n = %9.2g\n",
//                                      " all iterations:  ",*tgral,*sd,*chi2a);
                                if (nprn) {
                                        for (j=1;j<=ndim;j++) {
//                                              printf(" DATA FOR axis  %2d\n",j);
//                                              printf("%6s%13s%11s%13s%11s%13s\n",
//                                                      "X","delta i","X","delta i","X","delta
i");
                                                for (i=1+nprn/2;i<=nd;i += nprn+2) {
//
        printf("%8.5f%12.4g%12.5f%12.4g%12.5f%12.4g\n",
//                                                              xi[j][i],di[i][j],xi[j][i+1],
//
        di[i+1][j],xi[j][i+2],di[i+2][j]);
                                                }
                                        }
                                }
                        }
                        for (j=1;j<=ndim;j++) {
                                xo=d[1][j];
                                xn=d[2][j];
                                d[1][j]=(xo+xn)/2.0;
                                dt[j]=d[1][j];
                                for (i=2;i<nd;i++) {
                                        rc=xo+xn;
                                        xo=xn;
                                        xn=d[i+1][j];
                                        d[i][j] = (rc+xn)/3.0;
                                        dt[j] += d[i][j];
                                }
                                d[nd][j]=(xo+xn)/2.0;
                                dt[j] += d[nd][j];
                        }
                        for (j=1;j<=ndim;j++) {
                                rc=0.0;
                                for (i=1;i<=nd;i++) {
                                        if (d[i][j] < TINY) d[i][j]=TINY;
                                        r[i]=pow((1.0-d[i][j]/dt[j])/
                                                (log(dt[j])-log(d[i][j])),ALPH);
                                        rc += r[i];
                                }
                                rebin(rc/xnd,nd,r,xin,xi[j]);
                        }
                }
        }
}
#undef ALPH
#undef NDMX
#undef MXDIM
#undef TINY
```

```
//nr.h - Declares functions

#ifndef _NR_H_
#define _NR_H_

#ifndef _FCOMPLEX_DECLARE_T_
typedef struct FCOMPLEX {double r,i;} fcomplex;
#define _FCOMPLEX_DECLARE_T_
#endif /* _FCOMPLEX_DECLARE_T_ */

#ifndef _ARITHCODE_DECLARE_T_
typedef struct {
        unsigned long *ilob,*iupb,*ncumfq,jdif,nc,minint,nch,ncum,nrad;
} arithcode;
#define _ARITHCODE_DECLARE_T_
#endif /* _ARITHCODE_DECLARE_T_ */

#ifndef _HUFFCODE_DECLARE_T_
typedef struct {
        unsigned long *icod,*ncod,*left,*right,nch,nodemax;
} huffcode;
#define _HUFFCODE_DECLARE_T_
#endif /* _HUFFCODE_DECLARE_T_ */

#include <stdio.h>

#if defined(__STDC__) || defined(ANSI) || defined(NRANSI) /* ANSI */

void addint(double **uf, double **uc, double **res, int nf);
void airy(double x, double *ai, double *bi, double *aip, double *bip);
void amebsa(double **p, double y[], int ndim, double pb[],  double *yb,
        double ftol, double (*funk)(double []), int *iter, double temptr);
void amoeba(double **p, double y[], int ndim, double ftol,
        double (*funk)(double []), int *iter);
double amotry(double **p, double y[], double psum[], int ndim,
        double (*funk)(double []), int ihi, double fac);
double amotsa(double **p, double y[], double psum[], int ndim, double pb[],
        double *yb, double (*funk)(double []), int ihi, double *yhi, double fac);
void anneal(double x[], double y[], int iorder[], int ncity);
double anorm2(double **a, int n);
void arcmak(unsigned long nfreq[], unsigned long nchh, unsigned long nradd,
        arithcode *acode);
void arcode(unsigned long *ich, unsigned char **codep, unsigned long *lcode,
        unsigned long *lcd, int isign, arithcode *acode);
void arcsum(unsigned long iin[], unsigned long iout[], unsigned long ja,
        int nwk, unsigned long nrad, unsigned long nc);
void asolve(unsigned long n, double b[], double x[], int itrnsp);
void atimes(unsigned long n, double x[], double r[], int itrnsp);
void avevar(double data[], unsigned long n, double *ave, double *var);
void balanc(double **a, int n);
void banbks(double **a, unsigned long n, int m1, int m2, double **al,
        unsigned long indx[], double b[]);
void bandec(double **a, unsigned long n, int m1, int m2, double **al,
        unsigned long indx[], double *d);
void banmul(double **a, unsigned long n, int m1, int m2, double x[], double
b[]);
void bcucof(double y[], double y1[], double y2[], double y12[], double d1,
        double d2, double **c);
void bcuint(double y[], double y1[], double y2[], double y12[],
        double x11, double x1u, double x21, double x2u, double x1,
        double x2, double *ansy, double *ansy1, double *ansy2);
void beschb(double x, double *gam1, double *gam2, double *gampl,
        double *gammi);
```

```
double bessi(int n, double x);
double bessi0(double x);
double bessi1(double x);
void bessik(double x, double xnu, double *ri, double *rk, double *rip,
     double *rkp);
double bessj(int n, double x);
double bessj0(double x);
double bessj1(double x);
void bessjy(double x, double xnu, double *rj, double *ry, double *rjp,
     double *ryp);
double bessk(int n, double x);
double bessk0(double x);
double bessk1(double x);
double bessy(int n, double x);
double bessy0(double x);
double bessy1(double x);
double beta(double z, double w);
double betacf(double a, double b, double x);
double betai(double a, double b, double x);
double bico(int n, int k);
void bksub(int ne, int nb, int jf, int k1, int k2, double ***c);
double bnldev(double pp, int n, long *idum);
double brent(double ax, double bx, double cx,
     double (*f)(double), double tol, double *xmin);
void broydn(double x[], int n, int *check,
     void (*vecfunc)(int, double [], double []));
void bsstep(double y[], double dydx[], int nv, double *xx, double htry,
     double eps, double yscal[], double *hdid, double *hnext,
     void (*derivs)(double, double [], double []));
void caldat(long julian, int *mm, int *id, int *iyyy);
void chder(double a, double b, double c[], double cder[], int n);
double chebev(double a, double b, double c[], int m, double x);
void chebft(double a, double b, double c[], int n, double (*func)(double));
void chebpc(double c[], double d[], int n);
void chint(double a, double b, double c[], double cint[], int n);
double chixy(double bang);
void choldc(double **a, int n, double p[]);
void cholsl(double **a, int n, double p[], double b[], double x[]);
void chsone(double bins[], double ebins[], int nbins, int knstrn,
     double *df, double *chsq, double *prob);
void chstwo(double bins1[], double bins2[], int nbins, int knstrn,
     double *df, double *chsq, double *prob);
void cisi(double x, double *ci, double *si);
void cntab1(int **nn, int ni, int nj, double *chisq,
     double *df, double *prob, double *cramrv, double *ccc);
void cntab2(int **nn, int ni, int nj, double *h, double *hx, double *hy,
     double *hygx, double *hxgy, double *uygx, double *uxgy, double *uxy);
void convlv(double data[], unsigned long n, double respns[], unsigned long m,
     int isign, double ans[]);
void copy(double **aout, double **ain, int n);
void correl(double data1[], double data2[], unsigned long n, double ans[]);
void cosft(double y[], int n, int isign);
void cosft1(double y[], int n);
void cosft2(double y[], int n, int isign);
void covsrt(double **covar, int ma, int ia[], int mfit);
void crank(unsigned long n, double w[], double *s);
void cyclic(double a[], double b[], double c[], double alpha, double beta,
     double r[], double x[], unsigned long n);
void daub4(double a[], unsigned long n, int isign);
double dawson(double x);
double dbrent(double ax, double bx, double cx,
     double (*f)(double), double (*df)(double), double tol, double *xmin);
void ddpoly(double c[], int nc, double x, double pd[], int nd);
```

222

```
int decchk(char string[], int n, char *ch);
void derivs(double x, double y[], double dydx[]);
double df1dim(double x);
void dfour1(double data[], unsigned long nn, int isign);
void dfpmin(double p[], int n, double gtol, int *iter, double *fret,
        double (*func)(double []), void (*dfunc)(double [], double []));
double dfridr(double (*func)(double), double x, double h, double *err);
void dftcor(double w, double delta, double a, double b, double endpts[],
        double *corre, double *corim, double *corfac);
void dftint(double (*func)(double), double a, double b, double w,
        double *cosint, double *sinint);
void difeq(int k, int k1, int k2, int jsf, int is1, int isf,
        int indexv[], int ne, double **s, double **y);
void dlinmin(double p[], double xi[], int n, double *fret,
        double (*func)(double []), void (*dfunc)(double [], double[]));
double dpythag(double a, double b);
void drealft(double data[], unsigned long n, int isign);
void dsprsax(double sa[], unsigned long ija[], double x[], double b[],
        unsigned long n);
void dsprstx(double sa[], unsigned long ija[], double x[], double b[],
        unsigned long n);
void dsvbksb(double **u, double w[], double **v, int m, int n, double b[],
        double x[]);
void dsvdcmp(double **a, int m, int n, double w[], double **v);
void eclass(int nf[], int n, int lista[], int listb[], int m);
void eclazz(int nf[], int n, int (*equiv)(int, int));
double ei(double x);
void eigsrt(double d[], double **v, int n);
double elle(double phi, double ak);
double ellf(double phi, double ak);
double ellpi(double phi, double en, double ak);
void elmhes(double **a, int n);
double erfcc(double x);
double erff(double x);
double erffc(double x);
void eulsum(double *sum, double term, int jterm, double wksp[]);
double evlmem(double fdt, double d[], int m, double xms);
double expdev(long *idum);
double expint(int n, double x);
double f1(double x);
double f1dim(double x);
double f2(double y);
double f3(double z);
double factln(int n);
double factrl(int n);
void fasper(double x[], double y[], unsigned long n, double ofac, double hifac,
        double wk1[], double wk2[], unsigned long nwk, unsigned long *nout,
        unsigned long *jmax, double *prob);
void fdjac(int n, double x[], double fvec[], double **df,
        void (*vecfunc)(int, double [], double []));
void fgauss(double x, double a[], double *y, double dyda[], int na);
void fill0(double **u, int n);
void fit(double x[], double y[], int ndata, double sig[], int mwt,
        double *a, double *b, double *siga, double *sigb, double *chi2, double
*q);
void fitexy(double x[], double y[], int ndat, double sigx[], double sigy[],
        double *a, double *b, double *siga, double *sigb, double *chi2, double
*q);
void fixrts(double d[], int m);
void fleg(double x, double pl[], int nl);
void flmoon(int n, int nph, long *jd, double *frac);
double fmin(double x[]);
void four1(double data[], unsigned long nn, int isign);
```

```
void fourew(FILE *file[5], int *na, int *nb, int *nc, int *nd);
void fourfs(FILE *file[5], unsigned long nn[], int ndim, int isign);
void fourn(double data[], unsigned long nn[], int ndim, int isign);
void fpoly(double x, double p[], int np);
void fred2(int n, double a, double b, double t[], double f[], double w[],
        double (*g)(double), double (*ak)(double, double));
double fredin(double x, int n, double a, double b, double t[], double f[],
double w[],
        double (*g)(double), double (*ak)(double, double));
void frenel(double x, double *s, double *c);
void frprmn(double p[], int n, double ftol, int *iter, double *fret,
        double (*func)(double []), void (*dfunc)(double [], double []));
void ftest(double data1[], unsigned long n1, double data2[], unsigned long n2,
        double *f, double *prob);
double gamdev(int ia, long *idum);
double gammln(double xx);
double gammp(double a, double x);
double gammq(double a, double x);
double gasdev(long *idum);
void gaucof(int n, double a[], double b[], double amu0, double x[], double
w[]);
void gauher(double x[], double w[], int n);
void gaujac(double x[], double w[], int n, double alf, double bet);
void gaulag(double x[], double w[], int n, double alf);
void gauleg(double x1, double x2, double x[], double w[], int n);
void gaussj(double **a, int n, double **b, int m);
void gcf(double *gammcf, double a, double x, double *gln);
double golden(double ax, double bx, double cx, double (*f)(double), double tol,
        double *xmin);
void gser(double *gamser, double a, double x, double *gln);
void hpsel(unsigned long m, unsigned long n, double arr[], double heap[]);
void hpsort(unsigned long n, double ra[]);
void hqr(double **a, int n, double wr[], double wi[]);
void hufapp(unsigned long index[], unsigned long nprob[], unsigned long n,
        unsigned long i);
void hufdec(unsigned long *ich, unsigned char *code, unsigned long lcode,
        unsigned long *nb, huffcode *hcode);
void hufenc(unsigned long ich, unsigned char **codep, unsigned long *lcode,
        unsigned long *nb, huffcode *hcode);
void hufmak(unsigned long nfreq[], unsigned long nchin, unsigned long *ilong,
        unsigned long *nlong, huffcode *hcode);
void hunt(double xx[], unsigned long n, double x, unsigned long *jlo);
void hypdrv(double s, double yy[], double dyyds[]);
fcomplex hypgeo(fcomplex a, fcomplex b, fcomplex c, fcomplex z);
void hypser(fcomplex a, fcomplex b, fcomplex c, fcomplex z,
        fcomplex *series, fcomplex *deriv);
unsigned short icrc(unsigned short crc, unsigned char *bufptr,
        unsigned long len, short jinit, int jrev);
unsigned short icrc1(unsigned short crc, unsigned char onech);
unsigned long igray(unsigned long n, int is);
void iindexx(unsigned long n, long arr[], unsigned long indx[]);
void indexx(unsigned long n, double arr[], unsigned long indx[]);
void interp(double **uf, double **uc, int nf);
int irbit1(unsigned long *iseed);
int irbit2(unsigned long *iseed);
void jacobi(double **a, int n, double d[], double **v, int *nrot);
void jacobn(double x, double y[], double dfdx[], double **dfdy, int n);
long julday(int mm, int id, int iyyy);
void kendl1(double data1[], double data2[], unsigned long n, double *tau,
double *z,
        double *prob);
void kendl2(double **tab, int i, int j, double *tau, double *z, double *prob);
void kermom(double w[], double y, int m);
```

```
void ks2d1s(double x1[], double y1[], unsigned long n1,
        void (*quadvl)(double, double, double *, double *, double *, double *),
        double *d1, double *prob);
void ks2d2s(double x1[], double y1[], unsigned long n1, double x2[], double
y2[],
        unsigned long n2, double *d, double *prob);
void ksone(double data[], unsigned long n, double (*func)(double), double *d,
        double *prob);
void kstwo(double data1[], unsigned long n1, double data2[], unsigned long n2,
        double *d, double *prob);
void laguer(fcomplex a[], int m, fcomplex *x, int *its);
void lfit(double x[], double y[], double sig[], int ndat, double a[], int ia[],
        int ma, double **covar, double *chisq, void (*funcs)(double, double [],
int));
void linbcg(unsigned long n, double b[], double x[], int itol, double tol,
        int itmax, int *iter, double *err);
void linmin(double p[], double xi[], int n, double *fret,
        double (*func)(double []));
void lnsrch(int n, double xold[], double fold, double g[], double p[], double
x[],
        double *f, double stpmax, int *check, double (*func)(double []));
void load(double x1, double v[], double y[]);
void load1(double x1, double v1[], double y[]);
void load2(double x2, double v2[], double y[]);
void locate(double xx[], unsigned long n, double x, unsigned long *j);
void lop(double **out, double **u, int n);
void lubksb(double **a, int n, int *indx, double b[]);
void ludcmp(double **a, int n, int *indx, double *d);
void machar(int *ibeta, int *it, int *irnd, int *ngrd,
        int *machep, int *negep, int *iexp, int *minexp, int *maxexp,
        double *eps, double *epsneg, double *xmin, double *xmax);
void matadd(double **a, double **b, double **c, int n);
void matsub(double **a, double **b, double **c, int n);
void medfit(double x[], double y[], int ndata, double *a, double *b, double
*abdev);
void memcof(double data[], int n, int m, double *xms, double d[]);
int metrop(double de, double t);
void mgfas(double **u, int n, int maxcyc);
void mglin(double **u, int n, int ncycle);
double midexp(double (*funk)(double), double aa, double bb, int n);
double midinf(double (*funk)(double), double aa, double bb, int n);
double midpnt(double (*func)(double), double a, double b, int n);
double midsql(double (*funk)(double), double aa, double bb, int n);
double midsqu(double (*funk)(double), double aa, double bb, int n);
void miser(double (*func)(double []), double regn[], int ndim, unsigned long
npts,
        double dith, double *ave, double *var);
void mmid(double y[], double dydx[], int nvar, double xs, double htot,
        int nstep, double yout[], void (*derivs)(double, double[], double[]));
void mnbrak(double *ax, double *bx, double *cx, double *fa, double *fb,
        double *fc, double (*func)(double));
void mnewt(int ntrial, double x[], int n, double tolx, double tolf);
void moment(double data[], int n, double *ave, double *adev, double *sdev,
        double *var, double *skew, double *curt);
void mp2dfr(unsigned char a[], unsigned char s[], int n, int *m);
void mpadd(unsigned char w[], unsigned char u[], unsigned char v[], int n);
void mpdiv(unsigned char q[], unsigned char r[], unsigned char u[],
        unsigned char v[], int n, int m);
void mpinv(unsigned char u[], unsigned char v[], int n, int m);
void mplsh(unsigned char u[], int n);
void mpmov(unsigned char u[], unsigned char v[], int n);
void mpmul(unsigned char w[], unsigned char u[], unsigned char v[], int n,
        int m);
```

```
void mpneg(unsigned char u[], int n);
void mppi(int n);
void mprove(double **a, double **alud, int n, int indx[], double b[],
        double x[]);
void mpsad(unsigned char w[], unsigned char u[], int n, int iv);
void mpsdv(unsigned char w[], unsigned char u[], int n, int iv, int *ir);
void mpsmu(unsigned char w[], unsigned char u[], int n, int iv);
void mpsqrt(unsigned char w[], unsigned char u[], unsigned char v[], int n,
        int m);
void mpsub(int *is, unsigned char w[], unsigned char u[], unsigned char v[],
        int n);
void mrqcof(double x[], double y[], double sig[], int ndata, double a[],
        int ia[], int ma, double **alpha, double beta[], double *chisq,
        void (*funcs)(double, double [], double *, double [], int));
void mrqmin(double x[], double y[], double sig[], int ndata, double a[],
        int ia[], int ma, double **covar, double **alpha, double *chisq,
        void (*funcs)(double, double [], double *, double [], int), double
*alamda);
void newt(double x[], int n, int *check,
        void (*vecfunc)(int, double [], double []));
void odeint(double ystart[], int nvar, double x1, double x2,
        double eps, double h1, double hmin, int *nok, int *nbad,
        void (*derivs)(double, double [], double []),
        void (*rkqs)(double [], double [], int, double *, double, double,
        double [], double *, double *, void (*)(double, double [], double [])));
void orthog(int n, double anu[], double alpha[], double beta[], double a[],
        double b[]);
void pade(double cof[], int n, double *resid);
void pccheb(double d[], double c[], int n);
void pcshft(double a, double b, double d[], int n);
void pearsn(double x[], double y[], unsigned long n, double *r, double *prob,
        double *z);
void period(double x[], double y[], int n, double ofac, double hifac,
        double px[], double py[], int np, int *nout, int *jmax, double *prob);
void piksr2(int n, double arr[], double brr[]);
void piksrt(int n, double arr[]);
void pinvs(int ie1, int ie2, int je1, int jsf, int jc1, int k,
        double ***c, double **s);
double plgndr(int l, int m, double x);
double poidev(double xm, long *idum);
void polcoe(double x[], double y[], int n, double cof[]);
void polcof(double xa[], double ya[], int n, double cof[]);
void poldiv(double u[], int n, double v[], int nv, double q[], double r[]);
void polin2(double x1a[], double x2a[], double **ya, int m, int n,
        double x1, double x2, double *y, double *dy);
void polint(double xa[], double ya[], int n, double x, double *y, double *dy);
void powell(double p[], double **xi, int n, double ftol, int *iter, double
*fret,
        double (*func)(double []));
void predic(double data[], int ndata, double d[], int m, double future[], int
nfut);
double probks(double alam);
void psdes(unsigned long *lword, unsigned long *irword);
void pwt(double a[], unsigned long n, int isign);
void pwtset(int n);
double pythag(double a, double b);
void pzextr(int iest, double xest, double yest[], double yz[], double dy[],
        int nv);
double qgaus(double (*func)(double), double a, double b);
void qrdcmp(double **a, int n, double *c, double *d, int *sing);
double qromb(double (*func)(double), double a, double b);
double qromo(double (*func)(double), double a, double b,
        double (*choose)(double (*)(double), double, double, int));
```

```
void qroot(double p[], int n, double *b, double *c, double eps);
void qrsolv(double **a, int n, double c[], double d[], double b[]);
void qrupdt(double **r, double **qt, int n, double u[], double v[]);
double qsimp(double (*func)(double), double a, double b);
double qtrap(double (*func)(double), double a, double b);
double quad3d(double (*func)(double, double, double), double x1, double x2);
void quadct(double x, double y, double xx[], double yy[], unsigned long nn,
        double *fa, double *fb, double *fc, double *fd);
void quadmx(double **a, int n);
void quadvl(double x, double y, double *fa, double *fb, double *fc, double
*fd);
double ran0(long *idum);
double ran1(long *idum);
double ran2(long *idum);
double ran3(long *idum);
double ran4(long *idum);
void rank(unsigned long n, unsigned long indx[], unsigned long irank[]);
void ranpt(double pt[], double regn[], int n);
void ratint(double xa[], double ya[], int n, double x, double *y, double *dy);
void ratlsq(double (*fn)(double), double a, double b, int mm, int kk,
        double cof[], double *dev);
double ratval(double x, double cof[], int mm, int kk);
double rc(double x, double y);
double rd(double x, double y, double z);
void realft(double data[], unsigned long n, int isign);
void rebin(double rc, int nd, double r[], double xin[], double xi[]);
void red(int iz1, int iz2, int jz1, int jz2, int jm1, int jm2, int jmf,
        int ic1, int jc1, int jcf, int kc, double ***c, double **s);
void relax(double **u, double **rhs, int n);
void relax2(double **u, double **rhs, int n);
void resid(double **res, double **u, double **rhs, int n);
double revcst(double x[], double y[], int iorder[], int ncity, int n[]);
void reverse(int iorder[], int ncity, int n[]);
double rf(double x, double y, double z);
double rj(double x, double y, double z, double p);
void rk4(double y[], double dydx[], int n, double x, double h, double yout[],
        void (*derivs)(double, double [], double []));
void rkck(double y[], double dydx[], int n, double x, double h,
        double yout[], double yerr[], void (*derivs)(double, double [], double
[]));
void rkdumb(double vstart[], int nvar, double x1, double x2, int nstep,
        void (*derivs)(double, double [], double []));
void rkqs(double y[], double dydx[], int n, double *x,
        double htry, double eps, double yscal[], double *hdid, double *hnext,
        void (*derivs)(double, double [], double []));
void rlft3(double ***data, double **speq, unsigned long nn1,
        unsigned long nn2, unsigned long nn3, int isign);
double rofunc(double b);
void rotate(double **r, double **qt, int n, int i, double a, double b);
void rsolv(double **a, int n, double d[], double b[]);
void rstrct(double **uc, double **uf, int nc);
double rtbis(double (*func)(double), double x1, double x2, double xacc);
double rtflsp(double (*func)(double), double x1, double x2, double xacc);
double rtnewt(void (*funcd)(double, double *, double *), double x1, double x2,
        double xacc);
double rtsafe(void (*funcd)(double, double *, double *), double x1, double x2,
        double xacc);
double rtsec(double (*func)(double), double x1, double x2, double xacc);
void rzextr(int iest, double xest, double yest[], double yz[], double dy[], int
nv);
void savgol(double c[], int np, int nl, int nr, int ld, int m);
void score(double xf, double y[], double f[]);
void scrsho(double (*fx)(double));
```

227

```
double select(unsigned long k, unsigned long n, double arr[]);
double selip(unsigned long k, unsigned long n, double arr[]);
void shell(unsigned long n, double a[]);
void shoot(int n, double v[], double f[]);
void shootf(int n, double v[], double f[]);
void simp1(double **a, int mm, int ll[], int nll, int iabf, int *kp,
        double *bmax);
void simp2(double **a, int m, int n, int *ip, int kp);
void simp3(double **a, int i1, int k1, int ip, int kp);
void simplx(double **a, int m, int n, int m1, int m2, int m3, int *icase,
        int izrov[], int iposv[]);
void simpr(double y[], double dydx[], double dfdx[], double **dfdy,
        int n, double xs, double htot, int nstep, double yout[],
        void (*derivs)(double, double [], double []));
void sinft(double y[], int n);
void slvsm2(double **u, double **rhs);
void slvsml(double **u, double **rhs);
void sncndn(double uu, double emmc, double *sn, double *cn, double *dn);
double snrm(unsigned long n, double sx[], int itol);
void sobseq(int *n, double x[]);
void solvde(int itmax, double conv, double slowc, double scalv[],
        int indexv[], int ne, int nb, int m, double **y, double ***c, double
**s);
void sor(double **a, double **b, double **c, double **d, double **e,
        double **f, double **u, int jmax, double rjac);
void sort(unsigned long n, double arr[]);
void sort2(unsigned long n, double arr[], double brr[]);
void sort3(unsigned long n, double ra[], double rb[], double rc[]);
void spctrm(FILE *fp, double p[], int m, int k, int ovrlap);
void spear(double data1[], double data2[], unsigned long n, double *d, double
*zd,
        double *probd, double *rs, double *probrs);
void sphbes(int n, double x, double *sj, double *sy, double *sjp, double *syp);
void splie2(double x1a[], double x2a[], double **ya, int m, int n, double
**y2a);
void splin2(double x1a[], double x2a[], double **ya, double **y2a, int m, int
n,
        double x1, double x2, double *y);
void spline(double x[], double y[], int n, double yp1, double ypn, double
y2[]);
void splint(double xa[], double ya[], double y2a[], int n, double x, double
*y);
void spread(double y, double yy[], unsigned long n, double x, int m);
void sprsax(double sa[], unsigned long ija[], double x[], double b[],
        unsigned long n);
void sprsin(double **a, int n, double thresh, unsigned long nmax, double sa[],
        unsigned long ija[]);
void sprspm(double sa[], unsigned long ija[], double sb[], unsigned long ijb[],
        double sc[], unsigned long ijc[]);
void sprstm(double sa[], unsigned long ija[], double sb[], unsigned long ijb[],
        double thresh, unsigned long nmax, double sc[], unsigned long ijc[]);
void sprstp(double sa[], unsigned long ija[], double sb[], unsigned long
ijb[]);
void sprstx(double sa[], unsigned long ija[], double x[], double b[],
        unsigned long n);
void stifbs(double y[], double dydx[], int nv, double *xx,
        double htry, double eps, double yscal[], double *hdid, double *hnext,
        void (*derivs)(double, double [], double []));
void stiff(double y[], double dydx[], int n, double *x,
        double htry, double eps, double yscal[], double *hdid, double *hnext,
        void (*derivs)(double, double [], double []));
void stoerm(double y[], double d2y[], int nv, double xs,
        double htot, int nstep, double yout[],
```

```
        void (*derivs)(double, double [], double []));
void svbksb(double **u, double w[], double **v, int m, int n, double b[],
        double x[]);
void svdcmp(double **a, int m, int n, double w[], double **v);
void svdfit(double x[], double y[], double sig[], int ndata, double a[],
        int ma, double **u, double **v, double w[], double *chisq,
        void (*funcs)(double, double [], int));
void svdvar(double **v, int ma, double w[], double **cvm);
void toeplz(double r[], double x[], double y[], int n);
void tptest(double data1[], double data2[], unsigned long n, double *t, double
*prob);
void tqli(double d[], double e[], int n, double **z);
double trapzd(double (*func)(double), double a, double b, int n);
void tred2(double **a, int n, double d[], double e[]);
void tridag(double a[], double b[], double c[], double r[], double u[],
        unsigned long n);
double trncst(double x[], double y[], int iorder[], int ncity, int n[]);
void trnspt(int iorder[], int ncity, int n[]);
void ttest(double data1[], unsigned long n1, double data2[], unsigned long n2,
        double *t, double *prob);
void tutest(double data1[], unsigned long n1, double data2[], unsigned long n2,
        double *t, double *prob);
void twofft(double data1[], double data2[], double fft1[], double fft2[],
        unsigned long n);
void vander(double x[], double w[], double q[], int n);
void vegas(double regn[], int ndim, double (*fxn)(double [], double), int init,
        unsigned long ncall, int itmx, int nprn, double *tgral, double *sd,
        double *chi2a);
void voltra(int n, int m, double t0, double h, double *t, double **f,
        double (*g)(int, double), double (*ak)(int, int, double, double));
void wt1(double a[], unsigned long n, int isign,
        void (*wtstep)(double [], unsigned long, int));
void wtn(double a[], unsigned long nn[], int ndim, int isign,
        void (*wtstep)(double [], unsigned long, int));
void wwghts(double wghts[], int n, double h,
        void (*kermom)(double [], double ,int));
int zbrac(double (*func)(double), double *x1, double *x2);
void zbrak(double (*fx)(double), double x1, double x2, int n, double xb1[],
        double xb2[], int *nb);
double zbrent(double (*func)(double), double x1, double x2, double tol);
void zrhqr(double a[], int m, double rtr[], double rti[]);
double zriddr(double (*func)(double), double x1, double x2, double xacc);
void zroots(fcomplex a[], int m, fcomplex roots[], int polish);

#else /* ANSI */
/* traditional - K&R */

void addint();
void airy();
void amebsa();
void amoeba();
double amotry();
double amotsa();
void anneal();
double anorm2();
void arcmak();
void arcode();
void arcsum();
void asolve();
void atimes();
void avevar();
void balanc();
void banbks();
```

```
void bandec();
void banmul();
void bcucof();
void bcuint();
void beschb();
double bessi();
double bessi0();
double bessi1();
void bessik();
double bessj();
double bessj0();
double bessj1();
void bessjy();
double bessk();
double bessk0();
double bessk1();
double bessy();
double bessy0();
double bessy1();
double beta();
double betacf();
double betai();
double bico();
void bksub();
double bnldev();
double brent();
void broydn();
void bsstep();
void caldat();
void chder();
double chebev();
void chebft();
void chebpc();
void chint();
double chixy();
void choldc();
void cholsl();
void chsone();
void chstwo();
void cisi();
void cntab1();
void cntab2();
void convlv();
void copy();
void correl();
void cosft();
void cosft1();
void cosft2();
void covsrt();
void crank();
void cyclic();
void daub4();
double dawson();
double dbrent();
void ddpoly();
int decchk();
void derivs();
double df1dim();
void dfour1();
void dfpmin();
double dfridr();
void dftcor();
void dftint();
```

230

```
void difeq();
void dlinmin();
double dpythag();
void drealft();
void dsprsax();
void dsprstx();
void dsvbksb();
void dsvdcmp();
void eclass();
void eclazz();
double ei();
void eigsrt();
double elle();
double ellf();
double ellpi();
void elmhes();
double erfcc();
double erff();
double erffc();
void eulsum();
double evlmem();
double expdev();
double expint();
double f1();
double f1dim();
double f2();
double f3();
double factln();
double factrl();
void fasper();
void fdjac();
void fgauss();
void fill0();
void fit();
void fitexy();
void fixrts();
void fleg();
void flmoon();
double fmin();
void four1();
void fourew();
void fourfs();
void fourn();
void fpoly();
void fred2();
double fredin();
void frenel();
void frprmn();
void ftest();
double gamdev();
double gammln();
double gammp();
double gammq();
double gasdev();
void gaucof();
void gauher();
void gaujac();
void gaulag();
void gauleg();
void gaussj();
void gcf();
double golden();
void gser();
```

```
void hpsel();
void hpsort();
void hqr();
void hufapp();
void hufdec();
void hufenc();
void hufmak();
void hunt();
void hypdrv();
fcomplex hypgeo();
void hypser();
unsigned short icrc();
unsigned short icrc1();
unsigned long igray();
void iindexx();
void indexx();
void interp();
int irbit1();
int irbit2();
void jacobi();
void jacobn();
long julday();
void kendl1();
void kendl2();
void kermom();
void ks2d1s();
void ks2d2s();
void ksone();
void kstwo();
void laguer();
void lfit();
void linbcg();
void linmin();
void lnsrch();
void load();
void load1();
void load2();
void locate();
void lop();
void lubksb();
void ludcmp();
void machar();
void matadd();
void matsub();
void medfit();
void memcof();
int metrop();
void mgfas();
void mglin();
double midexp();
double midinf();
double midpnt();
double midsql();
double midsqu();
void miser();
void mmid();
void mnbrak();
void mnewt();
void moment();
void mp2dfr();
void mpadd();
void mpdiv();
void mpinv();
```

```
void mplsh();
void mpmov();
void mpmul();
void mpneg();
void mppi();
void mprove();
void mpsad();
void mpsdv();
void mpsmu();
void mpsqrt();
void mpsub();
void mrqcof();
void mrqmin();
void newt();
void odeint();
void orthog();
void pade();
void pccheb();
void pcshft();
void pearsn();
void period();
void piksr2();
void piksrt();
void pinvs();
double plgndr();
double poidev();
void polcoe();
void polcof();
void poldiv();
void polin2();
void polint();
void powell();
void predic();
double probks();
void psdes();
void pwt();
void pwtset();
double pythag();
void pzextr();
double qgaus();
void qrdcmp();
double qromb();
double qromo();
void qroot();
void qrsolv();
void qrupdt();
double qsimp();
double qtrap();
double quad3d();
void quadct();
void quadmx();
void quadvl();
double ran0();
double ran1();
double ran2();
double ran3();
double ran4();
void rank();
void ranpt();
void ratint();
void ratlsq();
double ratval();
double rc();
```

233

```
double rd();
void realft();
void rebin();
void red();
void relax();
void relax2();
void resid();
double revcst();
void reverse();
double rf();
double rj();
void rk4();
void rkck();
void rkdumb();
void rkqs();
void rlft3();
double rofunc();
void rotate();
void rsolv();
void rstrct();
double rtbis();
double rtflsp();
double rtnewt();
double rtsafe();
double rtsec();
void rzextr();
void savgol();
void score();
void scrsho();
double select();
double selip();
void shell();
void shoot();
void shootf();
void simp1();
void simp2();
void simp3();
void simplx();
void simpr();
void sinft();
void slvsm2();
void slvsm1();
void sncndn();
double snrm();
void sobseq();
void solvde();
void sor();
void sort();
void sort2();
void sort3();
void spctrm();
void spear();
void sphbes();
void splie2();
void splin2();
void spline();
void splint();
void spread();
void sprsax();
void sprsin();
void sprspm();
void sprstm();
void sprstp();
```

234

```
void sprstx();
void stifbs();
void stiff();
void stoerm();
void svbksb();
void svdcmp();
void svdfit();
void svdvar();
void toeplz();
void tptest();
void tqli();
double trapzd();
void tred2();
void tridag();
double trncst();
void trnspt();
void ttest();
void tutest();
void twofft();
void vander();
void vegas();
void voltra();
void wt1();
void wtn();
void wwghts();
int zbrac();
void zbrak();
double zbrent();
void zrhqr();
double zriddr();
void zroots();

#endif /* ANSI */

#endif /* _NR_H_ */
```

235

```
//nrutilcpp.h - Declares functions

#ifndef _NR_UTILS_H_
#define _NR_UTILS_H_

static double sqrarg;
#define SQR(a) ((sqrarg=(a)) == 0.0 ? 0.0 : sqrarg*sqrarg)

static double dsqrarg;
#define DSQR(a) ((dsqrarg=(a)) == 0.0 ? 0.0 : dsqrarg*dsqrarg)

static double dmaxarg1,dmaxarg2;
#define DMAX(a,b) (dmaxarg1=(a),dmaxarg2=(b),(dmaxarg1) > (dmaxarg2) ?\
        (dmaxarg1) : (dmaxarg2))

static double dminarg1,dminarg2;
#define DMIN(a,b) (dminarg1=(a),dminarg2=(b),(dminarg1) < (dminarg2) ?\
        (dminarg1) : (dminarg2))

static double maxarg1,maxarg2;
#define FMAX(a,b) (maxarg1=(a),maxarg2=(b),(maxarg1) > (maxarg2) ?\
        (maxarg1) : (maxarg2))

static double minarg1,minarg2;
#define FMIN(a,b) (minarg1=(a),minarg2=(b),(minarg1) < (minarg2) ?\
        (minarg1) : (minarg2))

static long lmaxarg1,lmaxarg2;
#define LMAX(a,b) (lmaxarg1=(a),lmaxarg2=(b),(lmaxarg1) > (lmaxarg2) ?\
        (lmaxarg1) : (lmaxarg2))

static long lminarg1,lminarg2;
#define LMIN(a,b) (lminarg1=(a),lminarg2=(b),(lminarg1) < (lminarg2) ?\
        (lminarg1) : (lminarg2))

static int imaxarg1,imaxarg2;
#define IMAX(a,b) (imaxarg1=(a),imaxarg2=(b),(imaxarg1) > (imaxarg2) ?\
        (imaxarg1) : (imaxarg2))

static int iminarg1,iminarg2;
#define IMIN(a,b) (iminarg1=(a),iminarg2=(b),(iminarg1) < (iminarg2) ?\
        (iminarg1) : (iminarg2))

#define SIGN(a,b) ((b) >= 0.0 ? fabs(a) : -fabs(a))

void nrerror(char error_text[]);
double *vector(long nl, long nh);
double **matrix(long nrl, long nrh, long ncl, long nch);
void free_vector(double *v, long nl, long nh);
void free_matrix(double **m, long nrl, long nrh, long ncl, long nch);

#endif
```

```
//rkdumb.h - Declares functions

#ifndef RKDUMB
#define RKDUMB

void rkdumb(double vstart[], int nvar, double x1, double x2, int nstep,
        void (*derivs)(double, double [], double []));


#endif      // RKDUMB
```

Appendix C

Fire Sciences Laboratory experimental fire data

238

Table C1. Basic fuel, environment and fire behavior data for the FiSL experimental fires.

Fire	U (m s^{-1})	δ (m)	w_a (kg m^{-2})	MC (fraction)	ROS (m s^{-1})	I_B (kW m^{-1})	D_F (m)	τ_r (s)	Pred. H_F (m)
FISL1	1.34	0.10	0.61	0.06	0.039	390	1.3	34	1.0
FISL2	1.34	0.10	0.61	0.07	0.042	421	1.4	34	1.0
FISL3	1.34	0.10	0.61	0.06	0.035	351	1.2	34	1.0
FISL4	1.34	0.10	0.61	0.06	0.041	406	1.4	34	1.0
FISL5	1.34	0.10	0.61	0.05	0.038	382	1.3	33	1.1
FISL6	2.64	0.10	0.61	0.07	0.076	763	2.8	37	1.1
FISL7	1.34	0.10	0.61	0.06	0.037	371	1.3	35	1.0
FISL8	0.00	0.10	0.61	0.07	0.007	69	0.1	11	0.4
FISL9	1.79	0.10	0.61	0.07	0.045	451	1.6	35	1.1
FISL10	0.91	0.10	0.61	0.04	0.044	442	1.9	43	1.0
FISL11	0.00	0.05	0.30	0.06	0.005	24	0.0	9	0.2
FISL12	0.89	0.05	0.30	0.06	0.077	386	0.8	10	0.6
FISL13	1.80	0.05	0.30	0.06	0.059	296	1.1	18	0.7
FISL14	2.65	0.05	0.30	0.06	0.167	837	3.7	22	0.7
FISL15	0.89	0.05	0.61	0.06	0.048	476	0.9	18	0.8
FISL16	1.79	0.05	0.61	0.05	0.050	497	1.8	35	0.8
FISL17	2.65	0.05	0.61	0.06	0.087	864	6.3	72	0.9
FISL18	0.00	0.05	0.61	0.05	0.004	38	0.1	31	0.0
FISL19	0.88	0.17	1.20	0.06	0.029	576	1.6	55	1.2
FISL20	1.79	0.17	1.20	0.05	0.033	656	1.7	52	1.4
FISL21	0.89	0.08	0.30	0.05	0.044	218	1.0	23	0.7
FISL22	1.79	0.08	0.30	0.05	0.077	382	1.4	18	0.8
FISL23	0.89	0.18	1.20	0.05	0.032	636	0.8	25	1.3
FISL24	1.79	0.17	1.20	0.05	0.033	656	3.0	91	1.4
FISL25	2.80	0.10	0.61	0.05	0.100	1004	5.0	50	1.1

Table C1 (Continued). Maximum flame temperature and upward heat flux results for the FiSL experimental fires.

Fire	T_{Fmax} (K)	Peak $q_{r,11}$ (kW m^{-2})	Peak $q_{c,11}$ (kW m^{-2})	$\Sigma_{peak}q_{r,11}$ (kW m^{-2})	$\Sigma_{peak}q_{c,11}$ (kW m^{-2})	$\Sigma q_{r,11}$ (kW m^{-2})	$\Sigma q_{c,11}$ (kW m^{-2})	Av. $\Delta q_{r,11}$ (kW m^{-2} s^{-1})	Av. $\Delta q_{c,11}$ (kW m^{-2} s^{-1})
FISL1	1159	15.2	9.3	272.4	124.3	620.3	391.5	0.40	0.22
FISL2	1228	9.7	8.3	255.0	158.7	571.3	381.5	0.17	0.15
FISL3	1096	7.9	11.1	231.2	124.2	480.8	437.7	0.15	0.33
FISL4	1176	9.2	11.5	247.2	108.4	561.4	438.7	0.18	0.32
FISL5	1135	9.2	10.2	242.7	204.4	538.4	482.1	0.16	0.18
FISL6	1109	12.8	8.7	188.3	186.5	559.0	368.3	0.37	0.12
FISL7	1209	10.8	9.0	253.2	133.5	526.5	343.3	0.21	0.19
FISL8	612	2.5	5.6	107.9	192.3	333.7	372.0	0.02	0.06
FISL9	1097	11.8	9.2	226.1	152.2	564.9	308.3	0.26	0.15
FISL10	1233	10.5	9.0	271.2	184.3	552.6	335.6	0.15	0.13
FISL11	869	0.6	2.2	4.9	76.4	2.9	119.6	0.00	0.03
FISL12	743	6.2	2.9	131.9	32.3	266.8	128.3	0.15	0.07
FISL13	1006	6.4	4.4	123.5	68.2	243.2	174.6	0.18	0.10
FISL14	922	7.9	5.6	96.0	78.7	225.4	208.8	0.34	0.15
FISL15	957	5.4	4.5	185.5	74.7	499.6	314.1	0.08	0.12
FISL16	1057	7.4	6.3	154.1	160.9	488.1	349.5	0.12	0.06
FISL17	1222	9.5	7.6	175.2	149.9	554.2	361.6	0.23	0.16
FISL18	832	1.5	5.5	43.2	168.7	183.1	415.1	0.02	0.05
FISL19	1335	20.3	34.9	331.1	460.4	1163.4	988.4	0.27	0.40
FISL20	1275	24.2	26.5	317.7	430.0	1350.3	871.0	0.45	0.23
FISL21	913	4.3	4.0	92.6	37.9	248.6	146.4	0.09	0.17
FISL22	1026	5.8	5.0	100.7	49.9	248.3	160.1	0.19	0.10
FISL23	1179	17.1	10.4	373.9	112.6	1254.8	526.6	0.29	0.17
FISL24	1249	17.7	19.6	147.3	229.6	1291.0	643.7	0.46	0.28
FISL25	1294	11.9	13.5	154.6	161.3	649.5	342.2	0.42	0.26