Pergamon

Computers Math. Applic. Vol. 31, No. 8, pp. 37–47, 1996
Copyright©1996 Elsevier Science Ltd
Printed in Great Britain. All rights reserved
0898-1221/96 $15.00 + 0.00

S0898-1221(96)00029-6

The Simulated Cooling of the Hot-Rolled Structural Steel Sections

A. D. RICHARDSON
British Steel Technical, Teesside Technology Centre
Middlesbrough, U.K.

J. R. DORMAND
School of Computing & Mathematics, University of Teesside, Teesside, U.K.

(Received and accepted September 1995)

Abstract—Temperature models based on the finite difference, ADI and Runge-Kutta methods have been written in order to establish the most efficient algorithm when simulating the cooling of newly hot-rolled steel sections under a variety of cooling conditions. For air-cooling, the most efficient results were obtained using extended-stability Runge-Kutta methods, together with adaptive step-size control procedures. CPU time-savings of around 85% were achieved when an existing finite difference based section air-cooling model was modified to run using a specially developed, highly stable, second-order Runge-Kutta formula with the method of lines. The ADI approach gave the most efficient results for water spray cooling, producing accurate results in approximately half the CPU time required by the finite difference method.

Keywords—Parabolic partial differential equations, Stiff systems of equations, Runge-Kutta methods, Extended regions of absolute stability, Semidiscretisation, Thermal modelling of steel.

1. INTRODUCTION

When attempting to optimise the cooling conditions of newly hot-rolled steel sections, a knowledge of how the proposed cooling strategy will affect the evolution of the temperature distribution within the section is essential in order to predict residual thermal stress development and any resultant undesirable buckling of the steel. Initially at around 900 to 1000°C, it normally takes several hours for the steel to reach room temperature. Over much of this time the section will be air cooled, losing heat by radiation and convection. However for short periods the surface of the beam may be sprayed with water, resulting in much more rapid temperature loss. In order to simulate thermal stress development and buckling, existing process models use a fine finite difference mesh for the temperature calculations. As a result, stability requirements severely limit the time step size and simulation of the process of cooling to room temperature may take several hours of computer time. The purpose of this paper is to describe a more efficient alternative to the conventional explicit finite difference (FD) algorithm which will give large savings in computing time.

When the step-size of an explicit process is limited by stability requirements, unconditionally stable implicit techniques must be considered. However, the nonlinearities caused by temperature dependent thermal properties and boundary conditions make each step rather more expensive than in the conventional explicit process. The Alternating-Direction-Implicit (ADI) method, giving tridiagonal systems of equations, is comparatively efficient but is only easily applicable to domains with simple rectilinear geometry. For general application the attraction of explicit techniques is clear.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

In recent years, much research has been devoted to the development of Runge-Kutta (RK) formulae with accuracy and stability characteristics chosen to suit particular types of problem. Although RK methods are specifically designed for ordinary rather than partial differential equations, they may be applied to the heat conduction equation after discretisation which is applied to the space variables only, leaving time as the single independent variable. This process is known as *semidiscretisation* and the overall procedure is often described as the *Method of Lines*.

We demonstrate below that an extended-stability RK routine proves much more efficient in simulating the cooling process than the conventional explicit finite difference scheme, under conditions where accuracy considerations do not severely limit time-step size.

2. THE HEAT CONDUCTION PROBLEM

We aim to establish the most efficient temperature calculation routines for the simulation of steel sections cooling under a variety of conditions, the most common being air and spray cooling. As the geometry of steel sections is relatively complex, initial modelling work was performed on a simple 2-dimensional mesh, representing a steel block of rectangular cross-section. Once the ideal temperature calculation routines had been established for each cooling condition, the modelling work could then be extended to cover steel sections.

The heat conduction equation in two space dimensions (x, y) may be written

$$\frac{\partial u}{\partial t} = \alpha(u)\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right), \qquad u(x, y, 0) = \Phi(x, y), \tag{1}$$

where u is the temperature, t is the time, and α is the diffusivity. This is a parabolic initial value problem. The boundary conditions depend on the type of cooling under consideration, and these are of the form

$$q = -K(u)\frac{\partial u}{\partial n}, \tag{2}$$

where K is the thermal conductivity. For air cooling, the heat flux is

$$q = V\sigma\varepsilon\left[U_b^4 - U_o^4\right] + h_a\left[U_b - U_o\right], \tag{3}$$

where σ is Stefan's constant, ε is the emissivity, h_a is the convective heat transfer coefficient, and the suffices b and o indicate boundary and ambient temperatures. During air cooling there will be an exchange of radiated heat energy between any adjacent steel surfaces, such that the net radiative heat flux from a given point on a surface will be reduced to a fraction V (radiation view factor) of its original value. For water-spray cooling

$$q = h_s\left[U_b - U_o\right], \tag{4}$$

where h_s is the spray heat transfer coefficient. The finite difference discretisation of (1) and (2) is well known and can be found in any standard text [1]. A more rigorous approach would include the gradient of the diffusivity but in practice this may be neglected in favour of tabulation. The thermal conductivity may be treated in a similar fashion and so the only nonlinear temperature dependence is associated with the air cooling boundary condition (3).

3. RUNGE-KUTTA METHODS

3.1. Embedded RK Methods

RK methods provide a convenient method of solving initial value problems of the type

$$y' = f(x, y(x)),\ y(x_0) \text{ given}, \qquad y \in \mathbb{R}^m. \tag{5}$$

Embedded methods use two RK formulae of orders p and q (RKp and RKq), where $q = p + 1$, which share the same function evaluations, to calculate two estimates of $y(x_{n+1}) = y(x_n + h_n)$ at each step. Using the symbol $\hat{y}$ to denote the higher order solution, and adopting the RK notation given in [2], the two estimates for $y(x_{n+1})$ can be written

$$\hat{y}_{n+1} = \hat{y}_n + h_n \sum_{i=1}^{s} \hat{b}_i f_i, \quad y_{n+1} = \hat{y}_n + h_n \sum_{i=1}^{s} b_i f_i, \tag{6}$$

where

$$f_i = f\left(x_n + c_i h_n, \hat{y}_n + h_n \sum_{j=1}^{i-1} a_{ij} f_j\right).$$

Both RK formulae have s stages, so s function evaluations are performed at each step. The local error e_{n+1} is given approximately by

$$e_{n+1} = y_{n+1} - \hat{y}_{n+1}, \tag{7}$$

so the embedded method gives an estimate of the local error without the need for extra function evaluations, and this estimate can be used to ensure that the size of each step is small enough to avoid local errors larger than a tolerance value T. At each step, the local error is estimated, and the step-size modified according to the optimal reduction formula [2]:

$$h_{n+1} = h_n \left(\frac{\alpha}{r_{n+1}}\right)^{1/q}, \quad r_{n+1} = \frac{\|e_{n+1}\|_\infty}{T}, \tag{8}$$

where $r_{n+1} < 1$ for an acceptable step and α is a strategy parameter, a typical value being 0.5. The ability to estimate local error and modify the step-size at each step is an advantage not shared by FD methods.

The equations of condition that must be satisfied by the RK coefficients a_{ij}, b_i, and c_i, for RK formulae of order up to 6, are given in [2].

3.2. Stability of RK Formulae

Applying RK techniques to the scalar problem $y' = \lambda y$, we obtain the difference equation

$$y_{n+1} = P(r) y_n, \quad r = h\lambda. \tag{9}$$

For absolute stability (decreasing solution), we require $|P(r)| < 1$, a condition which can be satisfied by choosing a sufficiently small h. Given the system of equations $y' = Ay$, where the matrix A is $m \times m$, with eigenvalues $\lambda_i, i = 1, 2, \ldots, m$, the corresponding stability requirement is

$$|P(h\lambda_i)| < 1, \qquad i = 1, 2, \ldots, m.$$

Since λ may be complex, it is possible to define a region in the complex plane (for $h\lambda_i$) in which stability is satisfied. If all the eigenvalues are real, then the step-size is limited only by the extent of the region of absolute stability including the real axis. For an RKp formula of s stages, it can be shown [3] that the stability polynomial $P(r)$ is of degree s, given by

$$\begin{aligned} P(r) &= 1 + r \sum_i b_i + r^2 \sum_i b_i c_i + r^3 \sum_{ij} b_i a_{ij} c_j + r^4 \sum_{ijk} b_i a_{ij} a_{jk} c_k \\ &\quad + \cdots + r^s \sum_{ijk\ldots vw} b_i a_{ij} a_{jk} \ldots a_{vw} c_w \\ &= 1 + \sum_{i=1}^{s} W_i r^i, \end{aligned} \tag{10}$$

where $W_i = 1/i!, i = 1, 2, \ldots, p$.

3.3. The Method of Lines with RK Formulae

Consider the application of the heat conduction equation (1) to the rectangular domain

$$0 \leq x \leq N\Delta x, \qquad 0 \leq y \leq M\Delta y.$$

The explicit FD formula for the solution of (1), obtained by discretising both spatial and time variables, is

$$U_{i,j}^{n+1} = U_{i,j}^{n} + \frac{\alpha_{ij}\Delta t}{(\Delta x)^2}\left[U_{i+1,j}^{n} - 2U_{i,j}^{n} + U_{i-1,j}^{n}\right] + \frac{\alpha_{ij}\Delta t}{(\Delta y)^2}\left[U_{i,j+1}^{n} - 2U_{i,j}^{n} + U_{i,j-1}^{n}\right]. \tag{11}$$

The method of lines adopts the alternative approach of semi-discretising (1), leaving t as a continuous variable. Setting $u(x_i, y_j, t) = U_{ij}(t)$, $i = 0, 1, \ldots, N$; $j = 0, 1, \ldots, M$, and discretising the space derivatives in (1), we obtain the system of ordinary differential equations

$$\frac{dU_{ij}(t)}{dt} = \alpha_{ij}\left[\frac{\delta_x^2}{(\Delta x)^2} + \frac{\delta_y^2}{(\Delta y)^2}\right] U_{ij}(t), \tag{12}$$

which has the same form as (5) and thus may be solved conveniently with RK methods.

For a linear cooling problem, the matrix associated with the system of ODEs (12) has negative real eigenvalues [3], so for an RK formula with absolute stability in $(-\beta, 0)$, the time step will satisfy

$$\Delta t = h \leq \frac{\beta}{|\lambda_{\max}|}. \tag{13}$$

In the problem of interest, the boundary conditions (3) will cause nonlinearity. However it is normally assumed that this factor will not perturb significantly the matrix eigenvalues.

3.4. Comparison of RK and FD Formulae

Application of the explicit finite difference formula to (1) is equivalent to the use of Euler's method (a first order RK with $s = 1$) to solve the ODE system (12). For Euler's method, $\beta = 2$, and the resulting step-size limit, for the FD method applied to the linear problem, is

$$h_{\mathrm{FD}} = \frac{1}{2\alpha\left(1/(\Delta x)^2 + 1/(\Delta y)^2\right)}.$$

The corresponding limit for an RK method with real negative stability interval $(-\beta, 0)$ is

$$h_{\mathrm{RK}} = \frac{\beta}{4\alpha\left(1/(\Delta x)^2 + 1/(\Delta y)^2\right)}. \tag{14}$$

Since multistage RK formulae allow $\beta \gg 2$ as well as higher orders of accuracy ($p > 1$), it is natural to compare their performance with that of the simpler method. Extra stages mean extra function evaluations and so, for higher efficiency, the step-size increase factor for our new RK method must be greater than the cost per step increase factor. Assuming that the computational cost per RK step is proportional to the number of function evaluations s, and that stability rather than truncation error limits the step-size, we define an efficiency factor to be

$$E = \frac{\beta}{s}. \tag{15}$$

For the explicit finite difference method we have $E = 2$.

3.5. Choice of RK Method

Two main types of section cooling are under consideration, air cooling and spray cooling. Experience with FD calculations suggests that numerical stability may be limiting time step size for air cooling. With spray cooling, which is much faster, this may not be the case.

We aim to find an embedded RK formula pair with the highest possible β/s ratio, where β derives from the lower of the two stability limits.

The development of explicit RK formulae suitable for the efficient integration of parabolic PDEs has been described by Van der Houwen [4], who presented three families of low order methods (RK2, RK3, and RK4). Each family was generated from a set of stability polynomials (10) chosen to satisfy the appropriate order conditions and to have the maximum possible negative real absolute stability limit. These polynomials are closely related to Chebyshev polynomials but the order constraint is imposed. For the first-order case, an appropriately shifted Chebyshev polynomial suffices. The stability limits of these RK formulae are given in Table 1, and these values increase with the the number of stages and hence with the degree of the stability polynomial. Plots of the polynomials show them to be similar to minimax polynomials.

Table 1. Stability characteristics of Van der Houwen's formulae.

Stages (s)	Stability Limit β		
	RK2	RK3	RK4
3	6.26		
4	12.05	6.03	
5	19.46	10.54	6.06
6	28.50	16.05	9.97
7	39.19	22.56	14.59
8	51.52	30.07	19.93
9	65.49	38.60	25.98
10	81.11	48.11	32.74
11	98.37	58.64	40.22
12	117.27	70.17	48.11

Families of embedded RK formulae (RK5(4) and RK6(5)), which include members with enlarged regions of absolute stability and small truncation error norms were presented by Dormand and Prince [2,5]. However the efficiency factors (15) of these formulae are smaller than that of the Euler method and considerably less than those of the Van der Houwen families. Consequently they may be useful only where step-sizes are controlled by truncation error.

A set of low-order embedded RK formulae with extended but not maximised stability has been published by Fehlberg [6]. However these formulae do not use local extrapolation.

3.6. Stable Embedded RK Pairs

From Table 1, the RK2 formulae of Van der Houwen have the highest efficiency factors. To permit error estimation and step-size control it is convenient to embed a first-order formula in one of these RK2 schemes. The resulting RK2(1)sS formula pair can be displayed in the tabular form:

$$\begin{array}{c|ccccc||c|c} 0 & & & & & & 0 & b_1 \\ c_2 & a_{21} & & & & & 0 & b_2 \\ c_3 & 0 & a_{32} & & & & 0 & b_3 \\ \vdots & \vdots & \vdots & \ddots & & & \vdots & \vdots \\ \frac{1}{2} & 0 & 0 & \cdots & \frac{1}{2} & & 1 & b_s \end{array} \tag{16}$$

where $c_i = \sum_{j=1}^{i-1} a_{ij}$, and the coefficients $b_1, \ldots, b_s$ refer to the lower-order (first-order) formula, and must satisfy $\sum_{i=1}^{s} b_i = 1$.

It appears from Table 1 that the efficiency factor E increases almost linearly with the number of stages, and we have constructed RK2 formulae with up to 15 stages. With $s = 15$, we obtain $\beta = 183.9$, giving $E = 12.3$. Larger s proves to be very difficult because of nonconvergence of the Remez iteration [7] which was used to determine the RK2 stability polynomial. Fortunately numerical tests will indicate that larger s may not be more efficient in practice. For the first-order member of the embedded pair, the real stability limit should be at least as large as that of the RK2, and so the shifted Chebyshev polynomial

$$P(r) = T_{10}\left(1 + \frac{r}{100}\right), \qquad r = h\lambda,$$

is a convenient choice for the stability polynomial. This is stable for $r \in [-200, 0]$. The appropriate b_i are obtained from equation (10), and the second-order coefficients are given in Table 2.

Table 2. Coefficients for the 15-stage RK2.

i	c_i
2	$7.31536209789414863265785810147042 80 \times 10^{-4}$
3	$1.62768249078385033540949811937603 05 \times 10^{-3}$
4	$2.74251865675580909457144467001700 02 \times 10^{-3}$
5	$4.15449172306511342902901685342646 92 \times 10^{-3}$
6	$5.98096762556488515371742261085801 21 \times 10^{-3}$
7	$8.40408417769247614355647784591643 67 \times 10^{-3}$
8	$1.17192177333562886219462352142706 81 \times 10^{-2}$
9	$1.64319549447918933642063489753714 01 \times 10^{-2}$
10	$2.34683772312831896269684100859603 82 \times 10^{-2}$
11	$3.46800683813598669340913423363907 25 \times 10^{-2}$
12	$5.42298864699797153477280534000302 13 \times 10^{-2}$
13	$9.31428969067228577588794390190929 38 \times 10^{-2}$
14	$1.88349717158753304283643804451992 82 \times 10^{-1}$
15	$5.00000000000000000000000000000000 00 \times 10^{-1}$

4. LINEAR TEST PROBLEM

To test the efficiency of the embedded RK2(1) pairs, they have been applied to the one-dimensional equation

$$u_t = u_{xx},$$

with boundary and initial conditions

$$u(0,t) = u(1,t) = 0; \; u(x,0) = \sin \pi x, \qquad x \in [0,1].$$

The true solution is

$$u(x,t) = \exp\left(-\pi^2 t\right) \sin \pi x. \tag{17}$$

Semidiscretization yields the linear system

$$\frac{du_i}{dt} = \frac{u_{i-1} - 2u_i + u_{i+1}}{(\Delta x)^2}, \qquad i = 1, 2, \ldots, m-1, \quad t \in [0,5]. \tag{18}$$

With initial and boundary conditions satisfying the PDE, this system of equations has solution

$$u_i(t) = \exp\left\{-4m^2 \sin^2\left(\frac{\pi}{2m}\right) t\right\} \sin\left(\frac{i\pi}{m}\right),$$

and as $\Delta x = 1/m \to 0$, this will converge to the solution (17).

Figure 1 shows the efficiency curves for four second order formulae applied to equation (18) with $m = 21$. Three of the formulae are the 6, 9, and 12-stage processes defined by Van der Houwen, and the fourth is the 15-stage method given in Table 2. Step-size control was achieved using error estimates based on embedding. The solution has a fast transient phase in which increasing step-sizes are predictable according to the asymptotically valid formula (8). The effective duration of this phase is dependent on the stability properties of the integrator, being greatest ($t \sim [0, 0.5]$) for the 15-stage formula. After the transient phase is complete, the steplength is governed by stability and so the formula (8) is no longer appropriate, although the embedding still furnishes the error estimate. A modification of the Watts algorithm [8],

$$h_{n+1} = \psi_n h_n, \qquad \psi_n = \left(\frac{h_n}{h_{n-1}}\right)\left[\left(\frac{\alpha}{r_{n+1}}\right)\left(\frac{r_n}{r_{n+1}}\right)\right]^{1/q}, \tag{19}$$

where $\psi_{\min} < \psi < \psi_{\max}$, in which the step-size is prevented from rapid variation, proved to be an efficient means of controlling the steplength during the stiff phase. At most tolerances, the solution included a single rejected step near the start of the stiff phase.

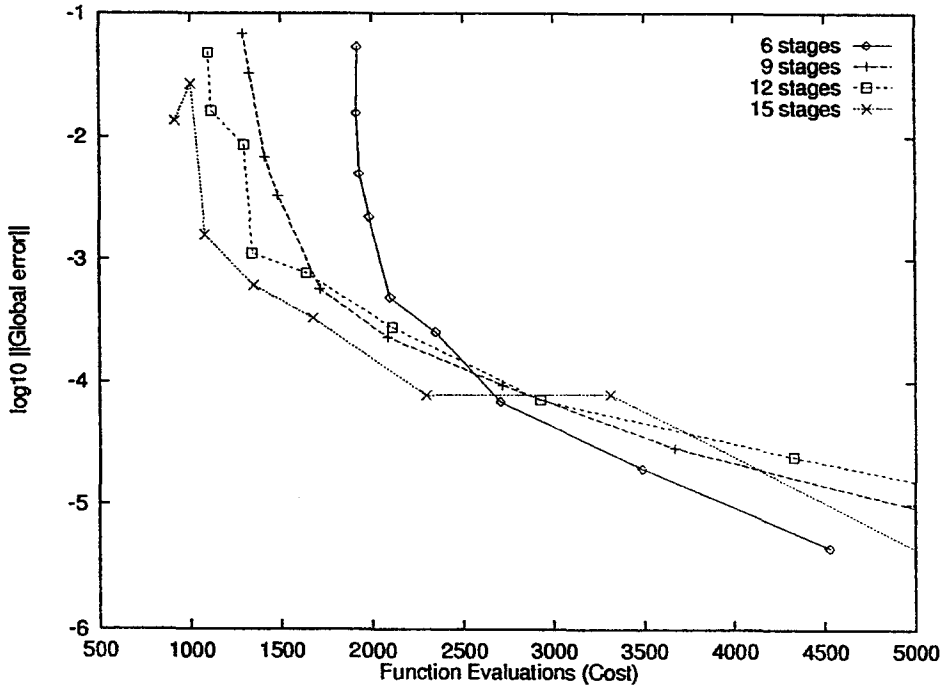

Figure 1. Efficiency curves for RK2 formulae with extended stability.

The 15-stage formula proves is the most efficient at lax tolerances, yielding global error $\|\varepsilon\| > 10^{-4}$, where the norm is taken over all steps and components of the solution. For more stringent tolerances, the stability property is less important and the 6-stage formula is the best of the four shown here. It seems unlikely that an increase in the number of stages would produce a significantly improved performance for the accuracy required in this work.

It should be noted that the more conventional RK5(4)7FS formula requires more than 10000 function evaluations to produce a global error 10^{-3} for problem (18).

5. APPLICATION TO COOLING OF STEEL

5.1. Preliminary

The semidiscretisation method with special RK processing has been applied to the cooling of hot-rolled steel sections. The cooling process is modelled in two space dimensions (equation (1))

with various boundary conditions (Section 2 and Figure 2). For mild steel, the diffusivity (α) is temperature dependent, and so the numerical stability limit will not be constant throughout the cooling period. As the steel cools from 900°C to 20°C the diffusivity increases by a factor of about 3. This implies a decreasing step length for the time integration. The step size control procedure developed in the test problem serves to vary the steplength appropriately without too many rejected steps. The extended stability property does not permit an embedded pair satisfying the step size equilibrium condition of Hall [9].

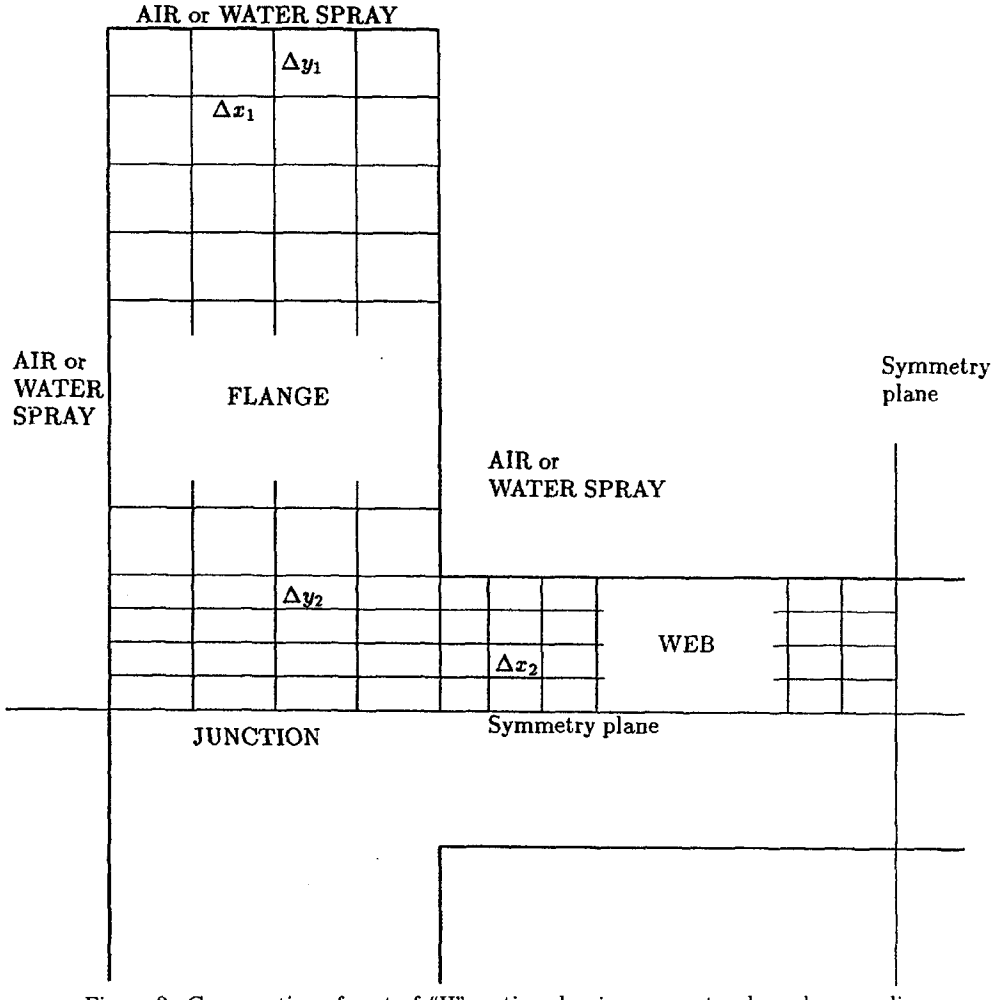

Figure 2. Cross-section of part of "H"-section showing symmetry, boundary conditions, and mesh geometry.

The accuracies of the special RK and FD processes were checked for a rectangular domain using a high-order RK integrator [10] with a stringent tolerance.

5.2. Air-Cooling Results

The 15-stage embedded RK2 formula (RK2(1)15s) has been used to simulate the air-cooling of a steel "H"-section (Figure 2), from a uniform initial temperature of 1000°C, over a time period of 15,000 seconds. Final temperatures were in the range 192°C to 202°C (see Figure 3). Results were compared with those obtained using the conventional FD method. A comparatively lax local error tolerance of 1°C was used, appropriate to the industrial application. As in the one-dimensional test problem, the solution had an initial fast transient phase, during which increasing steplength could be successfully predicted using the optimal reduction formula (8). This corresponds to the period of very rapid heat loss from the steel boundary at the start of

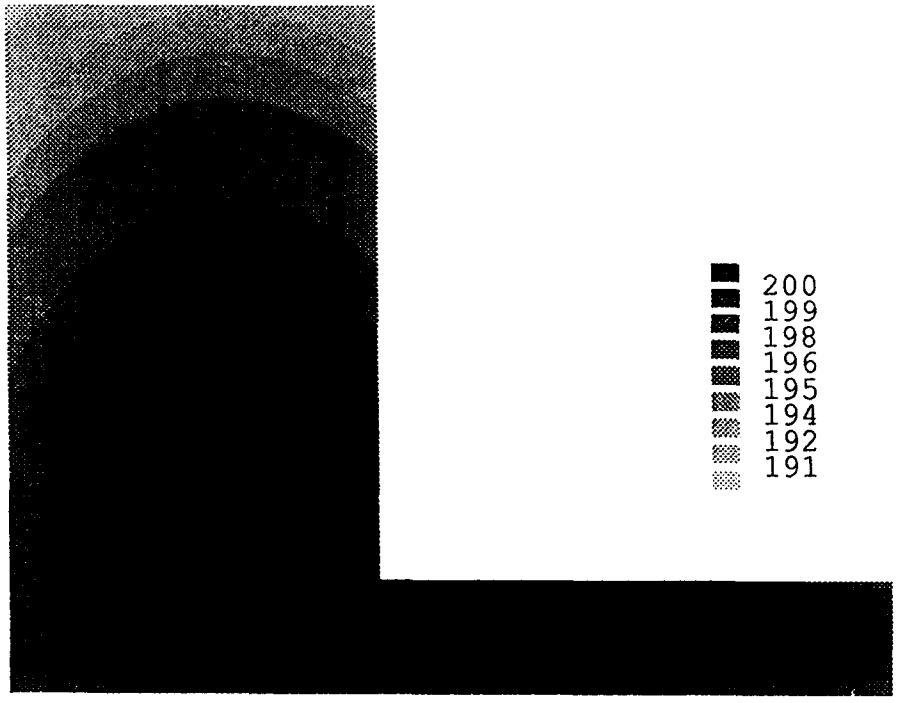

Figure 3. Temperature profile after 15000 s of air cooling.

cooling. The solution subsequently entered a stiff phase for the remainder of the cooling period. As expected, the asymptotically valid step-size control formula (8) was found to be inappropriate during the stiff phase, and the modified Watts algorithm (19) developed for the test problem was required to prevent costly step-size oscillations about the numerical stability limit. The onset of stiffness was detectable by the occurrence of a rejected step with a local error estimate many times greater than the tolerance.

The step-size sequence achieved using the RK2(1)15s formula with the Watts algorithm is shown in Figure 4. In the transient phase rapid step-size variation was allowed, step length increases and decreases being limited to factors 2.0 and 0.1, respectively. During the stiff phase, far more stringent step change limit factors of 1.02 and 0.75 were imposed, preventing significant numbers of step rejections and maintaining the step length close to the (declining) numerical stability limit. As the stiff phase dominates the overall cooling period, the air-cooling simulation proved to be highly efficient. CPU-time usage was less than 1/7 of that required for the conventional FD approach, and throughout the simulation, mesh point temperatures differed by at most 2°C in the two solutions. This discrepancy is negligible in the industrial context.

Air-cooling simulations were also performed using the alternating-direction-implicit (ADI) method, which like the RK2(1)15s formula is second order, but is unconditionally stable. Numerical experiments revealed that good accuracy could be achieved using step-sizes of similar length to those used with the RK2(1)15s formula. With a lower computational cost per step, considerable additional CPU-time savings can be achieved. However, as the ADI method is only easily applicable to domains with simple rectilinear geometry, it could only be readily applied to a restricted range of BS products.

5.3. Spray-Cooling Results

The extended stability RK formulae developed were used to simulate the cooling of steel sections with water sprays (Figure 2). With heat transfer coefficients for many industrial spray-cooling systems in the range 10,000 to 30,000 W m^{-2} k^{-1} the rate of heat loss involved is considerably higher than for air-cooling, and is typically over 100 times greater immediately after hot-rolling. Consequently, ambient temperature is reached in minutes rather than hours.

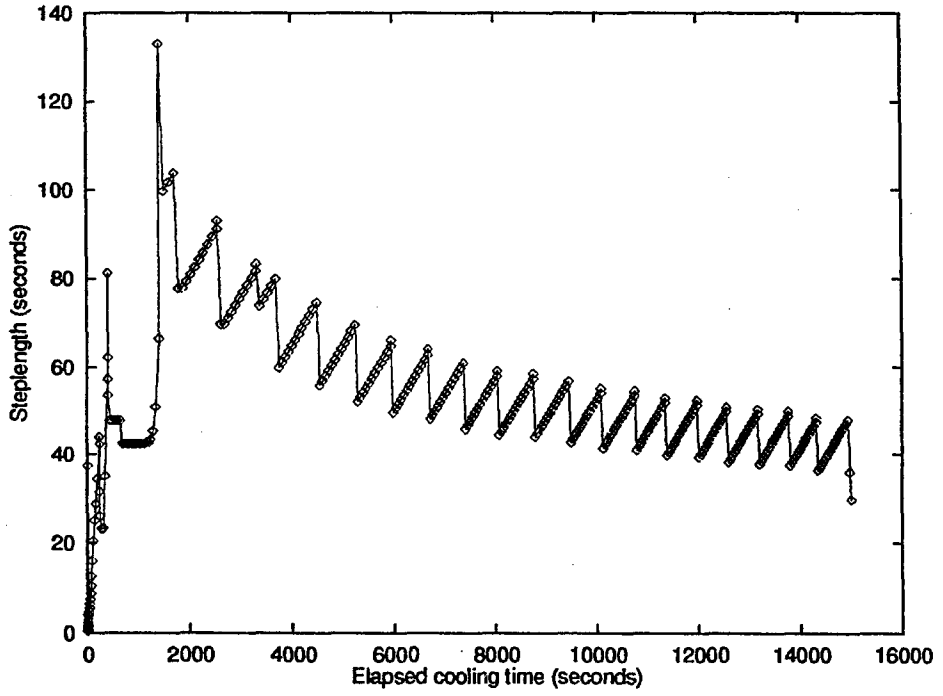

Figure 4. Step-size sequence with air cooling problem.

As expected, the initial transient solution phase was found to dominate the cooling period. Even with lax local error tolerances small step-sizes were required to accurately model the rapid cooling. The highly stable RK2(1) formulae were found to be less efficient than the FD method, on account of the high computational cost per step. Improved results were obtained by using higher order stable RK4(3) formulae, but the efficiency achieved was highly sensitive to the heat transfer characteristics of the spray system used, and also to the extent of spray application around the steel boundary. For particularly rapid rates of heat extraction, the FD method with its low computational cost per step required less CPU-time usage than any of the stiff RK formulae developed.

Spray-cooling simulations were also performed using higher order methods including the alternating direction implicit (ADI) method. For rapid cooling due to heavy spraying, the higher order methods permit larger step-sizes and tend to be superior to the FD method. Steps are computationally more expensive than with the FD approach but stability considerations are less important as cooling becomes more rapid. Numerical experiments revealed that for most spray-cooling applications, the ADI method gave reasonable results (accurate to within 5°C) using time steps considerably longer than is possible with the FD approach. CPU-time savings of up to 50% were achieved by adopting the ADI formula.

6. CONCLUSIONS

A study of different methods for simulating the cooling of newly hot-rolled steel sections has revealed that more efficient alternatives to the traditional FD approach exist for the cooling systems currently in use. For air cooling conditions, under which the solution exhibits stiff behaviour over most of the cooling period, the use of extended stability embedded RK2(1) formulae has allowed cooling simulations to be performed using only a small fraction of the CPU-time required by the FD approach. The application of a specially developed highly stable RK2(1)15s formula, together with a modified version of the Watts step-size control procedure in which suitably chosen parameters prevent costly step length oscillations about the numerical stability limit, has led to CPU-time savings in excess of 85%.

Under spray cooling conditions the step-size was constrained by accuracy rather than numerical stability for most of the cooling period. Consequently the highly stable RK formulae developed did not perform efficiently, on account of the high computational cost per step. CPU-time savings of up to 50% were achieved with the second-order unconditionally stable ADI method, but this method can only be readily applied to sections with very simple rectilinear geometries. For sections with more irregular boundaries, an explicit approach (perhaps FD) remains the most suitable option.

The adoption of the RK2(1)15s formula and/or the ADI method within the process models for section cooling has greatly reduced the time and expense involved in running industrial simulations. This should be of considerable benefit to current research programmes investigating optimum cooling strategies for a wide range of steel products.

REFERENCES

1. G.D. Smith, *Numerical Solution of Partial Differential Equations*, 3rd edition, Oxford University Press, (1985).
2. J.R. Dormand and P.J. Prince, A family of embedded Runge-Kutta formulae, *J. Comput. Appl. Math.* **6**, 19–26 (1980).
3. J.D. Lambert, *Numerical Methods for Ordinary Differential Systems*, John Wiley & Sons, (1991).
4. P.J. van der Houwen, Explicit Runge-Kutta formulas with increased stability boundaries, *Numer. Math.* **20**, 149–164 (1972).
5. J.R. Dormand and P.J. Prince, A reconsideration of some embedded Runge-Kutta formulae, *J. Comput. Appl. Math.* **15**, 203–211 (1986).
6. E. Fehlberg, Low-order classical Runge-Kutta formulas with stepsize control and their application to some heat transfer problems, NASA Technical Report No. TR-315, (1969).
7. C.T. Fike, *Computer Evaluation of Mathematical Functions*, Prentice Hall, (1968).
8. H.A. Watts, Step size control in ODE solvers, *Trans. of Soc. for Computer Simulation* 1, 15–25 (1984).
9. G. Hall, Equilibrium states of Runge Kutta schemes, *ACM Trans. on Math. Soft.* **11** (3), 289–301 (1985).
10. J.R. Dormand and P.J. Prince, Higher order embedded Runge-Kutta formulae, *J. Comput. Appl. Math.* **7**, 67–75 (1981).