Parallel CFD'93 - Implementation and Results Using Parallel Computers, Elsevier Science Publisher B.V., 1993.

Parallel Computation of a Combustion Problem

Frederic Desprez*
LIP CNRS-URA 1398, ENS de Lyon
69364 Lyon Cedex 07, France
desprez@lip.ens-lyon.fr

Marc Garbey†
LAN, Université Claude Bernard Lyon 1
69622 Villeurbanne cedex, France
mgarbey@lan1.univ-lyon1.fr

February 23, 1994

Abstract

We present the parallel computation of a combustion model with a two steps sequential reaction mechanism. Our main interest is the study of pattern formation, bifurcations and nonlinear interactions between the two chemical reactions; The combustion occurs in a frontal regime that leads to intense computation. We show how to match to the frontal structure of the problem an adaptive domain decomposition of the PDE, and we construct a massively parallel algorithm. The algorithm has been implemented with high efficiency on the hypercube iPSC/860 and the Paragon by overlapping communications and computations.

1 Introduction

The goal of this paper is to show that a properly designed domain decomposition offers an efficient algorithm to solve a reaction diffusion model of combustion on a massively parallel architecture.

The combustion model under investigation is a typical example of a highly nonlinear multiple scale problem. In multiple scale problems one can clearly take advantage of the special structure of the solution in the design of a domain decomposition method as shown in [8] Also multiple scale problems lead to very intense computation because one has to obtain an accurate representation of the solution in the so called transition layers where the solution is stiff; in addition the unstabilities of a combustion front are formed in the transition layers. Also it is of special importance to solve accurately this zone of stiffness. We use an adaptive scheme because the ratio of the space scale in the layers divided by the space scale in the region where the solution is not stiff is a very small number.

We have chosen to use an adaptive pseudospectral method that have been proved to be very accurate to solve such multiple scaled problems [1, 2, 3, 4, 7, 11] and its references. One of the essential feature of the method is that one can keep a regular data structure by mean of mapping and domain decomposition for an adaptive method that solves a two dimensional *stiff* problem with a complex front structure. It is therefore more easy to implement such algorithm on a massively parallel architecture.

However we face two difficulties in the implementation of our parallel algorithm: first we use a high order numerical scheme to discretize the equation in space and unfortunately one may found a strong upper bound to the size of the problem due to ill-conditioning. This is not a

*This work was supported by MRE grant No. 974, the CNRS-NSF grant No. 950.22/ 07 and the research program C3

†This work was supported by Cromey's foundation

priori a good way to exploit a massively parallel architecture since we will not be able to rely on the Gustafson's law [12]. Second we are only interested in the elapsed time necessary to solve our bifurcations because we address some scientific computing issues that are important to us in the understanding of a two step sequential mechanism. To measure the performance of our algorithm, we will compare our result in terms of accuracy and elapsed time to one of the best sequential code we have used. This sequential code solve the same model problem and was optimized on a vector computer.

The plan of the paper is as follows. In section 2, we describe the model. In section 3, we present briefly the numerical method and refer for more details to [2, 8]. In section 4, we describe the parallel algorithm. In section 5, we finally present the implementation of the scheme and gives the results.

2 The Model

We consider a combustion model with the sequential reaction mechanism:

$$A \rightarrow \mu B \rightarrow C,$$

where μ is the stoichiometric coefficient for the first reaction. Reactant A is converted to an intermediate species B prior to being burned and converted to the final product C. This highly simplified model have been used with considerable success in modeling the combustion of hydrocarbons [17]. In this paper, we consider the case of a cylindrical flame stabilized by a line source of fuel. The problem involves three variables— θ, a scaled temperature, C_1, the concentration of reactant A, and C_2, the concentration of the intermediate species B, all functions of the polar coordinates r and ψ— and corresponds to a thermodiffusive model for a two-step Arrhenius reaction [14]. The equations satisfied by θ, C_1 and C_2 are,

$$\begin{aligned}
\frac{\partial \theta}{\partial r} &= \Delta\theta - \frac{K}{r}\frac{\partial \theta}{\partial r} + \alpha Da_1 C_1 R_1(\theta) + (1-\alpha) Da_2 C_2 R_2(\theta), \\
\frac{\partial C_1}{\partial r} &= \frac{1}{L_1}\Delta C_1 - \frac{K}{r}\frac{\partial C_1}{\partial r} - Da_1 C_1 R_1(\theta), \\
\frac{\partial C_2}{\partial r} &= \frac{1}{L_2}\Delta C_2 - \frac{K}{r}\frac{\partial C_2}{\partial r} + Da_1 C_1 R_1(\theta) - Da_2 C_2 R_2(\theta),
\end{aligned}$$

where R_1 and R_2 are the nonlinear source terms:

$$R_1(\theta) = \exp\frac{\beta_1(\theta-\theta_0)}{1+\gamma_1(\theta-\theta_0))}, \; R_2(\theta) = \exp\frac{\beta_2(\theta-1)}{1+\gamma_2(\theta-1))},$$

Δ is the Laplacian in polar coordinates. The parameters are L_1 and L_2, the Lewis numbers; Da_1 and Da_2, the Daemker numbers; N_1 and N_2 the activation energy of each chemical reaction; α the heat release of the first reaction; and K, a measure for the strength of the fuel injection; $\theta_0 \in [0,1]$ is a (scaled) temperature close to the temperature at which the first reaction occurs. β_i is proportionnal to the activation energy N_i. The coefficients γ_i are function of the temperature of the unburned fresh mixture, the final adiabatic burned temperature and θ_0. The activation energy of each chemical reaction is large and consequently β_i; $i = 1, 2$ as well as one of the Daemker number at least are large. The derivation of this model but for cartesian geometry has been done in [14, 15].

θ, C_1 and C_2 are periodic functions of ψ. The boundary conditions satisfied by θ, C_1 and C_2 are

$$\theta \rightarrow 0,\; C_1 \rightarrow 1,\; C_2 \rightarrow 0, \text{ as } r \rightarrow 0; \quad \theta \rightarrow 1,\; C_1 \rightarrow 0,\; C_2 \rightarrow 0, \text{ as } r \rightarrow \infty.$$

For the computational domain we take $(r, \psi) \in (r_0, r_1) \times (0, 2\pi)$, where $0 < r_0 < r_1 < \infty$; r_0 is sufficiently small and r_1 sufficiently large that the boundary conditions can be replaced by the Dirichlet conditions,

$$\theta(r_0, \psi) = 0,\ C_1(r_0, \psi) = 1,\ C_2(r_0, \psi) = 0;$$
$$\theta(r_1, \psi) = 1,\ C_2(r_1, \psi) = 0,\ C_2(r_1, \psi) = 0.$$

Because the activation energy of the chemical reaction is large, the flame is a thin layer, and the combustion problem is of singular perturbation type. Since we have two chemical reactions we have two combustion fronts. Depending on the value of the parameters these two fronts may merge or being widely separated. We are mainly interested in the nonlinear interaction between these two fronts and the pattern formation of cellular flames. The asymptotic analysis of thermodiffusive model with a two step chemical reaction can be found in [13, 14, 15]. In these papers different regimes have been considered. A linear stability analysis of planar freely propagating flames suggest the possibility of cellular flame when the first Lewis number L_1 is sufficiently smaller than 1. This is analogous to the case of a single reaction mechanism. The main goal of this paper is to show how to carry out the direct numerical simulation of such cellular flame on a parallel computer.

3 The Numerical Method

Our numerical method is based on an adaptive pseudo-spectral domain decomposition. We first describe briefly the standard pseudo-spectral method [5, 10]. Let us consider the simple one dimensional model:

$$u_t \ = \ u_{xx} \ + \ R(u),\ -1 \leq x \leq 1,$$

The solution is approximated by expanding u as a finite sum of Chebyshev polynomials

$$u \sim u_J = \sum_{j=0}^{J} a_j T_j(x).$$

where the coefficients a_j are obtained from collocation that is u_J satisfies the equation at a set of $J+1$ points $x_j,\ j = 0..J$. We use the Gauss-Lobatto points:

$$x_j \ = \ cos(\frac{j\Pi}{J}),\ j = 0..J\ .$$

The major advantage of pseudo-spectral method compare to finite difference method is the enhanced accuracy: the pseudospectral approximation is infinit order accurate provide u is sufficiently differentiable. As a draw back, the pseudospectral approximation is prone to errors when the approximate function is stiff.

The main difficulty to compute the solution of our combustion problem are the zone of rapid variation of the temperature and concentration of the reactants. More precisely, we have one transition layer and possibly two separate transition layers that correspond to the sharp combustion front(s): the non linear terms $Da_1 C_1 R_1(\theta)$ and $Da_2 C_2 R_2(\theta)$ are of order one *only* in these layers where the solution is stiff. The all dynamics of the combustion process is driven by these thin layers; consequently it is critical to compute very accurately these layers.

In order to take advantage of the high accuracy of the standard pseudospectral method, we have to introduce an adaptive scheme. It is classical now to use a coordinate transform that stretches the layer as one do in the construction of an asymptotic expansion [3, 6, 11]. For concreteness we consider the singular perturbation problem:

$$u_t \ = \ u_{xx} \ + \ NR(u),\ -1 \leq x \leq 1,$$

where N is a large parameter. We suppose that we have a layer located at x_0. Then we introduce a one parameter family of mapping:

$$s \in [-1, 1] \rightarrow y = f(s, \varepsilon) \in [-1, 1]$$

where ε is a small parameter that describes how one concentrates the collocation points in the physical space. For example, if x_0 is at one end of the interval, we use a mapping of boundary layer type as:

$$f(s, \varepsilon) = \mp(\frac{4}{\pi}\ atan(\varepsilon\ tan(\frac{\pi}{4}\ (\pm s - 1)). \tag{3.1}$$

We use a plus sign (respt a minus sign) for $x_0 = 1$ (respt $x_0 = -1$). If $x_0 = 0$, we use a mapping of transition layer type as:

$$f(s, \varepsilon) = \varepsilon\ tan(\ s\ atan(\ \varepsilon^{-1}\ )). \tag{3.2}$$

If e_J represents the numerical error, we can use an a priori estimate [4, 3] :

$$|e_J|\ \leq \frac{C^t}{N^p}||u||_\varepsilon,$$

to obtain the best stretching parameter ε such that $||u||_\varepsilon$ is minimum. One can show also that ε is related to the thickness of the layer [8].

If our singular perturbation problem has two different layers located in $x_0 \neq x_1$, the previous change of coordinate cannot solves both layers. So we introduce to solve accurately both layers an additional degree of freedom in our adaptivity method with a domain decomposition method see [7, 9]. We can split the domain of computation $[-1, 1]$ into two subdomains in such a way that each subdomain contains only one layer.

In fact, even in the case where we want to solve a single inner layer a domain decomposition might be useful. A number of numerical experiments [2, 8] suggest that it is better to solve an inner layer located at x_0 with two subdomains, the interface between the subdomain located in x_0 and the mapping of boundary layer type (3.1) in each subdomain rather than a single domain with the mapping of transition layer type (3.2). For the combustion problem that we investigate, the accuracy is at least equivalent, but also the natural concentration of the Chebyshev points makes the adaptive approximation less sensitive to the error on the position of the inner layer. Also in the regime where we have two widely separated fronts $x_0 \neq x_1$, we use four subdomains, two of each to solve each inner layer.

The way that the interface can be found in the algorithm has not yet been defined. As suggested in [1], we can use an a priori estimate [4, 3] parametrized by the interface position as well as the stretching parameters of each subdomain. However the resulting minimization problem is relatively complex to solve because it depends on several parameters. Since the interface(s) position(s) corresponds to the location of the layer(s), we have chosen the interface(s) location(s) to be the maximum of each reaction term(s) $Da_i C_i R_i$; this choice is motivated by the asymptotic analysis of the model problem and we have checked that the asymptotic criterium on the interface location is in good agreement with the adaptivity criterium in [2].

Because we compute some cellular flame, the position of the combustion front may depends on the angle: so we have to average the position of the fronts defined above with respect to the angle. The boundary layer mapping is quiet robust and makes this adaptive domain decomposition working well even if the front oscillates as in Figure 1. In particular the case $L_1 = 0.37$ and $L_2 = 0.9$ corresponds to a 4-cells with a single front structure; we have tested the accuracy of the solution with various grids and number of grid points. We have also compared this solution obtained with a two subdomains and boundary layer mappings strategy with a single domain code that use a transition layer mapping [8]. The accuracy of our computation in the maximum norm with 12000 degrees of freedom is of order $10-4$. This accuracy is analogous to the accuracy of the solution given by the sequential code.

Finally as we will see, the domain decomposition add to the potential of parallelism of our algorithm.

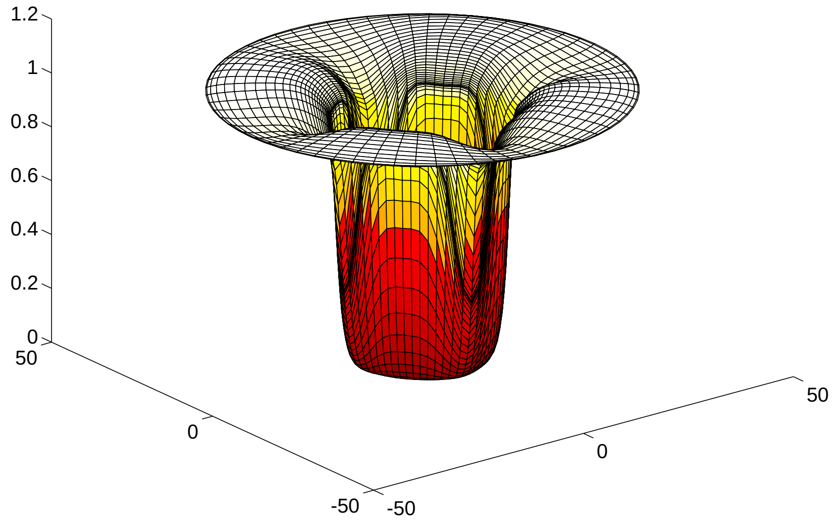

Figure 1: Temperature field with a stationary 4 cells solution.

4 The Parallel Algorithm

We are first going to describe the algorithm with the scalar equation:

$$\frac{\partial u}{\partial t} = \Delta u + F(u); \quad (r, \psi) \in \Omega = [r_o, r_1] \times [0, 2\pi],$$

where Δ is the laplacian in cylindrical coordinate. We consider the following Euler scheme:

$$\frac{u^{n+1} - u^n}{t^{n+1} - t^n} = D_{rr} u^{n+1} + \frac{1}{r^2} D_{\psi\psi} u^n + F(u^n), \tag{4.3}$$

where D_{rr} is the operator of differentiation with respect to the radius r and $D_{\psi\psi}$ is the operator of differentiation with respect to the angle ψ.

Let $\{\psi_i, i = 1..n_a\}$ a regular discretization with respect to the angle of step size $h = \frac{2\pi}{n_a - 1}$ and u_i the restriction of the solution at $\psi = \psi_i$.

We can rewrite (4.3) as:

$$\hat{D} u_i^{n+1} = \frac{1}{r^2} D_{\psi\psi} u_i^n + F(u_i^n), \; for \; i = 1..n_a. \tag{4.4}$$

We first use the explicit dependence on the angle in this scheme in order to have a parallel algorithm. This choice will be justify later on. We consider a finite difference approximation of order 6 of $\frac{1}{r^2}\frac{\partial^2 u}{\partial \psi^2}$: consequently $D_{\psi\psi} u_i^n$ depends only on u_j for $j = i-3, ..., i+3$. In order to overcome the constraint on the time step due to the explicit treatment of $\frac{\partial^2 u}{\partial \psi^2}$, we use a second order finite difference formula of space step $2h$ in the subdomain Ω^* of Ω restricted to the disk $r < r^*$. r^* will be defined later. Each iteration in time can be summarized as:

do in parallel for each angle $\psi = \psi_i$

- task 1: compute $F(u_i^n)$ and read u_j for $j = i-3, ..., i+3$, $j \neq i$.

- task 2: compute $D_{\psi\psi} u_i^n$.
- task 3: compute the solution of the linear system (4.4).

The previous algorithm can be applied in a similar way to our system of non linear PDE's with $u_i^n = (\Theta_i^n, C_{1,i}^n, C_{2,i}^n)$. In addition to the Euler scheme, we have used a predictor-corrector scheme of order two for the time marching scheme, but this does not add any difficulties from the point of view of getting a parallel algorithm.

We can justify the fact that the scheme is explicit in ψ in the following way: the front of combustion is mainly stiff in the radial direction and consequently we have for the temperature $\frac{\partial^2\theta}{\partial\psi^2} \prec\prec \frac{\partial^2\theta}{\partial r^2}$ as well as for C_1 and C_2. In addition the time step in our experiment was mainly restricted by the explicit treatment of the nonlinear reaction terms when $r^* = 4$ and $n_a = 64$ instead of the explicit treatment of $\frac{\partial^2\theta}{\partial\psi^2}$. In the disk $r < r*$, the dependence of the unknown Θ, C_1, C_2 on ψ is extremely weak because the cellular unstability start near the combustion front and then is rapidly damped by the diffusion process. Also the convection of the fuel is in the external direction and it limits the propagation of the unstability in the direction of the point source of injection of the fuel. So a low order approximation of the second order derivative with respect to the angle in the subdomain Ω^* does not affect the global quality of the result. In our experiments, the second order space derivative with respect to the angle of the unknowns are of order 10^{-6} at most in Ω^*

In addition to the parallelism based on the explicit terms discussed above, we have two others levels of parallelism that have to be considered. First our system of PDE is decoupled within each iteration step and the computation can therefore be split into three sub-blocks of processors corresponding to each equations. Second is the use of the domain decomposition along the radius. Let n_d be the number of radial symetric subdomains that we use. For each angle the linear system (4.4) can be solved with n_d parallel processes only coupled thru the resolution of the interface problem which is a tridiagonal linear system of size $n_d - 1$ [8, 9]. Future work will exploit this additional potential of parallelism and we restrict ourselves to use the parallelism based on the explicit angle dependence.

We have implemented the numerical simulation on a ring of P processors. Data are distributed among the ring in a natural way.

The ring is embedded in in a chosen topology (hypercube or grid), depending on the target machine (iPSC/860 or Paragon). The parallel algorithms of the first level of parallelization require lots of communications between processors owning neighboring angles. To avoid conflicts, we use a data distribution that ensure that communications occur just between neighbors.

The computation of the finite difference of order six try to maximize the overlap of communications by computations. Using the previous mapping, at step k, every processor computes an internal set of angles without any communications. But boundary angles need data which are mapped on neighbors on the left and on the right.

To reduce the number of startup costs, the different data of the temperature and concentrations are put together in a buffer (B_{left} and B_{right} on Figure 2) and sent in one asynchronous communication step. Because our target machines are 1-port, we use an "odd/even" communication scheme. At the first step, processors with an even (odd) logical number exchange their buffers with their left's (right's) neighbor. The cost of message buffering is negligible in front of the startup costs.

So the communication can be done as soon as possible (overlapped by other computations, i.e. computation of second order derivatives with respect to ψ of interior points, computation of the first order space derivatives with respect to r and computation of the reaction terms). The corresponding algorithm is given on Figure 2. `NBexchange` is a non-blocking exchange routine.

5 Implementation of the Parallel Algorithm and results

To maximize the overlap of communications by computations and to minimize the cost of the communications, data for θ, C_1 and C_2 are exchanged in the same communication (to avoid

```
my_id = f^{-1}(my_id_log)
⋮
store data of θ, C1 and C2 in B_left for the left
store data of θ, C1 and C2 in B_right for the right
if (my_id_log is even) then
   NBexchange B_left with the left
else
   NBexchange B_right with the right
endif
if (f¹(my_id) is even) then
   NBexchange B_right with the right
else
   NBexchange B_left with the left
endif
Compute ∂²θ/∂ψ², ∂²C1/∂ψ² and ∂²C2/∂ψ² for the interior points independent of the neighbors
Compute the derivatives ∂θ/∂r, ∂C1/∂r and ∂C2/∂r
Compute the source terms Da_1 C_1 R_1 and Da_2 C_2 R_2
wait for the exchanges to complete
Compute ∂²θ/∂ψ², ∂²C1/∂ψ² and ∂²C2/∂ψ² for the interior points dependent of the neighbors
Compute the Right Hand Sides of the linear systems.
Solve the linear systems
⋮
```

Figure 2: One part of the parallel algorithm on a ring.

Nbr of proc.	n_a			
	32	64	128	256
1	38.29	76.021	152.07	/
2	22.64	44.36	85.45	175.0
4	11.80	22.66	44.36	88.46
8	6.19	11.78	22.66	44.38
16	/	6.21	11.82	22.76
32	/	/	6.22	11.83
64	/	/	/	6.21

Table 1: Timing results in seconds for the Intel iPSC/860 with up to 64 processors

extra startup costs). To ensure that links of the ring are used in a full-duplex mode, we used a synchronization of the processors two-by-two [16] using a exchange of a null message.

Timings are given in Table 1 and 2. These results are obtained for a small number of Chebyshev points $N = 24$ and for $n_d = 4$ and n_a varying from 32 to 256. "/" entries in the table mean that the problem was too large for the memory size of the target machine or the number of discrete angles n_a was to small for the number of processors of the ring.

We notice that speedups are very good, even for a small number of angles per processor and that the algorithm scales very well, i.e. when increasing the number of processors by a factor of two as well as n_a, the execution time stays roughly the same. Timing results on the Paragon machine are 35% better than on the iPSC/860. This improvment in the performance

Nbr of proc.	n_a 32	64	128	256
1	26.87	53.61	107.03	/
2	16.47	31.83	62.71	125.00
4	8.91	16.60	32.03	63.10
8	5.04	8.95	16.65	32.13
16	/	5.06	8.96	16.67
32	/	/	5.08	8.99
64	/	/	/	5.10

Table 2: Timing results in seconds for the Intel Paragon with up to 64 processors

is essentially a consequence of the fact that the processor i860 XP of the Paragon is 25% faster than the i860 XR of the iPSC860. Also, the compiler is not the same between the two machines. On the Paragon machine, communication bandwidth is five times bigger than on the iPSC/860 (12.8Mb/s compared to 2.5Mb/s). However, as our communications are fully overlapped, we only measure the parallel computation time. The first two rows of table 1 and 2 show the cost of messages buffering and communications initializations.

In conclusion, we notice that the parallel code on the Paragon using 16 nodes is two time faster than a fully vectorized sequential code on one node of CRAY YMP for the computation of the same problem with analogous accuracy.

Acknowledgments

The authors want to thanks J.Dongarra because some experiments were done on the iPSC/860 and Paragon machines of ORNL during the visit of the first author at the University of Tennessee. The production code was elaborated on the Paragon of the Centre Pour le Développement du Calcul Scientifique Parallèle of Lyon.

References

[1] A. Bayliss, T. Belytdchko, D. Hansen, and M. Minkoff. Adaptive Multi-Domain Spectral Methods. In Meurant Scroggs Keyes, Chan and Voigt, editors, *5th SIAM Conference on Domain Decomposition Methods for Partial Differential Equations.* SIAM Philadelphia, 1992.

[2] A. Bayliss, M. Garbey, and B.J. Matkowsky. Pseudo-Spectral Domain Decomposition and the Approximation of Multiple Layers. Northwestern University Preprint, Evanston, IL, 1993.

[3] A. Bayliss, D. Gottlieb, B.J. Matkowsky, and M. Minkoff. An Adaptive Pseudo-Spectral Method For Reaction Diffusion Problems. *J. Comput. Phys.*, 81:421–443, 1989.

[4] A. Bayliss, B.J. Matkowsky, and M. Minkoff. Adaptive Pseudospectral Computation of Cellular Flame Stabilized by a Point Source. *Appl. Math. Letters*, 1:19–24, 1987.

[5] C. Canuto, M.Y. Hussaini, A. Quarteroni, and T.A. Zang. *Spectral Methods in Fluid Dynamics.* Springer Verlag - New York, 1987.

[6] W. Eckhaus. Asymptotic Analysis of Singular Perturbations. In *Studies in Mathematics and its Applications*, volume 9. North-Holland Publ. Co., 1979.

[7] U. Ehrenstein, H. Guillard, and R. Peyret. Flame Computations by a Chebyshev Multi-Domain Method. volume 9, pages 499–515, 1989.

[8] M. Garbey. Domain Decomposition to Solve Layers and Asymptotic. *SIAM Journal on Scientific Computing*, 15(4), July 1994. To appear.

[9] D. Gottlieb and R.S. Hirsh. Parallel pseudospectral domain decomposition techniques. ICASE Report, 1986.

[10] D. Gottlieb and S.A. Orszag. Numerical analysis of spectral methods: Theory and applications. In *C.B.M.S.-N.S.F. Conference Series in Applied Mathematics*. SIAM, Philadelphia, 1977.

[11] H. Guillard and R. Peyret. On the Use of Spectral Methods for the Numerical Solution of Stiff Problems. *Comput. Methods Appl. Mech. Engrg.*, 66:17–43, 1988.

[12] J.L. Gustafson. The Scaled-Sized Model: A Revision of Amdahl's Law, 1988.

[13] S.B. Margolis and B.J. Matkowsky. Steady and Pulsating Modes of Sequential Flame Propagation. *Comb. Sci. Tech.*, 27:193–213, 1982.

[14] J. Pelaez. Stability of Premixed Flames with Two Thin Reactions Layers. *SIAM J. Appl. Math.*, 47:781–799, 1987.

[15] J. Pelaez and A. Linan. Structure and Stability of Flames with Two Sequential Reactions. *SIAMJAP*, 45:503–522, 1985.

[16] S.R. Seidel, M.H. Lee, and S. Fotebar. Concurrent Bidirectional Communications on the Intel iPSC/860 and iPSC/2. In *Sixth Distributed Memory Computing Conference*, pages 283–286, 1991.

[17] C.K. Westbrook and F.L. Dryer. Simplified Reaction Mechanism for the Oxidation of Hydrocarbon Fuels in Flames. *Comb. Sci. Tech.*, 27:31–43, 1981.