Combust. Theory Modelling **1** (1997) 41–63. Printed in the UK PII: S1364-7830(97)78225-1

# Computationally efficient implementation of combustion chemistry using *in situ* adaptive tabulation

S B Pope

Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853, USA

Received 26 August 1996, in final form 21 January 1997

**Abstract.** A computational technique is described and demonstrated that can decrease by three orders of magnitude the computer time required to treat detailed chemistry in reactive flow calculations. The method is based on the *in situ* adaptive tabulation (ISAT) of the accessed region of the composition space—the adaptation being to control the tabulation errors. Test calculations are performed for non-premixed methane–air combustion in a statistically-homogeneous turbulent reactor, using a kinetic mechanism with 16 species and 41 reactions. The results show excellent control of the tabulation errors with respect to a specified error tolerance; and a speed-up factor of about 1000 is obtained compared to the direct approach of numerically integrating the reaction equations. In the context of PDF methods, the ISAT technique makes feasible the use of detailed kinetic mechanisms in calculations of turbulent combustion. The technique can also be used with reduced mechanisms, and in other approaches for calculating reactive flows (e.g. finite difference methods).

## 1. Introduction

A detailed description of combustion chemistry typically involves tens of species, hundreds of reactions, and timescales from $10^{-9}$ s to over 1 s. It is computationally demanding to solve the conservation equations including detailed chemistry for simple two-dimensional laminar flames (Smooke *et al* 1989). For more complex flows, especially turbulent flames, the computational cost of making calculations including detailed chemistry is usually deemed to be prohibitive. This paper introduces a new computational technique for implementing detailed chemistry that reduces the CPU time required by, typically, three orders of magnitude. The technique—*in situ* adaptive tabulation (ISAT)—is described and demonstrated here in the context of PDF methods for turbulent combustion (Pope 1985), but it can also be used in conjunction with other approaches. In addition, it can be used with full kinetic mechanisms (e.g. 50 species), with skeletal mechanisms (e.g. 20 species), or with reduced mechanisms (e.g. 4 degrees of freedom) with comparable benefits.

## 2. Formulation of the problem

At any point and time in a reactive gaseous flow, the thermochemical state of the mixture can be characterized by the mass fractions $Y_i$ $(i = 1, 2, \ldots, n_s)$ of the $n_s$ species, the enthalpy $h$, and the pressure $P$. We consider the broad class of flows in which $P$ differs by a very small fraction from a fixed reference pressure $P_0$, so that, given $P_0$, the state is determined by

$$\widehat{\phi} = \{\hat{\phi}_1, \hat{\phi}_2, \ldots, \hat{\phi}_{n_s+1}\} = \{Y_1, Y_2, \ldots, Y_{n_s}, h\}. \quad (1)$$

The components of $\widehat{\phi}$ are not linearly independent, because the mass fractions sum to unity. There may also be other dependences related to element and enthalpy conservation. If there are $n_\ell$ linear dependences, then there are

$$D = n_s + 1 - n_\ell, \tag{2}$$

degrees of freedom in the thermochemistry. We define the *composition*

$$\phi = \{\phi_1, \phi_2, \ldots, \phi_D\}, \tag{3}$$

to be a linearly independent subset of $\widehat{\phi}$. Given $P_0$ and a knowledge of the linear dependences, the thermochemical state of the fluid is completely determined by $\phi$.

It is valuable to treat the composition $\phi$ as a vector or, equivalently, as a point in the $D$-dimensional *composition space*. All physically possible values of $\phi$ define the *realizable region* of the composition space. In particular, in the realizable region the species mass fractions are non-negative.

In the computational implementation of PDF methods (Pope 1985), the fluid within the solution domain is represented by a large number of computational particles. The composition of each particle evolves according to the set of ordinary differential equations:

$$\frac{\mathrm{d}\phi(t)}{\mathrm{d}t} = \boldsymbol{S}(\phi[t]) + \boldsymbol{M}(t), \tag{4}$$

where $\boldsymbol{S}$ is the rate of change due to chemical reactions, and $\boldsymbol{M}$ is the rate of change due to transport. Enthalpy is conserved in adiabatic isobaric reactions, so that the corresponding component of $\boldsymbol{S}$ is zero. Given the chemical kinetics of the system, the other components of $\boldsymbol{S}$ are known in terms of $\phi$. For the species, the transport is due to molecular diffusion—or mixing: for enthalpy it is due to thermal conductivity and radiation. We refer to $\boldsymbol{M}$ as the *mixing* term.

It is important to appreciate the different timescales involved in equation (4). Chemical timescales are given by the inverses of the absolute values of the eigenvalues of the Jacobian matrix $\partial S_i/\partial\phi_j$. In combustion problems these timescales typically range from $10^{-9}$ s to 1 s or more. At any time $t$, the mixing term can be expressed uniquely as

$$\boldsymbol{M} = (\phi^B - \phi)/\tau_{\mathrm{mix}}, \tag{5}$$

where $\phi^B$ is a composition on the boundary of the realizable region and the positive quantity $\tau_{\mathrm{mix}}$ is identified as the mixing timescale. Typically $\tau_{\mathrm{mix}}$ is no smaller than $10^{-3}$ s or $10^{-4}$ s.

The composition evolution equation (4) is usually solved by a simple splitting method. For example, the solution is advanced from time $t_0$ for a small time step $\Delta t$ by the following procedure.

(i) From the initial condition $\phi(t_0)$, the mixing equation

$$\frac{\mathrm{d}\phi(t)}{\mathrm{d}t} = \boldsymbol{M}(t), \tag{6}$$

is integrated for a time $\Delta t$, and the solution is denoted by $\phi^0$.

(ii) From the initial condition $\phi^0$, the reaction equation

$$\frac{\mathrm{d}\phi(t)}{\mathrm{d}t} = \boldsymbol{S}(\phi[t]), \tag{7}$$

is integrated for a time $\Delta t$, to obtain $\tilde{\phi}(t_0 + \Delta t)$—the approximation to $\phi(t_0 + \Delta t)$.

A simple analysis shows that, in the limit as $\Delta t$ tends to zero, the (local) error on each time step $|\tilde{\phi}(t_0 + \Delta t) - \phi(t_0 + \Delta t)|$ is $O(\Delta t^2)$ so that the (global) error for a fixed time interval is $O(\Delta t)$—i.e. the method is first-order accurate. In practice, the time step is selected to be small compared to the mixing timescale (e.g. $\Delta t = \frac{1}{10}\tau_{\text{mix}}$), but this may be several orders of magnitude larger than the smallest chemical timescale. Although the simple analysis is not valid for such a choice of $\Delta t$, numerical tests nevertheless confirm that the method is indeed first-order accurate (Yang and Pope 1996a).

The problem considered here is the computationally efficient solution of the reaction equation (7) for a fixed time interval $\Delta t$ from a large number of different initial conditions $\phi^0$.

In a full-scale PDF method calculation, there may be 500 000 particles, and the solution may be advanced for 2000 timesteps. Consequently, the solution to the reaction equation is required (of order) $10^9$ times.

As depicted in figure 1, the solution to the reaction equation from the initial condition $\phi(t_0) = \phi^0$ corresponds to a trajectory in composition space. At large times the trajectory tends to the point corresponding to a mixture in chemical equilibrium (with the same enthalpy and elemental composition as $\phi^0$). For fixed $\Delta t$, the solution $\phi(t_0 + \Delta t)$ is a unique function of $\phi^0$, denoted by $\boldsymbol{R}(\phi^0)$, which is called the *reaction mapping*: the integration of the reaction equation for the time $\Delta t$ maps the initial condition $\phi(t_0) = \phi^0$ to the reacted value $\phi(t_0 + \Delta t) = \boldsymbol{R}(\phi^0)$.

An alternative statement of the problem considered is, therefore, the computationally-efficient evaluation of the reaction mapping $\boldsymbol{R}(\phi^0)$ for (of order) $10^9$ initial conditions $\phi^0$ arising from the reactive flow calculation.

While we continue the present development in the context of PDF methods, it should be appreciated that other approaches to reactive flows entail a similar problem. For example, in a finite-difference or finite-volume calculation of a laminar or turbulent reactive flow, (4) governs the evolution of the composition at a grid node or cell-centre. In that case, the term $\boldsymbol{M}(t)$ also accounts for advection. If the solution is advanced in timesteps (or pseudo timesteps) $\Delta t$, and (4) is split via (6) and (7), then precisely the same problem arises: namely, to determine the mapping $\boldsymbol{R}(\phi^0)$ for every grid node on every timestep.

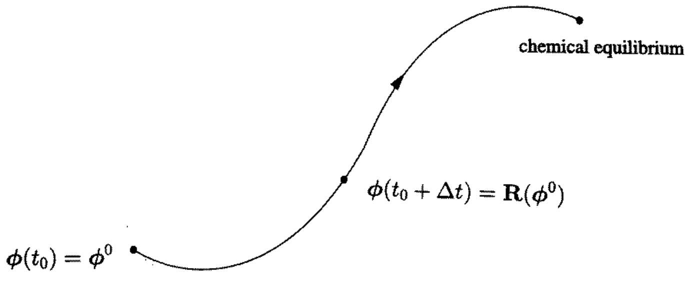

**Figure 1.** Sketch of the reaction trajectory from the initial condition $\phi^0$. The reaction mapping $\boldsymbol{R}(\phi^0)$ is the composition after reaction for a time $\Delta t$.

## 3. Approaches

### 3.1. Direct integration

The straightforward approach to determine the mapping $\boldsymbol{R}(\phi^0)$ is to integrate the reaction equation numerically. We refer to this as *direct integration*, or DI. For a detailed reaction mechanism (e.g. with more than 10 species), because the source term $\boldsymbol{S}(\phi[t])$ is expensive to evaluate, and because the coupled ordinary differential equations (7) are stiff, direct integration is computationally expensive. For example, with the 16-species mechanism for methane combustion used in the tests reported below, the average CPU time required to determine the mapping is 0.2 s†. Hence $10^9$ evaluations would require 6 years of CPU time, which is generally deemed to be prohibitive.

With the exception of the parallel work of Yang and Pope (1996b), no other approach applicable to detailed mechanisms has been demonstrated.

### 3.2. Reduction, storage and retrieval

An alternative approach, with a long and varied history, is to reduce radically the number of degrees of freedom in the description of the chemistry. Perhaps the most familiar of these approaches is mechanism reduction in which, through a combination of steady-state and partial-equilibrium assumptions, the minor species concentrations are determined from the major species (see e.g. Smooke 1991). Other methods include: constrained equilibrium (Keck and Gillespie 1971); computational singular perturbation (Lam and Goussis 1988); and, intrinsic low-dimensional manifolds (Maas and Pope 1992a, b, 1994). All these approaches reduce the number of degrees of freedom in the description of the chemistry, typically to four or less; but this is at the cost of introducing assumptions, often of uncertain generality and accuracy.

With a reduced mechanism, direct integration can be used to determine the mapping; but there are also other possibilities. The usual technique employed in PDF methods involves *tabulation* (e.g. Chen *et al* 1995, Taing *et al* 1993). There is a *pre-processing* stage in which direct integration is used to construct a table of values of the mapping $\boldsymbol{R}(\phi^0)$. In the simplest implementation, a regular mesh covers the realizable region of the composition space. Each mesh node represents a composition $\phi$, and the corresponding mapping $\boldsymbol{R}(\phi)$ is determined by direct integration and is then stored in the table. In the PDF method calculation, for each particle on each time step, the mapping is determined by a *table-look-up*, usually as a multi-linear interpolation. The interpolation error that is incurred can be controlled by refining the mesh.

Tabulation is just one example of the more general concept of *storage and retrieval*. Other techniques that have been used include neural networks (Christo *et al* 1996) and orthogonal polynomials (Turanyi 1994).

Criteria by which storage and retrieval techniques can be judged are:

(i) the CPU time required to create the store
(ii) the memory required for the store
(iii) inaccuracies in the retrieved mapping (e.g. interpolation errors)
(iv) the CPU time required to retrieve the mapping
(v) the degree to which the technique is generally applicable and automated.

If the chemistry is reduced to two or three degrees of freedom, tabulation fares well by criteria (i)–(iv). Interpolation errors can be made acceptably small with a table with modest

† All computations reported here were performed on an SGI Indigo 2 workstation.

storage requirements. The retrieval time may be three orders of magnitude smaller than the time to perform a direct integration, and the time to create the store is negligible when amortized over $10^9$ retrievals.

The direct application of tabulation to detailed kinetics is not feasible. For example, with $D = 10$ degrees of freedom in the description of the chemistry, and with a relatively coarse mesh of 10 nodes in each direction, the resulting table would require $D10^D = 10^{11}$ words of storage; and one multi-linear interpolation would require at least $D2^D \approx 10\ 000$ operations.

## 4. *In situ* adaptive tabulation (ISAT)

The present approach is a tabulation technique that overcomes the apparent obstacles just mentioned. In full, the method is: *in situ*, unstructured, adaptive tabulation of the accessed region, with control of retrieval errors.

### *4.1. Accessed region*

For a given reactive flow (or for a calculation thereof), the *accessed region* of the composition space is defined as the set of all compositions $\phi$ that occur in the flow (or calculation).

A crucial observation is that the accessed region is much smaller than the realizable region (of which it is a subset). To give just one illustration: consider a steady, laminar, axisymmetric flame involving, say, 50 species. The realizable region is a convex polytope in 50-space. But the accessed region is just a two-dimensional manifold, irrespective of the number of species and the complexity of the chemistry and transport. From some perspectives this result may be surprising and remarkable. From another, it is obvious: for all compositions that occur can be parametrized by the two spatial coordinates (i.e. $\phi = \phi[x, r]$).

In order to use a tabulation method for a particular flow, it is sufficient to tabulate the accessed region, rather than the whole of the realizable region. While the latter depends solely on the species involved, the accessed region depends on many aspects of the flow including the kinetics, the transport processes and the boundary conditions. It may have an irregular shape, and it is not known prior to the performance of the calculation. For these reasons the table is not constructed in a preprocessing phase, but rather it is built up during the reactive flow calculation. Each entry in the table corresponds to a composition that occurs in the calculation. This is referred to as *in situ* tabulation.

### *4.2. Linearized mapping*

The table contains information about the mapping $\boldsymbol{R}(\phi)$ at particular tabulation points, denoted generically by $\phi^0$. Figure 2 shows a sketch of the reaction trajectory from $\phi^0$, and also that from a nearby *query point* $\phi^q$.

As shown in the figure, displacements in the initial condition ($\delta\phi$) and in the mapping ($\delta\boldsymbol{R}$) are defined such that

$$\phi^q = \phi^0 + \delta\phi, \tag{8}$$

and

$$\boldsymbol{R}(\phi^q) = \boldsymbol{R}(\phi^0) + \delta\boldsymbol{R}. \tag{9}$$

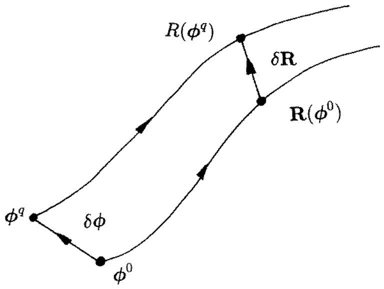

**Figure 2.** Sketch of the mappings from a tabulation point $\phi^0$ and a query point $\phi^q$ showing the definitions of the displacements $\delta\phi$ and $\delta\boldsymbol{R}$.

In the table, in addition to $\phi^0$ and $\boldsymbol{R}(\phi^0)$, the *mapping gradient matrix* $\boldsymbol{A}(\phi^0)$ is also stored: this is defined by

$$A_{ij}(\phi) \equiv \frac{\partial R_i(\phi)}{\partial \phi_j}. \tag{10}$$

From these tabulated quantities, a linear approximation to $\boldsymbol{R}(\phi^q)$ is obtained, i.e.

$$\boldsymbol{R}(\phi^q) \approx \boldsymbol{R}^\ell(\phi^q) \equiv \boldsymbol{R}(\phi^0) + \delta\boldsymbol{R}^\ell, \tag{11}$$

where

$$\begin{aligned} \delta\boldsymbol{R}^\ell &\equiv \boldsymbol{A}\delta\phi \\ &= \delta\boldsymbol{R} + O(|\delta\phi|^2). \end{aligned} \tag{12}$$

The accuracy of this approximation is considered in the following two subsections.

The mapping gradients $\boldsymbol{A}$ are related to sensitivity coefficients. Let the reaction equation (7) be solved from the the initial condition $\phi(0) = \phi^0$. Then the first order sensitivity coefficients with respect to initial conditions are defined by

$$B_{ij}(\phi^0, t) \equiv \frac{\partial \phi_i(t)}{\partial \phi_j^0}, \tag{13}$$

and then the mapping gradients are

$$\boldsymbol{A}(\phi^0) = \boldsymbol{B}(\phi^0, \Delta t). \tag{14}$$

It is readily deduced from (7) that $\boldsymbol{B}$ evolves according to the linear system of ordinary differential equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{B}(\phi^0, t) = \boldsymbol{J}(\phi[t])\boldsymbol{B}(\phi^0, t), \tag{15}$$

where $\boldsymbol{J}$ is the Jacobian

$$J_{ij}(\phi) \equiv \frac{\partial S_i(\phi)}{\partial \phi_j}, \tag{16}$$

and the initial condition is

$$\boldsymbol{B}(\phi^0, 0) = \boldsymbol{I}. \tag{17}$$

In the computational implementation of the method, (7) and (15) are solved together using the DDASAC code (Caracotsios and Stewart 1985) to obtain $\boldsymbol{R}(\phi^0)$ and $\boldsymbol{A}(\phi^0)$.

It is important to appreciate the significance and properties of the mapping matrix $\boldsymbol{A}$. Let the singular value decomposition (SVD) of $\boldsymbol{A}$ be

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T, \tag{18}$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are unitary matrices, and $\boldsymbol{\Sigma}$ is the diagonal matrix of singular values, $\sigma_1 \geqslant \sigma_2 \geqslant \cdots \geqslant \sigma_D \geqslant 0$. Then the linear approximation (12) can be rewritten

$$\delta \widehat{R}_i^\ell = \sigma_i \delta\tilde{\phi}_i, \quad i = 1, 2, \ldots, D, \tag{19}$$

where $\delta\widehat{\boldsymbol{R}}^\ell$ and $\delta\tilde{\boldsymbol{\phi}}$ are the displacements expressed in different bases:

$$\delta\widehat{\boldsymbol{R}}^\ell \equiv \boldsymbol{U}^T\delta\boldsymbol{R}^\ell, \tag{20}$$

$$\delta\tilde{\boldsymbol{\phi}} \equiv \boldsymbol{V}^T\delta\boldsymbol{\phi}. \tag{21}$$

Thus $\sigma_i$ represents the sensitivity of the mapping $\boldsymbol{R}(\boldsymbol{\phi})$ to a perturbation of $\boldsymbol{\phi}$ in the $i$th singular direction.

Both $\boldsymbol{R}$ and $\boldsymbol{A}$ are defined for a specified time interval $\Delta t$. In the limit as $\Delta t$ tends to zero, $\boldsymbol{A}$ tends to the identity $\boldsymbol{I}$, and so all the singular values are unity. (But recall that $\Delta t$ is not small compared to the smallest chemical timescale.) At the other extreme, as $\Delta t$ tends to infinity, $\boldsymbol{R}(\boldsymbol{\phi})$ tends to the equilibrium composition determined by the elemental composition and enthalpy. If there are $n_e$ elements and $n_s$ species, then there may be as many as $(n_s - n_e)$ directions† in composition space in which the elemental composition and enthalpy are constant. To each of these directions there is a corresponding zero singular value.

Figure 3 is a scatter plot of the singular values for about 200 representative samples of the mapping matrix $\boldsymbol{A}$ taken from the test calculation (described below) for non-premixed methane combustion. A skeletal reaction mechanism is employed which results in there being $D = 14$ degrees of freedom in the description of the thermochemistry. The first observation to be made from figure 3 is that many singular values are essentially zero: in nearly all cases there are five near-zero singular values, and in some cases there are 10. These near-zero singular values correspond to linear combinations of reactions that come to partial equilibrium in the time interval $\Delta t$.

It may also be observed from figure 3 that in all cases the first singular value $\sigma_1$ is greater than unity, and can be greater than 100, corresponding to extreme sensitivity.

### 4.3. Local error

For a query point $\boldsymbol{\phi}^q$ close to a tabulation point $\boldsymbol{\phi}^0$, the difference between the exact and linearized mappings is

$$\boldsymbol{R}(\boldsymbol{\phi}^q) - \boldsymbol{R}^\ell(\boldsymbol{\phi}^q) = \delta\boldsymbol{R} - \delta\boldsymbol{R}^\ell, \tag{22}$$

(where $\boldsymbol{R}^\ell$ and $\delta\boldsymbol{R}^\ell$ are defined by (11) and (12)). The local error could simply be defined as the magnitude of this difference, but it is preferable to allow for an appropriate scaling of the different components of $\boldsymbol{R}$. Accordingly we introduce a *scaling matrix* $\boldsymbol{B}$ and define the local error as

$$\begin{aligned}\varepsilon &\equiv |\boldsymbol{B}(\boldsymbol{R}[\boldsymbol{\phi}^q] - \boldsymbol{R}^\ell[\boldsymbol{\phi}^q])| \\ &= |\boldsymbol{B}(\delta\boldsymbol{R} - \delta\boldsymbol{R}^\ell)|.\end{aligned} \tag{23}$$

(Note that the simplest specification $\boldsymbol{B} = \boldsymbol{I}$ yields $\varepsilon = |\delta\boldsymbol{R} - \delta\boldsymbol{R}^\ell|$.)

† The precise number depends on the nature of the $n_\ell$ linear dependences between the components of $\widehat{\boldsymbol{\phi}}$, equation (1).

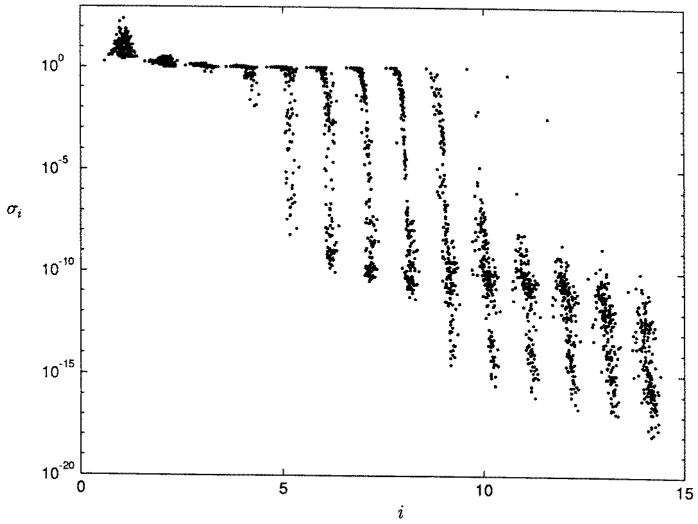

**Figure 3.** Singular values $\sigma_i$ of the mapping matrix $\boldsymbol{A}$ for about 200 representative compositions from the test case described in section 5.1. Each band corresponds to an index $i$. Within each band, the horizontal position of the point is based on temperature: cold compositions to the left, hot to the right.

The scaled composition is defined by

$$\psi = \boldsymbol{B}\phi. \tag{24}$$

In the tests reported below, $\boldsymbol{B}$ is taken to be the diagonal matrix such that, for species, $\psi_i$ is the mole fraction normalized by (an estimate of) the maximum mole fraction range in the problem†; whereas for enthalpy, $\psi_i$ is the enthalpy normalized by (an estimate of) the maximum enthalpy range in the problem.

### *4.4. Region of accuracy*

The accuracy of the method is controlled by using the linear approximation $\boldsymbol{R}^{\ell}$ at $\phi^0$ only if the local error is less than the specified tolerance $\varepsilon_{\mathrm{tol}}$. The *region of accuracy* is defined to be the connected region containing $\phi^0$ consisting of points $\phi^q$ for which the local error $\varepsilon$ does not exceed the tolerance $\varepsilon_{\mathrm{tol}}$. This is sketched in figure 4.

In order to investigate the region of accuracy, we consider the *constant approximation* in place of the linear approximation. That is, we consider the approximation

$$\boldsymbol{R}(\phi^q) \approx \boldsymbol{R}^c(\phi^q) \equiv \boldsymbol{R}(\phi^0), \tag{25}$$

with the corresponding local error

$$\varepsilon_c \equiv |\boldsymbol{B}(\boldsymbol{R}[\phi^q] - \boldsymbol{R}^c[\phi^q])|. \tag{26}$$

Now in this equation, $\boldsymbol{R}(\phi^q) - \boldsymbol{R}^c(\phi^q)$ is simply $\delta\boldsymbol{R}$; which, to leading order, is given by $\boldsymbol{A}\delta\phi$, equation (12). Thus we obtain the estimate

$$\varepsilon_c = |\boldsymbol{B}\boldsymbol{A}\delta\phi|. \tag{27}$$

† The same value of mole fraction is used to normalize all of the species mole fractions.

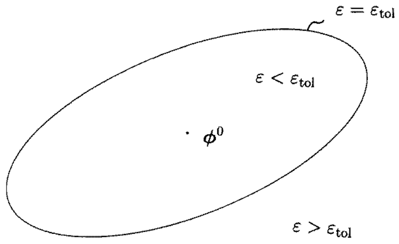
$\varepsilon = \varepsilon_{\text{tol}}$
$\varepsilon < \varepsilon_{\text{tol}}$
$\phi^0$
$\varepsilon > \varepsilon_{\text{tol}}$

**Figure 4.** Sketch of the region of accuracy about the point $\phi^0$.

On the boundary of the region of accuracy for the constant approximation, the error $\varepsilon_c$ equals the tolerance $\varepsilon_{\text{tol}}$, or

$$\varepsilon_{\text{tol}}^2 = \varepsilon_c^2 = \delta\boldsymbol{\phi}^T \boldsymbol{A}^T \boldsymbol{B}^T \boldsymbol{B}\boldsymbol{A}\delta\boldsymbol{\phi}. \tag{28}$$

The symmetric positive semi-definite matrix $\boldsymbol{A}^T \boldsymbol{B}^T \boldsymbol{B}\boldsymbol{A}/\varepsilon_{\text{tol}}^2$ can be expressed as

$$\boldsymbol{A}^T \boldsymbol{B}^T \boldsymbol{B}\boldsymbol{A}/\varepsilon_{\text{tol}}^2 = \boldsymbol{Q}^T \boldsymbol{\Lambda} \boldsymbol{Q}, \tag{29}$$

where $\boldsymbol{Q}$ is a unitary matrix and $\boldsymbol{\Lambda}$ is a diagonal matrix, with non-negative diagonal elements $\lambda_1, \lambda_2, \dots, \lambda_D$. Consequently (28) can be rewritten

$$1 = \delta\boldsymbol{\phi}^T \boldsymbol{Q}^T \boldsymbol{\Lambda} \boldsymbol{Q} \delta\boldsymbol{\phi}, \tag{30}$$

showing that the region of accuracy is a hyper-ellipsoid. The half-lengths of the principal axes are $\ell_i = 1/\sqrt{\lambda_i}$.

Suppose, further, that the scaling matrix $\boldsymbol{B}$ is taken to be the identity. Then, using the SVD of $\boldsymbol{A}$ (18), the left-hand side of (29) can be written

$$\begin{aligned} \boldsymbol{A}^T \boldsymbol{B}^T \boldsymbol{B}\boldsymbol{A}/\varepsilon_{\text{tol}}^2 &= \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T/\varepsilon_{\text{tol}}^2 \\ &= \boldsymbol{V}\left(\frac{\boldsymbol{\Sigma}}{\varepsilon_{\text{tol}}}\right)^2 \boldsymbol{V}^T. \end{aligned} \tag{31}$$

In this case, therefore, we have

$$\boldsymbol{\Lambda} = \left(\frac{\boldsymbol{\Sigma}}{\varepsilon_{\text{tol}}}\right)^2, \tag{32}$$

so that the half-lengths of the principal axes of the hyper-ellipsoid are

$$\ell_i = \varepsilon_{\text{tol}}/\sigma_i. \tag{33}$$

Clearly, the large range of singular values $\sigma_i$ (see figure 3) has substantial impact on the shape of the region of accuracy.

The development so far in this subsection applies to the constant approximation whereas the ISAT method employs the linear approximation. The corresponding analysis for the linear approximation is more complicated, and yields a tensor equation for the region of accuracy of the form

$$G_{ijk\ell}\delta\phi_i\delta\phi_j\delta\phi_k\delta\phi_\ell \leqslant 1. \tag{34}$$

Rather than using (34), the ISAT method represents and estimates the region of accuracy according to the following three rules.

(i) It is assumed that the region of accuracy for each tabulation point $\boldsymbol{\phi}^0$ is adequately approximated by a hyper-ellipsoid, called the *ellipsoid of accuracy* (EOA). The EOA is represented by the unitary matrix $\boldsymbol{Q}$ and the diagonal matrix $\boldsymbol{\Lambda}$: the query point $\boldsymbol{\phi}^q = \boldsymbol{\phi}^0 + \delta\boldsymbol{\phi}$ is inside the EOA if

$$\delta\boldsymbol{\phi}^T \boldsymbol{Q}^T \boldsymbol{\Lambda} \boldsymbol{Q} \delta\boldsymbol{\phi} \leqslant 1. \tag{35}$$

(ii) Based on (29), an initial (conservative) estimate of the EOA is

$$\boldsymbol{Q}^T \boldsymbol{\Lambda} \boldsymbol{Q} = \tilde{\boldsymbol{A}}^T \boldsymbol{B}^T \boldsymbol{B} \tilde{\boldsymbol{A}} / \varepsilon_{\mathrm{tol}}^2, \tag{36}$$

where $\tilde{\boldsymbol{A}}$ is a modification of $\boldsymbol{A}$. Specifically the SVD of $\tilde{\boldsymbol{A}}$ is

$$\tilde{\boldsymbol{A}} = \boldsymbol{U} \tilde{\boldsymbol{\Sigma}} \boldsymbol{V}^T, \tag{37}$$

where the singular value $\tilde{\sigma}_i$ is the maximum of the singular value of $\boldsymbol{A}$, $\sigma_i$, and $\frac{1}{2}$. This modification is to prevent small singular values from generating unduly large principal axes (see (33)).

(iii) If, in the course of the calculation, a query point $\boldsymbol{\phi}^q$ is encountered that is within the region of accuracy (i.e. $\varepsilon \leqslant \varepsilon_{\mathrm{tol}}$) and yet it is outside the current estimate of the EOA, then the EOA is *grown*.

This growing process is illustrated in figure 5. In the given circumstances, the new EOA is the hyper-ellipsoid of minimum volume, centred at $\boldsymbol{\phi}^0$, which encloses both the original EOA and the point $\boldsymbol{\phi}^q$.

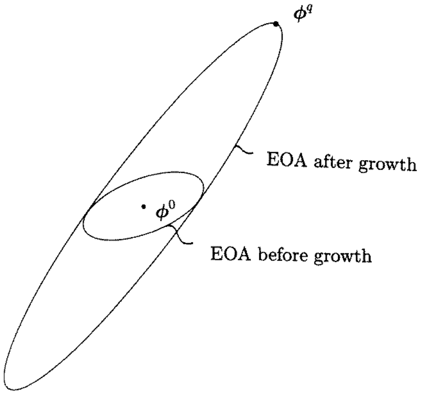

**Figure 5.** Sketch showing the growth on an ellipsoid of accuracy to include the point $\boldsymbol{\phi}^q$ at which the error $\varepsilon$ is less than the tolerance $\varepsilon_{\mathrm{tol}}$.

It should be appreciated that these rules contain assumptions and approximations, so that it cannot be guaranteed that the error $\varepsilon$ is indeed below the tolerance $\varepsilon_{\mathrm{tol}}$ for all points within the EOA. Tests reported below show that this strategy based on EOAs provides excellent—but not perfect—control of local errors.

### *4.5. Tabulation*

Figure 6 gives an overview of the functioning of the *in situ* adaptive tabulation (ISAT) procedure in terms of computer program modules. Initially, the reactive flow code provides ISAT with the timestep $\Delta t$, the scaling matrix $\boldsymbol{B}$ used to define the error, and the error tolerance $\varepsilon_{\mathrm{tol}}$. Subsequently, very many times, the reactive flow code gives ISAT a query composition $\phi^q$, and ISAT returns the corresponding mapping $\boldsymbol{R}(\phi^q)$ (to the required accuracy).

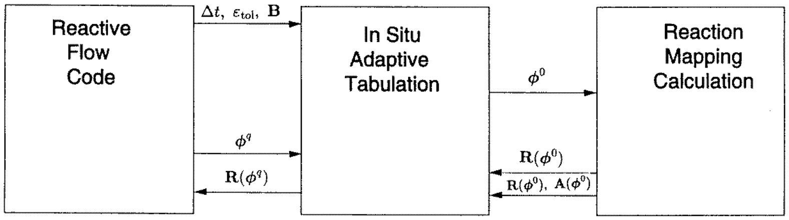
Reactive Flow Code
$\Delta t$, $\varepsilon_{\mathrm{tol}}$, $\mathbf{B}$
In Situ Adaptive Tabulation
$\phi^q$
$\mathbf{R}(\phi^q)$
$\phi^0$
Reaction Mapping Calculation
$\mathbf{R}(\phi^0)$
$\mathbf{R}(\phi^0)$, $\mathbf{A}(\phi^0)$

**Figure 6.** Overview of the interconnection between a reactive flow code, the *in situ* adaptive tabulation algorithm, and routines to determine the mapping and its gradient.

As the reactive flow calculation proceeds, and queries are received, ISAT builds a table by a process soon to be described. The ISAT module obtains the information it needs to build the table from the mapping module (which consists of chemistry and ODE routines). Given the timestep $\Delta t$ and a composition $\phi^0$, the mapping module returns the mapping $\boldsymbol{R}(\phi^0)$ and (optionally) the mapping gradient matrix $\boldsymbol{A}(\phi^0)$.

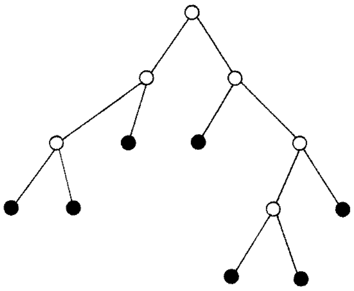

**Figure 7.** Sketch of the binary tree. At each leaf • there is a record; at each node ○ there is information about the cutting plane.

The table that is generated in the ISAT method consists of a *binary tree*—as sketched in figure 7—a set of *records*, one for each leaf of the tree, and a set of *cutting planes*, one for each node of the tree. Each record consists of:

$\phi^0$: the tabulation point<br>
$\boldsymbol{R}(\phi^0)$: the reaction mapping<br>
$\boldsymbol{A}(\phi^0)$: the mapping gradient<br>
$\boldsymbol{Q}$: the EOA unitary matrix<br>
$\boldsymbol{\lambda}$: the diagonal elements of the EOA matrix $\boldsymbol{\Lambda}$.

The first three items ($\phi^0$, $\boldsymbol{R}$ and $\boldsymbol{A}$) are computed once and are fixed, whereas $\boldsymbol{Q}$ and $\boldsymbol{\lambda}$ change whenever the EOA is grown.

On receipt of the first query $\phi^q$, the first record is generated (with $\phi^0 = \phi^q$), and the binary tree is initialized to a single leaf (corresponding to the single record). The exact value of the mapping $\boldsymbol{R}(\phi^0)$ is returned.

For subsequent queries, the algorithm is as follows.

(i) Given the query composition $\phi^q$, the binary tree is traversed until a leaf (denoted by $\phi^0$) is reached. (The tree is constructed so that $\phi^0$ is, in some sense, 'close' to $\phi^q$.)

(ii) Equation (35) is used to determine whether or not $\phi^q$ is within the EOA.

(iii) If $\phi^q$ is within the EOA, then the linear approximation,

$$\boldsymbol{R}^{\ell}(\phi^q) = \boldsymbol{R}(\phi^0) + \boldsymbol{A}(\phi^0)(\phi^q - \phi^0), \tag{38}$$

is returned. This, the first of three possible outcomes, is a *retrieve* $(R)$.

(iv) Otherwise (i.e. $\phi^q$ is outside the EOA) a direct integration is performed to determine the mapping $\boldsymbol{R}(\phi^q)$, and the error $\varepsilon$ is measured, equation (23).

(v) If the error $\varepsilon$ is within the tolerance, then the EOA is grown, and $\boldsymbol{R}(\phi^q)$ is returned. This outcome is a *growth* $(G)$.

(vi) Otherwise (i.e. $\varepsilon > \varepsilon_{\text{tol}}$) a new record is generated based on the query composition $\phi^q$. As sketched in figure 8, a *cutting plane* is defined by a vector $\boldsymbol{v}$ and a scalar $a$ such that all points $\phi$ with $\boldsymbol{v}^T\phi > a$ are deemed to be on the right of the cutting plane: all other points are on the left. The orientation of $\boldsymbol{v}$ is chosen so that $\phi^0$ is on the left and $\phi^q$ is on the right. The binary tree is then modified as sketched in figure 9. The original leaf $(\phi^0)$ is replaced by a node at which the cutting plane $(\boldsymbol{v}, a)$ is stored. The left child is then the leaf corresponding to $\phi^0$, and the right child $\phi^q$. This outcome is an *addition* $(A)$.

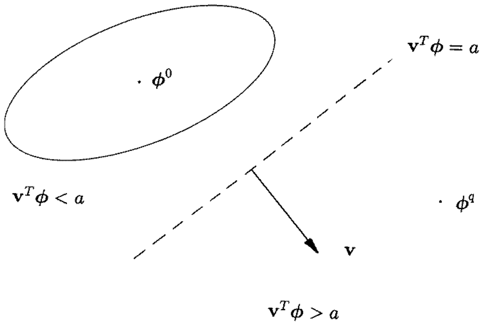

**Figure 8.** Sketch of the cutting plane $\boldsymbol{v}^T\phi = a$ in relation to the original tabulation point $\phi^0$ and the point to be added $\phi^q$.

In step (i), ideally one would like to find the leaf $\phi^0$ that is closest to $\phi^q$ in the sense that the error in the linear approximation is minimized. However, it is computationally expensive to find this leaf. The binary tree search used instead is computationally inexpensive, and yields a leaf $\phi^0$ that is likely to be close to $\phi^q$.

The growth process in step (v) is described in section 4.4 and sketched in figure 5.

The cutting plane (see figure 8) used in step (vi) is most easily defined in a transformed space. Consider the linear transformation that maps the EOA to the unit hypersphere. In the space defined by this transformation, the cutting plane is the perpendicular bisector of the line between $\phi^0$ and $\phi^q$.

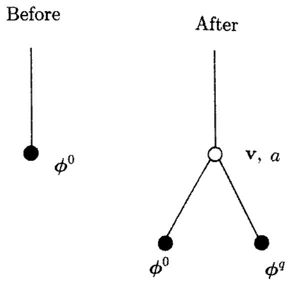

**Figure 9.** Sketch of part of the binary tree before and after the addition of the record at $\phi^q$.

## 5. Test results

Extensive tests have been performed to determine the performance of the ISAT method. The test case (described in section 5.1) is a pairwise mixing stirred reactor (PMSR), which corresponds to a zero-dimensional PDF calculation. In section 5.2, results are presented demonstrating that both local and global errors are controlled by the error tolerance $\varepsilon_{\text{tol}}$. The performance of the method in terms of table size, speed-up compared to direct integration, and other characteristics is described in section 5.3.

### *5.1. Pairwise mixing stirred reaction (PMSR)*

The partially-stirred reactor (PaSR, Correa and Braaten 1993) used previously to test implementations of combustion chemistry has the undesirable property that (in the steady state), the accessed region is a one-dimensional manifold: the composition of each particle is a unique function of its residence time. The PMSR, on the other hand, is designed to yield a much larger accessed region, and hence provides a more stringent test.

At any time $t$, the PMSR consists of an even number $N$ of particles, the $i$th particle having composition $\phi^{(i)}(t)$. With $\Delta t$ being the specified time step, at the discrete times $k\Delta t$ ($k$ integer) events occur corresponding to *outflow*, *inflow* and *pairing*, which can cause $\phi^{(i)}(t)$ to change discontinuously. Between these discrete times, the composition evolves by a *mixing* fractional step and a *reaction* fractional step.

The particles are arranged in pairs: particles 1 and 2, 3 and 4, ..., $N-1$ and $N$ are partners. The mixing fractional step consists of pairs ($p$ and $q$, say) evolving by

$$\frac{\mathrm{d}\phi^{(p)}}{\mathrm{d}t} = -(\phi^{(p)} - \phi^{(q)})/\tau_{\text{mix}}, \tag{39}$$

$$\frac{\mathrm{d}\phi^{(q)}}{\mathrm{d}t} = -(\phi^{(q)} - \phi^{(p)})/\tau_{\text{mix}}, \tag{40}$$

where $\tau_{\text{mix}}$ is the specified mixing timescale.

In the reaction fraction step, each particle evolves by the reaction equation

$$\frac{\mathrm{d}\phi^{(i)}}{\mathrm{d}t} = \boldsymbol{S}(\phi^{(i)}). \tag{41}$$

With $\tau_{\text{res}}$ being the specified residence time, outflow and inflow consist of selecting $\frac{1}{2}N\Delta t/\tau_{\text{res}}$ pairs at random and replacing their compositions with inflow compositions, which are drawn from a specified distribution.

With $\tau_{\text{pair}}$ being the specified pairing timescale, $\frac{1}{2}N\Delta t/\tau_{\text{pair}}$ pairs of particles (other than the inflowing particles) are randomly selected for pairing. Then these particles and the inflowing particles are randomly shuffled so that (most likely) they change partners.

The values of the parameters used in the test calculation are given in table 1. There are three inflowing streams: air (79% $N_2$, 21% $O_2$) at 300 K; methane at 300 K; and a pilot stream consisting of an equilibrium, stoichiometric fuel/air mixture at 2376 K. The mass flow rates of these streams are in the ratio 0.85:0.05:0.1. Initially $(t = 0)$, all particles are set to the pilot-stream composition. The pressure is atmospheric throughout.

**Table 1.** Parameters used in the PMSR tests.

| | | |
|---|---|---|
| Number of particles | $N$ | 100 |
| Time step | $\Delta t$ | 0.1 ms |
| Residence time | $\tau_{\text{res}}$ | 10 ms |
| Mixing timescale | $\tau_{\text{mix}}$ | 1 ms |
| Pairing timescale | $\tau_{\text{pair}}$ | 1 ms |

The chemical mechanism used is the skeletal mechanism for methane–air combustion reported by Yang and Pope (1996a). It consists of 4 elements, 16 species and 41 reactions. There are three linear dependences between the species mass fractions, but the enthalpy is not linearly dependent on the species. Thus there are 14 degrees of freedom in the description of the thermochemistry (i.e. $D = 14$).

To illustrate the performance of the PMSR, figure 10 shows the evolution of the mean temperature (i.e. the arithmetic mean of the particle temperatures). During the first one or two residence times (10–20 ms), the mean temperature drops from the initial value of 2376 K to fluctuate around 1600 K. For longer times, the mean temperature is statistically stationary, but, because of the random selection of particles, the significant fluctuations persist.

Figure 11 shows the temperature of the first two particles $T^{(1)}(t)$ and $T^{(2)}(t)$. Continuous evolution may be observed, punctuated by discontinuities corresponding to inflow/outflow and pairing.

### 5.2. Control of errors

The first test reported examines the control of the local error $\varepsilon$. The PMSR simulation is performed to a time of $t = 50$ ms, corresponding to 500 time steps and 50 000 queries. For each query $\phi^q$, the ISAT method is used to determine the mapping, and in addition (for these tests only) the exact result $\boldsymbol{R}(\phi^q)$ is obtained by direct integration so that the local error $\varepsilon$ can be measured directly. For this test, as functions of time, figure 12 shows: the error tolerance (specified to be $\varepsilon_{\text{tol}} = 0.0008$); the average (over all queries to date) of the local error incurred; the maximum local error incurred to date; and the fraction of queries to date for which the error exceeds the tolerance. The main conclusions are that the error is within the error tolerance for over 99% of the 50 000 queries, and that the largest error incurred is about $2\frac{1}{2}$ times the tolerance. Thus the algorithm based on ellipsoids of accuracy (EOA) to control the local error is quite satisfactory.

The global error $\varepsilon_G$ over the $K = 500$ time step is defined by

$$\varepsilon_G \equiv \frac{1}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N} |\boldsymbol{B}(\phi^{(n)}[k\Delta t] - \phi_{\text{DI}}^{(n)}[k\Delta t])|, \tag{42}$$

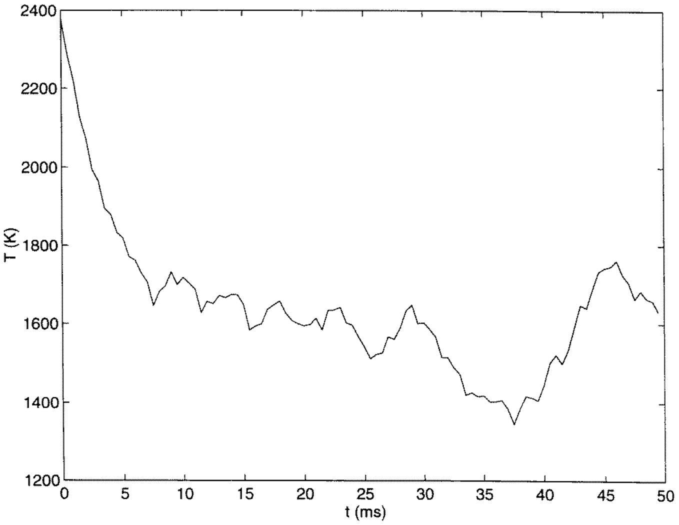

**Figure 10.** Mean temperature against time for the PMSR test.

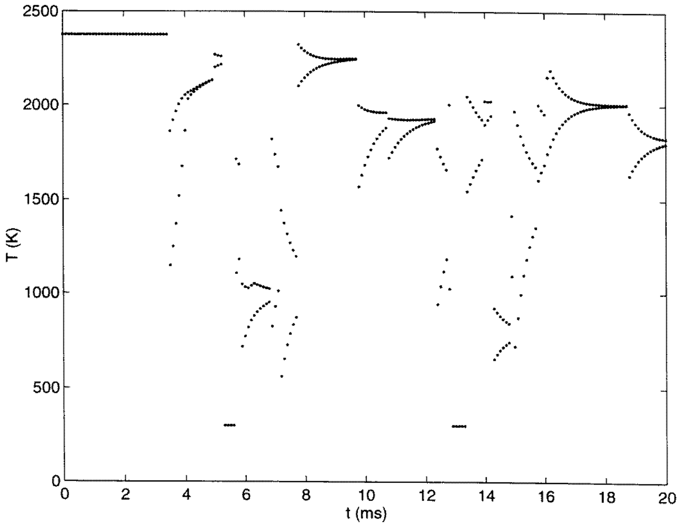

**Figure 11.** Temperatures of particles 1 and 2 for the first 200 time steps.

where $\phi^{(n)}(k\Delta t)$ is the composition of the $n$th particle on the $k$th time step, and $\phi^{(n)}_{\mathrm{DI}}(k\Delta t)$ is the corresponding value obtained when all the mappings are obtained by direct integration. Recall that $\boldsymbol{B}$ is a matrix used to scale the compositions appropriately to measure the error. Figure 13 shows the global errors measured for different specifications of the error tolerance $\varepsilon_{\mathrm{tol}}$. Clearly the algorithm to control the local error is also successful in controlling the global error.

To illustrate the levels of error, figure 14 (for the major species $CO_2$) and figure 15 (for

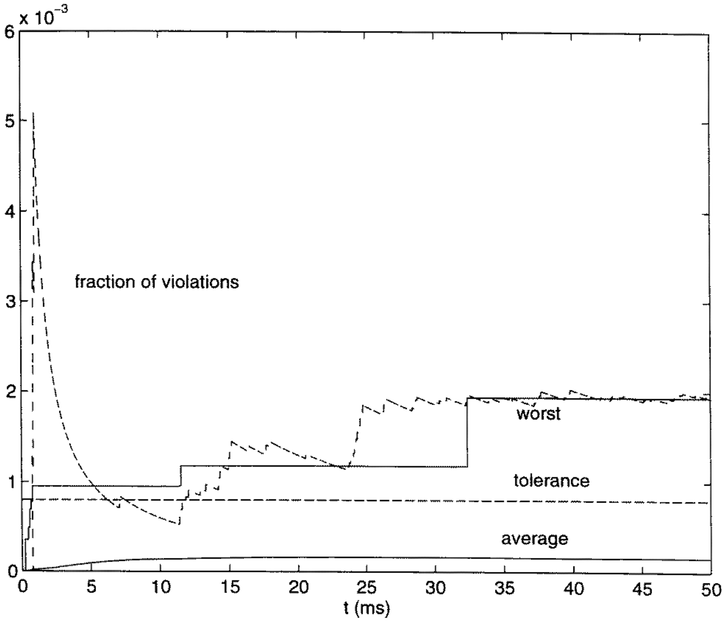

**Figure 12.** Test of local error control. Plotted against time: the error tolerance $\varepsilon_{\text{tol}} = 0.0008$; the average error incurred; the worst error made to date; the fraction of queries resulting in a tolerance violation, $\varepsilon > \varepsilon_{\text{tol}}$.

the minor species H) contrast the calculated composition of the first particle obtained from direct integration and from the ISAT method with different specified error tolerances.

### 5.3. Performance of ISAT

The ISAT method is intended to be very efficient when the total number of queries $Q$ is large—of order $10^9$, say. To study the performance of the method for large $Q$, the PMSR calculation is performed for over 2 million time steps to yield over $2 \times 10^8$ queries. The error tolerance is specified to be $\varepsilon_{\text{tol}} = 0.0128$.

Figure 16 shows the cumulative CPU time $T_h$ (in hours) as a function of the number of queries. Initially the curve rises steeply, but before $Q = 10^8$ it reaches the straight line

$$T_h = 1.41 + Q/(19 \times 10^6). \qquad (43)$$

In the initial region, a significant amount of CPU time is used in adding to the table ($A$) and in growing the EOAs ($G$). Roughly, (43) indicates that these operations ($A$ and $G$) together consume about $1\frac{1}{2}$ CPU hours. For $Q$ greater than $10^8$, essentially all the CPU time is used for retrieves ($R$), and the asymptotic rate of $19 \times 10^6$ queries per CPU hour is achieved.

Evidently, using ISAT, $10^9$ queries require 54 hours CPU time, compared to over 6 years using direct integration—a speed-up of around $10^3$.

Figure 17 is a log–log plot of the speed-up factor against the number of queries $Q$. This

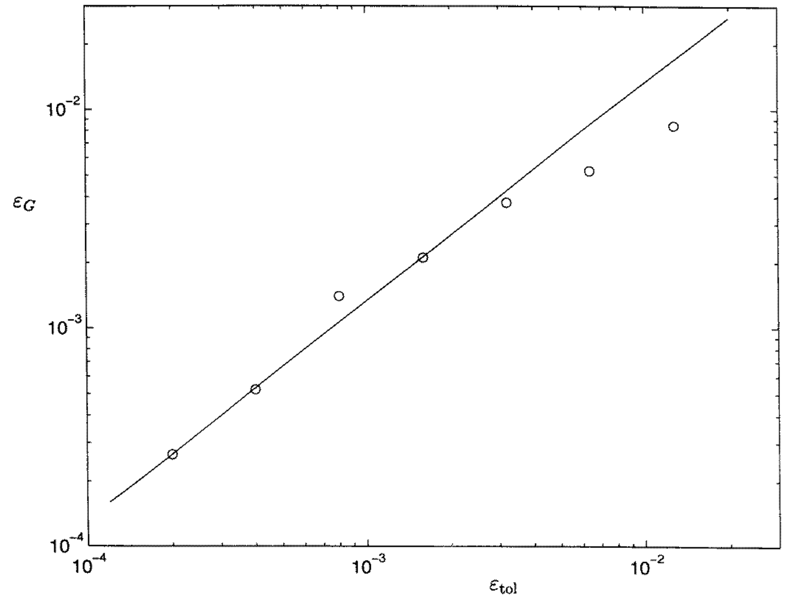

**Figure 13.** Global error $\varepsilon_G$ as a function of the specified local error tolerance $\varepsilon_{\text{tol}}$. A line of unit slope is shown for reference.

factor is defined as

$$\text{speed-up factor} = \frac{\text{CPU time for } Q \text{ queries using DI}}{\text{CPU time for } Q \text{ queries using ISAT}}. \tag{44}$$

(The numerator is evaluated as $Q$ times 0.187s, which, based on 50 000 queries, is the average time for one direct integration.) It may be observed that the speed-up is modest (about 10) until $Q$ exceeds $10^6$, because of the additions and growths that are performed.

The cumulative number of additions $N_A$ and growths $N_G$ performed is shown in figure 18. These curves rise very steeply as the table is initially formed, and then they rise much more slowly. It is interesting that there is no sign of their reaching asymptotes. Instead, their continual growth is well represented by power laws:

$$N_A \sim Q^{0.18} \text{ and } N_G \sim Q^{0.2}. \tag{45}$$

It is useful to consider the CPU times used by each of the three processes—addition ($A$), growth ($G$), and retrieve ($R$). The average CPU times for each of these processes are denoted by $T_A$, $T_G$ and $T_R$, respectively. It is found that these are in the ratio 10:1:$10^{-3}$ (with $T_G = 0.187$ s). Regarding the outcome of a query ($A$, $G$ or $R$) as random, the probabilities $P_A$, $P_G$ and $P_R$ can be ascribed to each (with $P_A + P_G + P_R = 1$). Clearly these probabilities change as the calculation proceeds: according to (45), $P_A$ and $P_G$ decrease as

$$P_A \sim Q^{-0.82} \text{ and } P_G \sim Q^{-0.80}, \tag{46}$$

and hence $P_R$ tends to unity. The expected CPU time $\bar{T}$ is then

$$\bar{T} = P_R T_R + P_A T_A + P_G T_G$$

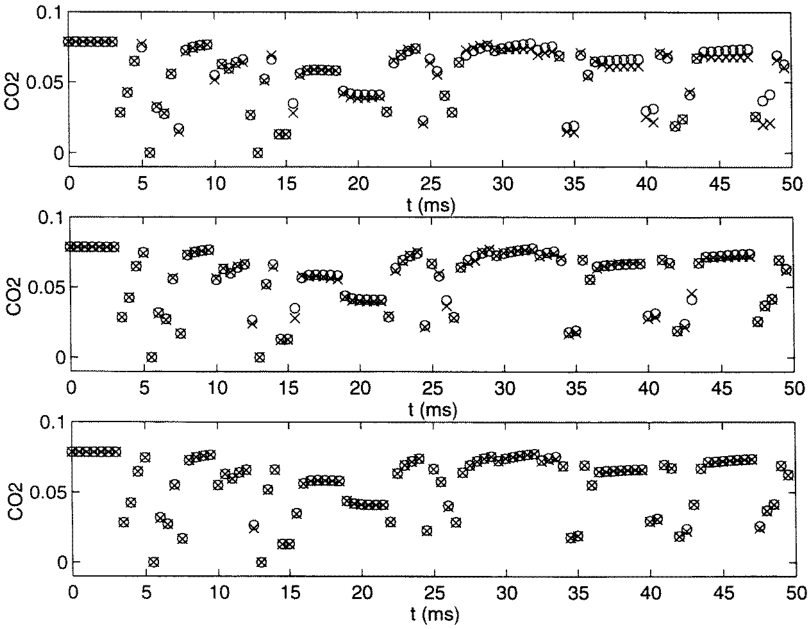

**Figure 14.** The effect of the error tolerance on the calculation of major species. Mole fraction of CO$_2$ for the first particle (shown on every fifth time step) in the PMSR test: ○, direct integration; ×, ISAT with error tolerances (from top to bottom) $\varepsilon = 0.0128$, 0.0032 and 0.0008.

$$
\begin{aligned}
&= T_R \left\{ P_R + \frac{P_A}{T_R/T_A} + \frac{P_G}{T_R/T_G} \right\} \\
&\approx T_R \left\{ 1 + \frac{P_A}{10^{-4}} + \frac{P_G}{10^{-3}} \right\}, \qquad (47)
\end{aligned}
$$

where the last line pertains to large $Q$ and the present ratios of $T_A$, $T_G$ and $T_R$.

The desired asymptotic behaviour is for every query to be a retrieve, so that $\bar{T}$ is as small as possible, i.e. $\bar{T} = T_R$. The important observation to be made from (47) is that to approach this asymptote it is not necessary for $P_A$ and $P_G$ to be zero: all that is required is

$$P_A \ll T_R/T_A \text{ and } P_G \ll T_R/T_G. \qquad (48)$$

Indeed, once $P_A$ is small compared to $T_R/T_A$, there is no performance penalty in replacing additions by direct integrations, so that the table no longer increases in size.

Figure 19 provides another view of the distribution of CPU time between the three processes. As a function of the number of queries $Q$, the figure shows the fraction of the cumulative CPU time devoted to each process. As expected, early on, additions dominate as the table grows rapidly. But by $Q = 10^5$, growth becomes the most time-consuming process. Eventually retrievals dominate.

The time taken to traverse the binary tree is proportional to the depth of the leaf, i.e. the number of nodes that are encountered in the traverse. As a function of the number of queries, figure 20 shows the maximum depth in the tree, and the average depth—the arithmetic mean over all traverses to date. Evidently the tree is well-behaved with the average depth being less than 10.

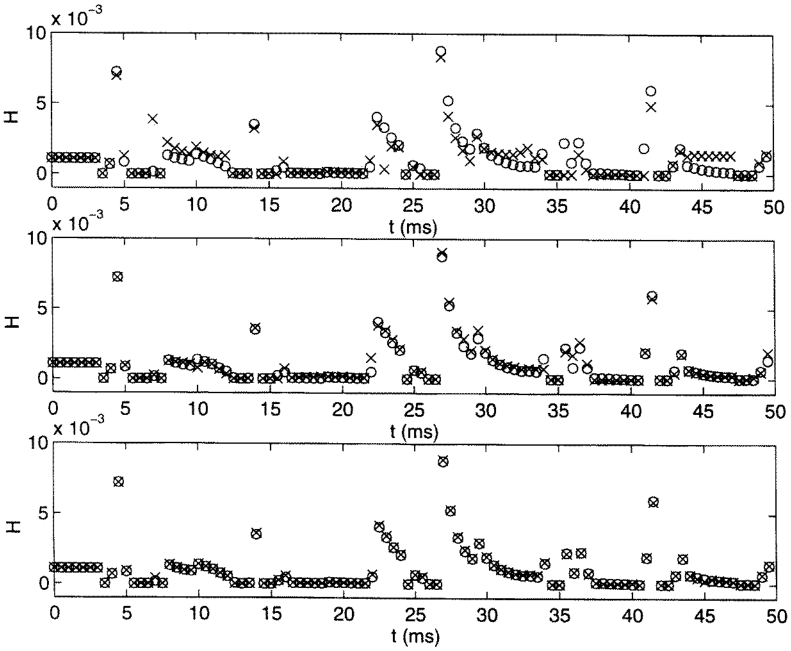

**Figure 15.** The effect of the error tolerance on the calculation of minor species. Mole fraction of H for the first particle (shown on every fifth time step) in the PMSR test: ○, direct integration; ×, ISAT with error tolerances (from top to bottom) $\varepsilon = 0.0128$, 0.0032 and 0.0008.

## 6. Discussion and conclusion

An *in situ* adaptive tabulation (ISAT) procedure has been developed and demonstrated as a computationally efficient method for implementing chemical reactions in reactive flow calculations. For the test case considered here (involving a skeletal mechanism for methane combustion with 14 degrees of freedom) a speed-up of a factor of 1000 is achieved compared to directly integrating the reaction equations numerically.

The essential ideas and ingredients of the approach are as follows:

(i) A splitting is employed so that the reaction problem is reduced to determining the mapping $\boldsymbol{R}(\phi)$, which is the solution to the reaction equation (7) after a time $\Delta t$ from the initial condition $\phi$.

(ii) A table is built up *in situ*, as the reactive flow calculation is performed, so that only the accessed region of the composition space is tabulated.

(iii) A table entry (or record) consists of: a composition $\phi^0$; the mapping $\boldsymbol{R}(\phi^0)$; the gradient of the mapping $\boldsymbol{A}(\phi^0)$; and the specification of an ellipsoid of accuracy (EOA) within which the linear approximation

$$\boldsymbol{R}(\phi^q) \approx \boldsymbol{R}(\phi^0) + \boldsymbol{A}(\phi^0)(\phi^q - \phi^0), \tag{49}$$

is accurate.

(iv) The EOA is initialized and grown to ensure that (with high probability) the error involved in (49) is within a specified tolerance.

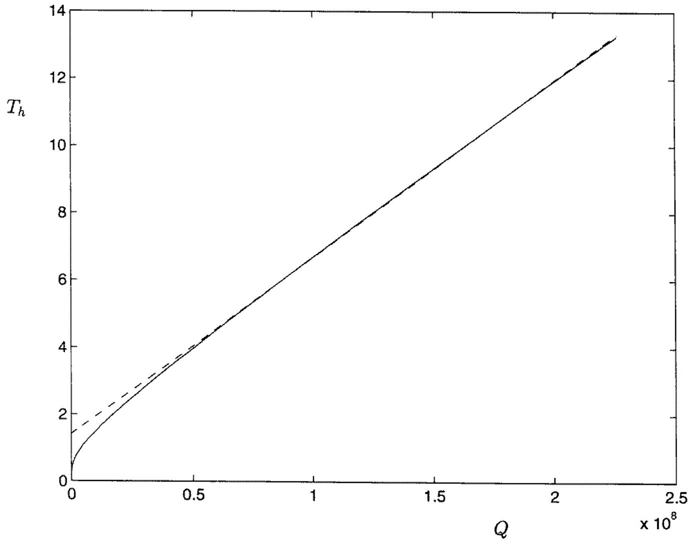

**Figure 16.** CPU time in hours $T_h$ against number of queries $Q$. The dashed line is $T_h = 1.41 + Q/[19 \times 10^6]$.

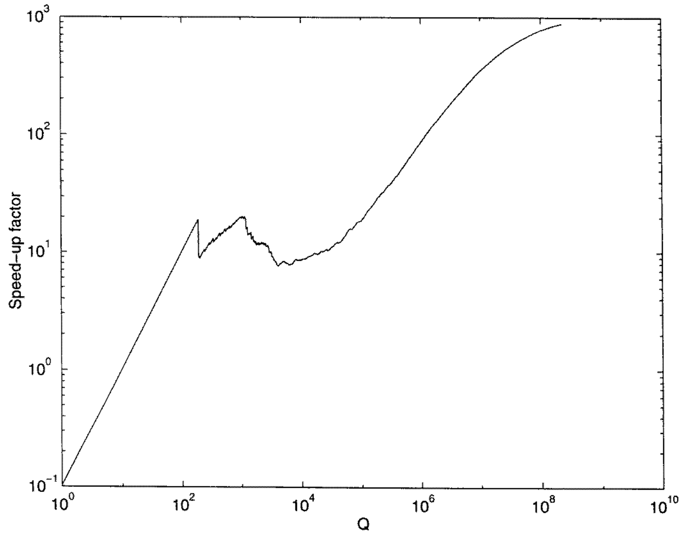

**Figure 17.** Speed-up factor (relative to direct integration) against number of queries $Q$.

(v) The records are stored in a binary tree which, given a query composition $\phi^q$, can be traversed to obtain a table entry $\phi^0$ which in some sense is close to $\phi^q$.

(vi) As the calculation proceeds, with increasing probability, the query composition $\phi^q$ lies within the EOA of a table entry $\phi^0$, so that the mapping is accurately and efficiently retrieved via (49).

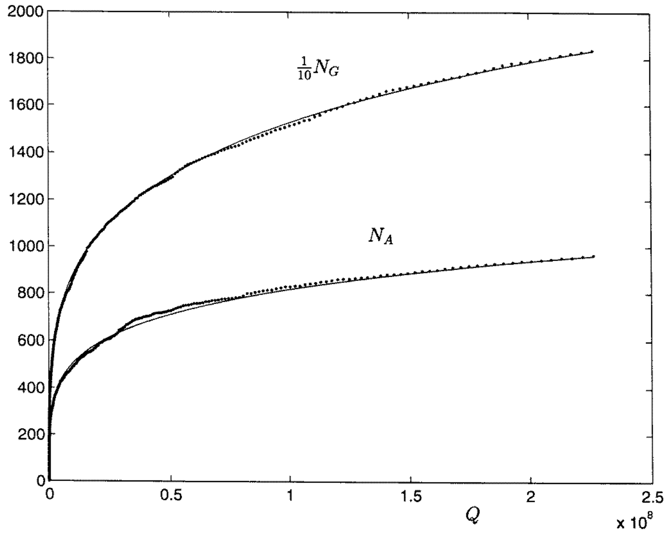

**Figure 18.** Number of additions $N_A$ and number of growths $N_G$ against number of queries. The lines are empirical power-law fits: $N_A = 33.3Q^{0.18} - 100$, and $N_G = 439Q^{0.20} - 2215$.

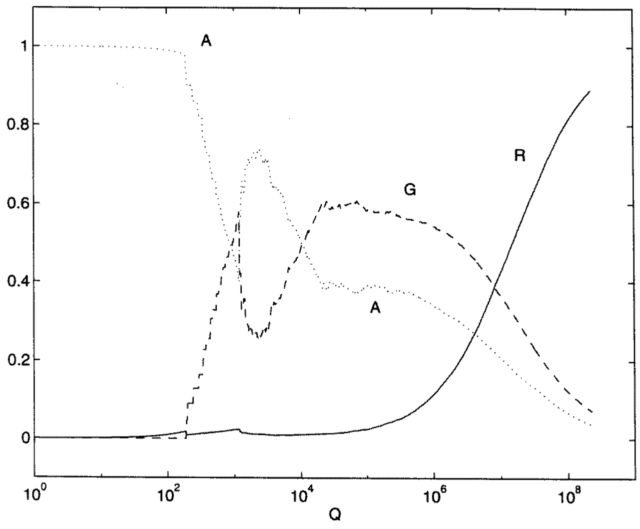

**Figure 19.** Fraction of total CPU time used for additions ($A$), growths ($G$) and retrieves ($R$) against number of queries $Q$.

A different *in situ* tabulation method is described by Yang and Pope (1996b). That approach uses the constant (rather than linear) approximation, and performs the tabulation in cells of fixed size, so that there is no adaptive control of the error. For a given error tolerance, the work and storage requirements scale linearly with the number of degrees of

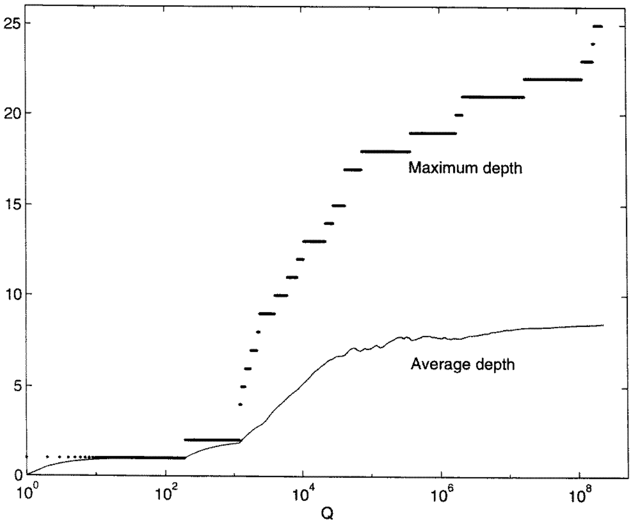

**Figure 20.** Maximum and average depth of leaves in the binary tree against number of queries $Q$.

freedom $D$, whereas in the current ISAT method the work and storage scale as $D^2$. A simple analysis shows that the total storage scales as

$$D^p \left( \frac{1}{\varepsilon_{\mathrm{tol}}} \right)^{\tilde{D}/p},$$

where $p$ is the order of the method ($p = 1$ for the constant approximation and $p = 2$ for the linear approximation), and $\tilde{D}$ is the effective dimension of the accessed region. It is evident, therefore, that for an accurate tabulation ($\varepsilon_{\mathrm{tol}} \to 0$), the current method using the linear approximation ($p = 2$) requires less storage overall.

In the present ISAT method, the order $D^2$ storage requirement stems equally from the linear approximation (i.e. the mapping gradient matrix $\boldsymbol{A}$) and from the use of EOAs to control tabulation errors (i.e. the unitary matrix $\boldsymbol{Q}$). An alternative possibility is to use ISAT with the constant approximation—which halves the storage requirement per record. But even for large error tolerance, it is likely that this algorithm requires more than twice the number of records (for given accuracy, compared to the linear approximation) and so, in fact, has a greater storage requirement.

The speed-up of 1000 achieved by the ISAT method is sufficient to enable full PDF calculations of turbulent flames with skeletal or detailed kinetics (i.e. of order 20 degrees of freedom). The method can also be used with reduced kinetics, with comparable benefits (providing that the reduced mechanism yields a smooth mapping). Nevertheless, some improvements to the method are desirable and possible. Principally, it is desirable for the asymptotic speed-up to be achieved after fewer queries; and it is desirable for the tabulation to operate with fewer degrees of freedom than there are in the thermochemistry. These and other improvements are subjects of ongoing research.

## Acknowledgments

I am grateful to Gal Berkooz, Paul Chew and Bo Yang for comments and suggestions on various aspects of this work.

## References

Caracotsios M and Stewart W E 1985 *Comput. Chem. Eng.* **9** 359

Chen J-Y, Chang W-C and Koszykowski M 1995 *Combust. Sci. Technol.* **110–111** 505

Christo F C, Masri A R, Nebot E M and Pope S B 1996 *Proc. 26th Int. Symp. on Combustion* (Pittsburgh, PA: The Combustion Institute) in press

Correa S M and Braaten M E 1993 *Combust. Flame* **94** 469

Keck J C and Gillespie D 1971 *Combust. Flame* **17** 237

Lam S H and Goussis D A 1988 *Proc. 22nd Int. Symp. on Combustion* (Pittsburgh, PA: The Combustion Institute) p 931

Maas U A and Pope S B 1992a *Combust. Flame* **88** 239

——1992b *Proc. 24th Int. Symp. on Combustion* (Pittsburgh, PA: The Combustion Institute) p 103

——1994 *Proc. 25th Int. Symp. on Combustion* (Pittsburgh, PA: The Combustion Institute) p 1349

Pope S B 1985 *Prog. Energy Combust. Sci.* **11** 119

Smooke M D, Mitchell R E and Keyes D E 1989 *Combust. Sci. Technol.* **67** 85

Smooke M D (ed) 1991 *Reduced Kinetic Mechanisms and Asymptotic Approximations for Methane–Air Flames* (Lecture Notes in Physics **384**) (Berlin: Springer)

Taing S, Masri A R and Pope S B 1993 *Combust. Flame* **95** 133

Turanyi T 1994 *Comput. Chem.* **18** 45

Yang B and Pope S B 1996a An investigation of the accuracy of manifold methods and splitting schemes in the computational implementation of combustion chemistry, submitted for publication

——1996b Development and implementation of accurate and efficient combustion chemistry for gas turbine combustor simulations, submitted for publication