Journal of Computational Physics **163,** 216–248 (2000)
doi:10.1006/jcph.2000.6572, available online at http://www.idealibrary.com on IDEAL®

The Random Projection Method for Hyperbolic Conservation Laws with Stiff Reaction Terms

Weizhu Bao[1] and Shi Jin[2]

School of Mathematics, Georgia Institute of Technology, Atlanta, Georgia 30332
E-mail: wbao@math.gatech.edu, jin@math.gatech.edu

Received June 9, 1999; revised May 17, 2000

We propose a random projection method for numerical simulations of hyperbolic conservation laws with stiff source terms arising from chemically reactive flows:

$$U_t + F(U)_x + G(U)_y = \frac{1}{\varepsilon}\Psi(U).$$

In this problem, the chemical time scales may be orders of magnitude faster than the fluid dynamical time scales, making the problem numerically stiff. A classic spurious numerical phenomenon, the incorrect propagation speeds of discontinuities, occurs in underresolved numerical solutions. We introduce a random projection method for the reaction term by replacing the ignition temperature with a uniformly distributed random variable. The statistical average of this method corrects the spurious shock speed, as will be proved with a scalar model problem and demonstrated by a wide range of numerical examples in inviscid detonation waves in both one and two space dimensions.

1. INTRODUCTION

Hyperbolic systems with stiff source terms arise, among many other applications, in the modeling of chemically reactive flows. Restricting our attention to inviscid flows, we describe the underlying physical equations by the compressible Euler equations of gas dynamics, coupled with source terms representing the chemical reaction. In two space dimensions these equations take the form

$$U_t + F(U)_x + G(U)_y = \frac{1}{\varepsilon}\Psi(U), \tag{1.1}$$

[1] On leave from Department of Applied Mathematics, Tsinghua University, Beijing 100084, People's Republic of China.

[2] Research supported in part by NSF Grant DMS-9704957. Current address: Department of Mathematics, University of Wisconsin, Madison, Wisconsin 53706. E-mail: jin@math.wisc.edu.

0021-9991/00 $35.00

where U is the vector of dependent variables with components mass, momentum, total energy, and density or concentration for each species in the reacting mixture. The flux functions $F(U)$ and $G(U)$ describe the fluid dynamical convection, while the source term $\Psi(U)$ arises from the chemistry of the reacting species. ε is the reaction time.

The kinetics equations often include reactions with widely varying time scales. Moreover, many of the chemical time scales may be orders of magnitude faster than the fluid dynamical time scales. This can lead to problems of numerical stiffness. Even a stable (for example, with an implicit source) scheme could lead to spurious numerical propagation speed when the reaction time scale is not properly resolved numerically. This phenomenon was first observed by Colella *et al.* [7], who made a study of the limiting behavior with increasing stiffness for various model systems. In particular, they looked at the Euler equations coupled with a single chemical variable representing the mass fraction of unburnt gas in a detonation wave. These waves have the structure of a fluid dynamic shock that raises the pressure to some peak value, followed immediately by a reaction zone that brings the pressure back down to a new equilibrium value. On coarse grids it is not possible to resolve this combustion spike and the best one can hope for is a single discontinuity linking the two equilibrium values and moving at the correct speed. They obtained the correct combustion spike and correct speed of discontinuities only with very fine grids; i.e., the reaction zone was completely resolved (at least 30 grid points in the reaction zone). In contrast, they observed that on coarse grids (i.e., when the reaction zone was not resolved) the numerical solutions were qualitatively incorrect. The computed solution consisted of a weak detonation wave, in which all the chemical energy was released, followed by fluid dynamic shock traveling more slowly. The numerical reaction wave always traveled much faster than the physical one, and the speed becomes one grid point per time step when the ratio of the time step over the reaction time is very large. Ben-Artzi [1] observed the same phenomenon in solving reactive flows using the solution of the generalized Riemann problem. Using scalar models, LeVeque and Yee [16] studied this spurious numerical phenomenon. They showed that this phenomenon is due to the introduction of nonequilibrium values through numerical dissipation in the advection step, which triggers the reaction too early. See also [18].

Since the numerical viscosity is the cause of the incorrect speed, a natural idea is to avoid any numerical viscosity. In [5] Chorin introduced the random choice method for reacting flows, which originated from the classical Glimm scheme [13] for hyperbolic systems. Since the random choice method does not introduce numerical viscosity, no spurious waves will occur. In [11], Engquist and Sjogreen proposed a temperature extrapolation method, which uses the extrapolated temperature value from outside the shock profile, and obtained the correct detonation speed.

Since numerical viscosity is an essential feature for modern high-resolution shock-capturing methods, it is highly desirable to develop methods for reacting flows that, instead of avoiding the numerical viscosity, make correct use of it. The random projection method proposed here is such a method. Our method consists of the typical two steps: solving the homogeneous hyperbolic conservation laws by a standard modern shock-capturing method and then performing a projection step for the stiff reaction term. In the underresolved regimes, where the numerical time step is much greater than the reacting time, all ODE solvers for the reaction term essentially reduce to a projection operator (which will be called the **deterministic projection** in this paper), which projects the chemically nonequilibrium data into the equilibrium ones according to the value of the temperature relative to the ignition temperature. It is this projection that causes the incorrect speed because the numerically

smeared nonequilibrium temperature, once above the ignition temperature, will trigger the chemical reaction too early, forcing the front to move. Unless the ignition temperature is suffciently high this will always happen [2, 18]. This is purely a grid effect. Since there is no correlation between the numerical shock location and the grid, a natural and robust way to handle this numerical difficulty is through a random method. Our idea is, in the projection step, to **replace the ignition temperature with uniformly distributed random variable**, defined in a suitable domain. Upon a suitable choice of the sampling sequence, at each time step, the front either moves one grid point or does not move, but the statistical average yields the correct speed!

For model scalar conservation laws with stiff source terms, we prove that the first-order random projection method can indeed capture the correct speed of discontinuity (i.e., obtain the correct jumps in the correct locations) with first-order accuracy.

The random projection method clearly differs from the random choice method of Chorin [5]. The random choice method is a Godunov type method that uses the Riemann or even the generalized Riemann (for hyperbolic systems with source terms) solver. The random projection method uses the randomness only for the reaction term, while in the convection step, any modern shock capturing methods, including the efficient methods free of Riemann solvers and local characteristic decompositions, can be used.

The paper is organized as follows. In Section 2 we first introduce the random projection method for a scalar model problem. We prove that this method, when the first order upwind or Lax–Friedrichs method is used for the nonlinear convection, yields the correct shock speed with an error of $O(h|\ln h|)$, where h is the grid size. We also present numerical examples for this scalar model. Furthermore, we apply this method to problems with the stiff Ricatti source. In Section 3 we extend this method to the computation of two-dimensional scalar hyperbolic conservation laws with stiff source terms. Numerical results show that the random projection method yields the correct speed of reaction front. In Section 4 we extend this method to the computation of one-dimensional detonation waves. Numerical examples for a variety of detonation waves are presented. In Section 5 we extend this method to the computation of two-dimensional detonation waves. In Section 6 we draw some conclusions.

2. ONE-DIMENSIONAL SCALAR PROBLEMS

We first introduce the random projection method for a model problem, studied in [12, 16]. Consider the hyperbolic conservation law with stiff source term

$$u_t + f(u)_x = \frac{1}{\varepsilon}(u-\alpha)(1-u^2), \qquad -1 < \alpha < 1, \tag{2.1}$$

with piecewise constant initial data

$$u(x,0) = u_0(x) = \begin{cases} 1, & x \le x_0, \\ -1, & x > x_0. \end{cases} \tag{2.2}$$

Here ε is the reaction time; f is a convex function of u, i.e., $f''(u) > 0$; and x_0 is a given point.

The source term in (2.1) admits three local equilibria, the unstable one $u = \alpha$ and the stable ones $u = \pm 1$. When the solution is at equilibrium, the reaction term has no effect. Thus

the exact solution is a shock discontinuity connecting $u = 1$ with $u = -1$ and propagating to the right with a speed determined by the Rankine–Hugoniot jump condition

$$s = \frac{1}{2}[f(1) - f(-1)]. \tag{2.3}$$

Namely,

$$u(x,t) = \begin{cases} 1, & \text{if } x \le x_0 + st, \\ -1, & \text{if } x > x_0 + st. \end{cases} \tag{2.4}$$

Note that the speed does not depend on the specific value of α. This is the key to the success of the random projection method, which we introduce next.

Let h be the grid size and let k be the time step. The numerical solution is evaluated at the points (ih, nk), $i = 0, \pm 1, \pm 2, \ldots$, $n = 0, 1, 2, \ldots$. Let u_i^n approximate $u(ih, nk)$ and let u^n be the solution vector of $u(\cdot, nk)$ at time $t_n = nk$. When the reaction term is resolved, i.e., $k = O(h) \ll \varepsilon$, any method that works well for the homogeneous hyperbolic conservation law still works well here. Here we are interested in the underresolved case, where $k = O(h) \ge \varepsilon$.

A standard numerical method, which allows an underresolved discretization, is the fractional step method that solves the homogeneous convection

$$u_t + f(u)_x = 0 \tag{2.5}$$

followed by the reaction step

$$u_t = \frac{1}{\varepsilon}(u - \alpha)(1 - u^2). \tag{2.6}$$

Let $S_c(k)$ denote the discrete operator for the homogeneous system (2.5) over a time step of duration k, and let $S_r(k)$ be the numerical integrator for the ODE system (2.6). Then the fraction step method takes the form

$$u^{n+1} = S_r(k)\, S_c(k)\, u^n. \tag{2.7}$$

One may use any high-resolution shock capturing method for $S_c(k)$. Let $u^* = S_c(k)u^n$. When $k \gg \varepsilon$, the solution of the ODE (2.6) approaches the equilibrium states ± 1 exponentially fast. Whether it approaches 1 or -1 depends on whether u is bigger or smaller than α [12]. Thus $S_r(k)$ becomes effectively the **deterministic projection** operator:

$$S_r(k): \qquad u_j^{n+1} = \begin{matrix} 1, & \text{if } u_j^* > \alpha \\ -1, & \text{if } u_j^* \le \alpha \end{matrix}, \quad \text{for all } j. \tag{2.8}$$

This standard method was studied in [16], where it was found that, if $k \gg \varepsilon$, the numerical shock moves one grid point per time step. This is the case when the smeared value u^* is above the critical value α, which will be projected into 1 by the projection step, forcing the shock to advance one grid point. If the smeared value u^* is below α, u will always be projected to -1, and then the shock will not move at all. Whether u^* is greater or less than α depends on the CFL number.

2.1. *The Random Projection Method*

The idea of the random projection method is to replace the unstable equilibrium, or the critical value, α, with a uniformly distributed random sequence $\theta_n \in (-1, 1)$. Let $u^* = S_c(k)u^n$. We replace S_r by S_θ, where

$$S_\theta(k): \qquad u_j^{n+1} = \begin{cases} 1, & \text{if } u_j^* > \theta_n \\ -1, & \text{if } u_j^* \le \theta_n \end{cases}, \quad \text{for all } j. \tag{2.9}$$

In this method, one random value of $\{\theta_n \mid n = 0, 1, 2, \ldots\}$ will be selected per time step for all grid points.

This idea is based on the observation that there is no correlation between the center of the shock and the grid. One hopes that the statistical average will automatically correct the wrong speed.

2.2. *The Choice of the Random Sequence θ_n*

In [9], several sampling schemes were proposed and compared for the random choice method of Chorin [4, 5]. Here and in our practical computations, we always use van der Corput's sampling scheme. The merit of this scheme is that it produces an equidistributed sequence on the interval [0, 1], and among all known uniformly distributed sequences the deviation of van der Corput's sequence is minimal [14]. The detailed scheme is the following. Let $1 \le n = \sum_{k=0}^{m} i_k 2^k$, $i_k = 0, 1$, be the binary expansion of the integer n. Then the van der Corput sequence, with range in [0, 1], is given by

$$\vartheta_n = \sum_{k=0}^{m} i_k 2^{-(k+1)}, \qquad n = 1, 2, \ldots. \tag{2.10}$$

We rescale it in order to get a sequence θ_n on $[-1, 1]$:

$$\theta_n = 2\vartheta_n - 1, \qquad n = 1, 2, \ldots. \tag{2.11}$$

In order to get the error estimate of our random projection method based on the van der Corput sequence, we recall some properties of the van der Corput sampling sequence $\{\vartheta_n, n = 1, 2, \ldots\}$ [14]. It is rather illuminating to list its first few elements:

$$\vartheta_1 = \frac{1}{2}, \quad \vartheta_2 = \frac{1}{4}, \quad \vartheta_3 = \frac{3}{4}, \quad \vartheta_4 = \frac{1}{8},$$
$$\vartheta_5 = \frac{5}{8}, \quad \vartheta_6 = \frac{3}{8}, \quad \vartheta_7 = \frac{7}{8}, \quad \vartheta_8 = \frac{1}{16},$$

One can see that

$$\vartheta_1 = \frac{1}{2}, \qquad \vartheta_n = \begin{cases} < \frac{1}{2}, & \text{if } n \text{ is even,} \\ > \frac{1}{2}, & \text{if } n > 1 \text{ is odd.} \end{cases}$$

In general, if one divides the unit interval into the subintervals $(r2^{-s}, (r+1)2^{-s})$, $r = 0, 1, \ldots, 2^s - 1$, then for each r there is exactly one n, $2^s - 1 < n \le 2^{s+1} - 1$, such that

$\vartheta_n \in [r2^{-s}, (r+1)2^{-s})$. Let

$$N\{j = n_1 + 1, \ldots, n_2; \vartheta_j \in I\} \tag{2.12}$$

denote the number of j, $n_1 < j \le n_2$, such that ϑ_j is contained in an interval $I \subset [0, 1]$. Let

$$\delta(\boldsymbol{\vartheta}, n_1, n_2, I) \equiv \frac{1}{n_2 - n_1} N\{j = n_1 + 1, \ldots, n_2; \vartheta_j \in I\} - |I| \tag{2.13}$$

be the difference between the proportion of times that ϑ_j is contained in $I \subset [0, 1]$ and let $|I|$ be the length of I. The following result [14] is rather useful in our error estimate in the next section.

LEMMA 2.1. *For the binary van der Corput sampling sequence*,

$$\delta(\boldsymbol{\vartheta}, n_1, n_2, I) \le \frac{3\ln(2(n_2 - n_1)) + 1}{n_2 - n_1}, \qquad \forall\, n_1, n_2, I. \tag{2.14}$$

The above inequality shows that the binary van der Corput sampling sequence is equidistributed on the interval [0, 1] since $\lim_{n_2 \to \infty} \delta(\boldsymbol{\vartheta}, n_1, n_2, I) = 0$ for each fixed n_1, I. Thus the sequence $\{\theta_n \mid n = 1, 2, \ldots\}$ is equidistributed on the interval $[-1, 1]$.

Remark 2.1. Of course, the van der Corput sampling sequence is not random (it is actually pseudo-random [14]). Since this method has the random spirit and is originated from the classical random choice method, we still call the method a random projection method. It is an interesting project to investigate the behavior of other random or pseudo-random sequences for this method.

2.3. *An Error Estimate on the Numerical Shock Speed*

Now we will prove that the random projection method, when combined with the first-order upwind method or the Lax–Friedrichs method for the convection term, can capture the correct location of discontinuities for the scalar model problem (2.1) with first-order accuracy. The proof and results are similar to the error estimate of Glimm's scheme for the homogeneous conservation laws [6].

First we consider the upwind method. Without loss of generality, we assume $f'(u) > 0$. Consider the first-order upwind method for the convection, followed by the random projection:

$$S_c(k): \qquad u_j^* = u_j^n - \frac{k}{h}\left(f\left(u_j^n\right) - f\left(u_{j-1}^n\right)\right), \tag{2.15}$$

$$S_\theta(k): \qquad u_j^{n+1} = \begin{cases} 1, & \text{if } u_j^* > \theta_n, \\ -1, & \text{if } u_j^* \le \theta_n. \end{cases} \tag{2.16}$$

It is easy to see that the above method preserves the monotonicity property of the upwind method. Under the usual CFL condition, i.e., $\sup_{j,n} |f'(u_j^n)|k/h < 1$, the above algorithm is stable.

First, at any time step n, there is an $l(n) = j_0$, j_0 an integer, such that

$$u_j^n = \begin{cases} 1, & \text{if } j \le l(n), \\ -1, & \text{if } j > l(n). \end{cases} \tag{2.17}$$

Here we assume that x_0, the initial point of discontinuity, is a grid point; i.e., $x_0 = l(0)h$. After the convection step (2.15), one has

$$u_j^* = \begin{cases} 1, & \text{if } j \le l(n), \\ \mu, & \text{if } j = l(n)+1, \\ -1, & \text{if } j > l(n)+1, \end{cases} \tag{2.18}$$

where

$$1 > \mu = \frac{k}{h}(f(1) - f(-1)) - 1 = \frac{2ks}{h} - 1 \equiv 2\lambda - 1 > -1 \tag{2.19}$$

with $\lambda = ks/h$. In the second step, the random projection (2.16) gives

$$u_j^{n+1} = \begin{cases} 1, & \text{if } j \le l(n), \\ -1, & \text{if } j > l(n)+1; \end{cases} \qquad u_{l(n)+1}^{n+1} = \begin{cases} 1, & \text{if } \mu > \theta_n, \\ -1, & \text{if } \mu \le \theta_n. \end{cases} \tag{2.20}$$

Thus one has

$$l(n+1) = \begin{cases} l(n)+1, & \text{if } \mu > \theta_n, \\ l(n), & \text{if } \mu \le \theta_n. \end{cases} \tag{2.21}$$

Therefore the discontinuities in the approximate solution at each time step either move one grid point to the right or do not move. We now examine the accumulative effect over many time steps. In fact, from (2.21), noting (2.19), (2.12), and (2.11), one gets

$$\begin{aligned} l(n) &= l(0) + N\{j = 1, \ldots, n; \theta_j \in [-1, \mu)\} \\ &= l(0) + N\{j = 1, \ldots, n; \vartheta_j \in [0, (\mu+1)/2)\} \\ &= l(0) + N\{j = 1, \ldots, n; \vartheta_j \in [0, \lambda)\}. \end{aligned} \tag{2.22}$$

Combining (2.22), (2.17), (2.4), (2.13), and (2.14) with $n_1 = 0$, $n_2 = n$, and $I = [0, \lambda)$, we obtain

$$\begin{aligned} |x_0 + st_n - l(n)h| &= |st_n - hN\{j = 1, \ldots, n; \vartheta_j \in [0, \lambda)\}| \\ &= |st_n - nh\{\lambda + \delta(\vartheta, 0, n, [0, \lambda))\}| \\ &= nh|\delta(\vartheta, 0, n, [0, \lambda))| \le h(1 + 3|\ln(2t_n/k)|) \\ &= h[1 + 3|\ln(2st_n/(\lambda h))|]. \end{aligned} \tag{2.23}$$

This gives the accuracy for the upwind method.

Next we consider the Lax–Friedrichs method followed by random projection:

$$S_c(k): \qquad u_j^* = u_j^n - \frac{k}{2h}\big(f(u_{j+1}^n)\big) - f(u_{j-1}^n) + \frac{\sqrt{ak}}{2h}(u_{j+1}^n - 2u_j^n + u_{j-1}^n), \quad (2.24)$$

$$S_\theta(k): \qquad u_j^{n+1} = \begin{cases} 1, & \text{if } u_j^* > \theta_n, \\ -1, & \text{if } u_j^* \le \theta_n. \end{cases} \quad (2.25)$$

In (2.24), the positive constant $\sqrt{a} \ge |f'(u)|$ for all u. After the convection step (2.24), one has

$$u_j^* = \begin{cases} 1, & \text{if } j \le l(n) - 1, \\ \mu_1, & \text{if } j = l(n), \\ \mu_2, & \text{if } j = l(n) + 1, \\ -1, & \text{if } j > l(n) + 1, \end{cases} \quad (2.26)$$

where

$$1 > \mu_1 = \frac{k}{h}(s - \sqrt{a}) + 1 > \mu_2 = \frac{k}{h}(s + \sqrt{a}) - 1 > -1. \quad (2.27)$$

In the second step, the random projection (2.25) gives

$$u_j^{n+1} = \begin{cases} 1, & \text{if } j \le l(n) - 1, \\ -1, & \text{if } j > l(n) + 1, \end{cases} \quad (2.28)$$

while

$$u_{l(n)}^{n+1} = \begin{cases} 1, & \text{if } \mu_1 > \theta_n, \\ -1, & \text{if } \mu_1 \le \theta_n. \end{cases} \qquad u_{l(n)+1}^{n+1} = \begin{cases} 1, & \text{if } \mu_2 > \theta_n, \\ -1, & \text{if } \mu_2 \le \theta_n. \end{cases} \quad (2.29)$$

Thus one has

$$l(n+1) = \begin{cases} l(n) - 1, & \text{if } \mu_1 < \theta_n, \\ l(n), & \text{if } \mu_2 < \theta_n \le \mu_1, \\ l(n) + 1, & \text{if } \mu_2 \ge \theta_n. \end{cases} \quad (2.30)$$

Therefore the numerical shock, after each time step, could move to the left by one grid point, not move, or move to the right by one grid point. We now examine the accumulative effect over many time steps, which should take into account the possibilities of both moving forward and backward. By the definition of N in (2.12),

$$\begin{aligned} l(n) &= l(0) + N\{j = 1, \dots, n; \theta_j \in [-1, \mu_2]\} - N\{j = 1, \dots, n; \theta_j \in (\mu_1, 1]\} \\ &= l(0) + N\left\{ j = 1, \dots, n; \vartheta_j \in \left[0, \frac{\mu_2 + 1}{2}\right] \right\} \\ &\quad - N\left\{ j = 1, \dots, n; \vartheta_j \in \left[\frac{\mu_1 + 1}{2}, 1\right] \right\} \\ &= l(0) + N\{j = 1, \dots, n; \vartheta_j \in [0, \lambda_2] - N\{j = 1, \dots, n; \vartheta_j \in (\lambda_1, 1]\}, \end{aligned} \quad (2.31)$$

where $\lambda_i = \frac{1}{2}(\mu_1 + 1)$ for $i = 1, 2$. Using the definition of μ_1, μ_2 in (2.27), one obtains

$$\begin{aligned}
&|x_0 + st_n - l(n)h| \\
&\quad = |st_n - hN\{j = 1, \ldots, n; \vartheta_j \in [0, \lambda_2]\} + hN\{j = 1, \ldots, n; \vartheta_j \in (\lambda_1, 1]\}| \\
&\quad = |st_n - nh[\lambda_2 + \delta(\boldsymbol{\vartheta}, 0, n, [0, \lambda_2])] + nh[1 - \lambda_1 + \delta(\boldsymbol{\vartheta}, 0, n, (\lambda, 1])]| \\
&\quad = \Big| snk - nk\frac{1}{2}(s + \sqrt{a}) - nk\frac{1}{2}(s - \sqrt{a}) - nh\delta(\boldsymbol{\vartheta}, 0, n, [0, \lambda_2]) \\
&\qquad + nh\delta(\boldsymbol{\vartheta}, 0, n, (\lambda_1, 1])\Big| \\
&\quad \le nh|\delta(\boldsymbol{\vartheta}, 0, n, [0, \lambda_2])| + nh|\delta(\boldsymbol{\vartheta}, 0, n, (\lambda_1, 1])| \le 2h(1 + 3|\ln(2t_n/k)|) \\
&\quad = 2h(1 + 3|\ln(2st_n/(\lambda h))|). \qquad (2.32)
\end{aligned}$$

This is the accuracy estimate for the Lax–Friedrichs method.

The above analyses can be summarized by the following theorem:

THEOREM 2.1. *Given $T > 0$, the difference between the shock location of the exact solution, $x_0 + st_n$, and the numerical one, $l(n)h$, as determined by the random projection method*

$$u^{n+1} = S_\theta(k) S_c(k)\, u^n,$$

where $S_c(k)$ is the first-order upwind or Lax–Friedrichs method, has the estimate

$$|x_0 + st_n - l(n)h| \le C(T)h|\ln h|, \qquad (2.33)$$

for any n such that $0 < t_n \le T$ and fixed $\lambda = sk/h$, where $C(T)$ is a positive constant depending on $|\ln(T)|$.

From this inequality, one can see that the error is first-order ($O(h|\ln h|)$).

2.4. *Numerical Examples*

EXAMPLE 2.1. We solve (2.1) and (2.2) with $f(u) = \frac{u^2}{2} + u$, $\alpha = 0$, $\varepsilon = 10^{-4}$, and $x_0 = 0.2$ on the interval [0, 1] with 101 grid points using the random projection method (2.15)–(2.16). The mesh size $h = 0.01$. For this example the speed of discontinuity

$$s = \frac{f(1) - f(-1)}{2} = 1.$$

Figure 1 shows the computed results at $t = 0.2$, 0.4, and 0.6 using different time steps $k = 0.004, 0.001$, and 0.0001. The numerical solutions indeed capture the shock propagation with the correct speed.

As a comparison, we use the explicit method

$$u_i^{n+1} = u_i^n - \frac{k\big(f\big(u_i^n\big) - f\big(u_{i-1}^n\big)\big)}{h} + \frac{k}{\varepsilon}\big(u_i^n - \alpha\big)\big(1 - \big(u_i^n\big)^2\big). \qquad (2.34)$$

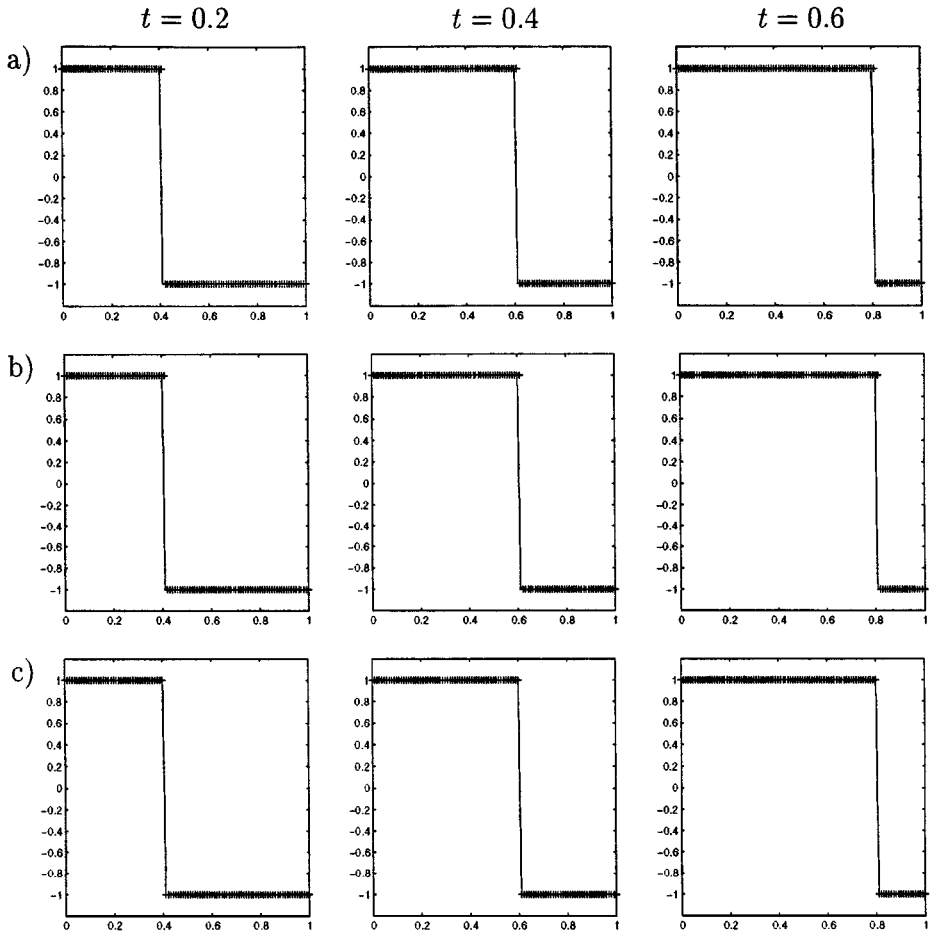

FIG. 1. Numerical results using the random projection method for example 2.1. $\varepsilon = 10^{-4}$, $h = 0.01$. $-$: True solution; $++$: computed solutions. (a) $k = 0.004$; (b) $k = 0.001$; and (c) $k = 0.0001$.

For this method, one has to choose $k < \varepsilon$ for numerical stability, even if $h \gg \varepsilon$. Figure 2 shows the computed results for (a) $\varepsilon = 0.1$ and $k = 0.001$; (b) $\varepsilon = 0.01$ and $k = 0.0001$; (c) $\varepsilon = 0.001$ and $k = 0.00001$; and (d) $\varepsilon = 0.0001$ and $k = 0.000001$. In this example, the correct locations of discontinuity at $t = 0.0$, 0.2, 0.4, and 0.6 are at $x = 0.2$, 0.4, 0.6, and 0.8 respectively. We observe that when the mesh resolves the reaction zone, as in cases (a) and (b), one can get the correct shock speed. In contrast, when the mesh does not resolve the reaction zone, as in cases (c) and (d), one cannot get the correct solution even with a time step smaller than ε. Instead the numerical shock speed is zero. This indicates that the incorrect shock speed is induced by the spatial underresolution rather than the temporal one.

EXAMPLE 2.2. We solve (2.1) and (2.2) with $f(u) = e^u, \alpha = 0, \varepsilon = 10^{-4}$, and $x_0 = 0.2$ over the interval [0, 1] with 101 grid points. The mesh size $h = 0.01$. For this example the speed of discontinuity $s = (f(1) - f(-1))/2 = (e - e^{-1})/2$ which is an irrational number. Figure 3 shows the numerical solutions at $t = 0.2$, 0.4, and 0.6 using different time steps $k = 0.001$, and 0.0001. In this example, the shock speed is correctly captured by the random projection method.

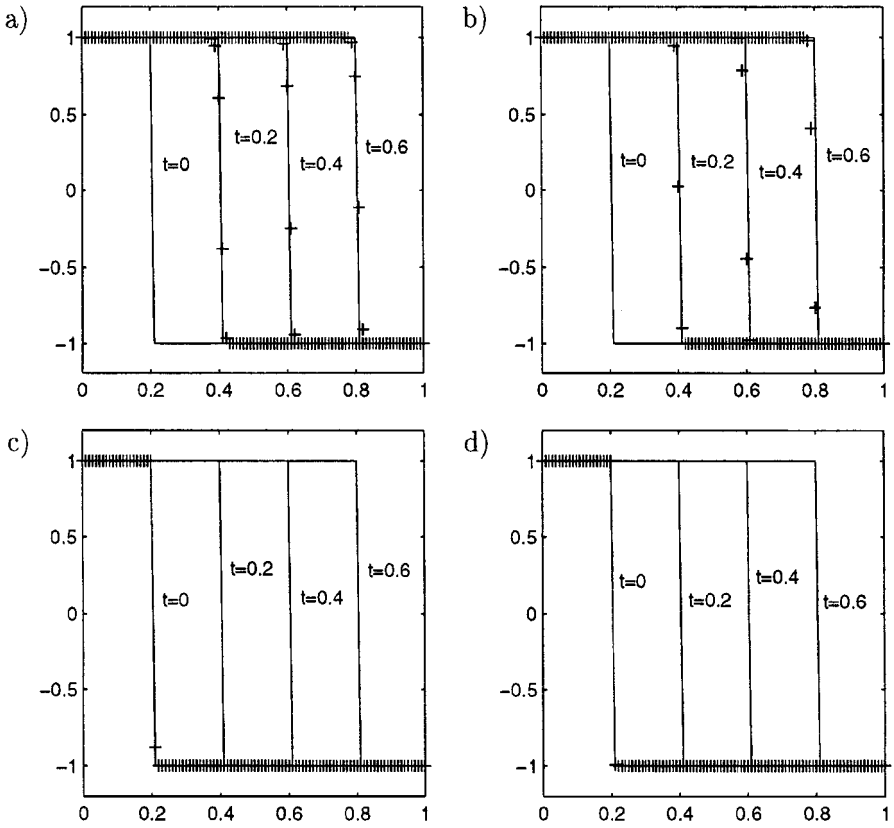

FIG. 2. Numerical results using the explicit method (2.34) for Example 2.1. $h = 0.01$. $-$: True solution; $++$: computed solutions. (a) $\varepsilon = 0.1$, $k = 0.001$; (b) $\varepsilon = 0.01$, $k = 0.0001$; (c) $\varepsilon = 0.001$, $k = 0.00001$; (d) $\varepsilon = 0.0001$, $k = 0.000001$.

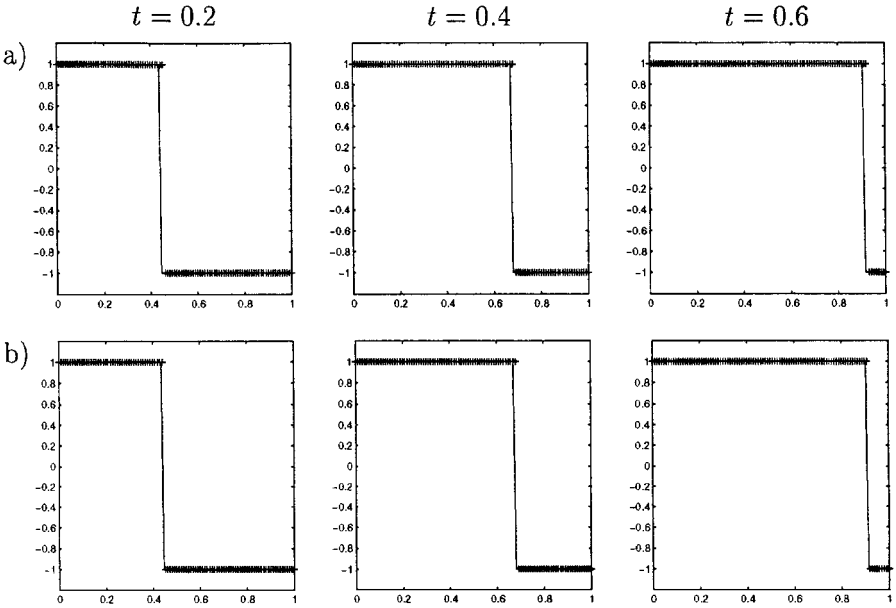

FIG. 3. Numerical results by using the random projection method for Example 2.2. $\varepsilon = 10^{-4}$, $h = 0.01$. $-$: True solution; $++$: computed solutions. (a) $k = 0.001$; (b) $k = 0.0001$.

Figures 1 and 3 show that the random projection method can capture the correct speeds of the discontinuities for scalar hyperbolic conservation laws with stiff source terms even when the reaction scale is not resolved spatially or temporally. The location of the shock may be off by a few grid points at each time step, but such a deviation never grows in time during our experiments to longer than that shown in the figures.

2.5. *Hyperbolic Equations with the Ricatti Sources*

The random projection method can be applied to numerical simulation of hyperbolic conservation laws with stiff, Ricatti sources:

$$u_t + f(u)_x = \frac{1}{\varepsilon} u(1-u). \tag{2.35}$$

Without loss of generality we start with the piecewise constant initial data

$$u(x,0) = u_0(x) = \begin{cases} 1, & x \le x_0, \\ 0, & x > x_0. \end{cases} \tag{2.36}$$

Here ε is a small parameter, f is a convex function of u, i.e., $f''(u) > 0$, and x_0 is any given point.

The Ricatti source in (2.35) admits two local equilibria, the unstable one $u = 0$ and the stable one $u = 1$ [20]. When the solution is at equilibrium, the Ricatti source has no effect. Thus the exact solution is a shock discontinuity connecting $u = 1$ with $u = 0$ and propagating to the right with a speed determined by the Rankine–Hugoniot jump condition:

$$\bar{s} = f(1) - f(0). \tag{2.37}$$

Namely,

$$u(x,t) = \begin{cases} 1, & \text{if } x \le x_0 + \bar{s}t, \\ 0, & \text{if } x > x_0 + \bar{s}t. \end{cases} \tag{2.38}$$

The idea of the random projection method is to replace the unstable equilibrium, $u = 0$, with the uniformly distributed random sequence $\vartheta_n \in (0, 1)$, defined in (2.10). Let $u^* = S_c(k)u^n$ with $S_c(k)$ being the discrete operator for the homogeneous equation (2.5) over a time step. The random projection operator S_ϑ is defined as

$$S_\vartheta(k): \qquad u_j^{n+1} = \begin{cases} 1, & \text{if } u_j^* > \vartheta_n, \\ 0, & \text{if } u_j^* \le \vartheta_n, \end{cases} \quad \text{for all } j. \tag{2.39}$$

In this method, one random value of ϑ_n will be selected per time step for all grid points. Combining with the convection step, we get

$$S_{\mathrm{rp}}(k): \qquad u^{n+1} = S_\vartheta(k) S_c(k) u^n. \tag{2.40}$$

We can also obtain a similar error estimate for the numerical shock speed for problems with the Ricatti source.

THEOREM 2.2. *Given $T > 0$, the difference between the shock location in the exact solution of the problem* (2.35), $x_0 + \bar{s}t_n$, *and the numerical one*, $l(n)h$, *as determined by the random projection method* (2.40) *in which $S_c(k)$ is the first-order upwind or Lax–Friedrichs method, estimated as*

$$|x_0 + \bar{s}t_n - l(n)h| \le C(T)h|\ln h|, \tag{2.41}$$

for any $0 < t_n \le T$, and fixed $\bar{\lambda} = \bar{s}k/h$, where $C(T)$ is a positive constant depending on $|\ln(T)|$.

From this inequality, one can see that the error is first order ($O(h|\ln h|)$).

Remark 2.2. When one considers the problem (2.35) with the Ricatti source $(1/\varepsilon)u(1-u)$ replaced by $(1/\varepsilon)u(1+u)$, our random projection method still works after the sampling sequence ϑ_n is replaced by another sequence $\tilde{\theta}_n = \vartheta_n - 1$, which is equidistributed on the interval $[-1, 0]$.

3. TWO-DIMENSIONAL SCALAR PROBLEMS

The random projection method can be extended to two space dimensions in a straightforward way. Consider the two-dimensional scalar hyperbolic conservation law with stiff reaction term

$$u_t + f(u)_x + g(u)_y = \frac{1}{\varepsilon}\psi(u) \equiv \frac{1}{\varepsilon}(u-\alpha)(1-u^2), \qquad -1 < \alpha < 1, \tag{3.1}$$

with piecewise constant initial data

$$u(x, y, 0) = u_0(x, y) = \begin{cases} 1, & (x, y) \in \Omega_0 \subset \mathbb{R}^2, \\ -1, & (x, y) \in \mathbb{R}^2 \setminus \Omega_0. \end{cases} \tag{3.2}$$

Here ε is the reaction time, f and g are convex functions of u, i.e., $f''(u) > 0$ and $g''(u) > 0$, and Ω_0 is a given domain in $\mathbb{R}^2$.

Let the grid points $(x_i, y_j) = (ih, jh)$, $i, j = \ldots, -1, 0, 1, \ldots$, with equal mesh spacing h. The time levels $t_n = nk$, $k = 0, 1, 2, \ldots$, are also uniformly spaced with time step k. Let $u^n_{i,j}$ be the approximate solution of u at $(x_i, y_j, t_n) = (ih, jh, nk)$. Similarly to the one-dimensional scalar model problem, we use the fraction step method consisting of the convection step for

$$u_t + f(u)_x + g(u)_y = 0 \tag{3.3}$$

followed by the reaction step

$$u_t = \frac{1}{\varepsilon}(u-\alpha)(1-u^2). \tag{3.4}$$

Let $S_{f,g}(k)$ be any high resolution shock capturing method for (3.3), and let $u^* = S_{f,g}(k)u^n$. The random projection for the reaction is given by

$$S_\theta(k): \qquad u^{n+1}_{ij} = \begin{cases} 1, & \text{if } u^*_{ij} > \theta_n, \\ -1, & \text{if } u^*_{ij} \le \theta_n, \end{cases} \quad \text{for all } i, j, \tag{3.5}$$

where

$$\theta_n = 2\vartheta_n - 1,$$

with ϑ_n being the van der Corput sampling sequence on the interval [0, 1] defined in (2.10).

Our fractional step method is then

$$u^{n+1} = S_\theta(k) S_{f,g}(k) u^n. \tag{3.6}$$

The stability condition for this algorithm is the usual CFL condition determined by the convection part. The initial data are discretized as

$$u_{ij}^0 = \begin{cases} 1, & (x_i, y_j) = (ih, jh) \in \bar{\Omega}_0, \\ -1, & \text{otherwise.} \end{cases} \tag{3.7}$$

Now we will test algorithm (3.6) with two numerical examples. In our numerical computations in this section, the operator $S_{f,g}(k)$ is the first-order upwind scheme using dimensional splitting.

EXAMPLE 3.1. We solve (3.1) with $f(u) = g(u) = u$, $\alpha = 0$, and $\varepsilon = 10^{-6}$. The initial condition is

$$u(x, y, 0) = u_0(x, y) = \begin{cases} 1, & x + y \le 1, \\ -1, & x + y > 1. \end{cases} \tag{3.8}$$

This problem is solved on the domain $[0, 1]^2$ with 101×101 grid points using the 2D random projection method (3.6). The mesh size $h = 0.01$. Dirichlet boundary conditions are used at inflow boundaries ($x = 0$, and $y = 0$), and outflow boundary conditions are used on the other two sides of the boundary. For this example the speed of the discontinuity is 2 in both x and y directions; see [10]. Figure 4 shows the computed results of the discontinuity front (i.e., contour of $u = 0$) at $t = 0.2$, 0.4, and 0.45 for time steps $k = 0.005$ and 0.001

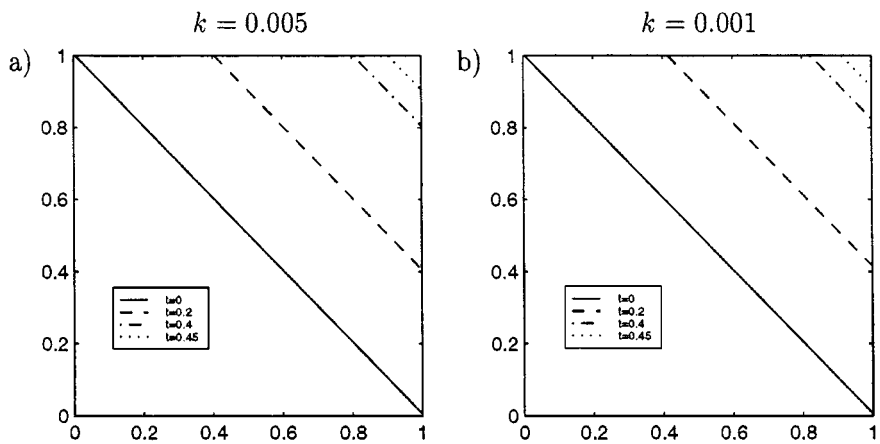

FIG. 4. Numerical results for Example 3.1 using the 2D random projection method (3.6). $\varepsilon = 10^{-6}$, $h = 0.01$. (a) $k = 0.005$; (b) $k = 0.001$.

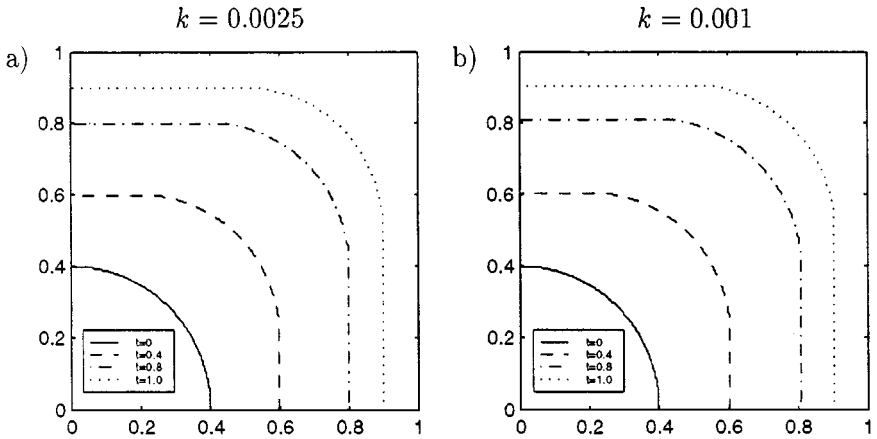

FIG. 5. Numerical results for Example 3.2 by using the 2D random projection method (3.6). $\varepsilon = 10^{-6}$, $h = 0.005$. (a) $k = 0.0025$; (b) $k = 0.001$.

respectively. At these times the x- and y-coordinates of the exact solutions will be 0.4, 0.8, and 0.9 respectively. The shock propagation was accurately captured.

EXAMPLE 3.2. Now consider the nonlinear convection with reaction

$$u_t + (u^2/2)_x + (u^2/2)_y = 10^6 u(1-u)(u-1/2), \tag{3.9}$$

with initial condition

$$u(x, y, 0) = u_0(x, y) = \begin{cases} 1, & x^2 + y^2 \le 0.16, \\ 0, & x^2 + y^2 > 0.16. \end{cases} \tag{3.10}$$

This problem is interesting because of the nonlinear flux functions and the curved shock front. The equilibria are different from those of (3.1), so we have to use the van der Corput sampling sequences $\theta_n = \vartheta_n$. We solve this problem on the domain $[0, 1]^2$ with 201×201 grid points using the 2D random projection method (3.6). So the mesh size $h = 0.005$. Homogeneous Neumann boundary conditions are used at the inflow boundaries ($x = 0$ and $y = 0$), while outflow boundary conditions are imposed on the other sides of the boundary. For this example the shock front moves with speed 0.5 along the axes [10]. Figure 5 shows the computed results of the discontinuity front (i.e. contour of $u = 0.5$) at $t = 0.4$, 0.8, and 1.0 for time steps $k = 0.0025$ and 0.001 respectively. At these times the exact solutions have the x- and y-coordinates 0.6, 0.8, and 0.9 respectively. The numerical solutions give good approximate shock locations at these times.

4. ONE-DIMENSIONAL DETONATION WAVES

In this section, we introduce the random projection method for the computation of one-dimensional stiff detonation waves. The key idea is to **make the ignition temperature random.**

Consider the Euler equations that model the time-dependent flow of an inviscid, compressible, reactive gas in one space dimension. Without heat conduction and viscosity, the

equations take the form

$$U_t + F(U)_x = \frac{1}{\varepsilon}\Psi(U), \tag{4.1}$$

$$U = \begin{pmatrix} \rho \\ m \\ e \\ \rho z \end{pmatrix}, \quad F(U) = \begin{pmatrix} m \\ m^2/\rho + p \\ m(e+p)/p \\ mz \end{pmatrix}, \quad \Psi(U) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\rho z e^{-T_0/T} \end{pmatrix} \equiv \begin{pmatrix} 0 \\ 0 \\ 0 \\ \psi(U) \end{pmatrix}.$$

The dependent variables $\rho(x,t)$, $m(x,t)$, $e(x,t)$, and $z(x,t)$ are the density, momentum, total energy, and fraction of unreacted fluid, respectively. The pressure is given by

$$p = (\gamma - 1)\left(e - \frac{m^2}{2\rho} - q_0\rho z\right)$$

and the temperature is defined as $T = p/\rho$. Let $u = m/\rho$ be the velocity. The parameters q_0, T_0, γ, and ε correspond to chemical heat release, ignition temperature, c_p to c_v ratio, and reaction time.

We will focus on the discontinuous solutions of detonation waves. For these waves the viscosity is not as important as for the slower deflagration wave solutions.

Equations (4.1) are referred to the reactive Euler equations with Arrhenius kinetics [8]. We will also consider (4.1) with

$$-\frac{1}{\varepsilon}\rho z e^{-T_0/T}$$

replaced by the Heaviside kinetics

$$-\frac{1}{\varepsilon}\rho z H(T - T_0),$$

where $H(x) = 1$ for $x > 0$, and $H(x) = 0$ for $x < 0$. Actually the stiffness issue with the Heaviside kinetics is more severe [11].

Now we will describe the random projection method for solving the problem (4.1) with piecewise constant initial data

$$(\rho(x,0), u(x,0), p(x,0), z(x,0)) = \begin{cases} (\rho_{\mathrm{l}}, u_{\mathrm{l}}, p_{\mathrm{l}}, 0), & \text{if } x \le x_0, \\ (\rho_{\mathrm{r}}, u_{\mathrm{r}}, p_{\mathrm{r}}, 1), & \text{if } x > x_0, \end{cases} \tag{4.2}$$

where x_0 is a given point. For simplicity these data are chosen so that the detonation moves to the right. The case where the detonation moves to the left can be dealt with similarly. Let $U_j^n = (\rho_j^n, m_j^n, e_j^n, (\rho z)_j^n)$ be the approximate solution of $U = (\rho, m, e, \rho z)$ at $(x_j, t_n) = (jh, nk)$. As with the scalar model problem, a fractional step method is used. In the first step, the inviscid compressible Euler equations,

$$U_t + F(U)_x = 0, \tag{4.3}$$

are solved using a standard shock capturing method $S_F(k)$. Since any shock capturing method will have a few grid points in the shock profile, the corresponding temperature values, once above the critical temperature T_0, may trigger a too early chemical reaction, causing non-physical waves [1, 7].

4.1. *Random Projection Methods for the Reaction Term*

In the second step of the fractional step method, we solve the chemical reaction term,

$$\rho_t = 0, \qquad m_t = 0, \qquad e_t = 0, \qquad (\rho z)_t = \frac{1}{\varepsilon}\psi(U), \tag{4.4}$$

using the random projection method. Here the ignition temperature T_0 is made random. This can be successful since the speed of the front does not depend on the specific value of T_0, as long as it is in the range between the left and right states. Let $U^* = S_F(k)U^n$. Our first choice for the random projection operator S_θ is defined as follows. Let

$$\theta_n = (T_\mathrm{l} - T_\mathrm{r})\vartheta_n + T_\mathrm{r}, \qquad T_l = p_\mathrm{l}/\rho_\mathrm{l}, \qquad T_\mathrm{r} = p_\mathrm{r}/\rho_\mathrm{r},$$

with ϑ_n (see (2.10) for detail) being the van der Corput sampling sequence on the interval [0, 1]. For all j,

$$\begin{aligned} S_0(k): \qquad & \rho_j^{n+1} = \rho_j^*, \qquad m_j^{n+1} = m_j^*, \qquad e_j^{n+1} = e_j^*, \\ & z_j^{n+1} = \begin{cases} 0, & \text{if } T_j^* > \theta_n, \\ 1, & \text{if } T_j^* \le \theta_n. \end{cases} \end{aligned} \tag{4.5}$$

This operator will be referred to as **global random projection**, since the projection operator $S_\theta(k)$ is used for all grid points.

The combination of the two steps gives the fractional step method as

$$S_1(k): \qquad U^{n+1} = S_\theta(k)S_F(k)U^n. \tag{4.6}$$

Since the reaction zone is local, it makes more sense to do the random projection around the reaction zone. In fact, the global projection could produce incorrect solutions when an intermediate state T_m behind the detonation (thus corresponding to $z = 0$, namely the burnt state) emerges between T_l and T_r. Since the random numbers are chosen over the interval $[T_\mathrm{r}, T_\mathrm{l}]$, and if $T_\mathrm{m} \ge T_\mathrm{l}$, the global random projection treats the state of T_m as burnt and the results will be corrected. However, if $T_\mathrm{r} < T_\mathrm{m} < T_\mathrm{l}$, then once a random number is above T_m, the state of T_m will be treated as an unburnt state and chemical reaction will take place in that state, which yields the wrong solution. This will be shown by Example 4.4 in this section.

We can avoid the problem of global random projection by the following **local random projection method**, which performs the random projection only near the detonation front. This guarantees that a state, once burnt, remains burnt. The location of the front can be easily determined from the value of z.

First notice that, although z may have some intermediate values between 0 and 1 after the convection step, the projection step always makes z either 0 or 1. Therefore, at any time step t_n, there is an $l(n) = j_0$, j_0 an integer, such that

$$z_j^n = \begin{cases} 0, & \text{if } j \le l(n), \\ 1, & \text{if } j > l(n). \end{cases}$$

Here $l(n)$ is the location of the jump for numerical solution of z in the approximate solution at time t_n. Since the projection operator will make the detonation move at most a few grid points, the new location of the detonation front is only a few grids away from that of the

previous time step. Thus one need only to check the value of T at several points near $l(n)$ for the value of $l(n+1)$. Therefore, (4.5) can be replaced with the local projection around the front

$$\begin{aligned}
\tilde{S}_\theta(k)\colon\quad & p_j^{n+1} = p_j^*, \qquad m_j^{n+1} = m_j^*, \qquad e_j^{n+1} = e_j^*, \\
& \text{Set } l(n+1) := l(n) - 1, \qquad (4.7) \\
& \text{For } l = l(n)-1, l(n), \ldots, l(n)+d \quad \text{do: } l(n+1) = l, \quad \text{if } T_l^* > \theta_n; \\
& z_j^{n+1} = \begin{cases} 0, & \text{if } j \le l(n+1), \\ 1, & \text{if } j > l(n+1); \end{cases} \qquad (4.8) \\
& \text{End if}
\end{aligned}$$

where d is the number of smeared points in the shock layer. From our numerical experience, for Chapman–Jouguet (C–J) detonations and strong detonations, $d = 1$ works very well. In the above algorithm, only $d + 2$ points will be scanned.

The fractional step method based on this local projection is

$$S_2(k)\colon \qquad U^{n+1} = \tilde{S}_\theta(k) S_F(k) U^n. \qquad (4.9)$$

For numerical comparison, we also describe the **deterministic** method, in which one always uses the given deterministic ignition temperature T_0. In the case of Heaviside kinetics, an implicit method for (4.4) gives

$$\begin{aligned}
S_{\mathrm{r}}(k)\colon\quad & p_j^{n+1} = p_j^*, \qquad m_j^{n+1} = m_j^*, \qquad e_j^{n+1} = e_j^*, \\
& (\rho z)_j^{n+1} = \begin{cases} (\rho z)_j^*, & \text{if } T_j^{n+1} \le T_0, \\ (\rho z)_j^*/(1 + k/\varepsilon), & \text{if } T_j^{n+1} > T_0. \end{cases} \qquad (4.10)
\end{aligned}$$

Since ρz decreases when T becomes larger than T_0, one gets the same result if the test $T_j^{n+1} > T_0$ is replaced by $T_j^* > T_0$ [11], thereby making it unnecessary to solve any nonlinear equations. Then the corresponding algorithm is

$$S_3(k)\colon \qquad U^{n+1} = \bar{S}_{\mathrm{r}}(k) S_F(k) U^n, \qquad (4.11)$$

where

$$\begin{aligned}
\bar{S}_{\mathrm{r}}(k)\colon\quad & p_j^{n+1} = p_j^*, \qquad m_j^{n+1} = m_j^*, \qquad e_j^{n+1} = e_j^*, \\
& (\rho z)_j^{n+1} = \begin{cases} (\rho z)_j^*, & \text{if } T_j^* \le T_0, \\ (\rho z)_j^*/(1 + k/\varepsilon), & \text{if } T_j^* > T_0. \end{cases} \qquad (4.12)
\end{aligned}$$

The stability condition for all three algorithms is the usual CFL condition determined by the convection terms.

4.2. *Numerical Results for Detonation Wave Problems*

Now we will test the above three algorithms S_1, S_2, and S_3 by a variety of numerical examples, including the C–J detonation and strong detonations. In our computations, we

always use the Heaviside kinetics. Among the numerical examples in this and the next sections, the convection step is solved using the second-order TVD relaxed scheme [15], which is free of any Riemann solver and local characteristic decomposition. We choose $d = 5$ in (4.7)–(4.8) in our computations in this subsection.

EXAMPLE 4.1 (A Chapman–Jouguet (C–J) Detonation). We consider the case of ozone decomposition C–J detonation discussed and computed in [1, 7]. We use the CGS units and the following parameter values:

$$\gamma = 1.4, \qquad q_0 = 0.5196 \times 10^{10}, \qquad \frac{1}{\varepsilon} = K = 0.5825 \times 10^{10}, \qquad T_0 = 0.1155 \times 10^{10}.$$

The initial data are piecewise constants defining a C–J detonation as a single wave (recall that in the Chapman–Jouguet model a C–J detonation corresponds to a sonic detonation or, in other words, a sharp reaction wave that moves at the minimal speed relative to the unburnt gas; see [8] for more details), given by

$$(\rho, u, p, z) = \begin{cases} (\rho_0, u_0, p_0, 1), & \text{if } x > 0.005, \\ (\rho_{\mathrm{CJ}}, u_{\mathrm{CJ}}, p_{\mathrm{CJ}}, 0), & \text{if } x \le 0.005, \end{cases}$$

where $p_0 = 8.321 \times 10^5$, $\rho_0 = 1.201 \times 10^{-3}$, $u_0 = 0$, and $p_{\mathrm{CJ}} = 6.270 \times 10^6$, $\rho_{\mathrm{CJ}} = 1.945 \times 10^{-3}$, $u_{\mathrm{CJ}} = 4.162 \times 10^4$. In fact, for any given initial state $(\rho_0, u_0, p_0, 1)$ on the right, one can obtain the C–J initial state on the left by [5, 8]

$$p_{\mathrm{CJ}} = -b + (b^2 - c)^{1/2}, \tag{4.13}$$

$$\rho_{\mathrm{CJ}} = \frac{\rho_0[p_{\mathrm{CJ}}(\gamma + 1) - p_0]}{\gamma p_{\mathrm{CJ}}}, \tag{4.14}$$

$$D_{\mathrm{CJ}} = \left[\rho_0 u_0 + (\gamma p_{\mathrm{CJ}} \rho_{\mathrm{CJ}})^{1/2}\right] / \rho_0, \tag{4.15}$$

$$u_{\mathrm{CJ}} = D_{\mathrm{CJ}} - (\gamma p_{\mathrm{CJ}} / \rho_{\mathrm{CJ}})^{1/2}, \tag{4.16}$$

where $b = -p_0 - \rho_0 q_0(\gamma - 1)$, $c = p_0^2 + 2(\gamma - 1) p_0 \rho_0 q_0 / (\gamma + 1)$, and D_{CJ} is the speed of the sharp front (in a C–J detonation).

Observe that the values p_{CJ}, ρ_{CJ}, and u_{CJ} depend only on p_0, u_0, ρ_0, and q_0 (and not on K, i.e., ε or T_0). The speed of the sharp front in Example 4.1 is $D_{\mathrm{CJ}} = 1.088 \times 10^5$. In this example the width of the reaction zone is approximately 5×10^{-5}. See [1, 7].

This problem is solved on the interval [0, 0.05]. The "exact" solutions are obtained by using the deterministic method (4.11) and choosing $h = 5 \times 10^{-6}$ (i.e., 10,001 grid points on the interval [0, 0.05]) and $k = 5 \times 10^{-12}$. This mesh size and time step resolve the reaction scale. Now we compare the results obtained by the random projection methods and the deterministic method when the reaction zone is underresolved. The solutions are displayed at $t_1 = 5 \times 10^{-8}$, $t_2 = 1 \times 10^{-7}$, and $t_3 = 2 \times 10^{-7}$ in Figs. 6–8. The grid size $h = 5 \times 10^{-4}$ (i.e., 101 grid points for the interval [0, 0.05]) for all numerical experiments.

Figure 6 shows the results of the global random projection method (4.6) with $k = 5 \times 10^{-10}$; Fig. 7 shows the results of the deterministic method (4.11) with the same time step $k = 5 \times 10^{-10}$; and Fig. 8 shows the results of the deterministic method (4.11) with a much

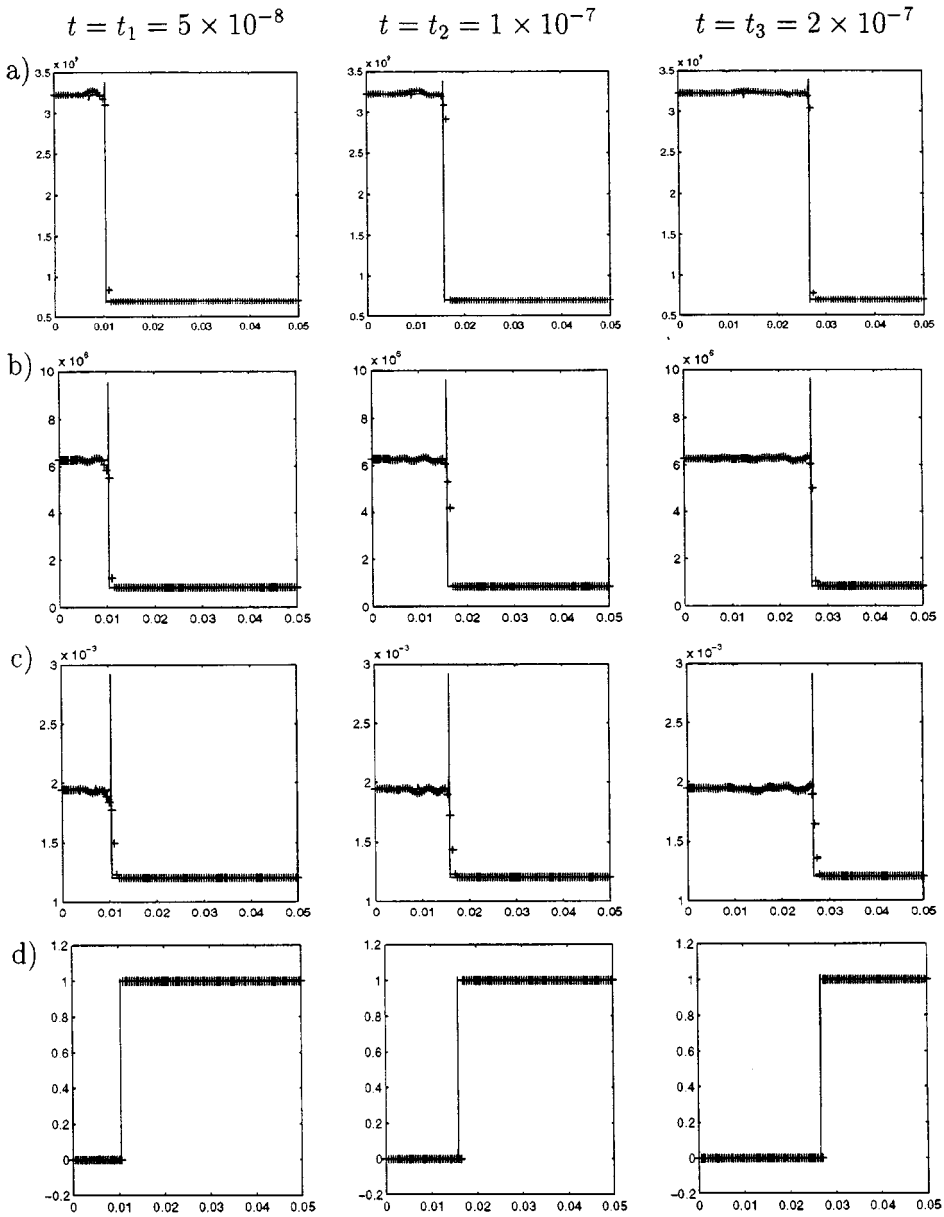

FIG. 6. Numerical results for a C–J detonation in Example 4.1 by using the global random projection method (4.6). $h = 5 \times 10^{-4}, k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

smaller time step $k = 5 \times 10^{-12}$. (In this case the time step resolves the reaction time but the spatial discretization does not resolve the reaction zone.)

In Fig. 6, the global random projection method captures the correct speed of the C–J detonation with underresolved spatial and time discretizations. There are some mild statistical fluctuations behind the detonation, which is the nature of a random method. Figures 7 and 8 show that, when the reaction zone is underresolved spatially, no matter how small the time step is, one cannot obtain the correct speed of the discontinuity by using the explicit or

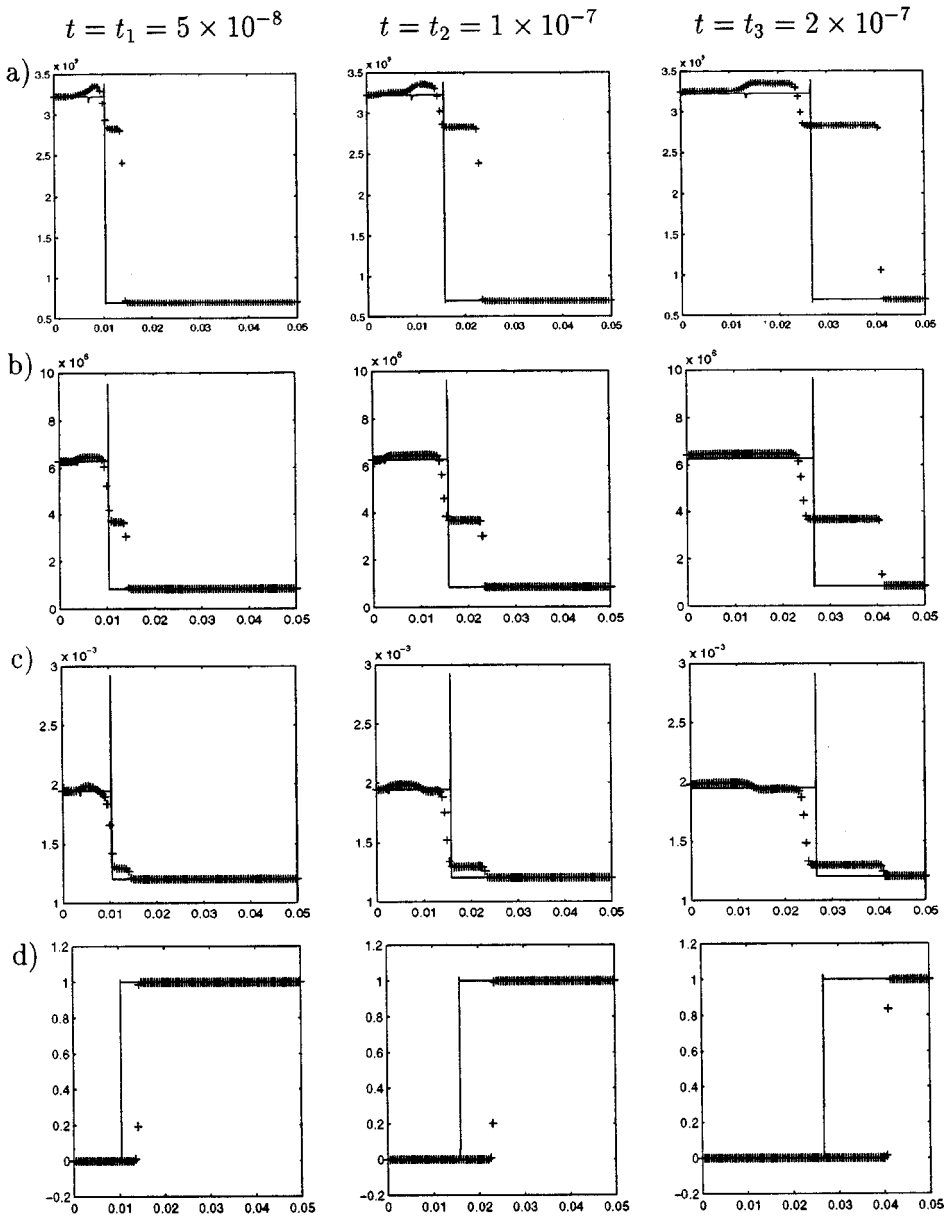

FIG. 7. Numerical results for Example 4.1 by using the deterministic method (4.11). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

implicit deterministic method for the chemical terms. A spurious weak detonation appears ahead of the detonation wave. This seems to suggest that the stiffness problem is due to the spatial rather than the temporal underresolution.

EXAMPLE 4.2 (A Strong Detonation). The initial data in this example are

$$(\rho, u, p, z) = \begin{cases} (\rho_0, u_0, p_0, 1), & \text{if } x > 0.005, \\ (\rho_1, u_1, p_1, 0), & \text{if } x \le 0.005, \end{cases}$$

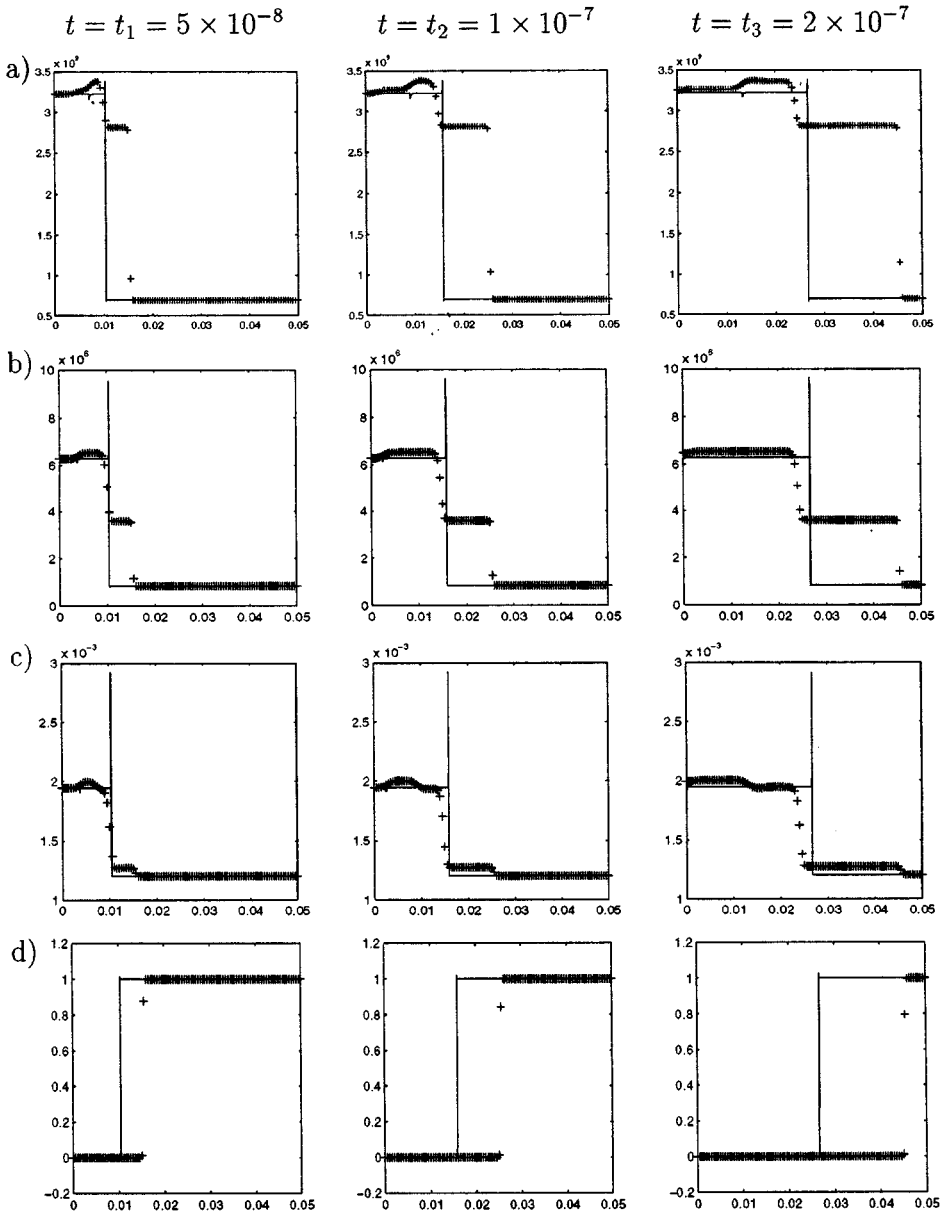

FIG. 8. Numerical results for Example 4.1 by using the deterministic method (4.11). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-12}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

where $u_1 = 9.162 \times 10^4 > u_{\mathrm{CJ}}$, $\rho_1 = \rho_{\mathrm{CJ}}$, $p_1 = p_{\mathrm{CJ}}$, and p_0, u_0, ρ_0, p_{CJ}, u_{CJ}, ρ_{CJ}, and all the other parameters (i.e. γ, q_0, K, and T_0) are the same as those in Example 4.1. The exact solution consists of a right-moving strong detonation [5], a right-moving contact discontinuity, and a stationary shock.

The "exact" solution is obtained similarly to that in Example 4.1. Figures 9 and 10 show the numerical solutions by the local random projection method and the deterministic method respectively. The local random projection method can capture the correct speed of

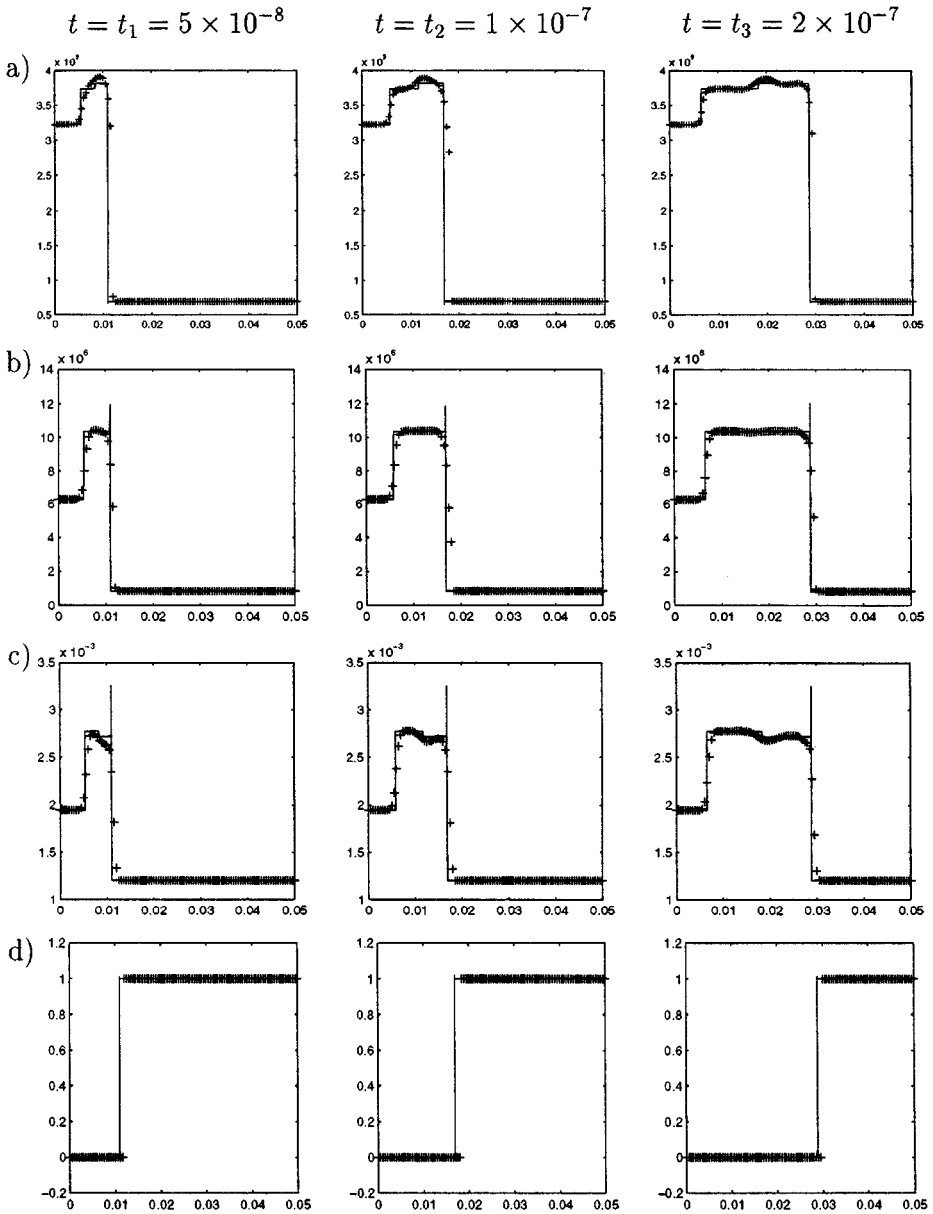

FIG. 9. Numerical results for Example 4.2 by using the local random projection method (4.9). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

the discontinuity of the strong detonation wave, among other waves, although the reaction zone is not resolved spatially or temporally. However, the deterministic method produces spurious waves.

EXAMPLE 4.3 (Another Strong Detonation). The initial data are

$$(\rho, u, p, z) = \begin{cases} (\rho_0, u_0, p_0, 1), & \text{if } x > 0.005, \\ (\rho_1, u_1, p_1, 0), & \text{if } x \le 0.005, \end{cases}$$

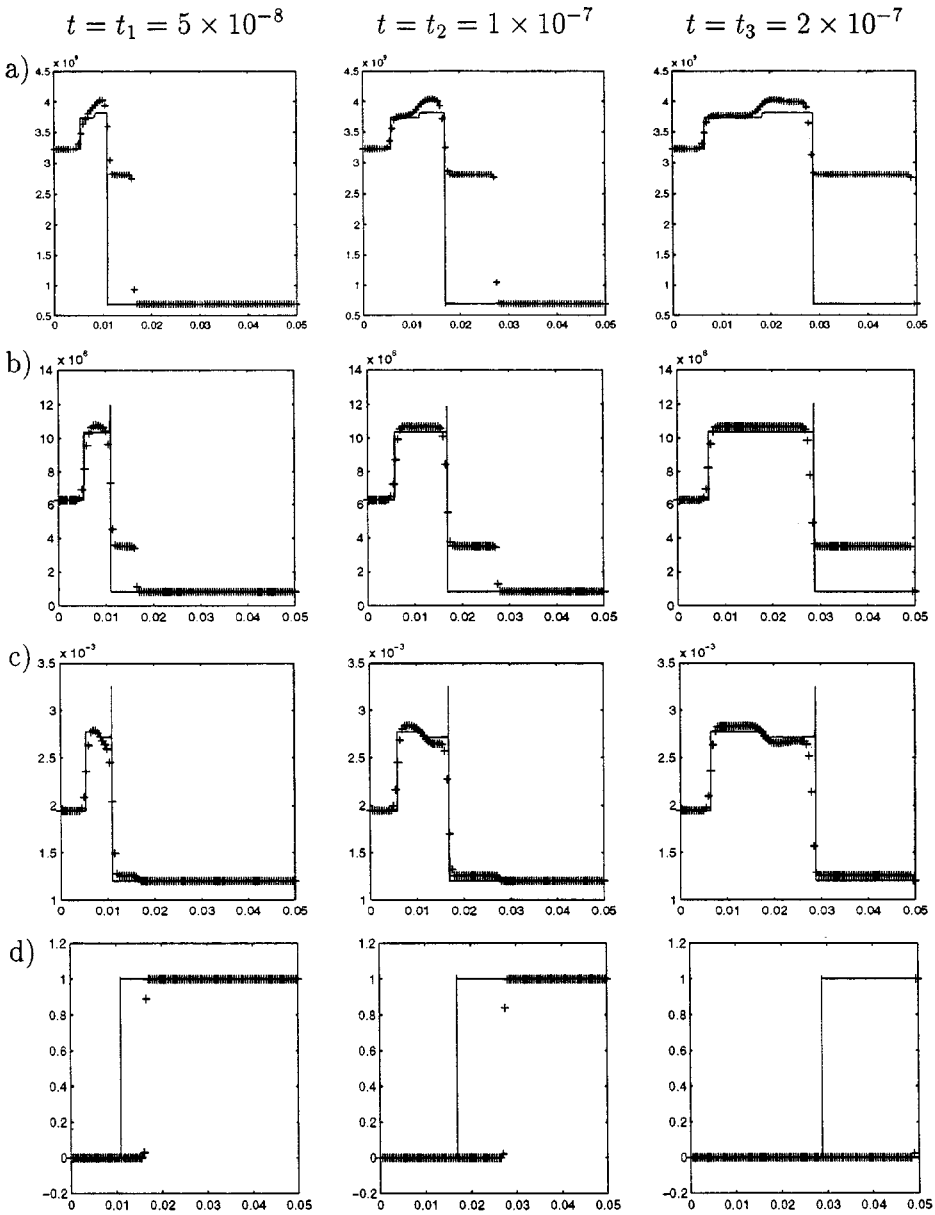

FIG. 10. Numerical results for Example 4.2 by using the deterministic method (4.11). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

where $u_1 = 9.162 \times 10^4 > u_{CJ}$, $\rho_1 = \rho_{CJ}$, $p_1 = 8.27 \times 10^6 > p_{CJ}$, and $p_0, u_0, \rho_0, p_{CJ}, u_{CJ}$, ρ_{CJ}, and all other parameters are the same as those in Example 4.1. The wave pattern in this example is the same as that in Example 4.2. The exact solution is obtained like that in Example 4.1. Figures 11 and 12 show the numerical solutions obtained by the local random projection method and the deterministic method, respectively. The local random projection method captures the propagation of all waves, while the deterministic method gives a spurious weak detonation.

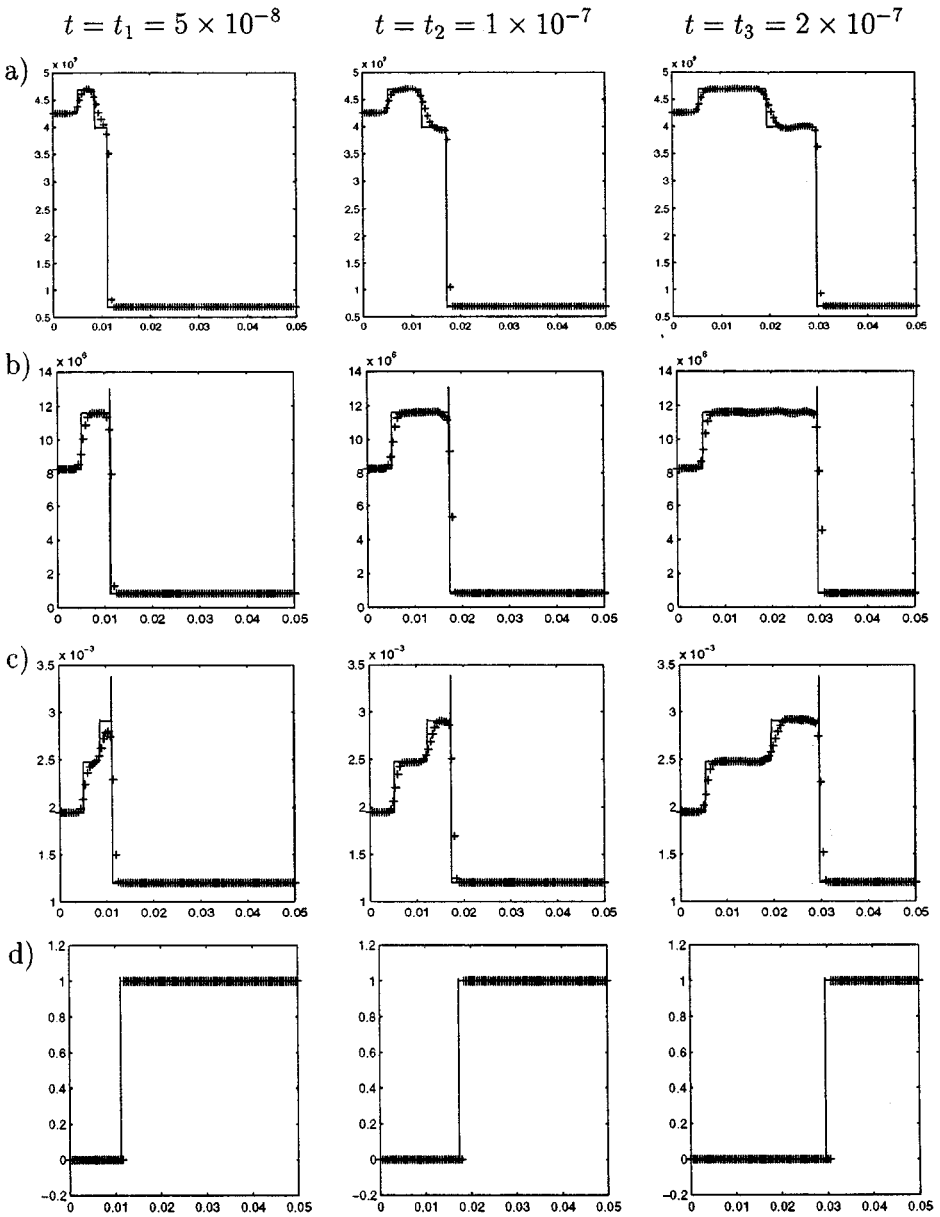

FIG. 11. Numerical results for Example 4.3 by using the local random projection method (4.9). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

EXAMPLE 4.4. The initial data are

$$(\rho, u, p, z) = \begin{cases} (\rho_0, u_0, p_0, 1), & \text{if } x > 0.005, \\ (\rho_1, u_1, p_1, 0), & \text{if } x \le 0.005, \end{cases}$$

where $u_1 = u_{CJ}$, $\rho_1 = \rho_{CJ}$, and $p_1 = 8.27 \times 10^6 > p_{CJ}$ while p_0, u_0, ρ_0, p_{CJ}, u_{CJ}, ρ_{CJ}, and other parameters are the same as those in Example 4.1. The solution consists of a

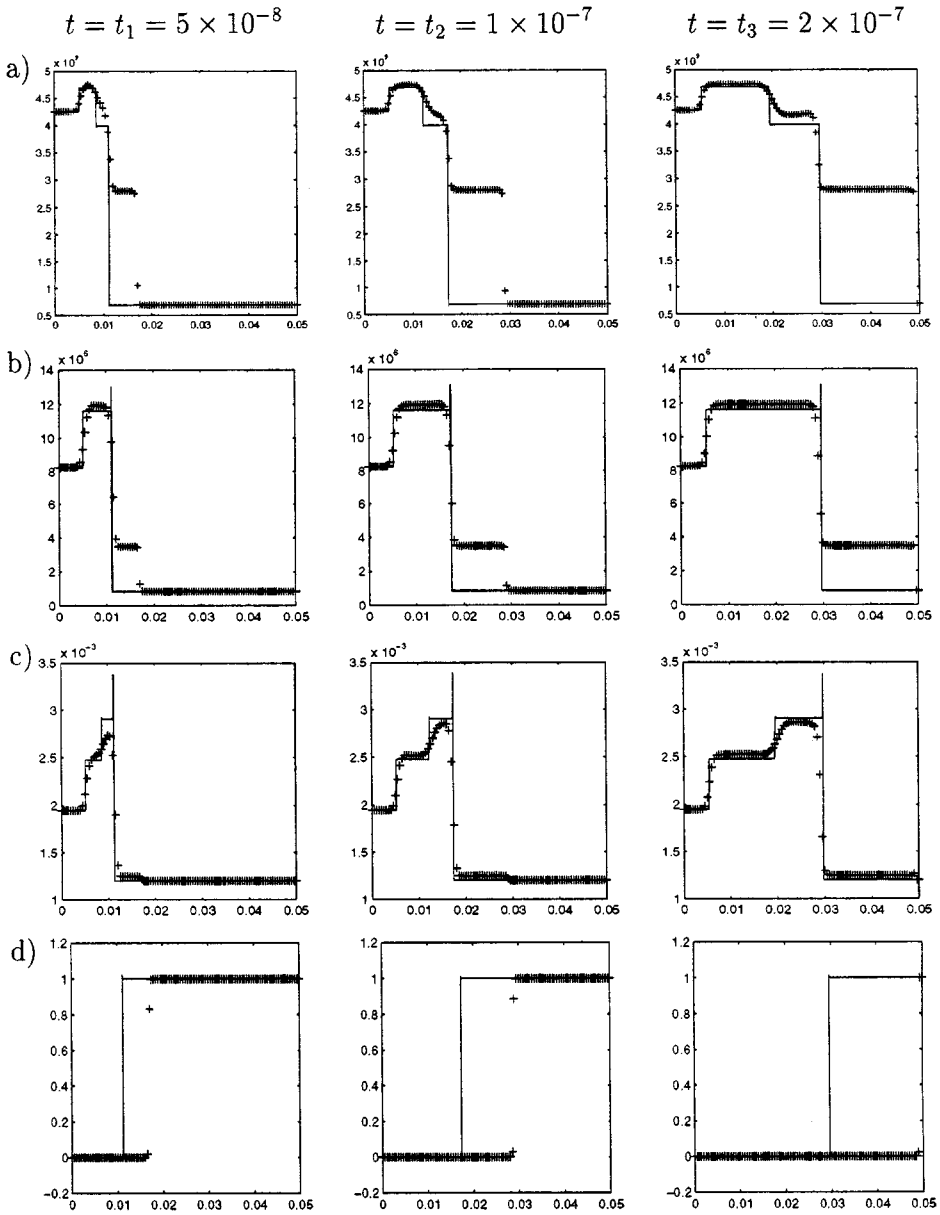

FIG. 12. Numerical results for Example 4.3 by using the deterministic method (4.11). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

right-moving strong detonation wave, a right-moving contact discontinuity, and a left-moving rarefaction wave. The exact solution is obtained that in Example 4.1. We take $h = 5 \times 10^{-4}$ and $k = 5 \times 10^{-10}$ and output the solutions at time $t_1 = 5 \times 10^{-8}$, $t_2 = 1 \times 10^{-7}$, and $t_3 = 2 \times 10^{-7}$.

Figure 13 shows the numerical solutions using the local random projection method (4.9), which captures all wave propagations. Figure 14 shows those by the deterministic method (4.11), which produced a spurious weak detonation.

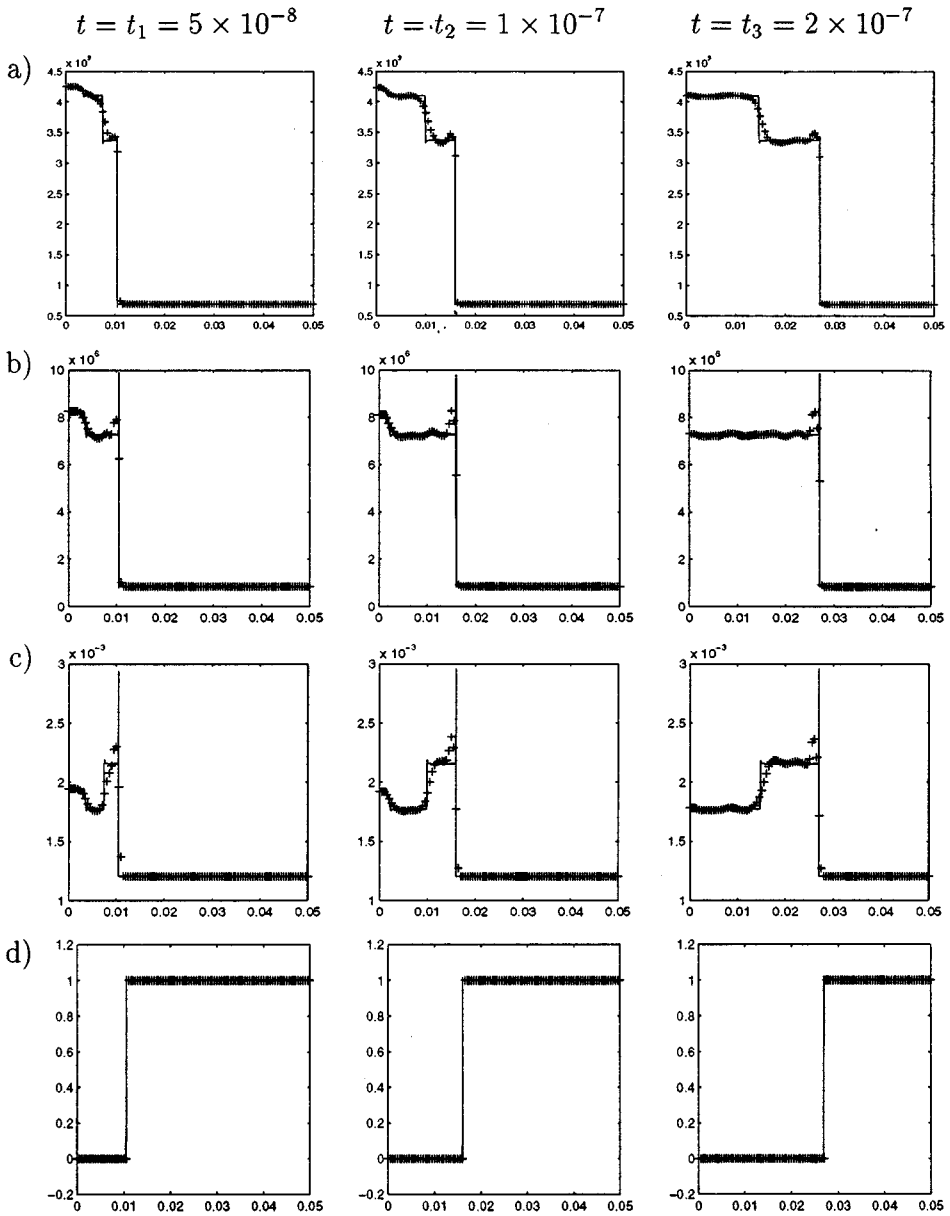

FIG. 13. Numerical results for Example 4.4 by using the local random projection method (4.9). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

We also computed the solution with the global random projection method and depict the numerical solutions in Fig. 15. The solution is essentially wrong, due to the improper projection away from the detonation front. Here the temperature, T_{m}, of the equilibrium state just behind the detonation front is less than T_{l}. The global random projection method triggers the location of the right-moving contact discontinuity rather than the location of the detonation front in the time steps when $T_{\mathrm{m}} < \theta_n < T_{\mathrm{l}}$. This causes the wrong numerical solutions. In summary, the local random projection method can capture the correct speeds

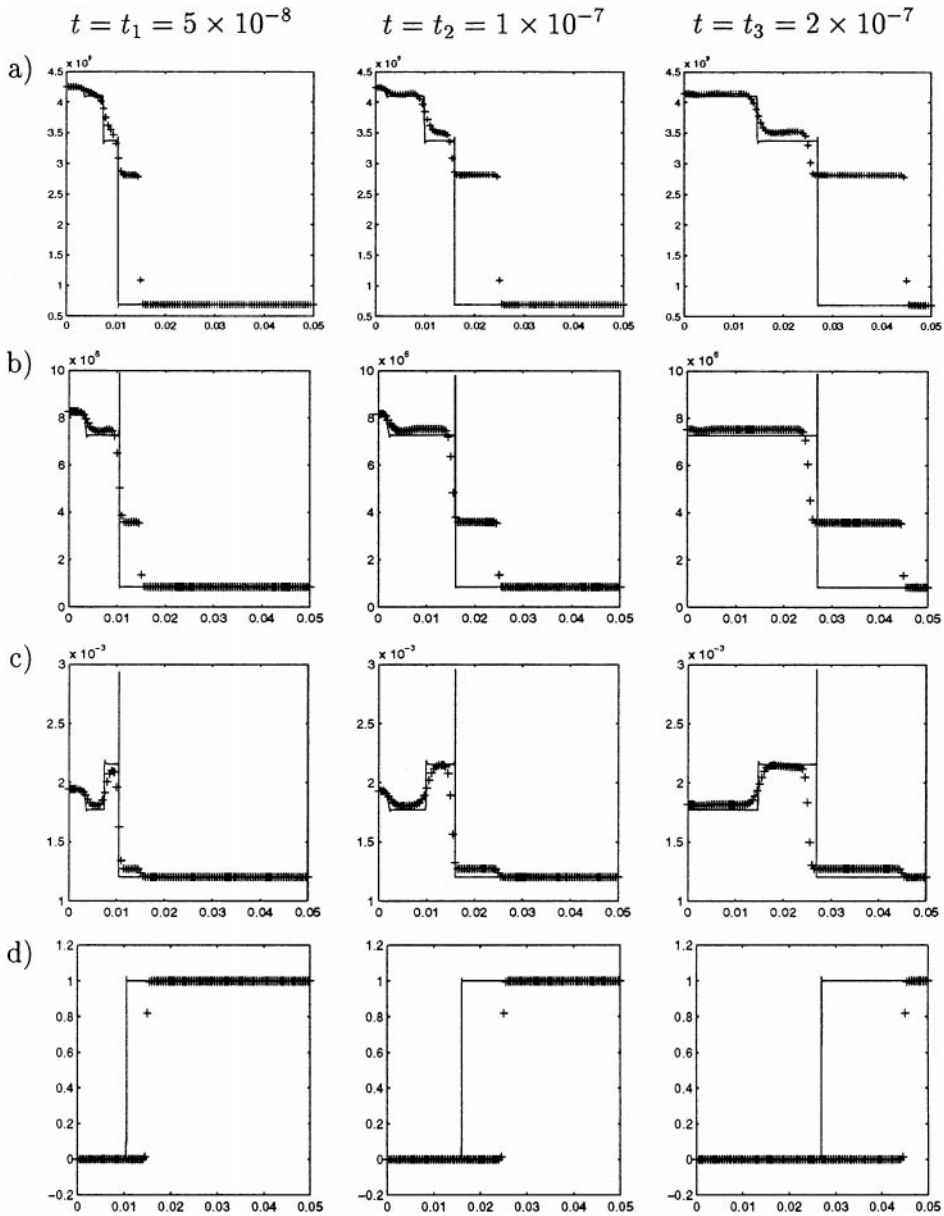

FIG. 14. Numerical results for Example 4.4 using the deterministic method (4.11). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

of both C–J and strong detonations in all the examples presented here, when the numerical discretization does not resolve the reaction scale. The global random projection algorithm (4.6) produces similar results for the first three examples; however, it fails to produce good approximate solutions in Example 4.4.

In practice, we recommend using the local random projection method (4.9) because it is more efficient and works well for all the cases we have considered.

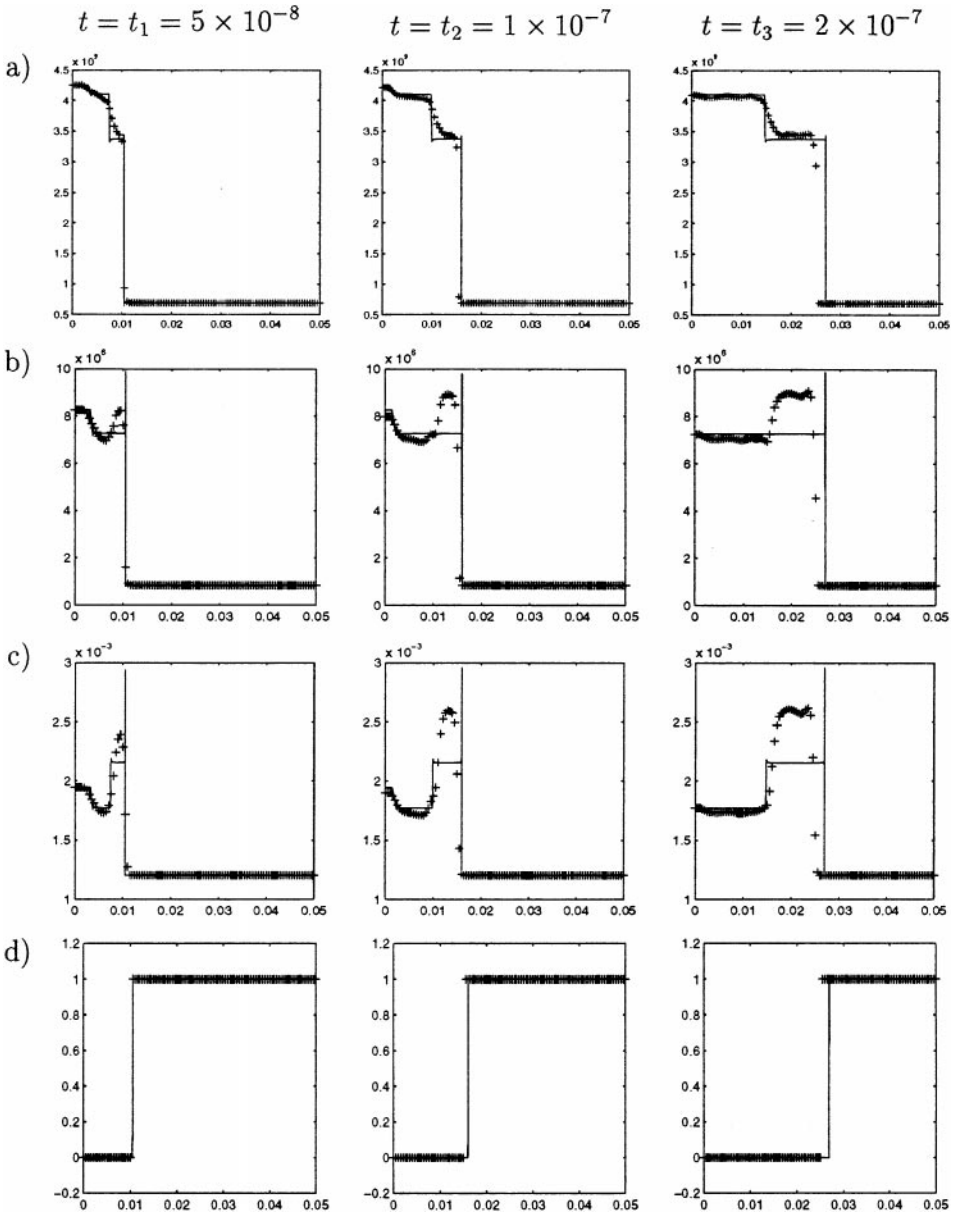

FIG. 15. Numerical results for Example 4.4 using the global random projection method (4.6). $h = 5 \times 10^{-4}$, $k = 5 \times 10^{-10}$. $-$: Exact solutions; $++$: computed solutions. (a) Temperature T; (b) pressure p; (c) density ρ; and (d) z.

5. TWO-DIMENSIONAL DETONATION WAVES

Now we extend the local random projection method to the computation of two-dimensional detonation waves. The equations are

$$U_t + F(U)_x + G(U)_y = \frac{1}{\varepsilon}\Psi(U), \tag{5.1}$$

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ e \\ \rho z \end{pmatrix}, \quad F(U) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (e+p)u \\ \rho uz \end{pmatrix}, \quad G(U) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (e+p)u \\ \rho vz \end{pmatrix}, \quad \Psi(U) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \psi(U) \end{pmatrix}.$$

The dependent variables $\rho(x, y, t)$, $u(x, y, t)$, $v(x, y, t)$, $e(x, y, t)$, and $z(x, y, t)$ are the density, x- and y-velocities, energy, and fraction of unreacted fluid, respectively. The pressure is given by

$$p = (\gamma - 1)\left(e - \rho\frac{u^2 + v^2}{2} - q_0\rho z\right)$$

and the temperature is defined as $T = p/\rho$. The parameters q_0, T_0, γ, and $1/\varepsilon$ are the same as those in one-dimensional detonation waves.

For simplicity, we consider the detonation waves in a two-dimensional channel. Without loss of generality, the initial data are

$$\begin{aligned}(\rho(x, y, 0), u(x, y, 0), v(x, y, 0), p(x, y, 0), z(x, y, 0)) \\ = \begin{cases} (\rho_l, u_l, 0, p_l, 0), & \text{if } x \le \xi(y), \\ (\rho_r, u_r, 0, p_r, 1), & \text{if } x > \xi(y), \end{cases}\end{aligned} \tag{5.2}$$

where $\xi(y)$ is a given function of y. Let $U_{ij}^n = (\rho_{ij}^n, \rho_{ij}^n u_{ij}^n, \rho_{ij}^n v_{ij}^n, e_{ij}^n, (\rho z)_{ij}^n)$ be the approximate solution of U at $(x_i, y_j, t_n) = (ih, jh, nk)$. As in the one-dimensional case, the fractional step method is used. In the first step, the inviscid compressible Euler equations

$$U_t + F(U)_x + G(U)_y = 0 \tag{5.3}$$

are solved by a high-resolution shock capturing method $S_{F,G}(k)$. In the second step, the chemical reaction term

$$\rho_t = 0, \qquad m_t = 0, \qquad n_t = 0, \qquad e_t = 0, \qquad (\rho z)_t = \frac{1}{\varepsilon}\psi(U) \tag{5.4}$$

is solved using the local random projection method.

At any time step, for each j, there is an $l_j(n) = j_n$, j_n an integer, such that

$$z_{ij}^n = \begin{cases} 0, & \text{if } i \le l_j(n), \\ 1, & \text{if } i > l_j(n). \end{cases}$$

Here $l_j(n)$ is the location of the jump for the numerical solution of z at the grid line $y = y_j$ and at time t_n. Since after each time step, the front moves by d grid points, $l_j(n+1)$ is at most d grid points away from $l_j(n)$. Thus one need only check d points near $l_j(n)$. Assume that after the convection step $U^* = S_{F,G}(k)U^n$, for each j, the front at time t_{n+1} is located at $l_j(n+1)$, which can be easily determined from the value of z. Then the following local

projection around the front is used in the second step:

$$\hat{S}_\theta(k): \quad \rho_{ij}^{n+1} = \rho_{ij}^*, \quad m_{ij}^{n+1} = m_{ij}^*, \quad n_{ij}^{n+1} = n_{ij}^*, \quad e_{ij}^{n+1} = e_{ij}^*,$$

For j do

$$\text{Set } l_j(n+1) := l_j(n) - 1, \tag{5.5}$$

$$\text{For } l = l_j(n) - 1,\ l_j(n), \ldots, l_j(n) + d \quad \text{do}: l_j(n+1) = l, \quad \text{if } T_{lj}^* > \theta_n;$$

$$z_{ij}^{n+1} = \begin{cases} 0, & \text{if } i \le l_j(n+1), \\ 1, & \text{if } i > l_j(n+1). \end{cases} \tag{5.6}$$

Then the corresponding algorithm is

$$U^{n+1} = \hat{S}_\theta(k) S_{F,G}(k) U^n. \tag{5.7}$$

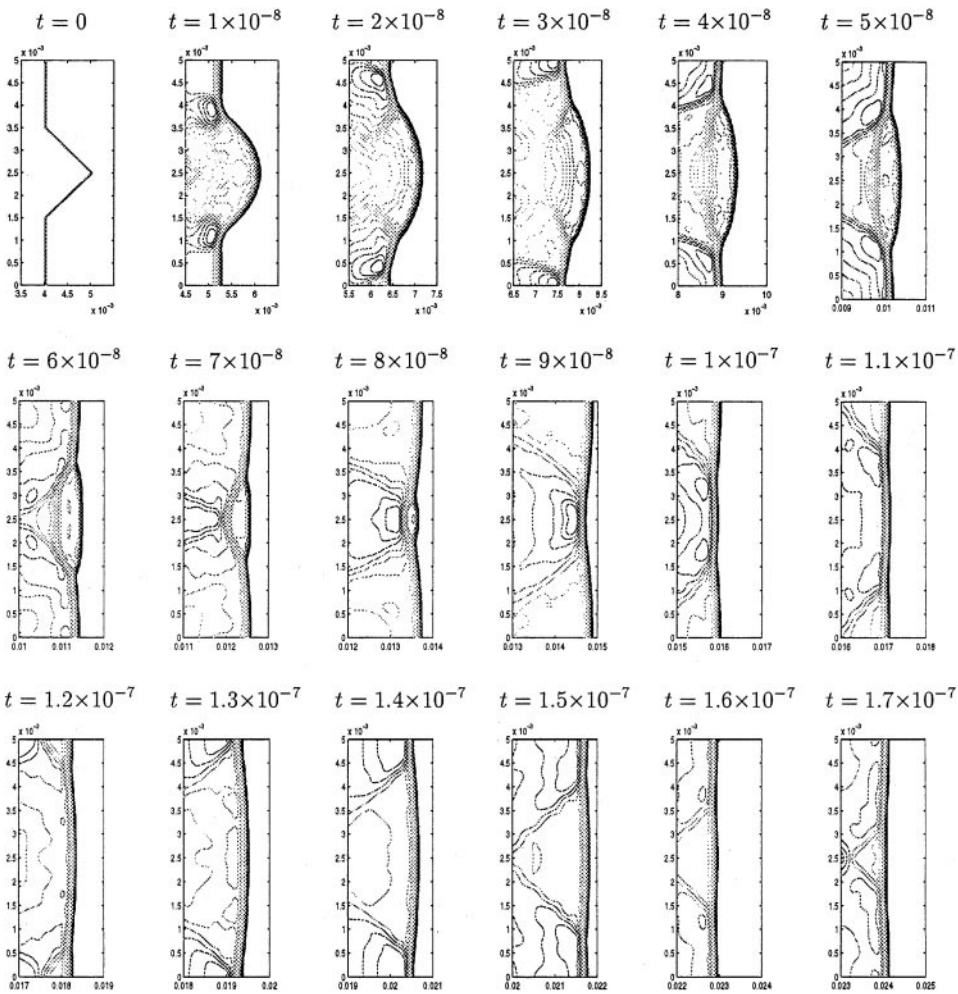

FIG. 16. Numerical results of density for Example 5.1 using the 2D local random projection method. $h = 5 \times 10^{-5}$, $k = 5 \times 10^{-11}$.

The stability condition for this algorithm is still the usual CFL condition determined from the convection term.

Now we will test algorithm (5.7) with a numerical example. We choose $d = 1$ in (5.5)–(5.6) for the following example since the detonation is strong.

EXAMPLE 5.1 (A Two-Dimensional Detonation Wave). We consider the problem (5.1) in a two-dimensional channel, where the upper and lower boundaries are solid walls. We choose the same γ, q_0, ε, and T_0 as in Example 4.1. The initial data in (5.2) are $\rho_r = \rho_0$, $u_r = u_0$, $p_r = p_0$, $\rho_l = \rho_{CJ}$, $u_l = 8.162 \times 10^4 > u_{CJ}$, and $p_l = p_{CJ}$ with ρ_l, u_l, p_l, ρ_{CJ}, u_{CJ}, and p_{CJ} given in Example 4.1.

One important feature of this solution is that the triple points travel in the transverse direction and bounce back and forth against the upper and lower walls, forming a cellular pattern [19].

This problem is solved on $[0, 0.025] \times [0, 0.005]$ with a 501×101 mesh, and

$$\xi(y) = \begin{cases} 0.004 & |y - 0.0025| \geq 0.001, \\ 0.005 - |y - 0.0025| & |y - 0.0025| < 0.001. \end{cases}$$

Thus the mesh size $h = 5 \times 10^{-5}$. The time step is $k = 5 \times 10^{-11}$.

Figure 16 shows the evolution of density contours in time. One can see that the local projection method produces the desired profile for the triple points. In contrast, the triple points, obtained by the usual deterministic method [11] cease to move after some time.

6. CONCLUSIONS

We propose a random projection method for numerical simulation of the hyperbolic conservation laws with stiff source terms arising from chemically reactive flows. Our method consists of two typical steps: solving the homogeneous hyperbolic conservation law by a standard modern shock capturing method and using a project method for the stiff reaction term. In the reaction step, we replace the deterministic ignition temperature with a uniformly distributed random variable in a suitable domain. This random projection method captures the propagation of the detonation waves with the correct speed when the reaction scale is not numerically resolved. For model scalar conservation laws with a stiff source term, we prove that this method, when the van der Corput sampling sequence is used, captures the correct shock speed with first-order accuracy. Extensive numerical examples for one- and two-dimensional scalar problems and one- and two-dimensional detonation waves demonstrate the effectiveness of this novel method.

In the future we will conduct more numerical experiments to validate the applicability of this new method. To conclude, we quote a remark by Chorin [4]: A certain randomness is a property of many real flows, and thus a method which exhibits randomness is not necessarily less desirable than a method which yields fully predictable answers.

ACKNOWLEDGMENT

We thank the referee for valuable suggestions on improving the presentation of the paper.

REFERENCES

1. M. Ben-Artzi, The generalized Riemann problem for reactive flows, *J. Comput. Phys*. **81**, 70 (1989).
2. A. C. Berkenbosch, E. F. Kaasschieter, and R. Klein, Detonation capturing for stiff combustion chemistry, *Combust. Theory Model*. **2**, 313 (1998).
3. A. Bourlioux, A. Majda, and C. Roytburd, Theoretical and numerical structure for unstable one-dimensional detonations, *SIAM J. Appl. Math*. **51**, 303 (1991).
4. A. J. Chorin, Random choice solution of hyperbolic systems, *J. Comput. Phys*. **22**, 517 (1976).
5. A. J. Chorin, Random choice methods with applications to reacting gas flow, *J. Comput. Phys*. **25**, 253 (1977).
6. P. Colella, Glimm's method for gas dynamics, *SIAM J. Sci. Stat. Comput.* **3**, 76 (1982).
7. P. Colella, A. Majda, and V. Roytburd, Theoretical and numerical structure for reacting shock waves, *SIAM J. Sci. Stat. Comput.* **7**, 1059 (1986).
8. R. Courant and K. O. Friedrichs, *Supersonic flow and Shock Waves* (Interscience, New York, 1971).
9. T. Elperin and O. Igra, About the choice of uniformly distributed sequences to be used in the random choice method, *Comput. Methods Appl. Mech. Eng.* **57**, 181 (1986).
10. B. Engquist and B. Sjogreen, Numerical approximation of hyperbolic conservation laws with stiff terms, in *Proc. Third Int. Conf. on Hyperbolic Problems* (Studentlitteratur, Lund, 1991), p. 848.
11. B. Engquist and B. Sjogreen, *Robust Difference Approximations of Stiff Inviscid Detonation Waves*, CAM Report 91-03 (UCLA, 1991).
12. H. Fan, S. Jin, and Z. H. Teng, Zero reaction limit for hyperbolic conservation laws with source terms, *J. Differential Equations*, to appear.
13. J. Glimm, Solutions in the large for nonlinear hyperbolic systems of equations, *Commun. Pure Appl. Math.* **18**, 697 (1965).
14. J. M. Hammersley and D. C. Handscomb, *Monte Carlo Methods* (Methuen, London, 1965).
15. S. Jin and Z. P. Xin, The relaxation schemes for systems of conservation laws in arbitrary space dimensions, *Commun. Pure Appl. Math.* **48**, 235 (1995).
16. R. J. LeVeque and H. C. Yee, A study of numerical methods for hyperbolic conservation laws with stiff source terms, *J. Comput. Phys*. **86**, 187 (1990).
17. A. Majda, A qualitative model for dynamic combustion, *SIAM J. Appl. Math.* **41**, 70 (1981).
18. R. B. Pember, Numerical methods for hyperbolic conservation laws with stiff relaxation. I. Spurious solutions, *SIAM J. Appl. Math.* **53**, 1293 (1993).
19. R. A. Strehlow, The nature of traverse waves in detonation, *Astronaut. Acta* **14**, 539 (1969).
20. H. C. Yee and P. K. Sweby, *Nonlinear Dynamics & Numerical Uncertainties in CFD*, NASA TM 110398, 1998.