Efficient Computation for a Reaction-Diffusion System with a Fast Reaction with Continuous and Discontinuous Initial Data Using COMSOL Multiphysics

Guan Wang, Aaron Churchill, Matthias K. Gobbert, and Thomas I. Seidman

Department of Mathematics and Statistics, University of Maryland, Baltimore County

{wangg1,achurc1,gobbert,seidman}@umbc.edu

Abstract

In a chemical reaction-diffusion system with three species, the interaction between diffusion and a fast reaction between two species creates a transient, sharp, internal interface. Besides the three species model, we use a two component model that is equivalent in the asymptotic limit as the reaction becomes infinitely fast. We use COMSOL Multiphysics for the simulations and use the time-dependent analysis of the general form of the problem for accurate representation of derivatives of the non-linear terms. For a representative continuous initial condition, detailed comparisons of results in one spatial dimension confirm that the two component model is the asymptotic limit of the three species model, but is significantly more numerically efficient. Additional studies in both one and two spatial dimensions confirm that COMSOL Multiphysics obtains the same steady state solution also for several initial conditions with jump discontinuities.

1 Introduction

Our objective is to study the diffusion controlled reactions

$$\mathrm{A} + \mathrm{B} \xrightarrow{\lambda} \mathrm{C}, \qquad \mathrm{A} + \mathrm{C} \xrightarrow{\mu} (*).$$

Here λ and μ are reaction coefficients with the units scaled such that $\lambda \gg \mu = 1$. We denote the concentrations of the three chemical species A, B, and C by $u(x,t)$, $v(x,t)$, and $w(x,t)$ respectively. Standard chemical kinetics then give a three species system of partial differential equations for these concentrations as

$$\left.\begin{aligned} u_t &= u_{xx} - \lambda uv - uw \\ v_t &= v_{xx} - \lambda uv \\ w_t &= w_{xx} + \lambda uv - uw \end{aligned}\right\} \text{ for } x \in (0,1) \text{ and } t \in (0, t_{\mathrm{fin}}]. \tag{1.1}$$

The boundary conditions considered are

$$\left.\begin{aligned} u &= \alpha \\ v_x &= 0 \\ w_x &= 0 \end{aligned}\right\} \text{at } x = 0, \qquad \left.\begin{aligned} u_x &= 0 \\ v &= \beta \\ w_x &= 0 \end{aligned}\right\} \text{at } x = 1. \tag{1.2}$$

This setup represents the case of a membrane, whose length is scaled to the unit interval, between tanks with unlimited supplies of A and B, respectively.

Singular perturbation analysis for the corresponding stationary problem in one spatial dimension is available in [2, 5], which prove the existence of an internal layer with width $\mathcal{O}(\varepsilon)$ and height $1/\mathcal{O}(\varepsilon)$ with scaling $\varepsilon = \lambda^{-1/3}$. This transient three species model in one spatial dimension has already been considered computationally in [3, 6]. Numerical studies on three initial conditions with continuous initial conditions in [6] allow us to conclude that the moving, sharp, internal layer in the transient problem has the same scaling as in the stationary problem. Studies in [3] consider $\lambda = 10^6$ and 10^9 and lead us to conclude that both values are already in the asymptotic regime, which gives strong guidance as to what to expect from the problem in the asymptotic limit. The recent paper [4] introduces a two component model that is equivalent to the above three species model in the asymptotic limit $\lambda \to \infty$, but promises to be significantly more computationally efficient, since it lacks the sharp internal layer present in the three species model. The internal layers become implicit in the two component model, showing up as discontinuities in spatial derivatives for the original species rather than a term λuv in the equation.

The goals of this paper are to provide sufficient evidence that the two component model, denoted as $\lambda = \infty$ for short, can be accurately solved and that it is more efficient than the three species model itself in computing those species. We do this by comparing computational efficiency and accuracy in 1-D with $\lambda = 10^6$, $\lambda = 10^9$, and $\lambda = \infty$ (the latter one using the two component model). We then do a short numerical study to provide evidence that the two component model works well with discontinuous initial conditions. A version of the two component model is then provided for two spatial dimensions and also tested with two discontinuous initial conditions.

2 Derivation of the Two Component Model

Computationally, the greatest difficulty in handling the system (1.1) is the occurrence of $q = \lambda uv$ in each of the equations. Chemical modeling and the rigorous analysis available for the corresponding steady state system in 1-D show that this pointwise reaction rate is very large where relevant — in a narrowly concentrated reaction zone: we expect this to be negligible where one or the other of A, B dominate, but to be significant where the diffusion transports these to meet each other at the interface and react rapidly there. For computation one needs fairly accurate determination of $\int_{\text{loc}} q\,dx$ so one must adequately resolve the q profile — a 'spike' in 1-D, a 'wall' in 2-D, etc. — with the further difficulty that the location of this 'spike' is not known a priori. Since the difference $\mathrm{A} - \mathrm{B}$ and the sum $\mathrm{B} + \mathrm{C}$ are each conserved in the first reaction, this difficult reaction term cancels when one combines the pairs of equations to get

$$\left.\begin{array}{l}(u-v)_t = (u-v)_{xx} - uw \\ (v+w)_t = (v+w)_{xx} - uw\end{array}\right\} \text{ for } x \in (0,1) \text{ and } t \in (0, t_{\text{fin}}]. \tag{2.1}$$

The term q no longer appears explicitly, but the system (2.1) is no longer self-contained: we cannot recover the three species concentrations u, v, w from the two components

$$u_1 := u - v, \qquad\qquad u_2 := v + w \tag{2.2}$$

to determine the second reaction term uw. On the other hand, we always have $u, v \geq 0$ and in the limit $\lambda \to \infty$ we also have the complementarity condition

$$uv \equiv 0$$

(meaning that uv is negligible for large λ, even though, when this is multiplied by $\lambda \gg 1$, q can become large). Thus, when $u_1 > 0$ we must have $u \neq 0$ so $v = 0$ and $u_1 = u$, while $u_1 < 0$ similarly gives $u = 0$ whence $u_1 = -v$. It then becomes possible to take

$$u = u_1^+ := \max\{u_1, 0\}, \qquad v = -u_1^- := -\min\{u_1, 0\}, \qquad w = u_2 + u_1^- \tag{2.3}$$

and so $uw = u_1^+(u_2 + u_1^-) = u_1^+ u_2$. The formulas in (2.2) and (2.3) in effect constitute forward and backward transformations between the three species and the two component models. The corresponding two component model is then

$$\left.\begin{array}{l}u_{1,t} = u_{1,xx} - u_1^+ u_2 \\ u_{2,t} = u_{2,xx} - u_1^+ u_2\end{array}\right\} \text{ for } x \in (0,1) \text{ and } t \in (0, t_{\text{fin}}]. \tag{2.4}$$

We emphasize that this is exactly valid only in the limit, but is approximately correct for large λ. There is no difficulty in obtaining the initial conditions for the two components $u_1 = u - v$ and $u_2 = v + w$ from those given for u, v, w, but we must also adjoin boundary conditions for the two component system. For this, we again use (2.3), now in (1.2), to get

$$\left.\begin{array}{l}u_1 = \alpha \\ u_{2,x} = 0\end{array}\right\} \text{at } x = 0, \qquad \left.\begin{array}{l}u_1 = -\beta \\ u_{2,x} = -u_{1,x}\end{array}\right\} \text{at } x = 1, \tag{2.5}$$

so the complete two component system (2.4)–(2.5) becomes self-contained, although at the price of a rather unusual coupling in the second boundary condition at $x = 1$.

Computationally, the two component model can be expected to be much easier to work with since the solutions are more regular: the derivative discontinuities of the species concentrations just match across the interfaces to make $u_{1,x}$ and $u_{2,x}$ continuous there where $u_x = [u_1^+]_x$, etc., are not. This report explores precisely the relative computational efficiency of the two component model.

3 Numerical Studies in 1-D

The software used for numerical studies is COMSOL Multiphysics version 3.4 with COMSOL Script, under Windows XP Professional Version 2002 Service Pack 3, with an Intel Core 2 CPU 2.13 GHz and 2 GB of RAM. COMSOL is a finite element package designed to handle various real-world physics problems. We use bi-linear nodal finite elements on a uniform mesh of the 1-D domain with N elements, the default BDFk family as ODE solver with default tolerances, and the default linear solver UMFPACK. In order to allow COMSOL to form all derivatives of the coefficient functions analytically, we use the time-dependent analysis of the PDE in General Form in COMSOL.

3.1 Comparison of the Solutions for the Three Species and Two Component Models

For the 1-D problem, let the domain $\Omega = (0, 1)$ and $0 \le t \le 20$, and we consider both $\lambda = 10^6$ and 10^9. We let the initial conditions be of the continuous form

$$u_{\text{ini}}(x) = \begin{cases} 4(0.25 - x)\alpha, & \text{if } 0 \le x \le 0.25, \\ 0, & \text{if } 0.25 \le x \le 0.5, \\ 64(0.50 - x)(x - 0.75)\gamma, & \text{if } 0.5 \le x \le 0.75, \\ 0, & \text{if } 0.75 \le x \le 1, \end{cases}$$

$$v_{\text{ini}}(x) = \begin{cases} 0, & \text{if } 0 \le x \le 0.25, \\ 64(0.25 - x)(x - 0.50)\delta, & \text{if } 0.25 \le x \le 0.5, \\ 0, & \text{if } 0.5 \le x \le 0.75, \\ 4(x - 0.75)\beta, & \text{if } 0.75 \le x \le 1, \end{cases}$$

$$w_{\text{ini}}(x) = 0.$$

Here we pick $\alpha = 1.6$, $\beta = 0.8$, and $\gamma = \delta = 0.25$. For a more comprehensive study of these types of initial conditions, see [6].

Using the transformations for the two component model with $\lambda = \infty$, its initial conditions are

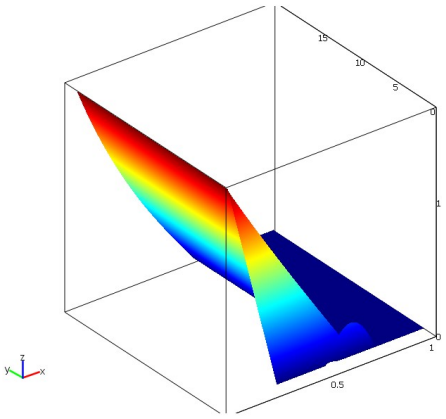
$$u_{1,\text{ini}}(x) = \begin{cases} 4(0.25 - x)\alpha, & \text{if } 0 \le x \le 0.25, \\ -64(0.25 - x)(x - 0.50)\delta, & \text{if } 0.25 \le x \le 0.5, \\ 64(0.50 - x)(x - 0.75)\gamma, & \text{if } 0.5 \le x \le 0.75, \\ 4(0.75 - x)\beta, & \text{if } 0.75 \le x \le 1. \end{cases}$$

$$u_{2,\text{ini}}(x) = \begin{cases} 0, & \text{if } 0 \le x \le 0.25, \\ 64(0.25 - x)(x - 0.50)\delta, & \text{if } 0.25 \le x \le 0.5, \\ 0, & \text{if } 0.5 \le x \le 0.75, \\ 4(x - 0.75)\beta, & \text{if } 0.75 \le x \le 1, \end{cases}$$

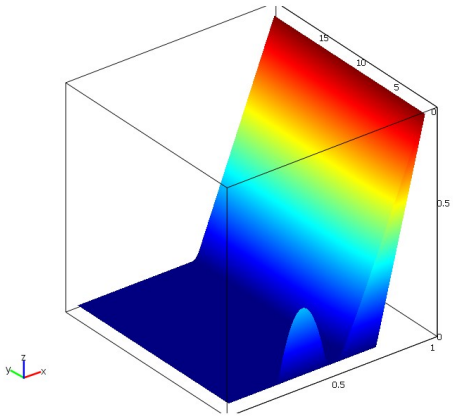
The following figures are simulations for the solutions and interfaces at the final times for the three species model with $\lambda = 10^6$ in Figure 1 and $\lambda = 10^9$ in Figure 2 and the two component model in Figure 3. The waterfall plots in (a), (b), and (c) in each figure represent the concentrations of $u(x, t)$, $v(x, t)$ and $w(x, t)$, respectively, over (x, t). Subplot (d) shows the movement of the interface over time $0 \le t \le 20$, where the curve is the actual interface, and subplot (e) is a zoom of the interesting region in (d) over $0 \le t \le 0.1$. We see in this simulation that after a short period of time, the interface settles down and moves slowly to where the steady state of the interface occurs, around $x^* \approx 0.6$. However, over the first short period of time, the interface moves rapidly. With this plot, we can see the interfaces begin at $x = 0.25$, $x = 0.5$ and $x = 0.75$, and coalesce to a single interface very quickly.

Comparing Figure 1 and Figure 2, we see very similar movement in the interfaces. The waterfall plots also appear incredibly similar. This offers further evidence that as we take $\lambda \to \infty$, we will obtain similar results. For a more in-depth study of initial conditions of this form, see [3].

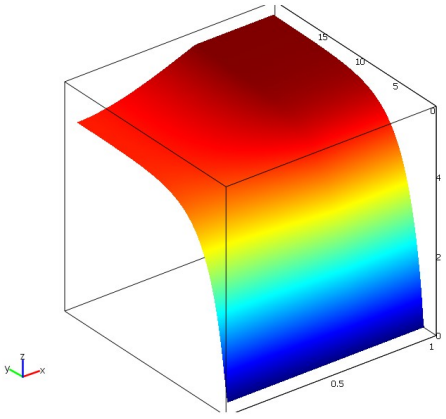
Simulation results for solutions and interface for the two component model with $\lambda = \infty$ are shown as Figure 3. Comparing them with Figures 1 and 2 obtained by the three species model, we observe they are very similar. Thus we can visually confirm that the two component model is reasonable and that there is an asymptotic consistency in the concentrations and interfaces when $\lambda \to \infty$.



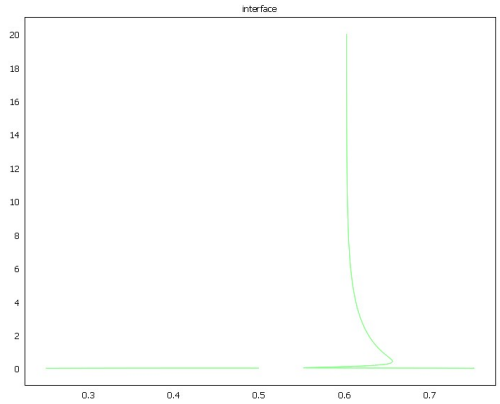
(a) u vs. (x, t)



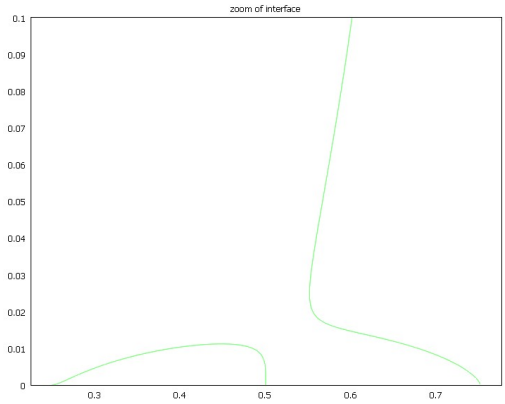
(b) v vs. (x, t)



(c) w vs. (x, t)

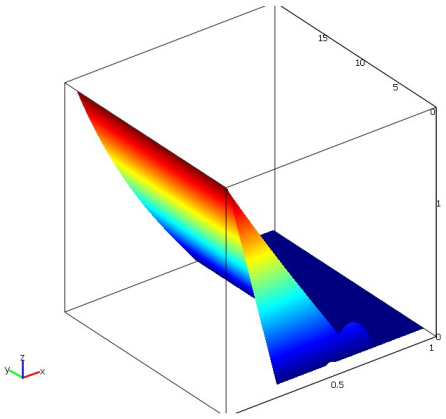


(d) interface plotted on (x, t)

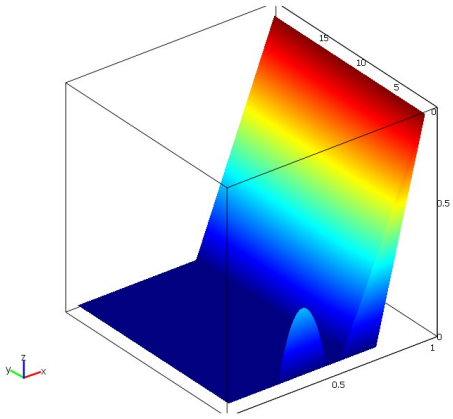


(e) zoomed interface plotted on (x, t)

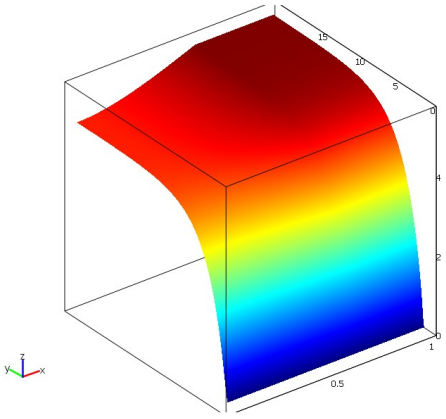
Figure 1: Simulations of the three species model with $\lambda = 10^6$ with continuous initial conditions.



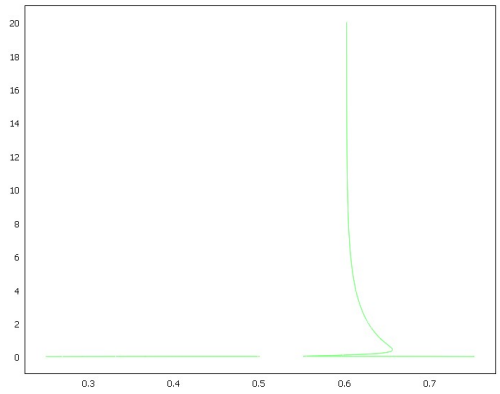
(a) u vs. (x, t)



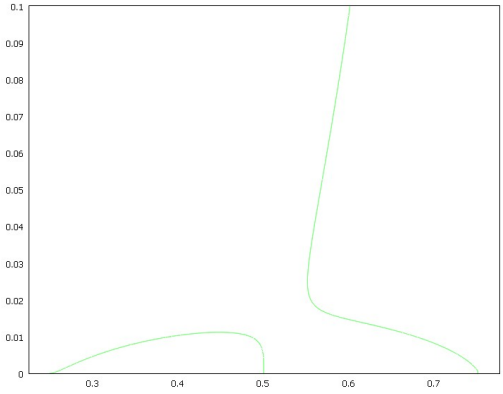
(b) v vs. (x, t)



(c) w vs. (x, t)

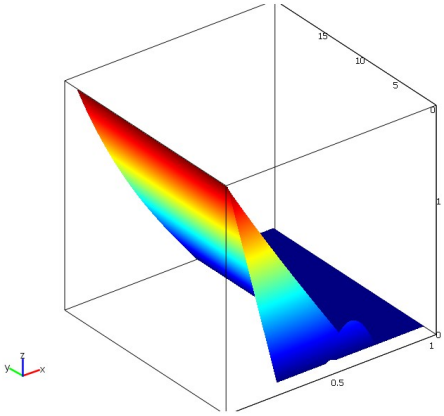


(d) interface plotted on (x, t)

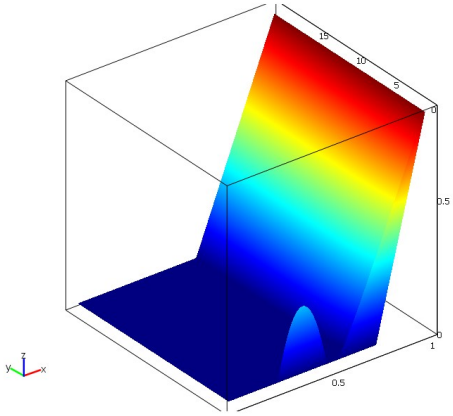


(e) zoomed interface plotted on (x, t)

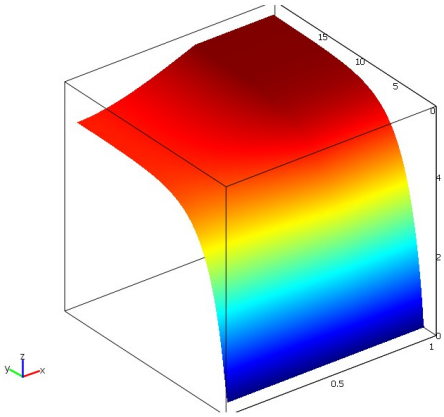
Figure 2: Simulations of the three species model with $\lambda = 10^9$ with continuous initial conditions.



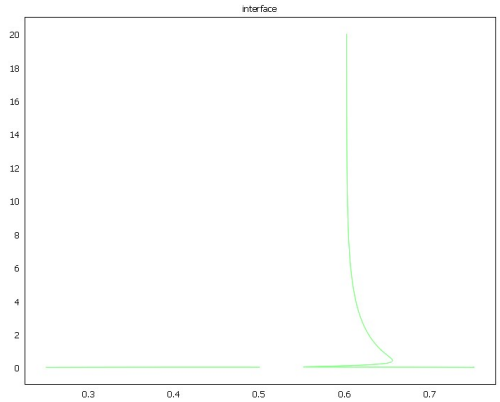
(a) u vs. (x, t)



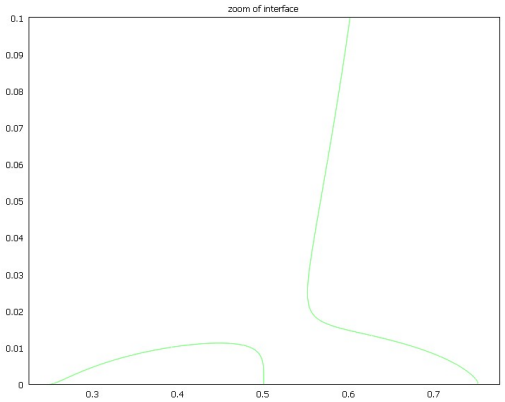
(b) v vs. (x, t)



(c) w vs. (x, t)



(d) interface plotted on (x, t)



(e) zoomed interface plotted on (x, t)

Figure 3: Simulations of the two component model with $\lambda = \infty$ with continuous initial conditions.

3.2 Accuracy and Efficiency Study

As our goal in this report is to verify that the two component model is indeed numerically superior to the three species model, we need to validate this claim quantitatively. So in order to do this, we compare two measures of accuracy and two measures of efficiency, over a variety of finite element meshes with N elements. The first measure of accuracy is the closeness of the location of the interface x^* at the final time of the transient problem to that of the actual steady state interface, which is $x^*_{\text{steady}} = 0.601806640625$; this value was obtained by simulations of the three species steady state problem on a high resolution mesh with $N = 8192$ and will be considered the true value for x^* in the following. The second measure of accuracy is the time t_{co} required for all three interfaces from the initial conditions above to coalesce into a single interface. Our measures of efficiency are fairly standard; we consider the number of time steps taken by the ODE solver and the total computation time in seconds taken by COMSOL Script. The data are summarized in the Table 1.

As for accuracy, the table shows that for corresponding numbers of elements N, the location of the interface at the final time is the same for each of $\lambda = 10^6,\ 10^9$ and ∞. The location x^* changes slightly with increasing mesh resolution, because finer meshes have mesh points that are closer to the true value of x^* than coarser meshes. But in all cases of N, the location determined for x^* is the best or one of the equivalently best values possible on that mesh. Therefore, we conclude that the values for x^* are as accurate as possible for each mesh size. Further, for each of $\lambda = 10^6,\ 10^9$, and ∞, the coalescing times t_{co} are very close for different numbers of mesh resolutions. Further, for a fixed grid size, the value of t_{co} is similar across all of $\lambda = 10^6,\ 10^9$ and ∞. Thus we confirm that not only does it look like the two component model is giving similar results to the three species model for large λ, but indeed it is giving essentially identical results.

Now from a numerical standpoint, the interesting data are in the last two columns of Table 1. As we can see, the number of time steps required for a single λ over different grid sizes does not vary considerably.

Table 1: Summary of accuracy and efficiency data for simulations (a) of the three species model with $\lambda = 10^6$, (b) of the three species model with $\lambda = 10^9$, and (c) of the two component model with $\lambda = \infty$ with continuous initial conditions.

(a) $\lambda = 10^6$				
N	x^*	t_{co}	time steps	computation time
128	0.6015625000000000	0.0112182294336213	552	8.86000000
256	0.6015625000000000	0.0112334958601752	465	8.31200000
512	0.6015625000000000	0.0111585034464878	524	10.93700000
1024	0.6015625000000000	0.0111818581067737	513	14.21900000
2048	0.6020507812500000	0.0111876013688106	516	18.35900000
4096	0.6018066406250000	0.0112575048005446	524	26.57700000
(b) $\lambda = 10^9$				
N	x^*	t_{co}	time steps	computation time
128	0.6015625000000000	0.0111557076128189	1079	21.46800000
256	0.6015625000000000	0.0111679246895327	1340	30.06200000
512	0.6015625000000000	0.0111606912657899	1483	37.84300000
1024	0.6015625000000000	0.0111663934378730	1461	55.17000000
2048	Error: out of memory			
4096	Error: out of memory			
(c) $\lambda = \infty$				
N	x^*	t_{co}	time steps	computation time
128	0.6015625000000000	0.0112999361134998	318	2.89100000
256	0.6015625000000000	0.0112894187459985	297	2.89100000
512	0.6015625000000000	0.0112529236404671	285	3.03100000
1024	0.6015625000000000	0.0112219691583494	275	3.51600000
2048	0.6020507812500000	0.0112475828870521	277	4.98500000
4096	0.6018066406250000	0.0112277130521462	276	8.51500000

However, as we increase λ, the number of time steps required to complete the computation increases considerably. We see that for $\lambda = 10^9$, the solver could not even finish the computations for $N = 2048$ and 4096, because the amount of data per time step is so large combined with the large number of time steps required. Now, when we consider the $\lambda = \infty$ case, we see immediately that both the number of steps taken by the solver and the amount of time are much smaller.

Thus, now that we have confirmed that the two component model is both accurate and efficient, this gives strong motivation for using the two component model whenever possible. Thus, we will now use this for other cases, such as discontinuous initial conditions and the 2-D version of the problem.

3.3 With Discontinuous Initial Conditions

It appears that the location of the interface in the steady state is consistent if replaced by various initial conditions, so long as the values of α and β remain the same. We demonstrate this next with giving initial conditions that are not continuous. Let

$$u_{\text{ini}}(x) = \begin{cases} \alpha, & \text{if } 0 \le x \le 0.25, \\ 0, & \text{if } 0.25 \le x \le 0.5, \\ \gamma, & \text{if } 0.5 \le x \le 0.75, \\ 0, & \text{if } 0.75 \le x \le 1, \end{cases}$$

$$v_{\text{ini}}(x) = \begin{cases} 0, & \text{if } 0 \le x \le 0.25, \\ \delta, & \text{if } 0.25 \le x \le 0.5, \\ 0, & \text{if } 0.5 \le x \le 0.75, \\ \beta, & \text{if } 0.75 \le x \le 1, \end{cases}$$

$$w_{\text{ini}}(x) = 0.$$

We will use the two component model, considering the efficiency, so the corresponding initial conditions are

$$u_{1,\text{ini}}(x) = \begin{cases} \alpha, & \text{if } 0 \le x \le 0.25, \\ -\delta, & \text{if } 0.25 \le x \le 0.5, \\ \gamma, & \text{if } 0.5 \le x \le 0.75, \\ -\beta, & \text{if } 0.75 \le x \le 1, \end{cases}$$

$$u_{2,\text{ini}}(x) = \begin{cases} 0, & \text{if } 0 \le x \le 0.25, \\ \delta, & \text{if } 0.25 \le x \le 0.5, \\ 0, & \text{if } 0.5 \le x \le 0.75, \\ \beta, & \text{if } 0.75 \le x \le 1. \end{cases}$$

Repeating the same numerical process in COMSOL as in the previous section, we obtain numerical results for different numbers of elements. The waterfall plots of concentrations and interface contour lines in Figure 4 follow the same pattern as the previous plots. As we can see in Table 2, the values of x^* are exactly the same, up to mesh size, as in all previous cases. As is to be expected, the coalesce time is different, since the initial conditions are significantly different from before.

Table 2: Summary of accuracy and efficiency data for simulations of the two component model with $\lambda = \infty$ with discontinuous initial conditions.

$\lambda = \infty$ with discontinuous initial conditions				
N	x^*	t_{co}	time steps	computation time
128	0.6015625000000000	0.0071984569556600	381	7.78100000
256	0.6015625000000000	0.0072820951086411	375	8.04700000
512	0.6015625000000000	0.0073262162291875	377	8.78200000
1024	0.6015625000000000	0.0074251203058940	372	8.53100000
2048	0.6020507812500000	0.0071796820107567	392	9.64100000
4096	0.6018066406250000	0.0071809613380149	406	11.90600000

4 Numerical Studies in 2-D

4.1 Problem Statement and Derivation of the Two Component Model

The 1-D problem generalizes directly to 2-D. As confirmed by the 1-D case, the two component model is both accurate and efficient, and so we only consider the two component model in the 2-D case here. The original problem in 2-D case is

$$\left.\begin{array}{l} u_t = u_{xx} + u_{yy} - \lambda uv - uw \\ v_t = v_{xx} + v_{yy} - \lambda uv \\ w_t = w_{xx} + w_{yy} + \lambda uv - uw \end{array}\right\} \text{ for } (x,y) \in (0,1)\times(0,1) \text{ and } t \in (0, t_{\text{fin}}] \tag{4.1}$$

with boundary conditions

$$\left.\begin{array}{l} u = \alpha(y) \\ v_x = 0 \\ w_x = 0 \end{array}\right\} \text{at } x = 0, \qquad \left.\begin{array}{l} u_x = 0 \\ v = \beta(y) \\ w_x = 0 \end{array}\right\} \text{at } x = 1, \qquad u_y = v_y = w_y = 0 \quad \text{at } y = 0, 1, \tag{4.2}$$

where $\alpha(y)$ and $\beta(y)$ are functions of $y \in (0,1)$. Following the same derivation as before, we obtain the following two component model in two spatial dimensions. We observe that it is almost identical to the two component model in one spatial dimension. Observe that as $\lambda \to \infty$, we see that $uv = 0$ and so we let $u_1 = u - v$ and $u_2 = v + w$. Thus the equations of the equivalent two component model are given by

$$\left.\begin{array}{l} u_{1,t} = u_{1,xx} + u_{1,yy} - u_1^+ u_2 \\ u_{2,t} = u_{2,xx} + u_{2,yy} - u_1^+ u_2 \end{array}\right\} \text{ for } (x,y) \in (0,1)\times(0,1) \text{ and } t \in (0, t_{\text{fin}}] \tag{4.3}$$

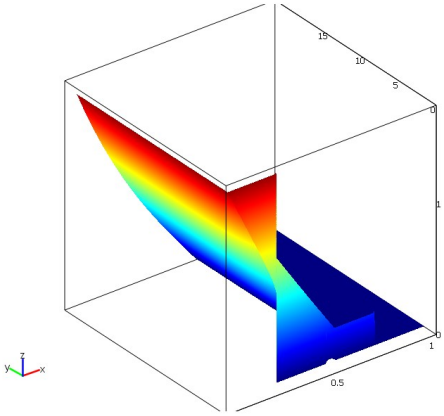
with boundary conditions

$$\left.\begin{array}{l} u_1 = \alpha(y) \\ u_{2,x} = 0 \end{array}\right\} \text{at } x = 0, \qquad \left.\begin{array}{l} u_1 = -\beta(y) \\ u_{2,x} = -u_{1,x} \end{array}\right\} \text{at } x = 1, \qquad u_{1,y} = u_{2,y} = 0 \quad \text{at } y = 0, 1 \tag{4.4}$$

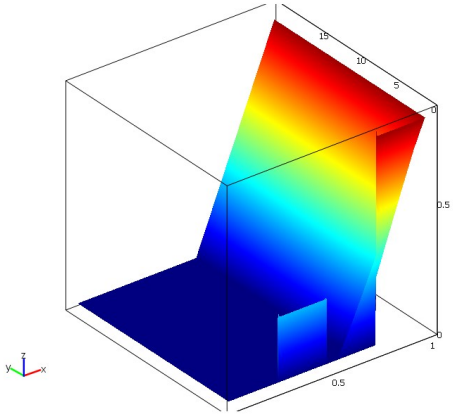
For a more complete derivation and discussions on the 2-D model, see the forthcoming report [1].

4.2 Numerical Studies

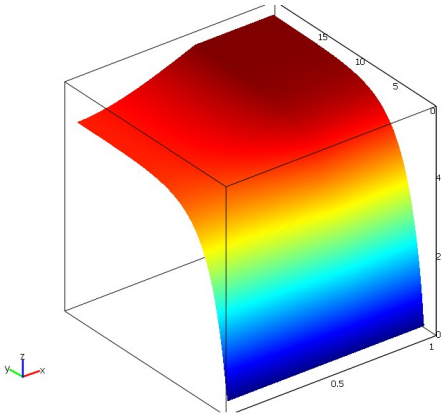
The initial conditions in the 2-D case can become very complicated. To test the model, we want to ensure that the problem has a known steady state solution. To this end, we let $\alpha(y) \equiv \alpha$ and $\beta(y) \equiv \beta$ be constant, then the 2-D problem has the same steady state solution as the 1-D problem at every point y.



(a) u vs. (x, t)



(b) v vs. (x, t)

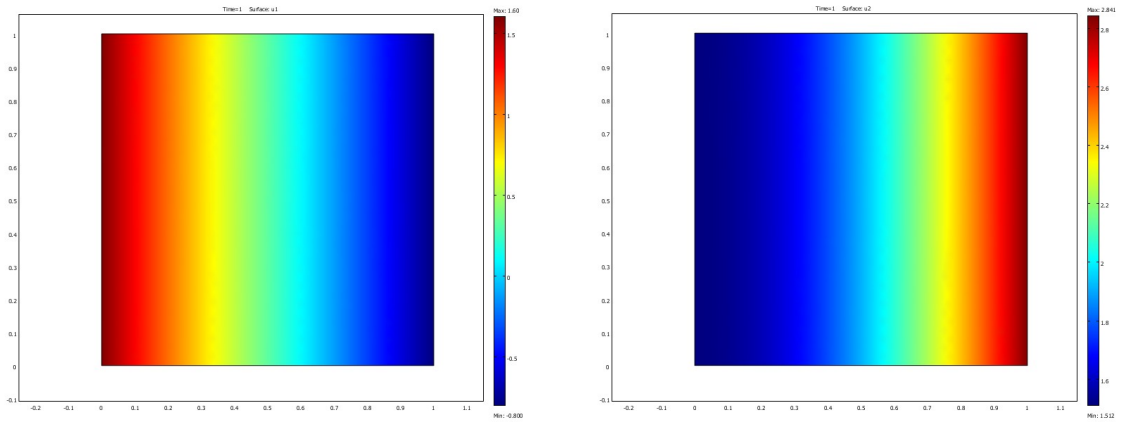


(c) w vs. (x, t)

(d) interface plotted on (x, t)

(e) zoomed interface plotted on (x, t)

Figure 4: Simulations of the two component model with $\lambda = \infty$ with discontinuous initial conditions.



(a) u_1 vs. (x, y) (b) u_2 vs. (x, y)

Figure 5: u_1 and u_2 at the final time for the two component model with jump discontinuities in the initial condition without y dependence.

In a first test, we also choose the initial conditions not to depend on y, thus the simulation should proceed just like in 1-D. Concretely, we pick the initial condition with jump discontinuities in u and v directly generalized from the initial condition with jump discontinuities in 1-D in Section 3.3, namely

$$u_{\text{ini}}(x,y) = \begin{cases} 4(0.25-x)\alpha, & \text{if } 0 \le x \le 0.25, \\ 0, & \text{if } 0.25 \le x \le 0.5, \\ 64(0.50-x)(x-0.75)\gamma, & \text{if } 0.5 \le x \le 0.75, \\ 0, & \text{if } 0.75 \le x \le 1, \end{cases}$$

$$v_{\text{ini}}(x,y) = \begin{cases} 0, & \text{if } 0 \le x \le 0.25, \\ 64(0.25-x)(x-0.50)\delta, & \text{if } 0.25 \le x \le 0.5, \\ 0, & \text{if } 0.5 \le x \le 0.75, \\ 4(x-0.75)\beta, & \text{if } 0.75 \le x \le 1, \end{cases}$$

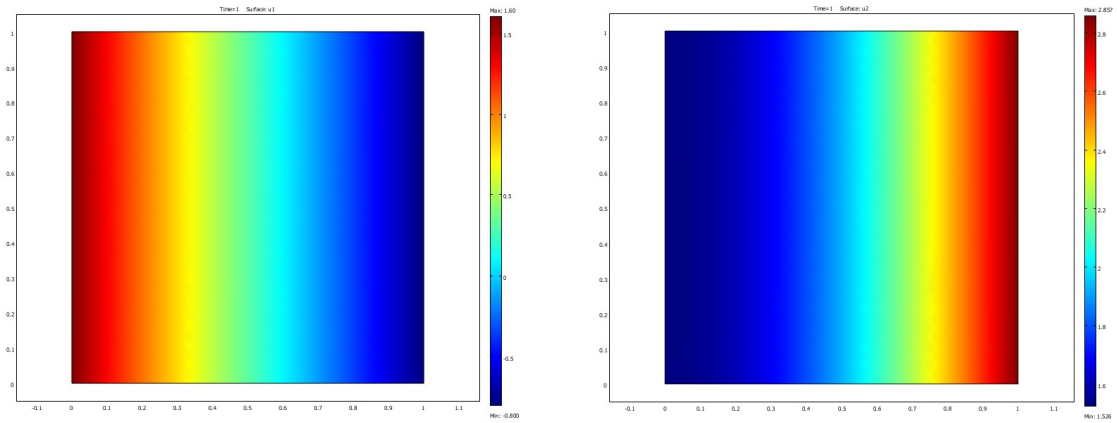
$$w_{\text{ini}}(x,y) = 0.$$

And for the two component model, the initial conditions are

$$u_{1,\text{ini}}(x,y) = \begin{cases} 4(0.25-x)\alpha, & \text{if } 0 \le x \le 0.25, \\ -64(0.25-x)(x-0.50)\delta, & \text{if } 0.25 \le x \le 0.5, \\ 64(0.50-x)(x-0.75)\gamma, & \text{if } 0.5 \le x \le 0.75, \\ 4(0.75-x)\beta, & \text{if } 0.75 \le x \le 1, \end{cases}$$

$$u_{2,\text{ini}}(x,y) = \begin{cases} 0, & \text{if } 0 \le x \le 0.25, \\ 64(0.25-x)(x-0.50)\delta, & \text{if } 0.25 \le x \le 0.5, \\ 0, & \text{if } 0.5 \le x \le 0.75, \\ 4(x-0.75)\beta, & \text{if } 0.75 \le x \le 1. \end{cases}$$

In a very similar process in COMSOL as to the 1-D case, we can find the numerical solution for u_1 and u_2. Figure 5 shows their shape at the final time. For each y, we can see that the shape is quite similar to the final shape in the 1-D case. From Figure 5 (a), we see that the interface $u_1 = 0$ is about at $x^* \approx 0.6$, which is consistent with 1-D case.



(a) u_1 vs. (x, y) (b) u_2 vs. (x, y)

Figure 6: u_1 and u_2 at the final time for the two component model with jump discontinuities in the initial condition with y dependence.

To demonstrate how wild the initial conditions can become and we can still obtain reasonable results, we try the following, quite different initial conditions. They have the same jump discontinuities as the previous initial condition, but with a y dependence in the location of the initial interfaces.

$$u_{\text{ini}}(x,y) = \begin{cases} \alpha, & \text{if } x \le 0.5 \text{ and } (x-0.25)^2 + (y-0.25)^2 > \frac{1}{64}, \\ \gamma, & \text{if } (x-0.75)^2 + (y-0.75)^2 \le \frac{1}{64}, \\ 0, & \text{otherwise,} \end{cases}$$

$$v_{\text{ini}}(x,y) = \begin{cases} \beta, & \text{if } x > 0.5 \text{ and } (x-0.75)^2 + (y-0.75)^2 > \frac{1}{64}, \\ \delta, & \text{if } (x-0.25)^2 + (y-0.25)^2 \le \frac{1}{64}, \\ 0, & \text{otherwise,} \end{cases}$$

$$w_{\text{ini}}(x,y) = 0.$$

The corresponding initial conditions for the two component model are

$$u_{1,\text{ini}}(x,y) = \begin{cases} \alpha, & \text{if } x \le 0.5 \text{ and } (x-0.25)^2 + (y-0.25)^2 > \frac{1}{64}, \\ -\delta, & \text{if } (x-0.25)^2 + (y-0.25)^2 \le \frac{1}{64}, \\ \gamma, & \text{if } (x-0.75)^2 + (y-0.75)^2 \le \frac{1}{64}, \\ -\beta, & \text{if } x > 0.5 \text{ and } (x-0.75)^2 + (y-0.75)^2 > \frac{1}{64}, \end{cases}$$

$$u_{2,\text{ini}}(x,y) = \begin{cases} \beta, & \text{if } x > 0.5 \text{ and } (x-0.75)^2 + (y-0.75)^2 > \frac{1}{64}, \\ \delta, & \text{if } (x-0.25)^2 + (y-0.25)^2 \le \frac{1}{64}, \\ 0, & \text{otherwise.} \end{cases}$$

Letting $\alpha = 1.6$, $\beta = 0.8$, and $\gamma = \delta = 0.25$, we repeat the process as before, and obtain Figure 6 showing the final shape of u_1 and u_2. We observe that they are consistent with the shapes found with the previous initial conditions.

References

[1] Aaron Churchill, Matthias K. Gobbert, and Thomas I. Seidman. Efficient computation for a reaction-diffusion system with a fast reaction in two spatial dimensions using COMSOL Multiphysics. Tech. Rep., UMBC High Performance Computing Facility, University of Maryland, Baltimore County, in preparation.

[2] Leonid V. Kalachev and Thomas I. Seidman. Singular perturbation analysis of a stationary diffusion/reaction system whose solution exhibits a corner-type behavior in the interior of the domain. *J. Math. Anal. Appl.*, vol. 288, pp. 722–743, 2003.

[3] Michael Muscedere and Matthias K. Gobbert. Parameter study of a reaction-diffusion system near the reactant coefficient asymptotic limit. *Dynamics of Continuous, Discrete and Impulsive Systems Series A Supplement*, pp. 29–36, 2009.

[4] Thomas I. Seidman. Interface conditions for a singular reaction-diffusion system. *Discrete and Cont. Dynamical Systems*, to appear.

[5] Thomas I. Seidman and Leonid V. Kalachev. A one-dimensional reaction/diffusion system with a fast reaction. *J. Math. Anal. Appl.*, vol. 209, pp. 392–414, 1997.

[6] Ana Maria Soane, Matthias K. Gobbert, and Thomas I. Seidman. Numerical exploration of a system of reaction-diffusion equations with internal and transient layers. *Nonlinear Anal.: Real World Appl.*, vol. 6, no. 5, pp. 914–934, 2005.