Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **5**, No. 4, pp. 407-422

DOI: 10.4208/aamm.13-13S01
August 2013

Explorations and Expectations of Equidistribution Adaptations for Nonlinear Quenching Problems

Matthew A. Beauregard* and Qin Sheng

Department of Mathematics, Baylor University, TX 76798-7328, USA

Received 14 June 2012; Accepted (in revised version) 8 September 2012

Available online 7 June 2013

Dedicated to Professor Graeme Fairweather on the occasion of his 70^{th} birthday.

Abstract. Finite difference computations that involve spatial adaptation commonly employ an equidistribution principle. In these cases, a new mesh is constructed such that a given monitor function is equidistributed in some sense. Typical choices of the monitor function involve the solution or one of its many derivatives. This straightforward concept has proven to be extremely effective and practical. However, selections of core monitoring functions are often challenging and crucial to the computational success. This paper concerns six different designs of the monitoring function that targets a highly nonlinear partial differential equation that exhibits both quenching-type and degeneracy singularities. While the first four monitoring strategies are within the so-called *primitive* regime, the rest belong to a later category of the *modified* type, which requires the priori knowledge of certain important quenching solution characteristics. Simulated examples are given to illustrate our study and conclusions.

AMS subject classifications: 65K20, 65M50, 35K65

Key words: Degeneracy, quenching singularity, adaptive difference method, arc-length, monitoring function, splitting method.

1 Introduction

Temporal and spatial adaptations have been playing an important role for computing the numerical solution of singular or near singular differential equations. Commonly, adaptations stem from the equidistribution of a particular monitor function [3]. For singular problems, the appropriate choice of a monitor function is not clear, as say for blow-up problems where the monitor function is chosen to minimize the local truncation error. Still the ultimate goal of the employed strategy is to optimize discretization steps for

*Corresponding author.
Email: Matthew_Beauregard@baylor.edu (M. A. Beauregard), Qin_Sheng@baylor.edu (Q. Sheng)

matching key physical properties of solutions, or easing the domain geometric sophistication. Adaptive mechanisms are often achieved through monitoring closely the most sensitive features of the multi-physical system anticipated, such as the velocity of fluids, location of wave fronts, and evidence of potential singularities. In this paper, through a frequently used two-dimensional reaction-diffusion equation of the quenching type, we discuss several effective adaptation designs. The exploration brings to the surface subtle issues in quenching computations and offers a cautionary reminder that a particularly tailored adaptation must be carefully screened prior to employment.

Let Ω be an open unit square. A typical two-dimensional degenerate quenching model can be comprised as

$$\phi(x,y)u_t = \alpha u_{xx} + \beta u_{yy} + f(u), \qquad (x,y)\in\Omega, \qquad t>0, \tag{1.1a}$$

$$u(x,y,t) = 0, \qquad (x,y)\in\Gamma, \qquad t>0, \tag{1.1b}$$

$$u(x,y,0) = u_0(x,y), \qquad (x,y)\in\Omega, \tag{1.1c}$$

where Γ is the boundary of Ω, $\alpha\geq\beta>0$ are constants, and $\phi(x,y)=\phi(y,x)>0$, $(x,y)\in\Omega$. A degeneracy occurs if ϕ diminishes at certain points on Γ. The source term f is highly nonlinear, positive, and approaches infinity as $u\to 1^-$. We adopt the standard nomenclature for quenching, first proposed by [7], that is, the solution u is said to *quench* if the time derivative u_t becomes unbounded in finite time. That time is called the *quenching time*. As discussed in [1, 5], a single point quenching singularity of (1.1a)-(1.1c), if occurs, must locate on the line segment of $y=x$, $0<x<1$. An interesting nuance of higher dimensional quenching problems is that depending on the size *and* shape of the domain the solution may or may not quench. Calculating *critical quenching domains* and times has been a primary focus of numerical and theoretical analysis [1, 4–6, 8, 11, 12, 14–17]. The numerical approaches have contained a mix of uniform and nonuniform grids often employing temporal adaptation solely. It still remains to be seen how to best adapt the spatial grid to improve on the overall numerical accuracy, efficiency, and robustness, let alone the effects of such adaptations on the computation itself.

In quenching phenomena, it has been shown that if a solution quenches at a finite value, then the rate of change function, that is, the temporal or spatial derivative, blows up faster than an exponential rate [5, 13]. This leads to the following procedure for temporal adaptation based on the equidistribution of u_t. We adopt the implicit equation for a new time step in each advancement,

$$\left(u'_{k+1}-u'_k\right)^2\cdot e_j+\left(\tau_k^{(i)}\right)^2=\left(u'_k-u'_{k-1}\right)^2\cdot e_j+\tau_{k-1}^2, \tag{1.2}$$

where u_k, u_{k+1} are numerical solutions at temporal levels k and $k+1$, respectively, and $e_j\in\mathbb{R}^N$ is the jth unit vector, $1\leq j\leq N$. The notation $(u)^p$ means that each of the vector's components is raised to the power p, and the initial step τ_0 is given. The updated temporal step τ_k is taken to be the minimum, that is,

$$\tau_k=\min_{1\leq j\leq N}\tau_k^{(j)}.$$

Therefore, the Eq. (1.2) can be readily reformulated to yield

$$\tau_k^2=\tau_{k-1}^2+\min_i\left\{\left(u_k'-u_{k-1}'\right)^2-\left(u_{k+1}'-u_k'\right)^2\right\},\quad k=1,2,\cdots. \tag{1.3}$$

In practical computations, temporal adaptations such as those via (1.3) are not necessary till a pre-quenching appears. At that time, solutions at multiple time levels prior to quenching are ready to use. Of course if this is not the case, such as with a numerical restart, the temporal step can be initially reduced to accumulate the necessary information to perform a successful temporal adaptation. Therefore solution data at multiple levels is always at ready for future adaptation.

The mechanism to adapt the underlying mesh as quenching is approached is less certain. The standard approach is to equidistribute a certain monitor function, typically the arc-length of the solution or one its derivatives. However, it is not clear to what effect different principles will have on the underlying computation. In this paper, we address two questions. First, what adaptation principle provides the most benefit to tracking the quenching location? Second, what numerical complications arise from the adaptation and how is this manifested in the solution?

The organization of the paper is as follows. The next section details the splitting algorithm as presented in [2]. In Section 3, six different spatial adaptive principles are introduced. The first four we call primitive as they use common target functions such as the solution or its derivatives. The last two we call modified as they use a priori knowledge of the quenching solution and its characteristics. Examples and subsequent discussions are then given in Section 4. Section 5 summarises our conclusions and expectations of this research.

2 Numerical method

The following semidiscretized approximation to the quenching equation (1.1a) on a nonuniform mesh is proposed,

$$\begin{aligned}(u_t)_{i,j}=&\frac{\alpha}{\phi(x_i,y_j)}\left(\frac{2}{h_{i-1}(h_{i-1}+h_i)}u_{i-1,j}-\frac{2}{h_ih_{i-1}}u_{i,j}+\frac{2}{h_i(h_{i-1}+h_i)}u_{i+1,j}\right)\\&+\frac{\beta}{\phi(x_i,y_j)}\left(\frac{2}{h_{j-1}(h_{j-1}+h_j)}u_{i,j-1}-\frac{2}{h_jh_{j-1}}u_{i,j}+\frac{2}{h_j(h_{j-1}+h_j)}u_{i,j+1}\right)\\&+\frac{f(u_{i,j})}{\phi(x_i,y_j)},\quad i,j=1,\cdots,N,\end{aligned} \tag{2.1}$$

where (x_i,y_j), $i,j=0,1,\cdots,N+1$, are the spatial grid points, $0<h_k\equiv x_{k+1}-x_k=y_{k+1}-y_k\ll 1$, $k=0,1,\cdots,N$, and $u_{i,j}(t)$ is an approximation of the exact solution of (1.1a)-(1.1c) at the grid point (x_i,y_j,t). The discretized equations can be placed into a vector form, namely,

$$v'=Pv+Rv+g(v), \tag{2.2a}$$

$$v(0)=v_0, \tag{2.2b}$$

where $v'=dv/dt$, $v(t)=(u_{1,1},u_{2,1},\cdots,u_{N,1},u_{1,2},\cdots,u_{N,N})^\intercal$ based on a natural ordering of components [15], and P and R are $N^2\times N^2$ matrices, that is,

$$P=\alpha B(I_N\otimes T),\quad R=\beta B(T\otimes I_N),$$

where I_N is the $N\times N$ identity matrix, $\otimes$ denotes the Kronecker product, and

$$g(v)=\Big(\frac{f(u_{1,1})}{\phi(x_1,y_1)},\cdots,\frac{f(u_{N,N})}{\phi(x_N,y_N)}\Big)^\intercal. \tag{2.3}$$

The matrix P is block diagonal with tridiagonal blocks while R is block tridiagonal with diagonal blocks. The matrix B is diagonally defined as,

$$B=\text{diag}(B^{(j)}),\quad B^{(j)}=\text{diag}(\psi_{i,j}^{-1})\quad\text{and}\quad \psi_{i,j}=\phi(x_i,y_j),$$

for $i,j=1,\cdots,N$. The matrix T is tridiagonal with upper, main, and lower diagonals of

$$u_k=\frac{2}{h_k(h_k+h_{k-1})},\qquad k=1,\cdots,N-1, \tag{2.4a}$$

$$m_k=-\frac{2}{h_kh_{k-1}},\qquad k=1,\cdots,N, \tag{2.4b}$$

$$l_k=\frac{2}{h_k(h_{k+1}+h_k)},\qquad k=1,\cdots,N-1. \tag{2.4c}$$

The advantage of the semidiscretization is that readily availably time-stepping algorithms can be used. Ultimately this amounts to approximating the matrix exponential of $t(P+R)$. We consider a linearized Peaceman-Rachford splitting method to acquire the fully discretized variable time-step splitting scheme,

$$\begin{aligned}v_{k+1}=&\Big(I-\frac{\tau_k}{2}R\Big)^{-1}\Big(I-\frac{\tau_k}{2}P\Big)^{-1}\Big(I+\frac{\tau_k}{2}P\Big)\Big(I+\frac{\tau_k}{2}R\Big)\Big[v_k+\frac{\tau_k}{2}g(v_k)\Big]\\&+\frac{\tau_k}{2}g\big(v_k+\tau_k\big(Cv_k+g(v_k)\big)\big)+\mathcal{O}(\tau_k^2),\quad k=0,1,\cdots,\end{aligned} \tag{2.5}$$

where $C=P+R$, v_0 is the given initial vector and τ_k is the variable time step to be determined by a suitable adaptation mechanism. For illustrative purposes, a typical solution profile with its time derivative prior to quenching is given in Fig. 1.

Key analysis of this algorithm has been acquired, mainly proving that the method matches the most important feature, that the solution monotonically increases toward a steady state solution or quenches in finite time. Herewith we provide two main results that show if certain criteria are satisfied that the discrete solution monotonically increases and is stable to perturbations in a weak sense. For detailed proofs, the reader is referred to [2].

Theorem 2.1. *(Monotonicity). For any beginning time step $\ell\geq 0$, if*

(i) $\Delta x_j=\Delta y_j=h_j$ *such that* $\min_j\{h_jh_{j-1}\}<4\alpha/13M$ *for all available* j, *where* $M=f(\tau_0 f_0\psi_{\min}^{-1})$;

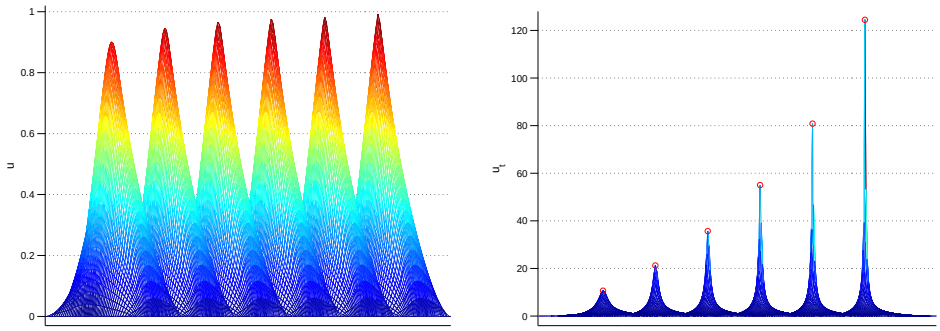
(ii) $\tau_k<4^{-1}\min_j\{h_jh_{j-1}\}\psi_{\min}\alpha^{-1}$ *for all* $k\geq\ell$;

(iii) *either*

1. $Cv_\ell+g(v_\ell)+\tau_\ell PRg(v_\ell)/4>0$ *and* $\tau_k=\tau_\ell$ *for all* $k\geq\ell$, *if uniform temporal steps are used;*
2. $Cv_\ell+g(v_\ell)>0$ *and* τ_k *is sufficiently small, if nonuniform temporal steps are used,*

then the sequence $\{v_k\}_{k\geq\ell}$ *generated by the adaptive splitting method* (2.5) *increases monotonically until unity is exceeded by a component of the solution vector* v_k, *or converges to the steady solution of the problem for both constant and variable* τ_k, $k\geq\ell$.

Theorem 2.2. *Stability. The adaptive splitting method* (2.5) *with the nonlinear term frozen is unconditionally stable in the von Neumann sense.*



(a) (b)

Figure 1: Computed (a) function u and (b) derivative function u_t prior to quenching. Projections of u and u_t are taken from 88, 24, 19, 4, 2 and 1 temporal positions (from the left to right) before quenching. An exponentially graded spatial grid of the size 301×301 is used with a nonlinear source function of $f(u)=1/(1-u)$ and $\phi(x,y)=\sqrt{x^2+y^2}$. While the terminal maximal value of u is 0.99219123737588, the peak value of u_t reaches 124.4559481079604.

3 Spatial adaptation

It is known that as quenching is approached, the solution function u remains bounded while its time derivative grows beyond exponentially fast at the quenching location. In the most general cases, the quenching location is not known. Thus to layout an accurate mesh focused about the quenching location is impossible before the computation. Adaptation provides a remarkable alternative to improve the mesh as the solution advances. Recall Fig. 1. Any mesh adaptation strategy should contain two essential features. First, distribute points near the quenching location as determined by the physics of the discrete solution, mitigating human intervention and construction. Secondly, minimize computational cost by allowing a fewer number of grid points to be used, hence reducing the

size of the involved matrices. Naturally, the methods to be presented attempt to address these features. Computational examples will also be utilized to highlight their degrees of effectiveness.

The adaptation of the grid is calculated via an equidistribution of a certain monitor function, in this case the arc-length of a target function. Substantial work has been done for general adaptations using equidistribution techniques (see [3] and references therein). For (1.1a)-(1.1c) the quenching location occurs along the line $y=x$. Therefore the equidistribution of the grid can be viewed along this line, hence one-dimensional. Define z as the local coordinate along the line $y=x$, hence $z\in[0,\sqrt{2}]$ and $z=\sqrt{2}x=\sqrt{2}y$. Let $w(z,t)$ be a given target function. The arc-length, L, of $w(z,t)$ is

$$L=\int_0^{\sqrt{2}}\sqrt{1+w_z^2}dz.$$

We then look to find grid points that satisfy the equidistribution equations, namely,

$$\frac{L}{N-1}=\int_{z_i}^{z_{i+1}}\sqrt{1+w_z^2}dz,$$

where $z_0=0$, $z_{N+1}=\sqrt{2}$, and $i=1,\cdots,N$. This generates N equations for the grid points $z_1,z_2,\cdots,z_N$ which can be solved using standard quadrature methods. A convenient approach is to solve for z_1 then z_2 and so forth. This minimizes computational storage as the entire system of equations need not be stored at any particular moment. The generation of the new spatial grid points z_i are then mapped to x_i and y_i accordingly.

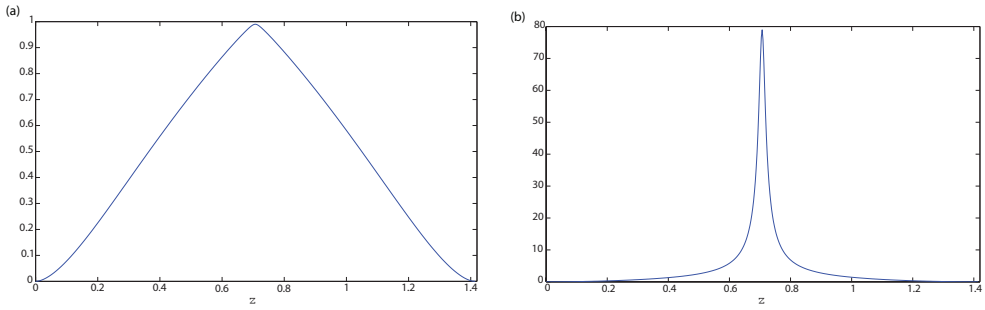
For simplicity, let $U(z,t)=u(x,x,t)$ the solution along the line $y=x$. Common choices of target function are $U(z,t)$, $U_t(z,t)$, $U_z(z,t)$, or $U_{tz}(z,t)$. Each target function is explored here. In the following these four methods will be referred to as *Prim 1, 2, 3*, and *4*, respectively.

We now motivate two additional target functions. It is clear from Fig. 1 that the solution profile remains bounded while the time derivative is growing quickly at the quenching location. The maximum location of both the solution and its derivative are the same. Naturally, it is ideal for a mesh to have more points near the maximum values in hopes of better tracking the quenching location and blow-up phenomena. Do the above principles exhibit this feature? The simple answer is no. To justify this answer, the reader may be inclined to examine the equidistribution of the arc-length of $f(x)=e^{-x^2}$ and $f'(x)$. The problem lies in that the spatial derivative a t the maximum is identically zero.

Knowing that the quenching singularity happens near a maximum and that the solution grows exponentially fast near the singularity leads us to an alternative approach. First, consider the plot of u and u_t along the center line using data from Fig. 1 close to the onset of quenching, this is shown in Fig. 2.

It is clear that Fig. 2(b) exhibits the expected beyond exponential blow-up profile. We elect to fit this curve to a two-sided exponential $y(z)$, namely

$$y(z)=\begin{cases}\kappa_1 e^{\gamma_1 z}, & 0<z<z^*,\\ \kappa_2 e^{\gamma_2 z}, & z^*<z<\sqrt{2},\end{cases}$$



(a) (b)

Figure 2: A plot of (a) u and (b) its derivative function u_t prior to quenching in the z-coordinate. The modified strategies attempt to fit the solution curves to an exponential function in the least squares sense.

where κ_i and γ_i fitting parameters chosen in the least squares sense and z^* is the maximum location. The equidistribution equations then become,

$$\frac{1}{N}\int_0^{z^*}\sqrt{1+\left(\gamma_1\kappa_1 e^{\gamma_1 z}\right)^2}dz=\int_{z_i}^{z_{i+1}}\sqrt{1+\left(\gamma_1\kappa_1 e^{\gamma_1 z}\right)^2}dz,$$
$$\frac{1}{N}\int_{z^*}^{\sqrt{2}}\sqrt{1+\left(\gamma_2\kappa_2 e^{\gamma_2 z}\right)^2}dz=\int_{z_{N+i}}^{z_{N+i+1}}\sqrt{1+\left(\gamma_2\kappa_2 e^{\gamma_2 x}\right)^2}dz,$$

where $z_0=0$, $z_{N+1}=z^*$, $z_{2N+1}=\sqrt{2}$, and $i=1,\cdots,N$. This method does not prevent the arc-length growing unboundedly as quenching is approached, however any monitoring of u_t or its derivatives through an arc-length principle will have a similar issue. A common approach is to clip the solution at a certain value, effectively fixing the number of points near the maximum [17]. Alternatively, a minimum step size controller can be implemented. Here, no such alternative mechanism is used.

Lastly, we propose a similar method that fits the solution to a two-sided exponential. Although Fig. 2(a) does not indicate an exponential pattern the benefit to this approach is that the arc-length will remain bounded as quenching is approached while still distributing points near the maximum location. These last two modified methods will be called *MOD 1* and *MOD 2*, respectively.

The adaptation of the mesh will require the solution to be interpolated onto the new mesh, (X_i,Y_j). For simplicity we consider a bilinear interpolant, that is,

$$\hat{u}(X_i,Y_j)=(1-\hat{\alpha})(1-\hat{\beta})u(x_i,y_j)+\hat{\alpha}(1-\hat{\beta})u(x_{i+1},y_j)+\hat{\alpha}\hat{\beta}u(x_{i+1},y_{j+1})+(1-\hat{\alpha})\hat{\beta}u(x_i,y_j),$$
$$\hat{\alpha}=\frac{X_i-x_i}{h_i},\quad \hat{\beta}=\frac{Y_j-y_j}{h_j},$$

where $u(x_i,y_j)$ is the solution's value at the old mesh location (x_i,y_j), $h_k=x_{k+1}-x_k=y_{k+1}-y_k$, and $\hat{u}$ is the interpolated solution which will be used in the next iteration in the computation.

4 Examples

The aforementioned spatial adaptive principles discussed are now examined computationally. In each example we will present many of the interesting features that illuminate potential advantages and disadvantages of the adaptation. For each, the temporal step is fixed in order to best observe the effect that the spatial adaption has on the computation. The first example is quite intensive and begins by illustrating a subtle problem that arises in the newly formed matrices as a result of the adaptation. It is shown that computation with overly refined and focused grids can potentially ruin the reliability of the computation. Beginning the adaptation early in the computation is a viable option, potentially avoiding this complication. However, a one-time violation is shown to not disrupt the end result. The next example puts many of the newly refined ideas to practice in the context of a computation involving a degeneracy. The quenching location is not known and the refined algorithm is compared to a computation with a fixed grid with a far greater number of grid points.

In each of the computations the nonlinear source and degenerate term will be $f(u)=1/(1-u)$ and $\phi(x,y)=(x^2+y^2)^{q/2}$, respectively [5, 9, 10].

4.1 Example 1

Let q=0, α=0.1 and β=0.1. It is well known that in this case, the quenching location occurs at the center of the domain $(0.5,0.5)$. The quenching time in this situation is computed to be $T^*=0.587354$ using a uniform mesh with 301×301 points and $dt=5\times10^{-7}$.

We consider the six adaptive principles on the numerical solution at the onset of quenching. The solution and the time derivative along the line $y=x$ are shown in Figs. 3(a) and (b) in the iteration just prior to quenching. In addition, the meshes constructed using each of the six adaptive principles are given in Fig. 4(a)-(f).

It is clear that MOD 1 and 2 have advantages over Prim 1 and 2, in that the mesh is focused about the quenching location rather than near other points in the domain. Both Prim 3 and 4 do focus the mesh near the quenching location, however Prim 3 places points near the boundary where the solution is growing the least while Prim 4 is not symmetric about $z=\sqrt{2}/2$, a consequence of the error in the derivative function u_{zz}. These are clear drawbacks to these primitive methods.

The adaptation will require an update of the temporal step in order to satisfy hypothesis (ii) of Theorem 2.1. Of course, the change in the mesh and temporal size will require the interpolation of the current solution onto the new mesh. In addition, a reconstruction of the involved matrices is necessary. How does this reconstruction influence the monotonicity of the solution? Recall, the third hypothesis of Theorem 2.1. This delicately depends on the following theorem.

Theorem 4.1. *Let*

$$Cv_\ell+g(v_\ell)+\frac{\tau_\ell^2}{4}PRg(v_\ell)>0$$

and $\tau_k=\tau_\ell$ *for all* $k\geq\ell$, *where* $\ell\geq 0$ *is any beginning time step. If*

$$\tau_k<\frac{1}{4}\min_{1\leq i\leq N}\{h_ih_{i-1}\}\psi_{\min}\alpha^{-1},$$

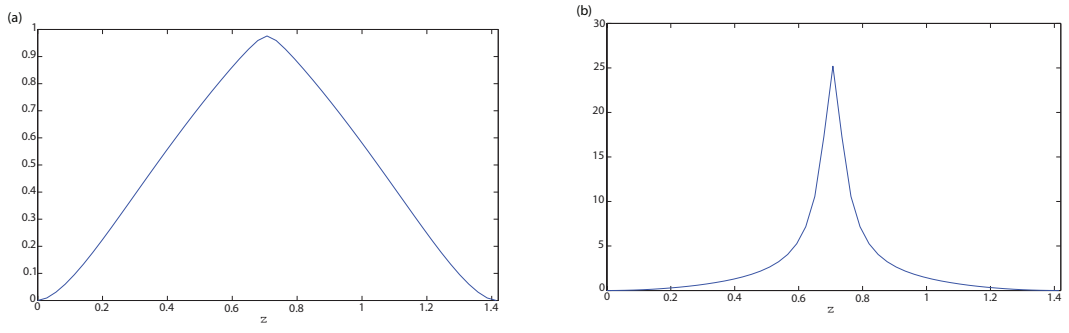
then $v_{k+1}>v_k$ *for all* $k\geq\ell$. *The sequence* $\{v_k\}_{k=\ell}^{\infty}$ *is therefore monotonically increasing.*

The result is a slight variation of that in [5]. In fact, it is straightforward to show that for a null initial condition, $v_\ell=0$, that

$$Cv_\ell+g(v_\ell)+\frac{\tau_\ell^2}{4}PR\,g(v_\ell)>0,$$

hence the theorem can then be proven by using a mathematical induction. Up to the point of the first spatial adaption the discrete solution will have monotonically increased, guaranteed by the theory. Of course, if the algorithm had begun using one of the newly constructed meshes (see Fig. 4(a)-(f)) the solution would monotonically increase. However, in the iteration that adaptation presents itself v_ℓ, the newly constructed interpolated solution, is not zero and contains some interpolation error. Does this error in the solution in conjunction with the update of the matrices violate the criteria of Theorem 4.1? Indeed, Table 1 indicates a violation has occurred by all of the mentioned adaptation principles! Of course, this is a direct result of small interpolation errors, but even with higher order interpolators the problem persists.

What does this violation do to the continued computation? Does the solution still monotonically increase with an exception of a small number of points? We now consider continuing independent computations using each of the newly constructed meshes from Fig. 4(a)-(f). Table 2 shows the ensuing computation's quenching time T_1, in contrast to the quenching time T_2, calculated by beginning the initial computation with the adapted mesh. Notice that the computations involving overly focused meshes have led to a drastic overshoot in the quenching time as compared to the previous result of 0.587354. In



(a) (b)

Figure 3: (a) The solution u and (b) its time derivative u_t along the line $y=x$. Herewith we have $\max u=0.98$ and $\max u_t=25.19$.

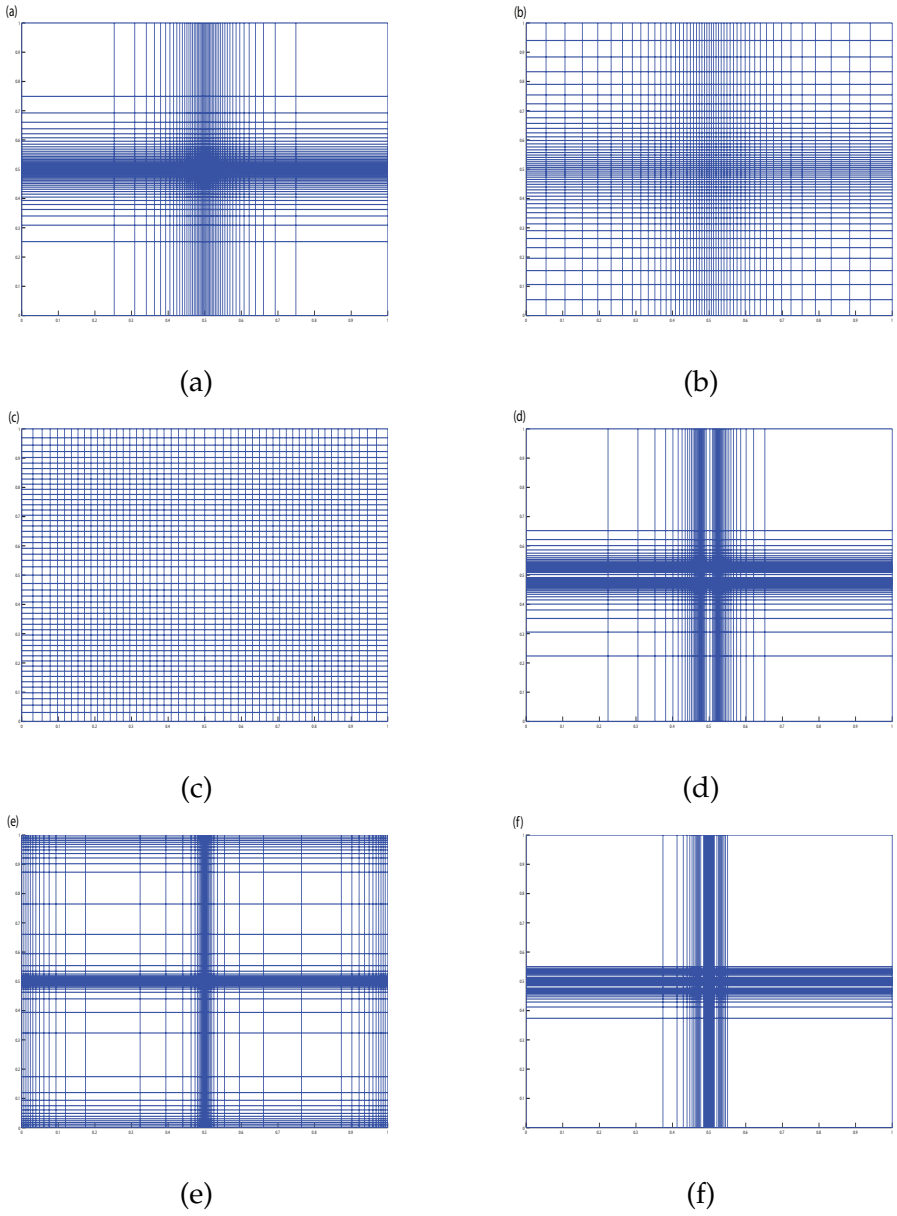


Figure 4: The meshes in the iteration just prior to quenching have been constructed using the 6 adaptive principles; (a-b) MOD 1-2 and (c-f) Prim 1-4, respectively. The meshes generated by Mod 1, Mod 2, Prim 3, and 4 have the smallest grid sizes near the quenching location, while Prim 1 and 2 contains a gap as a result of the stationary point in the spatial derivative. The pictures are given over a 51×51 mesh for the purpose of saving digital memory sizes and to better view the constructed grids.

fact Prim 2 and 4's computation were terminated when the calculation continued 4 times beyond the predicted value! Hence, it is not merely necessary to place more points near the quenching location, rather it must be done in a smooth fashion such that the disparity

Table 1: The minimal value of $Cv_\ell+g(v_\ell)$ in the advancement just prior to quenching, where v_ℓ is the interpolated solution and C is the matrix reconstructed after adaptation in the mesh generated by the six different adaptation mechanisms. To ensure a monotonically increasing discrete solution the value needs to be greater than zero. As τ_k is sufficiently small the additional term $(\tau_\ell^2/4)PRg(v_\ell)$ has been neglected.

Principle	*MOD* 1	*MOD* 2	*Prim* 1	*Prim* 2	*Prim* 3	*Prim* 4
$Cv_\ell+g(v_\ell)$	-4900.2285	-21.9991	-6.55012	-6964.6189	-13757.49860	-107056.6364

Table 2: A comparison of quenching times T_1 and T_2, found using a uniform spatial mesh and adapting the mesh in the iteration prior to quenching and using the adapted mesh as the initial mesh in the computation, respectively. We observe good agreement between T_1 and the computed quenching time $T=0.587354$ found using a uniform mesh, even though there is a violation of Theorem 4.1 in the continued computation. There is a lack of agreement between T_1 and T_2 for MOD 1, Prim 2, and Prim 4 which is attributed to the overly focused mesh near the quenching location resulting in a large disparity in h_k.

Principle	*MOD* 1	*MOD* 2	*Prim* 1	*Prim* 2	*Prim* 3	*Prim* 4
T_1	0.585980	0.587450	0.587335	0.587515	0.587465	0.587480
T_2	1.048925	0.596675	0.594350	N/A	0.591440	N/A

between neighboring grid points is not too large. It is easy to recognize that Figs. 4(b), (c) and (e) offer the least disparity in h_k as compared with (a), (d), and (f). Interestingly, T_1 for all of the adaptations are reasonably close to the previously calculated value. In addition, the discrete solution does monotonically increase with exception of a few points close to the boundary. Hence, a one-time violation is not too damaging, if at all, rather it is continued and early violation of the monotonicity of the solution that renders the calculation useless.

With a large change in the underlying meshes interpolation errors are presumed to be significant. Hence, we may expect that if the adaptation happens early in the computation no violation should occur. Let's consider the same computation maintaining a uniform mesh throughout, but at each iteration we formulate the meshes, calculate the newly interpolated solution, and reconstruct the matrices as if an adaption was going to occur. Then, we calculate $\sigma_\ell=\text{sign}(Cv_\ell+g(v_\ell))$ at each iteration, this is shown in Fig. 5.

Notice that Prim 2 and 3 are always in violation! Similarly, Prim 4 is always in violation outside of a few rare situations later in the computation. This suggests that using u_t, u_z or u_{tz} is not advantageous, at least not without a slight modification, such as only adapting a collection of points, or using a higher order interpolator. In contrast, MOD 1-2 and Prim 1 shows no violation early in the computation. This seems to indicate that if there is only a small movement of grid points in the adaptation that no violation will occur. Hence, early adaptation is necessary to attenuate this effect. Recall, Prim 1 still has the pesky drawback of placing too few points at the maximum location of the solution, while MOD 1-2 were developed in part to correct this. On the other hand, MOD 1 tends too focus the mesh too much which may potentially result in a loss of accuracy away from the quenching location or increase potential interpolation errors. This occurs as a result of the arc-length u_t tending to infinity as quenching occurs. Therefore, MOD 2 presents a nice and surprising alternative.

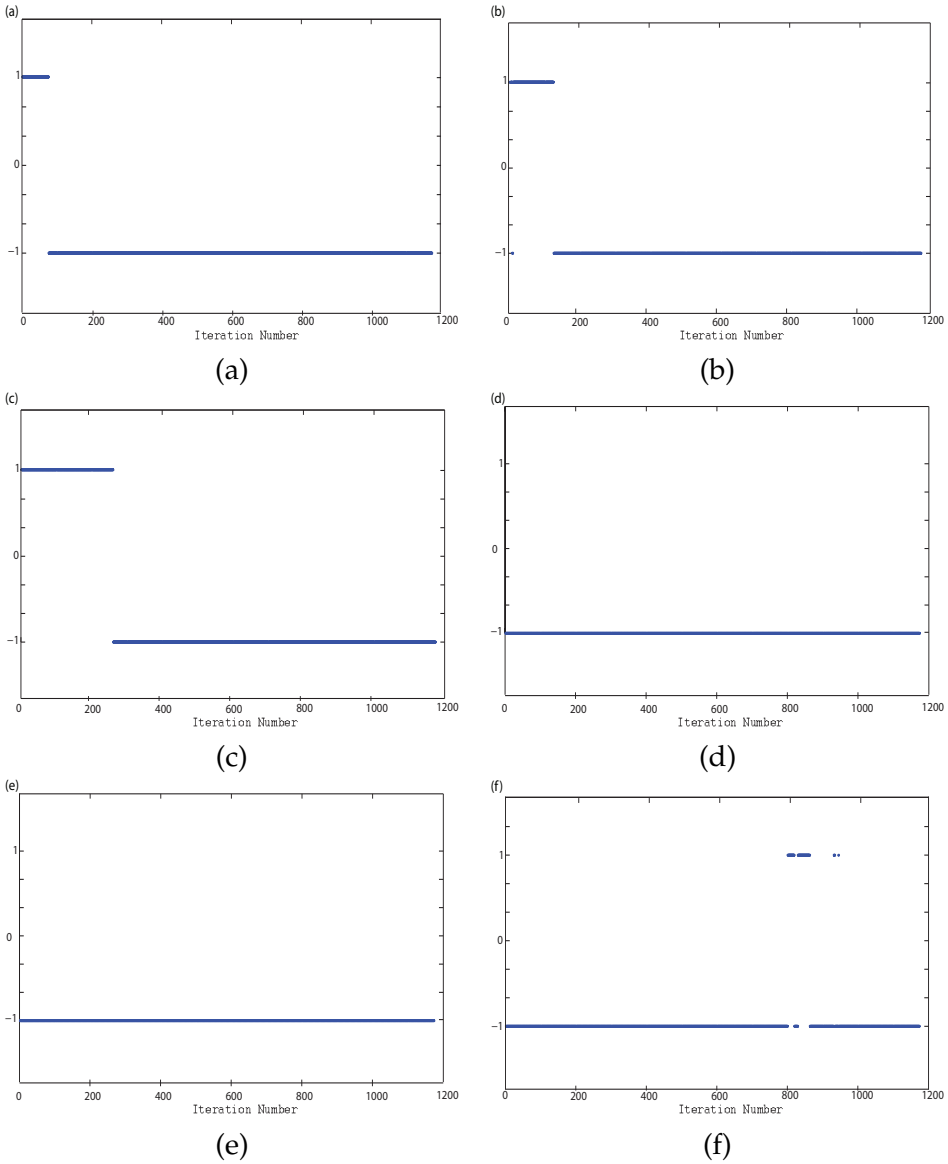


Figure 5: A plot of σ_ℓ at iteration ℓ using the adaptive principles: (a-b) MOD 1-2, (c-f) Prim 1-4, respectively. A negative value indicates a violation in Theorem 4.1, resulting in a loss of monotonicity in the discrete solution.

4.2 Example 2

Let $q=1$ in ϕ. In light of previous example's observations, we compute the quenching time and determine its location using MOD 2 and compare this to that found using a fixed 201×201 uniform mesh. Adaptations are done at iterations for which $\max u=0.01j$ for $j=1,\cdots,99$. This ensures the mesh is adapted early, while maintaining a small number of adaptations. No limitation on the minimum spatial step-size is given. The adaptive

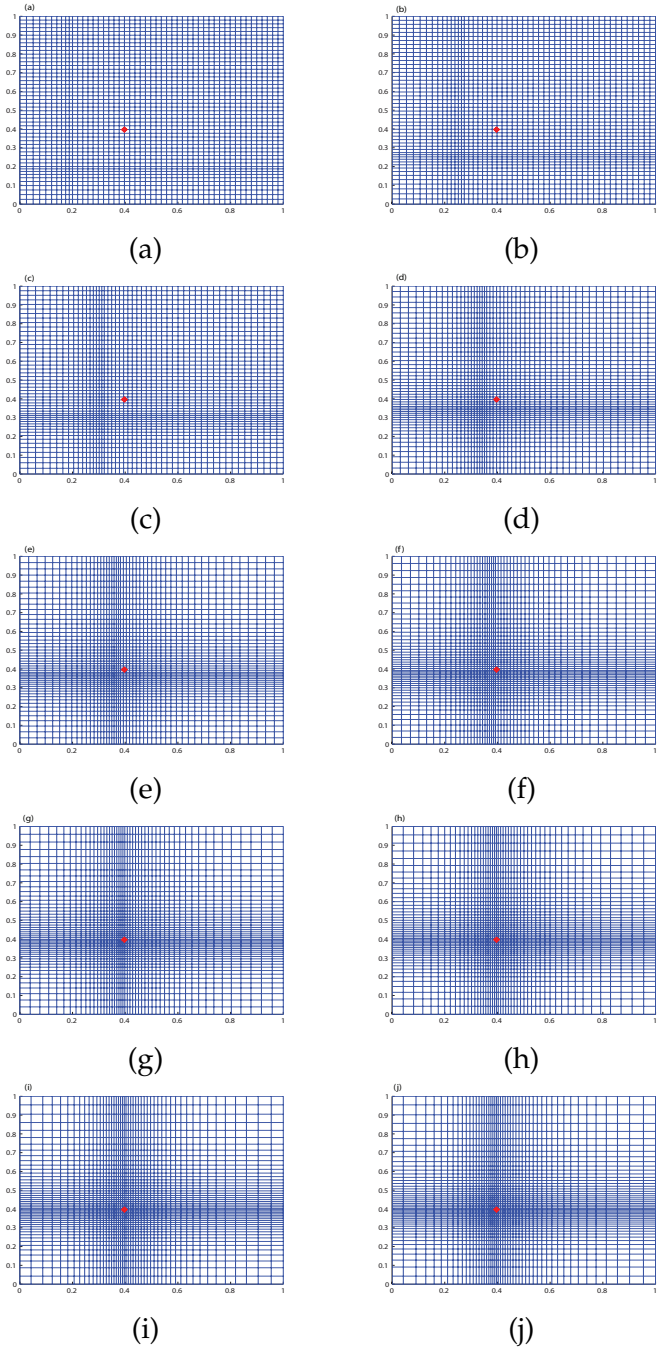


Figure 6: A plot of the constructed meshes using the MOD 2 at iterations for which the $\max_{x,y} u = 0.1j$ for $j = 1,\cdots,9$, and the iteration just prior to quenching (a)-(j), respectively. The adaptations quickly focus about the quenching location indicated by the closed red circle on each plot.

computation begins with a uniform 51×51 mesh.

The quenching time for MOD 2 is calculated to be $T_{m2}=0.392966$ in close agreement to $T_{fixed}=0.390574$ calculated using the fixed fine mesh. Of course, the quenching can be further refined by implementing temporal adaptation. Fig. 6 presents the evolution of the underlying grids at iterations for which $\max_{x,y}u=0.1,\cdots,0.9$, and after the last adaptation.

The quenching location is found at $(0.40328433,0.40328433)$ while the algorithm without refinement stops at $(0.405,0.405)$. Similarly, a computation using a fixed 101×101 grid results in a calculated quenching location of $(0.4\bar{1},0.4\bar{1})$. Hence, the spatial adaptation has allowed the algorithm to compute an extremely accurate quenching location using relatively few grid points, resulting in quicker computation as a consequence of smaller matrices. The minimum step-size is $h_k=0.00653$ and is found at that quenching location with its nearest neighbor along the line $y=x$.

Interestingly, the mesh situates around the quenching location rather quickly, by the time $\max_{x,y}u=0.6$ there is little movement around that location. This is easily recognized on closer examination of Figs. 6(g)-(j) which shows a lack of change in the meshes beyond this stage in the computation. However, the algorithm does continue to slightly focus the mesh about the quenching location in the remainder of the computation.

5 Conclusions

In this paper, we explore spatial adaptation in the context of a nonlinear reaction-diffusion equation of the quenching type. The degeneracy at the boundary and strong nonlinearity that creates the singularity in the solution's time derivative brings about the need for developing appropriate adaptations in both space and time. We employ and compare six different adaptive procedures. Each procedure attempts to establish a more suitable nonuniform grid through the equidistribution of the arc-length of a targeted function. The target functions are comprised from either the solution u or one of its derivative functions.

The first four primitive methods have been used in our recent computations for quenching, however, the presented examples indicate the potential drawbacks. Namely, the monitoring of the arc-length of a singular function can be problematic as the entire grid coalesces near the singularity. This can be avoided by modifying the adaptation itself, such as setting a minimum step-size. However, in doing so, the equidistribution equations will no longer be satisfied. Secondly, directly monitoring the solution or its spatial derivatives is less than desirable. This is because the existence of a stationary point near the quenching location places too few mesh points in the areas where it is needed the most.

In conjunction with these findings we develop two modified methods which follow the solution (MOD 2) or its time derivative (MOD 1) through least squares approximate functions. At first glance, MOD 1 seems to be a natural candidate but as quenching

approaches the target function's arc-length tends to infinite, a similar problem noted previously. MOD 2 is then suggested as a primary alternative to monitoring the solution and developing meshes.

One of the most important features of the quenching solution is that it monotonically increases toward quenching or steady state. The discrete solution will follow this behavior provided certain criteria are satisfied. The adaptations in the mesh cause a refinement in the temporal step, update of the involved matrices, and requires the interpolation of the current solution onto the newly constructed mesh. It is shown that this may result in a violation of the criteria for monotonicity. Although a one-time violation seems to not inherently destroy the algorithm, incessant and early violations can lead to erroneous results. Of course, any initial mesh used to begin the computation with a null initial condition will result in a monotonically increasing solution. However, it is found that using overly refined meshes near the quenching location may lead to inaccuracies in the computed quenching time. These observations indicate that MOD 2 is a suitable choice for such problems, while the alternative methods, particularly involving a function whose arc-length tends to infinite as quenching is approached, should be avoided.

The quenching time and location of a fully degenerate problem is then discussed and shares comparable success to the results found using a higher resolution and fixed mesh. While it is true, that the quenching times are comparable, the extra resolution in the quenching location is quite significant via a proper adaptive algorithm. Moreover, it is seen that the algorithm situates the mesh rather quickly around the quenching location and thereafter little movement is seen in the continued computation. This indicates early rather than later adaption is crucial for accuracy near the quenching location. In particular, the little movement in the mesh means unnecessary interpolation error is avoided later in the algorithm. This suggests a possibility of combining adaptive mechanisms when the solution exhibits certain type of behavior, such as using MOD 1 early in the computation and switching to MOD 2 in the later stages. Hence the mechanism can be better crafted for the solution's particular stage in the computation. In either case, the beneficial features of a fully adaptive method are extremely useful for analyzing quenching problems containing a degeneracy. However, much work is still needed to understand how the underlying interpolation errors and the disparity in the newly constructed mesh have on the numerical accuracy of the computation.

Acknowledgments

The authors would like to thank the referees for the helpful suggestions, for which greatly improved the clarity and readability of the manuscript.

References

[1] M. A. BEAUREGARD AND Q. SHENG, *A semi-adaptive compact splitting method for the numerical*

solution of 2-dimensional quenching problems, Appl. Math. Comput., 218 (2012), pp. 1240–1254.

[2] M. A. BEAUREGARD AND Q. SHENG, *Solving degenerate quenching-combustion equations by an adaptive splitting method on evolving grids*, Comput. Struct., 122 (2013), pp. 33–43.

[3] C. BUDD, W. HUANG AND R. RUSSELL, *Adaptivity with moving grids*, Acta Numer., (2009), pp. 111–241.

[4] C. CHAN, *A quenching criterion for a multi-dimensional parabolic problem due to a concentrated nonlinear source*, J. Comput. Appl. Math., 235 (2011), pp. 3724–3727.

[5] H. CHENG, P. LIN, Q. SHENG AND R. TAN, *Solving degenerate reaction-diffusion equations via variable step peaceman-rachford splitting*, SIAM J. Sci. Comput., 25(4) (2003), pp. 1273–1292.

[6] P. FERREIRA, *Numerical quenching for the semilinear heat equation with a singular absorption*, J. Comput. Appl. Math., 228 (2009), pp. 92–103.

[7] H. KAWARADA, *On solutions of initial-boundary problem for* $u_t = u_{xx} + \frac{1}{1-u}$, Publ. Res. Inst. Math. Sci., 10 (1975), pp. 729–736.

[8] A. KHALIQ AND Q. SHENG, *On the monotonicity of an adaptive splitting scheme for two-dimensional singular reactiondiffusion equations*, Int. J. Comput. Math., 84 (2007), pp.795–806.

[9] C. KIRK AND W. OLMSTEAD, *Blow-up in a reactive-diffusive medium with a moving heat source*, J. Zeitschrift Angew. Math. Phys., 53 (2002), pp. 147–159.

[10] H. LEVINE, *Quenching, nonquenching, and beyond quenching for solutions of some parabolic equations*, Ann. Math. Pure Appl., 4 (1989), pp. 243–260.

[11] K. LIANG, P. LIN, M. ONG AND R. TAN, *A splitting moving mesh method for reaction-diffusion equations of quenching type*, J. Comput. Phys., 215 (2006), pp. 757–777.

[12] K. LIANG, P. LIN AND R. TAN, *Numerical solution of quenching problems using mesh-dependent variable temporal steps*, Appl. Numer. Math., 57 (2007), pp. 791–800.

[13] F. N'GOHISSE AND T. BONI, *Quenching time of some nonlinear wave equations*, Arch. Mathematicum, 45 (2009), pp. 115–124.

[14] N. NOUAILI, *A liouville theorem for a heat equation and applications for quenching*, Nonlinearity, 24 (2011), pp. 797–832.

[15] Q. SHENG, *Adaptive decomposition finite difference methods for solving singular problems*, Frontiers Math. China, 4 (2009), pp. 599–626.

[16] Q. SHENG AND H. CHENG, *An adaptive grid method for degenerate semilinear quenching problems*, Compt. Math. Appl., 39 (2000), pp. 57–71.

[17] Q. SHENG AND A. KHALIQ, Adaptive Method of Lines, Chapter 9, CRC Press, London and New York, 2001.