FULLY DISCRETE ADAPTIVE METHODS FOR A BLOW-UP PROBLEM

CRISTINA BRÄNDLE, PABLO GROISMAN, AND JULIO D. ROSSI

Abstract. We present adaptive procedures in space and time for the numerical study of positive solutions to the following problem,

$$\begin{cases} u_t(x,t) = (u^m)_{xx}(x,t) & (x,t) \in (0,1) \times [0,T), \\ (u^m)_x(0,t) = 0 & t \in [0,T), \\ (u^m)_x(1,t) = u^p(1,t) & t \in [0,T), \\ u(x,0) = u_0(x) & x \in (0,1), \end{cases}$$

with $p, m > 0$. We describe how to perform adaptive methods in order to reproduce the exact asymptotic behavior (the blow-up rate and the blow-up set) of the continuous problem.

1. Introduction

We are interested in developing fully discrete adaptive numerical approximations for the following problem,

$$\begin{cases} u_t(x,t) = (u^m)_{xx}(x,t) & (x,t) \in (0,1) \times [0,T), \\ (u^m)_x(0,t) = 0 & t \in [0,T), \\ (u^m)_x(1,t) = u^p(1,t) & t \in [0,T), \\ u(x,0) = u_0(x) & x \in (0,1). \end{cases} \tag{1.1}$$

In order to have a regular solution, we assume that u_0 is positive and compatible with the boundary condition. Parabolic problems with nonlinear boundary conditions like (1.1) appear in several branches of applied mathematics. They have been used to model, for example, chemical reactions, heat transfer or population dynamics. See [23] and the references therein.

The constant T denotes the maximal time of existence of the solution. If T is infinite we say that the solution is global. However, it is known that, for certain choices of the parameters p and m, the solution may become unbounded in finite time, which means that this solution is

1991 *Mathematics Subject Classification.* 35B40, 35K20, 65M50.

Key words and phrases. numerical blow-up, nonlinear diffusion, nonlinear boundary conditions, adaptive schemes.

defined for $t \in [0, T)$, with

$$\limsup_{t \nearrow T} \|u(\cdot, t)\|_{L^\infty} = \infty.$$

This phenomena is known as blow-up, [16].

Let us summarize the known results concerning solutions of (1.1).

If $m > 1$ the equation is known as *porous medium equation.* It is proved in [16] that for every positive initial data the solution blows up if and only if $p > 1$. Moreover the blow-up rate for increasing in space initial data is given by,

- $\|u(\cdot, t)\|_{L^\infty} \sim (T - t)^{-\frac{1}{p-1}}$ if $1 < p \leq m$,
- $\|u(\cdot, t)\|_{L^\infty} \sim (T - t)^{-\frac{1}{2p-m-1}}$ if $p > m$.

Here and throughout the paper $f(t) \sim g(t)$ means that there exist two positive constants c, C, such that $cg(t) \leq f(t) \leq Cg(t)$.

For this kind of problems it is important to study the set of points where the solution becomes unbounded. To this end the blow-up set is defined as

$$B(u) = \{x : \text{ there exist } \ x_n \to x, t_n \nearrow T, \text{ such that } \ u(x_n, t_n) \to \infty\},$$

In [10, 13, 16] it is proved that the blow-up set for solutions to (1.1) is given by

- $B(u) = [0, 1]$ if $1 < p \leq m$ (global blow-up),
- $B(u) = \{1\}$ if $p > m$ (single point blow-up).

In the case $0 < m \leq 1$ (known as *fast diffusion equation* if $0 < m < 1$ or *heat equation* for $m = 1$) the existence of blow-up solutions depends on m and p. In fact if $p \leq \frac{m+1}{2}$ every solution is global (see [16]) and every positive solution blows up if $p > \frac{m+1}{2}$. In this case the blow-up rate is given by

- $\|u(\cdot, t)\|_{L^\infty} \sim (T - t)^{-\frac{1}{2p-m-1}}$,

and the blow-up set is

- $B(u) = \{1\}$ (single point blow-up).

It is remarkable that the blow-up condition depends strongly on the diffusivity, through the exponent m. Indeed, the condition changes across the critical value $m = 1$, and, if $m > 1$ the properties of the solution change also across the value $p = m$.

Numerical approximations for (1.1) have been already studied in [13], where the authors deal with a semidiscrete numerical approximation in a fixed mesh of size h, keeping the time variable t continuos. They compare the behavior of the continuous problem with the discrete one and observed that significant differences appear. For example, they proved that positive solutions of the numerical problem blow up if and only if $p > 1$. Hence, in the case $0 < m < 1$, if $\frac{m+1}{2} < p \leq 1$ the continuos solutions blow up in finite time, while the numerical approximations are globally defined. Moreover, it is shown that the blow-up rate for the numerical approximation, U, is given by

$$\|U(t)\|_\infty \sim (T_h - t)^{-\frac{1}{p-1}}.$$

Therefore the blow-up rate does not coincide with the expected for the continuous problem when $p > m$. In addition, it is shown that the numerical blow-up set is $[1-Lh, 1]$. The constant L depends only on p and m, hence the numerical blow-up sets concentrates in a neighborhood of $B(u)$ as $h \to 0$.

In view of these facts, we conclude that usual methods with fixed grid are not well suited in order to reproduce the asymptotic behavior of problem (1.1): the blow-up cases do not coincide and even if both, the discrete and the continuous solutions blow up, the blow-up rates can be different.

The aim of this paper is to combine ideas from [2] and [14] to obtain fully discrete adaptive schemes that reproduce the blow-up cases, blow-up rates and blow-up sets. We deal with two different improvements of the previous works simultaneously. On the one hand we extend the adaptive in space ideas of [14] to the case of nonlinear diffusion. On the other hand we use the ideas from [2] to develop a method to integrate the time variable adequately.

We use two different methods in order to define the mesh refinement: the first one adds points near the boundary when the numerical solution becomes large. The second method moves the last point near the boundary when the solution becomes large. This procedure is inspired in the moving mesh algorithms developed in [6, 7, 8, 18] combined with the time step adaptive procedures of [2]. In our case we take advantage of the a priori knowledge of the spatial location of the singularity at $x = 1$ and, instead of moving the whole mesh, we concentrate only the last point near the boundary, leaving the rest of the mesh fixed. This allows us to use a unified approach to analyze rigorously both schemes simultaneously. One advantage of this moving procedure is keeping the size of the problem to be solved constant in time.

Since we are dealing with totally discrete schemes we also need to determine how to integrate in time. First of all we present an explicit Euler method with adaptive time steps. The implicit Euler scheme is briefly mentioned. We also describe how to discretize using second order Runge-Kutta scheme. These discretizations may be combined with any of the two mentioned mesh refinements (adding points or moving point) to get new numerical methods. Higher order schemes can also be considered for the time discretization.

For previous work on numerical approximations of solutions with blow-up we refer to [1, 2, 3, 5, 8, 11, 12, 20, 21, 22] the survey [4] and references therein. In [2, 5, 8, 14], adaptive numerical methods are described for problems with linear diffusion, that is $m = 1$.

To state our main result we need to introduce some notation. Let $u_{h,\lambda}$ be the numerical solution of (1.1). We use h to denote the size of the spatial mesh and λ is the parameter used in the time discretization. We will say that $u_{h,\lambda}$ blows up if there exists a finite time $T_{h,\lambda}$ such that

$$\lim_{t^j \nearrow T_{h,\lambda}} \|u_{h,\lambda}(\cdot, t^j)\|_\infty = +\infty.$$

In this case, we will call $T_{h,\lambda}$ the numerical blow-up time. The main theorem of the paper states that, for an explicit Euler scheme, the numerical solution, $u_{h,\lambda}$, reproduces the properties of the continuous blowing up solution, u.

Theorem 1.1. *Let u be an increasing in x, smooth solution of* (1.1), *that blows up at time T, and $u_{h,\lambda}$ the numerical approximation. Then*

(i) *for every $\tau > 0$, the numerical solution $u_{h,\lambda}$ converges to the continuous one uniformly in $[0, 1] \times [0, T - \tau]$. In fact, there exists a constant $C = C(\tau)$ such that*

$$\|u - u_{h,\lambda}\|_{L^\infty([0,1]\times[0,T-\tau])} \le C(h^2 + \lambda).$$

(ii) *The numerical solution $u_{h,\lambda}$ blows up if and only if $p > 1$ when $m > 1$ and if and only if $p > (m + 1)/2$ when $m \le 1$.*

(iii) *The numerical blow-up rate is given by*

$$\lim_{t^j \nearrow T_{h,\lambda}} (T_{h,\lambda} - t^j)^{\frac{1}{(2p-m-1)}} \|u_{h,\lambda}(\cdot, t)\|_\infty = \Gamma_1, \quad \text{if } p > m,$$

$$\lim_{t^j \nearrow T_{h,\lambda}} (T_{h,\lambda} - t^j)^{\frac{1}{(p-1)}} \|u_{h,\lambda}(\cdot, t)\|_\infty = \Gamma_2, \quad \text{if } p \le m.$$

(iv) *The numerical blow-up time, $T_{h,\lambda}$, converges to the continuous one when h and λ tend to 0,*

$$\lim_{(h,\lambda)\to(0,0)} |T_{h,\lambda} - T| = 0.$$

Moreover, when $m = 1$, there exist $\alpha > 0$ and $C > 0$ such that

$$|T_{h,\lambda} - T| \le C(h^2 + \lambda)^\alpha.$$

(v) *Finally, the numerical blow-up set is given by*

$$B(u_{h,\lambda}) = \begin{cases} \{1\} & \text{if } p > m, \\ [0,1] & \text{if } p \le m. \end{cases}$$

If higher order methods are considered the conclusions of the theorem still holds, but the order of convergence increases.

Organization of the paper: In Section 2 we describe the numerical adaptive procedures. Section 3 is devoted to the proof of the main result, Theorem 1.1. To begin the analysis we prove that numerical approximations converge uniformly in sets of the form $[0,1]\times[0,T-\tau]$. Next we prove that the scheme reproduces the blow-up rate and set. We also prove that the numerical blow-up time converges to the continuous one.

2. Adaptive numerical schemes

To obtain a fully discrete adaptive scheme we proceed in three steps:

(1) space discretization;
(2) time discretization with adaptive time steps;
(3) mesh refinement as time evolves.

2.1. Space discretization.

For the space discretization we propose piecewise linear finite elements with mass lumping. Consider a partition (that can be non-uniform), $x_1, ..., x_{N+1}$ of $[0,1]$ of size h ($h = \max(x_i - x_{i-1})$) and its associated standard piecewise linear finite element space V_h. Let $\{\varphi_j\}_{1\le j\le N}$ be the usual Lagrange basis of V_h. We denote by $u_h(t)$ the semidiscrete approximation obtained by the finite element method with mass lumping (that in this case coincides with a finite differences discretization) and $U(t) = (u_1(t), \dots, u_{N+1}(t))$ the values of the numerical approximation at the nodes x_k, at time t. Writing,

$$u_h(t) = \sum_{j=1}^{N} u_j(t)\varphi_j,$$

a simple computation shows that $U(t)$ satisfies the following system of ordinary differential equations (see [9]):

$$\begin{cases} MU'(t) = -AU^m(t) + BU^p(t), \\ U(0) = u_0^I. \end{cases} \tag{2.1}$$

Here M is the mass matrix obtained with lumping, A is the stiffness matrix and u_0^I is the Lagrange interpolation of the initial datum, u_0.

2.2. Time discretization.

Several methods can be used to discretize the time variable. Nevertheless, we will only focus on explicit and implicit adaptive Euler schemes and a second order explicit Runge-Kutta method. Higher order Runge-Kutta methods can be considered as well.

Euler methods. If we apply an explicit Euler method to (2.1), we get

$$\begin{cases} MU^{j+1} = MU^j - \tau_j A(U^j)^m + \tau_j B(U^j)^p, \\ U^0 = u_0^I, \end{cases}$$

where $U^j = (u_1^j, \dots, u_{N+1}^j)$ is the discretization of $U(t)$ at time t^j.

The time steps are selected, following [2], according to the rule

$$\begin{aligned} \tau_j (u_{N+1}^j)^{2p-m} &= \lambda, \qquad \text{if } p > m, \\ \tau_j (u_{N+1}^j)^{p} &= \lambda, \qquad \text{if } p \leq m, \end{aligned} \tag{2.2}$$

where λ is the parameter of the method that has to be selected small in order to verify the usual stability condition of explicit schemes. Remark that the time step τ_j becomes smaller as the solution increases.

In a similar way we can consider an implicit method by evaluating the discrete laplacian (A) at time t^{j+1}. That is

$$\begin{cases} MU^{j+1} = MU^j - \tau_j A(U^{j+1})^m + \tau_j B(U^j)^p, \\ U^0 = u_0^I. \end{cases}$$

Note that the scheme is not totally implicit since the nonlinear boundary condition u^p is evaluated at time t^j while the discrete laplacian is evaluated a time t^{j+1}. This discretization has the advantage that, if $m = 1$, the explicit evaluation of $(U^j)^p$ avoids the problem of solving a nonlinear system in each step. For $m \neq 1$ one can consider either $B(U^j)^p$ or $B(U^{j+1})^p$, since the nonlinearity is originated by the diffusion term. The time steps τ_j are chosen in the same way as before, but in this case no stability restrictions have to be imposed.

However, we will show that if U^j is large enough, the required stability restriction for the explicit method is always verified. Therefore any implicit method can be used until the stability condition is satisfied, and from that moment on the explicit scheme is recommended.

Runge-Kutta methods. To produce higher order methods we proceed following ideas from [2]. We introduce a new time variable $s = s(t)$ for the semidiscrete problem (2.1), such that $\hat{U}(s) = U(t(s))$. We select s in such a way that $\hat{U}(s)$ is globally defined. Hence any method with fixed time step can be applied to $\hat{U}(s)$. The time rescaling is given by

$$\begin{cases} \dfrac{dt}{ds} = \dfrac{1}{u_{N+1}^{\eta}(t)}, \\ t(0) = 0. \end{cases} \tag{2.3}$$

The parameter η will be chosen below in terms of p and m. Observe that since U is positive, $t(s)$ defines a one to one function. In terms of $(\hat{U}(s), s)$, the system (2.1) takes the form

$$\begin{cases} M\hat{U}'(s) = F(\hat{U}), \\ \hat{U}(0) = u_0^I, \end{cases} \tag{2.4}$$

where F is given by

$$F(\hat{U}) = \frac{1}{\hat{u}_{N+1}^{\eta}(s)} \left(-A\hat{U}^m(s) + B\hat{U}^p(s) \right). \tag{2.5}$$

We can approximate (2.3)–(2.4) by any method with fixed step λ. This allows us to define adaptive in time procedures for (2.1). For example, if we apply an explicit Euler method we obtain

$$\begin{cases} \dfrac{t^{j+1} - t^j}{\lambda} = \dfrac{1}{(u_{N+1}^j)^{\eta}}, \\ t(0) = 0, \end{cases}$$

$$\begin{cases} \dfrac{M\hat{U}^{j+1} - M\hat{U}^j}{\lambda} = \dfrac{1}{(\hat{u}_{N+1}^j)^{\eta}} \left(-A(\hat{U}^j)^m + B(\hat{U}^j)^p \right), \\ \hat{U}(0) = u_0^I. \end{cases}$$

Calling $\tau_j = t^{j+1} - t^j$, these equations can be rewritten in terms of $U^j = \hat{U}^j$ and τ_j as

$$\begin{cases} MU^{j+1} - MU^j = \tau_j \left(-A(U^j)^m + B(U^j)^p \right), \\ \tau_j = \dfrac{\lambda}{(u_{N+1}^j)^{\eta}}. \end{cases}$$

Observe that this coincides with the adaptive explicit Euler method previously described. So, as before we take η as

$$\eta = \begin{cases} 2p - m & \text{if } p > m, \\ p & \text{if } p \leq m. \end{cases}$$

If we apply the half step second order R-K method to (2.3)–(2.4) we obtain

$$\begin{cases} \dfrac{t^{j+1}-t^j}{\lambda} = \dfrac{1}{(\hat{u}^j_{N+1} + \frac{\lambda}{2}F(\hat{U}^j)_{N+1})^\eta}, \\ t(0) = 0, \end{cases}$$

$$\begin{cases} \dfrac{M\hat{U}^{j+1} - M\hat{U}^j}{\lambda} = F(\hat{U}^j + \dfrac{\lambda}{2}F(\hat{U}^j)), \\ \hat{U}(0) = u^I_0. \end{cases}$$

The vector function F is defined as in (2.5). This system can be rewritten in the original variables as

$$\begin{cases} MU^{j+1} - MU^j = \tau_j \left(-A(U^j + \frac{\lambda}{2}F(U^j))^m + B((U^j + \frac{\lambda}{2}F(U^j))^p\right), \\ \tau_j = \dfrac{\lambda}{(u^j_{N+1} + \frac{\lambda}{2}F(U^j)_{N+1})^\eta}. \end{cases}$$

Using these ideas any Runge-Kutta method can be used to approximate (2.4). This provides higher order adaptive in time procedures.

2.3. Mesh refinement.

Once the method to integrate the time variable is selected, it remains to show how to perform the spatial mesh refinement. To this end let us pay special attention to the numerical solution at the point $x_{N+1} = 1$, where the continuous solution develops the singularity.

The numerical mesh refinement that we describe is based on the scale invariance of the problem in the half line with the nonlinear boundary condition placed at $x = 0$, $-(u^m)_x(0,t) = u^p(0,t)$. We mean the following: there exists a self-similar blow-up solution in the half-line of the form $u_S(x,t) = (T-t)^{-\alpha}\psi(\xi)$, $\xi = x(T-t)^{-\beta}$, where $\alpha = 1/(2p-m-1)$, $\beta = (p-m)/(2p-m-1)$. This solution $u_S(x,t)$ gives the behavior near the blow-up time T for solutions of (1.1). Our numerical schemes use this fact to add or move points of the mesh near $x = 1$ trying to reproduce the scaling invariance in the half-line.

Adding points. For the sake of simplicity we will consider the explicit Euler method for time discretization to describe the spatial refinement. We will describe how to proceed just in the case $p > m$ since for this choice of parameters fixed mesh methods do not reproduce the asymptotic behavior. If $p \le m$, as will be shown, the adaptive procedure described here does not refine the mesh and hence the method is a time discretization of the system analyzed in [13].

Writing the equation satisfied by u^j_{N+1} explicitly we obtain the following,

$$u^{j+1}_{N+1} - u^j_{N+1} = \tau_j \left(\frac{2}{h_N^2}((u^j_N)^m - (u^j_{N+1})^m) + \frac{2}{h_N}(u^j_{N+1})^p \right),$$

where $h_N = 1 - x_N$. As we want the numerical blow-up rate to be

$$u^j_{N+1} \sim (T_{h,\lambda} - t^j)^{-\frac{1}{(2p-m-1)}}$$

we impose the last node u^j_{N+1} to satisfy

$$c_1(u^j_{N+1})^{2p-m} \le \frac{u^{j+1}_{N+1} - u^j_{N+1}}{\tau_j} \le c_2(u^j_{N+1})^{2p-m}. \tag{2.6}$$

A straightforward calculation shows that this implies the behavior described above. Recalling (2.2), we get

$$c_1\lambda \le u^{j+1}_{N+1} - u^j_{N+1} \le c_2\lambda. \tag{2.7}$$

On the other hand, imposing (2.6) is equivalent to,

$$c_1(u^j_{N+1})^{2p-m} \le \frac{2}{h_N^2}((u^j_N)^m - (u^j_{N+1})^m) + \frac{2}{h_N}(u^j_{N+1})^p \le c_2(u^j_{N+1})^{2p-m}. \tag{2.8}$$

As the scheme with fixed mesh blows up at the last node, so we have that

$$R(j;h_N) = \frac{\frac{2}{h_N^2}((u^j_N)^m - (u^j_{N+1})^m) + \frac{2}{h_N}(u^j_{N+1})^p}{(u^j_{N+1})^{2p-m}} \to 0,$$

as j increases. Let t^{j_1} be the first time such that $R(j;h_N) \le c_1$ (hereafter we will use t_i to denote t^{j_i}). Namely t_1 is the first time where $R(j_1;h_N)$ does not verify the lower bound c_1. At that moment, we add a point z between x_N and $x_{N+1} = 1$ to the mesh and give the value to $u_{h,\lambda}(z,t_1)$ such that the slope of the line between $(z,(u_{h,\lambda})^m(z,t_1))$ and $(1,(u_{h,\lambda})^m(1,t_1))$ is the same as the slope between $(x_N,(u_{h,\lambda})^m(x_N,t_1))$ and $(1,(u_{h,\lambda})^m(1,t_1))$. Hence we have a new value for the length of the last interval, $[z,1]$,

$$h_{N,1} = 1 - z < h_N = 1 - x_N.$$

In other words, we add the new node z at $1 - h_{N,1}$ in such a way that,

$$\frac{1}{h_N}((u^{j_1}_N)^m - (u^{j_1}_{N+1})^m) = \frac{1}{h_{N,1}}((u^{j_1}_z)^m - (u^{j_1}_{N+1})^m).$$

Therefore, we obtain that $R(j_1; h_{N,1})$ satisfies

$$\begin{aligned}
R(j_1; h_{N,1}) &= \frac{\frac{2}{h_{N,1}^2}((u_z^{j_1})^m - (u_{N+1}^{j_1})^m) + \frac{2}{h_{N,1}}(u_{N+1}^{j_1})^p}{(u_{N+1}^{j_1})^{2p-m}} \\
&= \frac{\frac{2}{h_{N,1}}((u_z^{j_1})^m - (u_{N+1}^{j_1})^m) + 2(u_{N+1}^{j_1})^p}{h_{N,1}\ (u_{N+1}^{j_1})^{2p-m}} \\
&= \frac{\frac{2}{h_N}((u_N^{j_1})^m - (u_{N+1}^{j_1})^m) + 2(u_{N+1}^{j_1})^p}{h_{N,1}\ (u_{N+1}^{j_1})^{2p-m}} \\
&= \frac{h_N}{h_{N,1}} R(j_1; h_N) > c_1.
\end{aligned}$$

With this new mesh $x_1, ..., x_N, z, x_{N+1}$ we have that

$$R(j_1; h_{N,1}) > c_1,$$

and we can continue the method with initial time step τ_j in the new mesh. This gives a solution $u_{h,\lambda}$ that verifies (2.8) in a time interval $[t_1, t_2]$ where at time t_2, the quotient $R(j_2; h_{N,1})$ fails again the lower bound c_1. At that time we have to add another point in the last interval. As before this increases $R(j_2; h_{N,2})$ and we can continue in a new mesh, made up by the old one plus the point that we have added near the boundary $x = 1$. This procedure generates an increasing sequence of times t_i (the times at which we refine the spatial mesh), an increasing sequence of added points accumulating at $x = 1$ and a numerical solution $u_{h,\lambda}(x, t)$ defined for $(x, t) \in [0, 1] \times [0, T_{h,\lambda})$.

Let us point out that, when using this procedure, the system verified by the numerical solution changes whenever we add a new point to the mesh. Indeed, U^j is a solution of

$$\left\{ \begin{array}{l} M_i U^{j+1} = M_i U^j - \tau_j A_i (U^j)^m + \tau_j B_i (U^j)^p, \\ U^0 = u_0^I. \end{array} \right. \tag{2.9}$$

The matrices A_i, B_i and M_i are modified each time the mesh is refined. In fact, as we add a new point to the mesh, an equation is added to the system, and hence we need to add a new row (which takes into account the added point) to the three matrices. The matrix A_i is modified in the elements that correspond to the new point. The matrix B_i has zeros in all of its rows but the last one, where it collects the boundary condition. Nevertheless, the matrix A_i preserves the properties of the original matrix A, i.e.

$$a_{ii} > 0, \qquad a_{ij} \leq 0, \text{ if } j \neq i \qquad \sum_{k=1}^{N(j)+1} a_{kj} \geq 0,$$

where $N(j)$ is the dimension of U^j. For simplicity, we will keep the original nomenclature of the nodes, i.e

$$U^j = \overbrace{(u_1^j, u_2^j, \ldots, u_N^j, u_{N+1}^j)}^{N(j)+1} = (u_1^j, u_2^j, \ldots, u_{N(j)}^j, u_{N(j)+1}^j),$$

and we will assume that the mesh is fixed and uniform, whenever we can do it. If this is not the case, we will mention it explicitly. Under these conditions, we obtain the following discretization for the original problem (1.1)

(2.10)
$$\begin{cases} \dfrac{u_1^{j+1} - u_1^j}{\tau_j} & = \dfrac{2((u_2^j)^m - (u_1^j)^m)}{h^2}, \\ \dfrac{u_i^{j+1} - u_i^j}{\tau_j} & = \dfrac{(u_{i-1}^j)^m - 2(u_i^j)^m + (u_{i+1}^j)^m}{h^2}, \ 2 \leq i \leq N, \\ \dfrac{u_{N+1}^{j+1} - u_{N+1}^j}{\tau_j} & = \dfrac{2((u_N^j)^m - (u_{N+1}^j)^m)}{h^2} + \dfrac{2(u_{N+1}^j)^p}{h}, \end{cases}$$

and the initial datum $u_i^0 = u_0(x_i)$.

It remains to be more concrete on the election of the sequence h_i and the constants c_1, c_2. If we take,

$$c_1 = \Gamma_1^{-(2p-m-1)}(2p-m-1)^{-1},$$

and h_i and c_2 such that

$$\frac{h_i}{h_{i+1}} \to 1, \quad \text{and} \quad c_2(t_i) = \frac{h_i}{h_{i+1}} R(j, h_i), \tag{2.11}$$

we get

$$c_1 \leq R(j, h_{i+1}) \leq \frac{h_i}{h_{i+1}} R(j, h_i) \sim c_1.$$

In fact, if h_i is chosen such that

$$h_{i+1}(u_{N+1}^j)^{p-m} = \frac{2}{c_1} - \frac{A}{u_{N+1}^j}, \tag{2.12}$$

we have

$$
\begin{aligned}
c_2 &= \frac{h_i}{h_{i+1}} R(j_i, h_i) = \frac{\frac{2}{h_{i+1}}((u_z^{j_i})^m - (u_{N+1}^{j_i})^m) + 2(u_{N+1}^{j_i})^p}{h_{i+1}\ (u_{N+1}^{j_i})^{2p-m}} \\
&= \frac{\frac{2}{h_{i+1}}((u_z^{j_i})^m - (u_{N+1}^{j_i})^m) + 2(u_{N+1}^{j_i})^p}{(u_{N+1}^{j_i})^p \left(\frac{2}{c_1} - \frac{A}{u_{N+1}^{j_i}}\right)} \\
&= \frac{\frac{2}{h_{i+1}}((u_z^{j_i})^m - (u_{N+1}^{j_i})^m)}{(u_{N+1}^{j_i})^p \left(\frac{2}{c_1} - \frac{A}{u_{N+1}^{j_i}}\right)} + c_1 \left(\frac{1}{1 - \frac{c_1 A}{2u_{N+1}^i}}\right) \\
&\le c_1 \left(\frac{1}{1 - \frac{c_1 A}{2u_{N+1}^i}}\right) \to c_1.
\end{aligned}
$$

Observe that

$$
\frac{\tau_j}{h_{i+1}} \le C\lambda \frac{(u_{N+1}^j)^{p-m}}{(u_{N+1}^j)^{2p-m}} = \frac{C\lambda}{u_{N+1}^j}.
$$

Hence usual stability conditions for explicit schemes are verified if u_{N+1}^j is large.

In addition we will show that this selection of h_i makes

$$
h_i = 1 - x_N \sim C(T_{h,\lambda} - t^j)^\beta,
$$

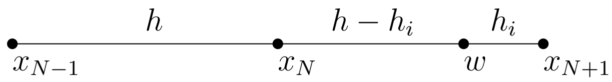
hence the method reproduces both the asymptotic behavior and the spatial structure of the continuous solution near the blow up time.

Observe that if $p \le m$, $R(j, h_N)$ is bounded from below and the method keeps the mesh fixed.

Moving point. An alternative procedure to refine the mesh near the singularity is to move a point towards $x_{n+1} = 1$. This method is inspired in the moving mesh methods described in [8].

We begin with a mesh composed by two types of nodes, a uniform mesh of size $h = 1/N$ and a node placed between $x_N = (N-1)h$ and $x_{N+1} = 1$, that we are going to move when appropriate. Let us call $0 = x_1 < .... < x_N < x_{N+1} = 1$ the fixed mesh and w the moving node, $x_N < w < x_{N+1}$.

As before, let us consider an explicit Euler procedure for the time discretization and let us use the mesh composed by the x_i and the w together for the space discretization. If we denote by z the numerical



Moving point method

solution at the moving node w, we arrive to the following system

$$\begin{cases} \dfrac{u_1^{j+1} - u_1^j}{\tau_j} & = \dfrac{2\left((u_2^j)^m - (u_1^j)^m\right)}{h^2}, \\ \dfrac{u_i^{j+1} - u_i^j}{\tau_j} & = \dfrac{(u_{i-1}^j)^m - 2(u_i^j)^m + (u_{i+1}^j)^m}{h^2}, \ 2 \le i \le N-1 \\ \dfrac{u_N^{j+1} - u_N^j}{\tau_j} & = \dfrac{2\left((u_{N-1}^j)^m (h-h_i) - (u_N^j)^m (2h - h_i) + (z^j)^m h\right)}{h(h-h_i)(2h-h_i)} \\ \dfrac{z^{j+1} - z^j}{\tau_j} & = \dfrac{2\left((u_N^j)^m (h_i) - (z^j)^m h + (u_{N+1}^j)^m (h - h_i)\right)}{(h-h_i) h h_i} \\ \dfrac{u_{N+1}^{j+1} - u_{N+1}^j}{\tau_j} & = \dfrac{2\left((z^j)^m - (u_{N+1}^j)^m\right)}{h_i^2} + \dfrac{2(u_{N+1}^j)^p}{h_i}, \end{cases}$$

and the initial data $u_i^0 = u_0(x_i)$.

The *moving point method* uses the ideas developed for the *adding points method* to move the point when $R(j, h_N) \le c_1$. When the constant c_1 is reached the node w is moved towards $x_{N+1} = 1$ choosing the new length $h_i = 1 - w$ in the same way as in the adding points method. This new value makes $R(j, h_{N,1}) > c_1$ and the method continues until the constant c_1 is reached again.

Let us remark that the criteria that we use to modify the mesh is the same in the adding points method and in the moving point method. This allows us to make a unified approach in the course of the proofs contained in the following section.

3. Proof of the main results

We may now proceed with the proof of Theorem 1.1. We will do it with full details for the adding points scheme with an explicit Euler method for the time variable and briefly point out the main differences with the other schemes described in Section 2. We state first some preliminary results.

Lemma 3.1. *If U^0 is increasing, then the numerical solution U^j is increasing for every j. Hence it satisfies $\max_i u_i^j = u_{N+1}^j$, for every j.*

Proof. As U^0 is increasing then $u_{i+1}^0 > u_i^0$, let us see that this holds for every $j > 0$. Assume not, then there exists a first positive j_0 and an index i such that $u_{i+1}^{j_0} \le u_i^{j_0}$. From the equations satisfied by u_{i+1}^j and u_i^j we have that at j_0 it holds $u_{i+1}^{j_0+1} - u_{i+1}^{j_0} - (u_i^{j_0+1} - u_i^{j_0}) > 0$, a contradiction that proves the result. □

From now on we will assume that the discrete initial data U^0 is increasing, $u_i^0 \le u_{i+1}^0$ for all $1 \le i \le N+1$.

Definition 3.2. We will say that Z^j is a supersolution (resp. a subsolution) of (2.10) if verifies

$$\begin{cases} MZ^{j+1} \ge MZ^j - \tau_j A(Z^j)^m + \tau_j B(Z^j)^p, \quad \text{(resp. } \le) \\ Z^0 \ge z_0^I, \quad \text{(resp. } \le). \end{cases}$$

Lemma 3.3. *Let $\overline{U}^j$ be a supersolution of* (2.10) *and $\underline{U}^j$ a subsolution, such that $\underline{U}^0 < \overline{U}^0$. If the time step verifies*

$$\tau_j < \frac{Ch_i^2}{\|\overline{U}^j\|_\infty^m}, \tag{3.1}$$

then $\underline{U}^j < \overline{U}^j$.

Proof. Let us define $Z^j = \overline{U}^j - \underline{U}^j$. By an approximation argument we can assume that we have strict inequalities in (2.10). Then Z^j is a solution of the following system

$$\begin{cases} \dfrac{MZ^{j+1} - MZ^j}{\tau_j} > -A((\overline{U}^j)^m - (\underline{U}^j)^m) + B((\overline{U}^j)^p - (\underline{U}^j)^p), \\ Z^0 > 0. \end{cases}$$

If the Lemma is false, then there exist a first time t^{j+1} and a node x_i, such that $z_i^{j+1} \leq 0$ and $z_i^j > 0$. Then we have,

$$\begin{aligned}
z_i^{j+1} &> z_i^j - \frac{\tau_j}{m_i}\left(\sum_{k=1}^{N+1} a_{ki}((\overline{u}_i^j)^m - (\underline{u}_i^j)^m) + b_i((\overline{u}_i^j)^p - (\underline{u}_i^j)^p)\right) \\
&= z_i^j\left(1 - \frac{C\tau_j}{m_i} a_{ii}\left((\overline{u}_i^j)^m - (\underline{u}_i^j)^m\right)\right) \\
&\quad + \frac{\tau_j}{m_i}\left(\sum_{k\neq i} a_{ki}\left((\overline{u}_i^j)^m - (\underline{u}_i^j)^m\right) z_k^j + b_i((\overline{u}_i^j)^p - (\underline{u}_i^j)^p)\right) \\
&> z_i^j\left(1 - \frac{C\tau_j}{m_i} a_{ii}\|\overline{U}^j\|_\infty^m\right) \geq z_i^j\left(1 - \frac{C\tau_j}{h_i^2}\|\overline{U}^j\|_\infty^m\right) \geq 0,
\end{aligned}$$

a contradiction. □

Corollary 3.4. *If $\overline{U}^j$ is a supersolution of* (2.10)*, such that*

$$\|\overline{U}^j\|_\infty \leq C u_{N+1}^j,$$

then Lemma 3.3 *holds for all j, if $p > m$. If $p \leq m$, the lemma holds as long as* (3.1) *is valid.*

3.1. Convergence of the numerical schemes.

We are now ready to establish an uniform convergence result: for any $\tau > 0$ we want that $u_{h,\lambda} \to u$ (when $h, \lambda \to 0$) uniformly in $[0,1]\times[0,T-\tau]$. This is a natural requirement since on such an interval the continuous solution is regular.

Lemma 3.5. *Let $u(x,t) \in C^{4,2}([0,1]\times[0,T-\tau])$ be a positive solution of* (1.1) *and let $u_{h,\lambda}$ its discrete approximation obtained by any of the adaptive schemes described in Section* 2*. Then there exists a constant C depending on τ such that,*

$$\|u - u_{h,\lambda}\|_{L^\infty([0,1]\times[0,T-\tau])} \leq C(h^2 + \lambda).$$

Proof. In the course of the proof we will consider an uniform mesh. We are able to do this since in the time interval $[0, T-\tau]$ both, the numerical and the continuous solution, are bounded if h is small enough.

If we rewrite the system (2.10) in terms of $Z = (U^j)^m$, we obtain

$$\left\{\begin{array}{ll}
\dfrac{(z_1^{j+1})^{1/m} - (z_1^j)^{1/m}}{\tau_j} = \dfrac{2(z_2^j - z_1^j)}{h^2}, & \\
\dfrac{(z_i^{j+1})^{1/m} - (z_i^j)^{1/m}}{\tau_j} = \dfrac{(z_{i-1}^j - 2z_i^j + z_{i+1}^j)}{h^2}, & 2 \leq i \leq N, \\
\dfrac{(z_{N+1}^{j+1})^{1/m} - (z_{N+1}^j)^{1/m}}{\tau_j} = \dfrac{2(z_N^j - z_{N+1}^j)}{h^2} + \dfrac{2(z_{N+1}^j)^{p/m}}{h}, &
\end{array}\right.$$

and the initial datum $z_i^0 = (u_0(x_i))^m$. Let $v_i^j = u^m(x_i, t^j)$, with u the solution of the continuous problem (1.1). The error function

$$e_i^j = z_i^j - v_i^j$$

satisfies, for $2 \le i \le N$,

$$\begin{aligned}
&\frac{1}{m}\xi^{(1-m)/m}\frac{e_i^{j+1}-e_i^j}{\tau_j}\\
&\quad= \frac{(e_{i-1}^j - 2e_i^j + e_{i+1}^j)}{h^2} - \frac{1-m}{m^2}\zeta^{(2-m)/m}\frac{v_i^{j+1}-v_i^j}{\tau_j}|\xi-\eta|\\
&\qquad+C_1(h^2+\lambda)\\
&\quad\le \frac{(e_{i-1}^j - 2e_i^j + e_{i+1}^j)}{h^2} - \frac{1-m}{m^2}\zeta^{(2-m)/m}\frac{v_i^{j+1}-v_i^j}{\tau_j}(v_i^{j+1}-v_i^j)|e_i^{j+1}|\\
&\qquad+C_1(h^2+\lambda),
\end{aligned}$$

where ξ is an intermediate value between z_i^{j+1} and z_i^j, η is between v_i^{j+1} and v_i^j and ζ is an intermediate value between ξ and η. Taking into account that there exist constants, c and C, such that $c \le z_i^j \le C$ for every $j \in [0, j_0]$ we have,

$$\frac{e_i^{j+1}-e_i^j}{\tau_j} \le \frac{C_1(e_{i-1}^j - 2e_i^j + e_{i+1}^j)}{h^2} + C_2|e_i^{j+1}| + C_3(h^2+\lambda). \tag{3.2}$$

Making analogous calculations for the first and the last nodes, we get

$$\frac{e_1^{j+1}-e_1^j}{\tau_j} \le \frac{C_1(e_2^j - e_1^j)}{h^2} + C_2|e_1^{j+1}| + C_3(h^2+\lambda), \tag{3.3}$$

and

$$\begin{aligned}
\frac{e_{N+1}^{j+1}-e_{N+1}^j}{\tau_j} \quad &\le \frac{C_1(e_N^j - e_{N+1}^j)}{h^2} + C_2|e_{N+1}^{j+1}|\\
&\qquad + \frac{C_3}{h}\kappa^{(p-m)/m}|e_{N+1}^j| + C_4(h^2+\lambda),
\end{aligned} \tag{3.4}$$

where κ is an intermediate value between z_{N+1}^j and v_{N+1}^j, and hence κ is also bounded.

Now we use a comparison argument. The error e_i^j is a subsolution of a discretization of the problem

$$\begin{cases} e_t = e_{xx} + e + C(h^2+\lambda) & (x,t) \in (0,1)\times[0,T-\tau), \\ e_x(0,t) = 0 & t \in [0,T-\tau), \\ e_x(1,t) = Ce(1,t) & t \in [0,T-\tau), \\ e(x,0) = C(h^2+\lambda) & x \in (0,1). \end{cases} \tag{3.5}$$

Since solutions of problem (3.5) are bounded by

$$|e(x,t)| \le C\exp\{CT\}(h^2+\lambda),$$

we can construct a supersolution $\overline{e}_i^j$ of (3.2)–(3.4), the system verified by e_i^j, by taking $\overline{e}_i^j = e(x_i, t^j)$. Therefore we obtain a similar bound for the error function e_i^j. Arguing similarly with $-e_i^j$ we obtain

$$|e_i^j| \le C\exp\{CT\}(h^2+\lambda), \quad t^j \in (0, T-\tau],$$

and this proves the convergence of the method. □

3.2. Numerical blow-up.

We prove now that solutions of the numerical problem (2.10) blow up in the same range of exponents as the solutions of the continuous problem do.

Let us recall, for notational purposes

$$t^j = \sum_{k=0}^{j} \tau_k.$$

Observe that since, for each j, the maximum of the numerical solution is attained at the last node, in order to have a numerical solution that blows up in finite time $T_{h,\lambda}$, we have to see that the series converge and that $u_{N+1}^j \to \infty$ as j increases.

Lemma 3.6. *The numerical solution $u_{h,\lambda}$ blows up if and only if $p > 1$, when $m > 1$ and $p > (m+1)/2$, when $m \le 1$.*

Proof. Since U^0 is increasing, by Lemma 3.1 we have that U^j reaches its maximum at $i = N+1$.

On the other hand, from (2.7) we obtain $u_{N+1}^j \sim \lambda j$, and hence

$$u_{N+1}^j \to \infty, \qquad \text{as } j \to \infty. \tag{3.6}$$

We still have to show conditions which imply that the time $T_{h,\lambda}$ at which (3.6) occurs is finite. Indeed, for $p > m$, we get

$$T_{h,\lambda} = \sum_{k=1}^{\infty} \tau_k = \sum_{k=1}^{\infty} (u_{N+1}^j)^{-(2p-m)}\lambda \sim C \sum_{k=1}^{\infty} (\lambda j)^{-(2p-m)}\lambda.$$

Observe that this series converges or diverges according to $2p - m > 1$ or $2p - m \leq 1$.

Analogously, for $p \leq m$, we get

$$T_{h,\lambda} = \sum_{k=1}^{\infty} \tau_k = \sum_{k=1}^{\infty} (u_{N+1}^j)^{-p}\lambda \sim C \sum_{k=1}^{\infty} (\lambda j)^{-p}\lambda.$$

□

3.3. Blow-up rate.

This section deals with the blow-up rate of the numerical solutions. We show that they coincide with the one for the continuous problem.

Lemma 3.7. *If $p > m$ (for $m > 1$) or $p > (m+1)/2$ (for $m \leq 1$), then every positive solution that blows up at time $T_{h,\lambda}$ verifies*

$$\|U^j\|_\infty = u_{N+1}^j \sim (T_{h,\lambda} - t^j)^{\frac{-1}{2p-m-1}}.$$

Proof. From Lemma 3.6 we know that $u_{h,\lambda}$ blows up in finite time $T_{h,\lambda} = \sum_{k=1}^{\infty} \tau_k$. In order to prove the blow-up rate we observe that for any $k > j$, we have

$$u_{N+1}^k \geq u_{N+1}^j + c_1\lambda(k - j).$$

Therefore,

$$\begin{aligned}
&(T_{h,\lambda} - t^j) \\
&\quad = \sum_{k=j+1}^{\infty} \tau_k = \sum_{k=j+1}^{\infty} \frac{\lambda}{(u_{N+1}^k)^{2p-m}} \leq \sum_{k=j+1}^{\infty} \frac{\lambda}{(u_{N+1}^j + c_1\lambda(k-j))^{2p-m}} \\
&\quad = \sum_{l=1}^{\infty} \frac{\lambda}{(u_{N+1}^j + c_1\lambda l)^{2p-m}} \leq \int_0^{\infty} \frac{\lambda}{(u_{N+1}^j + c_1\lambda s)^{2p-m}} ds \\
&\quad = C \int_{u_{N+1}^j}^{\infty} \frac{1}{z^{2p-m}} dz = \frac{C}{(u_{N+1}^j)^{2p-m-1}}.
\end{aligned}$$

Thus,

$$u_{N+1}^j \leq C(T_{h,\lambda} - t^j)^{\frac{-1}{2p-m-1}}.$$

On the other hand, since $u^k_{N+1} \le u^j_{N+1} + c_2\lambda(k-j)$, using the same ideas as before, we get

$$u^j_{N+1} \ge C(T_{h,\lambda} - t^j)^{\frac{-1}{2p-m-1}},$$

and we conclude

$$u^j_{N+1} \sim (T_{h,\lambda} - t^j)^{\frac{-1}{2p-m-1}},$$

as expected from the description of the method. Moreover, if we take into account that c_1 and c_2 were chosen as

$$c_1 = \Gamma_1^{-(2p-m-1)}(2p-m-1)^{-1}, \qquad c_2(t^j) \sim c_1,$$

we have

$$\lim_{j\to\infty} u^j_{N+1}(T_{h,\lambda} - t^j)^{\frac{1}{2p-m-1}} \to c_1.$$

□

Observe that from the blow-up rate we have

$$h_i \sim (u^j_{N+1})^{-p+m} \sim C(T_{h,\lambda} - t^j)^{\frac{p-m}{2p-m-1}} = C(T_{h,\lambda} - t^j)^{\beta},$$

and, as mentioned before, the selfsimilar behavior is reproduced.

Lemma 3.8. *If $p \le m$, then every positive solution that blows up at time $T_{h,\lambda}$ verifies*

$$\|U^j\|_\infty = u^j_{N+1} \sim (T_{h,\lambda} - t^j)^{-\frac{1}{p-1}}.$$

Proof. It is similar to the previous one, using $\tau_j(u^j_{N+1})^p = \lambda$. □

3.4. Convergence of the numerical blow-up times.

We gather now the proof of convergence of the numerical blow-up times, $T_{h,\lambda}$, to the continuous one, T as $(h,\lambda) \to (0,0)$. When $m = 1$ we state a stronger result, that gives a bound for $|T_{h,\lambda} - T|$ in terms of h and λ.

Lemma 3.9. *If the numerical solution $u_{h,\lambda}$ converges uniformly to the continuous one, u, in sets of the form $[0,1] \times [0, T-\tau]$ and verifies the blow-up rates described in lemmas 3.7 and 3.8, then the numerical blow-up times converge to the continuous one:*

$$\lim_{(h,\lambda)\to(0,0)} T_{h,\lambda} = T.$$

Observe that the schemes we are considering verify the assumptions of this lemma. In fact, in Lemma 3.5 we proved the uniform convergence and from Lemmas 3.7 and 3.8 we get the blow-up rate. Recall that in the proof of convergence we used a comparison result that needs a stability restriction on λ when the method is explicit.

Proof. From the blow-up rate we have that,

$$T_{h,\lambda} - t^j \le \frac{C}{(u_{N+1}^j)^\gamma},$$

where $\gamma = 2p - m - 1$ or $\gamma = p - 1$ according to $p > m$ or $p \le m$. Given $\varepsilon > 0$, we can choose K large enough in order to have

$$\frac{C}{K^\gamma} < \frac{\varepsilon}{2}.$$

Let $\tau < \varepsilon/4$ such that the continuous solution verifies $u(1, T-2\tau) > K$. On the other hand, from the uniform convergence in $[0,1]\times[0,T-\tau]$, for every h, λ small enough, there exists j_0 such that $T - 2\tau < t^{j_0} < T - \tau$ and $u_{N+1}^{j_0} > K$. Hence,

$$\begin{aligned} |T_{h,\lambda} - T| &\le |T_{h,\lambda} - t^{j_0}| + |t^{j_0} - T| \le \frac{C}{(u_{N+1}^{j_0})^\gamma} + 2\tau \\ &\le \frac{C}{K^\gamma} + 2\tau \le \varepsilon. \end{aligned}$$

This ends the proof. □

Lemma 3.10. *If $m = 1$, then there exist $\alpha > 0$ and $C > 0$ such that*

$$|T_{h,\lambda} - T| \le C(h^2 + \lambda)^\alpha.$$

Proof. The idea of the proof is as follows: first we get bounds for the first j_0, such that the error verifies

$$\begin{aligned} &\|u_{h,\lambda}(\cdot, t^j) - u(\cdot, t^j)\|_{L^\infty([0,1])} < 1, \quad \text{for all } j \in [0, j_0], \\ &\|u_{h,\lambda}(\cdot, t^{j_0+1}) - u(\cdot, t^{j_0+1})\|_{L^\infty([0,1])} \ge 1, \end{aligned} \tag{3.7}$$

then we use this bound to prove the convergence result.

To get a bound for j_0 let us look at our scheme (2.10). From the description of the numerical scheme we have

$$\frac{u_{N+1}^{j+1} - u_{N+1}^j}{\tau_j} = \frac{2(u_N^j - u_{N+1}^j)}{h^2} + \frac{2(u_{N+1}^j)^p}{h}. \tag{3.8}$$

If we call $v_i^j = u(x_i, t^j)$ we get that v satisfies

$$\frac{v_{N+1}^{j+1} - v_{N+1}^j}{\tau_j} = \frac{2(v_N^j - v_{N+1}^j)}{h^2} + \frac{2(v_{N+1}^j)^p}{h} + C(h^2+\lambda)(T-t)^\theta, \tag{3.9}$$

where θ depends only on p. Let us remark that $C(T-t)^\theta$ is a bound for the fourth spatial derivatives of $u(x,t)$, a regular solution of (1.1).

Subtracting (3.8) from (3.9) we have for $e^j_{N+1} = u^j_{N+1} - v^j_{N+1}$,

$$
\begin{aligned}
&\frac{e^{j+1}_{N+1} - e^j_{N+1}}{\tau_j} \\
&\quad \leq \frac{C_1(e^j_N - e^j_{N+1})}{h^2} + \frac{C_2}{h}\xi^{p-1}|e^j_{N+1}| + C_3(h^2+\lambda)(T-t)^\theta,
\end{aligned}
\tag{3.10}
$$

where ξ is an intermediate value between z^j_{N+1} and v^j_{N+1}. Using that $j \in [0, j_0)$, the bound (3.7) and the blow-up rate, we get

$$\xi^{p-1} \leq C(T-t)^{-\frac{1}{2}}.$$

Hence (3.10) gives

$$\frac{e^{j+1}_{N+1} - e^j_{N+1}}{\tau_j} \leq \frac{C_1(e^j_N - e^j_{N+1})}{h^2} + C_2\frac{|e^j_{N+1}|}{h(T-t)^{\frac{1}{2}}} + C_3(h^2+\lambda)(T-t)^\theta.$$

This inequality is a discretization of the problem

$$
\begin{cases}
e_t = e_{xx} + C(T-t)^\theta(h^2+\lambda) & (x,t) \in (0,1)\times[0,T), \\
e_x(0,t) \leq 0 & t \in [0,T), \\
e_x(1,t) \leq C(T-t)^{-\frac{1}{2}}e(1,t) & t \in [0,T), \\
e(x,0) \leq C(h^2+\lambda) & x \in (0,1).
\end{cases}
\tag{3.11}
$$

Below, we will construct a supersolution for problem (3.11), such that

$$\overline{e}(x,t) \leq C(h^2+\lambda)(T-t)^{-\eta}, \tag{3.12}$$

and with the fourth spatial derivatives positives in $[0,1]$ and the first two time derivatives positives in $[0,T)$.

We take $e^j_i = \overline{e}(x_i, t^j)$, which is a supersolution. Applying a comparison argument we obtain $e^j_i \leq C(T-t^j)^{-\eta}(h^2+\lambda)$. Similarly, applying the same ideas to $\tilde{e}^j_i = v^j_i - u^j_i$, we get $e^j_i \geq -C(T-t^j)^{-\eta}(h^2+\lambda)$. Therefore

$$|e^j_i| \leq C(T-t^j)^{-\eta}(h^2+\lambda).$$

We get a bound for j_0 as follows: since

$$1 \leq |e^{j_0+1}_i| \leq C(T-t^{j_0+1})^{-\eta}(h^2+\lambda),$$

we get

$$(T-t^{j_0+1}) \leq C(h^2+\lambda)^{\frac{1}{\eta}}.$$

With this bound for $(T-t^{j_0+1})$ we obtain the desired result. Indeed,

$$|T_{h,\lambda} - T| \leq |T_{h,\lambda} - t^{j_0+1}| + |T - t^{j_0+1}| \leq |T_{h,\lambda} - t^{j_0+1}| + C(h^2+\lambda)^{\frac{1}{\eta}}.$$

On the other hand

$$\begin{aligned}|T_{h,\lambda} - t^{j_0+1}| &\le |T_{h,\lambda} - t^{j_0}| + |t^{j_0} - t^{j_0+1}| \le C(u_{N+1}^{j_0})^{-2(p-1)} + \lambda \\ &\le C(v_{N+1}^{j_0} + 1)^{-2(p-1)} + \lambda \le C(T - t^{j_0}) + \lambda \\ &\le C(h^2+\lambda)^{\frac{1}{\eta}} + \lambda \le C(h^2+\lambda)^{\frac{1}{\eta}}.\end{aligned}$$

It remains to be more concrete in the construction of the supersolution to (3.11). This kind of construction was previously done in [14]. We include it here for the sake of completeness.

Let us consider the following problem,

$$\begin{cases} E_t = E_{xx} + C(T-t)^{\theta}(h^2+\lambda) & (x,t) \in (0,1)\times[0,T), \\ E_x(0,t) = 0 & t\in[0,T), \\ E_x(1,t) = C(T-t)^{-1/2}E(1,t) & t\in[0,T), \\ E(x,0) = C(h^2+\lambda) & x\in(0,1). \end{cases}$$

In order to build a supersolution, $\overline{E}$, to this problem such that

$$\overline{E}(x,t) \le C(h^2+\lambda)(T-t)^{-\eta}$$

and with the first four spatial derivatives positives in $[0,1]$, we look for a supersolution of the form

$$\overline{E}(x,t) = C(h^2+\lambda)(T-t)^{\theta}a(x,t),$$

with $a(x,t)$ a solution of

$$\begin{cases} a_t = a_{xx} & (x,t) \in (0,1)\times[0,T), \\ a_x(0,t) = 0 & t\in[0,T), \\ a_x(1,t) = C(T-t)^{-1/2}a(1,t) & t\in[0,T), \\ a(x,0) = a_0(x) & x\in(0,1). \end{cases} \tag{3.13}$$

As we want $\overline{E}$ to have their first four spatial derivatives positives, we impose that $a_0(x)$ is a smooth compatible initial datum with the first four spatial derivatives positives. The positivity of the derivatives is preserved for every $t\in[0,T)$. Now let us see that there exists r such that

$$a(x,t) \le \frac{C}{(T-t)^r}. \tag{3.14}$$

To this end we want to construct a supersolution, $v(x,t)$, to (3.13) such that (3.14) holds. This can be easily done by the following procedure: take $v(x,t)$ a solution of (1.1) with the boundary condition given by $v_x(1,t) = v^q(1,t)$, with q small, and initial datum v_0 such that $v(x,t)$ blows up exactly at time T, see [24] for a proof of the fact that for every time T there exist an initial datum such that $v(x,t)$ blows up at time

T. From the known blow-up rate for solutions of (1.1), see [15, 19], we get that $v(x,t)$ verifies

$$v(1,t) \sim \frac{L}{(T-t)^{\frac{1}{2(q-1)}}},$$

with $L^{q-1} \to +\infty$ as $q \searrow 1$, see [15] for an explicit formula for $L(q)$. Let us fix q such that $L^{q-1}(q) > C$. This choice leads to a supersolution of (3.13), since

$$v_x(1,t) = v^q(1,t) = v^{q-1}v(1,t) \geq \frac{L^{q-1}}{(T-t)^{\frac{1}{2}}} v(1,t).$$

This ends the proof. □

3.5. The blow-up set.

Now we turn our interest to the blow-up set of the numerical solution. We want to look at the set of points, x, such that $u_{h,\lambda}(x,t) \to +\infty$ as $t^j \nearrow T_{h,\lambda}$. Throughout this section we assume that the parameters h and λ are fixed.

We begin our analysis with the case $p > m$. In [2] and [13] it is proved that for a fixed mesh the numerical blow-up set is given by $B(U) = [1 - Lh, 1]$, where L depends only on p and m. Notice that as L does not depend on h, the blow-up set converges to the continuous blow-up set $B(u) = \{1\}$, as $h \to 0$.

We remark that the adding points method, adds at least L points near $x = 1$ when the solution blows up, therefore the parameter h_i goes to zero and formally the blow-up set for the method must be $B(u_{h,\lambda}) = \{1\}$. In order to prove this, we first study the propagation of blow-up.

Lemma 3.11. *Assume $p > m$. If $u_{k+1}^j \leq j^{\alpha_{k+1}}$, $0 < \alpha_{k+1} \leq 1$, then*

$$u_k^j \leq j^{\alpha_k}, \quad \alpha_k = 1 - (2p - m - m\alpha_{k+1}).$$

Proof. Recall that from Lemma 3.1 and Lemma 3.7 we have that

$$\max_i u_i^j = u_{N+1}^j \sim C(T_{h,\lambda} - t)^{-\frac{1}{2p-m-1}} \sim j.$$

Therefore,

$$\begin{aligned} u_k^{j+1} - u_k^j &= \frac{\tau_j}{h^2}((u_{k-1}^j)^m - 2(u_k^j)^m + (u_{k+1}^j)^m) \\ &\leq \frac{C\tau_j}{h^2}(u_{k+1}^j)^m \leq \frac{C\lambda}{(u_{N+1}^j)^{2p-m}} j^{m(\alpha_{k+1})} \\ &\leq \frac{C\lambda}{j^{2p-m}} j^{m\alpha_{k+1}} = \frac{C\lambda}{j^{2p-m-m\alpha_{k+1}}}. \end{aligned}$$

This bound implies that

$$u_k^{j+1} - u_k^0 = \sum_{l=0}^{j} \left(u_k^{l+1} - u_k^l \right) \le \sum_{l=0}^{j} \frac{C\lambda}{l^{2p-m-m\alpha_{k+1}}},$$

which means that

$$u_k^{j+1} \le u_k^0 + Cj^{-(2p-m-m\alpha_{k+1})+1}$$

. ∎

In view of this, u_k^j is bounded if $\alpha_k = 1 - (2p - m - m\alpha_{k+1}) \le 0$. Now, given a blow-up node x_k at most $u_{k-1}^j, \dots, u_{k-L}^j$ blow up and (at least) $u_1^j, \dots, u_{k-L-1}^j$ are bounded. Indeed, since $u_k^j \le cj$ we can apply the previous lemma with $\alpha_k \le 1$, hence

$$\begin{array}{rcl} \alpha_k & \le & 1 \\ \alpha_{k-1} & \le & 1 - (2p - 2m) \\ \alpha_{k-2} & \le & 1 - (2p - 2m) - m(2p - 2m) \\ & \vdots & \\ \alpha_{k-l} & \le & 1 - (2p - 2m) \displaystyle\sum_{i=0}^{l} m^i. \end{array}$$

Let L be the first integer such that $(2p-2m)\sum_{i=0}^{L+1} m^i \ge 1$, then u_{k-L-1}^j is bounded.

Lemma 3.12. *Assume that $u_{h,\lambda}$ is given by the adding points method. If $p > m$ then $B(u_{h,\lambda}) = \{1\}$.*

Proof. Let $\bar{x} < 1$, we claim that the numerical solution is bounded in $[0, \bar{x}]$, i.e there exists $C = C(\bar{x}) > 0$ such that $u_{h,\lambda}(x,t) \le C$, for all $x \in [0, \bar{x}]$, $t \in [0, T_{h,\lambda})$. Since we adapt near $x = 1$ collapsing (at least) L points in $\{1\}$ as the solution gets large, we can consider a node $x_{k+1} > \bar{x}$ such that, as $j \to \infty$, there exits $L + 1$ nodes between $\bar{x}$ and x_{k+1}. Hence, from the previous computations, for t^j close to $T_{h,\lambda}$, we get that for all nodes $x_i < \bar{x}$, u_i^j is bounded. This implies that $B(u_{h,\lambda}) = \{1\}$. ∎

For the moving point method the result is different:

Lemma 3.13. *Assume that $u_{h,\lambda}$ is given by the moving point method. If $p > m$ then $B(u_{h,\lambda}) \subseteq [1 - Lh, 1]$, where L is given as above.*

Proof. Let w be the moving node and z the numerical solution at this node. Since $z^j \le u^j_{N+1} \le Cj$, we have

$$\begin{aligned} u^{j+1}_N - u^j_N &= \frac{2\tau_j \big((u^j_{N-1})^m (h-h_i) - (u^j_N)^m (2h-h_i) + (z^j)^m h\big)}{(h-h_i)h(2h-h_i)} \\ &\le \frac{2Cj^{m-2p}(z^j)^m h}{(h-h_i)h(2h-h_i)} \le Cj^{2m-2p}. \end{aligned}$$

Hence, we get that

$$u^{j+1}_N \le u^j_N + Cj^{2m-2p+1}.$$

This means that the node u^j_N is bounded if $2m-2p+1 \le 0$. Otherwise, we apply Lemma 3.11 and get that at most the nodes $w, x_N, \dots, x_{N-L+1}$ blow up. Therefore, we obtain $B(u_{h,\lambda}) \subseteq [1-Lh, 1]$. □

We still have to prove that when $p \le m$ the blow-up set is given by $[0,1]$, i.e. there is global blow-up. Recall that in this range of parameters, the methods adapt just the time variable, letting the spacial mesh fixed.

Inspired in the self-similar variables introduced in [17] we define (Y^j) as

$$y^j_i = u^j_i j^{-1}, \qquad 1 \le i \le N+1. \tag{3.15}$$

In the sequel of the proof we will use Δy^{j+1}_i to denote

$$\frac{y^{j+1}_i - y^j_i}{\tau_j / j^{-(p-1)}},$$

This can be thought as $\tau_j/j^{-(p-1)}$ to be the time step in the new variables. With this notation Y^j verifies

$$\left\{ \begin{aligned} \Delta y^{j+1}_1 = &\quad -\frac{2j^{m-(p-1)}\big((y^j_2)^m - (y^j_1)^m\big)}{(j+1)h^2} - \frac{j^{-(p-1)}u^j_i}{\tau_j j(j+1)}, \\ \Delta y^{j+1}_i = &\quad -\frac{j^{m-(p-1)}\big((y^j_{i-1})^m - 2(y^j_i)^m + (y^j_{i+1})^m\big)}{(j+1)h^2} \\ &\qquad -\frac{j^{-(p-1)}u^j_i}{\tau_j j(j+1)}, \qquad 2 \le i \le N, \\ \Delta y^{j+1}_{N+1} = &\quad -\frac{2j^{m-(p-1)}\big((y^j_N)^m - (y^j_{N+1})^m\big)}{(j+1)h^2} \\ &\qquad +\frac{2j^{p-(p-1)}(y^j_{N+1})^p}{(j+1)h} - \frac{j^{-(p-1)}u^j_i}{\tau_j j(j+1)}, \end{aligned} \right. \tag{3.16}$$

and the initial datum becomes

$$y_i^0 = j^{-1} u_0(x_i), \qquad 1 \le i \le N+1.$$

From the blow-up rate proved in Lemma 3.8, we have that the vector Y^j is bounded, i.e

$$Y^j \le C, \quad \text{for all } j.$$

Moreover, there exist positive constants, c and C such that the last node verifies

$$c \le y_{N+1}^j \le C, \quad \text{for all } j.$$

Lemma 3.14. *If $1 < p < m$ then*

$$u_k^j \sim \frac{C}{(T_{h,\lambda} - t)^{\frac{1}{p-1}}} \sim Cj, \qquad\qquad 1 \le k \le N,$$

therefore $B(u_{h,\lambda}) = [0,1]$. This phenomena is known as uniform global blow-up.

Proof. From the equation for the last node in (3.16), the expression for τ_j and the asymptotic behavior we get

$$y_{N+1}^{j+1} - y_{N+1}^j \ge \frac{a_j}{j^{1+p-m}} - \frac{c}{j}, \tag{3.17}$$

where $a_j = (y_{N+1}^j)^m - (y_N^j)^m$ and c is a positive constant. If we write y_{N+1}^{j+1} as

$$y_{N+1}^{j+1} = \sum_{l=0}^{j} \left(y_{N+1}^{l+1} - y_{N+1}^l \right) + y_{N+1}^0,$$

from (3.17), we get

$$y_{N+1}^{j+1} \ge \sum_{l=0}^{j} \left(\frac{a_l}{l^{1+p-m}} - \frac{c}{l} \right) + y_{N+1}^0.$$

On the other hand, since y_{N+1}^j is bounded, we have

$$\left(\frac{a_j}{j^{1+p-m}} - \frac{c}{j} \right) j \to 0,$$

which implies that

$$\frac{a_j}{j^{p-m}} \to c.$$

Hence, since $p < m$, we get

$$0 \le (y_{N+1}^j)^m - (y_N^j)^m = a_j \to 0, \quad \text{as } j \to \infty.$$

This means that $y_N^j \ge c$. Applying the same argument with y_k^j we obtain that $y_{k-1}^j \ge c$ for all $2 \le k \le N$, which in terms of the original variables leads to

$$u_k^j \ge c(T_{h,\lambda} - t^j)^{-\frac{1}{p-1}} \quad \text{for } 1 \le k \le N.$$

The reverse inequality is obtained trivially from the fact that $u_k^j \le u_{N+1}^j \le c(T_{h,\lambda} - t^j)^{-1/(p-1)}$. We obtain uniform global blow-up. □

Lemma 3.15. *If $p = m$ then*

$$u_k^j \sim \frac{C}{(T_{h,\lambda} - t)^{\frac{1}{p-1}}} \sim Cj, \qquad 1 \le k \le N,$$

therefore $B(u_{h,\lambda}) = [0, 1]$.

Proof. As we have already seen $u_{N+1}^j \ge cj$. Let us prove that from this estimate we get that all nodes behave as the last one, i.e. $u_k^j \ge cj$, for all $1 \le k \le N$, and hence the blow-up is global.

First of all observe that U^j is a supersolution of

$$M(U^{j+1} - U^j) \ge -\tau_j A(U^j)^m + \tau_j Bcj^m.$$

Let us look for a subsolution of this problem of the form $V^j = jZ$, where $Z = (z_1, \dots, z_{N+1})$, and does not depend on j. Then Z must verify

$$CMZ \le -AZ^m + Bc,$$

which can be written explicitly as

$$\left\{ \begin{array}{lll} Cz_1 & \le \dfrac{2(z_2^m - z_1^m)}{h^2}, & \\ Cz_i & \le \dfrac{z_{i-1}^m - 2z_i^m + z_{i+1}^m}{h^2}, & 2 \le i \le N \\ Cz_{N+1} & \le \dfrac{2(z_N^m - z_{N+1}^m)}{h^2} + c. & \end{array} \right. \tag{3.18}$$

In order to find $Z > 0$ verifying the above system, we proceed as follows: we fix $z_1 > 0$ which will be determined later and compute $z_2, \dots, z_{N+1}$ in terms of z_1. We choose z_2 in such a way that the first inequality in (3.18) is verified with an equality. Iterating this procedure we obtain $z_3, \dots, z_{N+1}$ using the first N equations. It remains to check that the last inequality holds. Observe that z_{N+1} is a continuous positive function of z_1 and that $z_{N+1} = 0$ when $z_1 = 0$. Therefore the last inequality is verified, if z_1 is small enough. Hence, as the spatial mesh is fixed, we may use a standard comparison argument, which is proved in a similar way as Lemma 3.3, and get $v_i^j = z_i j \le u_i^j$, $1 \le i \le N$.

Remark that, as in the case $p < m$, the upper bound $u_i^j \le cj$ follows easily from the behaviour of the last node. Therefore, we obtain uniform complete blow-up. □

References

[1] L. M. Abia, J. C. Lopez-Marcos, J. Martinez. *Blow-up for semidiscretizations of reaction diffusion equations.* Appl. Numer. Math. Vol. 20, (1996), 145-156.

[2] G. Acosta, R. Duran and J. D. Rossi. *An adaptive time step procedure for a parabolic problem with blow-up.* Computing. Vol. 68 (4), (2002), 343-373.

[3] G. Acosta, J. Fernandez Bonder, P. Groisman and J.D. Rossi. *Numerical approximation of a parabolic problem with a nonlinear boundary condition in several space dimensions.* Discr. Cont. Dyn. Sys. B, Vol. 2 (2), (2002), 279-294.

[4] C. Bandle and H. Brunner. *Blow-up in diffusion equations: a survey.* J. Comp. Appl. Math. Vol. 97, (1998), 3-22.

[5] M. Berger and R. V. Kohn. *A rescaling algorithm for the numerical calculation of blowing up solutions.* Comm. Pure Appl. Math. Vol. 41, (1988), 841-863.

[6] C. Budd and G. Collins. *An invariant moving mesh scheme for the nonlinear diffusion equation.* Appl. Numer. Math. Vol. 26, (1998), 23-39.

[7] C. Budd, S. Chen and R. D. Russell. *New self-similar solutions of the nonlinear Schrodinger equation with moving mesh computations.* J. Comp. Phys. Vol. 152, (1999), 756-789.

[8] C. J. Budd, W. Huang and R. D. Russell. *Moving mesh methods for problems with blow-up.* SIAM Jour. Sci. Comput. Vol. 17(2), (1996), 305-327.

[9] P. Ciarlet. The finite element method for elliptic problems. North Holland, (1978).

[10] C. Cortázar, M. Elgueta and O. Venegas. *On the Blow-up set for $u_t = (u^m)_{xx},\ m > 1$, with nonlinear boundary conditions.* Preprint.

[11] R. G. Duran, J. I. Etcheverry and J. D. Rossi. *Numerical approximation of a parabolic problem with a nonlinear boundary condition.* Discr. and Cont. Dyn. Sys. Vol. 4 (3), (1998), 497-506.

[12] J. Fernández Bonder and J. D. Rossi. *Blow-up vs. spurious steady solutions.* Proc. Amer. Math. Soc. Vol. 129 (1), (2001), 139-144.

[13] R. Ferreira, P. Groisman and J. D. Rossi. *Numerical blow-up for a nonlinear problem with a nonlinear boundary condition.* Math. Models Methods Appl. Sci. M3AS. Vol. 12 (4), (2002), 461-484.

[14] R. Ferreira, P. Groisman and J.D. Rossi. *Adaptive numerical schemes for a parabolic problem with blow-up.* IMA J. Numer. Anal. Vol. 23 (3), (2003), 439–463.

[15] M. Fila and P. Quittner. *The blowup rate for the heat equation with a nonlinear boundary condition.* Math. Methods Appl. Sci., Vol. 14, (1991), 197–205.

[16] J. Filo. *Diffusivity versus absorption through the boundary*. J. Differ. Equations, Vol. 99, (1992), n. 2, 281–305.

[17] Y. Giga and R. V. Kohn. *Nondegeneracy of blow up for semilinear heat equations*. Comm. Pure Appl. Math. Vol. 42, (1989), 845-884.

[18] W. Huang, Y. Ren and R. D. Russell. *Moving mesh partial differential equations (MMPDEs) based on the equidistribution principle*. SIAM J. Numer. Anal. Vol 31, (1994), 709-730.

[19] B. Hu and H. M. Yin. *The profile near blowup time for solution of the heat equation with a nonlinear boundary condition*. Trans. Amer. Math. Soc., Vol. 346(1), (1994), 117–135.

[20] M. N. Le Roux. *Semidiscretizations in time of nonlinear parabolic equations with blow-up of the solutions*. SIAM J. Numer. Anal. Vol. 31, (1994), 170-195.

[21] T. Nakagawa. *Blowing up of a finite difference solution to* $u_t = u_{xx} + u^2$. Appl. Math. Optim., 2, (1976), 337-350.

[22] T. Nakagawa and T. Ushijima. *Finite element analysis of the semilinear heat equation of blow-up type*. Topics in Numerical Analysis III (J.J.H Miller, ed.), Academic Press, London, (1977), 275-291.

[23] C. V. Pao, "Nonlinear parabolic and elliptic equations". Plenum Press, New York, 1992.

[24] J. D. Rossi. *The blow-up rate for a system of heat equations with nontrivial coupling at the boundary*. Math. Methods Appl. Sci., vol 20, (1997), 1-11.

C. Brändle. Depto. Matemáticas, U. Autónoma de Madrid, 28049 Madrid, Spain.

E-mail address: cristina.brandle@uam.es

P. Groisman. Inst. Cálculo, FCEyN., UBA, (1428) Buenos Aires, Argentina.

E-mail address: pgroisma@dm.uba.ar

J.D. Rossi. Depto. Matemática, FCEyN, UBA, (1428) Buenos Aires, Argentina.

E-mail address: jrossi@dm.uba.ar