Computation of Space-Time Patterns via ALE Methods

V. Thümmler*,[1], A. Weddemann[2]
[1] Department of Mathematics, Bielefeld University, [2] Department of Physics, Bielefeld University
* Corresponding author: Bielefeld University, thuemmle@math.uni-bielefeld.de

Abstract: Partial differential equations which exhibit solutions that are spatial-temporal patterns are often found in biological and chemical systems, e.g. when describing pattern formation in reaction-diffusion systems. Special classes of such patterns are relative equilibria and relative periodic orbits, which are solutions that in an appropriately comoving frame of reference are stationary and periodic respectively. Examples are (modulated) traveling waves in 1d and 2d or (meandering) spiral waves in 2d. We compare different moving mesh methods to compute such patterns, especially the Freezing method and Arbitrary Eulerian Lagrangian (ALE) methods.

Keywords: spatial-temporal pattern, ALE, moving mesh, freezing, PDAE

1 Introduction

Spatial temporal patterns arise as solutions of reaction-diffusion equations, which are parabolic PDEs of the form

$$u_t = D\Delta u + f(u), \quad u(\cdot, 0) = u^0, \qquad (1)$$

where $x \in \mathbb{R}^d$, $u(x,t) \in \mathbb{R}^m$, $D \in \mathbb{R}^{m,m}$, $t \in I = [0,T]$. Such patterns can be traveling waves, rotating spirals or scroll waves. When simulating such systems by standard time-stepping schemes one faces two problems: the pattern may move out of any finite computational domain (e.g. in case of traveling waves) and the mesh is not adapted to the solution, i.e. mesh points are not equally distributed along the graph of its components. To compute such patterns, different methods can be used: Firstly, the Freezing method, where the whole mesh is moved with the velocity of the traveling wave. Since this velocity is also unknown, it has to be computed in parallel with the solution. This can be achieved by adding an integral constraint, a so called phase condition to (1). Thus the partial differential equation (PDE) (1) is transformed into a partial differential algebraic equation (PDAE). Secondly, Arbitrary Eulerian Lagrangian (ALE) methods, which are moving mesh methods where each point of the mesh is moved individually according to some prescribed movement. As for other moving mesh methods, the movement of the individual mesh points can be governed by an equidistribution principle for an appropriate monitor function, which for example measures the gradient or the curvature of the solution. Thirdly, one can combine the above methods: in addition to moving the whole mesh with the velocity of the traveling wave the individual mesh points are adapted according to an appropriate monitor function using the ALE method.

2 Methods

For the ease of presentation we will describe the freezing method as well as the ALE method only for one space dimension ($d = 1$). The generalization to higher dimensions ($d > 1$) is straightforward.

2.1 Freezing traveling waves

The main idea for the freezing method is to rewrite equation (1) using the ansatz $v(\xi,t) = u(x-\gamma(t),t)$, $\gamma : I \to \mathbb{R}$ in a moving frame of reference and to compute its (time-dependent) velocity $\lambda(t) = \dot\gamma(t)$ via an additional integral constraint $0 = \int_{\mathbb{R}} u_x^0(\xi)^T(v(\xi,t) - u^0(\xi))\,d\xi$, a so called phase condition (for more details see [1, 17]). This leads to a partial differential algebraic equation (PDAE) for (v,λ), namely

$$v_t = Dv_{\xi\xi} + \lambda v_\xi + f(v), \;\; v(\cdot,0) = u^0 \quad (2a)$$

$$0 = \int_{\mathbb{R}} u_x^0(\xi)^T(v(\xi,t) - u^0(\xi))\,d\xi. \qquad (2b)$$

By this ansatz a traveling wave solution $u(x,t) = \bar v(x - \bar\lambda t)$ of (1) with profile $\bar v$ and velocity $\bar{\lambda}$ is transformed into a stationary solution $(\bar{v}, \bar{\lambda})$ of (2). Thus we can view solving (2) as a transient method to compute traveling waves (or more generally, relative equilibria) of (1). Alternatively, this method can viewed as a special case of a moving mesh method where all mesh points (including the boundary) are moved with the same velocity, namely the velocity of the traveling wave.

2.2 Moving meshes with ALE

A more general method which allows to move all mesh points individually with a prescribed mesh velocity is the Arbitrary Euler Lagrangian method (ALE). This method has been developed in the context of fluid dynamics [8] and a multitude of results dealing with different aspects of this method exist (for an overview see [5]). Here we will describe ALE only as it is implemented in the moving mesh application mode in Comsol Multiphysics according to [4].

Let $\Omega_t \subset \mathbb{R}$ be a moving domain for $t \in I = [0, T]$. We define the ALE mapping $\mathcal{A}_t : \Omega_0 \to \Omega_t$, with $\mathcal{A}_0(\xi) = \xi$ with inverse $\mathcal{A}_t^{-1}$ and derivative $\mathcal{DA}_t$. For simplicity we assume that $\mathcal{A}_t$ has as much regularity w.r.t. ξ and t as required (for more details see [6]). Define the map $b : \mathbb{R} \times I \to \Omega_0,\ (x,t) \mapsto \mathcal{A}_t^{-1}(x)$ and note that $b_x(\mathcal{A}_t(\xi), t) = \mathcal{DA}_t^{-1}(\xi)$. We restrict (1) to a moving domain $x \in \Omega_t \subset \mathbb{R}$ (using appropriate boundary conditions) and obtain the weak form for test functions $\hat{u} : \Omega_t \times I \to \mathbb{R}^m$

$$0 = \int_{\Omega_t} \hat{u}(x,t)^T u_t(x,t) + \hat{u}_x(x,t)^T u_x(x,t) - \hat{u}(x,t)^T f(u(x,t)) dx.$$

With the coordinate transform $x = \mathcal{A}_t(\xi)$ this is equivalent to

$$\begin{aligned} 0 = \int_{\Omega_0} & [\hat{u}(\mathcal{A}_t(\xi), t)^T u_t(\mathcal{A}_t(\xi), t) \\ & + \hat{u}_x(\mathcal{A}_t(\xi), t)^T u_x(\mathcal{A}_t(\xi), t) \\ & - \hat{u}(\mathcal{A}_t(\xi), t)^T f(u(\mathcal{A}_t(\xi), t))]\ \mathcal{DA}_t(\xi)\, d\xi. \end{aligned}$$

With $v(\xi, t) = u(\mathcal{A}_t(\xi), t)$ and

$$\begin{aligned} u_x(\mathcal{A}_t(\xi), t) &= v_\xi(\xi, t)\mathcal{DA}_t^{-1}(\xi) \\ u_t(\mathcal{A}_t(\xi), t) &= v_t(\xi, t) + v_\xi(\xi, t)\ w(\xi, t), \end{aligned}$$

where $w(\xi, t) = b_t(\mathcal{A}_t(\xi), t)$ we obtain

$$\begin{aligned} 0 = \int_{\Omega_0} & \Big[\hat{v}^T\, (v_t + v_\xi\, w - f(v)) \\ & + (\hat{v}_\xi \mathcal{DA}_t^{-1})^T v_\xi \mathcal{DA}_t^{-1}\Big] \mathcal{DA}_t\, d\xi \end{aligned} \tag{3}$$

for time independent test functions $\hat{v} : \Omega_0 \to \mathbb{R}^m,\ \xi \mapsto \hat{v}(\xi) = \hat{u}(\mathcal{A}_t(\xi), t)$.

For the ALE method the choice of the equation which describes the movement of the mesh points is crucial. To compute the mesh velocity w we use the equidistribution principle which is widely used in the context of moving mesh PDEs in 1d [9, 12, 15]. Here a monitor function M is defined which is related to the expected error, e.g. the gradient or the curvature of the solution. This differs from other ALE applications where the mesh movement is determined from physical principles or prescribed externally. Based on this principle, different moving mesh PDEs have been derived in [9, 10]. One of them (MMPDE5) which can be put into the general form assumed by Comsol Multiphysics reads

$$\tau u_t = (M(u)\, u_x)_x. \tag{4}$$

For our application we use the gradient monitor

$$M(u) = \sqrt{\alpha + \|u_x\|^2}. \tag{5}$$

The parameter τ determines the speed of mesh adaptation whereas α controls the distribution of the mesh points. One possibility is to choose α such that all mesh points are equally distributed w.r.t. to the arclength of the solution u.

There are many other possible methods to determine the mesh velocity w. In [18] the velocity of the mesh is determined by minimizing a mesh energy integral. This method specializes to the equidistribution principle in 1d. Another approach in two space dimensions based on a so called geometric conservation law is developed in [2, 3]. In [11] the mesh velocity is governed by entropy production. Femlab has also been used with monitor functions in [16] using a static redistribution method and interpolation onto the new mesh (but not ALE).

2.3 Freezing and ALE

The combination of the above methods tries to combine the best of both methods. Here the transformed equation (2) is solved using the ALE method with the gradient monitor. In this case (3) reads

$$0 = \int_{\Omega_0} \Big[\hat{v}^T \left(v_t + v_\xi(w - \lambda\,\mathcal{D}\mathcal{A}_t^{-1}) - f(v)\right) + (\hat{v}_\xi \mathcal{D}\mathcal{A}_t^{-1})^T v_\xi \mathcal{D}\mathcal{A}_t^{-1}\Big] \mathcal{D}\mathcal{A}_t d\xi.$$

Note, that by adding the convective term λv_ξ the whole domain (including the boundaries) is moved with the velocity λ which is determined by the phase condition, whereas $w\, v_\xi$ moves the mesh points according to the monitor function M.

It remains to note that the addition of convective terms, which is necessary in all three cases, affects the stability of the time integration even when using implicit methods. This has been demonstrated in [14, 12] and analyzed in [6, 7, 15]. In [13] upwinding has been used to deal with that problem in the context of finite difference methods, whereas for finite element methods the use of stabilization methods such as streamline diffusion might be helpful [11].

3 Implementation

The PDAE (2) is implemented in Comsol Multiphysics by coupling a general PDE mode for (2a) to a boundary mode (in 1d) or a weak point mode (in 2d) which enforces the phase condition (2b). The coupling between this constraint and the other equations is implemented using integration coupling variables.

The velocity w of the ALE mesh is determined via (4) using the monitor function (5) using a general PDE mode and is used in the moving mesh application mode to drive the mesh movement.

4 Numerical Results

In the following we demonstrate the above methods using different examples which exhibit traveling waves and modulated traveling waves.

4.1 Nagumo, 1d

The scalar Nagumo equation

$$u_t = u_{xx} + u(1-u)(u - \tfrac{1}{4}) \qquad (6)$$

is our first test example. An explicit traveling wave solution which connects the stationary points $u_- = 1$, $u_+ = 0$ is given by

$$\bar{v}(x) = \left(1 + \exp\left(\frac{x}{\sqrt{2}}\right)\right)^{-1}, \quad \bar{\lambda} = -\tfrac{\sqrt{2}}{4}.$$

In Fig. 1 we see the result of the ALE computation for (6) with $N = 76$ mesh points on the interval $[-75, 75]$ and $\tau = 1e-3$. Here the boundary of the domain is fixed and only the inner mesh points are adapted according to the monitor function given in (5). The mesh points follow the movement of the traveling wave which eventually leaves the computational domain.

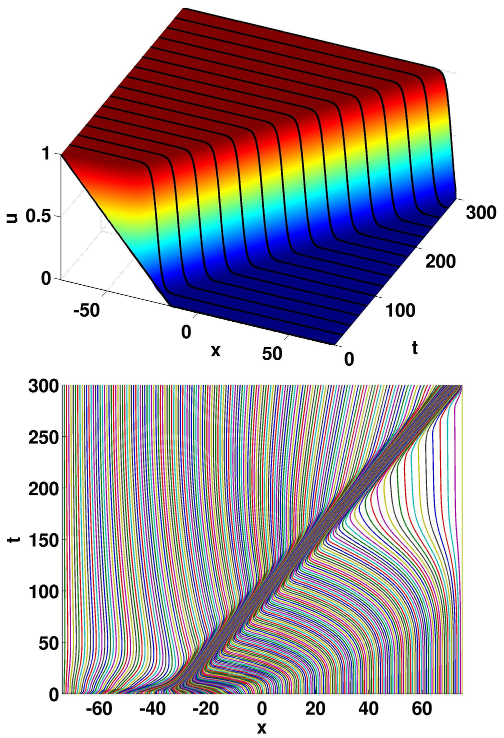

Figure 1: Nagumo, ALE, solution u and mesh.

In Fig. 2 we solve the transformed PDAE (2) which corresponds to (6) with $N = 51$ on $[-50, 50]$. By this transformation we have removed the overall motion of the traveling wave. The mesh points are adapted according to the monitor function (5) with $\alpha = 1e-4$.

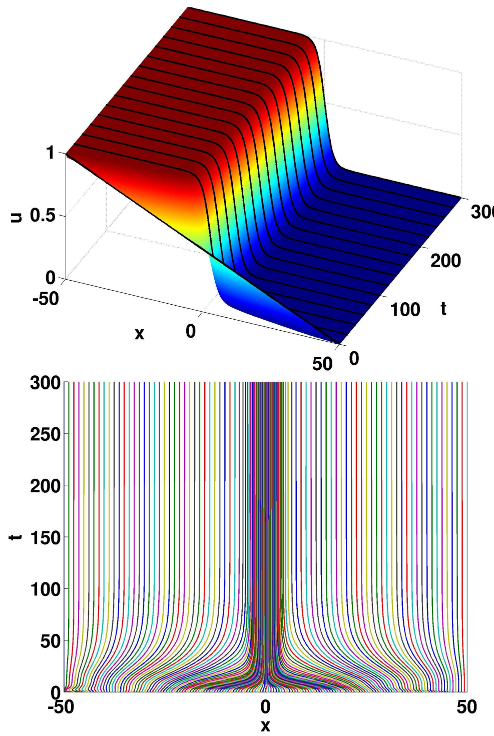

Figure 2: Nagumo, ALE for frozen system, solution u and mesh.

In Fig. 3 the time evolution of the velocity λ is compared for a uniform mesh and for the corresponding ALE computation. In both cases $N = 51$ mesh points have been used. While λ converges quickly to a value which deviates from the exact velocity $\bar{\lambda}$ by an amount which is determined by the mesh size for the uniform mesh, it converges more slowly but to a more accurate value for the ALE computation.

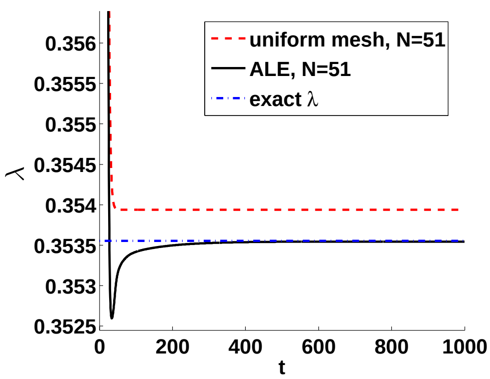

Figure 3: Nagumo, ALE vs. uniform mesh, time evolution of λ.

4.2 Autocat, 1d

A two component system which models an autocatalytic reaction is given by

$$\begin{aligned} u_t &= \delta u_{xx} - uv^m\,, \\ v_t &= v_{xx} + uv^m\,. \end{aligned} \tag{7}$$

Here $u(x,t)$ is the concentration of the reactant and $v(x,t)$ is the concentration of the autocatalyst. We suppose that $(u,v) \to (0,1)$ as $x \to -\infty$ and $(u,v) \to (1,0)$ as $x \to +\infty$. For different values of the exponent m, different types of solutions can be observed. For $m = 2$ we there exist traveling waves as found in the Nagumo example, whereas for $m = 9$ we find modulated traveling waves, i.e. solutions which are periodic in the PDAE framework (2). Since the results for $m = 2$ are similar to the results for the Nagumo system we omit them here and illustrate only the case $m = 9$, $\delta = 0.1$.

We display the v-component and the movement of the mesh points of the ALE computation with $\tau = 0.01$ in Fig. 4 for the PDE (7) and in Fig. 5 for the corresponding PDAE.

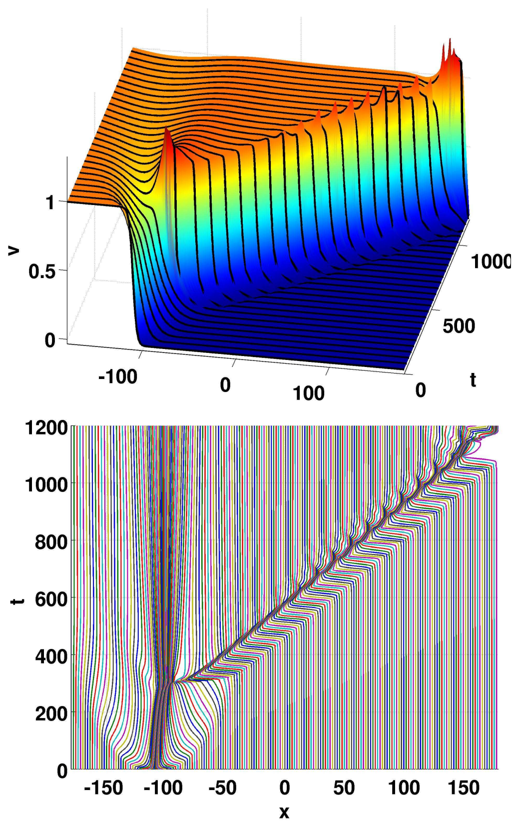

Figure 4: Autocat, ALE, solution v and mesh.

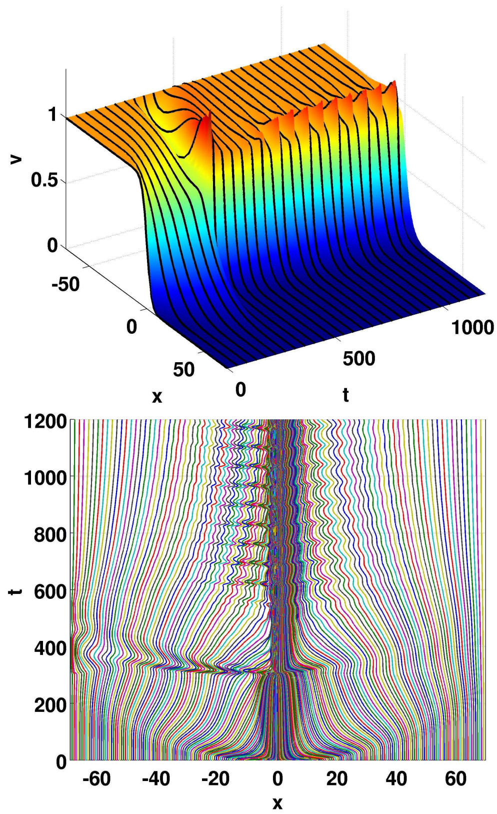

Figure 5: Autocat, ALE for frozen system, solution v and mesh.

Here we used $N = 361$ on the interval $[-180, 180]$ in the non-frozen and $N = 71$ on $[-70, 70]$ in the frozen case. In both cases the onset of a periodic motion which is characterized by regular bursts of the v-component can be observed after a transient phase. For the non-frozen system the wave eventually leaves the domain whereas it converges to a time-periodic solution for the frozen system.

Both components of this solution at time $t = 1200$ are shown in Fig. 6.

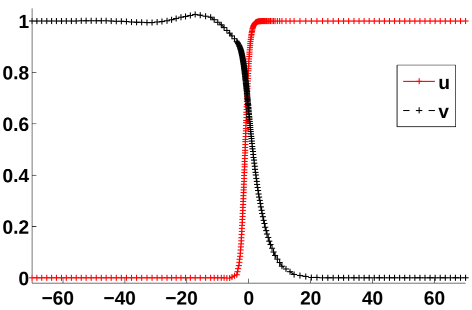

Figure 6: Autocat, ALE, u, v at $t = 1200$

In Fig. 7 we compare the time evolution of λ for a uniform mesh with the result for the ALE computation. In both cases we have used $N = 71$ mesh points. In case of the uniform mesh the solution converges to a spurious equilibrium solution, whereas the ALE solution converges to the same time-periodic solution as for the computation with a much finer mesh ($N = 1401$).

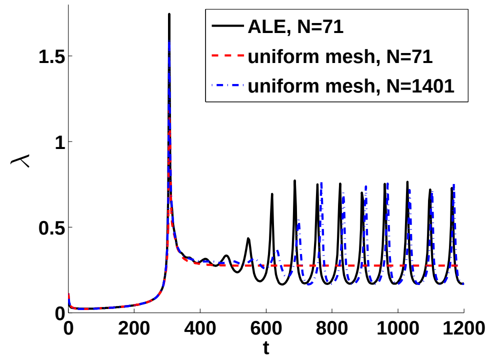

Figure 7: Autocat, ALE vs. uniform mesh, time evolution of λ.

4.3 Autocat, 2d

In two space dimensions the autocatalytic system reads

$$\begin{aligned} u_t &= \delta \Delta u - u v^m, \\ v_t &= \Delta v + u v^m . \end{aligned} \tag{8}$$

Here mesh velocities w_x and w_y in both directions have to be determined for the ALE computation. This is achieved using monitor functions M_x and M_y which measure the derivative of the solution (u, v) in x- and y-direction in the same way as in the 1d case.

As in the 1d case, there exists a stable planar traveling wave for the parameters $m = 2$, $\delta = 2$.

We compare the uniform mesh and the ALE computation with $N_x = 101$, $N_y = 51$ mesh points in x- and y-direction on a $L_x = 400$ by $L_y = 300$ rectangle, starting with the spatially periodic functions

$$u^0(x, y) = \frac{1}{1 + \exp((40 + \cos(\frac{10\pi x}{L_x}) - y)/4)}$$

and $v^0 = 1 - u^0$.

Fig. 8 shows the resulting meshes color coded with the v-component at $t = 5000$. In case of the uniform mesh one obtains a spurious solution whereas the ALE computation converges to the correct planar wave which is also obtained by using a much finer uniform mesh. Correspondingly it can be seen in Fig. 9 that the velocity λ converges to a spurious value in case of the rough uniform mesh, whereas it converges to the correct value in case of the ALE computation.

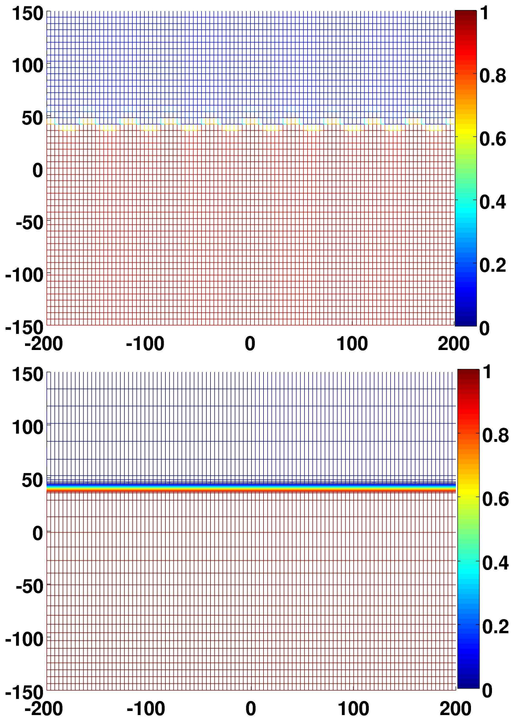

Figure 8: Autocat, ALE vs. uniform mesh, mesh at $t = 5000$.

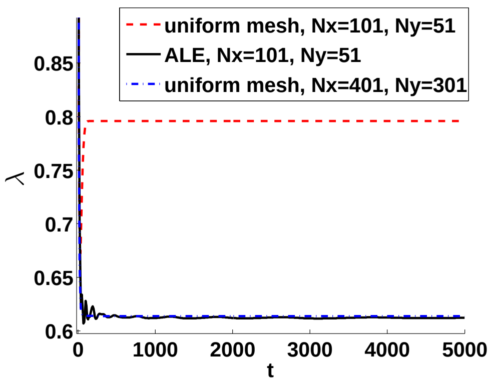

Figure 9: Autocat, ALE vs. uniform mesh, time evolution of λ.

5 Discussion

The previous numerical experiments show that a combination of the freezing method and ALE can be used to increase the accuracy of the computation of (modulated) traveling waves. While a comoving frame which moves with the velocity of the traveling wave can be computed with the Freezing method, the mesh points can be adapted to the solution using appropriate monitor functions with the ALE method. This leads to an improved accuracy for the profile v as well as for the velocity λ.

While for the Freezing method one has to solve an additional integral constraint, one needs to solve an additional PDE for the ALE method (doubling the number of unknowns for scalar problems). The convergence to a stationary solution of the PDAE (2) is quite fast for a uniform grid. This convergence is slowed down for the ALE method by the addition of (4). Here the choice of the parameter τ is crucial: if it is too large the convergence to the stationary solution is slow, if it is too small, the stiffness of equation (4) leads to numerical problems, e.g. instability of the time-stepping scheme and inverted mesh elements due to crossing of mesh points. In [9, 10] different approaches of regularizing a discrete version of this equation which is obtained using finite differences are given, which unfortunately do not have an obvious continuous analog.

6 Conclusion

We have compared different methods for computing traveling waves which are a special case of spatial-temporal varying patterns, with the ALE method, the freezing method and a combination of both. We have demonstrated the advantages and disadvantages of the respective methods using different numerical examples. Since the Freezing method can be used for more general relative equilibria, such as spiral waves in 2d [1, 17], it remains to test the combination with mesh adaptation in that case. Here the hope is that problems such as mesh entanglement which arise due the rotation of the solution can be avoided, since the rotation can be transformed out completely via the Freezing method. Together with considering ways of accelerating the convergence to the stationary solution this will be the subject of further investigations.

References

[1] W.-J. Beyn and V. Thümmler. Freezing solutions of equivariant evolution equations. *SIAM Journal on Applied Dynamical Systems*, 3(2):85–116, 2004.

[2] W. Cao, W. Huang, and R. D. Russell. A moving mesh method based on the geometric conservation law. *SIAM J. Sci. Comput.*, 24(1):118–142 (electronic), 2002.

[3] W. Cao, W. Huang, and R. D. Russell. Approaches for generating moving adaptive meshes: location versus velocity. *Appl. Numer. Math.*, 47(2):121–138, 2003.

[4] Comsol, AB. *Femlab ALE Models Reference Guide*, 2004. http://www.comsol.com.

[5] J. Donéa, J.-P. Ponthot, A. Rodríguez-Ferran, and A. Huerta. *Arbitrary Lagrangian–Eulerian Methods*, chapter 14. John Wiley and Sons, 2004.

[6] L. Formaggia and F. Nobile. A stability analysis for the arbitrary Lagrangian Eulerian formulation with finite elements. *East-West J. Numer. Math.*, 7(2):105–131, 1999.

[7] L. Formaggia and F. Nobile. Stability analysis of second-order time accurate schemes for ALE-FEM. *Comput. Methods Appl. Mech. Engrg.*, 193(39-41):4097–4116, 2004.

[8] C. W. Hirt, A. A. Amsden, and J. L. Cook. An arbitrary Lagrangian-Eulerian computing method for all flow speeds [J. Comput. Phys. **14** (1974), no. 3, 227–253]. *J. Comput. Phys.*, 135(2):198–216, 1997.

[9] W. Huang, Y. Ren, and R. D. Russell. Moving mesh methods based on moving mesh partial differential equations. *J. Comput. Phys.*, 113(2):279–290, 1994.

[10] W. Huang, Y. Ren, and R. D. Russell. Moving mesh partial differential equations (MMPDES) based on the equidistribution principle. *SIAM J. Numer. Anal.*, 31(3):709–730, 1994.

[11] E. M. Knobbe. Mesh movement governed by entropy production. In S. N. Laboratories, editor, *13th International Meshing Roundtable*, pages 265–276, 2004.

[12] S. Li and J. M. Hyman. An adaptive moving mesh method with locally refined nested grids for PDEs. Technical report, Los Alamos National Laboratory, 1999.

[13] S. Li and L. Petzold. Moving mesh methods with upwinding schemes for time-dependent PDEs. *Journal of Computational Physics*, 131:368–377, 1997.

[14] S. Li, L. Petzold, and Y. Ren. Stability of moving mesh systems of partial differential equations. *SIAM J. Sci. Comput.*, 20(2):719–738 (electronic), 1998.

[15] J. A. Mackenzie and W. R. Mekwi. An analysis of stability and convergence of a finite-difference discretization of a model parabolic PDE in 1D using a moving mesh. *IMA J. Numer. Anal.*, 27(3):507–528, 2007.

[16] A. Ramage and C. J. Newton. Adaptive grid methods for q-tensor theory of liquid crystals: A one-dimensional feasibility study. Technical report, Department of Mathematics, University Strathclyde, 2006.

[17] V. Thümmler. Numerical bifurcation analysis of relative equilibria with Femlab. In *Comsol Anwenderkonferenz 2006*. Femlab GmbH, 2006.

[18] P. A. Zegeling and H. P. Kok. Adaptive moving mesh computations for reaction-diffusion systems. *J. Comput. Appl. Math.*, 168(1-2):519–528, 2004.