

# Moving Adaptive Grids (part 3)

P. A. Zegeling

Mathematical Institute  
The Netherlands



Universiteit Utrecht

Winterschool  
December 1-2, 2018, Moscow

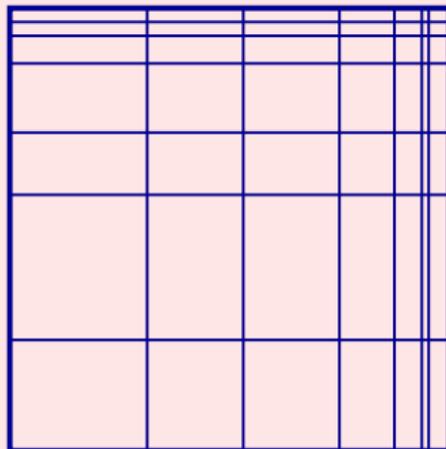
# Contents of part 3

## Possible extensions to higher space dimensions:

- Tensor-grid approach
- (Gradient-weighted) moving finite elements
- The deformation method
- The Monge-Ampère approach
- MM-PDEs
- Near-equidistribution, Winslow... ⇒ Part4

# A naive extension to 2D... [1]

Tensorgrids:



# A naive extension to 2D... [2]

## The transformed PDE model

$$\frac{\partial u}{\partial t} = \epsilon \Delta u - \beta(u, x, y, t) \cdot \nabla u + s(u, x, y, t)$$

$$x = x(\xi, \theta), \quad y = y(\eta, \theta), \quad t = \theta$$

$$\mathcal{J} := x_\xi y_\eta$$



$$\begin{aligned} & \mathcal{J} \partial_\theta u - \partial_\xi u \partial_\eta y \partial_\theta x - \partial_\eta u \partial_\xi x \partial_\theta y = \\ & \epsilon \left[ \partial_\xi \left( \frac{\partial_\eta y}{\partial_\xi x} \frac{\partial_\xi u}{\partial_\xi x} \right) + \partial_\eta \left( \frac{\partial_\xi x}{\partial_\eta y} \frac{\partial_\eta u}{\partial_\eta y} \right) \right] - \\ & \beta_1 \partial_\eta y \partial_\xi u - \beta_2 \partial_\xi x \partial_\eta u + s(u, x(\xi, \theta), y(\eta, \theta), \theta) \end{aligned}$$

(short notation)

## A naive extension to 2D... [3]

The adaptive grid ( $\sim$  transformation) satisfies

$$\begin{aligned}\partial_{\xi} [(\mathcal{S}_1(\mathcal{J}_1) + \tau \partial_{\theta} \mathcal{J}_1) \mathcal{W}_1] &= 0 \\ \partial_{\eta} [(\mathcal{S}_2(\mathcal{J}_2) + \tau \partial_{\theta} \mathcal{J}_2) \mathcal{W}_2] &= 0 \quad (\tau \geq 0)\end{aligned}$$

with smoothing operator

$$\mathcal{S} = \mathcal{I} - \sigma(\sigma + 1)(\Delta\xi)^2 \partial_{\xi\xi}^2 \quad (\sigma \geq 0)$$

and

$$\mathcal{J}_1 := \partial_{\xi} x, \quad \mathcal{J}_2 := \partial_{\eta} y$$

$$\begin{aligned}\omega_1 &= \sqrt{1 + \alpha \max_{\eta} [\partial_{\xi} u]^2} \\ \omega_2 &= \sqrt{1 + \alpha \max_{\xi} [\partial_{\eta} u]^2} \quad (\alpha \geq 0)\end{aligned}$$

# A naive extension to 2D... [4]

## THEOREM

1.  $\mathcal{J} > 0, \quad \forall \theta \geq 0, \quad (\xi, \eta) \in [0, 1] \times [0, 1]$

‘regularity of mapping’

2. 
$$\left| \frac{\partial_{\xi_m \xi_m}^2 X}{\partial_{\xi_m} X} \right| \leq 1 / \sqrt{\sigma(\sigma + 1) \Delta \xi_m}, \quad m = 1, 2$$

$(\xi_1 := \xi, \quad \xi_2 := \eta)$

‘smoothness of mapping’

# A naive extension to 2D... [5]

for  $\sigma = \tau = 0 \implies$

$$\partial_{\xi_m} [\mathcal{J}_m \omega_m] = 0, \quad m = 1, 2$$

**Solution:**

$$\xi(x, t) = \int_{x_l}^x \omega_1 d\bar{x} / \int_{x_l}^{x_r} \omega_1 d\bar{x}$$

$$\eta(y, t) = \int_{y_l}^y \omega_2 d\bar{y} / \int_{y_l}^{y_u} \omega_2 d\bar{y}$$

[Equidistribution in each direction]

# A naive extension to 2D... [6]

'Grid energy':

The grid PDEs are the Euler-Lagrange equations for minimizing

$$\mathcal{I}_1(\xi) = \int_{x_l}^{x_r} \frac{1}{\omega_1} (\partial_x \xi)^2 dx, \quad \mathcal{I}_2(\eta) = \int_{y_l}^{y_u} \frac{1}{\omega_2} (\partial_y \eta)^2 dy$$

'grid distribution represents equilibrium state of a spring system'  
(state of minimum energy)

THEOREM (semi-discretized version)

1.  $\Delta x_{i,j}(\theta) := x_{i,j}(\theta) - x_{i-1,j}(\theta) > 0, \forall \theta, \forall i, j$

'no grid points crossing'

2.  $\frac{1}{1+1/\sigma} \leq \frac{\Delta x_{i+1,j}(\theta)}{\Delta x_{i,j}(\theta)} \leq 1 + 1/\sigma, \forall \theta, \forall i, j$

'local quasi-uniformity'  $\rightsquigarrow \sigma = \mathcal{O}(1)$

# Application: a rotating cone [1]

$$\partial_t u = \Delta u + f(x, y, t)$$

$$u^*(x, y, t) = \exp(-80((x - r(t))^2 + (y - s(t))^2))$$

with

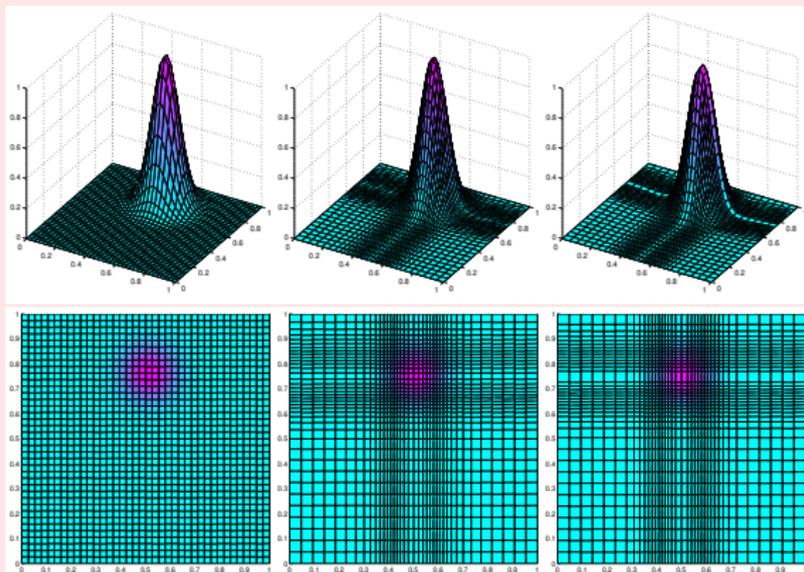
$$r(t) = 0.25(2 + \sin(\pi t)), \quad s(t) = 0.25(2 + \cos(\pi t))$$

A rotating step pulse that moves around in circles with a constant speed and with no change of shape during the movement

$39 \times 39$  grid (uniform starting grid),  $\alpha = 1$   $\sigma = 1$ ,  $\tau = 10^{-3}$ ,  
 $tol = 10^{-3}$

from Zegeling, J. of Comp. & Appl. Maths., 166, 2004

# Application: a rotating cone [2]



# Application: a whirlpool model [1]

$$\partial_t u = -\frac{v_t}{v_{t,max}} \frac{y}{r} \partial_x u + \frac{v_t}{v_{t,max}} \frac{x}{r} \partial_y u$$

with

$$r = \sqrt{x^2 + y^2 + \epsilon}, \quad v_t = \tanh(r) / \cosh^2(r)$$

$$v_{t,max} = 0.385, \quad u|_{t=0} = -\tanh(y)/2, \quad \partial_n u|_{\partial\Omega} = 0$$

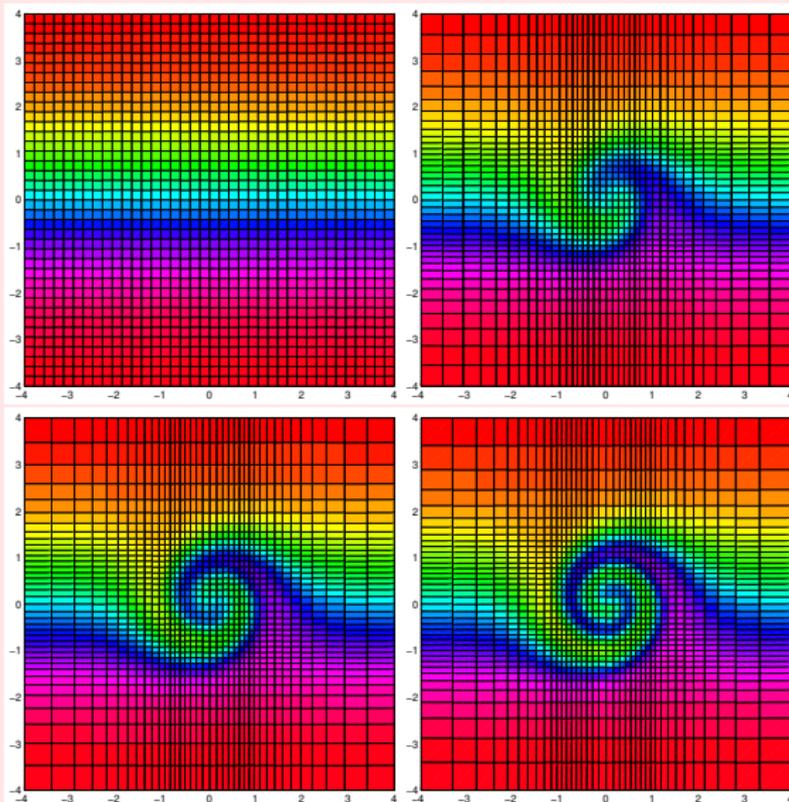
$$\Omega = [-4, 4]^2, \quad t \in [0, 4]$$

The formation of cold and warm fronts in a two-dimensional setting with a rotational velocity field  $\Rightarrow$  twists the front ( $\sim$  daily-weather maps)

39  $\times$  39 grid (uniform starting grid)

$\alpha = 1$ ,  $\sigma = 1$ ,  $\tau = 10^{-3}$ ,  $tol = 10^{-3}$

# Application: a whirlpool model [2]



# Application: Burgers' equation [1]

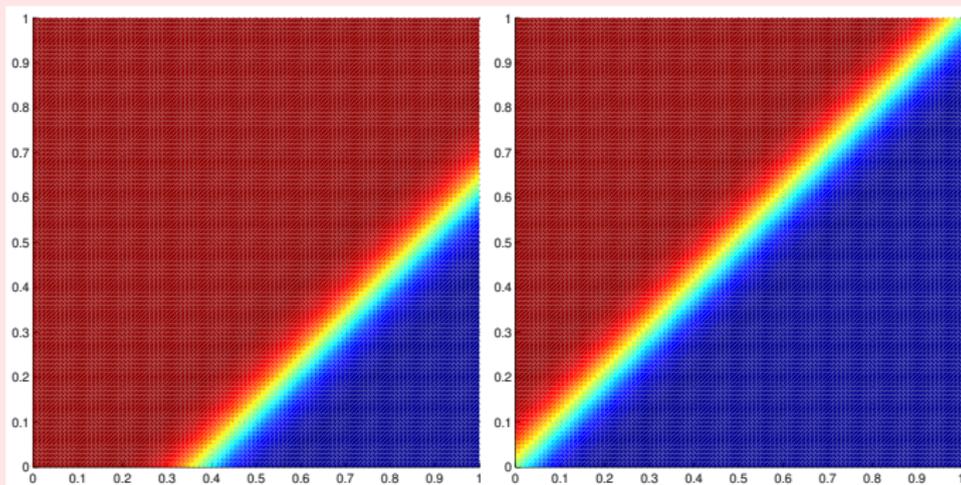
$$\partial_t u = \epsilon \Delta u - u \partial_x u - \left(\frac{3}{2} - u\right) \partial_y u$$

$$u^*(x, y, t) = \frac{3}{4} - \frac{1}{4} \frac{1}{1 + \exp \frac{-4x+4y-t+2}{32\epsilon}}$$

'Burgers' equation' (scalar version): describes a wave front with a steep transition area of thickness  $\mathcal{O}(\epsilon)$  that moves under an angle of  $135^\circ$  with the positive  $x$ -axis

('counterexample' for tensorgrid approach)

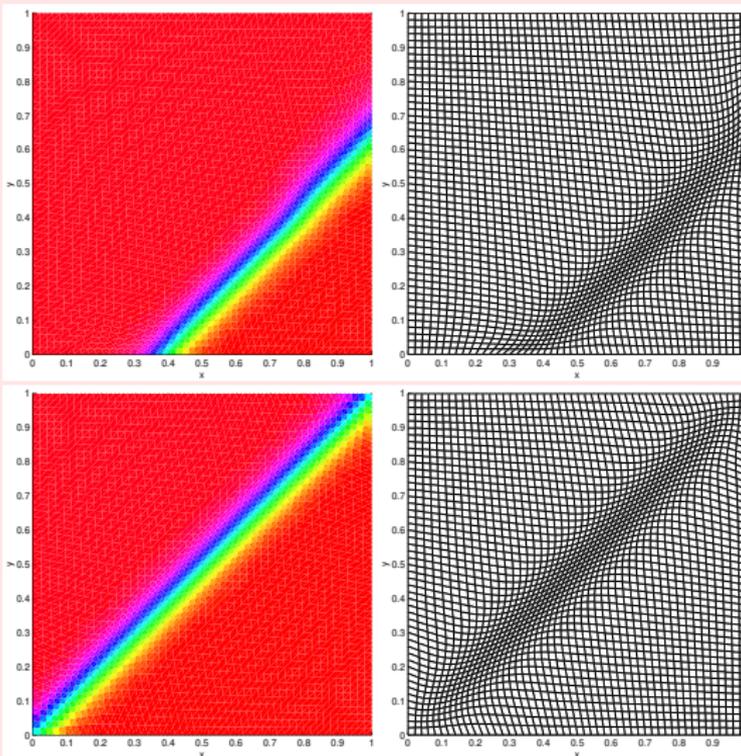
# Application: Burgers' equation [2]



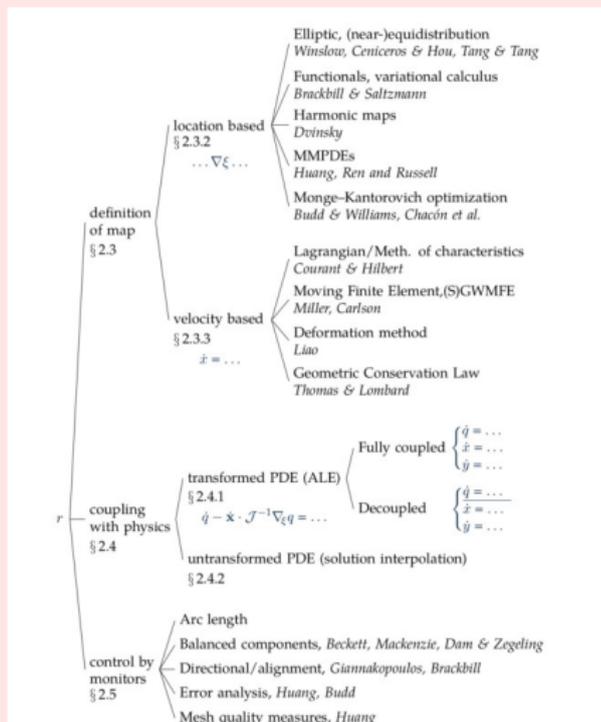
(exact solution)

# Application: Burgers' equation [3]

Adaptive moving grid solutions; see Part 4:



# Overview: adaptive moving meshes



# Moving finite elements [1]

Miller, Baines, Carlson, ...:

A two-dimensional moving grid technique (MFE) based on the minimization of the PDE residual is obtained by approximating the PDE solution  $u$  with piecewise-linear finite element basis functions (see Baines [6], Miller et al [34, 35], Zegeling [49]). There are several ways to describe this method. Here, we follow the concept of the transformation between the physical and computational domain:

$$u \approx U = \sum_{j \in J} U_j(\theta) \alpha_j(\xi, \eta), \quad x \approx X = \sum_{j \in J} X_j(\theta) \alpha_j(\xi, \eta), \quad y \approx Y = \sum_{j \in J} Y_j(\theta) \alpha_j(\xi, \eta), \quad (20)$$

where  $\alpha_j$  are the standard ‘hat’ functions on 2D having a limited support and  $J$  stands for the index set of the grid points. Substituting (20) into the time-dependent PDE model gives, in general, a non-zero PDE residual  $U_t - \mathcal{L}(U)$ . To obtain equations for the grid movement, a minimization procedure (‘least squares’) is applied with respect to the, yet unknown, variables  $\dot{U}_i, \dot{X}_i, \dot{Y}_i$  of the following quantity

$$\int_{\Omega_{\xi, \eta}} (\dot{U} - U_x \dot{X} - U_y \dot{Y} - \mathcal{L}(U))^2 \mathcal{J} \, d\xi d\eta \quad \forall i \in J. \quad (21)$$

# Moving finite elements [2]

Here  $\mathcal{J}$  denotes the Jacobian of the transformation. After re-writing (21) in the physical co-ordinates, we obtain the system

$$\begin{aligned} \int_{\Omega} (U_t - \mathcal{L}(U)) \alpha_i dx dy &= 0, \quad \forall i \in J, \\ \int_{\Omega} (U_t - \mathcal{L}(U)) U_x \alpha_i dx dy &= 0 \quad \forall i \in J, \\ \int_{\Omega} (U_t - \mathcal{L}(U)) U_y \alpha_i dx dy &= 0 \quad \forall i \in J. \end{aligned} \quad (22)$$

Working out the innerproducts and adding small regularization terms  $P_{1,2}$  and  $Q_{1,2}$  to keep the finite-element parametrization non-degenerate, yields for  $i \in J$ :

$$\begin{aligned} \sum_{l \in J} \langle \alpha_i, \alpha_l \rangle \dot{U}_l + \langle \alpha_i, \beta_l \rangle \dot{X}_l + \langle \alpha_i, \gamma_l \rangle \dot{Y}_l &= \langle \alpha_i, \mathcal{L}_i(U) \rangle \\ \sum_{l \in J} \langle \beta_i, \alpha_l \rangle \dot{U}_l + \langle \beta_i, \beta_l \rangle \dot{X}_l + \langle \beta_i, \gamma_l \rangle \dot{Y}_l + P_1(\epsilon_1^2) &= \langle \beta_i, \mathcal{L}_i(U) \rangle + Q_1(\epsilon_2^2) \\ \sum_{l \in J} \langle \gamma_i, \alpha_l \rangle \dot{U}_l + \langle \gamma_i, \beta_l \rangle \dot{X}_l + \langle \gamma_i, \gamma_l \rangle \dot{Y}_l + P_2(\epsilon_1^2) &= \langle \gamma_i, \mathcal{L}_i(U) \rangle + Q_2(\epsilon_2^2), \end{aligned}$$

# Moving finite elements [3]

where  $\beta_i = -U_x \alpha_i$ ,  $\gamma_i = -U_y \alpha_i$  and  $\langle \bullet, \bullet \rangle$  is the standard  $L_2$ -innerproduct. Using  $\eta_2 = (\dots, U_i, X_i, Y_i, \dots)^T$  as before this can be re-written as:

$$\mathcal{A}_{mfe}(\eta_2, \epsilon_1^2) \eta_2 = G_{mfe}(\eta_2, \epsilon_2^2). \quad (23)$$

The small parameters  $\epsilon_1^2$  and  $\epsilon_2^2$  serve to keep the extended mass-matrix  $\mathcal{A}_{mfe}$  and the right-handside  $G_{mfe}$  non-singular, respectively. It is worthwhile to note that the previous derivation can easily be done in higher space dimensions as well.

# Moving finite elements [4]

The more sophisticated GWMFE (see Carlson et al [14, 15]) uses an additional gradient-weighting term in the innerproducts of the form  $\langle w(\nabla U) \bullet, \bullet \rangle$ . However, in general, the results shown below hold, for the greater part, also for GWMFE, possibly with some minor modifications.

Some properties of the moving grid for MFE:

Consider now the PDE (2) in one or two space dimensions. In one space dimension it can be shown, Zegeleing et al [48], that for  $\#J \rightarrow \infty$  and  $\epsilon_1^2 = \epsilon_2^2 = 0$  the grid moves as a perturbed method of characteristics:

$$\frac{\partial x}{\partial \theta} = \beta + \delta \left( 2 \frac{u_{xxx}}{u_{xx}} - 3 \frac{\xi_{xx}}{\xi_x} \right), \quad (24)$$

where  $\xi$  is the spatial co-ordinate in the computational domain. Numerical solutions of (23) for Burgers' equation (14), clearly indicating property (24), are given in Figure 10. From equation (24) it can be derived that for steady-state situations ( $\frac{\partial x}{\partial \theta} = \frac{\partial u}{\partial t} = 0$ ) an equidistribution-like relation holds for the grid:

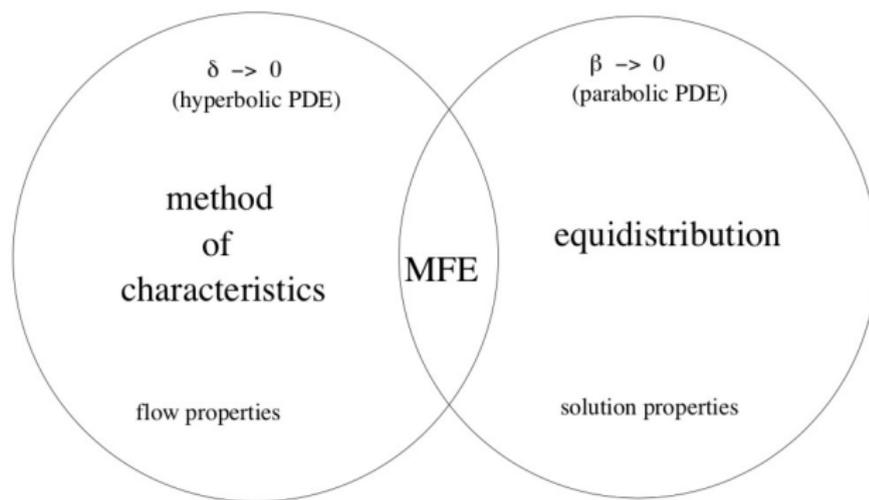
$$\frac{\partial x}{\partial \xi} |u_{xx}|^{2/3} |u_x|^{1/3} = \text{constant}. \quad (25)$$

In two space dimensions it is known that the grid moves in a similar way:

$$\frac{\partial x}{\partial \theta} = \beta_1 + \delta \phi_1, \quad (26)$$

$$\frac{\partial y}{\partial \theta} = \beta_2 + \delta \phi_2.$$

# Moving finite elements [5]



# MM-PDEs [1]

Huang, Russell, ...:

### 2.3.2.7 Huang, Ren and Russell's MMPDE approach

In 1994, Huang, Ren and Russell [HRR94] presented an adaptive moving mesh method in 1D, based on moving mesh PDEs (MMPDEs) that strive to achieve the equidistribution principle. Later, they extended some of the better MM-PDEs to two dimensions, now motivated from the theory of harmonic maps, which results in gradient flow equations [HR97a, HR99]. In 2001 Huang describes the practical aspects of the actual implementation [Hua01a]. Subsequent work of the same author concentrates on proper monitor functions, which we will discuss in Section 2.5.3.

In the MMPDE approach, the mesh map is explicitly time-dependent, i.e.,  $x(\xi, t)$ . Several one-dimensional MMPDEs are proposed; one that lies very close to equidistribution (2.21) is MMPDE5:

$$\dot{x} = \frac{1}{\tau} \frac{\partial}{\partial \xi} \left( \omega \frac{\partial x}{\partial \xi} \right), \quad (2.40)$$

where  $\tau > 0$  is a time-relaxation parameter to fit the speed of mesh movement approximately to the typical physical time scales (see page 43). Clearly, the mesh points are moved towards regions of large  $\omega$  and the mesh speed is zero when exact equidistribution is attained.

## MM-PDEs [2]

where  $\tau > 0$  is a time-relaxation parameter to fit the speed of mesh movement approximately to the typical physical time scales (see page 43). Clearly, the mesh points are moved towards regions of large  $\omega$  and the mesh speed is zero when exact equidistribution is attained.

In the general, multidimensional gradient-flow formulation [HR99], the mesh functional is generalized to:

$$I := \int_{\Omega} \left[ (\nabla \xi)^T G_1^{-1} (\nabla \xi) + (\nabla \eta)^T G_2^{-1} (\nabla \eta) \right] dx, \quad (2.41)$$

where the symmetric positive definite matrices  $G_1$  and  $G_2$  are the monitor functions. When  $G_1 = G_2 = G/\sqrt{g}$  it is a genuine energy functional (2.39), resulting in a harmonic map, but many other choices have been considered. The functional derivatives  $-\delta I/\delta \xi$  and  $-\delta I/\delta \eta$  are the directions in which  $I$  descends the fastest. The Euler-Lagrange equations yields the functional derivatives for (2.41):

$$\frac{\delta I}{\delta \xi} = -\nabla \cdot (G_1^{-1} \nabla \xi), \quad \frac{\delta I}{\delta \eta} = -\nabla \cdot (G_2^{-1} \nabla \eta). \quad (2.42)$$

These functional derivatives define the movement of the mesh points:

$$\frac{\partial \xi}{\partial t} = -\frac{P_1}{\tau} \frac{\delta I}{\delta \xi}, \quad \frac{\partial \eta}{\partial t} = -\frac{P_2}{\tau} \frac{\delta I}{\delta \eta}, \quad (2.43)$$

where  $P_1$  and  $P_2$  are operators with positive spectrum that allows one to change the descent directions and  $\tau$  changes the time scale of the mesh equation. For example, limiting the above equations to one dimension,  $P = (\frac{\omega}{\xi})^2 I$  gives the above one-dimensional MMPDE5 (2.40), where  $I$  is the identity operator. Other choices for  $P$  result in other MMPDEs by Huang et al.

Mackenzie and coworkers have employed MMPDEs for a wide range of applications. Amongst these are phase-field equations modeling fluid solidification and other state transitions [MR02, BMR06], and the Hamilton-Jacobi equations modeling front propagation [MN07].

# Deformation method [1]

Liao, Anderson, de la Peña, ...:

### 2.3.3.3 Deformation method

In 1992, Liao and Anderson [LA92] proposed a new way of adaptive mesh generation, motivated by problems in extending harmonic maps and Winslow's

variable diffusion to three dimensions. Over the following years, Liao and coworkers further developed the deformation method. This method yields a mesh map whose Jacobian determinant can be prescribed exactly, namely:

$$J = \det \nabla_{\xi} \mathbf{x}(\xi, t) = \frac{1}{\omega(\mathbf{x}(\xi, t))}, \quad \text{where } \mathbf{x} \in \Omega \subset \mathbb{R}^n, t > 0. \quad (2.73)$$

It achieves equidistribution in any number of dimensions for all times.

The main theorem is inspired by Moser and Dacorogna's work [Mos65, DM90] on diffeomorphisms on manifolds with prescribed Jacobian. The coordinate map is an automorphism, e.g., on the unit cube. If the physical domain is different from the computational domain ( $\Omega \neq \Omega_c$ ), an additional transformation can be included, e.g., a linear scaling on rectangular domains or a curvilinear transformation around an airfoil. Liao et al. [LPS94] show that this does not invalidate the original proof.

The existence of a map that satisfies the equidistribution property (2.73) is valid for any number of dimensions, the proof is by construction. We discuss a time-dependent map here, for the static case we refer to the original publications.

Suppose we have a monitor function  $\omega > 0$  that is 'normalized' such that at each time it satisfies:

$$\int_{\Omega} (\omega(\mathbf{x}, t) - 1) \, d\mathbf{x} = 0. \quad (2.74)$$

# Deformation method [2]

The idea is to evolve the mesh map  $\mathbf{x}(\xi, t)$  according to a well-chosen velocity field  $\mathbf{v}(\xi, t)$ . The first step is to find a field that satisfies:

$$\nabla_{\xi} \cdot \mathbf{v}(\xi, t) = -\frac{\partial}{\partial t} \omega(\xi, t). \quad (2.75)$$

This is a scalar equation for the unknown vector-valued function  $\mathbf{v}$ , so it is underdetermined. A vector field can always be written as the sum of the gradient of a potential and the curl of some other vector potential. Liao et al. [LJL99] neglect the second term, because they impose zero curl ( $\nabla_{\xi} \times \mathbf{v} = 0$ ) in order to allow points to move along the boundaries. Substituting this form of  $\mathbf{v}$  in (2.75) gives:

$$\mathbf{v} := \nabla_{\xi} a \implies \nabla_{\xi}^2 a = -\frac{\partial}{\partial t} \omega. \quad (2.76)$$

Now the velocity field is the solution to a Poisson equation, completed by Neumann boundary conditions that keep the points on the boundary:

$$\frac{\partial a}{\partial n} = 0.$$

The second step is to solve the ODE system:

$$\dot{\mathbf{x}} = \frac{\mathbf{v}(\mathbf{x}, t)}{\omega(\mathbf{x})}, \text{ for } t > 0, \mathbf{x}(\xi, 0) = \mathbf{x}_0(\xi). \quad (2.77)$$

These two steps will make sure that the resulting mesh satisfies equidistribution, as we will show in Theorem 2.3.3. Interestingly, though, they are also equivalent with the GCL-based approach (see next section), since the substitution of (2.77) in (2.75) yields  $\nabla_{\xi} \cdot (\omega \dot{\mathbf{x}}) = -\omega_t$ , which is (2.85).

# Deformation method [3]

**Theorem 2.3.3** (Equidistribution for deformation in  $n$  dimensions). Let  $\mathbf{x}_0$  be an initial mesh map that satisfies  $J(\mathbf{x}_0) = 1/\omega(\mathbf{x}_0(\xi), 0)$ . The time-dependent mesh map obtained from (2.76) and (2.77) satisfies  $J = 1/\omega(\mathbf{x}, t)$  for all  $t > 0$ .

*Proof.* We only need to prove  $\frac{d}{dt}(J(\mathbf{x})\omega(\mathbf{x}, t)) = 0$  for all  $t > 0$ .

$$\begin{aligned} \frac{d}{dt}(J\omega) &= \omega \frac{d}{dt}(J) + J \frac{d}{dt}(\omega(\mathbf{x}, t)) \\ &= \omega \frac{d}{dt}(J) + J(\dot{\mathbf{x}} \cdot \nabla \omega + \frac{\partial}{\partial t} \omega(\mathbf{x}, t)) \\ &= \omega \frac{d}{dt}(J) + J \left( \frac{\mathbf{v}}{\omega} \cdot \nabla \omega + \frac{\partial}{\partial t} \omega(\mathbf{x}, t) \right). \end{aligned} \quad (2.78)$$

The time derivative of the Jacobian matrix can be derived using (2.77):

$$\frac{d}{dt}J = \frac{d}{dt} \nabla_{\xi} \mathbf{x}(\xi, t) = \nabla_{\xi} \left( \frac{d}{dt} \mathbf{x}(\xi, t) \right) = \nabla_{\xi} \left( \frac{\mathbf{v}}{\omega} \right) = \nabla \left( \frac{\mathbf{v}}{\omega} \right) \nabla_{\xi} \mathbf{x}, \quad (2.79)$$

which is a matrix differential equation, for which holds in general:

$$\frac{d}{dt}X(t) = A(t)X(t) \implies \frac{d}{dt} \det(X(t)) = \text{trace}(A(t)) \det(X(t)).$$

We can now introduce the determinant in (2.79):

$$\frac{d}{dt}J = \frac{d}{dt} \det(\mathcal{J}) = \left( \nabla \cdot \left( \frac{\mathbf{v}}{\omega} \right) \right) J = \left( \frac{1}{\omega} \nabla \cdot \mathbf{v} + \mathbf{v} \cdot \nabla \left( \frac{1}{\omega} \right) \right) J. \quad (2.80)$$

Finally, the definition of the velocity field (2.75) implies  $\nabla \cdot \mathbf{v}(\mathbf{x}, t) = -\frac{\partial}{\partial t} \omega(\mathbf{x}, t)$ , so that (2.80) simplifies to:

$$\begin{aligned} \frac{1}{J} \frac{d}{dt}(J\omega) &= \left( -\frac{\partial}{\partial t} \omega + \omega \mathbf{v} \cdot \nabla \left( \frac{1}{\omega} \right) \right) \\ &\quad + \left( \frac{\mathbf{v}}{\omega} \cdot \nabla \omega + \frac{\partial}{\partial t} \omega \right) \\ &= 0. \end{aligned}$$

□

# Monge-Ampère approach

## Budd, Williams, Sulman, ...:

### 2.3.2.10 Monge-Kantorovich optimization

The latest approach for adaptive mesh generation is Monge-Kantorovich optimization. The key difference of this method with the other methods discussed, is the following. Instead of optimizing some *combination* of adaptation criteria and mesh quality measures, the Monge-Kantorovich approach *enforces* local equidistribution and then optimizes some mesh quality measure under this constraint. Monge's mapping problem [Mon81] and Kantorovich's associated optimization problem [Kan42] go back a long time. Only recently, Budd and Williams [BW06] employed it for mesh adaptation using relaxation. Delzanno et al. [DCF\*08] solve the full nonlinear system instead of using relaxation.

Monge-Kantorovich optimization aims to find a mapping that satisfies for any set  $A_c \subset \Omega_c$  (which maps to  $A := \{\mathbf{x}(\xi) \mid \xi \in A_c\} \subset \Omega$ ):

$$\int_{A_c} d\xi = \int_A \omega(\mathbf{x}, t) d\mathbf{x}, \quad \text{i.e., } \omega J = 1, \quad (2.58)$$

and—under these constraints—minimizes the point displacement:

$$\int_{\Omega_c} \frac{\|\mathbf{x} - \xi\|_2^2}{2} d\xi. \quad (2.59)$$

The constrained minimization problem can be put into variational form by including the equidistribution constraint with a local Lagrange multiplier:

$$I_{MK} := \int_{\Omega_c} \frac{\|\mathbf{x} - \xi\|_2^2}{2} + \lambda(\xi)(\omega(\mathbf{x})J - 1) d\xi. \quad (2.60)$$

The Euler-Lagrange equations imply  $\mathbf{x} - \xi = \nabla \lambda$ , i.e., the map  $\mathbf{x}(\xi) := \xi + \nabla_\xi \Phi$  is a gradient map. Inserting this into equidistribution relation (2.58) yields the *Monge-Ampère* equation to the displacement potential  $\Phi$ :

$$\nabla_\xi^2 \Phi + H(\Phi) = \frac{1}{\omega(\mathbf{x}, t)} - 1, \quad (2.61)$$

where  $H$  denotes the determinant of the Hessian matrix:

$$H(\Phi) := \frac{\partial^2 \Phi}{\partial \xi^2} \frac{\partial^2 \Phi}{\partial \eta^2} - \left( \frac{\partial^2 \Phi}{\partial \xi \partial \eta} \right)^2. \quad (2.62)$$

Budd and Williams solve their Monge-Ampère equation by approximation. Using temporal relaxation, an approximate potential tends to the exact solution over time. The Monge-Ampère equation in its relaxed form is parabolic (PMA), from which a convex potential and thus Jacobian positivity can be proved. This approach is similar to the relaxation approach that yields the MMPDEs (Section 2.3.2.7). In fact, the PMA equation simplifies to one of the MMPDEs in 1D.