Cartographic Displacement in Generalization: Introducing Elastic Beams

Mattias Bader & Mathieu Barrault
GIS Division - Geography Department - University of Zürich (Switzerland)
{mbader, barrault}@geo.unizh.ch

Abstract: This paper introduces a major improvement to the cartographic displacement technique based on the snakes [12], introduced to cartography [6] and improved since then[2]. It explains how polylines can be modeled as beam structures so as to better preserve shape and graph characteristics when vertices displacement is required. This paper first recapitulates through an example the main issues of cartographic displacement. It then introduces the mathematical fundations of the beam structure model and highlights its improvements. It then shows how this model takes place in the iterative process introduced by the snakes to better simultaneously solve multiple symbols-overlap conflicts within a road network. This geographical feature is used as an example although this method can be extended to any features modeled as polylines, as discussed all along this paper.

1 Introduction

Cartographic symbols width as minimal legible distance instanciate the need for displacing graphical representation of geographical entities apart from close hampering neighbours. If displacement means a loss of location accuracy, it also implies degradations of spatial relationships between displaced conflicting objects the other neighbours. Such information gets crucial while scale decreases: displacement as a generalisation operation must then both solve igniting conflicts but also propagate these displacements so as not to create new conflicts and preserve particular spatial relationships. This raises questions as the it can be unsderstood in the following example.

Generalisation of buildings, on an island, (fig. 1-a) illustrates the issue. Too small at the final scale, they are first enlarged to respect the minimal size (fig. 1-b). Their scaling generates an overlap between two of them, which then must be shifted apart. If the middle one is shifted along the largest translation (fig. 1-e), a new conflict occurs with the third building, which then must also be displaced. Thus, the initial displacement is diffused to the neighborood. One may also want to preserve relative distances between buildings; but then, the last building will lie in water (fig. 1-c). The displacement needs to be diffused to the island boundary, which, if merely translated, creates a conflict with the left-hand building (fig. 1-d). To avoid an infinite loop, it must be found a more adequate displacement operation. In order not to displace too many objects, the displacement diffusion must be cushioned. This can be done by decreasing variations of spatial relationships between objects. But the displacement can also be cushioned through feature's shape[1]. Fig. 1-h shows how the island boundary may fully cushions the second building displacement. But such a solution sacrifies two building locations and the island shape when alternative solutions can offer better solutions. Fig. 1-g illustrates how a better initial displacement avoids the island boundary to be involved, where figure 1-f shows how selecting the other conflicting object only transform the island's shape. Moreover the displaced section (top-left) can better cushion the displacement (Variation of curvature is less perceived on sinuous sections). The island boundary is less transformed.

Since feature cushioning efficiency relies on norm and direction of the igniting displacement, assessing which object to shift, of how and strong the cushion must be, is almost impredictable without a comprehensive model of what to preserve and to sacrify (shapes and all kind of spatial relationships: proximities, topology, clusters, prallelism, and so on...). Such model does not exist yet and would probably be extremely time-consuming. Moreover, more than one conflict occur during the generalisation process. Displacement must take into account this multiplicity. A one-step approach on each conflict can then not solve such a complex issue.

This paper does not pretend to solve all conflicts by displacement but rather proposes to describe how novel techniques brought from other fields like physics may contribute to an automated generalisation process. We

[1] We will term *propagation* the cushioning through a feature's shape and a *diffusion* a cushioning done through the spatial relations (proximities, topological connections,...

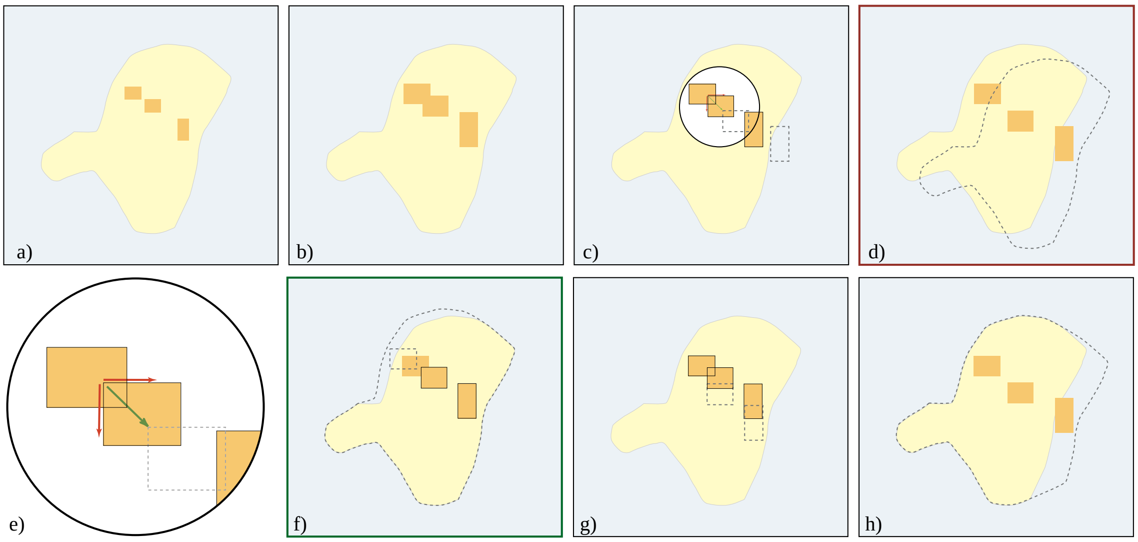

Figure 1: Displacement Issues - What to displace, where and how long, so as to involve as less neighboring features.

introduce in the next chapter the snakes principles, adapted to cartographic needs by [6]. We then enonciate an improved model of road shape behaviour against forces, based on beams modeling. We finally present results to defend the improvement and discuss this model against other approaches.

In order to simplify the reading, we only focus on the road network geogrphical feature and conflicts arising between road symbols, although it can be applied to any feature (resp. set of features) modeled by a polyline (resp. set of polylines) provided a different setup.

2 Improving the Snakes

Snakes methods has beend developed in the field of Computer Vision [12]. Based on a wrigling convergence of a polyline, Burghardt proposed a first aoplication of this method to map generalisation [6], which has been rapidly improved to better suit more stringent constraints [2].

2.1 Snakes principles

Snakes relies on two interesting principles which particularly fit to the cartographic displacement operation.

- A tiny step by tiny step approach:Snakes method is an iterative approach which slowly converge to a balance between external and internal influences.

- An optimised solution, at each step, is provided between multiple and contradictory influences. Adapted to polylines transformation, these discretised objects fit to the use of the Finite Element Method to supply a satisfying approximated solution of the optimisation of a continuous function.
  This function is the sum of external and internal energies. In cartography, external energies may correspond to conflicts and be modeled by repulsive forces applied to conflicting sections, whose norms should decrease up to 0. The stronger the conflict, the higher the external energy.
  Internal energy corresponds to the difference between the current state of each object and its ideal state. In generalisation, the ideal state will be for instance the initial road shape, or the initial arrangement of buildings, modeled as a graph whose shape to be preserved.

This approach allows to apply simultaneous forces to each vertex (by merging them in an average force) but moreover to different vertices at the same time. The optimisation integrates them: if opposite strong forces applied to close vertices may finally break the shape but solve conflicts, the next step brings it back to a closer state to the ideal since only inner energy must be minimised.

For linear features like roads, one of the issue consists in modeling modification of shapes. A road can provide various sections of heterogeneous shapes. Burghardt's approach models this difference by the sum of first and second estimated derivatives variations along the road. This solution treats independantly x and y road axis . The real two-dimensional aspect is not taken into account. If sinous roads do not suffer of this approximation, stiffer roads showed the limits of this measure. We thus propose a better modeling of polyline's shape modification based on the beams: The theoratical model of a beam structure is reminded. We then describe how to adapt this model to generalisation needs, and then get more in details of the whole process. We conclude with examples and comparisons with previous techniques.

2.2 Roads as Beam Structure

At its core, our method treats road as elastic components. *Each road segment is identified as a single beam*. By a beam, we understand a long and slender elastic bar. The beam elements are *connected to vertices over stiff hinges* (see Figure 2). Deformation thus does not take place in the vertices, but along the beams: the model is (largely) made independent of the vertex sampling.

Junction nodes are also modeled as hinges inside this structure. Their behavior does not differ from any other hinge, except that they make more than two slots available to plug in the incident beams.

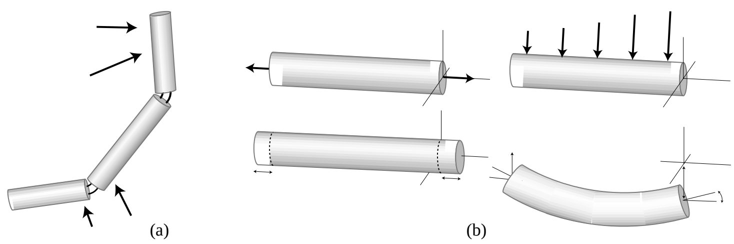

Figure 2: (a) A road is interpreted as a set of elastic beams connected by stiff hinges. (b) Beams under loads are stretched/compressed or bent.

2.3 Proximity Conflicts as Foundation of Forces

To trigger displacement, proximity conflicts between road segments are understood as source of forces. These forces act on the beam elements and provoke its deformation. By pushing conflicting elements apart, displacement conflicts are solved. The deformation of the beams affects the entire structure. The global deformation models the propagation of displacement through the road network.

The deformation of beams under load, described in the mechanics of material, is constituted by two characteristics (see Figure 2) : on the one hand, a force applied in direction of a beam's longitudinal axis exerts a *compression or stretching*; on the other hand, under transversal force, the beam is bent (*bending*). Both can be mathematically formalised.

3 Technical Foundation

In this section, a numerical method is described how the concept established in the former section can be implemented. The section is divided into 3 parts. First, the theory of beams is short addressed. Then, a Finite Element

Method (FEM) is described to solve the beam model for the forces, deformations, and displacements developer in following sections. Finally we pass over to an iterative treatment of the problem, by turning the FEM-resulting equation system into an evolutionary system.

3.1 Beams

The governing differential equation for beam deflection in two dimensions is derived in textbooks on mechanics of material (see e.g. [9] or [3]). Here, we separate the description of the beam deformation with respect to the applied forces: If the force act along the axis of the beam (normal stress), we observe an elongation or compression; If the beam is transversely loaded, it is bended (and displaced).

Since a finite element method is used later, we use a variational formulation of the problem. This variational formulation is related to the energy stored in the elements. The energy in a beam under *axial load* (compression) is expressed by

$$J(u) = \frac{1}{2}\int_0^L AE\left(\frac{du}{dx}\right)^2 - \int_0^L Awudx \tag{1}$$

where A denotes the cross-sectional area of the beam, L the length of the beam, E the modulus of elasticity (a material constant), and $u(x)$ the displacement function along the beam (describing compression resp. elongation).

The energy under *transversal load* (bending) is given by

$$J(v) = \frac{1}{2}\int_0^L EI\left(\frac{d^2v}{dx^2}\right)dx - \int_0^L wvdx \tag{2}$$

where I denotes the moment of inertia of the beam, and $v(x)$ the y-displacements of the beam. The first terms in (1) and (2) correspond to the internal strain energy. The second terms give the potential of the external loading. Here and throughout this article, we assume A, E, and I to be constant in a beam. But these values may vary later between beams.

3.2 Finite Element Method

We sketch in short the use of a finite element analysis for the implementation of the beam structure problem. A reader not familiar with finite element methods is referred to [8] and [5].

3.2.1 Element Equations with Respect to Bending

First, the equilibrium equations for a general beam element under transversal load (bending) are built. Therefore, the displacement function $v(x)$ is expressed in terms of the displacements v_1, v_2 and rotation θ_1, θ_2 at the beam nodes P_1, P_2. Using Hermite polynomials

$$\begin{aligned} H_{01}(x) &= 1 - 3\frac{x^2}{L^2} + 2\frac{x^3}{L^3} & H_{02}(x) &= 3\frac{x^2}{L^2} - 2\frac{x^3}{L^3} \\ H_{11}(x) &= x\left(1 - 2\frac{x}{L} + \frac{x^2}{L^2}\right) & H_{12}(x) &= x\left(\frac{x^2}{L^2} - \frac{x}{L}\right) \end{aligned} \tag{3}$$

this deflection function $v(x)$ may be written as

$$v(x) = \mathbf{H}\,\mathbf{v} \tag{4}$$

where

$$\mathbf{H} = (H_{01}\quad H_{11}\quad H_{02}\quad H_{12})\mathbf{v} = (v_1\quad \theta_1\quad v_2\quad \theta_2)^T \tag{5}$$

Inserting the displacement function (4) in the functional (2) gives the potential energy in matrix form

$$U(v) = \frac{1}{2}\int_0^L \mathbf{v}^T\mathbf{H}''^T EI\,\mathbf{H}''\mathbf{v}\,dx - \int_0^L \mathbf{v}^T\mathbf{H}^T w\,dx$$

In respect of the principle of minimum potential energy, minimize $U(v)$ with respect to $\mathbf{v}$ and set the result equal to zero to obtain the matrix equation that defines the local beam finite element

$$0 = \int_0^L \mathbf{H}''^T EI\,\mathbf{H}''\mathbf{v}\,dx - \int_0^L \mathbf{H}^T w\,dx \tag{6}$$

By substituting the second derivatives of the Hermitan polynomials (3) in (6), performing the matrix manipulations, and integrating the terms, the matrix equation for a general beam element under transversal load is found

$$\frac{EI}{L^3}\begin{pmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{pmatrix}\begin{pmatrix} v_1 \\ \theta_1 \\ v_2 \\ \theta_2 \end{pmatrix} = \begin{pmatrix} wL/2 \\ wL^2/2 \\ wL/2 \\ -wL^2/2 \end{pmatrix} \tag{7}$$

3.2.2 Element Equations with Respect to Axial Load

The matrix equation for a general beam under axial load is derived the same way. Again, the horizontal deflection $u(x)$ is expressed in terms of the nodal displacements u_1 and u_2. A Linear interpolation satisfies the continuity requirements. Therefore, with

$$G_1 = \frac{l-x}{l} \qquad \text{and} \qquad G_2 = \frac{x}{l}$$

the deflection $u(x)$ is may be expressed by

$$u(x) = \mathbf{G}\,\mathbf{u} \tag{8}$$

with $\mathbf{G} = (G_1 \;\; G_2)$ and $\mathbf{u} = (u_1 \;\; u_2)$. The same procedure as for transversal load is followed now: Insert (8) in (1), minimized the resulting potential energy with respect to u, perform the matrix manipulations, and integrate the equation. The final equations describing a general beam under axial load is then

$$\frac{AE}{L}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} N_1 \\ N_2 \end{pmatrix} \tag{9}$$

where N_1 and N_2 denote axial forces acting at the start- and end- node of the elastic bar.

3.2.3 Combining and Orienting the Element Equations

Axial and transverse deformations have been treated uncoupled so far. We gain the complete local element matrix equation by *combining* (7) and (9)

$$\frac{E}{l}\begin{pmatrix} A & 0 & 0 & -A & 0 & 0 \\ 0 & 12I/l^2 & 6I/l & 0 & -12I/l^2 & 6I/l \\ 0 & 6I/l & 4I & 0 & -6I/l & 2I \\ -A & 0 & 0 & A & 0 & 0 \\ 0 & -12I/l^2 & -6I/l & 0 & 12I/l^2 & -6I/l \\ 0 & 6I/l & 2I & 0 & -6I/l & 4I \end{pmatrix}\begin{pmatrix} u_1 \\ v_1 \\ \theta_1 \\ u_2 \\ v_2 \\ \theta_2 \end{pmatrix} = \begin{pmatrix} N_1 \\ R_1 \\ M_1 \\ N_2 \\ R_2 \\ M_2 \end{pmatrix} \tag{10}$$

where N_i denote the axial forces, and R_i and M_i the shear and moment end actions dependant upon the external beam loading.

Note that this equation holds only for a beam placed in a local coordinate system wherein the x-axis is oriented aligned to the beam's principle axis.

To express the element properties with respect to a common global coordinate system, equation (10) has to be adjusted for the orientation of the beam. This requires to supplement the previous steps by a rotation of input values. This step hides no pitfalls (see [4]), therefore we give the directly the resulting system equation.

$$K^G = \frac{E}{L}\begin{pmatrix} Ac^2 + \frac{12I}{L^2}s^2 & \left(A - \frac{12I}{L^2}cs\right) & -\frac{6I}{L}s & -\left(Ac^2 + \frac{12I}{L^2}s^2\right) & -\left(A - \frac{12I}{L^2}\right)cs & -\frac{6I}{L}s \\ & As^2 + \frac{12I}{L^2}c^2 & \frac{6I}{L}c & -\left(A - \frac{12I}{L^2}\right)cs & -\left(As^2 + \frac{12I}{L^2}c^2\right) & \frac{6I}{L}c \\ & & 4I & \frac{6I}{L}s & -\frac{6I}{L}c & 2I \\ & & & Ac^2 + \frac{12I}{L^2}s^2 & \left(A - \frac{12I}{L^2}\right)c^2 & -\frac{6I}{L}c \\ & \text{symmetric} & & & As^2 + \frac{12I}{L^2}c^2 & -\frac{6I}{L}c \\ & & & & & 4I \end{pmatrix} \tag{11}$$

with $c = \cos\alpha$ and $s = \sin\alpha$.

3.2.4 Assembling

Equation (11) gives the necessary algebraic equations describing the characteristics of each element of our system. To constitute each system, we have to consider the orientation α of the beam, and incorporate the object and material coefficients A, I, and E.

The next step now is to combine all these equations to form a complete set governing the composite of elements. This system assembly procedure is based on our insistence of compatibility at the element nodes. At nodes, where elements are connected, the values of the unknown generalized nodal variables (u, v, θ) are the same for all elements connecting at that node. When these generalized displacements are matched at the nodes, the nodal stiffnesses and nodal loads for each of the elements sharing the node are added to obtain the net stiffness and the net load at the node.

Performing this assembly procedure, we end with a global equation system of type

$$\boldsymbol{K}\boldsymbol{d} = \boldsymbol{f}$$

where $d = (...)$ mirrors the nodal displacements and rotations, and $f = (...)$ corresponds to the forces acting on the beam network.

3.3 Boundary Conditions

Defining *forces only* is from a mathematical and cartographical point of view not feasible.

Mathematically speaking, the stiffness matrix $\boldsymbol{K}$ is a priori singular. No solution for the equation system is found in its current form. At least two essential boundary conditions have to be integrated into $\boldsymbol{K}$ to turn it regular.

This integration of essential boundary conditions is also cartographically welcomed. To ensure smooth transitions of roads when the partition is reinserted in a given map, we specify for the partition end-nodes zero displacement and zero slope deflection. This ensures smooth transitions when the partition is reinserted in a given map (C^0 and C^1 continuity).

Based on the cartographic background, we specify as further essential boundary condition on all junction nodes θ to be zero. This preserves the relative orientation of the junction incident roads, which is an important cartographic property of the junction.

For each vertex, we have to define either displacements (essential boundary conditions) or forces (natural boundary conditions). The former one provide a strong operator to cushion a shifted vertex, as described below. The latter method already proved [2] its capability to integrate optimised solutions between multiple conflicts. This is described in the next section.

3.4 Propagation

If setting the displacement of some vertices to a value unequal zero (in vector $\boldsymbol{d}$), and setting the forces for all other vertices to zero (in vector $\boldsymbol{f}$), displacement is studied in absence of external forces. This handling models propagation: the method computes the deformation of the beam structure for which its inner energy is minimized subject to the specified shiftings (no external forces). Such an application is useful when the required displacements are known as a result of a previous analysis. Displacements are 'built in' to the solution and consequently are satisfied exactly.

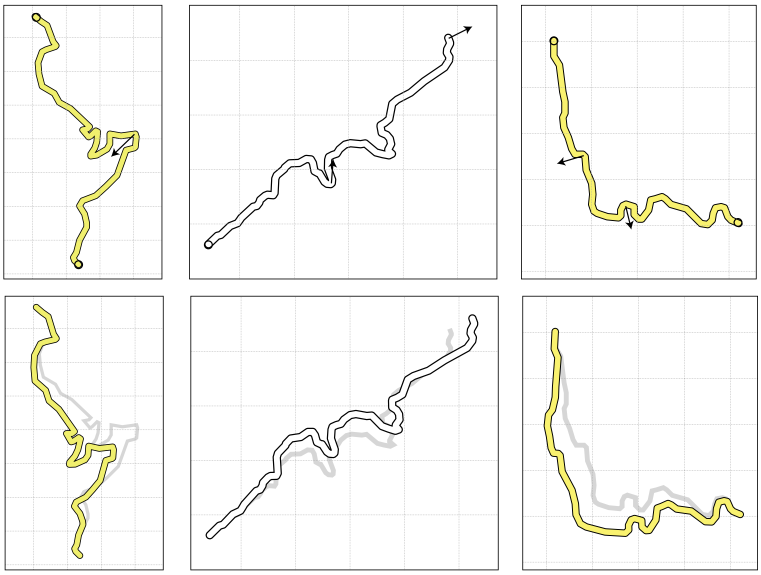

Figure 3: Cushioning shiftings within roads that are interpreted as chains of elastic bars. The initial line geometry, together with the required displacement is shown in the top row. Deformations by means of our method are shown below. On the middle and right road, more than one displacement vector is applied.

Figure 3 illustrates how solving the linear equation spreads the cushioning from igniting shiftings to the boundary conditions, namely the fixed nodes. These beams are equaly allowed to strech and to bend.
These examples use the ending nodes of the roads as boundary conditions, but it can easily be extended [2] to take a set of connected roads into account, while preserving connections and junctions angularities, as shown in fig. 5.

This beam structure models provides a powerful tool which allows to cushion vertices displacements through the road's shape and further to connected roads shape when it does not fit the requested displacement. The notion of fitness takes two entries in the beam model since the beam is driven by its bending and compressing capabilities. The way the setup emphasises one against the other makes the inner energy cost varying with respect to the angularity between the displacing vector and the beams orientations (cf. fig. 4).

When setting a road to more cushioning by streching and compressing, the minimisation will apply more cushioning those of the beams that are parallel to the shifting vector where perpendicular are less involved. A setup which emphasises the bending will make a "perpendicular" trade-off. A more complex setup could for instance prefer beams to bend when they belong to curved sections of a road while they better should strech or compress when modeling straight sections.

But the major issue remains how to find the right shifting. Since a good cushioning relies on the shifting vector characteristics AND since the right shifting vector relies on the neighborood, this application of the beams can not be fruitful in a dense environment where neighboring shapes are too involved. Snakes method proposed a more adapted iterative process which can get highly improved with the beams structure modeling.

4 The Whole Process: Slowly but Repeatedly

As illustrated in the introduction, defining the right shifting before-hand is rapidly impossible in a dense neighborood. Although propogation proposed by beams may be of interest for particular applications (manual editing

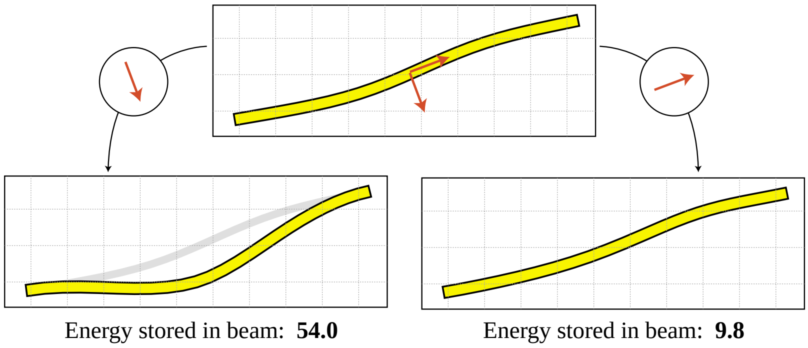

Figure 4: Regarding the displacement direction, cushioning cost fluctuates with respect to the priority given to either the bending or stretching behaviour of the beams. Here above, stretching is prefered. An orthogonal displacement requires then a higher cost.

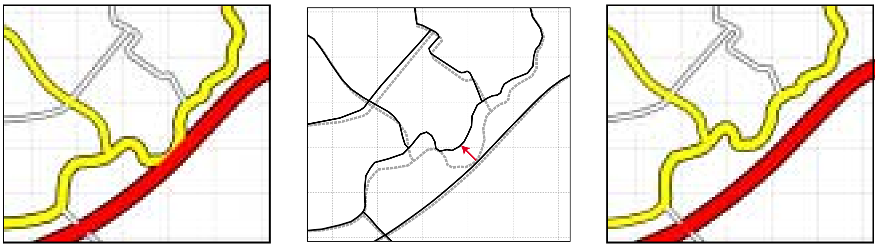

Figure 5: Situation before propagation (left). A displacement vector, imposed on the beam structure, is cushioned (middle). Situation after propagation (right).

for instance), its best application so far comes from the mix to the iterative process used by the snakes.

4.1 Displacement by Forces

The first chapter brought in the idea of balancing shape preservation (inner energies) and outer conflict (external energies). This approach has been chosen since it is extremely cumbersome to find a set of displacement vectors that would resolve a conflict situation in a single step. Rather, forces push onto the beam structure until it has taken on a situation where proximity conflicts are solved.

Obviously, the forces required to perform this deformation are not known beforehand. It is therefore hardly successful to define forces on the vertices and hope that the result meets the cartographic demands. If working with forces, only an iterative treatment is successful: Rather than one definite push, we will slowly put repeatedly push on the beams.

4.2 Inter-line Forces

Line segments between different lines, which are closer than a minimal threshold, have to be separated. To release deformations, forces act on the beams. It is a cartographic reasoning that defines the strength and the direction of these forces. At the heart of our force definition are the two following assumptions. First, the closer two lines, the higher the forces on the lines. Second, a valid displacement direction for a vertex in conflict points in the opposite direction of the reported minimal distance.

A simple approach that meets these criteria is described. Let us assume that the partition consists of n lines, denoted by $\mathcal{L}_1, \ldots, \mathcal{L}_n$. We focus a vertex P on an arbitrary line $\mathcal{L}_j$, $j \in (1, \ldots, n)$. The force $\mathbf{f}_P$ acting on this vertex P is defined by

$$\mathbf{f}_P^{(t)} = \sum_{\substack{i=1,\ldots,n \\ i \neq j}} \frac{\mathbf{v}_i}{|\mathbf{v}_i|} \frac{(r_{ij} - \min(|\mathbf{v}_i|, r_{ij}))}{r_{ij}}$$

where

r_{ji} : required min. distance between $\mathcal{L}_i$ and $\mathcal{L}_j$
$\mathbf{v}_i$: vector from P to the closest point on $\mathcal{L}_i$

Here, $\frac{\mathbf{v}_i}{|\mathbf{v}_i|}$ defines the direction of the force, while the latter term determines its magnitude (normalised to $[0, 1]$). Forces rise only if the distance between the vertex and a neighboring line drops below the tolerance distance r_{ij}. This distance r_{ij} respects the symbol width of the roads $\mathcal{L}_i$ and $\mathcal{L}_j$ and incorporates a minimal separation distance ('hardcore distance') defined by the user. The upper index t in $\mathbf{f}_P^{(t)}$ has been added to express the time dependent character of forces.

This model is rather rudimentary. Cartographic reasoning is poor. The force direction is solely determined by the shortest distance between a vertex and the conflicting lines. Later results will show that the force direction computed this way does not always exhibit the best displacement direction, which actually rely on neighboring avalaible room and involved road's shape characteristics. Also the strength of forces could be expressed differently. One could also incorporate the length of the symbol overlap zone to derive a force magnitude. Whether it is worth to establish a more sophisticated theory on how to apply forces is not studied here.
Nevertheless, it must be pointed out that pushing forces may also be calculated against other features than road symbols overlaps. Any conflicting cartographic feature may provide its own repulsion that this model will integrate as easily as too close roads.

4.3 Iterative Process

In contrast to most engineering tasks, we do not know the strength of the forces we want to apply on the structure beforehand. Rather, the forces are varied until proximity ensuring deformations are provoked. To meet this condition, we have to pass over to an iterative (evolutative) treatment.

The origin of such a handling, namely the treatment of snakes with a third energy term to represent the velocity energy of the lines, is not investigated (see [15]); merely the resulting formulas are overtaken. Reformulating the evolution problem using finite differences in time (with time step γ) we obtain

$$(\mathbf{1} + \gamma\mathbf{K})\,\mathbf{d}^t = \mathbf{d}^{t-1} + \gamma\mathbf{f}^{t-1} \tag{12}$$

where $\mathbf{1}$ denotes the identity matrix, and the upper sup-script denote the time (see [7]). This yielded to an implicit method, where the displacements at time t is derived using the displacements at time $t-1$ and the forces acting on this current situation $t-1$.

In each time step, we obtain a linear equation system. We compute the solution using a Cholesky factorization of $\mathbf{1} + \gamma\mathbf{K}$. This factorization needs be computed only once, since this term is constant over time. Note that K is very sparse; this property should be used for reducing storage and CPU requirements. Yet, this was not used in the current implementation.

A time step consists of the computation of the forces $\mathbf{f}^{(t-1)}$ and the subsequent application of (13) with these forces and the current line deformations. This results in new displacement functions $\mathbf{d}^{(t)}$ from which the new positions of the lines are derived. The next time step, i.e $\mathbf{f}^{(t)}$, is then computed based on this new geometry. The number of iteration is currently still set by the user. It was not investigated when to stop the loop since any further iteration will not derivate the result from its reached best state. Nevertheless, this investigation should still be carried on to improve speed performance.
Actually, our final equality to solve is described in matrix terms by as follow:

$$(\mathbf{1} + \gamma\mathbf{K})\mathbf{d}^{(t)} = \mathbf{d}^{(t-1)} + \gamma\mu\mathbf{f}^{(t-1)} \tag{13}$$

Our definition of forces computes magnitudes in the range of zero to one. These forces do not suffice to exert a deformation of the beams. The term μ had to be added as 'force-multiplier', scaling the forces to a reasonable ratio to the inner shape preserving strains.

The iteration process is guided by the parameters $m = \frac{1}{\gamma}$. m defines the inertia of the beam structure. The higher m, the less displacement can be expected in one time-step. The displacement is then weak and has local impact only. The propagation of the displacement through the line requires several time steps.

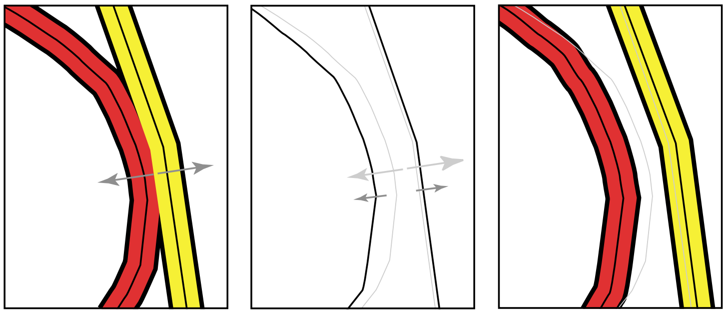

Figure 6: Instance of the iterative process in two steps. Repulsive Opposite forces are applied on both conflicting roads. Since the blue one (on the right) is stiffer, bending is less allowed, and perpendicular displacement is expensive. The minimisation proposes a small displacement. The red road, more sinuous can more cushion the force. At the next step, conflict still exists but require weaker forces. Although both lines have applied same forces, stifness differences make the most sinuous more cushioning the required displacements.

4.4 Example

A situation with symbol overlap, together with the solution computed by means of the presented beam method, is shown in Figure 7. Also, evolution of energy is displayed there.

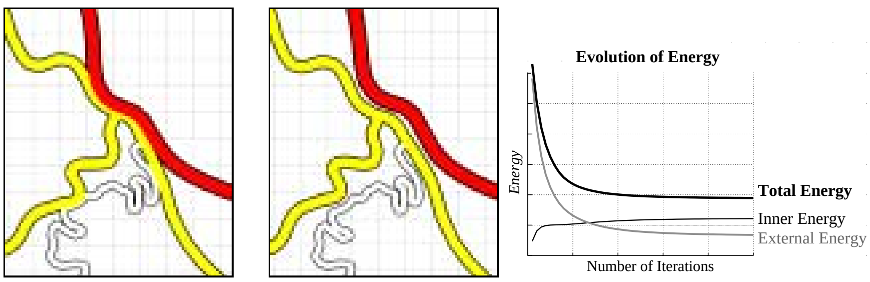

Figure 7: The evolution of energy (right) in the solution of a symbol overlap.

The example calls attention to two important issues.

First, it shows how the energy of the system does slowly strive towards a (local) minimum. This slow convergence raises the question when the iteration should be terminated. We did not investigate this question in our prototype; the number of iterations is simply defined by the user.

Second, from the visualization of the energy evolution it is apparent that the minima of inner and outer energy is reached at a state where the external energy does not vanish. The external forces, however, should not be in balance with the inner ones, but vanish instead. The presence of external energy indicates that lines are still too close. Hence, the required minimal distance between lines is not met. The separation distance depends hereby on the strength of the beam deformation and therefore on the magnitude of the conflict.

This problem is diminished by making the external forces dominating the inner ones. Yet, this treatment does not yet solve the problem entirely. Applicating the algorithm twice is also beneficial. The geometry of the lines after the first run is used as reference for the second run. This corresponds to a reset of the inner energy back to zero.
Alternative solution can also be proposed. A full generalisation process allows the use of other algorithms which then may be applied when the displacement process didn't fully sole all conflicts. Thius process integrates many conflicts to solve and ligthen most of it. Local tiny remaining conflicts should probably be solved by small shiftings easily propagated. On the other hand, bad results (still big conflicts or shapes too modified) may also highlight that situation can not be handled by single displacements but may require stronger generalisation decisions, like removing roads or simplifying some shapes. A constraining setup in a dense and full of conflict area can not be solved by displacement neither by any algorithm nor a human operator without breaking some of the initial rules. Optimisation techniques such as this one may also take part as the necessary assessing process to better define which generalisation operator or sequence of operators to finally apply.

4.5 Results

Two other examples closer to real situation, i.e. in more dense areas, are presented below in Figures 8 and 9.

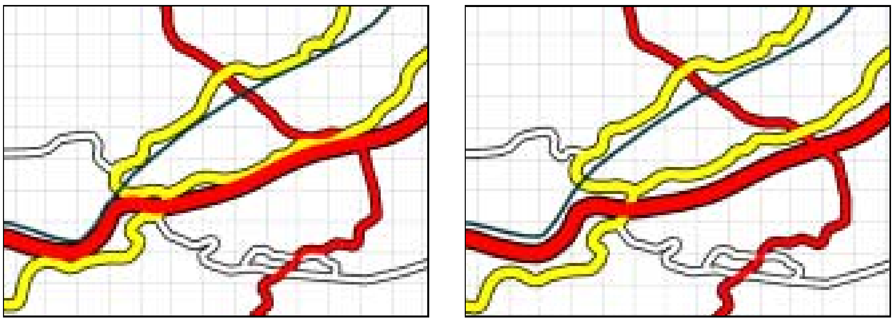

Figure 8: Situation before displacement (left) and after displacement (right).

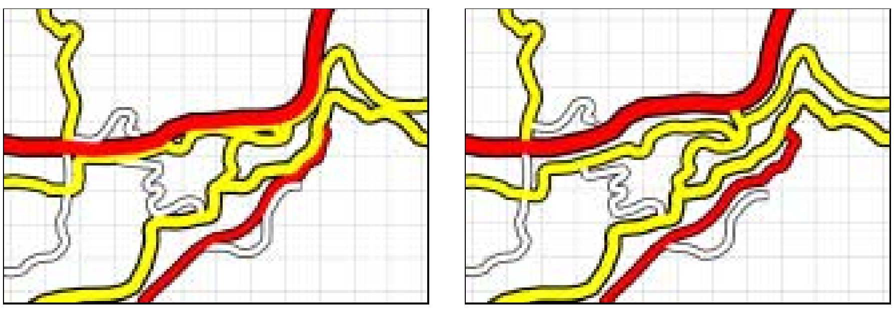

Figure 9: Situation before displacement (left) and after displacement (right).

The following data has gone into computation. First, regarding the setup of the stiffness matrices, we respected the road attributes (roads varying in symbology hold different attributes) to make major roads more resistant against bending (increasing I for more important roads). The parameter A, controlling compression, was not changed. No other line characteristics were used yet. To constrain propagation, a small term of positional accuracy was added.

Second, regarding the handling of junctions, which require more complex displacement directions in order to preserve alignments and perpendicularity of connected roads, we computed local corrections beforehand. This correction consisted in the enlargement of those segments that were hidden entirely by the symbolization of other roads ending in the same junction. The pre-computed enlargements are easily integrated in the equation system (see [1] for more information about it).

Third, with regard to the iterative process, we used the displacement triggering forces from the snakes algorithm. The number of iterations was set to 200. This high digit guarantees an advanced convergence, but was probably not required. The velocity of displacement propagation, defined over γ, was held constant for both examples.

Computer Resources The partition displayed in Figure 9 is described by 380 vertices. This requires the allocation of a 1140×1140 matrix ($\sim$ 1.3 million entries), since each beam vertex is described by 3 parameters (x_i, y_i, θ_i)). For computer memory- and CPU-reasons (Cholesky factorization) we recommend to take advantage of the sparsity of the matrix. Not even 1% of the matrix element differ from 0. As we did not yet make use of this property in our prototype implementation, computation time became notable.

Also, the distance measures to define the external energy consume sizeable resources, esp. as this computation is required in each iteration. Careful preprocessing is indispensable (spatial indexing). The algorithm is written in Java. On a Sun Ultra 30, it required 2.4 minutes CPU-time to compute the solution of Figure 9. The implementation definitely leaves room for accelerations.

5 Discussion

5.1 Quality of Results

The quality of displacement computed based on the presented beam structure is satisfying. While proximity conflicts are solved, the shape of roads is hardly distorted. The cartographic quality is in as much surprising, as we did not yet make use of the entire potential of beams: the beam properties did not yet vary within the lines, and thus no special cartographic reasoning was superimposed.

Main strengths of the presented method are:

- **Shape** The shape of lines under stress is modified in a cartographic reasonable manner, since important curvature changes are preserved. This proves the adequateness of beams for the model of propagation.

- **Multi-line Conflicts** Conflicts between several lines are solved. Proximity conflicts in dense areas are not just pushed forth and back, but indeed solved. The displacement is thereby shared reasonably between the roads. One major strength of such an approach holds in the diffusion of needed displacement. a bounding line without initial conflict may final be displaced soas to supply room to a conflict lying in the center of the selected dataset.

- **Network / Topology** In contrast to existing methods, displacement is propagated through the road networs. Junction nodes are not hold fixed. This is a prerequisite for successful displacement as intersections themselves are often in proximity conflict.

- **Possibilities for Cartographic Control** The description of bending and compression allows a detailed control of the displacement process. Further research is required here to make use of its full potential. Currently, the prototype does not make use of these possibilities, since the prerequisite analysis of roads is not sufficiently integrated.

Weaknesses are:

- **Computing Resources** The computational cost is large, both regarding computer memory and processing time, although much can be done to lighten this drawback.

- **Energy Minima / Equilibrium of forces** Depending on the complexity of a situation, an equilibrium of forces (resp. a minimization of energy) is found at different levels. The higher the remaining external forces (resp. the higher the remaining displacement potential), the farther away is the displacement from the required threshold. The method does not guarantee that the user-defined minimal distance is reached. To overcome this weakness, we could re-apply the method or, probably, a change in the underlying force definition would bring relief.

The algorithm presented in this work is implemented as *prototype*. It succeeded in solving proximity conflicts for the given selected, small datasets. Although the datasets in use contained complex situations, a validation of the concepts provided in the prototypes by means of large datasets is still outstanding.

Associated with this lack of broader experience is our inability to predict possible difficulties to setup suitable *algorithm parameters*. The algorithm contains parameters to stiffen element shapes, to scale forces, and to control the iterative process. Parameters *per se* are not a problem; it is in the nature of cartographic generalization that even the perfect displacement algorithm has to contain parameters by which the user can define the importance of different impacts to the solution. However, a simple way of setting these values is crucial, in the sense that the impact of a parameter's modification is predicatable. We believe that our algorithm meets this criterion, and that, for instance, machine learning techniques, such as described in [14], could accomplish this task.

5.2 Comparing Beams and Snakes

The setup of our method is, indeed, closely related to the snake techniques. The fundamental difference, and also advantage of beams, lies in the way how roads are modeled.

With beams, we treat the road segments as two-dimensional objects, which allows us to guide their bending and compression. This model has a serious impact on the solution, since lines are made aware of best displacement directions. Consider for example Figure 10. This situation manifests a perpendicular pattern of straight roads. The beam technique resolves the conflict in a cartographically reasonable manner (by preserving straightness and angularity of the entities) when we raise the parameter I to impede bending. The problem is then solved exclusively by compression and expansion.

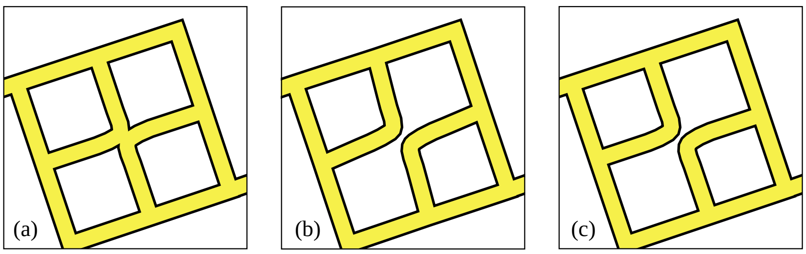

Figure 10: Roads in conflict (a). The solution of this conflict by an algorithm based on snakes (b), and an algorithm based on beams (c).

Directional sensitivity is a behavior not supported by snakes. With snakes, lines are first parameterized with respect to the line length s into the two forms $x(s)$ and $y(s)$. In the iterative process, these parametric forms are handled widely independently of each other. This handling makes it impossible to argue about the segment directions. No explicit control of bending and stretching is possible. The character of a situation can only insufficiently be preserved, as shown in Figures 10 and 11. The latter picture definitely highlights how beams better cushion displacements. Each step always sligthly displaces the conflicting section more along the main orientation of the road (since streching is prefered to bending). Even if the initial conflict is the same, results are different since shapes are distinct, which snakes does not guarantee.

5.3 Comparing beam elasticity and Hojholt's elasticity

It is the intuitive point of departure which is related to Hojholt's algorithm (see [10]),. Map objects are interpreted to consist of elastic material, and their deformation leads to a solution of proximity conflicts.

The method vary in the type of elasticity and in the way deformations are achieved. Hojholt's method is thought for areal displacement. It provides no control of line shapes and is therefore not suited for the application towards linear features. Further, it is not investigated in [10] and [11] how forces can be adjusted to provoke the required displacement. An iterative adjustment of forces over a minimization of internal and external energies is not forseen in Hojholt's algorithm. Yet, it could be adapted to this kind of handling.

6 Conclusion

This paper introduced the concept of *elastic beams* to road displacement in map generalization. The approach takes up on existing energy minimization methods, but overcomes noted weaknesses of other methods. Although several optimisation techniques already exist, snakes methods turned out to be the best appropriate to preserve linear features characteristics. Nevertheless, such an operator didn't meet the cartographic requirements when applied to straighter linear features. By integrating a real two dimensial control of shape's behaviour, the beam model significantly improves the snakes approach, based on energy minimsation and iterative process.

This paper shows how the concept of beams can be adapted to cartographic generalisation. The modeling of bending and stretching provides manifold ways to guide displacement. The algorithm has the ability to solve conflicts between several lines, to spread displacements through entire networks, and to clear symbol overlaps around junctions, while preserving line shapes sufficiently. The shown results clearly exceed the cartographic quality achieved by algorithms known in the literature.

One the one hand, this work supports the assumption of the superiority of optimization methods over sequential methods for road displacement. A global method evaluates *simultaneously* all the forces that act on (external forces) and within (internal forces) the observed object. The result of this simultaneous evaluation is that, even though forces are determined with a restricted local scope of reasoning, a deformation with global attributes is computed. If some forces are misaligned to the general trend of correction, they are overruled by the sum of the assembled other forces. Displacement is not computed detached from propagation; both processes are merged and executed at the same time. The solution of a proximity conflict is not solved locally, but the 'backyard' (spatial context) of roads is taken into consideration. At the mean time, shape distortion at the local level (e.g. individual bend, junction) is kept under control by the underlying beam model. This combination of both local and global control explains the power of the presented algorithm. It is important to notice that our optimization method does not make superfluous cartographic analysis nor reasoning. Only a thorough understanding of the road and network shapes allows to setup the algorithm parameters such that the quality of the results is maximized. Also, it is the analysis and improvement of junctions by means of cartographic considerations that makes the method suited for real world data. The benefit of optimization techniques is that they can manage all the information which are locally specified and put them in a global context. With sequential methods, we do not have the tools and concepts available to let all the available cartographic knowledge flow into a displacement computation.

One the other hand, resolving all conflicts and well preserving shapes can not be fully guaranteed in any situations because of the balance proposed by the minimisation of both energies. But such a failure can also be a first step to a better evaluation of the spatial area to manage in order to improve the best sequence of generalisation operators to apply trhough beahviours likes attempts and backtracks as proposed in the european project AGENT ([13]).

Acknowledgments

The authors express thanks to Dr. D. Burghardt and Dr. P. Hojholt for their valuable input to our work. Parts of this research was funded by the European Union Information Technologies Programme ESPRIT No. 24939 (AGENT).

References

[1] M. Bader. Energy Minimization Methods for Feature Displacement in Map generalization. Ph.D. thesis, University of Zurich, Switzerland, 2001.

[2] Barrault, M., Bader, M., and Weibel, R. Topology Preserving Conflict Removal between Symbolized Roads in Cartographic Generalization: Extending Snakes Methods. In: *Abstract for GIScience2000*.2000.

[3] A. Bedford and K. M. Liechti. Mechanics of Materials. Prentice-Hall, Inc., 2000.

[4] J. Betten. Finite Elemente für Ingenieure 1. Springer, Berlin Heidelberg, Germany, 1997.

[5] G. R. Buchanan. Theory and Problems of Finite Element Analysis. Schaum's Outline Series, McGraw-Hill Inc., 1995.

[6] Burghardt, D. and Meier, S. Cartographic Displacement using the Snakes Concept. In: W. Foerstner and L. Pluemer (editors), *Semantic Modeling for the Acquisition of Topografic Information from Images and Maps*. Birkhaeuser-Verlag, Basel. 1997.

[7] L. D. Cohen and I. Cohen. Finite-element methods for active contour models and balloons for 2-d and 3-d images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1131–1147, 1993.

[8] D. Henwood and J. Bonet. Finite Elements - A gentle introduction. Macmillan Press Ltd, 1996.

[9] R. C. Hibbeler. Mechanics of Materials. Prenitce-Hall, Upper Saddle River, New Jersey, fourth ed., 1999.

[10] P. Højholt. Solving local and global space conflicts in map generalization using a finite element method adapted from structural mechanics. In Proccedings 8th International Symposium on Spatial Data Handling, pages 679–689. Vancouver, Canada, 1998.

[11] P. Højholt. Solving space conflicts in map generalization: Using a finite element method. *Cartography and Geographic Information Science*, vol. 27(1):65–74, 2000.

[12] Kass, M., Witkin, A., and Terzopoulos, D. Snakes: Active Contour Models. In: *Proceedings of the First International Conference on Compute Vision.*, pp. 259–268. IEEE Computer Soc. Press. 1987.

[13] Lamy, S., Ruas, A., Demazeau, Y., Jackson, M., Mackaness, W., and Weibel, R. The Application of Agents in Automated Map Generalisation. In: *Proceedings of the 19th ICA/ACI Conference*, pp. 1225–1234. Ottawa. 1999.

[14] S. Mustière. Apprentissage automatique pour la généralisation cartographique. Ph.D. thesis, Université Pierre et Marie Curie (Paris 6), 2001. In press.

[15] W. M. Neuenschwander. Elastic Deformable Contour and Surface Models for 2-D and 3-D Image Segmentation. Ph.D. thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 1995.

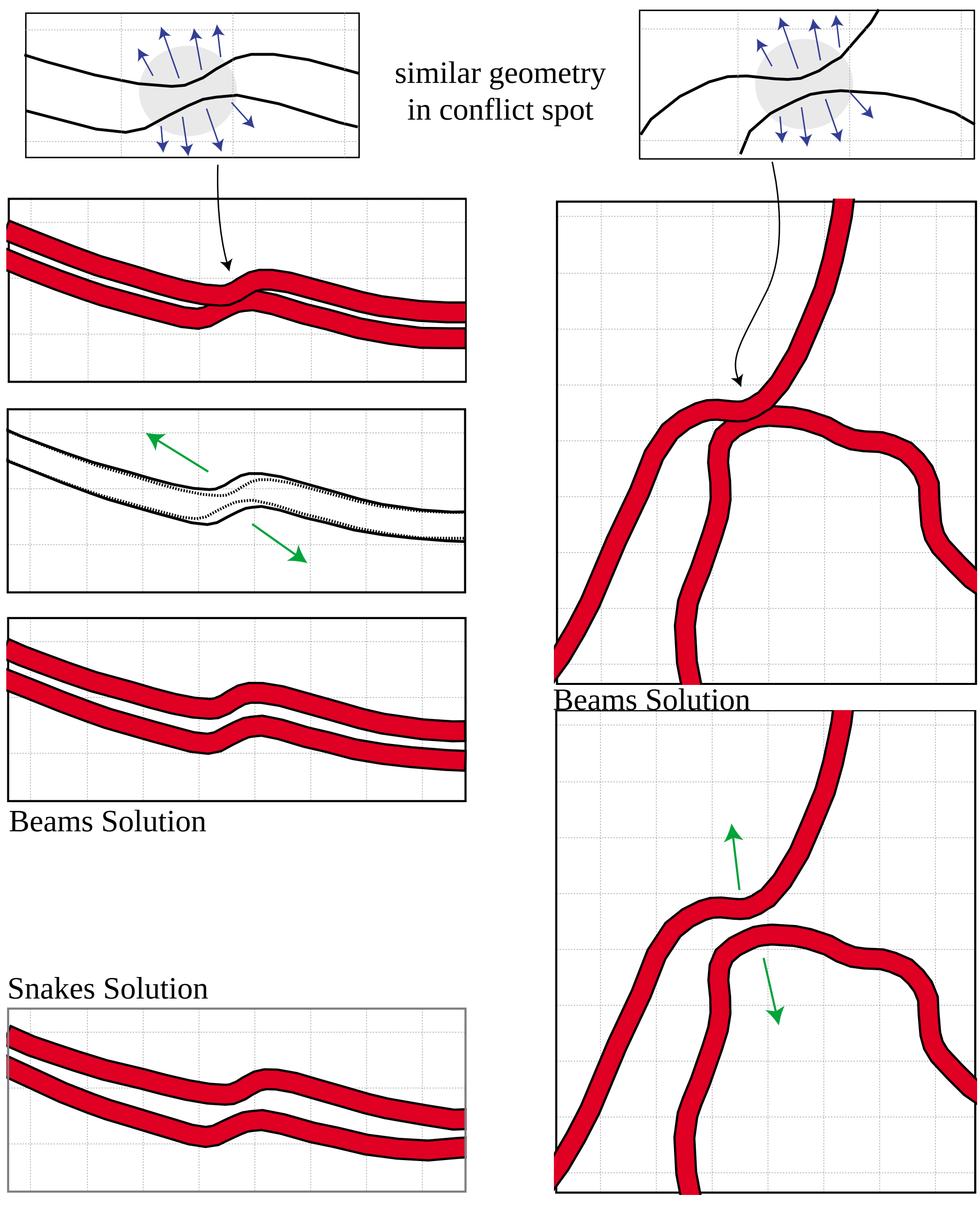

Figure 11: Influence of the model on the solution - Stretching is emphasised against bending - Only most parallel road sections to the displacements cushion the forces.