Meshing of heterogeneous unbounded domains

Benjamin Kettner[1] and Frank Schmidt[1]

Zuse Institute Berlin, Takustrasse 7, D-14195 Berlin, Germany kettner@zib.de

Traditionally, methods for solving partial differential equations such as the Finite Element method and the Finite Difference method can not deal with open region problems. On the other hand, when applying these methods, the computational domain has to be truncated to keep the computational costs at a reasonable level. To resolve this discrepancy, many different approaches have been followed. Techniques that treat the interior as well as the exterior problem by discretising the underlying partial differential equation rely on a discretisation of the exterior domain that satisfies several crucial conditions. We will discuss the problems these conditions pose and present an algorithm that will provide such a discretisation if one exists.

1 Background

The ambivalence between bounded computational domains and unbounded problem formulations is inherent in many applications. Each method applied to resolve this discrepancy requires the coupling of interior and exterior domains. This is either accomplished by the construction of transparent boundary conditions or by combining differentiation techniques for the solution of the coupled problem. The techniques we consider are the Perfectly Matched Layers (PML) Method as formulated by Bérenger in [1] and the Pole Condition formulated by F. Schmidt in [2]. Both require a discretisation of the unbounded exterior domain. Yet, while the triangulation of inhomogeneous bounded interior domains via constraint delaunay algorithms and Rupperts algorithms for refinement and mesh quality enforcement is already very elaborate [3, 4, 5], to our knowledge the discretisation of inhomogeneous exterior domains was not approached so far.
Surprisingly enough, it is sometimes non-trivial to find discretisations of heterogeneous exterior domains and we will show that even in the two dimensional case we can find simple examples where such a discretisation does not exist at all.

2 Discretisation in two dimensional space

In this section we will start by defining the conditions a discretisation of the exterior will have to fulfil in order to be admissible for our application. We will then introduce an algorithm that can produce such discretisations and prove the existence of solutions and the consistence of the algorithm. Both our applications require a discretisation of the exterior domain with quadrilaterals (two dimensions) or prisms (three dimensions) for our discretisation to be admissible we will require the following conditions:

A1: The quadrilaterals have to have one side on the boundary of the computational domain and the other side infinitely far from it
A2: Each quadrilateral or prism has to be homogenous in its interior. So if the inhomogeneities in the exterior domain such as waveguides in optics, are depicted as infinite rays leaving the computational domain, these rays must be contained in exactly one quadrilateral or prism for all times.
A3: For our application these prisms are cut off parallel to the boundary of the computational domain at a finite distance from it. It is necessary for both methods, that the so-created edges parallel to the computational domain form a closed polygon or in the three dimensional case a closed polyhedron, a scaled and distorted version of the computational domain.

While the conditions A1 and A2 are merely local criteria, condition A3 is a global condition to be fulfilled which is one of the difficulties to be mastered to gain a valid discretisation of the exterior domain.

2.1 Challenges

We will first depict the challenges of creating a valid discretisation and then show an example where no valid discretisation can exist. For our example we will choose the unit square as computational domain. Assume light scattering off an object containing four waveguides that leave this computational domain, each on one side of the square, and stretch to infinity. Now let these waveguides stretch into the directions $\mathbf{c_1}, \mathbf{c_2}, \mathbf{c_3}$ and $\mathbf{c_4}$ numbered counter clockwise starting from the waveguide leaving the domain through the edge on the x-axis (c.f. Fig. 1).
If one considers the corners of the computational domain, rays for the discretisation of the exterior, that attach to these corners have to lie within a cone that is spanned by the prolongation of the edges. If an inhomogeneity is given, this cone is narrowed by a side parallel to this inhomogeneity going through the corresponding corner. In Fig. 1 at the edges of the computational domain, these narrowed cones are marked grey and the discretisation rays are depicted as dashed lines. Note that each inhomogeneity only affects one cone. This is not the general case since in computational domains with inner angles larger than 90° some inhomogeneities might affect several edges.

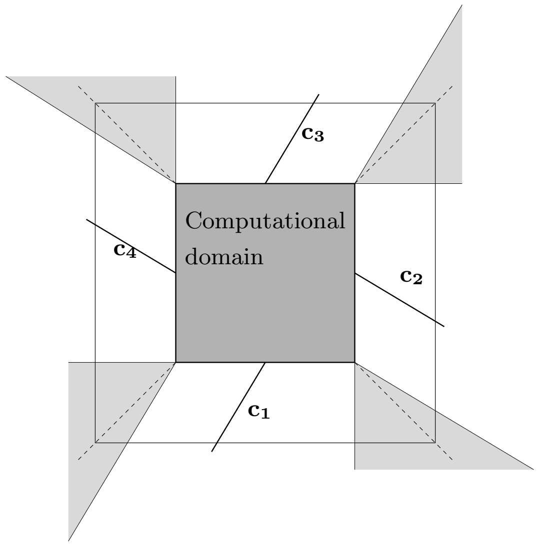

Fig. 1. Four waveguide inhomogeneities leaving the computational domain each through one edge. Dashed lines are exemplatory discretisation rays.

Let α_i denote the angle between $\mathbf{c_i}$ and the edge through which the inhomogeneity defined by $\mathbf{c}_i$ leaves the computational domain.

Lemma 1. *Let the computational Domain be the unit square and the inhomogeneities* $\mathbf{c}_i$ *be defined as before. Let* $0 < \alpha_i < \frac{\pi}{4}$ *for* $i = 1, ..., 4$*. Then there exists no discretisation of the exterior domain that satisfies the conditions A1, A2 and A3.*

Proof. The situation is depicted in Fig. 2. If the first quadrilateral of an exterior domain discretisation is cut off at a distance d_1 from the computational domain, then the closest cut off distance for its neighbour such that a line connecting the corner of the computational domain with the corner created by both cut offs for trigonometrical reasons has a distance of $d_2 = \frac{d_1}{\tan \alpha_1}$ to the computational domain. Since $\alpha_1 < 45°$, $d_2 > d_1$. So if all four inhomogeneities have angles less than 45°, by applying the same argument at each corner of the computational domain, we see that the minimal distance for the last cutoff is $d_4 > d_1$. Via d_1, the x-coordinate of the intersection between the first and the last cutoff is already fixed while $-d_4$ determines the y-coordinate. For the discretisation to be conforming to the given criteria, said intersection point has to lie in the cone formed by the x-axis and $\mathbf{c}_1$. Yet a line parallel to the second edge of the computational domain with a distance of d_1 to it, intersects this cone only in the interval $[0, -d_1 * \tan(\alpha_1)]$ which contains only points whose y-coordinate is less than d_1. Thus both lines can not intersect in the given cone and thus not form a valid discretisation of the exterior domain.

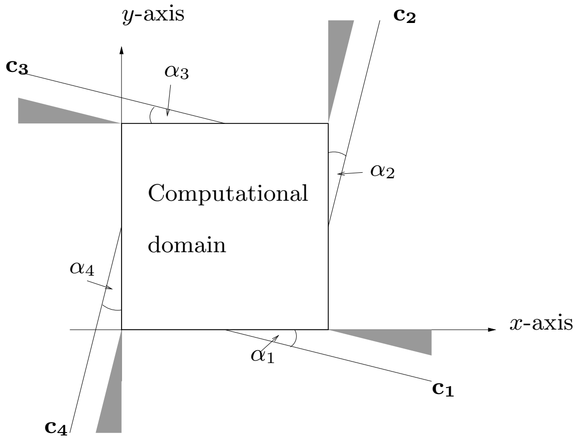

Fig. 2. Schematic of a computational domain together with constraints that make a valid discretisation of the exterior domain impossible.

2.2 Algorithmic solution

We will now present an algorithmic solution of the problem of creating a valid discretisation of the exterior domain if it exists and subsequently prove the existence of a solution and the consistence of the algorithm. The algorithm is given for 2D computational domains for graphic reasons, but it may also be extended to 3D computational domains.

Theorem 1. *Let the following algorithm find appropriate* ξ_i *in step 5 and thus not terminate in step 6:* **Algorithm:**

1. *Embed the problem into* $\mathbb{R}^3$ *("lifting"), leaving the computational domain in the* (x_1, x_2) *plane;*
2. *Define an axis* Ξ *that is orthogonal to the* (x_1, x_2) *plane and intersects the computational domain in an interior point;*
3. *For each edge* e_i *of the computational domain, define a plane* E_i *containing* e_i *and a point on* Ξ *having* $-\xi_i$ *as* x_3*-coordinate;*
4. *At each corner* V_i *of the computational domain, form the intersection of the planes* E_{i-1} *and* E_i *containing the adjacent edges* e_{i-1} *and* e_i *thus obtaining lines of intersection* $S_i \subset \mathbb{R}^3$*;*
5. *Select appropriate* ξ_i *such that the projections of the* S_i *into the* (x_1, x_2)*-plane do not conflict with any constraints given by inhomogeneities and do not intersect each other;*
6. *If such* ξ_i *exist, continue. Otherwise exit, there is no valid discretisation;*
7. *Intersect all* S_i *with a plane parallel to the* (x_1, x_2)*-plane;*
8. *Project the* S_i *into the* (x_1, x_2)*-plane. The points of intersection formed in the last step now define the corners for the quadrilaterals required for a discretisation of the exterior domain.*

Then the following holds true:

1. **Existence:** *The algorithm it produces a valid discretisation of the exterior domain, if one exists.*
2. **Consistence:** *For each valid discretisation of the exterior domain, there exist* ξ_1, ..., ξ_n *for which the algorithm will produce this discretisation.*

Proof. The first proposition is evident from the construction principle. If the algorithm terminates, in step 5, appropriate ξ_i were chosen. The resulting lines of intersection of the E_i will then form a valid discretisation of the exterior domain and since each E_i contains one edge e_i of the computational domain, the endpoints of the projected cut off intersection lines will form edges parallel to the edges of the computational domain which in turn form a closed polygon thus resulting in a valid discretisation of the exterior domain. To prove the second proposition, we lift the scaled computational domain created by the given discretisation of the exterior domain to a plane in $\mathbb{R}^3$ that is parallel to the (x_1, x_2)-plane. The corners of this lifted scaled version of the computational domain are then connected to the corners of the original computational domain with rays, yielding the E_i and thereby unambiguously defining the ξ_i.

2.3 Conclusions

The algorithm presented here may easily be extended to three dimensions. Yet the selection of the ξ_i in step 5 is an open issue that will require utilisation of optimisation algorithms for non-convex, non-linear programming problems. Also the lifting of the surface grids onto the parallel sides of the prisms in 3D is an open issue that will have to be tackled in the near future. Thus the first steps towards a fully automatic discretisation of the exterior domain have been presented, while more work will still have to be done to arrive there.

References

1. J. Bérenger, (1994) A perfectly matched layer for the absorption of electromagnetic waves. In: Springer, J. Comput. Phys.
2. F. Schmidt (2002) Solution of Interior-Exterior Helmholtz-Type Problems Based on the Pole Condition Concept: Theory and Algorithms. Habilitation Thesis, Free University Berlin, Fachbereich Mathematik und Informatik
3. J. R. Shewchuck (2002) Delaunay Refinement Algorithms for Triangular Mesh Generation In: Computational Geometry: Theory and Applications
4. P-L. George (1991) Mesh Generation: Application to Finite Elements, J. Wiley & Sons
5. J. F. Thompson, B. K. Soni and N. P. Weatherill (1999) Handbook of Grid Generation, CRC Press
6. Benjamin Kettner (2006) Ein Algorithmus zur prismatoidalen Diskretisierung von unbeschränkten Aussenräumen in 2D und 3D unter Einhalten von Nebenbedingungen. Diploma Thesis, Free University Berlin, Fachbereich Mathematik und Informatik