Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR-TP-2000-371

On the recovery of C^0-continuity of block-by-block adapted multi-block grids

R. Hagmeijer

Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR-TP-2000-371

On the recovery of C^0-continuity of block-by-block adapted multi-block grids

R. Hagmeijer

This report is based on a presentation to be held at the 7th International Conference on Numerical Grid Generation in Computational Field Simulations, Whistler, Canada, 25-28 September 2000.

Division:	Fluid Dynamics
Issued:	August 2000
Classification of title:	Unclassified

Contents

12 figures

(13 pages in total)

On the recovery of C^0-continuity of block-by-block adapted multi-block grids

R. Hagmeijer

National Aerospace Laboratory NLR
P.O. Box 90502, 1006 BM Amsterdam, The Netherlands
email: hagm@nlr.nl, r.hagmeijer@wb.utwente.nl

Abstract

An algorithm to recover C^0-continuity of a block-by-block adapted multi-block grid is proposed. Starting point for the algorithm is the notion that any adaptation or modification of a single-block structured grid is equivalent to a map from the unit cube onto itself. The algorithm is based on two types of operations: averaging of different grids and continuity recovery by modification of grids. The averaging operation is based on interpretation of grids in terms of monitor functions to drive adaptation. The continuity recovery operation is based on employing a modified form of Coons patch interpolation. An example in two-dimensions is presented which clearly demonstrates the potential of the proposed algorithm in the sense that C^0-continuity is recovered while the resolution and local orthogonality of the block-by-block adapted grid are approximately preserved.

Introduction

Consider a two-block grid consisting of two cube-shaped blocks that have one face in common. Suppose that initially there exists a boundary-conforming grid in each of the two blocks. As a consequence, two grids are present at the common face, each of them being a subset of the nodes of the grid in a neighbouring block. In the initial situation the two grids at the common face are identical. When we adapt the grids in the blocks separately, the two adapted grids at the common face are not identical in general. We will then call the compound grid discontinuous, based on the following definition:

Definition 1 *We will call a multi-block grid $\underline{C^0\text{-continuous}}$ if the following three conditions hold:*

1. *In each of the blocks there exists a boundary-conforming grid that is the image of a uniform grid in the unit cube under a bijective map with a Jacobian that is positive at all points of the unit cube.*
2. *In each face that is part of more than one block the grids associated with the different blocks are identical.*

3. *In each edge that is part of more than one face the grids associated with the different faces are identical.*

Similar to the problem of C^0-continuity at a common face of two blocks, the same kind of problem occurs at a common edge of a number of faces, introducing more complexity into the multi-block problem. The objective of the present paper is:

Objective: To provide an algorithm to modify a block-by-block adapted multi-block grid such that C^0-continuity over block interfaces is obtained, subject to the following two restrictions:

1. Adaptation characteristics should be preserved,
2. Orthogonality should be preserved approximately.

The first requirement evidently aims at keeping the modification of the block-by-block adapted grid as limited as possible in order to not destroy the obtained resolution. The second requirement is derived from general accuracy considerations.

The strategy chosen for the C^0-continuity algorithm is to employ a step-by-step approach, i.e., using a number of subsequent modifications of the maps that define the block-by-block adapted grid.

Global description of proposed algorithm

Consider the single-block part of an initial multi-block grid depicted in Fig.1 and the same part in Fig.2 after employing a block-by-block adaptation algorithm. The proposed C^0-continuity recovery algorithm is based on two types of operations:

1. Averaging of grids in block interfaces,
2. Continuity recovery by modification of grids in block and face interiors.

The averaging operation is indicated in Fig.3, showing new grid point locations on the two block-block interfaces, obtained by averaging two grids per interface, each of them being a subset of the nodes of the grid in a neighbouring block. The continuity recovery operation is indicated in Fig.4, showing a modified grid connected to the requested grid point locations on the block-block interfaces, while preserving the resolution and orthogonality depicted in Fig.3 to some degree.

Using these two operations we can define the C^0-continuity recovery algorithm in terms of four consecutive steps following block-by-block adaptation:

1. Construct averaged grids in block-faces based on grids in adjacent blocks,

2. Construct averaged grids in block-edges based on averaged grids in adjacent faces obtained in the first step,
3. Recover continuity between averaged grids in block-faces and averaged grids in block-edges by modification of the averaged grids in block-faces,
4. Recover continuity between grids in blocks and modified grids in block-faces by modification of the grids in blocks.

Equivalence of grid adaptation/modification and maps

An important notion in the work presented here is that single block grid adaptation/modification is equivalent to constructing a boundary-conforming continuous invertible map from the unit cube onto itself. To see this, we use the property of structured grids that any single-block grid in a given block $\overline{\Omega}_B$ in the physical domain can be generated by applying a boundary-conforming continuous invertible map x,

$$x : [0,1]^3 \mapsto \overline{\Omega}_B \subset I\!R^3, \tag{1}$$

to a uniform grid x_{ijk} in the unit cube:

$$x_{ijk} = x((\frac{i}{N_1}, \frac{j}{N_2}, \frac{k}{N_3})^T), \tag{2}$$

with $i \in \{0, ..., N_1\}$, $j \in \{0, ..., N_2\}$, $k \in \{0, ..., N_3\}$, where N_1, N_2 and N_3 are the number of cells in the three different coordinate directions in the unit cube.

Similarly, one could construct an adapted/modified grid by using an alternative map $\tilde{x}$,

$$\tilde{x} : [0,1]^3 \mapsto \overline{\Omega}_B \subset I\!R^3. \tag{3}$$

The maps x^{-1} and $\tilde{x}$ provide a map p from the unit cube onto itself:

$$p : [0,1]^3 \mapsto [0,1]^3, \quad p(\xi) = x^{-1}(\ \tilde{x}(\xi)). \tag{4}$$

Hence the alternative map $\tilde{x}(\xi)$ can be written as:

$$\tilde{x}(\xi) = x(p(\xi)), \tag{5}$$

which shows that single block grid adaptation/modification is equivalent to constructing a boundary-conforming continuous invertible map $p(\xi)$ from the unit cube onto itself. Fig.5 illustrates the compound map Eq.(5), where the map A represents $p(\xi)$ and the map M represents $x(p)$. The two domains in Fig.5 represent the parametric domains associated with the adapted and initial grid respectively.

It is noted that similar reasoning can be applied to grids in block-faces and block-edges. In case of a block-face grids are associated with maps operating on the unit square, while in case of a block-edge grids are associated with maps operating on the unit interval.

Averaging operation

Consider a block-face in which we have two different adapted grids associated with the grids in two adjacent blocks. Due to the equivalence of grid adaptation and maps, as described in the previous section, we can associate the two different grids with two different maps from the unit square onto itself, see the maps indicated as A_1 and A_2, respectively, in Fig.6. The key point of the grid averaging operation proposed here is to interpret the two maps in terms of four monitor functions and to employ these to drive an adaptation operation which again consists of a map from the unit square onto itself. Since each of the monitor functions consists of one of the two coordinates defined by one of the maps A_1 or A_2, the resolutions provided by both maps A_1 and A_2 will be represented by the averaged map A_{av}.

Two candidates are proposed here for the underlying adaptation procedure:

1. Laplace-Beltrami operator (see [1] and [2]),
2. Modified Anisotropic Diffusion (MAD) operator (see [3] and [4]).

The advantage of the Laplace-Beltrami operator is that when the maps A_1 and A_2 are identical the averaged map A_{av} will be also identical to A_1 and A_2, while the MAD operator does not satisfy this criterion. At the other hand, the advantage of the MAD operator is that it is guaranteed to produce a regular averaged map A_{av} [5], while such mathematical proof does not exist for the Laplace-Beltrami operator.

Continuity recovery operation

Consider a grid in a block-face with different grids on the edges as resulting from the averaging operation described above. To recover continuity between the grid in the face and the grids on the edges we start by employing the equivalence between grid adaptation/modification and maps again. In Fig.7 the face and averaged edge grids are depicted in the diagrams at the top, representing the parametric domains associated with the initial and block-by-block adapted grid respectively. To modify the face grid such that continuity with respect to the averaged edge grids is obtained we employ an extension of the compound map. The extension, consisting of the map C, see Fig.7, constitutes the continuity recovery operation.

Instead of modeling the map C directly we model it's inverse C^{-1}, see Fig.8. The advantage is that we can use information concerning the aspect ratio distribution of the block-by-block adapted grid in the physical domain without facing an implicit formulation. To illustrate this, let the map C be formally defined in terms of two functions $u(\xi, \eta)$ and $v(\xi, \eta)$, both mapping the unit square onto the unit interval. All information of the block-by-block adapted grid is available as functions of u and v, the coordinates in the parametric domain of the block-by-block adapted grid, see Fig.7. Hence, an explicit formulation for the map between $(u,v)^T$ and $(\xi,\eta)^T$ can be obtained by using u and v as the independent variables instead of ξ and η. The disadvantage that the obtained map $(\xi(u,v),\eta(u,v))^T$ must be inverted numerically is limited since the piecewise linear approach proposed in [4] is direct, robust and cost-effective.

The map C^{-1} is modeled by a modified Coons patch, see the Appendix. In contrast to a conventional Coons patch [6] where a function on the boundary of a unit square or cube is interpolated to the interior by means of linear interpolation functions, e.g. u and $1-u$, the modified Coons patch proposed here employs non-linear interpolation functions. For example, the function u is replaced by:

$$\{\int_0^u \phi(\overline{u},v,w)d\overline{u}\}/\{\int_0^1 \phi(\overline{u},v,w)d\overline{u}\}, \tag{6}$$

where $\phi(u,v,w)$ is an arbitrary positive function. Note that when $\phi(u,v,w)\equiv 1$ the conventional Coons patch is recovered. The objective of this modification is to make the interpolation controlable by the function ϕ. In case of the map C^{-1}, we take ϕ as:

$$\phi(u,v,w) = \{||x_u||/max(||x_v||,||x_w||)\}^2, \tag{7}$$

where $x \in I\!R^3$ is the physical coordinate. When the block-by-block adapted grid locally satisfies $||x_u||/max(||x_v||,||x_w||) << 1$ (small aspect ratio), the interpolation function Eq.(6) will locally be approximately independent of u which is a necessary condition to preserve orthogonality to some degree [3, 4].

As an example of the modeling of the map C^{-1} consider the grid in Fig.9 which could be the result of a typical block-by-block adaptation procedure showing high resolution areas along the circular boundaries and in the symmetry plane. In reality we have generated the grid as the image of a uniform grid in the unit square under a map M which is defined as follows.

$$M : [0,1]^2 \mapsto \Omega \subset I\!R^2, \quad M(p,q) = r(q)(cos(\theta(p)), sin(\theta(p)))^T, \tag{8}$$

with

$$\theta(p) = \left[\begin{array}{ll} \pi(\frac{1}{2} - 2^{m-1}|p-\frac{1}{2}|^m), & p < \frac{1}{2}, \\ \pi(\frac{1}{2} + 2^{m-1}|p-\frac{1}{2}|^m), & p \geq \frac{1}{2}, \end{array} \right. \tag{9}$$

and

$$r(q) = \left[\begin{array}{ll} 1 + 2^{m-1}q^m, & q < \frac{1}{2}, \\ 2 - 2^{m-1}(1-q)^m, & q \geq \frac{1}{2}. \end{array} \right. \tag{10}$$

In the present example we have taken $m = 5$. Let the averaging operation described in the previous section result in the grid point distribution on the inner circular boundary depicted in Fig.10, generated by the map $M(\hat{p}(p),0)$ with

$$\hat{p}(p) = p + c\,p(1-p), \quad c = 0.99. \tag{11}$$

Employing a conventional Coons patch to model C^{-1} results in the grid depicted in Fig.11 showing collapsed cells along the circular boundaries. In contrast, employing a modified Coons patch to model C^{-1} results in the grid depicted in Fig.12 showing regular cells everywhere. Note that in both cases the initial high resolution areas visible in Fig.9 are preserved.

Concluding remarks

The potential of the proposed algorithm to recover C^0-continuity of a block-by-block adapted multi-block grid is clearly demonstrated by an example in two-dimensions in the sense that C^0-continuity is recovered while the resolution and local orthogonality of the block-by-block adapted grid are approximately preserved. The two types of operations on which the algorithm is based, averaging of different grids and continuity recovery by modification of grids, can also be used in a multi-block environment with advanced topology identities such compound edges and faces [7].

References

[1] A.S. Dvinsky, *Adaptive Grid Generation from Harmonic Maps on Riemannian Manifolds*, J. Comput. Phys. 95, 450-476, 1991.

[2] S.P. Spekreijse, R. Hagmeijer and J.W. Boerstoel, *Adaptive Grid Generation by using the Laplace-Beltrami Operator on a Monitor Surface*, 5th International Conference on Numerical Grid Generation in Computational Fluid Dynamics and Related Fields, Starkville, Mississippi, April 1-5, 1996 (also NLR-TP-96297).

[3] R. Hagmeijer, *Grid Adaption Based on Modified Anisotropic Diffusion Equations Formulated in the Parametric Domain*, J. Comput. Phys. 115, 169-183, 1994 (also NLR-TP-94199).

[4] R. Hagmeijer, *Adaptation of Structured Grids Based on Weighted Least Squares Formulations* Ph.D. Thesis, Delft University of Technology, 1997 (also NLR-TP-97482).

[5] Ph. Clément, R. Hagmeijer and G. Sweers, *On the invertibility of mappings arising in 2D grid generation problems*, Numer. Math. 73: 37-51 (1996).

[6] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design, A Practical Guide*, Academic Press, 1992.

[7] S.P. Spekreijse and J.W. Boerstoel, *Multiblock Grid Generation, Part II: Multiblock aspects*, 27th Computational Fluid Dynamics Course, Von Karman Institute for Fluid Dynamics (VKI), Belgium, 25-29 March, 1996 (also NLR-TP-96338).

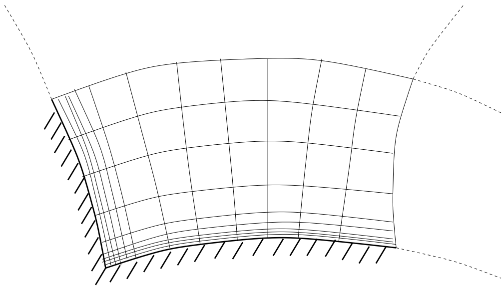

Figure 1: Single-block part of initial multi-block grid.

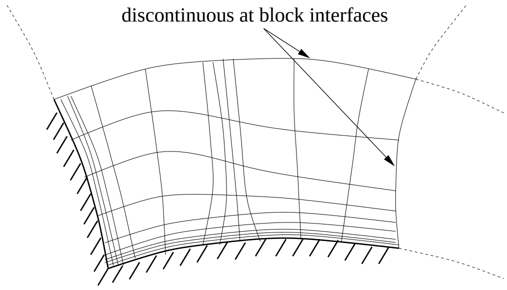

Figure 2: Single-block part of block-by-block adapted multi-block grid.

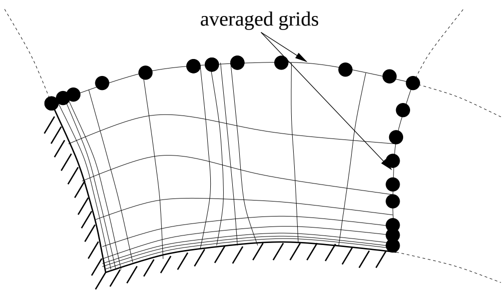

Figure 3: Averaged grids at block interfaces.

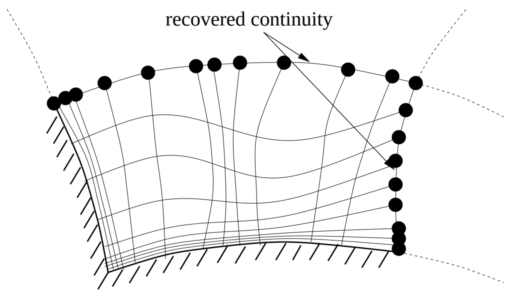

Figure 4: Recovered C^0-continuity.

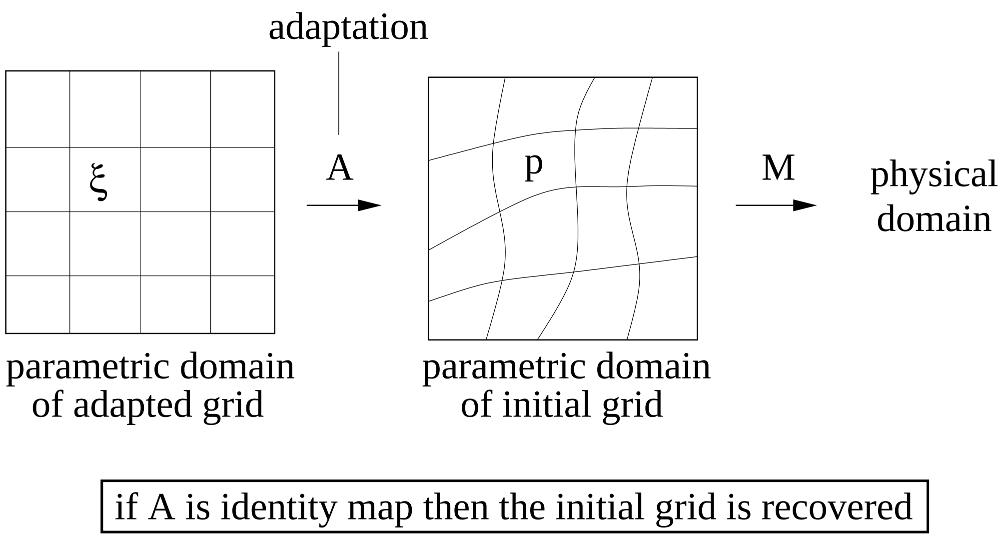

Figure 5: Adaptation represented by compound map.

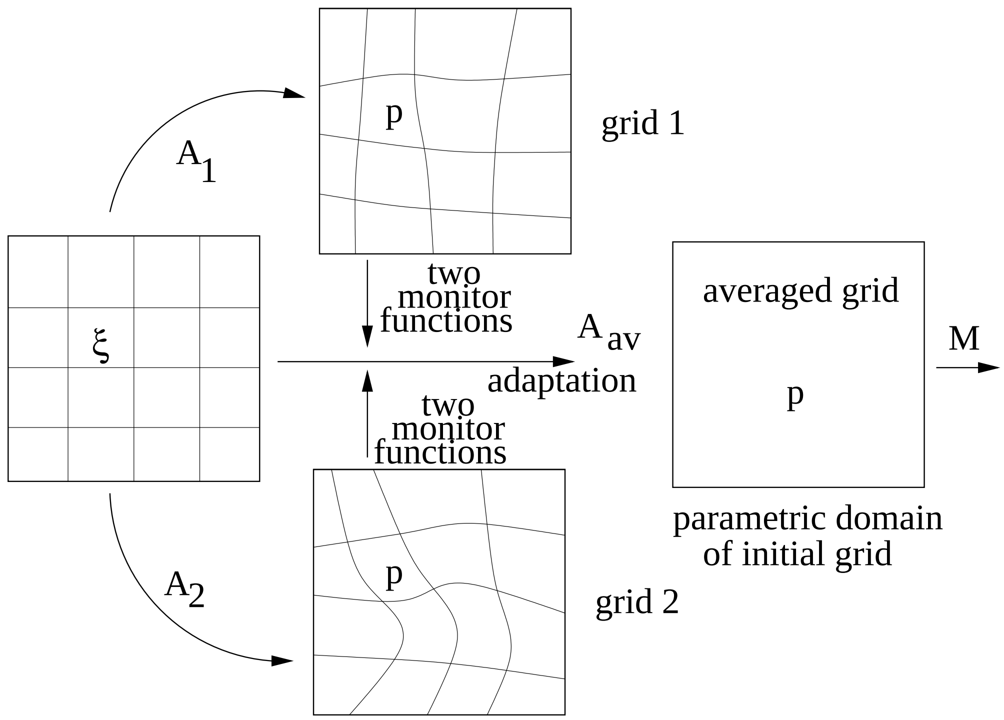

Figure 6: Averaging of two grids at block-face.

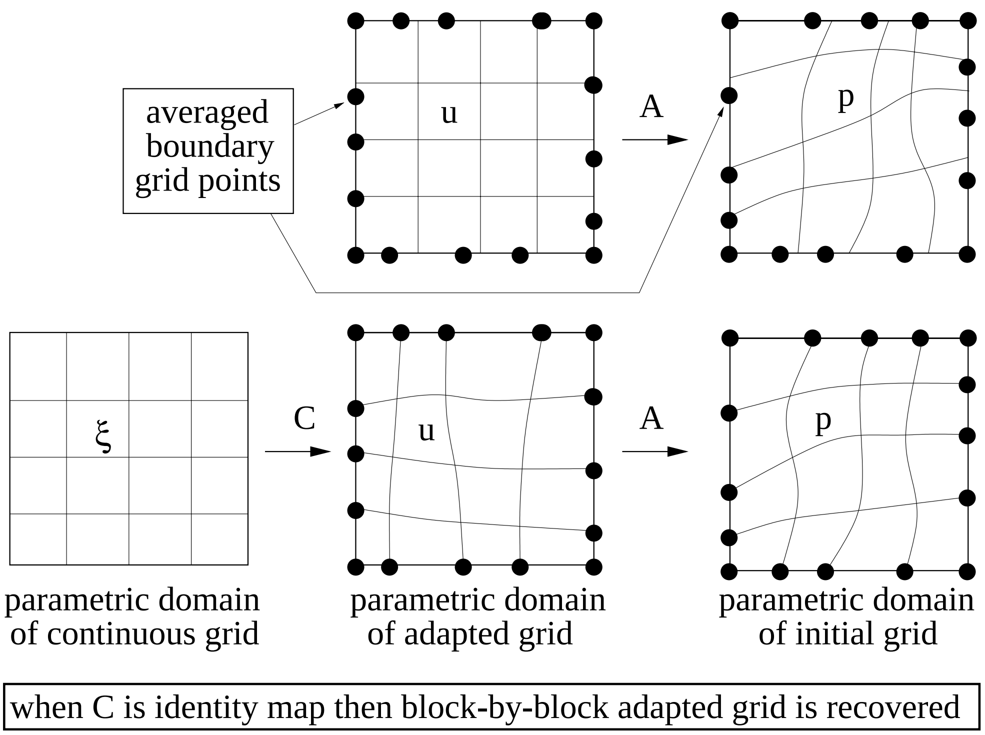

Figure 7: Continuity recovery represented by extended compound map.

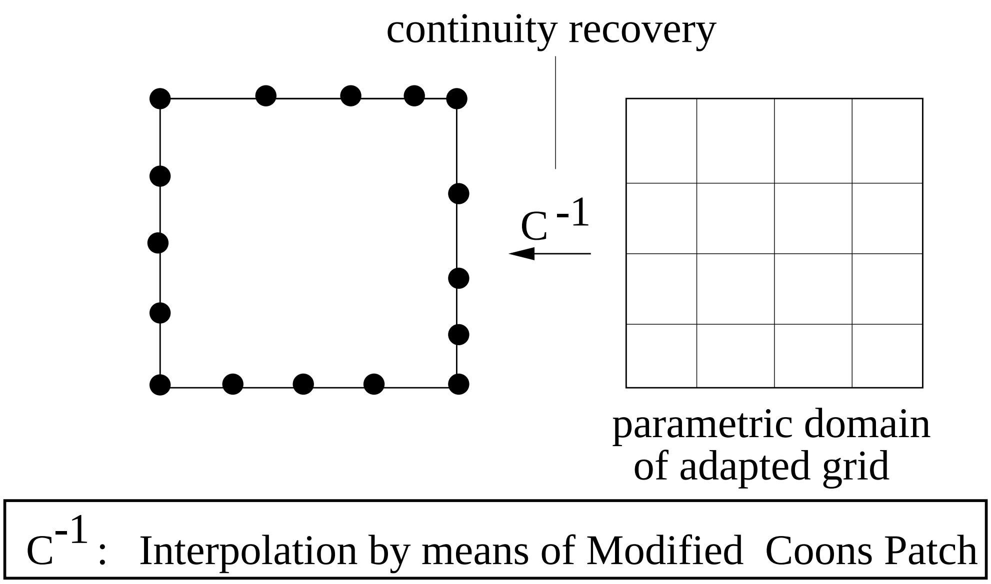

Figure 8: Advanced interpolation in parametric domain.

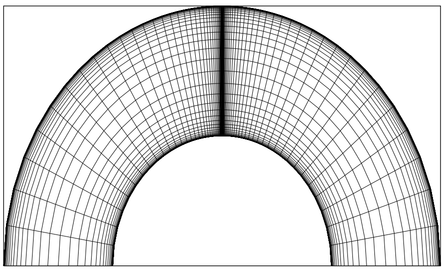

Figure 9: Adapted single-block grid around cylinder.

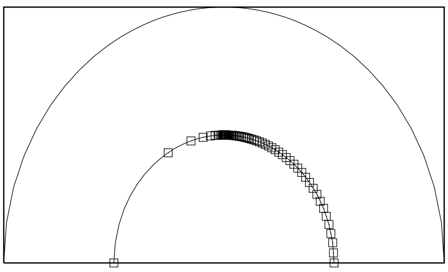

Figure 10: Requested grid on inner circular boundary.

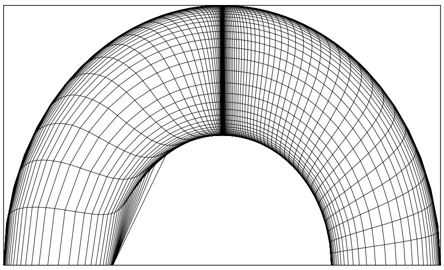

Figure 11: Recovered continuity using conventional Coons patch.

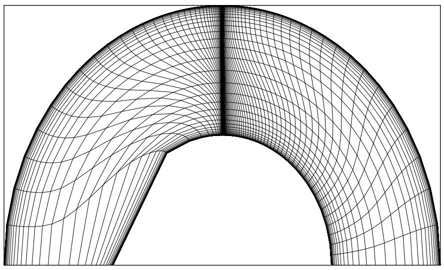

Figure 12: Recovered continuity using modified Coons patch.

Appendix: Modified Coons patch in three dimensions

Given $b(u,v,w) \in C^0 : \partial([0,1]^3) \mapsto \partial\Omega_b \subset I\!R^3$, and $x(u,v,w) \in C^1 : [0,1]^3 \mapsto \overline{\Omega}_B \subset I\!R^3$, a modified Coons patch in three dimensions is defined as:

$$
\begin{aligned}
& T_{3D}(u,v,w,b(u,v,w),x(u,v,w)) = \\
& \quad \sum_{i=0}^{1}\alpha_i(u,v,w)b(i,v,w) + \sum_{j=0}^{1}\beta_j(u,v,w)b(u,j,w) + \sum_{k=0}^{1}\gamma_k(u,v,w)b(u,v,k) \\
& - \sum_{i=0}^{1}\sum_{j=0}^{1}\alpha_i(u,v,w)\beta_j(u,v,w)b(i,j,w) \\
& - \sum_{i=0}^{1}\sum_{k=0}^{1}\alpha_i(u,v,w)\gamma_k(u,v,w)b(i,v,k) \\
& - \sum_{j=0}^{1}\sum_{k=0}^{1}\beta_j(u,v,w)\gamma_k(u,v,w)b(u,j,k) \\
& + \sum_{i=0}^{1}\sum_{j=0}^{1}\sum_{k=0}^{1}\alpha_i(u,v,w)\beta_j(u,v,w)\gamma_k(u,v,w)b(i,j,k),
\end{aligned}
$$

with

$$
\begin{aligned}
\alpha_1(u,v,w) &= 1-\alpha_0(u,v,w), \\
\alpha_0(u,v,w) &= 1-\{\int_0^u \phi(\overline{u},v,w)d\overline{u}\}/\{\int_0^1 \phi(\overline{u},v,w)d\overline{u}\}, \\
\beta_1(u,v,w) &= 1-\beta_0(u,v,w), \\
\beta_0(u,v,w) &= 1-\{\int_0^v \psi(u,\overline{v},w)d\overline{v}\}/\{\int_0^1 \psi(u,\overline{v},w)d\overline{v}\}, \\
\gamma_1(u,v,w) &= 1-\gamma_0(u,v,w), \\
\gamma_0(u,v,w) &= 1-\{\int_0^w \chi(u,v,\overline{w})d\overline{w}\}/\{\int_0^1 \chi(u,v,\overline{w})d\overline{w}\},
\end{aligned}
$$

$$
\begin{aligned}
\phi(u,v,w) &= \{\|x_u\|/max(\|x_v\|,\|x_w\|)\}^2, \\
\psi(u,v,w) &= \{\|x_v\|/max(\|x_u\|,\|x_w\|)\}^2, \\
\chi(u,v,w) &= \{\|x_w\|/max(\|x_u\|,\|x_v\|)\}^2.
\end{aligned}
$$