UPPSALA
UNIVERSITET

UPTEC F14040

Examensarbete 30 hp
September 2014

Summation By Part Methods for Poisson's Equation with Discontinuous Variable Coefficients

Thomas Nystrand

UPPSALA
UNIVERSITET

Teknisk- naturvetenskaplig fakultet
UTH-enheten

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
http://www.teknat.uu.se/student

Abstract

Summation By Parts Methods for the Poisson's Equation with Discontinuous Variable Coefficients

Thomas Nystrand

Nowadays there is an ever increasing demand to obtain more accurate numerical simulation results while at the same time using fewer computations. One area with such a demand is oil reservoir simulations, which builds upon Poisson's equation with variable coefficients (PEWVC). This thesis focuses on applying and testing a high order numerical scheme to solve the PEWVC, namely Summation By Parts - Simultaneous Approximation Term (SBP-SAT). The thesis opens with proving that the method is convergent at arbitrary high orders given sufficiently smooth coefficients. The convergence is furthermore verified in practice by test cases on the Poisson's equation with smoothly variable permeability coefficients. To balance observed lower boundary flux convergence, the SBP-SAT method was modified with additional penalty terms that were subsequently shown to work as expected. Finally the SBP-SAT method was tested on a semi-realistic model of an oil reservoir with discontinuous permeability. The correctness of the resulting pressure distribution varied and it was shown that flux leakage was the probable cause. Hence the proposed SBP-SAT method performs, as expected, very well in continuous settings but typically allows undesirable leakage in discontinuous settings. There are possible fixes, but these are outside the scope of this thesis.

Handledare: Gunilla Kreiss, Margot Gerritsen
Ämnesgranskare: Ken Mattsson
Examinator: Thomas Nyberg
ISSN: 1401-5757, UPTEC F14040

Sammanfattning

I modern tid har det blivit allt viktigare att kunna lösa stora ekvationsystem som beskrivs av partiella differential ekvationer. Detta gäller allt från ljud-utbredning, tusnamis, byggnader till djurarters populations-bestånd. En typ av partiella differential ekvationer som är särskilt efterfrågat och notoriskt svårt att lösa är Poisson's ekvation med diskontinuerliga koefficienter. Detta hanterats normalt med finita volym element metoder, men här testas en annan beprövad metod, nämligen Summation By Parts - Simulatuous Approximation Term (SBPSAT) som framgångsrikt har applicerats i många situationer som till exempel vågutbredning. Först visar vi matematiskt att metoden konvergerar mot den faktiska lösningen i analytiska fall och därefter appliserar vi metoden på flera tester med analytiska lösningar. Dessutom visar vi att metoden kan modiferas så att den konvergerar med extra hög noggrannhet på randen. Avslutningsvis använder vi metoden på ett verkligt problem som beskrivs av Poisson's ekvation med diskontinuerliga koefficienter, nämligen reservoar tryck-fördelningar. Resultatet av simuleringarna är delvis framgångsrika. För modeller med tillräckligt mjuka koefficienter ger SBP-SAT mycket noggranna lösningar. Dock om koefficienterna är diskontinuerliga kan vissa koefficient-kombinationer orsaka problem. Typiskt gäller detta diagonaler av höga eller låga koefficienter motsvarande porösa medier eller granit som löper genom domänen i 2D eller högre dimensioner. Dessa kan till exempel ge upphov till att lösningen ger flöde över områden där det inte bör vara något flöde, ett så kallat läckage.

"Insanity: doing the same thing over and over again and expecting different results."

Albert Einstein

Acknowledgements

I would like to start of by thanking my adviser and supervisor Gunilla Kreiss for her continuous support and patience. You gave me free hands but always had time to help whenever I needed. Your knowledge and help has been invaluable.

I would also like extend my gratitude to my other adviser, Margot Gerritsen. You showed me nothing but kindness and support. You were always ready, despite a busy schedule, to advice and help both on and off thesis matters.

Finally I would like to thank my subject reviewer, Ken Mattsson. You always take interest and pride in your work. No matter what question you explain throughly and in great detail. Your enthusiasm is inspiring and you were ready to offer guidance whenever I asked. Thank you for everything.

To all of the people who have helped me, including my advisers and subject reviewer, I greatly appreciate and admire the strong academic interest and passion you have taken in my thesis. I have been given an opportunity to work with the best. Thank you.

Contents

List of Figures

List of Tables

Abbreviations

SBP	Summation By Parts
SAT	Simultaneous Approximation Term
FEM	Finite Element Method
FVM	Finite Volume Method
MRST	MATLAB Reservoir Simulation Toolbox
SPE	Society of Petroleum Engineers
RHS	Right Hand Side
LHS	Left Hand Side
BFX	Boundary Flux term

The SBP method

As a short introduction to the SBP operators, which is a special version of finite differences, a few definitions are presented as in [1]. The numerical structure of the SBP operators is given in Appendix B

Definition. An explicit p^{th}-order accurate finite difference scheme with minimal stencil width of a Cauchy problem is called a p^{th}-order accurate narrow-stencil.

Definition. A difference operator $D_1 = H^{-1}Q$ approximating $\partial/\partial x$ on a bounded interval, using a pth-order accurate narrow-stencil at all points except a few near the boundaries, is said to be a pth-order accurate narrow-diagonal first-derivate SBP operator if H is diagonal and positive definite and $Q + Q^T = diag(-1, 0, \ldots, 0, 1)$.

Definition. Let $D_2^{(k)} = H^{-1}(-M^{(k)} + \bar{K}S)$ approximate $\partial/\partial x(k\partial/\partial x)$, where $k(x) > 0$, using a pth-order accurate narrow-stencil at all points except a few near the boundaries. $D_2^{(k)}$ is said to be a pth-order accurate narrow-diagonal second-derivative SBP operator, if H is diagonal and positive definite, $M^{(k)}$ is symmetric and positive definite, S approximates the first-derivative operator at the boundaries and $\bar{K} = diag(-k_1, 0 \ldots 0, k_n)$. Furthermore $M^{(k)} = D_1^T HKD_1 + R^{(k)}$ [2] where R is positive semidefinite and $K = diag(-k_1, k_2 \ldots k_{n-1}, k_n)$.

Chapter 1

Introduction and Background

The purpose of this thesis is to evaluate the performance of applying the Summation By Parts (SBP) method with weakly imposed boundary conditions through Simultaneous Approximation Terms (SAT) to the Poisson's equation with variable coefficients.

1.1 Purpose and Background

The Poisson's equation with variable coefficients (PEWVC) is the base for modeling various diffusion processes in nature such as steady state ground water movement, heat flow and oil reservoir flow. Both the ground water and oil reservoir use at its core the Darcy equation, where permeability is represented by the coefficients. Typically the Darcy equation is thereafter extended to include various phenomena to ensure a more accurate model. Since the material in ground water or reservoir oil flow generally vary heavily, the permeability coefficients tend to be discontinuous, which leads to a discontinuous domain flow.

To simulate these natural phenomenons with the help of computers there are many different schemes, such as Finite Element Methods (FEM), Finite Volume Methods (FVM) and Finite Differences (FD). The methods differ in how they model the problem which means that in practice they will differ in convergence order, applicable grid-types and complexity. However, they all have the same basic requirement; each method must converge toward the true solution. Not all schemes fulfill the last condition when applied to all natural phenomenons. This is the case for simulating the Darcy equation for modeling underground fluid flow which poses a particularly challenging problem due to its unusual large discontinuities. Nonetheless, there exist methods that converge toward the true solution for many problems, but only a few of these converge at higher orders.

Traditionally Finite Elements Methods, Finite Volume Methods and Finite Differences have been used to simulate the Poisson's equation with variable coefficients (see for instance [3] and [4]). FEM and FVM are in particular useful in an oil reservoir setting since the methods are simple, robust, efficient and can be applied on unstructured grids. Some versions have also been proved to be convergent despite large discontinuities, which is crucial for reservoir modeling. While successful, there is still room for improvements; a standard FVM typically gives first order flux convergence on the boundary. A high boundary flux convergence is highly desirable in many applications including oil reservoir modeling. The higher convergence could produce a more precise outflow and inflow for coarser grids which in turn means better estimates of the total oil flow and less spill.

The SBP-SAT method is one of the schemes which introduces high order convergence on Cartesian grids. SBP operators were introduced by Kreiss & Scherer 1974 and the full SBP-SAT method was introduced by Carpenter, M. H., Gottlieb, D. and Abarbanel, S. in 1994 [5] and has since been developed and used on a variety of problems. At its core the SBP-SAT is a modification of finite differences with special boundary treatment through penalty terms. The construction of the equation system guarantees that the discrete solution converges with a certain order toward the true solution.

Since the SBP-SAT method can in theory be used to construct arbitrary high convergence operators, the method has typically been very useful when high accuracy or computational efficiency is required. The speed is a consequence of using a higher order operator since one can design much coarser grids and aggressively reduce computational efforts while maintaining small errors. However SBP-SAT has been overlooked traditionally in many cases for two reasons. Firstly, SBP-SAT methods have in essence been restricted to Cartesian and curvilinear grids and secondly, many problems does not fulfill the necessary prerequisites to guarantee high order convergence. This first issue is not likely a huge problem anymore; with the advent of recent boundary treatment research, it is possible that special boundary operators which have the correct convergence order can be constructed (Such operators will not be used in this thesis). The second problem is harder. A typical example of not fulfilling prerequisites is if the posed problem is not sufficiently smooth. Without smoothness, the SBP-SAT is restricted to lower orders and its advantage is lost. Therefore it has been hard to justify using it on non-smooth problems over traditional methods such as FEM which can produce correct convergence on unstructured grids.

Since its introduction, SBP-SAT has nonetheless been successfully applied to a broad range of problems such as fluid dynamics and quantum physics. But it has not been tested on the Poisson's equation with discontinuous variable coefficients (to the authors knowledge). This thesis introduces SBP-SAT as an alternative solver for the Poisson's equation with variable coefficients and evaluates the SBP-SAT method's performance. The performance is compared to that of a well tested FVM solver. The thesis also tests if modifying the SBP-SAT method can guarantee better convergence on the boundary.

1.2 Research Ethics

In this thesis I have taken steps to ensure that the proposed SBP-SAT method is properly tested without bias. The reference solutions is produced using MRST [6]. This is a well-known open source oil reservoir simulator developed by SINTEF. It has been throughly tested on among others SPE Comparative Solution Project data [7]. Furthermore, the data used for this thesis is also drawn from the SPE Comparative Solution Project [7]. This data has been used by numerous large scale solvers since the 1990s. The data is obtained from real world measurements and aims to provide a common ground for algorithms and simulators to compare their performance. Another ethical aspect is that the thesis is centered around oil reservoir simulations which could raise potential environmental questions. This thesis maintains a strictly objective attitude toward the environmental aspect and seeks only to provide better tools for simulating pressure distributions in the ground. The simulation methods proposed in this thesis can be used to simulate ground water movement and heat flow just as well as oil reservoirs.

1.3 Thesis Structure

The thesis opens with a chapter that introduces a continuous model using the Poisson's equation with variable coefficients. By showing that a small change in indata produces a small change in outdata, a continuous inequality is obtained which is needed for proving SBP-SAT convergence. The following chapter, Chapter 3, seekes to mimic this inequality in discrete space when applying the SBP-SAT method on the PEWVC and subsequently prove convergence. To ensure that the discrete solution always converges to the one true solution, it is shown that the equation system is positive definite. The chapter concludes with the theory needed for improved boundary derivate convergence. Chapter 4 presents a test problem which is necessary to verify the correctness of the previous theory. The boundary convergence rate of this test is thereafter improved by using the earlier obtained boundary theory. Chapter 5 presents the results of applying the SBP-SAT method in more realistic environments with discontinuous media. Various targeted tests are performed to highlight potential weaknesses. The last chapter of numerical tests brings the simulations to full scale by using real world data. Finally the thesis is concluded with a discussion and a final remark on future use.

Chapter 2

Poisson's Equation - Continuous Theory

To apply the SBP-SAT method on the Poisson's equation and guarantee correct pressure simulation results of reservoirs it is necessary to prove convergence before running any simulations. This chapter, after introducing the problem examines the first step in proving convergence for the Poisson's equation.

2.1 Problem Formulation

The core of oil reservoir models are Darcy's law which translates into a time-independent elliptical partial differential equation on the following form

$$\begin{cases} -\boldsymbol{\nabla} \cdot (K\boldsymbol{\nabla} p) = f & x \in \Omega \\ n \cdot \boldsymbol{\nabla} p = g_n & x \in \Gamma_n \\ p = g_d & x \in \Gamma_d \end{cases} \tag{2.1}$$

This is commonly known as the Poisson's equation with variable coefficients. The RHS function f is a source term and K is the permeability matrix, which is assumed to be symmetrical and positive definite for all $x \in \Omega$ (for this thesis it is also, unless stated explicitly, considered diagonal). The K coefficients need not to be continuous although for the purpose of showing convergence, they will be assumed to be continuous. Function p is the final pressure variation. Γ_n represents the part of boundary that has normal conditions while Γ_d has Dirichlet conditions. Mixtures of the boundary types are also possible (and often favorable), but only discussed shortly during stability analysis. In equation (2.1), Γ_n and Γ_d together make up the entire boundary and do not overlap.

To show convergence of the SBP-SAT method we begin by noting that the Poisson's equation with positive definite variable coefficients is wellposed (see for instance [8]). For proofs of existence and uniqueness the reader is referred to literature. We do however show that the solution depends continuously on indata since it leads to an inequality that is needed to complete the convergence proof. A similar inequality is required in discrete space as well which is deferred to Chapter 3. This chapter continues with presenting the problem in one dimension and subsequently obtaining the continuous inequality.

2.2 The 1D Problem

The Poisson's equation (2.1) in the continuous space can be rewritten in 1D as

$$\begin{cases} -(k(x)u_x)_x = f, \quad k(x) > 0 \\ u = g_l, \quad x = 0 \\ u = g_r, \quad x = 1 \end{cases} \tag{2.2}$$

where, for simplicity the boundary conditions are specified as Dirichlet conditions in the domain $0 \leq x \leq 1$ and $k, f, u \in \Re$.

It is commonly known that by a change of variables the equation system 2.2 can always be transformed into the equivalent system;

$$\begin{cases} -(k(x)v_x)_x = \widetilde{f}, \quad k(x) > 0 \\ v = 0, \quad x = 0 \\ v = 0, \quad x = 1 \end{cases} \tag{2.3}$$

which has homogeneous boundary conditions and $k, \widetilde{f}, v \in \Re$. Henceforth homogeneous boundary conditions will be assumed in Chapter 2 and 3.

2.2.1 Inequalities and Notation

To simplify the notation the inner product of two functions $u, v \in L^2[0,1]$ is defined as:

$$(u,v) = \int_0^1 u^\dagger v \, \mathrm{d}x,$$

and the L_2 norm becomes,

$$\|u\|_k^2 = (u, ku), \;\; k > 0.$$

If $\| \cdot \|$ is used in a continuous context this is the same as $\| \cdot \|_{L^2(\Omega)}$ (standard L_2 norm).

We also present some inequalities that are needed in the continuous proof:

Cauchy-Schwarz: A continuous version of the Cauchy-Schwarz inequality is given by

$$(u, w) \leq \|u\| \|w\| \tag{2.4}$$

where u, w are real valued functions which are continuous in the closed interval $[0, 1]$

Poincaré: A continuous version of the Poincaré inequality is given by

$$\|u\|_{L^2(\Omega)} \leq C \|\boldsymbol{\nabla} u\|_{L^2(\Omega)} \tag{2.5}$$

Where $C > 0$ is a real valued constant, which depends only on Ω, a bounded connected open domain with Lipschitz boundary, and u must be zero on part of the boundary.

Next we present the theorem and proof for the continuous inequality needed for convergence.

2.2.2 Continuous Inequality

Theorem 2.2.1. *For a continuous solution $u(x) \in \Re$, $x \in [0, 1]$ to the 1D Poisson's equation (2.1) with homogeneous boundary conditions of which at least one is a Dirichlet condition there exists a constant $C \in \Re > 0$ such that*

$$\|u_x\|_{L^2(\Omega)} \leq C \|f\|_{L^2(\Omega)} \tag{2.6}$$

Proof. To prove the theorem, we being with multiplying equation (2.1) with $u^\dagger$ and integrate over the domain,

$$-(u,(k(x)u_x)_x) = (u,f). \tag{2.7}$$

Integrate by parts, using the Cauchy-Schwarz inequality (2.4) and using the homogeneous boundary conditions,

$$\begin{aligned} (u_x, k(x)u_x) - u^\dagger k(x)u_x|_0^1 = (u,f) \quad &\Rightarrow \\ \|u_x\|_k^2 \le \|u\|\|f\| \quad &\Rightarrow \\ \left[\inf_x k(x)\right]^2 \|u_x\|_{L^2(\Omega)}^2 \le \|u_x\|_k^2 \le \|u\|_{L^2(\Omega)}\|f\|_{L^2(\Omega)}, & \end{aligned} \tag{2.8}$$

we obtain

$$\|u_x\|_{L^2(\Omega)}^2 \le \frac{1}{\left[\inf_x k(x)\right]^2}\|u\|_{L^2(\Omega)}\|f\|_{L^2(\Omega)}. \tag{2.9}$$

Note that in the above reasoning, $\|\cdot\|_k^2$ is a norm since $0 < k_0 < k(x) < k_N < \infty$. Applying the Poincaré inequality 2.5 (since there is at least one homogeneous Dirichlet boundary condition) and dividing both sides with $\|u_x\|_{L^2(\Omega)}$ one finally obtains theorem 2.2.1

$$\|u_x\|_{L^2(\Omega)}^2 \le \frac{C_p}{\left[\inf_x k(x)\right]^2}\|u_x\|_{L^2(\Omega)}\|f\|_{L^2(\Omega)} \Rightarrow \tag{2.10}$$

$$\|u_x\|_{L^2(\Omega)} \le \frac{C_p}{\left[\inf_x k(x)\right]^2}\|f\|_{L^2(\Omega)}. \tag{2.11}$$

Setting $C = \frac{C_p}{\left[\inf_x k(x)\right]^2}$ completes the proof. □

2.3 The 2D Problem

The Poisson's equation (2.1) in 2D in a square domain can be written as

$$\begin{cases} -\boldsymbol{\nabla} \cdot K \boldsymbol{\nabla} p = f \\ \qquad p = g_E, \quad x = 1 \\ \qquad p = g_W, \quad x = -1 \\ \qquad p = g_N, \quad y = 1 \\ \qquad p = g_S, \quad y = -1 \end{cases} \tag{2.12}$$

where the boundary has been specified as Dirichlet conditions. The subscripts E, W, N and S refer to East, West, North and South boundary. The K is a 2-by-2 matrix defined by

$$K = \begin{bmatrix} k_{11}(x,y) & k_{12}(x,y) \\ k_{21}(x,y) & k_{22}(x,y) \end{bmatrix},$$

where $k_{12}(x,y) = k_{21}(x,y)$. Apart from symmetrical, K is assumed to be positive definite. For all simulations in this thesis, the K matrix is also diagonal. The full matrix is however kept in all proofs which means convergence applies for both diagonal and non-diagonal problems.

The continuous inequality in 2D space is defined analogously to the 1D space,

Theorem 2.3.1. *For a continuous solution $u(x,y) \in \Re$, $x \in [0,1], y \in [0,1]$ to the 2D Poisson's equation (2.12) with homogeneous boundary conditions of which at least one is a Dirichlet condition there exists a constant $C \in \Re > 0$ such that*

$$\|\boldsymbol{\nabla} u\|_{L^2(\Omega)} \leq C\|f\|_{L^2(\Omega)} \tag{2.13}$$

To show this (and thereafter convergence) in 2D it is straight forward to follow the 1D outline with only minor tweaking. The proofs are however cumbersome and the interested reader is referred to Appendix A.

Chapter 3

Poisson's Equation - Discrete Theory

In this chapter convergence of the SBP-SAT method is proven in 1D space (The details of the 2D proof resides in Appendix A). We show that there exists an inequality that mimics the continuous inequality from the Chapter 2. Using these two inequalities it is hence shown that a solution to the discrete method will converge to the analytical solution. By showing that the discrete equation system has exactly one solution it is furthermore concluded that the discrete system will converge to the true solution. The chapter rounds up with the theory needed to improve the boundary derivate convergence.

3.1 The 1D problem

To solve the equation system (2.1) using the SBP-SAT method (for definition see the introductory section The SBP method), narrow banded SBP operators are applied on a discretized domain using a set of equidistant points:

$$x_i = (i-1)h, \quad i = 1, 2, \dots, N, \quad h = \tfrac{1}{N-1} \tag{3.1}$$

The approximate solution at grid point x_i is denoted v_i and the discrete solution vector is $v = \Big[v_1, v_2, \dots, v_N\Big]^T$.

After applying the SBP-SAT method the equation system (2.1) is transformed into a system of equations on the following form,

$$-D_2^{(k)}v + SAT = f_v. \tag{3.2}$$

Where the $D_2^{(k)}$ is a second derivate SBP operator (see Appendix B) and f_v is the value of f - the analytical RHS - evaluated at each discrete domain point x_i. The SAT terms implement the boundary conditions by enforcing them weakly and v is the solution vector. Next we introduce some symbols that are needed for the discrete inequality proof, convergence and improved boundary.

3.1.1 Symbols and Definitions

A norm in the discrete case is written as:

$$(u, v)_H = u^T Hv, \ \|v\|_H^2 = v^T Hv,$$

where H is a positive definite diagonal matrix and $v, u \in \mathbb{R}^N$.

Some recurring symbols:

$$e_1 = \Big[1, 0, \dots, 0\Big]^T, \ e_N = \Big[0, \dots, 0, 1\Big]^T.$$

$$S_1 = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 \end{pmatrix} S, \ S_N = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \ddots & \vdots & \vdots \\ \vdots & \dots & 0 & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix} S,$$

where S is the boundary derivate matrix which is presented in Appendix B. $(KS)_i$ is the i^{th} row of the matrix product of K and S. $(KS)_i^T$ is interpreted as taking the transpose of the i^{th} row.

$$I_N = \underbrace{\left.\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \ddots & \vdots & \vdots \\ 0 & \dots & 1 & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix}\right\} Nrows}_{Ncolumns}$$

For the K coefficients we define k_i as the element located at position (i, i) in the K coefficient matrix and

$$\begin{aligned} \widetilde{k_L^p} &= \min\{k_1, k_2, ..., k_p\} \\ \widetilde{k_R^p} &= \min\{k_{n-p+1}, k_{n-p+2}, ..., k_n\}. \end{aligned} \tag{3.3}$$

Finally we introduce some discrete inequalities needed for the convergence proof.

Cauchy-Schwarz: A discrete version of Cauchy-Schwarz is given by

$$u^T w \leq \|u\|\|w\|, \tag{3.4}$$

where u, w are arbitrary vectors in an inner product vector space.

Poincaré: To complete the convergence proof a discrete version of the Poincaré inequality (2.5) is also required. A proof of this inequality was not obtained but numerical tests were conducted that indicate the discrete inequality's existence. Hence Matlab was used to determine if there exists a positive, real constant C such that all the eigenvalues to the matrix $M = CD_1^T H D_1 - H$ are positive. Here the matrices D_1 and H are defined in Appendix B. The first and/or last rows and columns are removed from M, which is the same as letting v vanish on the boundary. Such a C, independent of matrix size, was indeed found. The minimum allowed value of C has a slight dependancy on order of accuracy of D_1 and H, but for up to and including sixth order accuracy it could be set as $C = 0.3$. Showing that M has only positive eigenvalues is equivalent to setting $v^T(CD_1^T H D_1 - H)v \geq 0$. Hence we hypothesize that all gridpoints on a domain $[0, 1]$, which vanish at the endpoints, satisfy

$$\|v\|_H \leq C\|D_1 v\|_H, \tag{3.5}$$

where D_1 is a second or higher order SBP first derivate operator, v is a vector which vanishes on the boundary, C is a positive constant and H is the positive diagonal definite matrix defined in AppendixB.

3.2 Discrete inequality

Obtaining the discrete inequality corresponding to the continuous inequality (2.5) is slightly different depending on the boundary conditions for the SBP-SAT method. There are normally three different boundary types to consider: injected, which is strong Dirichlet, weak Dirichlet and weak Neumann conditions (or mixes of these).

Injected Boundary:

$$-\tilde{D}_2^{(k)}\tilde{v} = \tilde{f}_v + [\tilde{D}_2^{(k)}]_1 v_1 + [\tilde{D}_2^{(k)}]_N v_N \tag{3.6}$$

Neumann Boundary SAT:

$$-D_2^{(k)}v + \tau_0 H^{-1} e_1((KS)_1 v - g_l) + \tau_1 H^{-1} e_N((KS)_N v - g_r) = f_v \tag{3.7}$$

Weak Dirichlet Boundary SAT:

$$\begin{gathered} -D_2^{(k)}v + \tau_0 H^{-1}(KS)_1^T(e_1^T v - g_l) + \tau_1 H^{-1}(KS)_N^T(e_N^T v - g_r) + \\ \sigma_0 K e_1(e_1^T v - g_l) + \sigma_1 K e_N(e_N^T v - g_r) = f_v \end{gathered} \tag{3.8}$$

A few comments on the different boundary cases:

- Injection is a Dirichlet condition, which enforces the boundary strongly instead of weakly through a SAT term. This thesis is mainly concerned with the weak Dirichlet formulation since it has an advantage in that it can be used for interfaces, but injection is mentioned for later reference.

- The injected formulation uses a tilde to signify that the vectors and matrices in equation (3.2) have had the first and last rows removed. For the $D_2^{(k)}$, in addition to removing the rows, the first and last columns have been multiplied with v_1 and v_N respectively and moved to the RHS of the equation. The point values v_1 and v_N are the given Dirichlet boundary. The n^{th} column is represented by $[\cdot]_n$.

Below we present the proof for a discrete inequality that mimics the continuous inequality (2.6) for a system with Dirichlet boundary. To complete the proof for theorem 3.2.1 we had to utilize the discrete Poincaré (3.5), which requires at least one perfectly homogeneous Dirichlet boundary, which can only be guaranteed with an injected boundary. However, numerical tests with only weak Dirichlet boundary suggest that the weak Dirichlet conditions converge and we discuss the reason briefly in the proof. The proof can also be completed with for instance with Neumann and one strong Dirichlet condition.

Theorem 3.2.1. *The SBP-SAT approximation of the Poisson's equation* (2.2) *with one weak and one strong homogeneous Dirichlet boundary condition,*

$$\begin{gathered} -\tilde{D}_2^{(k)}\tilde{v} + \tau_1 \tilde{H}^{-1}(\tilde{KS})_N^T(\tilde{e}_N^T\tilde{v} - g_r) + \\ \sigma_1 \tilde{K}\tilde{e}_N(\tilde{e}_N^T\tilde{v} - g_r) = \tilde{f}_v + [\tilde{D}_2^{(k)}]_1 v_1, \end{gathered} \tag{3.9}$$

which satisfies the discrete Poincaré inequality (3.5) *will also satisfy*

$$\|\tilde{D}_1\tilde{v}\|_H \leq C\|\tilde{f}_v\|_H. \tag{3.10}$$

Here $\tau_1 = -1, \quad \sigma_1 \geq \frac{k_N \widetilde{k_R^p}}{\alpha h}$, α *is a positive constant dependent on the SBP order and the tilde signifies that the first row (and column of matrix) has been removed. The various operators are defined in appendix B. The RHS injected term,* $[\tilde{D}_2^{(k)}]_1 v_1$, *is zero since* $v_1 = 0$ *(but kept for clarity).*

Proof. To prove that equation (3.9) fulfills the inequality (3.10) we start by applying the homogeneous boundary conditions, which gives

$$\begin{gathered} -D_2^{(k)}v + \tau_1 H^{-1}(KS)_N^T e_N^T v + \\ \sigma_1 K e_N e_N^T v = f_v, \end{gathered} \tag{3.11}$$

where we for simplicity dropped the injected tilde signifier, but its understood that the first row and column has been removed henceforth. Multiplying by $v^T H$ we obtain

$$\begin{aligned} -v^T(-M^{(k)} + \bar{K}S)v + \tau_1 v^T (KS)_N^T e_N^T v + \\ \sigma_1 h v^T K e_N e_N^T v = v^T H f_v. \end{aligned} \tag{3.12}$$

Setting $\tau_1 = -1$ (using that $\bar{K} = diag(0, \ldots 0, k_n)$ since the first row and column is removed) equation (3.12) further simplifies to

$$\begin{aligned} v^T M^{(k)} v - v^T(e_N^T (KS)_N + (KS)_N^T e_N)v + \\ h\sigma_1 v^T K e_N e_N^T v = v^T H f_v. \end{aligned} \tag{3.13}$$

The next step is to ensure positive (semi-)definiteness of the LHS of equation (3.13) while rewriting it as LHS $= \|D_1 v\|_H + Terms$, where the $Terms$ must be positive semi-definite. To show that the $Terms \geq 0$, we treat each boundary condition separately and thereafter combine the injected and weak Dirichlet.

For injected and Neumann we start by splitting $M^{(k)}$ as (see for instance [9])

$$M^{(k)} = D_1^T K H D_1 + R^{(k)}.$$

where $R^{(k)}$ is positive semi definite. This means that for both Neumann and injected boundary conditions, the LHS can simply be rewritten as

$$\text{LHS} = v^T M^{(k)} v = v^T (D_1^T K H D_1 + R^{(k)}) v \geq v^T D_1^T K H D_1 v.$$

where the difference between the Neumann and injected condition is that the boundary column and rows are removed when injecting. However, weak Dirichlet conditions require some additional work since

$$\text{LHS} = v^T(M^{(k)} + SAT)v = v^T(D_1^T K H D_1 + R^{(k)} + SAT)v.$$

Here we assume momentarily that the $Terms = R^{(k)} + SAT$ contain weak Dirichlet contributions from both boundary sides (SAT $= -e_1^T (KS)_1 + e_N^T (KS)_N - (KS)_1^T e_1 + (KS)_N^T e_N + hK(\sigma_0 e_1 e_1^T v + \sigma_1 e_N e_N^T))$. Since $R^{(k)} + SAT$ cannot be proved to be positive semi-definite, instead the $M^{(k)}$ term is split as done in [9] (as well as [10] and [11]):

$$M^{(k)} = h\alpha\widetilde{k^p_L}(S)_1^T(S)_1 + h\alpha\widetilde{k^p_R}(S)_N^T(S)_N + \tilde{M}^{(k)}, \tag{3.14}$$

where $\tilde{M}^{(k)} \geq 0$. Here p is equal to the order of the operator and the $\widetilde{k^p_R}$ and $\widetilde{k^p_L}$ are defined in (3.3).

Gathering all terms ($Terms$ in equation (3.2) and contributions from equation (3.14)) as in [10] we require

$$v^T\tilde{M}^{(k)}v + w_1^TR_1w_1 + w_N^TR_Nw_N \geq 0, \tag{3.15}$$

where $w_{1,N} = [e_{1,N}v,\ (S)_{1,N}v]$ and

$$R_1 = \begin{pmatrix} \sigma_0k_1 & -k_1 \\ -k_1 & \frac{h\alpha}{\widetilde{k^p_L}} \end{pmatrix},\ R_N = \begin{pmatrix} \sigma_1k_N & k_N \\ k_N & \frac{h\alpha}{\widetilde{k^p_R}} \end{pmatrix}.$$

To ensure positivity $\sigma_0 \geq \frac{k_1\widetilde{k^p_L}}{\alpha h}, \quad \sigma_1 \geq \frac{k_N\widetilde{k^p_R}}{\alpha h}$ where α is a constant, which depends on the the order of operator (in this case the sign for σ is opposite to for instance [10] because of how the initial SAT terms were posed). As listed in [11] the α is about 0.36 for a second order operator and 0.25 for a fourth order operator.

To make sure the modified $\tilde{M}^{(k)}$, just as can be put on a form similar to $M^{(k)} = D_1^TKHD_1 + R^{(k)}$ (equation (3.2)), it is possible to choose between two alternatives presented as follows. Either α must be put significantly smaller (typically by a factor 10) than listed in [10] (here $\alpha = 0.25$ for a second fourth order operator) or we introduce a constant factor on the 'norm-term'. Hence we need to guarantee that

$$v^T\tilde{M}^{(k)}v \geq v^TCD_1^TKHD_1v, \tag{3.16}$$

where $C = diag(c_1, c_2 \ldots c_{n-1}, c_n),\ 0 < c_i \leq 1,\ 1 \leq i \leq n$. To find this C we write

$$\begin{aligned}
\tilde{M}^{(k)} =& M^{(k)} - h\alpha\widetilde{k_L^p}(S)_1^T(S)_1 - h\alpha\widetilde{k_R^p}(S)_N^T(S)_N \\
=& D_1^T KHD_1 + R^{(k)} - h\alpha\widetilde{k_L^p}(S)_1^T(S)_1 - h\alpha\widetilde{k_R^p}(S)_N^T(S)_N \\
=& CD_1^T KHD_1 + (I-C)D_1^T KHD_1 + \\
& R^{(k)} - h\alpha\widetilde{k_L^p}(S)_1^T(S)_1 - h\alpha\widetilde{k_R^p}(S)_N^T(S)_N \\
=& CD_1^T KHD_1 + \widetilde{R^{(k)}}
\end{aligned} \tag{3.17}$$

where we used equation (3.14) and equation (3.2) and I is the identity matrix. Hence we require that

$$\widetilde{R^{(k)}} = v^T[(I-C)D_1^T KHD_1 + R^{(k)} - h\alpha\widetilde{k_L^p}(S)_1^T(S)_1 - h\alpha\widetilde{k_R^p}(S)_N^T(S)_N]v \geq 0, \tag{3.18}$$

since it is already given that $CD_1^T KHD_1 \geq 0$ If S is equal to D_1 on the boundary C can be determined easily. However, for S with higher accuracy than D_1, we could only find numerical values for the C constants. (This numerical dependency can be somewhat avoided - see the trailing note).

Returning to the proof with one injected and one weak Dirichlet and assuming a proper C is chosen the LHS can be limited as follows:

$$v^T M^{(k)} v + SAT = v^T(D_1^T HKD_1 + R^{(k)})v + SAT \tag{3.19}$$
$$\geq v^T CD_1^T HKD_1 v + w_N^T R_N w_N, \tag{3.20}$$
$$\geq v^T CD_1^T HKD_1 v \tag{3.21}$$
$$\geq \min_i\{C_{ii}K_{ii}\}(vD_1)^T HD_1 v \tag{3.22}$$
$$= \min_i\{C_{ii}K_{ii}\}\|D_1 v\|_H^2, \tag{3.23}$$

where we used that $R \geq 0$ and $K > 0$ and $SAT = v^T(e_N^T(KS)_N + (KS)_N^T e_N)v + \alpha^{-1}v^T K e_N e_N^T v$.

Using a discrete version of the Poincaré's inequality 3.5 and the Cauchy-Schwartz inequality 3.4 one finally obtains

$$\begin{aligned}\underset{i}{min}\{C_{ii}K_{ii}\}\|D_1v\|_H^2 &\leq v^THf_v && (3.24)\\ &\leq \|v\|_H\|f_v\|_H && (3.25)\\ &\leq C_p\|D_1v\|_H\|f_v\|_H. && (3.26)\end{aligned}$$

Here we note that the using a discrete Poincaré is only guaranteed for injected conditions. However in practice, two weak condition will penalize the boundary such that v_1 and v_N is almost zero and the inequality is likely true for almost all vectors.

Dividing both sides with $\|D_1v\|$ and moving the constant completes the proof

$$\begin{aligned}\|D_1v\|_H &\leq \frac{C_p}{\underset{i}{min}\{C_{ii}K_{ii}\}}\|f_v\|_H && (3.27)\\ &= C\|f_v\|_H, && (3.28)\end{aligned}$$

which mimics theorem Theorem 2.2.1. □

Note. There are possible ways to avoid the numerical problems with the C coefficients, which are introduced by the weak condition. The first suggestion is to modify the initial boundary condition in (2.2) to

$$\begin{aligned}u+\gamma_l u_x &= g_l, \quad x=0\\ u+\gamma_r u_x &= g_r, \quad x=1.\end{aligned}$$

In this case it is not necessary to split $M^{(k)}$, and the $D_1^TKHD_1$ term can be used directly. Another option to avoid the numerical step could be to pretend that the equation has a time dependent term:

$$\delta u_t-(k(x)u_x)_x=f, \quad k(x)>0,$$

for a small δ. This method could possibly avoid introducing C in equation (3.16). See [9] and [10].

Note. The stability proof with Neumann and injected conditions are very similar to the proof above, but without the alpha split restriction (equation (3.14)) since both conditions remove the extra boundary terms. This means injected Dirichlet with SAT Neumann avoids the issue with determining a proper α and C coefficients in equation (3.16).

3.3 Convergence

Convergence for the proposed method is obtained by seeking the difference between the analytical solution u, applied to the discrete approximation equation (2.2) and the discrete solution v. This difference is shown to go to zero as the mesh space h goes to zero.

Theorem 3.3.1. *If approximating Poisson's equation* (2.2) *using the narrow banded SBP-SAT method with homogeneous Dirichlet conditions exactly as done in Theorem 3.2.1, the discrete solution vector, v, converges to the true solution, u, with at least h^p, where $2p$ is the order of SBP operator and the K coefficients are assumed to be sufficiently smooth.*

Proof. We start from the discrete SBP-SAT approximation (3.9) with $\tau_1 = -1, \quad \sigma_1 \geq \frac{k_N \widetilde{k_R}}{\alpha h}$ which gives,

$$-D_2^{(k)} v + H^{-1}(KS)_N^T (e_N^T v - g_r) + \sigma_1 K e_N (e_N^T v - g_r) = f_v + [\tilde{D}_2^{(k)}]_1 v_1. \tag{3.29}$$

Here the injected tilde is dropped for convenience but it is understood that the first row and column has been removed.

Inserting u, now defined as the analytical solution in each grid point to Poisson's equation with Dirichlet boundary conditions (2.2), into this equation yields

$$-D_2^{(k)} u = f_v + T_p(h) + [\tilde{D}_2^{(k)}]_1 u_1, \tag{3.30}$$

where $T_p(h)$ is the truncation error as given by Taylor expansions (using the assumption of smoothness), given by

$$T_p = C \left[\begin{array}{ccccccccccc} h^p & ... & h^p & h^{2p} & ... & h^{2p} & ... & h^p & ... & h^p \end{array} \right]^T \tag{3.31}$$

and C is a constant. Proceeding by defining the error ϵ as $u-v$ and subtracting equation (3.30) from equation (3.29) the following equation is obtained

$$-D_2^{(k)}\epsilon - H^{-1}(KS)_1^T\epsilon_1 + H^{-1}(KS)_N^T\epsilon_N + \sigma K e_1\epsilon_1 + \sigma K e_N\epsilon_N = T_p(h) \tag{3.32}$$

Next we proceed with the same steps as in the proof for the 1D stability, thus utilizing Theorem 3.2.1. The difference is merely a matter of substituting the vector v with ϵ and the right hand side of equation (3.9), f_v with $T_p(h)$. Completing the steps it is obtained that

$$\|D_1\epsilon\|_H \leq C\|T_p(h)\|_H. \tag{3.33}$$

This can be further simplified using the discrete version of the Poincaré's inequality 3.5 into

$$\|\epsilon\|_H \leq C\|T_p(h)\|_H \sim h^p, \tag{3.34}$$

where we yet again note that this is strictly only applicable given at least one injected condition.

Note. The estimated convergence is not sharp for the boundary conditions. This means that the true convergence is higher than what was obtained here, see remark in [10].

□

3.4 Positive Definitness I

To guarantee that the proposed SBP-SAT method not only converges, but also to the one true solution, the discrete solution for e.g. equation (3.11) must be physical and unique. This is the same as require positive definiteness in the LHS. By proving uniqueness, positive definiteness can be guaranteed as follows;

- Positive definiteness means that the SBP-SAT's final LHS has only positive eigenvalues
- It has already been shown in the convergence section 3.3 that that the LHS matrix is at least positive semi-definite (bigger or equal to zero eigenvalues)

- Uniqueness means that there are no eigenvalues equal to zero

The three points give positive definiteness.

Given that an SBP-SAT approximation fulfill a discrete Poincaré, the convergence theorem 3.3.1 and proof 3.3 can be used to show that the SBP-SAT approximation has a unique solution. Assuming two different discrete solutions, u and w, where $u \neq w$, for equation (3.29) and seeking the difference $u - w = q$ gives

$$-\tilde{D}_2^{(k)}\tilde{q} - \tilde{H}^{-1}(K\tilde{S})_N^T\tilde{e}_N^T\tilde{q} + \tilde{K}\tilde{e}_N\tilde{e}_N^T\tilde{q} = [\tilde{D}_2^{(k)}]_1 q_1 \tag{3.35}$$

Following the same steps as in the convergence proof 3.3 one arrives at

$$\|q\|_H \leq 0. \tag{3.36}$$

Equation (3.36) has only one solution: $q = 0$, which contradicts that $u \neq w$ and hence $u = w$.

The next section present a numerical way to show positive definiteness, which does not require the discrete Poincaré hypothesis.

3.5 Positive Definiteness II

In this section positive definiteness is verified by considering the structure of different SBP-SAT matrices. Looking at the structure means each boundary combination and order must be considered separately. There are in practice however a finite number of common boundary condition combinations and SBP orders, and completing the necessary steps for each combination is possible.

The aim is to show that the LHS always has only one solution, that is:

$$-v^T H D_2^{(k)} v + v^T H \text{SAT} v > 0 \tag{3.37}$$

The first step is to select a specific boundary condition combination. For instance injected on the left boundary and no flow Neumann on the right boundary. Expanding equation (3.37) and using the injected boundary gives

$$v^T(\widetilde{M^{(k)}} - \widetilde{\bar{K}S}) + v^T \widetilde{e_N^T K S_N} v > 0, \tag{3.38}$$

where the tilde signifies that the first row and column has been removed and v is now assumed to be an $n-1$ vector $v = \left[v_2, v_3, \ldots, v_N\right]^T$. Proceed with expanding the $\widetilde{M^{(k)}}$ term and canceling the boundary gives

$$v^T(\widetilde{D_1^T HKD_1} + \widetilde{R^{(k)}})v \tag{3.39}$$

Since $\widetilde{R^{(k)}} \leq 0$ and K is positive definite, equation (3.39) can be rewritten as

$$v^T(\widetilde{D_1^T HKD_1} + \widetilde{R^{(k)}})v \geq \min_i\{K_{ii}\} v^T \widetilde{D_1^T HD_1} v. \tag{3.40}$$

At this point the K coefficients influence on positive definiteness has been eliminated and it is sufficient to show that

$$v^T \widetilde{D_1^T HD_1} v + v^T \widetilde{H\text{SAT}} v > 0. \tag{3.41}$$

where the K coefficients have been removed from the SAT terms as well.

The inequality 3.41 is true if the LHS matrix is positive definite. Positive definiteness of the LHS can be shown by utilizing the Levy-Desplanques theorem found in [12]:

Theorem 3.5.1 (Levy-Desplanques). *A strictly diagonally dominant matrix or a diagonally dominant irreducible matrix with at least strict dominance in one row is nonsingular. In other words, let $A \in \mathbf{C}^{n,n}$ be a matrix satisfying the property*

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}| \qquad \forall i;$$

or any i given irreducibility, then $\det(A) \neq 0$

Here irreducibility is given as (obtained from wolfram mathworld online webpage [13]):

Irreducibilty: A square $n \times n$ matrix $A = a_{ij}$ is called irreducible if the indices $1, 2, ..., n$ cannot be divided into two disjoint nonempty sets $i_1, i_2, ..., i_\mu$ and $j_1, j_2, ..., j_\nu$ (with $\mu + \nu = n$) such that $a_{i_\alpha j_\beta} = 0$ for $\alpha = 1, 2, ..., \mu$ and $\beta = 1, 2, ..., \nu$.

Hence a matrix is irreducible if and only if it cannot be placed into block upper-triangular form by simultaneous row/column permutations. In addition, a matrix is irreducible if and only if its associated graph is strongly connected.

Since an irreducible diagonally dominant matrix is positive definite, the aim is to show that $v^T \widetilde{D_1^T H} D_1 v + v^T \widetilde{H\mathrm{SAT}} v$ is irreducible diagonally dominant. This is easiest done by typing out the matrix. The full $v^T \widetilde{D_1^T H} D_1 v + v^T \widetilde{H\mathrm{SAT}} v$ is as follows

$$h \begin{bmatrix} 0.75 & & -0.25 & & & & \\ & 0.5 & & -0.25 & & & \\ -0.25 & & 0.5 & & -0.25 & & \\ & \ddots & & \ddots & & \ddots & \\ & & -0.25 & & 0.5 & & -0.25 \\ & & & -0.25 & & 0.75 & -0.5 \\ & & & & -0.25 & -0.5 & 0.75 \end{bmatrix}, \tag{3.42}$$

where the first row and column of the $D_1^T H D_1$ term has been removed as dictated by the injected condition. All horizontal spaces indicate zero's. This matrix can be shown to be irreducible since for each node in the matrix connected graph, every node is reachable from every other node which means it is strongly connected. This is most easily determined by drawing the connected graph for a small representation of the matrix and then noting that the inner regions is a simple tri-diagonal repetition. This inner repetition will always be strongly connected to second adjacent nodes. Furthermore the matrix is diagonally dominant with at least one strict dominance (the first row is strictly dominant and the remaining rows are just dominant by equivalence). Using the Levy-Desplanques theorem 3.5.1, it is concluded that the matrix is non-singular. Since it was previously shown that the matrix is at least positive semidefinite, the matrix must be positive definite.

The same result holds for injected on the right boundary and Neumann condition on the left since the matrix is symmetric. Completing the exact same steps for the fourth and sixth order operators is straight forward. Furthermore, setting a weak Dirichlet on either boundary only changes the outline by including an extra $v^T(S^T + S)v$ term.

3.6 Improved Boundary Flux

To improve the boundary flux for the SBP-SAT method for problems such as in section 4.1 extra penalty terms are introduced. In 1D a possibility is to use the following scheme (assuming one weak Dirichlet and one Neumann boundary condition):

$$\begin{gathered} -D_2^{(k)}v+\tau_0 H^{-1}(KS)_1^T(e_1^T v-g_l)+\sigma K e_1(e_1^T v-g_l)+ \\ \tau_1 H^{-1}e_N((KS)_N v-g_r)+BFX=f_v. \end{gathered} \tag{3.43}$$

The BFX term is the extra penalty and is written as

$$\eta H^{-1}(KS_{fx})_N^T((KS_{fx})_N v-g_r) \tag{3.44}$$

The $\eta \leq 0$ is a constant which determines the strength of the penalty term. The S_{fx} term is a boundary derivative as previously defined. The difference from the S used in equation (3.43) is that S_{fx} should be of higher order (so that the new boundary penalty has higher convergence rate). For convenience an eighth order operator can be used.

The BFX term will not destroy the stability or convergence of (3.43) since the term is positive definite, symmetric and converges to the true solution. The added terms can however reduce the computational efficiency since the final matrix will contain more non-zero elements. It will also increase the matrix condition number to various degrees. A 2D extension of the BFX term is easily obtained by adding a second term for the new dimension and proceeding with the usual appropriate modifications.

Setting Neumann conditions on the east and west boundary of a 2D square domain, the extra added BFX penalty term is written as

$$\eta_E H^{-1}(KS_{fx})_E^T((KS_{fx})_E v-g_r)+\eta_W H^{-1}(KS_{fx})_W^T((KS_{fx})_W v-g_r), \tag{3.45}$$

where the matrix subscripts E and W signifies that the rows corresponding the East and West boundaries have been selected from the matrix (For matrix notations and structure see Appendix A).

Chapter 4

Numerical Results

This chapter contains numerical results based on the theory from previous chapters. This chapter is divided into three sections which treats sufficiently smooth problems, specifically problematic discontinuous for 1D and 2D problems and a real world test scenario, where data has been collected for the purpose of oil reservoir modeling.

4.1 Test Case I - Analytical Solution

In this section we verify the convergence analysis on the Poisson's equation with variable coefficients by comparing a sufficiently smooth analytical solution to the corresponding SBP-SAT solution. Both the $\log_{10}$ error and convergence order of the discrete solution is presented.

4.1.1 Convergence and Error Formulae

The convergence is measured as

$$q = \log_{10}\left(\frac{||u^t - v^{(N_1)}||_h}{||u^t - v^{(N_2)}||_h}\right) \Big/ \log_{10}\left(\frac{N_1}{N_2}\right)^{1/d}, \tag{4.1}$$

where the 1D the error norm is:

$$||u-v||_{l_2} = \sqrt{h\sum_{i=0}^{n-1}|u(x_i)-v_i|^2}. \tag{4.2}$$

The 2D error norm is:

$$||u-v||_{l_2} = \sqrt{h_x h_y\sum_{i=0}^{n-1}\sum_{j=0}^{m-1}|u(x_i,y_j)-v_{ij}|^2}. \tag{4.3}$$

Finally the boundary flux error is measured in 2D as

$$\sqrt{\frac{1}{\tilde{N}}\sum_p\sum_k\left|(\bar{n}\cdot\boldsymbol{\nabla}u)|_{x_p,y_k}-(Nv)_{pk}\right|^2}. \tag{4.4}$$

Here N represents the normal boundary derivate matrix which will be zero for inner points and give the boundary derivate for boundary points. $\tilde{N}$ is the total number of boundary points with Neumann conditions and the summation spans over p,k, which represent boundary points with Neumann conditions. $(\bar{n}\cdot\boldsymbol{\nabla}u)|_{x_p,y_k}$ is the continuous boundary derivate evaluated at point (x_p,y_k). An 8^{th} order D-operator is used in $\tilde{N}$ (given that the solution is sufficiently smooth) so that the numerical error in computing the gradient is less than the numerical error from solving the SBP-SAT equation.

4.1.2 Analytical Testing

For the first test, the Poisson's equation with variable coefficients (2.1) with the following settings is used:

$$K(x,y) = \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} = \begin{pmatrix} x^3y & 0 \\ 0 & \frac{x}{y-2} \end{pmatrix} \tag{4.5}$$

and

$$\begin{aligned} f =& 3x^2y\sin x\sin y + x^3y\cos x\sin y \\ &+ \frac{x}{(y-2)^2}\cos x\cos y + \frac{x}{y-2}\cos x\sin y, \end{aligned} \tag{4.6}$$

on a square domain $x\in[5,10], y\in[5,10]$. The boundary is defined with Neumann conditions on the upper and lower boundary (perpendicular to the y-axis) and Dirichlet on the left and right boundary (perpendicular to the x-axis) as follows:

$$g_{x=5}(x,y) = -x^3 y \sin x \sin y, \qquad x = 5 \tag{4.7}$$

$$g_{x=10}(x,y) = x^3 y \sin x \sin y, \qquad x = 10 \tag{4.8}$$

$$g_{y=5,10}(x,y) = \cos x \sin y, \qquad y = 5, 10 \tag{4.9}$$

The analytical solution is given by

$$u = \cos x \sin y, \tag{4.10}$$

and is plotted in Figure 4.1.

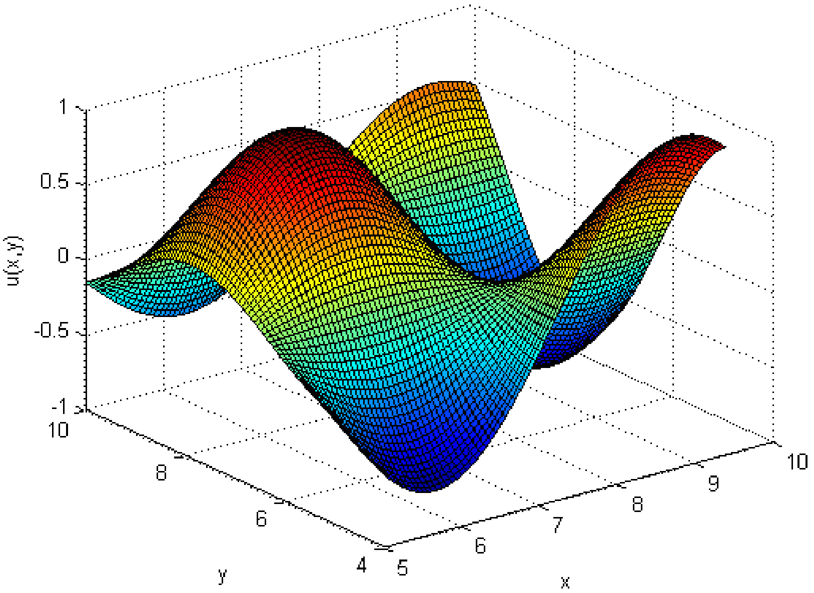

Figure 4.1: The analytical solution

Equation (4.10) was discretized and the SBP-SAT method was applied resulting in the following equation system

$$\begin{array}{ll}
-(D_{2x}^{(k_{11})} + D_{2y}^{(k_{22})})v & \\
+H_x^{-1}[(K_{11}S_x)^T]_E(e_{Ea}^T v - g_E) + \frac{1}{8h}K_{11}e_{Ea}(e_{Ea}^T v - g_E) & \\
-H_x^{-1}[(K_{11}S_x)^T]_W(e_{We}^T v - g_W) + \frac{1}{8h}K_{11}e_{We}(e_{We}^T v - g_W) & = f_v. \\
+H_y^{-1}e_{No}([(K_{22}S_y)]_N v - g_N) & \\
+H_y^{-1}e_{So}([(K_{22}S_y)]_S v - g_S) &
\end{array} \tag{4.11}$$

The meaning of the symbols in equation (4.11) is given in Appendix A. The resulting convergence obtained from applying the SBP-SAT method is presented in table and graphs (Table 4.1, Table 4.2, Figure 4.2, Figure 4.3). Each table lists the error and convergence for the solution using 2^{nd}, 4^{th} and 6^{th} order narrow banded SBP operators. Table 4.1 lists the total domain error while Table 4.2 lists the boundary flux error on the boundaries using Neumann condition (upper and lower boundaries). To summarize the convergence more clearly the same information is given in two separate plots; Figure 4.2 and Figure 4.3. The legend in each graph explains the different lines, which represent different SBP orders. Each figure shows six lines, three lines for the numerical SBP-SAT error and three lines for the analytical slope. The actual size of the analytical error bears no significance in the figures, the important information is the convergence slope, which should correspond to the numerically obtained slope. The analytical slope is based on a best case scenario. Hence we expect a convergence of 2, 4 and 5.5 for the corresponding SBP orders (see [14]). On the boundary the analytical order of accuracy is $p/2 + 1$ [10]. Furthermore, for a Neumann boundary error measure it is expected that one will loose one order of accuracy in the derivation process. Hence the analytical convergence slopes are put as $p/2$, which give a final convergence slope of 1, 2 and 3.

	2nd order		4th order		6th order	
N	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q
21 × 21	1.59e-02	0.00	5.71e-04	0.00	1.18e-04	0.00
31 × 31	6.97e-03	2.12	9.82e-05	4.52	1.34e-05	5.59
41 × 41	3.90e-03	2.08	2.86e-05	4.40	2.86e-06	5.52
51 × 51	2.48e-03	2.06	1.11e-05	4.33	8.65e-07	5.47
61 × 61	1.72e-03	2.05	5.17e-06	4.28	3.27e-07	5.44
71 × 71	1.26e-03	2.04	2.72e-06	4.24	1.44e-07	5.41
81 × 81	9.65e-04	2.04	1.56e-06	4.21	7.06e-08	5.39

Table 4.1: Total domain error and convergence

	2nd order		4th order		6th order	
N	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q
21 × 21	2.15e-02	0.00	1.91e-03	0.00	1.66e-03	0.00
31 × 31	1.22e-02	1.46	9.95e-04	1.68	5.44e-04	2.86
41 × 41	8.43e-03	1.32	6.29e-04	1.64	2.30e-04	3.08
51 × 51	6.52e-03	1.17	4.31e-04	1.73	1.17e-04	3.09
61 × 61	5.41e-03	1.05	3.11e-04	1.82	6.84e-05	3.02
71 × 71	4.67e-03	0.97	2.34e-04	1.89	4.39e-05	2.92
81 × 81	4.13e-03	0.93	1.81e-04	1.95	3.02e-05	2.84

Table 4.2: Neumann boundary derivate error and convergence

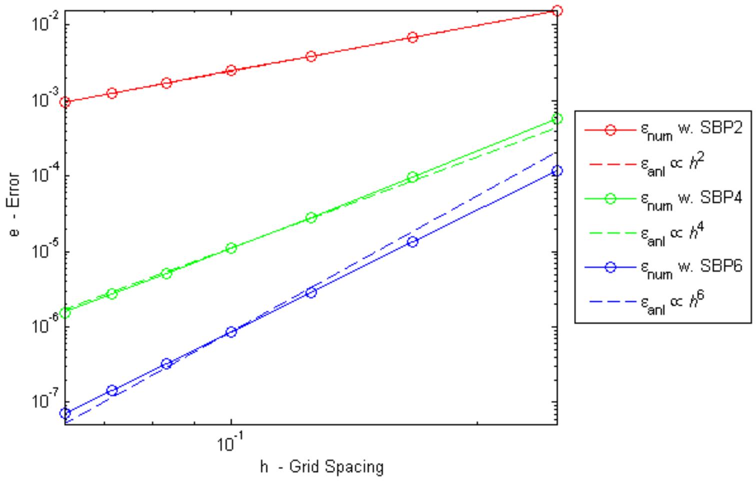

Figure 4.2: All SBP-SAT total domain convergence

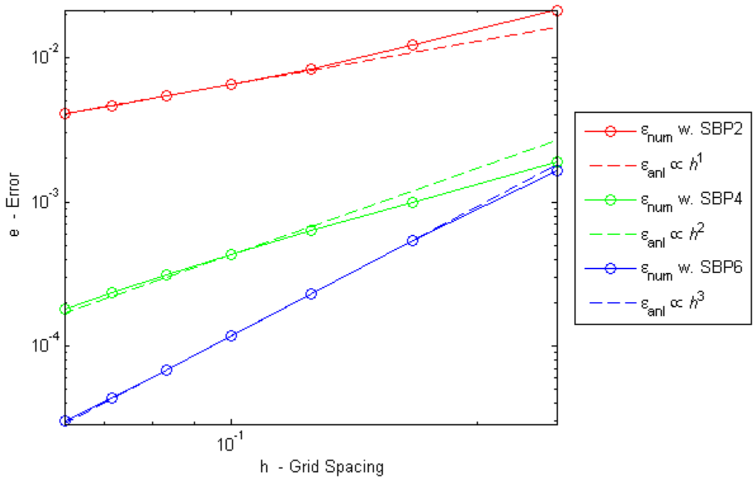

Figure 4.3: All SBP-SAT Neumann boundary convergence

Evaluating the results, it is clear that the obtained convergence slopes correspond well with the theoretical values. In all figures the simulation convergence lines match the analytical lines, both for the total error and boundary derivate convergence. There are some minor differences from the theory most easily detected in the tables, but it is all within an acceptable range.

4.2 Numerical Testing of Improved Boundary Flux

Previously in Section 3.6 it was suggested that the boundary flux convergence could be increased with an added penalty term. To test and verify this numerically the same test equation (4.5)-(4.10) used in Section 4.1 is reused here. The penalty coefficient is put as $\eta = -200$ and eight order D1 operators are used in the penalty term [1]. The Neumann boundary flux error and convergence from setting is presented in Table 4.4 and Figure 4.4. In the figure, the theoretical convergence lines for a non-penalized problem is kept as reference. The total domain error and convergence is presented in Table 4.3.

	2nd order		4th order		6th order	
N	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q
21 × 21	1.59e-02	0.00	5.70e-04	0.00	1.18e-04	0.00
31 × 31	6.97e-03	2.12	9.81e-05	4.52	1.34e-05	5.59
41 × 41	3.90e-03	2.08	2.86e-05	4.40	2.86e-06	5.52
51 × 51	2.48e-03	2.06	1.11e-05	4.33	8.64e-07	5.48
61 × 61	1.72e-03	2.05	5.17e-06	4.28	3.26e-07	5.45
71 × 71	1.26e-03	2.04	2.72e-06	4.24	1.43e-07	5.42
81 × 81	9.65e-04	2.04	1.56e-06	4.21	7.03e-08	5.40

Table 4.3: Total domain error and convergence with penalty

	2nd order		4th order		6th order	
N	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q
21 × 21	1.21e-05	0.00	5.15e-06	0.00	5.17e-06	0.00
31 × 31	2.10e-06	4.49	7.98e-07	4.79	6.19e-07	5.45
41 × 41	6.11e-07	4.42	2.06e-07	4.85	1.28e-07	5.63
51 × 51	2.38e-07	4.32	7.06e-08	4.89	3.70e-08	5.71
61 × 61	1.12e-07	4.21	2.93e-08	4.92	1.32e-08	5.74
71 × 71	6.01e-08	4.08	1.38e-08	4.93	5.52e-09	5.75
81 × 81	3.58e-08	3.95	7.22e-09	4.94	2.59e-09	5.74

Table 4.4: Neumann boundary derivate error and convergence with penalty

[1] Selecting an eight order operator leaves a minor extra error possibility since the same eight order operator was used to determine the boundary error. This will however only decrease the actual convergence

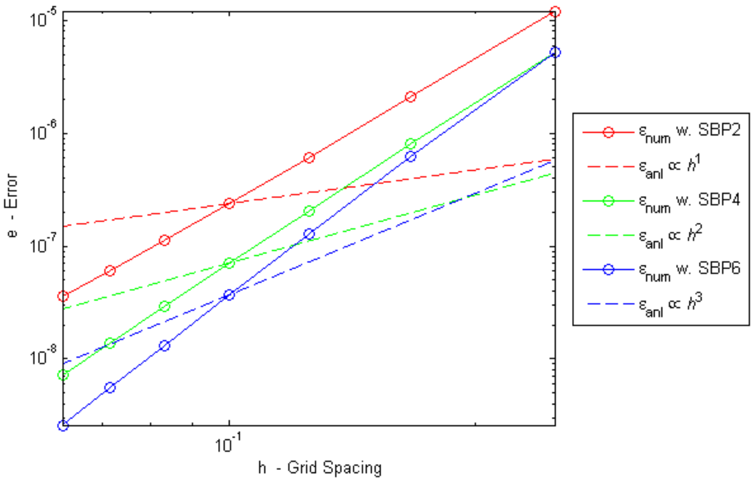

Figure 4.4: Neumann boundary derivate convergence with penalty term

Comparing Table 4.4 and Table 4.2, the latter obtained in section 4.1, there is a clear improvement in the convergence. With penalty, the Neumann boundary derivate error converges as 4, 5 and 5.75. Without penalty, the Neumann boundary derivate error converges as 1, 2 and 3. Comparing the domain error in Table 4.3 and Table 4.1 it is furthermore deduced that adding a penalty does not destory the solution.

As an alternative harder scenario a second test case is designed. The same equation ((4.5)-(4.10)) is reused, but in this case the Neumann condition is put on the left and right edge. The analytical boundary condition on the right edge is $n \cdot \nabla p = x^3 y \cos x \sin y$. Since this condition is $\propto x^3$, the Neumann boundary values are large. After setting $\eta = -20$ (more negative values caused increased domain errors), the resulting convergence and error is presented in Table 4.5, Table 4.6, Figure 4.5 and Figure 4.6.

	2nd order		4th order		6th order	
N	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q
21 × 21	4.54e+00	0.00	3.27e-02	0.00	1.01e-03	0.00
31 × 31	1.99e+00	2.12	9.41e-03	3.19	3.01e-04	3.10
41 × 41	1.11e+00	2.08	3.56e-03	3.48	7.54e-04	-3.29
51 × 51	7.08e-01	2.07	2.82e-04	11.62	1.07e-03	-1.62
61 × 61	4.92e-01	2.03	8.44e-04	-6.12	3.72e-04	5.92
71 × 71	3.62e-01	2.03	1.10e-03	-1.73	3.00e-03	-13.75
81 × 81	2.73e-01	2.15	2.43e-03	-6.05	9.71e-04	8.56

Table 4.5: Total domain error and convergence with penalty. Neumann condition on left and right boundary.

	2nd order		4th order		6th order	
N	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q	$log_{10}(\epsilon_{num})$	q
21 × 21	4.39e-06	0.00	2.85e-06	0.00	1.63e-06	0.00
31 × 31	1.44e-06	2.87	8.00e-07	3.26	4.06e-07	3.57
41 × 41	6.64e-07	2.76	3.07e-07	3.43	1.26e-07	4.19
51 × 51	3.68e-07	2.71	1.43e-07	3.48	4.89e-08	4.33
61 × 61	2.28e-07	2.67	7.67e-08	3.50	2.24e-08	4.37
71 × 71	1.53e-07	2.64	4.50e-08	3.51	1.15e-08	4.41
81 × 81	1.08e-07	2.62	2.84e-08	3.51	6.25e-09	4.60

Table 4.6: Neumann boundary derivate error and convergence with penalty. Neumann condition on left and right boundary.

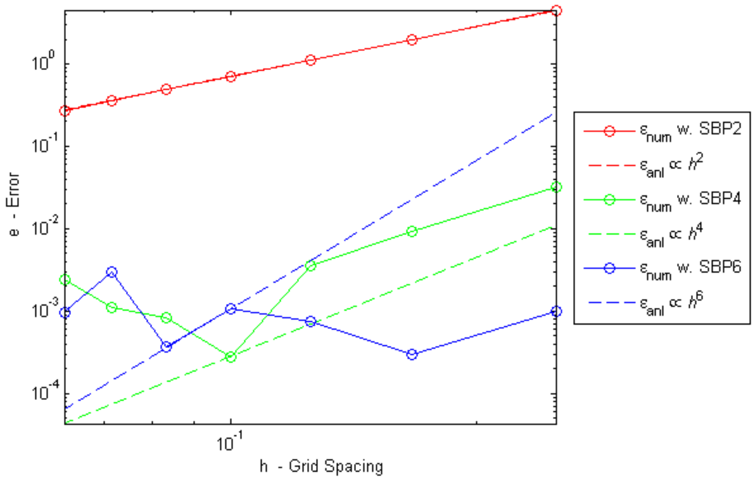

Figure 4.5: Total domain convergence with penalty term. Neumann condition is put on left and right boundary

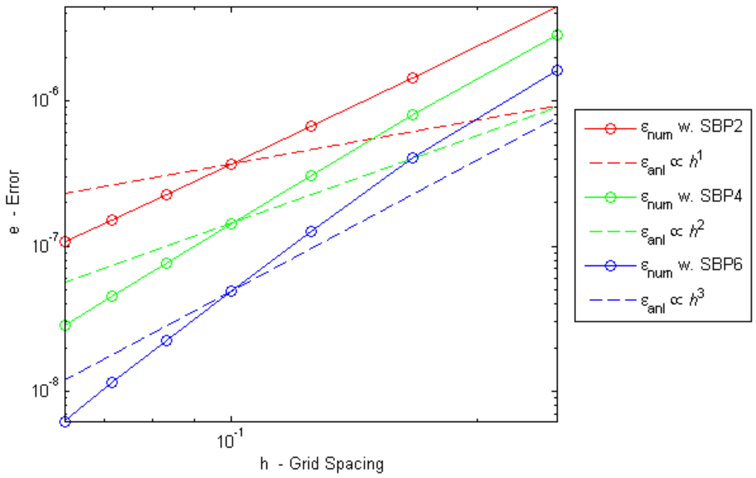

Figure 4.6: Neumann boundary derivate convergence with penalty term. Neumann condition is put on left and right boundary

Hence, setting Neumann condition on the left and right boundary stops the domain convergence completely as seen in Figure 4.5. The Neumann boundary derivate error does converge to the true solution, but at a more modest rate than previously observed. With Neumann boundary at top and bottom the convergence is 4, 5 and 5.75, but on left and right boundary for the same equation the convergence is 2.6, 3.5 and 4.6. The cause of the problem is the large boundary values on the right edge, of which the root problem is that $k_{11} = x^3y$. The penalty term values will become very large and in turn dramatically increase the condition number of the final matrix.

Note. In the above examples η was put to a constant for all orders of operators. This is not necessarily (and for most cases not) the best option. Typically a smaller value of η for lower SBP orders generates smaller errors, while higher operator orders require larger η.

4.3 Test Case II - Discontinuous Media

In this section a few test scenarios are presented on how the SBP-SAT method performs in a discontinuous environment. The focus is on highlighting the issues with applying the SBP-SAT on this type of problems. For every computed case a reference solutions is obtained using MRST. The problem parameters have consistently been scaled to fit within oil reservoir parameter ranges. Only solutions using second order SBP operators are presented and grid refining is done in a nearest neighbor sense. This means that in 1D, for each node, two new equidistant nodes are inserted in between previous nodes. In 2D, grid refining means subdividing each cell (area around node) into nine new cells. The new nodes are given K values corresponding to the K value of the closest node. The nearest neighbour refinement process is illustrated in Figure 4.7 and Figure 4.8. Nodes are represented by blue dots and its K value is assumed to be valid in a cell area around it as marked by the black and white colors.

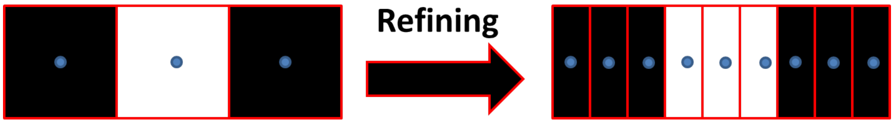

Figure 4.7: Nearest Neighbour refinement 1D.

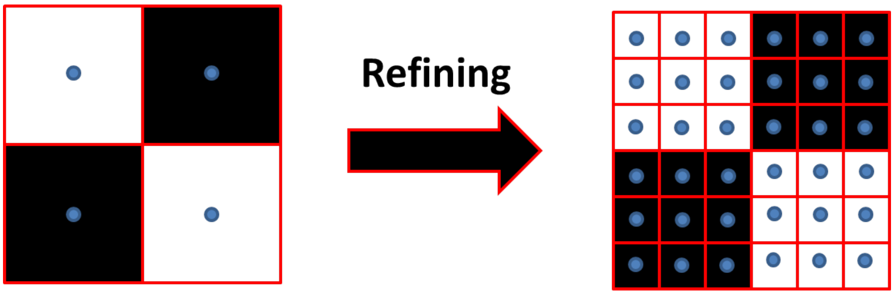

Figure 4.8: Nearest Neighbour refinement 2D.

The first test example is a simple step function in 1D. A typical real world situation of this is a region of stone on the left and a porous region on the right. The K coefficients are displayed in Figure 4.9. Each cell center represents a node. The output pressure distribution is displayed in Figure 4.10.

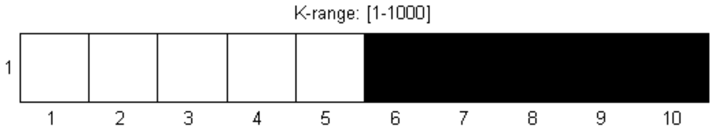

Figure 4.9: The K coefficients

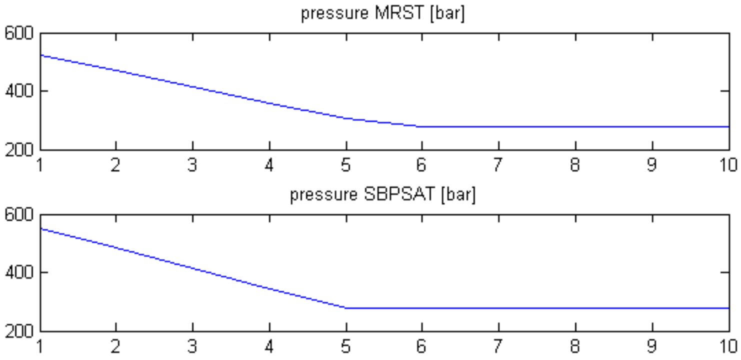

Figure 4.10: Final pressure distribution

From Figure 4.10 It is clear that the SBP-SAT method will 'cut off' the solution (at point 5) one point earlier than the actual step (at point 6). The MRST solution differ from the SBP-SAT solution in this regard. To distinguish which solution is more correct one could attempt to obtain an analytical solution. The problem is however that the analytical solution require some guessing. One could assume that the true K value changes from 0 to 1000 somewhere in the middle of point 5 and 6, or one could assume that the K value changes to 1000 right at point 5 or 6. Depending on which assumption is made, the analytical solution will look more or less like SBP-SAT. However a desirable characteristic of oil reservoir solvers is that it allows no leakage, which is used to select a preferred assumption. Leakage is the issue of observing a simulated flow (equal to a low pressure drop) over a region where one would expect no flow. For instance a small high permeability obstacle should not allow liquid to pass through unhindered. The leakage problem is examplified in the following problem tests which thereafter are used to distinguish the more correct assumption.

The second test example shows a scenario which corresponds to a region with a single obstruction. For example a small granite region in porous sand stone. The K coefficients are displayed in Figure 4.11 and the output pressure distribution is displayed in Figure 4.12.

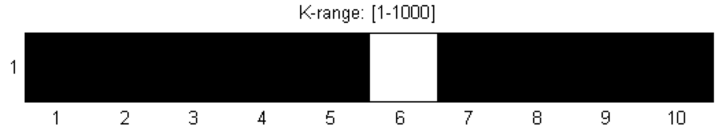

Figure 4.11: The K coefficients

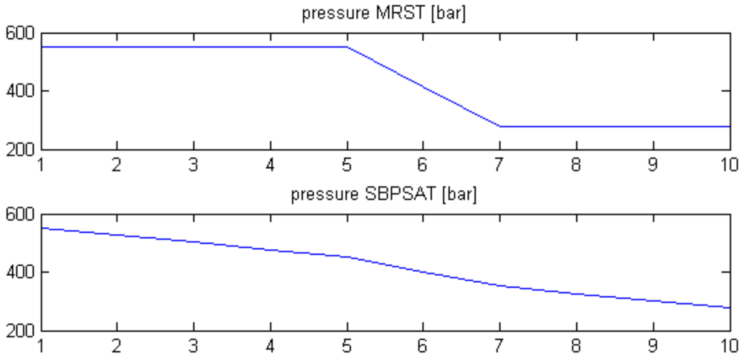

Figure 4.12: Final pressure distribution

The difference between the two solutions is obvious. The pressure drop is much greater for the MRST solution over the obstacle at point 6 to 7. The SBP-SAT method smooths the solution and will therefore almost ignore the obstacle. In other words, it favors leakage. This is not necessarily problematic in 1D since grid refining will reduce the differences as seen in Figure 4.13 and Figure 4.14 where the solutions will eventually converge toward the same solution.

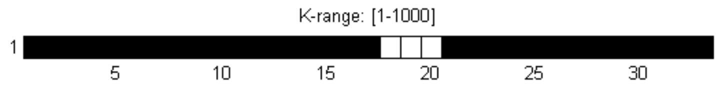

Figure 4.13: The K coefficients

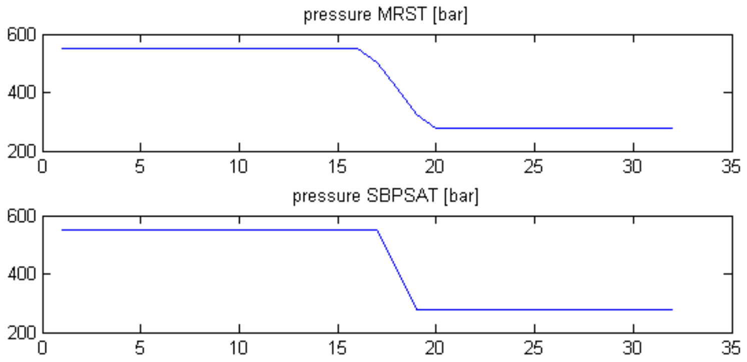

Figure 4.14: Final pressure distribution

In difference from 1D, all leakage problems cannot be solved in 2D by grid refining. This is illustrated in the next test which was specifically designed to produce a leakage problem. The test is a model of a thin diagonal of porous material running across for instance a granite region. The material domain is displayed (K coefficients) in Figure 4.15. Since the diagonal porous squares are strictly separated, one would expect there to be no or little flow following the diagonal. Figure 4.16 displays the resulting pressure distribution on a coarse grid. Figure 4.17 corresponds to the same K coefficients, but with one nearest neighbour refinement step.

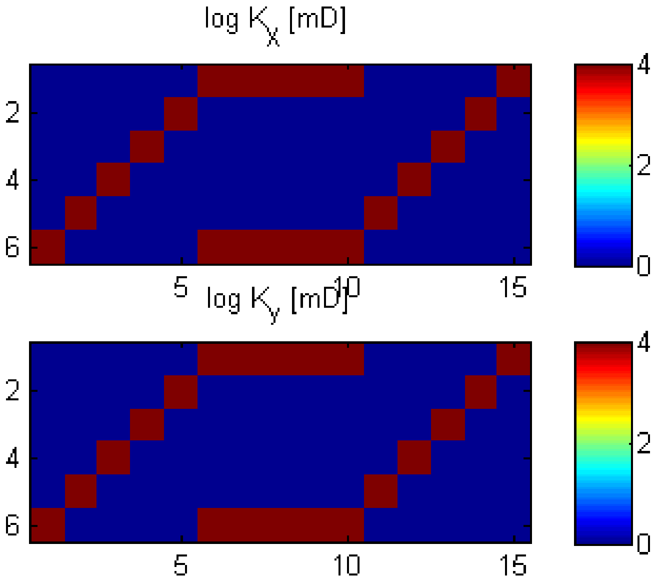

Figure 4.15: The K coefficients

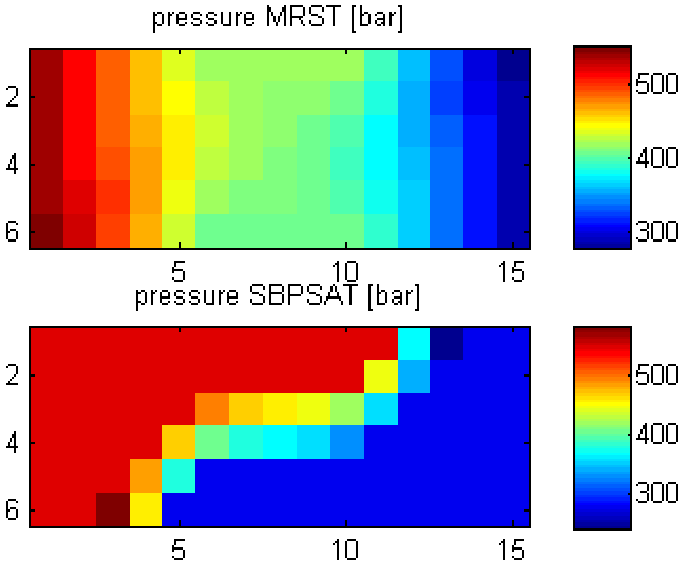

Figure 4.16: Final pressure distribution from a coarse grid.

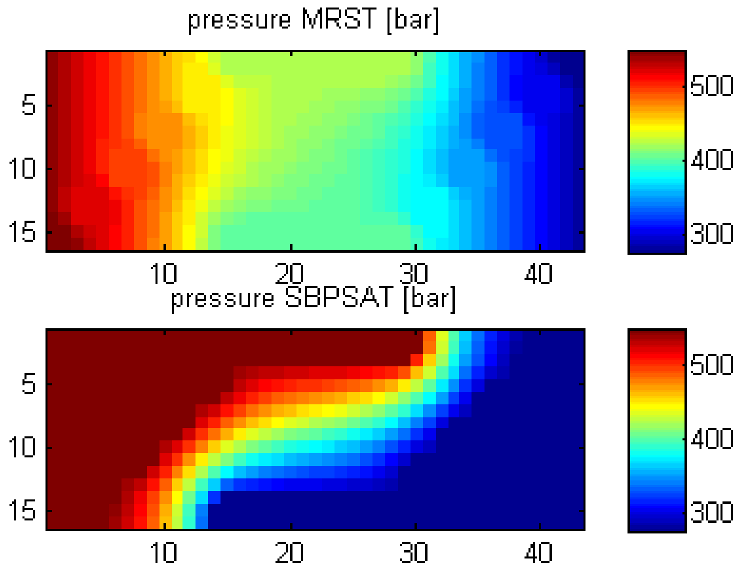

Figure 4.17: Final pressure distribution from a refined grid.

Figure 4.16 shows that MRST and SBP-SAT gives two very different solutions. The SBP-SAT method allows strong flow across the diagonal, while MRST allows minor flow. Figure 4.17 shows that refinement has no impact on the difference and that the solutions converge to different solutions. The problem lies in the corners of the coefficients. For a very sharp diagonal the SBP-SAT solution will allow leakage at the corners while MRST will not. And since leakage is undesirable (not natural) SBP-SAT converges to the wrong solution. Refining the grid to an arbitrary depth in a nearest neighbor approach will not reduce the problem; sharp corners will remain (see Figure 4.20). If one inverts the K coefficients (granite diagonal in porous media) in Figure 4.15 a similar problem emerges. The resulting pressure distribution is displayed in Figure 4.18.

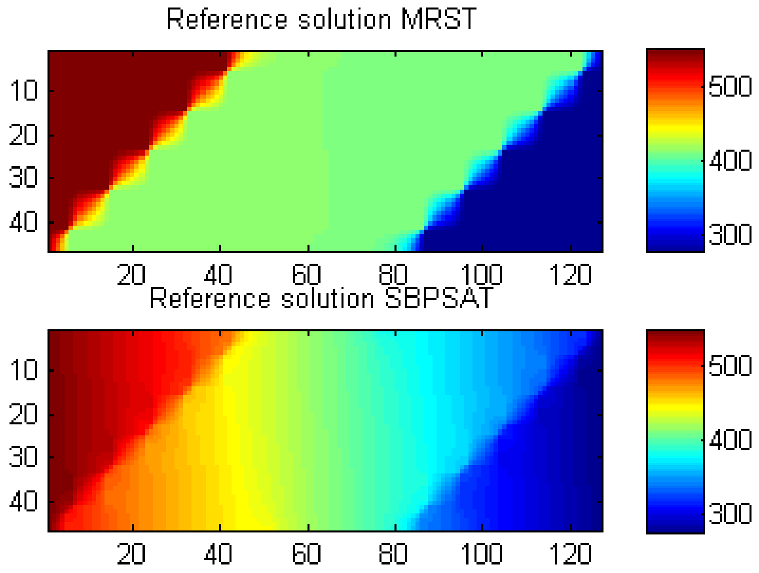

Figure 4.18: Final pressure distribution on a refined grid.

Figure 4.16 represents an already twice refined reference solution. Clearly further refinement will make no difference. Here flow across the granite section is not expected but still observed with the SPB-SAT method. The problem with leakage in 1D and 2D is exemplified in Figure 4.19 and Figure 4.20. It is clear that any leakage will remain as long as the refinement process is done by the nearest neighbor approach in 2D. The leakage stops with refinement in 1D where the diagonals never occur (For discussion on other refinement techniques see Section 6.1).

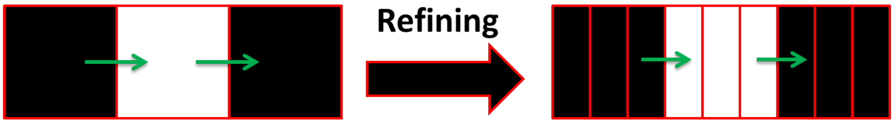

Figure 4.19: Refinement and leakage for SBP-SAT. Refining in 1D fixes the leakage over a single cell gap.

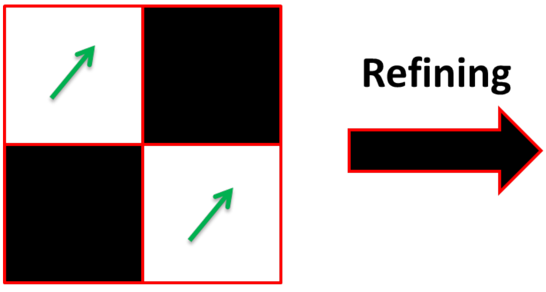

Figure 4.20: Refinement and leakage for SBP-SAT. In 2D a diagonal run of large or small K coefficients will cause problems despite refinement.

Note. Here it is also clear why tests with higher order operators have been omitted. For instance, a fourth order operator uses a five point stencil. Using more points on each side will exaggerate the leakage problem since the underlaying Taylor expansions assumes a sufficiently smooth solution. This means small non-smooth details from a not sufficiently smooth situation gets 'smoothed out'.

Note. Grid refining helps for poorly chosen α for the σ constant (a small σ with large K coefficients jumps can result in unstable boundary treatment, see section 3.2) in the weak Dirichlet penalty terms (3.14). This constant is most troublesome in non-smooth environments, where typically one must chose higher values. If the domain is grid refined, there will be a smooth (or flat) region along the boundary. This region is more natural for the SPB-SAT operators and an α of about $0.25/h$ is often sufficient.

4.4 SPE Comparative Solution Project

To test the performance of the SBP-SAT method in a more realistic environment, data from the SPE Comparative Solution Project is used, which supplies underground pearmability and porosity measurements from an oil reservoir. However, since the SBP-SAT method has been shown to produce undesirable results (section 4.3), the problem solved here is a simplification of the real problem. Only the underlaying Darcy equation is considered with 2D slices from the 3D data.

4.4.1 Test settings

Every layer are given the same conditions. A Neumann no flow condition is put on the upper and lower boundaries, while a Dirichlet pressure condition of 8000 psi is put on the right boundary and a Dirichlet pressure condition of 4000 psi is put on the left boundary. The SBP-SAT method implements all conditions weakly, with $\eta \sim -20/h$ for the Dirichlet boundary. Second order operators are used exclusively. To implement the Dirichlet conditions with minimum error (~ 0) one can also use injected conditions instead of weak Dirichlet on the right and left boundary. Here the decision was to keep the weak Dirichlet to show its functionality, since they would typically be used for interfaces and internal wells etc. (One could also obtain better results using mixed conditions ($u + u_x$) instead of simply Dirichlet)

4.4.2 Measuring Darcy velocity

Three different plots are presented for each test case, the K coefficients, the final pressure distribution and the Darcy velocity. For consistency, the Darcy velocity is calculated from the pressure distribution using SBP operators. The obtained Darcy velocity is however not a perfect estimate for two reasons. The first reason is that the analytical solution is not sufficiently smooth. A high order SBP derivate estimator gives inaccurate results as the Taylor expansions are no longer applicable. Hence a second order SBP operator was used as an estimator instead of the eighth order operator used in Section 4.1. However a second order SBP operator still assumes continuity in the first derivate, which is not guaranteed here. Secondly, with this method, the in and out fluxes are obtained on nodal values. MRST uses nodal values only for the pressure, but calculate the flux on the cell intersections. Similarly MRST sets the boundary conditions on the outer domain-boarder cell boundaries. If one calculates the fluxes directly in the nodes for MRST, it will not guarantee conservation. This means that using this method MRST will appear to produce unphysical solutions (more steady state inflow than outflow for instance). That said, an estimate, albeit crude, of the flux is still presented.

4.4.3 Tests

The first test is done on layer 4 of the SPE comparative solution project data. This is one of the upper, 'more nice' layers with less dramatic variations. The K coefficients are shown in Figure 4.21. The pressure distribution and Darcy velocity are shown in Figure 4.22 and Figure 4.23 respectively.

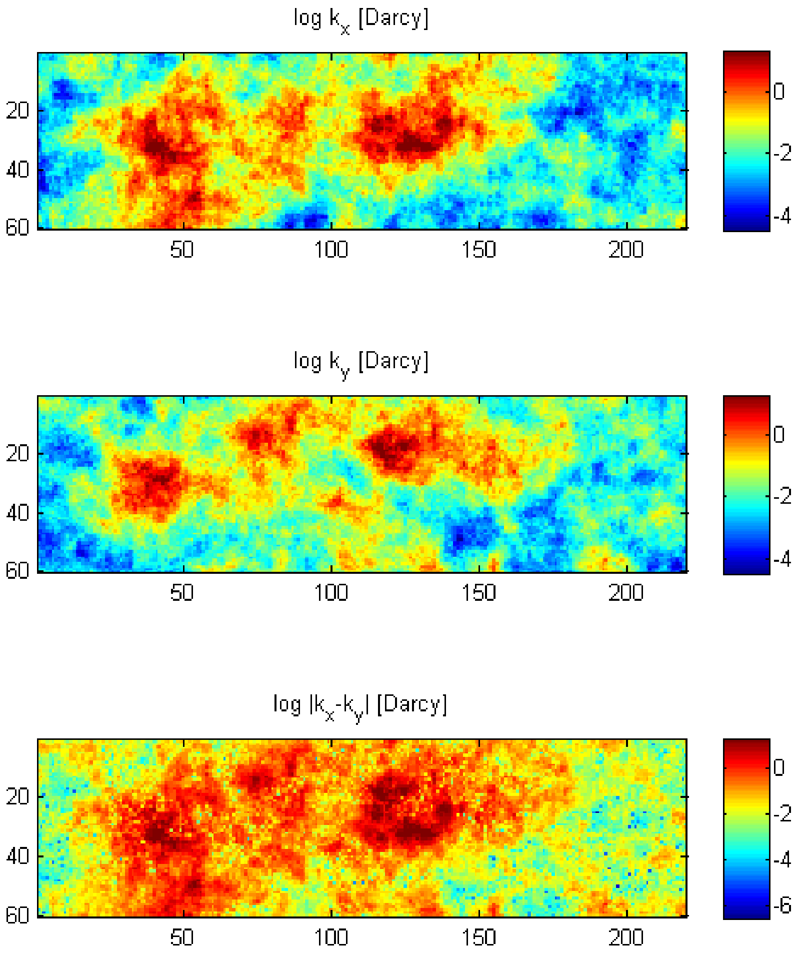

Figure 4.21: The K coefficients as well as the difference between them for layer 4.

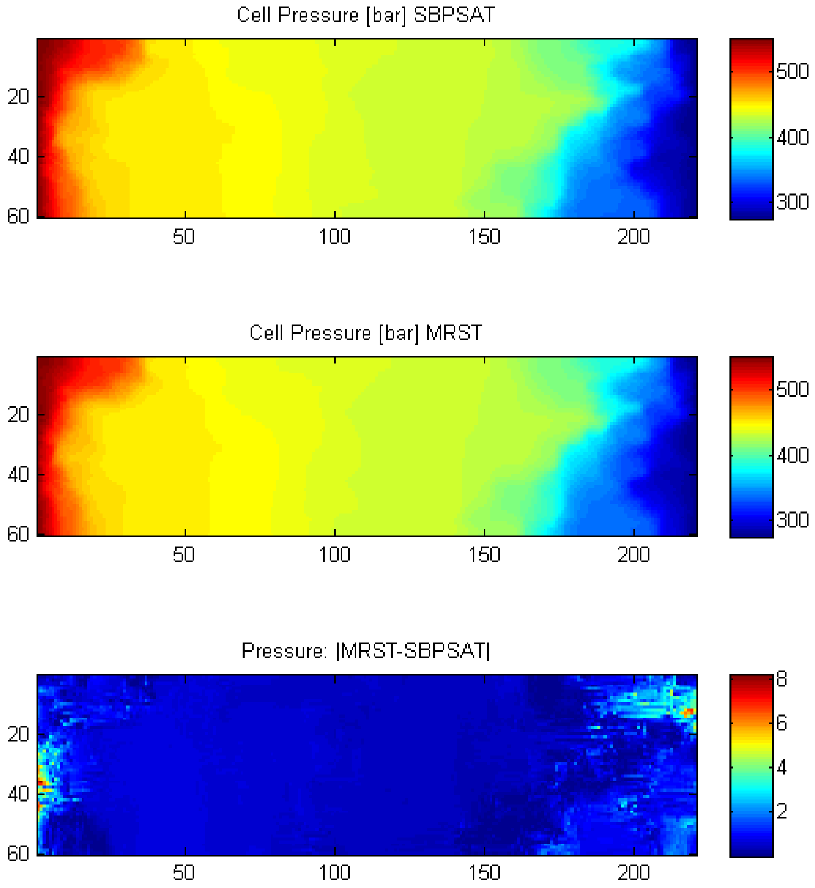

Figure 4.22: Final pressure distribution of layer 4.

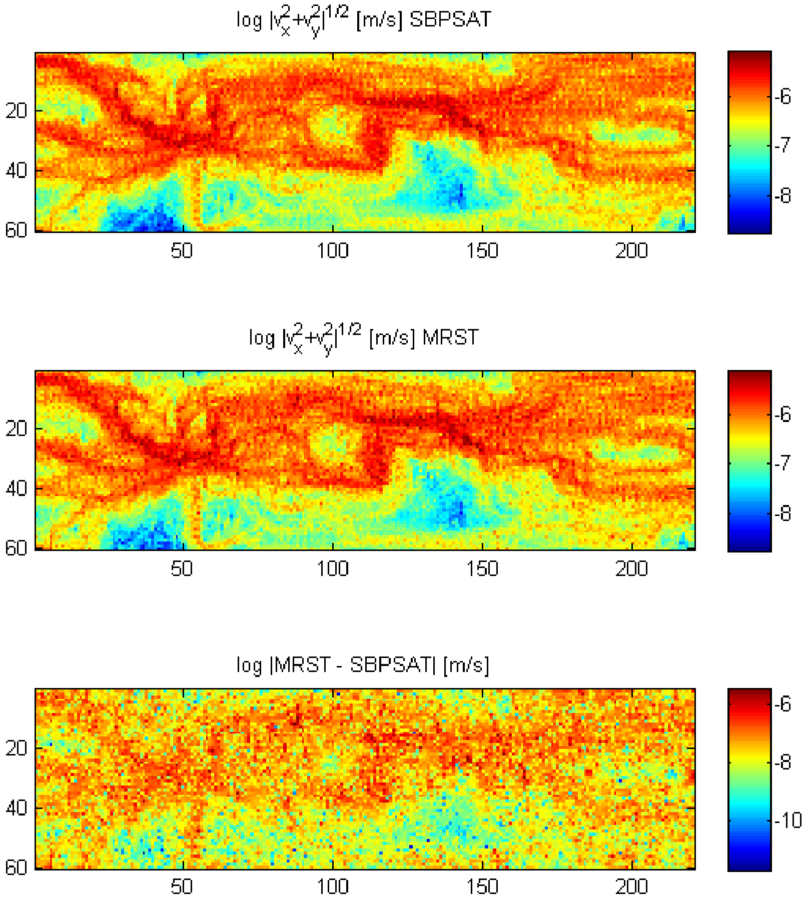

Figure 4.23: The Darcy velocity of layer 4.

The difference between the two solutions (Figure 4.22) is not severe (at maximum ~ 8 bar). There are however two regions, close to the boundary where the difference is somewhat larger as seen in Figure 4.22. Observing Figure 4.21, it is clear that the diverging areas do not correlate to the difference between K_x and K_y. Instead, the difference follows the pressure drop rim. At the rim corners, the SBP-SAT solution predicts a more dramatic pressure drop, most likely due to leakage. This difference is not as apparent in Figure 4.23, but another issue is noted here. The solutions differ in their predictions of the boundary Darcy velocity, which as previously explained is because calculating the fluxes in nodal values is not an optimal approach.

The next example layer is layer 53, which is considered one of the deeper and slightly 'tougher' layers. The K coefficients are shown in Figure 4.24. The pressure and Darcy velocity are shown in Figure 4.25 and Figure 4.26 respectively

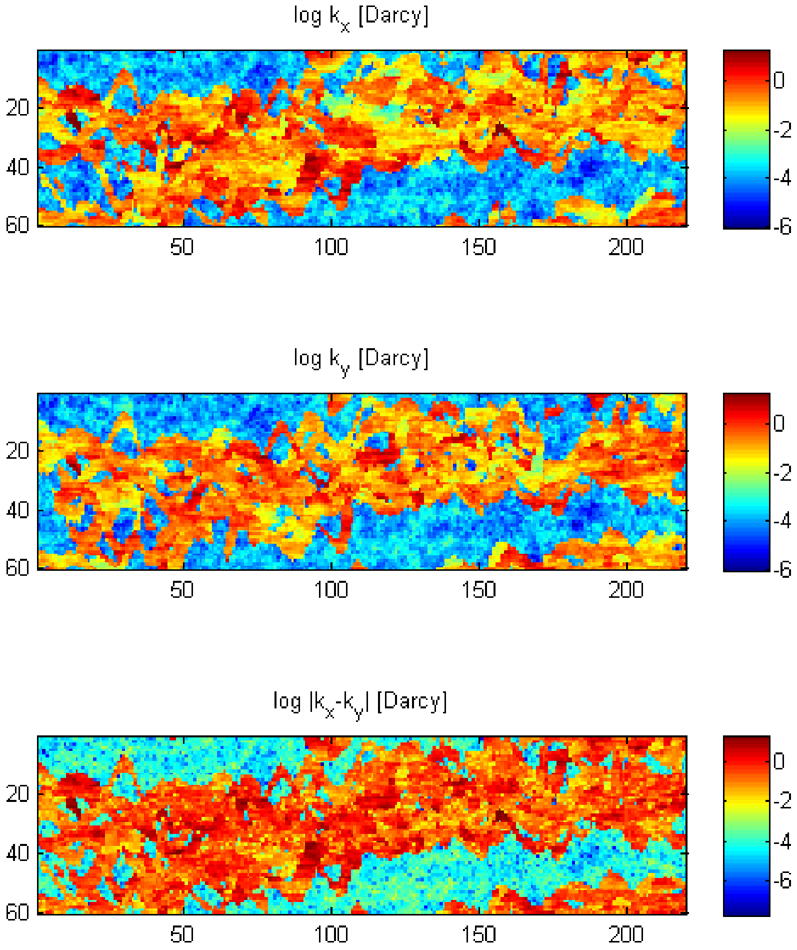

Figure 4.24: The K coefficients as well as the difference between them for layer 53.

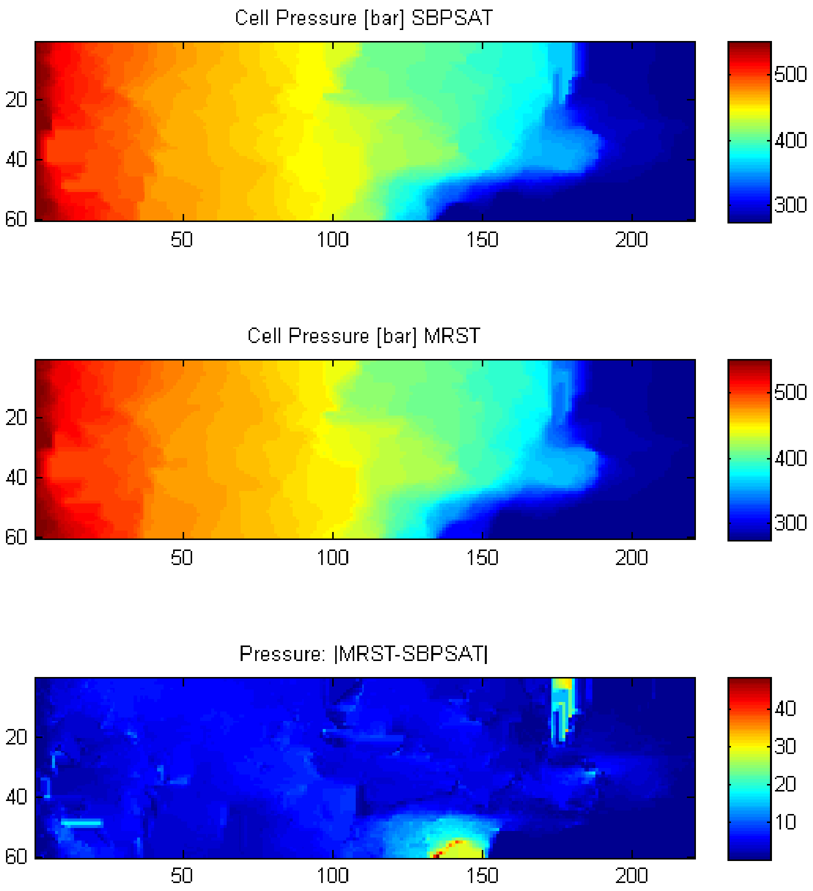

Figure 4.25: Final pressure distribution of layer 53.

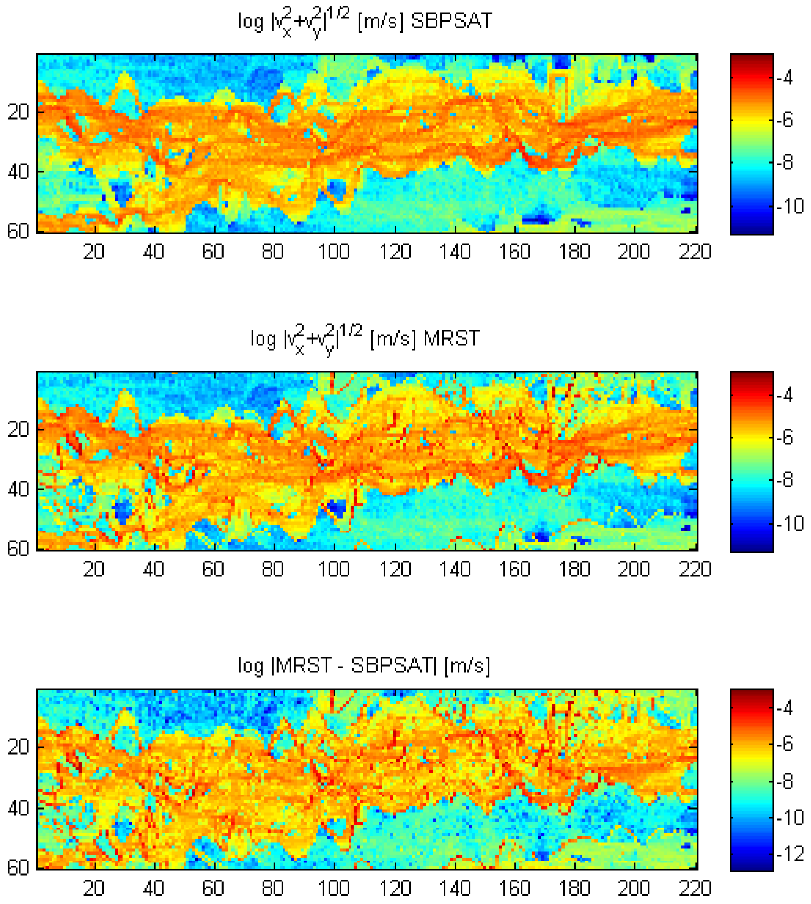

Figure 4.26: The Darcy velocity of layer 53.

There are two notable region differences between the solutions in Figure 4.25. The differences, located at $y = 0$ and $y = 60$ occurs clearly because of leakage as seen in Figure 4.24. The differences are also reflected in Figure 4.26, where the SBP-SAT method predicts a stronger flow at the 'difference' regions.

It should be noted, despite presenting layers with large differences in this section, that for most layers the issues in the pressure and Darcy velocity of applying SBP-SAT on discontinuous media was not as apparent as exemplified here.

Chapter 5

Analysis of Numerical Results

In the previous sections it was shown that the SBP-SAT method using narrow band operators does not result in optimal pressure distributions. Furthermore, it was pointed out that leakage was the main problem and that this is a problem in 2D despite refinement. This was finally shown to also be a problem in real world data. This result seemingly contrast section 3.3 where it is proved that the SBP-SAT solution should be convergent. This apparent inconsistency originates from that the convergence proof is built upon the assumption of continuity. The finite difference (and subsequently SBP) method uses Taylor expansions at its core. The Taylor expansion assumes that the approximated function is sufficiently smooth (the higher the accuracy, the higher function derivatives needs to be continuous). This works perfectly for analytically smooth cases, but the data drawn from real world situations is seldom continuous, such as the data (K coefficients) used in the previous sections. Just as it is not given that the true pressure distribution has enough continuous derivative's for higher order Taylor expansions. Typically the derivatives of the true solution will not be continuous over discontinuities. Traditionally the discontinuity problem is fixed with the introduction of interfaces (mimicking the process of solving the problem analytically), but this is unfeasible here, see Section 6.1. Hence one needs both continuity in the coefficients and the solution to guarantee that the discrete solution will converge toward the real solution. The lack of continuity in the posed problems therefore offers a theoretical explanation to why the SBP-SAT method theory was not applicable in a reservoir setting.

The MRST hybrid solver (which is the solver used in this thesis) takes a different approach. It uses a modified FEM where the equations force the flux across the cell boundaries to be continuous and does not require an assumption of continuous K coefficients. The selected method in MRST convergences despite a discontinuous domain. The fundamental difference between MRST and SBP-SAT becomes more apparent if comparing the flux coefficients (transmissibility) of a second order SBP operator with MRST. The coefficients are obtained by seeking the flux over cell boundaries, which in 1D this can be written as

$$F = T_{left}(p_{i-1} - p_i) + T_{right}(p_i - p_{i+1}), \tag{5.1}$$

where F is the total flux in or out of a cell and the T's are the transmissilility constants. MRST uses harmonic averaging of the K coefficients [4] and after rearranging the SBP terms, the corresponding SBP transmissibility constant is a regular averaging of the K coefficients. This mathematical difference of transmissibility constant further explains observed leakage; averaging K coefficients where one of the coefficients is very small (granite region) will allow a high flux to pass over the cell boundary while a harmonic average allows almost no flux. As an example consider a situation in 1D with three nodes of K values 1000, 1 and 1000 (a solid obstacle),

SBP-SAT:

$$\begin{align} F &= \frac{k_{i-1} + k_i}{2}(p_{i-1} - p_i) + \frac{k_{i+1} + k_i}{2}(p_i - p_{i+1}) \tag{5.2} \\ &= 500.5(p_{i-1} - p_i) + 500.5(p_i - p_{i+1}). \tag{5.3} \end{align}$$

MRST:

$$\begin{align} F &= \frac{k_{i-1}k_i}{k_{i-1} + k_i}(p_{i-1} - p_i) + \frac{k_{i+1}k_i}{k_{i+1} + k_i}(p_i - p_{i+1}) \tag{5.4} \\ &= 0.9990(p_{i-1} - p_i) + 0.9990(p_i - p_{i+1}). \tag{5.5} \end{align}$$

Clearly the SBP-SAT method allows a much higher flux to pass through the cell as compared to the MRST for the same pressure drop (and in this case it is undesirable to have high flux as noted in Section 4.3).

Chapter 6

Conclusion and Future Directions

6.1 Discussion of Future Directions

While fully functional in a smooth environment, a number of issues have been mentioned in the thesis regarding using narrow banded SBP-SAT operators on a discontinuous Poisson's equation. The main issue is the leakage which can cause undesirable solutions on 2D grids. There are numerous hypothetical ways to improve the SBP-SAT method such that the issue is eliminated (several have been mentioned earlier in the thesis). In this section, three suggestions are explored, which, to successfully remedy the issue in a useful 'SBP' manner, have a few requirements. The method should converge to the true solution, the method should be computationally efficient and the method should preferrable, given the right problem, allow for higher order convergence.

The first suggested fix is to use interfaces wherever there are discontinuities which is the traditional way of dealing with a discontinuity. At large it means solving different equation systems on different sufficiently continuous regions and then couple the regions with interfaces. This would guarantee convergence and also allow for higher order convergence. The problem is however that the final equations are likely to be very complex. One must distinguish between gradual changes and sudden changes in permeability and also account for that the discontinuities can run across the domain in arbitrary ways. Hence this fix requires some pre-processing, which can be potentially inefficient. However, although unlikely, it is possible that the cost of preprocessing is less than the gain of higher order convergence (e.g. for distinct channals). Hence this method is plausible but typically more suited for simulating a small local area with high resolution.

The second suggestion is to refine the grid with the application of smoothing. Hence, instead of using nearest neighbor refinement, one could use a Gaussian slope. This results in smoothly varying K coefficients, which means the SBP operators will converge on the 'true' solution. The issue in this approach is that an assumptions about the solution has to be made. Essentially the refinement is guessing what the data is beyond the actual resolution. Furthermore, additional grid refinement can mean a lot of extra work. One could take it one step further and try to localize the refinement around steep gradients and other problematic regions, but this step also requires extra computational work.

A third solution is to try and modify the SBP operators to use harmonic averaging combined with the SAT penalty terms on the boundary. This approach has the benefit (if successful) that it will converge to the true solution and will not generate complex equations or extra work. The main issue is that it might be difficult to produce high order operators and that this discretization in some sense means moving away from an SBP solution.

One concern about using SBP-SAT for solving the Poisson's equation is whether the inaccuracy in data is bigger than the gain of successfully applying a high order SBP-SAT method. There are several errors when measuring data for oil reservoirs. Except for grid resolution limits, the biggest error might be the measurement of the permeability (K coefficients). If these are measured to an accuracy of three decimals it might be futile to seek higher order solutions for the purpose of oil reservoir simulations. The reason is that there is almost always a region (for sufficiently coarse resolution) when using a higher order method will produce larger errors than its lower order equivalent. This is of course hard to deduce on beforehand.

A final remark is that despite the issues mentioned the SBP-SAT method works very well as long as the coefficients and solution are sufficiently smooth. Hence although it is not yet perfected for underground Darcy flow, it will perform very well wherever the Poisson's equation with variable coefficients are used on problems with the right settings.

6.2 Conclusions

Applying narrow banded SBP-SAT operators to the Poisson's equation with variable coefficients has been shown to generate results of varying accuracy. The method was proved to converge analytically and later verified by testing it on design problems given sufficiently smooth K coefficients. It was further shown that the boundary derivate can be penalized into converging substantially faster than for the unpenalized method. However, the SBP-SAT method was also applied to more realistic scenarios with strongly discontinuous coefficients and it was clear that the method will, in certain regions, not converge to a desired and expected solution. The cause of the (local) convergence issues was determined to be a combination of flux leakage (high flux in low flux expected regions) and grid refinement method. The method allows certain permeability structures (specifically K coefficient diagonals) in 2D to persistently give incorrect results (flux leakages) during grid refinement. It was pointed out that there is room for interpretation for what an analytically correct solution is. However, flux leakage is undesirable and thus SBP-SAT without some type of modification is not optimal for the purpose of simulating large and strongly density varying oil reservoirs. To remedy the problems, a few future modifications were suggested. The suggestions treated interfaces, grid refinement techniques and structure of SBP operators. Considering all of the above, an unaltered narrow banded SBP-SAT method version should be used on smooth analytical problems but so far avoided on discontinuous problems.

Appendix A

The 2D Problem

The proof mimcs the 1D outline and begins yet again with obtaining a norm bound on the solution in the continuous space, then proceed to seek existance of a smiliar discrete bound in the discrete space and finally use this to show convergence.

The domain is discretized on an $N \times M$ equidistant grid which is defined as

$$\begin{aligned} x_i &= (i-1)h_x, \quad i = 1, 2, \ldots, N, \quad h_x = \tfrac{1}{N-1} \\ y_j &= (j-1)h_y, \quad j = 1, 2, \ldots, M, \quad h_y = \tfrac{1}{M-1} \end{aligned}$$

Each point is associated with the solution at its coordinates. The full solution is represented by a vector as $v = \Big[\bar{v}_1, \bar{v}_2, \ldots, \bar{v}_N\Big]^T$ where $\bar{v}_j = \Big[v_{1j}, v_{2j}, \ldots, v_{nj}\Big]^T$.

The kronecker product is defined as follows:

$$C \otimes D = \begin{pmatrix} c_{1,1}D & \ldots & c_{1,q}D \\ \vdots & & \vdots \\ c_{p,1}D & \ldots & c_{p,q}D \end{pmatrix} \tag{A.1}$$

Introducing some recurring symbols:

$$e_{i,N} = \Big[0, \ldots, 0, 1, 0, \ldots, 0\Big]^T$$

where the vector is N values long and have a one at the ith position.

$$
\begin{aligned}
&D_{1x} = I_M \otimes D_1 && D_{1y} = D_1 \otimes I_N \\
&H_x = I_M \otimes H && H_y = H \otimes I_N \\
&e_{We} = I_M \otimes e_{1,N} && e_{So} = e_{1,M} \otimes I_N \\
&e_{Ea} = I_M \otimes e_{N,N} && e_{No} = e_{M,M} \otimes I_N \\
&S_{We} = I_M \otimes S_1 && S_{So} = S_1 \otimes I_N \\
&S_{Ea} = I_M \otimes S_N && S_{No} = S_M \otimes I_N \\
&\bar{H} = H_x H_y
\end{aligned}
\tag{A.2}
$$

$$K_{ij} = diag(k_{ij}(x_1, y_1) \dots k_{ij}(x_n, y_1) \dots k_{ij}(x_n, y_m)), \; ij \in \Big[1, 2\Big]$$

$$K = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix}$$

$$D_{2x}^{(k_{11})} = \begin{bmatrix} D_2^{(k_{11}(x,y_1))} & 0 & \dots & 0 \\ 0 & D_2^{(k_{11}(x,y_2))} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & D_2^{(k_{11}(x,y_m))} \end{bmatrix}$$

where $D_2^{(k_{11}(x,y_j))}$ is the standard D2 operator with variable coefficients for $x = [x_1, x_2 \dots x_n]$ at $y = y_j$.

Just as in the 1D case it is also assumed that the boundary is homogenuous since it will simplify the notation.

A.1 Continuous Inequality

Theorem A.1.1. *For a continuous solution $u(x, y) \in \Re \quad x \in [0, 1], y \in [0, 1]$ to the 2D Poisson's equation (2.12) with homogeneous boundary conditions of which at least one is a Dirichlet condition there is a constant $C \in \Re > 0$ such that*

$$\|\nabla u\|_{L^2(\Omega)} \leq C \|f\|_{L^2(\Omega)} \tag{A.3}$$

Proof. To prove the theorem, the energy method will be used. Multiply equation (2.12) with $u^\dagger$ and integrate over the domain,

$$-\,(u, \boldsymbol{\nabla}\cdot(K\boldsymbol{\nabla}u)) = (u, f). \tag{A.4}$$

Integrate by parts, and assuming homogeneous boundary conditions gives

$$\begin{multline}(u_x, k(x,y)_{11}u_x) + (u_x, k(x,y)_{12}u_y) + \\ (u_y, k(x,y)_{21}u_x) + (u_y, k(x,y)_{22}u_y) = (u, f). \tag{A.5}\end{multline}$$

Introducing

$$\bar{u} = \begin{pmatrix} u_x \\ u_y \end{pmatrix}, \tag{A.6}$$

the equation simplifies to

$$(\bar{u}, K\bar{u}) = (u, f). \tag{A.7}$$

Since $K > 0$ for each (x,y) pair it can be shown after some algebra that

$$\bar{u}^T K\bar{u} = \epsilon u_x^2 + \epsilon u_y^2 + \frac{(k_{11}-k_{12})^2 + (k_{22}-k_{12})^2}{k_{11}+k_{22}} \qquad > \epsilon(|u_x|^2 + |u_y|^2) \tag{A.8}$$

where $\epsilon = \frac{k_{11}k_{22}-k_{12}k_{21}}{k_{11}+k_{22}} > 0$. This means that

$$0 \le \epsilon_T(\|u_x\|^2 + \|u_y\|^2) < (\bar{u}, K\bar{u}) \le \|u\|\|f\|. \tag{A.9}$$

In the last step the Cauchy-Schwartz inequality (2.4) was used. Squaring the Poincaré inequality 2.5 one obtains

$$\|u\|^2 \le C_p^2\|\boldsymbol{\nabla}u\|^2 = C_p^2(\|u_x\| + \|u_y\|)^2 \le C_p^2(\|u_x\|^2 + \|u_y\|^2). \tag{A.10}$$

Taking the square root and using this result gives

$$\epsilon_T(\|u_x\|^2 + \|u_y\|^2) < C_p\sqrt{\|u_x\|^2 + \|u_y\|^2}\|f\|. \tag{A.11}$$

Dividing both sides with the norm and rewriting the right hand side one finally gets

$$\sqrt{\|u_x\|^2 + \|u_y\|^2} = \|\boldsymbol{\nabla} u\| < C_p \epsilon_T^{-1} \|f\| = C\|f\|. \tag{A.12}$$

□

To complete the convergence proof in 2D a discrete version of Poincaré similar to the 1D discrete Poincaré (3.5) is needed. The 2D discrete Poincaré inequality was numerically tested to be similar to the 1D inequality with the notable exception of the domain is now the unit square. Hence the same C as in the 1D case is used in 2D.

A.2 Discrete Inequality

To show the discrete inequality corresponding to the continuous equation (A.1.1) the following theorem is proved

Theorem A.2.1. *The SBP-SAT approximation with of the Poisson's equation* (2.12) *with weak homogeneous Dirichlet boundary conditions and an injected condition on one of the boundaries*

$$I \left\{ \begin{array}{ll} -(D_{2x}^{(k_{11})} + D_{2y}^{(k_{22})} + D_x K_{12} D_y + D_y K_{21} D_x)v & \\ +\tau_E H_x^{-1}[(K_{11}S_x)^T + (K_{12}D_y)^T]_E(e_{Ea}^T v - g_E) & \\ +\tau_W H_x^{-1}[(K_{11}S_x)^T + (K_{12}D_y)^T]_W(e_{We}^T v - g_W) & \\ +\tau_N H_y^{-1}[(K_{22}S_y)^T + (K_{21}D_x)^T]_N(e_{No}^T v - g_N) & \\ +\tau_S H_y^{-1}[(K_{22}S_y)^T + (K_{21}D_x)^T]_S(e_{So}^T v - g_S) & = f_v \\ +\sigma_E K_{11} e_{Ea}(e_{Ea}^T v - g_E) & \\ +\sigma_W K_{11} e_{We}(e_{We}^T v - g_W) & \\ +\sigma_N K_{22} e_{No}(e_{No}^T v - g_N) & \\ +\sigma_S K_{22} e_{So}(e_{So}^T v - g_S) & \end{array} \right\} \tag{A.13}$$

under the hypothesis (3.5) *also satisfies the inequality*

$$\|D_1 v\|_H \leq C\|f_v\|_H. \tag{A.14}$$

Here $\tau_E = -1, \tau_W = 1, \tau_N = -1, \tau_S = 1$, $\sigma_E \geq \frac{k_E \widetilde{k_E^p}}{\alpha h}$, $\sigma_W \geq \frac{k_W \widetilde{k_W^p}}{\alpha h}$, $\sigma_N \geq \frac{k_N \widetilde{k_N^p}}{\alpha h}$, $\sigma_S \geq \frac{k_S \widetilde{k_S^p}}{\alpha h}$ *and* α *is a positive constant dependent on the SBP order. The* *operators are defined in appendix B and the $I\{\}$ signifies that one of the four boundaries is injected.*

Note. The 'E', 'W', 'N' and 'S' subscripts at the square brackets, e.g. $[\cdot]_E$ indicates that only columns corresponding to East, West, North and South boundaries are kept.

Proof. Here we keep all the weak conditions, but note that at least one of the boundaries is injected. Assuming homogeneous Dirichlet boundary conditions, the discrete approximation (A.13) can be rewritten as

$$I\left\{\begin{array}{ll} -(D_{2x}^{(k_{11})}+D_{2y}^{(k_{22})}+D_xK_{12}D_y+D_yK_{21}D_x)v & \\ +\tau_E H_x^{-1}[(K_{11}S_{Ea})^T+(K_{12}D_y)^T]_E e_{Ea}^T v & \\ +\tau_W H_x^{-1}[(K_{11}S_{We})^T+(K_{12}D_y)^T]_W e_{We}^T v & \\ +\tau_N H_y^{-1}[(K_{22}S_{No})^T+(K_{21}D_x)^T]_N e_{No}^T v & \\ +\tau_S H_y^{-1}[(K_{22}S_{So})^T+(K_{21}D_x)^T]_S e_{So}^T v & = f_v. \\ +\sigma_E K_{11}e_{Ea}e_{Ea}^T v & \\ +\sigma_W K_{11}e_{We}e_{We}^T v & \\ +\sigma_N K_{22}e_{No}e_{No}^T v & \\ +\sigma_S K_{22}e_{So}e_{So}^T v & \end{array}\right\} \tag{A.15}$$

Dropping, the $I\{\cdot\}$ for convenience and multiplying by $v^TH_xH_y$, setting $\begin{bmatrix}\tau_E & \tau_W & \tau_N & \tau_S\end{bmatrix} = \begin{bmatrix}1 & -1 & 1 & -1\end{bmatrix}$ and using that $D_{2x}^{(K_{11})} = H_x^{-1}(-D_x^TH_xK_{11}D_x - R_x^{(K_{11})} - K_{11}S_{We} + K_{11}S_{Ea})$ and $D_{2y}^{(K_{22})} = H_y^{-1}(-D_y^TH_yK_{22}D_y - R_y^{(K_{22})} - K_{22}S_{So} + K_{22}S_{No})$ we obtain

$$Ds + Bs = v^T\bar{H}f_v, \tag{A.16}$$

where Ds and Bs are defined as follows

$$Ds = \begin{array}{l} -v^TH_y(-D_x^TH_xK_{11}D_x - R_x^{(K_{11})} - K_{11}S_{We} + K_{11}S_{Ea})v \\ -v^TH_x(-D_y^TH_yK_{22}D_y - R_y^{(K_{22})} - K_{22}S_{So} + K_{22}S_{No})v \\ -v^TH_xH_y(D_xK_{12}D_y + D_yK_{21}D_x)v \end{array} \tag{A.17}$$

$$Bs = \begin{aligned} &+v^T H_y[(K_{11}S_{Ea})^T + (K_{12}D_y)^T]_E e_{Ea}^T v \\ &-v^T H_y[(K_{11}S_{We})^T + (K_{12}D_y)^T]_W e_{We}^T v \\ &+v^T H_x[(K_{22}S_{No})^T + (K_{21}D_x)^T]_N e_{No}^T v \\ &-v^T H_x[(K_{22}S_{So})^T + (K_{21}D_x)^T]_S e_{So}^T v \\ &+\sigma_E h_x v K_{11} e_{Ea} e_{Ea}^T v \\ &+\sigma_W h_x v K_{11} e_{We} e_{We}^T v \\ &+\sigma_N h_y v K_{22} e_{No} e_{No}^T v \\ &+\sigma_S h_y v K_{22} e_{So} e_{So}^T v \end{aligned} \tag{A.18}$$

Regrouping the penalty terms gives, with newly defined Ds and Bs,

$$Ds + Bs = v^T \bar{H} f_v, \tag{A.19}$$

where Ds and Bs are defined as follows

$$Ds = \begin{aligned} &-v^T H_y(-D_x^T H_x K_{11} D_x - R_x^{(K_{11})})v \\ &-v^T H_x(-D_y^T H_y K_{22} D_y - R_y^{(K_{22})})v \\ &-v^T H_x H_y(D_x K_{12} D_y + D_y K_{21} D_x)v \\ &+v^T H_y[(K_{12}D_y)^T]_E e_{Ea}^T v - H_y[(K_{12}D_y)^T]_W e_{We}^T v \\ &+v^T H_x[(K_{21}D_x)^T]_N e_{No}^T v - H_x[(K_{21}D_x)^T]_S e_{So}^T v \end{aligned} \tag{A.20}$$

$$Bs = \begin{aligned} &+v^T H_y[(K_{11}S_{Ea})^T]_E e_{Ea}^T v \\ &-v^T H_y[(K_{11}S_{We})^T]_W e_{We}^T v \\ &+v^T H_x[(K_{22}S_{No})^T]_N e_{No}^T v \\ &-v^T H_x[(K_{22}S_{So})^T]_S e_{So}^T v \\ &+\sigma_E h_x v K_{11} e_{Ea} e_{Ea}^T v \\ &+\sigma_W h_x v K_{11} e_{We} e_{We}^T v \\ &+\sigma_N h_y v K_{22} e_{No} e_{No}^T v \\ &+\sigma_S h_y v K_{22} e_{So} e_{So}^T v \end{aligned} \tag{A.21}$$

Rewriting the equation and splitting up the mixed derivate terms by setting $D_x = H_x^{-1}(Q_x - 1/2e_{We}^T e_{We} + 1/2e_{Ea}^T e_{Ea})$ and $D_y = H_y^{-1}(Q_y - 1/2e_{So}^T e_{So} + 1/2e_{No}^T e_{No})$ as well as noting that $Q = -Q^T$ one can obtain (after some algebra) that

$$Bs + \begin{array}{l} +v^T D_x^T \bar{H} K_{11} D_x v + v^T H_y R_x^{(K_{11})} v \\ +v^T D_y^T \bar{H} K_{22} D_y v + v^T H_x R_y^{(K_{22})} v \\ +v^T D_x^T \bar{H} K_{12} D_y v + v^T D_y^T \bar{H} K_{21} D_x v \end{array} = v^T \bar{H} f_v. \tag{A.22}$$

If setting Neumann or injected boundary, $Bs = 0$ and it is enough to note that $H_y R_x^{(K_{11})} \geq 0$ and $H_x R_y^{(K_{22})} \geq 0$. However to ensure that $Bs \geq 0$ for weak Dirichlet, one needs to borrow an extra term by splitting up the Ds terms exactly as done in section 3.2 and shown in equation (3.14). Introducing

$$\bar{v} = \begin{pmatrix} D_x v \\ D_y v \end{pmatrix}, \tag{A.23}$$

and completing the same steps following equation (3.14), equation (A.19) simplifies into

$$\bar{v}^T C \bar{H} K \bar{v} \leq v^T \bar{H} f_v. \tag{A.24}$$

Here C is defined as

$$C = \begin{pmatrix} C^{11} & 0 \\ 0 & C^{22}, \end{pmatrix} \tag{A.25}$$

where $0 < c_{ii}^{11}, c_{ii}^{22} \leq 1$. Proceeding, completely analogous to the continuous case, equation (A.8) and the following equations (with the modification of keeping $\bar{H}$, which only changes the constant) can be rewritten as

$$0 \leq \epsilon_T (\|D_x v\|_{\bar{H}}^2 + \|D_y v\|_{\bar{H}}^2) < \bar{v}^T \bar{H} K \bar{v} \leq v^T \bar{H} f_v \leq \|v\|_{\bar{H}} \|f_v\|_{\bar{H}}, \tag{A.26}$$

where the Cauchy-Schwarz inequality 2.4 was used in the last step and ϵ_T is a constant. Using the discrete version of the Poincaré's inequality 2.5 (since one condition is injected) one obtains

$$\epsilon_T (\|D_x v\|_{\bar{H}}^2 + \|D_y v\|_{\bar{H}}^2) < C_p \sqrt{\|D_x v\|_{\bar{H}}^2 + \|D_y v\|_{\bar{H}}^2} \|f_v\|_{\bar{H}}. \tag{A.27}$$

Dividing both sides with $\sqrt{\|D_x v\|_{\bar{H}}^2 + \|D_y v\|_{\bar{H}}^2}$ and moving the constant gives

$$\sqrt{\|D_x v\|_{\bar{H}}^2 + \|D_y v\|_{\bar{H}}^2} = \|\boldsymbol{\nabla} v\|_{\bar{H}} < C_p \epsilon_T^{-1} \|f_v\|_{\bar{H}} = C\|f_v\|_{\bar{H}}, \tag{A.28}$$

which completely mimics theorem Theorem A.1.1 and thus completes the proof. □

Note. In the above proof a weakly imposed Dirichlet boundary was used. Similar proofs for robin, Neumann and injected Dirichlet are not shown here, but can be constructed by proceeding in the same manner as above.

A.3 Convergence

Convergence for the proposed method is obtained by seeking the difference between the analytical solution u, applied to the discrete approximation equation (2.2) and the discrete solution v. This difference is shown to go to zero as the mesh space h goes to zero.

Theorem A.3.1. *If approximating Poisson's equation* (2.12) *using the narrow banded SBP-SAT method with weak homogeneous Dirichlet conditions exactly as done in Theorem A.2.1, the discrete solution vector, v, converges to the true solution, u, with at least h^p, where $2p$ is the order of SBP operator and the K coefficients are assumed to be sufficiently smooth.*

Proof. We start from the discrete SBP-SAT approximation (A.13) with

$$\begin{bmatrix}\tau_E & \tau_W & \tau_N & \tau_S\end{bmatrix} = \begin{bmatrix}1 & -1 & 1 & -1\end{bmatrix},$$

which gives

$$I\left\{\begin{array}{l}-(D_{2x}^{(k_{11})}+D_{2y}^{(k_{22})}+D_xK_{12}D_y+D_yK_{21}D_x)v\\+H_x^{-1}[(K_{11}S_x)^T+(K_{12}D_y)^T]_E(e_{Ea}^Tv-g_E)\\-H_x^{-1}[(K_{11}S_x)^T+(K_{12}D_y)^T]_W(e_{We}^Tv-g_W)\\+H_y^{-1}[(K_{22}S_y)^T+(K_{21}D_x)^T]_N(e_{No}^Tv-g_N)\\-H_y^{-1}[(K_{22}S_y)^T+(K_{21}D_x)^T]_S(e_{So}^Tv-g_S)\qquad=f_v\\+\sigma_{E,W}K_{11}e_{Ea}(e_{Ea}^Tv-g_E)\\+\sigma_{E,W}K_{11}e_{We}(e_{We}^Tv-g_W)\\+\sigma_{N,S}K_{22}e_{No}(e_{No}^Tv-g_N)\\+\sigma_{N,S}K_{22}e_{So}(e_{So}^Tv-g_S)\end{array}\right\}\tag{A.29}$$

Dropping the $I\{\cdot\}$ for convenience and inserting u as the analytical solution to Poisson's equation with Dirichlet boundary conditions (2.12) into this equation yields

$$-(D_{2x}^{(k_{11})}+D_{2y}^{(k_{22})}+D_xK_{12}D_y+D_yK_{21}D_x)u=f_v+T_p(h)+Inj\tag{A.30}$$

Where Inj is the RHS injected terms and $T_p(h)$ is the truncation error, given by

$$T_p=C\left[\begin{array}{ccc}T_1&\dots&T_n\end{array}\right]^T\tag{A.31}$$

and

$$T_i=\left[\begin{array}{ccccccccccc}h^p&...&h^p&h^{2p}&...&h^{2p}&...&h^p&...&h^p\end{array}\right]^T.\tag{A.32}$$

and C is a constant. Proceeding by defining the error ϵ as $u-v$ and subtracting equation (A.30) from equation (A.29) the following equation is obtained

$$\begin{aligned}
&-(D_{2x}^{(k_{11})} + D_{2y}^{(k_{22})} + D_x K_{12} D_y + D_y K_{21} D_x)\epsilon \\
&+H_x^{-1}[(K_{11}S_x)^T + (K_{12}D_y)^T]_E \epsilon_{Ea} \\
&-H_x^{-1}[(K_{11}S_x)^T + (K_{12}D_y)^T]_W \epsilon_{We} \\
&+H_y^{-1}[(K_{22}S_y)^T + (K_{21}D_x)^T]_N \epsilon_{No} \\
&-H_y^{-1}[(K_{22}S_y)^T + (K_{21}D_x)^T]_S \epsilon_{So} \qquad\qquad = T_p(h) \\
&+\sigma_{E,W} K_{11} e_{Ea} \epsilon_{Ea} \\
&+\sigma_{E,W} K_{11} e_{We} \epsilon_{We} \\
&+\sigma_{N,S} K_{22} e_{No} \epsilon_{No} \\
&+\sigma_{N,S} K_{22} e_{So} \epsilon_{So}
\end{aligned} \tag{A.33}$$

Next we proceed with the same steps as in the proof for the 2D stability, thus utilizing Theorem A.2.1. The difference is merely a matter of substituting the vector v with ϵ and the right hand side of equation (A.13), f_v with $T_p(h)$. Completing the steps it is obtained that

$$\|\boldsymbol{\nabla}\epsilon\|_{\bar{H}} \leq C\|T_p(h)\|_H. \tag{A.34}$$

This can be further simplified using a discrete version of the Poincaré's inequality 2.5 into

$$\|\epsilon\|_H \leq C\|T_p(h)\|_H \sim h^p \tag{A.35}$$

At this point it is also clear that the above steps can also be applied on Neumann and injected boundary conditions with identical convergence.

□

Note. The estimated convergence is not sharp for the boundary conditions. This means that the true convergence should be higher than what was obtained here.

A.4 Positive Definiteness

The 2D positive defininteness discussion is at large a complete copy of the 1D positive definintenes discussion. The only difference is noting that the 2D matrices consists of copies of 1D matrices which has already been shown to be positive definite. This section is also omitted.

Appendix B

Operators

Here we present the compatible second-, fourth- sixth- and eighth-order accurate narrow-diagonal SBP operators, defined in [2]. The first-derivative SBP operators are given by $H^{-1}Q$, where Q can be found in [15]. The second-derivative operator, $D_2^{(b)} = H^{-1}(-M^{(b)} + \bar{B}S)$, approximate $\partial/\partial x\,(b\,\partial/\partial x)$, where $b(x) > 0$, using a pth-order accurate narrow-stencil. $M^{(b)}$ is symmetric and positive semi-definite, S approximates the first-derivative operator at the boundaries and $\bar{B} = diag\,(-b_0, 0\ldots, 0, b_N)$.

B.1 Second-order accurate

For the second-order case we have

$$S = \frac{1}{h}\begin{bmatrix} -\frac{3}{2} & 2 & -\frac{1}{2} & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & \frac{1}{2} & -2 & \frac{3}{2} \end{bmatrix}, \quad H = \frac{h}{2}\begin{bmatrix} 1 & & & & \\ & 2 & & & \\ & & \ddots & & \\ & & & 2 & \\ & & & & 1 \end{bmatrix}.$$

The operators $D_2^{(2)}$ and $C_2^{(2)}$ are given by:

$$D_2^{(2)} = \begin{bmatrix} 1 & -2 & 1 & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 & \\ & & & 1 & -2 & 1 \\ & & & 1 & -2 & 1 \end{bmatrix}, \quad C_2^{(2)} = \begin{bmatrix} 0 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & 0 \end{bmatrix}.$$

The $B_2^{(2)}$ matrix is given by $(B_2^{(2)})_{i,i} = b_i$.

The left boundary closure of $M^{(b)}$ (given by a 3×3 matrix) is given by

$$\begin{bmatrix} \frac{1}{2}\, b_1 + \frac{1}{2}\, b_2 & -\frac{1}{2}\, b_1 - \frac{1}{2}\, b_2 & 0 \\ -\frac{1}{2}\, b_1 - \frac{1}{2}\, b_2 & \frac{1}{2}\, b_1 + b_2 + \frac{1}{2}\, b_3 & -\frac{1}{2}\, b_2 - \frac{1}{2}\, b_3 \\ 0 & -\frac{1}{2}\, b_2 - \frac{1}{2}\, b_3 & \frac{1}{2}\, b_2 + b_3 + \frac{1}{2}\, b_4 \end{bmatrix}.$$

The corresponding right boundary closure is given by replacing $b_i \rightarrow b_{N+1-i}$ for $i = 1..4$ followed by a permutation of both rows and columns.

The interior stencil of $M^{(b)}$ at row i is given by $(i = 4 \ldots N-3)$:

$$\begin{aligned} m_{i,i-1} &= -\frac{1}{2}\, b_{i-1} - \frac{1}{2}\, b_i \\ m_{i,i} &= \frac{1}{2}\, b_{i-1} + b_i + \frac{1}{2}\, b_{i+1} \\ m_{i,i+1} &= -\frac{1}{2}\, b_i - \frac{1}{2}\, b_{i+1} \end{aligned}$$

B.2 Fourth-order accurate

The third-order accurate boundary derivative operator S is given by,

$$S = \frac{1}{h}\begin{bmatrix} -\frac{11}{6} & 3 & -\frac{3}{2} & \frac{1}{3} & & & \\ & 0 & & & & & \\ & & & \ddots & & & \\ & & & & & 0 & \\ & & & -\frac{1}{3} & \frac{3}{2} & -3 & \frac{11}{6} \end{bmatrix}.$$

The discrete norm H is given by

$$H = h\begin{bmatrix} \frac{17}{48} & & & & & \\ & \frac{59}{48} & & & & \\ & & \frac{43}{48} & & & \\ & & & \frac{49}{48} & & \\ & & & & 1 & \\ & & & & & \ddots \end{bmatrix}.$$

The difference operator $D_3^{(4)}$:

$$\begin{bmatrix} -1 & 3 & -3 & 1 & & & & & & \\ -1 & 3 & -3 & 1 & & & & & & \\ d3_1 & d3_2 & d3_3 & d3_4 & d3_5 & d3_6 & & & & \\ & & -1 & 3 & -3 & 1 & & & & \\ & & & \ddots & \ddots & \ddots & \ddots & & & \\ & & & & & -1 & 3 & -3 & 1 & \\ & & & & -d3_6 & -d3_5 & -d3_4 & -d3_3 & -d3_2 & -d3_1 \\ & & & & & & -1 & 3 & -3 & 1 \\ & & & & & & -1 & 3 & -3 & 1 \end{bmatrix},$$

where

$$\begin{array}{ll} d3_1 = -\frac{185893}{301051} & d3_4 = -\frac{36887526683}{54642863857} \\ d3_2 = \frac{79000249461}{54642863857} & d3_5 = \frac{26183621850}{54642863857} \\ d3_3 = -\frac{33235054191}{54642863857} & d4_6 = -\frac{4386}{181507} \end{array}.$$

The difference operator $D_4^{(4)}$:

$$\begin{bmatrix}
1 & -4 & 6 & -4 & 1 & & & & & & & & \\
1 & -4 & 6 & -4 & 1 & & & & & & & & \\
1 & -4 & 6 & -4 & 1 & & & & & & & & \\
 & 1 & -4 & 6 & -4 & 1 & & & & & & & \\
 & & \ddots & \ddots & \ddots & \ddots & \ddots & & & & & & \\
 & & & & 1 & -4 & 6 & -4 & 1 & & & & \\
 & & & & & 1 & -4 & 6 & -4 & 1 & & & \\
 & & & & & 1 & -4 & 6 & -4 & 1 & & & \\
 & & & & & 1 & -4 & 6 & -4 & 1 & & &
\end{bmatrix}.$$

The left boundary closure of the diagonal matrix $C_3^{(4)}$ is given by:

$$\begin{bmatrix} 0 & 0 & \frac{163928591571}{53268010936} & \frac{189284}{185893} & 1 & \cdots \end{bmatrix}.$$

The right boundary closure of the diagonal matrix $C_3^{(4)}$ is given by:

$$\begin{bmatrix} \cdots & 1 & \frac{189284}{185893} & 0 & \frac{163928591571}{53268010936} & 0 & 0 \end{bmatrix}.$$

The left boundary closure of the diagonal matrix $C_4^{(4)}$ is given by:

$$\begin{bmatrix} 0 & 0 & \frac{1644330}{301051} & \frac{156114}{181507} & 1 & \cdots \end{bmatrix}.$$

The corresponding right boundary closure is given by a permutation of both rows and columns.

The $B^{(4)}$ matrices are given by

$$\begin{aligned}
(B_3^{(4)})_{i,\,i} &= \tfrac{1}{2}(b_i + b_{i+1}) \\
(B_4^{(4)})_{i,\,i} &= b_i
\end{aligned} \quad .$$

The interior stencil of $-M^{(b)}$ at row i is given by $(i = 7 \ldots N-6)$:

$$\begin{aligned}
m_{i,\,i-2} &= \tfrac{1}{6}\, b_{i-1} - \tfrac{1}{8}\, b_{i-2} - \tfrac{1}{8}\, b_i \\
m_{i,\,i-1} &= \tfrac{1}{6}\, b_{i-2} + \tfrac{1}{6}\, b_{i+1} + \tfrac{1}{2}\, b_{i-1} + \tfrac{1}{2}\, b_i \\
m_{i,\,i} &= -\tfrac{1}{24}\, b_{i-2} - \tfrac{5}{6}\, b_{i-1} - \tfrac{5}{6}\, b_{i+1} - \tfrac{1}{24}\, b_{i+2} - \tfrac{3}{4}\, b_i \quad . \\
m_{i,\,i+1} &= \tfrac{1}{6}\, b_{i-1} + \tfrac{1}{6}\, b_{i+2} + \tfrac{1}{2}\, b_i + \tfrac{1}{2}\, b_{i+1} \\
m_{i,\,i+2} &= \tfrac{1}{6}\, b_{i+1} - \tfrac{1}{8}\, b_i - \tfrac{1}{8}\, b_{i+2}
\end{aligned}$$

The left boundary closure of $-M^{(b)}$ (given by a 6×6 matrix) is given by

$$m_{1,\,1} = \tfrac{12}{17}\, b_1 + \tfrac{59}{192}\, b_2 + \tfrac{27010400129}{345067064608}\, b_3 + \tfrac{69462376031}{2070402387648}\, b_4$$

$$m_{1,\,2} = -\tfrac{59}{68}\, b_1 - \tfrac{6025413881}{21126554976}\, b_3 - \tfrac{537416663}{7042184992}\, b_4$$

$$m_{1,\,3} = \tfrac{2}{17}\, b_1 - \tfrac{59}{192}\, b_2 + \tfrac{213318005}{16049630912}\, b_4 + \tfrac{2083938599}{8024815456}\, b_3$$

$$m_{1,\,4} = \tfrac{3}{68}\, b_1 - \tfrac{1244724001}{21126554976}\, b_3 + \tfrac{752806667}{21126554976}\, b_4$$

$$m_{1,\,5} = \tfrac{49579087}{10149031312}\, b_3 - \tfrac{49579087}{10149031312}\, b_4$$

$$m_{1,\,6} = -\tfrac{1}{784}\, b_4 + \tfrac{1}{784}\, b_3$$

$$m_{2,\,2} = \tfrac{3481}{3264}\, b_1 + \tfrac{9258282831623875}{7669235228057664}\, b_3 + \tfrac{236024329996203}{1278205871342944}\, b_4$$

$$m_{2,\,3} = -\tfrac{59}{408}\, b_1 - \tfrac{29294615794607}{29725717938208}\, b_3 - \tfrac{2944673881023}{29725717938208}\, b_4$$

$$m_{2,\,4} = -\tfrac{59}{1088}\, b_1 + \tfrac{260297319232891}{2556411742685888}\, b_3 - \tfrac{60834186813841}{1278205871342944}\, b_4$$

$$m_{2,\,5} = -\tfrac{1328188692663}{37594290333616}\, b_3 + \tfrac{1328188692663}{37594290333616}\, b_4$$

$$m_{2,\,6} = -\tfrac{8673}{2904112}\, b_3 + \tfrac{8673}{2904112}\, b_4$$

$$m_{3,\,3} = \tfrac{1}{51}\, b_1 + \tfrac{59}{192}\, b_2 + \tfrac{13777050223300597}{26218083221499456}\, b_4 + \tfrac{564461}{13384296}\, b_5 + \tfrac{378288882302546512209}{270764341349677687456}\, b_3$$

$$m_{3,\,4} = \tfrac{1}{136}\, b_1 - \tfrac{125059}{743572}\, b_5 - \tfrac{48363400904421872227}{55258028846872997744}\, b_3 - \tfrac{17220493277981}{89177153814624}\, b_4$$

$$m_{3,\,5} = -\tfrac{10532412077335}{42840005263888}\, b_4 + \tfrac{1613976761032884305}{7963657098519931984}\, b_3 + \tfrac{564461}{4461432}\, b_5$$

$$m_{3,\,6} = -\tfrac{960119}{1280713392}\, b_4 - \tfrac{3391}{6692148}\, b_5 + \tfrac{33235054191}{26452850508784}\, b_3$$

$$m_{4,\,4} = \tfrac{3}{1088}\, b_1 + \tfrac{507284006600757858213}{475219048083107777984}\, b_3 + \tfrac{1869103}{2230716}\, b_5 + \tfrac{1}{24}\, b_6 + \tfrac{1950062198436997}{3834617614028832}\, b_4$$

$$m_{4,\,5} = -\tfrac{4959271814984644613}{20965546238960637264}\, b_3 - \tfrac{1}{6}\, b_6 - \tfrac{15998714909649}{37594290333616}\, b_4 - \tfrac{375177}{743572}\, b_5$$

$$m_{4,\,6} = -\tfrac{368395}{2230716}\, b_5 + \tfrac{752806667}{539854092016}\, b_3 + \tfrac{1063649}{8712336}\, b_4 + \tfrac{1}{8}\, b_6$$

$$m_{5,\,5} = \tfrac{8386761355510099813}{128413970713633903242}\, b_3 + \tfrac{2224717261773437}{2763180339520776}\, b_4 + \tfrac{5}{6}\, b_6 + \tfrac{1}{24}\, b_7 + \tfrac{280535}{371786}\, b_5$$

$$m_{5,\,6} = -\tfrac{35039615}{213452232}\, b_4 - \tfrac{1}{6}\, b_7 - \tfrac{13091810925}{13226425254392}\, b_3 - \tfrac{1118749}{2230716}\, b_5 - \tfrac{1}{2}\, b_6$$

$$m_{6,\,6} = \tfrac{3290636}{80044587}\, b_4 + \tfrac{5580181}{6692148}\, b_5 + \tfrac{5}{6}\, b_7 + \tfrac{1}{24}\, b_8 + \tfrac{660204843}{13226425254392}\, b_3 + \tfrac{3}{4}\, b_6$$

The corresponding right boundary closure is given by replacing $b_i \rightarrow b_{N+1-i}$ for $i = 1..8$ followed by a permutation of both rows and columns. Let $m_{i,j}$ be the coefficient at row i and column$j$ in $M^{(b)}$. The matrix $M^{(b)}$ is symmetric, which means that it is completely defined by the the upper triangular part, i.e., all $m_{i,j}$, $i = 1..N, j = i..N$.

B.3 Sixth-order accurate

The discrete norm H is defined:

$$H = h\begin{bmatrix} \frac{13649}{43200} & & & & & & & \\ & \frac{12013}{8640} & & & & & & \\ & & \frac{2711}{4320} & & & & & \\ & & & \frac{5359}{4320} & & & & \\ & & & & \frac{7877}{8640} & & & \\ & & & & & \frac{43801}{43200} & & \\ & & & & & & 1 & \\ & & & & & & & \ddots \end{bmatrix}.$$

The 5th-order accurate boundary derivative operator is given by:

$$S = \frac{1}{h}\begin{bmatrix} -\frac{25}{12} & 4 & -3 & \frac{4}{3} & \frac{1}{4} & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & \frac{1}{4} & -\frac{4}{3} & 3 & -4 & \frac{25}{12} \end{bmatrix}.$$

The difference operator $D_4^{(6)}$:

$$\begin{bmatrix} 1 & -4 & 6 & -4 & 1 & & & & & & & & & \\ 1 & -4 & 6 & -4 & 1 & & & & & & & & & \\ 1 & -4 & 6 & -4 & 1 & & & & & & & & & \\ & 1 & -4 & 6 & -4 & 1 & & & & & & & & \\ & & 1 & -4 & 6 & -4 & 1 & & & & & & & \\ d4_1 & d4_2 & d4_3 & d4_4 & d4_5 & d4_6 & d4_7 & d4_8 & d4_9 & & & & & \\ & & & & 1 & -4 & 6 & -4 & 1 & & & & & \\ & & & & \ddots & \ddots & \ddots & \ddots & \ddots & & & & & \\ & & & & & 1 & -4 & 6 & -4 & 1 & & & & \\ & & & & & d4_9 & d4_8 & d4_7 & d4_6 & d4_5 & d4_4 & d4_3 & d4_2 & d4_1 \\ & & & & & & & 1 & -4 & 6 & -4 & 1 & & \\ & & & & & & & & 1 & -4 & 6 & -4 & 1 & \\ & & & & & & & & & 1 & -4 & 6 & -4 & 1 \\ & & & & & & & & & 1 & -4 & 6 & -4 & 1 \\ & & & & & & & & & 1 & -4 & 6 & -4 & 1 \end{bmatrix},$$

where

$$\begin{array}{ll}
d4_1 = 0.43819837221111761389 & d4_6 = 0.44412187877629861379 \\
d4_2 = -1.3130959257572520973 & d4_7 = -0.14810645777705395814 \\
d4_3 = 0.94797803521609260191 & d4_8 = 0.06831624563425347872 7 \\
d4_4 = 0.62436372993537192414 & d4_9 = -0.0047580470461699605110 \\
d4_5 = -1.0570178311926582165 &
\end{array} .$$

The difference operator $D_5^{(6)}$:

$$\begin{bmatrix}
-1 & 5 & -10 & 10 & -5 & 1 & & & & & \\
-1 & 5 & -10 & 10 & -5 & 1 & & & & & \\
-1 & 5 & -10 & 10 & -5 & 1 & & & & & \\
 & -1 & 5 & -10 & 10 & -5 & 1 & & & & \\
 & & -1 & 5 & -10 & 10 & -5 & 1 & & & \\
d5_1 & d5_2 & d5_3 & d5_4 & d5_5 & d5_6 & d5_7 & d5_8 & d5_9 & & \\
 & & & & -1 & 5 & -10 & 10 & -5 & 1 & \\
 & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & \\
 & & & -1 & 5 & -10 & 10 & -5 & 1 & & \\
 & & -d5_9 & -d5_8 & -d5_7 & -d5_6 & -d5_5 & -d5_4 & -d5_3 & -d5_2 & -d5_1 \\
 & & & & -1 & 5 & -10 & 10 & -5 & 1 & \\
 & & & & & -1 & 5 & -10 & 10 & -5 & 1 \\
 & & & & & -1 & 5 & -10 & 10 & -5 & 1 \\
 & & & & & -1 & 5 & -10 & 10 & -5 & 1 \\
 & & & & & -1 & 5 & -10 & 10 & -5 & 1
\end{bmatrix},$$

where

$$\begin{array}{ll}
d5_1 = -0.52131894522031211822 & d5_6 = 0.99549078638797583937 \\
d5_2 = 2.2819596734201098934 & d5_7 = -0.86326027050414424911 \\
d5_3 = -3.7719045450737321464 & d5_8 = 0.32111212717600079458 \\
d5_4 = 2.9041350609575637367 & d5_9 = -0.027844354179627268409 \\
d5_5 = -1.3183695329638344819 &
\end{array} .$$

The difference operator $D_6^{(6)}$:

$$\begin{bmatrix}
1 & -6 & 15 & -20 & 15 & -6 & 1 & & & & \\
1 & -6 & 15 & -20 & 15 & -6 & 1 & & & & \\
1 & -6 & 15 & -20 & 15 & -6 & 1 & & & & \\
1 & -6 & 15 & -20 & 15 & -6 & 1 & & & & \\
d6_1 & d6_2 & d6_3 & d6_4 & d6_5 & d6_6 & d6_7 & d6_8 & d6_9 & & \\
d6_{11} & d6_{12} & d6_{13} & d6_{14} & d6_{15} & d6_{16} & d6_{17} & d6_{18} & d6_{19} & & \\
 & & & 1 & -6 & 15 & -20 & 15 & -6 & 1 & \\
 & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & & \\
 & 1 & -6 & 15 & -20 & 15 & -6 & 1 & & & \\
 & & d6_{19} & d6_{18} & d6_{17} & d6_{16} & d6_{15} & d6_{14} & d6_{13} & d6_{12} & d6_{11} \\
 & & d6_9 & d6_8 & d6_7 & d6_6 & d6_5 & d6_4 & d6_3 & d6_2 & d6_1 \\
 & & & & 1 & -6 & 15 & -20 & 15 & -6 & 1 \\
 & & & & 1 & -6 & 15 & -20 & 15 & -6 & 1 \\
 & & & & 1 & -6 & 15 & -20 & 15 & -6 & 1 \\
 & & & & 1 & -6 & 15 & -20 & 15 & -6 & 1
\end{bmatrix},$$

where

$$\begin{aligned}
d6_1 &= 0.75842303723660880327 & d6_{11} &= 0.13990577549425331936 \\
d6_2 &= -4.2539074383142963409 & d6_{12} &= 0.19188572768373149785 \\
d6_3 &= 9.5415070255750278950 & d6_{13} &= -4.2605618076252134271 \\
d6_4 &= -10.388676034100037193 & d6_{14} &= 13.699047136714733223 \\
d6_5 &= 4.6179225213125232458 & d6_{15} &= -21.096213322723799491 \\
d6_6 &= 1.0 & d6_{16} &= 18.054894179643345962 \\
d6_7 &= -1.8471690085250092983 & d6_{17} &= -8.6164088505538261656 \\
d6_8 &= 0.62695371915714817047 & d6_{18} &= 2.0586773175102798144 \\
d6_9 &= -0.055053822341965281964 & d6_{19} &= -0.17122615614350473339
\end{aligned}.$$

The left boundary closure of the diagonal matrix $C_4^{(6)}$ is given by:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 16.652411984262326459 & 1.1325448501150501650 & 1 & \cdots \end{bmatrix}.$$

The corresponding right boundary closure is given by a permutation of both rows and columns. The left boundary closure of the diagonal matrix $C_5^{(6)}$ is given by:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 80.195967307267258036 & 1 & \cdots \end{bmatrix}.$$

The right boundary closure of the diagonal matrix $C_5^{(6)}$ is given by:

$$\begin{bmatrix} \cdots & 1 & 0 & 80.195967307267258036 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The left boundary closure of the diagonal matrix $C_6^{(6)}$ is given by:

$$\begin{bmatrix}0 & 0 & 0 & 0 & 106.92521670797294276 & 11.429565737638910841 & 1 & \cdots\end{bmatrix}.$$

The corresponding right boundary closure is given by a permutation of both rows and columns.

The different $B^{(6)}$ matrices are given by

$$\begin{aligned}(B_4^{(6)})_{i,\,i} &= \tfrac{1}{3}(b_{i-1} + b_i + b_{i+1}) \\ (B_5^{(6)})_{i,\,i} &= \tfrac{1}{2}(b_i + b_{i+1}) \qquad\qquad . \\ (B_6^{(6)})_{i,\,i} &= b_i\end{aligned}$$

The interior stencil of $M^{(b)}$ at row i is given by $(i = 10 \ldots N-9)$:

$$\begin{aligned}
m_{i,\,i-3} &= \tfrac{1}{40}\,b_{i-2} + \tfrac{1}{40}\,b_{i-1} - \tfrac{11}{360}\,b_{i-3} - \tfrac{11}{360}\,b_i \\
m_{i,\,i-2} &= \tfrac{1}{20}\,b_{i-3} - \tfrac{3}{10}\,b_{i-1} + \tfrac{1}{20}\,b_{i+1} + \tfrac{7}{40}\,b_i + \tfrac{7}{40}\,b_{i-2} \\
m_{i,\,i-1} &= -\tfrac{1}{40}\,b_{i-3} - \tfrac{3}{10}\,b_{i-2} - \tfrac{3}{10}\,b_{i+1} - \tfrac{1}{40}\,b_{i+2} - \tfrac{17}{40}\,b_i - \tfrac{17}{40}\,b_{i-1} \\
m_{i,\,i} &= \tfrac{1}{180}\,b_{i-3} + \tfrac{1}{8}\,b_{i-2} + \tfrac{19}{20}\,b_{i-1} + \tfrac{19}{20}\,b_{i+1} + \tfrac{1}{8}\,b_{i+2} + \tfrac{1}{180}\,b_{i+3} + \tfrac{101}{180}\,b_i \quad . \\
m_{i,\,i+1} &= -\tfrac{1}{40}\,b_{i-2} - \tfrac{3}{10}\,b_{i-1} - \tfrac{3}{10}\,b_{i+2} - \tfrac{1}{40}\,b_{i+3} - \tfrac{17}{40}\,b_i - \tfrac{17}{40}\,b_{i+1} \\
m_{i,\,i+2} &= \tfrac{1}{20}\,b_{i-1} - \tfrac{3}{10}\,b_{i+1} + \tfrac{1}{20}\,b_{i+3} + \tfrac{7}{40}\,b_i + \tfrac{7}{40}\,b_{i+2} \\
m_{i,\,i+3} &= \tfrac{1}{40}\,b_{i+1} + \tfrac{1}{40}\,b_{i+2} - \tfrac{11}{360}\,b_i - \tfrac{11}{360}\,b_{i+3}
\end{aligned}$$

The left boundary closure of $M^{(b)}$ (given by a 9×9 matrix) is given by

$$m_{1,\,1} = 0.79126675946955820939\,b_1 + 0.29684720906380007429\,b_2 + 0.0031855190887964290152\,b_3$$

$$+0.016324040425909519534\,b_4 + 0.031603022440944150877\,b_5 + 0.031679647480161052996\,b_6$$

$$+0.031485777339472539205\,b_7$$

$$m_{1,\,2} = -1.0166893393503381444\,b_1 - 0.028456273704916113690\,b_3$$

$$-0.041280298383492988198\,b_4 - 0.13922814516201405075\,b_5 - 0.11957773256112017666\,b_6$$

$$-0.11942677565293334109\,b_7$$

$$m_{1,\,3} = 0.070756429372437150463\,b_1 - 0.18454761060241510503\,b_2 - 0.043641631471118923470\,b_4$$

$$+0.24323679072077324609\,b_5 + 0.15821270735372154440\,b_6 + 0.16023485783647863076\,b_7$$

$$m_{1,\,4} = 0.22519915328913532127\,b_1 - 0.16627487110970548953\,b_2 + 0.027105309616486712977\,b_3$$

$$-0.19166461859684399091\,b_5 - 0.076841171601990145944\,b_6 - 0.082195869498316975759\,b_7$$

$$m_{1,\,5} = -0.052244034642020563167\,b_1 + 0.044400639485098762210\,b_2 - 0.0010239765473093878745\,b_3$$

$$+0.074034846453161740905\,b_4 + 0.012416255689984968954\,b_6 + 0.071886528478926012827\,b_5$$

$$+0.013793629971047355034\,b_7$$

$$m_{1,\,6} = -0.018288968138771973527\,b_1 + 0.0095746331632217580607\,b_2 - 0.00081057845305764042779\,b_3$$

$$-0.0073488455877755196984\,b_4 + 0.010636019497239069970\,b_5 - 0.013159670383826183824\,b_6$$

$$-0.021179364788387535246\,b_7$$

$$m_{1,\,7} = 0.0019118885633161709274\,b_4 - 0.040681303555291499361\,b_5 + 0.013196749810737491670\,b_6$$

$$+0.025572665181237836763\,b_7$$

$$m_{1,\,8} = 0.015596528711367857640\,b_5 - 0.0064861841573315378995\,b_6 - 0.0091103445540363197401\,b_7$$

$$m_{1,\,9} = 0.00055939836966298630593\,b_6 - 0.0013848225351007963723\,b_5 + 0.00082542416543781006633\,b_7$$

$$m_{2,\,2} = 1.3063321571116676286\,b_1 + 0.25420017604573457435\,b_3 + 0.10438978280925626095\,b_4$$

$$+0.66723280210321129509\,b_5 + 0.46818193597227494411\,b_6 + 0.46764154101958369201\,b_7$$

$$m_{2,\,3} = -0.090914102699924646049\,b_1 + 0.11036113131714764253\,b_4 - 1.2903975449975188870\,b_5$$

$$-0.66396052487350447871\,b_6 - 0.66159744640052061842\,b_7$$

$$m_{2,\,4} = -0.28935573956534316666\,b_1 - 0.24213200040645927216\,b_3 + 1.1876702550280310277\,b_5$$

$$+0.39565981499041363328\,b_6 + 0.38600489217558000007\,b_7$$

$$m_{2,\,5} = 0.067127744758037639890\,b_1 + 0.0091471926820756301800\,b_3 - 0.18721961430038080217\,b_4$$

$$-0.13193585588531745301\,b_6 - 0.48715757368119118874\,b_5 - 0.10475163122754481381\,b_7$$

$$m_{2,\,6} = 0.023499279745900688694\,b_1 + 0.0072409053835651813164\,b_3 + 0.018583789963916794487\,b_4$$

$$-0.092896161339386761743\,b_5 + 0.12235132704188076670\,b_6 + 0.11135203204362950339\,b_7$$

$$m_{2,\,7} = -0.0048347914064469075906\,b_4 + 0.23106838326878204031\,b_5 - 0.10807741421960079917\,b_6$$

$$-0.11815617764273433354\,b_7$$

$$m_{2,\,8} = -0.083681414344034553537\,b_5 + 0.040934994667670546616\,b_6 + 0.042746419676364006921\,b_7$$

$$m_{2,\,9} = -0.0035765451326969831434\,b_6 + 0.0073893991241210786821\,b_5 - 0.0038128539914240955387\,b_7$$

$$m_{3\,3} = 0.0063271611471368738078\, b_1 + 0.1147318200715868 5275\, b_2 + 0.11667405542796800075\, b_4$$

$$+2.7666108082854440372\, b_5 + 1.0709206899608171042\, b_6 + 1.0131613910329730572\, b_7$$

$$m_{3\,4} = 0.020137694138847972466\, b_1 + 0.10337179946308864017\, b_2 - 2.9132216211517427243\, b_5$$

$$-0.87558073434822622598\, b_6 - 0.69099571834888124265\, b_7$$

$$m_{3\,5} = -0.0046717510915754628683\, b_1 - 0.027603533656377128278\, b_2 - 0.19792902986208699745\, b_4$$

$$+0.54029853383734330523\, b_6 + 1.2391775930319110779\, b_5 + 0.26280380502473582273\, b_7$$

$$m_{3\,6} = -0.0016354308669218878195\, b_1 - 0.0059524752758832596197\, b_2 + 0.019646827777442752194\, b_4$$

$$+0.32366400126390466006\, b_5 - 0.46595166932288709739\, b_6 - 0.22172727209417368594\, b_7$$

$$m_{3\,7} = -0.0051113531893524745496\, b_4 - 0.53558781637747543460\, b_5 + 0.33283351044897389336\, b_6$$

$$+0.20786565911785401579\, b_7$$

$$m_{3\,8} = 0.18243281741342895622\, b_5 - 0.10598160301968184459\, b_6 - 0.076451214393747111630\, b_7$$

$$m_{3\,9} = 0.0092090899634437994856\, b_6 - 0.015915028188724931671\, b_5 + 0.0067059382252811321853\, b_7$$

$$m_{4\,4} = 0.064092997759871869867\, b_1 + 0.093136576388046999489\, b_2 + 0.23063676246347492291\, b_3$$

$$+3.6894403082837166203\, b_5 + 1.1905503386876088738\, b_6 + 0.59124795468888565194\, b_7$$

$$m_{4\,5} = -0.014868958192656041286\, b_1 - 0.024870405993901607642\, b_2 - 0.0087129289077117541871\, b_3$$

$$-1.2635078373718242057\, b_6 - 0.30583173978439973269\, b_7 - 1.4706919260458029548\, b_5$$

$$m_{4\,6} = -0.0052051474298559556576\, b_1 - 0.0053630987475285424890\, b_2 - 0.0068971427657906095463\, b_3 \quad .$$

$$-0.78575245216674501017\, b_5 + 0.22911480054237346001\, b_7 + 0.99770643562927505292\, b_6$$

$$m_{4\,7} = 0.66972974880676622652\, b_5 - 0.50132473560721279390\, b_6 - 0.17951612431066454373\, b_7$$

$$m_{4\,8} = -0.20229090601117515652\, b_5 + 0.14534218580636584986\, b_6 + 0.056948720204809306656\, b_7$$

$$m_{4\,9} = -0.012004296184410038337\, b_6 - 0.0047769156693859238415\, b_7 + 0.016781211853795962179\, b_5$$

$$m_{5\,5} = 0.0034494550959102336252\, b_1 + 0.0066411834994278261016\, b_2 + 0.00032915450832718628585\, b_3$$

$$+0.33577217075764772000\, b_4 + 2.0964133295790264390\, b_6 + 0.23173232041831268550\, b_7$$

$$+0.0061078257643682645765\, b_8 + 0.71091258506833766956\, b_5$$

$$m_{5\,6} = 0.0012075440723041938061\, b_1 + 0.0014321166657521476075\, b_2 + 0.00026055826461832559573\, b_3$$

$$-0.033329411132516353908\, b_4 - 0.28082416973855326836\, b_7 - 0.027209080835250836084\, b_8$$

$$+0.10458654356829219874\, b_5 - 1.3484369866671155432\, b_6$$

$$m_{5\,7} = 0.0086710380841746926251\, b_4 + 0.17360734113554285637\, b_6 + 0.053313621252876254126\, b_8$$

$$-0.24249352624045263018\, b_5 + 0.15690152576785882706\, b_7$$

$$m_{5\,8} = -0.086316839802171222760\, b_6 + 0.026988423604709992435\, b_7 + 0.080981941477156510853\, b_5$$

$$-0.032764636390806391639\, b_8$$

$$m_{5\,9} = 0.0074620594845308550733\, b_6 - 0.00081216403616686789496\, b_7 + 0.00055227020881270902093\, b_8$$

$$-0.0072021656571766961993\, b_5$$

$$
\begin{aligned}
m_{6\,6} &= 0.00042272261734493450425\, b_1 + 0.00030882419443789644048\, b_2 + 0.00020625757066474306202\, b_3 \\
&\quad +0.0033083434042009682567\, b_4 + 0.58280470164050018158\, b_5 + 0.80541742203662154736\, b_7 \\
&\quad +0.13383632334100334433\, b_8 + 0.0055555555555555555556\, b_9 + 1.1903620718618930511\, b_6 \\
m_{6\,7} &= -0.00086070442526864133026\, b_4 - 0.17480747086739049893\, b_5 - 0.31325448501150501650\, b_8 \\
&\quad -0.025000000000000000000\, b_9 - 0.31691663053104292713\, b_7 - 0.66916070916479291611\, b_6 \\
m_{6\,8} &= 0.033546617916933521087\, b_5 - 0.33436200223869714050\, b_7 + 0.050000000000000000000\, b_9 \\
&\quad +0.21697906098076027508\, b_6 + 0.18383632334100334433\, b_8 \\
m_{6\,9} &= 0.029125184768230046430\, b_7 + 0.022790919164749163916\, b_8 - 0.030689859975187405305\, b_6 \\
&\quad -0.0017817995133473605962\, b_5 - 0.030555555555555555556\, b_9 \\
m_{7\,7} &= 0.00022392237357715991790\, b_4 + 0.12754377854309566738\, b_5 + 1.0116994839296081646\, b_6 \\
&\quad +0.96988172751725752475\, b_8 + 0.12500000000000000000\, b_9 + 0.0055555555555555555556\, b_{i-3} \\
&\quad +0.48231775430312815001\, b_7 \\
m_{7\,8} &= -0.037841139730330129499\, b_5 - 0.29975568851348273616\, b_6 - 0.30000000000000000000\, b_9 \qquad . \\
&\quad -0.025000000000000000000\, b_{i-3} - 0.39914868674468211784\, b_7 - 0.43825448501150501650\, b_8 \\
m_{7\,9} &= 0.046981462180226839339\, b_6 - 0.29668637874712374587\, b_8 + 0.050000000000000000000\, b_{i-3} \\
&\quad +0.17163557041460064817\, b_7 + 0.0030693461522962583624\, b_5 + 0.17500000000000000000\, b_9 \\
m_{8\,8} &= 0.012303289427168044554\, b_5 + 0.11836475296458983325\, b_6 + 0.94105118982279433342\, b_7 \\
&\quad +0.95000000000000000000\, b_9 + 0.12500000000000000000\, b_{i-3} + 0.0055555555555555555556\, b_{i-2} \\
&\quad +0.56994743445211445545\, b_8 \\
m_{8\,9} &= -0.023080678926719163396\, b_6 - 0.29866250537751494972\, b_7 - 0.30000000000000000000\, b_{i-3} \\
&\quad -0.025000000000000000000\, b_{i-2} - 0.0010477348605150508026\, b_5 - 0.42720908083525083608\, b_8 \\
&\quad -0.42500000000000000000\, b_9 \\
m_{9\,9} &= 0.0051393702211491099770\, b_6 + 0.12477232150094220014\, b_7 + 0.95055227020881270902\, b_8 \\
&\quad +0.95000000000000000000\, b_{i-3} + 0.12500000000000000000\, b_{i-2} + 0.0055555555555555555556\, b_{i-1} \\
&\quad +0.000091593624651536418269\, b_5 + 0.56111111111111111111\, b_9
\end{aligned}
$$

The corresponding right boundary closure is given by replacing $b_i \rightarrow b_{N+1-i}$ for $i = 1..12$ followed by a permutation of both rows and columns.

B.4 Eighth-order accurate

For the eighth order case, we did not manage to find boundary closures in $D^{(8)}_{5,6,7,8}$ and $C^{8}_{5,6,7,8}$ such that we get back the compatible narrow stencil SBP operator presented in [16]. This is not to say that such closure does not exist. However, we are free to chose for example the difference operator $D_5^{(8)}$:

$$\begin{bmatrix} -1 & 5 & -10 & 10 & -5 & 1 & & \\ -1 & 5 & -10 & 10 & -5 & 1 & & \\ -1 & 5 & -10 & 10 & -5 & 1 & & \\ & -1 & 5 & -10 & 10 & -5 & 1 & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix},$$

the difference operator $D_6^{(8)}$:

$$\begin{bmatrix} 1 & -6 & 15 & -20 & 15 & -6 & 1 & \\ 1 & -6 & 15 & -20 & 15 & -6 & 1 & \\ 1 & -6 & 15 & -20 & 15 & -6 & 1 & \\ 1 & -6 & 15 & -20 & 15 & -6 & 1 & \\ & 1 & -6 & 15 & -20 & 15 & -6 & 1 \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix},$$

the difference operator $D_7^{(8)}$:

$$\begin{bmatrix} 1 & -6 & 15 & -20 & 15 & -6 & 1 & & \\ -1 & 7 & -21 & 35 & -35 & 21 & -7 & 1 & \\ -1 & 7 & -21 & 35 & -35 & 21 & -7 & 1 & \\ -1 & 7 & -21 & 35 & -35 & 21 & -7 & 1 & \\ & -1 & 7 & -21 & 35 & -35 & 21 & -7 & 1 \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \end{bmatrix},$$

and the difference operator $D_8^{(8)}$:

$$\begin{bmatrix} 1 & -6 & 15 & -20 & 15 & -6 & 1 & & & \\ 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & \\ 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & \\ 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & \\ 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 & \\ & 1 & -8 & 28 & -56 & 70 & -56 & 28 & -8 & 1 \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \end{bmatrix}.$$

Based on the closures in the corresponding 4th and 6th order cases an initial guess is to set the first 6-8 points to zero in $C^8_{5,6,7,8}$ to zero to get a closure not to bad.

At least we know how to chose the diagonal matrices $B^{(8)}_{5,6,7,8}$ to obtain a narrow interior stencil:

$$\begin{aligned}
(B^{(8)}_5)_{i,i} &= \tfrac{1}{4}(b_{i-1} + b_i + b_{i+1} + b_{i+2}) \\
(B^{(8)}_6)_{i,i} &= \tfrac{3}{10}(b_{i-1} + \tfrac{4}{3} b_i + b_{i+1}) \\
(B^{(8)}_7)_{i,i} &= \tfrac{1}{2}(b_i + b_{i+1}) \\
(B^{(8)}_8)_{i,i} &= b_i
\end{aligned} \quad .$$

at $i = 1..N-2$. Since $B^{(8)}_{5,6,7,8}$ are multiplied with $C^8_{5,6,7,8}$ that are zero at the first few boundary points, we can as well put the corner points to zero in $B^{(8)}_{5,6,7,8}$.

This means that to the interior stencil of $M^{(b)}$ at row i is given by $(i = 13 \ldots N-12)$:

$$\begin{aligned}
m_{i,i-4} &= -\tfrac{1}{280}\, b_{i-2} - \tfrac{1}{210}\, b_{i-3} - \tfrac{1}{210}\, b_{i-1} + \tfrac{5}{672}\, b_{i-4} + \tfrac{5}{672}\, b_i \\
m_{i,i-3} &= \tfrac{2}{35}\, b_{i-2} + \tfrac{2}{35}\, b_{i-1} - \tfrac{1}{70}\, b_{i-4} - \tfrac{1}{70}\, b_{i+1} - \tfrac{1}{18}\, b_{i-3} - \tfrac{1}{18}\, b_i \\
m_{i,i-2} &= \tfrac{11}{105}\, b_{i-3} - \tfrac{2}{5}\, b_{i-1} + \tfrac{3}{280}\, b_{i-4} + \tfrac{3}{280}\, b_{i+2} + \tfrac{11}{105}\, b_{i+1} + \tfrac{31}{168}\, b_{i-2} + \tfrac{31}{168}\, b_i \\
m_{i,i-1} &= -\tfrac{13}{35}\, b_{i-2} - \tfrac{1}{15}\, b_{i-3} - \tfrac{1}{210}\, b_{i-4} - \tfrac{1}{15}\, b_{i+2} - \tfrac{13}{35}\, b_{i+1} - \tfrac{1}{210}\, b_{i+3} - \tfrac{5}{14}\, b_{i-1} - \tfrac{5}{14}\, b_i \\
m_{i,i} &= \tfrac{1}{1120}\, b_{i+4} + \tfrac{53}{280}\, b_{i-2} + \tfrac{17}{630}\, b_{i-3} + \tfrac{69}{70}\, b_{i-1} + \tfrac{1}{1120}\, b_{i-4} + \tfrac{53}{280}\, b_{i+2} + \tfrac{69}{70}\, b_{i+1} + \tfrac{17}{630}\, b_{i+3} + \tfrac{445}{1008}\, b_i \quad . \\
m_{i,i+1} &= -\tfrac{1}{210}\, b_{i+4} - \tfrac{1}{15}\, b_{i-2} - \tfrac{1}{210}\, b_{i-3} - \tfrac{13}{35}\, b_{i-1} - \tfrac{13}{35}\, b_{i+2} - \tfrac{1}{15}\, b_{i+3} - \tfrac{5}{14}\, b_{i+1} - \tfrac{5}{14}\, b_i \\
m_{i,i+2} &= \tfrac{3}{280}\, b_{i+4} + \tfrac{3}{280}\, b_{i-2} + \tfrac{11}{105}\, b_{i-1} - \tfrac{2}{5}\, b_{i+1} + \tfrac{11}{105}\, b_{i+3} + \tfrac{31}{168}\, b_{i+2} + \tfrac{31}{168}\, b_i \\
m_{i,i+3} &= -\tfrac{1}{70}\, b_{i+4} - \tfrac{1}{70}\, b_{i-1} + \tfrac{2}{35}\, b_{i+2} + \tfrac{2}{35}\, b_{i+1} - \tfrac{1}{18}\, b_i - \tfrac{1}{18}\, b_{i+3} \\
m_{i,i+4} &= -\tfrac{1}{280}\, b_{i+2} - \tfrac{1}{210}\, b_{i+1} - \tfrac{1}{210}\, b_{i+3} + \tfrac{5}{672}\, b_{i+4} + \tfrac{5}{672}\, b_i
\end{aligned}$$

Bibliography

[1] Martin Almquist, Ilkka Karasalo, and Ken Mattsson. Atmospheric sound propagation over large-scale irregular terrain. *Journal of Scientific Computing*, January 2014.

[2] Ken Mattsson. Summation by parts operators for finite difference approximations of second-derivatives with variable coefficients. *J. Comput. Physics*, October 2011.

[3] Hector D. Ceniceros Chohong Min, Frederic Giboub. A supra-convergent finite difference scheme for the variable coefficient poisson equation on non-graded grids. *Journal of Computational Physics*, 2006. ISSN 0021-9991. doi: http://dx.doi.org/10.1016/j.jcp.2006.01.046.

[4] Knut-Andersson Lie. *An Introduction to Reservoir Simulation Using MATLAB*. SINTEF ICT, Oslo, Norway, May 2014.

[5] Mark H. Carpenter, David Gottlieb, and Saul Abarbanel. Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: Methodology and application to high-order compact schemes. *Journal of Computational Physics*, 111(2):220 – 236, 1994. ISSN 0021-9991. doi: http://dx.doi.org/10.1006/jcph.1994.1057. URL http://www.sciencedirect.com/science/article/pii/S0021999184710576.

[6] Mrst - matlab reservoir simulation toolbox. http://www.sintef.com/Projectweb/MRST/, 2014. Accessed: 2014-08-18.

[7] Spe comparative solution project. http://www.spe.org/web/csp//, 2014. Accessed: 2014-08-18.

[8] Thomas C. Sideris Xu-Dong Liu. Convergence of finite difference methods for poisson's equation with interfaces, September 2001. URL arXiv:math/0108122v1. Accessed: 2014-08-18.

[9] Ken Mattsson, Frank Ham, and Gianluca Iaccarino. Stable boundary treatment for the wave equation on second-order form. *J. Sci. Comput.*, 41(3):366–383, December 2009. ISSN 0885-7474. doi: 10.1007/s10915-009-9305-1. URL http://dx.doi.org/10.1007/s10915-009-9305-1.

[10] Ken Mattsson, Frank Ham, and Gianluca Iaccarino. Stable and accurate wave-propagation in discontinuous media. *J. Comput. Phys.*, 227(19):8753–8767, October 2008. ISSN 0021-9991. doi: 10.1016/j.jcp.2008.06.023. URL http://dx.doi.org/10.1016/j.jcp.2008.06.023.

[11] Kristoffer Virta and Ken Mattsson. Acoustic wave propagation in complicated geometries and heterogeneousmedia. *Journal of Scientific Computing*, pages 1–29, 2014. ISSN 0885-7474. doi: 10.1007/s10915-014-9817-1. URL http://dx.doi.org/10.1007/s10915-014-9817-1.

[12] Levy-desplanques theorem. http://planetmath.org/levydesplanquestheorem, 2014. Accessed: 2014-08-18.

[13] Wolfram mathworld - reducible matrix. http://mathworld.wolfram.com/ReducibleMatrix.html, 2014. Accessed: 2014-09-08.

[14] M. Svärd and J. Nordström. On the order of accuracy for difference approximations of initial-boundary value problems. *J. Comput. Physics*, 218:333–352, October 2006.

[15] Ken Mattsson and Jan Nordström. Summation by parts operators for finite difference approximations of second derivatives. *J. Comput. Phys.*, 199(2):503–540, September 2004. ISSN 0021-9991. doi: 10.1016/j.jcp.2004.03.001. URL http://dx.doi.org/10.1016/j.jcp.2004.03.001.

[16] K. Mattsson, M. Svärd, and M. Shoeybi. Stable and accurate schemes for the compressible navier–stokes equations. *Journal of Computational Physics*, 227(4):2293 – 2316, 2008. ISSN 0021-9991. doi: http://dx.doi.org/10.1016/j.jcp.2007.10.018. URL http://www.sciencedirect.com/science/article/pii/S0021999107004627.