



Geração de campos tensoriais anisotrópicos para projeção de Helmholtz através de Simulated Annealing

Arthur Gonze Machado

JUIZ DE FORA
MARÇO, 2021

Geração de campos tensoriais anisotrópicos para projeção de Helmholtz através de Simulated Annealing

ARTHUR GONZE MACHADO

Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Bacharelado em Ciência da Computação

Orientador: Marcelo Bernardes Vieira

Co-orientador: Gilson Antonio Giraldi

JUIZ DE FORA

MARÇO, 2021

GERAÇÃO DE CAMPOS TENSORIAIS ANISOTRÓPICOS PARA PROJEÇÃO DE HELMHOLTZ ATRAVÉS DE SIMULATED ANNEALING

Arthur Gonze Machado

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS
EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTE-
GRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE
BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Marcelo Bernardes Vieira
D.Sc. em Ciência da Computação

Gilson Antonio Giraldi
Doutor em Engenharia de Sistemas e Computação

Marcelo Caniato Renhe
Doutor em Engenharia de Sistemas e Computação

Luiz Maurílio da Silva Maciel
Doutor em Engenharia de Sistemas e Computação

JUIZ DE FORA
11 DE MARÇO, 2021

Aos meus amigos e irmãos.

Aos pais, pelo apoio e sustento.

Resumo

Métodos que buscam a redução de erros numéricos são de grande valia para o contexto de simulações. No entanto, o desenvolvimento e emprego destes métodos possuem um grande custo computacional, além da complexidade dos problemas em que são aplicados. Este trabalho tem como proposta o desenvolvimento de um método meta-heurístico baseado em *Simulated Annealing* para gerar campos tensoriais positivos definidos. O campo tensorial gerado visa a redução do resíduo numérico proveniente da resolução da decomposição anisotrópica de Helmholtz e, com isso, reduzir o divergente do campo vetorial resultante deste processo. A partir disso, é definido um conjunto de métricas para comparação entre o método proposto e a decomposição isotrópica equivalente. Por fim, é realizada uma análise de desempenho da redução do divergente obtido pelo método, e são discutidos os pontos que, se aprimorados, podem melhorar o resultado obtido.

Palavras-chave: *Simulated Annealing*, Campos Tensoriais, Decomposição Anisotrópica de Helmholtz.

Abstract

Methods that reduce numerical errors have great value in the simulations context. However, the development and use of these methods have a high computational cost, in addition to the complexity of the problems they are applied to. This work proposes the development of a metaheuristic method based on Simulated Annealing to generate defined positive tensor fields. The generated tensor field aims to reduce the numerical residue from the resolution of Helmholtz's anisotropic decomposition and, with this, reduce the divergence of the resulting vector field. From this, a set of metrics is defined for comparison between the proposed method and the equivalent isotropic decomposition. Finally, a performance analysis of the divergence reduction obtained by the method is carried out, and the points that can improve the obtained result are discussed.

Keywords: *Simulated Annealing*, Tensor Fields, Helmholtz Anisotropic Decomposition.

Agradecimentos

Aos meus pais e irmãos, pelo encorajamento e apoio. Aos meus amigos pela companhia, e conselhos.

Aos professores Marcelo e Gilson pela orientação, amizade e principalmente, pela paciência, sem a qual este trabalho não se realizaria.

Aos professores do Departamento de Ciência da Computação pelos seus ensinamentos, e aos funcionários do curso, que durante esses anos, contribuíram de algum modo para o meu enriquecimento pessoal e acadêmico.

“É melhor acender uma vela do que praguejar contra a escuridão”.

Adágio, citado em O Mundo Assombrado pelos Demônios (Carl Sagan)

Conteúdo

Lista de Figuras	8
Lista de Tabelas	10
Lista de Abreviações	11
1 Introdução	12
1.1 Problema	13
1.2 Justificativa	13
1.3 Objetivos	14
2 Trabalhos Relacionados	15
3 Fundamentação Teórica	17
3.1 Decomposição de Helmholtz	17
3.1.1 Decomposição Isotrópica	17
3.1.2 Decomposição Anisotrópica	18
3.2 Campos Tensoriais	19
3.3 Simulated Annealing	21
4 Método Proposto	24
4.1 Modelo Computacional	26
5 Resultados	30
5.1 Campos Vetoriais Iniciais	31
5.1.1 Campo 1 - Descontínuo Vertical	33
5.1.2 Campo 2 - Descontínuo Diagonal	33
5.1.3 Campo 3 - Complexo	34
5.1.4 Campo 4 - Radial Rotacional	34
5.2 Projeção Isotrópica	34
5.2.1 Campo 1 - Descontínuo Vertical	36
5.2.2 Campo 2 - Descontínuo Diagonal	37
5.2.3 Campo 3 - Complexo	38
5.2.4 Campo 4 - Radial Rotacional	39
5.3 Projeção Anisotrópica	39
5.3.1 Campo 1 - Descontínuo Vertical	40
5.3.2 Campo 2 - Descontínuo Diagonal	45
5.3.3 Campo 3 - Complexo	50
5.3.4 Campo 4 - Radial Rotacional	55
5.4 Análise dos Dados	60
5.4.1 Valores Brutos	63
5.4.2 Proporções entre Valores	65
5.4.3 Reduções Obtidas	67
6 Conclusão	69

Lista de Figuras

3.1	Formatos dos tensores, com glifos quádracos, variando os autovalores (Kindlmann, 2004).	20
4.1	Fluxograma de funcionamento do Algoritmo 1	27
4.2	Fluxograma de funcionamento do Algoritmo 2	29
5.1	Campo 1 - Descontínuo Vertical em seu estado inicial.	33
5.2	Campo 2 - Descontínuo Diagonal em seu estado inicial.	33
5.3	Campo 3 - Complexo em seu estado inicial.	34
5.4	Campo 4 - Radial Rotacional em seu estado inicial.	34
5.5	Campo Tensorial Isotrópico.	36
5.6	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 1 com a projeção isotrópica.	36
5.7	Campo 1 após projeção isotrópica.	37
5.8	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 2 com a projeção isotrópica.	37
5.9	Campo 2 após projeção isotrópica.	38
5.10	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 3 com a projeção isotrópica.	38
5.11	Campo 3 após projeção isotrópica.	38
5.12	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 4 com a projeção isotrópica.	39
5.13	Campo 4 após projeção isotrópica.	39
5.14	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 1 após execução do Algoritmo 1.	41
5.15	Campo 1 projetado após execução do Algoritmo 1.	41
5.16	Campo tensorial anisotrópico obtido após execução do Algoritmo 1 no Campo 1.	42
5.17	Evolução da média quadrática do divergente do Algoritmo 1 no Campo 1.	42
5.18	Evolução da soma absoluta dos divergentes do Algoritmo 1 no Campo 1.	43
5.19	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 1 após execução do Algoritmo 2.	43
5.20	Campo 1 projetado após execução do Algoritmo 2.	43
5.21	Campo tensorial anisotrópico obtido após execução do Algoritmo 2 no Campo 1.	44
5.22	Evolução da média quadrática do divergente do Algoritmo 2 no Campo 1.	44
5.23	Evolução da soma absoluta dos divergentes do Algoritmo 2 no Campo 1.	45
5.24	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 2 após execução do Algoritmo 1.	46
5.25	Campo 2 projetado após execução do Algoritmo 1.	46
5.26	Campo tensorial anisotrópico obtido após execução do Algoritmo 1 no Campo 2.	47
5.27	Evolução da média quadrática do divergente do Algoritmo 1 no Campo 2.	47
5.28	Evolução da soma absoluta dos divergentes do Algoritmo 1 no Campo 2.	48

5.29	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 2 após execução do Algoritmo 2.	48
5.30	Campo 2 projetado após execução do Algoritmo 2.	49
5.31	Campo tensorial anisotrópico obtido após execução do Algoritmo 2 no Campo 2.	49
5.32	Evolução da média quadrática do divergente do Algoritmo 2 no Campo 2.	50
5.33	Evolução da soma absoluta dos divergentes do Algoritmo 2 no Campo 2.	50
5.34	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 3 após execução do Algoritmo 1.	51
5.35	Campo 3 projetado após execução do Algoritmo 1.	51
5.36	Campo tensorial anisotrópico obtido após execução do Algoritmo 1 no Campo 3.	52
5.37	Evolução da média quadrática do divergente do Algoritmo 1 no Campo 3.	52
5.38	Evolução da soma absoluta dos divergentes do Algoritmo 1 no Campo 3.	53
5.39	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 3 após execução do Algoritmo 2.	53
5.40	Campo 3 projetado após execução do Algoritmo 2.	53
5.41	Campo tensorial anisotrópico obtido após execução do Algoritmo 2 no Campo 3.	54
5.42	Evolução da média quadrática do divergente do Algoritmo 2 no Campo 3.	54
5.43	Evolução da soma absoluta dos divergentes do Algoritmo 3 no Campo 1.	55
5.44	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 4 após execução do Algoritmo 1.	56
5.45	Campo 4 projetado após execução do Algoritmo 1.	56
5.46	Campo tensorial anisotrópico obtido após execução do Algoritmo 1 no Campo 4.	57
5.47	Evolução da média quadrática do divergente do Algoritmo 1 no Campo 4.	57
5.48	Evolução da soma absoluta dos divergentes do Algoritmo 1 no Campo 4.	58
5.49	Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 4 após execução do Algoritmo 2.	58
5.50	Campo 4 projetado após execução do Algoritmo 2.	59
5.51	Campo tensorial anisotrópico obtido após execução do Algoritmo 2 no Campo 4.	59
5.52	Evolução da média quadrática do divergente do Algoritmo 2 no Campo 4.	60
5.53	Evolução da soma absoluta dos divergentes do Algoritmo 2 no Campo 4.	60
5.54	Dados relativos ao Campo 1.	63
5.55	Dados relativos ao Campo 2.	63
5.56	Dados relativos ao Campo 3.	64
5.57	Dados relativos ao Campo 4.	64
5.58	Proporções entre os Dados relativos ao Campo 1.	65
5.59	Proporções entre os Dados relativos ao Campo 2.	65
5.60	Proporções entre os Dados relativos ao Campo 3.	66
5.61	Proporções entre os Dados relativos ao Campo 4.	66
5.62	Reduções obtidas em relação ao Campo 1.	67
5.63	Reduções obtidas em relação ao Campo 2.	67
5.64	Reduções obtidas em relação ao Campo 3.	68
5.65	Reduções obtidas em relação ao Campo 4.	68

Lista de Tabelas

5.1	Dados relativos ao estado inicial dos campos 1 e 2.	32
5.2	Dados relativos ao estado inicial dos campos 3 e 4.	32
5.3	Dados relativos a projeção isotrópica dos campos 1 e 2.	35
5.4	Dados relativos a projeção isotrópica dos campos 3 e 4.	35
5.5	Dados relativos ao processo completo da projeção anisotrópica do Campo 1.	40
5.6	Dados relativos aos campos aceitos da projeção anisotrópica do Campo 1. .	41
5.7	Dados relativos aos processo completo da projeção anisotrópica do Campo 2.	45
5.8	Dados relativos aos campos aceitos da projeção anisotrópica do Campo 2. .	45
5.9	Dados relativos ao processo completo da projeção anisotrópica do Campo 3.	50
5.10	Dados relativos aos campos aceitos da projeção anisotrópica do Campo 3. .	51
5.11	Dados relativos ao processo completo da projeção anisotrópica do Campo 4.	55
5.12	Dados relativos aos campos aceitos da projeção anisotrópica do Campo 4. .	55

Lista de Abreviações

DCC Departamento de Ciência da Computação

UFJF Universidade Federal de Juiz de Fora

1 Introdução

Métodos que buscam reduzir resíduos numéricos são computacionalmente custosos, dada a complexidade inerente dos problemas em que são aplicados. A busca da redução de resíduos numéricos possui aplicação direta em problemas de simulação, onde a fidelidade do comportamento do objeto simulado no ambiente discreto deve ser o mais próximo possível de seu comportamento no contínuo. O não tratamento do resíduo numérico nesses casos pode gerar artefatos indesejados conforme a simulação evolui e, com isso, invalidar os resultados obtidos. No contexto de simulações, campos vetoriais são ferramentas fundamentais para a descrição do comportamento do objeto simulado. No entanto, em simulações de fluido, por exemplo, é necessário o tratamento do divergente atrelado a este campo vetorial.

O processo numérico de obtenção de um campo vetorial livre de divergente, utilizando a decomposição anisotrópica de Helmholtz, envolve um campo tensorial para modular o campo vetorial, sendo sujeito a erros associados às discretizações inerentes ao esquema numérico.

O presente trabalho define um método meta-heurístico para encontrar campos de tensores diádicos positivos definidos, que reduzem resíduos numéricos no processo de obtenção de um campo vetorial livre de divergente via decomposição anisotrópica de Helmholtz.

De forma alguma este trabalho desenvolve um método ideal para a redução do divergente do campo vetorial. O objetivo deste estudo é encontrar evidências de que campos tensoriais podem ser utilizados como parte inerente do meio para a redução do divergente do campo vetorial.

Considerando que é desconhecida a estrutura dos campos tensoriais que cumprem esse objetivo, os principais desafios serão a definição dos parâmetros de busca por esses campos, definição de campos vetoriais que evidenciam padrões nos campos tensoriais, e o desenvolvimento de algoritmos eficientes para essa busca.

1.1 Problema

O problema está presente ao se tentar obter um campo vetorial livre de divergente como resultado da decomposição anisotrópica de Helmholtz. Dado um campo de velocidades $\mathbf{u}(\mathbf{r})$, a operação de projeção é o processo de encontrar um campo vetorial $\mathbf{w}(\mathbf{r}) = P(\mathbf{u}(\mathbf{r}))$, que satisfaça a condição $\nabla \cdot \mathbf{w} = 0$. Muitos modelos de simulação forçam a condição usando a decomposição isotrópica de Helmholtz (Bridson, 2015).

O campo tensorial anisotrópico, conseqüentemente, necessita de um método de projeção próprio. Como discutido em Renhe et al. (2019), a decomposição isotrópica de Helmholtz pode violar as restrições locais, representadas pelo tensor. A projeção deve ter como resultado um campo vetorial que não só é livre de divergente, mas também atende às restrições de troca impostas pelo campo tensorial. O principal fato a ser explorado é a existência e unicidade da decomposição anisotrópica de Helmholtz apresentada em Dassios et al. (2002).

No espaço contínuo temos um campo vetorial livre de divergente, porém numericamente isso não ocorre. Para a obtenção da solução da Equação (3.11) é utilizado um método numérico. O resíduo numérico inerente à utilização desse método deve ser considerado, e a solução encontrada não será um campo totalmente livre de divergente. Em Vieira et al. (2019), um método numérico foi desenvolvido para projeção em grades regulares, via solução da Equação (3.11). Desta forma, será possível avaliar a utilização da projeção anisotrópica como veículo para reduzir erros residuais, em comparação a uma projeção isotrópica equivalente. A ideia é inicialmente evidenciar a hipótese através do uso da meta-heurística *Simulated Annealing*.

1.2 Justificativa

O desenvolvimento deste trabalho é relevante, pois a obtenção de um divergente nulo, ou o mais próximo possível, é um problema para áreas de simulação, principalmente para simulação de fluidos e animação. A relevância para animação está atrelada ao fato de que a redução do divergente evita o surgimento de artefatos indesejados e, assim, o resultado fica o mais próximo possível do desejado pelo animador. Esse estudo também poderá se

desdobrar em um futuro trabalho com o desenvolvimento de um método numérico para geração de campos tensoriais que reduzem o resíduo numérico, que poderá beneficiar os métodos empregados em simulações.

A escolha da utilização da meta-heurística é motivada por facilitar a busca por campos tensoriais que influenciam positivamente o divergente do campo vetorial, sem necessariamente se ter conhecimento prévio do campo tensorial desejado. A partir dos campos encontrados, será possível realizar uma busca por padrões em relação ao impacto do campo tensorial no comportamento do divergente do campo vetorial.

1.3 Objetivos

O objetivo primário e geral do trabalho é estudar e desenvolver um método meta-heurístico para grades regulares, visando obter campos tensoriais positivos definidos que reduzem erros numéricos residuais provenientes da operação de projeção. Nesse contexto destacamos os seguintes objetivos específicos:

- Definir atributos dos tensores que, ao serem modificados, poderão gerar campos tensoriais que influenciam positivamente o divergente do campo vetorial projetado. Ou seja, será feita uma análise de atributos como: escala do tensor, ângulo de rotação, proporção e valores de seus autovalores, e direções dos autovetores;
- Fazer uma análise morfológica dos campos tensoriais encontrados;
- Desenvolver um método para visualização do impacto no divergente do campo vetorial, e no comportamento do fluido.

2 Trabalhos Relacionados

O uso de campos tensoriais como a principal ferramenta deste trabalho para a resolução do problema apresentado se dá principalmente pelo resultado obtido em [Vieira et al. \(2019\)](#), onde foi desenvolvido um método para controle de fluidos, tanto em relação ao caminho, quanto ao estado final. Foi empregada uma abordagem baseada em malha regular, na qual se define um campo tensorial que permite desviar o fluido, alterando os vetores de velocidades de forma a atender os objetivos da animação. Essa abordagem vem sendo desenvolvida no Departamento de Ciência da Computação da UFJF nos últimos anos pelos orientadores deste trabalho ([Renhe et al., 2019](#); [Vieira et al., 2019](#); [Ribeiro et al., 2018](#)), tendo como proposta central a utilização de tensores definidos positivos de segunda ordem para controle de fluidos. O método utilizado é uma adaptação de [Stam \(1999\)](#), onde a velocidade e densidade do campo vetorial são discretizadas em uma representação Euleriana. O campo tensorial é inserido na formulação matemática de forma que ele altere localmente o momento linear do fluido. No entanto, os trabalhos anteriores focam em desenvolver campos tensoriais como uma ferramenta de controle do escoamento do fluido, enquanto neste trabalho esse objetivo é secundário. Os campos tensoriais aqui apresentados possuem o objetivo principal de diminuir o divergente do campo vetorial, mesmo que com isso o fluxo original seja afetado pelos novos tensores.

Além dos trabalhos anteriores desenvolvidos pelo departamento, o trabalho de [Dassios et al. \(2002\)](#) teve um papel fundamental, tanto na fundamentação teórica, ao tratar da existência e unicidade da decomposição anisotrópica de Helmholtz, quanto ao desenvolvimento dessa decomposição para o funcionamento do método proposto.

Os trabalhos [Haeser et al. \(2008\)](#); [Ingber \(1993\)](#) foram fundamentais para a escolha da meta-heurística utilizada neste trabalho. Em [Haeser et al. \(2008\)](#), o autor aprofunda nos aspectos teóricos do *Simulated Annealing* explicando como o algoritmo se relaciona com Cadeias de Markov, o algoritmo de Metropolis, e a distribuição de Boltzmann, que são os principais pontos explorados pelos algoritmos desenvolvidos nesse trabalho. Em [Ingber \(1993\)](#) o autor analisa, tanto teoricamente quanto na aplicação

prática, diversas versões da meta-heurística aqui empregada. O autor deixa claro pontos positivos e negativos do emprego dessas diferentes versões do *Simulated Annealing*. Além disso, o autor também apresenta alguns pontos que precisam de atenção ao se desenvolver a meta-heurística como, por exemplo, o cuidado que se deve ter quanto à variação de temperatura para manutenção do equilíbrio de aceitação probabilística do sistema.

3 Fundamentação Teórica

Para contextualização do problema a ser trabalhado neste documento, se faz necessário uma introdução teórica dos conceitos que aqui serão apresentados. Será necessário também uma revisão dos trabalhos relevantes, que irão contribuir para a interpretação dos resultados.

3.1 Decomposição de Helmholtz

A projeção anisotrópica de Helmholtz é fundamental para o desenvolvimento do método proposto neste trabalho dado que é através dela que obtemos um campo vetorial livre de divergente. No entanto, é necessário apresentar a decomposição clássica de Helmholtz, ou decomposição isotrópica, que serve de base teórica para a projeção anisotrópica.

3.1.1 Decomposição Isotrópica

Dado um domínio limitado e regular $\mathbf{D} \in \mathbb{R}^3$, e um campo vetorial arbitrário $\mathbf{f}(\mathbf{r})$ contínuo em $\bar{\mathbf{D}} = \mathbf{D} \cup \partial\mathbf{D}$, possuindo derivadas primeiras contínuas em \mathbf{D} . Então a decomposição isotrópica de Helmholtz é descrita por:

$$\mathbf{f}(\mathbf{r}) = \nabla\phi(\mathbf{r}) + \nabla \times \mathbf{A}(\mathbf{r}), \quad (3.1)$$

e

$$\nabla \cdot \mathbf{A}(\mathbf{r}) = 0, \quad (3.2)$$

onde $\phi(\mathbf{r})$ é um campo escalar, chamado de “potencial escalar”, e $\mathbf{A}(\mathbf{r})$ é um campo vetorial, chamado também de “vetor potencial”. A parte $\nabla\phi$ da Equação (3.1) é irrotacional, enquanto a parte $\nabla \times \mathbf{A}$ é solenoidal. O ponto principal da utilização da decomposição de Helmholtz é sua conclusão, onde é afirmado que o campo vetorial \mathbf{f} pode ser reconstruído a partir dos invariantes escalares e vetoriais de seu gradiente, via Equação (3.1), e as

seguintes equações:

$$\phi(\mathbf{r}) = \nabla \cdot \mathbf{u}(\mathbf{r}), \quad (3.3)$$

e

$$\mathbf{A}(\mathbf{r}) = -\nabla \times \mathbf{u}(\mathbf{r}), \quad (3.4)$$

onde, $\mathbf{u}(\mathbf{r})$ pode ser obtido com a solução da seguinte equação de Poisson em \mathbf{D} :

$$\Delta \mathbf{u}(\mathbf{r}) = \mathbf{f}(\mathbf{r}). \quad (3.5)$$

Além disso, a decomposição isotrópica de Helmholtz (Eq.(3.1)) é única a menos do aditivo de uma função harmônica, ou seja, podemos reescrever a equação da seguinte forma:

$$\mathbf{f}(\mathbf{r}) = [\nabla \phi(\mathbf{r}) + \nabla \mathbf{v}(\mathbf{r})] + [\nabla \times \mathbf{A}(\mathbf{r}) - \nabla \mathbf{v}(\mathbf{r})], \quad (3.6)$$

onde

$$\Delta \mathbf{v}(\mathbf{r}) = 0, \quad (3.7)$$

em \mathbf{D} .

3.1.2 Decomposição Anisotrópica

A reconstrução e unicidade também é obtida através da decomposição anisotrópica de Helmholtz, como mostrado por [Dassios et al. \(2002\)](#). A decomposição é dada por:

$$\mathbf{f}(\mathbf{r}) = \nabla_{\mathbf{T}} \phi(\mathbf{r}) + \nabla_{\mathbf{S}} \times \mathbf{A}(\mathbf{r}). \quad (3.8)$$

Agora considere a aplicação do \mathbf{S} -divergente e o \mathbf{T} -rotacional a esse campo $\mathbf{f}(\mathbf{r})$, após a decomposição de Helmholtz:

$$\nabla_{\mathbf{T}} \cdot \mathbf{f}(\mathbf{r}) = \nabla_{\mathbf{T}} \cdot \nabla_{\mathbf{T}} \phi(\mathbf{r}) + \nabla_{\mathbf{T}} \cdot \nabla_{\mathbf{S}} \times \mathbf{A}(\mathbf{r}), \quad (3.9)$$

$$\nabla_{\mathbf{S}} \cdot \mathbf{f}(\mathbf{r}) = \nabla_{\mathbf{S}} \cdot \nabla_{\mathbf{T}} \phi(\mathbf{r}) + \nabla_{\mathbf{S}} \cdot \nabla_{\mathbf{S}} \times \mathbf{A}(\mathbf{r}). \quad (3.10)$$

O objetivo é encontrar um campo vetorial livre de divergente através destas decom-

posições. Se ambos $\mathbf{S} = \mathbf{I}$ e $\mathbf{T} = \mathbf{I}$, a decomposição é $\nabla \cdot \mathbf{f}(\mathbf{r}) = \nabla \cdot \nabla \phi(\mathbf{r}) + \nabla \cdot \nabla \times \mathbf{A}(\mathbf{r}) = \nabla \cdot \nabla \phi(\mathbf{r})$, e o campo vetorial do termo $\nabla \times \mathbf{A}(\mathbf{r})$ é livre de divergente em respeito à base canônica.

Nosso interesse é usar campos tensoriais de forma que $\mathbf{S} \neq \mathbf{I}$ ou $\mathbf{T} \neq \mathbf{I}$. Primeiramente, considere $\mathbf{T} = \mathbf{I}$ e $\mathbf{S} \neq \mathbf{I}$, reduzindo a Equação (3.10) para:

$$\nabla_{\mathbf{S}} \cdot \mathbf{f}(\mathbf{r}) = \nabla_{\mathbf{S}} \cdot \nabla \phi(\mathbf{r}) + \nabla_{\mathbf{S}} \cdot \nabla_{\mathbf{S}} \times \mathbf{A}(\mathbf{r}) = \nabla_{\mathbf{S}} \cdot \nabla \phi(\mathbf{r}),$$

que pode gerar um campo vetorial livre de divergente $\nabla_{\mathbf{S}} \times \mathbf{A}(\mathbf{r})$. No entanto, esse campo é solenoidal em respeito ao campo tensorial $\mathbf{S}(\mathbf{r})$. Alternativamente, ao definir $\mathbf{T} \neq \mathbf{I}$ e $\mathbf{S} = \mathbf{I}$, obtém-se:

$$\nabla \cdot \mathbf{f}(\mathbf{r}) = \nabla \cdot \nabla_{\mathbf{T}} \phi(\mathbf{r}) + \nabla \cdot \nabla \times \mathbf{A}(\mathbf{r}) = \nabla \cdot \nabla_{\mathbf{T}} \phi(\mathbf{r}),$$

que pode gerar um campo livre de divergente $\nabla \times \mathbf{A}(\mathbf{r})$, solenoidal em respeito à base canônica. Desta forma, a projeção anisotrópica $P_{\mathbf{T}}(\cdot)$ de um campo vetorial arbitrário $\mathbf{f}(\mathbf{r})$ restringido pelo campo tensorial $\mathbf{T}(\mathbf{r})$ é obtida solucionando a equação \mathbf{T} -Poisson:

$$\nabla \cdot \nabla_{\mathbf{T}} \phi(\mathbf{r}) = \nabla \cdot \mathbf{f}(\mathbf{r}), \quad (3.11)$$

de modo a obter o campo vetorial projetado como:

$$P_{\mathbf{T}}(\mathbf{f}(\mathbf{r})) = \mathbf{f}(\mathbf{r}) - \nabla_{\mathbf{T}} \phi(\mathbf{r}), \quad (3.12)$$

que é livre de divergente em respeito a base canônica, sujeito à anisotropia de \mathbf{T} .

3.2 Campos Tensoriais

Tensores são estruturas matemáticas que são utilizadas para representar a relação entre objetos algébricos e grandezas matemáticas, físicas ou estatísticas. Essas estruturas variam sua forma de acordo com sua ordem, que por sua vez, varia de acordo com a dimensão. No contexto deste trabalho, o campo tensorial é uma função que associa a cada ponto do espaço um tensor de segunda ordem, localmente representados por matrizes

3×3 , positivas e definidas.

Um problema ao se trabalhar com tensores é a representação geométrica dessas estruturas matemáticas. Seguindo o trabalho de [Kindlmann \(2004\)](#), procurando evitar problemas de assimetria e ambiguidade visual, os tensores serão representados por superfícies quádricas. Um fator importante para essa representação de tensores são seus autovalores e autovetores, que ditarão seu formato e orientação respectivamente. A variação entre os autovalores do tensor ditará sua anisotropia, que possuirá uma representação linear, planar ou esférica. Sejam os autovalores λ_1 , λ_2 e λ_3 da matriz diagonal A de autovalores do tensor T , faz-se a seguinte relação:

- $\lambda_1 = \lambda_2 = \lambda_3$, tensor esférico, isotrópico
- $\lambda_1 > \lambda_2 = \lambda_3$, tensor linear
- $\lambda_1 = \lambda_2 > \lambda_3$, tensor planar

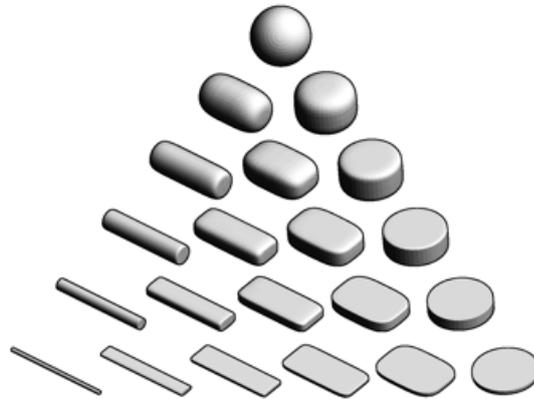


Figura 3.1: Formatos dos tensores, com glifos quádricos, variando os autovalores ([Kindlmann, 2004](#)).

Neste trabalho, a relação imposta entre o campo vetorial e o campo tensorial é a mesma descrita por [Vieira et al. \(2019\)](#). O tensor funciona como um modulador da velocidade e direção do fluido. O tensor altera essas propriedades localmente de acordo com a magnitude de seus autovalores e sua orientação em relação ao vetor correspondente. Os autovalores são os principais fatores de aceleração e desaceleração, caso o vetor esteja perfeitamente alinhado com o tensor. Caso contrário, o vetor será forçado à orientação do tensor. Portanto, o tensor também será um fator de alteração da direção do fluxo do fluido.

3.3 Simulated Annealing

Simulated Annealing, ou Recozimento Simulado, é um método probabilístico proposto por [Kirkpatrick et al. \(1983\)](#). Surgiu no contexto da mecânica estatística, sendo uma técnica com o propósito de solucionar problemas de otimização combinatória ([Haeser et al., 2008](#)). Esse método simula o funcionamento de um processo físico onde um material é levado a uma alta temperatura e é lentamente resfriado até que eventualmente se solidifique. Isso acontece quando ele se encontra em uma configuração de baixa energia ([Bertsimas et al., 1993](#)). Para que o *Simulated Annealing* seja utilizado, alguns elementos básicos devem ser verificados:

1. Um conjunto finito de estados E ;
2. Uma função custo $C \in \mathbb{R}$ e definida em E ;
3. Seja e um estado $\in E$, então $\forall e, \exists$ um conjunto de estados $E(e) \subset E - \{e\}$ chamado de conjunto de estados vizinhos a e ;
4. Uma função de arrefecimento da temperatura T ;
5. Um estado inicial $i \in E$.
6. Uma condição de parada.

Com os elementos acima satisfeitos, o algoritmo de *Simulated Annealing* consiste em uma cadeia de Markov discreta e não-homogênea ([Bertsimas et al., 1993](#)). A evolução do algoritmo acontece da seguinte forma: a partir de um estado inicial \mathbf{i} , é gerado, aleatoriamente, um novo estado \mathbf{v} . Uma vez que esse novo estado tenha sido gerado, o novo valor da função custo é verificado, e este valor é comparado com o do último estado aceito. Se o custo for menor, então este novo estado é aceito, e a busca se reinicia a partir dele. Caso o valor da função custo seja maior, uma função de probabilidade ditará a aceitação do estado. Essa meta-heurística possui várias versões e modificações presentes na literatura, como pode ser visto no trabalho de [Ingber \(1993\)](#). Portanto, existem diversas funções de probabilidade que podem ser utilizadas nessa etapa do processo. O algoritmo escolhido para geração dos campos tensoriais foi o *Boltzmann Annealing*, onde a função de aceitação probabilística de novos campos é dada pela seguinte equação:

$$\mathbf{P} = \exp\left(\frac{-\Delta\mathbf{C}}{T}\right), \quad (3.13)$$

onde $\Delta\mathbf{C}$ representa a diferença de “energia”, ou custo entre o novo estado e o último estado aceito, e \mathbf{T} faz referência à temperatura dada pela Equação (3.15). O valor de \mathbf{P} deverá ser maior ou igual a um número gerado aleatoriamente entre 0 e 1 para que o campo seja aceito. Essencialmente, isso é a distribuição de Boltzmann contribuindo para a função de particionamento da mecânica estatística do sistema (Ingber, 1993). Neste trabalho, os estados do *Simulated Annealing* são campos tensoriais positivos definidos. A função custo, ou parâmetro de comparação, é a média quadrática do divergente do campo projetado, dada por:

$$S = \frac{\sum_{n=1}^m (\text{divergente}_n)^2}{m}, \quad (3.14)$$

onde m é o tamanho do campo vetorial. A cada iteração do algoritmo uma modificação é feita no último campo tensorial aceito, e assim os passos de projeção são refeitos com esse novo campo gerado. O resultado dessa nova execução é então comparado com o resultado do estado anterior, e assim é decidido se o novo campo será aceito pelo algoritmo.

Em conjunto com a função de probabilidade de aceitação, há também uma função de arrefecimento da temperatura. O objetivo principal dessa função de arrefecimento é colaborar para que o algoritmo evite ótimos locais, tendo assim uma relação direta com a função de probabilidade de aceitação do algoritmo. A temperatura inicial deve ser suficientemente grande para que, inicialmente, o algoritmo consiga aceitar a maioria dos novos estados gerados e, com o passar do tempo, essa probabilidade de aceitação se torne cada vez menor (Haeser et al., 2008). A função de arrefecimento da temperatura utilizada é dada por uma generalização feita por Kirkpatrick et al. (1983) a partir do algoritmo de Metropolis Monte Carlo, dada por:

$$T(t) = \frac{1}{\log(t)}, \quad (3.15)$$

onde t é uma medida de tempo artificial para o resfriamento da temperatura. Essa medida de tempo pode ser representada por um número de iterações arbitrárias, ou realmente

uma quantia arbitrária de tempo. Nota-se que, ao relacionar as equações (3.15) e (3.13), a temperatura está diretamente relacionada com a probabilidade de aceitação dos estados, reafirmando que a temperatura inicial T_0 tem um papel importante para que o *Simulated Annealing* evite ótimos locais.

4 Método Proposto

O método utilizado é altamente dependente de parâmetros, tendo como entradas um campo vetorial, um campo tensorial inicial, e os parâmetros de execução do *Simulated Annealing*. Para isso, temos as seguintes etapas:

- Campo vetorial: na primeira etapa do método, o usuário deverá definir qual será o campo vetorial utilizado. O campo é fixo durante todo o processo do algoritmo.
- Campo tensorial inicial: na segunda etapa do processo deverá ser definido qual o campo tensorial utilizado para a execução inicial, ou seja, a execução que obterá o estado que será o ponto de partida do *Simulated Annealing*.
- Parâmetros de alteração do campo tensorial: após definido o campo tensorial inicial, deverão ser definidos os parâmetros que irão alterar o campo tensorial a cada iteração.
- Parâmetros de execução do *Simulated Annealing*: a meta-heurística possui seus próprios parâmetros de execução como, por exemplo, parâmetros relacionados à temperatura, e à função de probabilidade.
- Execução do *Simulated Annealing*: após a definição dos parâmetros e estados iniciais do método, o *Simulated Annealing* itera sobre o campo tensorial. A cada iteração o campo tensorial é modificado de acordo com os parâmetros definidos, sendo executado o passo de projeção do campo vetorial com o novo campo tensorial gerado. A cada iteração a meta-heurística decide se o novo campo tensorial possui um resultado aceitável, ou não, com relação à diminuição de resíduos da projeção.

Em todo o processo, os campos tensoriais utilizados deverão ser definidos positivos, portanto, as modificações feitas no campo deverão manter essa condição. As modificações feitas são uma série de escalonamentos e rotações aleatórias em tensores escolhidos aleatoriamente. No entanto, os valores aleatórios das variáveis que modificam o campo possuem

limites inferior e superior pré-estabelecidos. Além disso, os tensores presentes nas bordas do campo não são alterados. A não alteração desses tensores acontece para evitar problemas com condições de borda. As modificações são baseadas no seguinte princípio: se $\mathbf{S} \in R^2$ é uma matriz positiva definida. Essa matriz \mathbf{S} possui autovalores reais tais que $\lambda_1, \lambda_2 > 0$, e autovetores reais e_1 e e_2 , supostos normalizados, podendo ser escrita como:

$$S = \lambda_1 e_1 e_1^T + \lambda_2 e_2 e_2^T \equiv \lambda_1 \begin{pmatrix} e_{11} \\ e_{12} \end{pmatrix} \begin{pmatrix} e_{11} & e_{12} \end{pmatrix} + \lambda_2 \begin{pmatrix} e_{21} \\ e_{22} \end{pmatrix} \begin{pmatrix} e_{21} & e_{22} \end{pmatrix}.$$

Portanto, dada uma matriz de rotação \mathbf{R} , a operação $\mathbf{R}\mathbf{S}\mathbf{R}^T$, pode ser descrita como:

$$\mathbf{R}\mathbf{S}\mathbf{R}^T = \lambda_1 \mathbf{R}e_1 e_1^T \mathbf{R}^T + \lambda_2 \mathbf{R}e_2 e_2^T \mathbf{R}^T, = \lambda_1 (\mathbf{R}e_1) (\mathbf{R}e_1)^T + \lambda_2 (\mathbf{R}e_2) (\mathbf{R}e_2)^T.$$

Consequentemente, a matriz $\mathbf{R}\mathbf{S}\mathbf{R}^T$ é simétrica, positiva definida com autovalores λ_1, λ_2 e autovetores $(\mathbf{R}e_1)$ e $(\mathbf{R}e_2)$ que satisfaz:

$$(\mathbf{R}e_i) \cdot (\mathbf{R}e_i) \equiv (\mathbf{R}e_i)^T (\mathbf{R}e_i) = (e_i^T \mathbf{R}^T) (\mathbf{R}e_i) = 1, \quad i = 1 \text{ e } 2,$$

e

$$(\mathbf{R}e_1) \cdot (\mathbf{R}e_2) = (e_1^T \mathbf{R}^T) (\mathbf{R}e_2) = 0,$$

uma vez que uma matriz de rotação satisfaz $\mathbf{R}^T \mathbf{R} = \mathbf{I}$. Podemos escalar e rotacionar a matriz \mathbf{S} da seguinte forma:

$$(\sqrt{\alpha}\mathbf{R}) S (\sqrt{\alpha}\mathbf{R})^T = \lambda_1 (\sqrt{\alpha}\mathbf{R}) e_1 e_1^T (\sqrt{\alpha}\mathbf{R})^T + \lambda_2 (\sqrt{\alpha}\mathbf{R}) e_2 e_2^T (\sqrt{\alpha}\mathbf{R})^T,$$

para obter:

$$(\sqrt{\alpha}\mathbf{R}) S (\sqrt{\alpha}\mathbf{R})^T = (\alpha\lambda_1) (\mathbf{R}e_1) (\mathbf{R}e_1)^T + (\alpha\lambda_2) (\mathbf{R}e_2) (\mathbf{R}e_2)^T,$$

onde $\alpha \in \mathbb{R}_{\geq 0}$. Sendo assim, $(\sqrt{\alpha}R)S(\sqrt{\alpha}R)^\top$ é uma matriz simétrica com autovalores $(\alpha\lambda_1)$ e $(\alpha\lambda_2)$ e autovetores (Re_1) e (Re_2) .

4.1 Modelo Computacional

Os dois algoritmos desenvolvidos são distintos em seu funcionamento, mas como são fundamentados na mesma meta-heurística, alguns dos parâmetros são comuns a ambos. Porém, mesmo tendo parâmetros comuns, alguns dos parâmetros possuem valores diferentes para cada algoritmo. Um exemplo de parâmetro comum a ambos, e que possui o mesmo valor, é o limite de rotação dos tensores a cada iteração. Esse parâmetro possui um valor entre $-10^\circ \leq \theta \leq 10^\circ$. Outro exemplo, é o critério de parada dos algoritmos, ambos possuem um limite de iterações que ao ser atingida, os algoritmos param a execução e apresentam os resultados obtidos. O valor utilizado para esse critério de parada foi o limite de 30 mil iterações dos algoritmos. Para esclarecimento do fluxo de funcionamento dos algoritmos dois fluxogramas foram desenvolvidos, eles podem ser observados nas Figuras (4.1, 4.2)

Algoritmo 1

Apesar de ambos os algoritmos se basearem na mesma meta-heurística, algumas modificações foram feitas, não só em questão de valores dos parâmetros utilizados, mas também no comportamento do algoritmo. Em relação aos parâmetros que são comuns a ambos os algoritmos, há também o fator de escalonamento do tensor. Porém o valor para esse parâmetro difere entre os algoritmos. No primeiro algoritmo o fator de escalonamento varia da seguinte forma: $0,7 \leq \alpha \leq 1,3$. A temperatura usada no *Simulated Annealing* é constante, e possui um valor igual a $1e^{-9}$. Sendo assim, a variação na probabilidade de aceitação do algoritmo é a variação da média quadrática do divergente, além do número entre 0 e 1 gerado aleatoriamente a cada iteração que necessita da avaliação da função de probabilidade.

Em relação ao comportamento do algoritmo, uma característica importante é quanto a escolha do tensor que será modificado, a seleção é aleatória dentro dos limites de borda estabelecidos. Esse limite de borda está relacionado com a limitação imposta aos algoritmos para não modificarem tensores na borda do campo vetorial para evitar

problemas com condições de contorno. Se a alteração do tensor resultou em uma média quadrática do divergente menor que a do campo atualmente aceito, o mesmo tensor é selecionado para modificação e a alteração será a mesma. No caso do mesmo tensor ter sido selecionado mais de 25 vezes consecutivas, as alterações começam a ser aplicadas duas vezes, ou seja, o tensor é escalonado e rotacionado duas vezes na mesma iteração. Se o campo gerar um divergente pior, mesmo que o novo campo seja aceito pela função de probabilidade, ou se um mesmo tensor já foi selecionado mais de 50 vezes, um novo tensor será sorteado, assim como um novo fator de escalonamento e ângulo de rotação.

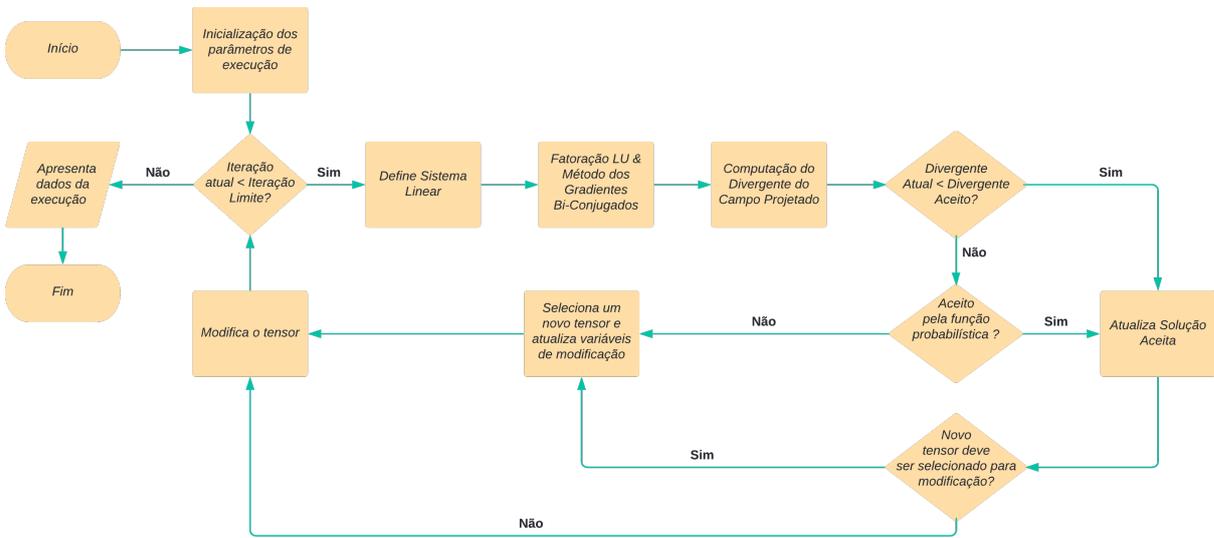


Figura 4.1: Fluxograma de funcionamento do Algoritmo 1

Algoritmo 2

Nesse segundo algoritmo o fator de escalonamento é menor que o anterior, obedecendo o seguinte limite: $0,85 \leq \alpha \leq 1,15$. O fator de escalonamento foi reduzido para evitar quebrar o equilíbrio de probabilidades do algoritmo, evitando mudanças bruscas na variação da média quadrática dos divergentes do campo. Essa preocupação é válida, pois diferentemente do algoritmo anterior, agora há o arrefecimento da temperatura. Sendo assim, há mais um fator variável na probabilidade de aceitação dos campos. Para melhor controle da probabilidade de aceitação, a temperatura não é definida diretamente, e sim pelos valores da probabilidade inicial e final, da seguinte forma:

$$t_i = \frac{-1}{\log(p_i)}, \quad (4.1)$$

e

$$t_f = \frac{-1}{\log(p_f)}, \quad (4.2)$$

onde t_i , p_i , t_f e p_f , são temperatura inicial, probabilidade inicial, temperatura final, e probabilidade final respectivamente. As probabilidades inicial e final possuem um valor entre 0 e 1, e a probabilidade final é menor que a probabilidade inicial. A função de arrefecimento da temperatura é dada por:

$$f = \left(\frac{t_f}{t_i} \right)^{\frac{1}{ciclos-1}}, \quad (4.3)$$

onde *ciclos* é o número de vezes que a temperatura irá diminuir durante o processo do *Simulated Annealing*. Dada essa variação da temperatura, de acordo como foi apresentado, é necessário fazer um ajuste da magnitude da temperatura em relação às variações da média quadrática do divergente, para que a função de probabilidade funcione como o esperado. Esse ajuste acontece da seguinte forma:

$$P = \exp\left(\frac{-\Delta E}{T * mag}\right), \quad (4.4)$$

onde:

$$\Delta E = |S_{atual} - S_{aceito}|, \quad (4.5)$$

e

$$mag = 10^{(\log(\Delta E))}, \quad (4.6)$$

onde T é a temperatura atual, S é o valor obtido na Equação(3.14), e mag é a magnitude da variação da média quadrática do divergente que é expressa por ΔE . Note que, ao inserir a Eq.(4.1), ou Eq.(4.2), na Eq.(4.4), as variações pequenas de divergente são favorecidas, enquanto as grandes alterações são evitadas.

Outra diferença importante entre os algoritmos se dá na seleção do tensor a ser modificado. Aqui, antes de um tensor ser selecionado, o algoritmo monta uma matriz com as 10 posições de maior divergente de cada linha do campo vetorial, e uma posição é sorteada considerando a matriz gerada. No que diz respeito às seleções consecutivas de tensores, apresentada como característica do algoritmo anterior, a única diferença aqui

5 Resultados

De modo a avaliar o método proposto, foi realizada uma bateria de testes com 4 campos vetoriais, utilizando os algoritmos descritos na Seção (4.1). A implementação atual foi desenvolvida para campos 2D, mas pode ser facilmente expandida para 3D. Os campos vetoriais utilizados se diferem em tamanho e geometria sendo, que o Campo 1 e o Campo 3 possuem uma malha de 78×110 , enquanto o Campo 2 e o Campo 4 são representados usando uma malha 100×100 . Os estados iniciais de cada campo podem ser visualizados na Seção (5.1).

Os testes foram executados em um ambiente 2D baseado na proposta apresentada por [Vieira et al. \(2019\)](#), utilizando a mesma máscara assimétrica desenvolvida para encontrar a solução do T-Laplaciano. Ela é necessária para montar o sistema de equações linear e encontrar a solução da projeção do campo vetorial. O método numérico utilizado foi o dos gradientes bi-conjugados. Apesar do ambiente utilizado ter sido desenvolvido com o propósito de solucionar um problema de dinâmica de fluidos, as ferramentas utilizadas por esse trabalho são essencialmente numéricas e livres da restrição de contexto de simulação de fluidos. A máquina utilizada para realização possui os seguintes componentes:

- Processador: Intel Core i5-9400F, 6 núcleos, 6 threads e 2.90-4.10 GHz;
- Memória RAM: 16 GB, 2666 MHz
- Sistema Operacional : Windows 10 64-bit
- IDE: MATLAB 2020a 64-bit

Cada campo vetorial possui uma característica específica, desenvolvida com o objetivo de evidenciar características importantes nos campos tensoriais resultantes. Os campos vetoriais 1 e 2 possuem uma descontinuidade muito abrupta, e o divergente está concentrado em uma parte específica do campo. Essa característica serve para verificar se os algoritmos tendem a aceitar modificações em tensores que estão presentes na parte do campo com maior divergente ou se não existe essa relação. O Campo 3 é referenciado

como complexo por ter topologia e distribuição complexa do divergente. Esse campo visa evidenciar como o método se comportaria em situações mais semelhantes às obtidas ao se utilizar, por exemplo, um campo vetorial de um fluido. O Campo 4 é um caso intermediário, onde a transição não é tão brusca, mas possui divergente concentrado em uma parte específica. Com essas características, também é possível observar se os algoritmos tendem a aceitar modificações que façam com que os tensores se alinhem mais com o fluxo associado.

Ambos os algoritmos tiveram 30 mil iterações para cada um dos campos vetoriais, e também tiveram o mesmo campo tensorial inicial, o campo tensorial isotrópico mostrado na Figura (5.5). Com isso, fica evidente a distinção das duas projeções, e o resultado possível de ser alcançado com ambas. Para fins de comparação e transparência numérica, os dados brutos coletados de cada uma das etapas do método proposto (estado inicial dos campos, projeção isotrópica e projeção anisotrópica), são apresentados em tabelas no início de cada uma das próximas seções.

Nos testes realizados com a projeção anisotrópica, que podem ser visualizados na Seção (5.3), há uma divisão entre “Processo Completo” e “Campos Aceitos”, onde os dados relativos ao primeiro são referentes a todos os campos gerados pelo algoritmo, enquanto os dados relativos ao segundo são referentes aos campos que foram aceitos pelo algoritmo. Os resultados obtidos em cada teste serão discutidos mais detalhadamente na Seção (5.4), onde também será apresentada a proporção dos resultados em relação aos dados do estado inicial, sem projeção, e em relação aos dados da projeção isotrópica.

Como visto na Seção (3.3), os campos tensoriais gerados foram comparados utilizando a média quadrática do divergente do campo vetorial resultante após a projeção. Porém, os algoritmos coletaram alguns dados além da métrica de comparação dos campos, sendo eles: divergentes máximos e mínimos obtidos durante a execução do algoritmo, e a soma absoluta dos divergentes de cada iteração.

5.1 Campos Vetoriais Iniciais

Os dados apresentados na Tabela (5.1) e na Tabela (5.2) possuem o propósito de evidenciar a eficiência numérica do método. O mesmo vale para os gráficos apresentados a seguir, que

além de servirem para evidenciar a eficiência do método, também deixam claro a influência dos campos tensoriais gerados. As tabelas indicam o maior e o menor valor de divergente encontrado em cada um dos campos, o valor da média quadrática dos divergentes, e a soma absoluta dos divergentes.

Dois gráficos são apresentados para cada campo. O primeiro gráfico contém informação relativa ao divergente do campo, como mostra a Figura (5.1a), enquanto o segundo gráfico deixa claro o fluxo dos vetores do campo e, além disso, mostra a magnitude dos vetores em diferentes regiões, como mostra a Figura (5.1b).

Tabela 5.1: Dados relativos ao estado inicial dos campos 1 e 2.

	Campo 1		Campo 2	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-0,5	0	-1	0
Média quadrática do divergente	0,004854		0,021621	
Soma Absoluta do divergente	71		183	

Tabela 5.2: Dados relativos ao estado inicial dos campos 3 e 4.

	Campo 3		Campo 4	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-1,01667	0,772688	-30	2
Média quadrática do divergente	0,026789		12,488496	
Soma Absoluta do divergente	869		10616	

5.1.1 Campo 1 - Descontínuo Vertical

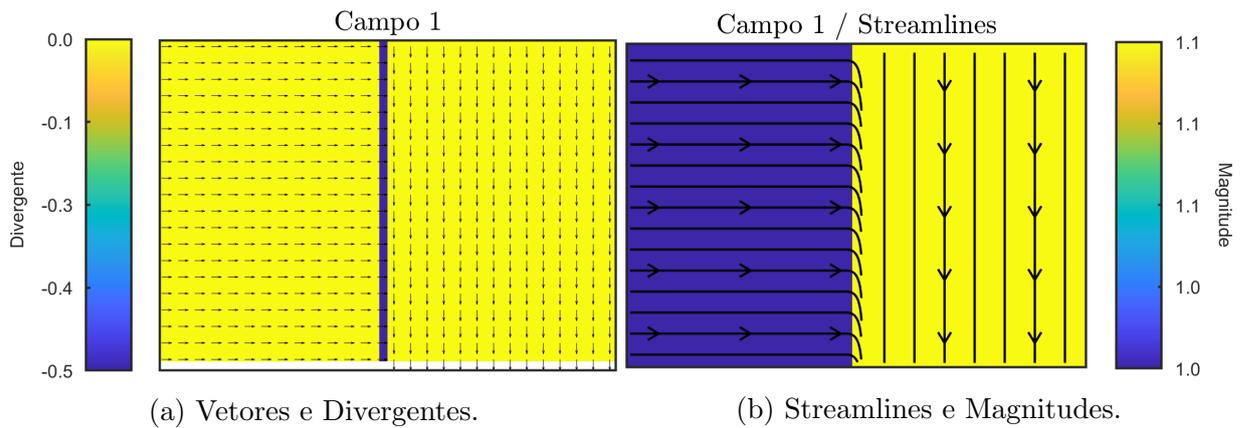


Figura 5.1: Campo 1 - Descontínuo Vertical em seu estado inicial.

5.1.2 Campo 2 - Descontínuo Diagonal

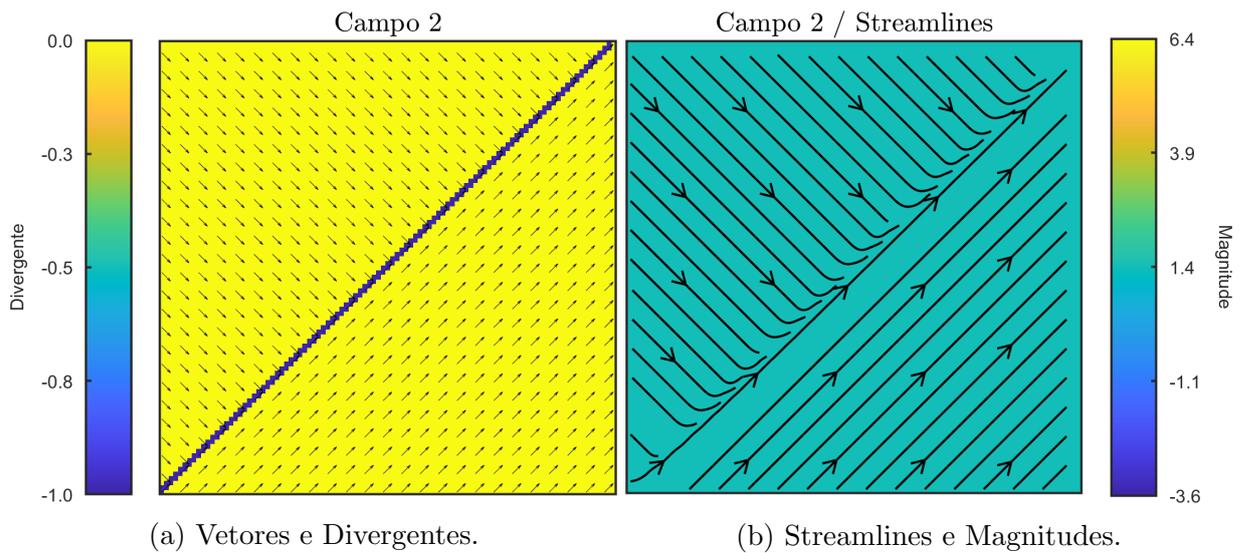


Figura 5.2: Campo 2 - Descontínuo Diagonal em seu estado inicial.

5.1.3 Campo 3 - Complexo

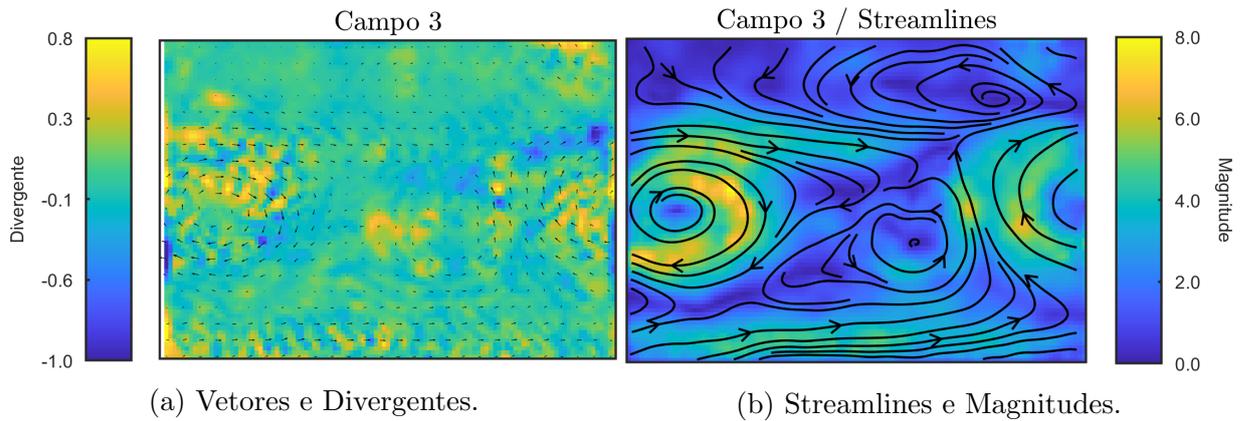


Figura 5.3: Campo 3 - Complexo em seu estado inicial.

5.1.4 Campo 4 - Radial Rotacional

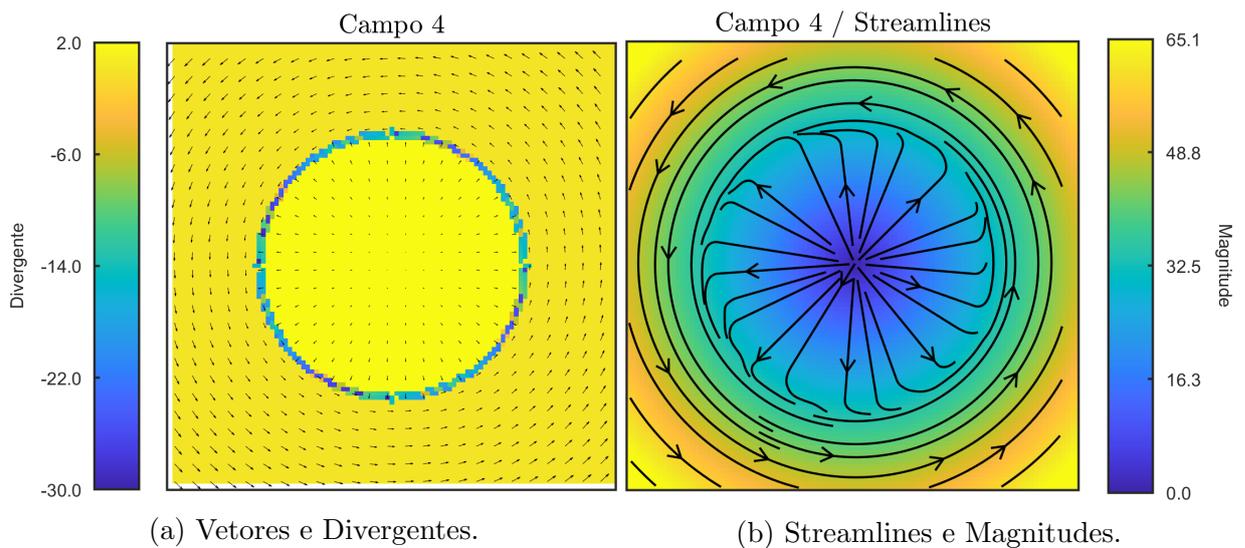


Figura 5.4: Campo 4 - Radial Rotacional em seu estado inicial.

5.2 Projeção Isotrópica

Assim como na seção anterior, os dados expostos nas Tabelas (5.3, 5.4), e os gráficos desta seção, possuem o único propósito de evidenciar a eficiência do método proposto.

Nota-se, porém, que os gráficos desta seção diferem dos gráficos da seção anterior. As primeiras figuras representam o campo vetorial irrotacional encontrado para cada um dos campos vetoriais desenvolvidos. Um exemplo pode ser visto na Figura (5.6). Esse campo, ao ser subtraído do campo original, como, por exemplo, o campo da Figura (5.1),

gera o campo projetado, que pode ser observado na Figura (5.7).

O campo tensorial utilizado para a projeção isotrópica pode ser visto na Figura (5.5). Nota-se que esse é o campo tensorial inicial para os algoritmos desenvolvidos apresentados na Seção (4.1).

Tabela 5.3: Dados relativos a projeção isotrópica dos campos 1 e 2.

	Campo 1		Campo 2	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-0,12499	0,126559	-0,25166	0,249997
Média quadrática do divergente	0,000606		0,002688	
Soma Absoluta do divergente	36		91	

Tabela 5.4: Dados relativos a projeção isotrópica dos campos 3 e 4.

	Campo 3		Campo 4	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-0,25957	0,250011	-13,5513	8,021646
Média quadrática do divergente	0,001085		1,956613	
Soma Absoluta do divergente	159		3094	

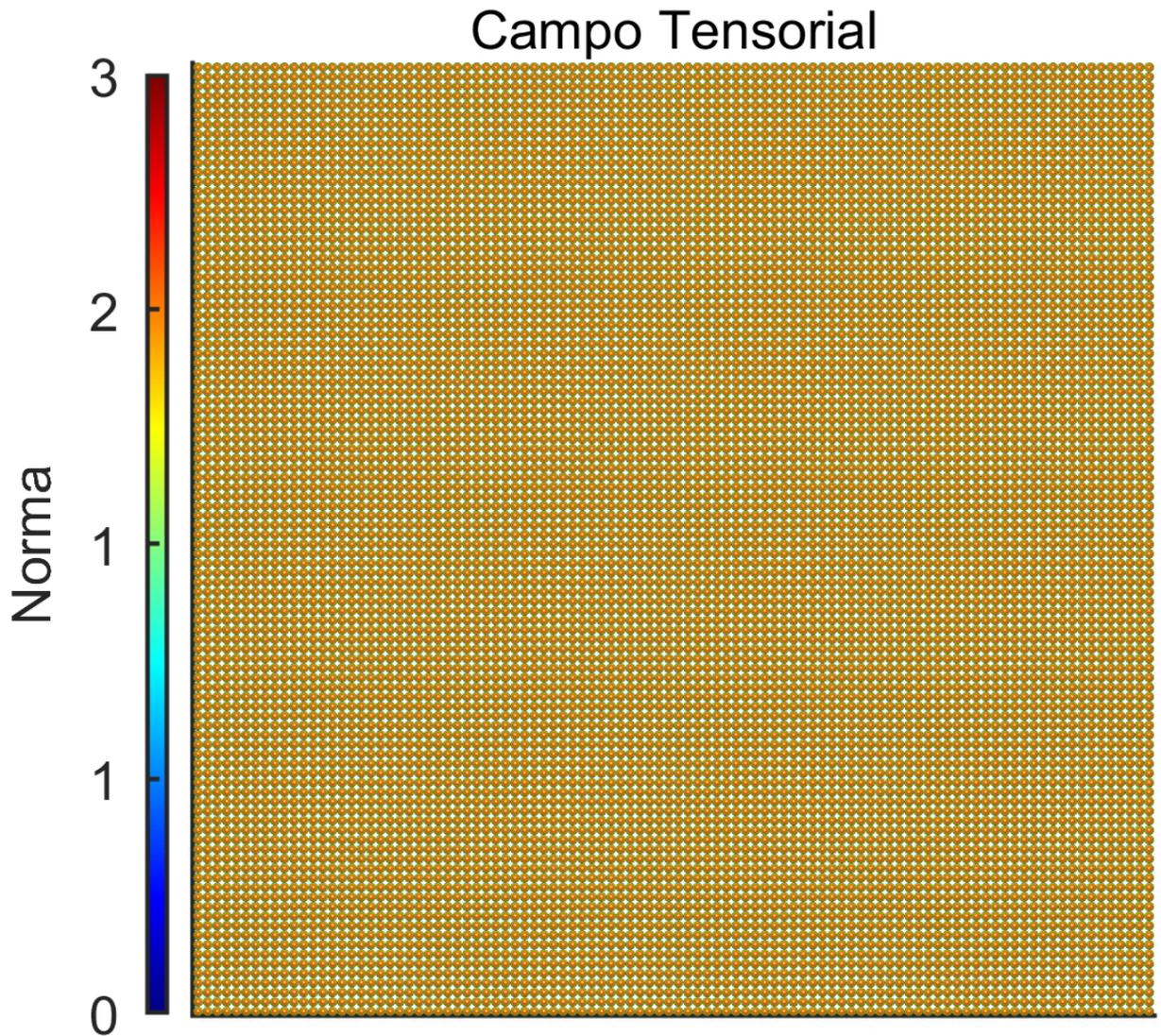


Figura 5.5: Campo Tensorial Isotrópico.

5.2.1 Campo 1 - Descontínuo Vertical

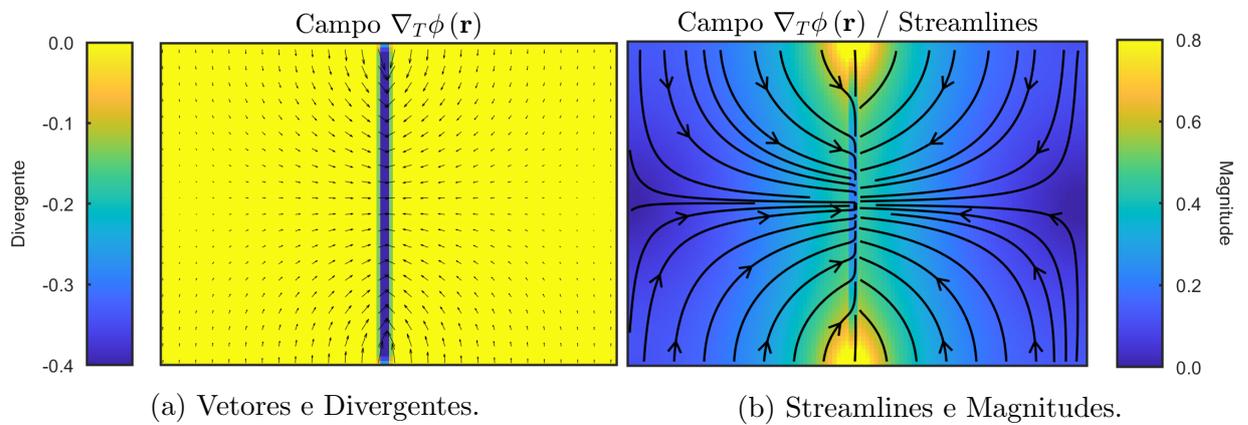


Figura 5.6: Campo vetorial T-irrotacional $\nabla_T \phi(\mathbf{r})$ para o Campo 1 com a projeção isotrópica.

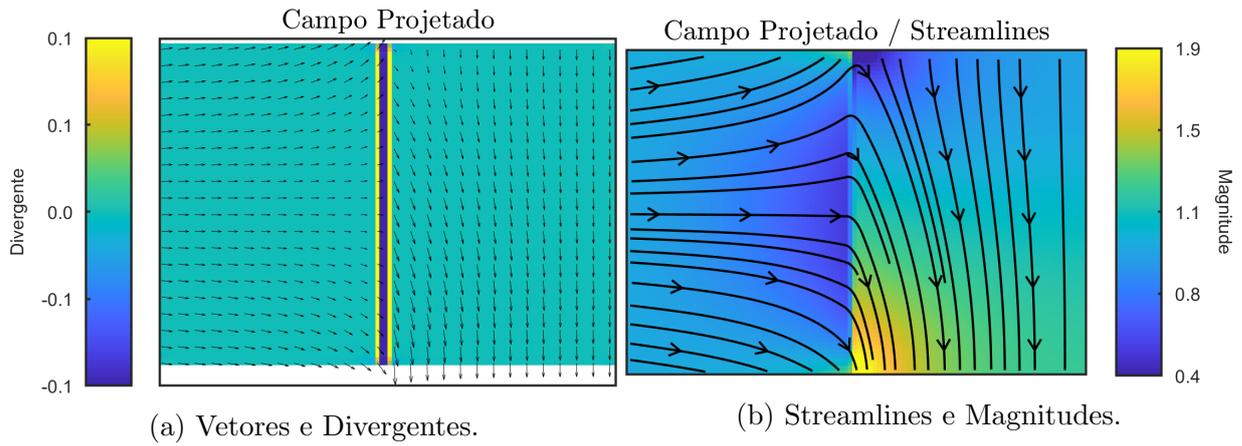


Figura 5.7: Campo 1 após projeção isotrópica.

5.2.2 Campo 2 - Descontínuo Diagonal

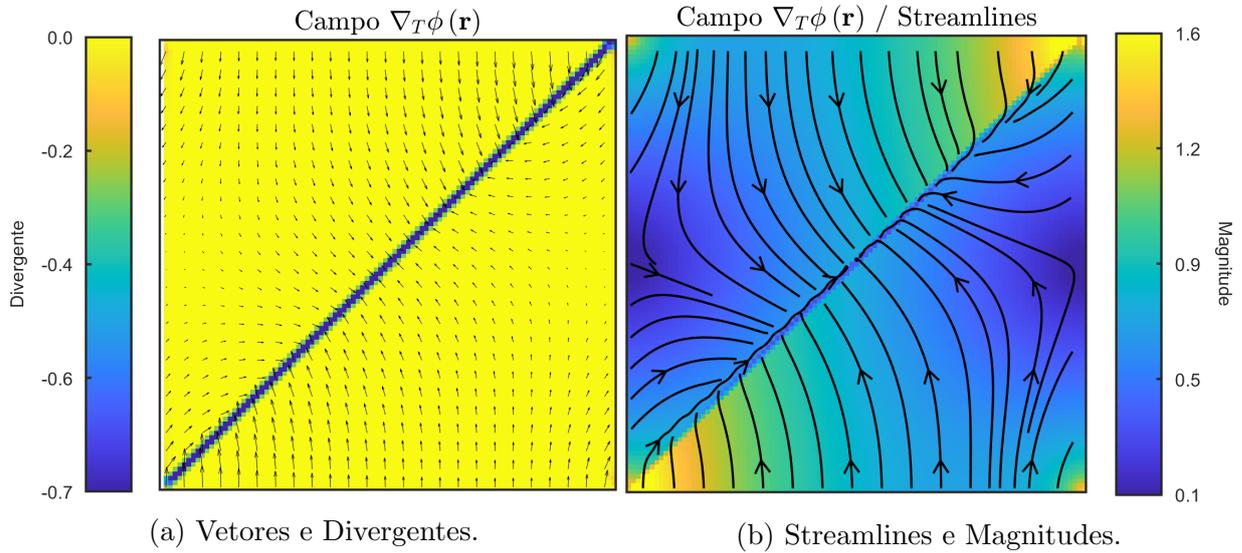


Figura 5.8: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 2 com a projeção isotrópica.

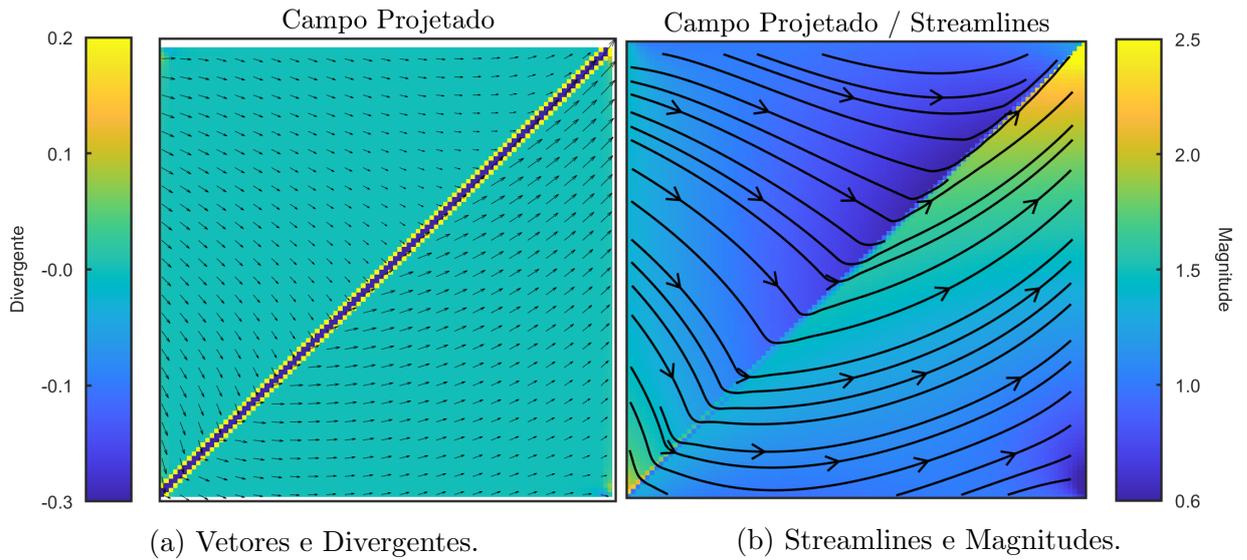


Figura 5.9: Campo 2 após projeção isotrópica.

5.2.3 Campo 3 - Complexo

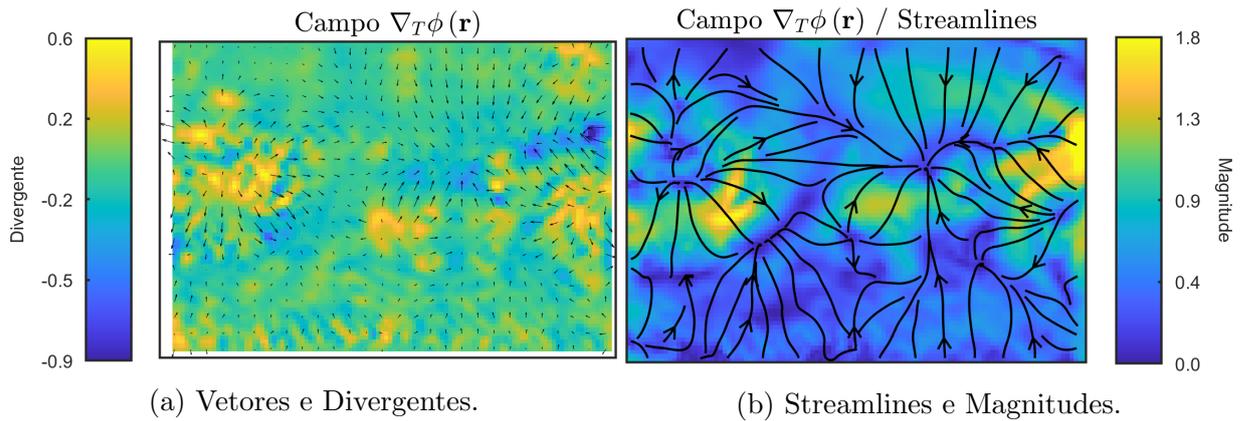


Figura 5.10: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 3 com a projeção isotrópica.

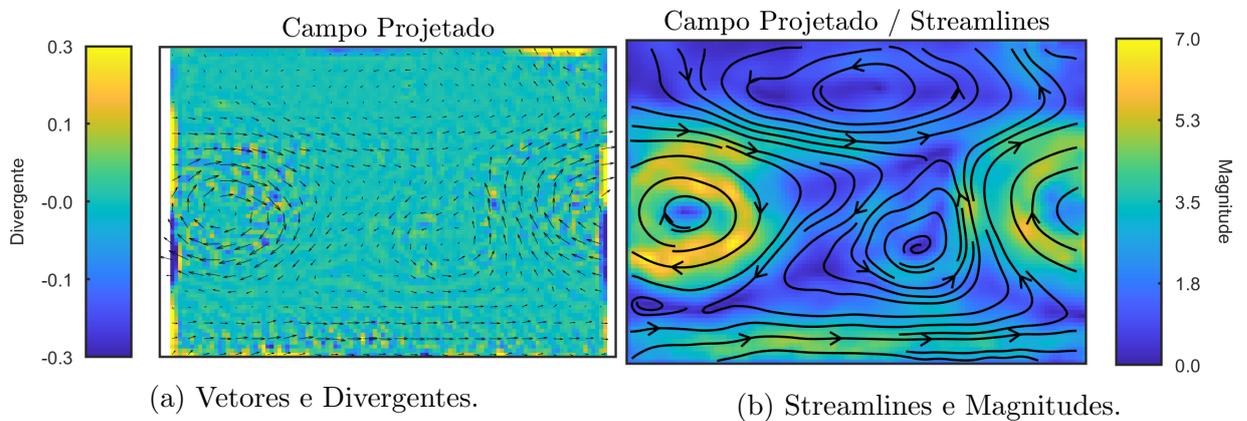


Figura 5.11: Campo 3 após projeção isotrópica.

5.2.4 Campo 4 - Radial Rotacional

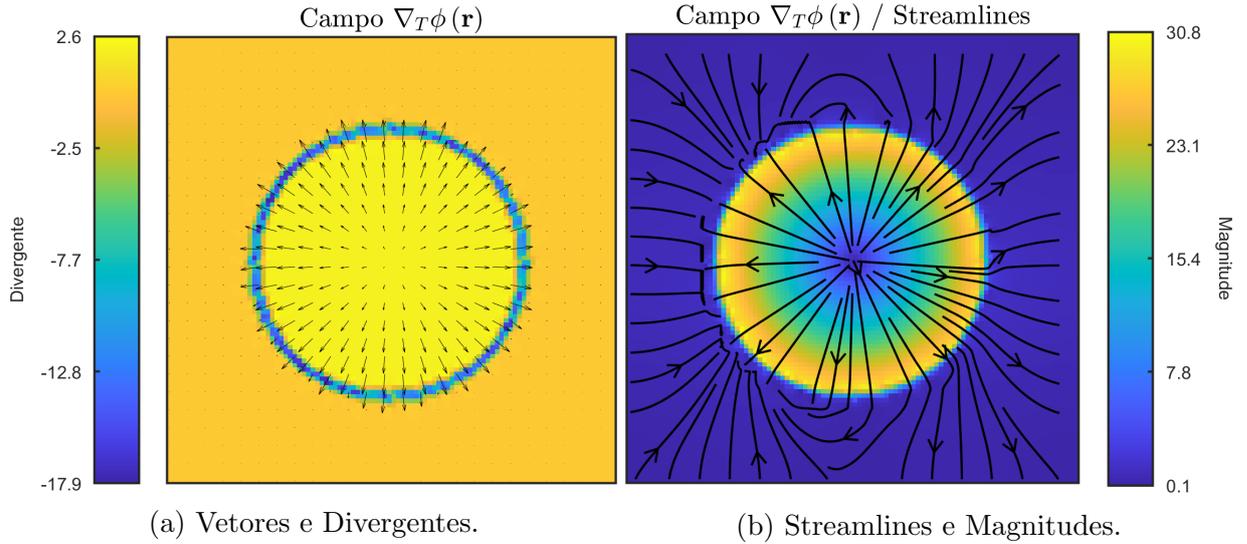


Figura 5.12: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 4 com a projeção isotrópica.

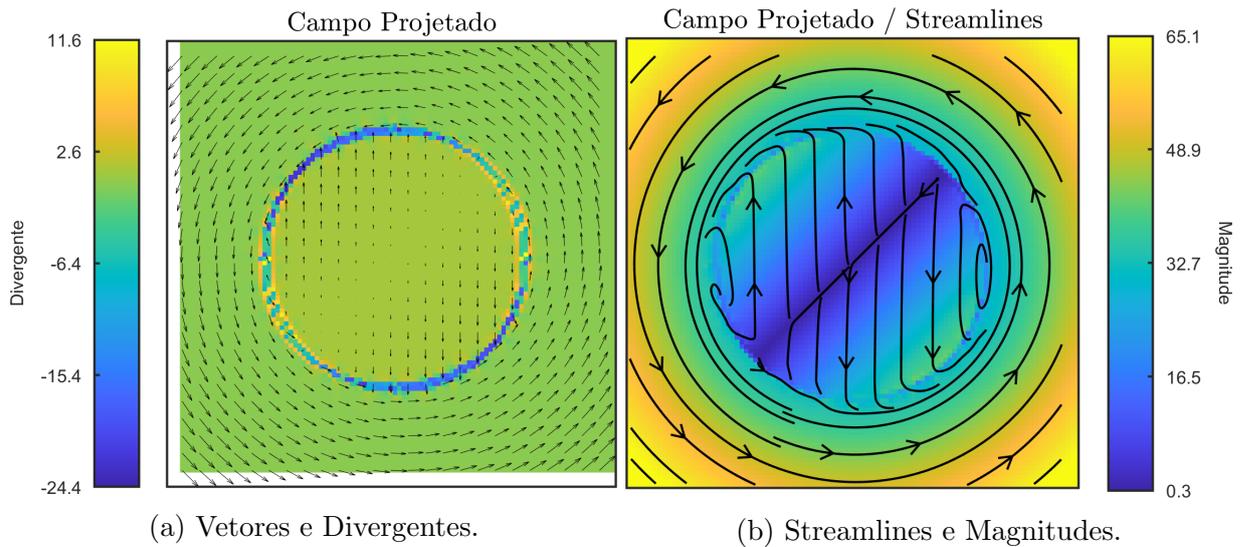


Figura 5.13: Campo 4 após projeção isotrópica.

5.3 Projeção Anisotrópica

Nesta seção são apresentados os resultados obtidos após a execução dos algoritmos para cada um dos campos apresentados na Seção (5.1). Esta seção possui subdivisões para cada um dos campos da Seção (5.1). Para cada campo são apresentados separadamente os resultados obtidos pelo Algoritmo 1, descrito na Seção (4.1), e pelo Algoritmo 2, descrito na Seção (4.1).

Quanto aos dados e figuras, existem algumas diferenças em relação à seção anterior. Nas tabelas, os valores da média quadrática e soma absoluta não são mais únicos. Aqui são apresentados os maiores e menores valores obtidos durante todo o processo de execução do algoritmo, como pode ser visto na Tabela (5.5). Além disso, é apresentada uma tabela para cada um dos campos, e em cada tabela estão contidos os dados obtidos por ambos os algoritmos. Os campos tensoriais obtidos pelos algoritmos também serão apresentados, como pode ser visto na Figura (5.16).

Quatro novas figuras foram introduzidas, referentes à evolução da média quadrática e à soma absoluta dos divergentes. As novas figuras separam os dados recolhidos em relação ao processo completo de execução, que pode ser visto, por exemplo, na Figura (5.17a), e os dados recolhidos apenas dos campos que foram aceitos, como pode ser visto, por exemplo, na Figura (5.17b). Note como as mudanças bruscas, que produzem um divergente pior, são evitadas pelos algoritmos. Esse detalhe fica mais evidente quando se compara a Figura (5.17a) e a Figura (5.17b). O algoritmo, por ser uma busca probabilística, pode piorar a média quadrática do divergente do campo vetorial, em relação ao seu estado inicial, ou seja, em relação ao resultado da projeção isotrópica. Contudo, isso acontece gradualmente, dado que os algoritmos tendem a evitar grandes alterações que pioram o divergente.

5.3.1 Campo 1 - Descontínuo Vertical

Tabela 5.5: Dados relativos ao processo completo da projeção anisotrópica do Campo 1.

	Algoritmo 1		Algoritmo 2	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-1,49462	2,046195	-0.25649	0.184882
Média quadrática do divergente	0.000183	0.001571	0.001079	0,001085
Soma Absoluta do divergente	30	40	47	48

Tabela 5.6: Dados relativos aos campos aceitos da projeção anisotrópica do Campo 1.

Dados	Algoritmo 1		Algoritmo 2	
	Mínimo	Máximo	Mínimo	Máximo
Divergente	-0,14863	0,156157	-0,25618	0,174628
Média quadrática do divergente	0,000183	0,000612	0,001079	0,001085
Soma Absoluta do divergente	30	35	47	48

Algoritmo 1

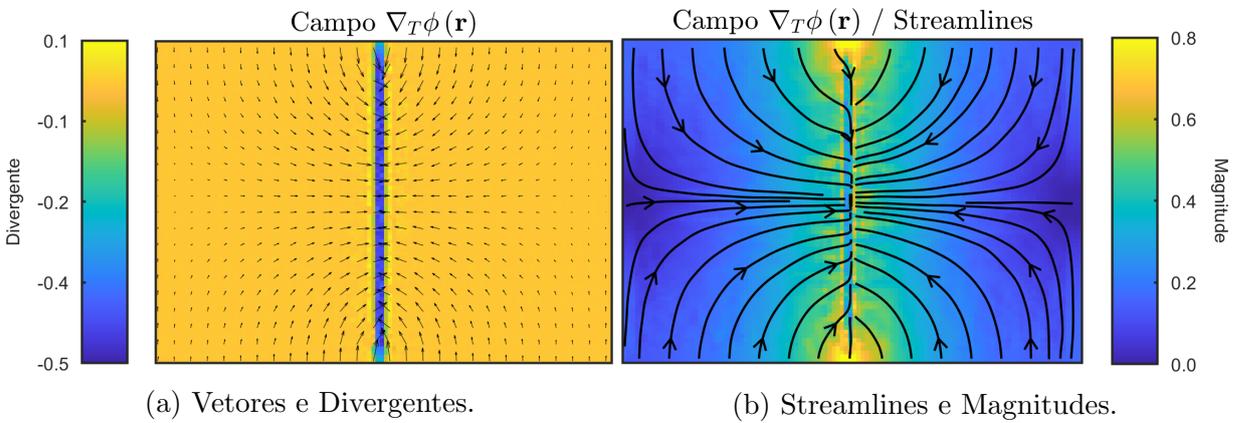


Figura 5.14: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 1 após execução do Algoritmo 1.

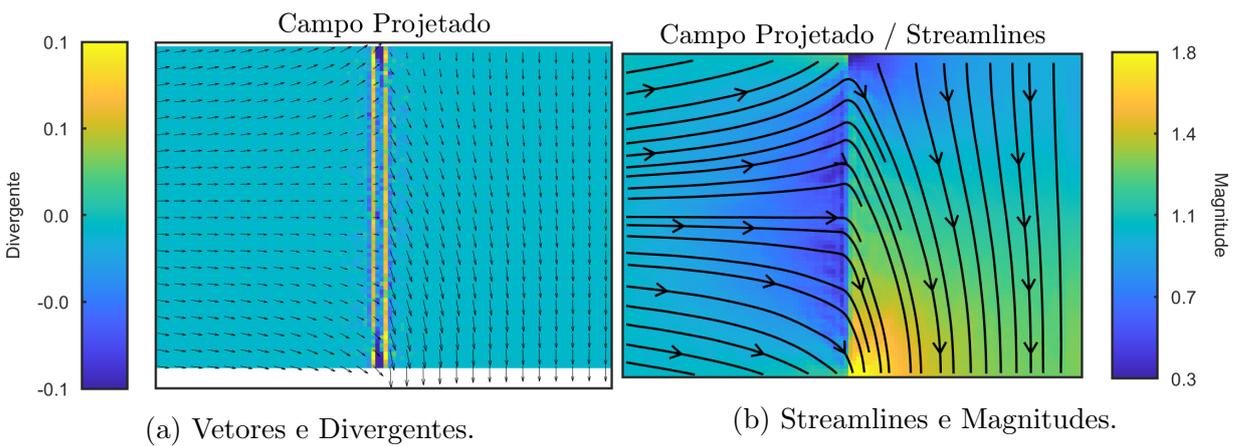


Figura 5.15: Campo 1 projetado após execução do Algoritmo 1.

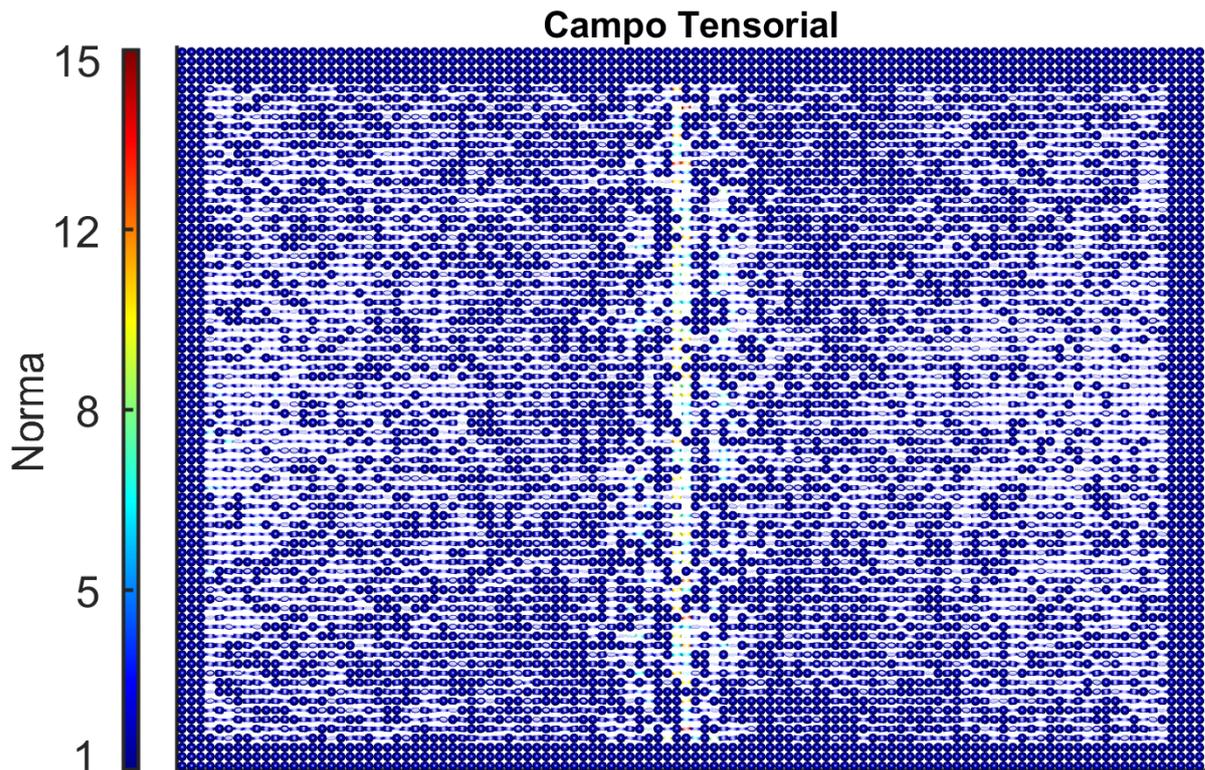


Figura 5.16: Campo tensorial anisotrópico obtido após execução do Algoritmo 1 no Campo 1.

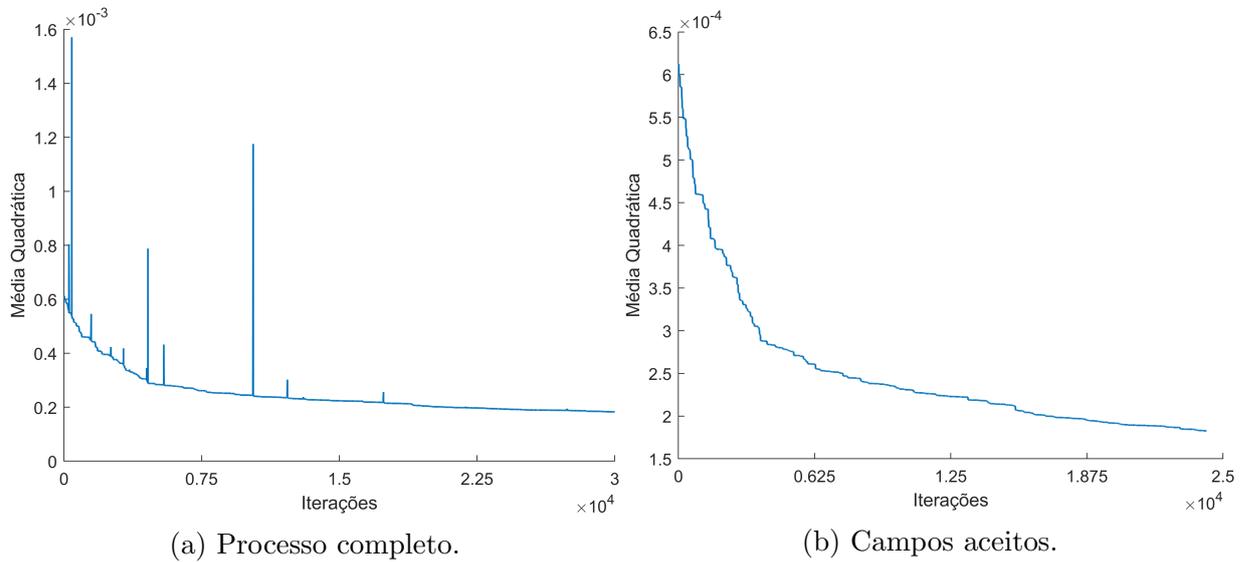


Figura 5.17: Evolução da média quadrática do divergente do Algoritmo 1 no Campo 1.

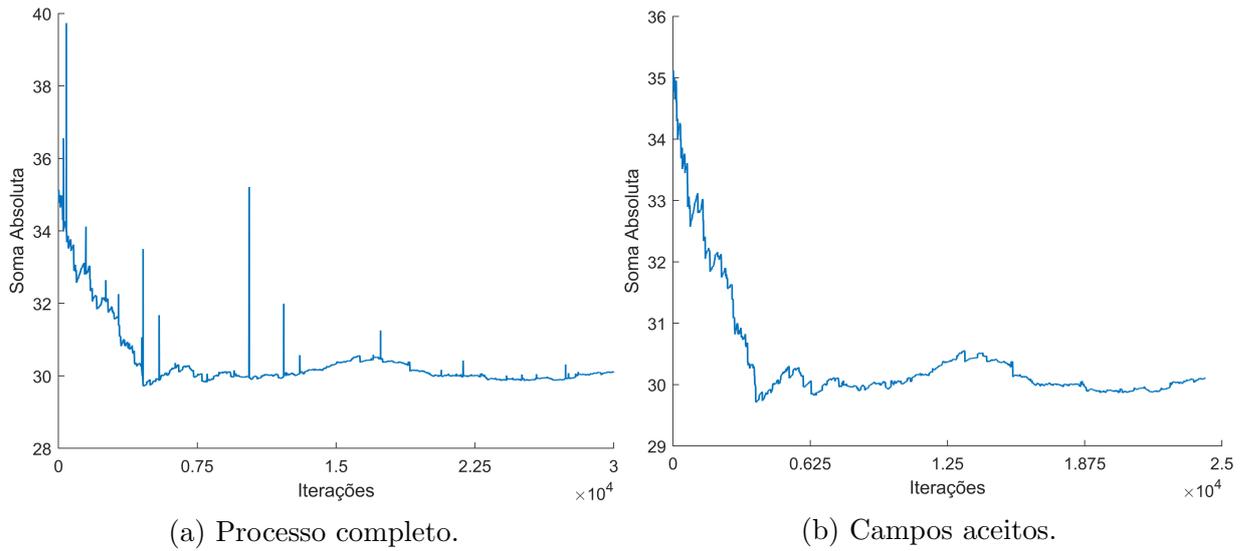


Figura 5.18: Evolução da soma absoluta dos divergentes do Algoritmo 1 no Campo 1.

Algoritmo 2

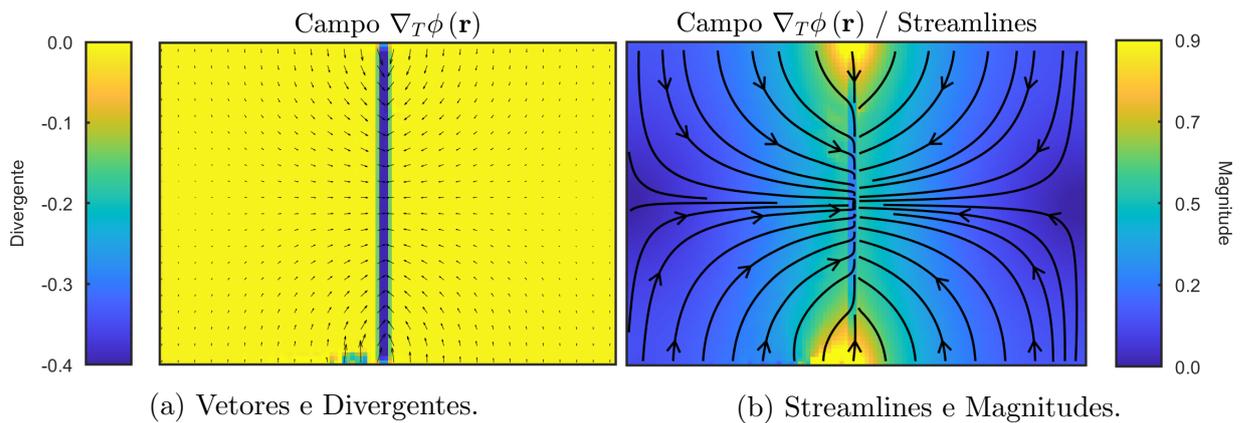


Figura 5.19: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 1 após execução do Algoritmo 2.

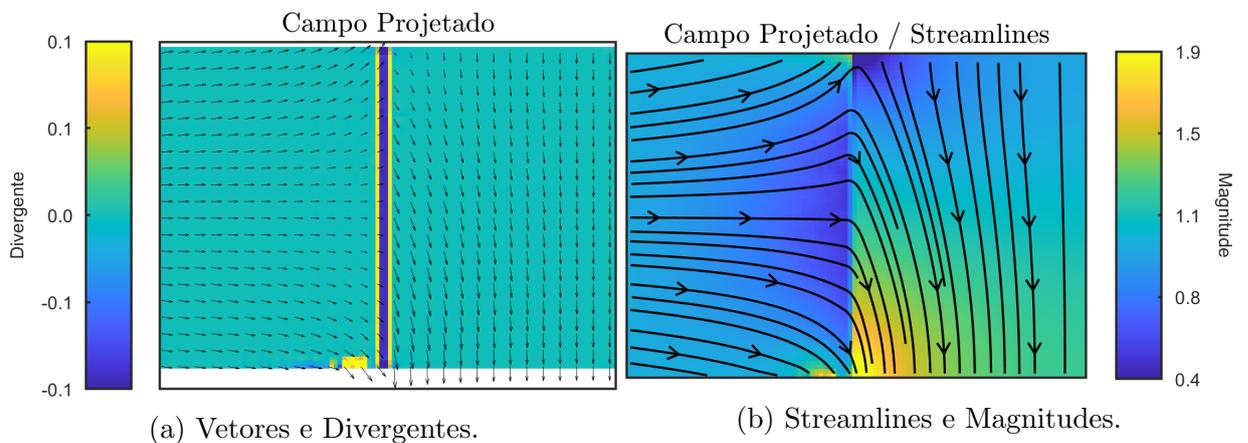


Figura 5.20: Campo 1 projetado após execução do Algoritmo 2.

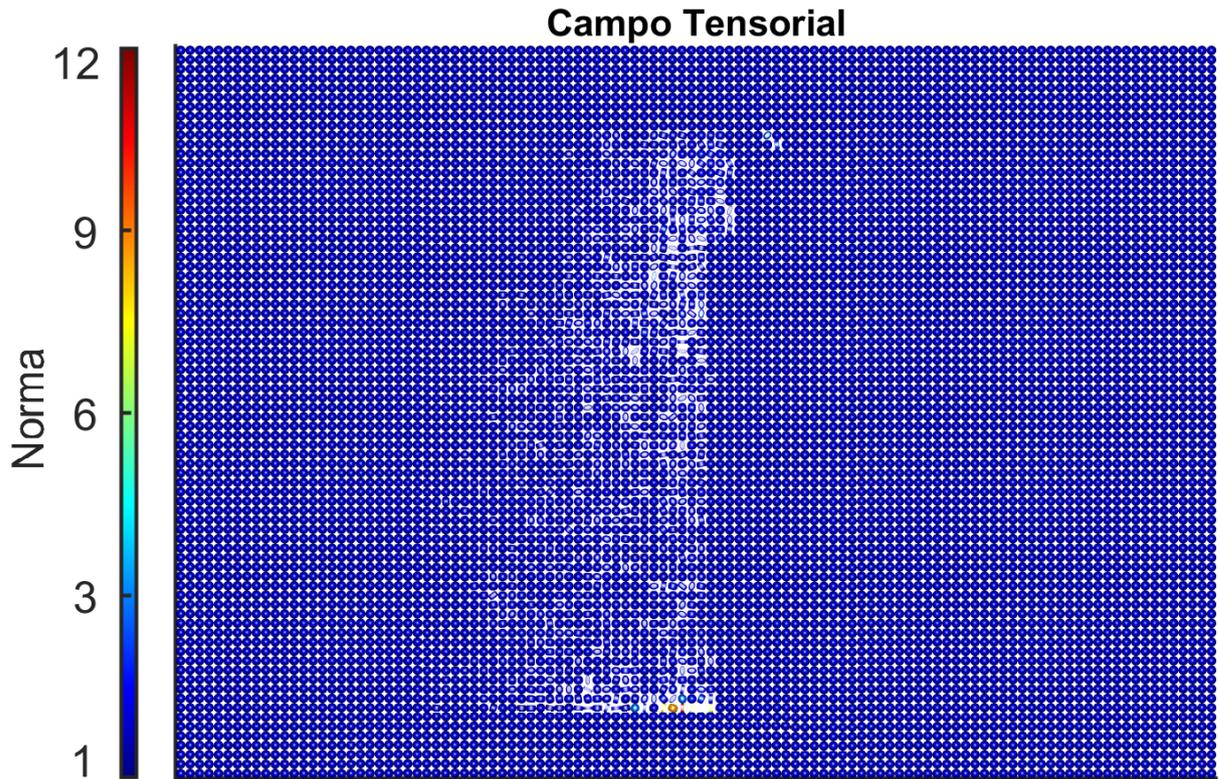


Figura 5.21: Campo tensorial anisotrópico obtido após execução do Algoritmo 2 no Campo 1.

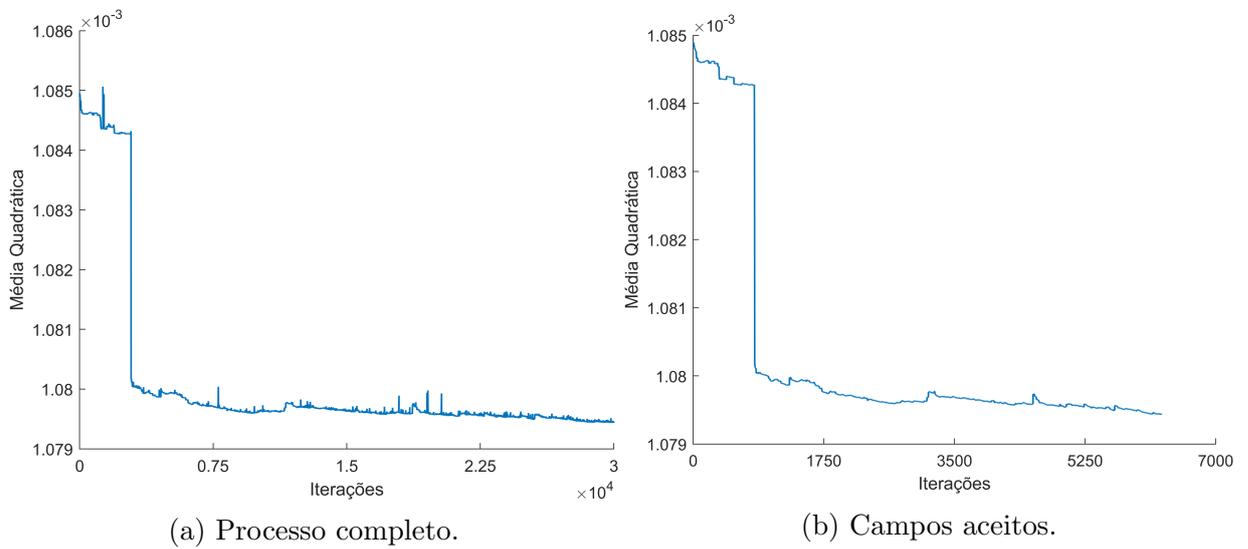


Figura 5.22: Evolução da média quadrática do divergente do Algoritmo 2 no Campo 1.

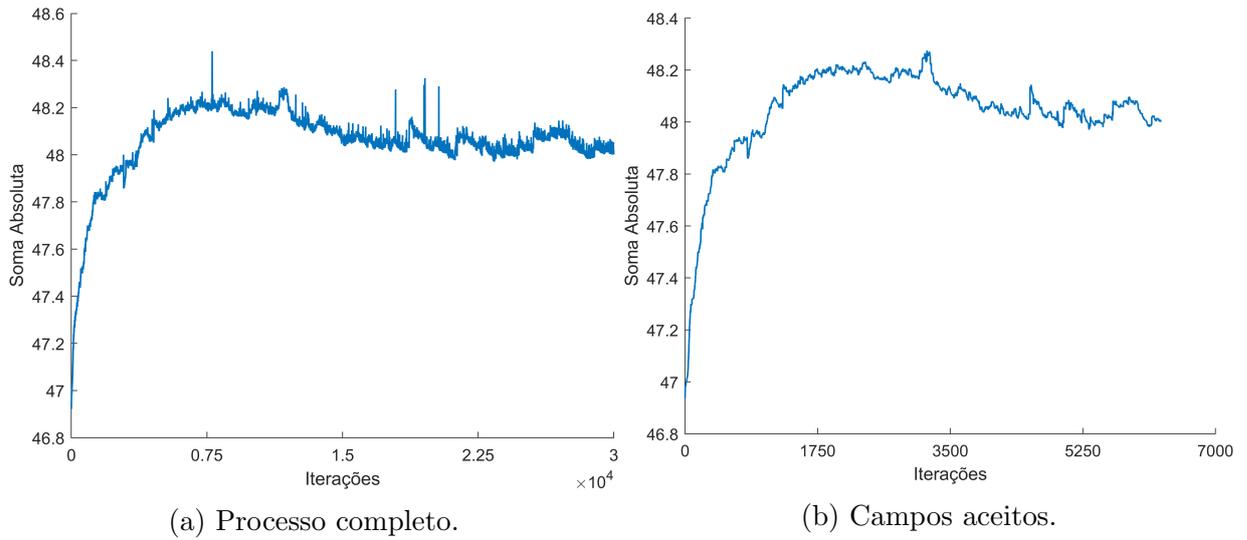


Figura 5.23: Evolução da soma absoluta dos divergentes do Algoritmo 2 no Campo 1.

5.3.2 Campo 2 - Descontínuo Diagonal

Tabela 5.7: Dados relativos aos processo completo da projeção anisotrópica do Campo 2.

	Algoritmo 1		Algoritmo 2	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-2,40345	3,652255	-0,44316	0,390782
Média quadrática do divergente	0,00133	0,004837	0,001159	0,002718
Soma Absoluta do divergente	88	103	71	90

Tabela 5.8: Dados relativos aos campos aceitos da projeção anisotrópica do Campo 2.

	Algoritmo 1		Algoritmo 2	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-0,32764	0,392234	-0,32722	0,308761
Média quadrática do divergente	0,001330	0,002717	0,001159	0,002718
Soma Absoluta do divergente	88	94	71	90

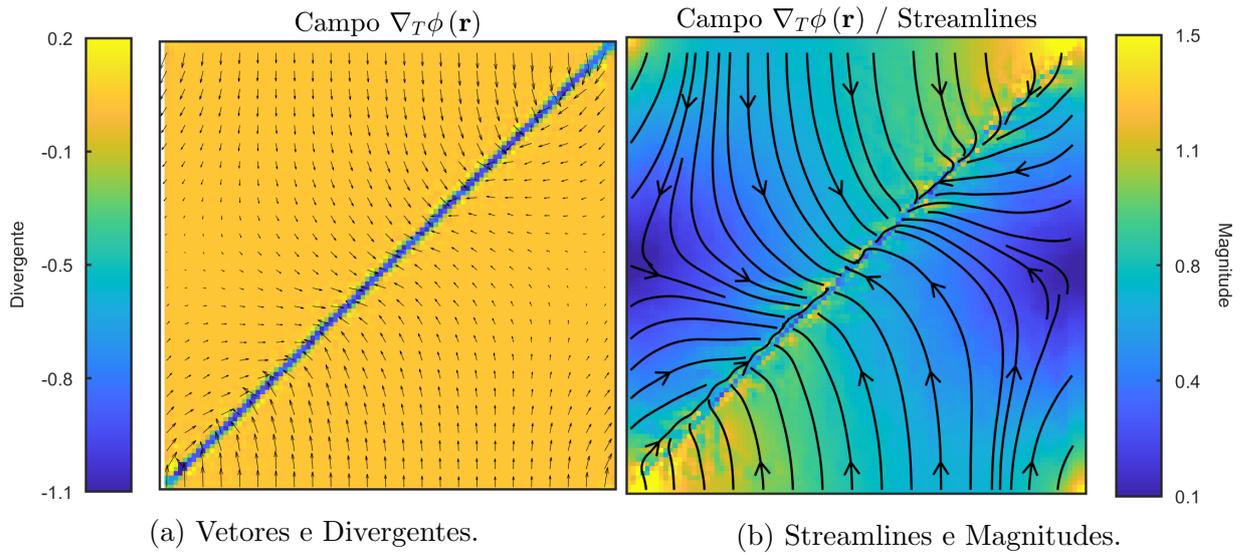
Algoritmo 1

Figura 5.24: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 2 após execução do Algoritmo 1.

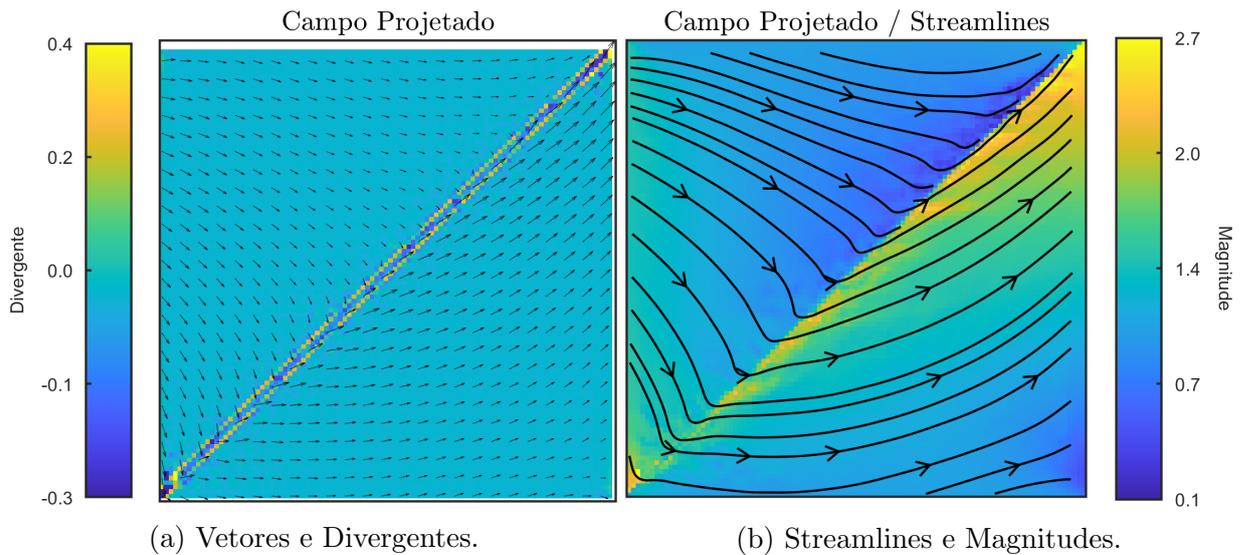


Figura 5.25: Campo 2 projetado após execução do Algoritmo 1.

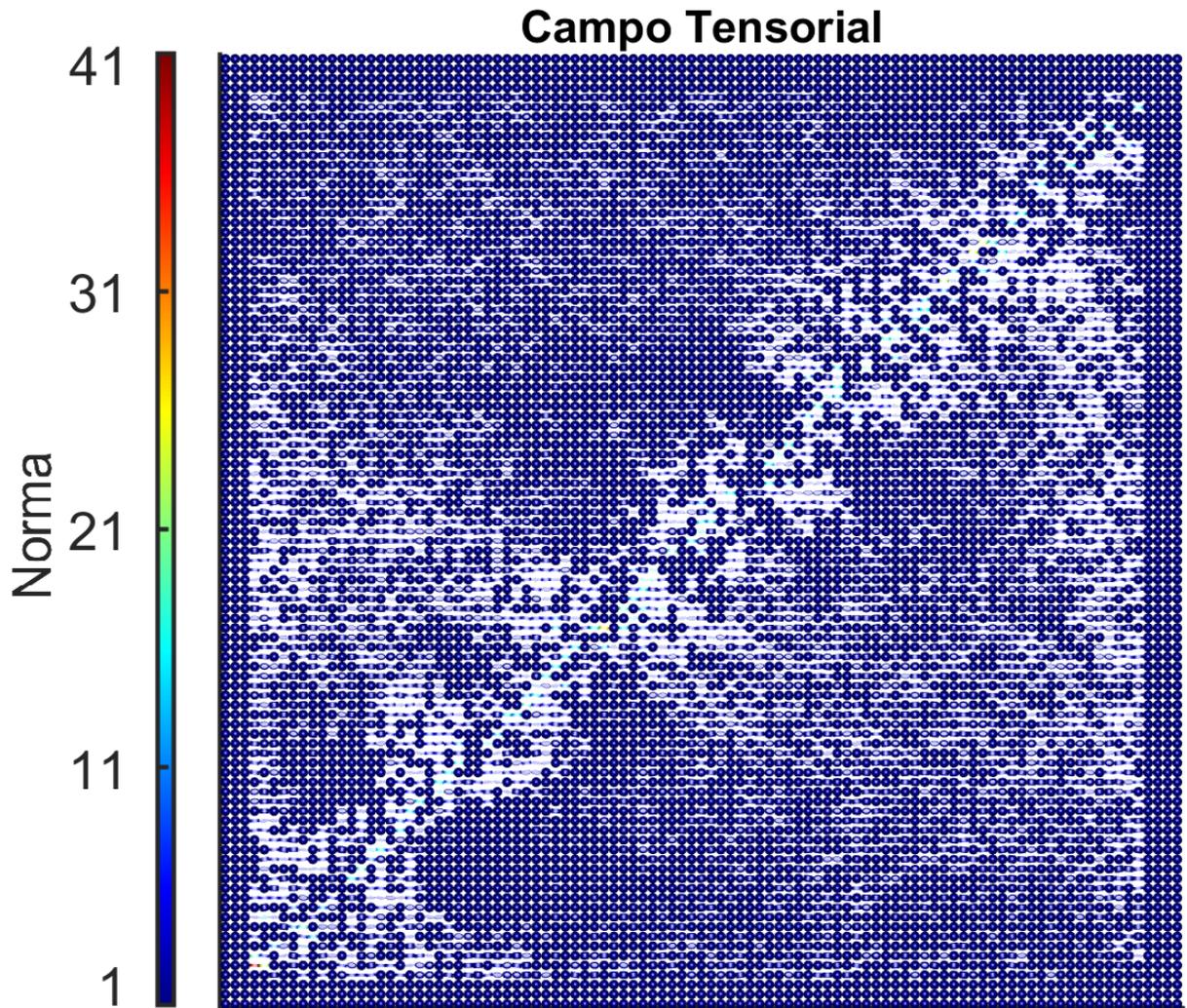


Figura 5.26: Campo tensorial anisotrópico obtido após execução do Algoritmo 1 no Campo 2.

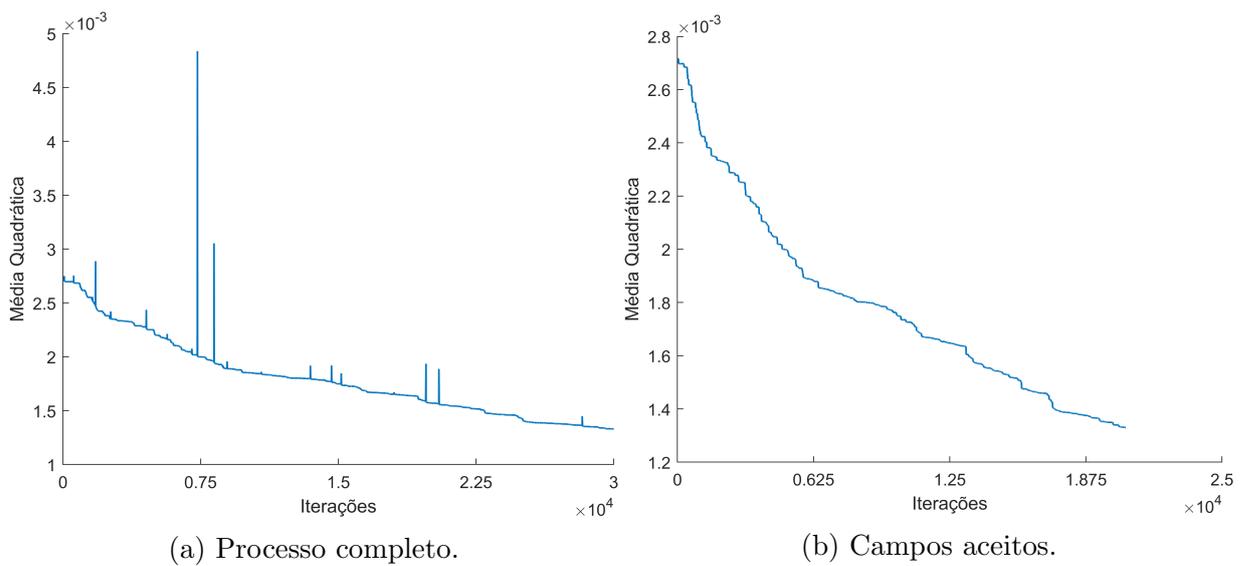


Figura 5.27: Evolução da média quadrática do divergente do Algoritmo 1 no Campo 2.

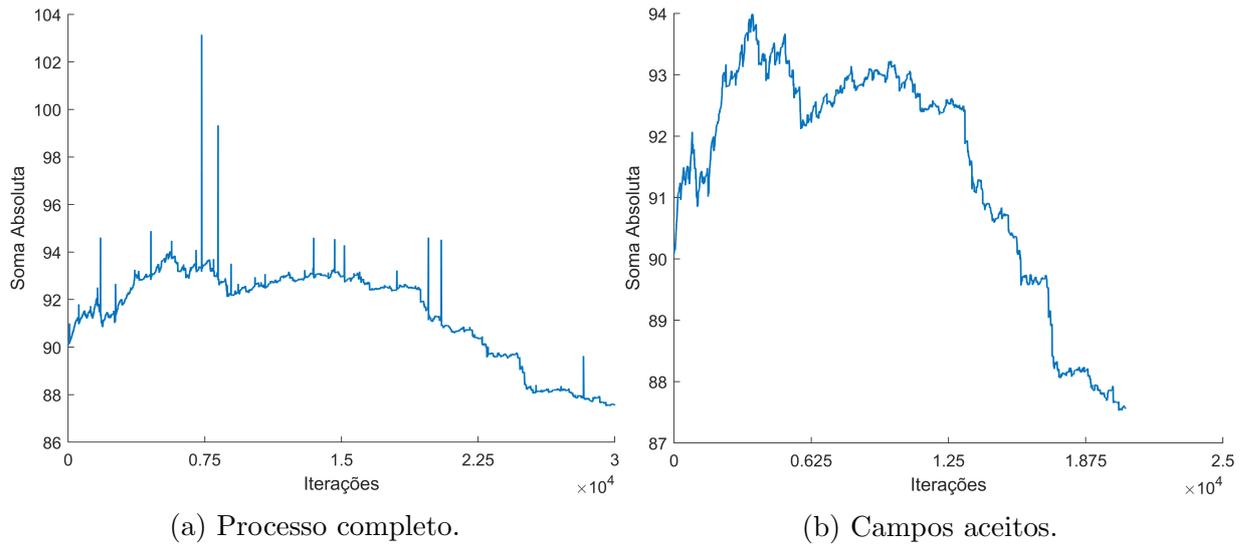


Figura 5.28: Evolução da soma absoluta dos divergentes do Algoritmo 1 no Campo 2.

Algoritmo 2

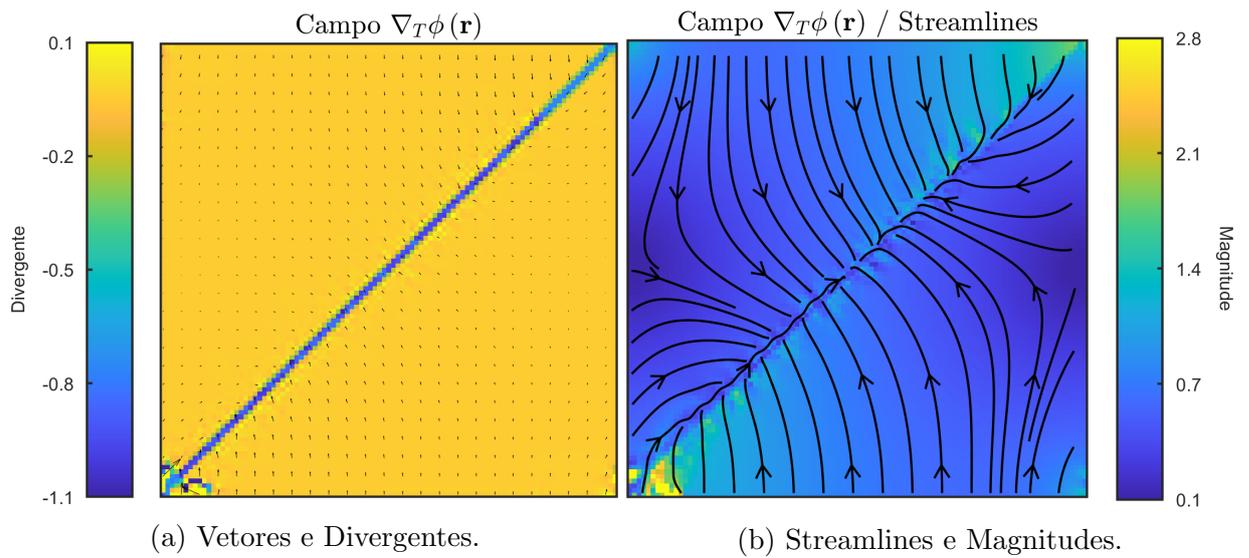


Figura 5.29: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 2 após execução do Algoritmo 2.

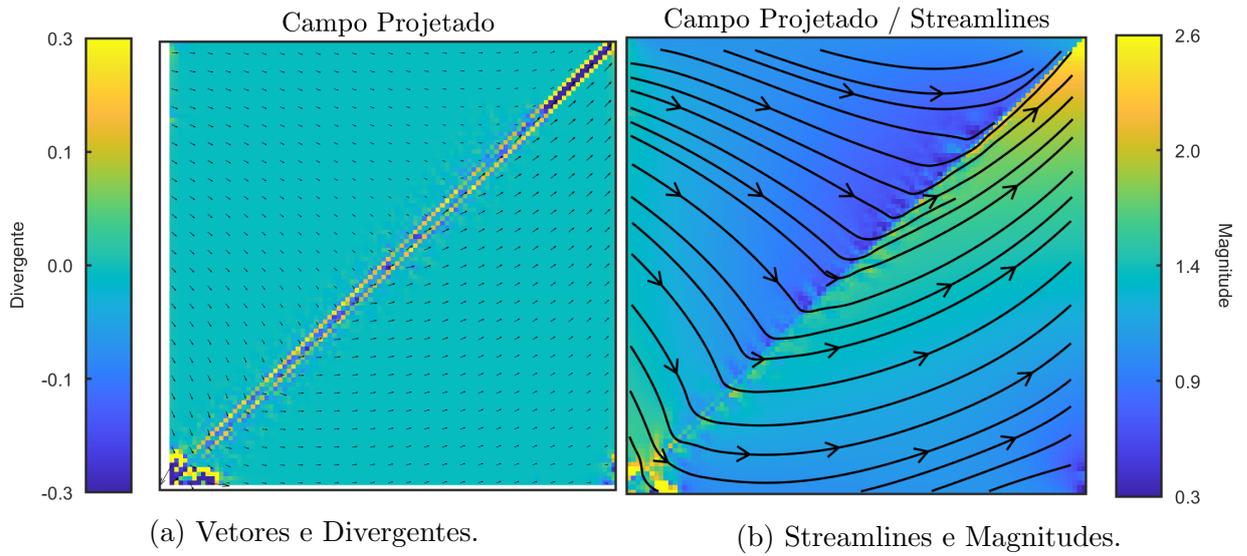


Figura 5.30: Campo 2 projetado após execução do Algoritmo 2.

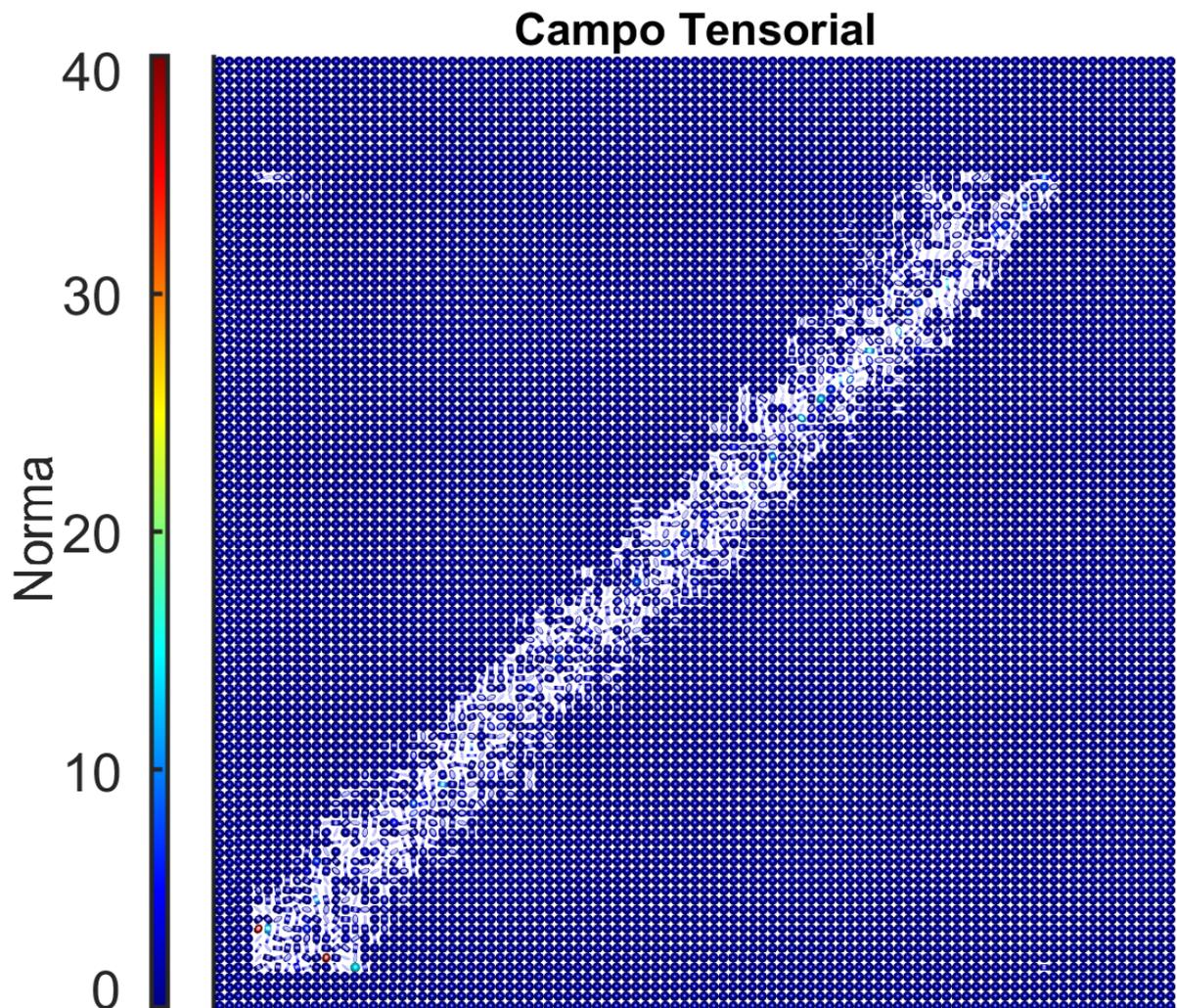


Figura 5.31: Campo tensorial anisotrópico obtido após execução do Algoritmo 2 no Campo 2.

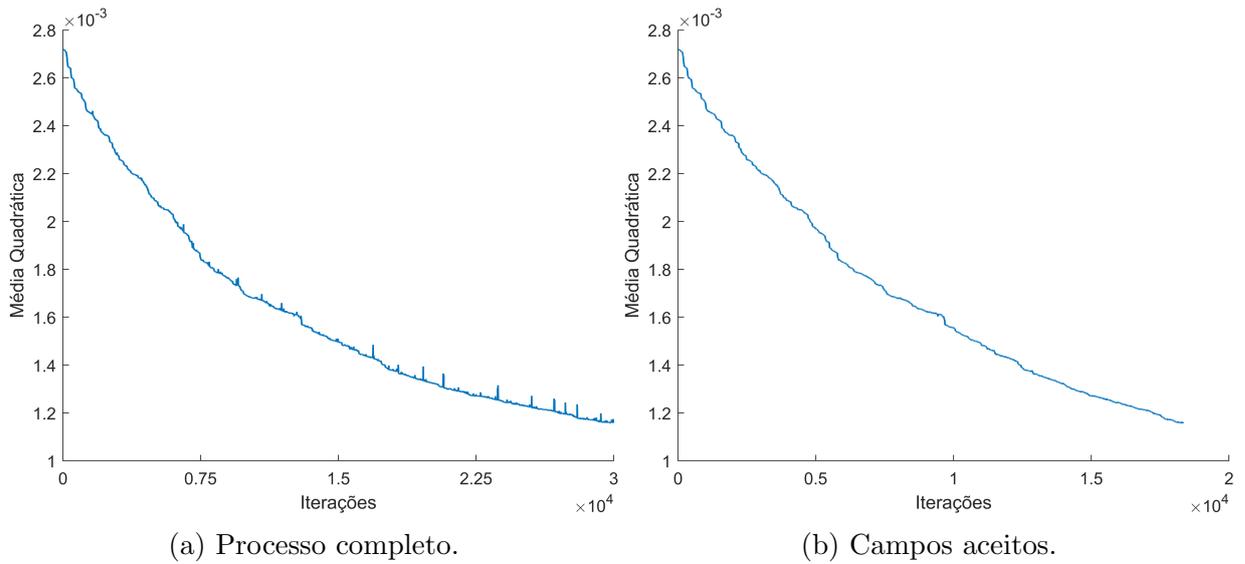


Figura 5.32: Evolução da média quadrática do divergente do Algoritmo 2 no Campo 2.

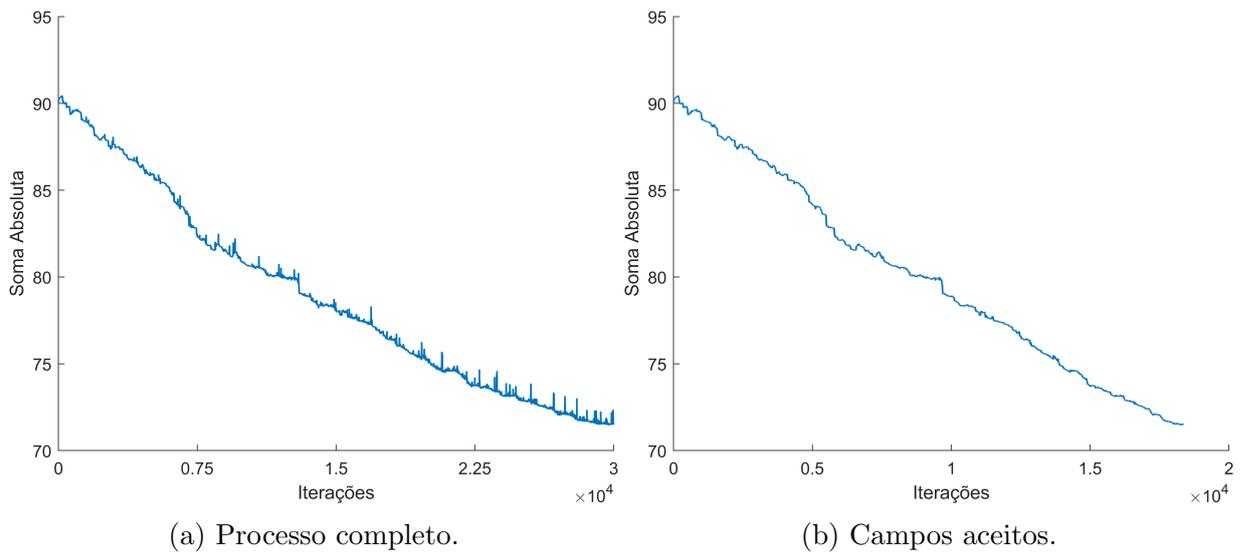


Figura 5.33: Evolução da soma absoluta dos divergentes do Algoritmo 2 no Campo 2.

5.3.3 Campo 3 - Complexo

Tabela 5.9: Dados relativos ao processo completo da projeção anisotrópica do Campo 3.

Dados	Algoritmo 1		Algoritmo 2	
	Mínimo	Máximo	Mínimo	Máximo
Divergente	-0,93048	1,159346	-0,26086	0,250036
Média quadrática do divergente	0,000556	0,001330	0,000837	0,001041
Soma Absoluta do divergente	113	154	137	152

Tabela 5.10: Dados relativos aos campos aceitos da projeção anisotrópica do Campo 3.

Dados	Algoritmo 1		Algoritmo 2	
	Mínimo	Máximo	Mínimo	Máximo
Divergente	-0,26018	0,250545	-0,26085	0,250012
Média quadrática do divergente	0,000556	0,001041	0,000837	0,001041
Soma Absoluta do divergente	113	152	137	152

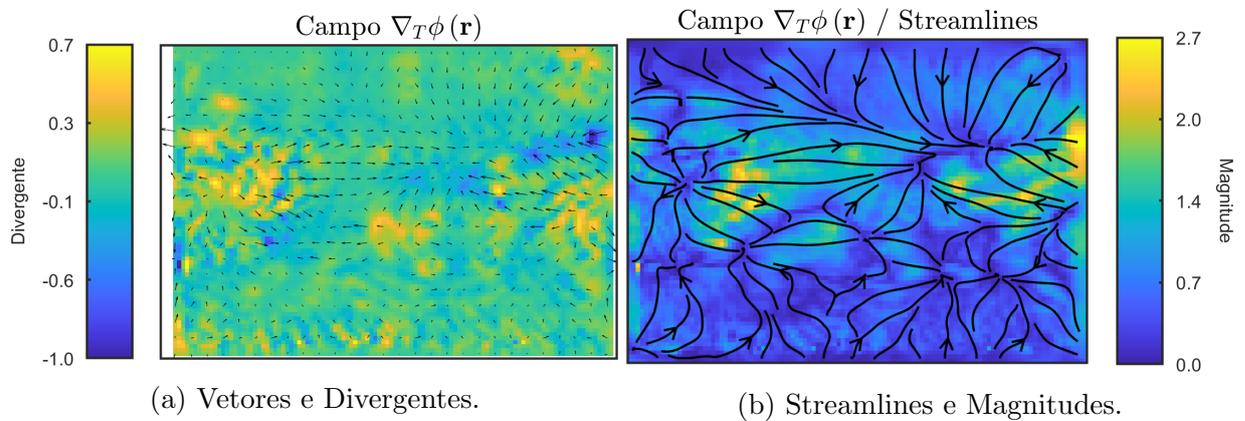
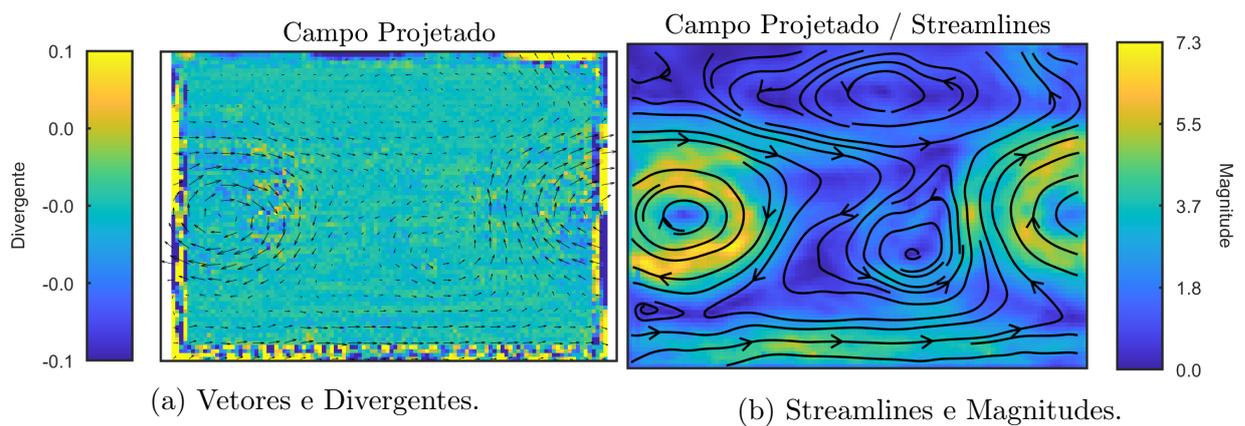
Algoritmo 1Figura 5.34: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 3 após execução do Algoritmo 1.

Figura 5.35: Campo 3 projetado após execução do Algoritmo 1.

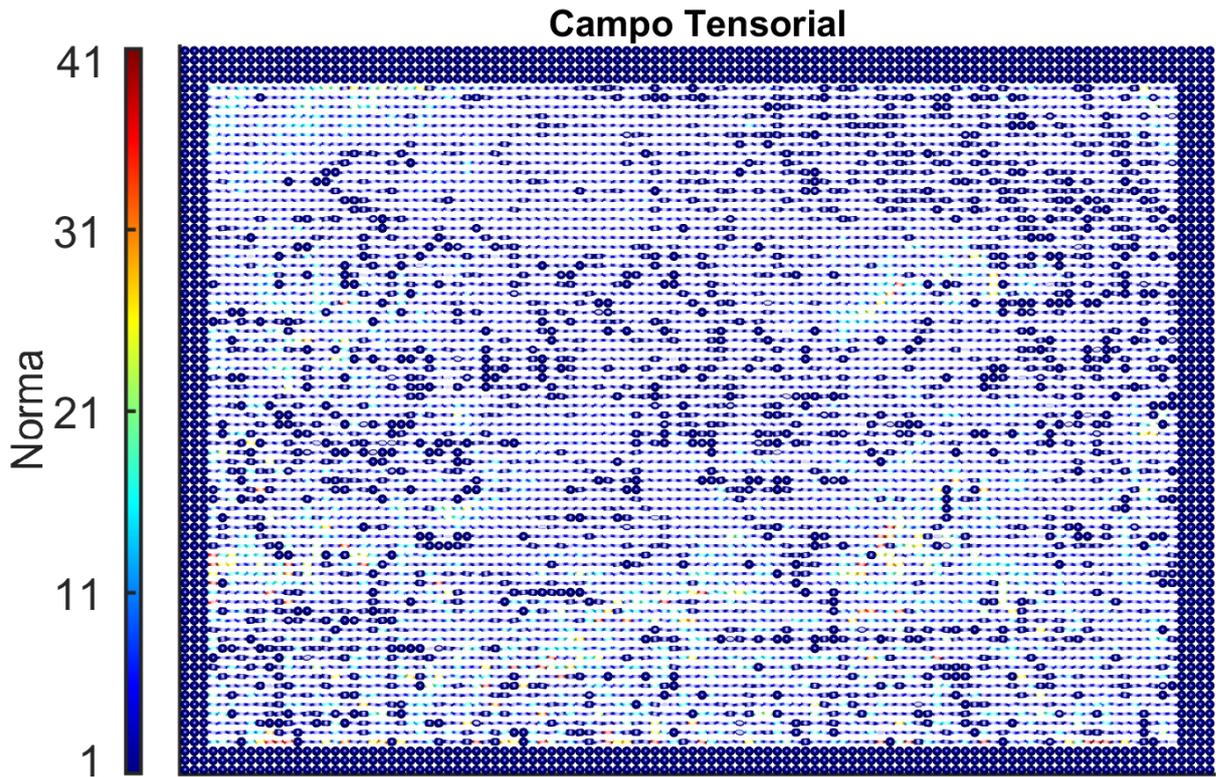


Figura 5.36: Campo tensorial anisotrópico obtido após execução do Algoritmo 1 no Campo 3.

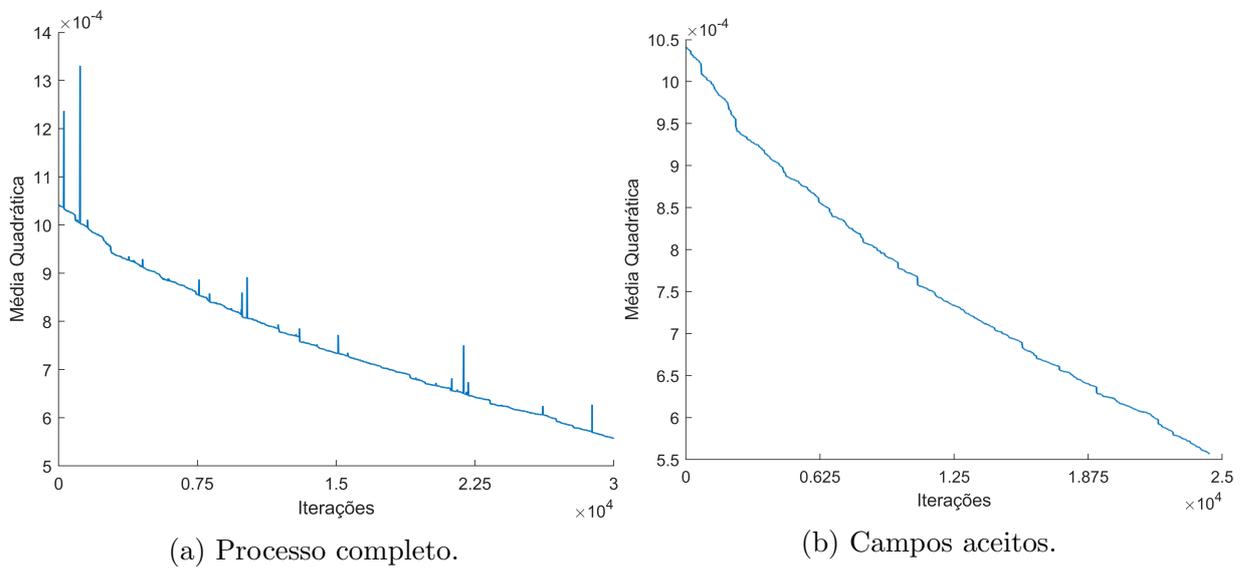


Figura 5.37: Evolução da média quadrática do divergente do Algoritmo 1 no Campo 3.

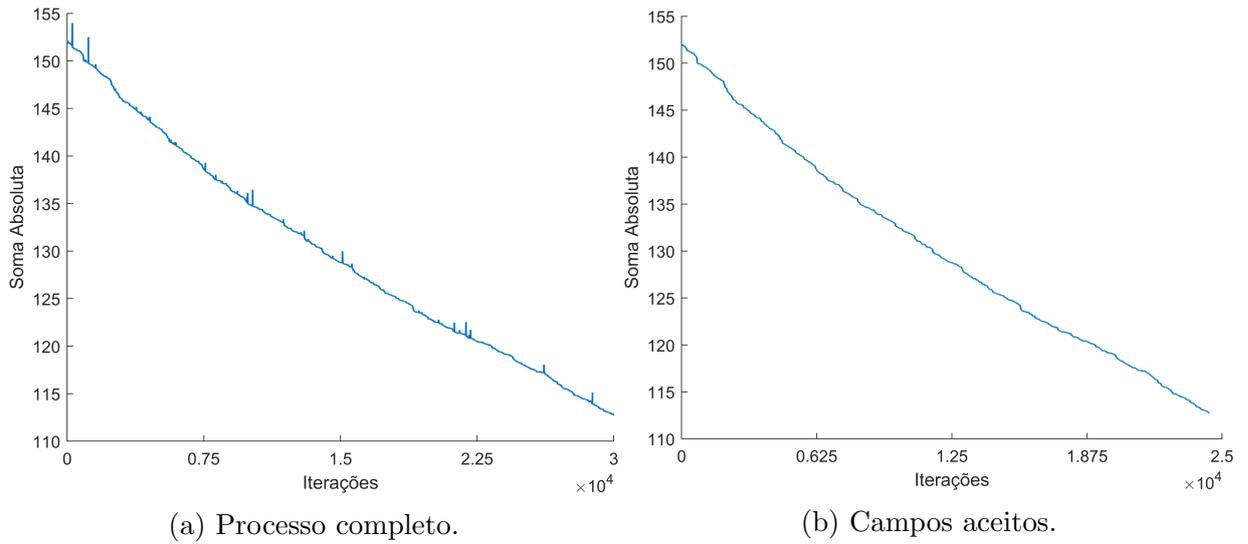


Figura 5.38: Evolução da soma absoluta dos divergentes do Algoritmo 1 no Campo 3.

Algoritmo 2

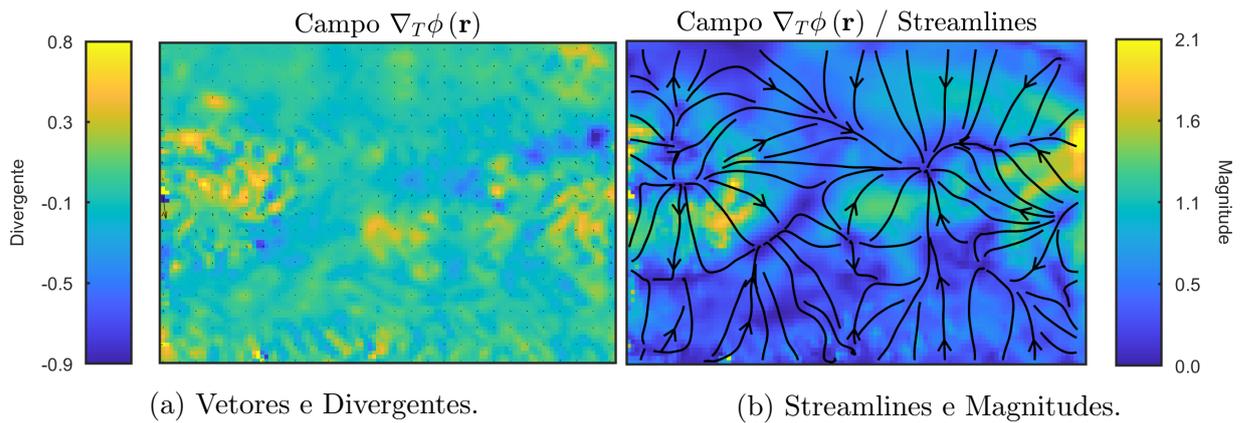


Figura 5.39: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 3 após execução do Algoritmo 2.

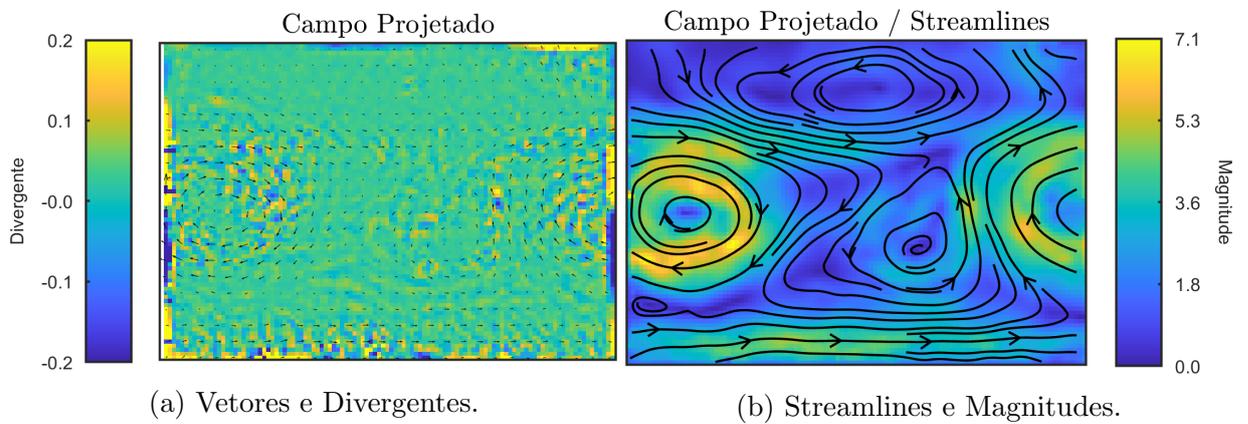


Figura 5.40: Campo 3 projetado após execução do Algoritmo 2.

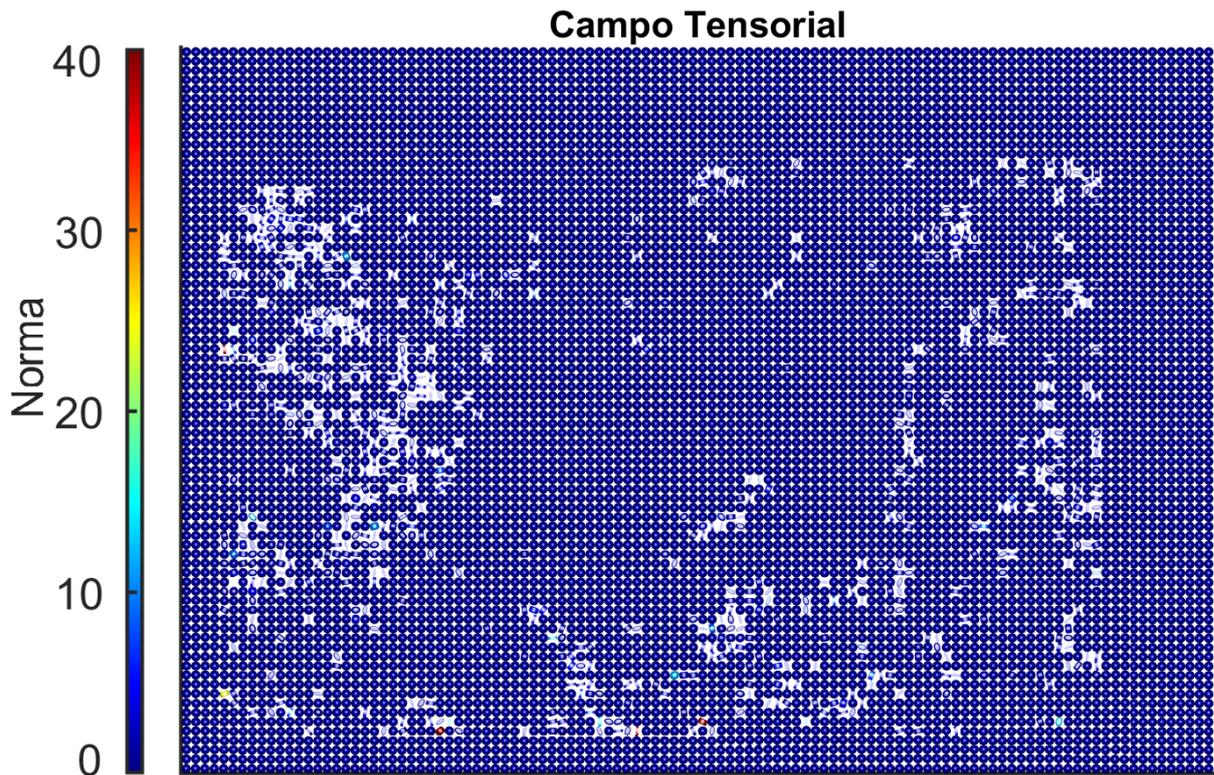


Figura 5.41: Campo tensorial anisotrópico obtido após execução do Algoritmo 2 no Campo 3.

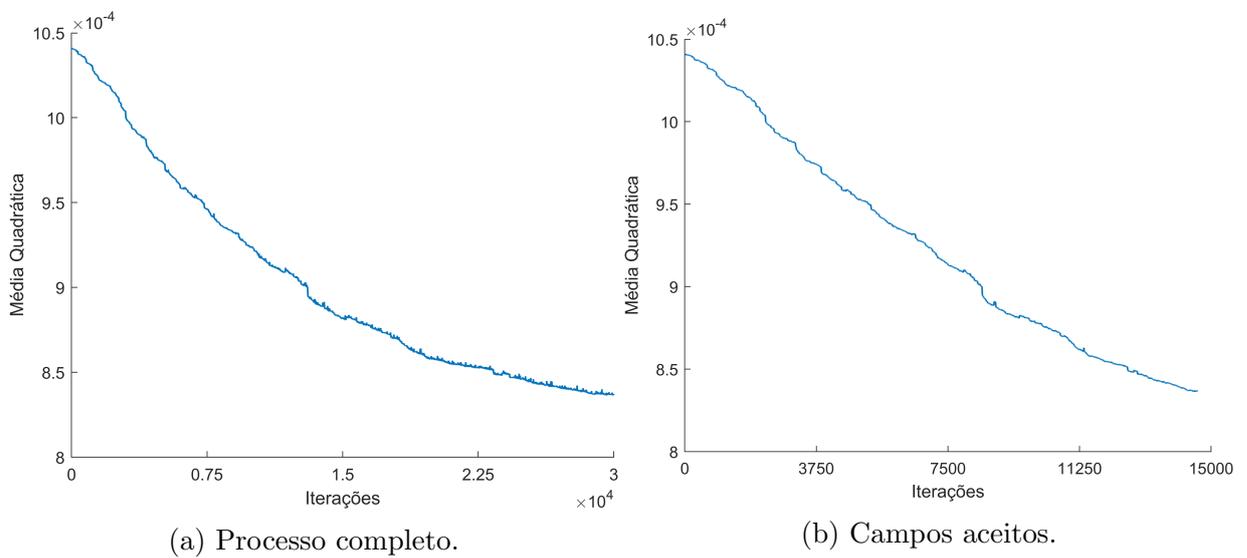


Figura 5.42: Evolução da média quadrática do divergente do Algoritmo 2 no Campo 3.

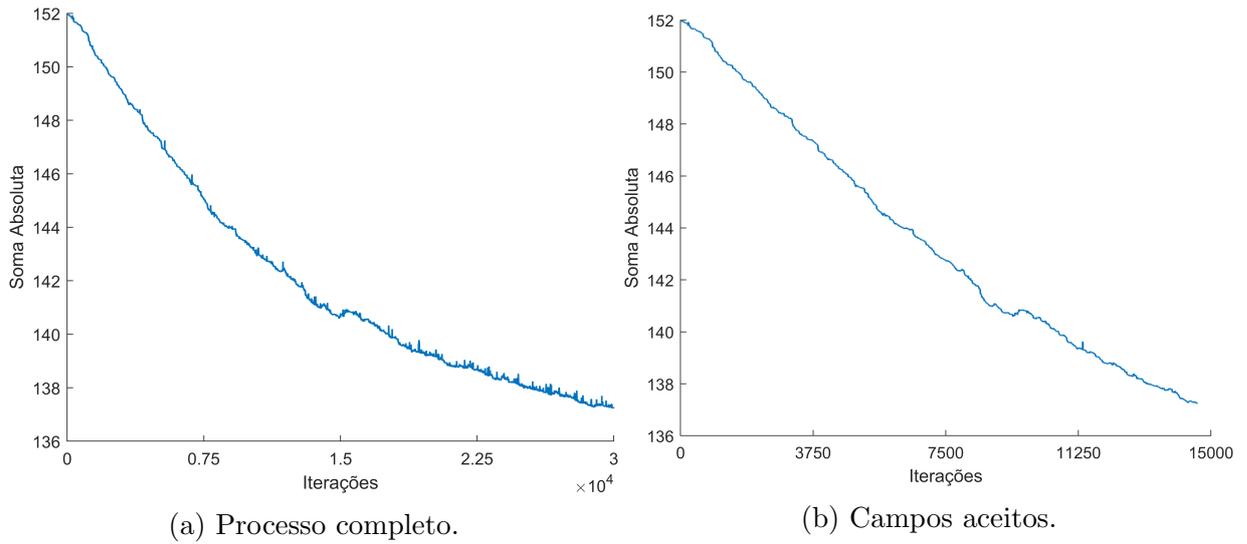


Figura 5.43: Evolução da soma absoluta dos divergentes do Algoritmo 3 no Campo 1.

5.3.4 Campo 4 - Radial Rotacional

Tabela 5.11: Dados relativos ao processo completo da projeção anisotrópica do Campo 4.

	Algoritmo 1		Algoritmo 2	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-163,873417	191,897217	-13,551342	8,064126
Média quadrática do divergente	1,311665	10,043111	1,772142	1,999380
Soma Absoluta do divergente	2978	3712	2974	3094

Tabela 5.12: Dados relativos aos campos aceitos da projeção anisotrópica do Campo 4.

	Algoritmo 1		Algoritmo 2	
Dados	Mínimo	Máximo	Mínimo	Máximo
Divergente	-14,205794	8,518323	-13,551341	8,064073
Média quadrática do divergente	1,311665	1,999380	1,772142	1,999287
Soma Absoluta do divergente	2978	3118	2975	3094

Algoritmo 1

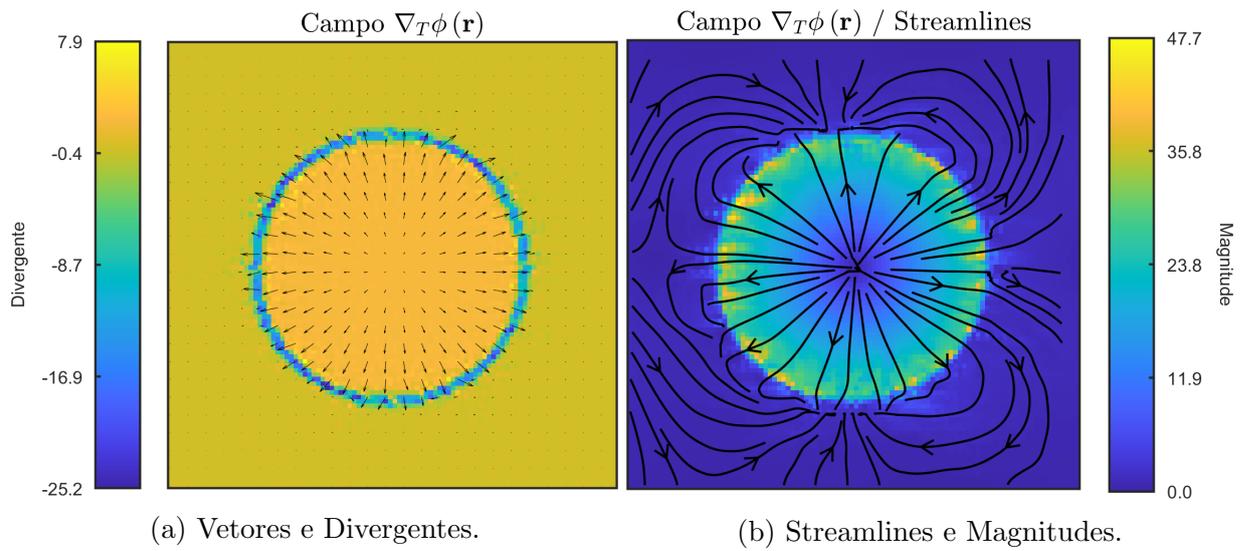


Figura 5.44: Campo vetorial T-irrotacional $\nabla_T \phi(\mathbf{r})$ para o Campo 4 após execução do Algoritmo 1.

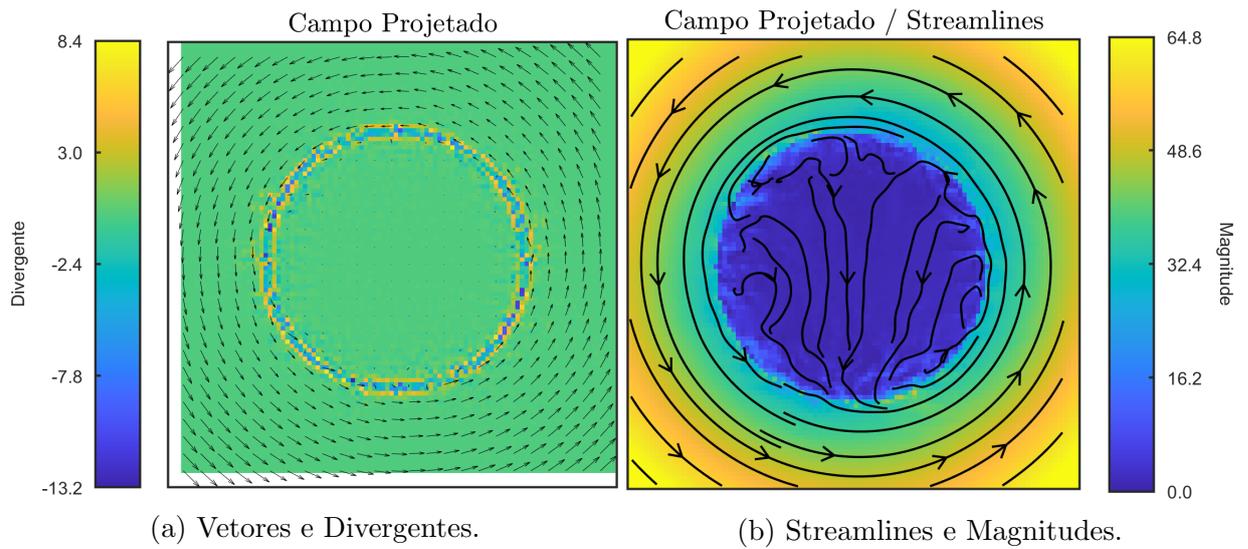


Figura 5.45: Campo 4 projetado após execução do Algoritmo 1.

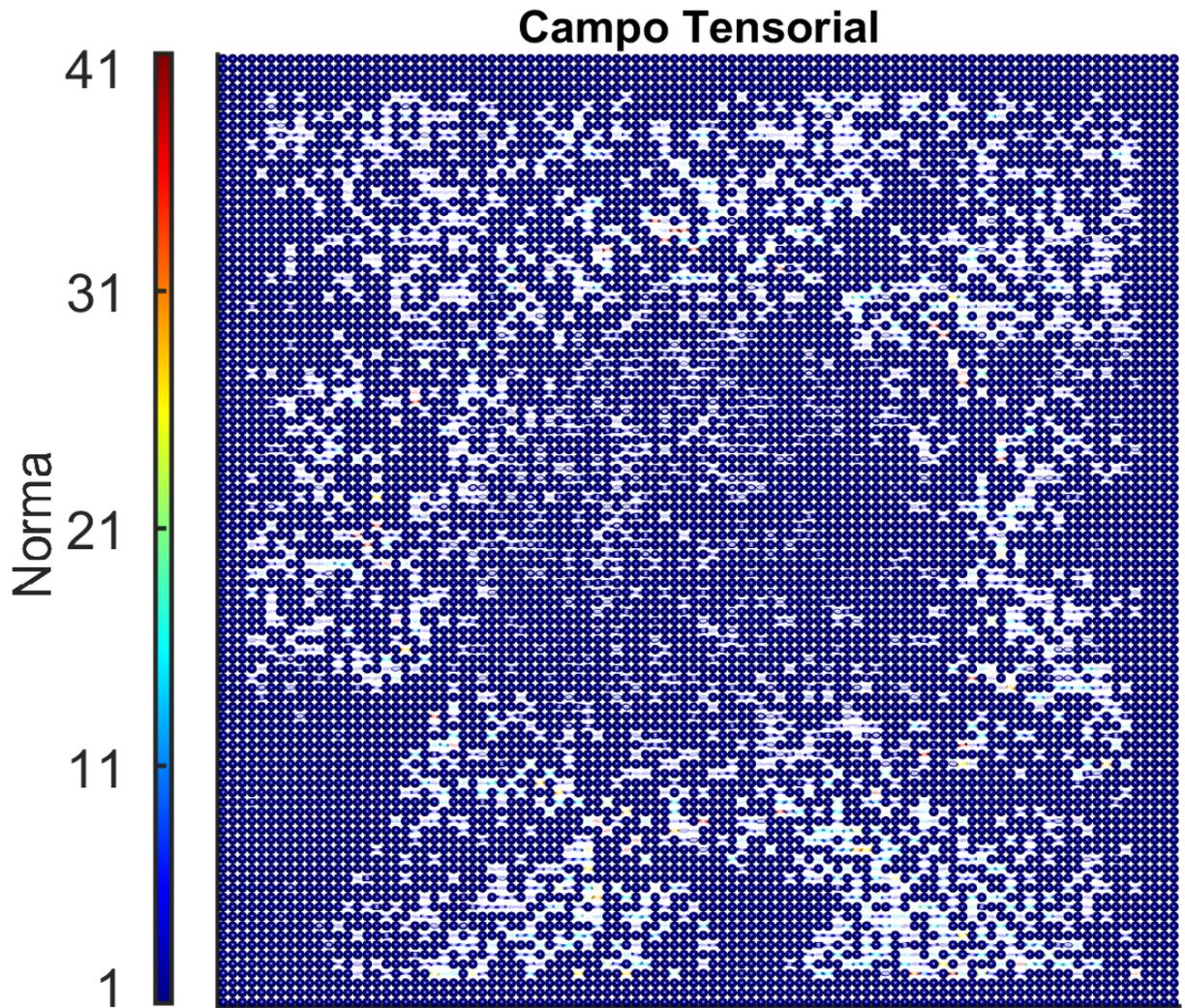


Figura 5.46: Campo tensorial anisotrópico obtido após execução do Algoritmo 1 no Campo 4.

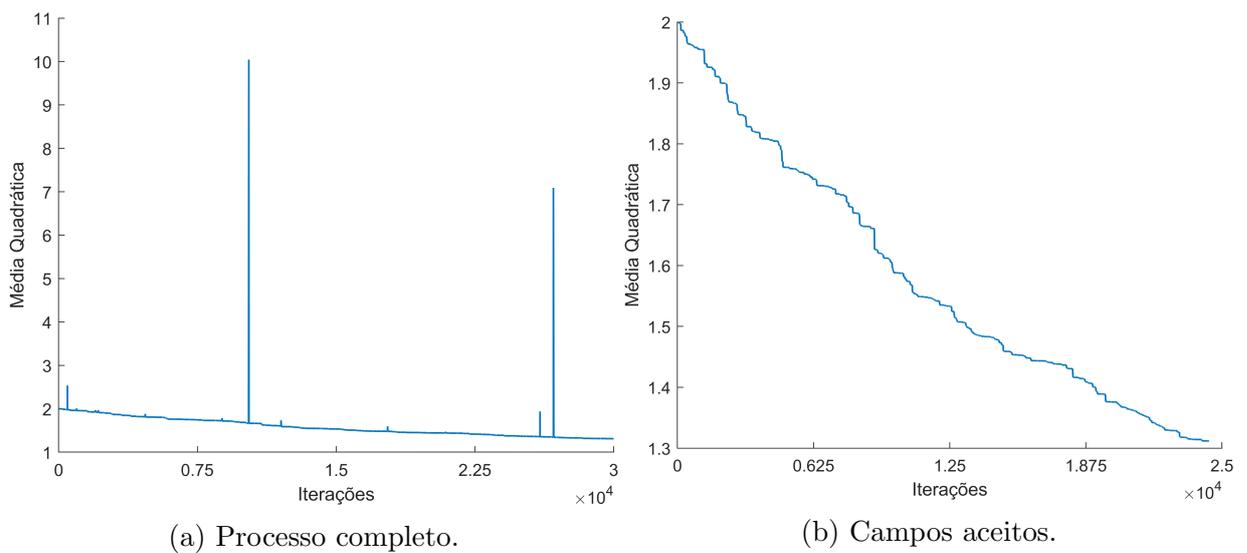


Figura 5.47: Evolução da média quadrática do divergente do Algoritmo 1 no Campo 4.

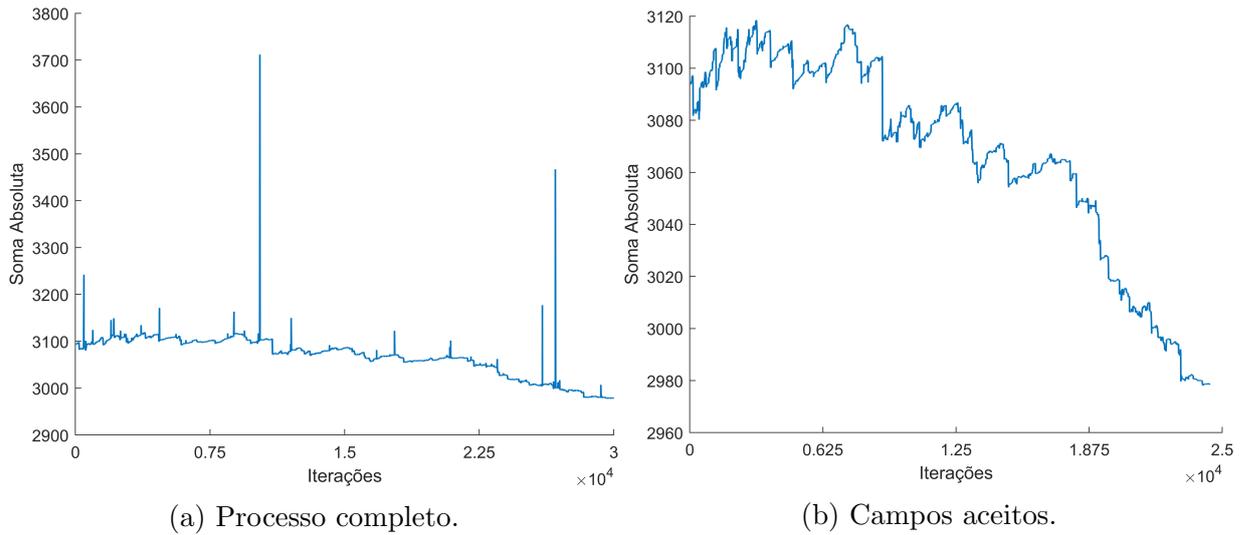


Figura 5.48: Evolução da soma absoluta dos divergentes do Algoritmo 1 no Campo 4.

Algoritmo 2

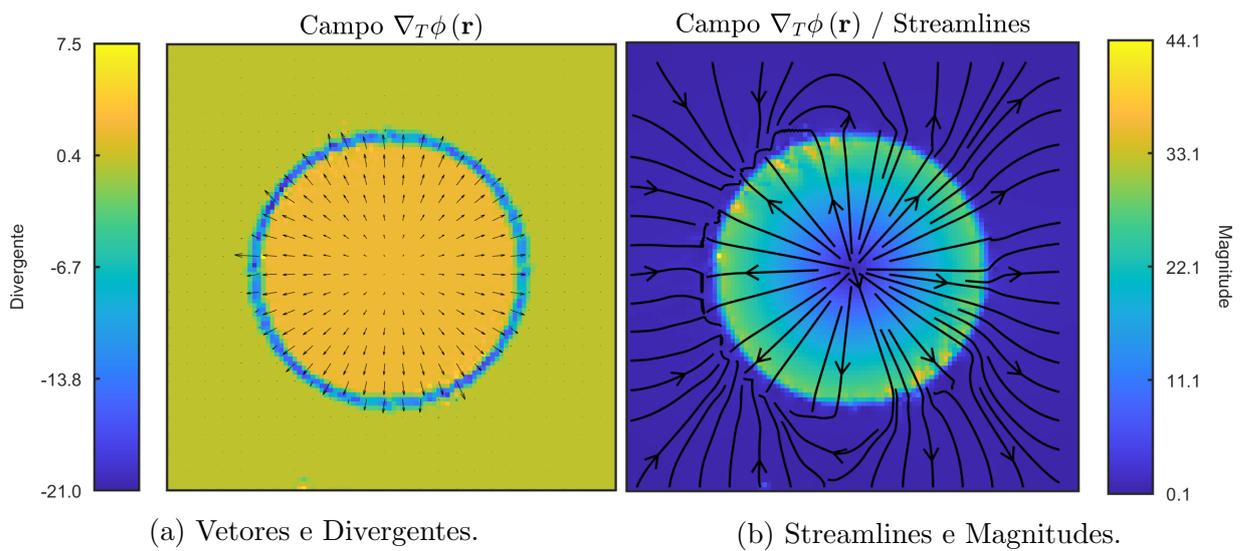


Figura 5.49: Campo vetorial T-irrotacional $\nabla_T\phi(\mathbf{r})$ para o Campo 4 após execução do Algoritmo 2.

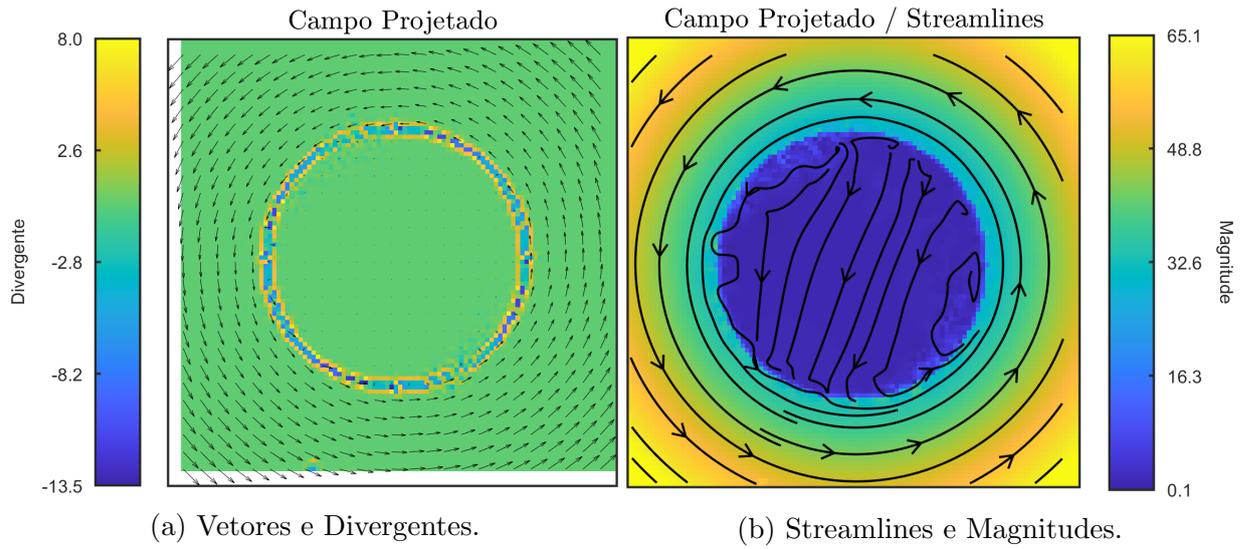


Figura 5.50: Campo 4 projetado após execução do Algoritmo 2.

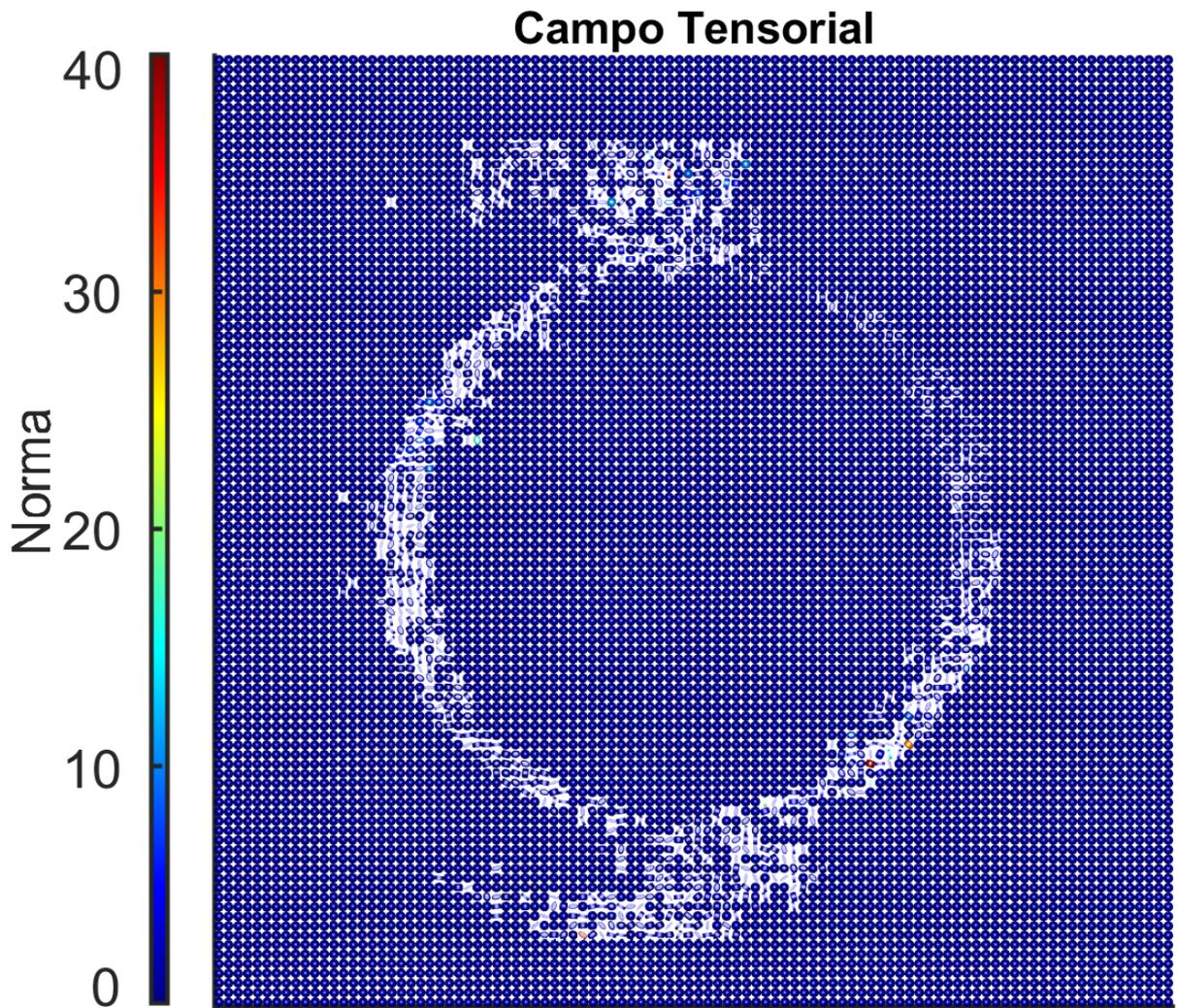


Figura 5.51: Campo tensorial anisotrópico obtido após execução do Algoritmo 2 no Campo 4.

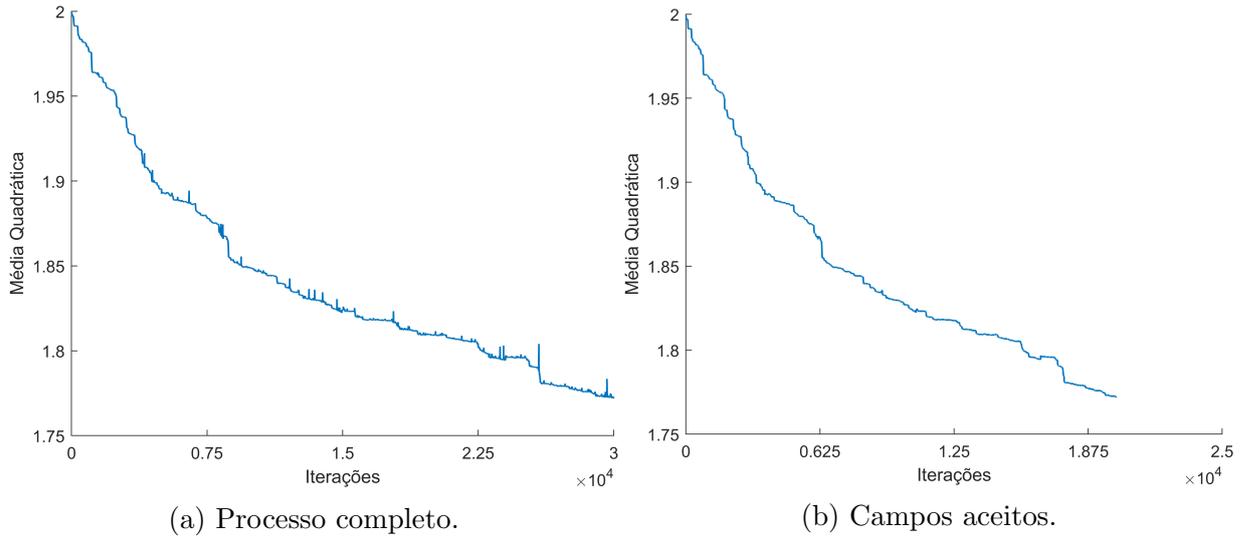


Figura 5.52: Evolução da média quadrática do divergente do Algoritmo 2 no Campo 4.

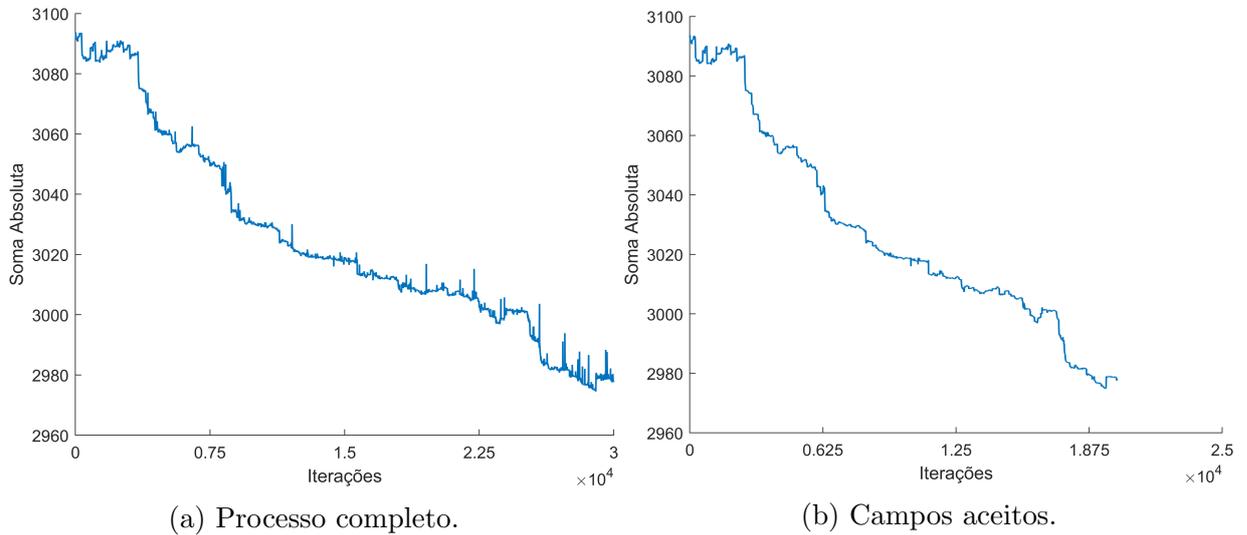


Figura 5.53: Evolução da soma absoluta dos divergentes do Algoritmo 2 no Campo 4.

5.4 Análise dos Dados

Para análise dos dados obtidos, é necessária a construção de gráficos que esclarecem a relação entre esses dados. Com esse propósito, três novos gráficos foram desenvolvidos. Todos os valores referentes aos algoritmos 1 e 2 são os mínimos encontrados, considerando os campos que foram aceitos pelos algoritmos.

O primeiro gráfico, que pode ser observado, por exemplo, na Figura (5.54), diz respeito aos valores brutos obtidos da média quadrática e da soma absoluta do divergente. O segundo, que pode ser observado, por exemplo, na Figura (5.58), diz respeito às proporções entre os valores dos dados. Por fim, no terceiro gráfico, que pode ser observado,

por exemplo, na Figura (5.62), diz respeito à redução da média quadrática do divergente, em cada um dos testes realizados. Este último gráfico possui o objetivo de evidenciar a redução observada ao analisar as proporções dos valores apresentados no segundo gráfico.

Nota-se na Figura (5.58) que o Algoritmo 2 encontrou problemas no teste relativo ao Campo 1 e, apesar da redução de divergente em relação ao campo sem projeção, o algoritmo piorou o resultado em relação à projeção isotrópica. Ao analisar o número de estados aceitos, informação presente na Figura (5.22b), e o campo tensorial resultante, Figura (5.21), o algoritmo teve dificuldades em fazer boas modificações no campo tensorial, dada a pequena quantidade de campos aceitos. Esse acontecimento pode estar relacionado com a matriz montada para seleção dos tensores. Todavia, apesar desse resultado, essa foi a única situação em que um dos algoritmos obteve um desempenho inferior à projeção isotrópica e, por isso, pode ser considerado uma exceção. No mesmo campo, o Algoritmo 1 obteve uma redução de 96% do divergente em relação ao campo sem projeção, e de 70% em relação à projeção isotrópica, como pode ser observado na Figura (5.62). A partir disso, nota-se uma redução de quase duas ordens de magnitude em relação ao divergente do campo sem projeção, como pode ser visto ao comparar os valores contidos na Figura(5.54a).

Os algoritmos conseguiram um resultado ainda mais expressivo nos testes realizados sobre o Campo 3, reduzindo duas ordens de magnitude do divergente relacionado ao campo sem projeção, como pode ser visto na Figura (5.56a). Os algoritmos alcançaram uma redução de 98% e 97% respectivamente, como pode ser visto na Figura (5.64).

Quanto aos campos tensoriais resultantes, nota-se que as modificações realizadas em tensores que estão nas posições de maior divergente tendem a ser aceitas. Essa característica pode ser deduzida pela coloração, formato e rotação dos tensores nas regiões de maior divergente, indicando que eles sofreram mais alterações que os demais. O impacto dos campos tensoriais no escoamento dos vetores pode ser notado ao comparar os campos projetados e seus respectivos estados iniciais. Os campos 1 e 2 foram desenvolvidos justamente com esse propósito e é possível ver que, em suas projeções anisotrópicas, há uma suavização onde inicialmente era uma transição brusca. Observa-se nos campos tensoriais resultantes que os tensores não tendem a se alinhar com os vetores, como inicialmente

era esperado. Geralmente eles são ortogonais aos vetores, além de tenderem a possuir um formato planar.

A diminuição do divergente também fica evidente, principalmente no Campo 3, onde o divergente é distribuído de diversas formas distintas por todo o campo. Nota-se, por exemplo, nas Figuras (5.35a) e (5.40a), a homogeneização das cores do divergente dos campos, e uma aproximação do valor 0.

Outro ponto interessante pode ser observado nos Campos 1 e 4, onde o divergente é concentrado em partes específicas do campo, e tendem a ser pontos de sorvedouro. Há o surgimento de pontos de fonte, ou seja, divergente acima de 0 ao redor da zona onde o divergente inicialmente estava concentrado. Essa característica pode ser vista ao se comparar as Figuras (5.15a) e (5.50a) com o estado inicial do campo, representados nas Figuras (5.1a) e (5.4a). Contudo, mesmo com o surgimento desses pontos de fonte nas bordas, o divergente é reduzido em seus pontos internos, característica visível pela homogeneização das cores em valores que tendem a 0.

Os resultados obtidos na projeção do Campo 4 também são notáveis. Esse campo possui pontos de fonte em seu centro, onde o fluxo dos vetores é radial, e na transição para o fluxo rotacional dos vetores existem pontos de sorvedouro. Na projeção isotrópica, que pode ser vista na Figura (5.13a), os pontos de fonte são suavizados no centro do campo, e passam a se concentrar nas bordas da zona onde estão os sorvedouros. No entanto, nas projeções anisotrópicas do Campo 4, que podem ser vistas nas Figuras (5.45a) e (5.50a), as fontes do centro são anuladas. Com isso, o fluxo no centro foi quase totalmente anulado. Isso mostra que os algoritmos tenderam a anular esse fluxo no centro e não redirecioná-lo como inicialmente era esperado.

5.4.1 Valores Brutos

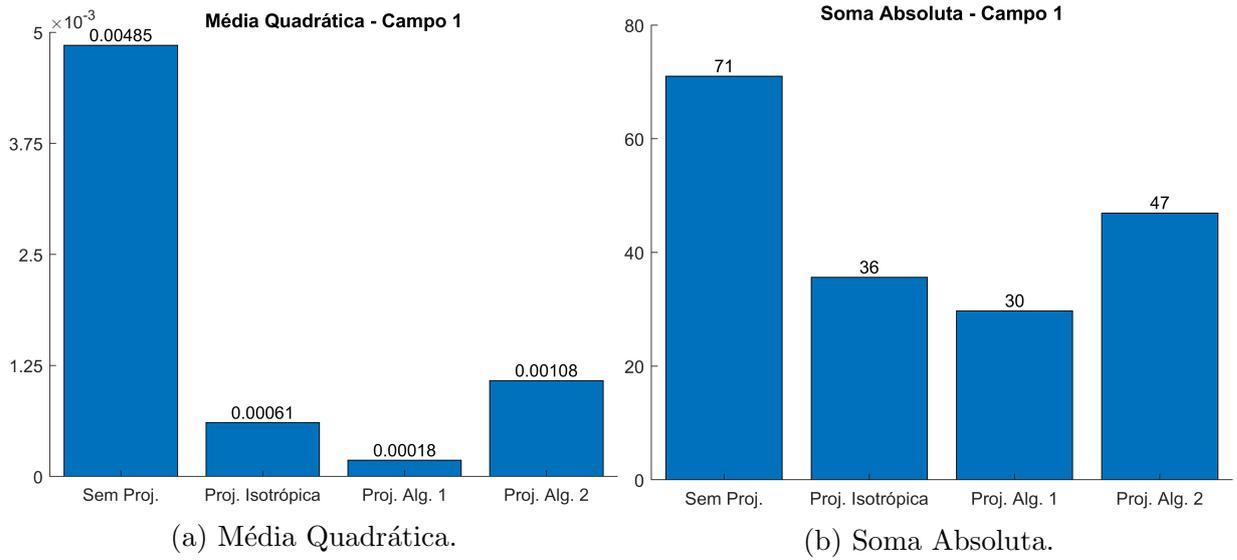


Figura 5.54: Dados relativos ao Campo 1.

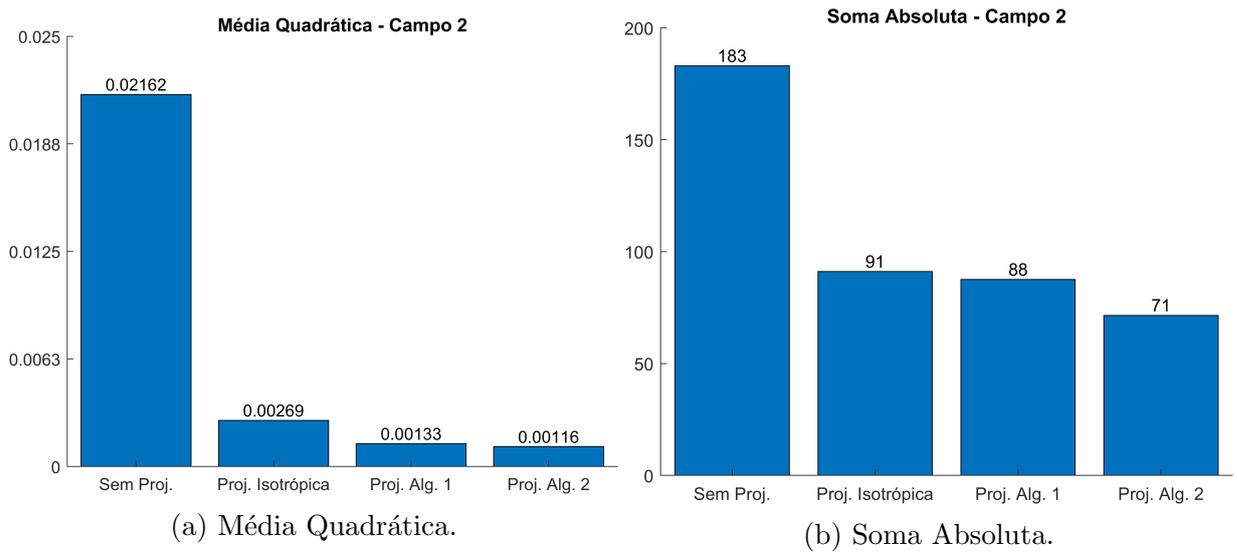


Figura 5.55: Dados relativos ao Campo 2.

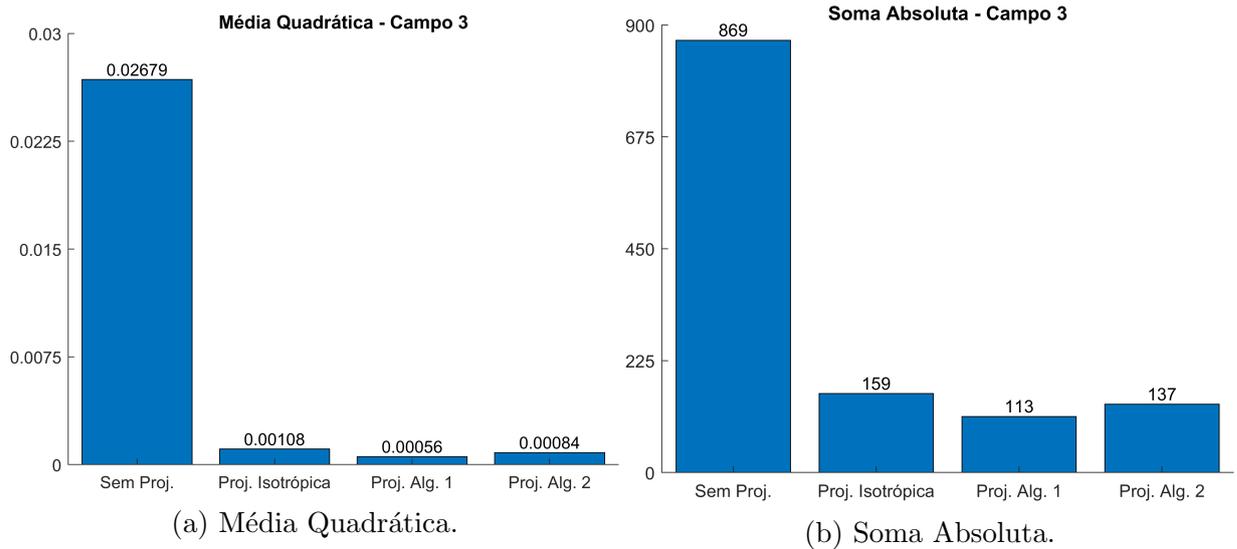


Figura 5.56: Dados relativos ao Campo 3.

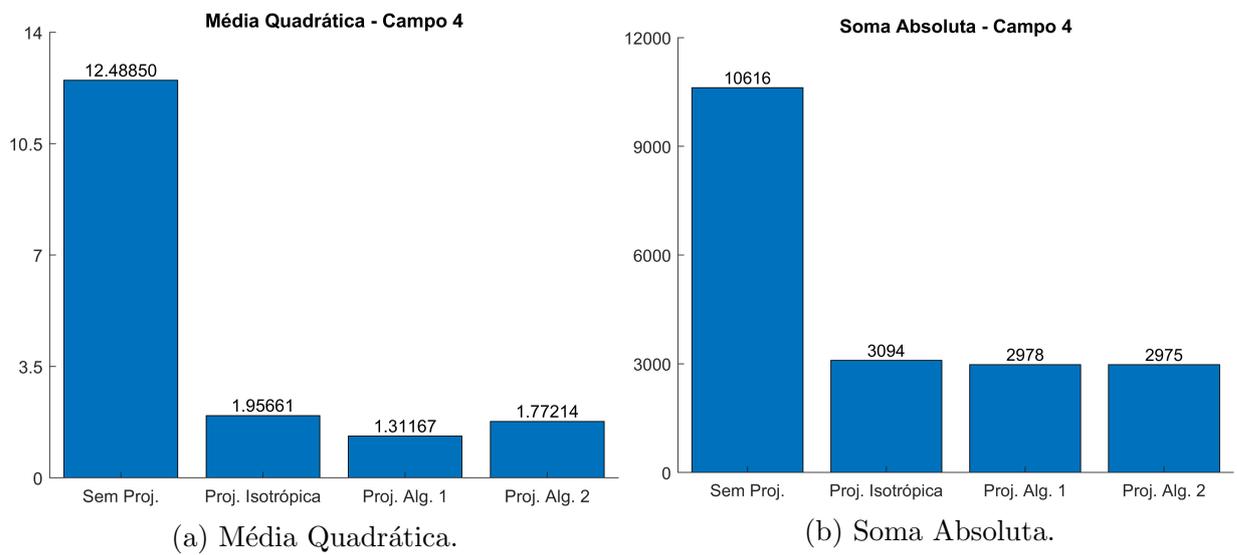


Figura 5.57: Dados relativos ao Campo 4.

5.4.2 Proporções entre Valores

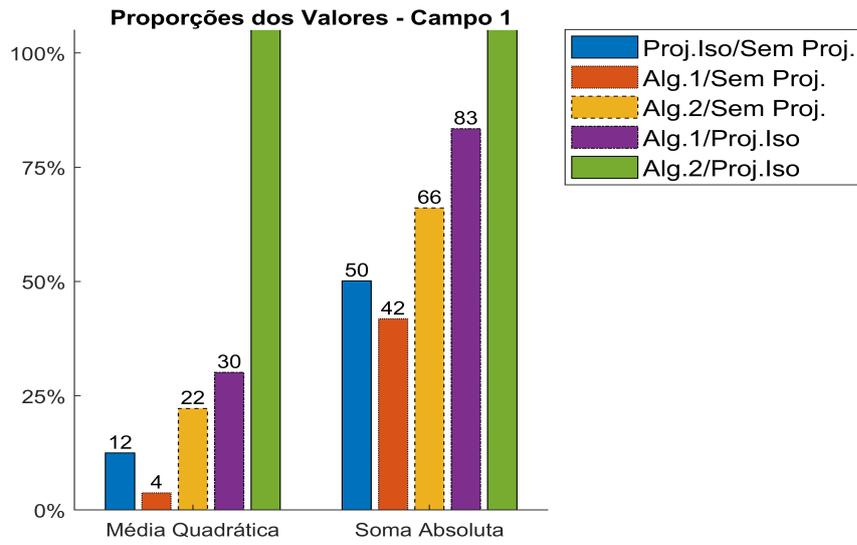


Figura 5.58: Proporções entre os Dados relativos ao Campo 1.

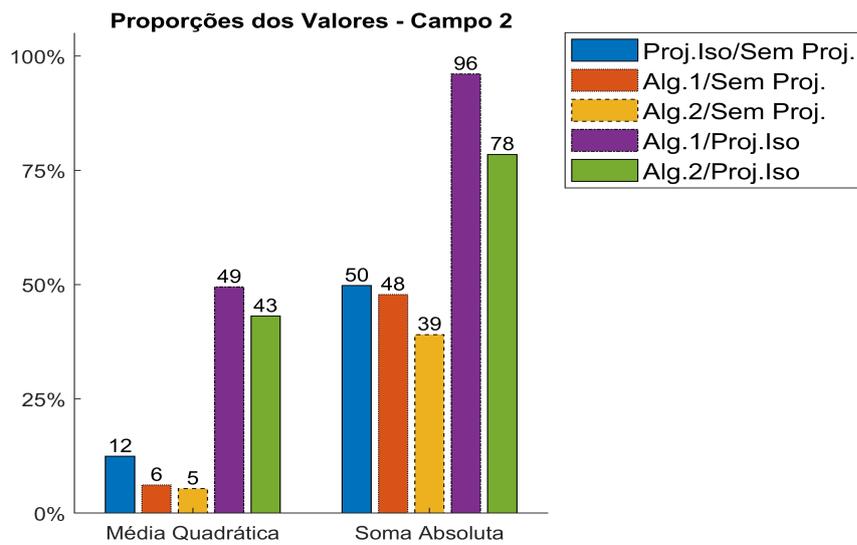


Figura 5.59: Proporções entre os Dados relativos ao Campo 2.

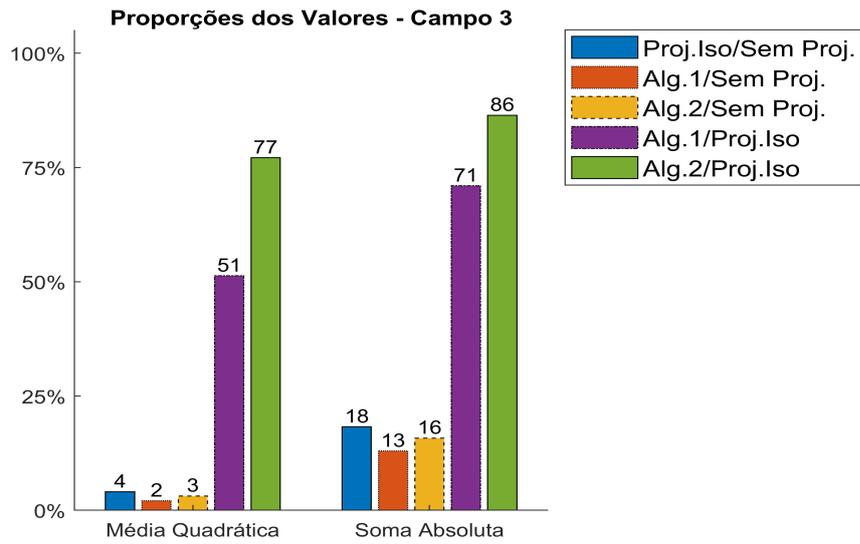


Figura 5.60: Proporções entre os Dados relativos ao Campo 3.

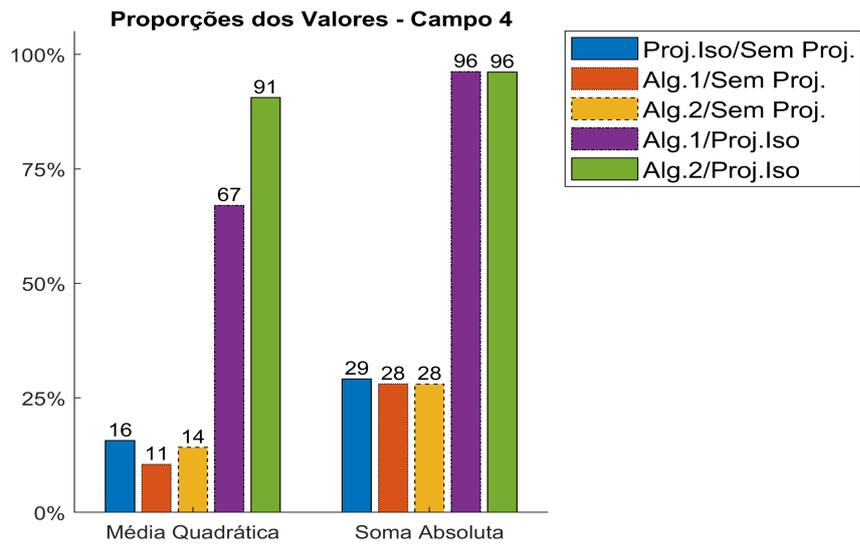


Figura 5.61: Proporções entre os Dados relativos ao Campo 4.

5.4.3 Reduções Obtidas

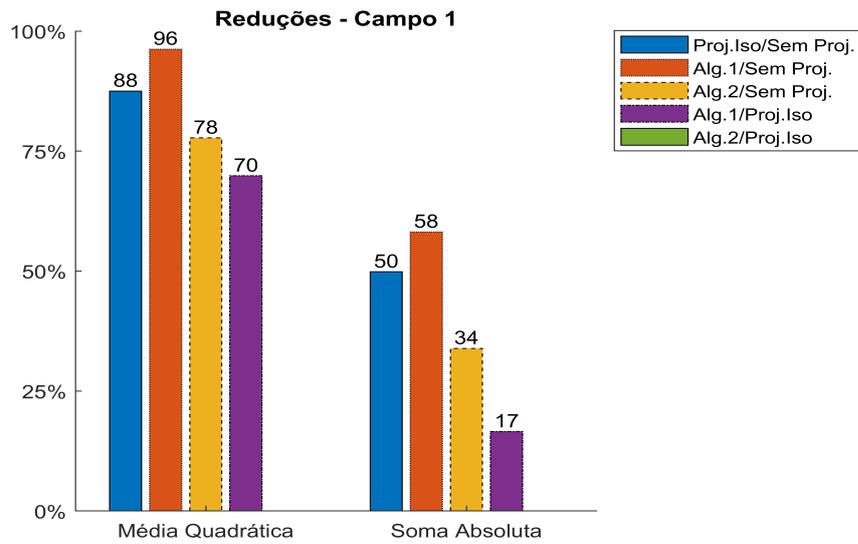


Figura 5.62: Reduções obtidas em relação ao Campo 1.

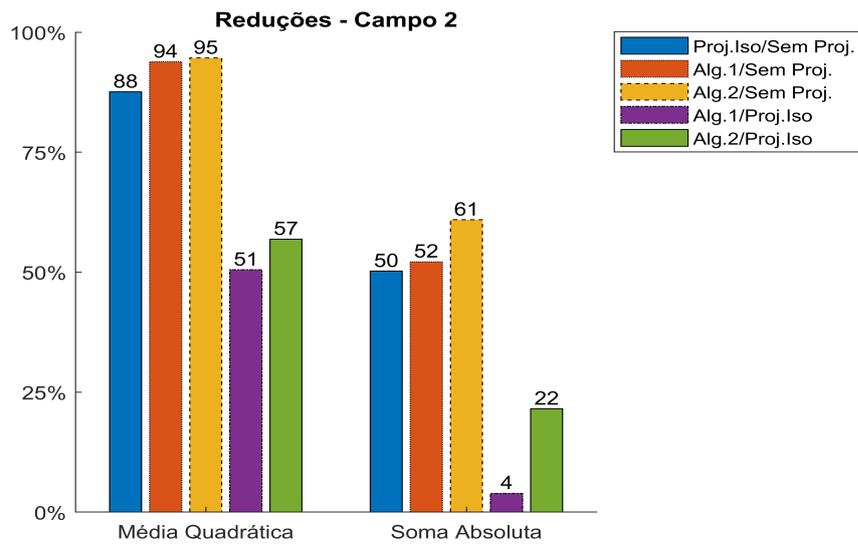


Figura 5.63: Reduções obtidas em relação ao Campo 2.

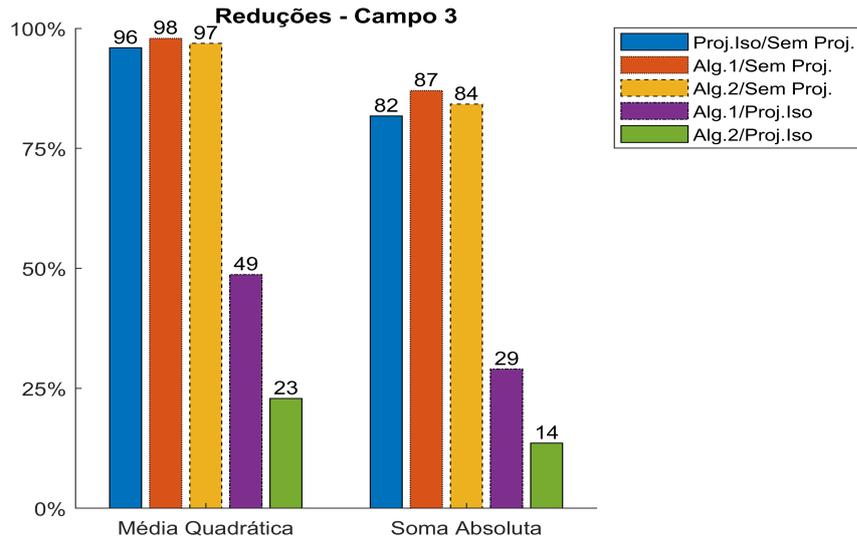


Figura 5.64: Reduções obtidas em relação ao Campo 3.

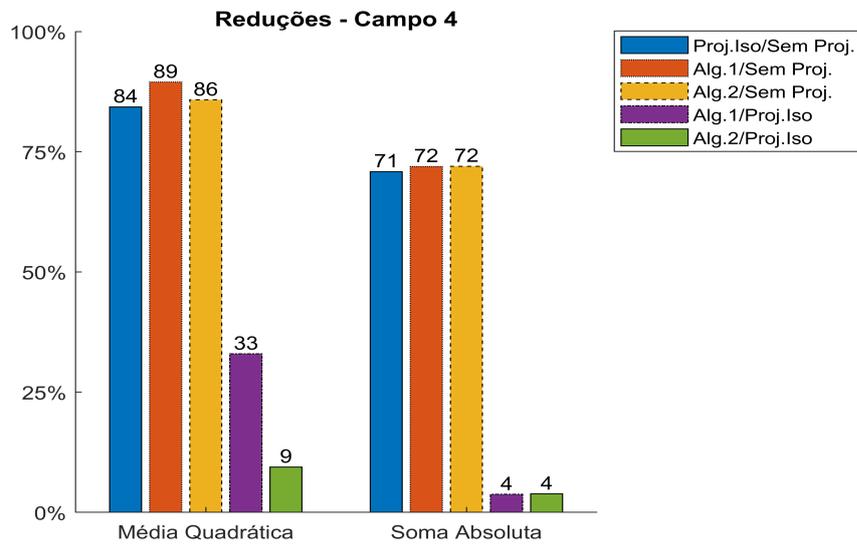


Figura 5.65: Reduções obtidas em relação ao Campo 4.

6 Conclusão

Neste trabalho foi desenvolvido um método que busca campos tensoriais utilizando a meta-heurística *Simulated Annealing*. Os campos foram gerados a partir de um campo vetorial arbitrário e de um campo tensorial isotrópico. Durante todo o desenvolvimento do método foram utilizados diversos campos vetoriais, diversas modificações foram realizadas nos algoritmos, diferentes configurações de parâmetros foram testadas, e também diferentes quantidades de iterações.

A partir dos resultados obtidos foi possível observar que o método desenvolvido tende a alterar significativamente os campos vetoriais originais e, em casos específicos, até anular o fluxo dos vetores em determinada região do campo. Portanto, o emprego do método pode não ser desejável caso o fluxo do campo tenha que se manter o mais fiel possível ao fluxo original.

Uma das principais limitações do método está relacionada ao elevado custo computacional, dado que para se obter uma redução significativa em relação à projeção isotrópica, são necessárias dezenas de milhares de iterações. As 30 mil iterações definidas como critério de parada para realização dos testes por exemplo, levaram cerca de 48 horas de execução com pequenas variações entre os testes. O emprego de grandes malhas pode tornar ainda mais proibitivo a utilização do método. Como uma possível solução para a redução do custo computacional, é possível o desenvolvimento de um estado inicial mais favorável para a execução dos algoritmos, dado que, neste trabalho, o ponto de partida foi um campo tensorial isotrópico. Com esse ponto de partida as informações relevantes do campo vetorial original, ou até de uma projeção isotrópica inicial, não foram aproveitadas para a construção de um estado que proporcionasse um custo computacional menor.

Com base nos resultados obtidos, é possível concluir que o método proposto pode obter um campo tensorial anisotrópico de modo a reduzir significativamente o divergente de um campo vetorial arbitrário utilizando a decomposição anisotrópica de Helmholtz. No entanto, é necessário um estudo mais aprofundado para que o método possa ser utilizado

em aplicações práticas.

Bibliografia

- Bertsimas, D.; Tsitsiklis, J. ; others. Simulated annealing. **Statistical science**, v.8, n.1, p. 10–15, 1993.
- Bridson, R. **Fluid simulation for computer graphics**. AK Peters/CRC Press, 2015.
- Dassios, G.; Lindell, I. V. Uniqueness and reconstruction for the anisotropic helmholtz decomposition. **Journal of Physics A: Mathematical and General**, v.35, n.24, p. 5139, 2002.
- Haeser, G.; Ruggiero, M. G. Aspectos teóricos de simulated annealing e um algoritmo duas fases em otimização global. **TEMA-Tendências em Matemática Aplicada e Computacional**, v.9, n.3, p. 395–404, 2008.
- Ingber, L. Simulated annealing: Practice versus theory. **Mathematical and computer modelling**, v.18, n.11, p. 29–57, 1993.
- Kindlmann, G. **Superquadric tensor glyphs**. In: Proceedings of the Sixth Joint Eurographics-IEEE TCVG conference on Visualization, p. 147–154, 2004.
- Kirkpatrick, S.; Gelatt, C. D. ; Vecchi, M. P. Optimization by simulated annealing. **science**, v.220, n.4598, p. 671–680, 1983.
- Renhe, M. C.; Vieira, M. B. ; Esperança, C. A stable tensor-based method for controlled fluid simulations. **Applied Mathematics and Computation**, v.343, p. 195–213, 2019.
- Ribeiro, A. A.; Renhe, M. C.; Vieira, M. B.; Giraldi, G. ; Esperança, C. **Anisotropic fluid modeling and simulation using tensor fields**. In: 31th SIBGRAPI, Foz do Iguaçu, Brazil, October 29 - November 01, 2018, 2018 (Submitted).
- Stam, J. **Stable fluids**. In: Proceedings of the 26th annual conference on Computer graphics and interactive techniques, p. 121–128. ACM Press/Addison-Wesley Publishing Co., 1999.
- Vieira, M. B.; Giraldi, G. A.; Ribeiro, A. C. A.; Renhe, M. C. ; Esperança, C. Anisotropic helmholtz decomposition for controlled fluid simulation. **Applied Mathematics and Computation**, December 2019, Submitted.