

UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI

Ana Roberta Melo

**PARALELIZAÇÃO EM MEMÓRIA DISTRIBUÍDA E MULTI-GPU DE  
UM SIMULADOR DA ELETROFISIOLOGIA CARDÍACA**

São João del-Rei

2023

UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI

Ana Roberta Melo

**PARALELIZAÇÃO EM MEMÓRIA DISTRIBUÍDA E MULTI-GPU DE  
UM SIMULADOR DA ELETROFISIOLOGIA CARDÍACA**

Dissertação apresentada como requisito para  
obtenção do título de mestre em Ciências no  
Curso de Mestrado do Programa de Pós Gra-  
duação em Ciência da Computação da UFSJ.

Orientador: Dr. Rafael Sachetto Oliveira

Universidade Federal de São João del-Rei – UFSJ

Mestrado em Ciência da Computação

Linha de pesquisa: Sistemas Distribuídos e Computação de Alto Desempenho

São João del-Rei

2023

Ana Roberta Melo

**PARALELIZAÇÃO EM MEMÓRIA DISTRIBUÍDA E MULTI-GPU DE  
UM SIMULADOR DA ELETROFISIOLOGIA CARDÍACA**

Dissertação apresentada como requisito para  
obtenção do título de mestre em Ciências no  
Curso de Mestrado do Programa de Pós Gra-  
duação em Ciência da Computação da UFSJ.

Trabalho aprovado. São João del-Rei, 8 de março de 2023:

---

**Dr. Rafael Sachetto Oliveira**

---

**Dra. Carolina Ribeiro Xavier**

---

**Dr. Rodrigo Weber dos Santos**

São João del-Rei

2023

*Este trabalho é dedicado a única mãe no mundo que poderia me aguentar, Maria de Fátima Moreira de Melo e, ao meu melhor amigo e companheiro, Mário Luís.*

## AGRADECIMENTOS

Ao meu orientador, Rafael Sachetto, por me indicar o tema aqui proposto, além da paciência e disponibilização de partes essenciais para este projeto. Agradeço-o também por me inspirar ao longo da graduação, mostrando o quão bom um Professor pode ser e dominar diversas áreas.

À minha mãe e meu namorado pelo imenso apoio para que tudo desse certo e, por acreditarem em mim quando nem eu acredito. Definição de amor, é o que sinto por vocês.

Aos meus irmãos, Luiz Felipe e Paulo Henrique, as minhas duas tia-mãe, Lucimary e Sônia, e aos pais do meu namorado, Maria e Mário, por cederem sua casa. Amo todos vocês, muito e sempre.

Agradeço ao Programa de Pós Graduação por todo o conhecimento compartilhado e à UFSJ pelo fomento e por ter sido meu segundo lar por anos. Aos membros da minha banca examinadora, agradeço por tão prontamente aceitarem julgar todo o conteúdo desta dissertação.

Por fim, mas não menos importante, agradeço aos verdadeiros amigos que sempre estão ao meu lado.

Todos vocês fizeram ser possível a finalização desse projeto, juntamente com outros fatores...

O meu muito obrigado, de coração!

Em memória de meu pai...

*"Home is the place where when you have to go, there they have to take you in."*

*"Lar é o lugar onde, quando você tem que ir, eles têm que te acolher."*

*(Robert Frost - The Death of the Hired Man)*

## RESUMO

As doenças cardiovasculares são as principais causas de morte no mundo e muitas dessas enfermidades requerem um entendimento profundo e detalhado das alterações eletrofisiológicas, para o estudo de novos fármacos e dispositivos clínicos para auxiliar no tratamento. Consequentemente, as simulações numéricas surgem como uma ferramenta relevante na investigação dessas alterações eletrofisiológicas em doenças cardíacas, mas uma das principais complexidades da aplicação de modelos matemáticos da eletrofisiologia cardíaca é recair em sistemas de equações diferenciais parciais (EDPs) e de equações diferenciais ordinárias (EDOs) de elevado número de incógnitas. O trabalho aqui desenvolvido consistiu na implementação de uma versão da solução das EDOs em memória distribuída e multi-GPU (*Graphics Processing Unit*), de um simulador da eletrofisiologia cardíaca já existente, como forma de acelerar a solução de modelos dessa classe. Para isso, foi adotado o modelo monodomínio de membrana celular e modelo de dinâmica celular Bondarenko, juntamente com passos de tempo adaptativos. O simulador foi executado em um domínio *benchmark*. Para realização dos testes, foram utilizadas placas de vídeo de diferentes desempenho. A combinação de duas placas de vídeo de menor desempenho possibilitou aceleração superior a 1,5 vezes para casos onde foi utilizada a malha mais refinada e, um balanceamento de carga entre placas de diferentes desempenhos foi o que proporcionou os melhores resultados. Os resultados mostraram que a implementação proposta foi capaz de acelerar a solução das EDOs em diferentes cenários, demonstrando ser uma ferramenta importante para simulações numéricas de problemas complexos de eletrofisiologia cardíaca.

**Palavras-chaves:** Memória distribuída; multi-GPU; eletrofisiologia cardíaca; simulação numérica; computação de alto desempenho.

## ABSTRACT

Cardiovascular diseases are the main causes of death on the world. Many of these diseases require a deep and detailed understanding of electrophysiological changes for to the study of new drugs and clinical devices to aid in the treatment. Consequently, numerical simulations emerge as a relevant tool in the investigation of these electrophysiological changes in heart disease. However, the complexity of applying mathematical models of cardiac electrophysiology falls into systems of differential equations with a high number of unknowns, demanding great computational effort. The work developed here consisted of the implementation of a distributed memory parallelization and multi-GPU (Graphics Processing Unit) version of an existing cardiac electrophysiology simulator with the aim of accelerate the solution of models of this class. The monodomain cell membrane model and Bondarenko cell dynamics model were adopted, together with adaptive time steps. The simulator was run in a benchmark domain. Graphics cards of different performance were used to carry out the tests. The combination of two lower performance graphics cards allowed acceleration of more than 1.5 times for cases where the most refined mesh was used. Load balancing between graphics cards of different performance provided the best results. The results showed that the proposed implementation was able to accelerate the solution of ODEs in different scenarios, proving to be an important tool for numerical simulations of complex cardiac electrophysiology problems.

**Key-words:** Distributed computing; multi-GPU; cardiac electrophysiology; numerical simulation; high performance computing.

## LISTA DE FIGURAS

Figura 1 – Forma representativa do PA ao longo do tempo, mostrando as cinco fases, para o átrio e ventrículo. Densidade de corrente interna em azul (despolarizante) e densidade de corrente externa em vermelho (repolarizante) . . . . .	22
Figura 2 – (a) Sistema condutor cardíaco; (b) Tempo de aparecimento do impulso elétrico nas diferentes partes do coração (em frações de segundo, após o aparecimento no nodo sinoatrial). A-V: atrioventricular; S-A: sinoatrial	24
Figura 3 – Descarga rítmica do nodo sinusal. Comparação do potencial de ação do nodo sinusal com o da fibra muscular ventricular . . . . .	25
Figura 4 – Alteração do PA pós infarto. (a) PA do grupo de controle e das áreas de fronteira e remotas da região infartada; (b) Platô <sub>75</sub> e Triangulação do PA; (c) Potencial de repouso e pico do PA; (d) Taxa de despolarização e repolarização. $V_m$ : potencial de membrana. $V_{rep}$ : potencial de repouso. $V_{pico}$ : pico do PA. $dV/dt_{max}$ : taxa de variação do potencial de membrana. DPA: duração do PA. número de amostras = 19-26 células/5-7 animais. ANOVA com pós teste Bonferroni. n.s.: não significativo. * $P < 0,05$ , ** $P < 0,01$ , *** $P < 0,001$ . . . . .	28
Figura 5 – PA ventricular e as principais correntes iônicas para os grupos de controle, IC-Remota e IC-Fronteira. (a) Correntes iônicas de influxo; (b) Correntes iônicas de efluxo. Número de amostras = 6–16 células/4–6 animais. $I_{NaL}$ : corrente de $Na^+$ por canais do tipo L. $I_{NCX}$ : corrente de troca de $Na^+/Ca^{2+}$ . $I_{CaL}$ : corrente de $Ca^{2+}$ por canais do tipo L. $I_{K1}$ : corrente retificadora de influxo de $K^+$ . $I_{Ks}$ : corrente de $K^+$ por canais lentos. $I_{Kr}$ : corrente de $K^+$ por canais rápidos . . . . .	30
Figura 6 – Analogia do comportamento elétrico da membrana celular com um circuito elétrico resistor-capacitor em paralelo . . . . .	31
Figura 7 – Exemplos de cadeias de Markov. (a) Um modelo de quatro estados com duas transições independentes - C: fechado; O: aberto; $I_O$ : aberto/inativado $I_C$ : fechado/inativado. $\alpha$ e $\beta$ : taxas de transição entre $I_O$ e $I_C$ e entre C e O. $\gamma$ e $\delta$ : taxas de transição entre C e $I_C$ e entre O e $I_O$ ; (b) Um modelo de três estados com transições dependentes de C para O e O para I . . . . .	39
Figura 8 – Circuito elétrico proposto por Hodgkin e Huxley representando a membrana e representação esquemática das correntes iônicas. $R_{Na}=1/g_{Na}$ ; $R_K=1/g_k$ ; $R_l=1/g_l$ . . . . .	40
Figura 9 – Representação esquemática das correntes iônicas do modelo Bondarenko	42
Figura 10 – Grafo resultante após a discretização de um cubo unitário . . . . .	49

Figura 11	– Exemplo da implementação proposta da computação em memória distribuída multi-GPU para solução das EDOs . . . . .	60
Figura 12	– Domínio <i>benchmark</i> adotado nos testes. S: local de estímulo do potencial de ação . . . . .	61
Figura 13	– Despolarização do domínio <i>benchmark</i> . . . . .	62
Figura 14	– Tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando duas placas de vídeo NVIDIA GeForce GTX 960. a: aceleração . . . . .	63
Figura 15	– Comparação do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio utilizando as placas NVIDIA GeForce GTX 960 e NVIDIA TITAN Xp individualmente . . . . .	65
Figura 16	– <b>Comparação do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio utilizando apenas paralelização distribuída em CPUs. np: número de processos</b> . . . . .	68
Figura 17	– Representação esquemática das correntes iônicas do modelo Luo Rudy	79
Figura 18	– Representação esquemática das correntes iônicas do modelo Ten Tusscher. RS: Retículo sarcoplasmático . . . . .	80

## LISTA DE TABELAS

Tabela 1 – Média e desvio padrão de cinco testes do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando duas placas de vídeo NVIDIA GeForce GTX 960. Linha de base: NVIDIA GeForce GTX 960 . . . . .	64
Tabela 2 – Média e desvio padrão de cinco testes do tempo de execução global do algoritmo em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando duas placas de vídeo NVIDIA GeForce GTX 960. Linha de base: NVIDIA GeForce GTX 960 . . . . .	64
Tabela 3 – Média e desvio padrão de cinco testes do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando uma placa de vídeo NVIDIA TITAN Xp e uma NVIDIA GeForce GTX 960. Linha de base: NVIDIA TITAN Xp . . . . .	66
Tabela 4 – Análise do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando uma placa de vídeo NVIDIA GeForce GTX 960 e uma NVIDIA TITAN Xp com balanceamento de carga. Amostra de cinco testes. Aceleração em relação à utilização de um processo com a placa de vídeo NVIDIA TITAN Xp FB: Fator de balanceamento . . . . .	66
Tabela 5 – Tempo de execução das EDOs em relação ao tempo de execução global do algoritmo de todos os testes realizados . . . . .	67
Tabela 6 – Tempo de execução da algoritmo utilizando apenas paralelização distribuída em CPUs . . . . .	69
Tabela A1 –Resumo do tempo de execução das EDOs de todos os testes realizados. BC: Balanceamento de carga . . . . .	85
Tabela A2 –Resumo do tempo de execução global do algoritmo de todos os testes realizados. BC: Balanceamento de carga . . . . .	85

## LISTA DE ABREVIATURAS E SIGLAS

3D	3 dimensões
A-V	Átrio-ventricular
ALG	Autonomous Leaves Graph
API	Application Programming Interface
ATP	Adenosina trifosfato
BC	Balanceamento de carga
BDK	Bondarenko
CICR	Corrente de liberação de cálcio induzida por cálcio
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture
DCV	Doenças cardiovasculares
DPA	Duração do potencial de ação
EDO	Equação diferencial ordinária
EDP	Equação diferencial parcial
FB	Fator de balanceamento
GC	Gradiente conjugado
GHK	Goldman-Hodgkin-Katz
GPGPU	General-Purpose Graphics Processing Unit
GPU	Graphics Processing Unit
HH	Hodgkin-Huxley
IC	Insuficiência cardíaca
MONOALG3D	Algoritmo de simulação 3D em modelo monodomínio
MPI	Message Passing Interface
MVF	Método dos volumes finitos

PA	Potencial de ação
RAM	Random Access Memory
RK2	Runge-Kutta de segunda ordem
RL	Rush-Larsen
RS	Retículo sarcoplasmático
S-A	Sinoatrial
SV	Vetor de estado
TT2	Ten Tusscher-Panfilov
TT3	Ten Tusscher-Panfilov 2
UFSJ	Universidade Federal de São João del-Rei

## LISTA DE SÍMBOLOS

$a$	Aceleração
$\alpha, \beta, \gamma, \delta$	Taxas de transição
$\beta_i$	Coefficiente de partição água-membrana
$\sigma_{\text{int}}$	Tensor de condutividade interno
$\sigma_{\text{ext}}$	Tensor de condutividade externo
$\phi_e$	Potencial extracelular
$\phi_i$	Potencial intracelular
$Ca^{2+}$	Íons de Cálcio
$c_{i\text{ext}}$	Concentração externa do íon $i$
$c_{i\text{int}}$	Concentração interna do íon $i$
$C_m$	Capacitor; capacitância da membrana
$D_i$	Coefficiente de difusão do íon $i$
$DPA_{50}$	Duração do potencial de ação com 50% de repolarização
$DPA_{95}$	Duração do potencial de ação com 95% de repolarização
$E_i$	Potencial de equilíbrio de Nernst para o íon $i$
$E_k$	Potencial de reversão de potássio
$E_r$	Potencial resultante
$F$	Constante de Faraday
$g_i$	Condutância máxima do íon $i$
$g_{i\text{ATP}}$	Condutância do íon $i$ dependente da adenosina trifosfato
$g_{i,\text{max}}$	Condutância máxima de $i$
$h$	Espessura da membrana
$i$	Íon $i$
$I_{b_i}$	Corrente de background do íon $i$

$I_C$	Corrente capacitiva
$I_{CaL}$	Corrente de íons de cálcio que atravessa o canal do tipo L
$I_{Cl,Ca}$	Corrente de íons de cloro ativada por íons de cálcio
$I_{ext}$	Corrente externa
$I_f$	Corrente de íons que atravessa o canal estimulado por hiperpolarização
$I_{int}$	Corrente interna
$I_{ion}$	Corrente iônica
$I_K$	Corrente de íons que atravessa o canal de potássio
$I_{K1}$	Corrente retificadora de influxo de íons de potássio
$I_{K(ATP)}$	Corrente de íons de potássio ativada por adenosina trifosfato
$I_{Kp}$	Corrente de platô de íons de potássio
$I_{Kr}$	Corrente de íons de potássio por canais rápidos
$I_{Ks}$	Corrente de íons de potássio por canais lentos
$I_{Kss}$	Corrente ativada por tensão de estado estacionário não inativante de íons de potássio
$I_{Kt0,f}$	Corrente de saída transitória de rápida recuperação de íons de potássio
$I_{Kt0,s}$	Corrente de saída transitória de lenta recuperação de íons de potássio
$I_{Kur}$	Corrente retificadora de íons de potássio por canais ultra-rápido
$I_l$	"Corrente de fuga"
$I_{leak}$	"Corrente de fuga" do retículo sarcoplasmático
$I_m$	Corrente transmembrânica
$I_{Na}$	Corrente de íons de sódio que atravessa o canal rápido
$I_{Na/Ca}, I_{NCX}$	Corrente de troca de íons sódio-cálcio
$I_{Na/K}$	Corrente bomba de sódio-potássio
$I_{NaL}$	Corrente de íons de sódio que atravessa o canal do tipo L
$I_{pCa}$	Corrente de platô de íons de cálcio

$I_{p(\text{Ca})}$	Corrente bomba de íons de cálcio
$I_{pK}$	Corrente de platô de íons de potássio
$I_{\text{rel}}$	Corrente de liberação de cálcio induzida por cálcio
$I_{\text{stim}}$	Corrente de estímulo
$I_t$	Corrente total
$I_{t0}$	Corrente que atravessa o canal transiente de efluxo
$I_{\text{up}}$	Corrente bomba para inserção de cálcio no retículo sarcoplasmático
$K_o$	Concentração de potássio extracelular
$K^+$	Íons de Potássio
$N_i$	Número de canais do íon i
$Na^+$	Íons de Sódio
$P_i$	Permeabilidade da membrana ao íon i
$Plato_{75}$	Potenciais de platô até 75% do potencial de ação
Q	Carga
R	Constante universal dos gases
$R_K$	Resistência potássio
$R_m$	Resistor não linear
$R_{Na}$	Resistência sódio
T	Temperatura absoluta
V	Potencial
$V_{\text{ext}}$	Potencial elétrico externo
$V_{\text{int}}$	Potencial elétrico interno
$V_m$	Potencial de membrana
$V_{\text{pico}}$	Pico do potencial de ação
$V_{\text{rep}}$	Potencial de repouso
$z_i$	Valência do íon i

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>18</b>
1.1	Objetivos e Contribuições	19
1.2	Organização do Texto	20
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA</b>	<b>21</b>
2.1	Eletrofisiologia cardíaca	21
2.2	Exemplos de patologias cardíacas	26
2.2.1	Arritmias cardíacas	26
2.2.2	Infarto do miocárdio	27
2.3	Modelagem da eletrofisiologia cardíaca	31
2.3.1	Modelo matemático para membrana celular	31
2.3.2	Modelos para corrente iônica	32
2.3.3	Modelos para o tecido	34
2.3.3.1	Modelo bidomínio	34
2.3.3.2	Modelo monodomínio	35
2.3.4	Modelos para canais iônicos	36
2.3.4.1	Modelo de dois estados	36
2.3.4.2	Modelo de subunidades	37
2.3.4.3	Modelo de cadeias de Markov	38
2.3.5	Modelos Celulares	40
2.4	Programação paralela, distribuída, memória compartilhada e GPU	43
2.5	Estratégias de implementação das simulações numéricas da eletrofisiologia cardíaca	45
<b>3</b>	<b>MATERIAIS E MÉTODOS</b>	<b>49</b>
3.1	Estrutura de dados ALG	49
3.2	Discretização das equações governantes	50
3.2.1	Método dos volumes finitos aplicado ao modelo monodomínio (EDP)	51
3.2.1.1	Discretização temporal (EDP)	51
3.2.1.2	Discretização espacial (EDP)	52
3.2.2	MVF aplicado ao ALG	53
3.2.3	Discretização das EDOs	55
3.2.3.1	Passo de tempo adaptativo (EDOs)	56
3.3	Implementação em computação distribuída e multi-GPU	58
3.3.1	Especificações do ambiente de execução	60
3.4	Domínio e modelo celular	61
<b>4</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>62</b>

<b>5 CONCLUSÃO E TRABALHOS FUTUROS</b> . . . . .	<b>70</b>
5.1 Conclusão . . . . .	70
5.2 Trabalhos futuros . . . . .	71
<b>REFERÊNCIAS</b> . . . . .	<b>72</b>
<b>APÊNDICES</b>	<b>78</b>
<b>APÊNDICE A MODELAGEM DA ELETROFISIOLOGIA CARDÍACA</b>	<b>79</b>
A.1 Modelo Luo Rudy . . . . .	79
A.2 Modelos Ten Tusscher, Ten Tusscher-Panfilov (TT2) e Ten Tusscher-Panfilov 2 (TT3) . . . . .	80
A.3 Modelo Oliveira (MONOALG3D) . . . . .	82
<b>APÊNDICE B MÉTODO DO GRADIENTE CONJUGADO (GC)</b> . .	<b>84</b>
<b>APÊNDICE C TABELA-RESUMO DOS TESTES</b> . . . . .	<b>85</b>

## 1 INTRODUÇÃO

Mais pessoas morrem anualmente de doenças cardiovasculares (DCV) do que de qualquer outra causa, sendo estimado que 17,9 milhões de pessoas morreram de DCV em 2016, representando 31% de todas as mortes globais (WHO, 2020). Um subgrupo das DCV são as doenças isquêmicas do coração, as quais segundo a Global Health Metrics (JAMES *et al.*, 2018), em 2017, tiveram incidência de 10.636.500 e prevalência de 126.451.500, no mundo. Entende-se por incidência a ocorrência de novos casos, enquanto a prevalência se refere a casos novos e casos existentes da doença.

O infarto, também chamado de ataque cardíaco, ocorre quando o fluxo de sangue é interrompido e não há recepção de sangue e oxigênio no coração suficientes para mantê-lo em atividade, ocorrendo necrose onde o fluxo foi interrompido. A principal causa de morte devido ao infarto do miocárdio é a fibrilação e, conseqüentemente, o desenvolvimento de arritmias, que ocorre após a isquemia cardíaca. O desenvolvimento de procedimentos terapêuticos, para um possível decréscimo na tendência de mortalidade, requer um entendimento profundo e detalhado das alterações eletrofisiológicas.

Nesse contexto, as simulações numéricas são uma importante ferramenta na investigação dessas alterações eletrofisiológicas em doenças cardíacas. Mais que isso, a crescente disponibilidade de dados clínicos possibilita uma melhor compreensão dos mecanismos fisiológicos e patológicos do sistema cardiovascular, de forma que os modelos matemáticos e numéricos fornecem uma ferramenta complementar relevante, capaz não apenas de reproduzir indicadores clínicos, mas também de prever e explorar cenários não vistos (PAGANI *et al.*, 2021).

O desenvolvimento de técnicas que permitam a integração da computação de alto desempenho à simulação numérica da eletrofisiologia cardíaca em domínios paciente-específico é uma necessidade. Técnicas recentes de terapias e diagnósticos personalizados em tempo real, tais como gêmeos digitais (*digital twins*) e medicina de precisão. Os conceitos de gêmeos digitais e medicina de precisão são interligados. Enquanto a medicina de precisão usa métodos científicos para personalizar o tratamento médico de acordo com os genes, anatomia, fisiologia e estilo de vida, os gêmeos digitais cardíacos são réplicas digitais de corações que integram continuamente dados clínicos do paciente adquiridos ao longo do tempo (PEIRLINCK *et al.*, 2021; GILLETTE *et al.*, 2021). Os gêmeos digitais podem auxiliar na tomada de decisão clínica, planejamento e diagnóstico, além de elevar o custo-benefício, segurança e ética em testes de novas terapias e dispositivos de eletrofisiologia cardíaca (GILLETTE *et al.*, 2021). No entanto, o desenvolvimento de gêmeos digitais de alta fidelidade totalmente personalizados em termos de anatomia e eletrofisiologia para construção de coortes virtuais em escala ainda estão em fase inicial (NIEDERER *et al.*,

2020).

Modelos matemáticos de eletrofisiologia cardíaca são sistemas dinâmicos que descrevem os processos elétricos que ocorrem no tecido cardíaco, em diferentes escalas, desde o nível celular até o tecidual. A variação do potencial elétrico através da membrana celular se deve ao desequilíbrio da concentração iônica entre os meios extracelular e intracelular. A troca dessas espécies, que causa a sucessiva despolarização e repolarização das células são possibilitadas pelos canais iônicos (PAGANI *et al.*, 2021). Os modelos atuais de eletrofisiologia cardíaca incluem informações detalhadas sobre os processos iônicos subjacentes ao potencial de ação, como as correntes iônicas de sódio, potássio e cálcio, trocadores, como o de Na/Ca e bombas, como a de Na/K. Embora incompleto, esse conhecimento tem sido usado para construir modelos computacionais que representam a atividade do coração como um todo, e de suas câmaras individuais (átrios e ventrículos), tecidos e células. Esses modelos também começam a ser usados para orientar as decisões sobre o tratamento de pacientes e o desenvolvimento de novos medicamentos pela indústria farmacêutica.

Os modelos celulares recentes que descrevem o comportamento da eletrofisiologia cardíaca exigem grande esforço computacional, devido principalmente ao grande número de variáveis de estado. Além disso, a utilização de domínios computacionais realísticos, obtidos por meio de tomografias computadorizadas, necessitam de malhas refinadas para obtenção de resultados confiáveis, estabilidade e convergência dos modelos numérico, o que contribui para o aumento do esforço computacional.

## 1.1 Objetivos e Contribuições

A complexidade da execução de simulações numéricas cardíacas em malhas com um grande número de nós e a aplicação de complexos modelos matemáticos de eletrofisiologia cardíaca cada vez mais realísticos, recaem em sistemas de equações diferenciais de elevado número de incógnitas demandando forte esforço computacional. Além disso, para uma melhor análise do problema, podem ser necessárias diversas simulações, sobre diferentes cenários (OLIVEIRA *et al.*, 2018). Por fim, os requisitos de memória das simulações também representam um problema, pois tornam-se cada vez maiores. Aliado a esses fatores, tem sido verificado que as soluções das EDOs podem representar o consumo de mais de 80% do tempo de execução das simulações de eletrofisiologia cardíaca (OLIVEIRA *et al.*, 2018b; OLIVEIRA, 2013). Dessa forma, a pesquisa aqui desenvolvida se propõe a reduzir o tempo de resolução das EDOs e, conseqüentemente, o tempo global, de forma a contribuir com uma ferramenta mais eficiente para simulação da eletrofisiologia cardíaca.

Assim, o objetivo principal dessa pesquisa consiste em implementar a paralelização em memória distribuída e multi-GPU (*Graphics Processing Unit*) da solução das EDOs

de um simulador de eletrofisiologia cardíaca já implementado - *MONOALG3D* - o qual por sua vez combina estratégias de unidade de processamento gráfico, passos de tempo adaptativos e paralelização em memória compartilhada.

Os objetivos específicos são:

1. Utilizar da computação distribuída visando reduzir o tempo global da solução de problemas de eletrofisiologia cardíaca e, com isso, possibilitar a resolução de problemas mais complexos.
2. Analisar a aceleração obtida na solução das EDOs pela paralelização em memória distribuída com a utilização de multi-GPUs.
3. Usar simultaneamente diferentes modelos de placas de vídeo para executar a solução das EDOs, e analisar seus desempenhos.

## 1.2 Organização do Texto

O restante desta dissertação está organizado da seguinte forma: no próximo capítulo é apresentado uma revisão bibliográfica sobre os aspectos teóricos da modelagem da eletrofisiologia cardíaca. No Capítulo 3, além dos métodos numéricos utilizados para a resolução dos modelos matemáticos empregados nas simulações da eletrofisiologia cardíaca, é descrito a proposta de um ambiente distribuído e multi-GPU para a simulação da eletrofisiologia cardíaca e, no Capítulo 4, são mostrados os resultados obtidos com a implementação da proposta e sua discussão. Finalmente, as conclusões e trabalhos futuros são apresentadas no Capítulo 5.

## 2 REVISÃO BIBLIOGRÁFICA

### 2.1 Eletrofisiologia cardíaca

Potenciais de ação são rápidas mudanças que ocorrem no potencial de membrana que se propagam por toda membrana celular. Um potencial de ação (PA) ocorre como resultado da concentração relativa de íons (predominantemente potássio, sódio e cálcio) e forças eletrostáticas através da membrana (GLOVER; BRUGADA, 2016). O PA cardíaco é ilustrado na Figura 1 para os átrios e ventrículos, e pode ser dividido em cinco fases principais, descritas a seguir, baseado em Hall (2017), Silverthorn (2010), Zipes *et al.* (2017), Andrade *et al.* (2016).

**Fase 0:** despolarização. Quando os potenciais de membrana se tornam mais positivos, eles provocam alteração conformacional nas comportas dos canais rápidos de sódio ( $I_{Na}$ ) ativados por voltagem abrindo-os rapidamente, tornando a membrana muito permeável a  $Na^+$ , levando o potencial de membrana a aproximadamente  $+20$  mV.

**Fase 1:** repolarização inicial. O mesmo aumento de voltagem também faz com que a comporta de inativação dos canais rápidos de  $Na^+$  se fechem por alteração conformacional. A célula inicia-se o processo de repolarização devido a saída de íons de potássio ( $K^+$ ) causada pela rápida abertura de canais de potássio (canais transientes de efluxo  $I_{t0}$ ) ativado pela despolarização.

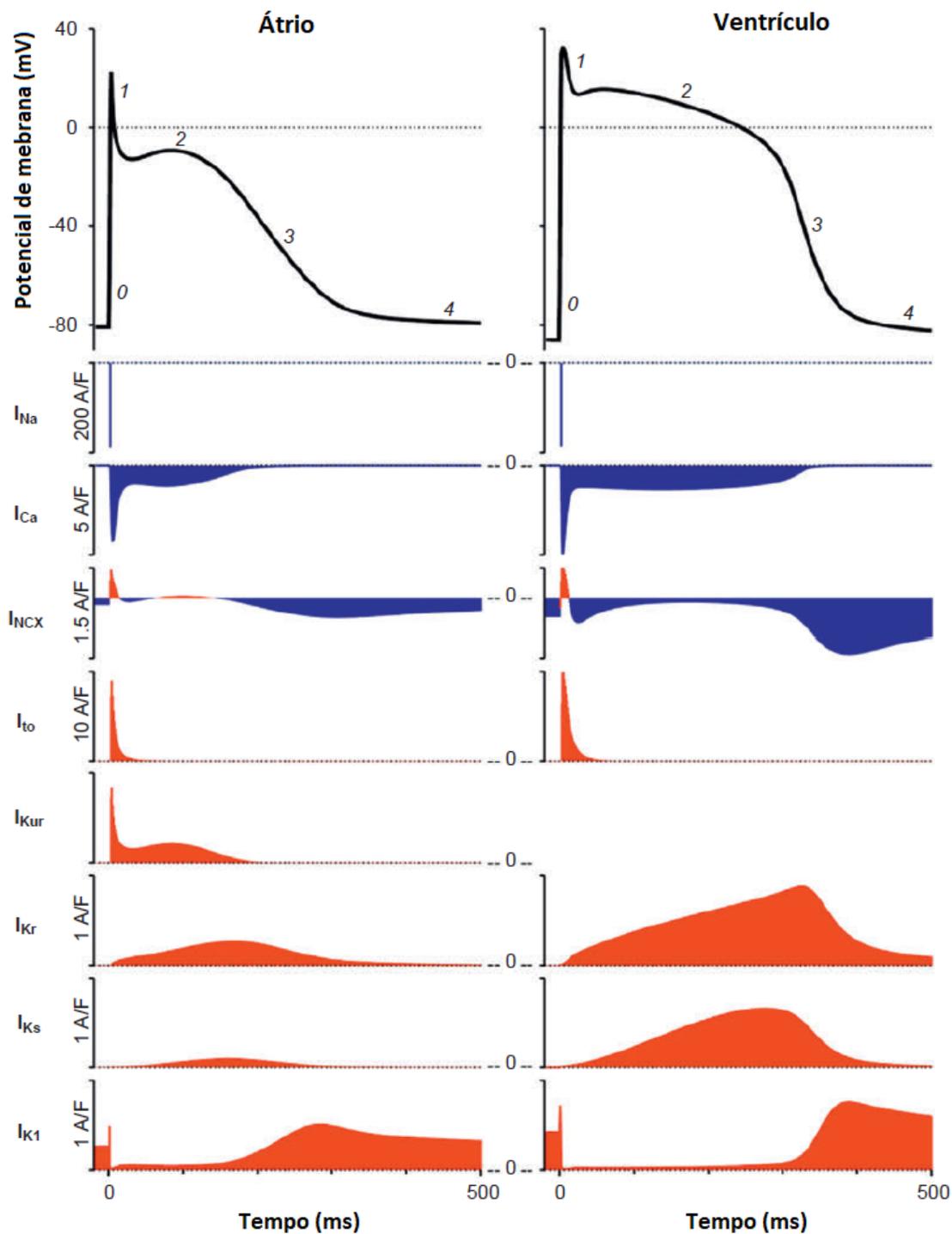
**Fase 2:** platô. A repolarização inicial é breve, de forma que um platô é formado. O platô presente no PA cardíaco dura de 0,2 a 0,3 s, fazendo com que a contração dos músculos do coração dure esse mesmo período. O platô é formado por dois motivos: aumento da permeabilidade do  $Ca^{2+}$  e diminuição da permeabilidade ao  $K^+$ . Durante as fases 0 e 1, canais de cálcio controlados por voltagem (canais de cálcio tipo L ( $I_{CaL}$ )), também chamados de canais lentos, são abertos lentamente, permitindo a entrada de  $Ca^{2+}$ . Além disso, canais rápidos de potássio são fechados, diminuindo a saída de  $K^+$ , impedindo o retorno do PA ao nível basal, formando-se assim o platô.

**Fase 3:** repolarização rápida. Os canais lentos de cálcio são lentos para abrir e também permanecem abertos por mais tempo. Quando finalmente se fecham e cessam o fluxo de  $Ca^{2+}$ , a permeabilidade ao  $K^+$  aumenta rapidamente, devido a completa abertura dos canais retificadores de potássio (canais de ativação ultra rápida ( $I_{Kur}$ ), rápida ( $I_{Kr}$ ) e lenta ( $I_{Ks}$ )). A abertura dos canais de potássio é lenta, se inicia no final da fase 0 e só é completada no fim do platô (início da fase 3). Com a abertura dos canais de potássio, os íons  $K^+$  rapidamente deixam o interior, e a célula retorna ao seu potencial de repouso, encerrando o PA (fase 4). Além disso, ao final da fase 3 ocorre a abertura dos canais da corrente retificadora de influxo ( $I_{K1}$ ), responsável por atuar nessa fase final de

repolarização e manter o potencial de repouso (fase 4).

**Fase 4:** potencial de membrana em repouso. As células miocárdicas têm um potencial de repouso de aproximadamente -90 mV.

Figura 1 – Forma representativa do PA ao longo do tempo, mostrando as cinco fases, para o átrio e ventrículo. Densidade de corrente interna em azul (despolarizante) e densidade de corrente externa em vermelho (repolarizante)



Fonte: Adaptado de Issa *et al.* (2018)

A Figura 1 mostra que o PA nos ventrículos possui platô mais duradouro, o que é explicado pela presença da corrente retificadora ultra-rápida ( $I_{K_{ur}}$ ) nos miócitos atriais, inexistente nos ventrículos, de forma a acelerar a repolarização nos átrios. A Figura 1 também mostra que os ventrículos possuem potencial de repouso mais eletronegativo (fase 4), pois a corrente retificadora de influxo ( $I_{K1}$ ) é maior nos mesmos, permitindo uma saída maior de  $K^+$ , o que deixa a membrana mais eletronegativa.

É importante citar que a corrente de troca sódio-cálcio ( $I_{Na/Ca}$ ) e a corrente bomba de sódio-potássio ( $I_{Na/K}$ ) desempenham papéis relevantes na manutenção das concentrações fisiológicas intracelulares de sódio, potássio e cálcio (ANDRADE *et al.*, 2016).

As células cardíacas são capazes de gerar PA espontaneamente, sem a necessidade de sinal do sistema nervoso, ou seja, são autoexcitáveis. Antes de analisar a autoexcitação e o mecanismo de ritmicidade cardíaca, é imprescindível conhecer o sistema especializado condutor e excitatório do coração que controla as contrações.

A condução da corrente elétrica se inicia com a despolarização do nodo sinusal, que é uma faixa pequena de músculo cardíaco especializado, sendo considerado o marca-passo normal do coração, pois é quem garante a geração do PA cardíaco e ritmicidade do mesmo. O nodo sinusal se conecta as vias intermodais, que por sua vez conduz a atividade elétrica até o nodo átrio-ventricular (A-V). O estímulo elétrico ao percorrer as vias intermodais, despolariza os átrios (SILVERTHORN, 2010).

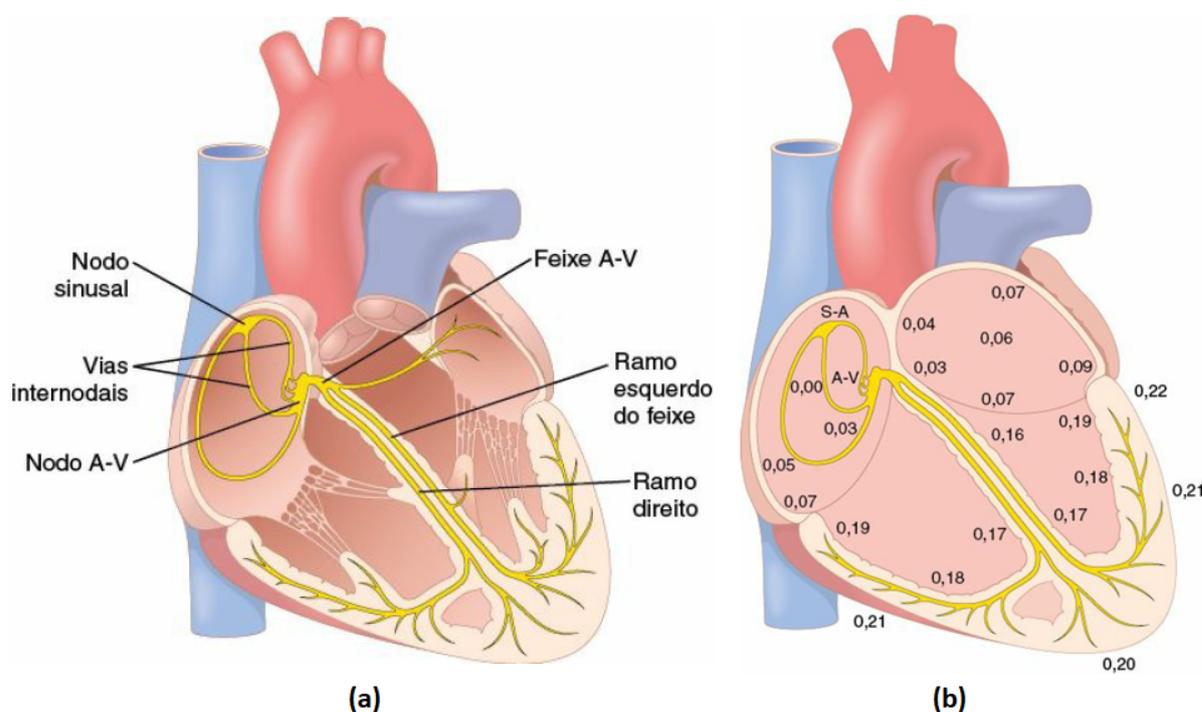
Na próxima etapa, o impulso é atrasado no nodo A-V devido ao reduzido número de junções comunicantes (*gap junctions*), entre as células das vias de condução, criando resistência a passagem de íons excitatórios. Esse retardo é fundamental, pois possibilita os átrios se contraírem e esvaziarem seu conteúdo nos ventrículos antes do início da contração ventricular (SHERWOOD; WARD, 2018). A seguir, os impulsos atingem o feixe A-V (feixe de His), levando o sinal até os ventrículos, por meio das fibras de Purkinje. O feixe A-V, possui característica que impede a reentrada do impulso dos ventrículos para os átrios. Os átrios e ventrículos são separados por barreiras fibrosas, funcionando como isolante, impedindo a passagem do impulso por qualquer outra via. A passagem do impulso por outras vias ou a reentrada do sinal, pode causar arritmias graves (WIDMAIER *et al.*, 2018).

As fibras de Purkinje, tem características funcionais opostas as fibras do nodo A-V, ou seja, são muito calibrosas e conduzem o PA rapidamente (com velocidades até 150 vezes maior que algumas fibras do nodo A-V). Isso permite que o impulso atinja todo restante do músculo ventricular quase instantaneamente (WIDMAIER *et al.*, 2018; SHERWOOD; WARD, 2018). As fibras de Purkinje, contém poucas miofibrilas, fazendo com que se contraíam pouco ou nada na transmissão do sinal. Quando a extremidade das fibras de Purkinje é atingida, o impulso é transmitido pela massa ventricular pelas próprias fibras musculares, de forma que a onda de despolarização segue o sentido base-ápice do coração

(WIDMAIER *et al.*, 2018; SHERWOOD; WARD, 2018).

A Figura 2 mostra o sistema condutor elétrico do coração, bem como o tempo de aparecimento do impulso elétrico em diferentes partes do órgão.

**Figura 2 – (a) Sistema condutor cardíaco; (b) Tempo de aparecimento do impulso elétrico nas diferentes partes do coração (em frações de segundo, após o aparecimento no nodo sinoatrial). A-V: atrioventricular; S-A: sinoatrial**



Fonte: Adaptado de Hall (2017)

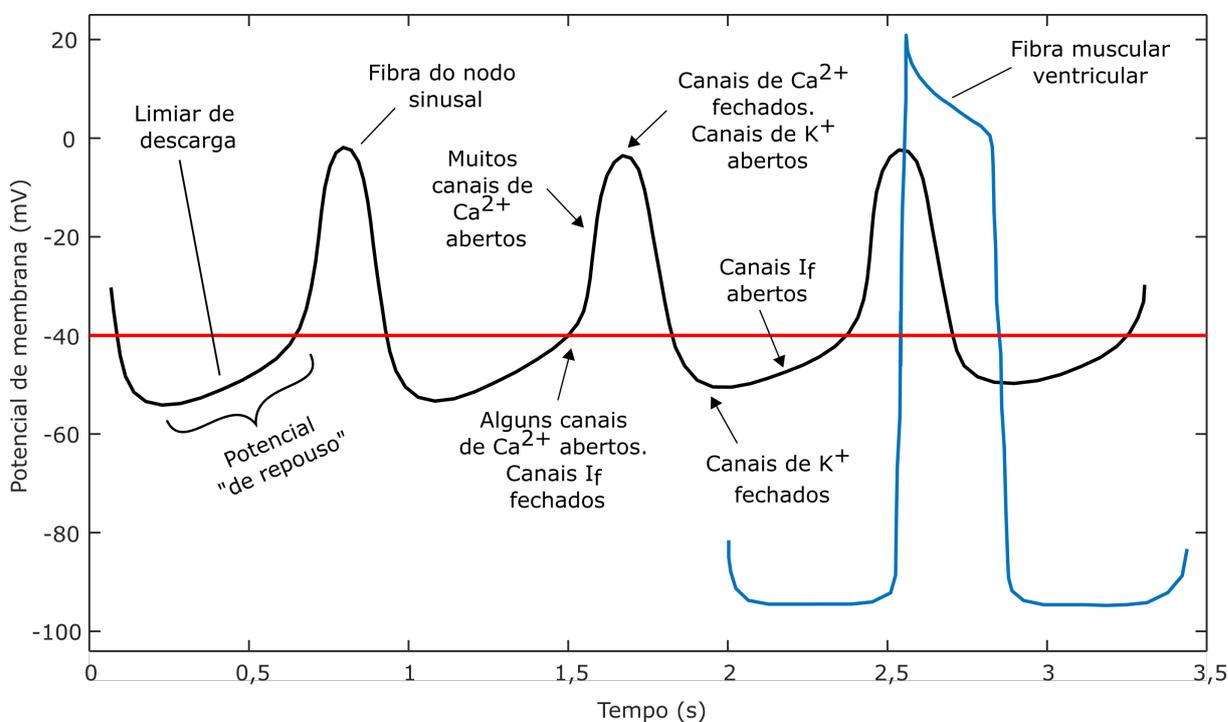
Como foi dito anteriormente, o responsável por gerar a frequência dos batimentos de todo o coração é o nodo sinusal (sinoatrial). Diferentemente de outras fibras miocárdicas, que possuem potencial de repouso entre -85 a -90 mV (fibras ventriculares, por exemplo), o nodo sinusal apresenta potencial de repouso de aproximadamente -55 a -60 mV (HALL, 2017). O que confere esse comportamento ao nodo sinoatrial é a presença de canais do tipo  $I_f$  (controlados por nucleotídeos cíclicos estimulados por hiperpolarização), que são canais permeáveis tanto ao  $K^+$  quanto ao  $Na^+$ , sendo que durante o potencial de repouso é mais permeável ao  $Na^+$  em relação ao  $K^+$ , de forma que o influxo líquido de carga positiva despolariza as células autoexcitáveis. Quando o potencial de membrana se torna mais positivo, os canais  $I_f$  são gradualmente fechados e alguns canais de  $Ca^{2+}$  se abrem, permitindo a continuação da despolarização. Mais canais de  $Ca^{2+}$  são abertos quando o limiar do potencial de membrana (aproximadamente -40 mV) é atingido, ocorrendo a fase de despolarização rápida do PA. É importante notar que esse processo é diferente das outras células excitáveis, pois ocorre devido aos canais de  $Ca^{2+}$  e não aos canais de  $Na^+$  controlados por voltagem. A fase de repolarização é parecida com a de outras células excitáveis, pois se dá devido ao fechamento dos canais de cálcio e abertura

lenta dos de potássio (SILVERTHORN, 2010; WIDMAIER *et al.*, 2018; SHERWOOD; WARD, 2018).

Perto do término de cada PA e por mais um breve período, a membrana se torna mais permeável ao  $K^+$ , de forma que o efluxo de potássio garante a eletronegatividade maior do que deveria ocorrer, deslocando o potencial de membrana para um valor mais próximo do potencial de Nernst do potássio. Esse estado é chamado de hiperpolarização e é importante pois garante que a membrana não se despolarize logo após ser polarizada. Entretanto, a condutância de  $K^+$  desaparece gradualmente, permitindo o aumento do potencial de membrana até o limiar de excitação (HALL, 2017; WIDMAIER *et al.*, 2018; SHERWOOD; WARD, 2018).

A Figura 3 mostra a descarga rítmica do nodo sinusal e a comparação do seu potencial de ação com o da fibra muscular ventricular. Observa-se o rápido PA em ponta nos ventrículos, que ocorre devido a abertura dos canais rápidos de sódio, além disso, a repolarização é mais rápida. A Figura 3 mostra também que o "potencial de repouso", também chamado de potencial de marcapasso no nodo sinusal é menos eletronegativo e ascendente, garantindo a ritmicidade dessas fibras.

**Figura 3 – Descarga rítmica do nodo sinusal. Comparação do potencial de ação do nodo sinusal com o da fibra muscular ventricular**



Fonte: Adaptado de Hall (2017)

A velocidade na qual o nodo sinusal se despolariza determina a frequência cardíaca. Se a permeabilidade das células excitáveis muda, a frequência cardíaca é alterada. Um aumento na permeabilidade ao  $K^+$  ou diminuição ao  $Ca^{2+}$ , diminui a frequência cardíaca,

pois retarda a despolarização, por outro lado o aumento da permeabilidade ao  $\text{Na}^+$  e  $\text{Ca}^{2+}$  acelera a despolarização e aumenta a frequência cardíaca (SILVERTHORN, 2010).

Outros locais do coração também podem apresentar autoexcitação, como o nodo A-V ou as fibras de Purkinje, porém a frequência normal do nodo sinusal é maior, de forma que o nodo sinoatrial atinge seu limiar primeiro. Se outras partes do coração desenvolver frequência rítmica maior que a do nodo sinusal, o mesmo deixa de ser o marca-passo natural do coração, ao passo que esse novo local passa a ser o marca-passo, chamado marca-passo ectópico (HALL, 2017).

## 2.2 Exemplos de patologias cardíacas

### 2.2.1 Arritmias cardíacas

A alteração na frequência cardíaca é chamada de arritmia. As arritmias podem se dividir em dois grupos, a braquicardias ( $< 60$  bpm), quando a frequência é abaixo da normal e a taquicardia quando a frequência é acima do normal ( $> 100$  bpm no adulto). A mais grave das arritmias é a fibrilação ventricular, que consiste na contração descoordenada do músculo decorrente de impulsos frenéticos que excitam primeiro uma parte do ventrículo e depois outra (FOX, 2019).

A base para fibrilação é o fenômeno denominado reentrada. Em uma sequência normal de impulsos, as ondas fisiológicas de excitação desaparecem espontaneamente após a ativação de todo o coração, pois o músculo já se encontra todo em estado refratário, sendo impossibilitado de conduzir o sinal, de forma que o impulso cessa. A reentrada ocorre quando um impulso de propagação persiste em re-excitá-lo após o período refratário. Em situações patológicas, as ondas de excitação podem ser bloqueadas em áreas circunscritas, girar em torno dessas zonas de bloqueio e entrar novamente no local da excitação original em ciclos repetitivos. A frente de onda não se apaga, mas propaga-se continuamente e, portanto, continua a excitar o coração, porque sempre encontra tecidos excitáveis (ISSA *et al.*, 2018).

Para que a reentrada ocorra, é necessário cumprir as seguintes condições: (i) Haver dois ou mais caminhos para a condução; (ii) Bloqueio unidirecional em uma via; (iii) Condução alternativa ao longo da outra via com atraso suficiente para invadir a via anteriormente bloqueada. Assim, uma das características mais importantes da fibrilação é a divisão de impulsos, de maneira a formar um ciclo vicioso, pois esses impulsos fazem surgir mais áreas de músculo refratário, e essas áreas produzem cada vez mais a divisão dos impulsos (GLOVER; BRUGADA, 2016).

Da mesma forma que ocorre a fibrilação ventricular, pode ocorrer a fibrilação atrial. Porém em comparação com a letalidade da fibrilação ventricular, a fibrilação atrial

é menos prejudicial. Da mesma forma que os ventrículos não bombearão sangue durante a fibrilação ventricular, os átrios também não o fazem na fibrilação atrial, porém ainda assim o sangue flui passivamente dos átrios e vai para os ventrículos (HALL, 2017).

### 2.2.2 Infarto do miocárdio

O infarto do miocárdio consiste na necrose (morte) de células do músculo cardíaco devido ao suprimento inadequado de sangue na área afetada. O infarto do miocárdio é causado principalmente devido ao acúmulo de placas ateroscleróticas, levando à formação de coágulo que bloqueia o fluxo sanguíneo. As placas ateroscleróticas, são resultado da deposição de colesterol no endotélio das artérias coronárias, de forma que essas áreas são invadidas por tecidos fibrosos, sendo calcificadas. O infarto também pode ocorrer devido a ruptura do vaso, compressão mecânica ou vasoconstrição (contração do vaso sanguíneo) (FOX, 2019).

A principal causa de morte devido a oclusão coronariana ocorrem devido a fibrilação que se desenvolve após o infarto (WIDMAIER *et al.*, 2018). A insuficiência cardíaca isquêmica consequente do infarto do miocárdio é caracterizada por uma remodelação estrutural e funcional que leva a uma alteração do do perfil e duração do PA. Porém, a compreensão das alterações que ocorrem nas correntes iônicas permanece incompletas (TOMASELLI; MARBÁN, 1999; HEGYI *et al.*, 2018).

A remodelação cardíaca é o conjunto de alterações moleculares, celulares e intersticiais, que se manifestam clinicamente por mudanças no tamanho, forma e função do coração, em resposta à determinada lesão cardíaca. A remodelação cardíaca é o processo fisiológico e patológico que tipicamente ocorre após o infarto do miocárdio (COHN *et al.*, 2000).

Na fase aguda do infarto do miocárdio desencadeia-se os processos de isquemia e necrose na região onde ocorreu a lesão. Já durante a fase de cicatrização dessa região, ocorre o processo de formação de fibrose, que substitui as células necróticas. A fibrose está relacionada ao aumento de colágeno, sintetizados pela hiperativação de fibroblasto, como reposta reparativa ou reativa à lesão (CHIONG *et al.*, 2011; SHI *et al.*, 2017). Com o aumento do colágeno, essas regiões se tornam não condutoras e não excitantes, podendo ocorrer bloqueios na condução elétrica, favorecendo o aparecimento de arritmias por reentrada (AZEVEDO *et al.*, 2016). Assim, o tecido cardíaco lesionado é altamente heterogêneo, composto de fibrose e necrose, além das regiões próximas ainda compostas de miócitos saudáveis.

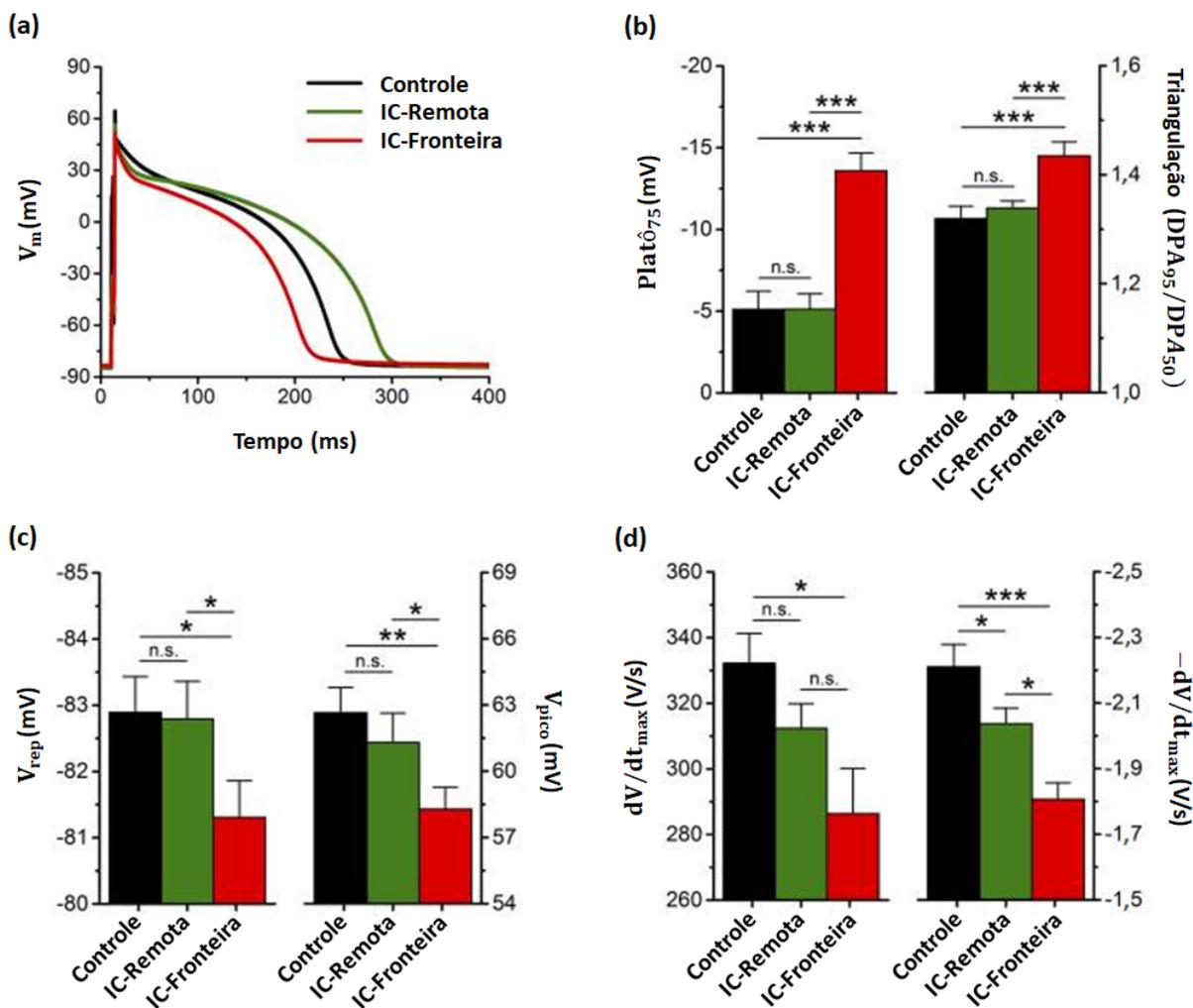
Outro processo desencadeado na remodelação cardíaca pós infarto é a hipóxia, que é a oxigenação deficiente do tecido, que pode alterar significativamente a eletrofisiologia cardíaca, estando relacionada também com a arritmia pós infarto (SACHETTO *et al.*,

2018).

Além do surgimento de arritmias, todas essas alterações anatômicas que ocorrem no tecido cardíaco podem re-excitar o tecido cardíaco vizinho, agindo como fontes de batimentos ectópicos (SACHETTO *et al.*, 2018).

Hegyí *et al.* (2018) realizaram um experimento para estudar a remodelação da insuficiência cardíaca (IC) após o infarto em um miocárdio suíno adulto. A análise do estudo foi feita após cinco meses da indução do infarto. Foram analisados dois locais diferentes da região infartada, o primeiro em uma zona próxima a fronteira da lesão e a segunda em uma zona remota. A Figura 4 mostra resultados encontrados para o PA para as zonas estudadas bem como para o grupo de controle que possuía miocárdio saudável.

Figura 4 – Alteração do PA pós infarto. (a) PA do grupo de controle e das áreas de fronteira e remotas da região infartada; (b) Platô<sub>75</sub> e Triangulação do PA; (c) Potencial de repouso e pico do PA; (d) Taxa de despolarização e repolarização.  $V_m$ : potencial de membrana.  $V_{rep}$ : potencial de repouso.  $V_{pico}$ : pico do PA.  $dV/dt_{max}$ : taxa de variação do potencial de membrana. DPA: duração do PA. número de amostras = 19-26 células/5-7 animais. ANOVA com pós teste Bonferroni. n.s.: não significativo. \* $P < 0,05$ , \*\* $P < 0,01$ , \*\*\* $P < 0,001$



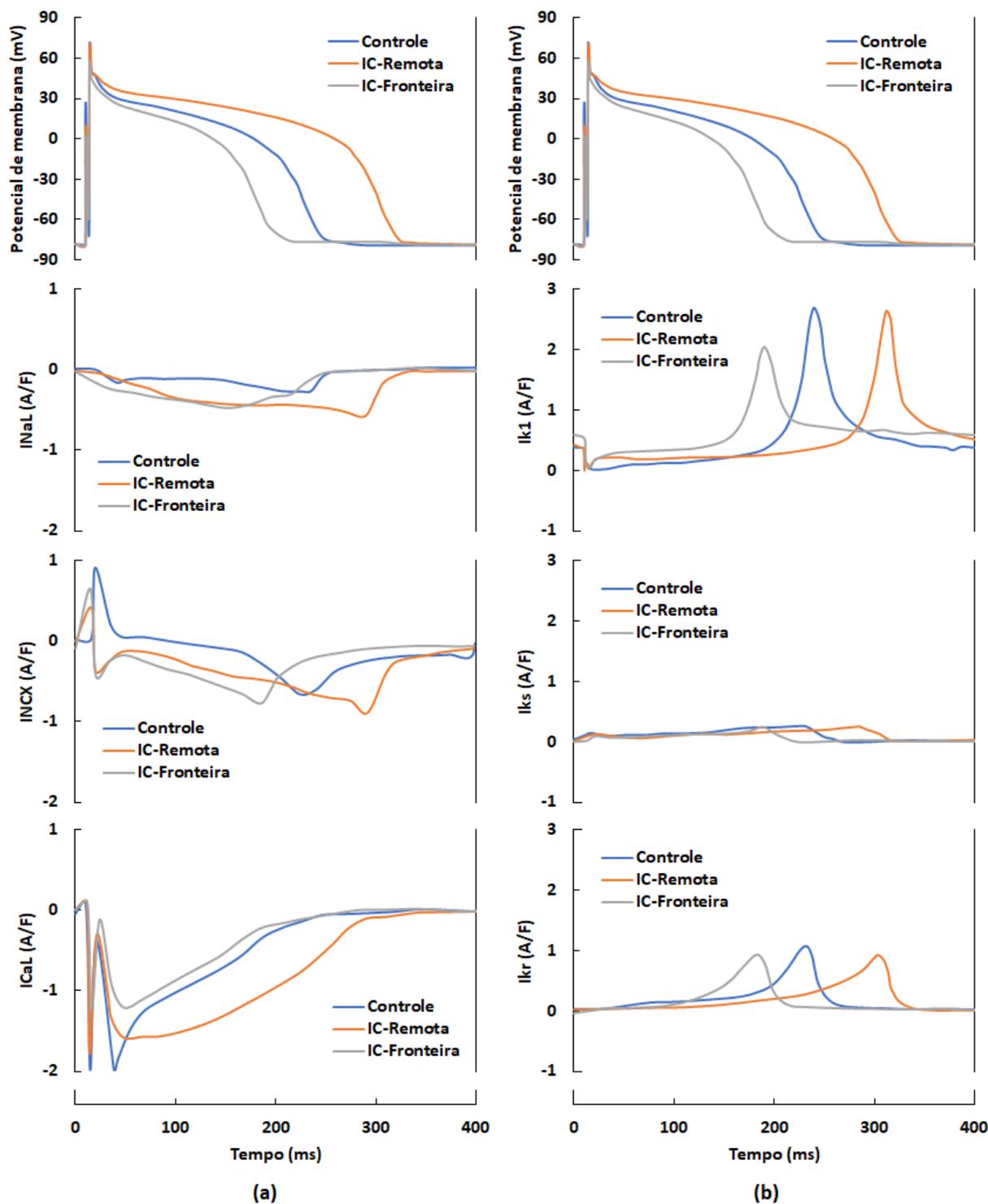
Fonte: Adaptado de Hegyí *et al.* (2018)

A Figura 4(a) mostra o comportamento dos perfis do PA. Na fronteira, os valores do potencial de membrana foram menores que do grupo de controle, ao passo que em zonas remotas no estágio de repolarização, o potencial foi maior do que no grupo de controle. Além disso a duração do potencial de ação (DPA) foi encurtada na fronteira do infarto e prolongada em zonas remotas. Os potenciais de platô até 75% do PA ( $\text{Platô}_{75}$ ) diminuíram na fronteira da IC, mas permaneceram inalterados na zona remota do IC (Figura 4(b)). Isso fez com que o fator de triangulação do PA, mensurado como a relação entre a DPA com 95% de repolarização ( $\text{DPA}_{95}$ ) e a DPA com 50% de repolarização ( $\text{DPA}_{50}$ ), fosse elevado na fronteira. Tanto o potencial de repouso ( $V_{\text{rep}}$ ), quanto o pico do potencial ( $V_{\text{pico}}$ ) foram menores na fronteira do infarto, ao passo que em zonas remotas, permaneceu estatisticamente inalterado (Figura 4(c)). Por fim, a Figura 4(d) exhibe a taxa de variação do potencial de membrana ( $dV/dt$ ) na despolarização e na repolarização. A velocidade máxima de avanço do PA das áreas remotas foi insignificante em relação ao grupo de controle, ao passo que da fronteira foi insignificante em relação a áreas remotas, porém significativa em relação aos indivíduos de controle, exibindo menores valores em relação à esse grupo. Por outro lado, a taxa de despolarização mostrou-se estatisticamente significativa para todos casos analisados, de forma que tanto da fronteira quanto em áreas remotas, a variação do potencial de membrana foi menor do que do grupo de controle.

Apesar do perfil do PA na fronteira epicardial da zona infartada ter aparência normalizada após dois meses da obstrução, a condução de frente de ondas parecem ter permanecido anormais. Para melhor compreender as variações nas correntes iônicas que entram e saem da célula ao longo do PA, para os casos analisados por Hegyi *et al.* (2018), os mesmos mediram os valores das principais correntes que deram origem aos PA visualizados na Figura 4. Os resultados encontrados por esses autores são mostrados na Figura 5.

Analisando as correntes de entrada, segundo Hegyi *et al.* (2018), a corrente nos canais de sódio do tipo L ( $I_{\text{NaL}}$ ) aumentou durante a fase de despolarização. Nas células IC-Remotas e nas IC-Fronteira, houve aumento também durante o platô, porém, no caso da IC em áreas remotas, houve aumento no número de  $\text{Na}^+$  totais que entram na célula durante o PA, diferentemente do caso da IC em áreas da fronteira, onde não houve diferença significativa em relação ao grupo de controle. A corrente de íons,  $I_{\text{NCX}}$  nas células de controle no início do PA foi direcionada para fora da célula e posteriormente mudou de sentido, sendo direcionada para o meio interno, atingindo o pico durante a repolarização. Já nas células IC-Fronteira e IC-Remota, essa tendência de corrente no sentido exterior, embora mostrada no gráfico, não existiu, pois na verdade, a corrente foi instantaneamente direcionada para dentro da célula.

Figura 5 – PA ventricular e as principais correntes iônicas para os grupos de controle, IC-Remota e IC-Fronteira. (a) Correntes iônicas de influxo; (b) Correntes iônicas de efluxo. Número de amostras = 6–16 células/4–6 animais.  $I_{NaL}$ : corrente de  $Na^+$  por canais do tipo L.  $I_{NCX}$ : corrente de troca de  $Na^+/Ca^{2+}$ .  $I_{CaL}$ : corrente de  $Ca^{2+}$  por canais do tipo L.  $I_{K1}$ : corrente retificadora de influxo de  $K^+$ .  $I_{Ks}$ : corrente de  $K^+$  por canais lentos.  $I_{Kr}$ : corrente de  $K^+$  por canais rápidos



Fonte: Adaptado de Hegyi *et al.* (2018)

Analisando as correntes de saída, ainda segundo Hegyi *et al.* (2018), a corrente

de potássio pelos canais rápidos ( $I_{Kr}$ ) foi pequena durante o platô e aumentou durante a repolarização (fase 3). A corrente dos canais lentos do retificadores  $K^+$  ( $I_{Ks}$ ) aumentou gradualmente durante o platô e caiu durante a repolarização, mas sua densidade foi menor que a do  $I_{Kr}$ . A corrente  $I_{K1}$  aumentou rapidamente no fim da repolarização, sendo que sua densidade e carga total não foi alterada no caso IC-Remota, mas foi diminuída no caso IC-Fronteira.

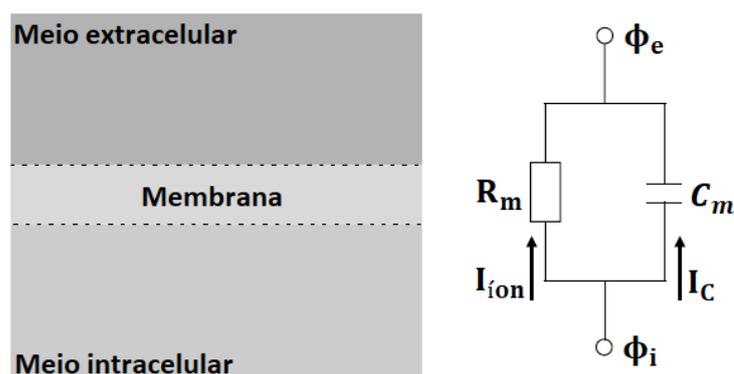
É interessante notar que as correntes mensuradas por Hegyi *et al.* (2018) são coincidentes com a tendência das curvas teorizadas, mostradas na Figura 1. Apenas as curvas de corrente de saída dos canais lentos  $I_{Ks}$  apresentam densidade menor que a curva da Figura 1. E a curva de corrente de  $I_{Na}$  é diferente porque aqui foi mensurado canais lentos, enquanto a Figura 1 mostra os canais rápidos (que são os principais responsáveis pela despolarização).

## 2.3 Modelagem da eletrofisiologia cardíaca

### 2.3.1 Modelo matemático para membrana celular

O comportamento elétrico da membrana celular, pode ser aproximado por uma analogia a um circuito elétrico resistor-capacitor em paralelo. Nessa analogia, um capacitor  $C_m$ , representa a separação de cargas entre os meios intracelular e extracelular pela membrana isolante e um resistor não linear  $R_m$  modela a passagem de corrente  $I_{ion}$  pelos canais iônicos (SACHSE, 2004). A Figura 6 ilustra essa estratégia.

**Figura 6 – Analogia do comportamento elétrico da membrana celular com um circuito elétrico resistor-capacitor em paralelo**



Fonte: Adaptado de Sachse (2004)

O potencial sobre a membrana  $V_m$  é definido pela diferença entre o potencial extracelular  $\phi_e$  e intracelular  $\phi_i$ . Para modelar o capacitor a partir dessa estratégia, o potencial de membrana  $V_m$ , proporcional a carga  $Q$ , é dado por:

$$V_m = \frac{Q}{C_m} \quad (2.1)$$

onde  $C_m$  é a capacitância da membrana.

O fluxo iônico muda o potencial da membrana, de forma que a corrente capacitiva  $I_C$  (variação da carga em relação ao tempo) pode ser calculada pela Equação 2.2, assumindo que a capacitância da membrana é constante ao longo do tempo.

$$\frac{dV_m}{dt} = \frac{d}{dt} \left( \frac{Q}{C_m} \right) = \frac{I_C}{C_m} \Rightarrow I_C = C_m \frac{dV_m}{dt} \quad (2.2)$$

A corrente transmembrânica ( $I_m$ ) é a soma da contribuição da corrente iônica com a corrente capacitiva:

$$I_m = I_{\text{ion}} + I_C \Rightarrow I_C = I_m - I_{\text{ion}} \quad (2.3)$$

Em uma célula isolada, como não há acúmulo de carga em nenhum dos lados da membrana, a corrente total através da membrana deve ser zero, logo, combinando as Equações 2.2 e 2.3:

$$C_m \frac{dV_m}{dt} + I_{\text{ion}} = 0 \Rightarrow \frac{dV_m}{dt} = -\frac{I_{\text{ion}}}{C_m} \quad (2.4)$$

O desafio a partir de então, é modelar a corrente iônica uma vez que diferentes células possuem diferentes tipos de canais iônicos, que podem possuir diferentes relações de corrente-voltagem. Além disso,  $I_{\text{ion}}$  pode ser dependente do tempo e do próprio potencial de membrana,  $V_m$ .

### 2.3.2 Modelos para corrente iônica

Antes de descrever os modelos de corrente iônica mais utilizados, é necessário definir o potencial de Nernst. A diferença entre as concentrações intracelular e extracelular, faz com que os íons se movam no sentido oposto ao do gradiente, porém a força do campo elétrico gerada pela diferença de potencial entre os meios, faz com que os íons se movam no sentido oposto ao da difusão. O equilíbrio será atingido quando os fluxos se igualarem, de forma que o potencial para que haja esse equilíbrio (fluxo nulo) para um íon qualquer, é chamado potencial de equilíbrio de Nernst,  $E_i$ , e é dado por (SACHSE, 2004; HALL, 2017):

$$E_i = \frac{RT}{z_i F} \ln \left( \frac{[c_i]_{\text{ext}}}{[c_i]_{\text{int}}} \right) \quad (2.5)$$

onde  $R$  é a constante universal dos gases (8,31 J/Kmol),  $T$  é a temperatura absoluta,  $z_i$  é a valência do íon  $i$ ,  $F$  é a constante de Faraday (9,648x10<sup>4</sup> C/mol),  $c_{i_{\text{ext}}}$  e  $c_{i_{\text{int}}}$  denotam as concentrações externa e interna do íon  $i$ , respectivamente.

O potencial de Nernst só se aplica no caso da membrana ser permeável a apenas um tipo de íon, de forma que se ela é permeável a mais tipos, é necessário adotar a equação de Goldman-Hodgkin-Katz (GHK), que é uma extensão da equação de Nernst. A equação de GHK para os íons de potássio, sódio e cloro, por exemplo, toma a seguinte forma (KEENER; SNEYD, 2009):

$$E_r = -\frac{RT}{F} \ln \left( \frac{P_K [c_{K^+}]_{int} + P_{Na} [c_{Na^+}]_{int} + P_{Cl} [c_{Cl^-}]_{ext}}{P_K [c_{K^+}]_{ext} + P_{Na} [c_{Na^+}]_{ext} + P_{Cl} [c_{Cl^-}]_{int}} \right) \quad (2.6)$$

sendo  $E_r$  o potencial resultante e  $P_i$  a permeabilidade da membrana ao íon  $i$ , a qual é expressa por (SACHSE, 2004):

$$P_i = \frac{D_i \beta_i}{h} \quad (2.7)$$

sendo  $D_i$  o coeficiente de difusão do íon  $i$  na membrana,  $\beta_i$  o coeficiente de partição água-membrana e  $h$  a espessura da membrana.  $D_i$  e  $\beta_i$  são dependentes do tipo de membrana e do íon  $i$ .

Quando o potencial transmembrânico é diferente do potencial de Nernst, há uma corrente através do canal. O modelo mais simples para descrever essa corrente iônica é o modelo linear (SACHSE, 2004; KEENER; SNEYD, 2009), dado por:

$$I_i = g_i (V_m - E_i), \quad (2.8)$$

onde  $g_i$  é a condutância do íon  $i$ .

Outro modelo de corrente iônica, conhecido como Goldman-Hodgkin-Katz (GHK), pode ser obtido por simplificação da equação de Nernst-Planck para eletrodifusão, considerando o campo elétrico constante na membrana (KEENER; SNEYD, 2009). Esse modelo é não-linear, e para um determinado íon  $i$  é dado por:

$$I_i = P_i \frac{z_i^2 F^2}{RT} V_m \frac{[c_i]_{int} - [c_i]_{ext} \left( e^{-\frac{z_i F V_m}{RT}} \right)}{1 - e^{-\frac{z_i F V_m}{RT}}} \quad (2.9)$$

É importante ressaltar que esses modelos descrevem a relação corrente-voltagem instantânea. Porém, a corrente total depende da corrente instantânea e da quantidade de canais abertos, que por sua vez varia com o tempo e depende do estado elétrico da membrana. Logo, pode ser necessário considerar modelos adicionais.

### 2.3.3 Modelos para o tecido

Como dito anteriormente, as células cardíacas conectam-se por junções do tipo *gap*, o que lhes permite comunicação instantânea entre células vizinhas, permitindo a transmissão do sinal elétrico em forma de íons.

Os modelos bidomínio e monodomínio são modelos que descrevem essa conexão das células pela qual a onda elétrica se propaga. São modelos contínuos de forma que suas equações descrevem o comportamento elétrico de forma macroscópica, onde as células do tecido são tratadas de forma homogênea, o que é adequado para maioria das situações (SACHSE, 2004; KEENER; SNEYD, 2009).

#### 2.3.3.1 Modelo bidomínio

Considere que cada ponto espacial do tecido é composto por uma fração dos espaços intracelular e extracelular. Considere também que os pontos possuem os potenciais elétricos  $V_{\text{ext}}$  (externo) e  $V_{\text{int}}$  (interno), e as correntes  $I_{\text{ext}}$  (externa) e  $I_{\text{int}}$  (interna). A relação entre o potencial e a corrente é dado pela lei de Ohm:

$$\begin{aligned} I_{\text{int}} &= -\sigma_{\text{int}} \nabla V_{\text{int}} \\ I_{\text{ext}} &= -\sigma_{\text{ext}} \nabla V_{\text{ext}} \end{aligned} \quad (2.10)$$

onde  $\sigma_{\text{int}}$  e  $\sigma_{\text{ext}}$  são tensores de condutividade.

Se não houver forças externas, a corrente total ( $I_t$ ) é conservada, logo:

$$\nabla \cdot I_t = \nabla \cdot (\sigma_{\text{int}} \nabla V_{\text{int}} + \sigma_{\text{ext}} \nabla V_{\text{ext}}) = 0 \quad (2.11)$$

Considerando que a corrente transmembrânica é igual a variação das correntes que entra ou sai, e considerando o sentido da corrente transmembrânica de dentro para fora, tem-se:

$$I_m = \nabla \cdot (\sigma_{\text{int}} \nabla V_{\text{int}}) = -\nabla \cdot (\sigma_{\text{ext}} \nabla V_{\text{ext}}) \quad (2.12)$$

Como mostrado anteriormente na Equação 2.3, a corrente transmembrânica é a soma da corrente iônica com a capacitiva, sendo a corrente capacitiva expressa na Equação 2.2, logo:

$$I_m = \beta \left( C_m \frac{\partial V_m}{\partial t} + I_{\text{ion}} \right) = \nabla \cdot (\sigma_{\text{int}} \nabla V_{\text{int}}) \quad (2.13)$$

onde  $\beta$  é uma constante utilizada para converter a corrente transmembrânica de unidade de área para unidade de volume. Essa constante é necessária pois tanto a corrente iônica,  $I_{\text{ion}}$ , como a capacitância,  $C_m$ , são convenientemente medidas por unidade de área da membrana.

Sabe-se também que o potencial da membrana é a diferença dos potenciais intracelular e extracelular:

$$V_m = V_{\text{int}} - V_{\text{ext}} \Rightarrow V_{\text{int}} = V_m + V_{\text{ext}} \quad (2.14)$$

Assim, utilizando a Equação 2.14, rearranjando as Equações 2.11 e 2.13 e, por fim, combinando essas três equações, obtém-se:

$$\begin{aligned} \nabla \cdot (\boldsymbol{\sigma}_{\text{int}} \nabla V_m) + \nabla \cdot (\boldsymbol{\sigma}_{\text{int}} \nabla V_{\text{ext}}) &= \beta \left( C_m \frac{\partial V_m}{\partial t} + I_{\text{ion}} \right) \\ \nabla \cdot (\boldsymbol{\sigma}_{\text{int}} \nabla V_m) + \nabla \cdot ((\boldsymbol{\sigma}_{\text{int}} + \boldsymbol{\sigma}_{\text{ext}}) \nabla V_{\text{ext}}) &= 0 \quad \text{em } \Omega \end{aligned} \quad (2.15)$$

As Equações em 2.15 são a formulação padrão do modelo bidomínio, proposta por Tung (1978), Miller e Geselowitz (1978).

### 2.3.3.2 Modelo monodomínio

O modelo monodomínio é uma redução do bidomínio, que vem do pressuposto de que os domínios intracelular e extracelular têm proporções iguais de anisotropia. Assim, considere que as condutividades intra e extracelular são proporcionais:

$$\boldsymbol{\sigma}_{\text{ext}} = \lambda \boldsymbol{\sigma}_{\text{int}} \quad (2.16)$$

onde  $\lambda$  é uma constante de proporcionalidade.

Assim, combinando as Equações 2.15 e 2.16 e simplificando, tem-se:

$$\left( \frac{\lambda}{(1+\lambda)} \nabla \cdot (\boldsymbol{\sigma}_{\text{int}} \nabla V_m) \right) = \beta \left( C_m \frac{\partial V_m}{\partial t} + I_{\text{ion}} \right), \quad (2.17)$$

fazendo  $\boldsymbol{\sigma} = \frac{\boldsymbol{\sigma}_{\text{int}} \lambda}{(1+\lambda)}$  obtém-se:

$$\nabla \boldsymbol{\sigma} \nabla V_m = \beta \left( C_m \frac{\partial V_m}{\partial t} + I_{\text{ion}} \right) \quad (2.18)$$

O modelo monodomínio é matematicamente e computacionalmente mais simples.

Potse *et al.* (2006) investigaram a diferença entre assumir o modelo monodomínio em comparação ao bidomínio em um coração humano e concluíram que as diferenças entre ambos foram insignificantes. Bourgault e Pierre (2010) também investigaram numericamente os dois métodos sobre dois aspectos diferentes: (i) discrepância entre os modelos no nível contínuo (e não apenas entre as equações discretizadas); (ii) estimativa do erro de discretização, fornecendo a resolução da malha necessária para executar uma simulação precisa. Os autores também concluíram que as diferenças entre os métodos foi muito pequena, menor que 1%, de forma que o erro entre os métodos foi menor que o erro de discretização das próprias malhas comumente utilizadas.

### 2.3.4 Modelos para canais iônicos

Existem diversos tipos de canais iônicos na membrana celular, de forma que seu comportamento pode ser modelado sob o ponto de vista em que sua condutividade é considerada constante ou variável. A condutividade dos canais iônicos muda ao longo do tempo devido a variação do potencial na membrana, pois essas variações influenciam nas proteínas que compõem o canal iônico, permitindo o canal alterar sua forma, como foi descrito anteriormente.

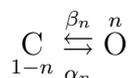
Os modelos mais utilizados que preveem esse comportamento são o de dois estados, o de subunidade e os baseados em cadeias de Markov, descritos nas subseções seguintes.

#### 2.3.4.1 Modelo de dois estados

É o modelo mais simples de canal iônico, o qual assume que o canal pode assumir dois estados: aberto (O) ou fechado (C). Seja  $n$  a proporção de canais abertos ( $n$  pertence  $[0,1]$ ), pode-se escrever a seguinte equação para a variação de  $n$  com o tempo (KEENER; SNEYD, 2009):

$$\frac{dn}{dt} = \alpha_n(1 - n) - \beta_n n, \quad (2.19)$$

onde os fatores  $\alpha_n$  e  $\beta_n$  são chamados de constantes de taxa de transição, e são dependentes do potencial transmembrânico, do tipo de canal iônico, da concentração iônica, entre outros fatores, mas não do tempo.  $\alpha_n$  é o número de vezes por segundo que um canal que está no estado fechado é aberto, enquanto  $\beta_n$  é o número de vezes por segundo que um canal que está no estado aberto é fechado. Em estado de equilíbrio  $\frac{dn}{dt}=0$ . Pode-se resumir o comportamento do modelo de dois estados no esquema abaixo.



Fazendo agora:

$$n_{\infty} = \frac{\alpha_n}{\alpha_n + \beta_n} \quad (2.20)$$

e:

$$\tau_n = \frac{1}{\alpha_n + \beta_n} \quad (2.21)$$

de forma que  $n_{\infty}$  é o valor do equilíbrio assintótico de  $n$  e  $\tau_n$  uma constante de tempo, a Equação 2.19 então pode ser convenientemente reescrita na forma:

$$\frac{dn}{dt} = \frac{n_{\infty} - n}{\tau_n} \quad (2.22)$$

Valores para  $n_{\infty}$  e  $\tau_n$  podem ser obtidos em testes experimentais (KEENER; SNEYD, 2009).

A condutância macroscópica de uma população de canais similares é dada por (OLIVEIRA, 2013):

$$g_i = N_i \cdot n \cdot g_{i,max} \quad (2.23)$$

com  $N_i$  representando o número de canais e  $g_{i,max}$  a condutância máxima do canal.

#### 2.3.4.2 Modelo de subunidades

O modelo de dois estados admite apenas uma unidade do canal iônico, ou ainda que o mesmo possua subunidades, porém ele só é considerado em um estado se todas subunidades também estão nesse estado, por exemplo, um canal estará aberto quando todas subunidades estão no estado aberto. Porém para modelar canais iônicos mais complexos e reais, é necessário admitir que ele tenha subunidades independentes, que podem estar abertas ou fechadas (HILLE, 2001; HODGKIN; HUXLEY, 1952).

Como visto anteriormente, os canais de sódio exibem o processo de ativação quando há um aumento rápido de condutividade (permeabilidade) e inativação quando a condutividade retorna a zero de forma lenta. Assim, para modelar esse comportamento dos canais, é necessário considerar a ativação e inativação. Considerando que o canal possua, por exemplo, três subunidades idênticas  $m$  associadas a ativação e uma subunidade  $h$  relacionada a inativação - considerando que as subunidades  $m$  e  $h$  são independentes - pode-se escrever (HODGKIN; HUXLEY, 1952):

$$g_i = m^3 h g_{i,max} \quad (2.24)$$

$$\frac{dm}{dt} = \alpha_m(1 - m) - \beta_m m \quad (2.25)$$

$$\frac{dh}{dt} = \alpha_h(1 - h) - \beta_h h \quad (2.26)$$

onde  $\alpha_m$ ,  $\beta_m$ ,  $\alpha_h$  e  $\beta_h$  são as taxas de transição. Nas condições iniciais de equilíbrio, a variável de ativação é 0 e de inativação é 1. No momento que ocorre o processo de ativação, que desencadeia a despolarização e, conseqüentemente o PA, o potencial da membrana ( $V_m$ ) se afasta do de Nernst e cresce rapidamente, fazendo com que tanto  $m$  quanto  $m_\infty$  se aproximem de 1. Por outro lado, ao mesmo tempo, é desencadeado o processo de inativação, onde  $h$  sai de 1 para 0, porém esse processo é mais lento, de forma que no pico  $I_i$  tende a  $g_{i,\max}(V_m - E_i)$ , para o caso do modelo linear de corrente iônica. Logo, tem-se um modelo com variação da condutividade.

### 2.3.4.3 Modelo de cadeias de Markov

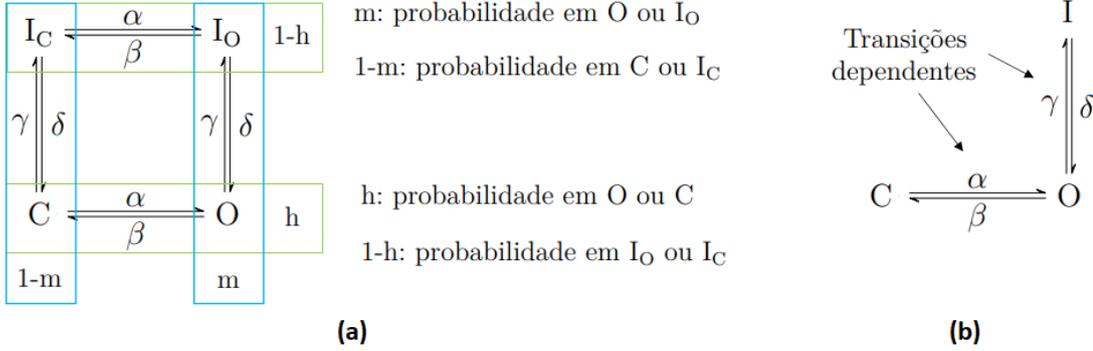
Os modelos anteriores são baseados no modelo desenvolvido por Hodgkin–Huxley (HH). Na formulação HH, os parâmetros de ativação/inativação ( $n$ ,  $m$ ,  $h$ ) não representam estados cinéticos específicos dos canais iônicos. À medida que mais informações sobre a ativação e desativação de canais iônicos foram obtidas, tornou-se claro a necessidade de modelos com representação explícita de estados de canais iônicos únicos (RUDY; SILVA, 2006).

Cadeias de Markov é um processo estocástico com estados discretos com a propriedade de que a distribuição de probabilidade do próximo estado depende apenas do estado atual e não do comportamento dos estados anteriores (MARKOV, 1971). Porém, as interações moleculares dos canais são geralmente estado-dependente, assim, as transições do modelo de Markov normalmente representam movimentos específicos de canais que foram caracterizados experimentalmente.

Nos modelos de cadeia de Markov, é comum assumir três estados diferentes: aberto, fechado e inativado. Inativação é quando o fluxo de íons é bloqueado por um mecanismo que não seja o fechamento do canal. O canal fechado está pronto para reabertura imediata, enquanto um canal inativado deve se recuperar da inativação antes de poder abrir (KARMAŽÍNOVÁ; LACINOVÁ, 2010).

Tome com exemplo a Figura 7 que mostra duas cadeias de Markov. Na Figura 7(a) são mostrados quatro estados hipotéticos: fechado (C) e aberto (O), mais dois estados de inativação, fechado/inativado  $I_C$  e aberto/inativado  $I_O$ . As taxas de transição  $\alpha$  e  $\beta$  entre  $I_O$  e  $I_C$  e entre C e O são idênticas, assim como as taxas de transição  $\gamma$  e  $\delta$  entre C e  $I_C$

**Figura 7 – Exemplos de cadeias de Markov. (a) Um modelo de quatro estados com duas transições independentes - C: fechado; O: aberto; I<sub>O</sub>: aberto/inativado I<sub>C</sub>: fechado/inativado.  $\alpha$  e  $\beta$ : taxas de transição entre I<sub>O</sub> e I<sub>C</sub> e entre C e O.  $\gamma$  e  $\delta$ : taxas de transição entre C e I<sub>C</sub> e entre O e I<sub>O</sub>; (b) Um modelo de três estados com transições dependentes de C para O e O para I**



Fonte: Adaptado de Rudy e Silva (2006)

e entre O e I<sub>O</sub>. Se forem usadas equações diferenciais para calcular a ocupação de cada estado, elas assumem a seguinte forma (RUDY; SILVA, 2006):

$$\begin{aligned}
 \frac{dC}{dt} &= \beta \cdot O + \delta \cdot I_C - (\alpha + \gamma) \cdot C, \\
 \frac{dO}{dt} &= \alpha \cdot C + \delta \cdot I_O - (\beta + \gamma) \cdot O, \\
 \frac{dI_C}{dt} &= \beta \cdot I_O + \gamma \cdot C - (\alpha + \delta) \cdot I_C, \\
 \frac{dI_O}{dt} &= \alpha \cdot I_C + \gamma \cdot O - (\beta + \delta) \cdot I_O.
 \end{aligned}
 \tag{2.27}$$

No modelo mostrado na Figura 7(a), as transições de ativação e inativação são independentes. A probabilidade de o canal estar em C ou em O é h, e a probabilidade de o canal estar em O ou em I<sub>O</sub> é m. Dessa maneira, como os canais são independentes, a probabilidade do canal estar aberto (O), por exemplo, é calculada pelo seu produto (m · h). Essa expressão é a mesma que teria sido derivada para a formulação de Hodgkin-Huxley. Assim, sob o pressuposto de um controle independente, o formalismo de Markov e o formalismo de Hodgkin-Huxley são intercambiáveis (RUDY; SILVA, 2006).

No entanto, experimentos mostraram que normalmente os processos de ativação e inativação de canais não são independentes. Uma versão simples do acoplamento de ativação e inativação, em um canal hipotético, é mostrada na Figura 7(b). Nesse modelo, a inativação de canal só pode ocorrer a partir do estado aberto, e a ativação e inativação de canal não envolvem transições independentes. Portanto, a suposição de ativação/inativação independente não é mais válida e o formalismo de Hodgkin-Huxley não pode ser aplicado. Na verdade, cada estado deve ser descrito individualmente por uma equação diferencial (RUDY; SILVA, 2006):

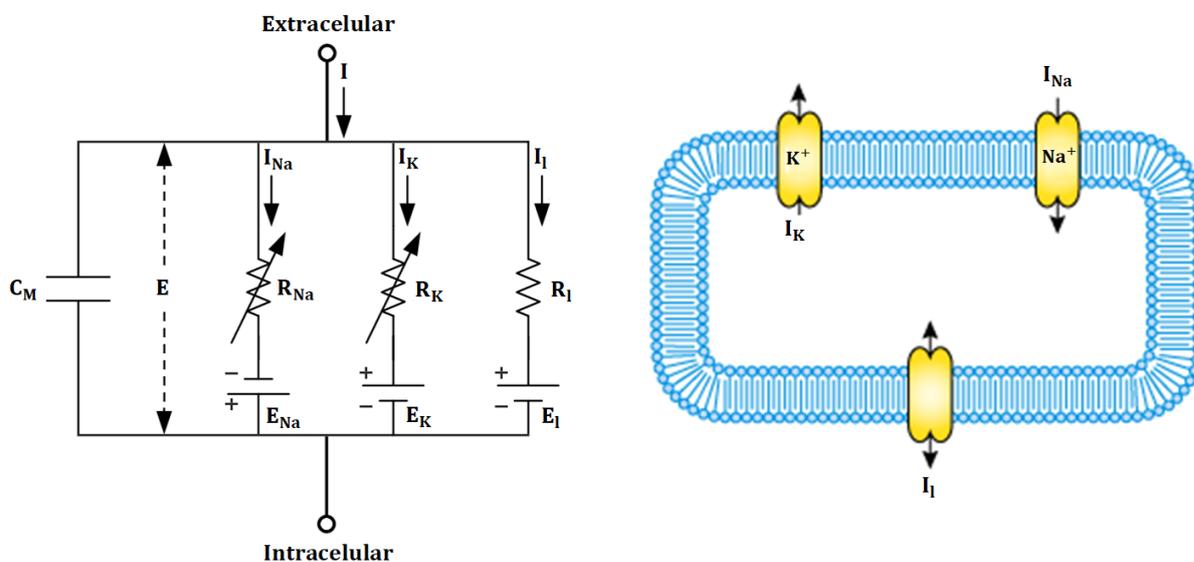
$$\begin{aligned} \frac{dC}{dt} &= \alpha \cdot C - \beta \cdot O, \\ \frac{dO}{dt} &= \alpha \cdot C + \delta \cdot I - (\beta + \gamma) \cdot O, \\ \frac{dI}{dt} &= \gamma \cdot O - \delta \cdot I. \end{aligned} \tag{2.28}$$

As cadeias de Markov abrem precedentes para modelar diferentes configurações de estados.

### 2.3.5 Modelos Celulares

O modelo Hodgkin-Huxley (HODGKIN; HUXLEY, 1952) foi o primeiro modelo de eletrofisiologia celular proposto, sendo a base de grande número de modelos mais atuais e complexos. Proposto para células nervosas, o modelo considera que a corrente iônica é dividida em íons de sódio e potássio ( $I_{Na}$  e  $I_K$ ) e uma pequena "corrente de fuga" ( $I_l$ ) composta por cloretos e outros íons. Cada componente da corrente iônica é determinado por uma força motriz que pode ser convenientemente medida como uma diferença de potencial elétrico e um coeficiente de permeabilidade que possui as dimensões de condutância. Nesse modelo, as resistências ao sódio ( $R_{Na}$ ) e ao potássio ( $R_K$ ) são funções do tempo e/ou do potencial de membrana, mas  $E_{Na}$ ,  $E_K$ ,  $E_l$ ,  $C_M$  e  $\bar{g}_l$  são consideradas constantes. A Figura 8 ilustra o circuito elétrico desse modelo, bem como mostra a representação das correntes iônicas.

Figura 8 – Circuito elétrico proposto por Hodgkin e Huxley representando a membrana e representação esquemática das correntes iônicas.  $R_{Na}=1/g_{Na}$ ;  $R_K=1/g_k$ ;  $R_l=1/g_l$



Fonte: Adaptado de Hodgkin e Huxley (1952), Physiome Project (2020a)

Baseado na Equação 2.4, a variação de potencial de membrana com o tempo é

dado por:

$$\frac{dV_m}{dt} = -\frac{1}{C_m}(I_{\text{ion}} + I_{\text{stim}}) \quad (2.29)$$

Sendo que  $I_{\text{stim}}$  é uma corrente de estímulo. A corrente transmembrânica do modelo HH é dada por:

$$I_{\text{ion}} = I_{\text{Na}} + I_{\text{K}} + I_1 \quad (2.30)$$

O modelo utiliza o modelo de corrente iônica linear (Equação 2.8), logo:

$$\begin{aligned} I_{\text{Na}} &= g_{\text{Na}}(V_m - E_{\text{Na}}), \\ I_{\text{K}} &= g_{\text{K}}(V_m - E_{\text{K}}), \\ I_1 &= \bar{g}_1(V_m - E_1). \end{aligned} \quad (2.31)$$

Para o canal iônico, é adotado o modelo de subunidades, de forma que a condutância do sódio é, de acordo com a Equação 2.24:

$$g_{\text{Na}} = m^3 h g_{\text{Na,max}} \quad (2.32)$$

onde  $g_{\text{Na,max}}$  é a condutância máxima de sódio e a variação temporal das variáveis de ativação e inativação,  $m$  e  $h$ , são dadas pelas Equações 2.25 e 2.26.

A condutância do potássio é função do potencial de membrana, mas não do tempo, sendo dada por:

$$g_{\text{K}} = g_{\text{K,max}} n^4 \quad (2.33)$$

onde  $g_{\text{K,max}}$  representa a condutividade máxima de potássio e a variação de  $n$  é dada pela Equação 2.19.

Valores para as taxas de transição podem ser encontrados em Hodgkin e Huxley (1952).

Um modelo celular mais realístico e complexo foi desenvolvido por Bondarenko *et al.* (2004), o qual corresponde a um modelo matemático do potencial de ação do miócito ventricular esquerdo de camundongo a partir de dados de *voltage clamp* das correntes subjacentes e transientes de  $\text{Ca}^{2+}$ . Na técnica de *voltage clamp* são utilizados pulsos de tensão em vez de pulsos de correntes para controlar o mecanismo da geração do potencial de ação. O modelo Bondarenko (BDK) é composto por 15 correntes transmembrânica

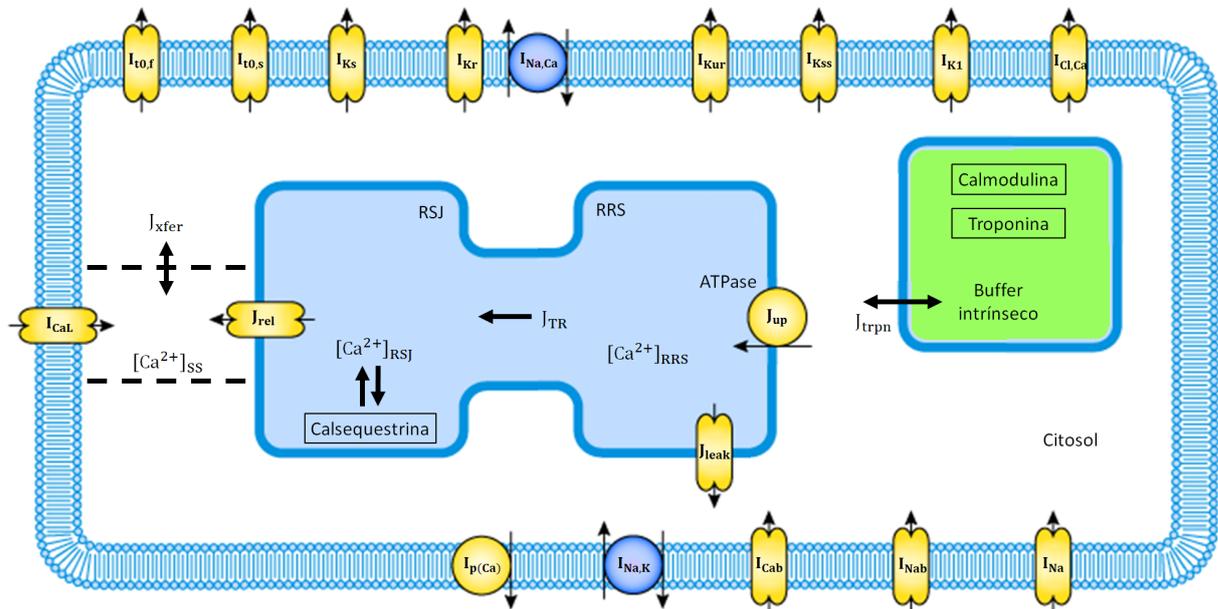
onde a maioria dos canais iônicos são representados por cadeias de Markov. A variação do potencial de membrana é dada pela Equação 2.4 e a corrente iônica é:

$$I_{ion} = I_{CaL} + I_{p(Ca)} + I_{Na/Ca} + I_{b_{Ca}} + I_{Na} + I_{b_{Na}} + I_{Na/K} + I_{K_{t0,f}} + I_{K_{t0,s}} + I_{K1} + I_{Ks} + I_{Kur} + I_{Kss} + I_{Kr} + I_{Cl,Ca} + I_{stim} \quad (2.34)$$

onde  $I_{Na}$  é a corrente rápida de  $Na^+$ ,  $I_{CaL}$  é a corrente do tipo L de  $Ca^{2+}$ ,  $I_{K_{t0,f}}$  é a corrente de saída transitória de rápida recuperação de  $K^+$ ,  $I_{K_{t0,s}}$  é a corrente de saída transitória de lenta recuperação de  $K^+$ ,  $I_{Kr}$  é a corrente retificadora rápida de  $K^+$ ,  $I_{Kur}$  é a corrente retificadora ultra rápida de  $K^+$ ,  $I_{Kss}$  é a corrente ativada por tensão de estado estacionário não inativante de  $K^+$ ,  $I_{K1}$  é a corrente retificadora interna independente do tempo de  $K^+$ ,  $I_{Ks}$  é a corrente retificadora lenta de  $K^+$ ,  $I_{Na/Ca}$  é a corrente de troca  $Na^+/Ca^{2+}$ ,  $I_{p(Ca)}$  é a corrente bomba de  $Ca^{2+}$ ,  $I_{Na/K}$  é a corrente bomba de  $Na^+/K^+$ ,  $I_{Cl,Ca}$  é a corrente  $Cl^-$  ativada por  $Ca^{2+}$ ,  $I_{b_{Ca}}$  e  $I_{b_{Na}}$  são as correntes de *background* de  $Ca^{2+}$  e  $Na^+$  e  $I_{stim}$  é a corrente externa de estímulo.

A Figura 9 mostra a representação esquemática das correntes iônicas do modelo BDK.

Figura 9 – Representação esquemática das correntes iônicas do modelo Bondarenko



Fonte: Adaptado de Bondarenko *et al.* (2004)

Os fluxos de  $Ca^{2+}$  dentro da célula são captados do  $Ca^{2+}$  do citosol para a rede do retículo sarcoplasmático (RS) ( $J_{up}$ ), liberação de  $Ca^{2+}$  do RS juncional ( $J_{rel}$ ), fluxo de  $Ca^{2+}$  da rede RS (RRS) para o RS juncional (RSJ) ( $J_{tr}$ ), fuga de  $Ca^{2+}$  do RS para o

citosol ( $J_{\text{leak}}$ ), fluxo de  $\text{Ca}^{2+}$  do volume do subespaço para o mioplasma em massa ( $J_{\text{xfer}}$ ) e fluxo de  $\text{Ca}^{2+}$  para a troponina ( $J_{\text{trpn}}$ ). O modelo inclui o buffer de  $\text{Ca}^{2+}$  pela troponina e calmodulina no citosol e pela calsequestrina no RS.

O modelo Bondarenko é composto por 41 EDOs. Descrições completas das correntes, parâmetros, constantes e das variáveis de estado desse modelo podem ser verificadas em Bondarenko *et al.* (2004).

Outros modelos baseados no modelo Hodgkin-Huxley, como o modelo de Luo Rudy para células do ventrículo de mamíferos e os modelos Ten Tusscher, Ten Tusscher Panfilov e Ten Tusscher Panfilov 2 mais específicos para células do ventrículo humano podem ser visualizados no Apêndice A.

## 2.4 Programação paralela, distribuída, memória compartilhada e GPU

Programação paralela compreende a identificação e redução de dependências entre as seções de um código e, para que seja de fato paralela, é necessário a existência de uma arquitetura paralela (por exemplo, um processador *multi-core*), enquanto a computação paralela envolve o uso de bibliotecas de paralelismo para examinar essas seções em arquiteturas paralelas (SITSYLITSYN, 2020).

A programação distribuída é uma metodologia adotada para aumentar a escalabilidade da execução de código paralelo usando uma rede de comunicações (SITSYLITSYN, 2020). A programação distribuída não é necessariamente usada para concorrência, pois o processamento pode ser distribuído e sequencial (SITSYLITSYN, 2020). Dessa forma, na programação distribuída, é necessário um modelo de passagem de mensagem, onde os processos que residem em diferentes nós de processadores possam se comunicar uns com os outros entregando mensagens pela rede (LI; ZHANG, 2018). A comunicação por mensagens pode ocorrer não apenas em um multiprocessador com variáveis compartilhadas, mas também em um *cluster* com armazenamento distribuído (LI; ZHANG, 2018). Nesse contexto, *Message Passing Interface* (MPI) é um padrão para comunicação de dados em computação paralela que foi desenvolvido para arquitetura de memória compartilhada, capaz de fornecer portabilidade, eficiência, flexibilidade e disponibilidade (SITSYLITSYN, 2020).

O compartilhamento de informações durante a execução está ligado diretamente ao tipo de arquitetura utilizado. É diferente quando usado memória compartilhada: as tarefas em execução compartilham um mesmo espaço de memória e a comunicação é através do acesso a uma área compartilhada realizada através de acessos do tipo *load* e *store* a uma área de endereçamento comum. Para utilização correta da área de memória compartilhada é necessário que os processos coordenem seus acessos utilizando primitivas de sincronização.

Nas simulações numéricas de eletrofisiologia cardíaca, a grande quantidade de variáveis de estado e parâmetros dos modelos celulares mais recentes, juntamente com malhas muito finas, recaem em um problema onde a solução requer o uso de algoritmos com maior nível de paralelismo em plataformas *multi-core* (MENA *et al.*, 2015; XIA *et al.*, 2015). Porém, o alcance de desempenho satisfatório depende de uma forte escalabilidade, ao mesmo tempo em que o desempenho pode ser deteriorado com o aumento do número de núcleos de computação, devido ao aumento do custo de comunicação entre os processos. Uma alternativa para contornar tais problemas é a utilização de unidades de processamento gráfico de propósito geral (*General Purpose Graphics Processing Unit - GPGPU*), as quais superam as arquiteturas de unidades de processamento centrais (*Central Processing Unit - CPU*) *multi-core* em termos de largura de banda de memória, mas têm desempenho inferior em termos de aritmética de ponto flutuante de precisão dupla (MENA *et al.*, 2015).

Uma GPU é um processador *unichip* usado principalmente para aumentar o desempenho de vídeo e gráficos (SINGH *et al.*, 2020). Enquanto uma CPU pode trabalhar em uma variedade de cálculos diferentes, uma GPU se concentra em uma tarefa específica com todo o seu poder de computação (SINGH *et al.*, 2020). CUDA (*Compute Unified Device Architecture*) é uma plataforma de computação paralela baseada em um modelo de programação chamado CUDA C, desenvolvido pela Nvidia que fornece uma extensa estrutura de programação adequada para GPGPU (DEHAL *et al.*, 2018). A estrutura do programa CUDA reflete a coexistência do *host* (CPU) e um ou mais dispositivos (GPUs) no computador (SINGH *et al.*, 2020; DEHAL *et al.*, 2018).

Obter desempenho a partir do escalonamento multi-GPU, ou seja, mais de uma placa de vídeo trabalhando juntas, no entanto, não é trivial, principalmente porque as GPUs atuais são conectadas aos processadores através da interface PCI Express, a qual limita o acesso da GPU à memória principal do sistema. Essa banda de conexão é cerca de quatro vezes menor que a da velocidade da própria memória *RAM* (*Random Access Memory*), e isso acaba sendo um gargalo toda vez que a GPU precisa de algum dado que está armazenado na memória *RAM* principal da máquina (LI *et al.*, 2019). O NVLink foi desenvolvido pela NVIDIA e é baseado na arquitetura PCI Express, funcionando como uma troca rápida de dados entre CPU e GPU, com o objetivo de eliminar os possíveis gargalos, fazendo com que a velocidade de comunicação seja igual à velocidade da memória do sistema. Um NVLink pode ser visto como um cabo com dois plugues terminais, enquanto cada GPU incorpora vários *slots* NVLink (LI *et al.*, 2019). A forma como esses *slots* são conectados por meio dos cabos NVLink determina a topologia e a largura de banda da rede da GPU (LI *et al.*, 2019). Vários cabos podem ser agrupados para aumentar a largura de banda quando estiverem ligando os mesmos terminais (LI *et al.*, 2019).

## 2.5 Estratégias de implementação das simulações numéricas da eletrofisiologia cardíaca

Os modelos celulares (descritos na Seção 2.3.5 e Apêndice A) incluem equações diferenciais ordinárias (EDO) e parciais (EDP), as quais são discretizadas em equações algébricas de forma que possam ser resolvidos numericamente de forma iterativa. A forma de discretização dessas equações depende do método numérico empregado, por exemplo, elementos finitos, diferenças finitas, volumes finitos, entre outros. Independentemente do método numérico, o domínio de aplicação de tais equações é subdividido em elementos menores, formando assim a malha.

Oliveira *et al.* (2012) avaliaram um algoritmo numérico da eletrofisiologia cardíaca de células ventriculares baseado em malha adaptativa e no método dos volumes finitos. Foi proposto um modelo monodomínio com modelo de corrente iônica de Luo Rudy. O sistema linear foi resolvido pelo método do gradiente conjugado. Os autores utilizaram o método de malha adaptativa chamado de *ALG (Autonomous Leaves Graph)* (BURGARELLI *et al.*, 2006). O *ALG* é uma estrutura de dados que pode ser integrado ao solver de sistema lineares para representar diferentes geometrias com refinamento adaptativo de complexos domínios. Oliveira *et al.* (2012) compararam os resultados encontrados na malha adaptativa com o mesmo problema solucionado de forma não adaptativa. Os resultados mostraram que, dependendo da configuração mínima e máxima do tamanho dos elementos, o *speed up* atingiu valores superiores a 80, com um erro menor que 2,3%. Foi mostrado também que o tempo gasto para reconstruir a matriz a cada refino ou desrefino da malha não foi significativo, sendo menor que 5% do tempo total de solução do sistema linear. Ao mesmo tempo, o tempo da solução do sistema linear pelo gradiente conjugado foi reduzido quando adotado a metodologia com malha adaptativa, onde a menor diferença foi de aproximadamente 90%.

Outra estratégia adotada para acelerar as simulações de eletrofisiologia cardíaca é a implementação dos modelos por meio de computação paralela e distribuída. É esperado que a divisão do trabalho em um determinado número de processos melhore consideravelmente o desempenho, reduzindo o tempo de execução da simulação numérica, porém outros fatores inerentes a essas estratégias tendem a aumentar o tempo de execução, tais como, o tempo de comunicação e a sincronização entre os processos (QUINN, 2003).

Oliveira *et al.* (2016) combinaram a estratégia de malha adaptativa *ALG* (OLIVEIRA *et al.*, 2012) com a solução das EDOs em GPU, usando CUDA, em um domínio *benchmark* (NIEDERER *et al.*, 2011) tridimensional. Além disso, para o cálculo das EDPs e montagem e remontagem da matriz adotou-se a paralelização em *OpenMP*. Na solução da EDP adotou-se o método do gradiente conjugado, a qual a solução foi acelerada por meio do pré-condicionador de Jacobi (SANTOS *et al.*, 2005). Foram realizados dois testes: (i)

modelo celular de Ten Tusscher-Panfilov (TT2) (TUSSCHER; PANFILOV, 2006a), onde para solução das EDOs adotou-se o método Rush-Larsen (RL) (RUSH; LARSEN, 1978), com passo de tempo de 50  $\mu$ s para EDP e EDOs; (ii) modelo celular BDK (BONDARENKO *et al.*, 2004), onde as EDOs foram solucionadas pelo método de Euler explícito, com passo de tempo de 50  $\mu$ s para EDP e 0,1  $\mu$  para as EDOs. O *speed up* foi calculado com base em uma malha fixa de 100  $\mu$ m de comprimento característico. Em (i) obteve-se *speed up* de 118,17 e 51,02 para malha mais e menos refinada, respectivamente. Já em (ii), o *speed up* foi ainda maior, 626,75 e 292,70 para malha mais e menos refinada, respectivamente.

Dando continuidade ao modelo combinado de malha adaptativa e paralelização, Oliveira *et al.* (2018b) propuseram agregar além das estratégias de Oliveira *et al.* (2016), o passo de tempo adaptativo e a paralelização das EDOs em *threads* em CPU e GPU. Ao utilizar o passo de tempo adaptativo pelo método de Euler (CAMPOS *et al.*, 2013), teve-se por objetivo reduzir o passo de tempo apenas nas regiões onde fosse realmente necessário (devido a elevados gradientes e problemas de instabilidade, por exemplo) e, aumentá-lo em demais regiões. Além disso, um terceiro teste foi realizado, o qual consistiu na simulação de uma geometria realística gerada por ressonância magnética do ventrículo esquerdo de um camundongo adulto. Nesse teste 3, foram adotadas as mesmas condições do teste 2. Como resultados, o *speed up* máximo para o teste 1 foi de 51, o qual foi atingido com a combinação de malha adaptativa, paralelização *OpenMP* e GPU (nesse teste não foi adotado o passo de tempo adaptativo). Para o teste 2, o *speed up* foi de 498, o qual foi atingido nas mesmas condições do teste 1 acrescido do passo de tempo adaptativo; e para o teste 3 com onda única e onda em espiral os *speed ups* foram 205 e 165, respectivamente, atingido nas mesmas condições do teste 2. Os valores de *speed up* dos casos 1 e 2 foram próximos ao de Oliveira *et al.* (2016), porém aqui a malha foi mais refinada, o que mostrou a eficiência da combinação de malha adaptativa, passo de tempo adaptativo e paralelização, além de ter mostrado a capacidade em simular geometrias mais realísticas. Assim como em Oliveira *et al.* (2016), o *speed up* foi calculado com base em uma malha fixa de 100  $\mu$ m de comprimento característico

As condições pós infarto devido ao processo isquêmico, como hipóxia, surgimento de arritmias, formação de fibrose, formação de marcapassos ectópicos, remodelação do tecido entre outros, também têm sido material de estudo no campo da modelagem numérica. Tusscher e Panfilov (2007) utilizando o modelo TT2 em uma geometria tridimensional, analisaram o papel da fibrose na propagação de ondas e o mecanismo de arritmia no tecido ventricular humano. Os autores mostraram que a fibrose retardou a propagação de ondas e aumentou a vulnerabilidade do tecido à quebra de ondas e formação de ondas em espiral. Foi possível detectar também que a fibrose aumentou o período de arritmias reentrantes e pôde suprimir a transição induzida pela restituição da taquicardia para a fibrilação.

Alonso *et al.* (2016) propuseram um modelo tridimensional para análise de reentrada e marcapassos ectópicos do tecido cardíaco. De forma geral, os autores mostraram que uma região confinada com microfibrese pôde gerar microreentradas, de forma que tais regiões ativas puderam atuar como marcapassos ectópicos para o tecido circundante. Os resultados por eles encontrados indicaram que a probabilidade de reentrada aumentava quando se passava de tecidos muito excitáveis (remodelação forte) para pouco excitáveis (remodelação intermediária) e diminuía quando se passava da remodelação pouco excitáveis para remodelação normal. Por fim, os autores também mostraram que a probabilidade de reentrada depende do tamanho da região fibrótica, de forma que é mais provável que ocorram nas maiores regiões.

Oliveira *et al.* (2018) investigaram o mecanismo de micro-reentrada que leva à geração de batimentos ectópicos próximos a áreas de infarto. A geometria de paciente específico utilizada foi obtida por meio de ressonância magnética, a qual incluía zonas de cicatrizes e fronteiras ao infarto. Foram analisadas duas regiões potenciais de geração de batimentos ectópicos, uma região maior com grande cicatriz e outra região de tamanho menor. Foi utilizado um modelo celular desenvolvido nesse mesmo trabalho - modelo Oliveira (Apêndice A.3). A região do infarto foi composta por células condutoras e não condutoras (fibrese), de forma que os padrões dos meios bifásicos foram criados por métodos estocásticos. Após a ativação normal por vários locais na superfície endocárdica, a propagação do PA fracionou dentro da região do infarto e deu origem a micro-reentradas sustentadas. Essas micro-reentradas deixava a região do infarto e re-excitava o tecido saudável circundante. As reentradas sustentadas ocorreram quando uma porcentagem de células não condutoras se aproximava do limiar de percolação. Também por meio de modelos computacionais de eletrofisiologia cardíaca, Sachetto *et al.* (2018) mostraram que o mecanismo por trás do início das batidas ectópicas foi o bloqueio unidirecional, o qual originava micro-reentradas sustentadas dentro da região com fibrese difusa e hipóxia, ou seja, hipóxia e fibrese foram suficientes para a criação de uma região focal que gerava os batimentos ectópicos.

Outro projeto para simulações da eletrofisiologia cardíaca, o qual abrange simulações a níveis celular, tecidual, órgãos e corporal é o *openCARP* (PLANK *et al.*, 2021), implementado em linguagem C++. Juntamente com a estrutura *carputils* baseada em Python, esse projeto visa automatizar experimentos *in silico*, por meio de desenvolvimento e compartilhamento de *pipelines* de simulação, incluindo todas as etapas de modelagem e simulação com o objetivo de aumentar a reprodutibilidade e a produtividade. O *openCARP* se mostrou mais eficiente do que outro simulador cardíaco, o *cbcbeat* (ROGNES *et al.*, 2017), superando-o em 8 a 9 vezes quando executado em série o *benchmark* proposto por Niederer *et al.* (2011) (JØRGENSEN, 2021). Por outro lado, o *cbcbeat* pareceu se beneficiar mais da paralelização do que o *openCARP* (JØRGENSEN, 2021). A maior demanda de tempo na solução do problema pelo *cbcbeat*, foi na solução das EDOs e na

sobrecarga ao calcular as etapas de integração (JØRGENSEN, 2021). O *openCARP* por sua vez, foi capaz de solucionar as EDOs quase 20 vezes mais rápido que o *cbcbeat* na execução serial (JØRGENSEN, 2021). Dentre outras aplicações, o *cbcbeat* foi utilizado no mapeamento de ativação eletroanatômica, que é um procedimento importante no diagnóstico de fibrilação atrial (COSTABAL *et al.*, 2020) e o *openCARP* foi utilizado, por exemplo, para o estudo da determinação das condições em que a estimulação de superfície de baixa energia desfibrila grandes ventrículos de mamíferos (MORENO *et al.*, 2022) e na quantificação do efeito da espessura da parede miocárdica nas amplitudes e duração de eletrocardiogramas (LEENKNEGT *et al.*, 2022).

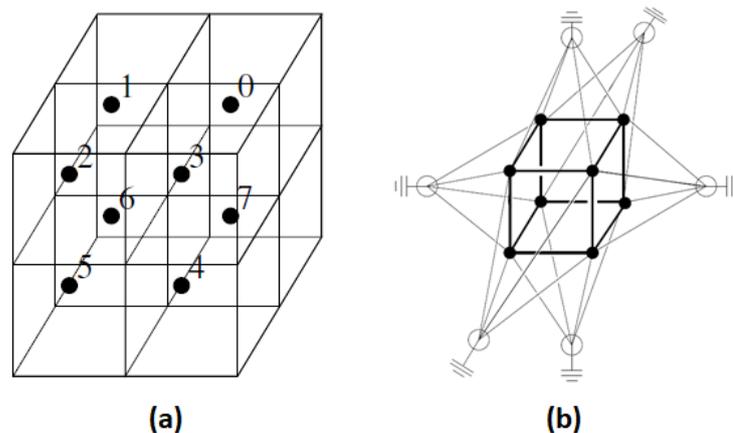
### 3 MATERIAIS E MÉTODOS

#### 3.1 Estrutura de dados ALG

*Autonomous Leaves Graph* (BURGARELLI *et al.*, 2006) é uma estrutura de dados baseada em grafos que lida com a comunicação de células em domínios discretizados para resolver numericamente EDPs evolutivas. A estrutura *ALG* pode ser integrada ao *solver* de sistemas lineares e representar diferentes geometrias. O tempo de processamento gasto na comunicação entre células vizinhas é independente do número de células presentes na discretização (ou seja,  $O(1)$  para cada célula). Essa estrutura de dados foi favoravelmente comparada às estruturas de dados baseadas em árvore comumente usadas (*quad-tree*) (BURGARELLI *et al.*, 2006). Neste trabalho, como será descrito a seguir, foi implementado o *ALG* para geometria tridimensional.

Considere a malha da Figura 10(a), a qual é composta por 8 células. Para construção da estrutura de dados *ALG*, é necessário formar o cubo unitário mostrado na Figura 10(b) por meio da conexão dos centros de cada elemento da malha da Figura 10(a). Como cada vértice de um cubo é ligado a outros 6 vértices (esquerda, direita, cima, baixo, frente e trás), ao formar o cubo unitário, aqueles centros de célula que não se ligam a outros centros, são conectados a novos vértices auxiliares (representados em branco na Figura 10(b)). Cada vértice auxiliar é ligado por arestas aos quatro vértices de cada face correspondente. Por fim, existe ainda outra ligação dos vértices auxiliares que aponta para um vértice chamado de terra, a qual é um ponteiro nulo. Para complementar a notação, as arestas indicam uma ligação dupla, de forma que quando a ligação é simples, a aresta é substituída por uma seta indicando a direção da ligação.

**Figura 10 – Grafo resultante após a discretização de um cubo unitário**



Fonte: Adaptado de Burgarelli (1998)

Dessa forma, a estrutura de dados *ALG* é composta por dois tipos de nós, os nós células (pretos) que são as células da malha propriamente dito e os nós de transição (brancos), que conectam células sem conexão com outras células, ou células com diferentes níveis de refinamento. Os nós células são compostos por 6 ponteiros para outros nós (pretos ou brancos) e mais dois ponteiros chamados *next* e *previous*, utilizados para ordenar os nós células no grafo. Além disso, os nós célula armazenam suas informações espaciais, informação do tipo de nó, nível de refinamentos e a grandeza de interesse. Já os nós de transição possuem 5 ponteiros, dos quais 4 ponteiros apontam para nós célula e o quinto ponteiro aponta para um nó de transição ou para um vértice terra. Os nós de transição também armazenam informações do tipo de nó e do nível de refinamento.

Nesta proposta, foi implementado o *ALG* para geometria tridimensional sem a utilização de malha adaptativa para refinamento e desrefinamento, pois como a nova versão do ALG3D implementa também a resolução do sistema linear associado a EDP utilizando a GPU, verificamos que não há mais ganho computacional associado a utilização de malhas adaptativas.

### 3.2 Discretização das equações governantes

Foi adotado o modelo monodomínio de eletrofisiologia cardíaca, o qual é dado em termos do potencial transmembrânico acoplado a um sistema de EDOs em termo das variáveis de estado  $\boldsymbol{\eta}$ . O conjunto de equações pode ser resumido nas Equações 3.1 e 3.2. Essas EDPs governam a cinética das variáveis de estado  $\boldsymbol{\eta}$  que influenciam a geração do potencial de ação e geralmente incluem a concentração intracelular de diferentes espécies de íons ( $K^+$ ,  $Na^+$ ,  $Ca_{2+}$ ) e a permeabilidade de diferentes canais de íons.

$$\beta C_m \frac{\partial V}{\partial t} + \beta I_{ion}(V, \boldsymbol{\eta}) = \nabla \cdot (\boldsymbol{\sigma} \nabla V) + I_{stim} \quad (3.1)$$

$$\frac{\partial \boldsymbol{\eta}}{\partial t} = f(V, \boldsymbol{\eta}) \quad (3.2)$$

Para solução das EDOs foi utilizado a técnica denominada operador de *splitting* (STRIKWERDA, 2004), que é uma técnica utilizada para dividir uma expressão complexa em uma sequência de expressões mais simples, de forma a facilitar a solução da EDO, permitindo utilizar então diferentes métodos numéricos para cada expressão. Além disso, a técnica do operador de *splitting* permite eliminar dependências complexas entre variáveis. No trabalho aqui desenvolvido, a parte da reação e difusão das equações do monodomínio foi dividida pelo operador de *splitting* Godunov (SUNDNES *et al.*, 2007), de forma a separar as EDOs da EDP, criando assim um sistema de EDOs não lineares e uma EDP parabólica linear (Equações 3.3, 3.4 e 3.5). Detalhes dessa metodologia podem

ser encontrados em Oliveira (2013), Santos *et al.* (2005).

$$\frac{\partial V}{\partial t} = \frac{1}{C_m} [-I_{ion}(V, \boldsymbol{\eta}_i) + I_{stim}] \quad (3.3)$$

$$\frac{\partial \boldsymbol{\eta}_i}{\partial t} = f(V, \boldsymbol{\eta}_i) \quad (3.4)$$

$$\beta C_m \frac{\partial V}{\partial t} = [\nabla \cdot (\boldsymbol{\sigma} \nabla V)] \quad (3.5)$$

### 3.2.1 Método dos volumes finitos aplicado ao modelo monodomínio (EDP)

A discretização espacial da EDP (Equação 3.5) resulta em um sistema linear de equações que deve ser resolvido a cada passo de tempo. O método dos volumes finitos (MVF) é um método matemático para discretização e solução de equações diferenciais. O desenvolvimento do MVF está ligado a problemas envolvendo leis de conservação, onde há o conceito de fluxo de uma grandeza,  $\phi$ , atravessando uma determinada fronteira de um volume de controle. O MVF pode ser resumido nos seguintes passos: (i) decomposição do domínio em volumes de controle; (ii) integração das equações que regem o fluxo de  $\phi$  em todos os volumes de controle; (iii) discretização - conversão das equações integrais resultantes em um sistema de equações algébricas; (iv) solução das equações algébricas por um método iterativo (VERSTEEG; MALALASEKERA, 2007).

#### 3.2.1.1 Discretização temporal (EDP)

A derivada temporal da Equação 3.5 foi discretizada pelo método de Euler implícito de primeira ordem, o qual é incondicionalmente estável, o que possibilitou a utilização de passos de tempo maiores, desde que esses valores não introduzissem erros numéricos significativos, pois por tratar-se de um método de primeira ordem, a precisão dos resultados pode ser significativamente reduzida. A discretização temporal da Equação 3.5 pelo método implícito de Euler é computada pela Equação 3.6.

$$\frac{\partial V}{\partial t} = \frac{V^{n+1} - V^n}{\Delta t_p} = f(V^{n+1}) \Rightarrow V^{n+1} = V^n + \Delta t_p f(V^{n+1}) \quad (3.6)$$

onde  $V^n$  é o potencial transmembrânico no instante de tempo  $t_n$  e  $\Delta t_p$  é o passo de tempo para EDP.

## 3.2.1.2 Discretização espacial (EDP)

O termo difusivo da Equação 3.5 foi discretizado baseado em Harrild e Henriquez (1997). Para isso, considere:

$$J = -\sigma \nabla V \quad (3.7)$$

onde  $J$  ( $\mu\text{A}/\text{cm}^2$ ) é a densidade do fluxo da corrente intracelular e:

$$\nabla \cdot J = -I_m \quad (3.8)$$

onde  $I_m$  ( $\mu\text{A}/\text{cm}^3$ ) é a corrente transmembrânica volumétrica e corresponde ao lado direito da Equação 3.5.

Integrando a Equação 3.8 sobre cada volume de controle:

$$\int_{\Omega} \nabla \cdot J dv = - \int_{\Omega} I_m dv \quad (3.9)$$

Aplicando o teorema da divergência (VERSTEEG; MALALASEKERA, 2007):

$$\int_{\Omega} \nabla \cdot J dv = - \int_{\partial\Omega} J \cdot \mathbf{n} ds \quad (3.10)$$

sendo  $\mathbf{n}$  o vetor normal à superfície  $s$ . Substituindo a Equação 3.10 na Equação 3.5:

$$\beta C_m \int_{\Omega} \frac{\partial V}{\partial t} dv = - \int_{\partial\Omega} J \cdot \mathbf{n} ds \quad (3.11)$$

Nesse trabalho foi adotada uma malha tridimensional formada por cubos regulares, logo cada célula tem volume  $h^3$ . Além disso, a integral de  $J$  foi discretizada pela soma dos fluxos nas 6 faces multiplicado pela devida área da face da célula ( $S_l$ ). Assim tem-se a Equação 3.5 discretizada espacialmente:

$$h^3 \beta C_m \frac{\partial V}{\partial t} = \sum_{l=1}^6 J_l S_l \quad (3.12)$$

É importante destacar que no MVF, a grandeza de interesse,  $\phi$ , que no caso aqui analisado é  $V$ , é armazenada no centro do volume de controle.

### 3.2.2 MVF aplicado ao ALG

Nessa seção será mostrado como a discretização espacial pelo MVF é aplicada na estrutura de dados ALG. Essa é a mesma metodologia utilizada por Oliveira *et al.* (2016), Oliveira *et al.* (2018b), Oliveira (2013).

Seja a célula hexaédrica centrada em  $(i,j,k)$ , o somatório dos fluxos ( $J_1$ ) sobre as faces ( $S_1$ ) é calculado por:

$$\sum_{l=1}^6 J_l S_l = S_1 J_{x_{i+1/2,j,k}} - S_2 J_{x_{i-1/2,j,k}} + S_3 J_{y_{i,j+1/2,k}} - S_4 J_{y_{i,j-1/2,k}} + S_5 J_{y_{i,j,k+1/2}} - S_6 J_{y_{i,j,k-1/2}} \quad (3.13)$$

Os fluxos  $J_{x_{m,n,o}}$ ,  $J_{y_{m,n,o}}$  e  $J_{z_{m,n,o}}$  são calculados nas faces da célula  $((m,n,o) = (i+1/2,j,k), (i-1/2,j,k), (i,j+1/2,k), (i,j-1/2,k), (i,j,k+1/2), (i,j,k-1/2))$  da seguinte maneira:

$$\begin{aligned} J_{x_{m,n,o}} &= \sigma_x(m, n, o) \left. \frac{\partial V}{\partial x} \right|_{(m,n,o)} \\ J_{y_{m,n,o}} &= \sigma_y(m, n, o) \left. \frac{\partial V}{\partial y} \right|_{(m,n,o)} \\ J_{z_{m,n,o}} &= \sigma_z(m, n, o) \left. \frac{\partial V}{\partial z} \right|_{(m,n,o)} \end{aligned} \quad (3.14)$$

As componentes do tensor das condutividades,  $\boldsymbol{\sigma} = [\sigma_x, \sigma_y, \sigma_z]$ , foram determinadas nas faces das células (embora seus valores estejam armazenados no centro) por uma média harmônica, como mostra a Equação 3.14:

$$\sigma_{x_{i+1/2,j,k}} = \frac{2\sigma_{x_{i,j,k}}\sigma_{x_{i+1,j,k}}}{\sigma_{x_{i+1,j,k}} + \sigma_{x_{i,j,k}}} \quad (3.15)$$

As demais condutividades,  $\sigma_{x_{i-1/2,j,k}}$ ,  $\sigma_{y_{i,j+1/2,k}}$ ,  $\sigma_{y_{i,j-1/2,k}}$ ,  $\sigma_{z_{i,j,k+1/2}}$  e  $\sigma_{z_{i,j,k-1/2}}$  foram calculadas de forma análoga à Equação 3.15.

Como mostrado nas Equações 3.14, os cálculos dos fluxos  $J$  envolvem a determinação do gradiente do potencial  $V$  nas interfaces. O cálculo desses gradientes foi realizado pelo método dos volumes finitos aplicado à malha da estrutura de dados ALG. Essa metodologia é mostrada a seguir para a direção  $x$ :

$$\begin{aligned}\frac{\partial V}{\partial x}\Big|_{(i+1/2,j,k)} &= \sum_{c=1}^{m_1} \frac{V_{r,c} - V_{i,j,k}}{h} \\ \frac{\partial V}{\partial x}\Big|_{(i-1/2,j,k)} &= \sum_{c=1}^{m_2} \frac{V_{i,j,k} - V_{l,c}}{h}\end{aligned}\quad (3.16)$$

onde  $m_1$  e  $m_2$  são o número de vizinhos à direita e à esquerda da célula centrada em  $(i,j,k)$ , respectivamente.  $V_{r,c}$  e  $V_{l,c}$  são os potenciais dos vizinhos à direita e à esquerda da célula  $(i,j,k)$ , respectivamente.

Dessa forma, a discretização dos fluxos (Equação 3.14), multiplicada pela área das faces das células se torna:

$$\begin{aligned}J_1 S_1 &= -\sigma_{x_{i+1/2,j,k}} \sum_{c=1}^{m_1} \frac{V_{r,c} - V_{i,j,k}}{h} S_1 \\ J_2 S_2 &= -\sigma_{x_{i-1/2,j,k}} \sum_{c=1}^{m_2} \frac{V_{i,j,k} - V_{l,c}}{h} S_2\end{aligned}\quad (3.17)$$

Para o trabalho aqui desenvolvido, foi considerado uma malha hexaédrica regular, logo as áreas das faces das células são  $S_1=h_1^2$  e  $S_2=h_2^2$  de forma que pode-se simplificar a Equação 3.17:

$$\begin{aligned}S_1 J_1 &= S_1 J_{x_{i+1/2,j,k}} = -\sum_{c=1}^{m_1} \sigma_{x_{r',c}} (V_{r,c} - V_{i,j,k}) h \\ S_2 J_2 &= S_2 J_{x_{i-1/2,j,k}} = -\sum_{c=1}^{m_2} \sigma_{x_{l',c}} (V_{i,j,k} - V_{l,c}) h\end{aligned}\quad (3.18)$$

As equações para as direções  $y$  e  $z$  são análogas as da direção  $x$ , descritas acima. Rearranjando e substituindo as discretizações na Equação 3.12 e considerando que  $n$  é o passo de tempo atual e  $*$  um passo de tempo intermediário:

$$C_m \frac{V_{i,j,k}^* - V_{i,j,k}^n}{\Delta t} = -\frac{(S_1 J_1^* - S_2 J_2^* + S_3 J_3^* - S_4 J_4^* + S_5 J_5^* - S_6 J_6^*)}{\beta k_{i,j,k}^3}\quad (3.19)$$

Fazendo também as devidas substituições das Equações 3.18 na Equação 3.19 e rearranjando, tem-se a PDE discretizada:

$$\begin{aligned}
& \alpha V_{i,j,k}^* - \sum_{c=1}^{m_1} \sigma_{x_{r',c}} (V_{r,c}^* - V_{i,j,k}^*) + \sum_{c=1}^{m_2} \sigma_{x_{l',c}} (V_{i,j,k}^* - V_{l,c}^*) \\
& - \sum_{c=1}^{m_3} \sigma_{y_{t',c}} (V_{t,c}^* - V_{i,j,k}^*) + \sum_{c=1}^{m_4} \sigma_{y_{b',c}} (V_{i,j,k}^* - V_{b,c}^*) \\
& - \sum_{c=1}^{m_5} \sigma_{z_{f',c}} (V_{f,c}^* - V_{i,j,k}^*) + \sum_{c=1}^{m_6} \sigma_{z_{bk',c}} (V_{i,j,k}^* - V_{bk,c}^*) = V_{i,j,k}^n \alpha
\end{aligned} \tag{3.20}$$

onde  $\alpha = (\beta C_m h_{i,j}^3) / \Delta t_p$ ;  $m_1, m_2, m_3, m_4, m_5$  e  $m_6$  correspondem ao número de vizinhos à direita, à esquerda, acima, abaixo, à frente e atrás, respectivamente;  $V_{r,c}, V_{l,c}, V_{t,k}, V_{b,k}, V_{f,k}$  e  $V_{bk,k}$  são os potenciais dos vizinhos à direita, à esquerda, acima, abaixo, à frente e atrás, respectivamente.

Essa equação deve ser resolvida para cada elemento da célula, formando assim, um sistema linear (matriz), o qual foi resolvido nesse trabalho pelo método do gradiente conjugado Apêndice B.

### 3.2.3 Discretização das EDOs

As EDOs que compõe o modelo de células cardíacas são equações que envolvem derivadas das funções com apenas uma variável independente, que no caso aqui em questão é o tempo. Dessa forma é necessário apenas a discretização temporal dessas equações.

Como foi utilizado o operador de *splitting* separando as EDOs da EDP, foi possível utilizar um método de discretização diferente para as EDOs do que o método de Euler implícito adotado para a EDP. Por exigir menos esforço computacional que outros modelos (OLIVEIRA, 2013), o método de Euler explícito foi utilizado nesse trabalho para discretização das EDOs. Segundo Oliveira (2013), em modelos complexos de eletrofisiologia, o passo de tempo exigido para outros modelos de discretização temporal, se torna da mesma ordem do método de Euler explícito, devido a problemas de estabilidade. No método de Euler explícito, a discretização temporal para uma variável  $u$ , qualquer, se torna:

$$\frac{\partial u^j}{\partial t} = \frac{u_{n+1}^j - u_n^j}{\Delta t_n} = f(u_n) \tag{3.21}$$

Apesar de sua simplicidade, o método de Euler explícito não é incondicionalmente estável, de forma que a utilização de passos de tempo muito pequenos pode ser necessária, o que acarreta um alto tempo de solução. Para contornar esse problema, foi utilizada uma metodologia para utilização de um passo de tempo adaptativo desenvolvida por Campos *et al.* (2011). Essa metodologia mostrou-se mais eficiente até mesmo que métodos implícitos,

como o de Euler implícito, na solução das EDOs que descrevem a eletrofisiologia cardíaca (OLIVEIRA, 2013), o que justifica a adoção dessa metodologia.

### 3.2.3.1 Passo de tempo adaptativo (EDOs)

Na metodologia desenvolvida por Campos *et al.* (2011), a escolha do novo passo de tempo envolve além da discretização pelo método de Euler explícito, a discretização pelo método de segunda ordem de Runge-Kutta (RK2) (BUTCHER, 2016). Para um sistema de  $k$  equações tem-se a seguinte forma:

$$\begin{aligned}
 u_{n+1}^{jEU} &= u_n^j \Delta t_n f^j(t_n, \vec{u}_n), \quad j = 1, \dots, k \\
 u_{n+1}^{jRK} &= u_n^j \frac{\Delta t_n}{2} (f^j(t_n, \vec{u}_n) + f^j(t_{n+1}, \vec{u}_{n+1})), \quad j = 1, \dots, k \\
 erro_n &= \left\| u_{n+1}^{EU} - u_{n+1}^{RK} \right\|_{\infty} \\
 \Delta t_{n+1} &= obter\_novo\_ \Delta t_n(erro_n, tol, \Delta t_n)
 \end{aligned} \tag{3.22}$$

onde  $u_{n+1}^{jEU}$  e  $u_{n+1}^{jRK}$  são as soluções obtidas pelos métodos de Euler explícito e RK2, respectivamente;  $obter\_novo\_ \Delta t_n$  é a função utilizada para computar o novo passo de tempo;  $\Delta t_n$  é o passo de tempo para EDO;  $erro_n$  é a norma infinita (máximo) da diferença das soluções obtidas pelos dois métodos, calculada da seguinte maneira:

$$\begin{aligned}
 erro_n &= \left\| u_{n+1}^{EU} - u_{n+1}^{RK} \right\|_{\infty} \\
 &= \left\| \left( u_n^j \Delta t_n f^j(t_n, \vec{u}_n) \right) - \left( u_n^j \frac{\Delta t_n}{2} (f^j(t_n, \vec{u}_n) + f^j(t_{n+1}, \vec{u}_{n+1})) \right) \right\|_{\infty} \\
 &= \left\| \frac{\Delta t_n}{2} (f^j(t_n, \vec{u}_n) - f^j(t_{n+1}, \vec{u}_{n+1})) \right\|_{\infty}
 \end{aligned} \tag{3.23}$$

Para estimar o novo passo de tempo, foi considerado um  $erro_n$  local da ordem de  $\Delta t_n^2$ . Assim, o novo passo de tempo deve ser limitado pela tolerância ( $tol$ ):

$$erro_{n+1} = \alpha \Delta t_{n+1}^2 \leq tol \tag{3.24}$$

fazendo  $erro_{n+1} = tol$ :

$$\frac{erro_n}{erro_{n+1}} = \frac{\alpha \Delta t_n^2}{\alpha \Delta t_{n+1}^2} \Rightarrow \Delta t_{n+1} = \Delta t_n \sqrt{\frac{tol}{erro_n}} \tag{3.25}$$

Se o erro obtido fosse maior do que a tolerância preestabelecida, a solução era descartada e a iteração  $n$  era recalculada com o  $\Delta t_{n+1}$ . Caso contrário, o método obtinha o novo passo de tempo e seguia para a iteração  $n+1$ .

Para solução do passo de tempo adaptativo, a princípio é necessário calcular o lado direito duas vezes em cada iteração, sendo uma para o método de Euler no instante  $t_n$  e outra para o método RK2 para  $t_{n+1}$ , o que dobraria o custo computacional. Segundo Campos (2011), uma maneira de contornar este problema é reutilizar  $f(t_{n+1})$  da iteração  $n$  como  $f(t_n)$  na iteração seguinte  $n+1$ , de forma que o método modificado utiliza apenas uma avaliação do lado direito por iteração. O Algoritmo 1 mostra o pseudocódigo do tempo adaptativo.

---

**Algoritmo 1** Método adaptativo no tempo
 

---

```

1:  $t_0 = dt\_inicial$ ;
2:  $LD_1 = lado\_direito(t_0, \vec{y}_0)$ ;
3: while  $t_n \leq t_{final}$  do
4:    $\vec{y}_{n+1} = LD_1 * \Delta t_n + \vec{y}_n$ ;
5:    $LD_2 = lado\_direito(t_{n+1}, \vec{y}_{n+1})$ ;
6:    $tol_{n+1} = absTol$ ;
7:    $erro_{n+1} = \max(0.5 * ||LD_1 - LD_2||)$ ;
8:    $\Delta t_{n+1} = obtem\_novo\_Delta t(erro_{n+1}, tol_{n+1})$ ;
9:   if  $\Delta t_{n+1} > max\_dt$  then
10:      $\Delta t_{n+1} = max\_dt$ ;
11:   end if
12:   if  $erro_n < tol_{n+1}$  then
13:      $\vec{y}_n = \vec{y}_{n+1}$ ;
14:      $LD_1 = LD_2$ ;
15:      $t_n = t_n + \Delta t_n$ ;
16:   end if
17:    $\Delta t_n = \Delta t_{n+1}$ ;
18: end while

```

---

Na linha 2 do Algoritmo 1, o lado direito das equações é computado pela primeira vez e armazenado em  $LD_1$ . Na linha 3 o laço iterativo é iniciado. Na linha 4 é calculado o método de Euler. Na linha 5 é calculado o “novo” lado direito,  $LD_2$ , o qual é utilizado para computar o erro e na iteração seguinte é utilizado para calcular o método de Euler. Nas linhas 6 e 7 são determinados a tolerância ( $tol_{n+1}$ ) e o erro ( $erro_{n+1}$ ), respectivamente. Na linha 8 é determinado o novo passo de tempo ( $\Delta t_{n+1}$ ). Na linha 9 verifica-se se o erro é menor que um valor máximo do passo de tempo preestabelecido e na linha 12 verifica-se se o erro é aceitável. Se o erro for aceitável, realizam-se as trocas de ponteiro de vetores e a variável do tempo é incrementada, caso contrário, os vetores e a variável do tempo são descartados e computados novamente na próxima iteração. Por fim, o passo de tempo é atualizado, na linha 17, de  $\Delta t_n$  para  $\Delta t_{n+1}$ . Informações mais detalhadas do passo de tempo adaptativo podem ser encontradas em Campos *et al.* (2011), Campos (2011) e Oliveira (2013).

Resumindo, as Equações 3.3, 3.4 e 3.5 tomam a forma discreta mostrada nas Equações 3.26, 3.27 e 3.28. Sendo  $n$  o passo de tempo atual, \* um passo de tempo intermediário

e  $n+1$  o passo de tempo.

$$C_m \frac{V_{i,j,k}^{n+1} - V_{i,j,k}^*}{\Delta t_n} = -I_{ion}(V_{i,j,k}^*, \boldsymbol{\eta}^n) \quad (3.26)$$

$$\frac{\boldsymbol{\eta}_{i,j,k}^{n+1} - \boldsymbol{\eta}_{i,j,k}^*}{\Delta t_n} = f(\boldsymbol{\eta}^n, V^*, t) \quad (3.27)$$

$$\begin{aligned} C_m \frac{V_{i,j,k}^* - V_{i,j,k}^n}{\Delta t_p} &= - \frac{(S_1 J_1^* - S_2 J_2^* + S_3 J_3^* - S_4 J_4^* + S_5 J_5^* - S_6 J_6^*)}{\beta k_{i,j,k}^3} \Rightarrow \\ \alpha V_{i,j,k}^* - \sum_{c=1}^{m_1} \sigma_{x_{r',c}} (V_{r,c}^* - V_{i,j,k}^*) + \sum_{c=1}^{m_2} \sigma_{x_{l',c}} (V_{i,j,k}^* - V_{l,c}^*) \\ &- \sum_{c=1}^{m_3} \sigma_{y_{t',c}} (V_{t,c}^* - V_{i,j,k}^*) + \sum_{c=1}^{m_4} \sigma_{y_{b',c}} (V_{i,j,k}^* - V_{b,c}^*) \\ &- \sum_{c=1}^{m_5} \sigma_{z_{f',c}} (V_{f,c}^* - V_{i,j,k}^*) + \sum_{c=1}^{m_6} \sigma_{z_{bk',c}} (V_{i,j,k}^* - V_{bk,c}^*) = V_{i,j,k}^n \alpha \end{aligned} \quad (3.28)$$

O pseudocódigo para solução do conjunto de equações para o modelo monodomínio da eletrofisiologia cardíaca é mostrado no Algoritmo 2, sendo a variável de estado do sistema de EDOs de cada volume finito, representado por um vetor de estado (SV).

---

**Algoritmo 2** Solução do modelo monodomínio
 

---

- 1: configura no modelo celular as condições iniciais;
  - 2: monta a matriz do monodomínio (Sistema linear da EDP);
  - 3: **while**  $t < t_{final}$  **do**
  - 4: atualiza SV para o modelo celular;
  - 5: resolve o modelo celular (EDO);
  - 6: atualiza SV para EDP;
  - 7: resolve o sistema linear (EDP) via gradiente conjugado;
  - 8:  $t = t + dt$ ;
  - 9: **end while**
- 

### 3.3 Implementação em computação distribuída e multi-GPU

A implementação em computação distribuída e multi-GPU da solução das EDOs, as quais não existiam no simulador de eletrofisiologia cardíaca, foram feitas com base no Algoritmo 2, sendo a configuração e execução da solução das EDOs distribuídas em nós de processamento usando estratégias da seguinte forma:

1. Paradigma mestre-escravo, onde  $N_p$  é o número de processos, um núcleo de processamento (denominado mestre) ficou responsável por decompor o problema em

tarefas e enviar mensagens aos demais  $N_p-1$  núcleos de processamento (denominados escravos) distribuindo as tarefas. Por sua vez, o ciclo de execução dos escravos consistiu em: obter por passagem de mensagem uma tarefa do mestre, processar a tarefa e enviar o resultado de volta para o mestre. Por fim, o mestre recolhe os resultados parciais dos escravos de modo a reunir o resultado final.

2. Somente o nó de processamento mestre faz a configuração do modelo celular (linha 1) e a execução das funções de montagem da matriz do monodomínio (linha 2). É feita a atualização de SV (linha 6) para EDP e a solução da EDP pelo gradiente conjugado (linha 7). Somente as linhas 4 e 5, responsáveis por atualizar SV para as EDOs e resolver as EDOs, foram feitas de forma distribuída para todos os nós de processamento.
3. Para acelerar ainda mais a solução, foi feita uma implementação multi-GPU para as soluções das EDOs. A EDP também utilizou da GPU.

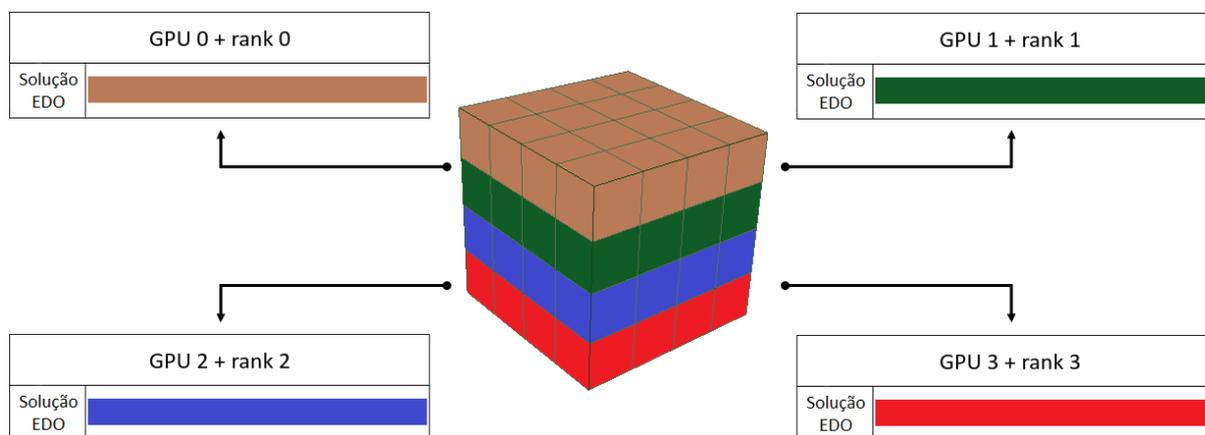
A estratégia 1 foi escolhida devido obtenção de bons desempenhos e um elevado grau de escalabilidade, e foi implementada em conjunto com a estratégia 2 e 3. Como não há dependência de dados entre as soluções dos sistemas de EDOs em cada volume finito, foi possível realizar a execução da solução de tais, em diferentes nós de forma direta.

O que causa a alteração nas variáveis de estado em cada passo de tempo é o estímulo do PA que atravessa o tecido excitável. O estímulo de cada volume finito foi representado por um vetor, *stim*, e a variável de estado do sistema de EDOs de cada volume finito foi representado por um vetor de estado (SV). Para a implementação da estratégia 2, em cada passo de tempo, o nó mestre atualizava o estímulo e enviava a cada nó escravo o estímulo referente a parte do domínio em que o nó era responsável por resolver. A seguir, os nós escravos resolviam as EDOs e enviavam ao nó mestre a solução referente à sua parte do domínio. O algoritmo seguia sua execução, e após a solução da EDP (feita em GPU e sem computação distribuída) as variáveis de estado eram atualizadas pelo nó mestre. O passo de tempo é incrementado e todo o processo do laço se repete.

Para implementação da estratégia 3, onde foi proposta a utilização de GPUs para solução das EDOs em conjunto com a computação distribuída, foi necessária a comunicação da CPU com a GPU para atualização das variáveis de estado, linha 4 do Algoritmo 2. Além disso, os resultados da solução do modelo celular (linha 5 do Algoritmo 2), foram necessários para solução da EDP (linha 7 do Algoritmo 2), o que também necessitou de comunicação da GPU para a CPU. Antes da execução da solução das EDOs em multi-GPU, o algoritmo distribui os nós de processamento entre as GPUs utilizadas, de forma que cada nó executa seus cálculos em uma determinada GPU, dessa forma, o número de processos é igual ao número de GPUs. A Figura 11 exemplifica por meio de um domínio

cúbico, a estratégia proposta de computação distribuída e multi-GPU, sendo rank o ID de cada processo criado.

**Figura 11 – Exemplo da implementação proposta da computação em memória distribuída multi-GPU para solução das EDOs**



Fonte: Elaborado pelo autor

A implementação paralela e em GPU foi realizada pela interface de programação de aplicações (*Application Programming Interface* - API) CUDA. A distribuição de tarefas e comunicação entre os nós foi realizada pelo padrão de interface para a troca de mensagens em máquinas paralelas com memória distribuída, MPI. Devido a limitações físicas, foram utilizadas somente duas GPUs nos testes, porém o algoritmo desenvolvido é genérico e permite a utilização de um maior número de GPUs. Conseqüentemente, devido a essas limitações, foram realizados testes com 1 e 2 processos.

Matematicamente, o fator de aceleração  $a$ , corresponde à razão entre o tempo gasto na computação de um dado problema utilizando-se de apenas um processador,  $t_s$ , e o tempo gasto por um programa que utiliza  $p$  processadores de forma paralela,  $t_p$ :

$$a(p) = \frac{t_s}{t_p} \quad (3.29)$$

### 3.3.1 Especificações do ambiente de execução

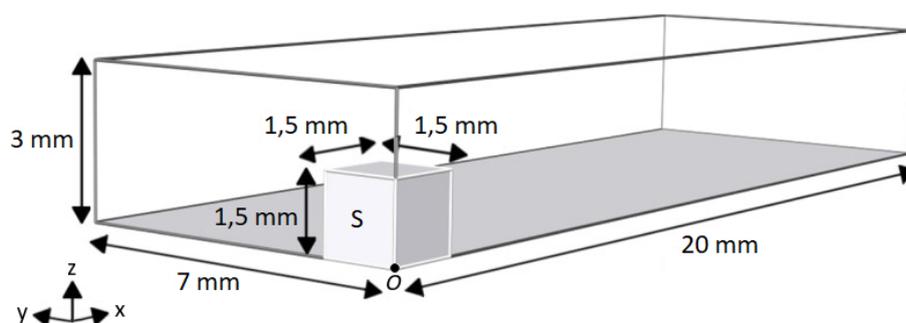
Os testes foram executados no sistema operacional Manjaro Linux, kernel 5.10 em *hardware* composto de processador Intel<sup>®</sup> Core<sup>™</sup> i7-6700K com frequência de *clock* de 4,00 GHz, 32 GB de memória RAM e 2 TB de disco rígido. Foram utilizadas três placas de vídeo, sendo duas NVIDIA GeForce GTX 960 e uma NVIDIA TITAN Xp. Para a implementação multi-GPU, primeiramente foram realizados testes em um ambiente homogêneo com duas placas de vídeo NVIDIA GeForce GTX 960 e, posteriormente, uma placa de vídeo NVIDIA GeForce GTX 960 foi combinada com a placa de vídeo NVIDIA TITAN Xp.

As placas de vídeo utilizadas foram as disponíveis no laboratório de elaboração dessa pesquisa, sendo importante utilizá-las da melhor forma possível, dando mais sobrevida aos recursos disponíveis.

### 3.4 Domínio e modelo celular

O domínio adotado para os testes da implementação aqui desenvolvida, é um *benchmark* proposto por (NIEDERER *et al.*, 2011), o qual pode ser visualizado na Figura 12.

Figura 12 – Domínio *benchmark* adotado nos testes. S: local de estímulo do potencial de ação



Fonte: Adaptado de (NIEDERER *et al.*, 2011)

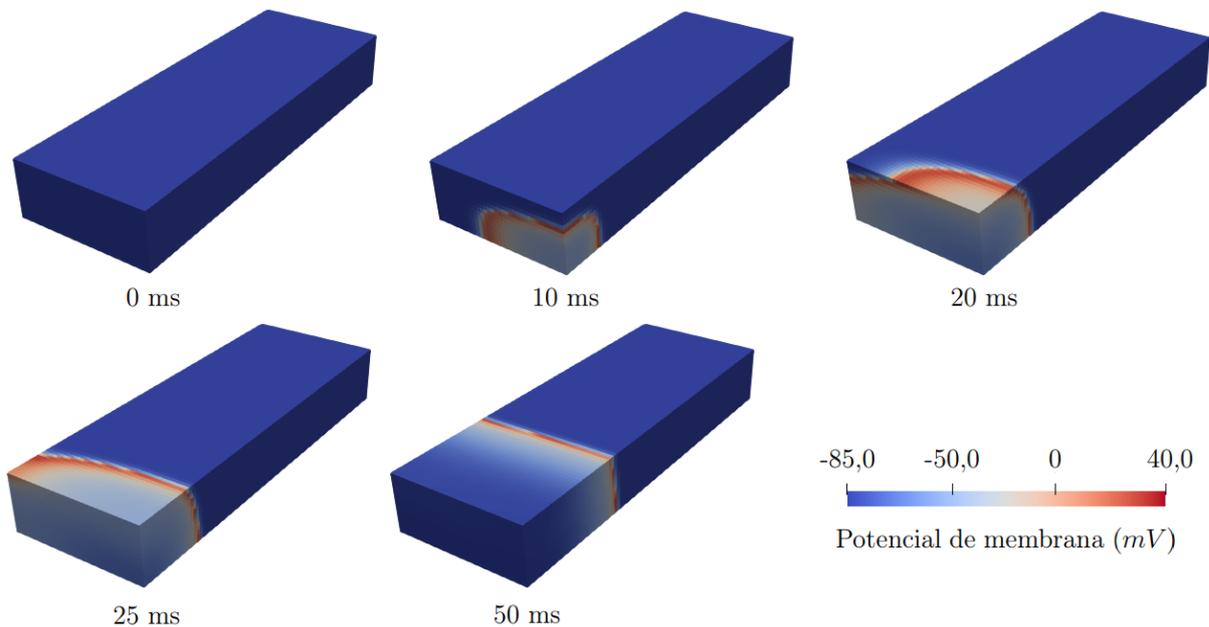
O *benchmark* proposto por Niederer *et al.* (2011) adota o modelo celular TT2, porém nesse trabalho, o *benchmark* foi modificado, de forma a adotar o modelo celular Bondarenko, pelo fato de além de ser um modelo celular muito utilizado, ser um modelo complexo, o qual exige mais poder computacional para ser resolvido e o uso de múltiplas GPUs é também justificável. Nesse modelo, o tecido é considerado anisotrópico e as condutividades são:  $\sigma_x = 0,0176$  S/m,  $\sigma_y = 0,1334$  S/m e  $\sigma_z = 0,0176$  S/m; a capacitância da membrana é  $C_m = 1$   $\mu\text{F}/\text{cm}^2$ . O protocolo de estímulo externo compreende uma corrente de  $-50.000$   $\mu\text{A}/\text{cm}^3$  aplicado durante 2 ms (NIEDERER *et al.*, 2011). O domínio foi discretizado espacialmente em elementos hexaédricos regulares de lado igual a  $500$   $\mu\text{m}$ ,  $400$   $\mu\text{m}$ ,  $250$   $\mu\text{m}$ ,  $200$   $\mu\text{m}$ ,  $125$   $\mu\text{m}$  e  $100$   $\mu\text{m}$ . O tempo de simulação adotado foi de 50 ms com passo de tempo das EDOs de  $0,0001$  ms e da EDP de  $0,01$  ms com tolerância de  $1 \times 10^{-15}$ .

## 4 RESULTADOS E DISCUSSÃO

A discretização do domínio *benchmark* em volumes finitos de comprimentos característicos de  $500 \mu m$ ,  $400 \mu m$ ,  $250 \mu m$ ,  $200 \mu m$ ,  $125 \mu m$  e  $100 \mu m$ , resultou em uma quantidade de volumes iguais a 3.360, 5.950, 26.880, 52.500, 215.040 e 420.000, respectivamente.

O resultado eletrofisiológico pode ser verificado na Figura 13, a qual mostra instantes da despolarização e repolarização do domínio *benchmark* ao longo da simulação.

**Figura 13 – Despolarização do domínio *benchmark***



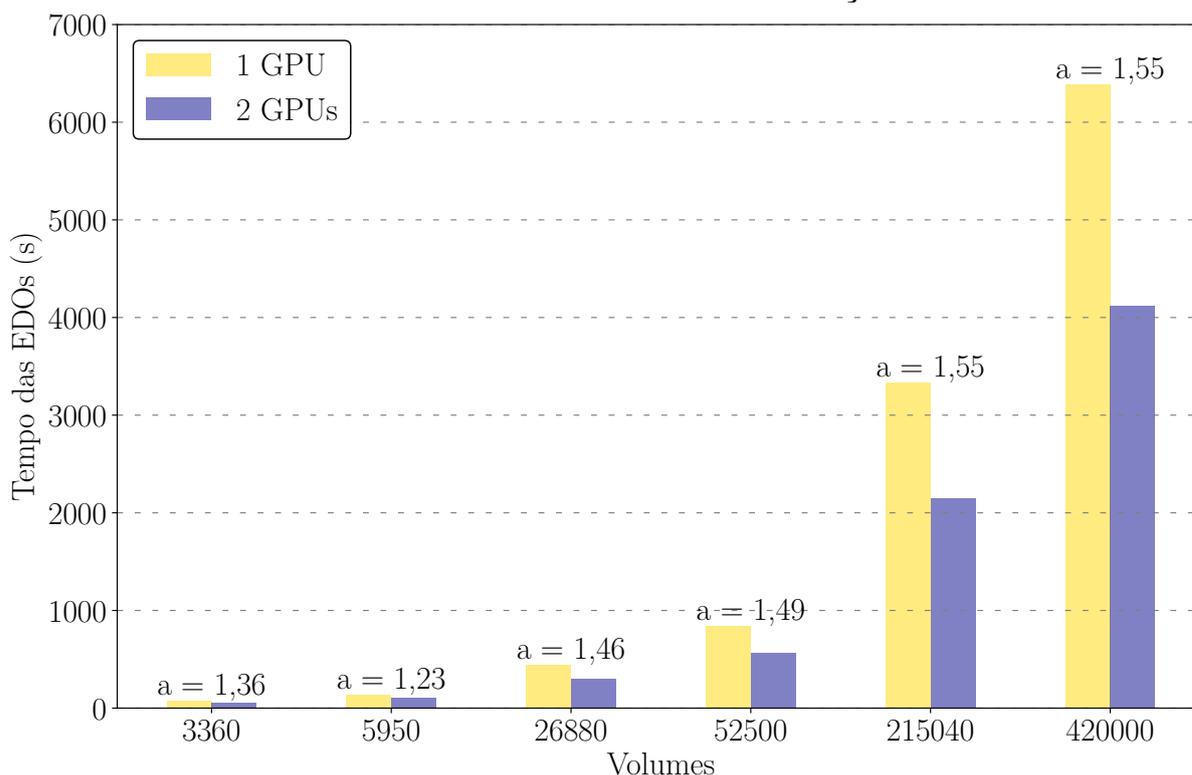
**Fonte: Elaborado pelo autor**

No instante de 0 ms o tecido encontrava-se totalmente em estado de repouso (aproximadamente -82 mV). No instante 10 ms já é possível observar a despolarização próximo à região de estímulo. O potencial de membrana foi elevado até aproximadamente 25 mV. A onda de despolarização seguiu por todo tecido, durante os demais instantes de tempo, além disso, é possível observar também a repolarização do tecido pós platô do estímulo durante a passagem do mesmo. Em instantes de tempo mais avançados, como o de 50 ms, é possível notar novamente partes do tecido já totalmente repolarizado, em estado de repouso. Observou-se também que a despolarização e repolarização foram mais rápida na direção *y*, devido a anisotropia do tecido cardíaco, onde a condutividade era maior nessa direção.

O primeiro teste realizado correspondeu a computação distribuída em multi-GPU utilizando as duas placas de vídeo NVIDIA GeForce GTX 960, de forma que ora foi criado um processo quando utilizado uma GPU, e ora dois processos quando utilizado

duas GPU, sendo cada processo executado em uma das placas. Obviamente, a linha de base é NVIDIA GeForce GTX 960. A Figura 14 mostra os resultados comparativos do tempo de execução das EDOs em uma e duas GPUs para o tempo de simulação de 50 ms, mostrando também a aceleração ( $a$ ) quando houve adoção de 2 GPUs.

**Figura 14 – Tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando duas placas de vídeo NVIDIA GeForce GTX 960.  $a$ : aceleração**



Fonte: Elaborado pelo autor

Os resultados apresentados na Figura 14 são referentes às médias de cinco execuções para cada caso. Esses valores médios, bem como o desvio padrão podem ser conferidos na Tabela 1, que mostra que em todos os casos o desvio padrão foi baixo em relação ao tempo de execução da EDOs, ou seja, não houve dispersão dos resultados em relação ao valor médio.

Com base na Figura 14, nota-se que o acréscimo de uma segunda GPU com as mesmas configurações não reduz pela metade o tempo de solução das EDOs, ou seja, não há uma aceleração ( $a$ ) igual a 2 devido ao tempo de comunicação entre CPU e GPU, e também GPU com CPU. É possível notar também que a aceleração tendeu a ser maior nos casos onde houveram maiores quantidades de volumes no domínio, o que mostra a relevância da proposta dessa pesquisa, permitindo a simulação de grandes domínios e malhas refinadas. Além disso, como o tempo de solução é acelerado, o modelo permite a adoção de um menor passo de tempo.

Com base na Figura 14 e Tabela 1 observa-se que de fato a utilização de multi-

GPUs acelerou o tempo de solução das EDOs e, conseqüentemente, reduziu o tempo global da solução do problema, como mostra a Tabela 2.

**Tabela 1 – Média e desvio padrão de cinco testes do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando duas placas de vídeo NVIDIA GeForce GTX 960. Linha de base: NVIDIA GeForce GTX 960**

Volumes	Quantidade de GPUs	Média (s)	Desvio padrão (s)
3.360	1	78,11	0,01
	2	57,54	0,02
5.950	1	135,85	0,08
	2	110,05	0,06
26.880	1	438,77	0,64
	2	300,57	0,55
52.500	1	844,59	0,28
	2	566,25	0,39
215.040	1	3331,75	0,61
	2	2145,62	0,21
420.000	1	6390,14	0,83
	2	4118,18	0,79

Fonte: Elaborado pelo autor

**Tabela 2 – Média e desvio padrão de cinco testes do tempo de execução global do algoritmo em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando duas placas de vídeo NVIDIA GeForce GTX 960. Linha de base: NVIDIA GeForce GTX 960**

Volumes	Quantidade de GPUs	Média (s)	Desvio padrão (s)
3.360	1	92,67	0,03
	2	72,56	0,10
5.950	1	152,06	0,09
	2	126,88	0,18
26.880	1	499,65	2,21
	2	367,78	1,91
52.500	1	1001,46	0,81
	2	739,61	2,29
215.040	1	4212,97	0,64
	2	3035,38	0,30
420.000	1	8163,59	0,81
	2	5905,78	0,72

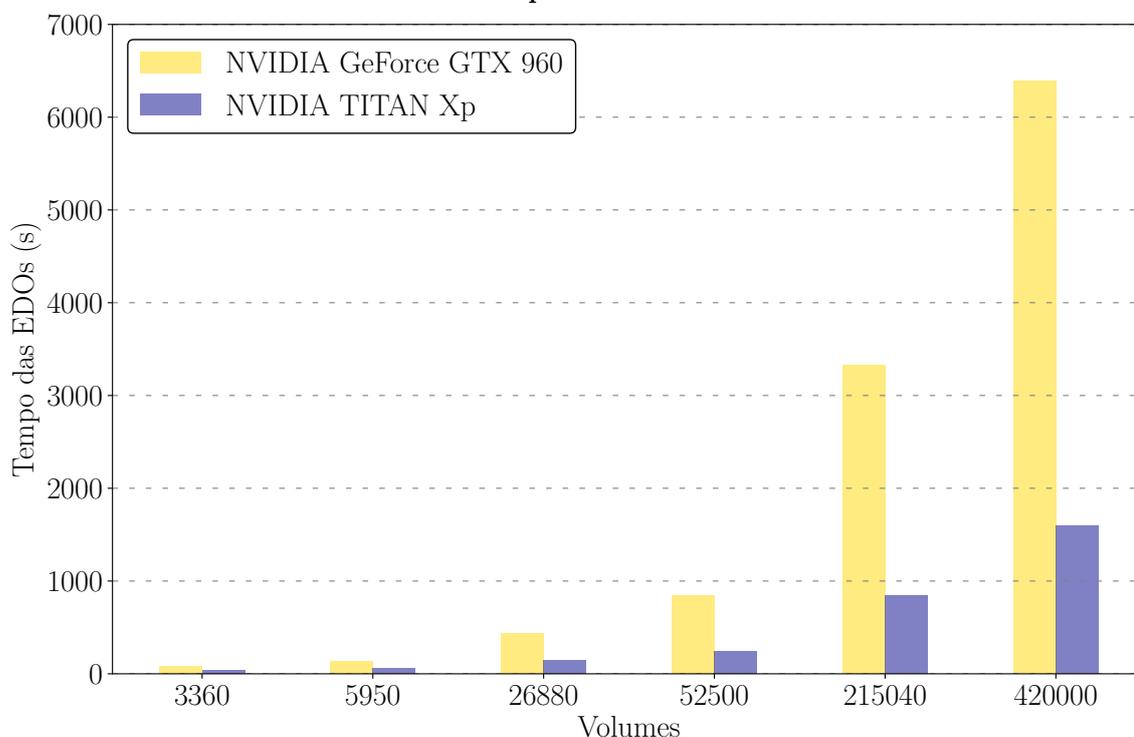
Fonte: Elaborado pelo autor

A adoção de duas placas NVIDIA GeForce GTX 960, mostra que mesmo com *hardwares* de menor desempenho é possível obter aceleração significativa nas simulações de eletrofisiologia cardíaca com modelos celulares complexos. Esse fato é mais um fator que justifica a implementação paralela proposta nessa pesquisa, se tornando relevante para parques tecnológicos em ambientes de simulação menos avançados ou já defasados

por questões financeiras, por exemplo, uma vez que placas de vídeo de última geração, com maior desempenho, necessitam de grande investimento.

Comparando o desempenho individual das placas de vídeo NVIDIA TITAN Xp e NVIDIA GeForce GTX 960, ou seja, quando utilizadas isoladamente em um único processo, como mostra a Figura 15, é notória a superioridade da NVIDIA TITAN Xp. Para os casos de 420.000, 215.040 e 52.500 volumes, por exemplo, a placa NVIDIA TITAN Xp solucionou as EDOs em 25,04%, 25,40% e 29,14%, respectivamente, do tempo de solução da placa NVIDIA GeForce GTX 960. Mais que isso, a execução de um processo na NVIDIA TITAN Xp resultou em tempos de solução das EDOs menores do que dois processos em duas placas NVIDIA GeForce GTX 960 (Tabela 1).

**Figura 15 – Comparação do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio utilizando as placas NVIDIA GeForce GTX 960 e NVIDIA TITAN Xp individualmente**



Fonte: Elaborado pelo autor

Um teste foi repetido com as mesmas configurações de duas placas NVIDIA GeForce GTX 960, porém uma das placas de vídeo foi substituída pela NVIDIA TITAN Xp, sendo esta a linha de base. A Tabela 3 mostra que diferentemente do esperado, ao acrescentar uma segunda GPU, o tempo de solução das EDOs aumentou consideravelmente. Tal fato é justificado pela diferença de desempenho entre as duas placas de vídeo, citado anteriormente. Como cada GPU ficou responsável por metade das EDOs (vide Figura 11) o processo executado na placa NVIDIA TITAN Xp terminava suas tarefas e ficava ociosa esperando a finalização das tarefas do processo executado na placa NVIDIA GeForce GTX 960. Isso pode ser comprovado ao se observar que o tempo de execução das duas placas

NVIDIA GeForce GTX 960 (Tabela 1), onde cada uma executava metade das tarefas, é próximo ao tempo de execução desse caso com duas placas diferentes.

**Tabela 3 – Média e desvio padrão de cinco testes do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando uma placa de vídeo NVIDIA TITAN Xp e uma NVIDIA GeForce GTX 960. Linha de base: NVIDIA TITAN Xp**

Volumes	Quantidade de GPUs	Média (s)	Desvio padrão (s)
3.360	1	32,76	0,13
	2	57,51	0,03
5.950	1	59,80	0,08
	2	109,92	0,05
26.880	1	140,77	0,24
	2	299,74	0,41
52.500	1	246,14	0,60
	2	568,77	0,66
215.040	1	846,36	1,69
	2	2143,94	0,79
420.000	1	1600,47	2,08
	2	4117,66	0,89

Fonte: Elaborado pelo autor

Observando a Tabela 3 e sabendo da discrepância de poder computacional das placas NVIDIA TITAN Xp e NVIDIA GeForce GTX 960 identificada anteriormente, como forma de otimizar ainda mais o tempo de execução, as placas de vídeo NVIDIA GeForce GTX 960 e NVIDIA TITAN Xp foram submetidas a um balanceamento de carga, sendo a NVIDIA TITAN Xp linha de base, e distribuindo de forma equilibrada e eficiente as diferentes partes da computação, de modo que os processadores fiquem ocupados durante a maior parte da execução. Os resultados podem ser visualizados na Tabela 4, onde, inicialmente, ao acrescentar mais uma GPU na solução havia desaceleração, passou a ter aceleração após o balanceamento, gerando um ganho maior que 2 no tempo de solução.

**Tabela 4 – Análise do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio em computação distribuída e multi-GPU utilizando uma placa de vídeo NVIDIA GeForce GTX 960 e uma NVIDIA TITAN Xp com balanceamento de carga. Amostra de cinco testes. Aceleração em relação à utilização de um processo com a placa de vídeo NVIDIA TITAN Xp FB: Fator de balanceamento**

Volumes	Média (s)	Desvio padrão (s)	Aceleração	FB
52.500	236,26	0,60	1,04	8,86
215.040	796,56	0,74	1,06	9,97
420.000	1487,83	0,89	1,08	9,98

Fonte: Elaborado pelo autor

Observou-se na prática que o fator ótimo de balanceamento de carga, ou seja, a porcentagem de processamento atribuída a cada GPU, era variável de acordo com o nível de discretização do domínio. Para essa análise da Tabela 4, só foram pós-processados

os valores referentes às malhas mais refinadas, pois são as de maior interesse, onde o tempo de execução é maior. Além disso, verificou-se na prática também que a aceleração tendeu a diminuir em malhas menos refinadas, pois como são menos elementos, o tempo de comunicação entre processos e GPUs se tornou mais expressivo do que o cálculo das EDOs propriamente dito. Nesses casos, a utilização de uma única GPU com melhor desempenho, como a NVIDIA TITAN Xp, mostrou-se mais vantajoso.

A Tabela 5 mostra o tempo de solução das EDOs em relação ao tempo de solução total do algoritmo, sendo possível notar que de fato as EDOs consomem bastante tempo da execução do algoritmo. Utilizando duas placas de vídeo NVIDIA GeForce GTX 960, a adição da segunda placa reduz ligeiramente essa relação dos tempos de solução e, à medida que a malha vai sendo refinada, essa relação vai se tornando mais significativa, ou seja, quanto mais elementos, menor o tempo percentual de solução das EDOs em relação ao tempo global. Esse fato é importante, uma vez que a implementação em paralelização distribuída e multi-GPU proposta nessa pesquisa tem como alvo malhas refinadas e modelos celulares complexos.

**Tabela 5 – Tempo de execução das EDOs em relação ao tempo de execução global do algoritmo de todos os testes realizados**

<b>Volumes</b>	<b>Nº GPUs</b>	<b>GTX 960 e GTX 960 (%)</b>	<b>TITAN Xp e GTX 960 (%)</b>	<b>TITAN Xp e GTX 960 com balanceamento (%)</b>
3.360	1	84,29	71,79	-
	2	79,30	80,83	-
5.950	1	89,34	80,59	-
	2	86,73	86,92	-
26.880	1	87,82	73,69	-
	2	81,72	83,87	-
52.500	1	84,34	62,83	-
	2	76,56	77,62	60,34
215.040	1	79,08	50,84	-
	2	70,69	72,16	49,05
420.000	1	78,28	49,61	-
	2	69,73	71,53	47,65

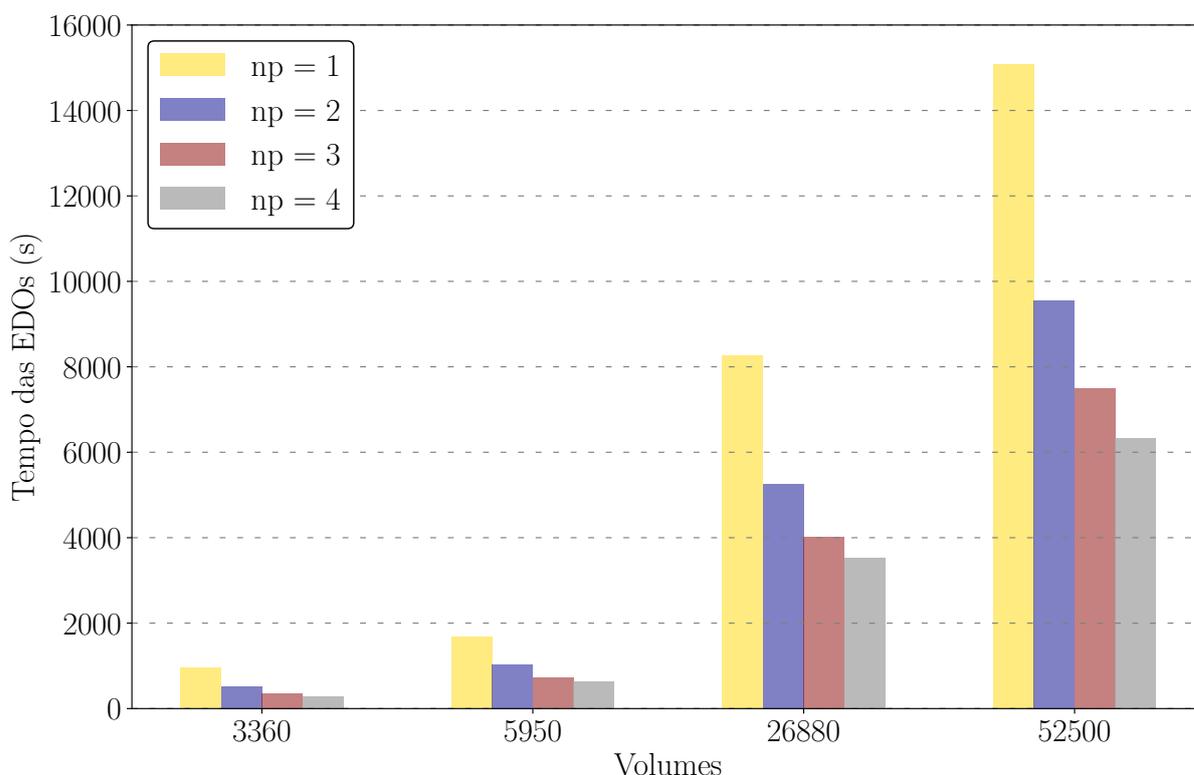
**Fonte: Elaborado pelo autor**

Por outro lado, ainda de acordo com a Tabela 5, com a disposição das duas placas de vídeo de diferentes desempenhos sem balanceamento de carga, fica claro que quando a NVIDIA TITAN Xp opera isoladamente, o tempo de solução das EDOs em relação ao tempo global também é reduzido, mais ainda do que no caso da utilização de duas NVIDIA GeForce GTX 960, sendo mais significativo também para malhas mais refinadas. Por fim, quando o balanceamento de carga foi realizado, observou-se que o tempo da solução das EDOs em relação ao tempo global forneceu os melhores resultados em termos de tempos absolutos, sempre reduzindo essa relação de tempo. Assim, fica demonstrado que

a implementação proposta nessa pesquisa foi de fato capaz de reduzir esse gargalo do tempo de solução das EDOs.

Também foram realizados testes levando em conta apenas a paralelização distribuída em CPUs, sem utilização de GPUs. O gráfico da Figura 16 mostra os resultados do tempo de execução das EDOs para esse caso, onde só foram testadas as discretizações de  $500 \mu m$ ,  $400 \mu m$ ,  $250 \mu m$  e  $200 \mu m$ .

**Figura 16** – Comparação do tempo de execução das EDOs em relação a diferentes níveis de discretização do domínio utilizando apenas paralelização distribuída em CPUs. np: número de processos



Fonte: Elaborado pelo autor

Embora o tempo de solução das EDOs tenha aumentado sem a presença de GPUs, a paralelização em memória distribuída se torna uma ferramenta interessante em ambientes com pouco recurso computacional, onde não é possível a disponibilidade de placas de vídeo. Como mostra a Figura 16, a adição de processos reduziu consideravelmente o tempo de solução das EDOs, de forma que quanto mais processos foram adicionados, maior foi a aceleração, conseqüentemente, menor a relação do tempo de solução das EDOs e do tempo de solução global. A Tabela 6 complementa os resultados da Figura 16, mostrando ainda o tempo global de execução do algoritmo, o tempo de solução das EDOs em relação ao tempo de solução global e a aceleração para cada caso.

Tabela 6 – Tempo de execução da algoritmo utilizando apenas paralelização distribuída em CPUs

Volumes	Nº processos	Tempo EDOs (s)	Tempo global (s)	Relação tempos (%)	Aceleração
3.360	1	955,74	1007,23	94,89	-
	2	514,22	572,70	89,79	1,86
	3	357,68	418,30	85,51	2,67
	4	292,13	356,46	81,95	3,27
5.950	1	1676,19	1729,06	96,94	-
	2	1040,28	1124,74	92,49	1,61
	3	715,58	803,84	89,02	2,34
	4	630,97	725,10	87,02	2,66
26.880	1	8272,85	10419,77	96,94	-
	2	5256,81	7558,03	92,49	1,58
	3	4020,08	4577,04	89,02	2,01
	4	3526,97	4101,57	87,02	2,39
52.500	1	15090,12	16704,74	90,33	-
	2	9548,53	15967,09	59,80	1,57
	3	7499,68	14011,45	53,53	2,06
	4	6317,01	12923,72	48,88	2,35

Fonte: Elaborado pelo autor

Em relação a Tabela 6, nota-se ainda que quanto mais refinadas foram as malhas em conjunto com a adição de processos, mais o tempo de solução das EDOs em relação ao tempo de solução global é reduzido.

## 5 CONCLUSÃO E TRABALHOS FUTUROS

### 5.1 Conclusão

Com a simulação de modelos celulares complexos aliados à domínios tridimensionais realísticos que requerem malhas muito refinadas, torna-se necessária a evolução dos algoritmos de simulação com o auxílio da computação de alto desempenho, como a utilização de processamento paralelo e distribuído em GPGPU, visando reduzir o tempo global. Implementações desse tipo podem fornecer detalhes da fisiopatologia cardíaca, como, a remodelação pós infarto, a análise das consequências causadas pelas cicatrizes formadas devido ao infarto, a propagação da frente de onda do potencial de ação em espiral, estudo de arritmias, entre outros.

A paralelização em memória distribuída e multi-GPU, para solução das EDOs em simulações da eletrofisiologia cardíaca, proposta nessa pesquisa, se mostrou eficiente no ponto de vista do desempenho computacional e utilização de memória. A estratégia multi-GPU para duas placas NVIDIA GeForce GTX 960 resultou em valores de aceleração próximos a 1,5 quando da utilização de 2 GPUs em relação a 1 GPU nos casos onde foi exigido maior poder computacional, ou seja, em malhas mais refinadas.

Utilizando dos recursos que se tem disponível, ou seja, a combinação da placa de melhor desempenho com a de desempenho inferior, foi comprovado a possibilidade de um desbalanceamento de carga devido ao ambiente não homogêneo, podendo ser resolvido com um simples balanceamento estático, gerando um ganho de mais de 2,4 vezes no tempo de solução do que quando está desbalanceado.

Tal balanceamento de carga foi o que gerou os menores tempos de solução das EDOs. Por outro lado, para malhas menos refinadas, foi mostrado que a placa de vídeo com melhor desempenho possibilitou um tempo menor de solução do que a utilização de duas placas de menor desempenho, mesmo após balanceamento de carga, pois, com poucos elementos, a comunicação entre CPUs e GPUs passou a ser mais significativo do que a solução das EDOs propriamente dito, ou seja, a troca de mensagens e as diversas cópias realizadas entre uma memória e outra fez com que o tempo de resposta fosse aumentado, causando impacto no desempenho geral nessas malhas.

Por fim, a escalabilidade do algoritmo proposto, permite a adição CPUs e GPUs, por meio de arquiteturas de *clusters*, de forma a acelerar ainda mais as simulações eletrofisiológicas. Dessa forma, a distribuição de tarefas em nós permite realizar uma quantidade maior de simulações do que a possível atualmente pelo estado da arte para um determinado intervalo de tempo. Além disso, a paralelização também permite simulações que utilizem mais memória, uma vez que as EDOs são distribuídas entre as GPUs. Assim, a

implementação multi-GPU aqui proposta torna-se uma ferramenta importante no estudo numérico de problemas complexos e realísticos de eletrofisiologia cardíaca, reduzindo o tempo global.

## 5.2 Trabalhos futuros

Ao longo da pesquisa, foram detectados alguns pontos para aprofundamento e continuidade do trabalho, os quais são sugeridos como trabalhos futuros:

- Realizar o balanceamento de carga entre as diferentes placas de vídeo em tempo de execução, ou seja, de forma dinâmica.
- Paralelizar a solução da EDP em memória distribuída e multi-GPU.
- Implementar e avaliar a paralelização híbrida.

## REFERÊNCIAS

- ALONSO, S.; SANTOS, R. W. dos; BÄR, M. Reentry and ectopic pacemakers emerge in a three-dimensional model for a slab of cardiac tissue with diffuse microfibrosis near the percolation threshold. *PloS one*, Public Library of Science San Francisco, CA USA, v. 11, n. 11, p. e0166972, 2016.
- ANDRADE, J. G. *et al.* *The Clinical Cardiac Electrophysiology Handbook*. 1. ed. Minneapolis: Cardiotext Publishing, 2016.
- AZEVEDO, P. S. *et al.* Cardiac remodeling: concepts, clinical impact, pathophysiological mechanisms and pharmacologic treatment. *Arquivos brasileiros de cardiologia*, SciELO Brasil, v. 106, n. 1, p. 62–69, 2016.
- BONDARENKO, V. E. *et al.* Computer model of action potential of mouse ventricular myocytes. *American Journal of Physiology-Heart and Circulatory Physiology*, American Physiological Society, v. 287, n. 3, p. H1378–H1403, 2004.
- BOURGAULT, Y.; PIERRE, C. Comparing the bidomain and monodomain models in electro-cardiology through convergence analysis. 2010.
- BURGARELLI, D.; KISCHINHEVSKY, M.; BIEZUNER, R. J. A new adaptive mesh refinement strategy for numerically solving evolutionary pde's. *Journal of Computational and Applied Mathematics*, Elsevier, v. 196, n. 1, p. 115–131, 2006.
- BURGARELLI, D. D. *Modelagem Computacional e Simulação Numérica Adaptativa de Equações Diferenciais Parciais Evolutivas Aplicadas a um Problema Termoacustico*. Dissertação (Mestrado) — PUC - Rio de Janeiro, 1998.
- BUTCHER, J. C. *Numerical methods for ordinary differential equations*. Chichester: John Wiley & Sons, 2016.
- CAMPOS, F. O. *et al.* Influence of ischemic core muscle fibers on surface depolarization potentials in superfused cardiac tissue preparations: a simulation study. *Medical & biological engineering & computing*, Springer, v. 50, n. 5, p. 461–472, 2012.
- CAMPOS, R. S. *Ferramentas web para descrição e simulação de modelos de células cardíacas*. Dissertação (Mestrado) — Universidade Federal de Juiz de Fora, 2011.
- CAMPOS, R. S. *et al.* Comparing high performance techniques for the automatic generation of efficient solvers of cardiac cell models. *Computing*, Springer, v. 95, n. 1, p. 639–660, 2013.
- CAMPOS, R. S.; LOBOSCO, M.; SANTOS, R. W. dos. Adaptive time step for cardiac myocyte models. *Procedia Computer Science*, Elsevier, v. 4, p. 1092–1100, 2011.
- CHIONG, M. *et al.* Cardiomyocyte death: mechanisms and translational implications. *Cell death & disease*, Nature Publishing Group, v. 2, n. 12, p. e244–e244, 2011.
- COHN, J. N. *et al.* Cardiac remodeling—concepts and clinical implications: a consensus paper from an international forum on cardiac remodeling. *Journal of the American College of Cardiology*, Journal of the American College of Cardiology, v. 35, n. 3, p. 569–582, 2000.

- COSTABAL, F. S. *et al.* Physics-informed neural networks for cardiac activation mapping. *Frontiers in Physics*, Frontiers Media SA, v. 8, p. 42, 2020.
- DEHAL, R. S. *et al.* Gpu computing revolution: Cuda. In: IEEE. *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*. Greater Noida, 2018. p. 197–201.
- FOX, S. *Human Physiology*. 15. ed. New York: Human Physiology, 2019.
- GILLETTE, K. *et al.* A framework for the generation of digital twins of cardiac electrophysiology from clinical 12-leads ecgs. *Medical Image Analysis*, Elsevier, v. 71, p. 102080, 2021.
- GLOVER, B. M.; BRUGADA, P. *Clinical Handbook of Cardiac Electrophysiology*. 1. ed. Cham: Springer, 2016.
- HALL, J. E. *Guyton & Hall - Tratado de fisiologia médica*. 13. ed. Rio de Janeiro: GEN Guanabara Koogan, 2017.
- HARRILD, D. M.; HENRIQUEZ, C. S. A finite volume model of cardiac propagation. *Annals of biomedical engineering*, Springer, v. 25, n. 2, p. 315–334, 1997.
- HEGYI, B. *et al.* Complex electrophysiological remodeling in postinfarction ischemic heart failure. *Proceedings of the National Academy of Sciences*, National Acad Sciences, v. 115, n. 13, p. E3036–E3044, 2018.
- HESTENES, M. R.; STIEFEL, E. *et al.* Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, v. 49, n. 6, p. 409–436, 1952.
- HILLE, B. *Ion channels of excitable membranes*. Sunderland: Sinauer Associates, Inc., 2001.
- HODGKIN, A. L.; HUXLEY, A. F. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, Wiley Online Library, v. 117, n. 4, p. 500–544, 1952.
- ISSA, Z. F.; MILLER, J. M.; ZIPES, D. P. *Clinical Arrhythmology and Electrophysiology: A Companion to Braunwald's Heart Disease*. 3. ed. Philadelphia: Elsevier, 2018.
- JAMES, S. L. *et al.* Global, regional, and national incidence, prevalence, and years lived with disability for 354 diseases and injuries for 195 countries and territories, 1990–2017: a systematic analysis for the global burden of disease study 2017. *The Lancet*, Elsevier, v. 392, n. 10159, p. 1789–1858, 2018.
- JØRGENSEN, E. B. *Benchmarking and Optimization of Cardiac Electrophysiology Solvers*. Dissertação (Mestrado) — University of Oslo, 2021.
- KARMAŽÍNOVÁ, M.; LACINOVÁ, L. Measurement of cellular excitability by whole cell patch clamp. *Physiol. Res*, Citeseer, v. 59, n. 1, p. S1–S7, 2010.
- KEENER, J. P.; SNEYD, J. *Mathematical physiology. 2: Systems physiology James Keener, James Sneyd*. 2. ed. New York: Springer, 2009.

- LEENKNEGT, L. *et al.* Insights into the intracardiac electrogram from analytical and numerical modelling. *Europace*, Oxford University Press, v. 24, n. Supplement\_1, p. euac053–567, 2022.
- LI, A. *et al.* Evaluating modern gpu interconnect: Pcie, nvlink, nv-sli, nvswitch and gpudirect. *IEEE Transactions on Parallel and Distributed Systems*, IEEE, v. 31, n. 1, p. 94–110, 2019.
- LI, Y.; ZHANG, Z. Parallel computing: review and perspective. In: IEEE. *2018 5th International Conference on Information Science and Control Engineering (ICISCE)*. Zhengzhou, 2018. p. 365–369.
- LUO, C.-h.; RUDY, Y. A model of the ventricular cardiac action potential. depolarization, repolarization, and their interaction. *Circulation research*, Am Heart Assoc, v. 68, n. 6, p. 1501–1526, 1991.
- MARKOV, A. A. Extension of the limit theorems of probability theory to a sum of variables connected in a chain. *Dynamic probabilistic systems*, Wiley New York, v. 1, p. 552–577, 1971.
- MENA, A.; FERRERO, J. M.; MATAS, J. F. R. Gpu accelerated solver for nonlinear reaction–diffusion systems. application to the electrophysiology problem. *Computer Physics Communications*, Elsevier, v. 196, p. 280–289, 2015.
- MILLER, W. T.; GESELOWITZ, D. B. Simulation studies of the electrocardiogram. i. the normal heart. *Circulation Research*, Am Heart Assoc, v. 43, n. 2, p. 301–315, 1978.
- MORENO, A. *et al.* Low-energy, single-pulse surface stimulation defibrillates large mammalian ventricles. *Heart Rhythm*, Elsevier, v. 19, n. 2, p. 308–317, 2022.
- NIEDERER, S. *et al.* Creation and application of virtual patient cohorts of heart models. *Philosophical Transactions of the Royal Society A*, The Royal Society Publishing, v. 378, n. 2173, p. 20190558, 2020.
- NIEDERER, S. A. *et al.* Verification of cardiac tissue electrophysiology simulators using an n-version benchmark. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, The Royal Society Publishing, v. 369, n. 1954, p. 4331–4351, 2011.
- OLIVEIRA, R. S. *Algoritmos paralelos e adaptativos no tempo e no espaço para simulação numérica da eletrofisiologia do coração*. Tese (Doutorado) — Universidade Federal de Minas Gerais, 2013.
- OLIVEIRA, R. S. *et al.* Ectopic beats arise from micro-reentries near infarct regions in simulations of a patient-specific heart model. *Scientific reports*, Nature Publishing Group, v. 8, n. 1, p. 1–14, 2018.
- OLIVEIRA, R. S. *et al.* An adaptive mesh algorithm for the numerical solution of electrical models of the heart. In: SPRINGER. *International Conference on Computational Science and Its Applications (ICCSA)*. Salvador, 2012. p. 649–664.
- OLIVEIRA, R. S. *et al.* Simulations of cardiac electrophysiology combining gpu and adaptive mesh refinement algorithms. In: SPRINGER. *International Conference on Bioinformatics and Biomedical Engineering*. Cham, 2016. p. 322–334.

- OLIVEIRA, R. S. *et al.* Performance evaluation of gpu parallelization, space-time adaptive algorithms, and their combination for simulating cardiac electrophysiology. *International journal for numerical methods in biomedical engineering*, Wiley Online Library, v. 34, n. 2, p. e2913, 2018b.
- PAGANI, S. *et al.* Data integration for the numerical simulation of cardiac electrophysiology. *Pacing and Clinical Electrophysiology*, Wiley Online Library, v. 44, n. 4, p. 726–736, 2021.
- PEIRLINCK, M. *et al.* Precision medicine in human heart modeling. *Biomechanics and modeling in mechanobiology*, Springer, v. 20, n. 3, p. 803–831, 2021.
- Physiome Project. *Learning CellML by example: The Hodgkin & Huxley model*. 2020. <https://models.physiomeproject.org/e/e5/tutorial/tutorial.html/view>. Acesso: 2020-06-11.
- Physiome Project. *Luo, Rudy, 1991*. 2020. <https://models.cellml.org/exposure/2d2ce7737b42a4f72d6bf8b67f6eb5a2>. Acesso: 2020-06-11.
- Physiome Project. *Ten Tusscher, Noble, Noble, Panfilov, 2004*. 2020. <https://models.physiomeproject.org/exposure/140813f9584b1108c1e7decf0a6f8099>. Acesso: 2020-06-11.
- PLANK, G. *et al.* The opencarp simulation environment for cardiac electrophysiology. *Computer Methods and Programs in Biomedicine*, Elsevier, v. 208, p. 106223, 2021.
- POTSE, M. *et al.* A comparison of monodomain and bidomain reaction-diffusion models for action potential propagation in the human heart. *IEEE Transactions on Biomedical Engineering*, IEEE, v. 53, n. 12, p. 2425–2435, 2006.
- QUINN, M. J. *Parallel Programming in C with MPI and Openmp*. 1. ed. New York: McGraw-Hill Education, 2003.
- REID, J. K. On the method of conjugate gradients for the solution of large sparse systems of linear equations. In: ACADEMIC PRESS. *Proc. Conference on Large sparse sets of linear equations*. New York, 1971. p. 231–254.
- ROGNES, M. E. *et al.* cbcbeat: an adjoint-enabled framework for computational cardiac electrophysiology. *J. Open Source Softw.*, v. 2, n. 13, p. 224, 2017.
- RUDY, Y.; SILVA, J. R. Computational biology in the study of cardiac ion channels and cell electrophysiology. *Quarterly reviews of biophysics*, NIH Public Access, v. 39, n. 1, p. 57, 2006.
- RUSH, S.; LARSEN, H. A practical algorithm for solving dynamic membrane equations. *IEEE Transactions on Biomedical Engineering*, IEEE, n. 4, p. 389–392, 1978.
- SACHETTO, R.; ALONSO, S.; SANTOS, R. W. D. Killing many birds with two stones: hypoxia and fibrosis can generate ectopic beats in a human ventricular model. *Frontiers in Physiology*, Frontiers, v. 9, p. 764, 2018.
- SACHSE, F. B. *Computational cardiology: modeling of anatomy, electrophysiology, and mechanics*. Heidelberg: Springer Science & Business Media, 2004. v. 2966.

- SANTOS, R. W. D. *et al.* Preconditioning techniques for the bidomain equations. In: *Domain Decomposition Methods in Science and Engineering*. Heidelberg: Springer, 2005. p. 571–580.
- SHAW, R. M.; RUDY, Y. Electrophysiologic effects of acute myocardial ischemia: a theoretical study of altered cell excitability and action potential duration. *Cardiovascular research*, Elsevier Science, v. 35, n. 2, p. 256–272, 1997.
- SHERWOOD, L.; WARD, C. *Human Physiology: From Cells to Systems*. 4. ed. Toronto: Nelson College Indigenous, 2018.
- SHI, H. *et al.* Metabolites of hypoxic cardiomyocytes induce the migration of cardiac fibroblasts. *Cellular Physiology and Biochemistry*, Karger Publishers, v. 41, n. 1, p. 413–421, 2017.
- SILVERTHORN, D. U. *Fisiologia Humana: Uma Abordagem Integrada*. 5. ed. Porto Alegre: Jones & Bartlett, 2010.
- SINGH, H. *et al.* Gpu and cuda in hard computing approaches: analytical review. *Proceedings of ICRIC 2019*, Springer, p. 177–196, 2020.
- SITSYLITSYN, Y. Methods and tools for teaching parallel and distributed computing in universities: a systematic review of the literature. In: EDP SCIENCES. *Proceedings of the International Conference on History, Theory and Methodology of Learning (ICHTML 2020)*. Kryvyi Rih, 2020. v. 75, p. 04017.
- STRIKWERDA, J. C. *Finite difference schemes and partial differential equations*. Philadelphia: SIAM, 2004.
- SUNDNES, J. *et al.* *Computing the electrical activity in the heart*. Heidelberg: Springer Science & Business Media, 2007. v. 1.
- TOMASELLI, G. F.; MARBÁN, E. Electrophysiological remodeling in hypertrophy and heart failure. *Cardiovascular research*, Elsevier Science, v. 42, n. 2, p. 270–283, 1999.
- TUNG, L. *A bi-domain model for describing ischemic myocardial dc potentials*. Tese (Doutorado) — Massachusetts Institute of Technology, 1978.
- TUSSCHER, K. H. T.; PANFILOV, A. V. Alternans and spiral breakup in a human ventricular tissue model. *American Journal of Physiology-Heart and Circulatory Physiology*, American Physiological Society, v. 291, n. 3, p. H1088–H1100, 2006.
- TUSSCHER, K. H. T.; PANFILOV, A. V. Cell model for efficient simulation of wave propagation in human ventricular tissue under normal and pathological conditions. *Physics in Medicine & Biology*, IOP Publishing, v. 51, n. 23, p. 6141, 2006.
- TUSSCHER, K. H. T.; PANFILOV, A. V. Influence of diffuse fibrosis on wave propagation in human ventricular tissue. *Europace*, Oxford University Press, v. 9, n. suppl\_6, p. vi38–vi45, 2007.
- TUSSCHER, K. H. *ten et al.* A model for human ventricular tissue. *American Journal of Physiology-Heart and Circulatory Physiology*, American Physiological Society, v. 286, n. 4, p. H1573–H1589, 2004.

VERSTEEG, H. K.; MALALASEKERA, W. *An introduction to computational fluid dynamics: the finite volume method*. Glasgow: Pearson education, 2007.

WHO, W. H. O. *World Health Organization - Cardiovascular Disease*. 2020. <https://www.who.int/health-topics/cardiovascular-diseases>. Acesso: 2020-06-11.

WIDMAIER, E. P.; RAFF, H.; STRANG, K. F. *Vander's Human Physiology*. 15. ed. New York: McGraw-Hill Education, 2018.

XIA, Y.; WANG, K.; ZHANG, H. Parallel optimization of 3d cardiac electrophysiological model using gpu. *Computational and mathematical methods in medicine*, Hindawi, v. 2015, 2015.

ZIPES, D.; JALIFE, J.; STEVENSON, W. *Cardiac Electrophysiology: From Cell to Bedside*. 7. ed. address=Philadelphia,: Elsevier, 2017.

## APÊNDICES

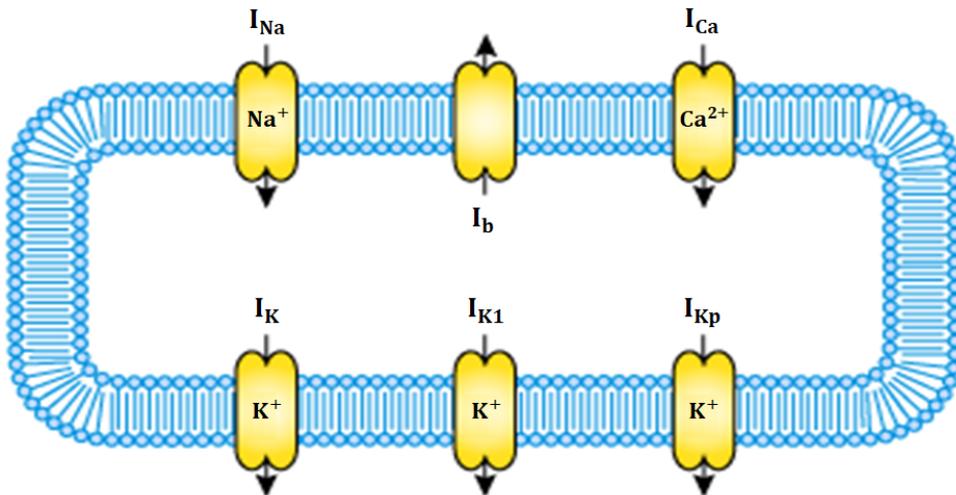
## A MODELAGEM DA ELETROFISIOLOGIA CARDÍACA

### A.1 Modelo Luo Rudy

O modelo de Luo Rudy (LUO; RUDY, 1991) foi desenvolvido baseado em células do ventrículo de mamíferos. A variação do potencial de membrana é dada pela mesma Equação 2.29 do modelo de Hodgkin-Huxley, porém são consideradas 6 correntes iônicas. Sendo quatro de efluxo: corrente de potássio dependente do tempo ( $I_K$ ), corrente de potássio independente do tempo ( $I_{K1}$ ), corrente de potássio de platô ( $I_{Kp}$ ) e, uma “corrente de fuga” chamada de *background* ( $I_b$ ) de potássio. Além de duas correntes de influxo: corrente rápida de sódio ( $I_{Na}$ ) e uma corrente lenta de cálcio ( $I_{CaL}$ ). A Equação A.1 mostra a corrente iônica total do modelo Luo Rudy e a Figura 17 mostra a representação esquemática das correntes iônicas através da membrana para esse modelo.

$$I_{\text{ion}} = I_{Na} + I_{Ca} + I_K + I_{K1} + I_{Kp} + I_b \quad (\text{A.1})$$

Figura 17 – Representação esquemática das correntes iônicas do modelo Luo Rudy



Fonte: Adaptado de Physiome Project (2020b)

Todas correntes iônicas do modelo são controladas por variáveis de ativação da forma:

$$\frac{dy}{dt} = \frac{y_{\infty} - y}{\tau_y} \quad (\text{A.2})$$

onde  $y$  é a variável de ativação/inativação em questão e os termos  $y_{\infty}$  e  $\tau_y$  são definidos

como:

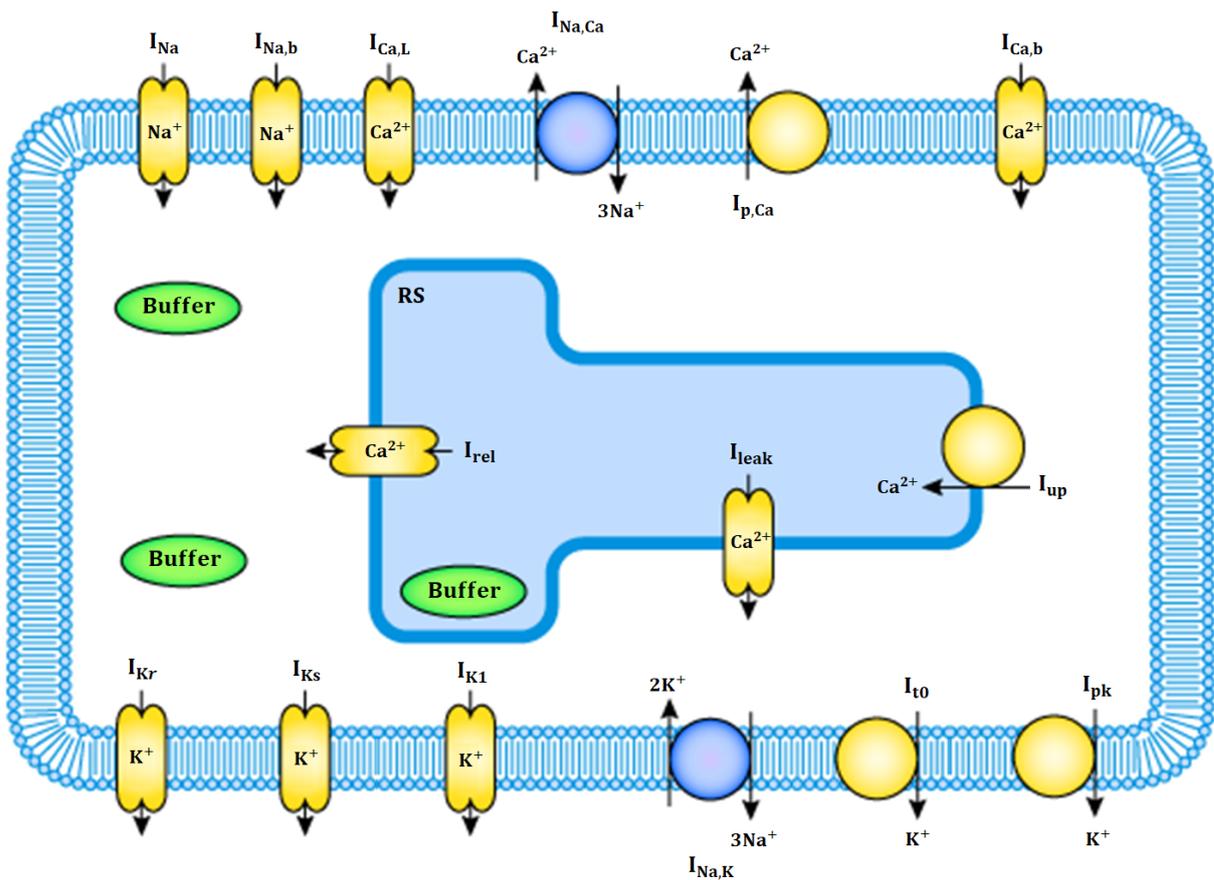
$$y_{\infty} = \frac{\alpha_y}{\alpha_y + \beta_y}, \quad \tau_y = \frac{1}{\alpha_y + \beta_y}, \quad (\text{A.3})$$

sendo  $\alpha_y$  e  $\beta_y$  funções de  $V_m$ . Expressões completas para  $\alpha_y$ ,  $\beta_y$  e correntes iônicas podem ser encontradas em Luo e Rudy (1991).

## A.2 Modelos Ten Tusscher, Ten Tusscher-Panfilov (TT2) e Ten Tusscher-Panfilov 2 (TT3)

O modelo de Ten Tusscher (TUSSCHER *et al.*, 2004), representado na Figura 18 foi desenvolvido para modelar células do ventrículo humano. O modelo inclui 11 correntes iônicas e duas correntes de trocadores e bombas.

Figura 18 – Representação esquemática das correntes iônicas do modelo Ten Tusscher. RS: Reticulo sarcoplasmático



Fonte: Adaptado de Physiome Project (2020c)

A variação do potencial de membrana é dada pela Equação 2.29 (mesma do modelo HH) e a corrente iônica é:

$$I_{\text{ion}} = I_{Na} + I_{K1} + I_{t0} + I_{Kr} + I_{Ks} + I_{CaL} + I_{NaCa} + I_{NaK} + I_{pCa} + I_{pK} + I_{bCa} + I_{bNa} \quad (\text{A.4})$$

sendo  $I_{Na}$  a corrente de sódio,  $I_{K1}$  a corrente retificadora de influxo,  $I_{t0}$  a corrente transiente de efluxo,  $I_{Kr}$  a corrente rápida de potássio,  $I_{Ks}$  a corrente lenta de potássio,  $I_{CaL}$  a corrente lenta de cálcio,  $I_{Na/Ca}$  a corrente do trocador  $Na^+/Ca^{2+}$ ,  $I_{Na/K}$  a corrente da bomba  $Na^+/K^+$ ,  $I_{pCa}$  e  $I_{pK}$  são correntes de platô de cálcio e potássio respectivamente,  $I_{bCa}$  e  $I_{bNa}$  correntes de *background* de cálcio e sódio.

As correntes iônicas no modelo Ten Tusscher são controladas por variáveis de ativação e inativação da mesma maneira que o modelo de Luo Rudy (Equações A.2 e A.3). O modelo considera ainda correntes intracelulares para modelar a dinâmica do cálcio. São elas: (i)  $I_{leak}$ : uma "corrente de fuga" do retículo sarcoplasmático (RS) para o citoplasma; (ii)  $I_{up}$ : uma bomba inserindo cálcio no RS; (iii)  $I_{rel}$ : uma corrente de liberação de cálcio induzida por cálcio. Por fim, o modelo considera também o armazenamento de cálcio no citoplasma e no RS, por meio de um *buffer*. Expressões completas para  $\alpha_y$ ,  $\beta_y$  e correntes iônicas, incluindo as correntes intracelulares, podem ser encontradas em Tusscher *et al.* (2004).

Posteriormente, Tusscher e Panfilov (2006a) desenvolveram uma nova versão para seu modelo de células ventriculares humanas, baseando-se em medições experimentais da restituição da DPA humano como potencial para estudar a fibrilação ventricular. Além disso, essa versão do modelo inclui uma descrição mais extensa da dinâmica intracelular do cálcio. Isso foi feito, pela inclusão da dinâmica do cálcio no subespaço, que controla a corrente de cálcio do tipo L e a liberação de cálcio induzida por cálcio (*calcium-induced calcium release* - *CICR*). A corrente *CICR* foi modelado por uma cadeia de Markov de quatro estados para o receptor de rianodina e incorporando tanto a inativação rápida, quanto a lenta da corrente de cálcio do tipo L dependente da voltagem.

O modelo de Ten Tusscher-Panfilov (TT2) possui 19 variáveis de estado descritas pelas EDOs, além de descrever a concentração intracelular de cálcio e as condutâncias dos canais iônicos. Com base no modelo original, além da modelagem da corrente *CICR*, apenas as correntes  $I_{CaL}$  e  $I_{Ks}$  são modificadas. Expressões completas para as correntes iônicas podem ser encontradas em Tusscher e Panfilov (2006a), Tusscher *et al.* (2004).

O modelo TT3 é baseado nos modelos Ten Tusscher e Ten Tusscher-Panfilov 2 (TT2), os quais são apresentados anteriormente. Tusscher e Panfilov (2006b) expandiram o modelo TT2 para formular um modelo de arritmia para células ventriculares humanas, capaz de simular arritmia de órgãos inteiros, e detalhado o suficiente para capturar os efeitos de processos no nível celular, como bloqueios de corrente e canalopatias (doenças que se desenvolvem devido a defeitos nos canais iônicos). O modelo foi capaz ainda de reproduzir corretamente os efeitos das ondas em espiral, típicas de arritmias.

A influência da dinâmica do cálcio na excitação cardíaca (presentes no modelo TT2) é removida no modelo TT3, sem prejudicar a eficácia do modelo. Na verdade, a dinâmica do cálcio é importante em estudos específico, tais como sobrecarga de cálcio,

liberação espontânea de cálcio e liberação de cálcio induzida por cálcio. Assim, a corrente iônica  $I_{CaL}$  foi alterada de maneira a permitir a redução do modelo. Para reduzir ainda mais o número de variáveis, os autores propõem utilizar aproximações de estado quase estacionário para algumas das variáveis de ativação/inativação de corrente que mudam rapidamente. Essas variáveis são relacionadas às correntes  $I_{t0}$ ,  $I_{Kr}$  e  $I_{CaL}$ . Por fim, valores de parâmetros da condutância das correntes  $I_{CaL}$ ,  $I_{Kr}$ ,  $I_{Ks}$  e  $I_{pK}$  também são alterados para adequar o modelo. Expressões completas para as correntes iônicas podem ser encontradas em Tusscher e Panfilov (2006b), Tusscher e Panfilov (2006a) e Tusscher *et al.* (2004).

### A.3 Modelo Oliveira (MONOALG3D)

Embora o modelo TT3 seja capaz de simular o efeito de uma condição desenvolvida de isquemia ou hipercalemia (alta concentração de potássio no sangue), ele não é capaz de investigar o desenvolvimento de condições de isquemia e hipercalemia, devido à ausência da dinâmica de sódio e potássio.

Baseado em estudos prévios (SHAW; RUDY, 1997; CAMPOS *et al.*, 2012), para incluir os efeitos de isquemia, Oliveira *et al.* (2018) propuseram a modificação do modelo TT3 para reproduzir hipóxia, hipercalemia e acidose (diminuição do pH sanguíneo, ou seja, excesso de ácido no sangue). No modelo de Oliveira, a hipóxia é modelada por inclusão da corrente de  $K^+$  ativada por adenosina trifosfato (ATP),  $I_{K(ATP)}$ , e pela modificação da condutividade do canal iônico do tipo L de cálcio  $P_{Ca(L)}$  que também é dependente da ATP. A corrente iônica para esse modelo se torna:

$$I_{ion} = I_{Na} + I_{K1} + I_{t0} + I_{Kr} + I_{Ks} + I_{CaL} + I_{Na/Ca} + I_{Na/K} + I_{pCa} + I_{pK} + I_{bCa} + I_{bNa} + I_{K(ATP)} \quad (A.5)$$

onde  $I_{t0}$  é a corrente que atravessa o canal transiente de efluxo,  $I_{pCa}$  é a corrente de platô de íons de cálcio,  $I_{pK}$  é a corrente de platô de íons de potássio,  $I_{bCa}$  é a corrente *background* de íons de cálcio,  $I_{bNa}$  é a corrente *background* de íons de sódio e  $I_{K(ATP)}$  é a corrente de íons de potássio ativada por adenosina trifosfato.

A formulação para  $I_{K(ATP)}$  e  $P_{Ca(L)}$  são expressados nas Equações A.6 e A.7:

$$\begin{aligned} I_{K(ATP)} &= g_{K_{ATP}}(V - E_k), g_{K_{ATP}} = G_{K_{ATP}} P_{ATP} ([K]_o / [K]_{o,normal})^n, \\ G_{K_{ATP}} &= 195 \times 10^{-6} / Nichols_{area} (nS/cm^2), \\ P_{ATP} &= \frac{1}{1 + \left( \frac{[ATP]_i}{k_{0,5}} \right)^H}, \end{aligned} \quad (A.6)$$

onde  $E_k$  é o potencial de reversão de potássio,  $Nichols_{area} = 5 \times 10^3$ ,  $n=0,24$ ,  $H=2$ ,

$k_{0,5}=0,250 \mu\text{M}$  (SHAW; RUDY, 1997).

$$P_{CaL,ATP} = \frac{1}{1 + \left(\frac{k_{0,5}}{[ATP]_i}\right)^H} \quad (\text{A.7})$$

sendo  $H=2,6$ ,  $k_{0,5}=1,4 \text{ mM}$ .

Por fim, a hipercalemia é introduzida pelo aumento da concentração de potássio extracelular ( $K_o$ ) e a acidose é modelada pela redução das condutâncias das correntes de cálcio do tipo L ( $g_{CaL}$ ) e de sódio ( $g_{Na}$ ).

Expressões completas para as demais correntes iônicas podem ser encontradas em Tusscher e Panfilov (2006b), Tusscher e Panfilov (2006a), Tusscher *et al.* (2004).

## B MÉTODO DO GRADIENTE CONJUGADO (GC)

Originalmente proposto por Hestenes *et al.* (1952) para solucionar um sistema linear no formato  $Ax = b$  de ordem  $n \times m$ , sendo  $A$  uma matriz simétrica e positiva definida. Considerando  $x_0$  um vetor arbitrário como estimativa inicial e  $p_1=r_0$ , o método do gradiente conjugado constrói duas sequências de vetores:  $r_k = r_{k-1} - \alpha_k Ap_k$  e  $p_k = r_{k-1} + \beta_k p_{k-1}$ , para  $k=1,2,\dots,n$ .

Os vetores devem satisfazer as condições de ortogonalidade e conjugação.

Os escalares  $\alpha$  e  $\beta$  são dados por:

$$\alpha_i = \frac{r_{k-1}^t r_{k-1}}{p_k^t A p_k}, \beta_i = \frac{r_{k-1}^t r_{k-1}}{r_{k-2}^t r_{k-2}} \quad (\text{B.1})$$

Assim, o algoritmo do gradiente conjugado, como proposto por Reid (1971), fica da forma mostrada no Algoritmo 3.

---

### Algoritmo 3 Método do gradiente conjugado

---

- 1:  $x_0$  estimativa inicial
  - 2:  $r_0 = b - Ax_0$
  - 3:  $k = 1$
  - 4:  $p_1 = r_0$
  - 5:  $\alpha_1 = (r_0^t r_0) / (p_1^t A p_1)$
  - 6:  $x_1 = x_0 + \alpha_1 p_1$
  - 7:  $r_1 = r_0 - \alpha_1 A p_1$
  - 8: **while**  $r_k \neq 0$  **do**
  - 9:    $k = k + 1$
  - 10:    $\beta_k = (r_{k-1}^t r_{k-1}) / (r_{k-2}^t A r_{k-2})$
  - 11:    $p_k = r_{k-1} + \beta_k p_{k-1}$
  - 12:    $\alpha_k = (r_{k-1}^t r_{k-1}) / (p_k^t A p_k)$
  - 13:    $x_k = x_{k-1} + \alpha_k p_k$
  - 14:    $r_k = r_{k-1} - \alpha_k A p_k$
  - 15: **end while**
  - 16:  $x = x_k$
-

## C TABELA-RESUMO DOS TESTES

Como forma de consolidar os resultados anteriores, as Tabelas A1 e A2 apresentam os resultados de todos os testes realizados para o tempo de solução das EDOs e para o tempo de solução global do algoritmo, respectivamente.

**Tabela A1 – Resumo do tempo de execução das EDOs de todos os testes realizados. BC: Balanceamento de carga**

Volumes	Quantidade de GPUs	GTX 960 e GTX 960 (s)	TITAN Xp e GTX 960 sem BC (s)	TITAN Xp e GTX 960 com BC (s)
3.360	1	78,11	32,76	-
	2	57,54	57,51	-
5.950	1	135,85	59,80	-
	2	110,05	109,92	-
26.880	1	438,77	140,77	-
	2	300,57	299,74	-
52.500	1	844,59	246,14	-
	2	566,25	568,77	236,26
215.040	1	3331,75	846,36	-
	2	2145,62	2143,94	796,56
420.000	1	6390,14	1600,47	-
	2	4118,18	4117,66	1487,83

Fonte: Elaborado pelo autor

**Tabela A2 – Resumo do tempo de execução global do algoritmo de todos os testes realizados. BC: Balanceamento de carga**

Volumes	Quantidade de GPUs	GTX 960 e GTX 960 (s)	TITAN Xp e GTX 960 sem BC (s)	TITAN Xp e GTX 960 com BC (s)
3.360	1	92,67	45,63	-
	2	72,56	71,16	-
5.950	1	152,06	74,20	-
	2	126,88	126,46	-
26.880	1	499,65	191,03	-
	2	367,78	357,38	-
52.500	1	1001,46	391,77	-
	2	739,61	732,76	391,53
215.040	1	4212,97	1664,89	-
	2	3035,38	2971,22	1624,10
420.000	1	8163,59	3226,02	-
	2	5905,78	5756,27	3122,72

Fonte: Elaborado pelo autor