



Qualificação - Mestrado

Estabilidade e Sincronização em Redes Dinâmicas Discretas
Acopladas: uma Abordagem Computacional

Universidade Federal do ABC
Centro de Matemática, Computação e Cognição (CMCC)

Candidato: Carlos Reynaldo Portocarrero Tovar (UFABC-CMCC)
Orientador: Luiz Carlos da Silva Rozante (UFABC-CMCC)

Santo André, 28 de setembro de 2019



Sumário

Motivação

Definições preliminares

Métodos propostos

Resultados Preliminares

Conclusões parciais



Seção 1 | Motivação



Estabilização e sincronização

- Sincronização é a coordenação de eventos para operar um sistema em uníssono.
- A Estabilidade está associada à ideia de permanência em um determinado estado por um determinado ente.
- No século XVII, Christiaan Huygens fez estudos sobre sincronismo de relógios de pêndulo.



Estabilização e sincronização

- Sincronização é a coordenação de eventos para operar um sistema em uníssono.
- A Estabilidade está associada à ideia de permanência em um determinado estado por um determinado ente.
- No século XVII, Christiaan Huygens fez estudos sobre sincronismo de relógios de pêndulo.



Estabilização e sincronização

- Sincronização é a coordenação de eventos para operar um sistema em uníssono.
- A Estabilidade está associada à ideia de permanência em um determinado estado por um determinado ente.
- No século XVII, Christiaan Huygens fez estudos sobre sincronismo de relógios de pêndulo.



Estabilidade e Sincronização - Video 1, Video 2

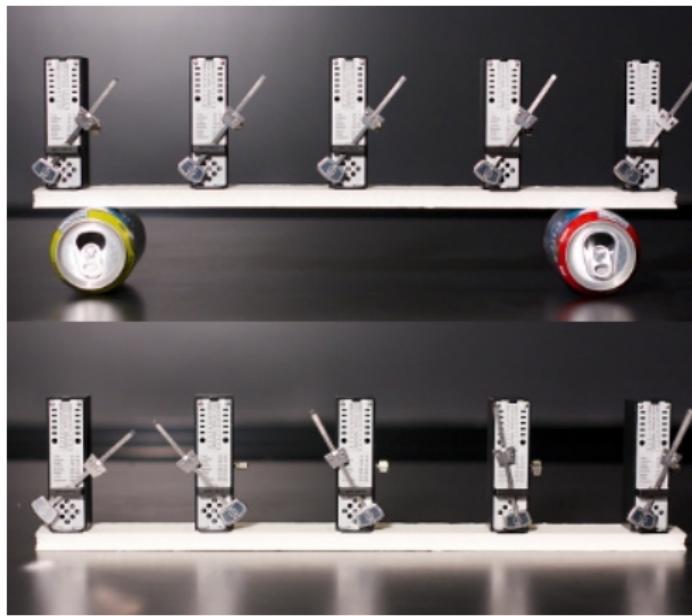


Figura: Metrônimos sincronizados (Fonte: [Harvard-University, 2002]).



Redes Dinâmicas Discretas (RDD)

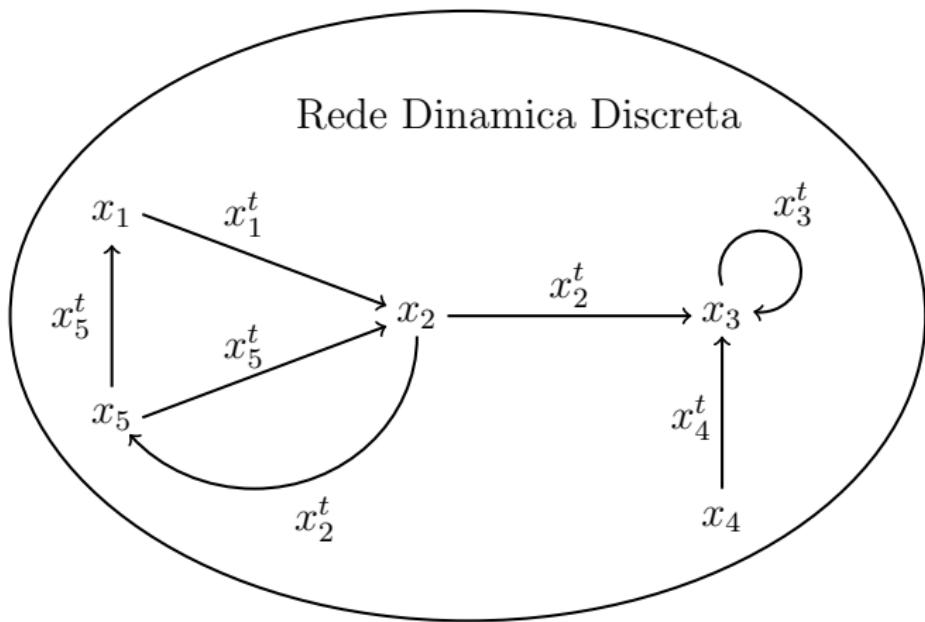


Figura: Representação topológica de uma RDD, com variável não regulada.



Redes Dinâmicas Discretas (RDD)

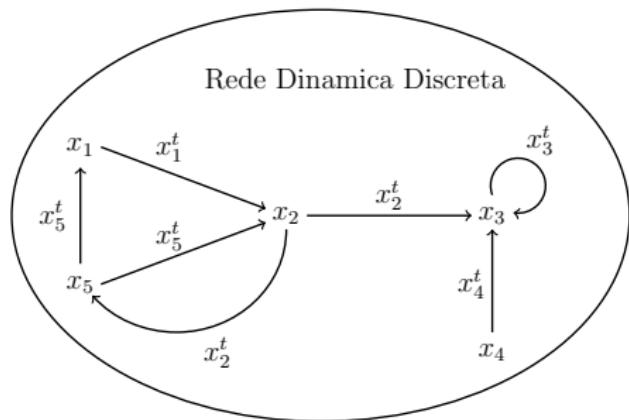


Figura: Exemplo uma RDD.

Possíveis valores das variáveis em uma RDD, ao longo do tempo(t). Ao passo de um estado a outro chamamos de *transição*.

t	x_1	x_2	x_3	x_4	x_5
0	5	23	5	10	2
1	6	1	11	10	6
2	7	10	8	10	5
3	8	12	1	10	7
...



Grafo de Transição de Estados

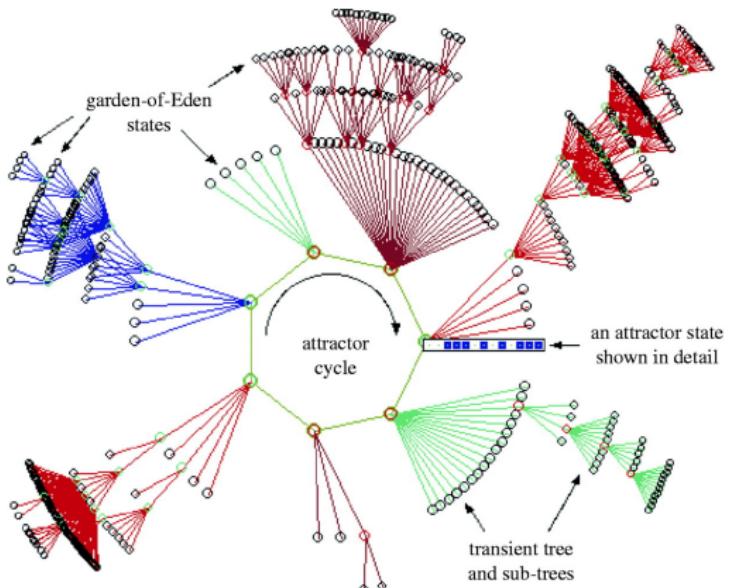


Figura: Grafo de Transição de Estados como um atrator só, mas pode ter mais atratores, grafo desconexo (Fonte: [Wuensche, 2011]).



Redes Dinâmicas Discretas Acopladas

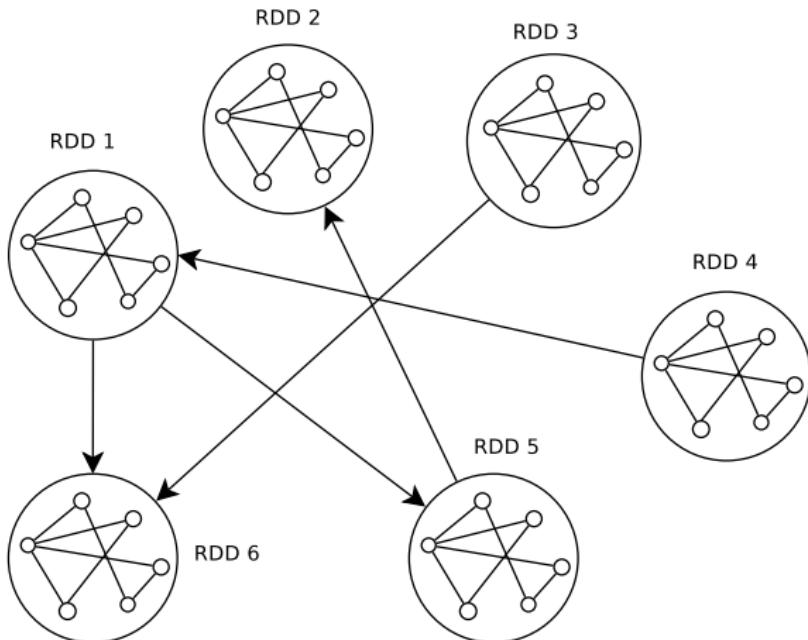


Figura: Exemplo uma RDDA de 6 RDDs.



Redes Dinâmicas Discretas Acopladas

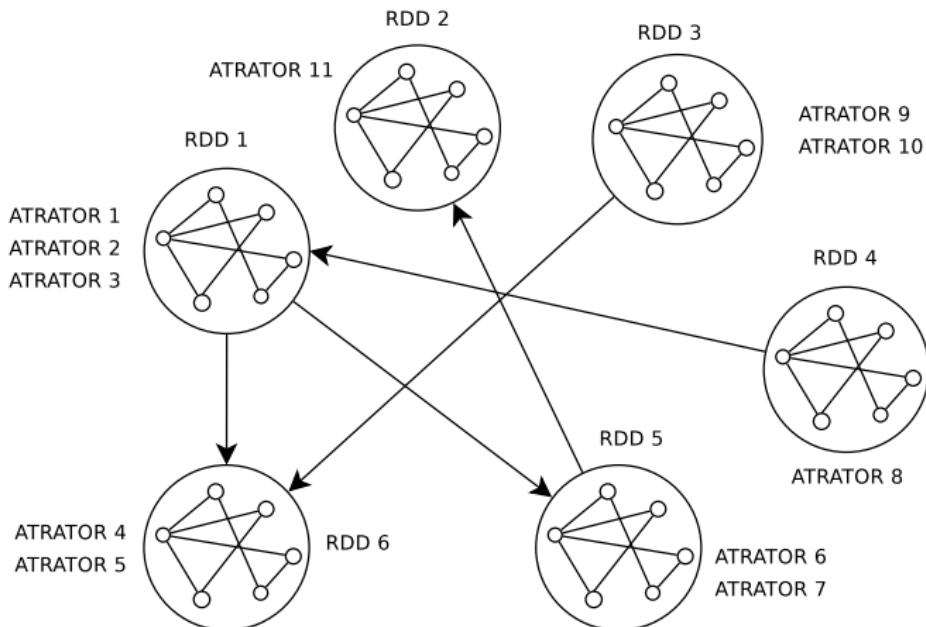


Figura: Exemplo uma RDDA de 6 RDDs, com seus respetivos atratores.



Redes Dinâmicas Discretas Acopladas

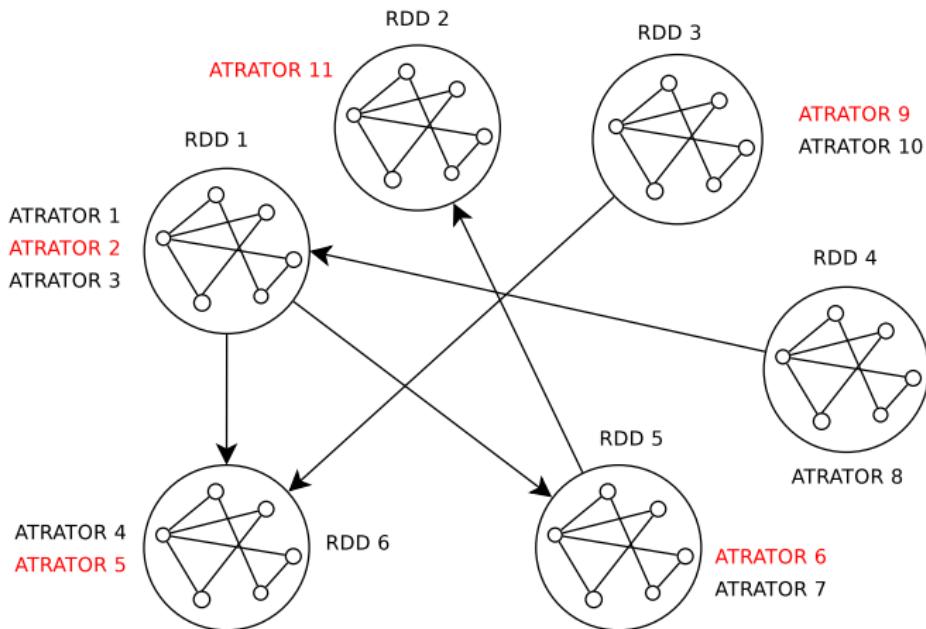


Figura: Campo de atratores 1, com entradas e saídas compatíveis.



Redes Dinâmicas Discretas Acopladas

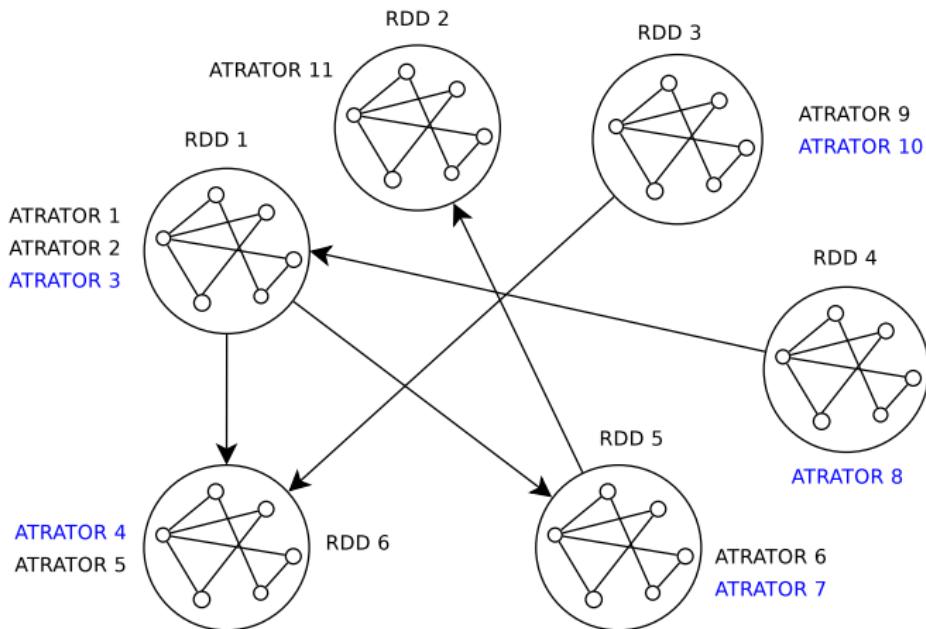


Figura: Campo de atratores 2, com entradas e saídas compatíveis.



Objetivos

- **Objetivo Geral:** Desenvolver e implementar um método eficiente que, dada uma RDDA como entrada, seja capaz de responder se ela contém ou não campos atratores estáveis.
- **Objetivos específicos:**
 - Adaptar o algoritmo de Dubrova e Teslenko (2011) e combiná-lo com o algoritmo de Carastan-Santos et al. (2017) para achar atratores locais nas RDDs pertencentes a uma RDDA.
 - Implementar uma abordagem original (baseada em SAT “classico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos, que é uma característica necessária à aplicação no contexto de RDDAs.
 - Construir atribuições estáveis como combinações de atratores locais dentro da RDDA.



Objetivos

- **Objetivo Geral:** Desenvolver e implementar um método eficiente que, dada uma RDDA como entrada, seja capaz de responder se ela contém ou não campos atratores estáveis.
- **Objetivos específicos:**
 - Adaptar o algoritmo de Dubrova e Teslenko (2011) e combiná-lo com o algoritmo de Carastan-Santos et al. (2017) para achar atratores locais nas RDDs pertencentes a uma RDDA.
 - Implementar uma abordagem original (baseada em SAT “classico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos, que é uma característica necessária à aplicação no contexto de RDDAs.
 - Construir atribuições estáveis como combinações de atratores locais dentro da RDDA.



Objetivos

- **Objetivo Geral:** Desenvolver e implementar um método eficiente que, dada uma RDDA como entrada, seja capaz de responder se ela contém ou não campos atratores estáveis.
- **Objetivos específicos:**
 - Adaptar o algoritmo de Dubrova e Teslenko (2011) e combiná-lo com o algoritmo de Carastan-Santos et al. (2017) para achar atratores locais nas RDDs pertencentes a uma RDDA.
 - Implementar uma abordagem original (baseada em SAT “classico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos, que é uma característica necessária à aplicação no contexto de RDDAs.
 - Construir atribuições estáveis como combinações de atratores locais dentro da RDDA.



Objetivos

- **Objetivo Geral:** Desenvolver e implementar um método eficiente que, dada uma RDDA como entrada, seja capaz de responder se ela contém ou não campos atratores estáveis.
- **Objetivos específicos:**
 - Adaptar o algoritmo de Dubrova e Teslenko (2011) e combiná-lo com o algoritmo de Carastan-Santos et al. (2017) para achar atratores locais nas RDDs pertencentes a uma RDDA.
 - Implementar uma abordagem original (baseada em SAT “classico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos, que é uma característica necessária à aplicação no contexto de RDDAs.
 - Construir atribuições estáveis como combinações de atratores locais dentro da RDDA.



Aplicações

Os processos de estabilização e sincronização em redes de entidades dinâmicas interagentes são quase onipresentes na natureza e desempenham um papel muito importante em muitos contextos diferentes [Arenas et al., 2008].

- Bioinformática
- Mineração de Dados
- Telecomunicações
- Redes de Distribuição
- Redes Sociais.



Seção 2 | Definições preliminares



Sistemas Dinâmicos Discretos

Um *sistema dinâmico discreto* consiste em um espaço de estados X e um mapeamento

$$f : X \rightarrow X$$

tal que f define a evolução no tempo do sistema.

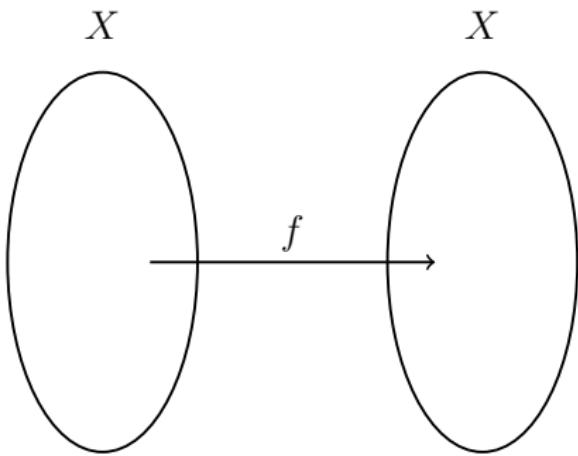


Figura: Função em um sistema dinâmico discreto.



Sistemas Dinâmicos Discretos Finitos

Um *sistema dinâmico discreto finito* (SDF) é um sistema dinâmico discreto (X, f) onde X é finito. Em outras palavras, seja um conjunto de variáveis

$$\mathcal{X} = \{x_1, x_2, \dots, x_n\}$$

que assumem valores num conjunto finito X e sejam as funções f_1, f_2, \dots, f_n tal que a função

$$f_i : X^n \rightarrow X$$

determina o estado da variável x_i .



Sistemas Dinâmicos Discretos Finitos

Assumindo passos de tempo discreto e atualização síncrona das variáveis em \mathcal{X} , o próximo estado (tempo $t + 1$) de um SDF é dado pelas funções de transição F aplicadas ao estado atual (tempo t) como segue:

$$\begin{cases} x_1^{t+1} = f_1(x_1^t, \dots, x_n^t), \\ x_2^{t+1} = f_2(x_1^t, \dots, x_n^t), \\ \dots \\ x_n^{t+1} = f_n(x_1^t, \dots, x_n^t). \end{cases} \quad (1)$$

Uma RDD é um caso particular de um SDF. Quando as variáveis \mathcal{X} só podem ter valores no domínio de 0 e 1, esta RDD é denominada como *Redes Booleana*.



Grafo de Transições de Estados (GTE)

A dinâmica de uma rede dinâmica discreta $j = (\mathcal{X}, X^n, F^j)$ pode ser representada por um grafo dirigido $G_j = (V, E)$, denominado *Grafo de Transições de Estado* (GTE), o qual é assim definido:

$$V = X^n, (x, y) \in E \leftrightarrow F^j(x) = y$$

Quando as arestas do grafo (transições) voltam para o mesmo estado x , representam um *atrator*.



Exemplos de GTE

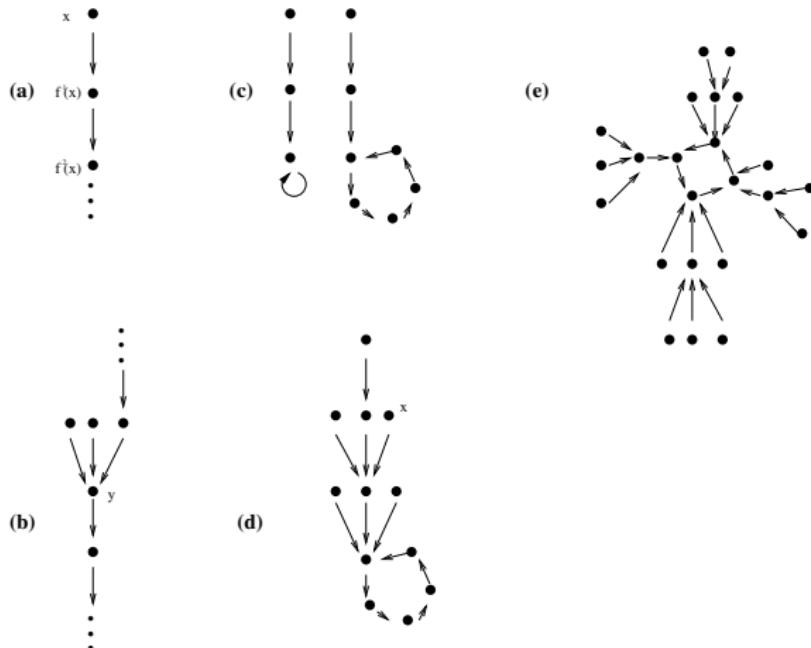


Figura: Características do GTE



Redes Dinâmicas Discretas Acopladas (RDDAs)

Considere uma rede de m entidades dinâmicas interagentes, onde cada uma delas entidades é modelada como uma RDD. Nesse sistema (uma “rede de redes”) podemos distinguir três elementos constituintes:

- ① O padrão de conexão – a estrutura da rede – entre as entidades dinâmicas, representado por um grafo dirigido G ;
- ② As regras que governam a dinâmica local das entidades, representadas por RDDs;
- ③ A natureza da interação entre as entidades, isto é, a forma como os acoplamentos inter-entidades (entre RDDs) ocorrem, que é definida por uma função de acoplamento.



Redes Dinâmicas Discretas Acopladas

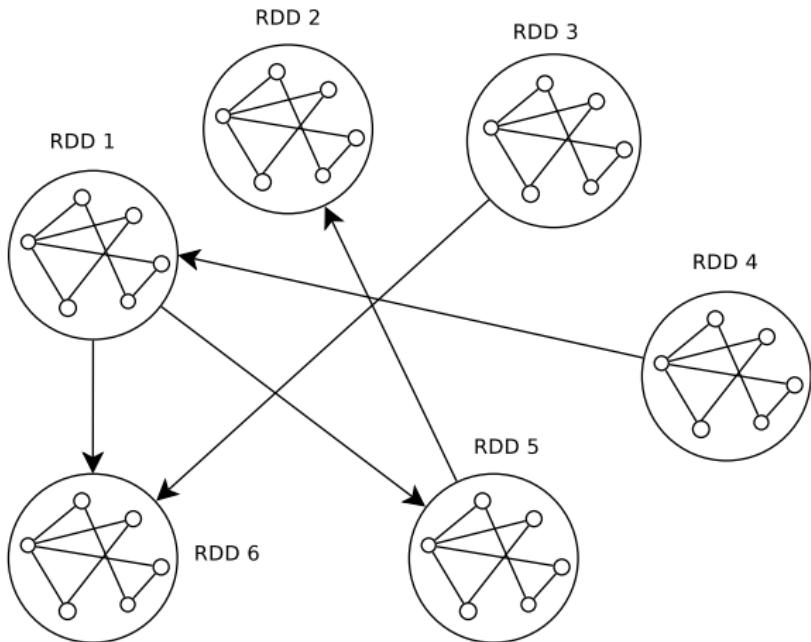


Figura: Exemplo uma RDDA de 6 RDDs.



Redes Dinâmicas Discretas Acopladas

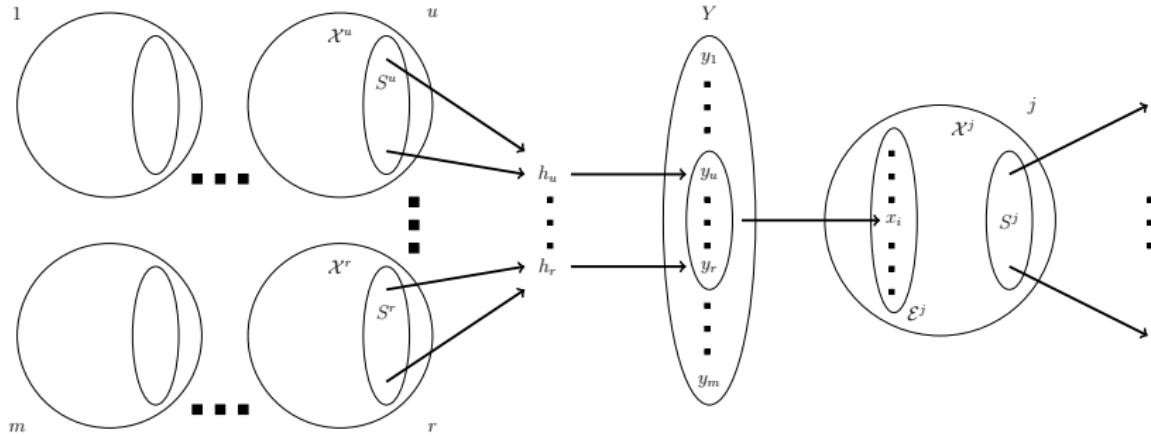


Figura: Uma representação gráfica da forma pela qual as RDDs u, \dots, r (que são vizinhas de j) exercem influência sobre j .



Redes Dinâmicas Discretas Acopladas (RDDAs)

As sinais de acoplamento y são formadas assim:

$$y_w(t) = h_w(x_p^w(t), \dots, x_q^w(t)).$$

Os estados no tempo $t + 1$ estão em função dos valores das variáveis do tempo anterior, mas também das sinais de acoplamento de as outras redes:

$$x_i^j(t+1) = \begin{cases} f_i^j(x_k^j(t), \dots, x_l^j(t), y_u(t), \dots, y_r(t)), & \text{se } i \in \mathcal{E}^j \\ f_i^j(x_k^j(t), \dots, x_l^j(t)), & \text{se } i \in \mathcal{X}^j \setminus \mathcal{E}^j, \end{cases} \quad (2)$$



Problema de Satisfatibilidade Booleana (SAT)

O SAT é o problema de determinar se existe uma valoração para as variáveis de uma dada fórmula Booleana F tal que esta valoração, volte F verdadeira, então ela é dita ser *satisfazível*, caso contrario ela é considerada *insatisfazível*. Por exemplo, tomando x_1, x_2, x_3, x_4 como as variáveis Booleanas e a expressão :

$$(x_1 \vee \neg x_3 \vee x_4) \wedge (\neg x_2 \vee x_3 \vee \neg x_4) \wedge (x_1 \vee \neg x_3 \vee x_4) \wedge (\neg x_2 \vee x_3 \vee \neg x_4)$$

Uma atribuição de variáveis possível é $\{x_1, x_3, \neg x_4\}$, então F é *satisfazível*.



Forma Normal Conjuntiva

Na lógica booleana, uma fórmula está em Forma Normal Conjuntiva do inglês *Conjunctive Normal Form* (CNF) se é uma conjunção (\wedge) de cláusulas, onde uma cláusula é uma disjunção (\vee) de literais.

$$(x_1 \leftrightarrow x_2) \rightarrow (\neg x_1 \wedge x_3) = (x_1 \vee x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3)$$

Qualquer fórmula booleana F pode-se converter a CNF.



Problema *Hitting Set*

O *Hitting Set problem* (HSP) pode ser definido como: Dado um conjunto finito \mathbb{X} e a coleção \mathcal{S} de subconjuntos de \mathbb{X} , deve-se achar um subconjunto $H \subseteq \mathbb{X}$ com a menor cardinalidade, tal que:

$$H \cap S \neq \emptyset, \forall S \in \mathcal{S} \quad (3)$$

Mais de um subconjunto de \mathbb{X} pode satisfazer as condições da cima. Nos chamamos de $\mathcal{H} = \{H_1, H_2, \dots, H_{|\mathcal{H}|}\}$ a coleção de possíveis soluções.



Problema *Hitting Set*

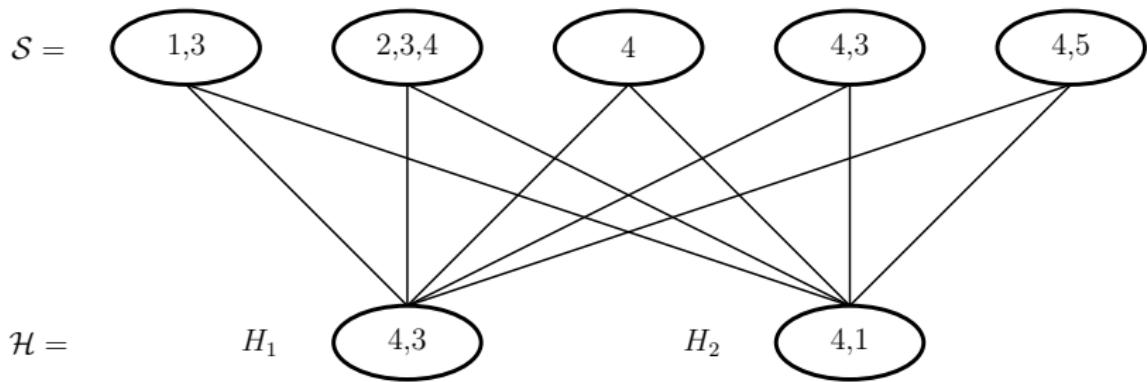


Figura: Exemplo de uma instância HSP. Coleção de subconjuntos e suas possíveis soluções.

Algoritmo de Dubrova e Teslenko

O algoritmo de Dubrova e Teslenko adota uma representação simbólica para as transições de estado, que correspondem a arestas no GTE. Isso permite detectar a existência de caminhos no GTE por meio da avaliação da satisfatibilidade de expressões lógicas, com a finalidade de achar vértices repetidos nos caminhos do GTE, os quais conformam os *atratores*.



Algoritmo de Dubrova e Teslenko

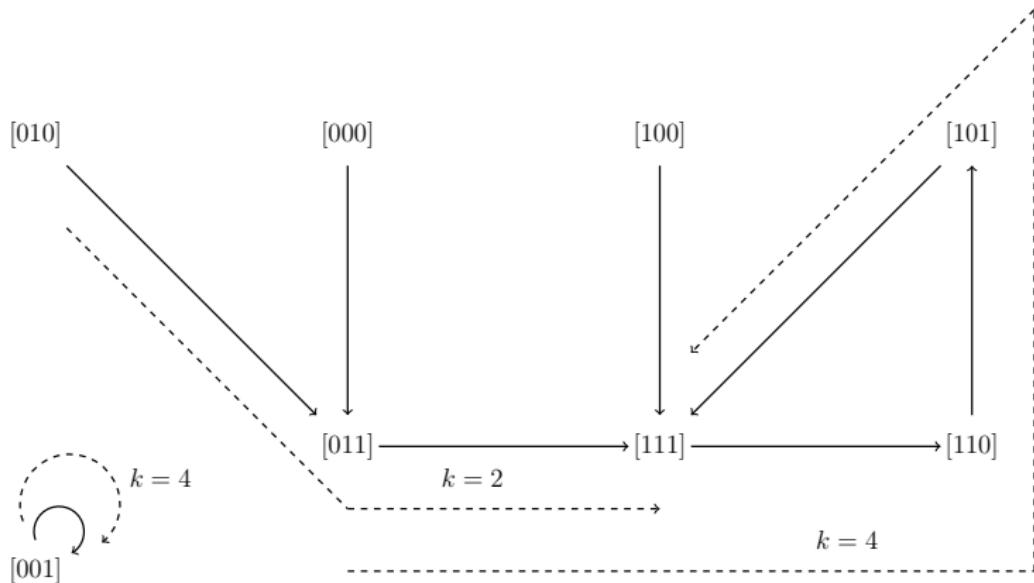


Figura: Grafo de transições de estados de uma rede Booleana de 3 variáveis.



Algoritmo de Dubrova e Teslenko

Denotamos por S_k uma sequência de transições de estados:

$$x^{t+0} \rightarrow x^{t+1} \rightarrow x^{t+2} \rightarrow \dots \rightarrow x^{t+k} \quad (4)$$

Uma transição de estado S_1 , $x^t \rightarrow x^{t+1}$, de acordo com o método de representação simbólica de [Burch et al., 1992] é dada por:

$$[x_1^{t+1} \leftrightarrow f_1(x^t)] \wedge [x_2^{t+1} \leftrightarrow f_2(x^t)] \wedge \dots \wedge [x_n^{t+1} \leftrightarrow f_n(x^t)] \quad (5)$$

Esta representação da transição S_1 é chamada de *formulação proposicional* de S_1 , a qual denotamos por \widehat{S}_1 .



Algoritmo de Dubrova e Teslenko

Para criar formulação proposicional \hat{S}_2 , se adiciona a transição S_1 , $x^{t+1} \rightarrow x^{t+2}$ com o operador (\wedge) a \hat{S}_1 , assim:

$$[x_1^{t+1} \leftrightarrow f_1(x^t)] \wedge [x_2^{t+1} \leftrightarrow f_2(x^t)] \wedge \dots \wedge [x_n^{t+1} \leftrightarrow f_n(x^t)] \wedge$$

$$[x_1^{t+2} \leftrightarrow f_1(x^{t+1})] \wedge [x_2^{t+2} \leftrightarrow f_2(x^{t+1})] \wedge \dots \wedge [x_n^{t+2} \leftrightarrow f_n(x^{t+1})]$$



Algoritmo de Dubrova e Teslenko

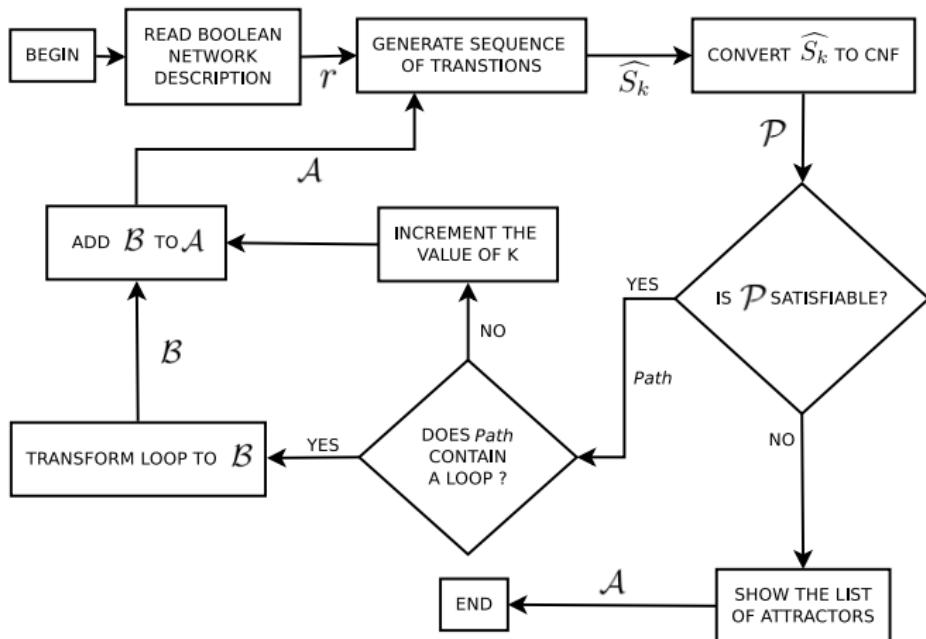


Figura: Fluxograma do algoritmo de Dubrova.



Algoritmo de Dubrova e Teslenko

Depois de cada iteração é preciso excluir os atratores achados do espaço de busca, o que é feito através da seguinte operação:

$$\widehat{S}_k \leftarrow \widehat{S}_k \wedge \neg \mathcal{A} \quad (6)$$

Onde a lista de atratores \mathcal{A} tem seus valores na Forma Normal Disjuntiva do inglês *Disjunctive Normal Form* (DNF) se é uma disjunção (\vee) de cláusulas, onde uma cláusula é uma conjunção (\wedge) de literais. Na qual os atratores são as clausulas e os valores de cada variável são os literais.



Seção 3 | Métodos propostos



Métodos Propostos

Nós dividimos a pesquisa em duas etapas:

- Passo 1: Calcular os atratores locais das RDDs que são parte de uma RDDA:
 - ① Abordagem Baseado no HSP
 - ② Abordagem Baseado no SAT
- Passo 2: Achar os campos atratores dentro da RDDA:



Uma Abordagem Baseada em Hitting Set

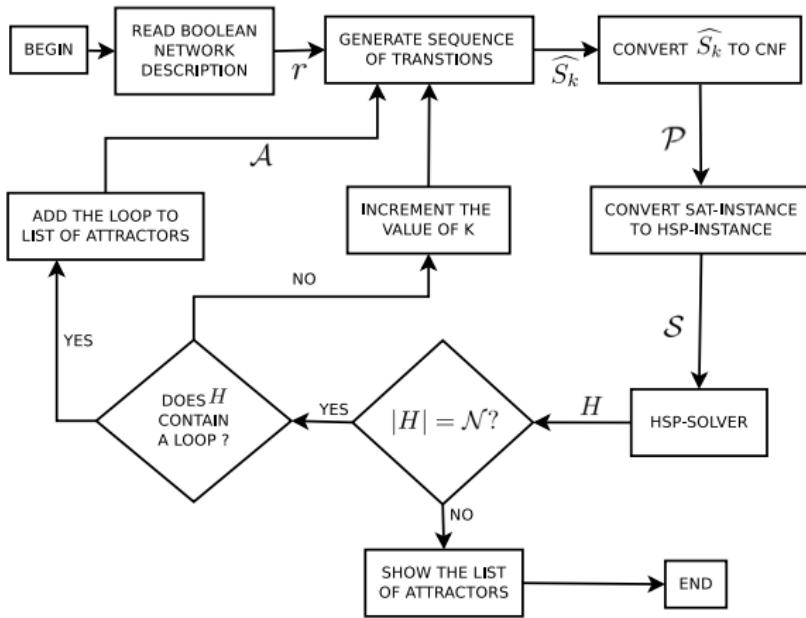


Figura: Fluxograma do algoritmo com HSP.



Uma redução simples e linear do problema SAT para HSP

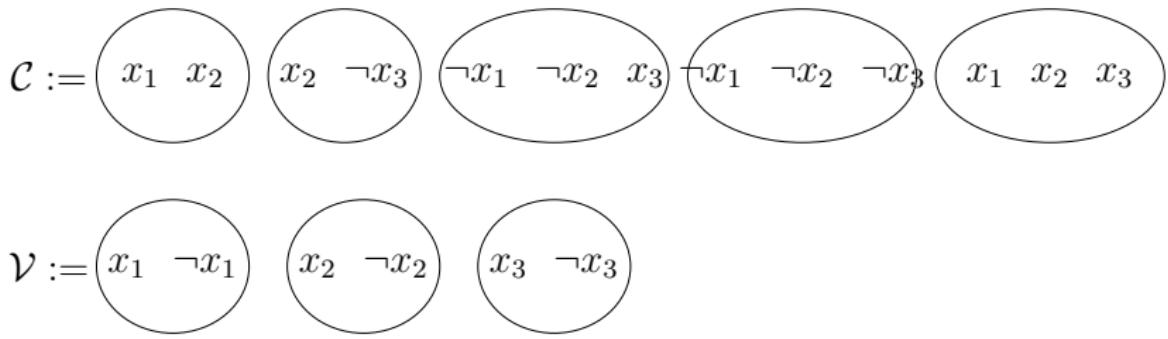


Figura: Exemplo de conjuntos \mathcal{C} e \mathcal{V} cuja união é a instância do HSP de $\mathcal{P} = (x_1 \vee x_2) \wedge (x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_1 \vee x_2 \vee x_3)$. Neste caso, observe que existe uma interpretação $(\neg x_1, x_2, x_3)$ satisfazendo \mathcal{P} pois cada conjunto em $\mathcal{C} \cup \mathcal{V}$ tem pelo menos um literal de $\{\neg x_1, x_2, x_3\}$.



Tratando Sinais de acoplamento em RDDs

Para achar atratores em RDDs que são parte de uma RDDA adaptamos o Algoritmo de Dubrova e Teslenko, além nós levamos as considerações:

- Essas RDDs tem como característica o uso dos sinais de acoplamento que relacionam uma RDD com outra
- Assumimos os sinais de acoplamento com valores fixos 0 ou 1.
- Para representar esta característica criamos uma RDD para cada possível combinação de valores dos sinais de acoplamento, o que gera o total de $2^n m$ RDDs, onde n é o numero de variáveis e m o numero de RDDs iniciais.
- Cada sinal de acoplamento representa somente uma variável externa.
- As funções de acoplamento I estão inclusas dentro da descrição das funções de transição F das RDDs na CNF



Tratando Sinais de acoplamento em RDDs

Para achar atratores em RDDs que são parte de uma RDDA adaptamos o Algoritmo de Dubrova e Teslenko, além nós levamos as considerações:

- Essas RDDs tem como característica o uso dos sinais de acoplamento que relacionam uma RDD com outra
- Assumimos os sinais de acoplamento com valores fixos 0 ou 1.
- Para representar esta característica criamos uma RDD para cada possível combinação de valores dos sinais de acoplamento, o que gera o total de $2^n m$ RDDs, onde n é o numero de variáveis e m o numero de RDDs iniciais.
- Cada sinal de acoplamento representa somente uma variável externa.
- As funções de acoplamento I estão inclusas dentro da descrição das funções de transição F das RDDs na CNF



Tratando Sinais de acoplamento em RDDs

Para achar atratores em RDDs que são parte de uma RDDA adaptamos o Algoritmo de Dubrova e Teslenko, além nós levamos as considerações:

- Essas RDDs tem como característica o uso dos sinais de acoplamento que relacionam uma RDD com outra
- Assumimos os sinais de acoplamento com valores fixos 0 ou 1.
- Para representar esta característica criamos uma RDD para cada possível combinação de valores dos sinais de acoplamento, o que gera o total de $2^n m$ RDDs, onde n é o numero de variáveis e m o numero de RDDs iniciais.
- Cada sinal de acoplamento representa somente uma variável externa.
- As funções de acoplamento I estão inclusas dentro da descrição das funções de transição F das RDDs na CNF



Tratando Sinais de acoplamento em RDDs

Para achar atratores em RDDs que são parte de uma RDDA adaptamos o Algoritmo de Dubrova e Teslenko, além nós levamos as considerações:

- Essas RDDs tem como característica o uso dos sinais de acoplamento que relacionam uma RDD com outra
- Assumimos os sinais de acoplamento com valores fixos 0 ou 1.
- Para representar esta característica criamos uma RDD para cada possível combinação de valores dos sinais de acoplamento, o que gera o total de $2^n m$ RDDs, onde n é o numero de variáveis e m o numero de RDDs iniciais.
- Cada sinal de acoplamento representa somente uma variável externa.
- As funções de acoplamento / estão inclusas dentro da descrição das funções de transição F das RDDs na CNF



Tratando Sinais de acoplamento em RDDs

Para achar atratores em RDDs que são parte de uma RDDA adaptamos o Algoritmo de Dubrova e Teslenko, além nós levamos as considerações:

- Essas RDDs tem como característica o uso dos sinais de acoplamento que relacionam uma RDD com outra
- Assumimos os sinais de acoplamento com valores fixos 0 ou 1.
- Para representar esta característica criamos uma RDD para cada possível combinação de valores dos sinais de acoplamento, o que gera o total de $2^n m$ RDDs, onde n é o numero de variáveis e m o numero de RDDs iniciais.
- Cada sinal de acoplamento representa somente uma variável externa.
- As funções de acoplamento / estão inclusas dentro da descrição das funções de transição F das RDDs na CNF



Tratando Sinais de acoplamento em RDDs

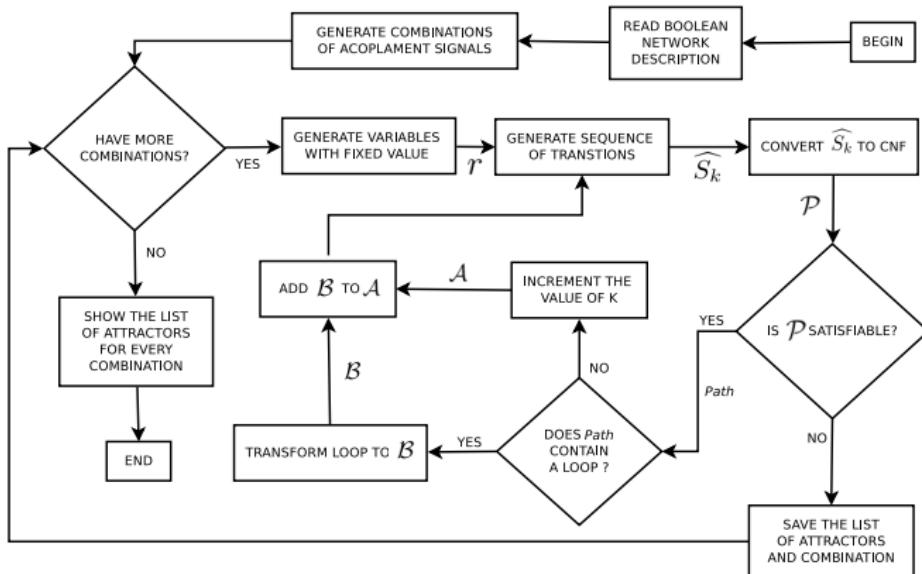


Figura: Fluxograma do algoritmo com SAT para RDDA.



Sincronização e Estabilidade em RDDAs

Campos de Atractores

A_1 $(1, a_1), (2, a_3), (3, a_2)$

A_2 $(1, a_4), (2, a_1), (3, a_3)$

Conjunto de Atractores

a_1, a_2, a_3, a_4, a_5

a_1, a_2, a_3, a_4, a_5

a_1, a_2, a_3, a_4, a_5

\mathcal{A}^1

\mathcal{A}^2

\mathcal{A}^3

RDDs

1

2

3

Figura: Campos de atratores cujas saídas e entradas sejam compatíveis.



Seção 4 | Resultados Preliminares



Ferramentas Utilizadas

- Os experimentos foram executados usando sistema operacional Ubuntu 18.04 em uma área de trabalho com processador Intel Core(TM) i7-3930K, processador de 3.17GHz, 12 CPUs, 32 GB de RAM e dois Nvidia GK 104 GeForce GTX boards.
- A implementação desta abordagem foi feita no Python 2.7.
- Se fez uso da biblioteca Satisfy a qual se encarga de deixar a formulação proposicional na CNF.
- O HSP-solver utilizado foi o feito por Danilo Carastan-Santos [Carastan-Santos et al., 2017].
- O SAT-solver utilizado foi o Minisat [Eén and Sörensson, 2003].



Resultados do abordagem Baseada em HSP

Foram feitos dois experimentos um de 3 e outro de 4 variáveis, para o experimento com 3 variáveis o tempo de execução foi de 3,55 segundos, mas para o experimento de rede de 4 variáveis, o tempo de execução foi de 9,15 horas.

Esse crescimento no tempo de execução depende do HSP-solver [Carastan-Santos et al., 2017], já que as RDDs com o algoritmo de Dubrova e Teslenko gera uma grande quantidade de clausulas, o que representa a entrada do pior caso para o HSP-Solver.



Resultados do abordagem Baseada em SAT

Gerador de RDDAs

Foi feito um gerador de RDDAs que tem como parâmetros:

- quantidade de redes.
- número de variáveis por rede.
- número de sinais de acoplamento.

O gerador foi configurado para que as RDDs tenham número aleatório de cláusulas nas funções F , que está no intervalo de 1 a 4, também foi usando um número aleatório para o número de variáveis por clausula, o qual esta no intervalo de 2 a 4.



Resultados do abordagem Baseada em SAT

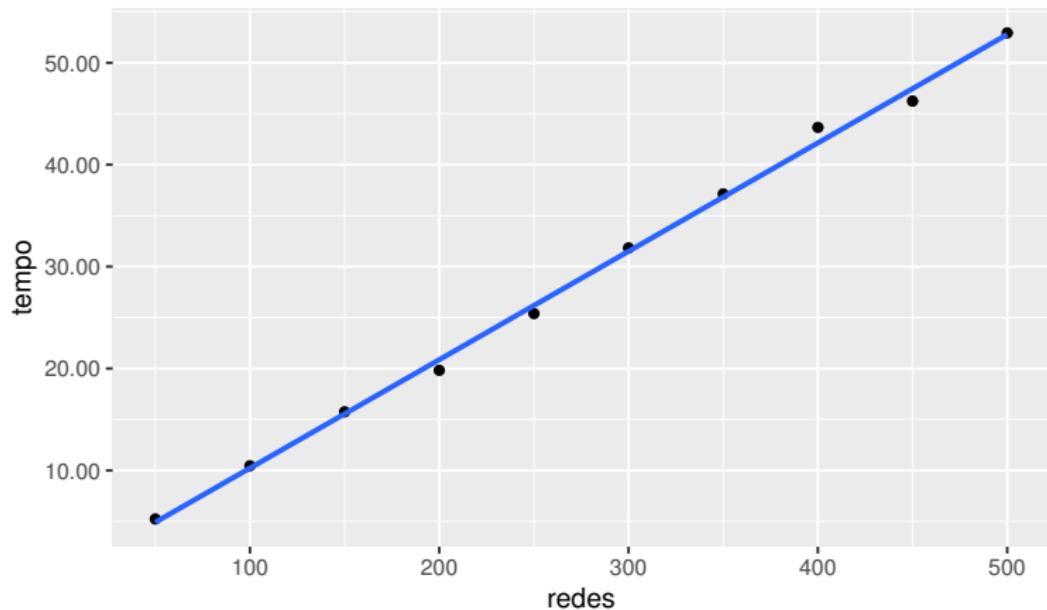


Figura: Gráfico da tabela 1



Resultados do abordagem Baseada em SAT

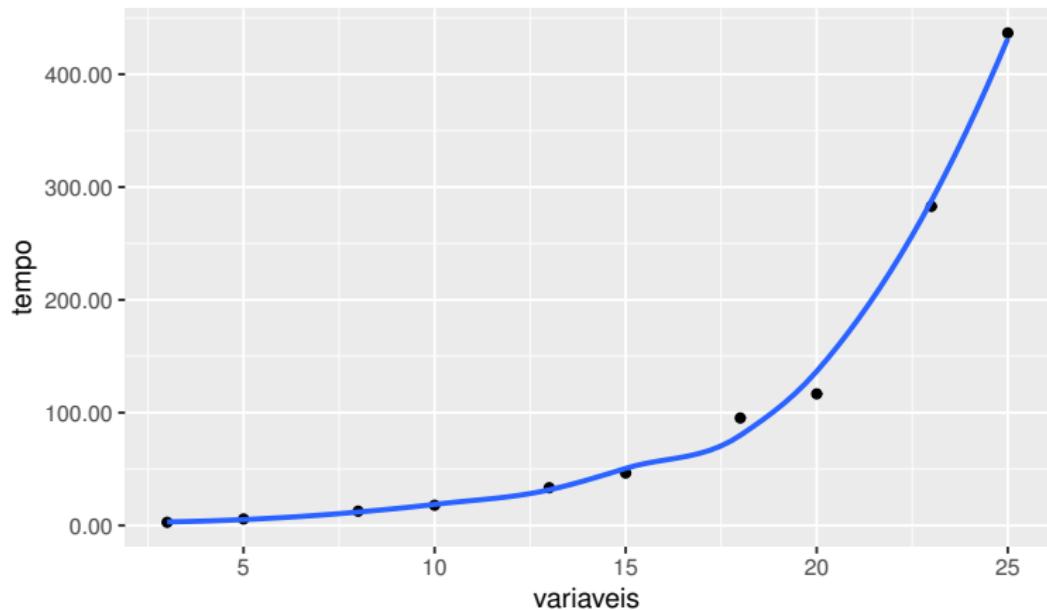


Figura: Gráfico da tabela 2



Resultados do abordagem Baseada em SAT

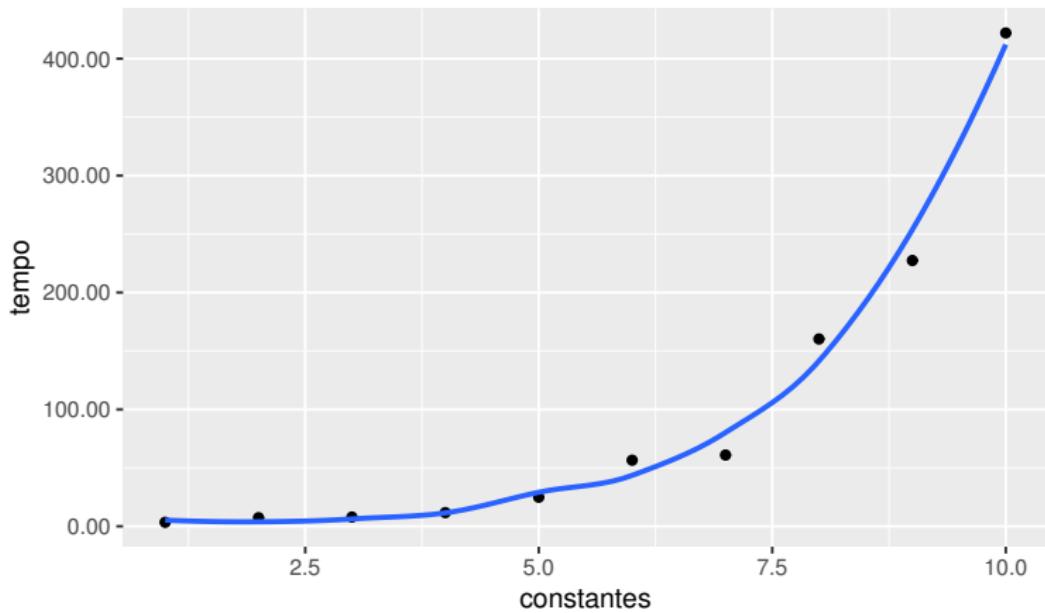


Figura: Gráfico da tabela 3



Seção 5 | Conclusões parciais



Contribuições

- Desenvolvemos uma modelagem de sistemas de entidades interagentes por redes dinâmicas discretas acopladas (RDDAs).
- Desenvolvemos e implementamos uma abordagem inovadora para o problema de detecção de atratores em redes Booleanas:
 - Adaptamos o algoritmo de Dubrova e Teslenko (2011), de modo a combiná-lo com o algoritmo de Carastan-Santos et al. (2017);
 - Desenvolvemos uma redução $[SAT \longrightarrow \text{Hitting Set}]$, o que é um resultado teórico.
- Implementamos a abordagem original (baseada em SAT “clássico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos.



Contribuições

- Desenvolvemos uma modelagem de sistemas de entidades interagentes por redes dinâmicas discretas acopladas (RDDAs).
- Desenvolvemos e implementamos uma abordagem inovadora para o problema de detecção de atratores em redes Booleanas:
 - Adaptamos o algoritmo de Dubrova e Teslenko (2011), de modo a combiná-lo com o algoritmo de Carastan-Santos et al. (2017);
 - Desenvolvemos uma redução $[SAT \longrightarrow \text{Hitting Set}]$, o que é um resultado teórico.
- Implementamos a abordagem original (baseada em SAT “clássico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos.



Contribuições

- Desenvolvemos uma modelagem de sistemas de entidades interagentes por redes dinâmicas discretas acopladas (RDDAs).
- Desenvolvemos e implementamos uma abordagem inovadora para o problema de detecção de atratores em redes Booleanas:
 - Adaptamos o algoritmo de Dubrova e Teslenko (2011), de modo a combiná-lo com o algoritmo de Carastan-Santos et al. (2017);
 - Desenvolvemos uma redução $[SAT \longrightarrow \text{Hitting Set}]$, o que é um resultado teórico.
- Implementamos a abordagem original (baseada em SAT “clássico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos.



Contribuições

- Desenvolvemos uma modelagem de sistemas de entidades interagentes por redes dinâmicas discretas acopladas (RDDAs).
- Desenvolvemos e implementamos uma abordagem inovadora para o problema de detecção de atratores em redes Booleanas:
 - Adaptamos o algoritmo de Dubrova e Teslenko (2011), de modo a combiná-lo com o algoritmo de Carastan-Santos et al. (2017);
 - Desenvolvemos uma redução [SAT \rightarrow Hitting Set], o que é um resultado teórico.
- Implementamos a abordagem original (baseada em SAT “clássico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos.



Contribuições

- Desenvolvemos uma modelagem de sistemas de entidades interagentes por redes dinâmicas discretas acopladas (RDDAs).
- Desenvolvemos e implementamos uma abordagem inovadora para o problema de detecção de atratores em redes Booleanas:
 - Adaptamos o algoritmo de Dubrova e Teslenko (2011), de modo a combiná-lo com o algoritmo de Carastan-Santos et al. (2017);
 - Desenvolvemos uma redução [SAT \rightarrow Hitting Set], o que é um resultado teórico.
- Implementamos a abordagem original (baseada em SAT “clássico”) proposta por Dubrova e Teslenko; porém, adaptada de modo a considerar a presença de sinais externos.



Limitações e Desafios

- Limitação: O crescimento no tempo de execução de HSP-solver de Carastan-Santos *et al* em função ao número de clausulas.
- Desafio: Otimizar o programa de HSP-solver para resolver instancias com número de clausulas grandes , maiores a 100.



Limitações e Desafios

- Limitação: O crescimento no tempo de execução de HSP-solver de Carastan-Santos *et al* em função ao número de clausulas.
- Desafio: Otimizar o programa de HSP-solver para resolver instancias com número de clausulas grandes , maiores a 100.



Considerações finais e próximas etapas

Etapa 1: Projeto do algoritmo para computação dos campos atratores estáveis.

Etapa 2: Implementação do algoritmo para computação dos campos atratores estáveis como combinações de atratores locais.

Etapa 3: Implementação distribuída do algoritmo para detecção dos atratores locais.

Etapa 4: Execução dos experimentos com o código integrado (“passo 1” e “passo 2”) e análise dos resultados.

Etapa 5: Redação e submissão de artigo 2, artigo 1 pronto para submissão.

Etapa 6: Redação da dissertação e defesa.

Considerações finais e próximas etapas

Tabela: Cronograma estimado de desenvolvimento das próximas etapas.

Atividades/Mês	Etapa 1	Etapa 2	Etapa 3	Etapa 4	Etapa 5	Etapa 6
07/2019	X	X	X			
08/2019		X	X	X		
09/2019				X	X	
10/2019					X	X
11/2019						X

Agradecimentos

- Universidade Federal do ABC.
- Carlos Fernando Montoya Cubas (UFABC) pela discussões frutíferas.
- Suportado pela CAPES.



Referências

-  Arenas, A., Díaz-Guilera, A., Kurths, J., Moreno, Y., and Zhou, C. (2008).
Synchronization in complex networks.
Physics Reports, 469:93–153.
-  Burch, J. R., Clarke, E. M., McMillan, K. L., Dill, D. L., and Hwang, L.-J. (1992).
Symbolic model checking: 1020 states and beyond.
Information and computation, 98(2):142–170.
-  Carastan-Santos, D., de Camargo, R. Y., Martins Jr, D. C., Song, S. W., and Rozante, L. C. (2017).
Finding exact hitting set solutions for systems biology applications using heterogeneous gpu clusters.
Future Generation Computer Systems, 67:418–429.
-  Dubrova, E. and Teslenko, M. (2011).
A sat-based algorithm for finding attractors in synchronous boolean networks.
IEEE/ACM Transactions on Computational Biology and Bioinformatics, 8(5):1393–1399.
-  Eén, N. and Sörensson, N. (2003).
An extensible sat-solver.
In *International conference on theory and applications of satisfiability testing*, pages 502–518. Springer.
-  Harvard-University (2002).
Synchronization of metronomes.
-  Kauffman, S. A. (1993).
The Origins of Order.