

REDES DE HOPFIELD

Trabalho de Inteligência Artificial

Alunos:

Bruno Mamede de Araújo Moura¹

Gustavo Henriques da Cunha¹

Vitor Hugo Palmié Peixoto¹

¹Departamento de Ciência da Computação - DCOMP
Coordenadoria do Curso de Ciência da Computação - CCOMP
Universidade Federal de São João del Rei - UFSJ

Agosto de 2024

- 1 Introdução
 - Contextualização
- 2 Arquitetura Utilizada
 - Redes de Hopfield
 - Redes de Hopfield-Tank
- 3 Problema do Caixeiro-Viajante
- 4 Modelagem do Problema
- 5 Implementação
- 6 Testes e Resultados
 - Testes
 - Resultados
- 7 Conclusão

Sumário

- 1 Introdução
 - Contextualização
- 2 Arquitetura Utilizada
 - Redes de Hopfield
 - Redes de Hopfield-Tank
- 3 Problema do Caixeiro-Viajante
- 4 Modelagem do Problema
- 5 Implementação
- 6 Testes e Resultados
 - Testes
 - Resultados
- 7 Conclusão

Considerações Iniciais

- Com o avanço da Inteligência Artificial na solução de diversos problemas, um campo de estudo de grande destaque é o das Redes Neurais Artificiais.
- Simulando os neurônios no cérebro humano, os algoritmos são boas ferramentas para resolver diversos problemas.
- Buscando entender melhor o funcionamento desses algoritmos, este trabalho estuda o Modelo de Hopfield [1] e uma variação, o modelo de Hopfield-Tank [2] [3], aplicado ao Problema do Caixeiro-Viajante.

Sumário

- 1 Introdução
 - Contextualização
- 2 Arquitetura Utilizada
 - Redes de Hopfield
 - Redes de Hopfield-Tank
- 3 Problema do Caixeiro-Viajante
- 4 Modelagem do Problema
- 5 Implementação
- 6 Testes e Resultados
 - Testes
 - Resultados
- 7 Conclusão

- As Redes de Hopfield pertencem a classe de Redes Neurais Recorrentes.
- Opera com conexões bidirecionais com pesos simétricos [4].
- Também pode ser classificada como uma Rede Neural de Memória Autoassociativa.

Redes de Hopfield

- Os nós ou neurônios, têm dois estados, indicando se o neurônio está "disparando" ou não.
- Dado uma configuração inicial, a rede, de forma aleatória e assíncrona, atualiza seus valores para cada neurônio.
- A rede funciona como uma memória dos dados.

Redes de Hopfield-Tank

- Evolução das Redes de Hopfield tradicionais, que são discretas.
- Trabalha em um espaço de decisão contínuo.
- Cada neurônio muda seu estado analógico de forma contínua e simultânea em paralelo.
- Através de equações de evolução e energia, a carga dos neurônios é atualizada e a energia converge para um mínimo.

Sumário

- 1 Introdução
 - Contextualização
- 2 Arquitetura Utilizada
 - Redes de Hopfield
 - Redes de Hopfield-Tank
- 3 Problema do Caixeiro-Viajante**
- 4 Modelagem do Problema
- 5 Implementação
- 6 Testes e Resultados
 - Testes
 - Resultados
- 7 Conclusão

Problema do Caixeiro-Viajante

- O problema do caixeiro-viajante (PCV) é um problema de otimização combinatória.
- É um problema difícil de resolver, pertencente à classe NP-completo.
- É comum usá-lo para testar implementações de heurísticas.

Problema do Caixeiro-Viajante

- Sendo $G = (V, A)$ um grafo com n vértices V e arestas A com um certo peso, é preciso encontrar um circuito de menor distância possível que passe por cada vértice uma única vez e retorne ao vértice inicial.
- A figura 1 mostra uma instância do PCV com 49 cidades [5].

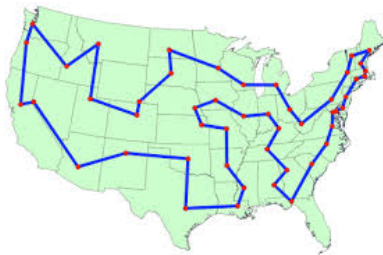


Figura 1: PCV com 49 cidades.

Sumário

- 1 Introdução
 - Contextualização
- 2 Arquitetura Utilizada
 - Redes de Hopfield
 - Redes de Hopfield-Tank
- 3 Problema do Caixeiro-Viajante
- 4 Modelagem do Problema**
- 5 Implementação
- 6 Testes e Resultados
 - Testes
 - Resultados
- 7 Conclusão

Modelagem do Problema

- Para modelar o PCV para a rede de Hopfield-Tank é necessário mapear os neurônios para uma representação da solução.
- Seguindo os autores do método [2], é usada uma matriz quadrada com o número de cidades, onde cada linha representa uma cidade e cada coluna uma posição no ciclo.

Modelagem do Problema

- A figura 2 mostra uma matriz de neurônios mapeada para uma solução [3].

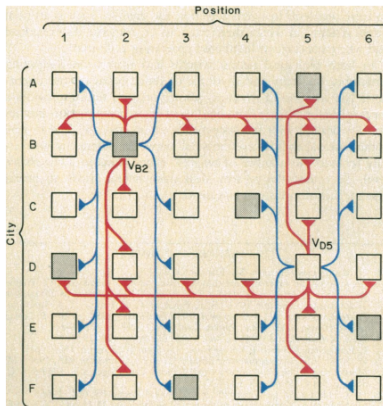


Figura 2: Representação da matriz de neurônios para uma solução.

Modelagem do Problema

- Em seguida, deve-se definir a função de energia.
- Os autores [2] sugerem a Equação 1, onde A , B , C e D são constantes que são definidas antes da execução do código.

$$\begin{aligned} E = & \frac{A}{2} \sum_X \sum_i \sum_{j \neq i} V_{Xi} V_{Xj} \\ & + \frac{B}{2} \sum_i \sum_X \sum_{X \neq Y} V_{Xi} V_{Yi} \\ & + \frac{C}{2} \left(\sum_X \sum_i V_{Xi} - n \right)^2 \\ & + \frac{D}{2} \sum_X \sum_{Y \neq X} \sum_i d_{XY} V_{Xi} (V_{Yi+1} + V_{Yi-1}). \end{aligned} \quad (1)$$

Modelagem do Problema

- Também é definida a fórmula para os pesos da matriz de conexões, dada pela Equação 2 [2].

$$\begin{aligned} E = & -A\delta_{XY}(i - \delta_{ij}) - B\delta_{ij}(1 - \delta_{XY}) \\ & - C - Dd_{XY}(\delta_{j,i+1} + \delta_{j,i-1}) \\ & [\delta_{i,j} = 1 \text{ se } i = j \text{ e } 0 \text{ caso contrário}]. \end{aligned} \quad (2)$$

Modelagem do Problema

- As constantes A , B e C fornecem as restrições básicas para o problema do PCV. Com a função de energia E , dado um estado inicial, o circuito deve convergir para uma matriz de permutação, como ilustrado na Figura 2, que representa uma solução para o problema.

Sumário

- 1 Introdução
 - Contextualização
- 2 Arquitetura Utilizada
 - Redes de Hopfield
 - Redes de Hopfield-Tank
- 3 Problema do Caixeiro-Viajante
- 4 Modelagem do Problema
- 5 Implementação**
- 6 Testes e Resultados
 - Testes
 - Resultados
- 7 Conclusão

- Para a realização dos testes neste trabalho, foi utilizada a biblioteca 'Netomaton' [6], desenvolvida por Luis Antunes para a linguagem Python. Esta implementação segue como base o artigo de Hopfield e Tank e possibilita o ajuste dos parâmetros antes da execução.

Sumário

- 1 Introdução
 - Contextualização
- 2 Arquitetura Utilizada
 - Redes de Hopfield
 - Redes de Hopfield-Tank
- 3 Problema do Caixeiro-Viajante
- 4 Modelagem do Problema
- 5 Implementação
- 6 Testes e Resultados**
 - Testes
 - Resultados
- 7 Conclusão

- Para a realização dos testes, foi utilizada uma instância pequena, com 10 cidades, o mesmo tamanho utilizado por Hopfield e Tank em seu artigo.
- O primeiro teste foi realizado utilizando os mesmos parâmetros que os autores encontraram em seu artigo [2], com $A = B = 500$, $C = 200$, $D = 500$ e $n = 15$.
- Outros três testes foram realizados com diferentes parâmetros
- Por fim, os resultados foram comparados com testes feitos usando a metaheurística *Simulated Annealing*.

- A tabela 1 ilustra o resultado dos testes realizados com a média de distância encontrada com 5 execuções.

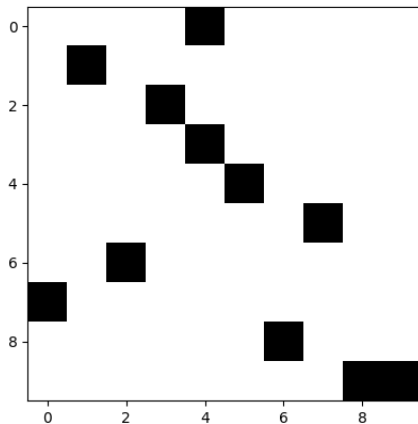
A	B	C	D	n	Média
500	500	200	500	15	3,06
300	300	100	300	12	3,09
400	400	150	400	12	3,23
500	500	200	500	15	3,97

Tabela 1: Tabela com os resultados dos testes.

- É importante observar que, embora a rede sempre convirja para um ponto de energia mínimo, por trabalhar com soluções inválidas, não é garantido que o resultado será válido.
- Para a construção da tabela, foram consideradas apenas as cinco execuções com soluções válidas.

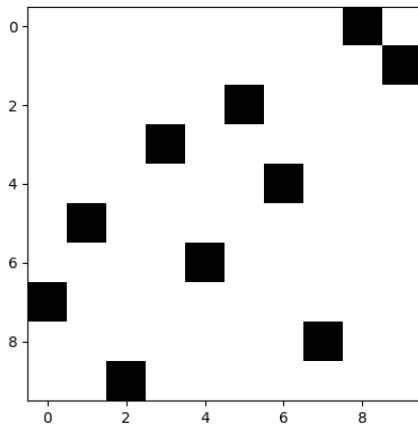
Resultados

- A Figura 3 mostra como fica uma matriz de neurônios em uma solução inválida.



Resultados

- A Figura 4 mostra como fica uma matriz de neurônios em uma solução válida.



Comparação com o *Simulated Annealing*

- Fazendo os mesmos testes com o *Simulated Annealing* com parâmetros já calibrados, foi constantemente obtido o valor de 2,55.
- Sendo assim, a rede neural não foi tão boa para o problema.

Sumário

- 1 Introdução
 - Contextualização
- 2 Arquitetura Utilizada
 - Redes de Hopfield
 - Redes de Hopfield-Tank
- 3 Problema do Caixeiro-Viajante
- 4 Modelagem do Problema
- 5 Implementação
- 6 Testes e Resultados
 - Testes
 - Resultados
- 7 Conclusão

Conclusão

- Pelos resultados alcançados, é possível concluir que a rede de Hopfield-Tank não é o método mais adequado para a solução do Problema do Caixeiro Viajante.
- No entanto, o desenvolvimento de redes neurais para problemas de otimização não deve ser desencorajado. Neste trabalho, foi utilizada uma das primeiras versões do algoritmo, e desde então, várias otimizações e melhorias foram realizadas, além do surgimento de novas arquiteturas.
- As redes neurais possuem uma alta capacidade de paralelização e podem funcionar bem em outros problemas.

Referências Bibliográficas

- [1] John J. Hopfield. “Neural networks and physical systems with emergent collective computational abilities”. Em: *Proceedings of the National Academy of Sciences of the United States of America* 79 (1982), pp. 2554–2558.
- [2] John J. Hopfield e David W. Tank. ““Neural” computation of decisions in optimization problems”. Em: *Biol Cybern* 52 (1985), pp. 141–152.
- [3] John J. Hopfield e David W. Tank. “Computing with Neural Circuits: A Model”. Em: *Science* 233 (1986), pp. 625–633.
- [4] Stuart J. Russell e Peter Norvig. *Artificial Intelligence: A Modern Approach*. 3rd. Upper Saddle River, New Jersey 07458.: Prentice Hall, 2010.

- [5] Usman Lateef, Rufai Idowu, Olayinka Olusanya e Adedeji Oluwaseun. “Solving Travelling Salesman Problem Using an Improved Ant Colony Optimization Algorithm”. Em: - 6 (jun. de 2021), pp. 158–170.
- [6] Luis Antunes. *netomaton*.
<https://github.com/lantunes/netomaton/tree/master>. 2021.