

Rodrigo Mendonça da Paixão  
Lucas Teles Agostinho

**Titulo a definir**

São Paulo – Brasil

2016

Rodrigo Mendonça da Paixão  
Lucas Teles Agostinho

## **Titulo a definir**

Pré-monografia apresentada na disciplina Trabalho de Conclusão de Curso I, como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação.

Centro Universitário Senac  
Bacharelado em Ciência da Computação

Orientador: Eduardo Heredia

São Paulo – Brasil

2016

# Lista de abreviaturas e siglas

GA	Algoritmos Genéticos
----	----------------------

# Sumário

1	INTRODUÇÃO . . . . .	4
1.1	Motivação . . . . .	4
1.2	Escopo . . . . .	4
1.3	Justificativa . . . . .	4
1.4	Objetivos . . . . .	4
1.5	Método de trabalho . . . . .	4
1.6	Organização do trabalho . . . . .	4
2	REVISÃO DE LITERATURA . . . . .	5
2.1	Algoritmos Genéticos para busca de caminhos . . . . .	5
3	PROPOSTA . . . . .	7
4	CRONOGRAMA . . . . .	8
	REFERÊNCIAS . . . . .	9

# 1 Introdução

1.1 Motivação

1.2 Escopo

1.3 Justificativa

1.4 Objetivos

1.5 Método de trabalho

1.6 Organização do trabalho

## 2 Revisão de Literatura

### 2.1 Algoritmos Geneticos para busca de caminhos

Foi demonstrado que algoritmos genéticos paralelos são eficientes para a resolução de problemas de busca de caminho, tal como o classico problema do caixeiro viajante, que consiste em dado um numero finito de cidades com seus custos de viagem entre elas, deve-se encontrar o caminho mais curto para viajar entre todas as cidades e voltar ao ponto inicial, o problema pode ser representado pelo modelo de um grafo direcionado ponderado, aplicando a mesma ideia, o problema seria encontrar o caminho de menor custo para percorrer todos os nós, de maneira analoga, as cidades seriam os nós e a distancia entre elas o peso das arestras (D.LOHN SILVANO P. COLOMBANO, 2000)(ALAOU; EL-GHAZAWI, 2000)(MUHLENBEIN, 2000).

\*NOTA\* Decorrer do POR QUE a escolha do uso de AG \*NOTA\*

A solução para este tipo de problema pode requer uma quantidade grande de processamento. Uma boa solução seria dividir o processamento do problema em pequenas partes e distribuir cada parte para um processador a parte, trabalhando de forma distribuida ou paralela. Varios modelos para essa finalidade foram propostos.

Um modelo interessante para parelização seria o de mestre-escravo, onde o mestre fica responsável na manutenção da população e execução dos operadores genéticos. A avaliação dos melhores indivíduos é distribuída para os demais escravos, O mestre envia um indivíduo a cada um dos escravos subjacentes. Cada escravo realiza a interpretação do problema, aplica a função de cálculo para a escolha dos melhores indivíduos e envia seus resultados ao mestre, que executa seleção dos indivíduos e a geração da nova população, repetindo o processo como um todo. Essa estrutura teve implicação satisfatoria para a automação de design de circuitos eletrônicos. (D.LOHN SILVANO P. COLOMBANO, 2000)

Outra forma de trabalhar com o modelo de mestre escravo, seria definir que cada um dos nós escravos subjacentes fica responsável por sua própria população. O nó central mestre, cria as populações iniciais e as distribui para os nós escravos. Cada nó escravo processa a evolução da população por um determinado número de gerações e então a submete ao mestre. O mestre então seleciona os melhores indivíduos dentre todas as populações dos nós escravos e os distribui novamente. Em cada nó escravo, os novos indivíduos distribuídos pelo mestre são inseridos na população corrente e o processo de evolução recomeça. A migração entre os escravos, que é controlado pelo nó mestre, implementa o mecanismo que regula a velocidade da convergência e oferece os meios de

escape dos mínimos locais. Entretanto a migração das populações dos nós escravos para o mestre e vice versa pode impor um certo grau de sobre carga, dependente do meio de comunicação entre os nós. Esse modelo obteve sucesso no mapeamento de tarefas em máquinas paralelas. (ALAOU; EL-GHAZAWI, 2000)

Podemos partir do ponto que cada indivíduo é o responsável por encontrar e reproduzir com um parceiro em sua vizinhança. O controle de seleção e reprodução se espalha pela população e o algoritmo deixa de ser centralizado em um mestre, com isso, diminui o grau de sincronização e facilita a paralelização. O processo do algoritmo é definir uma representação genética para o problema e criar a estrutura de vizinhança e sua população inicial. Cada indivíduo faz uma busca em sua vizinhança e seleciona um parceiro para a reprodução. Uma geração descendente é criada com o operador genético resultante. (MUHLENBEIN, 2000)

Podemos observar alguns problemas nos modelos apresentados (MOLE, 2002), no modelo de (D.LOHN SILVANO P. COLOMBANO, 2000) existe problema em explorar o paralelismo no calculo de verificação dos indivíduos não explorando para a reprodução e mutação. No modelo de (MUHLENBEIN, 2000), tem a possibilidade de utilizar vários métodos de busca de indivíduos da mesma população, sendo úteis em casos que a eficiência dos métodos de busca se mostram dependentes da instancia do problema. O modelo (ALAOU; EL-GHAZAWI, 2000), por todos os escravos devem enviar para o nó mestre, demanda uma grande capacidade de processamento no nó mestre, e proporciona a divisão das populações em pequenas ou de médio porte.

(MOLE, 2002) desenvolveu seu próprio modelo, utilizando o modelo de (ALAOU; EL-GHAZAWI, 2000) como inspiração. O modelo segue o conceito mestre-escravo, o mestre cria as populações e distribui a cada uma delas, os conjuntos de genes e parâmetros iniciais. O mestre é utilizado para a troca de indivíduos entre as populações, mantendo um indivíduo de cada população ate serem substituídos por um melhor e envia esses indivíduos para as populações que não seja a sua de origem. As populações são independentes, gerando seus indivíduos inicial com base nos genes enviados pelo mestre, aplicando seus próprios operadores de evolução e a população que determina os parceiros dos indivíduos.

## 3 Proposta



## 4 Cronograma

Para modelar a proposta descrita, sera seguido um cronograma semanal, este ira descrever semana a semana as tarefas que devem ser realizadas de forma a se concluir o trabalho.

Semana	Atividade
1 <sup>a</sup>	Analise de trafego de pacotes em rede, criar aplicação em Go para ler e realizar log das operações na rede
2 <sup>a</sup>	Criar base de dados de trafego em rede de forma controlada. Verificar como implementações de IDS atuais reagem a esses dados. / trabalhar na monografia
3 <sup>a</sup>	Implementar na aplicação de log de trafego um sistema de redes neurais
4 <sup>a</sup>	Realizar treinamento da RNA da aplicação com os <i>datasets</i> KDD'99 e o modelado para o projeto / trabalhar na monografia
5 <sup>a</sup>	Mensurar resultados da aplicação do modelo / trabalhar na monografia
6 <sup>a</sup>	Comparar com os resultados obtidos por outros IDS, e com resultados publicados de outros modelos baseados em RNA
7 <sup>a</sup>	Analisar se for possível como aprimorar os resultados do modelo
8 <sup>a</sup>	Implementar o modelo de POLVO-IIDS
9 <sup>a</sup>	Comparar com os resultados obtidos anteriormente / trabalhar na monografia
10 <sup>a</sup>	Implementar RNA clusterizado.
11 <sup>a</sup>	Comparar com os resultados obtidos anteriormente / trabalhar na monografia
13 <sup>a</sup>	Implementar modelo hibrido POLVO-IIDS Clusterizado
14 <sup>a</sup>	Comparar com os resultados obtidos anteriormente / trabalhar na monografia
15 <sup>a</sup>	Trabalhar na monografia - desenvolvimento
16 <sup>a</sup>	Trabalhar na monografia - resultados
17 <sup>a</sup>	Trabalhar na monografia - resultados
18 <sup>a</sup>	Apresentação dos resultados

## Referências

ALAOUI, O. F. S. M.; EL-GHAZAWI, T. A parallel genetic algorithm for task mapping on parallel machines. p. 9, 2000. Citado 2 vezes nas páginas 5 e 6.

D.LOHN SILVANO P. COLOMBANO, G. L. H. t. D. S. J. Parallel genetic algorithm for automated electronic circuit design. p. 10, 2000. Citado 2 vezes nas páginas 5 e 6.

MOLE, V. L. D. Algoritmos genéticos – uma abordagem paralela baseada em populações cooperantes. p. 97, 2002. Citado na página 6.

MUHLENBEIN, H. Evolution in time and space - the parallel genetic algorithm. p. 22, 2000. Citado 2 vezes nas páginas 5 e 6.