

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

BRUNO PANERAI VELLOSO

**PROPOSTA DE UMA METODOLOGIA BASEADA NA
TEORIA DA PERCOLAÇÃO PARA CONFIGURAÇÃO DOS
PARÂMETROS DE UM ALGORITMO DE INTELIGÊNCIA
DE ENXAMES**

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Mauro Roisenberg

Florianópolis, agosto de 2008

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

**PROPOSTA DE UMA METODOLOGIA BASEADA NA TEORIA DA
PERCOLAÇÃO PARA CONFIGURAÇÃO DOS PARÂMETROS DE
UM ALGORITMO DE INTELIGÊNCIA DE ENXAMES**

BRUNO PANERAI VELLOSO

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, área de concentração Sistemas de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Prof. Dr. Frank Augusto Siqueira

Banca Examinadora

Orientador: Prof. Dr. Mauro Roisenberg

Prof. Dr. Antonio Carlos da Rocha Costa

Prof. Dr. Ricardo Azambuja Silveira

Profa. Dra. Sivia Modesto Nassar

*”Cautelosas e prudentes,
O caminho atravessando,
As formigas diligentes
Vão andando, vão andando...”
Olavo Bilac*

Agradecimentos

A elaboração deste trabalho não seria possível sem a ajuda de inúmeras pessoas. Não é viável cita-las todas pois corro o risco de cometer involuntárias omissões imperdoáveis. No entanto algumas pessoas merecem meu mais sincero agradecimento, pois, sem a ajuda destas este trabalho se tornaria impossível.

À minha namorada/esposa/amiga/companheira Alessandra Fucolo Baptista sem a qual não encontraria os motivos corretos para elaboração deste trabalho. Pelo incentivo e ajuda na revisão e pelas palavras de motivação que me fizeram prosseguir e finalmente terminar este trabalho.

À minha mãe Elizabeth Bernardes Panerai pela preocupação, incentivo e auxílio nas revisões que me motivaram para que a elaboração deste trabalho se desse da melhor forma possível.

Ao meu pai e à minha madrastra Roberto Guardiola Velloso e Sônia Palacio Guido por acreditarem em mim e me apoiarem durante toda a execução deste trabalho.

Aos meus irmãos Roberto, Bernardo e Matheus por serem além de meus irmãos meus amigos.

Aos meus amigos Clayton, Luiz, Maurano, Gonçalves, De Bem, Rogério, Ralf, Alex, Patricia, Raquel, Mariana, as respectivas e respectivos deles e a todos os outros amigos que aqui não foram citados, por serem meus amigos e tornarem minha vida mais feliz.

Ao meu Orientador Mauro, pela orientação, paciência interminável e por todos os votos de confiança que recebi durante este trabalho.

Aos meus colegas do Hiperlab Alice, Marília e Ronnie pela ajuda e compreensão durante todas as fases deste trabalho.

À secretária do PPGCC Vera Lúcia Sodré Teixeira por tornar minha passagem por este programa muito mais fácil e prazerosa, auxiliando em todos os processos acadêmicos/burocráticos nos quais invariavelmente cometi erros.

À todos que me auxiliaram nesta empreitada e que não puderam ser citados aqui.

Eu agradeço a vocês.

Lista de Publicações

Bruno Panerai Velloso, Mauro Roisenberg - *Percolation analyses in a swarm based algorithm for shortest-path finding* - **Symposium on Applied Computing**, pp. 1861-1865, ACM-SAC 2008

Conteúdo

| | |
|--|-------------|
| Lista de Figuras | ix |
| Lista de Tabelas | xi |
| Resumo | xii |
| Abstract | xiii |
| 1 Introdução | 1 |
| 1.1 Motivação | 1 |
| 1.2 Caracterização do Problema | 3 |
| 1.3 Objetivos | 4 |
| 1.3.1 Objetivo Geral | 4 |
| 1.3.2 Objetivos Específicos | 5 |
| 1.4 Metodologia | 5 |
| 1.5 Organização do Trabalho | 6 |
| 2 Teoria da Percolação | 7 |
| 2.1 Percolação em Ambientes Finitos | 10 |
| 2.2 Zona Crítica | 13 |
| 2.3 Conclusão | 15 |
| 3 Algoritmos de <i>Swarm Intelligence</i> | 17 |
| 3.1 Ant Colony Optimization | 19 |
| 3.2 Agentes Ant Colony | 20 |

| | | |
|----------|---|-----------|
| 3.2.1 | Ant System(AS) | 21 |
| 3.2.2 | Ant Colony System(ACS) | 21 |
| 3.2.3 | Variações do Ant Colony System | 23 |
| 3.3 | Parâmetros | 23 |
| 4 | Metodologia e Simulações | 25 |
| 4.1 | Sistema Proposto | 26 |
| 4.2 | Metodologia Proposta | 27 |
| 4.3 | Aplicação da Metodologia | 30 |
| 4.3.1 | Definição do Algoritmo | 30 |
| 4.3.2 | Definição do Critério de Percolação | 30 |
| 4.3.3 | Análise dos Parâmetros | 31 |
| 4.3.4 | Simulações | 32 |
| 4.4 | Considerações finais sobre a metodologia proposta | 37 |
| 5 | Resultados | 38 |
| 5.1 | Resultados da aplicação da metodologia proposta | 42 |
| 5.2 | Cálculo de Parâmetros para o <i>Ant System</i> | 45 |
| 5.2.1 | Equações Interpoladas | 47 |
| 5.2.2 | Validação dos Resultados | 55 |
| 5.3 | Conclusões sobre os resultados | 58 |
| 6 | Conclusão | 60 |
| 7 | Apêndice | 63 |
| | Referências Bibliográficas | 65 |

Lista de Figuras

| | | |
|-----|--|----|
| 2.1 | <i>Simulação de incêndios florestais baseada na teoria da percolação - a)Estado Inicial b)Estado Final - Sem ocorrência de percolação - Fonte:[PRI 08]</i> | 8 |
| 2.2 | <i>Simulação de incêndios florestais baseada na teoria da percolação - a)Estado Inicial b)Estado Final - Com ocorrência de percolação - Fonte:[PRI 08]</i> | 8 |
| 2.3 | <i>Temperatura x Estado - Descontinuidade no Ponto Crítico</i> | 9 |
| 2.4 | <i>Grafo após aplicação da equação 2.1 para a)pequenos valores de P_c e b)grandes valores de P_c</i> | 10 |
| 2.5 | <i>Variância da Probabilidade de Percolação em um grafo finito</i> | 11 |
| 2.6 | <i>Raio de Correlação</i> | 12 |
| 2.7 | <i>Raio de Correlação</i> | 13 |
| 2.8 | <i>Percolação em ambientes finitos</i> | 14 |
| 2.9 | <i>Percolação em ambientes finitos</i> | 16 |
| 3.1 | <i>TETwalker - Robô de reconhecimento desenvolvido pela NASA Fonte: [NAS 05]</i> | 18 |
| 3.2 | <i>Comportamento de agentes ant colony - Fonte: [JUN 06]</i> | 20 |
| 3.3 | <i>Três ciclo da execução do Ant Colony System em um Grafo Fonte: [DOR 97]</i> | 22 |
| 3.4 | <i>Conjunto de parâmetros do AntNet - Fonte: [JUN 06]</i> | 24 |
| 4.1 | <i>Fluxograma da Metodologia Proposta</i> | 29 |
| 4.2 | <i>Saída do Simulador - Retorno de um agente com trilha de Feromônio incompleta</i> | 34 |
| 4.3 | <i>Saída do Simulador - Início do retorno de um agente e trilha de Feromônio</i> | 35 |

| | | |
|------|---|----|
| 4.4 | <i>Gráfico de Percolação x Densidade de Agentes obtido com a simulação 1</i> | 37 |
| 5.1 | <i>Gráfico de Percolação x Densidade de agentes para a simulação 2</i> | 40 |
| 5.2 | <i>Gráfico de Desvio Padrão x Densidade de Agentes - Simulação 2</i> | 40 |
| 5.3 | <i>Gráfico de Variância x Densidade de Agentes - Simulação 2</i> | 41 |
| 5.4 | <i>Gráfico de Tempo de simulação x Densidade de Agentes - Simulação 1 . .</i> | 41 |
| 5.5 | <i>Gráfico de Alimento Coletado x Densidade de Agentes - Simulação 2 . . .</i> | 42 |
| 5.6 | <i>Gráfico de Percolação x Densidade de Agentes - Simulação 3 - Em ambiente 50 x 50 com variação no valor de Depósito de Feromônio(DE) de 8 a 24</i> | 44 |
| 5.7 | <i>Gráfico de Percolação x Densidade de Agentes - Simulação 3 - Em ambiente 100 x 100 com variação no valor de Depósito de Feromônio(DE) de 13 a 37</i> | 44 |
| 5.8 | <i>Gráfico de Percolação x Densidade de Agentes - Simulação 3 - Em ambiente 150 x 150 com variação no valor de Depósito de Feromônio(DE) de 35 a 55</i> | 45 |
| 5.9 | <i>Polinômio interpolado para cálculo de C em ambiente 100x100</i> | 48 |
| 5.10 | <i>Polinômio interpolado para cálculo de C em ambiente 150x150</i> | 49 |
| 5.11 | <i>Polinômio interpolado para cálculo de C em ambiente 50x50</i> | 50 |
| 5.12 | <i>Polinômio interpolado x Valores das constantes</i> | 51 |
| 5.13 | <i>Limite inferior da zona crítica em um gráfico de variância da probabilidade de existência de percolação x densidade de agentes</i> | 52 |
| 5.14 | <i>Regressão Não Linear em função da quantidade de feromônio depositada para cálculo de Pc em ambiente 50x50</i> | 53 |
| 5.15 | <i>Regressão Não Linear em função da quantidade de feromônio depositada para cálculo de Pc em ambiente 100x100</i> | 54 |
| 5.16 | <i>Regressão Não Linear em função da quantidade de feromônio depositada para cálculo de Pc em ambiente 150x150</i> | 55 |
| 5.17 | <i>Polinômio interpolado x Valores das constantes - Calculo de Pc</i> | 57 |
| 5.18 | <i>Comparação entre os valores simulados e os valores interpolados</i> | 58 |

Lista de Tabelas

| | | |
|-----|-----------------------|----|
| 4.1 | Simulação 1 | 36 |
| 5.1 | Simulação 2 | 39 |
| 5.2 | Simulação 3 | 43 |
| 5.3 | Simulação 3 | 46 |
| 5.4 | Simulação 3 | 46 |
| 5.5 | Simulação 3 | 46 |
| 5.6 | Validação | 56 |
| 5.7 | Resultados | 56 |

Resumo

Algoritmos de inteligência de enxame (*swarm intelligence*) são algoritmos biologicamente inspirados no funcionamento social de grandes colônias de insetos ou outros animais e que caracterizam-se por possuir um elevado número de agentes com baixa complexidade e controle descentralizado. Estes agentes, através da interação com o ambiente e com outros agentes procuram prover boas soluções para problemas complexos de otimização em tempos aceitáveis.

O desempenho de um algoritmo de inteligência de enxame está intimamente relacionado à escolha adequada dos parâmetros que governam o funcionamento do algoritmo, em especial, a quantidade de agentes e a taxa de evaporação do feromônio.

Este trabalho procura analisar e modelar um algoritmo de inteligência de enxame, o algoritmo de sistema de formigas (*ant system*), baseado na teoria da percolação, teoria que estuda o comportamento de fenômenos críticos e de transição de fases, propondo uma metodologia que permita a garantia de convergência para a solução e a obtenção de valores próximos ao ótimo para os parâmetros do algoritmo, melhorando deste modo o desempenho das aplicações que façam uso desta tecnologia.

Palavras Chave: Inteligência de enxames, Inteligência de Coletiva, Teoria da Percolação, Fenômenos Críticos, Desempenho de Algoritmos.

Abstract

Swarm intelligence algorithms are biological inspired in the social behavior of great insects or other animals colonies. They are characterized by a large number of agents with low level of complexity and decentralized control. These agents, try to provide good solutions for complex optimization problems in reasonable times through interaction with the environment and other agents.

The swarm intelligence algorithm performance is closely related to the right choice of the parameters . These parameters are responsible for the algorithm behavior, in special, the amount of agents and the pheromone evaporation rate.

This work shows a modeling and analysis of a swarm intelligence based algorithm, the Ant System, based in the percolation theory. This theory studies the behavior of critical and phase transitions phenomena, proposing a methodology that allows the guarantee of solution convergence and the determination of closer optimal values for the algorithm parameters, improving the performance of the applications that use this technology.

Key Words: Swarm Intelligence, Percolation Theory, Critical Phenomena, Algorithm performance.

Capítulo 1

Introdução

1.1 Motivação

O interesse por algoritmos com inspiração biológica vem crescendo e suas aplicações são cada vez mais numerosas. A capacidade destes algoritmos de resolver problemas de complexidade elevada, os torna cada vez mais, alvos das pesquisas[ARK 98]. Dos algoritmos com inspiração biológica serão abordados neste trabalho assuntos relativos aqueles algoritmos classificados como *swarm intelligence*, termo inglês que pode ser traduzido como inteligência de enxames ou inteligência coletiva. Ao longo deste trabalho optou-se pelo uso do termo em inglês, uma vez que não existe um consenso na comunidade científica nacional quanto ao termo em português mais apropriado.

Os algoritmos de *swarm intelligence* fazem uso de um número elevado de agentes, de complexidade baixa. Como produto de sua interação com o ambiente ou entre si, produzem, na forma de uma propriedade emergente, a solução para o problema ao qual se destinam[ROU 04]. A inspiração biológica presente nestes algoritmos pode ser observada em sistemas que baseado em colônias de formigas implementam buscas em grafos[DOR 96] e rotas em redes de computadores[KWA 03], ou mesmo baseado na movimentação de cardumes de peixes implementam o reconhecimento de padrões em imagens[OMR 05].

Como característica estes algoritmos possuem um conjunto de parâmetros

que rege seu comportamento e, em conseqüência disto, seu desempenho: Tal conjunto de parâmetros possui valores empíricos normalmente ajustados de forma experimental para cada aplicação em particular.

Os algoritmos de *swarm intelligence* são utilizados em um vasta gama de aplicações, e o ajuste impreciso de seus parâmetros compromete o desempenho destas aplicações. Sistemas que façam uso de um número excessivo de agentes, e por conseqüência de recursos, podem apresentar desempenho incompatível com o esperado, tornando sua utilização restrita a dispositivos com alto poder computacional. Este tipo de restrição representa um problema em diversos casos. É cada vez maior o número de plataformas de baixo poder computacional que são utilizadas em larga escala, tais como celulares, aplicações embarcadas e palm pilots. Nestas plataformas a má utilização de recursos é proibitiva. Acredita-se que este problema, em alguns casos, possa ser contornado com uma correta configuração dos parâmetros dos algoritmos.

A escolha correta dos valores dos parâmetros dos algoritmos de *swarm intelligence* apresenta uma complexidade não elementar, uma vez que tais parâmetros não possuem um estudo exaustivo[JUN 06] e, em grande parte, não implicam, de forma clara, uma relação direta com o desempenho.

Existem parâmetros que apresentam alguma relação direta com o desempenho dos algoritmos, como é o caso do parâmetro que controla a quantidade de agentes criada no algoritmo. Este parâmetro pode ser vinculado de forma simples com o desempenho, pois espera-se que cada agente consuma uma determinada quantidade de memória e de processamento. Desta maneira um número muito grande de agentes provavelmente implica um uso excessivo de recursos.

O número de agentes é um parâmetro que pode ser vinculado ao desempenho sem maiores problemas, por outro lado, grande parte dos parâmetros dos algoritmos estudados não apresentam uma relação tão direta com o desempenho. Por exemplo supondo um parâmetro que controle a interação do agente com o ambiente, este parâmetro não apresenta claramente uma relação com o uso de recursos do algoritmo, mas sua influência sobre o comportamento emergente pode ser determinante para a otimização do desempenho.

O comportamento crítico dos algoritmos de *swarm intelligence* é conhecido, como pode ser observado em [LEU 03a] e [LEU 03b], onde o autor propõe um sistema capaz de construir estruturas geométricas com base no comportamento de uma colônia de agentes autônomos. Nestes trabalhos é caracterizada a transição de fase entre a obtenção ou não do objetivo com a variação dos valores dos parâmetros que configuram o algoritmo implementado. No entanto não foram encontrados, durante a pesquisa efetuada para execução do presente trabalho, trabalhos relacionados ao refinamento do desempenho deste tipo de algoritmos tendo por base o estudo de seu comportamento crítico.

Como principal motivação deste trabalho, é apresentada a possibilidade de otimização destas aplicações com o uso de um conjunto de parâmetros mais próximo ao conjunto considerado ótimo. Para isso busca-se obter um método que viabilize a modelagem matemática do sistema em questão. A criação de um modelo matemático para estes sistemas é muito atraente do ponto de vista da otimização de recursos, ma vez que torna possível o ajuste de parâmetros em tempo de execução, incrementando a performance mesmo em sistemas com características variáveis.

1.2 Caracterização do Problema

A existência de um comportamento crítico nos algoritmos de *swarm intelligence* implica que, sob determinadas condições, o algoritmo não é capaz de resolver os problemas a que se propõe. Aparentemente, este comportamento é oriundo do conjunto de valores dos parâmetros. Desta forma, conjuntos insatisfatórios de valores tornam a probabilidade de resolução dos problemas bastante pequena. Por outro lado valores adequados de parâmetros fazem com que o algoritmo encontre a solução do problema na grande maioria das vezes.

Este comportamento pode ser descrito como um fenômeno de transição de fase, ou seja, idealmente, a partir de determinados valores de parâmetros, o problema é sempre resolvido, enquanto que abaixo destes valores o algoritmo nunca obtém esta solução. Este fenômeno pode ser observado mais claramente variando-se o número de agentes utilizado por um algoritmo de *swarm intelligence*. Se este número for muito

pequeno, os agentes dificilmente obterão êxito, pois sua interação com o ambiente e entre os próprios agentes, base do funcionamento do algoritmo, será prejudicada. Por outro lado com um número maior de agentes, é provável que o algoritmo obtenha a solução do problema conforme esperado.

A técnica utilizada neste trabalho para a análise dos fenômenos de transição de fase é a Teoria da Percolação que estuda e modela, por meios estatísticos, o comportamento dos fenômenos críticos.

Neste trabalho é apresentada a proposta de uma metodologia para análise de algoritmos de *swarm intelligence*. Para tal é realizada a análise de um algoritmo que simula de forma simples o comportamento de uma colônia de formigas, conhecido como *Ant System*[DOR 96]. Este algoritmo é capaz de encontrar o menor caminho entre dois pontos de um grafo. Conforme esperado de algoritmos de *swarm intelligence*, o *Ant System* emprega um grande número de agentes de pequena complexidade e de fácil implementação e possui um conjunto de parâmetros que regem seu comportamento.

Por sua simplicidade e funcionalidade o *Ant System* é utilizado em diversas aplicações como: Robôs móveis, inteligência artificial, gerenciamento de redes, sistemas de transporte e outros[DOR 96][BAL 00].

A teoria da percolação tem como principal objetivo o estudo de fenômenos críticos, ou seja, que apresentam transição de fase[EFR 87]. É empregada neste trabalho com a finalidade de obter dados referentes às condições de contorno dos algoritmos de *swarm intelligence* e, desta forma, determinar os valores de parâmetros que tornam possível a resolução dos problemas com o uso mínimo de recursos.

1.3 Objetivos

1.3.1 Objetivo Geral

O objetivo geral deste trabalho é definido pela proposta de uma metodologia que torne possível a determinação de parâmetros para algoritmos do tipo *swarm intelligence* otimizando o desempenho de aplicações que façam uso deste tipo de tecno-

logia.

1.3.2 Objetivos Específicos

Como objetivos específicos pode-se citar:

- Evidenciar o comportamento percolativo dos algoritmos de *swarm intelligence*;
- Caracterizar o conjunto de parâmetros ótimos para os algoritmos estudados;
- Obter um modelo matemático do algoritmo implementado que mapeie as características do ambiente simulado com os valores dos parâmetros de interesse, possibilitando melhora no desempenho do algoritmo;
- Analisar com base na metodologia proposta um algoritmo do tipo *Ant System*;

1.4 Metodologia

Pode-se definir a metodologia utilizada neste trabalho como os seguintes passos:

- Estudo das características dos algoritmos de *swarm intelligence*;
- Estudo das técnicas envolvidas na teoria da percolação;
- Construção de um simulador para conjuntos variados de parâmetros de um algoritmo *Ant System*;
- Aplicar a metodologia proposta para análise dos algoritmos em um algoritmo do tipo *Ant System*
- Simulação de um algoritmo do tipo *Ant System* com o conjunto de parâmetros selecionados;
- Validar o modelo obtido comparando valores interpolados de parâmetros com valores obtidos nas simulações

1.5 Organização do Trabalho

Neste trabalho os assuntos abordados estão organizados em capítulos da seguinte forma:

- *Capítulo 1 - Introdução:* Apresenta a introdução ao assunto do trabalho, a motivação para sua execução e a organização do presente trabalho;
- *Capítulo 2 - Teoria da Percolação:* Apresenta uma revisão sobre os fundamentos da teoria da percolação e uma introdução sobre os aspectos e equações relevantes para execução deste trabalho;
- *Capítulo 3 - Swarm Intelligence:* Apresenta uma revisão sobre as técnicas de resolução de problemas baseadas em *swarm intelligence* e aborda os principais algoritmos utilizados, suas aplicações e características, apresentando exemplos do tipo de conjunto de parâmetros esperados destes algoritmos;
- *Capítulo 4 - Metodologia e Simulações:* Desenvolve a Metodologia alvo deste trabalho e aplica esta metodologia a um algoritmo implementado por meio do simulador desenvolvido com esse propósito;
- *Capítulo 5 - Resultados:* Apresenta e discute os resultados e análises resultantes da aplicação da metodologia proposta no capítulo anterior;
- *Capítulo 6 - Conclusão:* Compila as conclusões possíveis deste trabalho e propõe trabalhos futuros abordando aspectos relacionados a este trabalho;
- *Capítulo 7 - Apêndices:* Dados, simuladores e aplicativos utilizados para elaboração deste trabalho;

Capítulo 2

Teoria da Percolação

Neste capítulo é apresentada uma breve revisão da teoria da percolação e seus aspectos relevantes para o presente trabalho. A teoria da percolação foi inicialmente desenvolvida por Stauffer e Aharony [STA 92] para análise de formação de moléculas em substâncias químicas. Posteriormente, em sua abordagem contemporânea, foi utilizada para modelar o escoamento de um fluido em um meio permeável ou poroso. A teoria não define os termos meio e fluido, sendo estes abstratos possibilitando sua aplicação em sistemas diversos.

As aplicações da teoria da percolação abordam variados assuntos tais como: Estudo de propagação de ondas em meios aleatórios[RAI 08], análise de incêndios florestais[MAC 84], conectividade em redes de computadores não estruturadas[SAN 07], formação de polímeros[STA 82] e outros. A figura 2.1a mostra o estado inicial de uma simulação de incêndios florestais baseada em modelos percolativos. Para esta simulação os parâmetros relevantes foram definidos como a densidade de árvores, direção e velocidade do vento. Na figura 2.1a as árvores são representadas pelos pontos verdes. O vento não é representado, mas tem influência no estado final da simulação, apresentado na figura 2.1b. O objetivo da simulação é, que na eventualidade de um foco de incêndio, definir se o incêndio se propagará para todas as árvores próximas ou se extinguiu espontaneamente.

A figura 2.1b apresenta o estado final da simulação. Nesta simulação pode-se perceber que a grande maioria das árvores não foi consumida pelo fogo, perma-

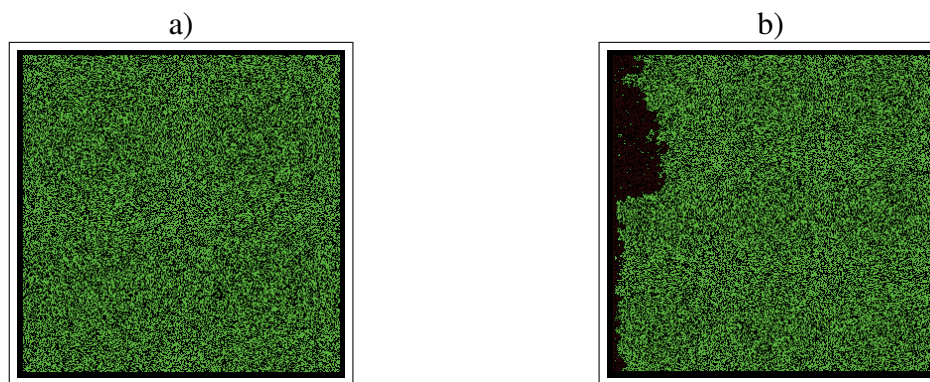


Figura 2.1: Simulação de incêndios florestais baseada na teoria da percolação - a)Estado Inicial
b)Estado Final - Sem ocorrência de percolação - Fonte:[PRI 08]

neccendo como apresentadas no estado inicial figura 2.1a. Neste caso define-se que não existiu percolação. Por outro lado, na figura 2.2a pode-se ver o estado inicial de uma simulação de incêndio florestal com a densidade de árvores maior. Neste caso como apresentado na figura 2.2b existe percolação, ou seja, o fogo se propaga para todas as árvores próximas.

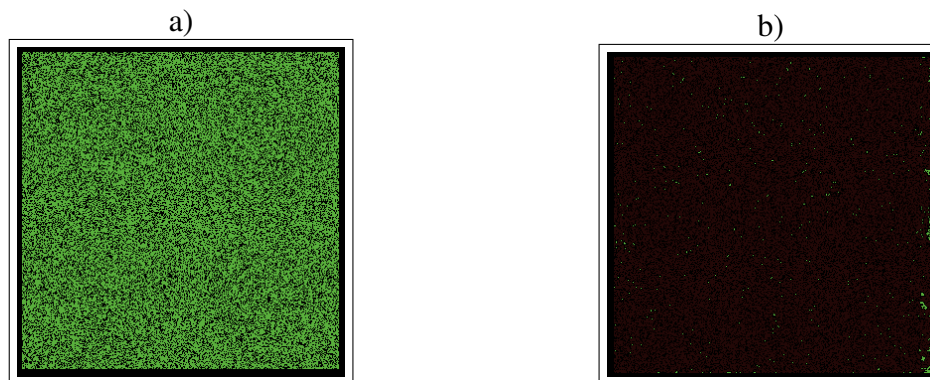


Figura 2.2: Simulação de incêndios florestais baseada na teoria da percolação - a)Estado Inicial
b)Estado Final - Com ocorrência de percolação - Fonte:[PRI 08]

A teoria da percolação é baseada no estudo dos fenômenos críticos, ou seja, fenômenos onde existe um ponto de troca abrupta de comportamento ou estado [EFR 87]. Tais fenômenos, por exemplo, a troca de estados físicos de líquido para gasoso, se representados em gráficos adequados, apresentam um ponto de descontinuidade

ou uma função do tipo degrau. No exemplo proposto, em um gráfico de estado por temperatura da água, o ponto crítico seria representado em 100°C conforme a figura 2.3.

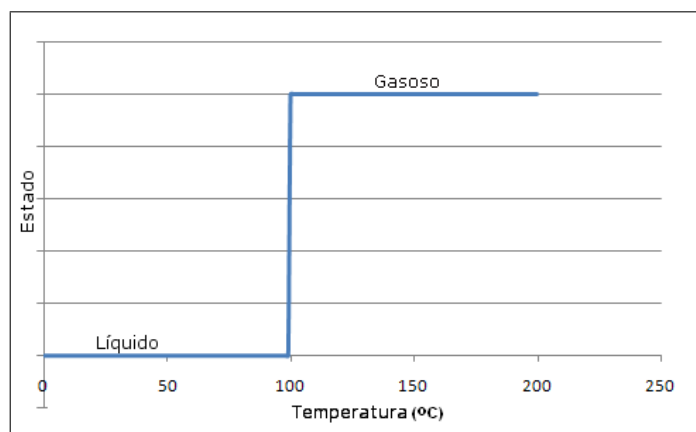


Figura 2.3: *Temperatura x Estado - Descontinuidade no Ponto Crítico*

Na figura 2.3 percebe-se o ponto de descontinuidade no gráfico em 100°C . Analisando o gráfico com base na teoria da percolação pode-se afirmar que nos pontos de coordenadas menores que 100 não há percolação e nos pontos de coordenadas maiores que 100 há percolação. O ponto de coordenada 100 é denominado ponto crítico (P_c).

Outro exemplo de aplicação da teoria da percolação trata de um sólido poroso que mergulhado em determinado fluido tem seus poros superficiais inundados. De acordo com a porosidade deste sólido, estes poros estarão ligados em maior ou menor número com os poros internos e estes com outros poros. Desta forma em determinada porosidade o fluido será capaz de transpassar o sólido de uma de suas faces até a face oposta. Na eminência deste evento é dito que o fluido percolou o sólido ou que existe percolação para determinada porosidade. [STA 92] Avaliando este fenômeno percebe-se que em porosidades menores que determinado valor V , não existe percolação e para valores maiores que V existe percolação. Logo V é o valor do ponto crítico P_c .

Pode-se modelar a percolação como um grafo onde os vértices representam os poros do material e as arestas as conexões entre os poros. Desta forma baseado em uma variável aleatória P pode-se marcar cada vértice x do grafo como ocupado ou

vazio, de acordo com a equação 2.1

$$Percola(x, P) = \begin{cases} ocupado, P > P_c \\ vazio, P < P_c \end{cases} \quad (2.1)$$

A aplicação da equação 2.1 em um grafo quadrado totalmente conectado pode ser representada pela figura 2.4a para valores de P_c menores que 0,2. Na figura 2.4b é apresentado o resultado para valores de P_c maiores que 0,8. Os pontos preenchidos representam vértices ocupados e os pontos vazados representam vértices vazios.

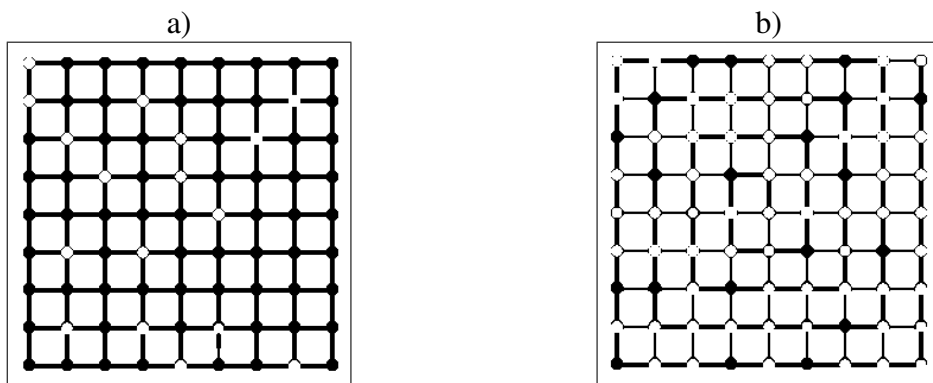


Figura 2.4: Grafo após aplicação da equação 2.1 para a)pequenos valores de P_c e b)grandes valores de P_c

Analisando as figuras 2.4 pode-se notar a existência de agrupamentos contínuos de vértices marcados como ocupados. Estes agrupamentos são denominados *clusters*[EFR 87]. A percolação ou sua existência é obtida quando a formação destes *clusters* aleatórios cria estruturas contínuas entre duas faces opostas.

2.1 Percolação em Ambientes Finitos

O cálculo da probabilidade de um determinado nó de um grafo estar ocupado ou vazio é o foco principal da teoria da percolação. Em ambientes finitos pode-se, de forma empírica, determinar tal probabilidade, que estará por sua vez vinculada às

características do grafo, e à existência de *clusters percolantes* e apresentará uma determinada variância para cada valor de probabilidade P , representando a imprecisão vinculada a esta medida.

Segundo [EFR 87]: “O desvio quadrático médio $\delta(N)$ diminui de forma exponencial proporcionalmente ao número N de nós(vértices), tendendo a zero quando $N \rightarrow \infty$.” Agregada a esta afirmação o autor propõe a equação 2.2

$$\delta(N) = \frac{C}{N^{1/2v}} \quad (2.2)$$

Onde N é o número de vértices, C e v são constantes numéricas referentes ao ambiente(tais como número e forma das aresta do grafo). A equação 2.2 foi determinada empiricamente com base em repetidos testes com grafos de dimensões diversas, mas pode ser determinada algebricamente como apresentado em [EFR 87]. A figura 2.5 apresenta o resultado esperado de variância para um grafo finito de dimensões fixas para diversos valores de probabilidade.

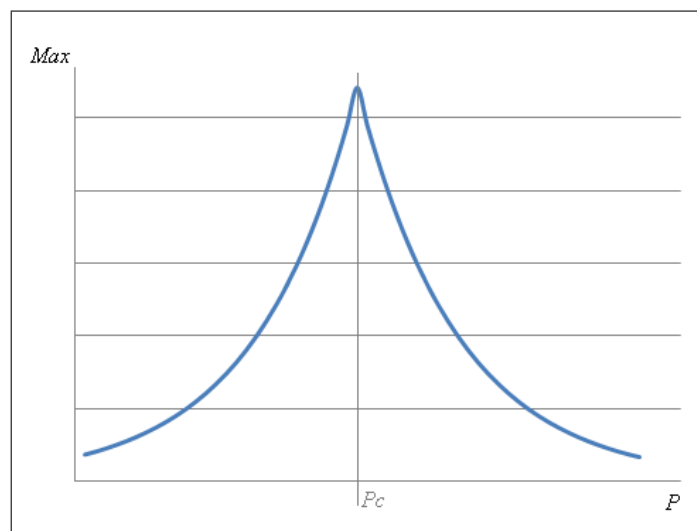


Figura 2.5: Variância da Probabilidade de Percolação em um grafo finito

A teoria da percolação prevê, para ambientes infinitos compostos de elementos infinitesimais, uma descontinuidade no ponto crítico P_c . Neste ponto é esperado um valor de variância indeterminado tendendo ao infinito. Na figura 2.5 pode-se perceber

que com a proximidade do ponto crítico, a variância, em um ambiente finito composto de elementos discretos, tende ao seu valor máximo. Neste ponto a incerteza relativa a existência ou não de percolação também tende a ser máxima.

Segundo B.I.Shklovski *appud* [EFR 87] um *Raio de Correlação* é a dimensão linear média, conforme figura 2.6, relativo a uma probabilidade P de existência de conexão entre dois nós da rede. Aplicado a um grafo quadrado o valor do raio de correlação representa o tamanho esperado de *clusters* e pode ser definido pela equação 2.3.

$$R = \frac{l}{|P - P_c|^v} \quad (2.3)$$

Onde R é o Raio de Correlação, l a periodicidade característica de conexão entre nós, P uma variável aleatória, P_c o Ponto crítico de Percolação para determinada rede e v é uma constante positiva denominada índice de raio de correlação.

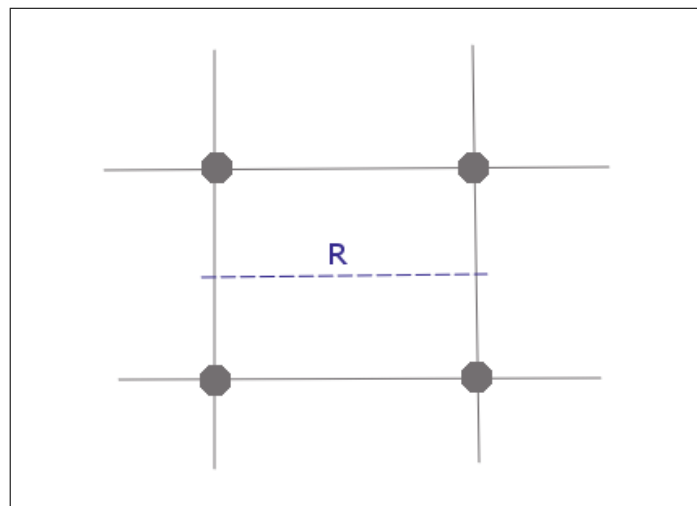


Figura 2.6: *Raio de Correlação*

Analisando a equação 2.3 percebe-se que na vizinhança do ponto crítico o valor de R tende ao infinito. O gráfico da figura 2.7 apresenta a relação entre o raio de correlação e a variância para ambientes finitos de tamanhos variados. Analisando este gráfico pode-se perceber a influência do uso de elementos finitos uma vez que quanto

menor as dimensões, menor o tamanho médio esperado dos agrupamentos. Quanto menor o raio de correlação R , maior a variância com relação à ocorrência de percolação, ou seja, quanto menor o meio maior a incerteza relacionada a existência de percolação.

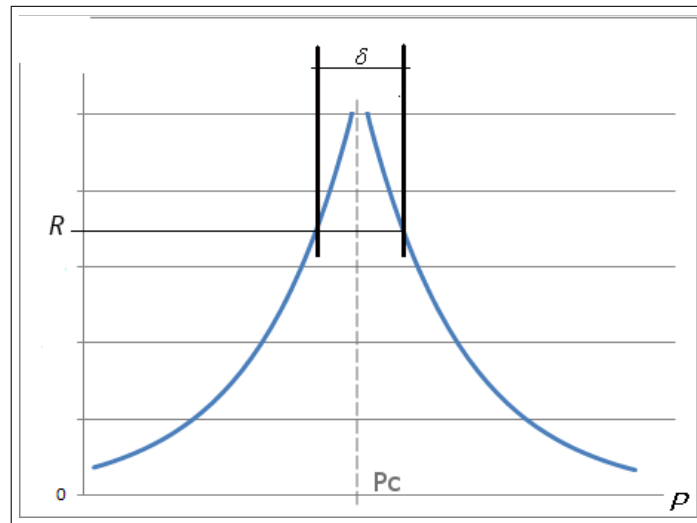


Figura 2.7: Raio de Correlação

O efeito esperado da existência de valores elevados de variância próximos ao ponto crítico é que este último se torna indistinto fazendo a transição de fase suave. Isto deve-se, ao fato da análise ser feita em ambientes finitos. O gráfico de transição de fase esperado é representado na figura 2.8.

2.2 Zona Crítica

Como se torna impossível determinar o ponto crítico de um sistema em ambientes finitos pode-se definir como zona crítica o intervalo onde é imprecisa a existência ou não de percolação, ou seja, o intervalo onde os valores de P possuem valores de variância para percolação não nulos[EFR 87]. Logo pode-se afirmar que:

$$Percola(x, P) = \begin{cases} ocupado, P \gg P_c \\ vazio, P \ll P_c \end{cases} \quad (2.4)$$

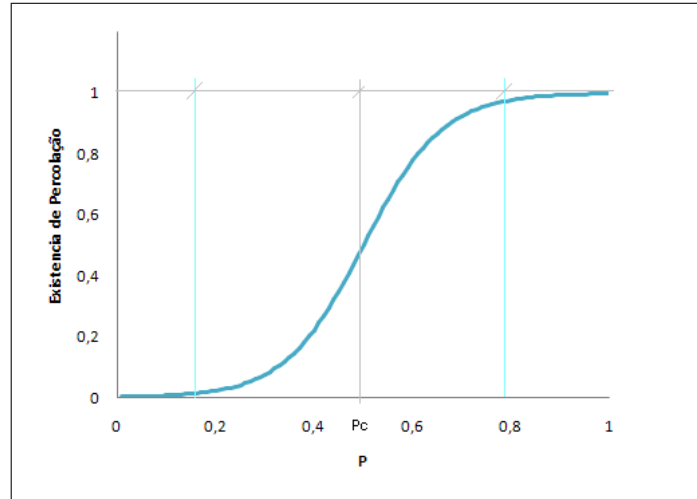


Figura 2.8: Percolação em ambientes finitos

Com base na equação 2.4 admitindo-se $R = L$, onde L é a dimensão lateral de um grafo quadrado, tem-se:

$$R = \frac{l}{|P - P_c|^v}$$

$$L = \frac{l}{|P - P_c|^v}$$

onde, $|P - P_c| = \delta$,

$$L = \frac{l}{\delta^v}$$

ou

$$\delta = \left(\frac{l}{L} \right)^{1/v} \quad (2.5)$$

O número total de vértices de um grafo pode ser calculado por:

$$N = \left(\frac{L}{l}\right)^d \quad (2.6)$$

Onde, d é a dimensão espacial para o caso do grafo quadrado $d = 2$. Com base nas equações 2.5 e 2.6 tem-se:

$$\delta(N) = \frac{C}{N^{1/dv}} \quad (2.7)$$

A equação 2.7 é a generalização da equação 2.2, aplicada a grafos quadrados. O valor de $\delta(N)$ é interpretado como o tamanho do intervalo denominado zona crítica e é o objeto de estudo da teoria da percolação em ambientes finitos. A figura 2.9 apresenta o gráfico de percolação em ambientes finitos e a zona crítica. Este gráfico pode ser interpretado da seguinte forma, sabe-se que para valores de P menores que 0,2 não existe a ocorrência de percolação nunca e para valores de P maiores que 0,8 sempre é obtida a percolação, no entanto para análise de ocorrências de P dentro do intervalo $[0, 2; 0, 8]$ não é possível afirmar nada com relação a percolação. Com base na influência do uso de elementos finitos nesta análise é correto afirmar que a amplitude do intervalo compreendido por $\delta(N)$ será menor quanto maior forem as dimensões do ambiente estudado.

2.3 Conclusão

Com base na teoria da percolação pode-se calcular, para fenômenos de transição de fase, a zona crítica que é o intervalo de valores ordenados onde é impossível prever a troca de fase ou a percolação. A zona crítica tem centro no ponto crítico P_c e sua amplitude é inversamente proporcional ao número de vértices do grafo. Para ambientes finitos é impossível calcular o valor de P_c , no entanto pode-se calcular a amplitude da zona crítica com base na equação 2.2, e por meio da análise dos valores de variância o valor aproximado de sua posição.

Portanto tem-se:

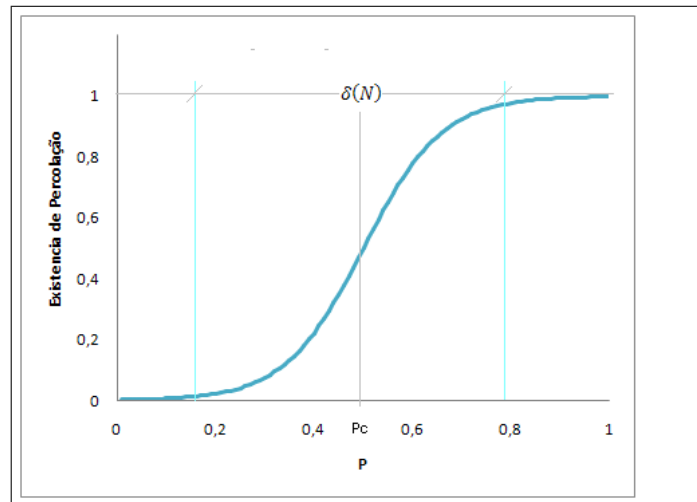


Figura 2.9: Percolação em ambientes finitos

$$Percola(x, P) = \begin{cases} \text{ocupado}, & P > Pc + \delta(N)/2 \\ \text{vazio}, & P < Pc - \delta(N)/2 \end{cases} \quad (2.8)$$

Capítulo 3

Algoritmos de *Swarm Intelligence*

São denominados algoritmos de *Swarm Intelligence*, os algoritmos que fazem uso de agentes simples, ou seja, com baixa complexidade de implementação e de atuação, sem controle centralizado e com interação com o ambiente. Estes agentes interagindo com o ambiente, de forma coletiva, são capazes de formar padrões complexos correspondentes a solução de determinado problema[ROU 04].

O uso de algoritmos de *Swarm Intelligence* tem se intensificado e seu campo de aplicações é amplo, tais como: Roteamento de redes[JUN 06], Reconhecimento de padrões[OMR 05], Mapeamento de Regiões[NAS 05], Simulações[RAI 08], Métodos de Otimização de Processos[RAI 08] e outros. As características básicas do algoritmo, tornam as implementações robustas, de baixo custo computacional e financeiro. Na figura 3.1 é apresentado o protótipo de um robô de reconhecimento baseada em *Swarm Intelligence* desenvolvido pelo NASA[NAS 05].

Os algoritmos de *Swarm Intelligence* tem por base a simulação de agentes com comportamentos simples e sua interação com o ambiente e com outros agentes. Normalmente em um algoritmo de *Swarm Intelligence* a solução de determinado problema é obtida pela configuração do ambiente, ou dos agentes, após determinado tempo de simulação ou até que determinada condição se verifique. Este tipo de efeito é comumente chamado de comportamento emergente, ou seja, um único agente não possui representatividade para o processo de solução de um problema, no entanto um número



Figura 3.1: *TETwalker* - Robô de reconhecimento desenvolvido pela NASA Fonte: [NAS 05]

mais elevado de agentes interagindo entre si e com o ambiente eventualmente obtém a solução do problema.

As características dos sistemas de *Swarm Intelligence* tornam este tipo de algoritmos atrativo do ponto de vista computacional, pois, baseando-se em comportamentos simples e agentes igualmente simples, a implementação não apresenta uma complexidade elevada. Deste modo, os recursos e desempenho geral tendem, em diversos casos, a ser tão ou mais apropriados que os de outros sistemas[DOR 96][OMR 05][CAR 98].

Existem diversas abordagens para construção de algoritmos de *Swarm*

Intelligence tais como: Particle Swarm[OMR 05]; Stochastic Search[MEY 06]; Ant Colony e outros. Um dos objetivos deste trabalho é apresentar uma metodologia coerente para a avaliação dos valores de parâmetros deste tipo de algoritmo. Para tal utilizou-se um sistema baseado nos algoritmos *Ant Colony Optimization*[LEU 03b].

3.1 Ant Colony Optimization

Os algoritmos do tipo *ant colony optimization*, propostos por Dorigo *et. al.*[BON 99][DOR 97][CAR 98][DOR 96], tem como um dos principais objetivos a determinação do menor caminho entre dois vértices de um grafo e seu comportamento é inspirado em uma colônia de formigas. A princípio todas as formigas estão localizadas no formigueiro e percorrem caminhos aleatórios a fim de localizar uma fonte de alimento. Se determinada formiga localiza uma fonte de alimento esta retorna para o ninho carregando uma parte do alimento, e deixando pelos caminho onde passa uma trilha de feromônio que pode ser detectado por outras formigas, desta forma guiando outros indivíduos ao local do alimento.

O feromônio por sua vez apresenta evaporação em função do tempo, e a constante reposição é feita pelas formigas que futuramente retornem ao ninho pelo mesmo caminho, desta forma criando uma trilha estável de feromônio. As formigas possuem uma maior probabilidade de percorrerem locais com maior concentração de feromônio. Estas características fazem com que o caminho de maior concentração de feromônio tenda a ser o menor caminho entre o ninho e o alimento[DOR 97].

Uma formiga que viaje pelo menor caminho leva menos tempo para repor o feromônio de cada local neste caminho, desta forma caminhos maiores tendem a apresentar menor concentração de feromônio e conseqüentemente atraem menos formigas. O maior número de formigas atraídas para o menor caminho elevam a concentração do feromônio deste. Existe uma probabilidade de que mesmo em concentrações altas de feromônio a formiga escolha outro caminho para seguir. Este fato é benéfico pois evita a convergência para caminhos não ótimos ou para mínimos locais.

Os parâmetros envolvidos no algoritmo, principalmente o Número de

Agentes (Formigas) e Taxa de Evaporação do Feromônio normalmente são estabelecidos de forma empírica e tem influência direta no desempenho e na utilização de recursos.

A figura 3.2 apresenta o comportamento esperado para os agentes de um algoritmos do tipo ant colony.

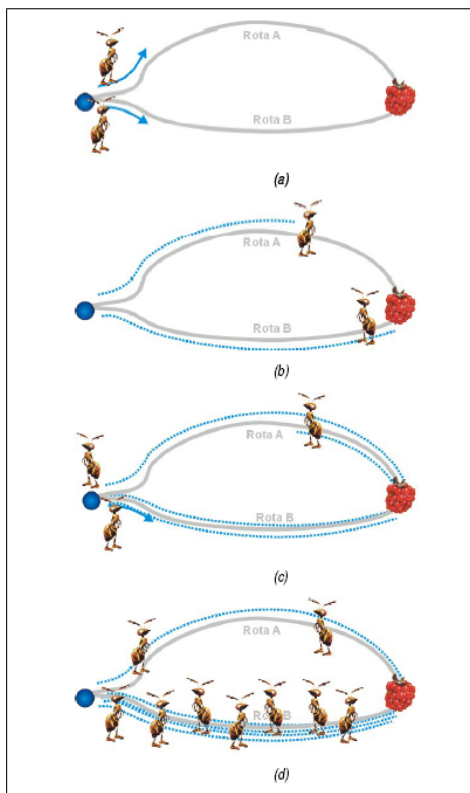


Figura 3.2: Comportamento de agentes ant colony - Fonte: [JUN 06]

3.2 Agentes Ant Colony

Os agentes de algoritmos do tipo *Ant Colony* compartilham as características esperadas de sistemas de *Swarm Intelligence* e por sua vez possuem controle descentralizado e baixa complexidade. Baseado na figura 3.2 o comportamento esperado dos agentes, e sua interação com o ambiente, pode ser descrito da seguinte forma:

- 3.2(a) - A priori os agentes se movem de forma a se afastarem do ninho em busca de alimento;

- 3.2(b) - Após localizar o alimento a formiga retorna ao ninho com uma parte do alimento. Na existência de mais de uma rota possível para o local do alimento é esperado que a formiga que se utilize da rota menor retorne primeiro ao ninho;
- 3.2(c) - Cada formiga que retorna ao ninho deposita feromônio no ambiente. Esta trilha pode ser seguida por outras formigas, sendo que as formigas tendem de forma estocástica a seguir a trilha com maior concentração de feromônio;
- 3.2(d) - O feromônio tem por característica a evaporação em função do tempo, logo sua concentração tende a se tornar maior no menor caminho visto que o período entre as reposições por outras formigas em retorno ao ninho é menor para esse caminho;

A formação de uma trilha de feromônio é a implementação da solução para o problema do menor caminho em um grafo, onde este grafo simula o ambiente dos agentes e os agentes podem se mover de um vértice para outro por meio das arestas do grafo.

3.2.1 Ant System(AS)

Proposto em 1991 por Dorigo e Di Caro[BON 99] foi o primeiro algoritmo a utilizar agentes com o comportamento baseado em formigas. É o protótipo precursor de outros algoritmos *ant colony optimization*.

As aplicações do Ant System foram voltadas para o problema do caixeiro viajante(Travel Salesman Problem - TSP) e testes sobre seu desempenho mostraram que este era melhor que de outros algoritmos clássicos[DOR 96] para problemas de dimensões(número de cidades) relativamente pequeno, no entanto, o Ant System não era capaz de resolver problemas com dimensões maiores.

3.2.2 Ant Colony System(ACS)

Baseado no algoritmo Ant System, o *Ant Colony System* propõe uma solução para a falha do precursor em problemas de dimensões maiores. Foi proposto por

Dorigo e Gambardella[DOR 97] em 1996.

As principais modificações com relação ao Ant System estão na atualização do feromônio, ou seja, na sua evaporação. *Ant Colony System* possui taxas de evaporação diferentes para cada trilha de feromônio criada pelos agentes. Desta forma a cada ciclo é verificado o melhor caminho atual, definido como aquele que no momento atual da simulação é avaliado como mais bem sucedido por uma função heurística, e com base neste caminho o feromônio de outros caminhos é decrementado, fazendo com que a chance de formigas seguirem o melhor caminho seja maior com relação aos outros.

A atualização do feromônio é feita pelo agente e não de forma global como proposto no *Ant System*. Essa estratégia tem por objetivo favorecer a criação de novos caminhos que não o melhor caminho atual, pois torna necessário que determinado caminho seja percorrido pelo menos por um agente antes de ser destruído pela evaporação, o que não ocorre com o *Ant System*.

Os novos caminhos são verificados ao final do ciclos e se forem julgados melhores que o melhor caminho atual, passam a possuir este status, tendo uma taxa de evaporação do feromônio inferior aos demais.

A figura 3.3 apresenta a evolução dos ciclo em algoritmo do tipo *Ant Colony System*, onde τ é proporcional a concentração de feromônio.

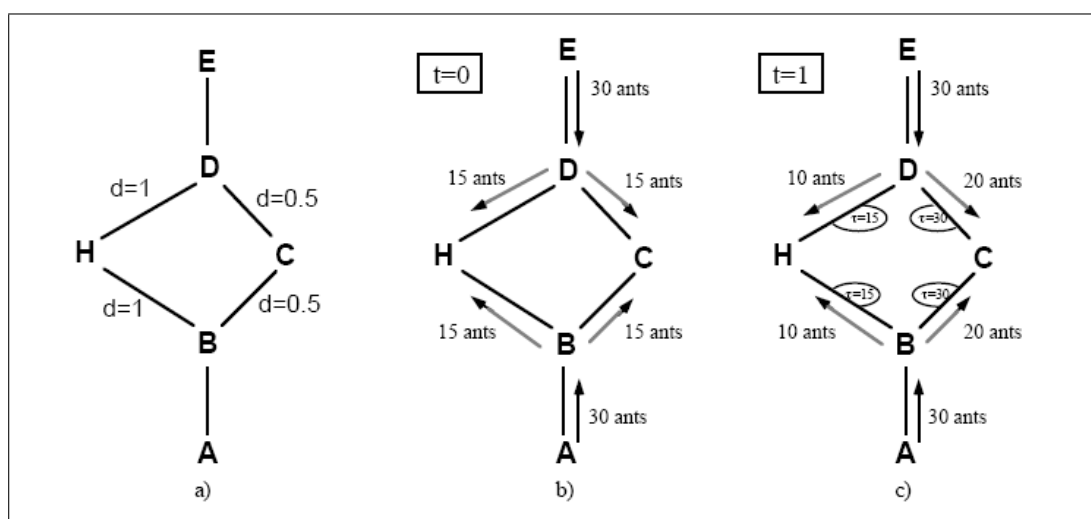


Figura 3.3: Três ciclo da execução do Ant Colony System em um Grafo Fonte: [DOR 97]

Foi executada uma modificação na regra de escolha de caminho do agente que neste algoritmo passa a considerar a existência de heurísticas locais, ou seja, própria da aplicação do algoritmo, tal como a distância entre as cidades para o TSP.

O algoritmo *Ant Colony System* foi testado e comparado com outros algoritmos com o mesmo propósito, e tanto no consumo de recursos como na qualidade da solução proposta o *Ant Colony System* apresentou resultados semelhantes ou melhores que outros algoritmos para problemas de variadas dimensões[CAR 04].

3.2.3 Variações do Ant Colony System

Com base no algoritmo *Ant Colony System* foram implementados diversos outros algoritmos, tais como AntNet[CAR 98], AntNet-FA[PAR 08], AntNet-FS[PAR 08], CAF[GHO 06], ABC-backward[GHO 06] e outros. Esses algoritmos tem os mais diversos propósitos, como roteamento em redes e procura de menor caminho. Possuem em comum as características do *Ant Colony System* e por consequência do *Ant System*. As diferenças básicas entre estes algoritmos encontram-se na aplicação de heurísticas locais e na adaptação para aplicações específicas, sendo assim tendem a apresentar um desempenho melhor que os algoritmos *Ant Colony System*, mas sua aplicabilidade é reduzida para os problemas para os quais foram desenvolvidos.

3.3 Parâmetros

Os sistemas de *Swarm Intelligence* possuem um determinado conjunto de parâmetros que configuram e modelam o comportamento do agente e do ambiente simulado, suas interações e efeitos. Tais parâmetros possuem valores empíricos e normalmente sofrem ajustes de forma não fundamentada.

Segundo [JUN 06] *appud* [CAR 04] e [SCH 96], com referência aos valores dos parâmetros do algoritmo AntNet, '*...é lembrado ainda que estes valores podem ser melhorados visto que não foram exaustivamente testados...*' .

Na figura 3.4 é apresentado um possível conjunto de parâmetros para

um algoritmo de *Swarm Intelligence* do tipo *AntNet*.

| Parâmetro | Valor | Descrição | Seção |
|------------|--------------------------|---|---------|
| Δt | 0.225 | Intervalo de criação dos agentes. | 4.1.2.1 |
| α | $\alpha \in [0, 1]$ 0.35 | Peso da correção heurística. | 4.1.3 |
| δ | $\delta \in [0, 1]$ 0.1 | Probabilidade de seleção uniforme. | 4.1.3 |
| ϵ | $\epsilon \geq 1$ 1.0 | Coefficiente de compressão da função exponencial. | 4.1.5 |
| η | $\eta \in (0, 1]$ 0.775 | Coefficiente da média exponencial. | 4.1.6.1 |
| c | $c \in (0, 1]$ 0.3 | Coefficiente do tamanho da janela móvel. | 4.1.6.1 |
| θ_1 | 0.7 | Peso para a razão entre o tempo de viagem atual e o melhor tempo para cálculo do reforço. | 4.1.6.3 |
| θ_2 | 0.3 | Peso para o valor da distância do valor de t de I_{inf} para o cálculo do reforço. | 4.1.6.3 |
| ζ | 0.6 | Coefficiente do nível de confiança atribuído. | 4.1.6.3 |
| a | $a \in \mathbb{R}^+$ 2.0 | Coefficiente da função de compressão. | 4.1.6.3 |

Figura 3.4: Conjunto de parâmetros do *AntNet* - Fonte: [JUN 06]

O objetivo da 3.4 é ilustrar a forma como são definidos os parâmetros nos algoritmos estudados. Pode-se notar que parâmetros, como por exemplo, Δt , que representa o intervalo de tempo entre a criação de agentes, e δ , que representa a probabilidade de um determinado agente seguir um caminho puramente aleatório, tem influência direta no comportamento dos agentes e conseqüentemente no desempenho geral do sistema. Os outros parâmetros podem não apresentar uma relação tão direta com o comportamento dos agentes, mesmos desta forma influenciam em maior ou menor grau o desempenho do sistema. Sendo assim pode-se concluir que o desempenho desses algoritmos esta fortemente relacionado com os parâmetros utilizados em sua configuração.

Capítulo 4

Metodologia e Simulações

Neste capítulo será apresentada a metodologia básica utilizada para a obtenção de parâmetros próximos aos valores ótimos. Esta metodologia esta fundamentada na teoria da percolação, ou seja, consiste na determinação dos valores de pontos críticos para determinadas características desejadas. Conforme abordado no capítulo 2 é impossível a determinação destes valores exatos para sistemas finitos e discretos, como é o caso dos algoritmos estudados. No entanto, por meio de artifícios numéricos podemos aproximar um valor com um erro admissível, este valor será determinado com base em simulações que tem por objetivo determinar o intervalo da zona crítica. Neste intervalo nada pode ser afirmado com relação a existência de percolação e por conseqüência, a convergência do algoritmo estudado para a solução do problema não pode ser garantida.

Para a metodologia proposta assume-se que o conjunto ótimo de parâmetros é aquele que com o mínimo de custo computacional soluciona o problema alvo, todas as vezes em que é executado. Admitindo-se um algoritmo de *swarm intelligence*, com o número de agentes ajustável, o conjunto de parâmetros ótimo para este algoritmo é aquele que com um número mínimo de agentes obtenha solução em 100% dos casos.

Para a aplicação da metodologia, proposta deste trabalho, foi implementado um simulador para um algoritmo baseado em *Ant Colony* e com base neste simulador gerou-se dados estatísticos para análise do desempenho do algoritmo com conjuntos variados de parâmetros com o objetivo de verificar se o comportamento emergente do

algoritmo com cada conjunto de parâmetros é capaz de construir um caminho estável de feromônio entre o ninho(origem) e o alimento(destino).

4.1 Sistema Proposto

A proposta deste trabalho é apresentar uma análise, com base na teoria da percolação, para algoritmos de *Swarm Intelligence*, com o objetivo de proporcionar um método válido para a definição dos parâmetros, de determinada aplicação, de forma objetiva.

Foi escolhido, como base para todas as simulações apresentadas neste trabalho, um algoritmo baseado no *Ant System* para o problema do menor caminho, esta escolha foi feita visando simplificar os passos propostos. Por ser um algoritmo mais simples que seus sucessores, as análises executadas sobre o *Ant System* mostraram-se mais sucintas e objetivas. Outro motivo para a escolha deste algoritmo é que por possuir um caráter genérico as análises apresentadas independem da aplicação esperada do algoritmo, ou seja, o comportamento do algoritmo não está relacionado com qualquer aplicação final deste. Acredita-se que sendo assim a metodologia e os passos propostos sejam validos para outros algoritmos de *Swarm Intelligence* que tenham características distintas do *Ant System*.

O objetivo do algoritmo implementado é que dado um ambiente de dimensões conhecidas e contendo um ninho, ponto de origem dos agentes/formigas, e um ponto com alimento, os agentes propostos devem ser capazes de formar um caminho estável de feromônio entre ninho e alimento, neste trabalho define-se caminho estável como uma trilha contínua de feromônio que ligue o alimento(destino) ou ninho(origem) e que possua quantidade suficiente de feromônio para manter-se desta forma. Este objetivo foi escolhido, pois representa o ponto fundamental do funcionamento de algoritmos desse tipo, ou seja, com a existência de um caminho estável entre alimento e ninho se torna possível o refinamento da solução com base no funcionamento subsequente do algoritmo. A escolha deste objetivo representa uma grande simplificação na implementação do algoritmo e por conseqüência um menor custo computacional para as simulações.

A simplificação do algoritmo tem por objetivo a redução no número de parâmetros com influência em seu comportamento. Isto é feito para tornar a metodologia proposta mais clara e objetiva, evitando problemas do domínio da aplicação. Para o sistema proposto os parâmetros adotados são: Número de Agentes, Dimensão do Ambiente e Taxa de Evaporação Relativa do Feromônio. Uma breve explanação sobre estes parâmetros se faz necessária:

- *Número de Agentes*: Quantidade absoluta de agentes utilizada em uma simulação;
- *Dimensão do Ambiente*: Tamanho total do ambiente proposto, nas simulações foi utilizado um grafo quadrado com conexões entre os vértices do tipo vizinhança-4, ou seja, cada vértice se conecta com os vértices a sua esquerda, direita, abaixo e acima;
- *Taxa de Evaporação Relativa do Feromônio*: Um valor diretamente proporcional a velocidade em que o feromônio depositado por determinado agente evapora. O quociente entre determinada quantidade de feromônio depositado e o número de ciclos de simulação necessários para que este evapore completamente.

Os parâmetros propostos foram aplicados ao algoritmo de forma a verificar a existência de pontos críticos de percolação.

4.2 Metodologia Proposta

Com base na teoria da percolação e no funcionamento geral dos algoritmos de *swarm intelligence* a metodologia proposta para a obtenção de parâmetros próximos aos ótimos pode ser discriminada da seguinte forma:

- *Passo 1*: Determinação do algoritmo desejado. O algoritmo usado para a resolução do problema deve ser definido a priori, pois as características do algoritmo tem influência direta nos valores obtidos, mesmo quando se tratam de algoritmos semelhantes;

- *Passo 2: Definição do Critério de Percolação.* Este critério representa a condição que, se verificada verdadeira, é considerada solução do problema alvo;
- *Passo 3: Análise dos Parâmetros.* No conjunto de todos os parâmetros devem ser identificados os parâmetros de maior relevância com relação ao Critério de Percolação escolhido, essa escolha deve ser feita com base no funcionamento do algoritmo e pode ser embasada em conhecimento prévio do comportamento geral do algoritmo ou em simulações com valores arbitrários;
- *Passo 4: Simulação do Algoritmo.* Através da utilização do subconjunto identificado no passo anterior simular o comportamento do algoritmo para valores variáveis dos parâmetros selecionados. Nota-se que a simulação possui uma complexidade exponencial, proporcional ao número de parâmetros selecionados, o que torna a cardinalidade deste conjunto crítica do ponto de vista das simulações. Nesta etapa é possível a aplicação de pseudo heurísticas a fim de minimizar o custo de simulação.
- *Passo 5: Identificação das Características Percolativas do algoritmo.* Com base na análise dos dados oriundos das simulações obter os valores dos pontos críticos e intervalos da zona crítica para o critério analisado. Nesta etapa é possível a identificação de parâmetros relevantes que não fazem parte do conjunto pré-definido, neste caso o novo parâmetro deve ser incluído neste conjunto e as simulações devem ser refeitas;
- *Passo 6: Modelagem do Sistema.* Com base nos valores de ponto crítico e intervalo de zona crítica é criada uma aproximação das equações características da percolação a fim de obter os valores interpolados para outras configurações do algoritmo. Desta forma, mesmo em ambientes com características variáveis o algoritmo deve apresentar melhora no desempenho, pois sendo assim é possível ajustar os valores dos parâmetros em questão durante a execução do algoritmo.

A figura 4.1 apresenta o fluxograma dos passos da metodologia proposta.

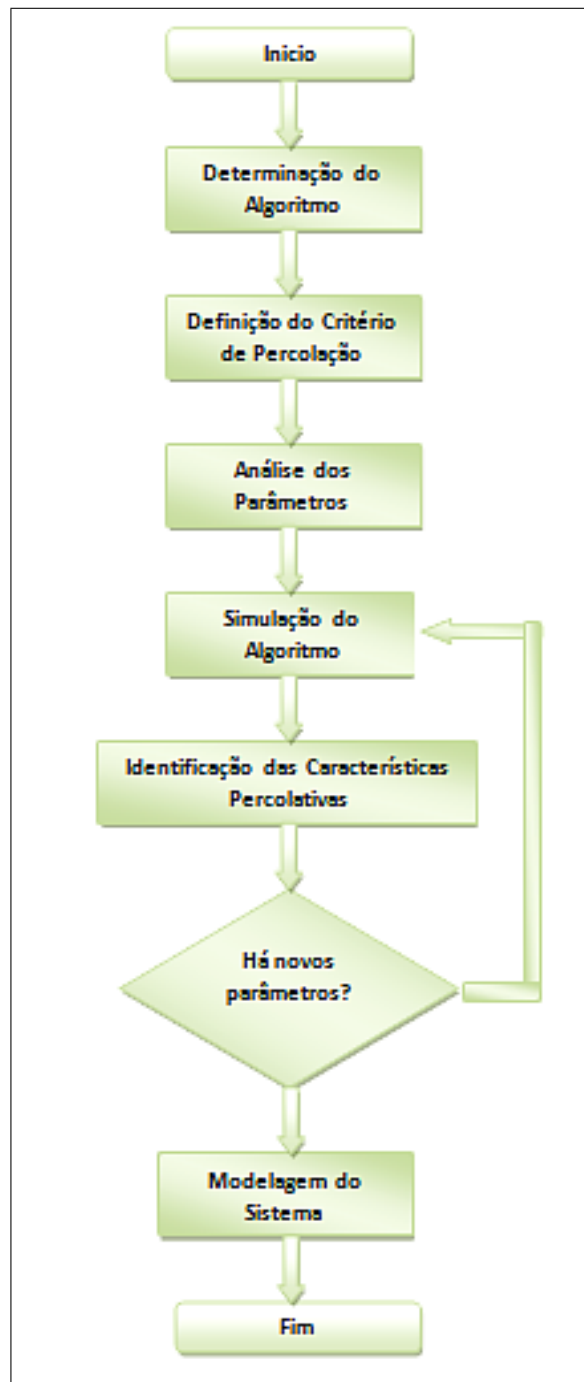


Figura 4.1: Fluxograma da Metodologia Proposta

O objetivo final da metodologia proposta é a obtenção das equações que modelam o comportamento do sistema baseado em determinados valores de parâmetros. A utilidade destas equações pode ser exemplificada por um algoritmo de roteamento ba-

seado em *swarm intelligence*, como o *AntNet*, para uma rede com *hosts* variáveis, ou seja, a rede tem tamanho e topologia dinâmicos, de forma que o algoritmo de roteamento seja capaz de se adaptar as variações das características da rede em tempo de execução mantendo o desempenho próximo ao ótimo em grande parte das situações.

Nestes algoritmos o roteamento impõe um fluxo de dados, pois muitas vezes cada agente é implementado por um pacote de dados na rede. Se a cada intervalo de tempo determinado, o algoritmo de roteamento verificar o número de computadores ativos em uma rede, com base neste número, seria possível configurar o algoritmo de roteamento para o mínimo de recursos necessários no momento, mantendo o fluxo de dados de roteamento próximo ao mínimo para a execução eficiente do roteamento, com menor utilização de recursos e sem perda de performance.

4.3 Aplicação da Metodologia

4.3.1 Definição do Algoritmo

De forma a analisar a metodologia proposta e exemplificar sua aplicação, será analisada a utilização de um algoritmo do tipo *Ant System* para a resolução do problema do menor caminho entre dois vértices de um grafo. A análise tem por objetivo a validação da metodologia e não apresenta uso prático do algoritmo *Ant System*, mas sim, uma situação hipotética e didática. O algoritmo foi simplificado a fim de minimizar sua complexidade tornando o entendimento mais claro e objetivo. As simplificações são apresentadas juntamente com a aplicação da metodologia e podem ser interpretadas como distorções da modelagem. Desta forma espera-se que o sistema proposto tenha familiaridade com os sistemas reais, mas seja menos complexo que estes, tornando a fase de simulação mais simples.

4.3.2 Definição do Critério de Percolação

Uma vez que o algoritmo foi definido como *Ant System*, deve-se definir o Critério de Percolação. Este critério representa a verificação de determinada condição

que seja considerada como solução para o problema proposto. No caso do *Ant System* o critério de percolação estabelecido foi a existência de um caminho estável de feromônio entre o ninho, vértice inicial, e o alimento, vértice final. O caminho estável é definido como uma trilha contínua de feromônio entre o ninho e o alimento que possua feromônio suficiente para manter-se desta forma por pelo menos mais um ciclo de simulação.

Este critério foi selecionado, pois com base em um caminho estável de feromônio a aplicação do algoritmo *Ant System* é capaz de obter o menor caminho possível ou no pior caso, como demonstrado em [DOR 96], o melhor caminho local. A existência deste caminho garante que o algoritmo encontrou uma solução para o problema. É importante salientar que a qualidade desta solução não é relevante neste estágio do trabalho, somente sua existência. O refinamento da solução é obtido pelo funcionamento contínuo do algoritmo.

4.3.3 Análise dos Parâmetros

A análise dos parâmetros do algoritmo proposto foi realizada através da observação do comportamento dos agentes em simulações com parâmetros conhecidos. Estas simulações apresentaram um comportamento que pode ser descrito da seguinte forma:

- *Característica 1:* Os agentes foram capazes de construir um caminho estável de feromônio entre o ninho (origem) e o alimento (destino), evento definido como Critério de Percolação para determinado conjunto inicial de parâmetros;
- *Característica 2:* O aumento das dimensões do ambiente sem modificação de outros parâmetros causou a não obtenção do Critério de Percolação;
- *Característica 3:* O aumento da taxa de evaporação do feromônio sem modificação de outros parâmetros causou a não obtenção do Critério de Percolação;
- *Característica 4:* O aumento do número de agentes causou a obtenção do Critério de Percolação para as duas situações anteriores (Características 2 e 3);

- *Característica 5:* Existe correlação entre número de agentes, taxa de evaporação, dimensões do ambiente e percolação.

Com base no comportamento descrito e observado no algoritmo, os parâmetros selecionados foram: Dimensões do Ambiente, Número de Agentes e Taxa de Evaporação.

4.3.4 Simulações

Algumas considerações foram tomadas para a construção do simulador:

- A taxa de evaporação deve ser suficiente para que um único agente não seja capaz de produzir um caminho contínuo entre o alimento e o ninho. Nas simulações é calculada como o suficiente para formação de um caminho de comprimento igual a uma fração da semi-diagonal do ambiente;
- O grafo do ambiente proposto é quadrado e não possui obstáculos; A percolação é obtida quando existe um caminho de feromônio entre o alimento e o formigueiro;
- O Número de ciclos de simulação deve ser suficiente para que se esgote a comida mesmo sem a presença de percolação;
- Cada conjunto de parâmetros deve ser testado diversas vezes a fim de diminuir os erros estatísticos e distorções relativas da utilização de um espaço finito com elementos discretos.

Com relação aos agentes propostos uma série de comportamentos foi implementada:

- Cada agente inicia seu ciclo de vida no formigueiro;
- A movimentação inicial é aleatória, no entanto cada agente possui uma probabilidade maior de se afastar do formigueiro. Isto se faz necessário uma vez que se o movimento for puramente aleatório o deslocamento tende a zero;

- O agente tem o tempo necessário para percorrer o perímetro completo do ambiente, se durante este tempo não localizar comida ou feromônio o agente retorna ao formigueiro e inicia um novo ciclo;
- Sempre que um agente cruza uma trilha de feromônio este segue a trilha no sentido que o distancia do formigueiro;
- Quando um agente encontra alimento, este retorna em direção ao formigueiro;
- Cada agente retira uma pequena parte do montante de alimento;
- Cada agente possui sensibilidade para o feromônio em sua vizinhança direta, Vizinhança Oito.

Desta forma podem-se perceber assim, algumas características dos agentes implementados: cada agente possui o conhecimento do local do formigueiro, o agente não tem conhecimento de outros agentes, o agente não possui memória e tem comportamento puramente reflexivo. Como apresentado a construção e características de cada agente o torna simples, como esperado a propriedade emergente do uso de diversos agentes é a resolução do problema de menor caminho.

As características dos agentes implementados diferem, em alguns aspectos, dos agentes clássicos dos sistemas *Ant System*. Estas diferenças no entanto, encontram-se no fato do algoritmo implementado não refinar a solução obtida. Esta escolha foi feita para tornar a simulação menos custosa em termos computacionais e, como o critério de percolação escolhido se verifica antes da solução final do problema de menor caminho, o comportamento obtido pelo simulador é próximo do esperado de um algoritmo *Ant System*.

A localização do formigueiro é fixa e, em todas as simulações, é localizado no centro do ambiente. A fonte de alimento tem localização aleatória mas, com restrições quanto à proximidade, podendo estar localizada somente nos extremos do ambiente.

Foi implementada a possibilidade de visualização do comportamento dos agentes com base no sistema de gráficos GNUPlot[GNU 08], com o objetivo de fa-

cilitar o acompanhamento da simulação. As figuras 4.2 e 4.3 apresentam dois estado do sistema. Esta facilidade foi utilizada durante a fase de testes do simulador, mas teve de ser abandonada durante as simulações, pois o custo computacional e o tempo de atualização de tela se mostraram elevados tornando as simulações inviáveis.

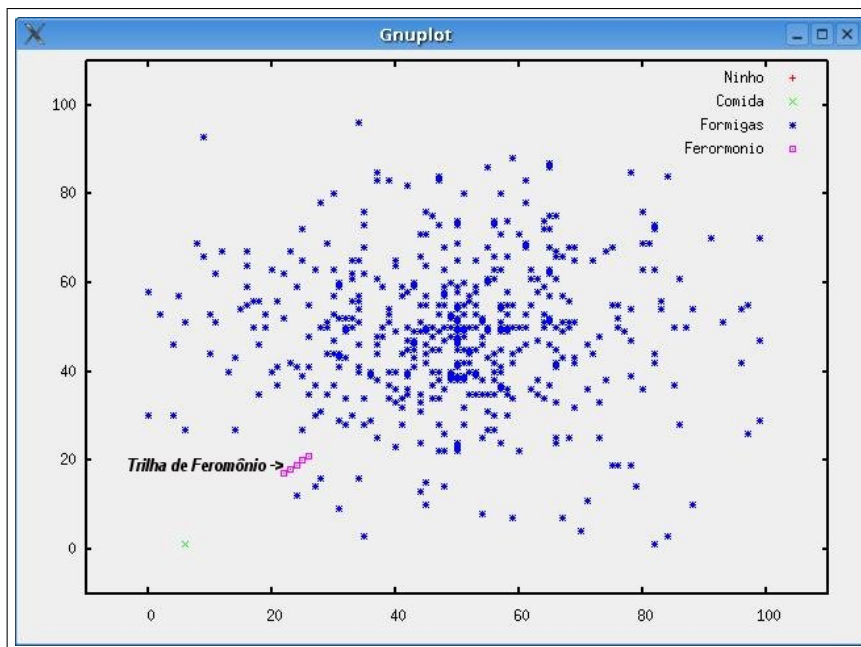


Figura 4.2: Saída do Simulador - Retorno de um agente com trilha de Feromônio incompleta

O simulador tem como saída um arquivo contendo dados sobre todos os ciclos da simulação de determinada propriedade, a saber: Tempo em ciclos, Número de Agentes, Resultado da Percolação, Alimento Consumido, Alimento Restante, Tamanho do Ambiente e Taxa de Evaporação.

As simulações realizadas tiveram por base o seguinte procedimento: determinadas propriedades foram estabelecidas, como o tamanho do ambiente, e são fixas para cada simulação. Com relação a este parâmetro é calculada a taxa de evaporação levando em conta as restrições já abordadas. Cada ciclo de simulação é encerrado por dois eventos: existe percolação, ou seja, foi encontrado um caminho entre alimento e formigueiro, ou o número máximo de iterações é atingido.

Neste trabalho entende-se por ciclo de simulação a execução completa do algoritmo, ou seja, cada ciclo de simulação tem por resultado principal a existência ou não da percolação. Da mesma forma uma iteração diz respeito a uma unidade de tempo do algoritmo ou a atualização do estado dos agentes. Nas simulações foi adotada uma densidade de agentes inicial, com relação à área total do ambiente, e uma densidade final, bem como um passo de incremento. Cada ciclo de determinada densidade é executado um determinado número de vezes e após a densidade é elevada com a soma do passo de incremento, seguindo desta forma enquanto a densidade final não for atingida.

Na tabela 4.1 é apresentado um conjunto de parâmetros submetido ao simulador como exemplo de funcionamento e denominado Simulação 1. O gráfico da figura 4.4 foi obtido com base nos dados oriundos da simulação.

Na tabela 4.1 as unidades apresentadas são: unidade de área(u.a.); unidade de tempo(u.t.); unidade de alimento(u.c.); unidade de feromônio(u.g.); unidade de agentes(u.f.). Estas unidades são próprias do sistema e tem seus correspondentes nas características do simulador. Por exemplo uma unidade de área é o espaço ocupado por

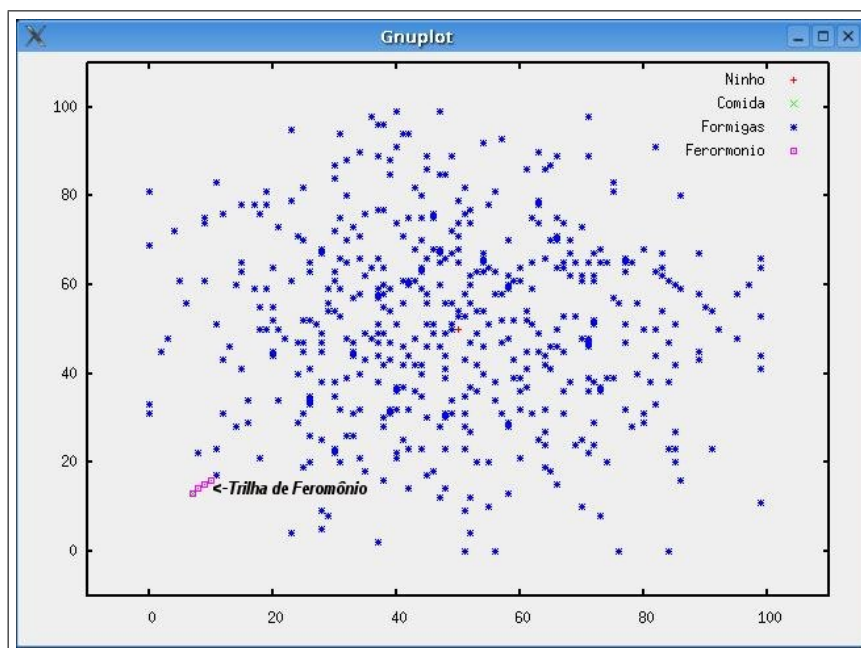


Figura 4.3: Saída do Simulador - Início do retorno de um agente e trilha de Feromônio

| Parametro | Valor |
|------------------------------------|------------------|
| Dimensões do Ambiente | 100x100 u.a. |
| Máximo de Iterações por Ciclo | 10000 u.t. |
| Alimento Disponível | 10000 u.c. |
| Densidade Inicial | 0,01 u.f. / u.a. |
| Densidade Final | 0,9 u.f. / u.a. |
| Depósito de Feromônio | 23 u.g. |
| Taxa de Evaporação | 1 u.g. / u.t. |
| Quantidade de Ciclos por Densidade | 50 |
| Passo de Incremento de Densidade | 0,1 u.f. / u.a |

Tabela 4.1: Parâmetros da simulação-exemplo

um agentes e também pode ser interpretado como um vértice do grafo. As unidades de alimento são constantes que medem o desempenho dos agentes na coleta de alimento utilizado na análise do algoritmo e não tem influência direta em seu funcionamento. Uma unidade de tempo corresponde a uma interação dentro de um ciclo de simulação e uma unidade de feromônio corresponde à quantidade mínima de feromônio necessária para ser detectado por um agente.

A análise da figura 4.4 mostra as características esperadas de um sistema percolativo, onde pode-se observar a mudança de fase em determinado ponto. Este ponto não é definido devido as distorções resultantes da utilização de um ambiente finito com elementos discretos. No entanto pode-se observar que para os pontos com densidade de agentes superiores a aproximadamente $0,15 u.f./u.a.$ o sistema converge para a solução do problema, isto é, a ocorrência de percolação. Em 100% dos casos, este ponto pode ser considerado o limite superior da zona crítica. Para valores de Densidade de Agentes aproximadamente $0,04 u.f./u.a.$ a percolação, ou seja, a solução do problema, foi encontrada apenas em 50% das simulações.

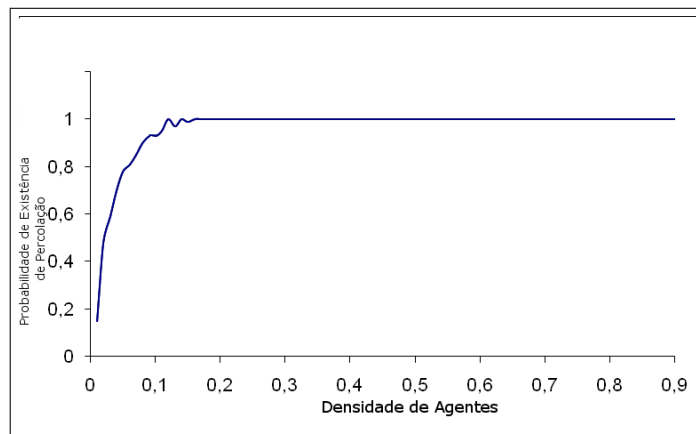


Figura 4.4: Gráfico de Percolação x Densidade de Agentes obtido com a simulação 1

4.4 Considerações finais sobre a metodologia proposta

Neste capítulo apresentou-se a metodologia proposta para análise paramétrica de algoritmos de *swarm intelligence* com base na teoria da percolação. A aplicação desta metodologia tem por componente fundamental o simulador do algoritmo proposto. Seu funcionamento foi ilustrado, em parte, pela aplicação dos passos iniciais da metodologia proposta. Uma análise aprofundada e o resultado de simulações mais complexa, bem como dos passos finais da metodologia são apresentados no capítulo 5.

Capítulo 5

Resultados

Neste capítulo é apresentada a aplicação dos passos finais da metodologia proposta no capítulo 4, para um sistema baseado em um algoritmo *Ant System*. A simulação apresentada no capítulo 4 (tabela 4.1 e figura 4.4) teve por objetivo ilustrar o funcionamento do simulador. Neste capítulo esta simulação será estendida com a finalidade de evidenciar o caráter percolativo dos sistemas estudados, uma nova simulação será utilizada a fim de obter-se dados mais completos e com características mais próximas as necessárias para a análise completa do algoritmo. Esta simulação possui um conjunto mais extenso de parâmetros e uma discretização maior dos elementos e será definida como Simulação 2. Isto é necessário a fim de abordar todos os parâmetros definidos como de interesse pela metodologia proposta. Na tabela 5.1 é apresentada a configuração da Simulação 2. Na figura 5.1 é apresentado o gráfico da análise de percolação.

O gráfico da figura 5.1 foi submetido a um filtro de média para retirar ruídos oriundos da discretização maior dos elementos, definido pelo parâmetro Passo de Incremento de Densidade, ou seja, cada ponto do gráfico apresentado é a média de um determinado número de pontos no seu entorno. No caso do filtro apresentado o tamanho do entorno é igual a 10. Este filtro é aplicado em todos os gráficos apresentados neste capítulo.

O gráfico da figura 5.1 apresenta as características esperadas, esta simulação foi interrompida antes da densidade máxima de agentes uma vez que após o limite supe-

| Parâmetro | Valor |
|------------------------------------|-------------------|
| Dimensões do Ambiente | 100x100 u.a. |
| Máximo de Iterações por Ciclo | 10000 u.t. |
| Alimento Disponível | 10000 u.c. |
| Densidade Inicial | 0,001 u.f. / u.a. |
| Densidade Final | 0,1 u.f. / u.a. |
| Depósito de Feromônio | 13 u.g. |
| Taxa de Evaporação | 1 u.g. / u.t. |
| Quantidade de Ciclos por Densidade | 50 |
| Passo de incremento de Densidade | 0,001 u.f. / u.a. |

Tabela 5.1: Parâmetros da simulação 2

rior da zona crítica, que coincide aproximadamente com a densidade final da simulação, os valores esperados são constantes e iguais a 1.

No gráfico da figura 5.2 é apresentado o gráfico dos valores de desvio padrão de cada ponto do gráfico de percolação. Pode-se notar, mais acentuadamente, as distorções características do uso de elementos finitos também presentes no gráfico da figura 5.1, e caracterizados por uma pequena elevação próxima ao zero. Segundo [LEU 03a] este efeito é esperado e decorrente do uso de um ambiente de dimensões finitas. Com base nestes valores pode-se apresentar o gráfico da figura 5.3 com os valores de variância dos pontos.

Baseado na teoria da percolação a variância do Ponto Crítico tende ao infinito. Nos gráficos da simulação, figura 5.3, observa-se um ponto onde a variância é máxima, mas não infinita. Isso se deve aos valores de percolação resultantes do simulador, 0 (zero) no caso de não haver percolação e 1 caso contrário. Desta forma os valores 0,5 para o desvio padrão e 0,25 para variância, são, por sua vez, os valores máximos possíveis nestas situações. Seria errado afirmar que estes valores indicam o ponto crítico de percolação uma vez que com base na teoria este ponto não pode ser definido para

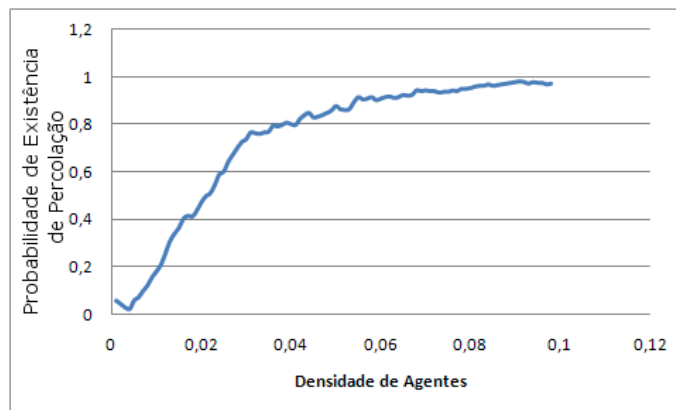


Figura 5.1: Gráfico de Percolação x Densidade de agentes para a simulação 2

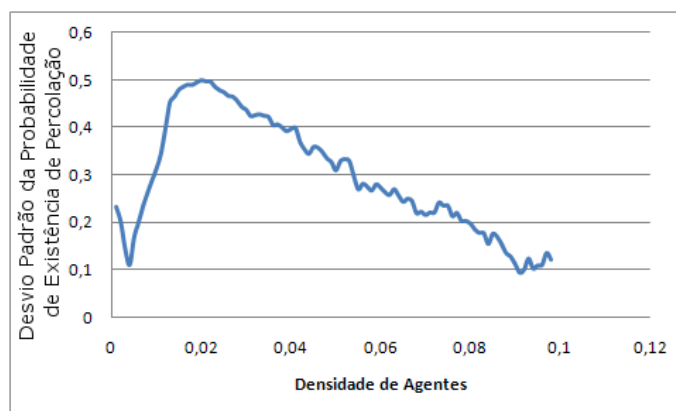


Figura 5.2: Gráfico de Desvio Padrão x Densidade de Agentes - Simulação 2

sistemas finitos e discretos, no entanto representam uma estimativa aceitável deste valor. Pode-se afirmar também, com relativo grau de segurança, que a zona crítica está definida dentro do intervalo onde os valores de variância são maiores que o mínimo.

Outras características podem ser analisadas com base nos dados oriundos da simulação 2, a figura 5.4 apresenta o gráfico que relaciona o tempo de simulação com a densidade de agentes.

O gráfico da figura 5.4 apresenta o tempo de simulação em iterações e para pequenos valores de densidade de agentes o tempo de simulação é próximo ao máximo definido de 10000 iterações, no entanto com a proximidade do limite superior da zona crítica este valor tende a um número de iterações próximo a 800. Uma simulação

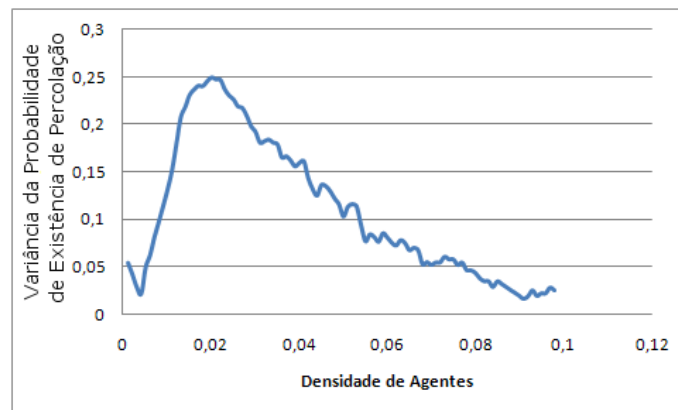


Figura 5.3: Gráfico de Variância x Densidade de Agentes - Simulação 2

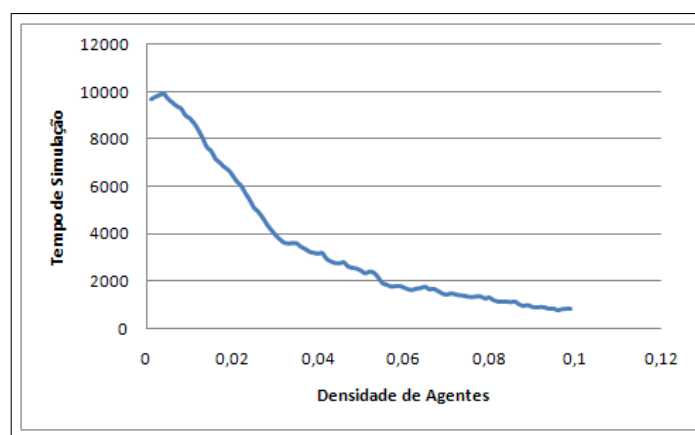


Figura 5.4: Gráfico de Tempo de simulação x Densidade de Agentes - Simulação 1

com valores mais elevados de densidades de agentes não apresentou uma melhora substancial neste tempo. Como o tempo em iterações não se relaciona com o tempo real de simulação é esperado que o uso de um número elevado de agentes degrade a performance geral do sistema, uma vez que um número maior de agentes implica um custo computacional maior e por sua vez um tempo de processamento maior. Com base neste fato pode-se afirmar que mesmo obtendo um número menor de iterações um algoritmo com um número maior de agentes apresentará um tempo total de execução maior que o necessário.

Na figura 5.5 é apresentado o gráfico de Alimento Coletado por Densidade de agentes para a simulação 2 este gráfico apresenta um resultado que corrobora

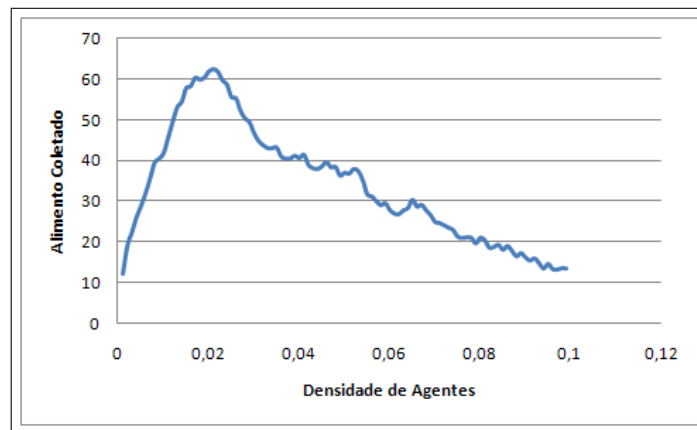


Figura 5.5: Gráfico de Alimento Coletado x Densidade de Agentes - Simulação 2

o resultado da análise da figura 5.4 uma vez que nas proximidades do limite superior da zona crítica a quantidade de alimento coletada equivale a de um único agente (10 u.c. definido para todas as simulações) indicando que um caminho estável de feromônio foi obtido na primeira vez em que a posição do alimento foi localizado por um agente, o que para valores menores de densidade de agentes, não é verdade. Nestas densidades mesmo após a localização do alimento os agentes não foram capazes de formar um caminho estável entre o alimento (destino) e o ninho (origem).

5.1 Resultados da aplicação da metodologia proposta

Os resultados da simulação 2 apresentam as características percolativas do algoritmo implementado no entanto, esta simulação não apresenta as propriedades necessárias para análise do algoritmo *Ant System* com base na metodologia proposta. Os parâmetros definidos como relevantes pela aplicação da metodologia são: Tamanho do ambiente; Densidade de Agentes e Taxa de evaporação de feromônio. Nas simulações 1 e 2 somente uma destas características é abordada, a densidade de agentes. Para a análise completa dos parâmetros relevantes deve-se definir outro conjunto de simulações, este conjunto em nada difere do conjunto inicial a não ser por variar todos os parâmetros relevantes. A definição da simulação pode ser vista na tabela 5.2 e esta será referenciada

| Parametro | Valor |
|------------------------------------|---|
| Dimensões do Ambiente | 50x05 - 100x100 - 150x150 u.a. |
| Máximo de Iterações por Ciclo | 10000 u.t. |
| Alimento Disponível | 10000 u.c. |
| Densidade Inicial | 0,001 u.f. / u.a. |
| Densidade Final | 0,1 u.f. / u.a. |
| Depósito de Feromônio | 50x05(8 - 12 - 16 - 20 - 24)u.g. 100x100(13 - 18 - 23 - 28 - 32 - 37)u.g. 150x150(35 - 40 - 45 - 50 - 55)u.g. |
| Taxa de Evaporação | 1 u.g. / u.t. |
| Quantidade de Ciclos por Densidade | 50 |
| Passo de incremento de Densidade | 0,001 u.f. / u.a |

Tabela 5.2: Parâmetros da simulação completa

como Simulação 3.

É importante salientar que os dados da tabela 5.2, referentes aos valores de Dimensão do Ambiente e Depósito de Feromônio, são valores discretos e não intervalos de valores simulados. Estes valores foram arbitrados com base no funcionamento prévio do sistema e dependem da aplicação proposta. Desta forma pretende-se obter equações que tornem possível a interpolação de outros valores de interesse.

Na figura 5.6 é apresentado o gráfico de percolação para a simulação 3 com as dimensões do ambiente definidas para 50x50 u.a.

Na figura 5.7 é apresentado o gráfico de percolação para a simulação 3 com as dimensões do ambiente definidas para 100x100 u.a.

Na figura 5.8 é apresentado o gráfico de percolação para a simulação 3 com as dimensões do ambiente definidas para 150x150 u.a.

De posse dos dados que dão origem as figuras 5.6, 5.7 e 5.8 pode-se obter os valores apresentados nas tabelas 5.3, 5.4 e 5.5 com a aplicação da metodologia

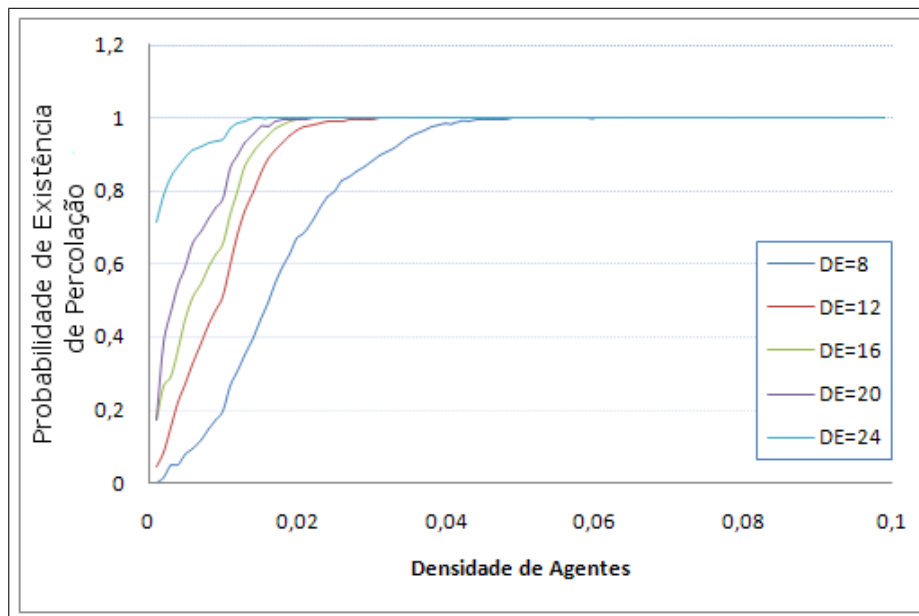


Figura 5.6: Gráfico de Percolação x Densidade de Agentes - Simulação 3 - Em ambiente 50 x 50 com variação no valor de Depósito de Feromônio(DE) de 8 a 24

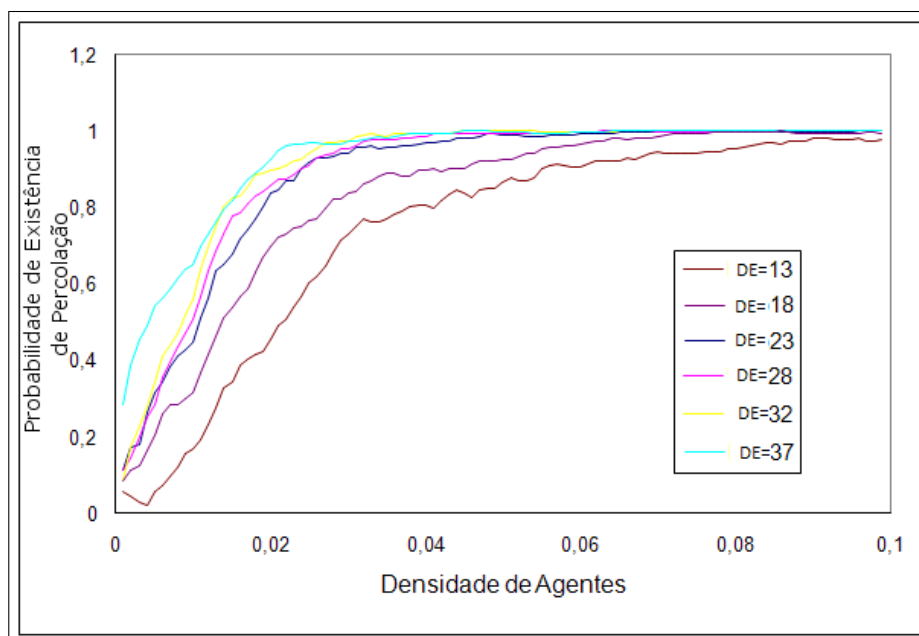


Figura 5.7: Gráfico de Percolação x Densidade de Agentes - Simulação 3 - Em ambiente 100 x 100 com variação no valor de Depósito de Feromônio(DE) de 13 a 37

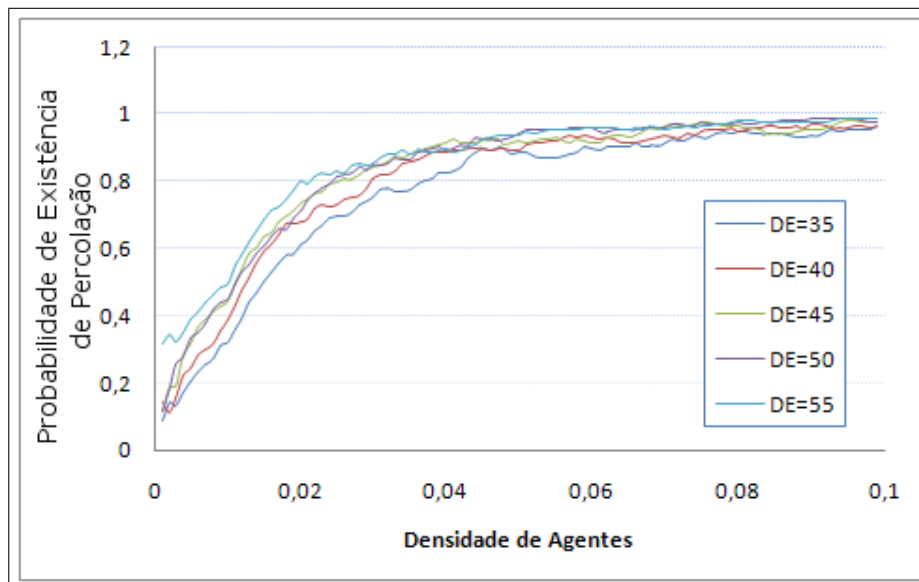


Figura 5.8: Gráfico de Percolação x Densidade de Agentes - Simulação 3 - Em ambiente 150×150 com variação no valor de Depósito de Feromônio (DE) de 35 a 55

discutida neste trabalho. A constante C é calculada com base na equação 2.2 com o valor de $v = 1,33$, conforme [EFR 87], este é o valor apropriado para problemas de duas dimensões, e sua determinação foge ao escopo deste trabalho.

Nas tabelas os valores da coluna DE são referentes a quantidade de feromônio depositado pelo agente, a coluna P_c contém os valores do Ponto Crítico encontrado para cada simulação, os valores da coluna $\neg \delta(N)$ são referentes ao limite superior da zona crítica assim como $\delta(N)$ contém os valores de amplitude da zona crítica de cada simulação. Por fim é apresentada a constante C calculada com base na equação 2.2.

5.2 Cálculo de Parâmetros para o *Ant System*

O cálculo dos valores dos parâmetros para o algoritmo *Ant System* leva em consideração os valores apresentados como resultado das simulações. Estes valores são obtidos por meio de interpolação, e a técnica numérica utilizada para tal não é o foco deste trabalho, podendo ser utilizada qualquer técnica que melhor se adapte ao problema proposto. As técnicas utilizadas neste trabalho são: Regressão não linear geométrica,

| DE | Pc | $-\delta(N)$ | $\delta(N)$ | C |
|-----------|-----------|--------------|-------------|----------|
| 35 | 0,017 | 0,150 | 0,266 | 11,509 |
| 40 | 0,016 | 0,140 | 0,248 | 10,730 |
| 45 | 0,014 | 0,120 | 0,212 | 9,173 |
| 50 | 0,012 | 0,100 | 0,176 | 7,615 |
| 55 | 0,010 | 0,092 | 0,164 | 7,096 |

Tabela 5.3: Conjunto de Valores de Ponto Crítico, Zona Crítica e Constante C para Ambiente 150x150 com Quantidade de Feromônio Depositado(DE) de 35 a 55

| DE | Pc | $-\delta(N)$ | $\delta(N)$ | C |
|-----------|-----------|--------------|-------------|----------|
| 13 | 0,0220 | 0,112 | 0,180 | 5,742 |
| 18 | 0,0140 | 0,072 | 0,116 | 3,700 |
| 23 | 0,0110 | 0,060 | 0,098 | 3,126 |
| 28 | 0,0100 | 0,041 | 0,062 | 1,978 |
| 32 | 0,0085 | 0,036 | 0,055 | 1,754 |
| 37 | 0,0040 | 0,031 | 0,054 | 1,722 |

Tabela 5.4: Conjunto de Valores de Ponto Crítico, Zona Crítica e Constante C para Ambiente 100x100 com Quantidade de Feromônio Depositado(DE) de 13 a 37

| DE | Pc | $-\delta(N)$ | $\delta(N)$ | C |
|-----------|-----------|--------------|-------------|----------|
| 8 | 0,0160 | 0,042 | 0,052 | 0,985 |
| 12 | 0,0100 | 0,027 | 0,034 | 0,644 |
| 16 | 0,0060 | 0,020 | 0,028 | 0,530 |
| 20 | 0,0035 | 0,018 | 0,029 | 0,549 |
| 24 | 0,0000 | 0,013 | 0,026 | 0,492 |

Tabela 5.5: Conjunto de Valores de Ponto Crítico, Zona Crítica e Constante C para Ambiente 50x50 com Quantidade de Feromônio Depositado(DE) de 8 a 24

técnica implementado por meio de uma fração onde se calcula o valor de denominador em função de um numerador variável por meio iterativo; Interpolação Polinomial de Newton que interpola valores pontuais por meio de um polinômio calculado com base no método das diferenças finitas. Estes métodos foram escolhidos, pois apresentam uma menor complexidade na construção das equações, mas podem ser substituídos por métodos mais eficientes. É desejável neste trabalho que as equações apresentem uma complexidade baixa a fim de tornar o entendimento geral do método mais claro.

5.2.1 Equações Interpoladas

A fim de obter-se equações genéricas para modelar o problema proposto, inicia-se neste caso pela interpolação de uma função para cálculo da amplitude da zona crítica. Esta amplitude pode ser obtida, segundo a teoria da percolação, pela equação 2.2 rerepresentada na equação 5.1. Esta equação aborda o caso geral para elementos infinitesimais e deve ser interpolada para cada um dos conjuntos de parâmetros simulados. A figura 5.9 apresenta a interpolação, por regressão não linear, para o cálculo do valor da constante C em um ambiente de 100×100 u.a., as figuras 5.10 e 5.11 apresentam as regressões não lineares para a constante C em ambiente de área igual a 150×150 u.a. e 50×50 u.a. respectivamente. É importante ressaltar que o número de pontos para a interpolação, resultantes destas simulações, é pequeno. Esta limitação se deve ao fato das simulações imporem um tempo total proibitivo, devido a limitações de processamento por parte dos computadores disponíveis.

$$\delta(N) = \frac{C}{N^{1/2v}} \quad (5.1)$$

A regressão não linear do gráfico da figura 5.9 é apresentada na equação 5.2. Com as equações 5.2 e 5.1 tem-se as equações 5.3 para problemas bidimensionais. A constante v mantém-se $v = 1,33$, conforme abordado anteriormente na sessão 5.1.

$$C_{100 \times 100}(DE) = \frac{68,833}{DE} \quad (5.2)$$

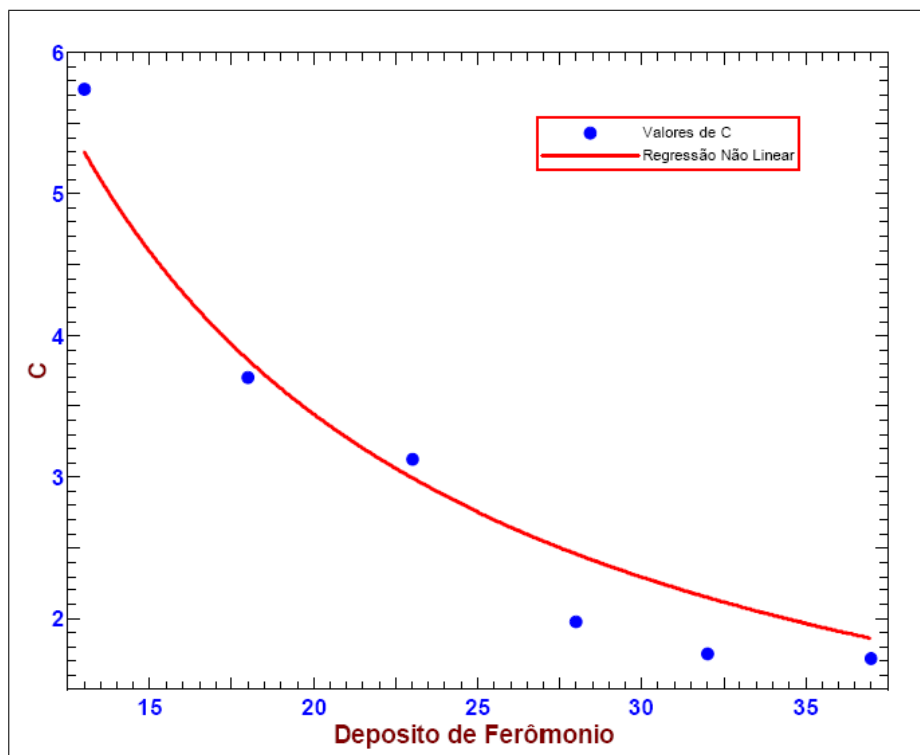


Figura 5.9: Polinômio interpolado para cálculo de C em ambiente 100×100

$$\delta_{100 \times 100}(N, DE) = \frac{68,833}{DE \cdot N^{1/2v}} \quad (5.3)$$

A regressão não linear do gráfico da figura 5.10 é apresentado na equação 5.4. Com as equações 5.4 e 5.1 tem-se a equações 5.5 para problemas bidimensionais.

$$C_{150 \times 150}(DE) = \frac{402,286}{DE} \quad (5.4)$$

$$\delta_{150 \times 150}(N, DE) = \frac{402,286}{DE \cdot N^{1/2v}} \quad (5.5)$$

A regressão não linear do gráfico da figura 5.11 é apresentado na equação 5.6. Com as equações 5.6 e 5.1 tem-se a equações 5.7 para problemas bidimensionais.

$$C_{50 \times 50}(DE) = \frac{8,396}{DE} \quad (5.6)$$

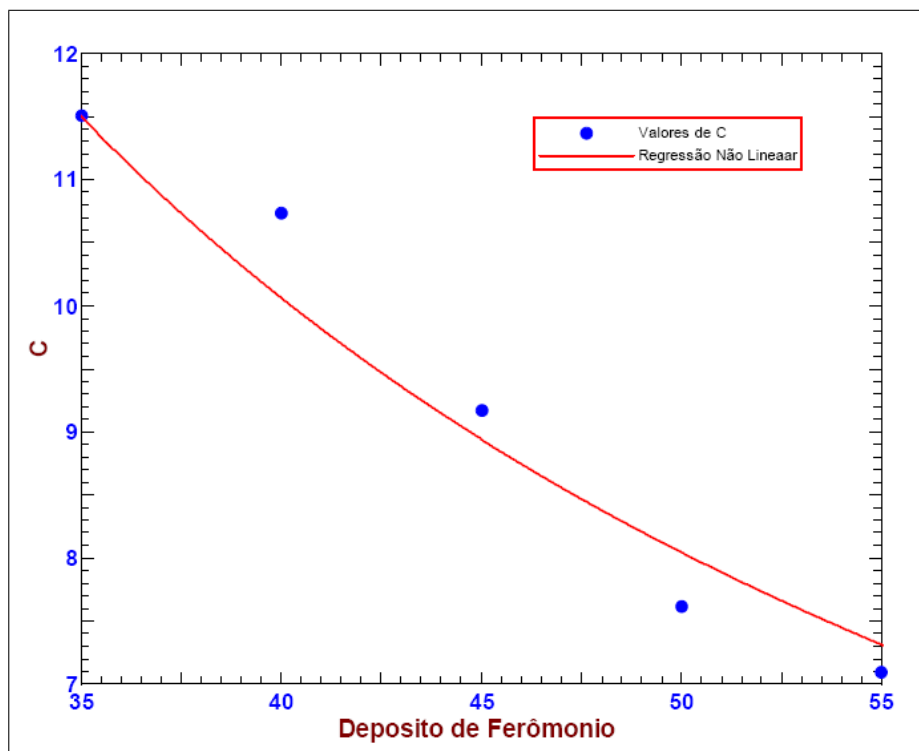


Figura 5.10: Polinômio interpolado para cálculo de C em ambiente 150×150

$$\delta_{50 \times 50}(N, DE) = \frac{8,396}{DE \cdot N^{1/2v}} \quad (5.7)$$

De posse das equações interpoladas por área de ambiente, pode-se por sua vez, com base na similaridade das equações 5.3, 5.5 e 5.7, interpolar por meio de polinômios, para a determinação dos valores das constantes numéricas apresentadas nos denominadores das equações, uma equação geral para cálculo da amplitude da zona crítica para o problema *Ant System* proposto. Na equação 5.8 é apresentado a forma final da solução proposta para cálculo da zona crítica, que possibilita a determinação da amplitude da zona crítica para algoritmos que possuem seu conjunto de parâmetros dentro dos intervalos definidos nas simulações.

O método de interpolação por polinômios foi escolhido por apresentar um erro menor que a regressão não linear usada nos passos anteriores, e por não comprometer o entendimento geral do processo, podendo, da mesma forma que nos métodos

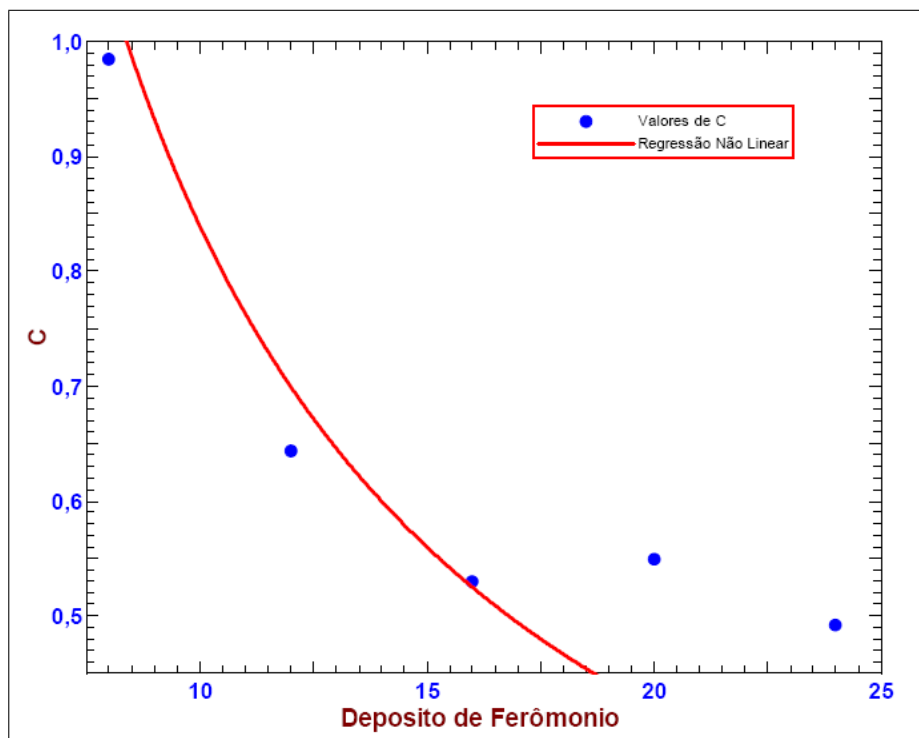


Figura 5.11: Polinômio interpolado para cálculo de C em ambiente 50x50

anteriores, ser substituído por outras técnicas que forem julgadas mais apropriadas.

$$\delta(N, DE) = \frac{a.N^2 + b.N + c}{DE.N^{1/2v}}, \text{ onde :}$$

$$a = 9,31 \times 10^{-7},$$

$$b = -0,003578,$$

$$c = 11,523 \quad (5.8)$$

A figura 5.12 apresenta o ajuste do polinômio interpolado aos valores das constantes das equações originais. O polinômio apresentado na figura mapeia uma função do tipo $f(N) \rightarrow C$ obtém-se desta forma os valores interpolados das constantes numéricas C das equações 5.3, 5.5 e 5.7 em função da quantidade de nós do ambiente N . Os dados completos referentes as análise dos métodos numéricos empregados encontram-se nos apêndices deste trabalho.

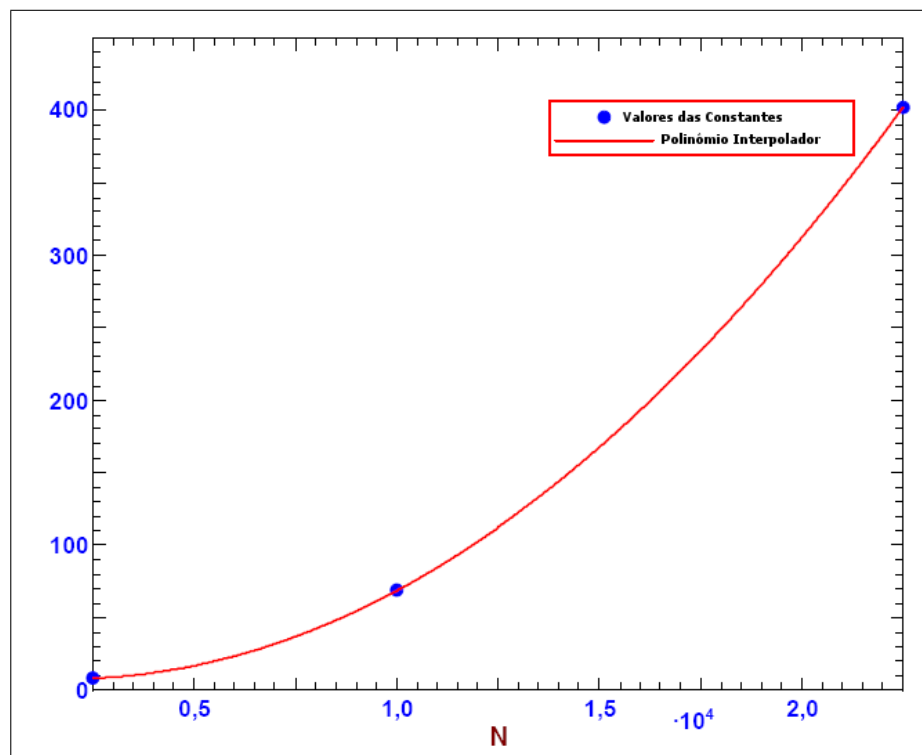


Figura 5.12: Polinômio interpolado x Valores das constantes

Com a obtenção da equação que torna possível o cálculo da amplitude aproximada da zona crítica resta o posicionamento deste valores, uma vez que a amplitude da zona crítica não implica diretamente os valores de densidade de agentes necessários. O centro da zona crítica encontra-se no ponto crítico, e este com base na teoria não pode ser determinado algebricamente. Aqui se propõe duas abordagens para a localização deste valor.

O primeiro método para posicionamento da zona crítica consiste em de forma empírica, por meio de nova simulação dos parâmetros desejados, localizar o início da zona crítica, ponto que pode ser determinado como a mínima densidade de agentes onde a variância da probabilidade de existência de percolação apresenta valores maiores que os mínimos.

O ponto determinado por esta nova simulação pode ser admitido como início da zona crítica e somado ao valor calculado na equação 5.8 representa o início da zona onde a percolação sempre é obtida, ou seja, o limite superior da zona crítica.

Sendo, este ponto que indica o valor aproximado do ótimo para a densidade de agentes. A figura 5.13 apresenta o ponto inicial da zona crítica em um gráfico de variância para uma aplicação hipotética.

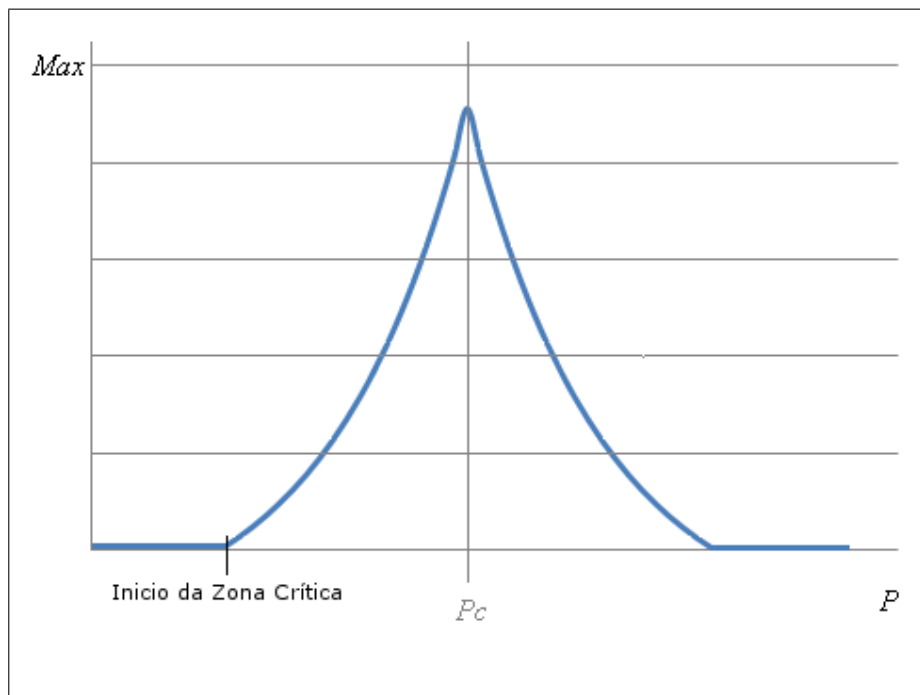


Figura 5.13: Limite inferior da zona crítica em um gráfico de variância da probabilidade de existência de percolação x densidade de agentes

O método de determinação da posição da zona crítica pela simulação dos valores iniciais, não apresenta a flexibilidade esperada da metodologia proposta. Por tratar-se de uma extensão das simulações iniciais não é indicado para todas as situações e aplicações, pois necessita de um nível elevado de processamento para sua aplicação prática. Este método não será profundamente avaliado neste trabalho, mas é apresentado como alternativa para aplicações onde não é desejável ou possível a aplicação de outro método mais apropriado para a determinação do ponto crítico.

O cálculo do ponto crítico pode ser efetuado por meio de uma nova interpolação. Utilizando as mesmas técnicas usadas na interpolação da equação da zona crítica obtemos as equações para a regressão não linear do ponto crítico para ambientes 50x50u.a., 100x100u.a. e 150x150u.a. figuras 5.14, 5.15 e 5.16 e equações 5.9, 5.10 e

5.11 respectivamente.

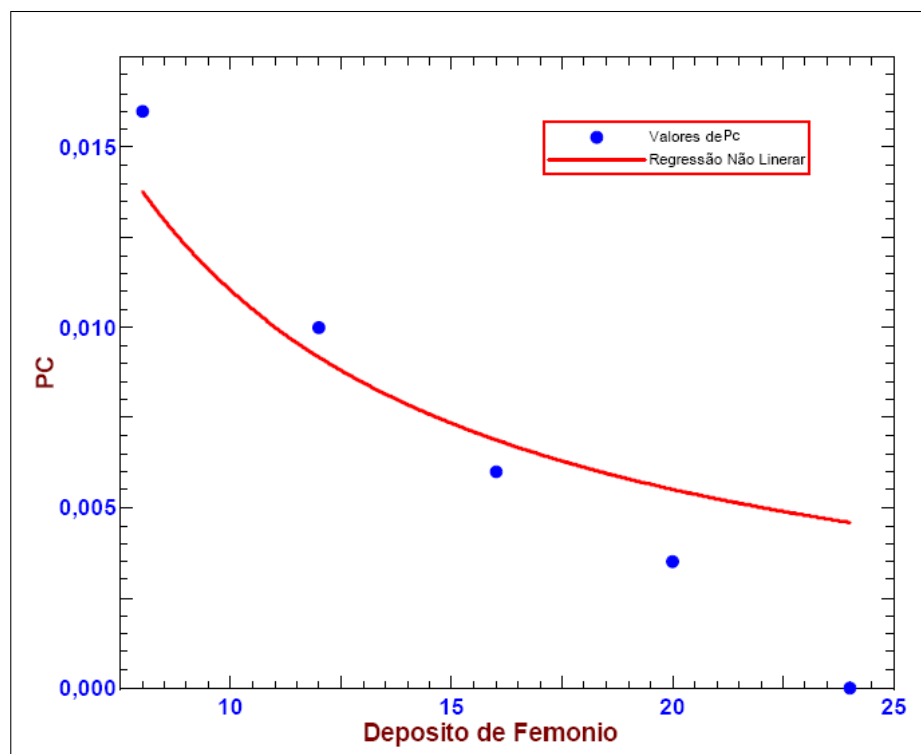


Figura 5.14: Regressão Não Linear em função da quantidade de feromônio depositada para cálculo de P_c em ambiente 50x50

$$P_{C_{50 \times 50}}(DE) = \frac{0,111}{DE} \quad (5.9)$$

$$P_{C_{100 \times 100}}(DE) = \frac{0,265}{DE} \quad (5.10)$$

$$P_{C_{150 \times 150}}(DE) = \frac{0,607}{DE} \quad (5.11)$$

Na figura 5.14 nota-se uma distorção nos valores do ponto crítico obtidos na simulação. Especificamente para o valor de depósito de feromônio igual a 24,

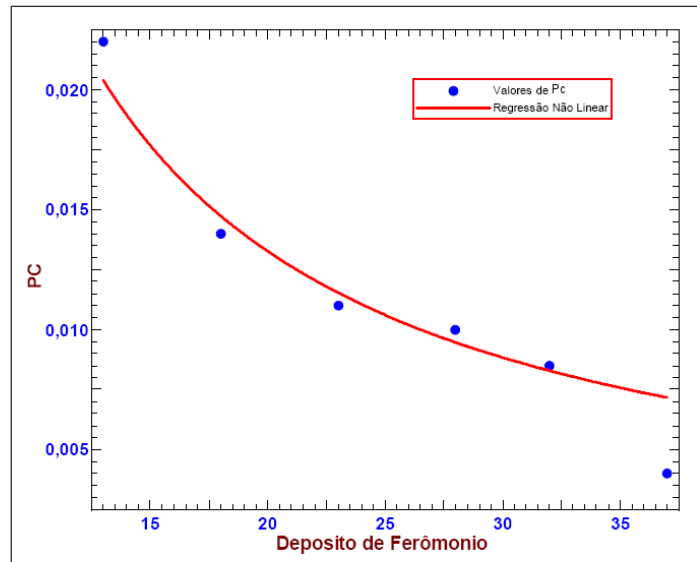


Figura 5.15: Regressão Não Linear em função da quantidade de feromônio depositada para cálculo de Pc em ambiente 100x100

onde o valor do ponto crítico é zero, isto ocorre pelo fato do valor real do ponto crítico ser inferior ao mínimo valor de densidade de agentes simulado 0,001 u.f./u.a..

De posse das equações interpoladas por área de ambiente para cálculo do ponto crítico, pode-se por sua vez, com base na similaridade das equações 5.9, 5.10 e 5.11 interpolar por meio de polinômios, para a determinação dos valores das constantes numéricas apresentadas nos denominadores das equações, uma equação geral para cálculo da amplitude do ponto crítico para o problema *Ant System* proposto. Na equação 5.12 é apresentado a forma final da solução proposta para cálculo do ponto crítico dentro dos intervalos de valores de parâmetros definidos na simulação.

$$Pc(N, DE) = \frac{a.N^2 + b.N + c}{DE}, \text{ onde :}$$

$$a = 3,351 \times 10^{-10},$$

$$b = 1,647 \times 10^{-5},$$

$$c = 0,0668 \tag{5.12}$$

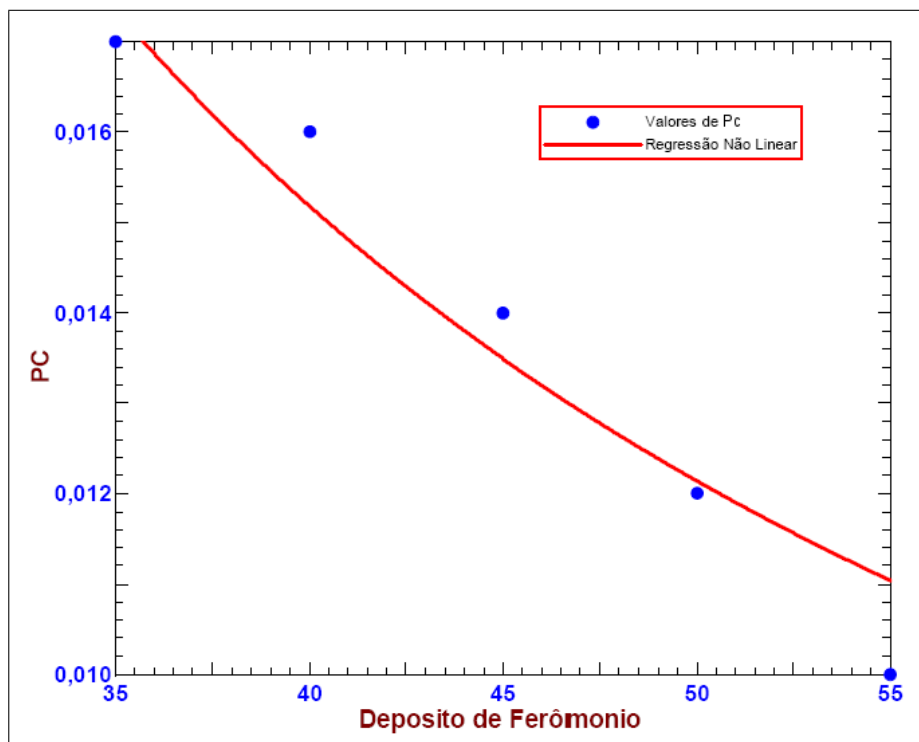


Figura 5.16: *Regressão Não Linear em função da quantidade de feromônio depositada para cálculo de Pc em ambiente 150x150*

A figura 5.17 apresenta o ajuste do polinômio interpolado aos valores das constantes das equações originais.

5.2.2 Validação dos Resultados

Como forma de validação dos resultados são apresentados os valores obtidos em simulação e os valores esperados, calculados com as equações 5.8 e 5.12, para parâmetros com valores intermediários aos valores usados para a obtenção destas equações. Os parâmetros propostos são apresentados na tabela 5.6.

Com base nas equações 5.8 e 5.12 obtemos os valores esperados da simulação que são apresentados na tabela 5.7 comparados com os valores obtidos pela simulação.

A figura 5.18 apresenta o gráfico de percolação para a simulação proposta para validação bem como o ponto crítico e o limite superior da zona crítica calcu-

| Parâmetro | Valor |
|------------------------------------|-------------------|
| Dimensões do Ambiente | 120x120 u.a. |
| Máximo de Iterações por Ciclo | 10000 u.t. |
| Alimento Disponível | 10000 u.c. |
| Densidade Inicial | 0,001 u.f. / u.a. |
| Densidade Final | 0,1 u.f. / u.a. |
| Depósito de Feromônio | 44u.g. |
| Taxa de Evaporação | 1 u.g. / u.t. |
| Quantidade de Ciclos por Densidade | 50 |
| Passo de incremento de Densidade | 0,001 u.f. / u.a. |

Tabela 5.6: Parâmetros da simulação usada para validação

| Parâmetro | Valor Calculado | Valor Simulado |
|---------------------------------|-----------------|----------------|
| Amplitude da Zona Crítica | 0,159085295 | 0,124 |
| Limite inferior da Zona Crítica | -0,071051618 | – |
| Ponto Crítico | 0,00849103 | 0,007 |
| Limite superior da Zona Crítica | 0,088033677 | 0,069 |
| Número de Agentes Ótimo | ≈ 1268 u.a. | ≈ 994 u.a. |

Tabela 5.7: Parâmetros calculados para validação e valores obtidos em simulação

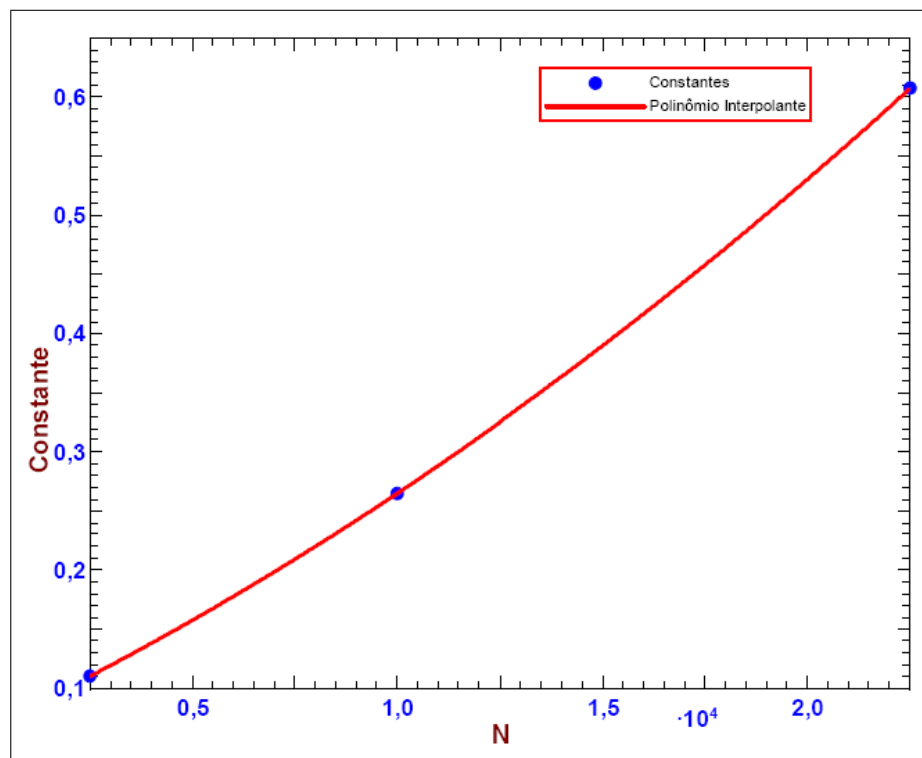


Figura 5.17: Polinômio interpolado x Valores das constantes - Cálculo de P_c

lado.

Na tabela 5.7 pode-se comparar os valores calculados pelas equações com os valores obtidos em simulação. Observa-se na tabela que mesmo os valores calculados apresentando erro, este erro não compromete a capacidade de solução do algoritmo de *swarm intelligence*, uma vez que implica em um número maior de agentes, degradando o desempenho do sistema, mas mantendo a capacidade de solução de problemas do algoritmo, apesar de os erros apresentados terem valor elevado. No entanto valores maiores de agentes e taxa de evaporação tem influência direta na qualidade da solução e devem ser levados em consideração durante a aplicação da metodologia. Os fatores que, acredita-se, influenciem nos erros encontrados são: o número e a relevância das simulações utilizadas, a distância do valor interpolado para o valor simulado mais próximo e o ajuste dos métodos numéricos utilizados. Um conjunto mais expressivo de simulações, assim como o uso de métodos numéricos mais eficientes, devem obter equações com valores mais precisos.

Os valores apresentados provavelmente não são aceitáveis para aplicações que desejam obter o melhor desempenho de algoritmos de *swarm intelligence*. Desta forma para a aplicação devem ser tomadas medidas, conforme abordado, que minimizem os erros de modelagem, estatísticos e numéricos.

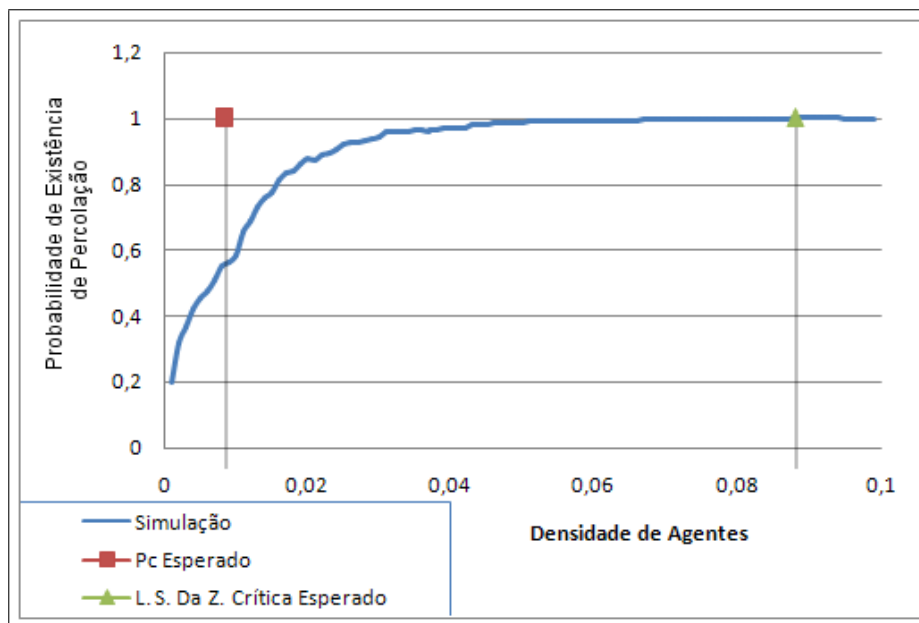


Figura 5.18: Comparação entre os valores simulados e os valores interpolados

Para o cálculo do número de agentes é utilizada a equação 5.13, que relaciona o limite superior da zona crítica (L_c) e o número de vértices do ambiente (N) com o número mínimo de agentes (A_g).

$$A_g(L_c, N) = L_c \cdot N \quad (5.13)$$

5.3 Conclusões sobre os resultados

Os resultados apresentados neste capítulo buscam validar a metodologia proposta por este trabalho. Como esperado, os valores obtidos com a utilização das equações interpoladas contêm erros com relação aos valores reais, isso se deve ao fato

do uso de métodos numéricos para a construção e modelagem do sistema. Estes erros devem ser levados em consideração quando da aplicação da metodologia em problemas de domínio real. No entanto, se julgados aceitáveis para a aplicação, não devem comprometer o funcionamento do sistema alvo.

Os dados e valores apresentados neste capítulo tem validade somente para o sistema proposto e os resultados obtidos aplicam-se ao sistema desenvolvido durante este trabalho. Outros sistemas devem, por sua vez, serem submetidos a aplicação completa da metodologia deste trabalho, afim de obterem dados relevantes. A determinação de uma equação genérica que modele os algoritmos de *swarm intelligence* ainda está por ser pesquisada e é proposta aqui como um trabalho futuro.

O comportamento do sistema foi estável e os dados obtidos nas simulações corroboram as premissas da metodologia. Também é possível perceber que a qualidade das equações, e conseqüentemente a dimensão do erro envolvido no parâmetro desejado, está diretamente ligada a complexidade e quantidade de simulações executadas e ao ajuste dos métodos numéricos de interpolação e regressão escolhidos.

Capítulo 6

Conclusão

Os métodos de *swarm intelligence* encontram cada vez mais aplicações, com as restrições impostas de processamento e a proliferação de plataformas diversas, e muitas vezes de recursos limitados. A otimização destes métodos torna-se indispensável tanto do ponto de vista computacional como econômico, como é o exemplo de redes e algoritmos de roteamento baseado em agentes para plataformas de telefonia móvel[KWA 03]. Neste trabalho foi proposta uma metodologia que tem por objetivo principal obter o conjunto de parâmetros próximo aos valores ótimos necessários para solução de determinado problema por um algoritmo de *swarm intelligence*.

Neste trabalho conclui-se que:

- Os algoritmos de *swarm intelligence* apresentam características percolativas, ou que tornam possível sua análise com base nesta teoria;
- A metodologia proposta atende aos requisitos enunciados. No entanto, neste trabalho, sua aplicação está vinculada ao problema em particular que pretende otimizar;
- A quantidade e qualidade das simulações utilizadas na elaboração das equações, alvo da metodologia, têm influência no erro resultante da modelagem;
- Os métodos numéricos de interpolação utilizados tem influência no erro resultante da modelagem;

- Parâmetros subdimensionados não garantem a solução do problema, por encontrarem-se dentro da zona crítica, em todas as vezes que o algoritmo for executado. Parâmetros superdimensionados acarretam o uso desnecessário de recursos. Sendo assim o uso de parâmetros superdimensionados ou subdimensionados acarreta degradação do sistema, do ponto de vista da otimização de recursos, ou seja, o conjunto de parâmetros mínimos também é o conjunto de parâmetros ótimos mas a influência dos parâmetros sobre a qualidade da solução é relevante;
- Os dados obtidos para a aplicação obedeciam o padrão de dados esperados do sistema, reforçando a validade da metodologia;
- Uma vez modelado o sistema o algoritmo implementado, alvo da otimização, pode ter seus parâmetros ajustados de forma dinâmica. Por meio do uso da equação estes parâmetros podem ser calculados sempre que se detecte uma mudança nas características do ambiente, mesmo em tempo de execução, dentro de limites operacionais;

A metodologia proposta neste trabalho foi implementada e suas características foram obtidas e avaliadas. Sua aplicação, no entanto, fica restrita a sistemas com características conhecidas em tempo de simulação. Os erros apresentados pela modelagem, com base em seu comportamento, têm influência no desempenho final da aplicação a ser otimizada. No entanto, estes erros por atuarem, invariavelmente no sentido de superdimensionamentos dos parâmetros envolvidos, não comprometem a capacidade de solução de problemas dos algoritmos estudados, mas provavelmente apresentam relação com a qualidade desta solução. Estas limitações tornam a aplicação do sistema restrita, mas não apresentam dados que invalidem a análise exposta. Ao contrário, indicam a importância da otimização destes algoritmos e o conseqüente ganho de desempenho decorrente desta otimização.

As conclusões obtidas neste trabalho, podem ser consideradas completas com relação à proposta apresentada. No entanto esta pesquisa não exaure os assuntos relacionados a otimização de algoritmos de *swarm intelligence* nem tão pouco a aplicação

da metodologia proposta em condições reais. Para tal são citadas aqui propostas de trabalhos futuros:

- Análise de algoritmos clássicos de *swarm intelligence* com o uso da metodologia proposta visando a obtenção de modelos matemáticos capazes de proporcionar ganho de desempenho para aplicações que utilizem estas tecnologias;
- Utilização da metodologia proposta em uma aplicação prática;
- Análise da influência das ferramentas matemáticas e de simulação no desempenho da metodologia;
- Estudo e pesquisa de equações genéricas que modelem sistemas de *swarm intelligence*.

Capítulo 7

Apêndice

O CD que acompanha este trabalho contém os códigos fonte do simulador implementado, bem como do aplicativo de análise matemática desenvolvido, com a finalidade de tratar os dados obtidos nas simulações.

Contém também os dados brutos utilizados nas simulações apresentados neste trabalho, bem como suas análises e gráficos. Todos os dados estão organizados em pastas que indicam a configuração utilizada, cada pasta contém os dados brutos separados em arquivos para cada ciclo de simulação. A totalização das médias e variâncias estão organizadas em arquivos separados indicado pelo nome do arquivo e acompanhados de uma planilha com as análises e gráficos.

Na pasta de nome Modelagem estão os arquivos e relatórios do software Rt-Plot utilizado para a execução de todas as interpolações e regressões apresentadas neste trabalho.

Referências Bibliográficas

- [ARK 98] ARKIN, R. C. **Behavior-Based Robotics**. Cambridge, MA: MIT Press, 1998.
- [BAL 00] BALCH, T.; HYBINETTE, M. Social potentials for scalable multi-robot formations. **IEEE International Conference on Robotics and Automation**, [S.l.], v.1, 2000.
- [BON 99] BONABEAU, E.; DORIGO, M.; THERAULAZ, G. **Swarm Intelligence: From Natural to Artificial Systems**. Cambridge, MA: MIT Press, 1999.
- [CAR 98] CARO, G. D.; DORIGO, M. Antnet: Distributed stigmergetic control for communications networks. **Journal of Artificial Intelligence Research**, [S.l.], v.9, n.1, 1998.
- [CAR 04] CARO, G. D. **Ant Colony Optimization and its application to adaptative routing in telecommunications network**. Faculdade des Sciences Appliquées, 2004. Tese de Doutorado.
- [DOR 96] DORIGO, M.; MANIEZZO, V.; COLORNI, A. Ant system: Optimization by a colony of cooperating agents. **IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics**, [S.l.], v.26, n.1, 1996.
- [DOR 97] DORIGO, M.; L. M. GAMBARDELLA, L. Ant colony system: A cooperative learning approach to the traveling salesman problem. **IEEE Transactions on Evolutionary Computation**, [S.l.], v.1, n.1, 1997.
- [EFR 87] EFROS, A. **Fisica y Geometria del Desorden**. Moscow: Ed. Mir, 1987. translated. Beloúsov, S.
- [GHO 06] GHOSEIRI, K.; MORSHEDSOLOUK, F. Acs-ts: Train scheduling using ant colony system. **Journal of Applied Mathematics and Decision Sciences**, [S.l.], v.1, 2006.
- [GNU 08] GNUPLOT. **GNU Plot**. Disponível em <<http://www.gnuplot.info/>>. Acesso em: Julho. Última Visita em 20.07.2008.
- [JUN 06] JUNIOR, B. G. **Análise de Algoritmos de Roteamento Baseados em Formigas**. Escola Politécnica - USP, 2006. Dissertação de Mestrado.

- [KWA 03] KWANG, M. S.; WENG, H. S. Ant colony optimization for routing and load-balancing: Survey and new direction. **IEEE Transaction on systems, man, and cybernetics**, [S.l.], v.33, n.5, September, 2003.
- [LEU 03a] LEUNG, H.; HOTHARI, R. Phase transition in a swarm algorithm for self-organized construction. **Physical Review**, [S.l.], v.68, n.Issue 4, 2003.
- [LEU 03b] LEUNG, H.; KOTHARI, R. **Self-Organized Construction of Spatial Structures by Swarms of Autonomous Mobile Agents**. College of Engineering. University of Cincinnati., 2003. Dissertação de Mestrado.
- [MAC 84] MACKAY, G.; JAN, N. Forest fires as critical phenomena. **Journal of Physics A: Mathematical and General**, [S.l.], v.17, n.14, 1984.
- [MEY 06] MEYER, K. D.; NASUTO, S. J.; BISHOP, M. Stochastic diffusion search: Partial function evaluation in swarm intelligence dynamic optimisation. **Studies in Computational Intelligence**, [S.l.], v.31, n.1, 2006.
- [NAS 05] NASA. **Shape-Shifting Robot Nanotech Swarms on Mars**. Disponível em <<http://www.nasa.gov/vision/universe/roboticexplorers/ants.html>>. Acesso em: Março. Última Visita em 01.07.2008 12:28.
- [OMR 05] OMRAN, M.; ENGELBRECHT, A. P.; SALMAN, A. Particle swarm optimization method for image clustering. **International Journal of Pattern Recognition and Artificial Intelligence**, [S.l.], v.19, n.3, 2005.
- [PAR 08] PARK, K.; KIM, Y. Analysis of antnet routing scheme by using queueing model. **Computer Communications**, [S.l.], v.In Press, 2008.
- [PRI 08] PRISMA, A. L. D. F. **Modelos de percolação e fogos florestais**. Disponível em <<http://cftc.cii.fc.ul.pt/PRISMA/capitulos/capitulo5/modulo6/topico1.php>>. Acesso em: junho de 2008.
- [RAI 08] RAINSFORD, T.; BENDER, A. Markov approach to percolation theory based propagation in random media. **IEEE Transactions on Antennas and Propagation**, [S.l.], v.56, n.Issue 5, 2008.
- [ROU 04] ROUFF, C. et al. Verification of emergent behaviors in swarm-based systems. **Proceedings - 11th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems, ECBS**, [S.l.], v., 2004.
- [SAN 07] SANTIS, E. D.; GRANDONI, F.; PANCONESI, A. Fast low degree connectivity of ad-hoc networks via percolation. **Lecture Notes in Computer Science**, [S.l.], v.4698, n.LNCS, 2007.

- [SCH 96] SCHOONDERWOERD, R. et al. Ant-based load balancing in telecommunications networks. **HP Labs 1996 Technical Reports**, [S.l.], v.1, 1996.
- [STA 82] STAUFFER, D.; CONIGLIO, A.; MIREILLE, A. Gelation and critical phenomena. **Advances in Polymer Science**, [S.l.], v.44, 1982.
- [STA 92] STAUFFER, D.; AHARONY, A. **Percolation Theory**. London: Taylor and Francis, 1992.

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)