



Modalidade do trabalho: Relatório técnico-científico
Evento: 2011 SIC - XIX Seminário de Iniciação Científica

DESENVOLVIMENTO DE UMA META-HEURÍSTICA PARA OTIMIZAÇÃO DA COBERTURA DE SINAL EM REDES SEM FIO¹

Marcelo Italo Vanelli², Sandro Sawicki³, Gerson Battisti⁴.

¹ Projeto de Pesquisa realizado no curso de Ciência da Computação. Trabalho inserido no âmbito do Projeto de Pesquisa Institucional intitulado “Heurísticas aplicadas na Distribuição de Pontos de Acesso em Redes Sem Fio”;

² Bolsista PIBIC/CNPq. Estudante do Curso de Ciência da Computação do Departamento de Ciências Exatas e Engenharias; E-mail: marcelo.vanelli@unijui.edu.br;

³ Professor do Departamento de Ciências Exatas e Engenharias. E-mail: sawicki@unijui.edu.br

⁴ Professor do Departamento de Ciências Exatas e Engenharias

Resumo

Este trabalho visa a distribuição de pontos de acesso em redes sem fio em ambientes complexos por meio da metaheurística Simulated Annealing [1]. Tal metaheurística baseia-se em uma técnica de busca local probabilística fundamentada na termodinâmica. Destaca-se pelas funções de perturbação, desfazer a perturbação, retornar o tamanho do problema e o seu custo. Essas funções foram implementadas, assim como outras auxiliares e estão citadas e explicadas ao longo do artigo. Na função que retorna o custo da perturbação foram incluídos os modelo matemático de sobreposição de círculos, necessários para calcular as intersecções de sinal. Testes usando o somatório de todas as sobreposições foram realizados para validar tal implementação e modelo matemático.

Palavras-chave: redes sem fio; heurística; otimização combinatória; modelo matemático;

Introdução

Apesar do amplo desenvolvimento alcançado pelas redes sem fio, ainda ficam questões que necessitam de avanço para oferecer garantias de qualidade para o correto funcionamento desses sistemas [10]. Uma questão essencial para garantir o desempenho e qualidade da rede sem fio, é a análise da localização dos pontos de acesso ou *access points* (APs).

Gómara et. al. [2] apresentam um método de análise de desempenho aplicado em um centro de eventos. Esse modelo reproduz no ambiente o comportamento dos usuários em um evento. O modelo proposto melhora os recursos de rede de forma flexível, sempre que na rede novos recursos forem exigidos.

Bisatto et. al [3] propõe a implantação de algoritmos para obtenção da localização de estações sem fio, baseada na técnica de trilateração. Os autores concluem que a técnica de trilateração obtém um aumento de erro de aproximadamente 1 metro no resultado final.





Modalidade do trabalho: Relatório técnico-científico
Evento: 2011 SIC - XIX Seminário de Iniciação Científica

Entretanto, essa abordagem possui maior facilidade de implementação em comparação com a análise realizada com o método *fingerprinting*.

Morais et al. [4] apresenta um modelo de localização de pontos de acesso baseada em sniffers. O objetivo desse trabalho é melhorar a precisão de um sistema de calibragem, estimando-se o decaimento de sinal provocado por obstáculos entre o ponto de acesso e o sniffer sem-fio. O modelo analítico de propagação de sinal utilizado para construção do modelo de localização é capaz de refletir de forma mais realista o ambiente monitorado. Esta modificação melhora a precisão em até 13% em erro médio, quando não se utiliza qualquer informação sobre o ambiente monitorado.

Lopes et. al. [6] descreve um modelo de propagação para interiores, baseado em uma análise teórica. O modelo de propagação proposto é aferido com medidas a 2,4GHz. Com base nessas medidas, descreve uma tabela de atenuação para os obstáculos no percurso do rádio em interiores.

Hoffmann e Gómez [8] apresentam a modelagem inicial de um protótipo de um Sistema de Informação Geográfica (SIG), o qual auxilia na tomada de decisão de posicionamento de torres de radio transmissão. O problema é abordado de duas maneiras, a primeira define os locais a serem instaladas as antenas visando o máximo de pontos possíveis, e a segunda visa o mínimo de antenas possíveis, dado um número total de pontos a serem atendidos.

Dessa forma, o projeto busca suprir a falta de uma ferramenta que resolva de forma eficiente o problema de cobertura dos *pontos de acesso* em ambientes que utilizam redes sem fio e que ao mesmo tempo tenha uma interface amigável ao usuário. Há também a necessidade de adequação do sistema frente a interferências magnéticas, espessura de paredes e obstáculos variados. Além disso, a área de desperdício de sinal deve ser a menor possível, assim como a possibilidade de deixar áreas sem cobertura. Neste sentido, este trabalho tem o objetivo de fixar múltiplos pontos de acesso em ambientes complexos para gerar a melhor cobertura de sinal por meio da metaheurística Simulated Annealing.

Metodologia

Inicialmente foram feitas leituras exploratórias com o objetivo de conhecer os diferentes algoritmos que poderiam ser utilizados na resolução do problema proposto pelo projeto. Dentre eles, destacaram-se os algoritmos genéticos, busca TABU, *quadratic placement*, GRASP (*Greedy Randomized Adaptive Search Procedure*) e *simulated annealing* [1, 14, 15, 16, 17]. Em seguida houve a comparação dos mesmos. Após escolhido o algoritmo, começou então um estudo mais aprofundado da linguagem de programação que tem um melhor desempenho, levando em consideração o projeto em questão.

Entre os vários métodos que poderiam ser usados para calcular a melhor disposição dos *access points* em uma área, o escolhido foi a metaheurística chamada *simulated annealing*. O principal motivo de escolher esse método foi porque possui uma habilidade de fornecer resultados muito próximos do ideal, o que é muito importante para aumentar o grau de precisão do programa em desenvolvimento. Outro motivo é o fato de ser uma



Modalidade do trabalho: Relatório técnico-científico
Evento: 2011 SIC - XIX Seminário de Iniciação Científica

metaheurística genérica, ou seja, pode ser usada em muitos tipos de aplicações, como por exemplo, biologia, na análise da estrutura de cadeias complexas de proteínas; militar, em problemas de defesa; física, problemas de otimização combinatória, como o problema do caixeiro-viajante.

Simulated Annealing foi baseada em um trabalho criado por Kirkpatrick [1] na década de 80. É uma técnica que aplica o algoritmo de Metrópolis para resolução de problemas de otimização. Resumidamente, *annealing* (recozimento) é um processo que se trata do aquecimento controlado de um sólido até tornar-se líquido, e em seguida ocorre um resfriamento também controlado até cristalização, transformando-se em uma estrutura molecular mais estável, devido a energia liberada ao longo do processo. Esse processo pode servir, por exemplo, para encontrar a solução com o menor custo, entre as várias soluções de um problema. Dessa forma, os estados físicos seriam equivalentes as soluções possíveis e a energia livre seria equivalente ao custo.

Assim como o método, o conhecimento sobre a linguagem de programação é de fundamental importância. Foi usado o compilador DevC++ e a linguagem de programação C.

Resultados e Discussão

A metaheurística começa com uma solução inicial, a partir desta solução inicia-se o processo de perturbação, nesse caso é a troca de lugar dos pontos de acesso. Com essa troca, calcula-se no método `GetCost()` o seu custo (o custo é representado pelo somatório de todas as sobreposições). Caso a troca for ruim e a temperatura for alta, existe uma grande probabilidade desta troca ser aceita; mas se temperatura for baixa, a tendência é que esta troca se desfça através do método `UndoPertrubation()`. Abaixo a figura 1 ilustra um fluxograma dos passos do algoritmo.

Modalidade do trabalho: Relatório técnico-científico
Evento: 2011 SIC - XIX Seminário de Iniciação Científica

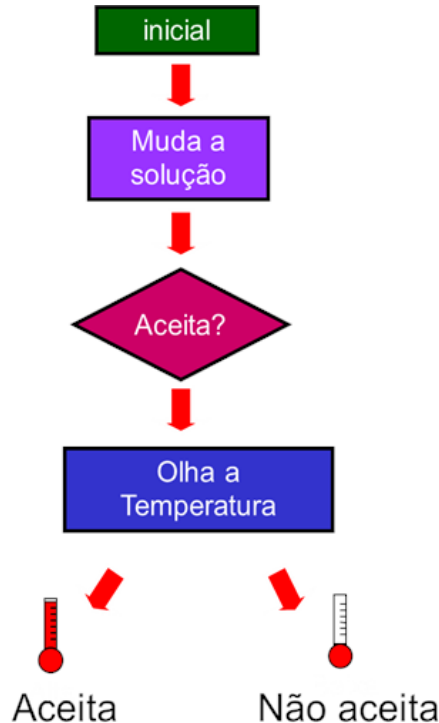


Figura 1: Fluxograma do Simulated Annealing

Inicialmente foram definidas três estruturas que armazenam os dados dos *AccessPoints*: `dadosString`, `dadosFloat` e `dadosAnt`. Essas estruturas armazenam o número identificador do AP, as coordenadas do mesmo e o raio da cobertura do sinal. A estrutura `dadosAnt` diferentemente das duas anteriores, armazena apenas a posição anterior dos APs, caso esses dados precisem ser resgatados.

Os dados que serão fornecidos para o programa estarão armazenados em um arquivo de texto com o nome de `Entrada.txt`. Os primeiros valores a serem armazenados são os referentes a área do local, ou planta. Os próximos dados a serem lidos são o número do AP, a posição e o raio de cobertura do sinal do AP. Enquanto não chegar ao fim do arquivo os dados são coletados linha por linha, sendo que cada linha é separada em quatro dados diferentes pelo caractere ‘ ‘ (espaço) e os dados armazenados nas respectivas variáveis de uma lista do tipo `dadosString`, aqui declarada como `S`.

Daqui em diante serão descritas as variáveis e funções contidas na classe `Vetor`, começando com a função de mesmo nome da classe, executada no momento em que a classe é chamada pela função principal. A função `Vetor` é que fará a conversão dos dados da estrutura `dadosString` declarada com o nome de `S`, que são do tipo `char` para o tipo `float`, armazenados na lista de estrutura `dadosFloat` declarada como `F`. No final da função `Vetor` há uma chamada para a função `GetCost()`. Essa função irá calcular o valor do custo a cada perturbação, avaliando se a sobreposição do sinal dos APs é maior que na etapa (perturbação) passada. Se a



Modalidade do trabalho: Relatório técnico-científico
Evento: 2011 SIC - XIX Seminário de Iniciação Científica

área de sobreposição anterior for maior que a área de sobreposição atual o custo incrementará em um. Em seguida ela retorna o valor do custo para `thresholdaccept.h`. Para ser possível a realização de todos os cálculos necessários para chegar a área de sobreposição de sinal, utiliza-se a biblioteca `cmath`. Nessa biblioteca estão disponibilizadas várias funções, como: seno, cosseno, raiz quadrada, potência e módulo de algum valor.

O *simulated annealing* realiza várias perturbações, por exemplo, entre as moléculas de algum sólido. O mesmo ocorre no programa, ao perturbar os APs, fazendo com que eles troquem, aleatoriamente, de posição. A função responsável por essa perturbação é chamada `Perturbation()`, que armazena todos seus dados em uma lista de dados auxiliar (`dadosAnt`), caso seja necessário voltar ao estado anterior, e logo após gera coordenadas aleatórias, por meio da função `rand`, para todos seus *access points*. Após atribuídas essas novas coordenadas, a função `GetCost` é chamada e seu valor atribuído a variável `novoCusto`, que servirá para calcular o delta.

A função `UndoPerturbation` faz com que seja possível o *simulated annealing* retornar ao seu estado anterior. Ela retorna o que estava na lista auxiliar para a lista `dadosFloat F` e o que estava em `custoAnterior` para a variável `custoAtual`. A função `Print`, que tem como principal objetivo a gravação dos dados em um arquivo de texto, chamado `Saida.txt`.

A função de avaliação computa o valor da temperatura inicial e sua variação ao longo do algoritmo. Determina também o comportamento da aceitação de *simulated annealing*. Por exemplo, na abordagem clássica desse algoritmo, começando-se com temperaturas altas, a variação da temperatura é de fundamental importância, pois ela determinará por quanto tempo ele será aleatório e por quanto será guloso. Decidirá, também, o impacto dessa mudança. Abaixo, o um pseudocódigo descrevendo os passos dessa função.

Função de Avaliação

Para cada iteração de n repetições

Ao final de cada iteração

Calcular a diferença de custo com a iteração anterior

$\Delta \text{Custo} = \text{Custo} (\text{Nova Solução}) - \text{Custo} (\text{Solução});$

Se $\text{abs}(\Delta \text{Custo}) < 0,7\%$, considera-se que não houve alteração

Se houver x iterações sem alteração, fim.



Modalidade do trabalho: Relatório técnico-científico
Evento: 2011 SIC - XIX Seminário de Iniciação Científica

A função de custo, descrita acima, considera o somatório de interseções entre todos os círculos (APs) [13]. Assim, o delta é representado pela diferença entre a nova solução e a solução atual, passando novamente para a aceitação ou não da alteração provida pela perturbação (nova localização dos pontos de acesso).

Para validar a implementação e o modelo matemático, a função de custo armazenou a cada perturbação o somatório de todas as sobreposições. Por exemplo, quando a área continha 3 pontos de acesso, a comparação se dava entre (AP1,AP2), (AP1,AP3) e (AP2,AP3), sendo AP o ponto de acesso e {1, 2, 3} o seu respectivo número. A mesma combinação acontece quando mais APs são inseridos.

A posição final dos pontos de acesso dentro da área é decidida pela metaheurística. Neste caso, o modelo matemático foi validado, entretanto, quando o somatório dos raios dos APs é menor do que a área de cobertura, os APs tendem a se afastar, deixando grandes espaços em branco (sem cobertura). Neste caso é necessária a descrição de mais restrições, principalmente, possibilitando a inserção automática de pontos de acesso ou até mesmo mudar a função de custo por meio da subtração da área total com a área de cobertura. Neste caso, quanto maior a cobertura, melhor o resultado da função custo.

Conclusões

Obteve-se sucesso ao fazer o programa se comunicar com o usuário através de leitura e gravação em arquivo e, principalmente, organizar os *access points* em um ambiente, reduzindo ao mínimo a área de sobreposição. A heurística *simulated annealing* provou ser muito eficiente ao organizar de uma melhor forma os APs.

Contudo, ainda há a necessidade de realizar a comunicação entre o *simulated annealing* e a interface gráfica. Além de incluir no raio de cobertura de sinal para cada AP a resistência das paredes e dos demais obstáculos. Neste caso, o modelo de custo a ser inserido deverá prever a perda de sinal sempre que encontrar um obstáculo.

Os trabalhos futuros consistem em: estudos sobre resistência dos materiais para descobrir qual o impacto da ondas frente a obstáculos; inserção de ambientes complexos; Redundância de cobertura; Visualização da cobertura em interface gráfica; Sombras, locais dentro do ambiente, mas sem cobertura.

Agradecimentos

Em especial ao CNPq e a UNIJUI.

Referências

- [1] KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by Simulated Annealing. Science, Number 4598, 13 May 1983, [S.l.], v.220, 4598, p.671–680, 1983.
- [2] GÒMARA, S.; JUIZ, C.; Estudio en rendimiento de redes en un palacio de congresos mediante simulacion; In: XXXV, CLEI 2009.



Modalidade do trabalho: Relatório técnico-científico
Evento: 2011 SIC - XIX Seminário de Iniciação Científica

- [3] BISATTO, Ana P.; PERES, A. Localização de Estação Sem Fio Utilizando Trilateração; X Salão de Iniciação Científica e Trabalhos Acadêmicos, 2009.
- [4] MORAES, Luis F.; NUNES, Bruno; FERNANDES, Rafael. Utilizando Características do Ambiente Monitorado Para Aumentar a Precisão na Localização de Dispositivos Wi-Fi Evitando a Calibragem. In: SRBC, 2008.
- [6] LOPES, R.; FREIXO, P.; SERRADOR, A., Modelo de Propagação para WLANs. In: JETC'05, Nov. 2005.
- [8] HOFFMANN, L. T. ; GÓMEZ A. T.; Desenvolvimento de um Protótipo de um Sistema de Informação Geográfica para Auxílio à Tomada de Decisão de Posicionamento de Torres de Radiotransmissão. TCC, Unissinos, 2002.
- [10] CARRIÓN, D.; Implementação de um Ponto de Acesso Seguro para Redes 802.11b Baseado No Sistema Operacional Openbsd.
- [13] DORNELLES, M. ; BATTISTI, Gerson ; SAWICKI, S. . Uma Proposta para Detecção de Sobreposição de Sinais em Redes Sem Fio. In: Simpósio de Tecnologia da Informação, 2010. STIN 2010.
- [14] VISWANATHAN, N.; PAN, M.; CHU, C. C.-N. **FastPlace: an analytical placer for mixed-mode designs**. In: ISPD, 2005, New York, NY, USA, p.221–223.
- [15] HMETIS HOMEPAGE, Disponível via WWW em < http://glaros.dtc.umn.edu/gkh_ome_hmetis/hmetis/overview>. Maio de 2009.
- [16] HU, B.; ZENG, Y.; MAREK-SADOWSKA, M. **mFAR: fixed-points-addition-based vlsi placement algorithm**. In: ISPD, 2005, New York, NY, USA. p.239–241.
- [17] KAHNG, A. B.; REDA, S.; WANG, Q. **APlace: a general analytic placement framework**. In: ISPD, New York, NY, USA, 2005. p.233– 235.