

# ALGORITMOS GENÉTICOS EM PROJETOS DE ENGENHARIA: APLICAÇÕES E PERSPECTIVAS FUTURAS

Heitor Silvério Lopes

Centro Federal de Educação Tecnológica do Paraná  
Departamento de Eletrônica e CPGEI  
Av. 7 de setembro, 3165 – CEP 80230-901 – Curitiba, PR  
e-mail: [hslopes@cpgei.cefetpr.br](mailto:hslopes@cpgei.cefetpr.br)

**Resumo:** Algoritmos Genéticos (AGs) são técnicas de busca e otimização baseadas no modelo Darwiniano da evolução dos seres vivos. Nas últimas décadas, inúmeras aplicações de AGs têm surgido nos campos das Engenharias e da Computação, mostrando a sua larga aplicabilidade. Neste trabalho são descritas algumas aplicações de AGs em subáreas dentro da Engenharia Elétrica, particularmente enfocando a natureza dos problemas envolvidos. O papel de algoritmos genéticos como método de engenharia é discutido, bem como são examinadas algumas perspectivas futuras.

**Palavras-Chave:** Algoritmos Genéticos, Engenharia Elétrica, Projeto.

**Abstract:** Genetic Algorithms (GAs) are optimization and search techniques based on the Darwinian model of the evolution of living beings. In last decades, several applications of GAs have appeared, showing its wide range of applicability both in Engineering and Computer Science areas. In this work, some applications of GAs to particular niches in Electrical Engineering are described, with emphasis on the nature of the problem. It is discussed the role of GAs as an engineering method, as well as some future perspectives are examined.

**Keywords:** Genetic Algorithms, Electrical Engineering, Project.

## 1 INTRODUÇÃO

### 1.1 Computação Evolucionária

Algoritmos Genéticos (AGs) são métodos de busca e otimização baseados no conceito Darwiniano da evolução dos seres vivos e em fundamentos da genética. Algoritmos Genéticos é um dos paradigmas da área de Computação Evolucionária, da qual também fazem parte a Programação Genética, os Sistemas Classificadores, as Estratégias Evolutivas e a Programação Evolucionária, embora estes três últimos sejam de menor expressão. A Computação Evolucionária, por sua vez, é parte da grande área denominada de Inteligência Computacional (juntamente com Redes Neurais Artificiais e Sistemas Fuzzy), que é caracterizada por

manipulação numérica (e não simbólica) do conhecimento, adaptabilidade e tolerância a informações imprecisas.

Este artigo é organizado da seguinte maneira: no restante desta seção, é apresentada uma breve introdução sobre o mecanismo básico de funcionamento de AGs, bem como são discutidos alguns aspectos da aplicação de AGs a problemas reais. Nas seções 2, 3, 4 e 5 e respectivas subseções, são apresentados diversos problemas específicos da área de Engenharia Elétrica, os quais foram abordados (nas referências citadas) com a técnica de AGs. Esta parte do trabalho tem objetivo ilustrativo, onde a ênfase dada é à natureza do problema envolvido, e não à solução proposta propriamente dita. Na seção 6 é discutido o papel de AGs como método de engenharia e também algumas perspectivas futuras são vislumbradas em relação ao dueto AGs/Engenharia. Por fim, na última subseção, a conclusão pessoal do autor fecha o artigo.

### 1.2 Algoritmos Genéticos

Nesta seção é apresentado um breve resumo sobre o funcionamento de algoritmos genéticos. Uma investigação mais didática e detalhada pode ser encontrada em livros-texto consagrados na área tais como Goldberg (1989) e Mitchell (1996).

Otimização é a palavra-chave quando se trata AGs. Muitos problemas de engenharia e de computação são problemas de otimização ou podem ser modelados desta maneira. Em otimização, um conjunto de variáveis do problema deve ser escolhido de tal maneira a maximizar (ou minimizar) uma função de ganho (ou de custo), a qual representa o critério de otimização. Isto pode ser feito por métodos algébricos, numéricos ou heurísticos, através de uma busca no espaço multidimensional das variáveis do problema. Para muitos problemas existem métodos convencionais eficazes. Porém, tais métodos podem exibir um desempenho fraco ou até mesmo falhar quando a natureza do problema envolve não-linearidade, ruído, descontinuidade, multimodalidade ou espaços de busca proibitivamente grandes. É nestas situações que AGs demonstram a sua utilidade e robustez.

O primeiro passo para a aplicação de AGs a um problema real é a codificação das variáveis do problema. Cada variável é discretizada em um determinado intervalo e representada com um conjunto de bits, sendo que o conjunto de variáveis codificado é chamado de "cromossomo". Um ou mais cromossomos associados para formar um "indivíduo". É importante ressaltar que o AG manipula os indivíduos (variáveis codificadas) e não as variáveis propriamente ditas. A codificação, em si, é uma questão importante na aplicação de AGs para problemas reais. Uma codificação inadequada pode tornar o problema difícil ou mesmo impossível para o AG.

Na sequência, um conjunto de indivíduos é criado formando uma "população". Esta população inicial pode ser aleatória, ou constituída com base em conhecimento prévio sobre a natureza do problema. A existência de uma população de possíveis soluções caracteriza a exploração paralela do espaço de busca. Esta é uma vantagem de AGs em relação a métodos convencionais que normalmente exploram uma possível solução de cada vez.

Cada indivíduo da população é uma possível solução para o problema. Assim, é necessário alguma medida de qualidade dos indivíduos, de tal maneira a discriminar as melhores das piores soluções. Esta medida de adequabilidade (em relação à solução para o problema) é conhecida como "*fitness*". O cálculo do *fitness* é um ponto crítico para o algoritmo, já que, em última análise, é a função de *fitness* que está sendo otimizada.

Com base no *fitness* dos indivíduos, estes são selecionados de tal maneira a privilegiar as melhores soluções em detrimento das piores. Este procedimento imita o processo de seleção natural que guia a evolução das espécies. Os indivíduos são selecionados por métodos probabilísticos (ou mesmo determinísticos) de modo a poderem gerar descendentes, implementando, assim o mecanismo da sobrevivência do mais apto.

Os indivíduos selecionados são submetidos a modificações probabilísticas através de "operadores genéticos", usualmente recombinação (*crossover*) e mutação. A recombinação toma dois indivíduos e combina partes de ambos para formar dois novos descendentes. A mutação, por outro lado, atua em um indivíduo em particular, muda aleatoriamente um bit de sua composição. A recombinação atua como busca local, enquanto que a mutação realiza uma busca global do espaço de busca.

Como consequência da seleção e da modificação pelos operadores genéticos, uma nova geração de indivíduos é criada e que substitui a anterior. A nova população é submetida à avaliação e posterior seleção e modificação. Este processo é repetido iterativamente, esperando-se que a cada geração a qualidade média dos indivíduos aumente. Ao longo de um determinado número de gerações é possível que soluções muito boas para o problema sejam geradas, ou mesmo que a solução ótima seja encontrada.

### 1.3 Aplicações de AGs

Ao longo dos últimos anos, AGs têm sido amplamente popularizados, basicamente devido à sua ampla aplicabilidade em áreas tão distintas tais como: engenharias, desenho industrial, pesquisa operacional, computação, bioquímica e biologia, composição musical, e ciências sociais. Em particular, nas diversas modalidades de engenharia, AGs têm sido amplamente aplicados (Dasgupta & Michalewicz, 1997).

Porém, ainda não há uma convenção de como classificar tais áreas de aplicações, já que AGs são métodos genéricos e muitas vezes a sua aplicação é multidisciplinar. É fato que atualmente existe muita sobreposição entre áreas de Engenharia, Computação e Pesquisa Operacional. Assim, uma taxonomia precisa baseada em áreas e subáreas pode não ser possível. Tendo isto em mente, neste trabalho foram selecionados alguns nichos da subárea de Engenharia Elétrica, de maneira a agrupar convenientemente as aplicações abordadas, porém sem ser exaustivo. Certamente, uma análise mais profunda de certas aplicações permitiria o seu agrupamento em mais de uma área, como será mostrado posteriormente.

Uma outra faceta importante é o aspecto metodológico da aplicação. Tem sido relativamente frequente a hibridização de AGs com outras técnicas de busca e otimização mais convencionais, tais como *branch-and-bound*, busca tabu ou ainda *simulated annealing*. Nestes casos, é explorado aquilo que de melhor cada método é capaz de oferecer: o AG é direcionado a uma busca global, varrendo paralelamente um grande espaço de busca, enquanto que os outros métodos exploram regiões específicas do espaço de busca, promovendo uma busca local sequencial.

Na grande maioria das aplicações do mundo-real, os AGs utilizados são de complexidade e sofisticação muito além do SGA-*Simple Genetic Algorithm* proposto por Goldberg (1989). Operadores específicos, técnicas e parâmetros de controle avançados e codificações sofisticadas são lugar-comum nas implementações de AGs mais modernas, e que objetivam resolver problemas de complexidade e dimensionalidade elevados, independentemente da área específica de aplicação.

## 2 AGs EM TELECOMUNICAÇÕES

### 2.1 Alocação de freqüências para sistemas de comunicações

O problema da alocação de freqüências (*Frequency Assignment Problem* - FAP) é um problema de otimização discreta do tipo NP-completo. O FAP é equivalente ao problema clássico da computação de colorimento de grafos. Um sistema de comunicações tem uma faixa útil de freqüências que é limitada. Esta faixa normalmente é subdividida em vários canais de freqüência central. Numa determinada região a distribuição de freqüências deve ser tal que o número de canais alocados supra a demanda sem utilizar canais adjacentes, de modo a minimizar as interferências entre canais. De maneira análoga, os canais alocados para uma determinada região têm que ser suficientemente distantes (no espectro) dos canais alocados para todas as regiões adjacentes, de modo a minimizar as interferências entre regiões. O problema é dinâmico, já que a demanda varia em função do tempo e da localização. Também é sujeito a fortes restrições: o número total de canais disponíveis é limitado; há um número mínimo (em função da demanda) e máximo (em função da infra-estrutura física e disponibilidade de canais) de canais possíveis de serem alocados para uma região; há um limite máximo de interferência aceitável dentro de cada região e entre regiões adjacentes.

Além da complexidade do problema, a alocação de freqüências envolve otimização multiobjetivos, pois pode-se aplicar um método qualquer de otimização tendo em vista inúmeros e contraditórios aspectos do funcionamento do sistema. Por

exemplo, pode-se desejar maximizar o tráfego total (isto é, suprir a demanda de canais para todas as regiões simultaneamente), minimizando ao mesmo tempo o número total de canais utilizados. Também pode-se otimizar o sistema do ponto de vista da minimização das interferências dentro de cada região e/ou a minimização das interferências do sistema como um todo. São várias as abordagens do problema para redes de comunicação por rádio, como descrito em (Cuppini, 1994; Kapsalis *et al*, 1995). Estes trabalhos demonstram a eficiência de AGs para a alocação de freqüências em redes de grande complexidade, sendo que os resultados são comparáveis ou melhores do que aqueles obtidos por outras técnicas.

Uma aplicação particular de AGs nesta categoria é a alocação de canais para redes de comunicação de telefonia celular. Nesta aplicação, a dimensionalidade pode atingir 6000 células e 12000 restrições (Crisan & Mühlenbein, 1998), tornando-se um problema de difícil solução por métodos convencionais. De fato, tem sido reportado (Dorne e Hao, 1995; Hao & Dorne, 1996) que AG's obtêm melhores resultados e em menos tempo para problemas reais quando comparado com algoritmos tradicionais do tipo *Simulated Annealing* e algoritmos de colorimento de grafos. A utilização de AGs se torna cada vez mais atrativa à medida que a dimensionalidade do problema aumenta, pois paulatinamente os métodos computacionais tradicionais se tornam ineficientes.

## 2.2 Planejamento de sistemas de comunicações

A rede de acesso a serviços de comunicações usualmente é constituída de pares de fios de cobre que ligam o usuário final à uma central de comutação. Com o aumento da demanda de serviços que requerem alta velocidade de comunicação (tais como acesso intenso à Internet e vídeo sob demanda), uma alternativa à rede física convencional é a substituição por cabos de fibras ópticas. Este tipo de instalação é conhecido como PON (*Passive Optical Network*), pois além das fibras ópticas, também utiliza dispositivos ópticos como divisores passivos. Em função da atenuação no sinal causada por estes dispositivos, existe a restrição de que haja somente dois pontos de divisão do sinal entre a central de comutação e o usuário final (nó primário e nó secundário). Uma outra restrição importante é a limitação do número de ramificações de cada nó. Em geral o produto do número de ramificações dos dois nós deve ser menor do que uma constante determinada por critérios técnicos (por exemplo, 32). Assim, os pontos principais a serem otimizados no planejamento de uma rede PON são: o número de divisões em cada nó e a localização precisa de cada nó na rede física, de modo a minimizar a atenuação do sinal e a quantidade de fibras utilizadas. Além disto, o roteamento dos dutos por onde os cabos devem passar, bem como os pontos de colocação dos divisores passivos, são fortemente restringidos pelos dutos subterrâneos já existentes (ou postes, no caso de cabeamento aéreo), o seu uso (outros cabeamentos pré-existentes), o seu acesso (ruas e alçapões de acesso) e finalmente o custo da construção de dutos específicos em função da localização geográfica dos clientes. Outros fatores de engenharia tais como a confiabilidade da rede e a qualidade do serviço impõem restrições técnicas adicionais à rede. Um fato que torna ainda mais complicada a implantação de uma rede de acesso é que o planejamento deve levar em conta também o crescimento futuro da demanda dentro de um determinado horizonte de tempo. Estes fatores todos fazem com que o planejamento de uma rede desta natureza seja uma

tarefa bastante complexa e com inúmeras possibilidades, sendo também um caso de otimização multiobjetivos. O uso de AGs para este tipo de planejamento é uma alternativa válida (Brittain *et al*, 1997) e que traz grande flexibilidade, principalmente pela facilidade de incorporar as restrições no problema e pela possibilidade de fornecer diversas soluções possíveis.

Um problema de natureza muito semelhante ao anteriormente discutido, é o planejamento de redes de comunicação de dados geograficamente distribuídas. O problema consiste em se estabelecer ligações (*links*) entre determinados nós de uma rede onde se concentram outras subredes de computadores. A configuração da topologia do *backbone* de comunicações deve levar em consideração não só a demanda de tráfego presente e futura entre os nós, como também a capacidade e a confiabilidade das conexões e o método de roteamento e controle em cada nó (Pierre & Legault, 1998). Este problema de projeto é conhecido como sendo *NP-hard*. A utilização de AGs para esta tarefa se mostra mais eficiente do que métodos tradicionais à medida que o número de nós e conexões aumenta. O uso de AGs como ferramenta computacional de projeto de redes tem sido reportado com sucesso tanto para a otimização e expansão de redes pré-existentes (Kumar *et al*, 1995), quanto para a otimização da topologia e do fluxo de tráfego entre nós em novos projetos (Dengiz *et al*, 1997; Pierre & Legault, 1998).

A questão do roteamento em redes de computadores de longa distância (WAN-Wide Area Network), é um problema mais geral do que o anteriormente mencionado. Numa rede WAN podem existir inúmeros nós, usualmente com muitas conexões (*links*) cada. Os mesmos parâmetros já citados no parágrafo precedente norteiam o projeto do esquema de roteamento em cada nó. Entretanto, dada à característica dinâmica da Internet, muitas vezes um roteamento dinâmico é mais eficiente. Cox, Davis e Qui propuseram o uso de AGs para resolver o problema de roteamento dinâmico antecipado em tempo-real (Cox *et al*, 1991). Neste trabalho é levado em consideração o estado corrente da rede como um todo, a lista de solicitações em espera, bem como o tráfego corrente em cada segmento. O AG proposto busca no espaço de todas as permutações possível de tal maneira a obter a melhor configuração de roteamento da rede, suprindo cada solicitação com a menor rota possível e com a largura de banda necessária. Outra abordagem semelhante é apresentada por Sevenster e Engelbrecht mostrando a eficácia de AGs para o problema de roteamento em redes de comunicações (Sevenster & Engelbrecht, 1996).

Algo semelhante ao planejamento de redes de acesso a sistemas de comunicação, anteriormente discutido, é o problema do planejamento de redes locais de computadores utilizando a tecnologia WLAN (*Wireless Local Area Network*). O uso de WLANs tem aumentado recentemente e o seu futuro deve ser muito promissor à medida que os seus custos venham a cair. Esta tecnologia de redes locais é apropriada para ambientes onde o cabeamento físico é de difícil implementação ou mesmo impossível. Exemplos de aplicação seriam ambientes industriais onde as condições ambientais não permitem a colocação de cabos, estações de trabalho móveis ou robôs/véículos autoguiados. Dada uma determinada configuração de pontos terminais (estações de trabalho), é necessário posicionar adequadamente um certo número de estações rádio-base que provêm a cobertura de rádio adequada para a área. O custo de uma rede WLAN é diretamente proporcional ao número de estações rádio-base e, portanto, é necessário minimizar o número destas estações,

posicionando-as estrategicamente. O número e o posicionamento das estações rádio-base são influenciados por outros fatores: a distribuição dos pontos terminais na área, a intensidade de tráfego de cada terminal, e principalmente as características de propagação do sinal entre cada estação rádio-base e cada terminal (atenuação do sinal e interferência entre canais). Por ser um problema multiobjetivos relativamente difícil, diversos métodos heurísticos já foram propostos para a solução deste problema. A utilização de um AG hierárquico demonstrou excelentes resultados permitindo a obtenção de um conjunto de Pareto que satisfaça os objetivos simultâneos a serem otimizados (Man *et al*, 1999; Tang *et al*, 1997). Estes resultados mostram que AGs são uma ferramenta adequada e eficiente de projeto de engenharia, permitindo ao projetista da rede a flexibilidade de escolher entre possíveis soluções satisfatórias para uma determinada configuração.

### 2.3 Equalização de canal de comunicação digital

A interferência intersimbólica está freqüentemente presente em sistemas de comunicação digital. A distorção do sinal resultante deste tipo de interferência é devida ou à restrita faixa espectral alocada para o canal ou aos problemas na propagação do sinal. No receptor, a interferência intersimbólica deve ser, de alguma maneira, compensada através da equalização do canal, com o objetivo de reconstruir os símbolos transmitidos. O sistema de equalização do receptor é ajustado a partir de um treinamento prévio onde as características implícitas do canal de comunicação são "aprendidas" pelo equalizador. A partir deste ponto, o equalizador é capaz de se adaptar, dentro de certas limitações, às variações dinâmicas das características do canal de comunicação, podendo, assim, restaurar o sinal recebido. Entretanto, por inúmeras razões, nem sempre é possível realizar o treinamento, como no caso de transmissões em *broadcasting*. Neste tipo de situação o equalizador deve se ajustar baseando-se somente nas observações de ruído do canal, sem acesso a uma fonte de referência. Isto é conhecido como equalização cega, para a qual inúmeras técnicas algorítmicas já foram desenvolvidas. Chen *et al* (1997) propuseram a utilização de um AG para ajustar a função de custo HOC (*high-order cumulants*) na equalização cega de canais de comunicação digital, obtendo bons resultados. Nesta aplicação, a abordagem do problema com AGs mostrou ser mais eficiente e rápida do que a abordagem com *Simulated Annealing*. De maneira análoga, White e Flockton (1994) também reportaram com sucesso a utilização de AGs para equalização de canais de comunicação digital.

## 3 AGs EM ELETRÔNICA

### 3.1 Projeto de circuitos eletrônicos analógicos e digitais

No projeto de um circuito eletrônico, objetiva-se encontrar uma estrutura de um circuito e o valor de seus componentes que façam com que as especificações iniciais do circuito sejam atingidas. Estas especificações podem ser, por exemplo, em relação ao ganho/atenuação, banda passante, resposta em freqüência, deslocamento de fase, tempo de resposta, distorção harmônica, etc. Supondo-se inicialmente que a estrutura do circuito seja fixa, pode-se utilizar o AG para fazer uma busca no espaço dos valores dos componentes (em geral, valores discretos) que otimize o circuito, levando-se em consideração um ou mais objetivos (Horrocks & Spittle, 1993). Este problema certamente de trata de otimização multiobjetivos, já

que o desempenho de um circuito eletrônico pode ser mensurado de inúmeras formas diferentes, consoante às suas especificações. Esta abordagem é particularmente interessante para circuitos de grande complexidade. A utilização de AGs para a otimização dos valores/tipos dos componentes eletrônicos de um circuito pressupõe o acoplamento com um *software* de simulação de circuitos eletrônicos do tipo SPICE ou semelhante (Haikarainen *et al*, 1996), principalmente para circuitos analógicos. Para o caso da síntese de circuitos digitais, utilizando AGs (Arslan *et al*, 1993; Coello *et al*, 1997), o próprio programa faz a simulação do circuito, que pode ser descrito por equações lógicas. Casos particulares do uso de AGs para o projeto de circuitos digitais baseados em FPGAs (*Field Programmable Gate Array*) também são relatados recentemente (Miller & Thomsons, 1998; Thomson, 1997).

Com certas restrições, um AG também poderia ser utilizado para uma busca não no espaço de valores dos componentes, porém, no espaço de busca da topologia do circuito. Entretanto esta tarefa envolve uma complexidade muito maior, visto que os graus de liberdade muito maiores (até mesmo, infinitos graus de liberdade poderiam existir). A codificação usual de AGs pressupõe estruturas de tamanho fixo, não oferecendo, assim, a flexibilidade necessária para tratar expressões de tamanho variável. Para tanto, o paradigma da Programação Genética – PG (Koza, 1992) é mais adequado. De fato, Koza e colaboradores (Koza *et al*, 1997; Koza *et al*, 1999) têm realizado intensas pesquisas na utilização de PG para a síntese de circuitos eletrônicos analógicos e digitais e obtido bons resultados com esta abordagem.

A discussão anterior refere-se primordialmente a circuitos analógicos que utilizam componentes discretos. Na área de processamento digital de sinais (PDS), um sinal analógico é amostrado digitalmente e as funções de um circuito é realizada por *software*, através da manipulação matemática de um determinado número de amostras do sinal. Uma das áreas mais importantes dentro de PDS é o projeto de filtros digitais. Existe uma grande quantidade de tipos de filtros digitais, usualmente divididos em filtros de resposta finita ao impulso (FIR) e de resposta infinita (IIR), sendo que, em geral, há um método matemático predefinido para cálculo de seus coeficientes. Algumas vezes, entretanto, a obtenção dos coeficientes é complexa ou demanda algoritmos iterativos podendo, em certos casos, ter uma complexidade computacional duplamente exponencial (Undrill *et al*, 1997). Neste tipo de aplicação, cada coeficiente do filtro é uma variável contínua, portanto, o espaço de busca é, a princípio, infinito, apenas limitado pela precisão desejada. Inúmeras variações de AGs têm sido propostas para o cálculo de coeficientes de filtros, em especial para filtros recursivos e filtros *stack* (Cemes & Ait-Boudaoud, 1993; Flockton & White, 1993; Tang *et al*, 1998a; White & Flockton, 1994), demonstrando, assim a sua aplicabilidade e eficácia.

### 3.2 Projeto de circuitos integrados

Uma outra faceta desta área de aplicação é o projeto de circuitos integrados em VLSI (*very large scale integration*). A aplicação de AGs nesta área em particular tem sido bastante intensa. Algumas ferramentas de *software* tradicionais, como por exemplo, VHDL e Verilog, já estão integrando AGs para otimização de diversas tarefas no projeto, *layout* e teste de circuitos integrados VLSI (Mazumder & Rudnik, 1999).

A síntese de circuitos integrados VLSI consiste, basicamente, na transformação de uma especificação funcional do circuito em uma representação de baixo nível como, por exemplo, esquemáticos e *layouts*. Com a crescente demanda por circuitos integrados (CIs) cada vez mais rápidos e mais complexos, uma otimização eficiente do seu projeto pode economizar quantias imensas quando o componente é produzido em larga escala. Os problemas básicos neste tipo de otimização são: a minimização da área da pastilha de silício (que é função da quantidade de unidades lógicas), minimização da potência consumida (que é função da quantidade de unidades lógicas e da velocidade de funcionamento) e a maximização da velocidade de funcionamento do circuito. Martin e Knight (1993) apresentam um sistema computacional baseado em AGs para a otimização da síntese de CIs para dois problemas típicos de projeto. Os resultados obtidos atestam a eficiência da abordagem para a tarefa, bem como sugerem que a otimização por AGs possa ser utilizada também em outras fases do ciclo de projeto de CIs.

A nível de projeto físico de um CI VLSI, isto é, de layout, AGs também têm uma grande aplicabilidade. Uma vez definido o projeto eletrônico do circuito, os componentes são agrupados em unidades funcionais denominadas de macrocélulas. Estas macrocélulas são blocos retangulares de tamanho variado e com terminais nas suas bordas. Os terminais são conectados por condutores a outros terminais em outras macrocélulas. Alguns terminais devem ser conectados externamente ao CI. O *layout* define as posições das macrocélulas e as rotas dos condutores que as interligam. O problema do *layout* em VLSI é basicamente igual ao problema do roteamento de placas de circuito impresso, anteriormente discutido. Entretanto neste problema é desejável que tanto a definição da posição das macrocélulas quanto o roteamento em si sejam feitos de maneira automática, o que torna o problema ainda mais complexo computacionalmente. Uma solução para este problema foi proposto por Schnecke e Vornberger (1997), utilizando AGs paralelos. A comparação desta técnica com outras descritas na literatura, utilizando um *benchmark*, demonstrou a superioridade e escalabilidade da abordagem com AGs. Outros trabalhos recentemente publicados também enfatizam o sucesso do uso de AGs tanto para a tarefa de roteamento de VLSI (Davidenko *et al.*, 1997; Lienig, 1997), como também a otimização da colocação das macrocélulas (Wang & Chen, 1995) e para teste e simulação de falhas de circuitos (Rudnik *et al.*, 1997).

### 3.3 Projeto de fiação impressa e montagem

Bastante semelhante à discussão anterior, é o projeto do *layout* de placas de circuito impresso (PCI) multilaminadas. Existem inúmeros *softwares* disponíveis para roteamento automático de PCI, sendo que a maioria deles é de um custo bastante elevado. Num projeto de PCI, a disposição física dos componentes é, em geral, definida pelo projetista, em função de fatores não só estéticos (tipo e tamanho dos componentes, distribuição organizada sobre a PCI, etc), como também técnicos (dissipação de potência, proximidade de conectores, facilidade de acesso e manutenção dos componentes, etc). Após a definição da localização de cada componente eletrônico é necessário interligar fisicamente os terminais dos componentes entre si através de filetes de cobre (trilhas). Estas ligações seguem o esquema elétrico definido pelo projeto do circuito eletrônico. O caminho que percorre uma trilha de um terminal até outro não pode cruzar outra trilha na mesma superfície, o que causaria um curto-circuito. O roteamento automático

realizado por certos *softwares* consiste em encontrar sequencialmente trilhas não-conflitantes entre si interligando os pontos necessários, até que todos os terminais sejam convenientemente conectados. Isto é realizado utilizando-se um, dois ou múltiplos planos de interligação, o que caracteriza placas de face simples, face dupla ou multilaminadas. Os algoritmos de roteamento automático usualmente são derivados da teoria dos grafos, porém nem sempre são eficientes para PCIs com muitos componentes e ligações. A busca de uma solução ótima para o roteamento de PCIs é inviável para a maioria dos casos, já que o custo computacional desta solução é extremamente elevado. A utilização de AGs para o problema de roteamento automático de PCIs é ainda uma área de pouca pesquisa, porém já com resultados promissores (Tanomaru & Oka, 1995).

Correlacionado ao projeto da PCI está o processo de fabricação das placas de circuitos eletrônicos. A nível industrial, a montagem dos componentes eletrônicos sobre a PCI (para posterior soldagem) é feita por máquinas automáticas que seguem uma programação específica. Minimizar o tempo de colocação/soldagem significa diminuir custos de produção. Este é um problema semelhante ao clássico "caixeiro viajante", porém mais complexo, pois é restrito por questões de tempo, prioridades, roteamento, etc. AGs também foram utilizados satisfatoriamente para este problema na indústria eletrônica (Lindhorst, 1998).

## 4 AGs EM SISTEMAS ELÉTRICOS DE POTÊNCIA

A operação de um sistema de geração e distribuição de energia elétrica envolve, entre outras atividades, o planejamento, a previsão e o despacho de carga. O planejamento visa estruturar o crescimento da oferta de energia elétrica para o futuro, e é feito com base em um horizonte de até vários anos. A previsão de carga consiste na estimativa antecipada da energia elétrica a ser consumida em uma região, e é baseada principalmente em dados históricos. A grosso modo, execução propriamente dita da previsão de carga é o despacho de carga. Esta atividade consiste no gerenciamento integrado do sistema geração-distribuição. Isto é, ativar quantas unidades geradoras de energia forem necessárias para suprir a demanda decarga em diversas regiões. Deve ser levado em consideração não só a flutuação da demanda com a hora do dia, como também as capacidades de geração das usinas e de transmissão das linhas. Variáveis adicionais, tais como a compra e venda de energia de outras concessionárias, variações climáticas inesperadas e manutenções programadas tornam o despacho de carga uma tarefa de alta complexidade que demanda, além profissionais altamente especializados, *softwares* auxiliares sofisticados. A literatura recente tem mostrado uma grande quantidade de aplicações de AGs no problema do despacho de carga (Bakirtzis *et al.*, 1994; Chen & Chang, 1995; Li *et al.*, 1994; Wong & Wong, 1994). A principal característica de tais aplicações é o lucro gerado para as operadoras em função da otimização do sistema, que passa a operar de maneira mais econômica e segura. Este problema é particularmente interessante para AGs não só pela sua complexidade, mas também por ser dinâmico.

Um subproblema do despacho de carga, bastante conhecido de empresas geradoras de eletricidade (em especial dos setores de geração de energia), é a questão do comissionamento de unidades geradoras. O problema se trata da decisão de quantas e quais unidades geradoras de energia devem ser utilizadas

num determinado momento. O tempo de colocação em funcionamento de uma unidade geradora pode ser grande, quando comparado com a velocidade de variação da demanda de energia. De maneira análoga, não é econômico o funcionamento de unidades geradoras quando não há demanda. Além disto também existem inúmeros aspectos que impõem outras restrições ao problema: o número, capacidade, tipo e localização geográfica de cada unidade geradora, o tempo de acionamento e desligamento de uma unidade geradora, a disponibilidade e a capacidade das linhas de transmissão, a quantidade e localização da demanda de energia e finalmente a previsão da demanda. A maneira tradicional de abordagem deste problema é através de programação inteira-mista com métodos de relaxação, entretanto Hassoun (1994) utilizou um método global de otimização através de AGs e redes neurais para resolver o problema. Este projeto provou que o uso destas técnicas avançadas pode trazer uma considerável economia para empresas do setor elétrico. Especificamente quando as unidades de geração são termoelétricas, fatores econômicos, além dos técnicos, são preponderantes: cada usina tem custos de operação distintos e obviamente deseja-se também minimizar os custos totais de operação do sistema. Dasgupta e McGregor (1994) e ainda Michalewicz *et al* (1996) mostraram o uso de AGs para problemas desta natureza.

A energia elétrica gerada deve ser transmitida por linhas de transmissão de alta tensão até os pontos de consumo (incluindo rebaixamento de tensão). Um sistema integrado de geração e distribuição de energia elétrica normalmente tem alguma redundância na transmissão, de tal modo que a energia pode chegar a um ponto de carga por mais de um caminho. Desligar uma linha de transmissão pode significar sobrecarga de outras e, talvez, *blackouts* localizados. Entretanto, mesmo assim a manutenção preventiva deve ser regularmente executada. Logo, o planejamento da manutenção de linhas de transmissão de alta tensão deve ser muito cuidadoso e organizado. O plano de manutenção deve minimizar os custos, levando em consideração a demanda regional de energia, as capacidades dos geradores e suas disponibilidades, a capacidade de transmissão da linha e das demais linhas que não serão desligadas simultaneamente. Nesta aplicação, AGs também têm se mostrado úteis em problemas reais. Por exemplo, na National Grid Plc, empresa que administra as linhas de transmissão na Grã-Bretanha, este complexo planejamento é realizado pelos seus engenheiros com o auxílio de AGs (Langdon, 1996).

Inúmeras outras aplicações de AGs em sistemas elétricos de potência têm sido reportadas na literatura, tais como a localização ótima de seccionadores em redes de distribuição de baixa tensão (Levitin, 1994), otimização da potência reativa (Iba, 1994; Lee *et al*, 1995), otimização da temporização de relés de proteção automáticos (Alander *et al*, 1997), planejamento e expansão de sistemas de geração (Fukuyama & Chiang, 1996) e distribuição em baixa e média tensão (Barczynski *et al*, 1999) e até mesmo determinação de parâmetros de motores de indução (Haque *et al*, 1995).

## 5 AGs EM MECATRÔNICA E CONTROLE

### 5.1 Robótica

A robótica é uma área de fronteira entre as Engenharias Elétrica e Mecânica, o que é corroborado pelo recente surgimento da Engenharia Mecatrônica.

AGs têm sido mais freqüentemente aplicados a um problema clássico (e suas variações) na área de robótica: o planejamento de trajetórias tanto para manipuladores robóticos, quanto para robôs móveis. O planejamento de trajetórias é um problema do tipo ordem-dependente, posto que é fundamental seguir uma seqüência de pontos (posições/orientações no espaço de trabalho do robô) para que o efetuador possa atingir o ponto objetivo desejado. De maneira análoga, para robôs móveis, o robô deve seguir uma trajetória no plano de tal maneira a evitar obstáculos e atingir um determinado ponto. Uma trajetória pode ser gerada e/ou otimizada satisfatoriamente com AGs desde que a codificação utilizada permita a representação de tamanho variável dos indivíduos. Isto é necessário pois cada indivíduo representará uma possível trajetória, e esta pode ter um número variável de pontos (Davidor, 1991a). A otimização da trajetória pode ser feita com base em um ou mais critérios bem definidos, por exemplo, o menor deslocamento possível entre o ponto inicial e final, o menor tempo possível de deslocamento, a menor energia gasta no trajeto, etc. Isto tudo torna o problema da geração de trajetórias altamente multimodal e multidimensional. Adicionalmente pode-se requerer que a trajetória passe por determinados pontos, o que restringe a variação possível do ângulo de determinadas juntas de um manipulador. Estas restrições quando consideradas no conjunto representam um problema de elevada complexidade e o uso de AGs tem se mostrado altamente eficiente quando comparado com métodos tradicionais (Davidor, 1991a; Lee & Lee, 1997; Toogood *et al*, 1995).

Uma variante do planejamento de trajetórias é a determinação da cinemática inversa de manipuladores, isto é, dado um ponto no espaço de trabalho do robô, determinar qual é a coordenada no espaço de juntas. Este problema é bem mais complexo do que a cinemática direta (que tem uma única solução), pois pode ter nenhuma, apenas uma, ou múltiplas soluções. Neste caso em particular, AGs tem sido eficazes para a determinação de trajetórias no espaço das juntas, tanto para manipuladores não-redundantes, quanto para manipuladores redundantes (Gebara Júnior *et al*, 1999; Silveira *et al*, 1994).

O objetivo-chave da robótica móvel é fazer com que o robô navegue através de um ambiente até uma posição preestabelecida, sem coldir com os obstáculos presentes. É desejável também que o robô possa navegar através de um grande número de diferentes configurações do ambiente. Tradicionalmente este problema tem sido atacado através de técnicas de Inteligência Artificial simbólica, porém com resultados ainda aquém do ideal. AGs têm sido utilizados para a otimização dos parâmetros de controle da navegação do robô (Ram *et al*, 1994) e, em especial, no planejamento da trajetória (Davidor, 1991b).

### 5.2 Identificação de Sistemas

O princípio básico da identificação de sistemas é: a partir de um modelo conhecido de estrutura e de um conjunto de entradas e saídas, estimar os parâmetros do referido modelo. Em alguns casos o modelo do sistema é bem conhecido e através de métodos matemáticos é possível a determinação eficaz do valor dos parâmetros. Entretanto, quando a estrutura não é perfeitamente conhecida ou o sistema envolve não-linearidades, ruído ou outros fatores complicadores, as abordagens tradicionais podem não ser viáveis. Nestes casos a abordagem do problema utilizando AGs passa a ser interessante, em particular para sistemas não-lineares, contínuos ou discretos. A identificação de sistemas é uma área de intensa aplicação principalmente em engenharia de controle

e AGs podem ser utilizados para a identificação em sistemas contínuos e discretos, lineares e não-lineares, identificação de polos e zeros ou mesmo parâmetros físicos de um sistema (Kristinnson & Dumont, 1992; Tan & Li, 1997). Também AGs foram utilizados para a identificação de parâmetros de sistemas *fuzzy* (Yen & Gillespie, 1995), e para a modelagem de sistemas físicos específicos (Tzes *et al*, 1998). Iba *et al* (1993) apresentam um AG diferente para a identificação de sistemas. Nesta proposta a representação cromossômica é estruturada de tal maneira a evitar a explosão combinatória à medida que aumenta o número de parâmetros do sistema investigado. Esta abordagem, de fato, é um AG mais próximo da Programação Genética, que efetivamente consegue manipular com mais eficiência problemas desta natureza (Koza, 1992).

### 5.3 Sistemas de Controle

O projeto de sistemas de controle em geral é baseado em métodos algébricos clássicos que são eficientes para a maioria das aplicações práticas. Não obstante, AGs também podem ser aplicados em problemas de controle linear e não-linear e de sistemas dinâmicos que envolvam uma complexidade mais elevada (Fleming & Fonseca, 1993; Krishnakumar & Goldberg, 1992; Morrison & Stengel, 1997; McGregor *et al*, 1992; Michalewicz *et al*, 1992). De igual modo, a aplicação prática em controle de processos industriais, tais como mostrado em Chipperfield & Fleming (1996), Coelho & Coelho (1998) e Nordvik & Renders (1991), são usuais.

Porém, onde AGs têm tido mais amplamente utilizados é no projeto de controladores com lógica *fuzzy*, uma área de intensa pesquisa e farta bibliografia. A grosso modo, o uso de AGs em sistemas de controle *fuzzy* pode ser dividido em duas grandes categorias: otimização de funções de pertinência e otimização da base de regras. No primeiro caso, as funções de pertinência que caracterizam os conjuntos *fuzzy* podem ter uma grande variedade de formas e, em geral, são parametrizadas. A escolha correta da forma e dos parâmetros das funções de pertinência não é trivial, principalmente considerando-se um sistema com muitas variáveis lingüísticas. AGs para este tipo de aplicação em geral objetivam definir a forma (e os parâmetros) do conjunto total de funções de pertinência de um controlador *fuzzy* (Karr, 1991). O outro caso, a otimização da base de regras, tem sido ainda mais pesquisado, pois consiste de um problema bem mais desafiador. A questão crítica é a explosão combinatória que ocorre no projeto de controladores com um grande número de variáveis e de funções de pertinência. Por exemplo, com  $n$  variáveis e  $m$  funções de pertinência, podem existir  $n^m$  regras possíveis. Quanto maior o número de regras, mais tempo de processamento será consumido, podendo até mesmo inviabilizar o uso em tempo-real do controlador. O conjunto de regras deve ser, portanto, o menor possível e o mais adequado possível para a função de controle. Esta é uma tarefa muito difícil por ser multimodal e descontínua, o que normalmente demanda algum conhecimento empírico do sistema controlado. O uso de AGs para esta tarefa pode facilitar sobremaneira o projeto de controladores lógicos *fuzzy* (Baitinger & Kropp, 1993; Cho *et al*, 1997; Park *et al*, 1993; Pham & Karaboga, 1991). Não obstante, inúmeros trabalhos utilizam AGs de forma integrada no projeto automático de controladores lógicos *fuzzy*, otimizando tanto as funções de pertinência quanto a base de regras *fuzzy* (Carse *et al*, 1996; Chiang *et al*, 1997; Heider & Drabe, 1997; Homaifar & McCormick, 1995; Tang *et al*, 1998b). Esta metodologia tem demonstrado ser muito superior ao tradicional procedimento de tentativa e erro que muitas vezes é inviável em termos práticos.

## 6 DISCUSSÃO E CONCLUSÃO

### 6.1 AGs como método de engenharia

Otimização é um tópico que aparece usualmente em quase toda publicação importante principalmente nas áreas de Engenharia. Ao desenvolver um projeto, não necessariamente o engenheiro o faz de maneira otimizada. Um projeto real de engenharia envolve usualmente uma grande quantidade de parâmetros que devem ser definidos de maneira a satisfazer um conjunto de restrições de projeto ao mesmo tempo que satisfaz uma ou mais especificações. Assim, otimizar um projeto de engenharia demanda experiência e tempo, e pode ser uma tarefa bastante difícil, mesmo para engenheiros experientes. A utilização de AG's pode facilitar a tarefa de projeto, convergindo rapidamente para um conjunto quase-ótimo (ou mesmo ótimo) de parâmetros, manipulando inúmeras limitações impostas pelo usuário ou pelo projeto. Neste artigo foram apresentadas inúmeras referências de trabalhos de engenharia (enfocando a área de Engenharia Elétrica) que efetivamente utilizaram com sucesso AGs como ferramenta.

O processo criativo e inovador implementado por AGs é facilmente comprehensível: a partir de um conjunto de possíveis soluções para um problema, seleciona as melhores, aproveita o que de melhor cada uma tem, recombina-as, adapta-as e gera novas soluções. Este refinamento iterativo é o que intuitivamente se faz para resolver problemas não só de engenharia, com também de outras áreas.

AGs são facilmente adaptáveis a inúmeras classes de problemas, são robustos e são fáceis de hibridizar com outras técnicas. Além disto, exploram satisfatoriamente espaços de busca de grande dimensionalidade, assim como manipulam com certa facilidade um grande número de restrições, fatores estes comuns em projetos de engenharia. O custo computacional de implementação é baixo, sendo um algoritmo simples e também facilmente paralelizável. Estes fatores todos tornam AGs métodos muito atraentes para a área de engenharia.

### 6.2 Perspectivas Futuras

Projeto é normalmente uma atividade chave em engenharia e é considerada uma atividade que requer uma considerável criatividade aliada a profundo conhecimento técnico. Até mesmo a definição própria do termo "projeto" pode ser complicada, posto que ele pode ser interpretada de diversas maneiras, dependendo da tarefa a ser executada. Embora tenham sido investidos muitos esforços no sentido de se desenvolver programas e sistemas para o auxílio a projeto em diversas áreas da engenharia, tais programas são notoriamente difíceis de desenvolver e com eficiência limitada. Neste trabalho foram citadas inúmeras tarefas de projeto de engenharia, com por exemplo, planejamento de redes de comunicação, projeto de circuitos eletrônicos e otimização de sistemas produção e distribuição de energia elétrica. As ferramentas atualmente disponíveis de auxílio a projeto em engenharia são fortemente baseadas no conhecimento do processo de projeto. Isto as torna profundamente dependentes da área específica de aplicação, sendo muito difícil. A utilização de AGs como ferramenta de projeto nos diversos exemplos aqui descritos, atesta o surgimento de um novo elemento como importante ferramenta de auxílio a engenheiros. Embora neste trabalho seja mostrada apenas uma pequena gama de aplicações numa área específica, AGs têm sido aplicados com sucesso em inúmeras outras áreas das

Engenharias (Computação, Mecânica, Química, Civil, etc). Isto mostra que além de robustez e eficiência, AGs também são métodos genéricos. Num contexto mais amplo, não só AGs como também outros paradigmas da Computação Evolucionária também estão se popularizando. No passado, a filosofia Projeto Auxiliado por Computador/Engenharia Auxiliada por Computador (do inglês CAD/CAE) modificaram radicalmente a prática profissional do engenheiro moderno, especialmente aqueles dedicados à área de projetos. É provável que brevemente venham a emergir novos conceitos: o Projeto Auxiliado por Algoritmos Genéticos (PAAG) e a Engenharia Auxiliada por Algoritmos Genéticos (EAAG). O *status-quo* da pesquisa atual em AGs aplicados indica que brevemente, estarão disponíveis sistemas de desenvolvimento de aplicações baseadas em AGs, dispensando o potencial usuário da necessidade de programação em baixo nível. Um sistema desta natureza será um tipo de *shell* (analogamente às primeiras ferramentas para desenvolvimento de sistemas baseados em regras em Inteligência Artificial). A disponibilização de tais ferramentas certamente virá a popularizar ainda mais o uso de AGs especialmente no âmbito de usuários que desenvolvem tarefas de planejamento e projeto em engenharia.

### 6.3 Conclusão

Certamente AGs vieram para ficar; não são apenas uma "tecnologia da moda"; tal afirmação é suportada pela imensa quantidade de aplicações bem sucedidas. O principal fator responsável pelo sucesso de AGs em aplicações de engenharia é relacionado à sua eficácia. Nos inúmeros exemplos citados neste trabalho e em muitos outros, a característica fundamental é que o AG de fato é capaz de ser uma ferramenta eficaz para a solução dos problemas envolvidos em projetos de engenharia. Esta eficácia traz em si um fator de ganho, em geral econômico, já que AGs em muitas aplicações vêm direta ou indiretamente gerar algum tipo de lucro, seja por obter melhor desempenho ou melhor qualidade do produto/processo. O uso de AGs em si é uma forma de inovação importante como método de engenharia. Possivelmente em breve testemunharemos a sua difusão e assimilação na prática profissional e nos currículos dos cursos de engenharia modernos.

## REFERÊNCIAS BIBLIOGRÁFICAS

- Alander, J.T., T. Mantere & G. Moghadampour (1997). Searching protection relay response time extremes using genetic algorithm - software quality by optimization.. *Proceedings of the 4<sup>th</sup> International Conference on Advances in Power System Control, Operation & Management*, Hong Kong, v. 1, p. 95-99.
- Arslan, T., E. Ozdemir, M.S. Bright, M.S. & D.H. Horrocks (1993). Genetic synthesis techniques for low-power digital signal processing circuits. *Proceedings of the 1<sup>st</sup> On-line Workshop on Soft Computing*, Nagoya, Japan, [sp].
- Baitinger, U.G. & K. Kropp (1993). Optimization of fuzzy logic controller inference rules using a genetic algorithm. *Proceedings of 1<sup>st</sup> European Congress on Fuzzy and Intelligent Technologies*, v. 2, p. 1090-1096.
- Bakirtzis, A.G., V. Petridis & S.A. Kazarlis (1994). Genetic algorithm solution to the economic dispatch problem. *IEE Proceedings part C: Generation, Transmission and Distribution*, v. 141, n. 4, p. 377-382.
- Barczynski, D., P. Helt, M. Parol & P. Piotrowski (1999). ANN and EA in electrical distribution network optimisation. In Szczepaniak, P.S., *Computational Intelligence and Applications*. Heidelberg: Physica-Verlag, p. 94-107.
- Brittain, D., J.S. Williams & C. McMahon (1997). A genetic algorithm approach to planning the telecommunications access network. *Proceedings of the 7<sup>th</sup> International Conference on Genetic Algorithms*, East Lansing, USA, p. 623-628.
- Carse, B., T.C. Fogarty & A. Munro (1996). Evolving fuzzy rule based controllers using genetic algorithms. *Fuzzy Sets and Systems*, v. 80, n. 3, p. 273-293.
- Cemes R. & D. Ait-Boudaoud (1993). Genetic approach to design of multiplierless FIR filters. *Electronic Letters*, v. 29, n. 24, p. 2087-2088.
- Chen, P-H. & H-C. Chang (1995). Large-scale economic dispatch by genetic algorithm. *IEEE Transactions on Power Systems*, v. 10, n. 4, p. 1919-1926.
- Chen, S., Y. Wu & S. McLaughlin (1997). Genetic algorithm for blind channel identification with higher order cumulant fitting. *IEEE Transactions on Evolutionary Computation*, v. 1, n. 4, p. 259-265.
- Chiang, C-K., H.Y. Chung & J-J. Lin (1997). A self-learning fuzzy logic controller using genetic algorithms with reinforcements. *IEEE Transactions on Fuzzy Systems*, v. 5, n. 3, p. 460-467.
- Chipperfield, A. & P.J. Fleming (1996). Evolutionary algorithms for control engineering. *Proceedings of 13<sup>th</sup> International Federation of Automatic Control World Congress*, San Francisco, USA, p. 181-186.
- Cho, H-J., K-B. Cho & B-H. Wang (1997). Fuzzy-PID hybrid control: automatic rule generation using genetic algorithms. *Fuzzy Sets and Systems*, v. 92, n. 3, p. 305-316.
- Coello, C.A., A.D. Christiansen & A. Hernández-Aguirre (1997). Automated design of combinational logic circuits using genetic algorithms. *Proceedings of the International Conference on Artificial Neural Nets and Genetic Algorithms*, Norwich, England, p. 335-338.
- Coelho, L.S. & A.A.R. Coelho (1998). Computational intelligence in process control: fuzzy, evolutionary, neural and hybrid approaches. *International Journal of Knowledge-Based Intelligent Engineering Systems*, v. 2, n. 2, p. 80-94.
- Cox, L.A., L. Davis & Y. Qiu (1991). Dynamic anticipatory routing in circuit-switched telecommunications networks. In Davis, L. (ed.), *Handbook of Genetic Algorithms*, New York: Van Nostrand Reinhold, p. 124-143.
- Crisan, C. & H. Mühlenbein (1998). The frequency assignment problem: a look at the performance of evolutionary search. *Artificial Evolution 3<sup>rd</sup> European Conference – Lecture Notes in Computer Science*, v. 1363, p. 263-273.

- Cuppini, M. (1994). A genetic algorithm for channel assignment problems. *European Transactions on Telecommunications and Related Technologies*, v. 5, n. 2, p. 285-294.
- Dasgupta, D. & Z. Michalewicz (1997). *Evolutionary Algorithms in Engineering Applications*. Berlin: Springer-Verlag.
- Dasgupta, D. & D.R. McGregor (1994). Thermal unit commitment using genetic algorithms. *IEE Proceedings - part C: Generation, Transmission and Distribution*, v. 141, n. 5, p. 459-465.
- Davidenko, V.N., V.M. Kureichik, & V.V. Miagkikh (1997). Genetic algorithm for restrictive channel routing problem. *Proceedings of the 7<sup>th</sup> International Conference on Genetic Algorithms*, East Lansing, USA, p. 636-642.
- Davidor, Y. (1991a). A genetic algorithm applied to robot trajectory generation. In Davis, L. (ed.), *Handbook of Genetic Algorithms*. New York: Van Nostrand Reinhold, p. 144-165.
- Davidor, Y. (1991b). *Genetic Algorithms and Robotics: a Heuristic Strategy for Robotics*. Singapore: World Scientific Publishing.
- Dengiz, B., F. Altiparmak & A. Smith (1997). Local search genetic algorithm for optimal design of reliable networks. *IEEE Transactions on Evolutionary Computation*, v. 1, n. 3, p. 179-188.
- Dorne, R. & J.K. Hao (1995). An evolutionary approach for frequency assignment in cellular radio networks. *Proceedings of the IEEE International Conference on Evolutionary Computation*, Perth, Australia, p. 539-544.
- Flockton, S.J. & M.S. White (1993). Application of genetic algorithms to infinite impulse-response adaptative filters. In Ruck, D.W. (ed.), *Science of Artificial Neural Networks II*, v. SPIE-1966, p. 414-419.
- Fleming, P.J. & C.M. Fonseca (1993). Genetic algorithms in control systems engineering. *Proceedings of 12<sup>th</sup> International Federation of Automatic Control World Congress*, Sidney, Australia, v. 2, p. 383-390.
- Fukuyama, Y. & H-D. Chiang (1996). A parallel genetic algorithm for generation expansion planning. *IEEE Transactions on Power Systems*, v. 11, n. 2, p. 955-961.
- Gebara Junior, M., A.F. Santos & H.S. Lopes (1999). Inverse kinematics of trajectories of redundant robotic manipulators using genetic algorithms. [submetido para publicação].
- Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, USA: Addison-Wesley.
- Haikarainen, S., H. Jokinen & M. Valtonen (1996). Genetic optimization in circuit simulation. *Proceedings of the Baltic Electronics Conference*, Tallin, Estonia, p. 381-384.
- Hao, J-K. & R. Dorne (1996). Study of genetic search for the frequency assignment problem. *Artificial Evolution 2<sup>nd</sup>* European Conference – Lecture Notes in Computer Science, v. 1063, p. 539-544.
- Haque, T., R. Nolan & J. Reynaud (1995). Parameter determination for induction motors. *Proceedings of the IEEE SOUTHEASTCON'94*, Miami, USA, p. 45-94.
- Hassoun, M.H. (1994). Optimization of the Unit Commitment Problem by a Coupled Gradient and by a Genetic Algorithm. *EPRI Report TR-103697*, Electric Power Research Institute, Palo Alto, USA.
- Heider, H. & T. Drabe (1997). Fuzzy system design with a cascaded genetic algorithm. *Proceedings of IEEE International Conference on Evolutionary Computation*, Indianapolis, USA, p. 585-588.
- Homaifar, A. & E. McCormick (1995). Simultaneous design of membership functions and rule sets for fuzzy controllers using genetic algorithms. *IEEE Transactions on Fuzzy Systems*, v. 3, p. 129-139.
- Horrocks, D.H. & M.C. Spittle (1993). Component value selection for active filters using genetic algorithms. In *Proceedings of the 1<sup>st</sup> On-line Workshop on Soft Computing*, Nagoya, Japan, [sp].
- Iba, H., T. Kurita, H. deGaris, H. & T. Sato (1993). System identification using structured genetic algorithms. *Proceedings of the 5<sup>th</sup> International Conference on Genetic Algorithms*, Urbana-Champaign, USA, p. 279-286.
- Iba, K. (1994). Reactive power optimization by genetic algorithm. *IEEE Transactions on Power Systems*, v. 9, n. 2, p. 685-692.
- Kapsalis, A., P. Chardaire, V.J. Raynard-Smith & G.D. Smith (1995). The radio link frequency assignment problem: a case study using genetic algorithms. *Evolutionary Computing AISB Workshop - Lecture Notes in Computer Science*, v. 993, p. 117-131.
- Karr, C.L. (1991). Design of an adaptative fuzzy logic controller using a genetic algorithm. *Proceedings of the 4<sup>th</sup> International Conference on Genetic Algorithms*, San Diego, USA, p. 450-457.
- Koza, J.R. (1992). *Genetic Programming*. Cambridge: MIT Press.
- Koza, J.R., F.H. Bennett III, H. Andre & M.A. Keane (1999). *Genetic Programming III*. San Francisco, USA: Morgan Kaufmann.
- Koza, J.R., F.H. Bennett III, D. Andre, M.A. Keane & F. Dunlap (1997). Automated synthesis of analog electrical circuits by means of genetic programming. *IEEE Transactions on Evolutionary Computation*, v. 1, n. 2, p. 109-128.
- Krishnakumar, K. & D.E. Goldberg (1992). Control system optimization using genetic algorithms. *Journal of Guidance, Control, and Dynamics*, v. 15, n. 3, p. 735-740.
- Kristinnson, K. & G.A. Dumont (1992). System identification and control using genetic algorithms. *IEEE*

- Transactions on Systems, Man and Cybernetics*, v. 22, n. 5, p. 1033-1046.
- Kumar, A., R.M. Pathak, Y.P. Gupta, & H.R. Parsaei (1995). Genetic algorithm based reliability optimization for computer network expansion. *IEEE Transactions on Reliability*, v. 44, n. 1, p. 63-72.
- Langdon, W.B. (1996). Scheduling planned maintenance of the national grid. *Evolutionary Computing AISB Workshop - Lecture Notes in Computer Science*, v. 993, p. 132-153.
- Lee, K.Y. & Y-M. Park (1995). Optimization method for reactive power planning by using a modified simple genetic algorithm. *IEEE Transactions on Power Systems*, v. 10, n. 4, p. 1843-1850.
- Lee, Y.D. & B.H. Lee (1997). Genetic trajectory planner for a manipulator with acceleration parameterization. *Journal of Universal Computer Science*, v. 3, n. 9, p. 1056-1073.
- Levitin, G., S. Mazal-Tov & D. Elmakis (1994). Optimal sectionalizer allocation in electric distribution systems by genetic algorithm. *Power Systems Research*, v. 31, n. 2, p. 97-102.
- Li, F., Y.H. Song & R. Morgan (1994). Genetic algorithms based optimisation approach to power system economic dispatch. *Proceedings of the 29<sup>th</sup> Universities Power Engineering Conference*, Galway, Ireland, v. 2, p. 680-683.
- Lienig, J. (1997). A parallel genetic algorithm for performance-driven VLSI routing. *IEEE Transactions on Evolutionary Computation*, v. 1, n. 1, p. 29-39.
- Lindhorst, G. (1998). Relational genetic algorithms: with application to surface mount technology placement machines. *Genetic Programming 1998: Proceedings of the 3<sup>rd</sup> Annual Conference*, Madison, USA, p. 543-550.
- Man, K.F., K.S. Tang, S. Kwong & W.A. Halang (1999). *Genetic Algorithms Concepts and Designs*. 2<sup>nd</sup> ed. Berlin: Springer-Verlag.
- Marrison, C.I. & R.F. Stengel (1997). Robust control system design using random search and genetic algorithms. *IEEE Transactions on Automatic Control*, v. 42, n. 6, p. 835-839.
- Martin, R.S. & J.P. Knight (1993). Genetic algorithms for the optimization of integrated circuits synthesis. *Proceedings of 5<sup>th</sup> International Conference on Genetic Algorithms*, Urbana-Champaign, USA, p. 432-438.
- Mazumder, P. & E.M. Rudnik (1999). *Genetic Algorithms for VLSI Design, Layout and Test Automation*. Englewood Cliffs, USA: Prentice-Hall.
- McGregor, D.R., M.O. Odetayo & D. Dasgupta (1992). Adaptive control of a dynamic system using genetic-based methods. *Proceedings of the 1992 IEEE International Symposium on Intelligent Control*, p. 521-525.
- Michalewicz, Z., D. Dasgupta, R.G. Le Riche & M. Schoenauer (1996). Evolutionary Algorithms for Constrained Engineering Problems. *Computers & Industrial Engineering Journal*, v. 30, n. 2, p. 851-870.
- Michalewicz, Z., C.Z. Janikow & J.R. Krawczyk. (1992). A Modified genetic algorithm for optimal control problems. *Computers & Mathematics with Applications*, v. 23, n. 12, p. 83-94.
- Miller, J.F. & P. Thomson (1998). Evolving digital electronic circuits for real-valued function generation using a genetic algorithm. *Genetic Programming 1998: Proceedings of the 3<sup>rd</sup> Annual Conference*, Madison, USA, p. 863-868.
- Mitchell, M. (1996). *An Introduction to Genetic Algorithms*. Cambridge: MIT Press.
- Nordvik, J.P. & J.M. Renders (1991). Genetic algorithms and their potential for use in process control: a case study. *Proceedings of the 4<sup>th</sup> International Conference on Genetic Algorithms*, San Diego, USA, p. 480-486.
- Park, S-H., Y-H. Kim, Y-K. Choi, H-C. Cho, & H-T. Jeon (1993). Self-organization of fuzzy rule base using genetic algorithm. *Proceedings of the 5<sup>th</sup> International Fuzzy Systems Association World Congress*, Seoul, Korea, p. 881-886.
- Pham, D.T. & D. Karaboga (1991). Optimun design of fuzzy logic controllers using genetic algorithms. *Journal of Systems Engineering*, v. 1, n. 2, p. 114-118.
- Pierre, S. & G. Legault (1998). A genetic algorithm for designing distributed computer network topologies. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, v. 28, n. 2, p. 249-258.
- Ram, A., R. Arkin, G. Boone & M. Pearce (1994). Using genetic algorithms to learn reactive control parameters for autonomous robotic navigation. *Adaptive Behavior*, v. 2, n. 3, p. 277-305.
- Rudnick, E.M., J.H. Patel, G.S. Greenstein, T.M. Niermann (1997). A genetic algorithm framework for test generation. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, v. 16, n. 9, p. 1034-1043.
- Schnecke, V. & O. Vornberger (1997). Hybrid genetic algorithms for constrained placement problems. *IEEE Transactions on Evolutionary Computation*, v. 1, n. 4, p. 266-277.
- Sevenster, A.A. & A.P. Engelbrecht (1996) GARTNet: a genetic algorithm for routing in telecommunications networks. *Proceedings of the Symposium on Control, Optimization and Supervision*, Lille, France, v. 2, p. 1106-1111.
- Silveira, C.H., L.S. Coelho & M.F.M. Campos (1994). The use of genetic algorithms for the evaluation of inverse kinematics of manipulators. *Anais do 1º Congresso Brasileiro de Redes Neurais*, Itajubá, MG.
- Tan, K.C. & Y. Li (1997). Evolutionary system identification in the time-domain. *Journal of Systems and Control Engineering*, part I, v. 211, n. 4, p. 319-323.

Tang, K.S., K.F. Man & K.T. Ko (1997). Wireless LAN design using hierarchical genetic algorithm. *Proceedings of the 7<sup>th</sup> International Conference on Genetic Algorithms*, East Lansing, USA, p. 629-635.

Tang, K.S., K.F. Man, S. Kwong & Z.F. Liu (1998a). Design and optimization of IIR filter structure using hierarchical genetic algorithms. *IEEE Transactions on Industrial Electronics*, v. 45, n. 3, p. 481-487.

Tang, K.S., K.F. Man, Z.F. Liu, & S. Kwong (1998b). Minimal fuzzy memberships and rules using hierarchical genetic algorithms. *IEEE Transactions on Industrial Electronics*, v. 45, n. 1, p. 162-169.

Tanomaru, J. & K. Oka (1995). Automatic wire routing using a customized genetic algorithm. *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, Vancouver, Canada, v. 2, p. 2971-2976.

Thomson, A. (1997). An evolved circuit, intrinsic in silicon, entwined with physics. In Higuchi, T, Iwata, M, e Liu, W (eds.), *Proceedings of the 1<sup>st</sup> International Conference on Evolvable Systems: From Biology to Hardware - Lecture Notes in Computer Science*, v. 1259, p. 390-405.

Toogood, R., H. Hao & C. Wong (1995). Robot path planning using genetic algorithms. *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, Vancouver, Canada, v. 1, p. 489-494.

Tzes, A., P-Y. Peng & J. Guty (1998). Genetic-based fuzzy clustering for DC-motor friction identification and compensation. *IEEE Transactions on Control Systems Technology*, v. 6, n. 4, p. 462-472.

Undrill, P.E., K.A. Delibasis & G.G. Cameron (1997). Stack filter design using a parallel implementation of genetic algorithms. *Journal of Universal Computer Science*, v. 3, n. 7, p. 821-834.

Yen, J. & W. Gillespie (1995). Integrating global and local evaluations for fuzzy model identification using genetic algorithms. *Proceedings of the 6<sup>th</sup> International Fuzzy Systems Association World Congress*, São Paulo, Brazil. v. 1, p. 121-124.

Wang, X-D. & T. Chen (1995). Performance and area optimization of VLSI system using genetic algorithms. *VLSI Design*, v. 3, n. 1, p. 43-51.

White, M.S. & S.J. Flockton (1994). Genetic algorithms for digital signal processing. *Evolutionary Computing AISB Workshop - Lecture Notes in Computer Science*, v. 865, p. 291-303.

Wong, K.P. & Y.W. Wong (1994). Genetic and genetic/simulated-annealing approaches to economic dispatch. *IEE Proceedings C: Generation, Transmission and Distribution*, v. 141, n. 5, p. 507-513.