

Otimização Matemática Através de Algoritmos Genéticos

Fernando Concatto¹

¹Bacharelado em Ciência da Computação – Universidade do Vale do Itajaí (UNIVALI)
Caixa Postal 360 – CEP 88302-202 – Itajaí – SC – Brasil

fernandoconcatto@edu.univali.br

Abstract. *Problems involving mathematical optimization are ubiquitous in many fields of science. Since many of them exhibit elevated computational complexity, several approximation methods have been developed to solve such problems. This paper describes the notion of genetic algorithms, which employs concepts of evolution and natural selection to approximate the highest or lowest value of a function, improving candidate solutions with the passing of generations.*

Resumo. *Problemas envolvendo otimização matemática são ubíquos em diversas áreas da ciência. Como grande parte desses problemas apresenta uma dificuldade computacional muito elevada, diversos métodos aproximados para resolvê-los foram desenvolvidos. Este trabalho descreve os algoritmos genéticos, que aplicam conceitos de evolução e seleção natural para aproximar o ponto máximo ou mínimo de uma função, aprimorando soluções candidatas com o decorrer das gerações.*

1. Introdução

O estudo de problemas e suas soluções compõe o cerne do campo da Ciência da Computação. Para efetuar a solução dos problemas, uma sequência finita de operações deve ser desenvolvida; tais sequências são conhecidas como algoritmos. Problemas abordados por cientistas da computação possuem uma enorme gama de formulações e dificuldades, alguns tão complexos que demandam mais do que o tempo de vida um ser humano para serem resolvidos.

Um exemplo de problema com esta característica é o Problema do Caixeiro Viajante, definido da seguinte forma: dado um conjunto de n cidades e as distâncias entre elas, qual a rota mais curta que parte de uma cidade, passa por todas as outras e retorna para a cidade inicial? Como existem $(n - 1)!/2$ rotas possíveis, descobrir a menor rota entre 30 cidades através de um algoritmo exato requer muito mais do que um milhão de anos de processamento [MacGregor and Chu 2011].

Para tratar de problemas com esse nível de complexidade, métodos **heurísticos** são usualmente aplicados. Algoritmos heurísticos são potencialmente capazes de resolver problemas complexos em pouquíssimo tempo, porém não garantem que a solução exata seja encontrada, fornecendo apenas aproximações, que geralmente são suficientes para grande parte dos problemas reais [Kokash 2005]. Exemplos de algoritmos heurísticos para problemas específicos incluem o algoritmo de Christofides para o problema do Caixeiro Viajante [Christofides 1976], mencionado anteriormente, e o algoritmo de Kernighan-Lin para particionamento de grafos [Kernighan and Lin 1970].

Entretanto, a categoria de algoritmos heurísticos que mais recebeu atenção da comunidade científica é a que envolve a resolução de **problemas de otimização**, caracterizada pela busca por um valor $x \in X$ que maximiza ou minimiza uma função $f : X \rightarrow \mathbb{R}$, geralmente chamada de **função objetivo**. Grande parte dos problemas, tanto inéditos quanto clássicos, podem ser formulados como problemas de otimização, incluindo o Problema do Caixeiro Viajante, onde a função objetivo pode ser descrita por:

$$\min f(x) = \sum_{i=1}^{n-1} d(x_i, x_{i+1}) + d(x_n, x_1) \quad (1)$$

onde x é o conjunto que descreve a rota candidata de tamanho n e $d(x_i, x_j)$ é a distância entre as cidades x_i e x_j . Desta forma, um algoritmo pode mover-se iterativamente pelo *espaço de busca*, que contém todas as rotas possíveis, buscando um conjunto x que minimize a função $f(x)$. Porém, este espaço de busca (assim como de vários outros problemas) apresenta uma quantidade excessivamente grande de elementos, exigindo então a aplicação de métodos heurísticos.

Problemas de otimização são objeto de estudo de diversas disciplinas, como pesquisa operacional, economia e as ciências naturais. A análise numérica é um dos campos com maior influência no estudo dos problemas de otimização, devido ao foco em métodos iterativos e aproximados, como o Método de Newton, através do qual é possível encontrar soluções ótimas quando aplicado à primeira derivada de uma função. Outro método capaz de encontrar pontos críticos é o Método dos Gradientes, que guia uma solução candidata para um ponto mínimo ou máximo a partir do gradiente de uma função. Este método é frequentemente aplicado em Redes Neurais Artificiais para conduzir a aprendizagem da rede [Haykin 1998].

Diversos outros métodos numéricos existem para encontrar pontos ótimos de uma função objetivo, como o método Simplex, um algoritmo exato para problemas de programação linear. Entretanto, a maior parte desses métodos só é capaz de convergir para um ótimo global quando o espaço de busca do problema é convexo, isto é, quando a função objetivo não possui **ótimos locais** [Lachtermacher 2009]. Ótimos locais consistem em pontos onde todas as soluções vizinhas possuem qualidade inferior à solução do ponto atual, porém soluções superiores podem estar localizadas em pontos distantes do presente. Desta forma, para funções com espaços de busca não-convexos, como é o caso em diversos problemas de otimização combinatória, os métodos supracitados são incapazes de encontrar um ótimo global. A figura 1 exemplifica graficamente um espaço de busca não-convexo.

Este trabalho busca descrever o método heurístico conhecido como Algoritmo Genético, que utiliza conceitos da biologia evolucionária para encontrar o ótimo global de uma função objetivo que pode possuir ótimos locais. Para este fim, a seção 2 introduz alguns conceitos relativos à evolução por seleção natural; por sua vez, a seção 3 apresenta a construção dos algoritmos genéticos, descrevendo sua estrutura fundamental; já a seção 4 define os operadores genéticos de seleção, mutação e reprodução. A seção 5 apresenta algumas aplicações dos algoritmos genéticos, e por fim, a seção 6 apresenta uma breve discussão sobre o método, concluindo o trabalho.

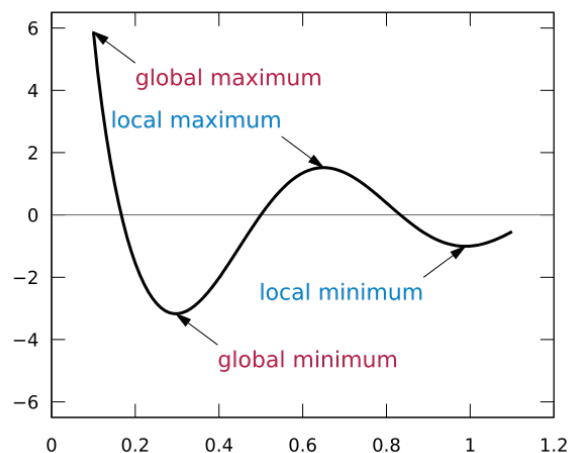


Figura 1. Exemplo de espaço de busca não-convexo com dois ótimos locais.

2. O processo evolucionário

O conceito de evolução de organismos é bastante antigo, datando da época dos filósofos gregos pré-socráticos [Hull 1967]. A noção de seleção natural foi desenvolvida por Charles Darwin, com a publicação de seu livro “A Origem das Espécies”, de 1859. Nele, Darwin propõe que populações evoluem ao longo de gerações através de um processo de seleção que favorece organismos com aptidão alta, medida através do sucesso reprodutivo. Indivíduos aptos produzem descendentes, que herdam boa parte de suas características (genes), e portanto também tendem a possuir elevado sucesso reprodutivo.

Desta maneira, a cada geração, os melhores indivíduos da população são escolhidos naturalmente para reprodução, passando adiante suas características, enquanto os indivíduos menos aptos tendem a desaparecer. Assim, a próxima geração será composta tanto pelos indivíduos aptos sobreviventes quanto por seus descendentes, ambos possuindo uma aptidão média superior à geração passada. Algoritmos genéticos buscam simular esse processo, modelando os mecanismos da natureza de forma computacional, com o intuito de otimizar uma função objetivo.

3. Os algoritmos genéticos

Algoritmos genéticos são classificados como meta-heurísticas, ou seja, são independentes do problema sendo resolvido. Qualquer problema que possa ser formulado por uma função objetivo pode ser solucionado por um algoritmo genético. Adicionalmente, algoritmos genéticos não possuem uma definição absolutamente rigorosa quanto aos detalhes dos procedimentos realizados; ao invés disso, são caracterizados pela existência de alguns elementos fundamentais [Mitchell 1998], vistos adiante.

3.1. Modelagem de cromossomos

O primeiro destes elementos é a especificação dos **cromossomos**, que caracterizam os indivíduos da população. Cromossomos representam as soluções do problema, e a maneira como eles são codificados é um dos principais fatores envolvidos no desenvolvimento de um algoritmo genético bem sucedido.

Historicamente, o método mais utilizado para modelar cromossomos foi a codificação por cadeias de bits, porém outros métodos podem ser empregados, especialmente para problemas mais complexos [Mitchell 1998]. Por exemplo, para o Problema do Caixeiro Viajante, apresentado na introdução deste trabalho, um cromossomo pode ser constituído pela sequência de cidades que especifica a rota a ser tomada.

3.2. Função de aptidão

O segundo elemento fundamental de um algoritmo genético é a função objetivo propriamente dita, denominada **função de aptidão** nesse contexto. A função de aptidão é responsável por avaliar a qualidade dos indivíduos, e é utilizada para selecioná-los para reprodução. Geralmente, devido ao caráter biológico, a função de aptidão é maximizada; caso a intenção seja a minimizar a função, é possível utilizar o inverso multiplicativo da função, definido como $1/f(x)$.

3.3. Operadores genéticos

Por fim, os elementos que concluem a base dos algoritmos genéticos são os **operadores genéticos**. Os principais operadores são a **seleção**, que utiliza a função de aptidão para selecionar indivíduos; a **reprodução** (ou cruzamento), que combina dois indivíduos para produzir um descendente; e a **mutação**, que aleatoriamente altera um ou mais genes do cromossomo. Estes operadores e as técnicas utilizadas para implementá-los são detalhados na seção 4.

3.4. Evolução

Com todos estes elementos definidos, resta especificar o procedimento de evolução. Este é um dos segmentos mais flexíveis de um algoritmo genético. Muitas variações existem para a implementação deste processo, algumas extremamente complexas e outras razoavelmente simples. Um exemplo básico, porém bastante eficiente para grande parte dos problemas, é apresentado adiante.

Inicialmente, uma população de n indivíduos é estabelecida, com seus cromossomos construídos aleatoriamente ou através de outros métodos. Como a otimização por algoritmos genéticos é um processo iterativo, os seguintes passos são repetidos até que uma solução satisfatória seja encontrada. A primeira etapa consiste no cálculo da aptidão de cada indivíduo x da população, através da função de aptidão $f(x)$. Na sequência, uma nova geração de indivíduos é produzida a partir da aplicação dos operadores genéticos.

Dois indivíduos são selecionados através do operador de seleção, que favorece indivíduos mais aptos. Então, um ou mais filhos são concebidos pelo operador de reprodução, herdando as características de seus pais. Cada filho passa pelo operador de mutação, possivelmente sofrendo alterações genéticas em seu cromossomo. Este procedimento é repetido até que uma quantidade suficiente de filhos seja gerada.

Por fim, a população de indivíduos que foram selecionados para reprodução é unida com a população de filhos concebidos, formando uma nova população de tamanho n . Este processo é repetido até que uma determinada condição de término seja atingida. O algoritmo 1 formaliza essa sequência de passos por meio de uma representação em pseudocódigo.

Algoritmo 1: Algoritmo genético básico

Entrada: Tamanho da população (n), chance de mutação (m)

Saída : Indivíduos evoluídos

```
1  $populacao \leftarrow gerarPopulacaoInicial(n)$ 
2 repeat
3   foreach  $individuo \in populacao$  do
4      $individuo.aptidao \leftarrow calcularAptidao(individuo)$ 
5    $filhos \leftarrow \emptyset$ 
6   while quantidade de filhos for insuficiente do
7      $pais \leftarrow selecionar(populacao)$ 
8      $filho \leftarrow reproduzir(pais)$ 
9     if  $aleatorio() < m$  then
10       $aplicarMutacao(filho)$ 
11       $filhos \leftarrow filhos \cup \{filho\}$ 
12    $populacao \leftarrow combinar(populacao, filhos)$ 
13 until condição de parada não for atingida
14 return  $populacao$ 
```

Ao término deste algoritmo, caso não haja nenhuma falha na definição dos cromossomos, função de aptidão ou operadores genéticos, a população final possuirá uma qualidade consideravelmente superior à inicial, dependendo da topologia do espaço de busca do problema e da quantidade de gerações decorridas. Entretanto, por se tratar de um algoritmo heurístico, não há garantia de que soluções ótimas serão encontradas, mesmo com uma grande quantidade de gerações.

4. Operadores genéticos

Os operadores genéticos constituem as técnicas utilizadas para guiar as soluções candidatas (indivíduos) em direção a estados cada vez mais próximos da solução ótima. Uma breve introdução aos operadores de seleção, reprodução e mutação foi realizada na seção 3.3; esta seção é dedicada a apresentar com detalhes as particularidades de cada operador.

4.1. Operador de seleção

O operador de seleção é responsável por escolher indivíduos para reprodução. O método de seleção deve enfatizar indivíduos aptos, para que seus descendentes também sejam aptos. Entretanto, também deve haver uma baixa chance de selecionar indivíduos com pouca aptidão – caso contrário, haverá pouca diversidade genética na população, dificultando ou até mesmo impedindo a evolução das soluções para um estado melhor [Mitchell 1998].

Diversos métodos de seleção foram propostos por pesquisadores no campo. A seguir, as técnicas conhecidas como *seleção por roleta* e *seleção por torneio* serão descritas, por serem amplamente utilizadas e razoavelmente simples.

4.1.1. Seleção por roleta

O método de seleção por roleta envolve a atribuição de uma chance de seleção para cada indivíduo proporcional à sua aptidão. Conceitualmente, este método constrói uma roleta onde cada casa corresponde a um indivíduo, com casas maiores representando indivíduos mais aptos. Esta técnica é uma das mais antigas, datada das primeiras pesquisas sobre algoritmos genéticos, e ainda é frequentemente utilizada hoje em dia.

Para efetuar a seleção por roleta, o primeiro passo a ser realizado é somar a aptidão de todos os indivíduos da população. Na sequência, um número aleatório r é escolhido, no intervalo de 0 até a soma calculada. Então, a sequência de indivíduos da população é percorrida, acumulando a aptidão de cada indivíduo visitado até que esse valor ultrapasse o número aleatório r . O último indivíduo visitado é escolhido para reprodução [Mitchell 1998].

4.1.2. Seleção por torneio

A seleção por torneio foi desenvolvida um pouco depois da seleção por roleta, e apresenta uma complexidade computacional significativamente menor. O método da roleta requer o somatório da aptidão dos indivíduos da população, que demanda uma quantidade de operações proporcional ao tamanho da população; a seleção por torneio não exibe essa propriedade

Na técnica da seleção por torneio, dois indivíduos são escolhidos da população de forma completamente aleatória. Então, um número é escolhido aleatoriamente no intervalo de 0 a 1. Se esse número for inferior a um parâmetro k previamente definido, o indivíduo mais apto entre eles será escolhido para reprodução; caso contrário, o indivíduo menos apto é selecionado [Mitchell 1998]. Geralmente, valores próximos de 0.75 são escolhidos para o parâmetro k .

4.2. Operador de reprodução

O operador de reprodução é um dos aspectos mais marcantes dos algoritmos genéticos. Este operador é responsável por combinar os cromossomos dos indivíduos selecionados, produzindo filhos com características similares às dos seus pais. Na maior parte das vezes, um filho possuirá aptidão superior à de seus pais, mas também existe a possibilidade da criação de um filho com aptidão muito inferior. A probabilidade de conceber um filho mais apto que seus pais depende de diversos fatores, como a modelagem dos cromossomos, a complexidade do problema e do próprio operador de reprodução.

A técnica mais básica para efetuar a reprodução é o **cruzamento de ponto único**. Neste método, os genes de um dos pais são copiados para o cromossomo do filho até que um ponto previamente especificado (fixo ou aleatório) seja atingido. A partir desse ponto, os genes do outro pai passam a ser copiados, até que o cromossomo do filho esteja totalmente formado.

Entretanto, para cromossomos mais complexos, com quantidade elevada de genes, a técnica de cruzamento de ponto único se mostra insuficiente; nestes casos, a maioria dos filhos apresentaria aptidão inferior à de seus pais. Desta forma, outros métodos de

reprodução foram desenvolvidos, como o **cruzamento de múltiplos pontos**, que é equivalente ao cruzamento de ponto único, porém com mais pontos de troca (geralmente dois); e o **cruzamento uniforme paramétrico**, onde a troca acontece a cada gene copiado, com chance igual a um parâmetro p [Mitchell 1998]. A figura 2 apresenta uma demonstração gráfica destes três métodos.

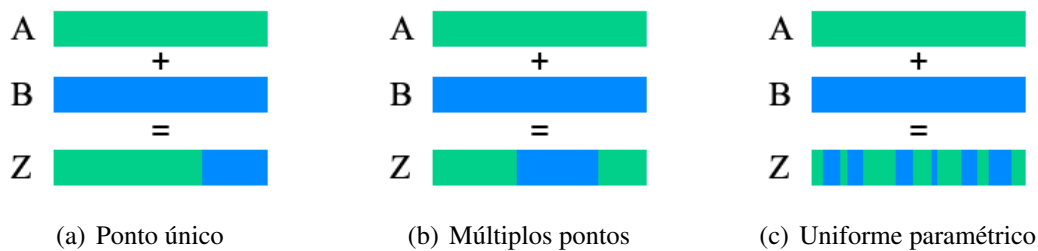


Figura 2. Métodos de reprodução entre dois pais A e B, produzindo um filho Z.

4.3. Operador de mutação

O último dos operadores de um algoritmo genético é o operador de mutação. Sua tarefa consiste em elevar a variação genética da população, impedindo que certos genes fiquem fixados em ótimos locais. Encontrar a solução ótima global evitando a fixação em ótimos locais é um problema extraordinariamente difícil, e o operador de mutação é uma das principais ferramentas utilizadas pelos algoritmos genéticos na resolução desse problema, juntamente com o operador de reprodução [Mitchell 1998].

O operador de mutação deve gerar alguma perturbação no cromossomo de um indivíduo. Esta perturbação pode tomar diversas formas, sendo que as mais comuns envolvem a alteração de um ou mais genes do cromossomo para um valor diferente do atual. Para o caso de cadeias de bits, o gene escolhido para mutação pode ter seu bit invertido; para o caso de valores reais, um número aleatório pode ser somado ou subtraído ao valor do gene, ou o valor pode ser simplesmente substituído por um número aleatório qualquer.

5. Aplicações dos algoritmos genéticos

Por ser uma meta-heurística, o método de otimização por algoritmos genéticos encontra utilização em uma enorme gama de problemas. Pelo mesmo motivo, entretanto, os algoritmos genéticos não garantem que a solução ótima será encontrada, e portanto devem ser utilizados somente em problemas onde soluções aproximadas são admissíveis. Geralmente, problemas com essa característica apresentam espaços de busca extremamente extensos, como o Problema do Caixeiro Viajante, apresentado na seção 1. Alguns exemplos práticos são descritos adiante.

Um caso recente de aplicação de algoritmos genéticos ocorreu durante o evento Opção Profissional por Área (OPA) de 2016, organizado pela Universidade do Vale do Itajaí. O evento tem como público-alvo jovens que se encontram no processo de escolha de um curso superior. Para apresentar o curso de Ciência da Computação, os integrantes do Laboratório de Inteligência Aplicada desenvolveram um algoritmo genético que evolui imagens inicialmente compostas por pixels aleatórios em direção a uma imagem real,

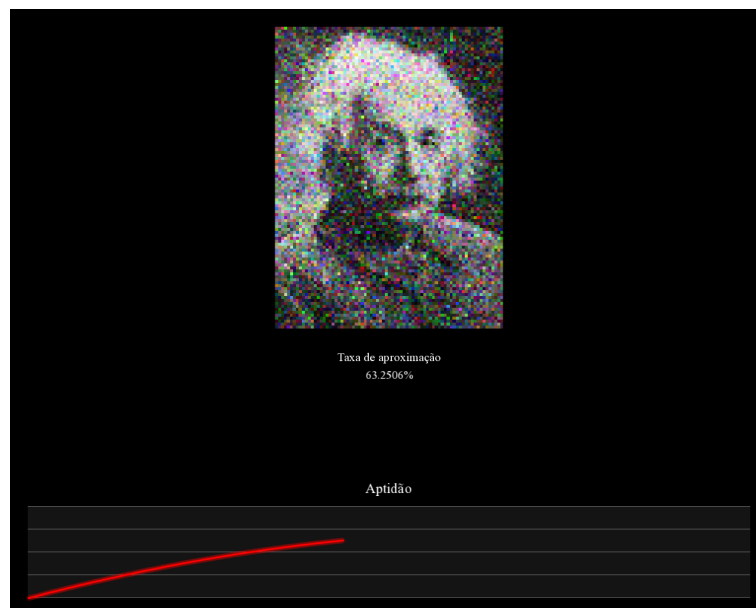


Figura 3. Imagem no processo de evolução em direção a uma fotografia do físico Albert Einstein através do algoritmo genético apresentado no OPA 2016.

desafiando os espectadores a descobrir qual era a imagem alvo enquanto introduziam conceitos de inteligência artificial e complexidade computacional. Um exemplo de execução deste algoritmo é apresentado na figura 3.

Outra aplicação de algoritmos genéticos, desta vez no cenário da análise de redes complexas, foi na detecção de comunidades, caracterizadas por grupos de indivíduos (também conhecidos como nodos ou vértices) fortemente conectados entre si. Neste contexto, os cromossomos representam possíveis divisões da rede em comunidades, e a função de aptidão calcula a qualidade da divisão, através de função de modularidade proposta por [Newman and Girvan 2004] ou alguma variação da mesma. Por meio de experimentos, a eficiência dos algoritmos genéticos na detecção de comunidades foi evidenciada, superando diversos métodos similares tanto em qualidade quanto em desempenho [Tasgin et al. 2007, Pizzuti 2008].

Algoritmos genéticos foram aplicados em muitos outros problemas com sucesso considerável. O trabalho de [Ross and Corne 1994] apresenta uma visão geral sobre esses problemas, incluindo a otimização de planejamento (*scheduling problem*), que consiste na organização de uma agenda, evitando conflitos espaciais e temporais, e também apresenta algumas aplicações no campo da engenharia, como no desenvolvimento de circuitos integrados, rotores de helicópteros, controle de braços robóticos, entre outras aplicações.

6. Discussão e conclusões

Neste trabalho, a técnica de otimização matemática por algoritmos genéticos foi examinada. As características fundamentais de um algoritmo genético foram apresentadas, juntamente com os conceitos de problemas de otimização, funções objetivo e espaços de busca. Adicionalmente, a construção de um algoritmo genético básico foi exibida.

A natureza heurística dos algoritmos genéticos faz com que seu uso seja propício em problemas complexos, onde a obtenção de uma solução exata demandaria um tempo

de busca intoleravelmente alto, e portanto uma solução aproximada é suficiente. Através de uma pesquisa bibliográfica, uma série de problemas com essas características foi identificada e apresentada; estes problemas foram solucionados com êxito através da utilização de algoritmos genéticos.

Por meio dos estudos realizados, foi possível constatar com clareza o potencial da resolução de problemas por otimização matemática, além da capacidade de resolução eficiente de tais problemas por algoritmos genéticos. Entretanto, como uma grande variedade de métodos heurísticos pode ser encontrada na literatura, as vantagens e desvantagens entre eles não é de fácil compreensão. Esta situação pode ser explorada em trabalhos futuros, através de análises comparativas entre algoritmos heurísticos; desta maneira, seria possível contextualizar os algoritmos genéticos em relação à outras técnicas com mais efetividade.

Referências

- Christofides, N. (1976). Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388, Graduate School of Industrial Administration, Carnegie Mellon University.
- Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition.
- Hull, D. L. (1967). The metaphysics of evolution. *The British Journal for the History of Science*, 3(4):309–337.
- Kernighan, B. W. and Lin, S. (1970). An efficient heuristic procedure for partitioning graphs. *The Bell System Technical Journal*, 49(2):291–307.
- Kokash, N. (2005). An introduction to heuristic algorithms. *Department of Informatics and Telecommunications*, (August).
- Lachtermacher, G. (2009). *Pesquisa Operacional na Tomada de Decisões*. Pearson.
- MacGregor, J. N. and Chu, Y. (2011). Human performance on the traveling salesman and related problems: A review. *The Journal of Problem Solving*, 3(2).
- Mitchell, M. (1998). *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA, USA.
- Newman, M. E. J. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Phys. Rev. E*, 69(2):026113.
- Pizzuti, C. (2008). GA-net: A genetic algorithm for community detection in social networks. In *Parallel Problem Solving from Nature – PPSN X*, pages 1081–1090. Springer Nature.
- Ross, P. and Corne, D. (1994). Applications of genetic algorithms. *AISB Quarterly on Evolutionary Computation*, 89:23–30.
- Tasgin, M., Herdagdelen, A., and Bingol, H. (2007). Community detection in complex networks using genetic algorithms. *arXiv preprint arXiv:0711.0491*.