

Comparando Algoritmos Genéticos e Transgenéticos para Otimizar a Configuração de um Serviço de Distribuição de Vídeo Baseado em Replicação Móvel

Luiz E. C. Leite¹ Guido Souza Filho² Marco Goldberg³ Elizabeth Gouvêa³
lecl@cin.ufpe.br guido@di.ufpb.br gold@dimap.ufrn.br beth@dimap.ufrn.br

¹CIN – UFPE, Campus Universitário, Recife, PE

²DI – CCEN – UFPB, Campus Universitário, João Pessoa, PB

³DIMAp – CCET – UFRN, Campus Universitário, Natal, RN

Abstract. *The video distribution service named DynaVideo uses mobile replication to adjust its configuration to demand variations. The service was designed to distribute video in a way that is independent of the video format and to interact with different types of clients. This paper presents a comparison between genetic and transgenetic algorithms developed to optimize the service configuration to a given demand, characterized by its clients' number and placing.*

Resumo. *O serviço de distribuição de vídeo denominado DynaVideo utiliza o conceito de replicação móvel para ajustar sua configuração de acordo com a variação da demanda. O serviço foi projetado para distribuição de vídeo de forma independente do formato do mesmo e para interagir com diferentes tipos de clientes. Este artigo apresenta uma comparação entre algoritmos genéticos e transgenéticos desenvolvidos para otimizar a configuração do serviço DynaVideo para uma dada demanda, definida em termos de número e localização dos clientes do serviço.*

1. Introdução

Aplicações que lidam com distribuição de vídeo, tais como a difusão de TV digital, normalmente trabalham com variações abruptas em sua demanda. O número, tipo e a localização dos clientes do serviço podem variar muito em um curto intervalo de tempo. Por exemplo, isso ocorre toda vez que um programa de grande audiência começa a ser exibido.

O serviço para distribuição de vídeo denominado DynaVideo (Dynamic Video Distribution Service) [1] foi concebido para distribuir vídeo de forma independente do seu formato e para interagir com diferentes tipos de clientes. A principal característica do DynaVideo é a possibilidade de ajustar dinamicamente a sua configuração para uma dada demanda. O ajuste é feito com base no conceito de replicação móvel. Réplicas do servidor de vídeo podem ser criadas em tempo real e movidas para pontos da rede que permitam o atendimento a uma determinada demanda. Para tal, as réplicas são

implementadas como código móvel. Com o intuito de tirar proveito da possibilidade de reconfiguração dinâmica do serviço, este trabalho apresenta heurísticas baseadas em algoritmos genéticos [3] transgenéticos [2], que foram desenvolvidas e implementadas para otimizar a configuração do sistema. Sempre que o sistema detecta uma variação em sua demanda, determinada pela entrada ou saída de um cliente, um algoritmo de otimização é acionado e, quando uma configuração melhor que a atual é encontrada, o serviço é reconfigurado.

A próxima seção apresenta os algoritmos utilizados para otimizar a configuração do serviço DynaVideo. A seção 3 apresenta os resultados de alguns testes realizados e a seção 4 apresenta as conclusões e perspectivas futuras desse trabalho.

2. Algoritmos de Otimização

Os algoritmos genéticos e transgenéticos são algoritmos que partem de um conjunto inicial de soluções para um dado problema (não necessariamente as melhores soluções). Cada solução é representada na forma de uma seqüência de valores (genes) denominada cromossomo. Esses cromossomos por sua vez são avaliados através de alguma função que indica quantitativamente a qualidade da solução. A cada iteração dos algoritmos, os cromossomos são manipulados visando melhorar o valor da sua função de avaliação. Ao término do algoritmo é escolhida a solução com melhor função de avaliação.

Nesse trabalho optou-se por utilizar uma *representação binária* para os cromossomos (cada gene só pode assumir um valor zero ou um), onde cada gene representa um servidor. Se um determinado servidor deve ser ativado, o gene que o representa assume valor um. Caso contrário, o referido gene assume valor zero. Essa representação indica apenas quais servidores estão ativos e quais estão inativos. A *função de avaliação* $f(c)$ de cada cromossomo deve, no entanto, representar: custo de ativação dos servidores $fs(c)$, os custos dos enlaces que interligam os servidores ativos $fe(c)$ e os custos das associações de cada cliente ao seu servidor $fc(i)$. Dessa forma, $f(i) = fs(i) + fe(i) + fc(i)$.

A função $fs(c)$ pode ser facilmente computada somando-se os custos de ativação de todos os servidores ativos na solução.

Para se determinar quais enlaces serão utilizados para conectar os servidores ativos, utilizou-se um algoritmo de geração de árvores de custo mínimo denominado algoritmo de PRIM [4]. A função $fe(c)$ é calculada fazendo-se o somatório dos custos associados a esses enlaces.

Para o cálculo de $fc(e)$, cada cliente deve ser associado ao servidor ativo capaz de atendê-lo com o menor custo, respeitando-se a capacidade do mesmo. O valor da função $fc(e)$ é então calculado fazendo-se o somatório dos custos de associação de cada cliente ao seu respectivo servidor.

Para *população inicial*, foram gerados 1000 cromossomos aleatoriamente.

Algoritmo Genético

A cada iteração do Algoritmo Genético, alguns indivíduos são selecionados para serem manipulados por operadores de Cruzamento e Mutação, gerando novos cromossomos.

Em seguida algumas soluções são descartadas através do método da roleta [3], de forma que o tamanho da população permaneça constante ao início de cada iteração.

Na *operação de cruzamento*, é gerada uma máscara (uma seqüência de bits) do mesmo comprimento dos cromossomos da população. Então, dois cromossomos são escolhidos através do método da roleta para permutarem seus genes entre si, apenas nas posições em que a máscara gerada assume valor 1. Neste trabalho, utilizou-se uma taxa de cruzamento de 30% da população a cada iteração do algoritmo.

O *operador de mutação* seleciona aleatoriamente duas posições em um cromossomo também escolhido aleatoriamente, e permuta os genes presentes nessas duas posições. Os demais genes são mantidos intactos. A *taxa de mutação* utilizada foi de 0,5 % do tamanho da população a cada iteração do algoritmo.

Algoritmos Transgenéticos

No caso do Algoritmo Transgenético, foi utilizado um algoritmo da classe ProtoG [2]. Na implementação realizada, inicialmente é gerado um cromossomo, denominado meme, da seguinte maneira: todos os servidores são colocados em uma lista ordenada pelo número de clientes que cada servidor consegue atender a um custo menor que o custo médio de atendimento a clientes de todos os servidores. Em seguida essa lista é percorrida do início ao fim e cada servidor é ativado no meme se essa ativação provocar uma melhoria no valor da função de avaliação do mesmo.

A cada iteração do algoritmo são realizadas duas operações: uma de intensificação e outra de diversificação. Na operação de intensificação, alguns genes do meme são selecionados de forma aleatória e são utilizados para substituir os genes equivalentes nos cromossomos da população. Se o cromossomo resultante possuir função de avaliação melhor do que a do cromossomo original, a operação é mantida. Caso contrário, ela é desfeita. Se o cromossomo resultante possuir função de avaliação melhor do que a do meme, o mesmo é substituído pelo cromossomo resultante. Na operação de diversificação, alguns genes gerados aleatoriamente são utilizados para atacar os cromossomos da população.

Tanto o algoritmo genético como o transgenético param após 1000 gerações sem que haja melhoria da melhor solução encontrada.

3.Resultados de Experimentos

Para se avaliar a performance do Algoritmo Genético e do Algoritmo Transgenético, foi desenvolvida uma ferramenta gráfica capaz de gerar instâncias do problema de otimização do DynaVideo. Utilizando-se essa ferramenta é possível que sejam especificados: o número de servidores com seus respectivos custos de ativação e capacidades de atendimento a clientes; o número de clientes que se deseja utilizar no problema; o custo dos enlaces conectando os servidores; e alguns parâmetros de rede para os enlaces conectando os clientes aos servidores, tais como: número de saltos, retardo, variação de retardo, taxa de erro e taxa de perda de pacotes. O custo de atendimento de um cliente por um determinado servidor é então calculado fazendo-se a soma ponderada desses parâmetros, onde pesos são atribuídos aos mesmos, de acordo com a influência que eles exercem sobre a qualidade do vídeo percebido pelos usuários finais.

Nas simulações realizadas, foram geradas instâncias aleatórias do problema de otimização do DynaVideo (os custos e capacidades dos servidores, custos dos enlaces entre servidores e parâmetros de rede foram gerados aleatoriamente), com diferentes quantidades de servidores e clientes. Cada instância foi utilizada como entrada tanto para o algoritmo Genético como para o Transgenético.

Nos experimentos realizados observou-se que para um número pequeno de servidores e clientes (até dez servidores ou até 50 clientes), os dois algoritmos obtêm os mesmos resultados. Entretanto, à medida que se aumenta o número de clientes e de servidores (mais de dez servidores e mais de cinquenta clientes), o algoritmo ProtoG apresenta resultados que chegam a ser até 43% melhores que os do Genético. Esse fato é justificado pelo tempo de convergência do algoritmo ProtoG ser menor que o do Genético. O tempo de convergência é de fundamental importância para o serviço DynaVideo, uma vez que a configuração do sistema não é estática. Assim, o cálculo de configurações otimizadas deve ser realizado em intervalos de tempo curtos. Considerando-se essas mesmas instâncias, os dois algoritmos obtiveram praticamente o mesmo resultado para intervalos de tempo superiores a trinta minutos.

4. Conclusões e Perspectivas Futuras

O serviço de distribuição de vídeo DynaVideo (Dynamic Video Distribution Service) foi projetado para ambientes genéricos de comunicação de dados, oferecendo suporte a diferentes formatos de vídeo e diferentes tipos de clientes. O artigo focou um dos pontos chave deste serviço: o algoritmo de otimização da configuração do serviço desenvolvido e implementado.

Uma vez que o serviço baseia-se em réplicas de servidores que podem mover-se pela rede, ele pode reconfigurar-se dinamicamente para atender a variações na demanda. Para computar configurações otimizadas, foram especificadas e implementadas heurísticas específicas para as características do serviço DynaVideo, baseadas em algoritmos genéticos e transgenéticos [2]. As duas implementações foram utilizadas para resolver problemas de diferentes tamanhos, em um simulador de redes desenvolvido especificamente para tal fim. Em alguns casos, a abordagem baseada em algoritmos transgenético conseguiu obter configurações para o serviço que eram até 43% mais baratas que as obtidas pela abordagem baseada em algoritmos genéticos. Entretanto, para problemas de pequeno porte ou mesmo permitindo-se que os algoritmos rodem por um tempo suficientemente grande, as duas abordagens apresentam resultados bastante parecidos.

Referências Bibliográficas

- [1] Leite, L. E., Filho, G., Batista, T., "DynaVideo - A Dynamic Video Distribution Service", *EG Multimedia 2001 proceedings book*, Springer-Wien, 2001.
- [2] Gouvêa, E., "Transgenética Computacional: Um Estudo Algorítmico", *Tese de Doutorado*, COPE/UFRN, Rio de Janeiro, 2001
- [3] Holland J., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, 1975
- [4] Prim, R. C., Shortest Connection Network and Some Generalizations, *Bell Sys. Tech J.*, Vol. 36, 1389-1401, 1957