

Algoritmos Genéticos para Operação Ótima de Sistemas de Recursos Hídricos em Tempo Real

Alcigeimes Batista Celeste, Koichi Suzuki e Akihiro Kadota

Ehime University, Department of Civil and Environmental Engineering, 3-Bunkyo-cho, Matsuyama, Ehime, 790-8577, Japão
e-mail: geimes@yahoo.com, ksuzuki@dpc.ehime-u.ac.jp, akado@dpc.ehime-u.ac.jp

Recebido: 28/10/02 – revisão: 03/02/03 – aceito: 18/07/03

RESUMO

Um modelo matemático de otimização baseado em algoritmos genéticos foi desenvolvido e aplicado para determinar a operação ótima em tempo real de um sistema de recursos hídricos sujeito a múltiplos usos e composto por um reservatório e um conjunto de poços. O sistema é responsável pelo abastecimento da cidade de Matsuyama, no Japão. O objetivo é encontrar a alocação ótima diária de água da barragem e dos poços para abastecimento e irrigação de acordo com a previsão de chuva num dado horizonte de operação. Para fins de comparação, o problema é resolvido também por uma técnica tradicional de otimização, mas os resultados mostram que o algoritmo genético encontra soluções melhores.

Palavras-chave: Operação de reservatórios, otimização, algoritmos genéticos.

INTRODUÇÃO

O crescente aumento na demanda de água, especialmente em regiões onde os recursos hídricos são escassos, requer um gerenciamento mais eficiente e busca por políticas cada vez mais elaboradas para a operação dos sistemas.

Sistemas de gerenciamento de recursos hídricos usam freqüentemente modelos matemáticos de otimização para identificar o melhor conjunto de planos e políticas que promovem uma distribuição justa e econômica da água. Os métodos mais comumente usados são Programação Linear (PL) e Programação Dinâmica (PD) (Yeh, 1985; Mujumdar e Ramesh, 1997; Belaineh et al., 1999). No entanto, algumas desvantagens dessas duas técnicas são a necessidade de relações lineares (no caso de PL) e a utilização de várias expressões matemáticas que precisam ser simplificadas para que os seus tamanhos e o custo de suas soluções sejam reduzidos (no caso de PD). Uma formulação mais geral e melhor tratamento das não linearidades são fornecidos através de Programação Não Linear (PNL) (Celeste et al., 1996), contudo os requisitos matemáticos são mais complicados e os algoritmos envolvidos não são eficientes para problemas complexos contendo funções multimodais, não diferenciáveis e não convexas. Algoritmos Genéticos (AGs) são técnicas de busca heurística as quais diversos autores têm mostrado que podem superar as limitações dos algoritmos tradicionais para a solução de problemas de otimização (Goldberg, 1989, Michalewicz, 1996; Michalewicz e Fogel, 2000). Embora existam muitas

aplicações de AGs em recursos hídricos, o seu uso para operação de sistemas não é abundante (Celeste et al., 2002; Wardlaw e Sharif, 1999).

Neste trabalho, um algoritmo genético foi construído e usado na resolução de um modelo de otimização para a operação ótima de um sistema de recursos hídricos em tempo real. Uma técnica baseada em Programação Quadrática Seqüencial (PQS) também é usada para a solução do modelo e o seu desempenho é comparado com o do AG.

SISTEMA DE ESTUDO E OBJETIVOS

O sistema estudado é responsável pelo abastecimento d'água da cidade de Matsuyama, capital da província de Ehime, localizada no norte da ilha de Shikoku, a menor das quatro ilhas principais do Japão. Matsuyama tem um clima moderado com uma temperatura média de 15,6 °C e uma precipitação média anual de 1.333 mm. Chove muito em junho e pouco em janeiro e a cidade recebe pouca neve e alguns tufões.

O sistema é composto por um reservatório (barragem do rio Ishite) sujeito a múltiplas finalidades e um conjunto de 26 poços localizados ao longo do rio Shigenobu como mostra a Figura 1. A barragem é usada para o controle de enchentes do rio Ishite e fornece metade do abastecimento da cidade enquanto que os poços fornecem a outra metade. O reservatório também é utilizado para irrigação da área norte do rio Ishite.

A cidade de Matsuyama sofre periodicamente com problemas originados pela escassez de água. Um exemplo disso foi o ano de 2002, no qual tinha havido falta de chuva na região. A Figura 2 mostra uma comparação da precipitação média e dos valores medidos (até começo de setembro de 2002) próximos ao reservatório, cujo nível d'água atingiu apenas 46,6% da capacidade em setembro. Além disso, a população da cidade tem crescido substancialmente fazendo com que também cresça a necessidade de um melhor gerenciamento dos recursos hídricos na região.

O principal objetivo da operação é determinar a

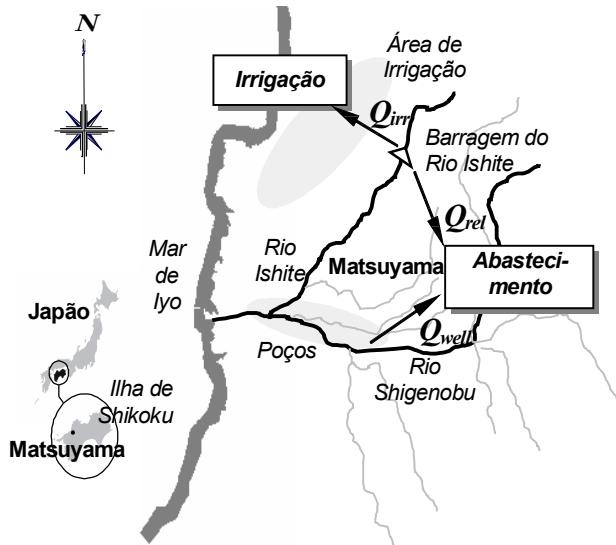


Figura 1 – Localização e layout do sistema.

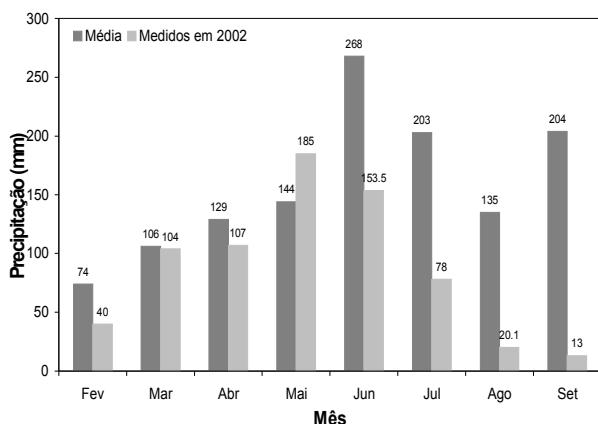


Figura 2 – Precipitação média e valores medidos até o começo de setembro de 2002 na região do reservatório.

melhor alocação de água da barragem e dos poços para abastecimento e irrigação para que atendam da melhor forma as respectivas demandas. Outra finalidade é manter

o volume do reservatório o mais próximo possível de um valor dado, para que o nível não caia consideravelmente. Além disso, a distribuição de água não pode comprometer a operação do sistema violando suas restrições e resultando no seu colapso.

O MODELO DE OTIMIZAÇÃO

As três principais variáveis de decisão do modelo são: Q_{rel} – alocação de água da barragem para abastecimento; Q_{well} – alocação de água dos poços para abastecimento; e Q_{irr} – alocação de água da barragem para irrigação.

De acordo com os objetivos de operação e considerando um horizonte de N dias, o modelo de otimização pode ser escrito através das expressões (1)-(7) abaixo. A função objetivo é formada pela soma dos desvios das variáveis de decisão de suas devidas metas. As restrições são compostas pela equação da continuidade, limitações dos componentes do sistema, etc.

$$\text{minimizar} \quad \sum_{t=1}^N \left\{ \left(\frac{Q_{rel}^t + Q_{well}^t - T_{dem}^t}{T_{dem}^t} \right)^2 + \left(\frac{Q_{irr}^t - T_{irr}^t}{T_{irr}^t} \right)^2 + \left(\frac{V_{stor}^t - T_{stor}}{T_{stor}} \right)^2 \right\} \quad (1)$$

sujeito a

1. Equação da continuidade para o reservatório:

$$V_{stor}^{t+1} = V_{stor}^t + V_{inf}^t - Q_{rel}^t - Q_{irr}^t - V_{spill}^t; \quad \forall t \quad (2)$$

2. O volume do reservatório (V_{stor}) pode variar apenas entre o volume morto (V_{stor}^{dead}) e a capacidade da barragem (V_{stor}^{max}):

$$V_{stor}^{dead} \leq V_{stor}^t \leq V_{stor}^{max}; \quad \forall t \quad (3)$$

3. O volume de água bombeado dos poços depende da precipitação (P) e não deve ultrapassar a capacidade das estações de tratamento de água subterrânea (Q_{well}^{max}):

$$Q_{well}^t \leq f(P^t); \quad \forall t \quad (4)$$

$$Q_{well}^t \leq Q_{well}^{max}; \quad \forall t \quad (5)$$

4. O volume de água retirado do reservatório não deve ultrapassar a capacidade das estações de tratamento de água superficial (Q_{rel}^{max}):

$$Q_{rel}^t \leq Q_{rel}^{max}; \forall t \quad (6)$$

5. Restrições de não negatividade:

$$Q_{rel}^t \geq 0; Q_{well}^t \geq 0; Q_{irr}^t \geq 0; \forall t \quad (7)$$

onde:

t : dia; T_{dem}^t : demanda para abastecimento no dia t ; T_{irr}^t : demanda para irrigação no dia t ; T_{stor} : meta de volume no reservatório; V_{inf}^t : afluxo d'água ao reservatório no dia t ; V_{spill}^t : descarga do vertedouro no dia t .

OPERAÇÃO EM TEMPO REAL

Como a operação do sistema acontece em tempo real, a alocação de água deve ser decidida sem perfeito conhecimento dos eventos futuros. O procedimento de operação em tempo real é formulado para aceitar dados de chuva previstos num horizonte de N dias. O modelo (1)-(7) é então resolvido para encontrar a alocação ótima para um período de N dias. Contudo, somente a alocação para o primeiro dia é implementada, pois as dos dias seguintes serão ajustadas durante as próximas execuções do modelo a partir das informações atualizadas de previsão. Sendo T o índice do dia atual e T_{tot} o número total de dias de operação, os passos do processo de simulação são como a seguir:

1. Atribuir $T = 1$ ($T = 1, 2, \dots, T_{tot}$) e especificar o volume inicial do reservatório;
2. Prever a chuva para os próximos N dias e usar a previsão para estimar os afluxos ao reservatório e o volume de água disponível nos poços para os próximos N dias;
3. Usar o modelo baseado em AGs para resolver o problema (1)-(7) e obter a alocação ótima de água da barragem e dos poços durante os próximos N dias;
4. Implementar apenas a alocação para o primeiro dia;
5. Atualizar o volume do reservatório usando o afluxo real ao reservatório e os defluxos calculados. Atribuir $T = T + 1$;
6. Se $T > T_{tot}$ parar; senão, voltar ao passo 2.

Para o passo 2, as estimativas dos afluxos ao reservatório e da água nos poços a partir da chuva prevista foram determinadas de duas formas: 1) uso de um modelo

de precipitação-vazão para determinar os afluxos a partir da chuva e 2) uso de um modelo de regressão linear múltipla para correlacionar chuva com nível de água médio nos poços e posteriormente determinar a quantidade de água disponível. Esses dois procedimentos são detalhados em Celeste (2002).

BREVE DESCRIÇÃO DO ALGORITMO GENÉTICO DESENVOLVIDO

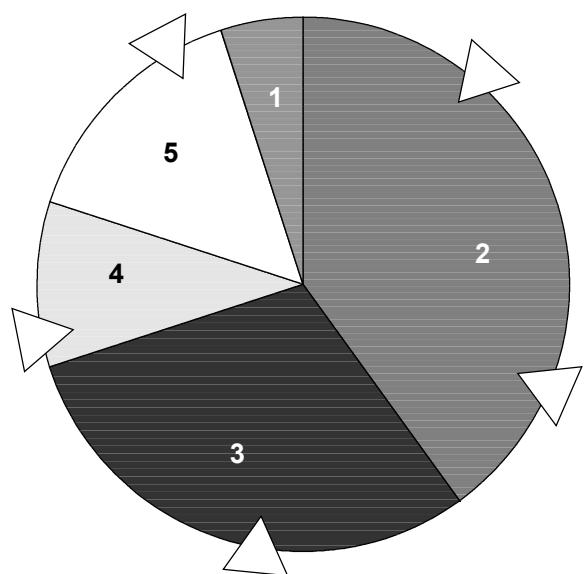


Figura 3 – Amostragem Estocástica Universal. Os pais selecionados são: 2 2 3 4 5.

Para a solução do modelo de otimização (1)-(7), um algoritmo genético foi desenvolvido e implementado no ambiente MATLAB. AGs imitam o processo de evolução por seleção natural, através de uma estratégia de "sobrevivência do mais apto". Em princípio, a busca começa com um conjunto de soluções aleatórias iniciais chamado de *população*. Indivíduos denominados *cromossomos* compõem a população. Cada cromossomo representa uma solução potencial para o problema e é descrito por uma cadeia de símbolos (geralmente bits binários). Cada solução é avaliada para que se tenha uma medida de *aptidão*. Uma nova população é então formada pela seleção dos indivíduos mais aptos. Alguns membros dessa nova população são submetidos a transformações através de operadores genéticos (*crossover* e *mutação*) para formar novas soluções, chamadas de *descendentes* ou *filhos*. Depois de sucessivas iterações, ou *gerações*, o algoritmo converge para o melhor cromossomo, que deve representar a solução ótima do problema. Geralmente, os indivíduos mais aptos tendem a se reproduzir e assim melhorar as gerações seguintes, mas

pode acontecer que indivíduos inferiores sobrevivam e também se reproduzam (Michalewicz, 1996; Gen e Cheng, 1997; Houck et al., 1995).

Já foi mostrado que a representação real das variáveis de decisão que compõem cada possível solução do problema é mais eficiente e produz melhores soluções em relação à representação binária tradicional (Michalewicz, 1996). O AG desenvolvido neste trabalho usa esse tipo de representação e incorpora *Amostragem Estocástica Universal* (SUS, do inglês Stochastic Universal Sampling) como procedimento de seleção. Proposto por Baker (1987), SUS é um método de seleção onde a população é representada por um gráfico em forma de torta da qual o tamanho de cada fatia é proporcional à aptidão do cromossomo que ela representa (Figura 3). Um número de ponteiros igual ao tamanho da população é colocado ao redor da torta e depois de um único giro o cromossomo apontado por cada ponteiro é escolhido para *recombinação* (crossover e mutação). Esse processo é melhor do que *Roda da Roleta*, aparentemente o mais popular procedimento de seleção, porque mantém a diversidade e previne que *super cromossomos* dominem a população (Gen e Cheng, 1997).

Seis tipos de operadores genéticos, como os usados no sistema GENOCOP de Michalewicz (1996), foram utilizados:

1. Mutação Uniforme: seleciona aleatoriamente uma variável x_k e atribui a ela um valor uniforme aleatório do intervalo (L_k, U_k) , os limites inferior e superior de x_k , respectivamente:

$$\text{Pai: } \mathbf{x} = (x_1, \dots, x_k, \dots, x_n) \quad (8)$$

$$\text{Filho: } \mathbf{x}' = (x_1, \dots, \overset{\cdot}{x_k}, \dots, x_n); \overset{\cdot}{x_k} \in (L_k, U_k)$$

2. Mutação Limite: variação da mutação uniforme que seleciona uma variável aleatoriamente e lhe atribui o valor L_k ou U_k :

$$\text{Filho: } \mathbf{x}' = (x_1, \dots, \overset{\cdot}{x_k}, \dots, x_n); \quad (9)$$

$$\overset{\cdot}{x_k} = \begin{cases} L_k & \text{se um número binário aleatório é 0} \\ U_k & \text{se um número binário aleatório é 1} \end{cases}$$

3. Mutação Não-Uniforme: seleciona uma variável aleatoriamente e lhe atribui um valor não uniforme:

$$\text{Filho: } \mathbf{x}' = (x_1, \dots, \overset{\cdot}{x_k}, \dots, x_n); \quad (10)$$

$$\overset{\cdot}{x_k} = \begin{cases} x_k + \psi(t, L_k - x_k) & \text{se um número binário} \\ & \text{aleatório é 0} \\ x_k + \psi(t, x_k - U_k) & \text{se um número binário} \\ & \text{aleatório é 1} \end{cases}$$

onde $\psi(t, y) = y \cdot r \cdot (1 - t/T)^b$; t é o número da geração; r é um número aleatório entre $(0,1)$; T é o número máximo de gerações; e b é um parâmetro que determina o grau de não-uniformidade do sistema.

4. Crossover Aritmético: produz duas combinações lineares complementares dos pais:

$$\mathbf{x}_1' = a \cdot \mathbf{x}_1 + (1-a) \cdot \mathbf{x}_2 \quad (11)$$

$$\mathbf{x}_2' = (1-a) \cdot \mathbf{x}_1 + a \cdot \mathbf{x}_2 \quad (12)$$

onde $a \in (0,1)$.

5. Crossover Simples: a “cabeça” (primeira seqüência de valores até uma posição aleatoriamente escolhida) de um indivíduo é conectada com a “cauda” (seqüência restante depois da posição de crossover) de outro e vice-versa produzindo dois novos filhos:

$$\text{Pai1: } \mathbf{x}_1 = (x_1, \dots, x_n) \quad (13)$$

$$\text{Pai2: } \mathbf{x}_2 = (y_1, \dots, y_n) \quad (14)$$

$$\text{Filho1: } \mathbf{x}_1' = (x_1, \dots, x_k, y_{k+1}, \dots, y_n) \quad (15)$$

$$\text{Filho2: } \mathbf{x}_2' = (y_1, \dots, y_k, x_{k+1}, \dots, x_n) \quad (16)$$

6. Crossover Heurístico: Realiza uma extrapolação linear entre os pais usando informação de aptidão:

$$\mathbf{x}_1' = \mathbf{x}_1 + r \cdot (\mathbf{x}_1 - \mathbf{x}_2) \quad (17)$$

$$\mathbf{x}_2' = \mathbf{x}_1 \quad (18)$$

onde r é um número aleatório entre 0 e 1, e \mathbf{x}_1 é melhor do que \mathbf{x}_2 em termos de aptidão. Se \mathbf{x}_1' é inviável, um outro valor de r é gerado e um outro filho é criado. Se depois de um certo número de tentativas a inviabilidade ainda permanecer, os filhos serão iguais aos pais.

No modelo (1)-(7), embora a função objetivo seja não linear, todas as restrições são do tipo linear (conjunto de restrições definidos pela expressão 2) ou restrições de domínio (desigualdades 3 a 7). Quando o conjunto de res-

trições é linear, os filhos gerados pelos operadores acima são sempre viáveis, considerando (L_k, U_k) como sendo o intervalo viável de uma variável x_k quando as outras variáveis são fixas (Michalewicz, 1996). L_k e U_k não são, portanto, necessariamente iguais aos verdadeiros valores dos limites das variáveis de decisão (l_k, u_k) como mostrado na Figura 4. Essa característica, requerida pelos operadores de mutação, é suficiente para lidar com as restrições do problema de otimização considerado.

A função de avaliação utilizada para determinar a aptidão de cada cromossomo na população é, em princípio, a função objetivo do problema de otimização. Contudo, depois que a função é avaliada para todos os cromossomos, a população é submetida ao chamado *ordenamento de aptidão*, um processo no qual os indivíduos são ordenados de acordo com os valores reais de aptidão e, em seguida, aptidões *reprodutivas* são atribuídas de acordo com um ordenamento linear. No caso de minimização, quanto menor o valor da função objetivo maior será a ordem e, consequentemente, a aptidão. Devido a isso, o efeito de um ou dois indivíduos extremos será desprezível no processo de seleção, independente de quanto maior ou menor os valores das suas aptidões são em relação ao resto da população (Beasley et al., 1993a,b).

Outra característica do algoritmo é o uso de um procedimento denominado *elitismo*, segundo o qual, em cada geração, o melhor cromossomo da população anterior é armazenado e substituído pelo pior indivíduo da nova população. Elitismo impede que o melhor cromossomo seja perdido de uma geração para outra através de crossover ou mutação.

Cada cromossomo da população inicial é gerado aleatoriamente. Se um indivíduo gerado não for viável, um outro cromossomo é gerado até que a viabilidade seja atingida.

RESULTADOS E DISCUSSÃO

Esta seção apresenta resultados da aplicação do modelo desenvolvido para alguns cenários.

Com finalidades de comparação, o problema de otimização foi resolvido não só pelo algoritmo genético descrito mas também por um procedimento baseado em cálculo, que é um programa integrante do *MATLAB Optimization Toolbox* (Optimization Toolbox user's guide, 1996), um conjunto de algoritmos especialmente desenvolvidos para resolver diversos tipos de problemas de otimização. O algoritmo escolhido usa um método de Programação Quadrática Sequencial (PQS), no qual um sub-problema de Programação Quadrática (PQ) é resolvido a cada iteração principal.

O uso do modelo em tempo real requer dados de entrada que são previsões reais de chuva. Contudo, esses dados não são disponíveis para anos passados. Dessa forma, todas as previsões de chuva foram assumidas como sendo “perfeitas”, ou seja, obtidas usando os verdadeiros dados de chuva medidos.

Todas as simulações foram realizadas para um ano de operação e um horizonte de operação de cinco dias. Assim, cada uma das três variáveis de decisão é formada por um vetor de cinco elementos (Tabela 1). Os valores das demandas e de volume alvo do reservatório foram escolhidos como na Tabela 2. A capacidade da barragem foi assumida como o volume de armazenamento até o nível de controle de cheias ($8,500,000 \text{ m}^3$). O volume morto do reservatório é de $2,200,000 \text{ m}^3$.

A Figura 5 mostra os resultados de uma operação usando a técnica de PQS. Essa operação usa os dados observados no ano de 1996 como chuva prevista. Nesta simulação, o volume inicial do reservatório foi considerado como sendo o valor máximo (reservatório cheio). Na Figura 5(a) são mostrados o comportamento do volume de água na barragem e as alocações ótimas para abastecimento e irrigação. A Figura 5(b) considera apenas os resultados para abastecimento e mostra o quanto a alocação total do reservatório e dos poços atendeu as demandas. A Figura 5(c) apresenta os resultados levando em conta apenas a irrigação e também mostra como as demandas foram atendidas. Pode-se notar que, para alguns dias do ano de operação, as demandas não foram totalmente atendidas.

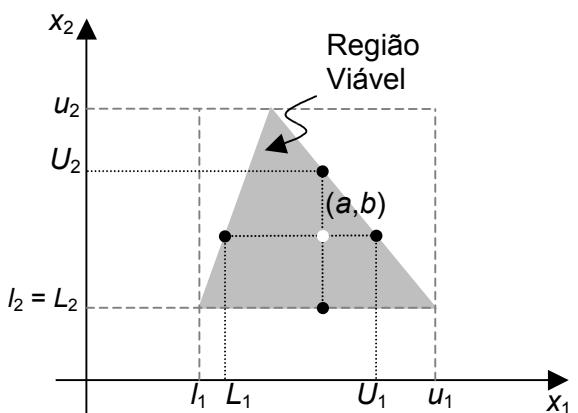


Figura 4 – Exemplo de determinação do intervalo viável para um dado ponto $(x_1, x_2) = (a, b)$ considerando um problema com duas variáveis de decisão x_1 e x_2 . (L_1, U_1) é o intervalo dentro do qual x_1 pode variar a partir de a fixando-se o valor de b . Similarmente, (L_2, U_2) é o intervalo dentro do qual x_2 pode variar a partir de b fixando-se o valor de a .

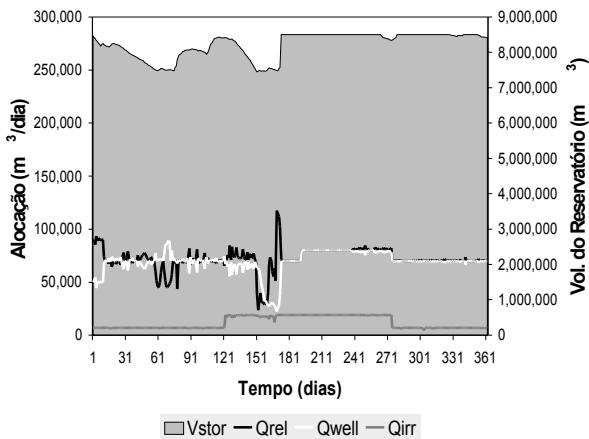


Figura 5(a) Volume do reservatório e alocações para 1996 usando PQS.

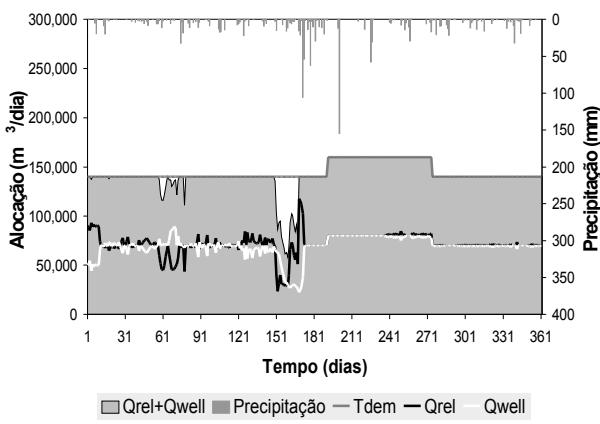


Figura 5(b) Abastecimento em 1996 usando PQS.

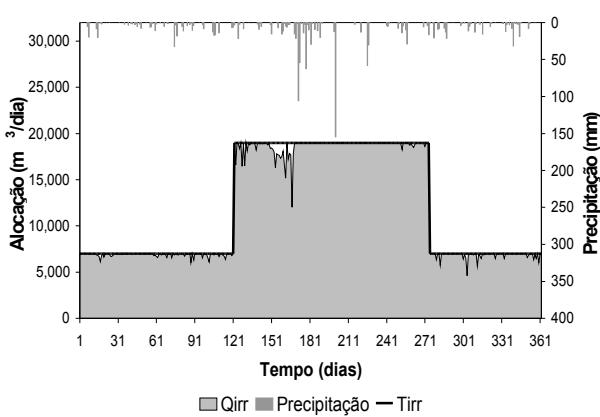


Figura 5(c) Irrigação em 1996 usando PQS.

Com o AG, várias simulações foram realizadas. A Figura 6 ilustra uma delas usando o mesmo cenário da simulação com PQS. Resultados para a operação com os

dados do ano de 1995 e com um volume inicial igual ao volume observado são mostrados na Figura 7. O tamanho da população e o número máximo de gerações foram iguais a 100 e 300, respectivamente.

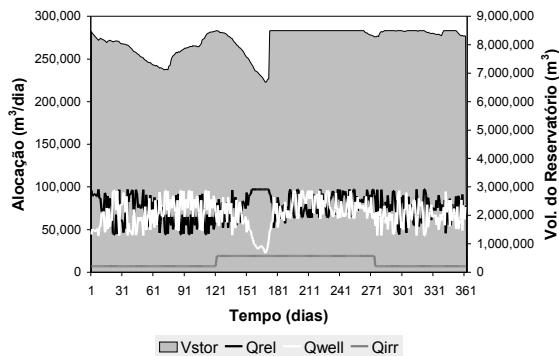


Figura 6(a) Volume do reservatório e alocações para 1996 usando o AG.

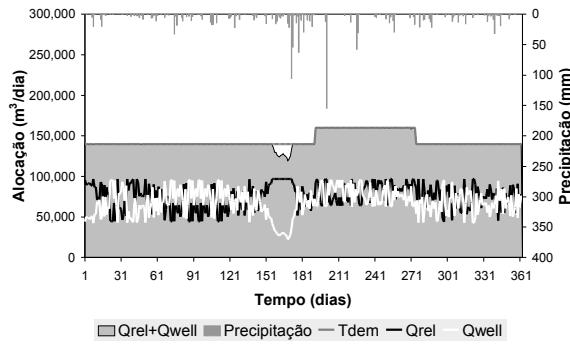


Figura 6(b) Abastecimento em 1996 usando o AG.

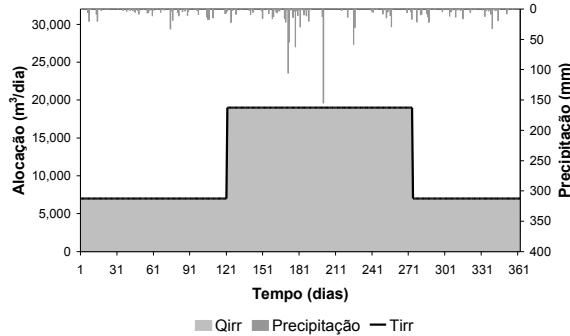


Figura 6(c) Irrigação em 1996 usando o AG.

De acordo com os resultados apresentados pode-se observar que o modelo usando o algoritmo genético produziu melhores soluções em relação ao resolvido pelo procedimento baseado em cálculo. Analisando os resultados obtidos para o mesmo cenário (Figuras 5 e 6) é notado que o método de PQS “decidiu” não liberar água

suficiente para atender as demandas durante certos dias (por exemplo, entre os dias 58 a 78 no caso de abastecimento urbano) enquanto que o AG mostrou que as demandas ainda poderiam ser satisfeitas nestes dias sem comprometer o sistema. Isso mostra que o algoritmo genético encontrou valores de função objetivo menores do que os encontrados pela PQS durante esses períodos. Para a mesma operação em 1996 usando o AG (Figura 6) as demandas para abastecimento não foram atendidas durante os dias 155 e 168, aproximadamente, principalmente porque a água disponível nos poços nesses dias era muito pouca. Embora a barragem tivesse água para liberar, isso não poderia ser feito pois a quantidade necessária superaria a capacidade das estações de tratamento.

Em 1995 (Figura 7), as demandas para abastecimento não foram atendidas para poucos dias em abril porque o volume do reservatório era baixo o que não permitia que a liberação da barragem fosse alta.

Tabela 1 – Implementação das variáveis.

Q_{rel}^1	...	Q_{rel}^5
X_1	...	X_5
Q_{well}^1	...	Q_{well}^5
X_6	...	X_{10}
Q_{irr}^1	...	Q_{irr}^5
X_{11}	...	X_{15}

Tabela 2 – Demandas e volume alvo na barragem.

Demandas de abastecimento	Jul-Set: 160,000 m ³ /dia Out-Jun: 140,000 m ³ /dia
Demandas de irrigação	Mai-Set: 19,000 m ³ /dia Out-Abr: 7,000 m ³ /dia
Volume alvo do reservatório	7,500,000 m ³

CONCLUSÕES

Modelos de otimização podem ajudar os planejadores de recursos hídricos na busca de alternativas de operação que auxiliem na tomada de decisão, especialmente em sistemas complexos que envolvem um grande número de variáveis e restrições.

Existem muitas aplicações de algoritmos genéticos na área de recursos hídricos mas apenas poucas delas são relacionadas à operação ótima de reservatórios. Para essa finalidade, programação linear e dinâmica são predominantes.

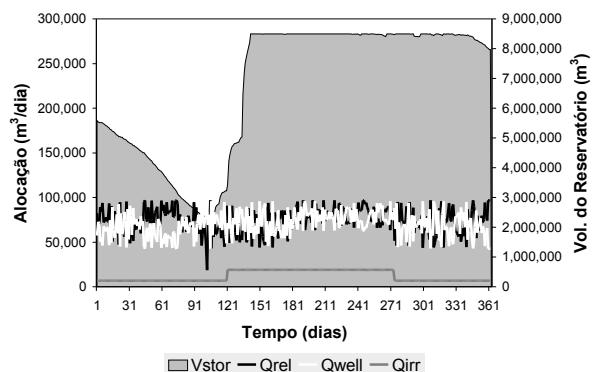


Figura 7(a) Volume do reservatório e alocações para 1995 usando o AG.

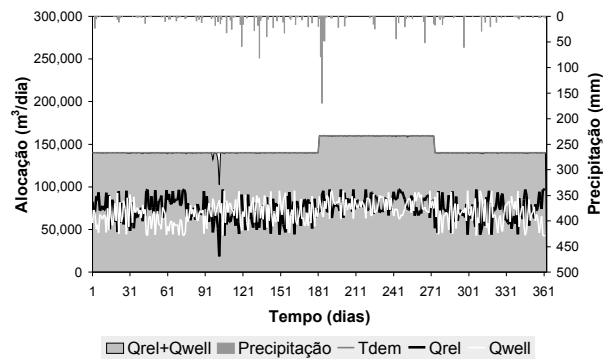


Figura 7(b) Abastecimento em 1995 usando o AG.

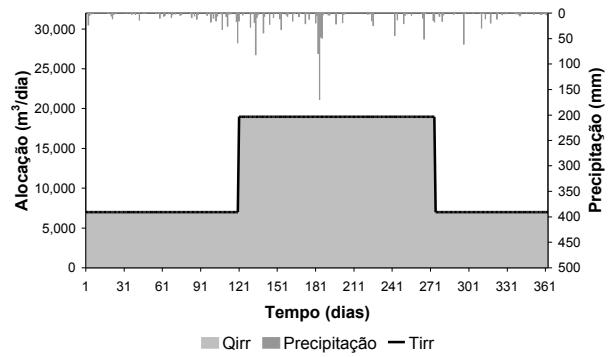


Figura 7(c) Irrigação em 1995 usando o AG.

Neste trabalho, um algoritmo genético foi desenvolvido e aplicado na resolução de um modelo de otimização para a operação ótima de um sistema de recursos hídricos em tempo real. O AG se mostrou mais eficiente do

que uma técnica de otimização tradicional baseada em cálculo em vista dos melhores resultados obtidos através das simulações.

Os resultados apresentados sugerem que algoritmos genéticos são técnicas robustas e eficazes também para problemas de operação de reservatórios, embora o caso analisado seja relativamente simples. Problemas de larga escala para operação e dimensionamento de reservatórios com maior número de variáveis envolvidas (e.g., Santana e Lanna, 2000), os quais têm sido resolvidos geralmente através de técnicas clássicas, são exemplos de casos onde uma abordagem por algoritmos genéticos seria interessante.

REFERÊNCIAS

- BAKER, J. E. (1987). Reducing bias and inefficiency in the selection algorithm. In *Proceedings of the Second International Conference on Genetic Algorithms*, Lawrence Erlbaum Associates, 14-21.
- BEASLEY, D., BULL, D. R. e MARTIN, R. R. (1993a). An overview of genetic algorithms: Part 1, fundamentals. *University Computing*, 15(2): 58-69.
- BEASLEY, D., BULL, D. R. e MARTIN, R. R. (1993b). An overview of genetic algorithms: Part 2, research topics. *University Computing*, 15(4): 170-181.
- BELAINEH, G., PERALTA, R. C. e HUGHES T. C. (1999). Simulation/optimization modeling for water resources management. *J. Water Resour. Plng. and Mgmt.*, 125(3): 154-161.
- CELESTE, A. B. (2002). Optimal real-time operation of a multipurpose water resource system using genetic algorithms. Tese de Mestrado, Department of Civil and Environmental Engineering, Ehime University, Matsuyama, Ehime, Japão.
- CELESTE, A. B., SUZUKI, K., WATANABE, M. e KADOTA, A. (2002). Optimal real-time reservoir operation using genetic algorithms. *8th JSCE Congress of Civil Engineering - Shikoku Division*, 197-198, Tokushima, Japão, Maio.
- CELESTE, A. B., CURÍ, R. C., e CURÍ, W. F. (1996). Um modelo de operação ótima de reservatórios sujeitos a múltiplos usos. *III Simpósio de Recursos Hídricos do Nordeste*. Vol. 1, 39-45, Salvador, Brasil, Dezembro.
- GEN, M., e CHENG, R. (1997). *Genetic algorithms & engineering design*. Wiley-Interscience, New York.
- GEN, M., e CHENG, R. (1997). *Genetic algorithms & engineering design*. Wiley-Interscience, New York.
- GOLDBERG, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Addison Wesley, Reading, Massachusetts.
- HOUCK, C., JOINES, J. e KAY, M. (1995). A genetic algorithm for function optimization: a Matlab implementation. NCSU-IE Technical Report 95-09, North Carolina State University.
- MICHALEWICZ, Z. (1996). *Genetic algorithms + data structures = evolution programs*. 3rd edition, Springer-Verlag, New York.
- MICHALEWICZ, Z. e FOGEL, D. B. (2000). *How to solve it: modern heuristics*. Springer-Verlag, New York.
- MUJUMDAR, P. P. e RAMESH, T. S. V. (1997). Real-time reservoir operation for irrigation. *Water Resour. Res.*, 33(5): 1157-1164.
- Optimization Toolbox user's guide - version 5*. (1996). The MathWorks Inc., Natick, Massachusetts.
- SANTANA, A. G. de e LANNA, A. E. L. (2000). Dimensionamento ótimo de um sistema de recursos hídricos de grande porte: o projeto arco-íris, nordeste do Brasil. *Rev. Bras. Rec. Híd.*, 5(4).
- WARDLAW, R. e SHARIF, M. (1999). Evaluation of genetic algorithms for optimal reservoir system operation. *J. Water Resour. Plng. and Mgmt.*, 125(1): 25-33.
- YEH, W. W-G. (1985). Reservoir management and operations models: a state-of-the-art review. *Water Resour. Res.*, 21(12): 1797-1818.

Genetic Algorithms For Real Time Optimum Operation of Water Resources Systems

ABSTRACT

A mathematical model for optimization based on genetic algorithms was developed and applied to determine real-time optimum operation of a water resources system subject to multiple uses and consisting of a reservoir and a number of wells. The system supplies the city of Matsuyama in Japan. The purpose of the study is to find the optimum daily allocation of water from the dams and the wells for supply and irrigation according to rainfall forecasts within a given horizon of operations. For purposes of comparison, the problem is also solved by a traditional optimization technique, but the results show that the genetic algorithm finds better solutions.

Key words: Reservoir operation, optimization, genetic algorithms.