LI Simpósio Brasileiro de Pesquisa Operacional
Limeira–SP, 02 a 06 de setembro de 2019

Uma metaheurística *iterated local search* baseada em simulação para o escalonamento de bombas hidráulicas em redes de distribuição de água

Tiago C. Oliveira, Willian C. S. Martinho e Rafael A. Melo
Computational Intelligence and Optimization Research Lab (CInO)
Departamento de Ciência da Computação, Universidade Federal da Bahia
Av. Adhemar de Barros, s/n, Salvador, BA 40170-110, Brasil
tiago.conceicao@ufba.br, willianmartinho@gmail.com e melo@dcc.ufba.br

RESUMO

Em uma rede de distribuição de água, a utilização de energia elétrica para o funcionamento das bombas hidráulicas pode aproximar-se de 90% do total consumido nesta. Uma vez que o valor cobrado pelo consumo de energia elétrica pode variar com o horário, o escalonamento das bombas hidráulicas em horários oportunos pode diminuir de forma significativa o custo com energia. Assim, o problema de escalonamento de bombas hidráulicas em redes de distribuição de água consiste em obter o menor custo monetário operacional, garantindo que a água seja entregue a todos os pontos de demanda, respeitando as restrições físicas da rede. Este trabalho propõe métodos heurísticos baseados em simulações, combinando-os em uma metaheurística *iterated local search* baseada em simulação. Experimentos computacionais preliminares mostram que a abordagem proposta é bastante promissora e competitiva com os métodos do estado-da-arte, alcançando resultados muito próximos dos melhores conhecidos em tempo de execução bastante reduzido.

PALAVRAS CHAVE. ***Iterated local search*; Escalonamento de bombas hidráulicas; Redes de distribuição de água.**

Tópicos (MH - Metaheurística; OC – Otimização Combinatória; SIM – Simulação; OA – Outras aplicações em PO)

ABSTRACT

In a water distribution network, the use of electrical energy for pumps operation can reach 90% of the total consumed in the network. The amount charged for the electric power consumption can vary along the day, thus an adequate pump scheduling can significantly reduce the cost with energy. The pump scheduling problem in water distribution networks consists in obtaining the lowest operating monetary cost, ensuring that water is delivered to all demand points without violating physical constraints of the network. This work proposes simulation based heuristic methods, combining them into a simulation based iterated local search metaheuristic. Preliminary computational experiments show that the proposed approach is very promising and competitive with state-of-the-art methods, achieving solutions which are very close to the best ones available in the literature while requiring reduced computational times.

KEYWORDS. Iterated local search; pumping scheduling; Water distribution network.

Paper topics (MH - Metaheuristic; OC – Combinatorial optimization; SIM – Simulation; OA – Others issues in PO)

https://proceedings.science/p/106921?lang=pt-br

1. Introdução

O fornecimento de água adequado é uma necessidade básica do ser humano e da sociedade moderna. Um sistema de distribuição de água (WDN, do inglês *water distribution network*) consiste em uma infraestrutura de diferentes elementos (bombas, reservatórios, canos, válvulas, entre outros) conectados afim de transportar água de diferentes nós fontes até os nós de demanda. Em muitos casos, a água é frequentemente bombeada até reservatórios que ficam em pontos elevados antes de ser distribuída. O custo operacional da utilização de bombas hidráulicas em uma rede de distribuição de água representa uma fração significativa do gasto total gerado na gestão operacional da rede. As bombas consomem grandes quantidades de energia elétrica para produzir fluxo da fonte até os tanques de armazenamento e nós de demanda. Elas também precisam ser reparadas e substituídas, resultando em custos de manutenção [López-Ibáñez et al., 2008]. Nesse sentido, o problema de escalonamento de bombas hidráulicas em redes de distribuição de água consiste em escalonar as bombas de forma a minimizar o custo monetário operacional, garantindo que água seja entregue a todos os pontos de demanda em uma rede de distribuição de água, respeitando as restrições hidráulicas da rede [Lansey e Awumah, 1994].

Em geral, o custo sobre a operação de uma bomba não pode ser facilmente quantificado [López-Ibáñez et al., 2008]. Uma maneira para estimar esse custo é definir o escalonamento como uma configuração que indica quando uma bomba deve ligar ou desligar por um determinado intervalo de tempo. Esse escalonamento deve respeitar um número máximo de comutações, sendo que, uma comutação frequente (liga/desliga) provoca desgaste das bombas e, consequentemente, custos futuros de manutenção. No entanto, encontrar escalonamentos que minimizam o custo operacional não é uma tarefa fácil, pois os requisitos de operação de uma bomba estão sujeitos a tarifas elétricas e demandas variáveis. Adicionalmente, devem condicionar pressão adequada a rede afim de fornecer água aos consumidores e proporcionar a recuperação total dos níveis dos tanques durante um ciclo operacional. Além disso, o problema é limitado por restrições hidráulicas não-lineares o que torna o NP-Difícil.

Nos últimos anos, uma quantidade significativa de trabalhos foi dedicada a otimização do escalonamento de bombas hidráulicas em rede de distribuição de água. Algoritmos exatos foram desenvolvidos usando métodos de programação linear [Pasha e Lansey, 2009] e programação não-linear [Dai et al., 2017]. Entretanto, sua aplicação é limitada ao tamanho do espaço de busca de soluções e do seu tempo de avaliação. Assim, o número de soluções que podem ser examinadas em um tempo satisfatório é extremamente restrito. Portanto, a aplicação de métodos tradicionais de otimização, tais como programação linear e não-linear, frequentemente envolve a formulação de modelos simplificados da rede e do sistema de equações hidráulicas para atender aos requisitos do algoritmo [López-Ibáñez et al., 2011].

Em particular, algoritmos evolucionários se mostraram bastante promissores para o problema [Mackle et al., 1995; Savic et al., 1997; Simpson et al., 1999; Boulos et al., 2001; Van Zyl et al., 2004]. A maioria desses trabalhos codificou o escalonamento como uma cadeia binária que representa o estado ligado/desligado de uma bomba durante um intervalo de tempo em que foram utilizados operadores específicos de recombinação e mutação. Esse tipo de abordagem é baseada em complicadas estruturas metaheurísticas inspiradas na natureza e sem muita conexão com os fundamentos do problema. Assim como criticado por De Corte e Sörensen [2013] na otimização do projeto de redes de distribuição de água, mas que também é valido para o problema de escalonamento de bombas, isso leva a implementações caixa preta que geralmente são complicadas, tornando os métodos mais difíceis de serem implementados e que sofrem com a falta de robustez em relação a diferentes conjuntos de dados de entrada.

https://proceedings.science/p/106921?lang=pt-br

Outras metaheurísticas também foram usadas, tais como *simulated annealing* [Goldman e Mays, 1999], *particle swarm optimization* [Wegley et al., 2000] e *ant colony optimization* [López-Ibáñez et al., 2008]. Essas abordagens apresentaram resultados promissores e foram capazes de superar algumas das limitações evidenciadas pelos métodos exatos. Porém, algumas das técnicas apresentam dificuldades ao lidar com o acréscimo de restrições e até mesmo estender o problema com características multiobjetivo.

O presente trabalho propõe uma metaheurística *iterated local search* (ILS) baseada em simulação para o problema de escalonamento de bombas hidráulicas em redes de distribuição de água. A abordagem utiliza-se de um simulador hidráulico para validar as soluções geradas pelo algoritmo, o que faz com que a complexidade do problema não afete consideravelmente o desempenho do processo de otimização. O algoritmo é simples em seu conceito e prática, efetivo e modular, permitindo melhorias em cada uma das partes separadamente.

O restante do artigo está organizado da seguinte forma. Na Seção 2, o problema do escalonamento de bombas hidráulicas em redes de distribuição de água é formalizado. A metaheurística *iterated local search* baseada em simulação é descrita na Seção 3. Resultados preliminares da abordagem proposta considerando instâncias *benchmark* utilizadas na literatura são sumarizados na Seção 4. Comentários finais são discutidos na Seção 5.

2. O problema do escalonamento de bombas hidráulicas em redes de distribuição de água

Dada uma rede de distribuição de água onde os padrões de demanda, níveis dos tanques e tarifas elétricas são especificados, o problema consiste em encontrar a melhor configuração para o escalonamento das bombas em um ciclo operacional, minimizando os custos operacionais totais e garantindo o serviço de distribuição adequado [López-Ibáñez et al., 2008]. Os custos de energia de uma bomba dependem do preço da energia no momento do bombeamento e da quantidade de energia consumida. O preço por unidade de energia é determinado pelas tarifas de eletricidade. Enquanto a quantidade real de energia consumida depende de vários parâmetros, como o fluxo, a energia hidráulica fornecida e sua eficiência. As restrições de um sistema de distribuição de água permitem a viabilidade do bombeamento, o que inclui as demandas atendidas em pressões adequadas e a recuperação da água fornecida pelos tanques ao final de um período de operação. Neste trabalho são tratadas as restrições de nível de água no tanque e comutações de estado da bomba, respectivamente.

Dessa forma, dada uma bomba p, o valor $C_{\mathrm{E}}(p)$ (em \$) a ser pago pela sua utilização é dado pela soma de duas parcelas. A primeira consiste na tarifa de pico em que deve-se avaliar $E_{\mathrm{d}}(p)$ (em kWh) como máximo de energia elétrica que esta pode utilizar e a tarifa de pico do consumo C_{d} (em $\$/kW$). Assim, com C_{d} e $E_{\mathrm{d}}(p)$ é possível obter o valor a ser pago por pico de consumo. A segunda parcela consiste no consumo pelo intervalo de tempo que é dado $E_{\mathrm{c}}(p,t)$ deve-se avaliar o consumo da bomba p no intervalo t, em caso de operação da bomba sempre em ponto de eficiência (razão entre a energia de entrada e energia de saída) $E_{\mathrm{c}}(p,t)$ permanece constante para todos os intervalos de tempo (N^T). A tarifa cobrada pelo consumo no intervalo de tempo t é dada por $C_{\mathrm{c}}(t)$, a qual pode ser variável, gerando assim tarifas onerosas em algumas horas do dia. E por fim, o intervalo de tempo em que a bomba permaneceu ligada $S(p,t)$ (em h), o qual é comumente dado em intervalos de uma hora. Assim, com $C_{\mathrm{c}}(t)$, $E_{\mathrm{c}}(p,t)$ e $S(p,t)$ é possível obter o valor a ser pago pelo consumo por intervalo de tempo de acordo a equação (1).

$$C_{\mathrm{E}}(p) = C_{\mathrm{d}}E_{\mathrm{d}}(p) + \sum_{t=0}^{N^T} C_{\mathrm{c}}(t)E_{\mathrm{c}}(p,t)S(p,t). \tag{1}$$

A quantidade de energia consumida $E_{\mathrm{c}}(p,t)$ por uma bomba p depende da sua vazão $Q(p,t)$ (em m^3/s), a energia hidráulica entrega ao sistema $h(p,t)$ (em m), a eficiência a qual opera $e(p,t)$ (em

https://proceedings.science/p/106921?lang=pt-br

$\%$) e a densidade do líquido bombeado γ (em kg/m^3) a qual é constante, como descrita na equação (2). Dessa forma, escalonamentos das bombas em ponto de eficiência podem ser aplicados para melhor uso energético.

$$E_{\mathrm{c}}(p,t) = \frac{10^{-3}\gamma Q(p,t)h(p,t)}{e(p,t)}. \tag{2}$$

2.1. Restrições do problema do escalonamento de bombas hidráulicas

Para se obter configurações de escalonamento viáveis, as soluções geradas pelo método proposto deve satisfazer as restrições do sistema que representam seus critérios de desempenho. Estes incluem restrições hidráulicas condicionadas as leis de conservação de massa e energia, limites mínimos e máximos nos níveis de armazenamento dos tanques, requisitos mínimos de pressão nos nós de demanda e um equilíbrio entre a oferta e a demanda dos tanques. A lei de conservação de massa deve ser satisfeita para cada componente da rede de distribuição de água. Sendo o fluxo o produto da massa e a taxa de tempo, esta lei estabelece que o volume de água que flui para um nó da rede, por unidade de tempo, deve ser igual ao volume de água para fora deste nó. Depreende-se então, que o fluxo varia conforme velocidade. A lei de conservação de energia deve ser satisfeita para cada ciclo fechado. Esta lei estabelece que a soma das quedas de pressão em um ciclo fechado é zero. As quedas de pressão nos sistemas de tubulação são causadas por cisalhamento na área de contato interna dos tubos e fricção com componentes da tubulação, como junções e válvulas. O simulador hidráulico usado, EPANET [Rossman, 1999], abstrai a complexidade das redes mantendo acurácia próxima de uma rede real, para que as soluções verificadas nesse sejam utilizadas nas redes reais sem ajustes extensivos. Grande parte dos trabalhos que precisam verificar as restrições físicas das redes usam algum tipo de simulador hidráulico, desta forma os cálculos que definem as restrições físicas apresentadas não foram elencados, mas podem ser encontrados em De Corte e Sörensen [2013]. Neste trabalho o EPANET será utilizado para verificar a viabilidade das soluções geradas, o custo monetário e quantidade de comutações.

Em López-Ibáñez [2009] são apresentadas formulações para as restrições de demanda e consumo nos tanques. O deficit do volume em um tanque k (ΔV_{k}) é definido pela diferença, em porcentagem, do volume final $V_{\mathrm{k,E}}$ pelo volume inicial $V_{\mathrm{k,S}}$, conforme a equação (3). Um deficit de volume negativo significa que há volume de água excedente no tanque, o qual não compensa a perda de água em outro tanque. O volume de tolerância de deficit é o parâmetro que define qual valor (ΔV^{tol}) é permitido para cada tanque, como a equação (4). Só valores maiores que ΔV^{tol} são acumulados para calcular o deficit total ΔV em todos os N^{TK} tanques da rede, porém para uma solução viável esse valor deve ser zero, como a equação (5).

$$\Delta V_{\mathrm{k}} = 100\frac{V_{\mathrm{k,S}} - V_{\mathrm{k,E}}}{V_{\mathrm{k,S}}}. \tag{3}$$

$$\Delta V'_{\mathrm{k}} = \begin{cases} \Delta V_{\mathrm{k}}, & se\ \Delta V_{\mathrm{k}} > \Delta V^{\mathrm{tol}}, \\ 0, & \text{caso contrário.} \end{cases} \tag{4}$$

$$\Delta V = \sum_{k=1}^{N^{\mathrm{t}}} \Delta V'_{\mathrm{k}} = 0. \tag{5}$$

A pressão em um nó de demanda k deve ser maior que a pressão mínima H_k^{min} requerida para que haja fluxo neste, a qual deve ser satisfeita durante um período t, obtendo assim $H_{\mathrm{k,t}}$, como a

inequação (6). A violação da restrição (6) é quantificada no deficit de pressão, como a equação (7) e agregada as outras em um único valor, como a equação (8), onde N^d é o número de nós de demanda e W_k é o peso relativo à importância da pressão em k. Assim como em López-Ibáñez [2009], a importância é considerada como única, sendo $W_k = 1, \forall k = 1...N^d$.

$$H_{\mathrm{k,t}} \geq H_k^{min}. \tag{6}$$

$$\Delta H_{\mathrm{k,t}} = \begin{cases} \frac{H_k^{min} - H_{\mathrm{k,t}}}{H_k^{min}}, & se\ H_{\mathrm{k,t}} < H_k^{min}, \\ 0, & \text{caso contrário.} \end{cases} \tag{7}$$

$$\Delta H = \sum_{t=1}^{N^T} \sum_{k=1}^{N^d} W_k \Delta H_{\mathrm{k,t}} = 0. \tag{8}$$

A comutação do estado de bomba equivale a ação de ligar uma bomba que não estava operando durante um intervalo de tempo anterior. O problema de reduzir os custos com bombeamento é, na maior parte das abordagens, atrelado ao problema com manutenção das bombas, neste caso tratado como a comutação do estado da bomba, a qual aumenta o seu desgaste [Lansey e Awumah, 1994]. A comutação do estado de bombas é comumente tratada na literatura como restrição ou como objetivo, em abordagem multiobjetivo. Em López-Ibáñez [2009] a comutação do estado de bombas é tratada como restrição, o qual define o número máximo de comutações de cada bomba (N_p^{sw}), e N^{sw} a soma das comutações de todas as bombas como na equação (9).

$$N^{sw} = \sum_{p=1}^{N^P} N_p^{sw}. \tag{9}$$

Limitar a comutação por cada bomba é uma alternativa a limitar o numero total de comutações em toda a rede de distribuição, assim evita que uma única bomba sofra várias comutações como na inequação (10). Tem-se então, a constante SW, que é o número máximo de comutações de estado de uma bomba em um período de escalonamento.

$$N_p^{sw} \leq SW, \quad \forall p \in \{1, ..., N^P\}. \tag{10}$$

2.2. Representação do escalonamento de bombas hidráulicas

Uma solução S equivale a uma representação binária que divide o horizonte de tempo em T intervalos menores de tempo e atribui o escalonamento das bombas a uma *string* de bits de tamanho N^T. Desta forma cada bit representa um estado da bomba (desligado/ligado) em um determinado intervalo de tempo como na Figura 1.

t_0	t_1	t_j	t_{N^T}
0/1	0/1	...	0/1

Figure 1: A figura mostra a representação binária para uma bomba p e N^T intervalos para um ciclo de operação. Para cada intervalo t_j, um bit $0/1$ (desligado/ligado) define o estado de operação da bomba. Imagem baseada no trabalho de López-Ibáñez [2009].

https://proceedings.science/p/106921?lang=pt-br

A partir de um escalonamento, o número máximo de comutação é dado pela quantidade de alternância entre 0 e 1 mais 1 se o escalonamento inicia com 1 e termina com 0. Desta forma, o número máximo de comutações é $N_p^{SW} \leq \lfloor N^T/2 \rfloor$

3. Metaheurística *iterated local search* baseada em simulação

Nesta seção é apresentada a metaheurística *iterated local search* baseada em simulação para o escalonamento de bombas hidráulicas em redes de distribuição de água. O algoritmo combina uma abordagem *iterated local search* com o simulador hídrico EPANET. O procedimento de validação de cada solução candidata encontrada é descrito na Subseção 3.1. A heurística *multi-start* geradora de soluções iniciais é apresentada na Subseção 3.2. As descrições das vizinhanças e dos métodos de busca local são explanadas na Subseção 3.4. A Subseção 3.5 descreve o funcionamento da perturbação das soluções. Por fim, a Subseção 3.6 descreve o pseudocódigo da metaheurística *iterated local search* baseada em simulação.

3.1. Processo de validação de solução

A verificação das soluções geradas pela metaheurística é realizada através do simulador EPANET. O processo de verificação é feito sempre que há uma nova solução, seja gerada pelos métodos de geração inicial, pelo método de perturbação ou pela busca local. A verificação do EPANET é realizada iterativamente, desta forma é possível obter informações dos componentes da rede a cada passo da execução, podendo ser interrompida assim que houver alguma irregularidade na simulação. Os possíveis estados de execução foram elencados por Rossman [1999] e serão utilizadas neste trabalho para quantificar as soluções.

Dentre os retornos fornecidos pelo EPANET estão o custo da energia consumida, a comutação de estado das bombas, os níveis de tanques e as pressões nos nós. Os resultados obtidos pelo EPANET são reutilizados pelo algoritmo para gerar novas soluções. A entrada para a abordagem é composta por: parâmetros para o algoritmo, proposta de demanda e dados sobre os componentes da rede hídrica, tais como bombas, canos, reservatórios e suas restrições.

3.2. Geração de solução inicial

O método de geração de solução inicial utiliza a representação explícita binária (Seção 2.2), gerada aleatoriamente e priorizando intervalos contíguos, não ultrapassando o número máximo de comutações. A solução inicial é composta por no máximo três intervalos, os quais são gerados conforme o Algoritmo 1, tem como entrada o máximo de iterações do *multi-start*, dado por ϕ, bem como valores para aumentar ou diminuir a probabilidade de ligar bombas hidráulicas em períodos de energia elétrica mais onerosos, definidos por α e β.

Algoritmo 1: Multi-start(ϕ, α, β)

```
1  f(s0') ← ∞;
2  y ← 0;
3  p ← 0;
4  while i < φ do
5      while p < N^P do
6          k ← 2 * aleatorio(0, 3);
7          while j < k do
8              if oneroso(y) then
9                  l ← 2βN^T/k;
10             else
11                 l ← 2αN^T/k;
12             y ← y + l;
13             p ← p ∪ aleatorio(1, l);
14             j ← j + 1;
15         s0 ← s0 ∪ p;
16         p ← p + 1;
17     w ← EPANET(s0);
18     if w = 0 then
19         i ← i + 1;
20         if f(s0) < f(s0') then
21             s0' ← s0;
22     s0 ← ∅;
23 retorne(s0');
```

Na linha 1 o custo da solução inicial é inicializada com valor infinito, para servir de comparação. Na linha 4 inicia o método *Multi-start*, o qual será iterado ϕ vezes, será realizado para todas as p bombas (linha 5) e retornará o menor valor gerado de $s0$. Na linha 6 é gerada a quantidade de comutações (é dada em pares, onde inicia e onde finda o intervalo de bomba ligada). Nas linhas 8, 9, 10 e 11 é verificado se o período é mais oneroso, caso seja, a probabilidade de ligar a bomba será mais baixa. Na linha 16 uma parte do intervalo que a bomba hidráulica ficará ligada será adicionado à bomba p, o tamanho desta parte de intervalo é aleatório e este processo continua até o fim do período. Na linha 17 é realizada a simulação e na linha 18 é verificado se houve erro ao simular. Não havendo erro na simulação, na linha 21, se o custo da solução $s0$ for menor que o custo da solução $s0'$, esta será atualizada para melhor solução até o momento. Na linha 23 o algoritmo retorna a melhor solução encontrada.

3.3. Vizinhanças

Neste trabalho duas vizinhanças são consideradas: a) diferença de um elemento extremidade (Figura 2); e b) diferença de um elemento interno (Figura 3).

3.3.1. Vizinhança com diferença de um elemento - *flip* extremidade

Dada uma solução s seu vizinho é definido como a diferença entre um elemento mais extremidade à sequência de períodos ligados. Assim, o movimento possível para alcançar um vizinho é desligando uma bomba p em um intervalo de tempo t_j, onde $0 \leq t_j \leq t_{N^T}$ e ($t_{j-1} = 0$ ou $t_{j+1} = 0$), como pode ser visto na Figura 2. O conjunto de vizinho de s é dado por $N_1(s)$.

https://proceedings.science/p/106921?lang=pt-br

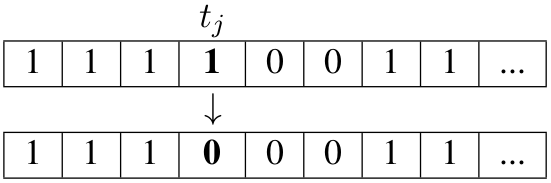

Figure 2: Estrutura de vizinhança com diferença de um elemento - *flip* extremidade.

3.3.2. Vizinhança com diferença de um elemento interno - *flip* interno

Dada uma solução s seu vizinho é definido como a diferença de um elemento contido na sequência de períodos ligados, que não sejam os períodos das extremidades da sequência (complementa a vizinhança *flip* extremidade). Assim, o movimento possível para alcançar um vizinho é desligando uma bomba p em um intervalo de tempo t_j, onde $0 \leq t_j \leq t_{N^T}$ e $(t_{j-1} = 1$ e $t_{j+1} = 1)$, como pode ser visto na Figura 3. O conjunto de vizinho de s é dado por $N_2(s)$.

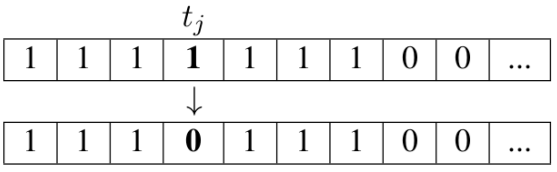

Figure 3: Estrutura de vizinhança *flip* interno.

3.4. Buscas locais

Para a metaheurística *iterated local search* baseada em simulação a busca é dividida em duas partes, *flip* extremidade e *flip* interno, detalhadas na Seção 3.3. Ambas possuem o mesmo critério de execução, como pode ser visto no Algoritmo 2: seleciona um vizinho a partir de uma solução corrente, realiza a simulação no EPANET e avalia o custo, se o custo da solução vizinha for menor que a da solução corrente, atualiza a solução corrente e atualiza a vizinhança. Este processo é repetido até que não haja mais vizinhos. O Algoritmo 2 recebe uma solução corrente s e o tipo k de vizinhança que irá explorar, sendo, assim, utilizado por ambas as vizinhanças.

Algoritmo 2: Busca-local(s, k)

```
1 while N_k(s) ≠ ∅ do
2     selecione randomicamente s' ∈ N_k(s);
3     w ← EPANET(s');
4     if w = 0 then
5         if f(s)' < f(s) then
6             s ← s';
7 retorne(s);
```

3.5. Perturbação

A perturbação é usada para diversificação, isto é, para escapar de um ótimo local. Uma perturbação é aplicada à solução para explorar outras vizinhanças. A perturbação modifica a solução conforme o grau ϵ que indica quantos períodos t serão modificados nas bombas, com pode ser visto na Figura 4. Os períodos e bombas modificados pela pertubação são escolhidos de maneira aleatória.

https://proceedings.science/p/106921?lang=pt-br

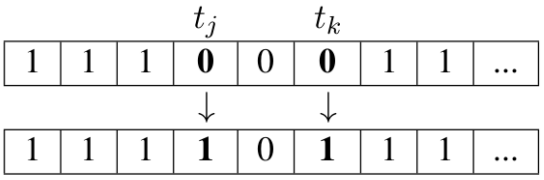

Figure 4: Estrutura de perturbação com $\epsilon = 2$.

A perturbação gera soluções respeitando a quantidade de comutações, conforme a Inequação 10.

Algoritmo 3: Perturbacao(s, ϵ)

```
1 i ← 0;
2 while i < ε do
3 |   p ← aleatorio(0, N^P);
4 |   t ← aleatorio-desligado(p);
5 |   if SW(p) ≤ 3 then
6 |   |   s ← ligar(p, t);
7 |   |   i ← i + 1;
8 retorne(s);
```

O Algoritmo 3 mostra como funciona a perturbação, o qual recebe uma solução s e o fator de perturbação ϵ. Na linha 3 é sorteada uma bomba hidráulica p aleatoriamente entre todas as bombas da rede de distribuição de água, em seguida, na linha 4 é sorteado um período t aleatoriamente entre os períodos que a bomba p está desligada. Na linha 5 é verificado se quantidade máxima de comutações não foi atingida. Na linha 7 será ligada e adicionada à s. Este processo repete-se até que a quantidade ϵ de tempo t seja ligada para a solução s. Por fim, na linha 9 é retornada a solução s perturbada.

3.6. *Iterated local search* baseada em simulação

A construção deste algoritmo foi baseada nos conceitos do *Iterated Local Search* (ILS). O ILS é uma metaheurística simples em seu conceito e prática, efetiva e de propósito geral (Lourenço et al. Lourenço et al. [2003]). Sua modularidade (solução inicial, busca local, pertubação e critério de aceitação) permite melhorias em cada uma das partes separadamente, assim como novas adequações sem modificações extensivas, como pode ser visto no Algoritmo 4.

Algoritmo 4: ILS-sim(Max-it)

1 $s_0 \leftarrow$ *Multi-start*(ϕ, α, β);
2 $s* \leftarrow s_0$;
3 $i \leftarrow 0$;
4 **while** $i <$ *Max-it* **do**
5 $s^{'} \leftarrow$ Busca-local$(s_0, 1)$;
6 $s^{"} \leftarrow$ Busca-Local$(s^{'}, 2)$;
7 **if** $f(s^{"}) < f(s^{*})$ **then**
8 $s^{*} \leftarrow s^{"}$;
9 **else**
10 $s' \leftarrow$ Perturbação$(s^{"}, \epsilon)$;
11 $s_0 \leftarrow s^{'}$;
12 $i \leftarrow i + 1$;

O Algoritmo 4, ILS-sim, na linha 1 gera ϕ soluções iniciais e utiliza a que possui menor valor para iniciar a busca local. A busca local consiste em duas partes, sendo: vizinhança *flip* extremidade; e vizinhança *flip* interno, conforme linhas 3 e 4. Caso a solução encontrada nesta iteração seja melhor que s^* (melhor solução até o momento), esta será atualizada, caso contrário realiza a pertubação, linha 7. A pertubação é iniciada se não houver melhoria em uma solução após as duas buscas. Esta solução perturbada é usada como entrada para o algoritmo de busca local.

4. Experimentos computacionais preliminares

Nesta seção são apresentadas as redes hídricas utilizadas para os experimentos computacionais e os parâmetros utilizados para execução dos algoritmos. Os resultados obtidos pela abordagem proposta são comparados com os do estado da arte.

4.1. Redes hídricas

As redes hídricas podem ser hipotéticas ou reais. As redes hipotéticas não existem no mundo real e seu propósito é testar o sistema em devidas circunstâncias. Doutro modo, as redes reais são representações das redes existentes. No contexto real as redes são compostas por canos, bombas, tanques e outros elementos, todavia para as simulações estes são abstraídos. Assim, são representados somente pelas informações que os caracterizam, um tanque, por exemplo, pode ser representado pela sua altura, raio e distância do chão. Dentre as redes utilizadas nos trabalhos apresentados, Vanzyl e Richmond são as mais usadas, sendo assim parâmetros viáveis para comparação entre as abordagens do estado da arte.

A rede Vanzyl, proposta em Van Zyl et al. [2004], é considerada pequena, todavia possui os componentes suficientes para uma rede complexa. Como pode ser vista na Figura 5 (a), esta é composta por três bombas, dois tanques e uma válvula, a qual previne que a água retorne. Adota como zero o valor para pico de consumo de demanda e inicia a simulação as 07 h, para um período de 24 h. As tarifas de pico são das 07:00 h às 24:00 h e o maior consumo de água, ocorre em dois períodos 07:00 h e 18:00 h.

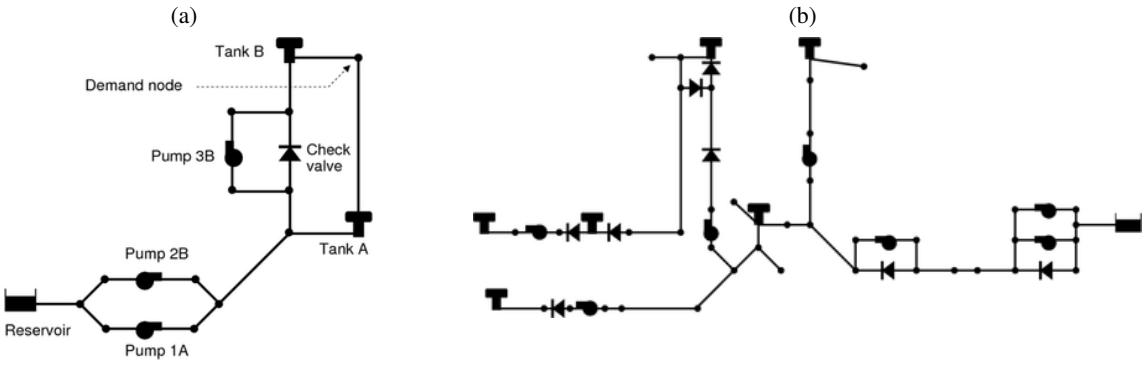

Figure 5: A figura mostra as redes usadas no trabalho. Em (a) a rede Vanzyl [Van Zyl et al., 2004], composta por três bombas, dois tanques e uma válvula, a qual previne que a água retorne. Em (b) a rede Richmond Atkinson et al. [2000] (representação reduzida na imagem), é uma rede real, que fica no Reino Unido. Esta é composta por sete bombas, seis tanques e 948 links e 836 nós de demanda.

A instância Richmond, utilizada pela primeira vez em Atkinson et al. [2000], é uma rede real, que fica no Reino Unido. Esta é composta por sete bombas, seis tanques, um reservatório, 948 links e 836 nós de demanda, como pode ser visto na Figura 5 (b). Assim como Vanzyl, adota como zero o valor para pico de consumo de demanda e as tarifas de pico são das 07:00 h às 24:00 h.

4.2. Resultados computacionais

Os experimentos foram realizados em uma máquina com Ubuntu 18.04.1 LTS kernel versão 4.15.0-46-generic, processador Intel(R) Core(TM) i5-3470 (3.20GHz) e 4GB de memória. Foram realizados com a instância proposta em Van Zyl et al. [2004] (Vanzyl) e a proposta em Atkinson et al. [2000] (Richmond). Essas são as instâncias usadas na maioria dos trabalhos, como em Van Zyl et al. [2004], López-Ibáñez et al. [2005], López-Ibáñez et al. [2008], Manuel [2009], López-Ibáñez et al. [2011], Giacomello et al. [2012], Abdallah e Kapelan [2017]. Para os teste foram realizadas 50 execuções, onde foram extraídos o melhor custo, pior custo e a média dos custos.

4.2.1. Vanzyl

Nas execuções expostas nesta seção utilizou-se os seguintes para a rede Vanzyl: Max-it = 1000, maxit-s0 = 100, $\epsilon = 2$, $\alpha = 0.7$ e $\beta = 0.3$.

Table 1: Resultados preliminares usando o algoritmo proposto.

Instância	Melhor ($)	Média ($)	Maior ($)	Tempo Médio(s)
Vanzyl	316.11	327.84	357.28	78.56
Richmond	91.16	103.14	115.52	456.6

Table 2: Resultados preliminares usando o algoritmo proposto e a correspondência com as soluções do estado da arte Vanzyl.

	ILS-sim	López-Ibáñez et al. [2008]	López-Ibáñez et al. [2005]	Van Zyl et al. [2004]	López-Ibáñez [2009]
Melhor ($)	316.11	326.50	325.00	344.19	315.90
Tempo médio computacional (s)	78.56	417.6	-	-	-
Número de comutações	3	3	~4.29	-	3
Processador	Intel Core i5-3470	Pentium 4 3.20 GHz	-	-	Pentium 4 3.20 GHz

https://proceedings.science/p/106921?lang=pt-br

Table 3: Resultados preliminares usando o ILS-VND (algoritmo proposto) e a correspondência com as soluções do estado da arte - Richmond.

	ILS-sim	López-Ibáñez et al. [2008]	Giacomello et al. [2012]	Van Zyl et al. [2004]	López-Ibáñez [2009]
Melhor ($)/ano	33,215.00	32,581.40	42,606.45	35,296.00	32,010.50
Tempo médio computacional (s)	456.60	8862	5400	75600	-
Número de comutações	3	3	-	-	3
Processador	Intel Core i5-3470	Pentium 4 3.20 GHz	Intel Core 2 Duo 2.2-GHz	Pentium 400 MHz	Pentium 4 3.20 GHz

4.2.2. Richmond

Para a instância Richmond, que é maior, com mais componentes e mais complexa para resolução pelo Epanet utilizou-se os seguintes parâmetros: Max-it = 50, maxit-s0 = 10, $\epsilon = 2$, $\alpha = 0.7$ e $\beta = 0.3$.

Alguns resultados de trabalhos citados na Tabela 2 e na Tabela 3 faltam valores como tempo médio computacional e número de comutações. A falta do valor para tempo médio computacional dá-se pela técnica utilizada, já que esta aplica avaliações de função, assim o tempo não é exposto pelos autores. Em outros casos, os valores faltantes para a quantidade de comutações não é informado no artigo. Vale lembrar também que as configurações de hardware e software são distintas dos trabalhos citados para este.

5. Comentários finais

Este trabalho apresentou uma nova abordagem para o problema de escalonamento de bombas hidráulicas em redes de distribuição de água. A implementação inicial da metaheurística *iterated local search* baseada em simulação mostrou-se promissora quando comparados os resultados preliminares obtidos com os resultados dos métodos estado-da-arte e com tempo de execução computacional bastante reduzido.

Os valores obtidos para a instância Vanzyl mostram que a implementação inicial da metaheurística *iterated local search* baseada em simulação chegou a soluções que desviam em apenas 0,06% do melhor valor conhecido, entretanto com tempo médio de execução muito menor que o dos demais métodos. Para a instância Richmond, a abordagem proposta chegou a soluções que desviam em apenas 3,76% do melhor valor conhecido e com tempo médio de execução menor que dos demais métodos.

O objetivo para obter soluções ainda melhores é a implementação de uma abordagem baseada em busca tabu, a qual auxiliará a busca a evitar soluções já encontradas. Pretende-se também propor novas estruturas de vizinhanças levando em consideração a iteração entre as bombas. Por fim, um esforço será realizado para melhorar a perturbação realizada nas soluções, considerando possíveis abordagens adaptativas.

References

Abdallah, M. e Kapelan, Z. (2017). Iterative extended lexicographic goal programming method for fast and optimal pump scheduling in water distribution networks. *Journal of Water Resources Planning and Management*, 143(11):04017066.

Atkinson, R., Van Zyl, J., Walters, G., e Savic, D. (2000). Genetic algorithm optimisation of level-controlled pumping station operation. *Water network modelling for optimal design and management*, p. 79–90.

Boulos, P., Wu, Z., Orr, C., Moore, M., Hsiung, P., e Thomas, D. (2001). Optimal pump operation of water distribution systems using genetic algorithms. In *AWWA Distribution System Symposium*. Denver, USA: American Water Works Association.

https://proceedings.science/p/106921?lang=pt-br

Dai, P. D., Van Dai, B., et al. (2017). Optimal pump scheduling to pressure management for large-scale water distribution systems. In *International Conference on Advanced Engineering Theory and Applications*, p. 532–541. Springer.

De Corte, A. e Sörensen, K. (2013). Optimisation of gravity-fed water distribution network design: A critical review. *European Journal of Operational Research*, 228(1):1–10.

Giacomello, C., Kapelan, Z., e Nicolini, M. (2012). Fast hybrid optimization method for effective pump scheduling. *Journal of Water Resources Planning and Management*, 139(2):175–183.

Goldman, F. E. e Mays, L. W. (1999). The application of simulated annealing to the optimal operation of water systems. In *WRPMD'99: Preparing for the 21st Century*, p. 1–16. American Society of Civil Engineers.

Lansey, K. E. e Awumah, K. (1994). Optimal pump operations considering pump switches. *Journal of Water Resources Planning and Management*, 120(1):17–35.

López-Ibáñez, M., Prasad, T. D., e Paechter, B. (2011). Representations and evolutionary operators for the scheduling of pump operations in water distribution networks. *Evolutionary computation*, 19(3):429–467.

López-Ibáñez, M. (2009). *Operational optimisation of water distribution networks*. PhD thesis, Edinburgh Napier University.

López-Ibáñez, M., Prasad, T. D., e Paechter, B. (2008). Ant colony optimization for optimal control of pumps in water distribution networks. *Journal of Water Resources Planning and Management*, 134(4):337–346.

López-Ibáñez, M., Prasad, T. D., e Paechter, B. (2005). Optimal pump scheduling: Representation and multiple objectives. In *Proceedings of the Eighth International Conference on Computing and Control for the Water Industry (CCWI 2005)*, volume 1, p. 117–122. Citeseer.

Lourenço, H. R., Martin, O. C., e Stützle, T. (2003). Iterated local search. In *Handbook of metaheuristics*, p. 320–353. Springer.

Mackle, G., Savic, G. A., e Walters, G. A. (1995). Application of genetic algorithms to pump scheduling for water supply. In *Genetic Algorithms in Engineering Systems: Innovations and Applications, 1995. GALESIA. First International Conference on (Conf. Publ. No. 414)*, p. 400–405. IET.

Manuel, L. (2009). Operational optimisation of water distribution networks. *Edinburgh, UK*.

Pasha, M. F. K. e Lansey, K. (2009). Optimal pump scheduling by linear programming. In *World Environmental and Water Resources Congress 2009: Great Rivers*, p. 1–10.

Rossman, L. A. (1999). The EPANET programmer's toolkit for analysis of water distribution systems. In *WRPMD'99: Preparing for the 21st Century*, p. 1–10. American Society of Civil Engineers.

Savic, D. A., Walters, G. A., e Schwab, M. (1997). Multiobjective genetic algorithms for pump scheduling in water supply. In *AISB international workshop on evolutionary computing*, p. 227–235. Springer.

https://proceedings.science/p/106921?lang=pt-br

Simpson, A. R., Sutton, D. C., Keane, D. S., e Sheriff, S. J. (1999). Optimal control of pumping at a water filtration plant using genetic algorithms. *Water industry systems: modeling and optimization applications*, p. 407–415.

Van Zyl, J. E., Savic, D. A., e Walters, G. A. (2004). Operational optimization of water distribution systems using a hybrid genetic algorithm. *Journal of water resources planning and management*, 130(2):160–170.

Wegley, C., Eusuff, M., e Lansey, K. (2000). Determining pump operations using particle swarm optimization. In *Building Partnerships*, p. 1–6. American Society of Civil Engineers.