

Algoritmos genéticos na determinação da trajetória de ondas ultrassônicas para geração de imagens tomográficas em elementos de concreto.

Genetic algorithms in the determination of ultrasonic wave trajectories for tomographic imaging of concrete elements.

Vinícius Moura Giglio (1); Vladimir Guilherme Haach (2).

(1) Departamento de Engenharia de Estruturas EESC/USP; vimourag@usp.br

(2) Professor Doutor, Departamento de Engenharia de Estruturas EESC/USP. vghaach@sc.usp.br
EESC/USP – Departamento de Engenharia de Estruturas – Av. Trabalhador São carlense, 400, Arnold Schmidt, CEP 13566-590, São Carlos - SP

Resumo

Os ensaios não destrutivos têm se mostrado muito eficientes no monitoramento da qualidade e na detecção de heterogeneidades do concreto, se firmando como uma alternativa de ensaio menos invasiva e com menor variabilidade estatística. Dentre esses, o ensaio de ultrassom tem se desenvolvido bastante. É possível combiná-lo com a técnica de tomografia computadorizada para gerar imagens do mapa de velocidades do pulso ultrassônico em seções transversais da estrutura. No entanto, para calcular essas velocidades, é necessário que se assuma uma hipótese a respeito da trajetória seguida pelos pulsos. O presente artigo tem como objetivo determinar a trajetória dos pulsos ultrassônicos em elementos de concreto utilizando algoritmos genéticos (AG), uma ferramenta de busca e otimização baseada nos princípios da seleção natural. Para este fim, um modelo do algoritmo foi implementado no *software* de geração de imagens tomográficas em estruturas TUSom. A fim de comparar os parâmetros de desempenho computacional e qualidade das soluções fornecidas, o algoritmo genético foi comparado com o algoritmo determinístico de Dijkstra em um exemplo de seção discretizada de concreto com defeitos preestabelecidos. Os resultados demonstraram que o algoritmo de Dijkstra apresentou melhor desempenho em problemas menores (com menos de 1000 nós), enquanto o algoritmo genético se mostrou superior nos outros casos. Assim, podem ser implementadas variações no algoritmo genético buscando melhor desempenho, sendo no momento uma abordagem mista a forma ideal de resolver o problema das trajetórias.

Palavra-Chave: Tomografia ultrassônica. Algoritmos genéticos. Ensaios não destrutivos.

Abstract

Nondestructive tests have shown efficient results in integrity monitoring of concrete structures, consolidating itself as a less invasive option, with smaller statistical variability. The ultrasonic pulse velocity test (UPV) is part of this group and one that has been developing over the past years. It is possible to combine the UPV and the tomographic imaging technique to create velocity maps of the ultrasonic pulse in structure's cross-sections. However, to measure the ultrasonic pulse velocity within the structure, it is necessary to propose a hypothesis about how the waves propagate. This article intends to evaluate the trajectory of ultrasonic pulses inside concrete structures using genetic algorithms (GA), a search and optimization tool based on the natural selection principles. To reach this goal, a model of genetic algorithm were implemented in a software of tomographic imaging in structures (TUSom). Intending to prove the computational performance and the quality of answers provided by the genetic algorithm, it was compared to the Dijkstra's algorithm in an example of discretized cross-section of concrete with simulated defects. The results showed that the Dijkstra's algorithm performed better in small problems (less than 1000 nodes) and the genetic algorithm was better in bigger ones. Therefore, variations in the genetic algorithm structure can be implemented seeking for better performance, but now the best approach is a mixed one, using both algorithms to solve the trajectories problem.

Keywords: Tomographic ultrasound imaging. Genetic algorithms. Nondestructive tests.

1 Introdução

O concreto é um dos materiais mais utilizados pela humanidade. Sua grande resistência à compressão, aliada a baixo custo e evolução no campo de estudo dos materiais, contribuiu para a sua popularização e permite a construção de qualquer tipo de obra. No entanto, estas apresentam vida útil finita. Além disso, estruturas de grande porte necessitam de constante monitoramento de sua integridade estrutural. Estas situações geram complicações na adoção de ensaios destrutivos. Nesse contexto, os ensaios não destrutivos têm se mostrado ferramentas com muito potencial, pois permitem ensaios rápidos (RAMÍREZ, 2015), menos invasivos e que eliminam muitas vezes a variabilidade estatística presente nos ensaios destrutivos convencionais, permitindo a análise de parâmetros ao longo do tempo (HOLA; SCHABOWICKZ, 2010). Sua utilização têm diversas finalidades, sendo as principais a determinação de propriedades mecânicas e a detecção de heterogeneidades e do grau de deterioração de estruturas.

Um destes ensaios é o ensaio de ultrassom, em que se utilizam transdutores que emitem e recebem pulsos ultrassônicos dentro de elementos estruturais, medindo o tempo de viagem destes pulsos. O ensaio ainda pode ser aliado à técnica de tomografia computadorizada, em que a partir de várias medições de tempo em pontos diferentes da estrutura é possível representar a distribuição do campo de velocidades do pulso ultrassônico em uma seção transversal da mesma, conhecida como tomograma.

Quando se executam experimentos com o uso do ultrassom, se conhecem os tempos de viagem dos pulsos. No entanto, as suas trajetórias ao longo da estrutura são incógnitas, visto que o pulso caminha pelos pontos de maior homogeneidade, que geram maior velocidade. Estas trajetórias geralmente são supostas lineares, o que corresponde a um material homogêneo, aproximação não coincidente com a realidade. Quando a hipótese anterior é abandonada e trajetórias mais próximas das reais são obtidas, o são de forma pouco eficaz através de algoritmos determinísticos, como o algoritmo de busca de Dijkstra, que testa todas as possibilidades de viagem do pulso dentro de uma malha discretizada no elemento de concreto, apresentando grande custo computacional.

Os algoritmos genéticos são métodos de otimização de natureza probabilística (metaheurísticas) que têm se mostrado muito eficazes na resolução de problemas em diversas áreas, surgindo como uma alternativa viável e menos custosa computacionalmente à solução do problema das trajetórias.

2 Ensaios não destrutivos

A propriedade de maior importância para o concreto é a resistência à compressão, pois costuma ser o fator limitante no dimensionamento de estruturas. Essa e outras propriedades costumam ser obtidas através de ensaios destrutivos de pequenas amostras coletadas em obra. No entanto, nem sempre é possível a moldagem ou retirada de corpos de prova. Um exemplo disso são estruturas antigas ou de grande porte, como barragens. Tem-se investido cada vez mais em ensaios não destrutivos (END) que sejam capazes de detectar propriedades e o grau de deterioração de estruturas de forma não invasiva. A heterogeneidade da estrutura de materiais como concreto e madeira torna o processo ainda mais complexo (MEHTA; MONTEIRO, 2014).

2.1 Ensaio de Ultrassom

Dentro do grupo dos ensaios não destrutivos, temos os métodos de propagação de ondas de tensão. O ensaio de ultrassom se enquadra nessa categoria e, como o próprio nome sugere, consiste na emissão e recepção de pulsos ultrassônicos (frequência de impacto acima de 20kHz) através de sensores chamados transdutores. A figura 1 mostra o arranjo do ensaio de ultrassom na modalidade transparência, em que um transdutor emite e outro recebe o sinal. Como a distância entre os transdutores é conhecida e o equipamento do ensaio fornece o tempo de viagem do pulso ultrassônico, calcula-se a sua velocidade de propagação através da equação 1, onde V representa velocidade (m/s), ΔS representa a distância entre os transdutores (m) e Δt representa o tempo de viagem da onda (s).

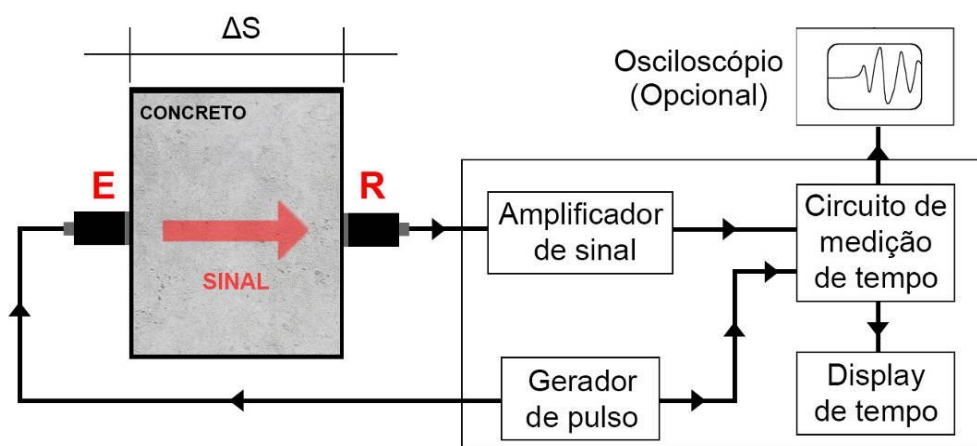


Figura 1 - Arranjo do ensaio de ultrassom na modalidade transparência (Adaptado de MALHOTRA e CARINO (2004)).

$$V = \frac{\Delta S}{\Delta t} \quad (\text{Equação 1})$$

Os parâmetros do pulso ultrassônico, como velocidade, amplitude, frequência e comprimento de onda sofrem variações ao passarem por zonas heterogêneas do material, permitindo a identificação de defeitos quando da existência de regiões de alta ou baixa velocidade e da atenuação de amplitude da onda emitida (RAMÍREZ, 2015).

2.2 Tomografia Computadorizada

A representação convencional do ensaio de ultrassom tenta representar o que ocorre em um corpo por meio de um gráfico unidimensional (figura 2.a). Aliando a técnica de tomografia computadorizada ao ensaio de ultrassom, à partir de várias medições de velocidade dos pulsos ultrassônicos dentro da estrutura, é possível reconstruir a distribuição de velocidades em uma seção transversal (bidimensional) da mesma, os chamados tomogramas (figura 2.b). A área em vermelho representa uma região de baixa velocidade quando comparada ao restante da seção, permitindo a identificação do defeito.

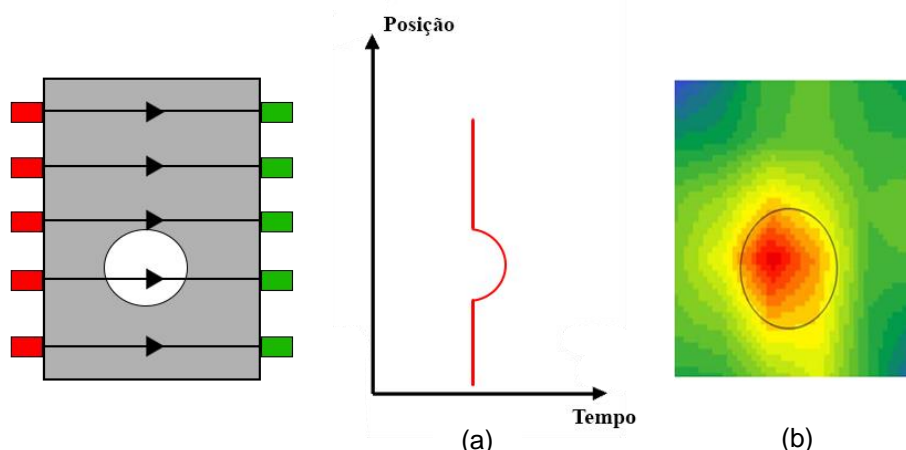


Figura 2 - Representação de um defeito com o uso do ultrassom: a) Unidimensional; b) Bidimensional com tomografia (Adaptado de RAMIREZ (2015)).

Existem dois caminhos adotados na geração de imagens tomográficas: a retroprojeção filtrada, que envolve o processo da transformada de Radon, e a reconstrução iterativa, baseada na solução algébrica de um sistema de equações (HAACH, 2017).

3 Trajetórias do pulso ultrassônico

3.1 O software TUSom

O *software* TUSom foi desenvolvido pelo Prof. Dr. Vladimir Guilherme Haach, com a colaboração de alunos de mestrado do Departamento de Engenharia de Estruturas da EESC/USP. À partir de medidas de tempo do ensaio de ultrassom, o *software* mapeia as velocidades do pulso ultrassônico em seções de uma estrutura, gerando tomogramas. O esquema da figura 3 ilustra resumidamente o funcionamento do *software* TUSom.

Na etapa de dados geométricos do *software*, a estrutura ensaiada com o ultrassom é discretizada em uma malha em função do tipo de transdutor utilizado (malhas 3x3cm na utilização de transdutores de 250kHz e 5x5cm para 54kHz). Em seguida, definem-se as coordenadas dos pontos de medição, tomadas geralmente como o ponto médio das arestas da malha. Após essa etapa, definem-se as trajetórias que as leituras realizam dentro da estrutura. Muitos autores adotam a aproximação de trajetórias retilíneas, como KWON, CHOI e SONG (2005), YANLI (2010), CHAI et al. (2010 e 2011), AGGELIS et al. (2011), PERLIN (2011), FERRARO, BOYD e CONSOLAZIO (2013), entre outros. Essa aproximação é razoável, mas não totalmente condizente com a realidade, visto que o pulso ultrassônico viaja pelos pontos de maior homogeneidade da estrutura.

Segundo JACKSON e TWEETON (1994, apud PERLIN, 2015, p. 88), quando a diferença de ordem de grandeza das velocidades apresentadas no meio não é maior que 10%, a aproximação da trajetória dos pulsos como retilíneos não resulta em grandes distorções nos resultados. Caso contrário, se recomenda a adoção de estratégias que mapeiem o trajeto do pulso ao longo da seção experimentada. Autores como PERLIN e PINTO (2019) se utilizam da ideia do algoritmo de Dijkstra para calcular trajetórias ótimas entre os diversos nós da malha discretizada.

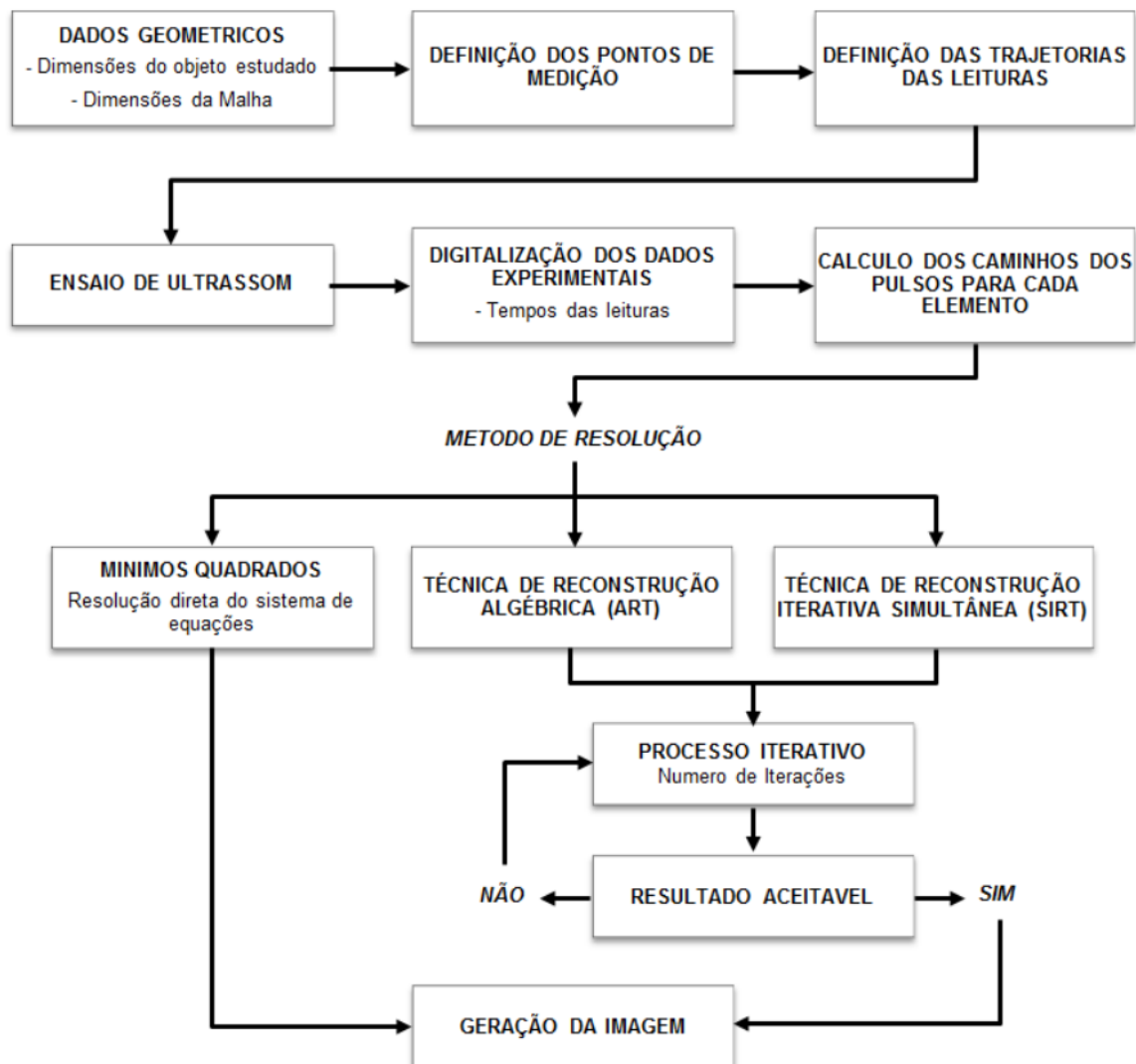


Figura 3 - Esquema de funcionamento do TUSom (RAMIREZ (2015)).

3.2 Algoritmo de Dijkstra

O algoritmo foi concebido inicialmente pelo cientista da computação Edsger Wybe Dijkstra, em 1956. Antes de introduzir o funcionamento do algoritmo, é necessário apresentar o conceito de grafos. Um grafo é um conjunto de dados que guarda relações entre si. Este possui nós, que são enumerados e contém alguma informação. As ligações entre os nós são chamadas de arestas e estabelecem as relações existentes entre eles, podendo apresentar pesos positivos ou negativos (ROMAN, 2017a).

A aplicação do algoritmo pode ser compreendida através de um exemplo: Tem-se um grafo com pesos positivos e deseja-se conhecer a menor distância entre os pontos a e d. Inicialmente, se assume a distância entre i e ele mesmo como valendo zero e entre i e qualquer outro vértice como sendo um valor "infinito". Também se armazenam os pontos precedentes a f ($p[f]$), adotados inicialmente como um valor nulo (-), visando ao final da aplicação do algoritmo saber o caminho trilhado (ROMAN, 2017b), vide figura 4.

Na inicialização do algoritmo, parte-se do vértice de menor distância de i . Marca-se então este vértice como visitado, e para todos os vértices adjacentes, se a nova distância entre i e o vértice em questão (conhecida, pois representa o peso da aresta) for menor que a estimativa adotada (inicialmente “infinita”), se atualiza a distância entre os dois nós (relaxamento da aresta). O algoritmo se repete até que todos os vértices sejam visitados. O algoritmo se mostra eficiente no cálculo da menor trajetória existente. Quando adotam-se muitos pontos de Dijkstra, a trajetória real é muito bem aproximada, mas a sua aplicação exige grande esforço computacional, visto que o algoritmo é determinístico, testando todas as possibilidades de viagem entre os nós da malha. Visando obter menores tempos de processamento no cálculo das trajetórias de pulsos ultrassônicos mais próximas das reais, foram aplicados os algoritmos genéticos.

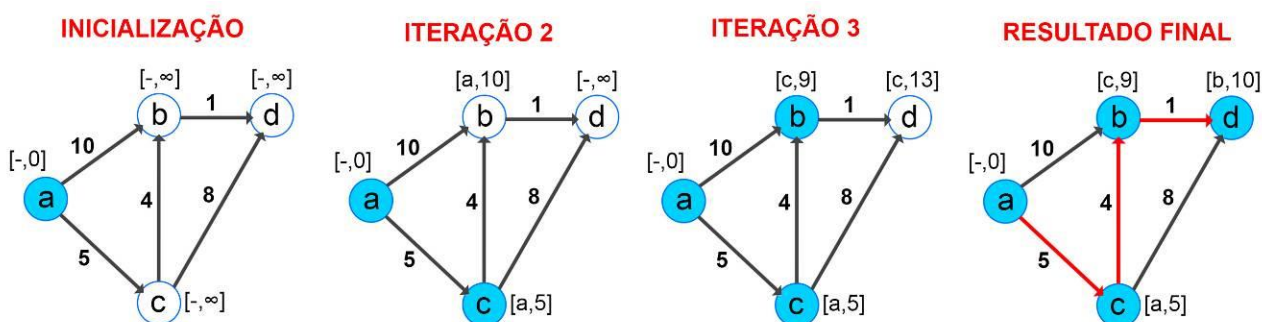


Figura 4 – Exemplo de aplicação do algoritmo de Dijkstra (Adaptado de ROMAN (2017b)).

3.3 Algoritmos Genéticos

Consiste num método de otimização de funções, dentro da categoria das chamadas metaheurísticas, e foi concebido na década de 1970 à partir da observação de fenômenos da natureza bem sucedidos. Basicamente, se aplicam operadores (chamados operadores genéticos) à uma função, rumando estatisticamente aos pontos de mínimo (não há necessidade de derivadas), o que resulta em maior sucesso para encontrar mínimos globais (HAUPT, HAUPT, 2004). Como o método se baseia em processos estocásticos, cada execução produz uma resposta diferente e não existe a garantia de que a melhor resposta absoluta (mínimo global) será sempre encontrada. Dessa forma, sempre que possível, devem-se adotar estratégias determinísticas, desde que apresentem desempenho computacional satisfatório, visto que sempre encontram a mesma solução, a melhor (LINDEN, 2006).

O funcionamento de um algoritmo genético pode ser explicado da seguinte forma: Cria-se, aleatoriamente, uma população de n indivíduos. Estes passam então por uma função de avaliação (*fitness*, em inglês) que mede o quão adaptados estes indivíduos são ao ambiente, de acordo com determinadas características valorizadas por essa função. Assim, os indivíduos são candidatos a solução de um problema de otimização e representam, por exemplo, a entrada x de uma função $f(x)$. No caso do problema ser de maximização, queremos descobrir o valor de x que faz a função $f(x)$ assumir o maior valor possível, dentro de um domínio previamente especificado.

O primeiro algoritmo genético, proposto por HOLLAND (1992), apresentava representação binária e três operadores genéticos essenciais: seleção, crossover e

mutação, ou seja, os indivíduos eram um conjunto de números binários que representavam cada uma das características de interesse (genes) para um problema. Inicia-se então a aplicação dos operadores: a seleção escolhe indivíduos aleatoriamente para reprodução e a probabilidade de um indivíduo ser selecionado é proporcional à sua avaliação. Em seguida, ocorre o crossover, que simula a reprodução sexuada, onde há troca de material genético (bits) entre os pais selecionados, gerando os filhos. Por fim, estes filhos podem sofrer mutação, tendo trechos de seus genes alterados espontaneamente. Tem-se então uma geração. O processo se repete até que se tenha dado número de gerações ou que determinado critério de parada seja atingido. A eficiência dos algoritmos genéticos em encontrar máximos/mínimos globais para funções reside no fato do crossover efetuar uma busca local de soluções (*exploitation*), enquanto a mutação efetua buscas globais menos frequentes (*exploration*) (LINDEN, 2006).

A aplicação ou não dos operadores genéticos é definida através de parâmetros como probabilidade de crossover (PC) ou de mutação (PM). Existem também outros operadores genéticos opcionais, como o elitismo, que transmite os melhores indivíduos da geração atual para a próxima sem a aplicação de operadores genéticos, visando conservar as melhores soluções encontradas até o momento (LINDEN, 2006).

4 Metodologia

No *software* TUSom, desenvolvido em linguagem Pascal, na plataforma Lazarus, foram implementados os algoritmos de Dijkstra e genético, alterando a etapa de definição das trajetórias que o pulso percorre dentro da estrutura. Em sua última etapa, o *software* calcula as velocidades nos nós da seção, definidos como o ponto de intersecção entre as linhas da malha definida. Estes serão chamados nós principais. Além dos nós principais, existe a opção de aumentar a densidade de nós, inserindo os chamados nós secundários. Nestes, a velocidade é definida à partir da interpolação linear das velocidades nos nós principais. Existem 9 níveis diferentes de densidade de nós, conforme figura 5. Quanto maior a densidade, mais próximo do comportamento real. Na aplicação dos algoritmos, é por estes nós que o pulso irá caminhar entre os pontos de medição origem e destino.

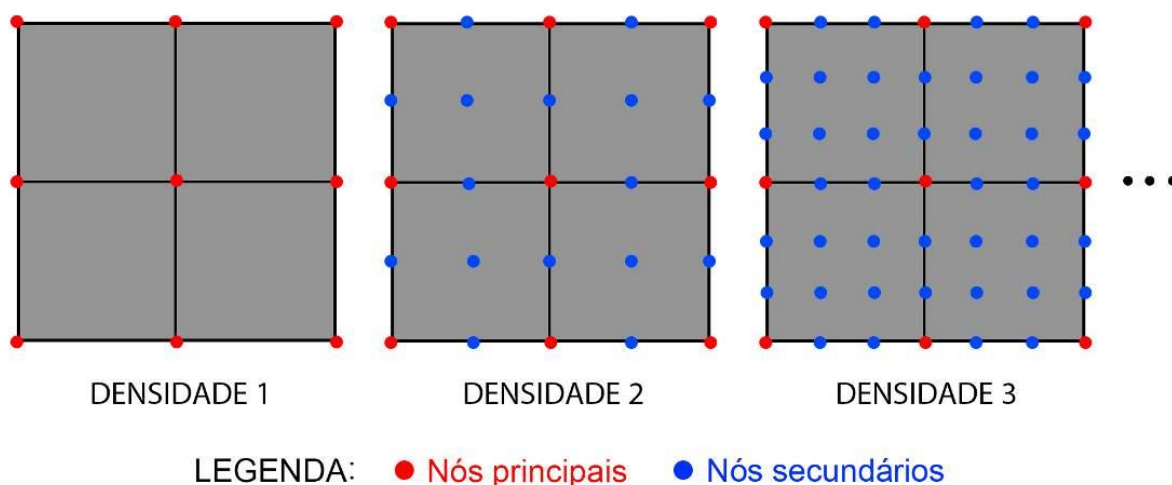


Figura 5 – Exemplo de malha discretizada e níveis de densidade de nós (Autor (2020)).

A implementação do algoritmo de Dijkstra seguiu os moldes do que foi exposto na seção 3.2 e sua utilização foi feita com o intuito de mapear os trajetos do pulso em um exemplo com defeitos preestabelecidos, variando os níveis de densidade de nós e permitindo a comparação dos resultados obtidos com o algoritmo genético em quesitos de tempo de execução e qualidade das respostas.

O exemplo utilizado foi uma seção 150x500mm, representando o corpo de prova padronizado do ensaio de flexão, usado também no ensaio de ultrassom (ABNT, 2016). Este foi dividido em uma malha de 3x10 elementos (44 nós principais), com defeitos pontuais (figura 6). Os defeitos foram representados adotando velocidade 1000m/s nos nós onde se encontram, enquanto o restante da malha possui velocidade 5000m/s, representando concreto íntegro. Além da malha adotada e dos defeitos preestabelecidos, a figura 6 também exibe as trajetórias retilíneas supostas inicialmente, sendo 160 ao todo.

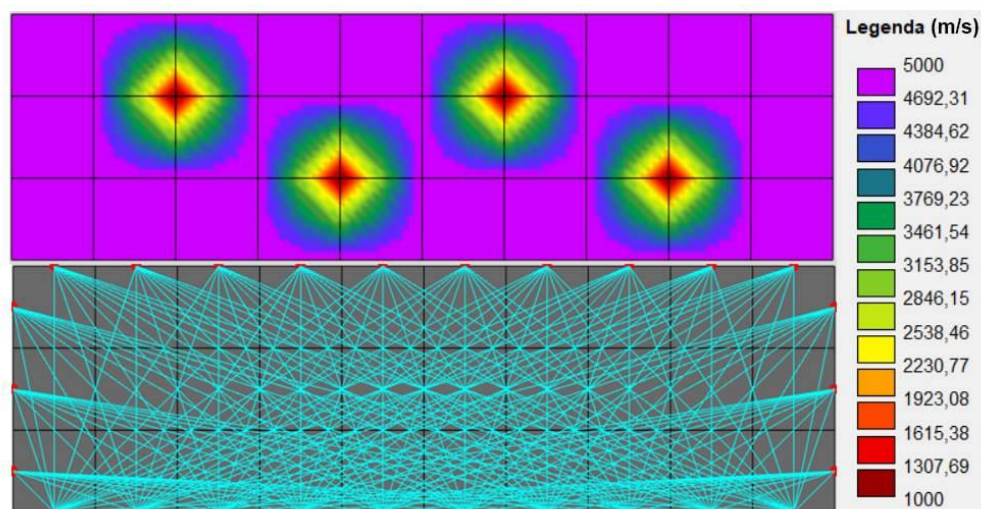


Figura 6 – Exemplo utilizado no mapeamento do pulso ultrassônico (Autor (2020)).

Pelo fato do algoritmo genético (AG) apresentar várias estruturas diferentes, sua implementação será melhor detalhada. Apesar da representação inicial de Holland adotar números binários, LINDEN (2006) recomenda que a representação seja sempre a mais natural possível, adequando a ferramenta ao problema. Assim, optou-se pela utilização de números inteiros para representar os indivíduos: cada um é um vetor (sorteado aleatoriamente) que representa os nós (principais ou secundários) pelos quais o pulso pode caminhar até o transdutor destino. A avaliação dos indivíduos é dada pelo inverso do tempo de viagem até o transdutor destino, tornando o problema um problema de maximização (o indivíduo com menor tempo de viagem tem a maior avaliação). O restante do indivíduo não contribui para sua avaliação. Apesar de serem gerados caminhos ruins inicialmente, estes são eliminados com o passar das gerações. Além disso, indivíduos que apresentam trajetórias retilíneas diretas entre transdutor emissor e receptor são penalizadas com um aumento em 10 vezes do seu tempo de viagem (visto que já conhecemos estas trajetórias de antemão e queremos evitar sua proliferação).

O operador de seleção utilizado foi o da roleta viciada, por sua grande difusão e simplicidade, o que gera rapidez, colaborando para um bom desempenho.

No que diz respeito ao crossover, adotou-se o operador de crossover uniforme, pelo fato deste tender a apresentar resultados superiores, apesar de ser mais devagar que o crossover de dois pontos (LINDEN, 2006). Neste operador, gera-se um bitstring aleatório que determina qual gene do filho provém de cada pai. Adaptou-se sua morfologia para o AG baseado em ordem (*ordering GA*), visto que a ordem em que os nós aparecem na composição de um indivíduo possuem influência na qualidade da resposta final. Ainda deve-se respeitar a restrição de que o pulso não pode passar duas vezes pelo mesmo nó. Para o operador de mutação, novamente utilizaram-se os conceitos do AG baseado em ordem, adotando o operador de mistura de sublistas, em que se sorteiam dois pontos de corte e se misturam os genes entre estes pontos. DAVIS (1991, apud LINDEN, 2006, p. 181) afirma que este operador se mostra mais agressivo que outros, como a permutação de elementos, atingindo resultados satisfatórios em suas pesquisas. Também adotou-se elitismo, com 2 elites por geração, armazenando as melhores soluções encontradas até então e passando-as diretamente à próxima geração. A figura 7 apresenta um exemplo de malha adotada, indivíduos gerados e da atuação dos operadores genéticos.

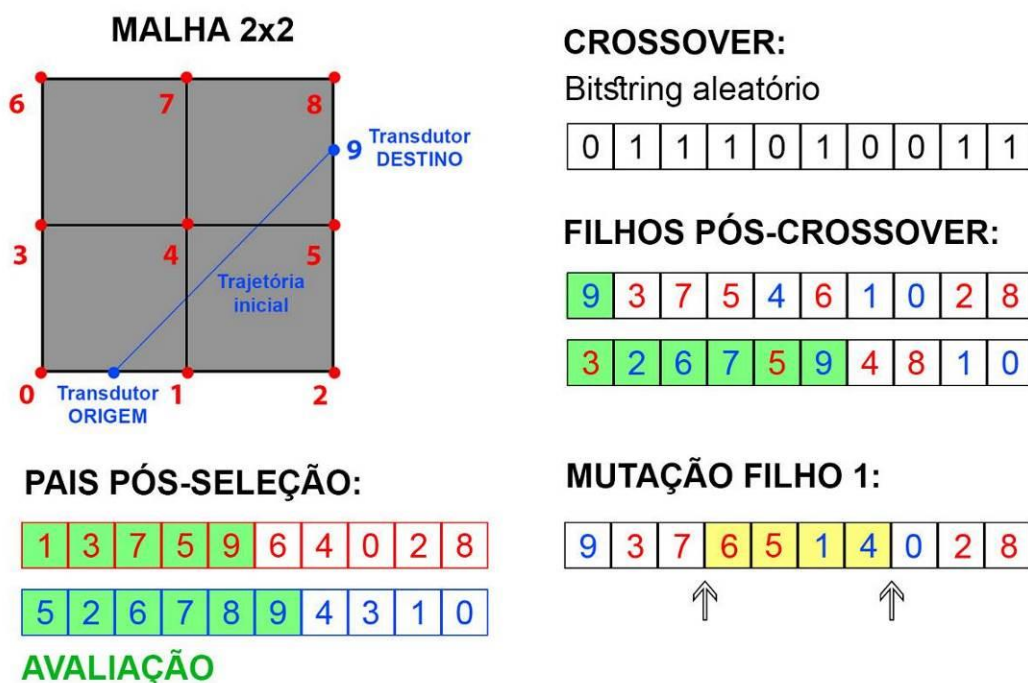


Figura 7 – Exemplo de funcionamento do algoritmo genético implementado (Autor (2020)).

A porcentagem dos operadores foi adotada como 90% para o crossover e 10% para a mutação, estimulando fortemente as buscas locais. Estes valores são adotados na maioria das publicações e têm atingido bons resultados. Alguns autores afirmam que a soma das porcentagens dos operadores deve ser 1 (LINDEN, 2006). Os parâmetros do AG que foram alterados ao longo de sua aplicação foram o número de indivíduos gerados e o de gerações adotadas, fator que possui maior impacto em seu desempenho. Alteraram-se estes em cada exemplo, para cada uma das densidades de nós, buscando atingir resultados com 10% de erro máximo, sendo o erro calculado em relação aos resultados obtidos pela aplicação do algoritmo de Dijkstra, de acordo com a equação 2.

ANAIS DO 62º CONGRESSO BRASILEIRO DO CONCRETO - CBC2020 – 62CBC2020

Como cada execução do AG gera resultados diferentes (método probabilístico), este foi executado duas vezes consecutivas, com os resultados finais representados pela média.

$$erro = \left(\frac{\Delta T_{AG} - \Delta T_{Dijkstra}}{\Delta T_{Dijkstra}} \right) \cdot 100 \quad (\text{Equação 2})$$

5 Resultados e discussão

A figura 8 exibe os resultados de tempo para a aplicação dos algoritmos de Dijkstra e genético para o exemplo estudado, com as densidades de malha de 1 a 6, onde a seção discretizada apresenta até 900 nós. Apesar dos resultados absolutos serem diferentes de acordo com as especificações de cada máquina onde o *software* for executado, o formato das curvas se mantém, visto que o código é sequencial (não paralelizado).

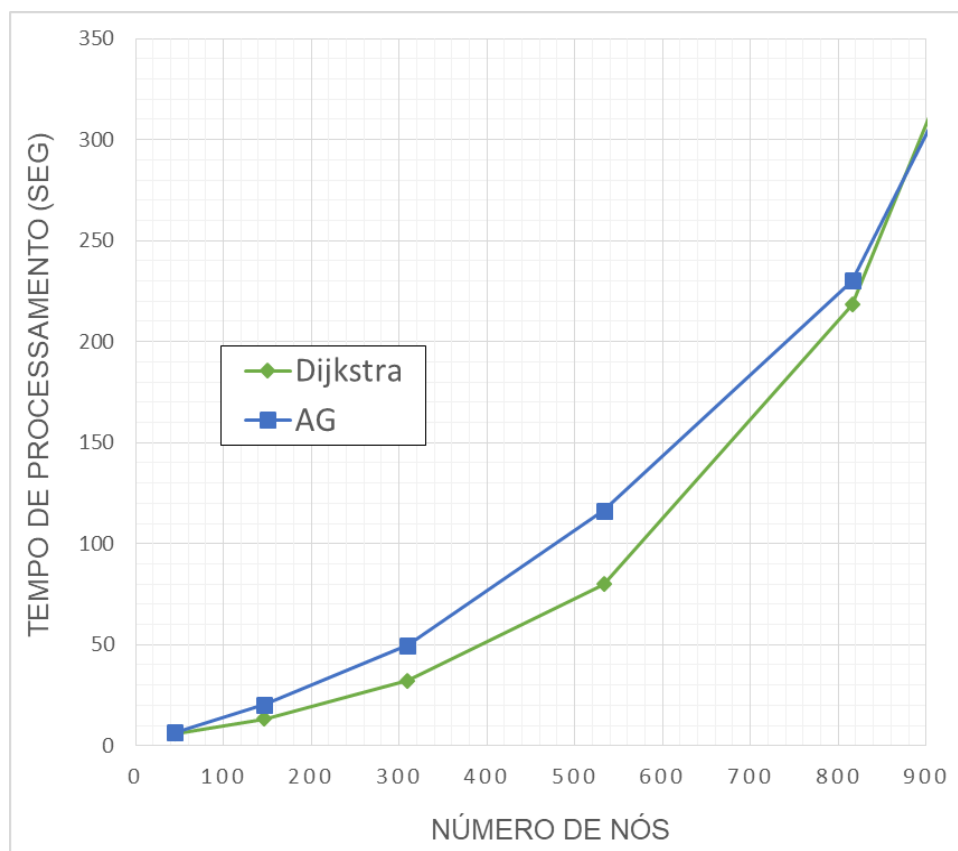


Figura 8 – Tempo de processamento dos algoritmos no exemplo até 900 nós (Autor (2020)).

Através da figura, é possível observar que, até um limite de cerca de 900 nós, o algoritmo de Dijkstra implementado apresenta desempenho superior ao do algoritmo genético. No que diz respeito à qualidade das respostas, o primeiro é sempre superior, visto que testa todas as possibilidades e calcula a melhor trajetória existente, apresentando sempre o mesmo resultado final, independentemente do número de execuções (algoritmo determinístico).

O resultado deve-se a vários motivos, dentre eles a adoção do operador de crossover uniforme no algoritmo genético, que se mostra eficiente no que diz respeito à qualidade dos resultados, mas é mais devagar que outros operadores “mais simples”, como são os crossovers de um ou dois pontos.

Também é possível que o modelo de criação de indivíduos adotado não seja o mais conveniente, visto que cada indivíduo é um vetor com todos os nós possíveis de serem visitados, visando a facilidade de aplicação dos operadores (todos indivíduos têm o mesmo tamanho) e o cumprimento da condição de não repetição de nós em um mesmo indivíduo. Isso faz com que, quando a malha apresenta densidade 5, cada indivíduo seja um vetor com aproximadamente 800 elementos, sendo que os que aparecem após o nó do transdutor destino não acrescentam em sua avaliação, mas exigem memória para armazenamento, instruções para criação e são operados normalmente. Um modelo com indivíduos menores ou que não seja criado aleatoriamente, ou seja, que respeite a vizinhança de cada um dos nós, criando um caminho inicial melhor, pode se mostrar mais eficiente na rapidez de convergência do método e/ou requerer menos esforço computacional.

Dessa forma, a utilização do algoritmo genético não se justifica no intervalo em questão e a preferência deve ser dada ao algoritmo de Dijkstra. No entanto, conforme a densidade de nós aumenta, o tempo de processamento se comporta de forma diferente para cada um dos algoritmos, conforme mostrado na figura 9.

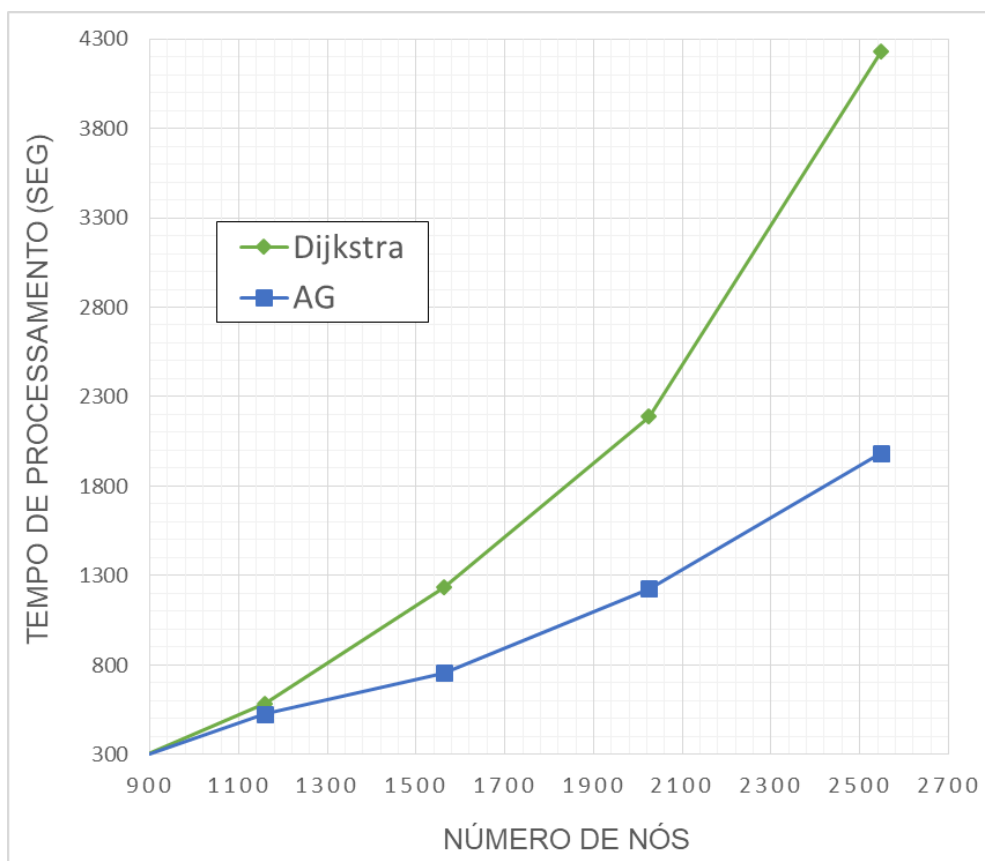


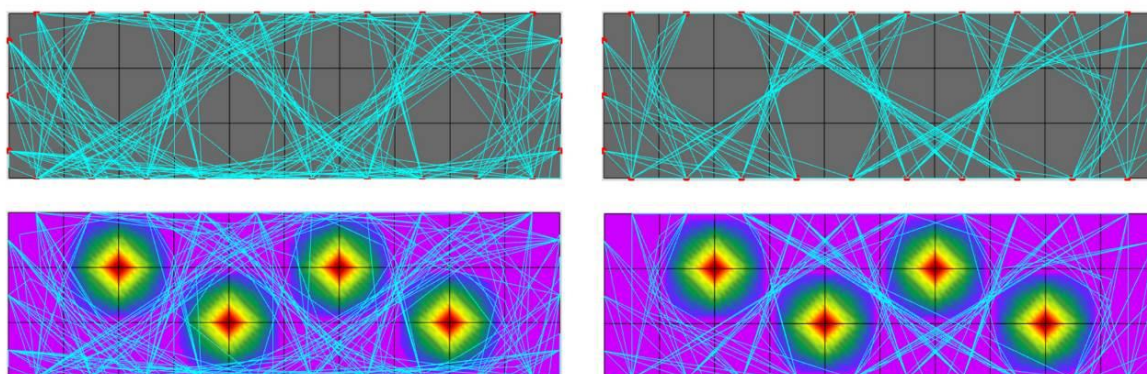
Figura 9 – Tempo de processamento dos algoritmos no exemplo após 900 nós (Autor (2020)).

Observando a figura, é possível concluir que o algoritmo genético é muito menos sensível ao crescimento da malha de nós do que o algoritmo de Dijkstra, apresentando tempos menores em todas as execuções a partir de 900 nós. Na densidade 9, quando o número de nós ultrapassa 2500, o algoritmo genético apresenta tempo de execução menor que a metade do tempo registrado para o algoritmo de Dijkstra. Isso se deve ao fato do último crescer exponencialmente junto com o número de nós, pois o número de possibilidades de visita na malha é $N!$, sendo N o número total de nós. Isso ocorre porque cada nó visitado elimina apenas uma possibilidade dentro de um espaço de busca muito grande.

O algoritmo genético também depende da malha de nós, que influencia a criação e o tamanho de cada indivíduo, além da quantidade de vezes que cada operador será acionado. No entanto, o AG depende muito mais do número de indivíduos e de gerações adotados, que foi relativamente pequeno dada a quantidade de nós: apenas 100 indivíduos e 30 gerações, com $PM = 0,9$ e $PC = 0,1$, foram suficientes para gerar indivíduos com erro satisfatório (igual ou menor a 10%) no exemplo em questão.

A experimentação também permitiu concluir que o aumento do tamanho da população se mostrou mais efetivo na obtenção de melhores soluções do que o aumento do número de gerações, visto que o último, quando excessivo, acabou destruindo boas soluções (ou soluções com potencial para serem boas) ao invés de ajudar a criar novas.

No geral, a média dos resultados apresentados pelo algoritmo genético se mostra bastante satisfatória. Apesar disso, algumas trajetórias desviam-se muito das ideais, chegando em alguns casos a apresentar erros de mais de 100%. A figura 10 apresenta uma comparação das trajetórias encontradas no exemplo discutido para malha de nós com densidade 5, quando da utilização do algoritmo genético e de Dijkstra.



ALGORITMO GENÉTICO

ALGORITMO DE DIJKSTRA

Figura 10 – Comparação das trajetórias encontradas com o uso de cada um dos algoritmos (Autor (2020)).

Percebe-se que existe menor sobreposição de trajetórias no caso do algoritmo genético, visto que este é probabilístico e apresenta resultados diferentes a cada execução, mas é possível observar também que algumas trajetórias passam pelas zonas de heterogeneidade do material, o que sabe-se que não ocorre na realidade. Isso poderia ser resolvido adotando uma abordagem que aplique métodos determinísticos apenas em trajetórias que apresentam erros muito altos, tirando proveito do melhor de cada um dos métodos.

6 Conclusões

Apesar de ainda existirem muitas possibilidades de estruturas diferentes, o algoritmo genético se mostrou promissor em sua aplicação, especialmente no âmbito de resolver problemas maiores, onde os número de nós adotados passa de mil. Isso se deve principalmente à menor sensibilidade dos algoritmos genéticos ao tamanho da malha de nós, que não impõe o mesmo ritmo de crescimento de dados que impõe ao algoritmo de Dijkstra.

No futuro, podem ser adotados tamanhos de população e probabilidades de crossover e mutação dinâmicos (mudam ao longo das gerações), outros operadores de crossover mais simples, buscando melhor desempenho, ou inspirados no algoritmo genético baseado em ordem, como o *edge recombination* proposto por MICHALEWICZ (2002) ou o PMX (mapeamento parcial). O operador de mutação também é custoso computacionalmente e pode ser substituído por simples permutação de elementos ou inversão de sublistas, ambos provenientes da formulação do AG baseado em ordem. Todo esse processo visa encontrar a estrutura ótima que alia bom desempenho computacional e qualidade de respostas satisfatória, antes da aplicação de técnicas de otimização mais agressivas, como a paralelização.

Outros métodos metaheurísticos ainda podem performar melhor na resolução do problema das trajetórias, como a otimização por colônia de formigas (ACO – *ant colony optimization*), o método de enxame de partículas (PSO – *particle swarm optimization*) ou até a adoção de métodos híbridos: AG com lógica fuzzy ou com redes neurais artificiais.

É possível observar também, à partir dos resultados obtidos, que os algoritmos determinísticos não devem ser desprezados, podendo ser utilizados em problemas com menor quantidade de nós. Dessa forma, outros algoritmos de busca determinísticos como o algoritmo A* (lê-se “A estrela”) podem ser testados, adotando um processo de solução misto (algoritmos determinísticos até determinado tamanho de malha, metaheurísticas à partir deste).

7 Referências

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR 5738: Concreto - Procedimento para moldagem e cura de corpos de prova**. Rio de Janeiro, ABNT, 2016.

AGGELIS, D.G. et al. Numerical simulation of elastic waves for visualization of defects. **Construction and Building Materials**, No. 4, vol. 25, 2011, p. 1503-1512.

CHAI, H. K. et al. Singleside access tomography for evaluating interior defect of concrete. **Construction and Building Materials**, No. 12, vol. 24, 2010, p. 2411-2418.

CHAI, H. K. et al. Tomographic reconstruction for concrete using attenuation of ultrasound. **NDT & E International**, No. 2, vol. 44, 2011, p. 206-215.

DAVIS, L.. **Handbook of genetic algorithms**. New York, Van Reinhold Nostrand, 1991.

HAACH, V. G.. **Aplicação do ultrassom como ensaio não destrutivo para a avaliação de elementos de concreto**. 2017. Texto sistematizado (Livre docência) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos.

HAUPT, R. L.; HAUPT, S. E.. **Practical Genetic Algorithms**. 2. ed. Hoboken, John Wiley & Sons, 2004.

HOLA, J.; SCHABOWICKZ, K.. State-of-the-art non-destructive methods for diagnostic testing of building structures – anticipated development trends. **Archives of civil and mechanical engineering**, No. 3, vol. 10, 2010, p. 5-18.

HOLLAND, J. H.. Genetic Algorithms. **Scientific American**, No. 1, vol. 267, 1992, p. 66-73.

JACKSON, M. J.; TWEETON, D. R.. **MiGRATOM – Geophysical tomography using wavefront migration and fuzzy constraints**. U.S. Department of Interior – Bureau of Mines. Wasinghton, 1994.

FERRARO, C.C.; BOYD, J.A.; CONSOLAZIO, R.G.. Evaluation of damage to bridge piers using pulse velocity tomography. **Construction and Building Materials**, vol. 38, 2013, p. 1303-1309.

KWON, J.; CHOI, S. J.; SONG, S. M.. Implementation and evaluation of the ultrasonic TOF tomography for the NDT of concrete structures. **Proceedings of SPIE 5674**, Computational Imaging III, vol. 47, 2005, 8p.

LINDEN, R.. **Algoritmos Genéticos** – Uma importante ferramenta da inteligência computacional, Rio de Janeiro, Brasport, 2006.

MALHOTRA, V. M.; CARINO, N. J.. **Nondestructive testing on concrete**. 2. ed. Boca Raton, CRC Press, 2004.

MEHTA, P. K.; MONTEIRO, P. J. M.. **Concreto** - Microestrutura, propriedades e materiais. 2. ed. São Paulo, IBRACON, 2014.

MICHALEWICZ, Z.; FOGEL, D. B.. **How to solve it: Modern heuristics**. 1. ed. Cambrigde, MIT Press, 2002.

PERLIN, L. P.. **Tomografia ultrassônica em concreto: Desenvolvimento de ferramenta computacional**. 2011. Dissertação (Mestrado em Engenharia Civil) – Centro Tecnológico, Universidade Federal de Santa Catarina, Florianópolis.



Anais do
62º Congresso Brasileiro do Concreto
CBC2020
Setembro / 2020



@ 2020 - IBRACON - ISSN 2175-8182

PERLIN, L. P.. **Tomografia ultrassônica em concreto e madeira: Desenvolvimento de ferramenta computacional.** 2015. Tese (Doutorado em Engenharia Civil) – Centro Tecnológico, Universidade Federal de Santa Catarina, Florianópolis.

PERLIN, L. P.; PINTO, R. C. A.. Use of the network theory to improve the ultrasonic tomography in concrete. **Ultrasonics**, vol. 96, 2019, p. 185-195.

RAMÍREZ, F. C.. **Detecção de danos em estruturas de concreto por meio de tomografia ultrassônica.** 2015. Dissertação (Mestrado em Engenharia Civil (Estruturas)) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos.

ROMAN, N. T.. **Estrutura de dados** – Aula 23: Grafos – Conceitos Básicos. 2017. Disponível em: <<https://www.youtube.com/watch?v=MC0u4f334ml>>. Acesso em: 06 out. 2019.

ROMAN, N. T.. **Estrutura de dados** – Aula 28: Grafos – Algoritmo de Dijkstra. 2017. Disponível em: <<https://www.youtube.com/watch?v=ovklTlgyJ2s>>. Acesso em: 06 out. 2019.

YANLI, C.. The shortest path ray tracing algorithm in concrete computerized tomography. **Information Technology and Applications**, 1, 2010, p. 392 - 396.