

Otimização da disposição de peças para Impressão 3D e Torres de Internet 5G

Resolução de dois problemas de engenharia usando a heurística *Firefly*

Francisco Power · Pedro Rolo

Submetido: 19/1/2021

Resumo Resolução de dois problemas de engenharia usando, como ferramenta de otimização, o algoritmo *Firefly*. Avaliação do desempenho deste algoritmo resolvendo o problema *Benchmark 2020 (A)*.

Palavras-Chave Otimização · *Firefly* · Heurísticas · Internet 5G · Impressão 3D

1 Introdução

A Internet 5G e a Impressão 3D são duas tecnologias cada vez mais importantes. Apresentam inúmeras vantagens face a tecnologias semelhantes, mas também têm desvantagens. É possível atenuar os efeitos desses problemas traduzindo-os em problemas de otimização e resolvendo-os como tal.

Neste trabalho foram abordados dois desafios, formulados como problemas de otimização e resolvidos usando uma meta-heurística, o algoritmo *Firefly*. Estudou-se, também, o desempenho e robustez deste algoritmo através da resolução do problema *Benchmark 2020*, o problema da minimização do peso de um redutor.

2 Apresentação dos problemas de otimização

2.1 Impressão 3D

A Impressão 3D não só é extremamente útil para prototipagem rápida, como também já é usada para pro-

duções de peças de geometria complexa de pequena e média escala. No entanto ainda é um processo relativamente lento.

O objetivo deste trabalho é desenvolver um programa que minimize o tempo de impressão, minimizando o percurso da extrusora em vazio. Conhecendo o formato das peças a imprimir e as dimensões da cama quente, o programa deverá dispor as peças na cama quente de forma ótima, aplicando uma determinada translação e rotação a cada peça.

O programa terá que ter em consideração o percurso da extrusora em todas as camadas da impressão e não apenas a forma da base.

2.2 Internet 5G

A Internet 5G traz inúmeras vantagens, nomeadamente a velocidade de transmissão de dados, o que permitirá grandes avanços na área de IoT, condução autónoma, entre outras áreas. No entanto, as ondas de comprimento de onda milimétrico têm um alcance bastante baixo, sendo também facilmente bloqueadas por edifícios e outras obstruções de tamanhos similares. O objetivo deste trabalho é descobrir, para uma certa área e um certo número de torres 5G, a distribuição que ofereça a máxima cobertura do espaço, tendo em conta o efeito das obstruções mencionado. Será, posteriormente, avaliado também o melhor número de torres 5G.

A área que se pretende afetar é obtida através de uma imagem de satélite, a qual é depois convertida num mapa de obstruções e restrições, como se pode ver na figura 1.

Francisco Power
n.º 84706
E-mail: franciscopower66@ua.pt

Pedro Rolo
n.º 84803
E-mail: pmrolo@ua.pt



Figura 1. Mapa de obstruções (vermelho) e zonas restritas (vermelho + azul).

3 Formulação dos problemas de otimização

3.1 Impressão 3D

Neste problema, o objetivo é minimizar a distancia que a extrusora percorre em vazio, ou seja, o somatório em todas as camadas da distancia no plano entre o ponto de saída da extrusora de uma peça, $[a_n, b_n]$, e o ponto de entrada da peça seguinte, $[c_{n+1}, d_{n+1}]$. Estes relacionam-se com o vetor de translação, $[x_{(t,n)}, y_{(t,n)}]$, e o ângulo de rotação de cada peça, θ_n , substituindo $[x_n, y_n]$ por $[a, b]$ ou $[c, d]$ na equação 1:

$$\begin{bmatrix} x_n \\ y_n \\ f(x_n, y_n) \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta_n) & -\sin(\theta_n) & 0 & x_{(t,n)} \\ \sin(\theta_n) & \cos(\theta_n) & 0 & y_{(t,n)} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{n_0} \\ y_{n_0} \\ f(x_{n_0}, y_{n_0}) \\ 1 \end{bmatrix} \quad (1)$$

, sendo f a função de forma de cada peça, $[x_{n_0}, y_{n_0}]$ as coordenadas x e y de um ponto da peça n estando esta centrada na origem do referencial e $[x_n, y_n]$ as coordenadas x e y do mesmo ponto após a transformação. Assim sendo, e considerando N o número de peças a imprimir e C o número máximo de camadas de impressão, pode-se formular o problema como apresentado seguidamente:

Variáveis de projeto:

$$\mathbf{x} = [x_{(t,n)}, y_{(t,n)}, \theta_n]^T \quad (2)$$

Função objetivo:

Minimizar:

$$F(\mathbf{x}) = \sum_{c=1}^C \left(\sum_{n=1}^{N-1} \left(\sqrt{(c_{n+1} - a_n)^2 + (d_{n+1} - b_n)^2} + \sqrt{(c_1 - a_N)^2 + (d_1 - b_N)^2} \right) \right)_c \quad (3)$$

Sujeito a:

$$\begin{aligned} g_1(\mathbf{x}) &= -x_n - Cc/2 \leq 0, \\ g_2(\mathbf{x}) &= x_n - Cc/2 \leq 0, \\ g_3(\mathbf{x}) &= -y_n - Lc/2 \leq 0, \\ g_4(\mathbf{x}) &= y_n - Lc/2 \leq 0, \\ g_5(\mathbf{x}) &= 3 - \sqrt{(c_{n+1} - a_n)^2 + (d_{n+1} - b_n)^2} \leq 0, \\ h(\mathbf{x}) &= A(f_n(x, y) \wedge f_{n+1}(x, y)) = 0, \end{aligned} \quad (4)$$

, sendo Cc o comprimento da cama quente e Lc a largura da cama quente e estando n a variar entre 0 e o número de peças.

3.2 Internet 5G

Para um determinado numero de torres, o objetivo é minimizar a zona não coberta, o que pode ser feito calculando a intensidade de sinal em cada ponto, seguido do cálculo da razão entre a área com uma intensidade de sinal inferior a uma intensidade mínima e a área total.

A intensidade de sinal num determinado ponto é dada pela equação 5:

$$I(x_n, y_n) = \frac{P}{4\pi((x - x_p)^2 + (y - y_p)^2)} \quad (5)$$

, sendo P a potencia do sinal, $[x, y]$ as coordenadas do ponto em análise e $[x_p, y_p]$ as coordenadas da torre 5G mais próxima e sem obstruções entre o ponto e a torre). Assim sendo, e considerando A a área do espaço excluindo as obstruções e considerando I_{min} a intensidade mínima de sinal admissível, podemos definir o problema como apresentado seguidamente.

Variáveis de projeto:

$$\mathbf{x} = [x_n, y_n]^T \quad (6)$$

Função objetivo:

minimizar:

$$F(\mathbf{x}) = \frac{\sum_{x=0}^{Ce} \left(\sum_{y=0}^{Le} a \right)}{A} \cdot 100, \quad a = \begin{cases} 0, & \text{se } I(x_p, y_p) > I_{min} \\ 1, & \text{se } I(x_p, y_p) \leq I_{min} \end{cases} \quad (7)$$

Sujeito a:

$$R(\mathbf{x}) = (x_n, y_n) \in A_{permitida} \quad (8)$$

e com

$$\begin{aligned} 0 &\leq x_n \leq Ce, \\ 0 &\leq y_n \leq Le \end{aligned} \quad (9)$$

, sendo C_e o comprimento e L_e a largura do espaço a cobrir.

É importante notar que os mapas de obstruções e áreas restritas para colocação de torres são imagens. Assim sendo, as coordenadas das posições das torres 5G são as coordenadas de determinados pixels da imagem, logo, discretas inteiras.

4 Análise de sensibilidade

A análise de sensibilidade tem como principal objectivo verificar o efeito da alteração de uma variável na função objectivo, mostrando assim o quão sensível é a função objectivo a mudanças dessa variável.

4.1 Impressão 3D

A análise de sensibilidade deste problema fez-se variando as coordenadas dos pontos de chegada e saída de objeto. Alterando a posição dos objetos altera-se o comprimento do percurso da extrusora, que é directamente proporcional ao tempo de impressão.

Para isso usou-se o programa Cura da Ultimaker de modo a simular os tempos de impressão de um conjunto de peças diferentes em posições diferentes. Foi possível confirmar que, quanto maior é a distancia entre os objetos, maior é o tempo de impressão, e que a orientação também influencia o seu tempo de impressão.

4.2 Internet 5G

Após programar o problema, foi possível observar que o valor resultado da função objectivo é menor no caso em que as torres estão muito juntas num espaço confinado, abrangendo uma pequena área, e significativamente maior no caso em que as torres estão mais afastadas e abrangendo uma área maior.

É também visível a existência de degraus na função objectivo, devido às obstruções, como se pode ver no gráfico da figura 2, em que os valores foram obtidos alterando apenas uma coordenada da posição de uma torre. Isto é algo de elevada importância na escolha do algoritmo de optimização.

5 Avaliação

5.1 Impressão 3D

Para poder fazer a avaliação do problema, foi ,primeiro, necessário obter as coordenadas de todos os pontos do percurso da extrusora. Para tal, foi gerado o código CN

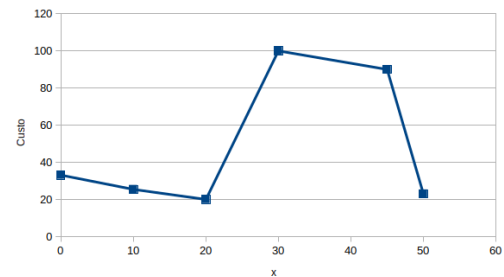


Figura 2. Variação do custo variando a coordenada x de uma torre 5G

no programa Cura, com 4 peças arbitrárias centradas na origem da cama quente. Um programa em Python lê os documentos .gcode e cria uma matriz de pontos, sendo possível extrair dessa matriz os pontos de início e fim do percurso da extrusora em cada camada. O algoritmo de optimização Firefly, descrito na secção 6, gera os valores de translação e rotação para cada peça e são aplicadas as transformações a todos os pontos de cada peça. Por fim, é calculado o valor da função objectivo, como descrito na secção 3.1.

5.2 Internet 5G

A avaliação do problema começa com a geração das coordenadas das torres pelo algoritmo Firefly, descrito em 6, e introdução das mesmas num programa python que, tendo também como argumentos de entrada o mapa de obstruções e restrições (figura 1), percorre todos os pontos do mapa, verifica quais as torres sem obstruções entre a elas e o ponto e , e de entre essas torres, calcula o valor da função objectivo, como descrito na secção 3.2.

6 Algoritmo de optimização

6.1 O algoritmo *Firefly*

A optimização deste problema pode ser feita com recurso a muitos algoritmos de optimização diferentes.

Tendo em consideração que uma das funções objectivo apresenta degraus, optou-se por não utilizar métodos baseados no gradiente da função.

Métodos heurísticos, apesar de nem sempre convergirem com exatidão no mínimo da função, permitem encontrar o ótimo global mesmo em funções com degraus, não contínuas.

O algoritmo *Firefly* é um método heurístico baseado na ideia de inteligência de enxames (*Swarm Intelligence*) [3], em particular, de pirilampos. O método assume que cada pirilampo é atraído por todos os outros, sendo essa atracção proporcional ao seu brilho e

```

x = rand(x_min, x_max) #inicializar populacao
I = f(x)
for t in range(0, iteracao_max):
    for i in range(1, populacao):
        for j in range(1, populacao):
            if I[j] < I[i]:
                x_novo = ... # mover pirilampo i na
                             direcao j segundo a equacao 8
                I[i] = f(x_novo)
        # guardar o otimo global
        alpha = alpha * damp

```

Tabela 1. Pseudo-código do Algoritmo *Firefly*

inversamente proporcional à distância entre os pirilampos. No contexto de otimização, o brilho do pirilampo é determinado pela função objetivo [4].

As posições iniciais dos pirilampos são geradas aleatoriamente e a deslocação de um pirilampo gera novas soluções.

6.2 Pseudo-código do algoritmo

O pseudo-código do algoritmo encontra-se na tabela 1 [4] [5].

Em cada avaliação, a nova posição do pirilampo i , x_i^{novo} , é dada pela equação 10:

$$x_i^{novo} = x_i^{ant} + \beta_0(x_j - x_i)e^{-\gamma d^2} + \alpha L \cdot \text{rand}(-1,1) \quad (10)$$

onde, β_0 é a atração entre pirilampos à distancia 0, γ é o coeficiente de absorção, $d = \|x_j - x_i\|$ é a distancia entre os pirilampos, α é o fator de aleatoriedade e L é a escala do problema [4] [2].

6.3 Penalizações externas

O algoritmo *Firefly* assume que o problema não tem restrições. Uma vez que os problemas em estudo têm restrições, é necessário transforma-los em problemas sem restrições. Isto pode ser feito aplicando penalizações às funções objetivo, como se pode ver na equação 11:

Procurar \mathbf{x} de modo a :

$$\begin{aligned} &\text{minimizar } F(\mathbf{x}, r_g) = f(\mathbf{x}) + P(\mathbf{x}, r_g, r_h), \\ &\text{sujeito a } x_i^{min} \leq x_i \leq x_i^{max}, \end{aligned} \quad (11)$$

sendo $P(\mathbf{x}, r_g, r_h)$ a penalização da função objetivo e $f(\mathbf{x})$ a função objetivo original.

No problema da impressão 3D, as penalizações podem ser calculadas de acordo com a equação 12, sendo r_g e r_h os fatores de penalização.

$$P(\mathbf{x}, r_g, r_h) = r_g \cdot \sum_{j=1}^m [\max\{0, g_j(\mathbf{x})\}]^2 + r_h \cdot (h(\mathbf{x}))^2 \quad (12)$$

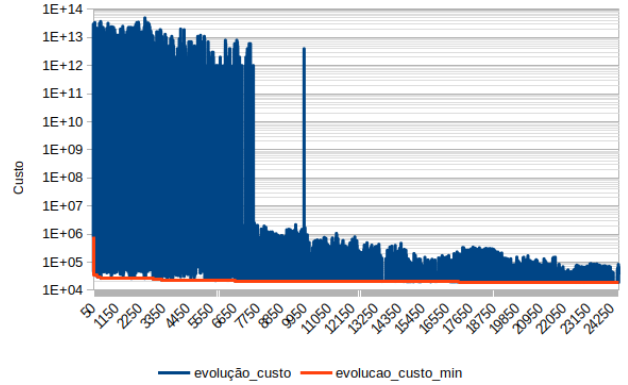


Figura 3. Evolução do custo mínimo da função objetivo da otimização de impressão 3D

| Peça | x_t | y_t | θ |
|------|--------|--------|----------|
| 1 | -63,27 | -35,56 | 3,78 |
| 2 | -36,23 | -37,49 | 3,59 |
| 3 | -46,37 | -8,73 | 5,60 |
| 4 | -21,17 | 1,99 | 2,11 |

Tabela 2. Valores ótimos para as translações e rotações das peças.

No caso do problema da internet 5G, o valor de penalização é fixo para cada torre que não seja posicionada numa zona permitida (equação 13).

$$P(\mathbf{x}, r_g) = r_g \cdot n((x_n, y_n) \notin A_{permitida}) \quad (13)$$

7 Resultados

Uma vez que os resultados destes problemas são desconhecidos, os parâmetros do algoritmo *Firefly* tiveram que ser ajustados e estudados com um problema com um mínimo conhecido, o Benchmark 2020, apresentado no apêndice A.

7.1 Impressão 3D

Os resultados foram obtidos com 4 objetos distintos uma cama quente de 200x200mm, tendo os testes sido efetuados com uma população de 75 e 10 iterações.

No gráfico da figura 3 é possível observar a evolução do custo e custo mínimo ao longo das avaliações, verificando-se a estabilização do custo mínimo. Observa-se também que nas avaliações há varias posições com custos extremamente elevados, o que se deve ao facto de essas posições não cumprirem as restrições impostas e serem penalizadas por tal.

Após 24528 avaliações da função objetivo, obteve-se o valor ótimo final de 18888,55, estando os valores finais de translação e rotação de cada peça na tabela 2.

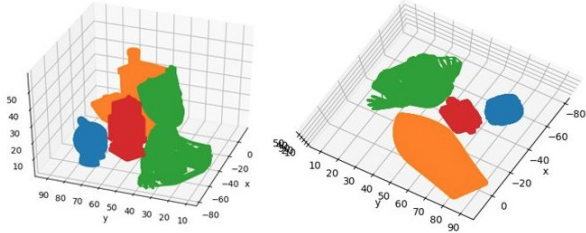


Figura 4. Peças dispostas de forma ótima.

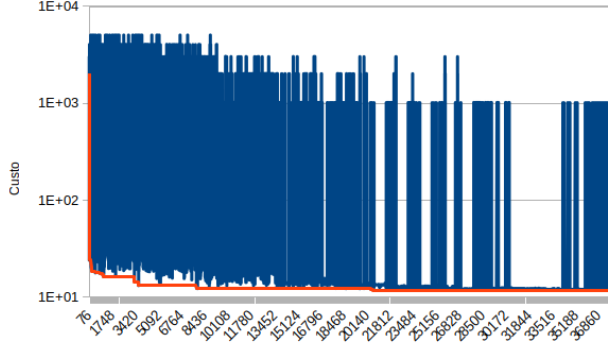


Figura 5. Evolução do custo mínimo da função objetivo da otimização da distribuição de torres 5G

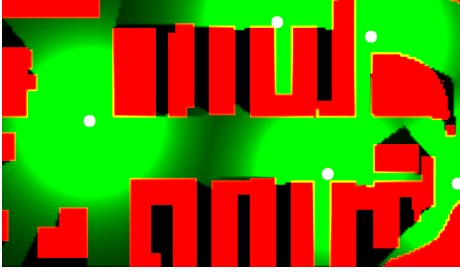


Figura 6. Disposição ótima de 5 torres 5G no espaço.

7.2 Internet 5G

Após efetuadas 37863 avaliações da função objetivo, com uma população de 100 e 10 iterações, obteve-se o gráfico da figura 5. O valor do custo ao longo das avaliações apresenta elevadas oscilações. Efetivamente, sendo a função objetivo extremamente sensível a mudanças das variáveis de otimização, a variação de uma delas por apenas um fator de 1 pode colocar a torre numa zona restrita, penalizando a função objetivo, o que provoca as oscilações observadas. No entanto, pode-se observar uma diminuição e estabilização do valor mínimo da função objetivo.

Efetuados vários ensaios, para 5 torres, para o mapa da figura 1, uma potencia de -33 dBW [7] e uma intensidade mínima de -67 dBm [6], obteve-se, após 37863 avaliações, o valor de menor custo de 11,73, associado à melhor posição apresentada na tabela 3. A figura 6 mostra, de forma mais visual, o resultado obtido.

| Torre | x_t | y_t |
|-------|-------|-------|
| 1 | 38 | 51 |
| 2 | 155 | 16 |
| 3 | 190 | 77 |
| 4 | 137 | 74 |
| 5 | 116 | 11 |

Tabela 3. Valores ótimos para as posições das torres 5G.

| n ^o Torre | custo |
|----------------------|-------|
| 3 | 18,41 |
| 5 | 11,73 |
| 7 | 9,39 |
| 9 | 8,83 |

Tabela 4. Valor da função objetivo em função do número de torres 5G.

Efetuando os mesmos testes, com a mesma população e número de iterações, para os casos de 3, 7 e 9 torres, os valores de custo obtidos são os apresentados na tabela 4.

8 Conclusões

Os resultados obtidos após a resolução destes dois problema de otimização não linear de escala intermédia foram bastantes satisfatórios.

No problema da impressão 3D, observa-se, como esperado, uma aproximação das peças, com as restrições cumpridas. No entanto, o gráfico da figura 3 mostra que os resultados não convergiram totalmente. Para tal acontecer, seria necessário que fossem feitas mais iterações. Outro aspeto a considerar é o facto que o valor em análise foi obtido com uma população de 75 indivíduos, em vez de 100, uma vez que os recursos computacionais disponíveis não permitiram um número de avaliações suficientemente elevado, com uma população de 100, o que permitiria obter um resultado melhor.

Para os vários valores obtidos na tabela 4, pode-se visualizar que o aumento número de torres aumenta a área coberta. No entanto, como a diferença entre a área coberta por 5 torres e por 9 torres é de apenas 2,9%, não compensaria investir em mais de 5 torres, daí terem sido apresentados resultados mais detalhados para 5 torres. Estes resultados foram bastante satisfatórios, sendo a área com cobertura 5G cerca de 90% da área total.

A Benchmark 2020/2021

A.1 Enquadramento do benchmark

O problema da minimização do peso de um redutor de engrenagens paralelas é um problema benchmark conhecido em otimização não-linear. Sendo um problema não linear de elevado grau de complexidade e com restrições, e sendo conhecida uma solução ótima, é ideal para testar o desempenho de novos algoritmos de otimização.

Com a resolução do Benchmark 2020, pretende-se estudar o desempenho do algoritmo *Firefly*, um algoritmo inspirado na natureza, baseado no conhecido Particle Swarm Optimization.

O problema em questão tem como objetivo minimizar o peso de um redutor de engrenagens paralelas de um andar, modificando sete variáveis [1], estando estas sujeitas a determinadas restrições.

A.1.1 Formulação do problema de otimização

O problema de otimização apresentado, pode ser formulado da seguinte forma [1]:

Procurar \mathbf{x} de modo a :

$$\begin{aligned} \text{minimizar } f(\mathbf{x}) = & 0.7854x_1x_2^2 \\ & \times (3.3333x_3^2 + 14.9334x_3 - 43.0934) \\ & - 1.508x_1(x_6^2 + x_7^2) \\ & + 7.477(x_6^3 + x_7^3) \\ & + 0.7854 * (x_4x_6^2 + x_5x_7^2), \end{aligned}$$

$$\text{sujeito a } g_1(\mathbf{x}) = \frac{27}{x_1x_2^2x_3} - 1 \leq 0,$$

$$g_2(\mathbf{x}) = \frac{397.5}{x_1x_2^2x_3} - 1 \leq 0,$$

$$g_3(\mathbf{x}) = \frac{1.93x_4^4}{x_2x_3x_6^4} - 1 \leq 0,$$

$$g_4(\mathbf{x}) = \frac{1.93x_4^4}{x_2x_3x_6^4} - 1 \leq 0,$$

$$g_5(\mathbf{x}) = \frac{\sqrt{\left(\frac{745x_4}{x_2x_3}\right)^2 + 16.9E6}}{110x_6^3} - 1 \leq 0,$$

$$g_6(\mathbf{x}) = \frac{\sqrt{\left(\frac{745x_5}{x_2x_3}\right)^2 + 157.5E6}}{85x_7^3} - 1 \leq 0,$$

$$g_7(\mathbf{x}) = \frac{x_2x_3}{40} - 1 \leq 0,$$

$$g_8(\mathbf{x}) = \frac{5x_2}{x_1} - 1 \leq 0,$$

$$g_9(\mathbf{x}) = \frac{x_1}{12x_2} - 1 \leq 0,$$

$$g_{10}(\mathbf{x}) = \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0,$$

$$g_{11}(\mathbf{x}) = \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0,$$

(14)

e com

$$2.6 \leq x_1 \leq 3.6,$$

$$0.7 \leq x_2 \leq 0.8,$$

$$17.0 \leq x_3 \leq 28.0,$$

$$7.3 \leq x_4 \leq 8.3,$$

$$7.8 \leq x_5 \leq 8.3,$$

$$2.9 \leq x_6 \leq 3.9,$$

$$5.0 \leq x_7 \leq 5.5$$

(15)

Tanto a função objetivo como as restrições são funções não lineares. Assim sendo, este problema é classificado com um problema de otimização não linear, de escala intermédia (uma vez que há mais de 5 variáveis de otimização), com variáveis de natureza determinística, seis contínuas e uma inteira.

A.1.2 Análise de sensibilidade

De modo a melhor compreender o problema, efetuou-se uma análise de sensibilidade. Com uma análise de sensibilidade "um-de-cada-vez", foi possível observar que a função objetivo é contínua, o que pode ajudar a seleccionar o método de otimização mais apropriado. Neste caso, seria possível a utilização de métodos baseados no gradiente da função, uma vez que a análise de sensibilidade mostra que a função é diferenciável.

A.2 Método de otimização e Implementação

A otimização deste problema pode ser feita com recurso a muitos algoritmos de otimização diferentes. O algoritmo em questão foi escolhido tendo em consideração os dois problemas referidos anteriormente, cuja solução é desconhecida e se pretende otimizar. Como analisado na secção 4, a função objetivo do problema da otimização das torre 5G apresenta degraus, não sendo possível calcular o seu gradiente.

Assim sendo, optou-se por não utilizar métodos baseados no gradiente da função. Métodos de procura direta também não são uma possibilidade, uma vez que são muito pouco eficientes em problemas com elevado grau de complexidade.

O método escolhido foi a heurística *Firefly*, já descrita na secção 6.

Tanto o algoritmo de otimização como o problema foram programados em linguagem de programação *Python*.

A.2.1 Penalizações externas

O algoritmo *Firefly* assume que o problema não tem restrições. Uma vez que o problema em estudo tem restrições de desigualdade, é necessário transformá-lo num problema sem restrições. Isto pode ser feito aplicando penalizações à função objetivo, a qual passa a ser:

Procurar \mathbf{x} de modo a :

$$\text{minimizar } F(\mathbf{x}, r_g) = f(\mathbf{x}) + P(\mathbf{x}, r_g), \quad (16)$$

$$\text{sujeito a } x_i^{min} \leq x_i \leq x_i^{max}, \quad (17)$$

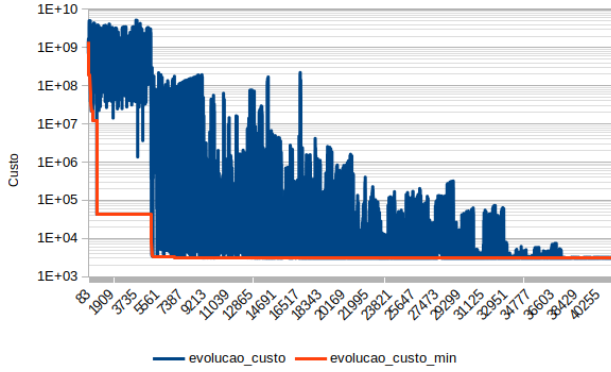


Figura 7. Custo em cada avaliação da função objetivo

sendo

$$P(\mathbf{x}, r_g) = r_g \cdot \sum_{j=1}^m [\max\{0, g_j(\mathbf{x})\}]^\gamma \quad (18)$$

A.3 Resultados

Para obter resultados fiáveis usando o algoritmo escolhido, foi necessário ajustar os parâmetros do algoritmo.

Como este algoritmo, para este problema, converge extremamente rápido, usou-se apenas 10 iterações para uma população de 100 indivíduos, efetuando, em média, 42 000 avaliações, uma vez que em cada iteração há um número de iterações aproximadamente igual a metade do quadrado do tamanho da população.

Os parâmetros escolhidos para o melhor resultado obtido foram: $\gamma = 1$ de forma a convergir mais depressa, menos iterações, compensado assim o valor do $\text{damp} = 0,4$, cuja redução diminui também a velocidade de convergência, $\alpha = 0,1$, de forma a que os valores aleatórios da distribuição normal não sejam demasiado distantes dos valores calculados na iteração anterior e, por fim, tem-se o valor de $\beta = 1,8$, que, como os outros valores, foi obtido através de vários testes.

Na figura 7, pode-se ver que, ao fim de 25000 a 30000 avaliações, já se tem valores relativamente próximos do valor ótimo e pode ser útil ou não fazer mais avaliações, dependendo da precisão requerida.

O melhor custo obtido foi de 2996,349406 com uma posição de $\mathbf{x} = [3,500001; 0,7; 17; 7,3; 7,8; 3,350216; 5,286684]$ para 41353 avaliações, bastante próximo da literatura que é de 2996,348165 com a posição de $\mathbf{x} = [3,499999; 0,7; 17; 7,3; 7,8; 3,350215; 5,286683]$, com 10^5 avaliações [1].

Utilizando o algoritmo *Firefly* na resolução de um problema de otimização *benchmark*, foi possível avaliar o seu desempenho e exatidão de resultados, além de ganhar uma maior compreensão dos parâmetros dos quais o algoritmo depende. Em comparação com o melhor valor obtido noutros estudos deste problema, o resultado obtido nestes ensaios apresenta um erro de apenas 0,0000416%. Trata-se de um algoritmo com uma muito boa exatidão de resultado para este problema. Além disso, sendo um algoritmo que converge depressa, caso seja necessário reduzir o custo computacional, será possível diminuir o número de avaliações (diminuindo a população ou o número de iterações) sem comprometer demasiado o resultado final.

Após efetuados 50 testes, sempre com os mesmos parâmetros, o resultado ótimo médio foi de 3008,16, 0,394% acima

do resultado mínimo obtido, sendo o desvio padrão deste conjunto de resultados de 80,4787, ou seja, 2,920% do melhor resultado. Com estes valores, pode-se concluir que se trata de um algoritmo de elevada robustez.

Referências

1. A. Andrade-Campos, J. Dias-de-Oliveira, Benchmark 2020: Weight minimisation of a speed reducer, *Non-Linear Optimisation in Engineering / MIEM.2020*, 2020.
2. Yang, Xin-She & Deb, Suash. (2010). Eagle Strategy Using Lévy Walk and Firefly Algorithms for Stochastic Optimization. *Studies in Computational Intelligence*.
3. Chakraborty, Amrita & Kar, Arpan. (2017). Swarm Intelligence: A Review of Algorithms
4. Firefly Algorithm - an overview | ScienceDirect Topics. (2020). Retrieved 14 December 2020, from <https://www.sciencedirect.com/topics/computer-science/firefly-algorithm>
5. WiraDKP/Firefly-Algorithm. (2020). Retrieved 17 December 2020, from <https://github.com/WiraDKP/Firefly-Algorithm>
6. Wifi, B., Wifi, E., Wifi, B. (2016). Best dBm Values for Wifi. Retrieved 14 January 2021, from <https://support.randomsolutions.nl/827069-Best-dBm-Values-for-Wifi>
7. Global 5G wireless deal threatens weather forecasts. (2019). Retrieved 14 January 2021, from <https://www.nature.com/articles/d41586-019-03609-x>