MAC0499 - Trabalho de formatura supervisionado

Minimizando objetos em problemas de empacotamento

Aluno: Francisco Nogueira C. Sobral [1]
Orientador: Ernesto G. Birgin [2]

Palavras-chave: Empacotamento de círculos, modelos, programação não linear.

Parte I
Trabalho técnico

1 Introdução

Problemas de empacotamento aparecem freqüentemente em situações da vida prática. Tentar empacotar um determinado número de produtos minimizando a quantidade de material necessário para a embalagem, ou transportar o maior número de produtos utilizando a menor frota de caminhões possível são exemplos que aparecem na área de logística e estão totalmente relacionados a empacotamento.

Há diversas maneiras para se resolver problemas de empacotamento. Encontrar uma solução ótima global para estes problemas foi provado ser um problema NP-completo, o que torna tal busca muito custosa em termos de tempo de processamento. Por esse motivo os métodos mais comuns utilizados para sua resolução são os heurísticos, que são em sua maioria rápidos, mas não garantem convergência para minimizadores globais.

Dentre estes métodos heurísticos, existem aqueles que utilizam modelos não lineares para resolver os problemas, procurando soluções através de algoritmos de programação não linear.

Para podermos estudar mais sobre a área de programação não linear e para testarmos a eficiência do programa ALGENCAN (um algoritmo para problemas de programação não linear), os modelos não lineares para problemas de empacotamento são o objeto de estudo deste trabalho. Dado um número fixo de itens circulares, o problema de encontrar o menor objeto no qual os itens podem ser empacotados é focado. Objetos circulares, triangulares, quadrados, retangulares e também *strips* são considerados.

[1] Departamento de Ciência da Computação - Instituto de Matemática e Estatística - Universidade de São Paulo. Este aluno foi auxiliado pelo CNPq (PIBIC/CNPq)

[2] Departamento de Computação - Instituto de Matemática e Estatística - Universidade de São Paulo

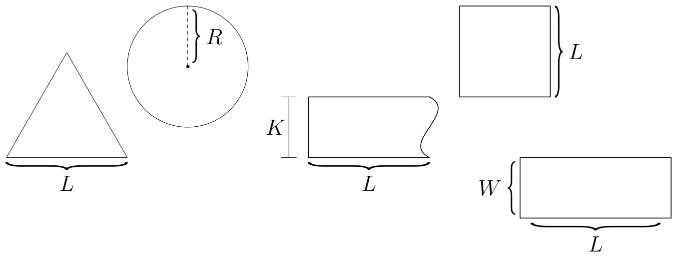


Figura 1: Objetos considerados nesse trabalho.

2 Conceitos e tecnologias estudadas

Muitos trabalhos abordam o problema de, dado um objeto e N itens idênticos, minimizar o tamanho do objeto no qual os N itens podem ser empacotados (ou, equivalentemente, maximizar o tamanho dos itens) utilizando técnicas de otimização contínua.

Em [17], o problema de empacotar itens circulares num objeto quadrado é abordado. A reformulação não linear do problema max-min de maximizar a menor distância entre os centros dos itens é resolvida utilizando o solver MINOS dentro do ambiente do GAMS. Uma estratégia multi-start é utilizada com o intuito de encontrar a solução global do problema de programação não linear. Experimentos numéricos mostram que a abordagem encontra as melhores soluções da literatura para até 30 itens, acha um resultado melhor para 15 itens e novos formatos de empacotamento para 28 e 29 itens.

Em [21] uma formulação diferente para o mesmo problema é apresentada. O modelo consiste em minimizar a função de energia $E = \sum_{1 \leq i < j \leq N} (\lambda / d_{ij})^m$, onde d_{ij} é a distância entre os pontos i e j, λ é um fator de escala e m é um inteiro positivo. As restrições que mantém os itens dentro do objeto são eliminadas pela mudança de variáveis $x_i = sin(\tilde{x}_i)$ e $y_i = sin(\tilde{y}_i)$. O problema irrestrito é resolvido por um algoritmo híbrido que mistura um algoritmo do tipo gradiente no início e um algoritmo do tipo Newton no fim. O resultado do algoritmo de otimização é ainda utilizado como ponto inicial para resolver um sistema de equações não-lineares, com o intuito de melhorar a precisão da solução. Foram resolvidos problemas para $21 \leq N \leq 50$ encontrando algumas soluções já conhecidas e melhorando outras.

O problema de empacotar círculos em círculos é abordado em [20]. Duas formulações não-lineares são introduzidas. Basicamente, ambas consistem em maximizar o raio dos itens circulares sujeito à não sobreposição entre os itens e ao fato dos itens pertencerem ao objeto. Um dos modelos utiliza coordenadas cartesianas e o outro coordenadas polares. Ambos modelos são resolvidos usando MINOS. Problemas com até 100 itens são resolvidos.

Um modelo para o empacotamento de itens circulares idênticos em triângulos eqüiláteros, baseado em maximizar a menor distância entre os centros dos itens, é apresentado em [19], onde dois métodos são apresentados para resolver o problema: a meta-heurística simulated annealing e um algoritmo do tipo quasi-Newton. Ajustes manuais são acrescentados às soluções obtidas pelos métodos de otimização.

Classes de valores de N para os quais são conhecidas soluções ótimas e outras classes onde suspeita-se conhecer os valores ótimos do problema de empacotar itens circulares em triângulos eqüiláteros são apresentados em [12].

O problema de empacotar itens circulares diferentes em *strips* (retângulos com uma das dimensões fixa) é abordado em [24]. O modelo consiste em minimizar a dimensão livre do retângulo sujeita à não sobreposição dos itens e a alocação dos itens no objeto. O método do gradiente reduzido e uma estratégia de restrições ativas são utilizados para resolver o modelo. Mais ainda, uma técnica específica é desenvolvida para escapar de minimizadores locais. Experimentos numéricos com problemas com entre 15 e 35 itens e uma comparação com um método do tipo Branch & Bound mostram as vantagens do método proposto.

Modelos não lineares também foram utilizados no problema de, dados um objeto fixo e itens idênticos com tamanho fixo, achar a maior quantidade de itens que podem ser empacotados. Em [9] é abordado o problema com itens circulares e em [8, 10] são abordados os problemas de empacotar itens retangulares com rotações ortogonais e com rotações livres, respectivamente.

Todas as abordagens mencionadas têm uma característica em comum. O número de restrições dos modelos não lineares é $O(N^2)$. Neste trabalho apresentamos formulações não lineares para o problema de empacotar itens circulares em diversos objetos. Apresentamos também uma estratégia baseada no problema dos N-corpos que permite avaliar os modelos com complexidade esperada $O(N)$. Em função disso, problemas com um número grande de itens podem ser abordados.

Os resultados de problemas de empacotamento podem ser interpretados como mínimos globais de problemas de programação não linear. Como os problemas de programação não linear eventualmente possuem muitos mínimos locais, é necessário resolvê-los utilizando vários pontos iniciais diferentes, uma técnica denominada *multi-start*.

3 Atividades realizadas

Nesta Seção descreveremos todo o trabalho realizado durante a iniciação científica. Primeiramente, é necessário apresentar duas nomenclaturas características de problemas de empacotamento: **item** é aquele que será empacotado e **objeto** é a forma que conterá os itens e terá sua área minimizada.

3.1 Modelos não lineares

3.1.1 Estrutura

Os modelos não lineares considerados neste trabalho possuem a seguinte estrutura:

Minimizar	a área do objeto
sujeito a	alocação dos itens no objeto,
	não sobreposição entre itens.

Esse modelo tem três partes principais: (i) a área do objeto a ser minimizada; (ii) as restrições de alocação dos itens no objeto; (iii) as restrições de não sobreposição entre itens. Claramente, a primeira depende apenas do objeto e a última apenas dos itens, enquanto que a restante depende dos dois, itens e objeto. Nesta Subseção mostraremos como modelar cada uma das partes para os casos de itens circulares e objetos circulares, triangulares, retangulares, quadrados e strips.

3.1.2 Sobreposição de itens

Seja $c_i = (c_i^x, c_i^y)^T \in \Re^2$ os centros dos N itens circulares com raios $r_i, i = 1, \ldots, N$. As restrições de não sobreposição entre itens, que são independentes da forma do objeto, podem ser modeladas como se segue:

$$d(c_i, c_j)^2 \geq (r_i + r_j)^2, \ \forall\, i \neq j, \tag{1}$$

ou

$$\sum_{i \neq j} \max\{0, (r_i + r_j)^2 - d(c_i, c_j)^2\}^2 = 0, \tag{2}$$

onde $d(\cdot, \cdot)$ é a distância Euclidiana. Ambas, (1) e (2), correspondem às distâncias entre os centros dos $N(N-1)/2$ pares de itens circulares. Claramente, (1) e (2) são equivalentes e qualquer conjunto de itens que satisfaça (1) ou (2) não possui sobreposição. Por motivos que serão explicados na Subseção 3.2 a equação (2) é a restrição de não sobreposição escolhida para a modelagem.

Vamos considerar agora a área do objeto e as restrições de alocação que dependem da forma do objeto. Se o objeto é circular, minimizar sua área é equivalente a minimizar seu raio R. Além do mais, o caso de um item circular com raio r_i pertencer ao objeto circular (o qual, sem perda de generalidade, pode ser considerado com centro na origem) pode ser modelado como $d(c_i, 0)^2 \leq (R - r_i)^2$ e $R \geq r_i$. No caso de um objeto quadrado, minimizar sua área é equivalente a minimizar seu lado L. Se assumirmos, sem perda de generalidade, que seu canto inferior esquerdo está fixo na origem, a alocação de itens circulares com raio r_i no objeto quadrado com lado L pode ser modelado por $r_i \leq c_i^x \leq L - r_i$ e $r_i \leq c_i^y \leq L - r_i$. Os casos em que o objeto é um triângulo, um retângulo ou um strip são análogos. A seguir são apresentados os modelos para cada problema, bem como o número de variáveis e restrições de cada um.

3.1.3 Círculos em um círculo

Dados N círculos $c_i = (c_i^x, c_i^y)$, de raios r_i, $i = 1, \ldots N$, desejamos encontrar o círculo de menor raio R que os contenha sem que se sobreponham. Sem perda de generalidade, consideramos que ele está centrado no ponto $(0, 0)$, como se pode ver na Figura 2.

Assim, o problema pode ser modelado como:

$$\begin{aligned} &min && R \\ &s.a. && \text{restrição de não sobreposição (2)} \\ & && {c_i^x}^2 + {c_i^y}^2 \leq (R - r_i)^2, \quad i = 1, \ldots N && (3) \\ & && R \geq r_{max}, && (4) \end{aligned}$$

onde $r_{max} = \underset{0 \leq i \leq N}{max} \{r_i\}$.

As restrições do tipo (3) e (4) são responsáveis por manter cada círculo dentro do círculo maior.

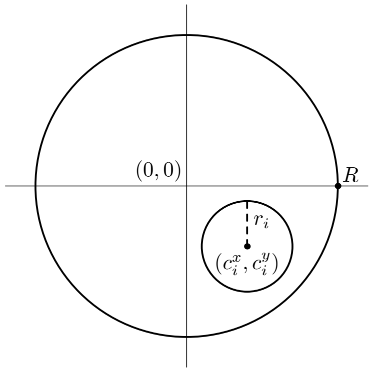


Figura 2: Círculo de centro (c_i^x, c_i^y) e raio r_i dentro de um outro círculo de centro na origem e raio R a ser minimizado.

O problema possui $2N + 1$ variáveis. Para garantir que nenhum círculo se sobreponha sobre outro, dois a dois, é necessária a restrição (2). As restrições da forma (3) são em número de N, que junto com (4) totalizam $N+1$ restrições.

3.1.4 Círculos em um *strip*

Dados N círculos $c_i = (c_i^x, c_i^y)$ com raios r_i, $i = 1, \ldots N$, desejamos encontrar o menor retângulo com um lado fixado (*strip*) que os contenha sem sobreposição. Encontrar tal retângulo significa encontrar a menor largura L de um retângulo com altura fixada K. Veja a Figura 3.

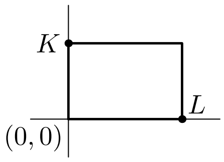


Figura 3: O *strip* considerado no modelo, com altura K fixada.

Existe uma aplicação deste problema na área têxtil. Basta considerar um rolo de tecido (que possui a altura fixada) e pensar que deseja-se utilizar a menor quantidade possível de tecido ao se fazer cortes nele. Isso é o mesmo que encontrar a menor largura de tecido que permite fazer cortes sem que eles se sobreponham (no nosso caso, os cortes são circulares).

Sem perda de generalidade, suporemos que o retângulo encontra-se no primeiro quadrante do plano cartesiano, com o vértice inferior esquerdo sobre a origem. Assim, o problema pode ser modelado como:

$$
\begin{array}{ll}
min & L \\
s.a. & \text{restrição de não sobreposição (2)} \\
 & r_i \leq c_i^x \leq L - r_i, \quad i = 1, \ldots, N \qquad (5) \\
 & r_i \leq c_i^y \leq K - r_i, \quad i = 1, \ldots, N \qquad (6) \\
 & L \geq 0 \qquad (7)
\end{array}
$$

As restrições (5) e (6) garantem que cada círculo c_i esteja dentro do *strip*. Embora a restrição (7) não seja necessária, visto que se $L < 2r_{max}$ o problema torna-se inviável, na prática percebemos, ao iniciá-la, uma melhoria nos resultados, impedindo que o método de resolução convirja para pontos estacionários inviáveis.

O problema possui $2N + 1$ variáveis. As restrições (5), (6) e (7) somam $2N + 2$ restrições.

3.1.5 Círculos em um quadrado

Dados N círculos $c_i = (c_i^x, c_i^y)$ de raios r_i, $i = 1, \dots N$, desejamos encontrar o menor quadrado de lado L que os contenha sem que se sobreponham dois a dois. Como no *strip*, sem perda de generalidade vamos supor que o quadrado está no primeiro quadrante do plano cartesiano com seu vértice inferior esquerdo sobre o ponto $(0, 0)$, como é mostrado na Figura 4.

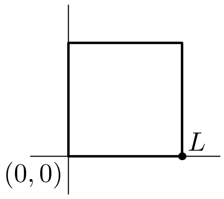


Figura 4: O quadrado considerado no modelo.

O problema foi modelado da seguinte maneira:

$$
\begin{array}{lll}
min & L & \\
s.a. & \text{restrição de não sobreposição (2)} & \\
 & r_i \leq c_i^x \leq L - r_i, \quad i = 1, \dots, N & \\
 & r_i \leq c_i^y \leq L - r_i, \quad i = 1, \dots, N & (8) \\
 & L \geq 0 & (9)
\end{array}
$$

Esse modelo assemelha-se ao do *strip*, a menos da restrição (8). Dessa maneira, o número de variáveis e de restrições desse problema é o mesmo que o do problema de colocar círculos em um *strip*, $2N + 1$ e $2N + 2$, respectivamente.

3.1.6 Círculos em um triângulo eqüilátero

Dados N círculos $c_i = (c_i^x, c_i^y)$ de raios r_i, $i = 1, \dots N$, desejamos encontrar o menor triângulo eqüilátero que os contenha sem que se sobreponham dois a dois. Consideramos o triângulo com um dos lados sobre o eixo x e o eixo y dividindo-o em duas partes simétricas (veja a Figura 5).

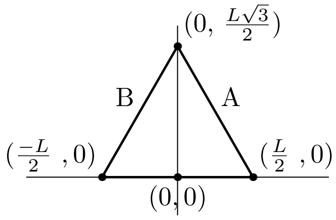


Figura 5: O triângulo escolhido

O modelo vem a seguir:

$$
\begin{aligned}
min \quad & L \\
s.a. \quad & \text{restrição de não sobreposição (2)} \\
& 6c_i^x + 2\sqrt{3}c_i^y - 3L + 4\sqrt{3}r_i \leq 0, \quad i = 1, \ldots, N && (10) \\
& -6c_i^x + 2\sqrt{3}c_i^y - 3L + 4\sqrt{3}r_i \leq 0, \quad i = 1, \ldots, N && (11) \\
& c_i^y \geq r_i, \quad i = 1, \ldots, N && (12) \\
& L \geq 0
\end{aligned}
$$

As restrições dos tipo (10) e (11), por não serem nada triviais, serão explicadas em detalhes. Consideremos as duas retas A e B da Figura 5, cujas equações são:

$$x + \frac{\sqrt{3}}{3}y = \frac{L}{2} \tag{13}$$

$$x - \frac{\sqrt{3}}{3}y = -\frac{L}{2} \tag{14}$$

Temos, então as seguintes restrições:

$$x + \frac{\sqrt{3}}{3}y \leq \frac{L}{2} \tag{15}$$

$$-x + \frac{\sqrt{3}}{3}y \leq \frac{L}{2} \tag{16}$$

Sejam $n_1 = \begin{bmatrix} 1 \\ \frac{\sqrt{3}}{3} \end{bmatrix}$ e $n_2 = \begin{bmatrix} -1 \\ \frac{\sqrt{3}}{3} \end{bmatrix}$ os vetores normais às retas (13) e (14) respectivamente (Figura 6), ou seja,

$$n_1^T x = \frac{L}{2} \quad \text{para todo x pertencente à (13)}$$

$$n_2^T x = \frac{L}{2} \quad \text{para todo x pertencente à (14)}$$

Um círculo c_i, de centro (c_i^x, c_i^y) e raio r_i, pertence à intersecção dos dois espaços afins se e somente se

$$(c_i^x, c_i^y) + r_i \frac{n_1}{||n_1||} \quad \text{satisfizer (15)}$$

$$(c_i^x, c_i^y) + r_i \frac{n_2}{||n_2||} \quad \text{satisfizer (16)}$$

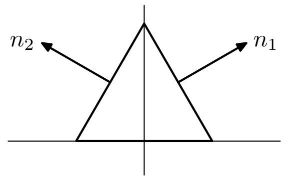


Figura 6: Os vetores normais n_1 e n_2

Calculando os valores, temos:

$$||n_1|| = \frac{2\sqrt{3}}{3} \quad \text{e} \quad ||n_2|| = \frac{2\sqrt{3}}{3}$$

o que nos dá

$$\frac{n_1}{||n_1||} = \begin{bmatrix} \frac{\sqrt{3}}{2} \\ \frac{1}{2} \end{bmatrix} \quad \text{e} \quad \frac{n_2}{||n_2||} = \begin{bmatrix} -\frac{\sqrt{3}}{2} \\ \frac{1}{2} \end{bmatrix}$$

Substituindo os pontos $(c_i^x, c_i^y) + r_i \frac{n_1}{||n_1||}$ e $(c_i^x, c_i^y) + r_i \frac{n_2}{||n_2||}$ nas desigualdades (15) e (16) respectivamente, chegamos às restrições:

$$\begin{aligned} 6c_i^x + 2\sqrt{3}c_i^y - 3L + 4\sqrt{3}r_i &\leq 0 \\ -6c_i^x + 2\sqrt{3}c_i^y - 3L + 4\sqrt{3}r_i &\leq 0 \end{aligned}$$

que, juntamente com a restrição (12), impede que os círculos fiquem fora do triângulo.

O número de restrições é $3N + 2$ e o número de variáveis é $2N + 1$.

3.1.7 Círculos em um retângulo

Nesse problema, dados N círculos $c_i = (c_i^x, c_i^y)$ de raios r_i, $i = 1, \ldots N$, queremos encontrar qual o retângulo de menor área que os contém sem sobreposição. Para modelar esse problema consideramos o retângulo no primeiro quadrante do plano cartesiano com o vértice inferior esquerdo na origem, como é mostrado na Figura 7.

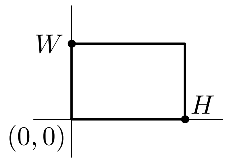


Figura 7: O retângulo considerado no modelo.

Segue o modelo do problema:

$$
\begin{aligned}
min \quad & WH \\
s.a. \quad & \text{restrição de não sobreposição (2)} \\
& r_i \leq c_i^x \leq W - r_i, \quad i = 1, \ldots, N && (17) \\
& r_i \leq c_i^y \leq H - r_i, \quad i = 1, \ldots, N && (18) \\
& W \geq 0 && (19) \\
& H \geq 0 && (20)
\end{aligned}
$$

Nas restrições, temos (17) e (18) que impedem os círculos de ficarem fora do retângulo. O número de variáveis tratado nesse problema agora é $2N + 2$ e o número de restrições é $2N + 3$.

3.2 Avaliação eficiente da sobreposição

O problema dos N-corpos consiste em calcular a força gravitacional entre N partículas no espaço tridimensional, onde cada partícula exerce uma força em todas as outras partículas, implicando em interações entre pares. Esse tipo de sistema físico aparece em diversos campos, como mecânica celestial, física de plasmas, mecânica dos fluidos e simulação de dispositivos semicondutores [15].

Apresentaremos a seguir uma abordagem baseada no problema dos N-corpos para reduzir a complexidade de avaliar a medida de sobreposição dos itens nos nossos problemas de empacotamento.

3.2.1 Método da Grade

As restrições de não sobreposição são em $O(N^2)$ e constituem a parte mais custosa para o cômputo do modelo. Na prática, isso inviabiliza a resolução de problemas grandes (acima de 100 círculos), pois requer muito tempo de CPU.

Vamos analisar novamente a restrição de sobreposição utilizada (2):

$$\sum_{i \neq j} \max\{0, (r_i + r_j)^2 - d(c_i, c_j)^2\}^2 = 0$$

Para o caso de itens idênticos, esta restrição tem uma grande vantagem na prática: nem todas as $O(N^2)$ distâncias precisam ser calculadas. Basicamente, se dois itens circulares k_1 e k_2 estão distantes um do outro, a contribuição do termo $\max\{0, (r_{k_1} + r_{k_2})^2 - d(c_{k_1}, c_{k_2})^2\}$ para a soma é nula. De fato, se os itens estão mais ou menos distribuídos no plano (como ocorre na prática), o número de pares que contribuem para a soma é linear. Além disso, os pares podem ser identificados (e sua distância, por conseqüência, calculada) em $O(N)$ operações. Esta estratégia tem sido usada com sucesso em problemas de conformação molecular, onde o número de itens pode ser muito grande (veja [18], entre outros).

O método se baseia numa idéia simples para realizar a verificação de não sobreposição entre itens: ao invés de calcular a sobreposição de um item com todos os outros, apenas os itens "vizinhos" são considerados para o cálculo, ou seja, aqueles com os quais há alguma chance de sobreposição.

Para dispor de forma eficiente dos vizinhos de um determinado item, constroe-se uma partição do espaço dividida em regiões idênticas de forma tal que só itens que ocupam a mesma região ou regiões adjacentes têm chance de se sobrepor. No caso de itens circulares idênticos com raio r, basta definir regiões quadradas de lado $2r$ como mostra a figura 8. Nesse caso dizemos que um item circular "ocupa" uma região se o seu centro pertence à região.

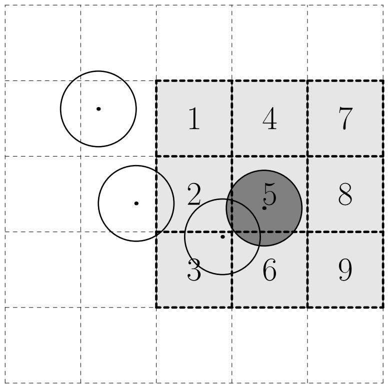


Figura 8: Os círculos cujos centros estão fora das regiões numeradas garantidamente não se sobreporão ao círculo destacado.

Dessa maneira, basta percorrer as regiões ocupadas da partição e verificar a sobreposição entre itens de regiões adjacentes. Esse método de construir a partição aliado à restrição de não sobreposição apresentada na Subseção 3.1 torna a avaliação extremamente eficiente.

3.2.2 Implementação

O tamanho de cada região da partição é dado por $\delta = max\{r_k\}$. Como podemos ver na Figura 9. Claramente, se dois itens não são adjacentes, não há chance de sobreposição, e a propriedade pedida é válida.

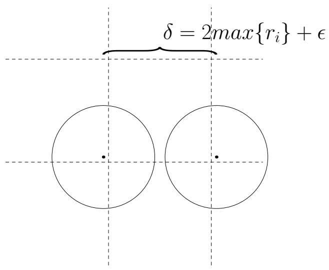


Figura 9: Dimensão escolhida para cada região.

Para determinarmos o tamanho da partição, calculamos um *upper bound* para a área do objeto considerado (valor para o qual certamente todos os itens são empacotados sem sobreposição) e determinamos o número de regiões necessárias para contê-lo. Além disso, fazemos as bordas da partição serem ilimitadas, para que qualquer objeto do espaço possa ser mapeado para alguma região.

Com a partição criada, para cada item, sua respectiva região é encontrada. A verificação de sobreposição é feita, então, somente entre itens localizados em regiões adjacentes.

O método começa com a partição vazia. Cada vez que a sobreposição deve ser avaliada, suas regiões são preenchidas. Cada região possui uma lista dos itens cujos centros estão na região. Há também uma lista de regiões não vazias que torna o percorrimento da partição mais eficiente.

Dessa maneira, um algoritmo para a avaliação eficiente da não sobreposição foi implementado.

0) $sobrep := 0$

1) Para cada item $i = 1, \ldots, N$ faça:

   1.1) Determine Reg_i, a região de i

   1.2) Adicione i à lista de itens cujos centros estão em Reg_i

   1.3) Se i for o primeiro item desta região, adicione-a à lista de regiões não-vazias

2) Para cada região não-vazia R e para cada item i em R faça:

   2.1) Para cada item j em uma região adjacente a R faça:

      2.1.1) $sobrep := sobrep + \max\{0, (r_i + r_j)^2 - d(c_i, c_j)^2\}$

3) Limpe a estrutura e devolva $sobrep$

4 Resultados e produtos obtidos

Com todos os modelos criados, experimentos numéricos foram realizados. Todos foram implementados em Fortran77, com a opção "-O4" e sistema operacional Linux. Um computador AMD Athlon 64 3200+ com 2202.87MHz e 1GB de memória RAM foi utilizado. Os resultados foram comparados com outros obtidos na literatura. Como critério de parada escolhemos 10000 tentativas de um mesmo problema (cada uma com um ponto inicial aleatório) ou 6 horas de uso de CPU.

Para resolver os problemas propostos foi utilizado um algoritmo baseado em Lagrangeanos Aumentados denominado ALGENCAN. Entre muitas outras características, este algoritmo permite que usemos informações de segunda ordem (hessianas) para auxiliar na convergência a minimizadores locais. Por esse motivo, as derivadas primeira e segunda de cada modelo foram calculadas e adicionadas ao código. Como estamos buscando minimizadores globais, uma estratégia de resolver um mesmo problema com muitos pontos iniciais aleatórios (denominada *multi-start*) foi utilizada.

Para podermos comparar os resultados, todos os testes foram realizados com itens circulares de raio unitário. As tabelas que seguem mostram alguns dos resultados obtidos. A primeira coluna indica a instância do problema resolvido, a segunda o menor valor encontrado para o problema de minimizar a área do objeto, a terceira, o mesmo resultado, mas associado ao problema (equivalente ao tratado neste trabalho) de maximizar o raio dos itens em um objeto de tamanho unitário, na quarta a densidade ($\sum$ área dos itens / área do objeto), na quinta coluna o número de tentativas para chegar à solução e na sexta o tempo (em segundos) gasto para encontrá-la. No caso do triângulo equilátero, não há a coluna associada ao problema equivalente.

Os resultados apresentados nas Tabelas 1 e 2, para empacotamento de itens circulares em círculos e em quadrados respectivamente, foram comparados com os encontrados em [23]. Para todos os testes os valores encontrados foram os mesmos.

Para o caso do empacotamento de itens círculares em triângulos equiláteros, comparamos apenas para $N = 1, \ldots, 15$ com aqueles apresentados em [11]. Para os outros valores, nada foi encontrado.

No empacotamento de itens circulares em strips, a única comparação encontrada ocorreu em [24], onde considera-se itens circulares de raios distintos entre si. A solução encontrada por nossa abordagem é mostrada na Figura 10. Nesse caso o valor obtido foi 18.0106808 contra 17.4912 encontrado pelos autores.

Para o empacotamento de itens circulares em retângulos os resultados não puderam ser comparados. As Figuras 11–19 apresentam figuras das soluções encontradas para objeto circular, triangular e quadrado.

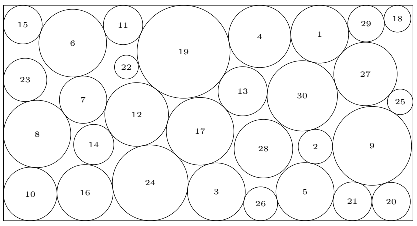


Figura 10: Empacotamento de 30 itens circulares de raios distintos em um strip com altura fixada $9,5$ e comprimento ótimo encontrado de 18.0106808.

Círculos em um círculo

# itens	min R	max r	densidade	# trials	time
1	1.0000000	1.0000000	1.0000000	1	0.00
2	1.9992025	0.5001995	0.5003990	1	0.01
3	2.1539074	0.4642725	0.6466469	2	0.00
4	2.4133556	0.4143608	0.6867795	1	0.00
5	2.7007299	0.3702703	0.6855004	6479	277.98
6	2.9992631	0.3334152	0.6669943	1	0.00
7	2.9993452	0.3334061	0.7781174	1	0.00
8	3.3036924	0.3026916	0.7329777	1956	60.31
9	3.6124088	0.2768236	0.6896817	17	0.21
10	3.8122937	0.2623093	0.6880615	1	0.01
11	3.9228482	0.2549168	0.7148085	1	0.02
12	4.0289172	0.2482056	0.7392725	1	0.02
13	4.2354639	0.2361016	0.7246718	1	0.02
14	4.3277753	0.2310656	0.7474784	1	0.02
15	4.5207092	0.2212042	0.7339697	3	0.10
16	4.6147706	0.2166955	0.7513110	2	0.06
17	4.7916271	0.2086974	0.7404281	1	0.04
18	4.8631145	0.2056295	0.7611031	1	0.04
19	4.8629412	0.2056369	0.8034439	1	0.04
20	5.1216840	0.1952483	0.7624378	1	0.03
21	5.2515062	0.1904216	0.7614678	1	0.08
22	5.4390865	0.1838544	0.7436537	20	1.71
23	5.5445274	0.1803580	0.7481674	1	0.16
24	5.6508911	0.1769632	0.7515837	123	17.55
25	5.7520698	0.1738505	0.7555996	7	2.79
26	5.8274546	0.1716015	0.7656240	6	1.06
27	5.9057954	0.1693252	0.7741176	4	0.83
28	6.0143379	0.1662693	0.7740738	3	0.61
29	6.1377932	0.1629250	0.7697922	23	4.64
30	6.1970919	0.1613660	0.7811697	1	0.22
31	6.2907438	0.1589637	0.7833533	96	23.64
32	6.4287657	0.1555509	0.7742742	40	11.83
33	6.4860429	0.1541772	0.7844302	50	15.85
34	6.6105133	0.1512742	0.7780519	5	1.95
35	6.6968333	0.1493243	0.7804212	37	14.53
36	6.7459650	0.1482368	0.7910689	2	0.62
37	6.7579417	0.1479741	0.8101638	7	2.08
38	6.9611797	0.1436538	0.7841839	6	2.19
39	7.0571914	0.1416994	0.7830704	7	2.93
40	7.1231248	0.1403878	0.7883497	7	2.86
41	7.2592805	0.1377547	0.7780306	132	67.17
42	7.3462124	0.1361246	0.7782557	260	151.14
43	7.4192211	0.1347850	0.7811813	8	4.92
44	7.4972874	0.1333816	0.7827883	79	50.52
45	7.5722837	0.1320606	0.7847996	988	699.06
46	7.6494233	0.1307288	0.7861410	167	126.47
47	7.7233644	0.1294773	0.7879249	99	78.72
48	7.7906959	0.1283582	0.7908402	29	24.51
49	7.8863814	0.1268009	0.7878445	123	109.40
50	7.9468092	0.1258367	0.7917434	91	90.57

Tabela 1: 1–50 itens circulares de raio 1 em um objeto circular.

Círculos em um quadrado

# itens	min L	max r	densidade	# trials	time
1	2.0000000	0.5000000	0.7853982	1	0.01
2	3.4095563	0.2932933	0.5404856	1	0.00
3	3.9271795	0.2546357	0.6110963	1	0.00
4	3.9992029	0.2500498	0.7857113	1	0.01
5	4.8271618	0.2071611	0.6741184	1	0.00
6	5.3270016	0.1877229	0.6642560	1	0.00
7	5.7308183	0.1744952	0.6695988	1	0.00
8	5.8626951	0.1705700	0.7312153	22	0.20
9	5.9990345	0.1666935	0.7856510	2	0.00
10	6.7591076	0.1479485	0.6876559	17	5.09
11	7.0210505	0.1424288	0.7010328	5	0.03
12	7.1437936	0.1399816	0.7387089	9	0.11
13	7.4630289	0.1339939	0.7332684	7	0.17
14	7.7309165	0.1293508	0.7358952	35	0.71
15	7.8626416	0.1271837	0.7622618	48	1.09
16	7.9989557	0.1250163	0.7856033	1	0.01
17	8.5309004	0.1172209	0.7338530	71	2.22
18	8.6552029	0.1155374	0.7548625	3	0.10
19	8.9061303	0.1122822	0.7525327	11	0.62
20	8.9768656	0.1113975	0.7797052	10	0.46
21	9.3563091	0.1068798	0.7536333	19	1.31
22	9.4630757	0.1056739	0.7718056	8	0.52
23	9.7263184	0.1028138	0.7638019	1220	174.07
24	9.8625920	0.1013932	0.7751379	25	2.02
25	9.9989094	0.1000109	0.7855695	15	1.08
26	10.3761299	0.0963750	0.7586692	17	1.45
27	10.4788069	0.0954307	0.7724849	5	0.47
28	10.6733915	0.0936909	0.7721524	5495	1100.32
29	10.8142428	0.0924706	0.7790326	2	0.23
30	10.9072688	0.0916820	0.7922078	30	4.28
31	11.1915357	0.0893532	0.7775569	19	3.12
32	11.3803879	0.0878705	0.7762216	9	1.34
33	11.4626236	0.0872401	0.7890341	1	0.08
34	11.7262981	0.0852784	0.7767959	137	30.22
35	11.8625591	0.0842988	0.7813779	75	21.50
36	11.9988793	0.0833411	0.7855449	26	5.80
37	12.1812037	0.0820937	0.7833777	379	98.42
38	12.2380586	0.0817123	0.7970919	88	54.84
39	12.2887513	0.0813752	0.8113327	2	0.47
40	12.6267385	0.0791970	0.7881838	16	5.31
41	12.7456259	0.0784583	0.7928872	7	2.49
42	12.8518647	0.0778097	0.7988530	24	7.69
43	13.0966666	0.0763553	0.7875838	7140	3459.99
44	13.1941865	0.0757910	0.7940307	201	139.51
45	13.3803861	0.0747363	0.7896327	520	403.40
46	13.4630603	0.0742773	0.7972970	49	19.26
47	13.6751048	0.0731256	0.7895623	339	227.82
48	13.8047808	0.0724387	0.7912835	1230	961.20
49	13.9472638	0.0716987	0.7913488	2704	2205.54
50	14.0083593	0.0713859	0.8004705	615	494.65

Tabela 2: 1–50 itens circulares de raio 1 em um objeto quadrado.

Círculos em um triângulo eqüilátero

# itens	min L	densidade	# trials	time
1	3.4641016	0.6045998	1	0.00
2	5.4567072	0.4873242	1	0.00
3	5.4596287	0.7302041	1	0.01
4	6.9208148	0.6058914	1	0.00
5	7.4575602	0.6522675	1	0.00
6	7.4598010	0.7822508	1	0.01
7	8.9185234	0.6385007	1	0.01
8	9.2864554	0.6730375	3	0.04
9	9.4582257	0.7299152	1	0.01
10	9.4594504	0.8108069	2	0.02
11	10.7212929	0.6943007	3	0.04
12	10.9216644	0.7298823	6	0.11
13	11.3989499	0.7258768	3	0.07
14	11.4587786	0.7735718	1	0.02
15	11.4592465	0.8287592	1	0.02
16	12.7047059	0.7191840	10	0.34
17	12.9207553	0.7387923	4	0.19
18	13.2841101	0.7400428	31	2.11
19	13.4398487	0.7631573	1	0.12
20	13.4587943	0.8010635	1	0.03
21	13.4591128	0.8410768	1	0.04
22	14.6035984	0.7484317	10	0.67
23	14.8749493	0.7541645	29	2.75
24	14.9219595	0.7820036	57	8.46
25	15.2871828	0.7761298	6	2.06
26	15.4502342	0.7902281	12	0.73
27	15.4586419	0.8197291	1	0.07
28	15.4590185	0.8500480	2	0.20
29	16.5974824	0.7637704	38	5.09
30	16.7229142	0.7782992	46	9.12
31	16.9216049	0.7854668	4	1.84
32	17.2395055	0.7811773	66	32.71
33	17.3975076	0.7910230	7	3.14
34	17.4523699	0.8098775	14	1.79
35	17.4586628	0.8330965	1	0.14
36	17.4589482	0.8568713	1	0.11
37	18.5222010	0.7824666	9	2.33
38	18.7200442	0.7867180	6	1.57
39	18.9099779	0.7912830	6	2.04
40	18.9220081	0.8105406	8	4.43
41	19.2862185	0.7997218	22	24.97
42	19.3994632	0.8096906	126	148.66
43	19.4546051	0.8242764	18	4.79
44	19.4585837	0.8431007	1	0.18
45	19.4588940	0.8622346	1	0.30
46	20.5182266	0.7927339	44	28.89
47	20.6965270	0.7960716	72	40.98
48	20.8749500	0.7991708	4	2.82
49	20.9218316	0.8121681	33	37.44
50	21.2378873	0.8042603	8	7.22

Tabela 3: 1–50 itens circulares de raio 1 em um triângulo eqüilátero.

5 Conclusões

Nesse trabalho foram apresentados modelos para o empacotamento de itens circulares em objetos circulares, retangulares, quadrados, triangulares e *strip*s.

Uma abordagem que visa diminuir a complexidade para a avaliação da restrição de não sobreposição foi utilizada e implementada. Testes numéricos foram realizados e os resultados foram comparados aos encontrados na literatura, mostrando a eficiência da nova abordagem. O tempo computacional gasto para resolver os modelos também foi reduzido, permitindo que problemas com um grande número de itens pudessem ser resolvidos.

Devido à generalidade dos modelos, a extensão para três dimensões torna-se trivial e será o proximo passo do trabalho.

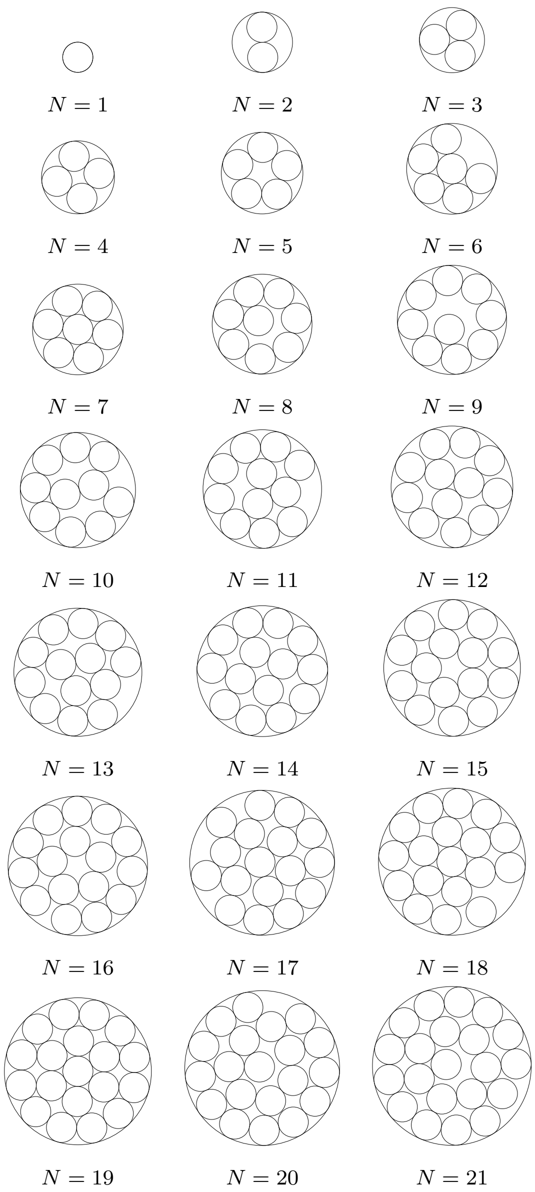


Figura 11: 1–21 círculos em uma circunferência

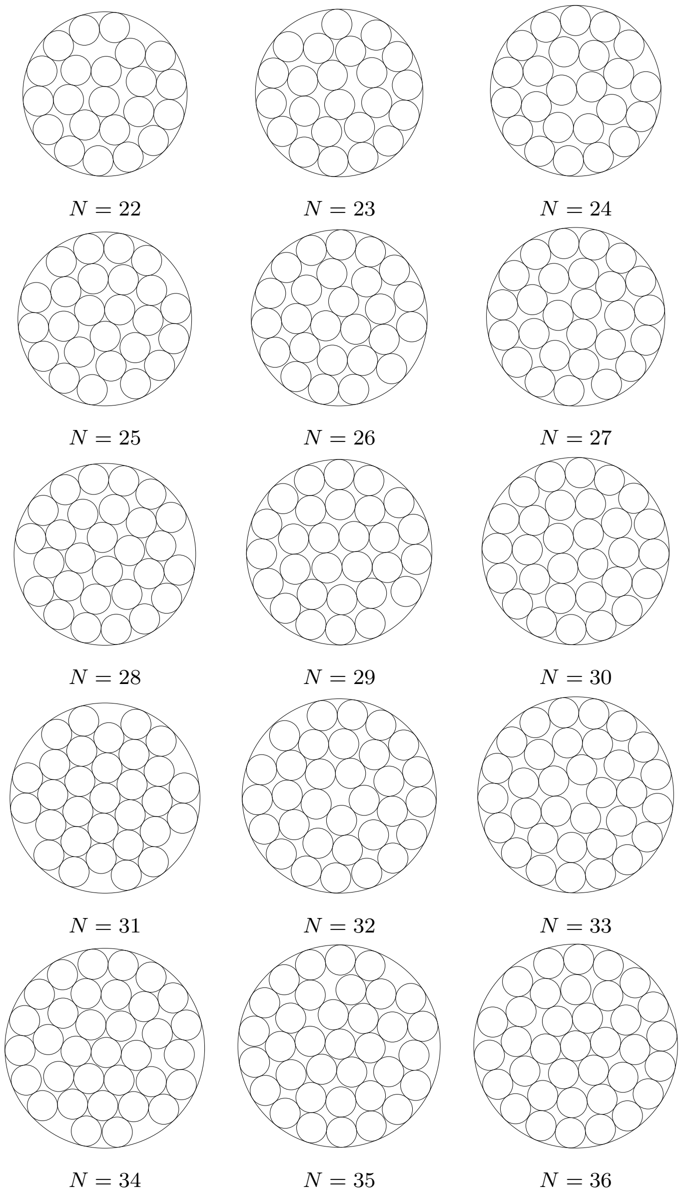
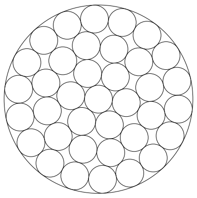
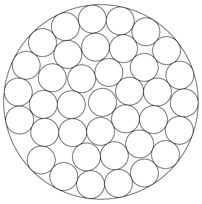


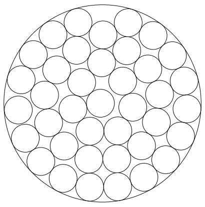
Figura 12: 22–36 círculos em uma circunferência



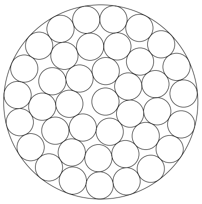
$N = 37$



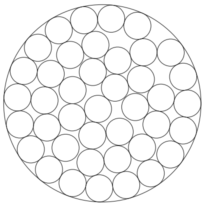
$N = 38$



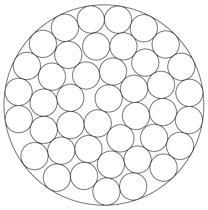
$N = 39$



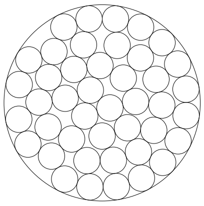
$N = 40$



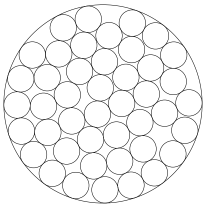
$N = 41$



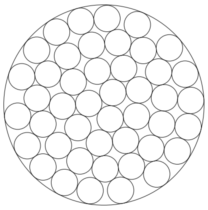
$N = 42$



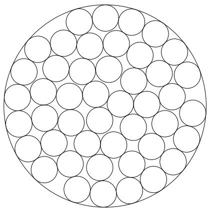
$N = 43$



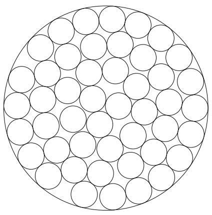
$N = 44$



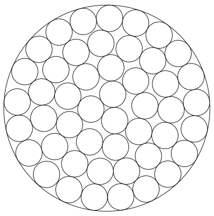
$N = 45$



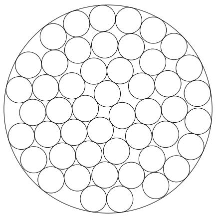
$N = 46$



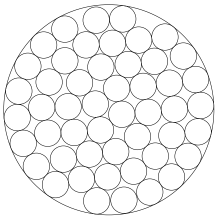
$N = 47$



$N = 48$



$N = 49$



$N = 50$

Figura 13: 37–50 círculos em uma circunferência

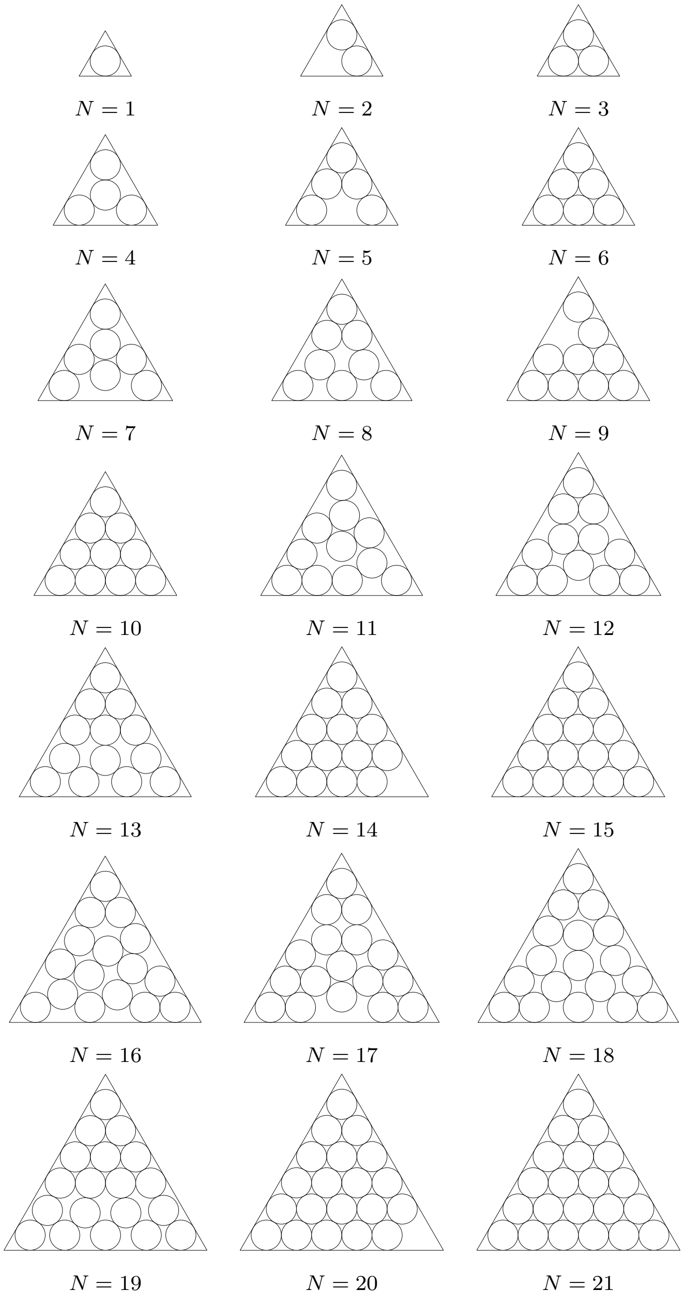


Figura 14: 1–21 círculos em um triângulo

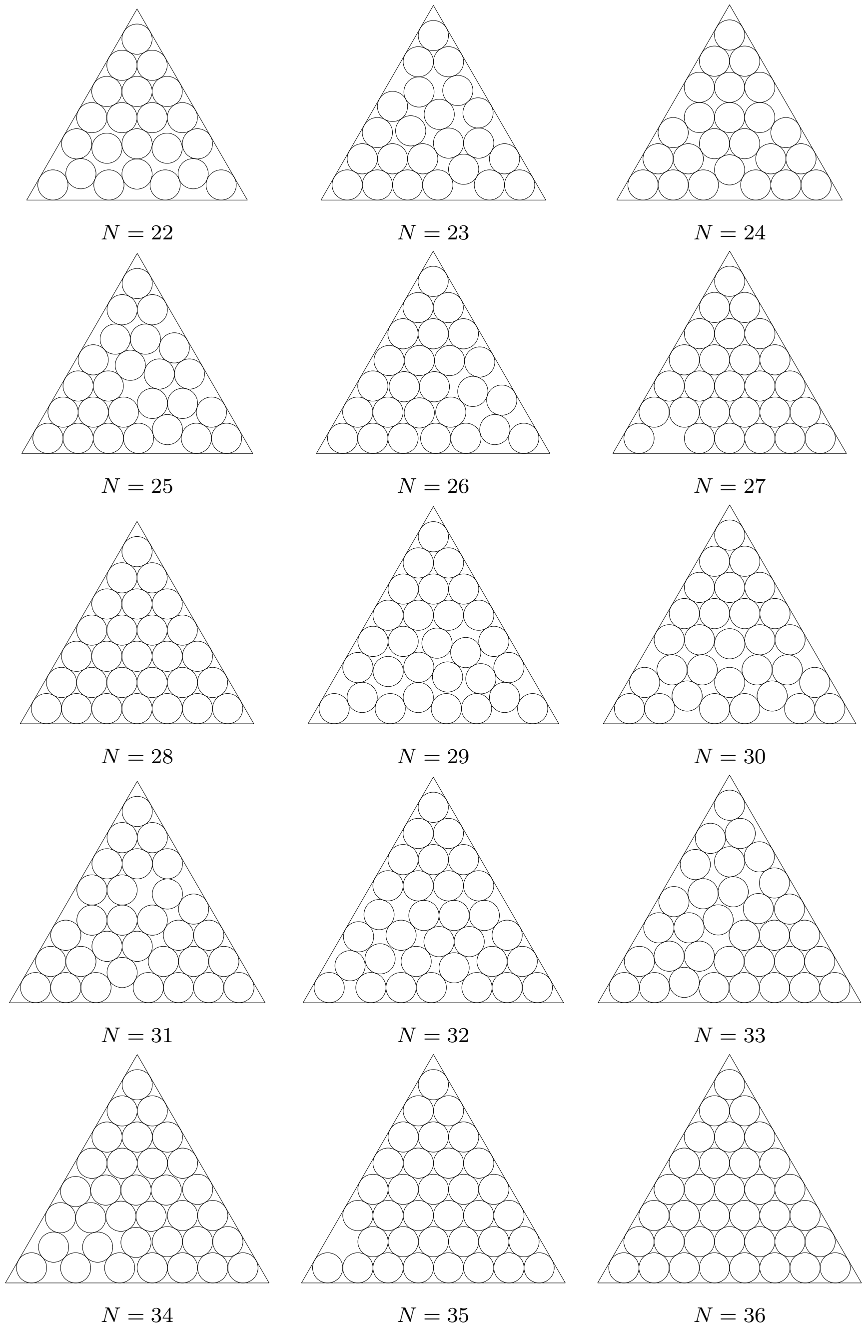


Figura 15: 22–36 círculos em um triângulo

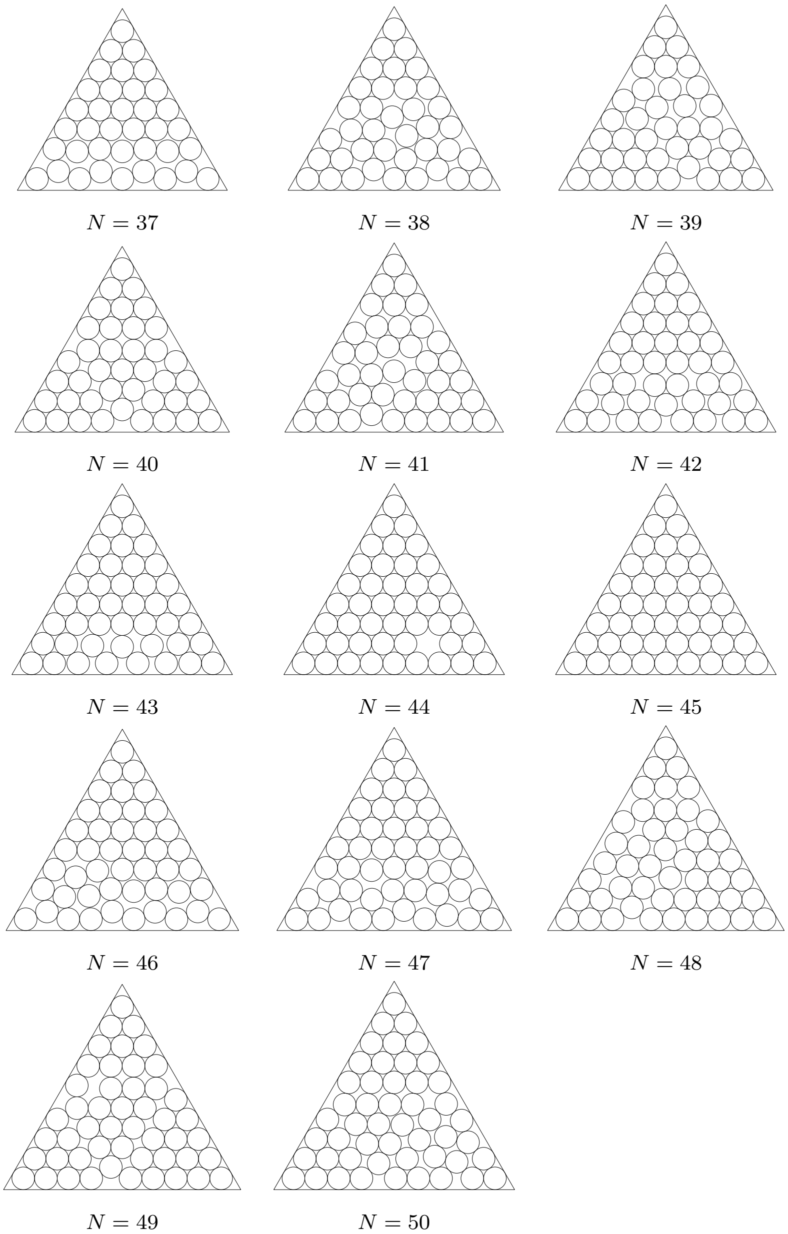


Figura 16: 37–50 círculos em um triângulo

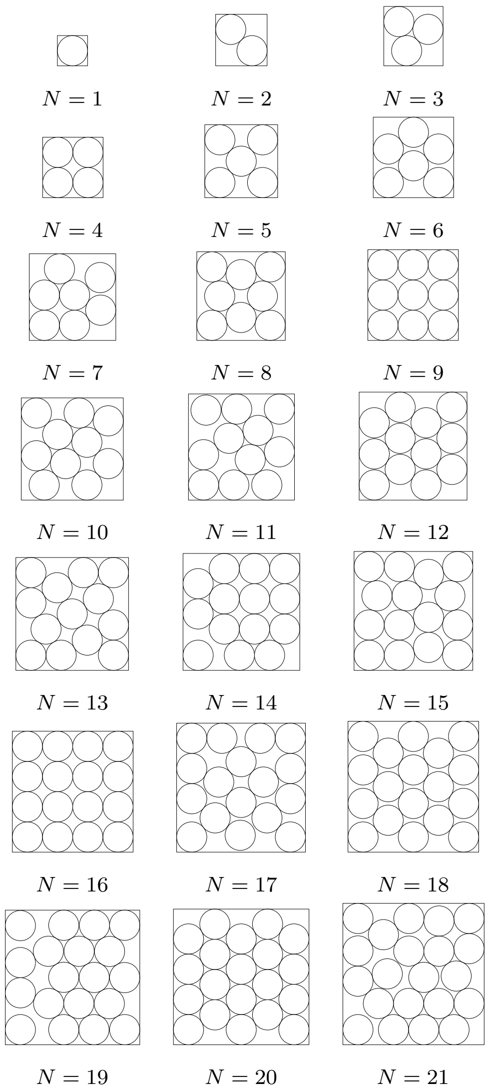
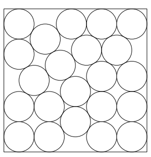
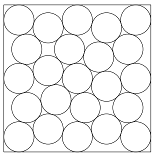


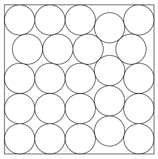
Figura 17: 1–21 círculos em um quadrado



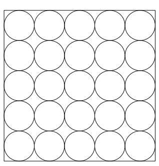
$N = 22$



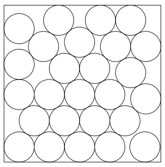
$N = 23$



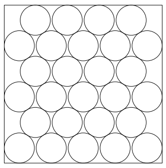
$N = 24$



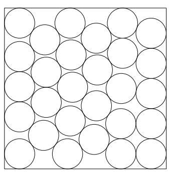
$N = 25$



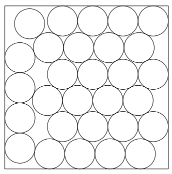
$N = 26$



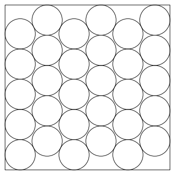
$N = 27$



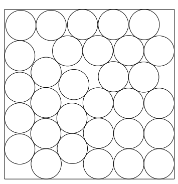
$N = 28$



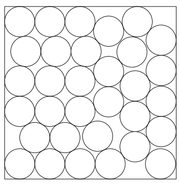
$N = 29$



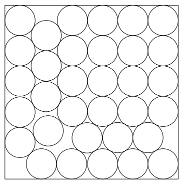
$N = 30$



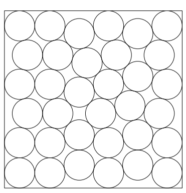
$N = 31$



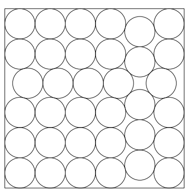
$N = 32$



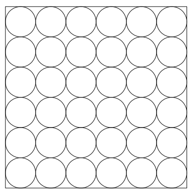
$N = 33$



$N = 34$



$N = 35$



$N = 36$

Figura 18: 22–36 círculos em um quadrado

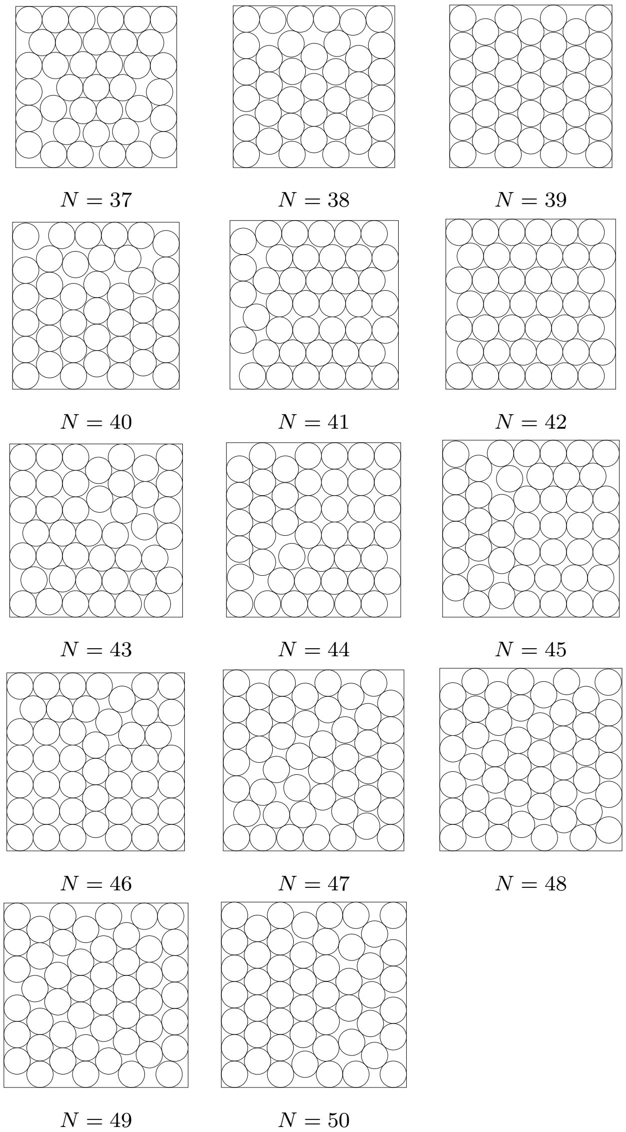


Figura 19: 37–50 círculos em um quadrado

Referências

[1] R. ANDREANI et al, Augmented Lagrangian methods under the Constant Positive Linear Dependence constraint qualification, submitted, 2004. Disponível em *http://www.ime.usp.br/˜egbirgin/*. Acesso em: Março de 2005.

[2] R. ANDREANI et al, On Augmented Lagrangian methods with general lower-level constraints, submitted, 2005. Disponível em *http://www.ime.usp.br/˜egbirgin/*. Acesso em: Março de 2005.

[3] A. W. Appel, An Efficient Program for Many-Body Simulations, *SIAM Journal on Scientific and Statistical Computing* v.6, n.1, pp 85–103, 1985.

[4] J. Barnes and P. Hut, A Hierarchical O(n log n) force calculation algorithm, *Nature*, 324, pp. 446–449, December 1986.

[5] E. G. Birgin, R. Castillo and J. M. Martínez, Numerical comparison of Augmented Lagrangian algorithms for nonconvex problems, *Computational Optimization and Applications*, to appear.

[6] E. G. Birgin, J. M. Martínez, A box constrained optimization algorithm with negative curvature directions and spectral projected gradients, *Computing [Suppl]* 15, pp. 49–60, 2001.

[7] E. G. Birgin and J. M. Martínez, Large-scale active-set box-constrained optimization method with spectral projected gradients, *Computational Optimization and Applications* 23, pp. 101–125, 2002.

[8] E. G. Birgin, J. M. Martínez, W. F. Mascarenhas and D. P. Ronconi, Method of Sentinels for Packing Objects whitin Arbitrary Regions, submitted.

[9] E. G. Birgin, J. M. Martínez and D. P. Ronconi, Optimizing the Packing of Cylinders into a Rectangular Container: A Nonlinear Approach, *European Journal of Operational Research* 160, pp. 19–33, 2005.

[10] E. G. Birgin, J. M. Martínez, F. H. Nishihara and D. P. Ronconi, Orthogonal packing of rectangular items within arbitrary regions by nonlinear optimization, submitted.

[11] E. Friedman, Erich's Packing Center, website (last updated used dated 10 november 2004) *http://www.stetson.edu/˜efriedma/packing.html*

[12] R. L. Graham and B. D. Lubachevsky, Dense packings of equual disks in an equilateral triangle: from 22 to 34 and beyond, *Electronic Journal of Combinatorics* 2, A1, 39 pp., 1995.

[13] L. Greengard and V. Rokhlin, A Fast Algorithm for Particle Simulations, *Journal of Computational Physics* v.73, n.2, pp. 325–348, December 1987.

[14] L. Greengard and V. Rokhlin, The rapid evaluation of potential fields in three dimensions, *Lecture notes in mathematics* 1360, pp. 121–141, 1988.

[15] R. W. Hockney and J. W. Eastwood, *Computer Simulation using Particles*, McGraw Hill, New York, 1981.

[16] J. Katzenelson, Computational structure of the N-body problem, *Siam Journal on Scientific and Statistical Computing* v.10, n.4, pp. 787–815, July 1989.

[17] C. D. Maranas, C. A. Floudas and P. M. Pardalos, New results in the packing of equal circles in a square, *Discrete Mathematics* 142, pp 287–293, 1995.

[18] J. M. Martínez and L. Martínez, Packing optimization for automated generation of complex system's initial configurations for molecular dynamics and docking, *Journal of Computational Chemistry* 24, pp. 819–825, 2003.

[19] J. B. M. Melissen and P. C. Schuur, Packing 16, 17 or 18 circles in an equilateral triangle, *Discrete Mathematics* 145, pp 333–342, 1995.

[20] N. Mladenović, F. Plastria and D. Urošević, Reformulation descent applied to circle packing problems, *Computers & Operations Research*, volume 32, issue 9, september 2005, pp 2419–2434

[21] K. J. Nurmela and P. R. J. Östergård, Packing over 50 equal circles in a square, *Discrete & Computational Geometry* 18, pp. 111–120, 1997.

[22] R. B. Schnabel, A view of the limitations, opportunities, and challenges in parallel nonlinear optimization, *Parallel Computing* 21, pp. 875–905, 1995.

[23] E. Specht, Packomania, website, última atualização datada de 02 de Março de 2005, *http://hydra.nat.uni-magdeburg.de/packing/cci/cci.html*

[24] Y. G. Stoyan and G. Yaskov, A mathematical model and a solution method for the problem of placing various-sized circles into a strip, *European Journal of Operational Research* 156, pp 590–600, 2004.

[25] H. Wang, W. Huang, Q. Zhang and D. Xu, An improved algorithm for the packing of unequal circles within a larger containing circle, *European Journal of Operational Research* 141, pp 440–453, 2002.

[26] F. Zhao, *An 0(N) algorithm for three-dimensional n-body simulations*, Technical Report, AI-TR-995, MIT Al Lab, Cambridge, MA, 1987.

Parte II
Relação entre o projeto e o BCC

Procurei fazer iniciação científica com o objetivo de estudar mais profundamente áreas relacionadas com disciplinas que geraram interesse durante a graduação. As disciplinas relacionadas com álgebra linear, álgebra linear computacional e otimização sempre tiveram maior prioridade na hora da escolha das optativas eletivas. Dentre todas, as que considerei mais relevantes para o meu trabalho foram: MAC0122 (Princípios de Desenvolvimento de Algoritmos), MAT0139 (Álgebra Linear para Computação), MAT0121 (Cálculo Diferencial e Integral II), MAT0221 (Cálculo Diferencial e Integral IV), MAC0338 (Análise de Algoritmos), MAC0315 (Programação Linear), MAC0427 (Programação Não Linear) e MAC0300 (Álgebra Linear Computacional). Uma disciplina, que não se relaciona muito com as acima citadas, mas que considero como uma das mais importantes que assisti do BCC, foi MAC0316 (Conceitos Fundamentais em Linguagens de Programação).

Antes da iniciação, porém, realizei uma monitoria para a disciplina de Programação Linear, para sentir um pouco de como é ter que estudar para algo que não vai ser avaliado. Um ano depois, terminei a disciplina de Programação Não Linear e escolhi que esta seria a área para uma iniciação científica.

Um dos primeiros desafios foi a escolha do tema, apesar de não ter levado muito tempo. Na hora parece sem importância, mas se a escolha for ruim a pesquisa sobre o assunto pode se tornar muito cansativa. No nosso caso, um tema inicial foi escolhido e ao longo do tempo fomos adaptando conforme íamos conhecendo tudo o que já tinha sido publicado e estudado sobre o assunto na literatura.

Em termos de pesquisa, a iniciação me rendeu inúmeras experiências no campo acadêmico. A procura de trabalhos já publicados na literatura é uma tarefa difícil e cansativa, mas necessária para que a pesquisa não produza resultados já conseguidos por outros autores, ou para coletarmos informações sobre a estrutura do problema e resultados comparativos. O tempo todo eu fui bastante cobrado quanto à produtividade, apesar de não poder deixar de lado as matérias da graduação, o que deu a impressão de ser um trabalho mesmo, com responsabilidades, datas de entrega, etc...

Uma grande dificuldade ocorreu com os testes realizados para o programa criado. Percebi que é muito importante ter certeza que certa parte de um programa está funcionando antes de partir para implementar outra. A falta dessa experiência fez com que os modelos tivessem que ser reescritos diversas vezes e, como para resolver os problemas o tempo gasto não era desprezível, esses erros levaram a uma grande perda de tempo.

Pelo motivo de quase todas as matérias que me interessaram serem teóricas, desenvolvi uma grande vontade de seguir para a área de pesquisa, ou a área acadêmica. A maior frustração de todas foi perceber como a pesquisa é pouco incentivada por empresas e pelo governo no Brasil. Muitos profissionais com doutorado têm dificuldade de encontrar emprego, devido ao seu alto grau de especialização. A carreira acadêmica, que era vista como a única alternativa, também está muito mal. As faculdades públicas sofrem com falta de pagamento e as particulares preferem contratar mestres do que doutores, pagando salários desproporcionais. Além disso, todos os estágios encontrados são somente na área de programação, e pedem que o aluno seja recém formado. Isso faz com que se crie um desânimo para fazer mestrado, por medo de haver falta de colocação no mercado.

Apesar disso, a continuação dos estudos será mantida. Pretendo fazer mestrado nessa área da iniciação (Otimização Contínua). O tema não foi relacionado com o estudado na iniciação científica, mas a escolha da área sim. Nos próximos anos pretendo ir em congressos para ver como são os profissionais nessa área e como ela está em outros lugares do Brasil e do mundo.

Com o orientador a relação foi muito profissional. O meu orientador foi o Ernesto Birgin e ele estava sempre presente quando eu precisa tirar dúvidas quanto à implementação, ou sobre a parte teórica. Sugeria mudanças e novas abordagens que melhoravam o desempenho da nossa aplicação. Quanto à qualidade dos relatórios, foi muito rígido, o que me fez ficar bastante crítico e me esforçar a aprender e escrever melhor os textos científicos. Aprendi a fazer revisões bibliográficas. Quando manifestei interesse em ir em congressos na área de nosso estudo, ele foi muito solícito, entrando em contato com os organizadores e procurando por órgãos de auxílio a estudantes. Além disso, também me aconselhou na escolha das optativas e até estimulou a fazer algumas disciplinas da pós-graduação como aluno especial.

A iniciação não foi um trabalho feito em grupo. Porém haviam outros orientandos com o Ernesto e criamos uma espécie de grupo. Além da iniciação pediram-nos para que ajudássemos a desenvolver uma interface entre o algoritmo para problemas de programação não linear ALGENCAN (o qual meu orientador participa do desenvolvimento) que estava escrito em Fortran77 e as demais linguagens utilizadas para resolver tais problemas. Essas interfaces foram (e ainda estão sendo) feitas por quase todos, e gerou uma grande interação entre todos os orientandos, uma coisa que eu nunca vi com outros professores. Tamanha foi a interação que quase todos foram juntos para um congresso de otimização que ocorreu em Julho deste ano.