A shape optimization approach to the problem of covering a two-dimensional region with minimum-radius identical balls

Antoine Laurain

Instituto de Matemática e Estatística, Universidade de São Paulo.

joint work with: Ernesto G. Birgin, Rafael Massambone, Arthur G. Santana

Concepción 06/2021

Packing of spheres and covering problem

- *Sphere packings, lattices and groups*
  J.H. Conway and N.J.A. Sloane.
- The packing and covering problems in the whole space are kind of dual.
- The solutions to these problems often involve lattices.
- Kepler conjecture (3D) solved by Thomas Hales in 1998.
- Optimal packing in 8D solved by Maryna Viazovska in 2016.

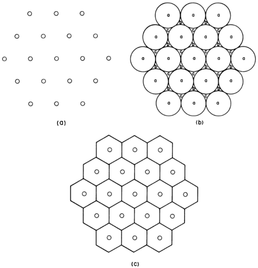


Figure 1.3. (a) The hexagonal lattice in the plane. (b) The corresponding sphere-packing (in this case a circle-packing). The points labeled *a* are the lattice points; the points labeled *b* and *c* are the "deep holes" in this lattice, the points of the plane furthest from the lattice. (c) The *Voronoi cells* are regular hexagons. (d) An enlargement of (b) and (c), showing the *packing radius* ρ of the lattice (the radius of the circles in (b)), and the *covering radius* R, the distance from a lattice point to a deep hole. For this lattice $R = 2\rho/\sqrt{3}$.

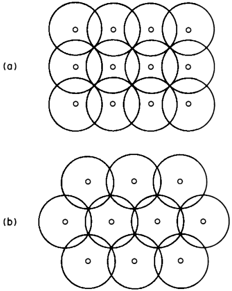


Figure 2.1. Covering the plane with circles. In (a) the centers belong to the square lattice $\mathbf{Z}^2$, in (b) they belong to the hexagonal lattice. (b) is a more efficient or *thinner* covering.

Covering problem

- We consider the problem of covering a subset A of $\mathbb{R}^2$ (not the whole plane).
- $A \subset \mathbb{R}^2$ and $\Omega(\boldsymbol{x}, r) = \cup_{i=1}^m B(x_i, r)$ with $\boldsymbol{x} := \{x_i\}_{i=1}^m$.
- Find $(\boldsymbol{x}, r) \in \mathbb{R}^{2m+1}$ such that $A \subset \Omega(\boldsymbol{x}, r)$ with minimum r.
- Optimization formulation:

$$\underset{(\boldsymbol{x}, r) \in \mathbb{R}^{2m+1}}{\text{Minimize}} \; r \; \text{ subject to } \; G(\boldsymbol{x}, r) = 0,$$

where

$$G(\boldsymbol{x}, r) := \mathrm{Vol}(A) - \mathrm{Vol}(A \cap \Omega(\boldsymbol{x}, r))$$

- $G(\boldsymbol{x}, r) = 0$ if and only if $A \subset \Omega(\boldsymbol{x}, r)$ up to a set of zero measure, i.e., when $\Omega(\boldsymbol{x}, r)$ covers A.

Covering problem

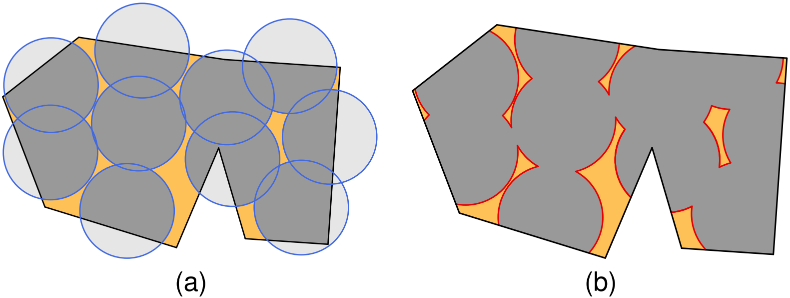


Figure: (a) represents a region A (orange color) to be covered by a union of balls $\Omega(\boldsymbol{x}, r)$. (b) represents, in red, $\partial\Omega(\boldsymbol{x}, r) \cap A$. Each $\mathcal{A}_i := \partial B(x_i, r) \cap \partial\Omega(\boldsymbol{x}, r) \cap A$ corresponds to the red arcs intersecting $\partial B(x_i, r)$. In this example, most sets $\mathcal{A}_i$ contain two or three maximal arcs and there is only one $\mathcal{A}_i$ with four maximal arcs.

Some important notations

- Given $x, y \in \mathbb{R}^n$, $x \cdot y = x^\top y \in \mathbb{R}$ and $x \otimes y = xy^\top \in \mathbb{R}^{n \times n}$.
- $B(x_i, r)$ denotes an open ball with center $x_i \in \mathbb{R}^2$ and radius r.
- For a sufficiently smooth set $S \subset \mathbb{R}^2$, $\nu_S(z)$ denotes the unitary-norm outwards normal vector to S at z.
- $\tau_S(z)$ is the unitary-norm tangent vector to ∂S at z (pointing counter-clockwise).
- When $S = B(x_i, r)$ we use $\nu_i(z) := \nu_{B(x_i,r)}$ and $\tau_i(z) := \tau_{B(x_i,r)}$.
- For intersection points $z \in \partial S \cap B(x_i, r)$, we also use the notation $\nu_{-i}(z) := \nu_S(z)$.

Second-order derivative of G

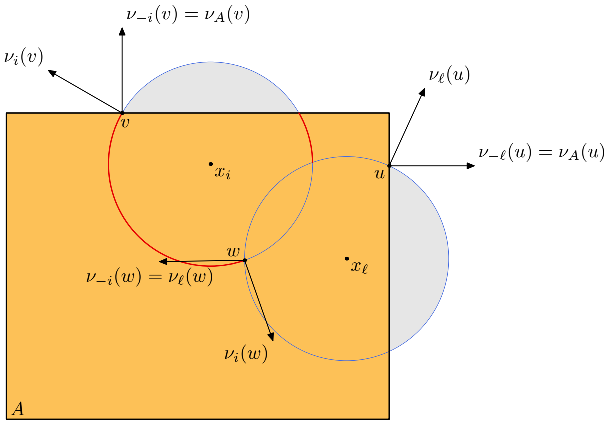


Figure: The set $\mathcal{A}_i = \partial B(x_i, r) \cap \Omega(\boldsymbol{x}, r) \cap A$ is composed of two arcs (in red). If $z \in \partial B(x_i, r) \cap \partial B(x_\ell, r)$ for some $\ell \neq i$, as for $z = w$, then $\nu_{-i}(z) = \nu_\ell(z)$, while if $z \in \partial B(x_i, r) \cap \partial A$, as for $z \in \{u, v\}$, then $\nu_{-i}(z) = \nu_A(z)$.

First and second-order derivative of G

- The derivatives of G can be computed using techniques of shape calculus [Delfour, Henrot, Murat, Pierre, Simon, Sokolowski, Zolésio].
- Assuming $(\boldsymbol{x}, r)$ is non-degenerate:

$$\nabla G(\boldsymbol{x}, r) = -\left(\int_{\mathcal{A}_1} \nu_1(z)\, dz, \cdots, \int_{\mathcal{A}_m} \nu_m(z)\, dz, \int_{\partial\Omega(\boldsymbol{x},r)\cap A} dz\right)^\top,$$

  with $\mathcal{A}_i := \partial B(x_i, r) \cap \partial\Omega(\boldsymbol{x}, r) \cap A$.
- The Hessian of G is given by:

$$\nabla^2 G(\boldsymbol{x}, r) = \begin{pmatrix} \nabla^2_{\boldsymbol{x}} G(\boldsymbol{x}, r) & \nabla^2_{\boldsymbol{x},r} G(\boldsymbol{x}, r) \\ \nabla^2_{\boldsymbol{x},r} G(\boldsymbol{x}, r)^\top & \nabla^2_r G(\boldsymbol{x}, r) \end{pmatrix},$$

  with the blocks $\nabla^2_{\boldsymbol{x}} G(\boldsymbol{x}, r) \in \mathbb{R}^{2m\times 2m}$, $\nabla^2_{\boldsymbol{x},r} G(\boldsymbol{x}, r) \in \mathbb{R}^{2m}$, and $\nabla^2_r G(\boldsymbol{x}, r) = \partial^2_r G(\boldsymbol{x}, r) \in \mathbb{R}$.

Second-order derivative of G

- $\mathcal{A}_i$ is a finite union of arcs of the circle $\partial B(x_i, r)$.
- $\mathbb{A}_i$ denotes the set of pairs (v, w) that represent the arcs in $\mathcal{A}_i$.
- $\mathbb{A}_i = \emptyset$ if $\mathcal{A}_i$ is a full circle.
- The scalar $\nabla_r^2 G(\boldsymbol{x}, r)$ is given by:

$$\nabla_r^2 G(\boldsymbol{x}, r) = -\frac{\mathrm{Per}(\partial\Omega(\boldsymbol{x},r)\cap A)}{r} - \sum_{i=1}^{m} \sum_{(v,w)\in\mathbb{A}_i} \left[\!\!\left[\frac{|L(z)|-\nu_{-i}(z)\cdot\nu_i(z)}{\nu_{-i}(z)\cdot\tau_i(z)} \right]\!\!\right]_v^w$$

- $[\![\Phi(z)]\!]_v^w := \Phi(w) - \Phi(v)$
- For an extreme z of an arc represented by $(v, w) \in \mathbb{A}_i$,

$$L(z) = \{\ell \in \{1, \dots, m\} \setminus \{i\} \mid z \in \partial B(x_\ell, r)\}.$$

Second-order derivative of G

- Matrix $\nabla^2_{\boldsymbol{x}} G(\boldsymbol{x}, r)$ is given by the 2×2 diagonal blocks

$$\partial^2_{x_i x_i} G(\boldsymbol{x}, r) = \frac{1}{r} \int_{\mathcal{A}_i} -\nu_i(z) \otimes \nu_i(z) + \tau_i(z) \otimes \tau_i(z)\, dz + \sum_{(v,w) \in \mathbb{A}_i} \left[\!\!\left[\frac{\nu_{-i}(z) \cdot \nu_i(z)}{\nu_{-i}(z) \cdot \tau_i(z)} \nu_i(z) \otimes \nu_i(z) \right]\!\!\right]_v^w$$

and the 2×2 off-diagonal blocks

$$\partial^2_{x_i x_\ell} G(\boldsymbol{x}, r) = \sum_{v \in \mathcal{I}_{i\ell}} \frac{\nu_i(v) \otimes \nu_\ell(v)}{\nu_\ell(v) \cdot \tau_i(v)} - \sum_{w \in \mathcal{O}_{i\ell}} \frac{\nu_i(w) \otimes \nu_\ell(w)}{\nu_\ell(w) \cdot \tau_i(w)},$$

- $\mathcal{I}_{i\ell} = \{v \in \partial B(x_\ell, r) \mid (v, \cdot) \in \mathbb{A}_i\}$
- $\mathcal{O}_{i\ell} = \{w \in \partial B(x_\ell, r) \mid (\cdot, w) \in \mathbb{A}_i\}$
- Note that $\mathcal{I}_{i\ell} = \mathcal{O}_{i\ell} = \emptyset$ for all $\ell \neq i$ if $\mathbb{A}_i = \emptyset$.

Second-order derivative of G

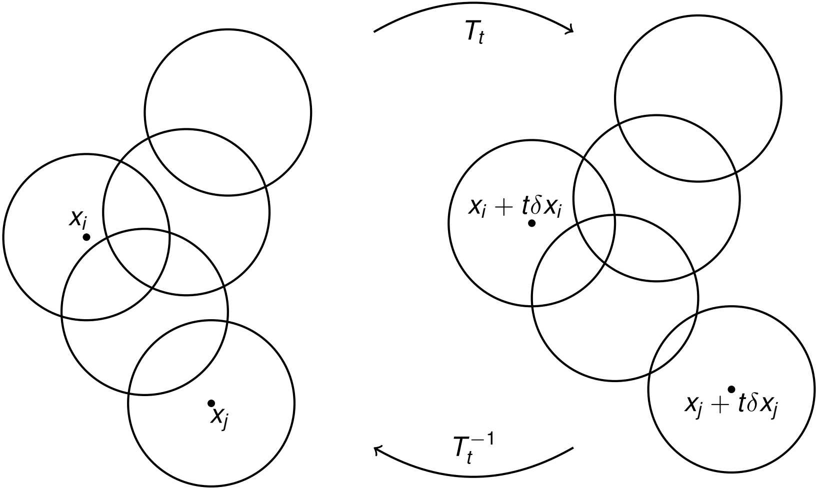
- $\nabla^2_{\boldsymbol{x},r} G(\boldsymbol{x}, r)$ is given by the 2-dimensional arrays

$$\partial^2_{x_i r} G(\boldsymbol{x}, r) = -\frac{1}{r}\int_{\mathcal{A}_i} \nu_i(z)\, dz + \sum_{(v,w)\in\mathbb{A}_i} \left[\!\left[\frac{\nu_{-i}(z)\cdot\nu_i(z)}{\nu_{-i}(z)\cdot\tau_i(z)}\nu_i(z) - \sum_{\ell\in L(z)} \frac{\nu_i(z)}{\tau_i(z)\cdot\nu_\ell(z)} \right]\!\right]_v^w$$

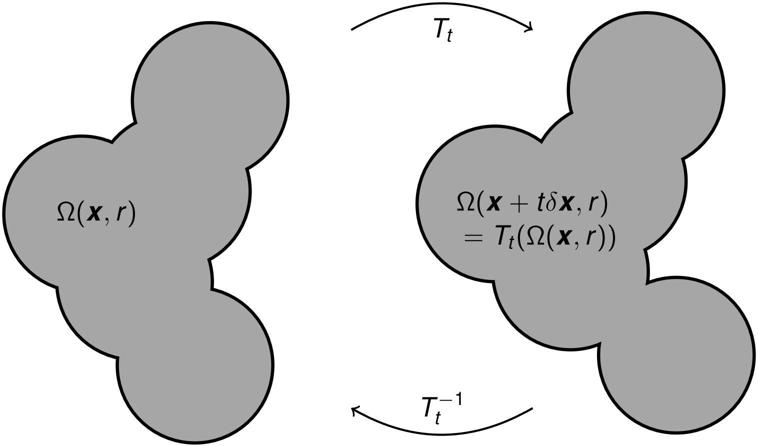
- For an extreme z of an arc represented by $(v, w) \in \mathbb{A}_i$,

$$L(z) = \{\ell \in \{1, \ldots, m\} \setminus \{i\} \mid z \in \partial B(x_\ell, r)\}.$$

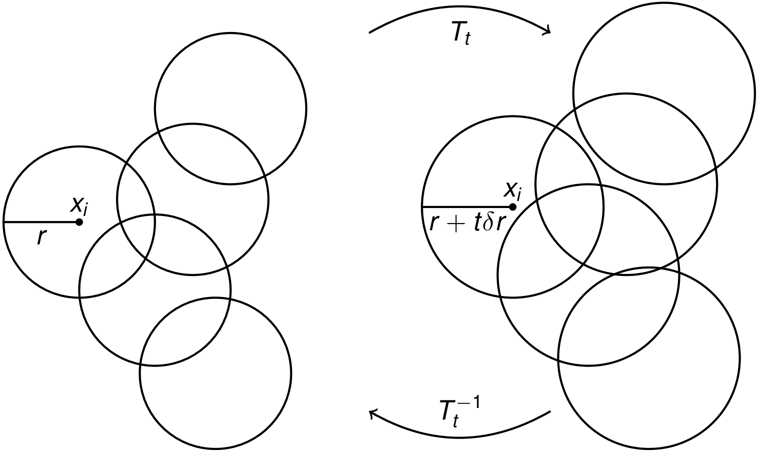
Center perturbations



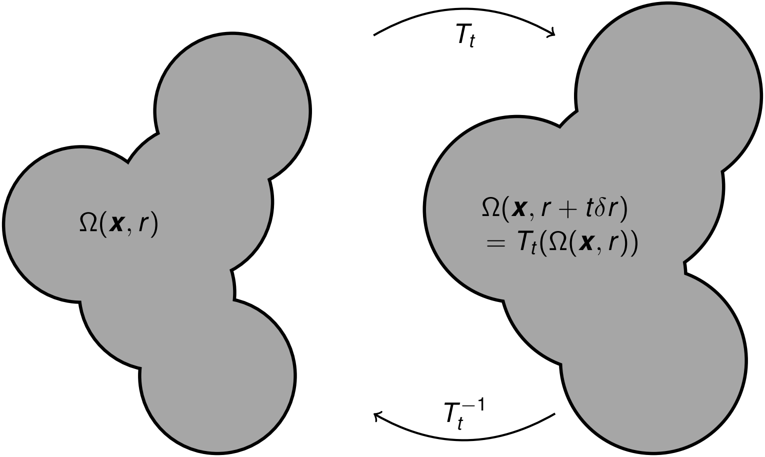
Center perturbations



Radius perturbations



Radius perturbations



Construction of bi-Lipschitz mappings T_t

- How can we build bi-Lipschitz mappings $T_t : \Omega(\boldsymbol{x}, r) \to \Omega(\boldsymbol{x} + t\delta\boldsymbol{x}, r)$ and $T_t : \Omega(\boldsymbol{x}, r) \to \Omega(\boldsymbol{x}, r + t\delta r)$?
- First, we observe that $\partial\Omega(\boldsymbol{x}, r)$ contains singular points (the circle intersections) and regular points.
- The motion of the singular points is fully determined by the center or radius perturbations.
- For instance, the motion of an intersection point in $\partial B(x_i + t\delta x_i, r) \cap \partial B(x_j + t\delta x_j, r)$ can be fully determined, for sufficiently small t, using the implicit function theorem.
- The motion of the regular points is underdetermined. Roughly speaking, one direction of T_t is prescribed (such as $t\delta x_i$ for center perturbations), while the orthogonal direction can be choosen “freely”.

- Thus, we are free to choose this orthogonal direction of T_t at regular points, as long as these constraints are satisfied:
  - T_t must be bi-Lipschitz
  - The value of T_t is prescribed at the singular points.
  - $T_t(\Omega(\boldsymbol{x}, r) \cap A) = \Omega(\boldsymbol{x} + t\delta\boldsymbol{x}, r) \cap A$ or $T_t(\Omega(\boldsymbol{x}, r) \cap A) = \Omega(\boldsymbol{x}, r + t\delta r) \cap A$.
- Since $\partial\Omega(\boldsymbol{x}, r)$ is a union of arcs, we can use local polar coordinates on each $B(x_i, r)$ to define the missing direction of T_t at regular points.
- Then we extend T_t to $\Omega(\boldsymbol{x}, r) \cap A$, in a way that preserves the bi-Lipschitz property of T_t.

Shape derivative for radius perturbations

- We can actually build a bi-Lipschitz mapping

$$T_t : \Omega(\boldsymbol{x}, r) \cap A \to \Omega(\boldsymbol{x}, r + t\delta r) \cap A.$$

- T_t allows us to use the following change of variables:

$$\begin{aligned} G(\boldsymbol{x}, r + t\delta r) &= \mathrm{Vol}(A \setminus \Omega(\boldsymbol{x}, r + t\delta r)) \\ &= \mathrm{Vol}(A) - \mathrm{Vol}(A \cap \Omega(\boldsymbol{x}, r + t\delta r)) \\ &= \mathrm{Vol}(A) - \int_{T_t(\Omega(\boldsymbol{x},r)\cap A)} dz \\ &= \mathrm{Vol}(A) - \int_{\Omega(\boldsymbol{x},r)\cap A} |\det DT_t(z)|\, dz. \end{aligned}$$

- Then the derivative is, with $V := \partial_t T_t|_{t=0}$,

$$\begin{aligned} \frac{d}{dt} G(\boldsymbol{x}, r + t\delta r)\Big|_{t=0} &= -\int_{\Omega(\boldsymbol{x},r)\cap A} \mathrm{div}\, V(z)\, dz \\ &= -\int_{\partial(\Omega(\boldsymbol{x},r)\cap A)} V(z) \cdot \nu(z)\, dz = -\delta r \int_{\partial\Omega(\boldsymbol{x},r)\cap A} dz \end{aligned}$$

Other shape derivatives

- The property $V(z) \cdot \nu(z) = \delta r$ on $\partial\Omega(\boldsymbol{x}, r) \cap A$ comes from the explicit construction of T_t on $\partial(\Omega(\boldsymbol{x}, r) \cap A)$.
- The calculation works in a similar way for center perturbations and for second-order derivatives.
- The main task is to build the appropriate T_t for each type of perturbation, and compute the corresponding $V := \partial_t T_t|_{t=0}$.

Other shape derivatives

- These derivatives were obtained assuming $(\boldsymbol{x}, r)$ is non-degenerate, i.e., when the following assumptions hold.
- **Assumption 1.** The centers $\{x_i\}_{i=1}^m$ satisfy $\|x_i - x_j\| \notin \{0, 2r\}$ for all $1 \leq i, j \leq m$, $i \neq j$ and $\partial B(x_i, r) \cap \partial B(x_j, r) \cap \partial B(x_k, r) = \emptyset$ for all $1 \leq i, j, k \leq m$ with i, j, k pairwise distinct.
- **Assumption 2.** $\Omega(\boldsymbol{x}, r)$ and A are compatible.
- This yields the following decomposition, with $\bar{k}$ independent of t:

$$\partial\Omega(\boldsymbol{x} + t\delta\boldsymbol{x}, r) \cap A = \bigcup_{k=1}^{\bar{k}} \mathcal{S}_k(t),$$

where $\mathcal{S}_k(t)$ are arcs parameterized by an angle aperture $[\theta_{k,v}(t), \theta_{k,w}(t)]$, and $t \mapsto \theta_{k,v}(t)$, $t \mapsto \theta_{k,w}(t)$ are continuous.

Compatibility

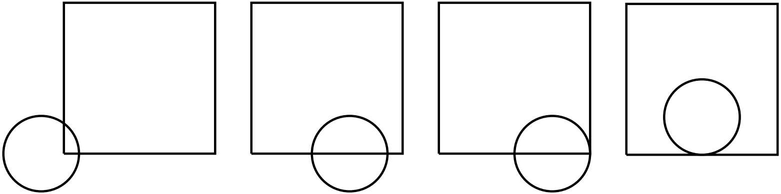


Figure: Compatibility of a ball and a square. From left to right: (a) compatible (b) compatible (c) not compatible (d) not compatible.

Example of degenerate case: two tangent disks

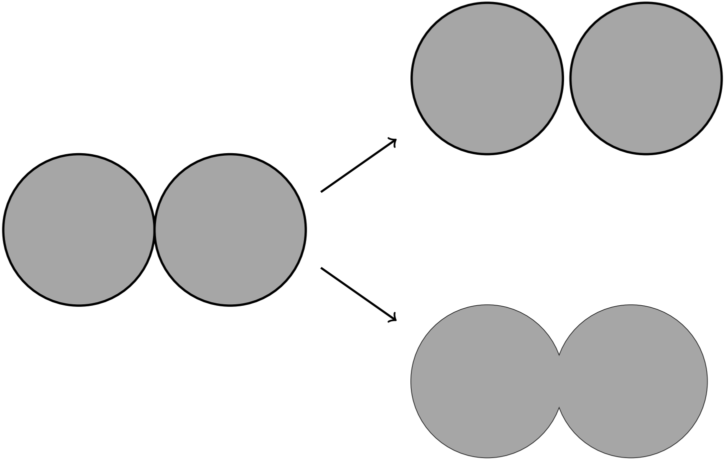
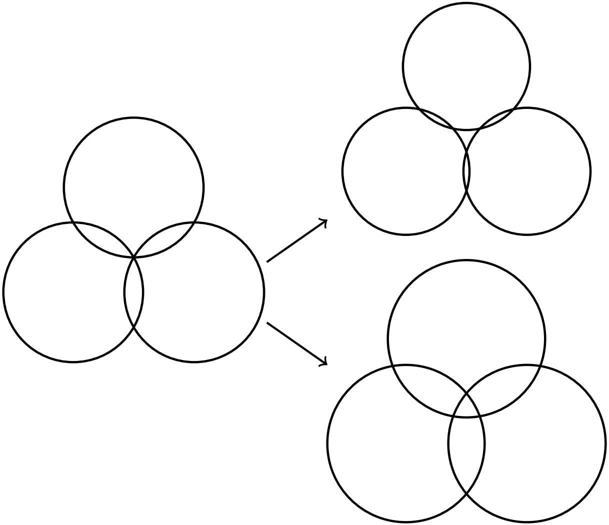


Figure: Two tangent disks $B(x_1, r)$ and $B(x_2, r)$ may either merge if $(x_1 - x_2) \cdot (\delta x_1 - \delta x_2) < 0$ or have an empty intersection if $(x_1 - x_2) \cdot (\delta x_1 - \delta x_2) > 0$.

Example of degenerate case: three disks



Analysis of singular cases

- Other singular cases: two superposed disks, a disk tangent to ∂A, etc ...
- Singular cases can be investigated using asymptotic analysis: G is sometimes differentiable, but seems to never be twice differentiable.
- Gateaux semidifferentiablity of the components of ∇G can often be proved.

Algorithm 1

- After discretization, the problem becomes a constrained nonlinear programming problem (with a linear objective function and a single difficult nonlinear constraint) of the form

$$\text{Minimize } f(\boldsymbol{x}, r) := r \text{ subject to } G_h(\boldsymbol{x}, r) = 0 \text{ and } r \geq 0$$

- We considered the safeguarded Augmented Lagrangian (AL) method Algencan [Andreani, Birgin, Martínez, Schuverdt].
- Algencan is based on the PHR AL function, in this case:

$$L_\rho(\boldsymbol{x}, r, \lambda) = f(\boldsymbol{x}, r) + \frac{\rho}{2}\left[G_h(\boldsymbol{x}, r) + \frac{\lambda}{\rho}\right]^2, \tag{1}$$

  for all $\rho > 0$, $r \geq 0$, and $\lambda \in \mathbb{R}$.
- Each iteration of the method consists in the approximate minimization of (1) subject to $r \geq 0$ followed by the update of the Lagrange multiplier λ and the penalty parameter ρ.

Numerical results for Algorithm 1

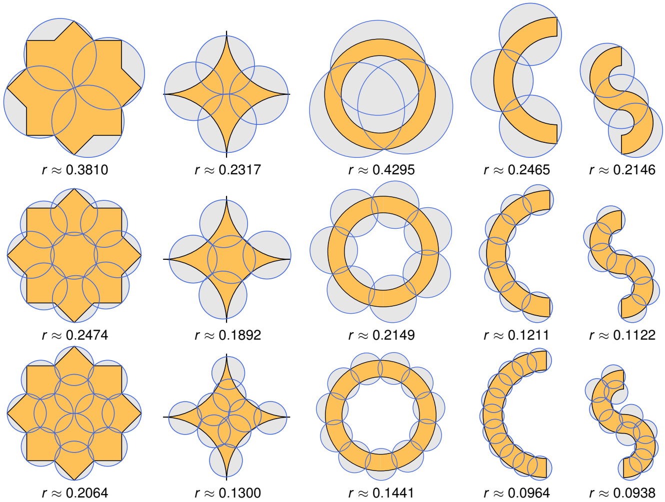


Figure: Solutions found for covering two-squares region with $m = 4, 9, 12$, peaked star region with $m = 4, 5, 9$, ring, half-ring, and two-half-rings regions with $m = 3, 7, 11$, and disconnected region with $m = 3, 7, 15$.

Numerical results for Algorithm 1

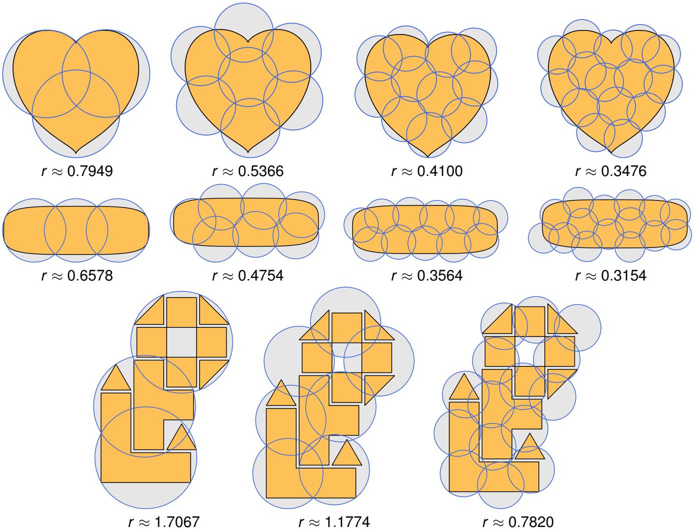
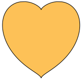
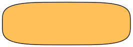
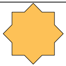
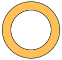


Figure: Solutions found for covering heart-shape and soap-shape regions with $m = 3, 7, 11, 15$, and disconnected region with $m = 3, 7, 15$.

Performance metrics for Algorithm 1

Alg. 1.1 computes G (complexity $O(1/h^2)$), Alg. 1.2 computes ∇G (complexity $O(1/h)$), “trial” is the number of the initial guess yielding the best solution, “outit” and “innit” are the number of outer and inner iterations of the AL optimization method.

Region A	m	r^*	trial	outit	innit	Alg. 1.1	Alg. 1.2	CPU Time
	3	0.7949	100	20	155	2188	249	59.08
	7	0.5366	69	15	50	214	117	7.92
	11	0.4100	89	12	68	303	130	12.77
	15	0.3476	78	13	77	311	138	15.46
	3	0.6578	70	12	76	402	134	4.61
	7	0.4754	30	13	119	1228	185	20.11
	11	0.3564	61	13	72	261	132	6.12
	15	0.3154	69	13	80	447	140	12.77
	4	0.3810	91	11	40	222	90	2.78
	9	0.2474	70	11	45	197	94	3.18
	12	0.2064	32	10	66	346	112	6.16
	4	0.2317	82	20	136	2221	230	14.55
	5	0.1892	32	10	61	251	107	1.70
	9	0.1300	59	10	56	248	107	1.84
	3	0.4295	12	10	40	186	86	0.49
	7	0.2149	36	10	35	155	78	0.58
	11	0.1441	23	12	94	337	152	1.50

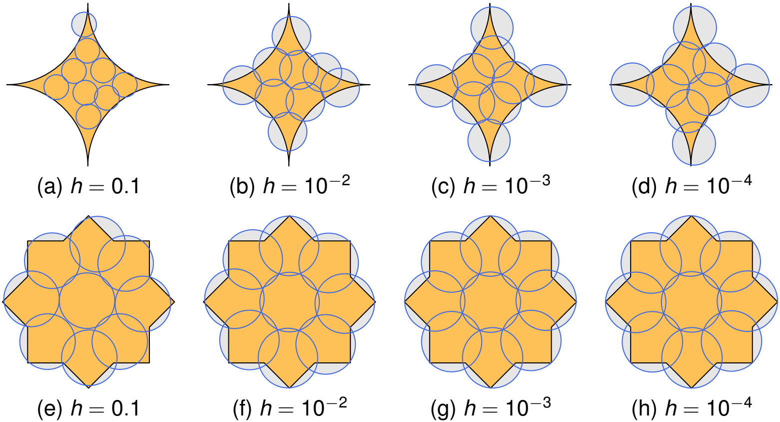


Figure: Solutions found varying $h \in \{0.1, 10^{-2}, 10^{-3}, 10^{-4}\}$ in problems (a–d) "two squares" and (e–h) "peaked star" with $m = 9$. The peaked star requires a smaller h to cover its small thin features.

Algorithm 1

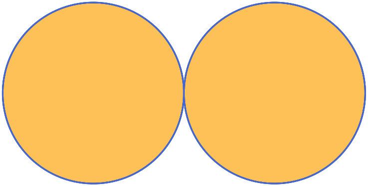


Figure: An example of a degenerate case: A is the union of two tangent unitary-diameter balls to be covered by $m = 2$ balls. In this case, ∇G does not exist. Even though this singular case is not covered by the theory, the solution, which is the set A itself, was found with a single run of the method.

Algorithm 2

- Algorithm 1 allows to find coverings of general shapes A, but is relatively slow when a fine discretization is required (i.e., a small h). This occurs when A presents small thin features. Algorithm 1 only uses G and ∇G.
- Algorithm 2 deals with the case $A = \cup_{j=1}^{p} A_j$ and $\{A_j\}_{j=1}^{p}$ are non-overlapping convex polygons. Algorithm 2 uses $G, \nabla G$ and $\nabla^2 G$. In this case $G, \nabla G$ and $\nabla^2 G$ can be computed analytically which leads to a fast and accurate algorithm.

Algorithm 2

- Compute Voronoi diagram with cells $\{V_i\}_{i=1}^m$ associated with the balls centers $x_1, \ldots, x_m$.
- Compute convex polygons $W_{ij} = A_j \cap V_i$ and $S_{ij} = W_{ij} \cap B(x_i, r)$ for $j = 1, \ldots, p$ and $i = 1, \ldots, m$.
- $\mathcal{K}_{A_j} = \{i \in \{1, \ldots, m\} \mid S_{ij} \neq \emptyset\}$
- Partition $A_j \cap \Omega(\boldsymbol{x}, r) = \bigcup_{i \in \mathcal{K}_{A_j}} S_{ij}, \quad j = 1, \ldots, p.$

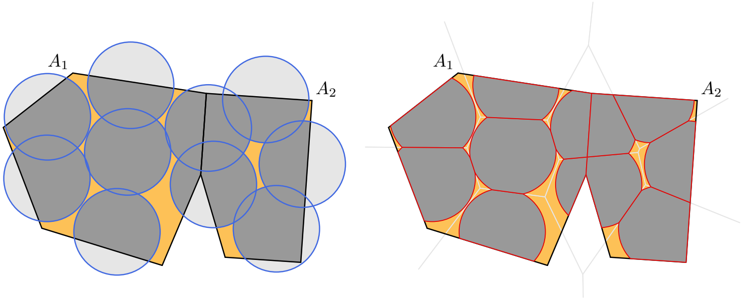


Figure: (left) $A = \cup_{j=1}^p A_j$ with $p = 2$ and $\Omega(\boldsymbol{x}, r) = \cup_{i=1}^m B(x_i, r)$ with $m = 10$. (right) Voronoi diagram and sets S_{ij}.

Algorithm 2

- Using the sets S_{ij}, $G, \nabla G$ and $\nabla^2 G$ can be computed analytically.
- $G(\boldsymbol{x}, r) = \mathrm{Vol}(A) - \mathrm{Vol}(A \cap \Omega(\boldsymbol{x}, r)) = \mathrm{Vol}(A) - \sum_{(i,j)\in\mathcal{K}} \mathrm{Vol}(S_{ij})$.
- Using Green's Theorem,

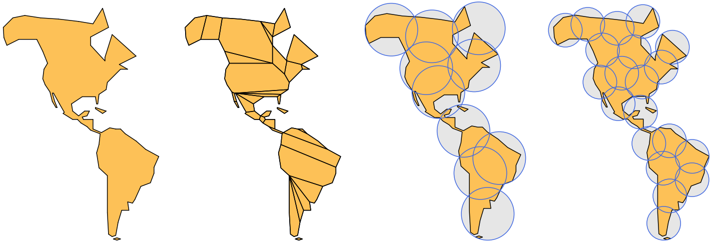
$$
\begin{aligned}
\mathrm{Vol}(S_{ij}) &= \int_{S_{ij}} dxdy = \int_{\partial S_{ij}} x\, dy \\
&= \sum_{[v,w]\in\mathbb{E}(S_{ij})} \int_0^1 x_{\mathcal{E}}(t)\, dy_{\mathcal{E}}(t) + \sum_{(v,w)\in\mathbb{A}(S_{ij})} \int_{\theta_v}^{\theta_w} x_{\mathcal{A}}(\theta)\, dy_{\mathcal{A}}(\theta)
\end{aligned}
$$

  and this can be computed analytically.
- Here $\mathbb{E}(S_{ij})$ is the set of edges of ∂S_{ij}, and $\mathbb{A}(S_{ij})$ is the set of arcs of ∂S_{ij}.
- Works in a similar way for ∇G and $\nabla^2 G$.

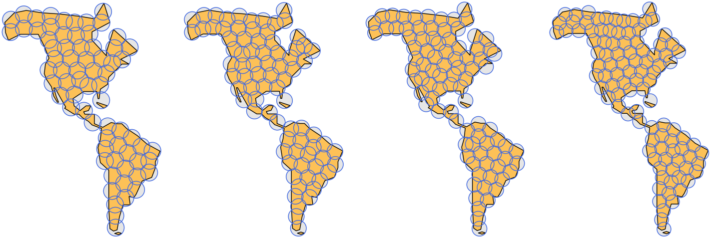
Algorithm 2

- The algorithms for computing $G, \nabla G$ and $\nabla^2 G$ depend on the computation of $\mathbb{E}_i$ and $\mathbb{A}_i$ for $i = 1, \dots, m$.
- Computing $\mathbb{E}_i$ and $\mathbb{A}_i$ requires to compute the Voronoi diagram (using Fortune's algorithm) and to compute $W_{ij} = V_i \cap A_j$ and $S_{ij} = W_{ij} \cap B(x_i, r)$.
- The worst-case time complexity of Algorithm 2 is $\mathcal{O}(m \log m + m \sum_{j=1}^{p} e_{A_j})$, where e_{A_j} is the number of sides of A_j.

Numerical results for Algorithm 2



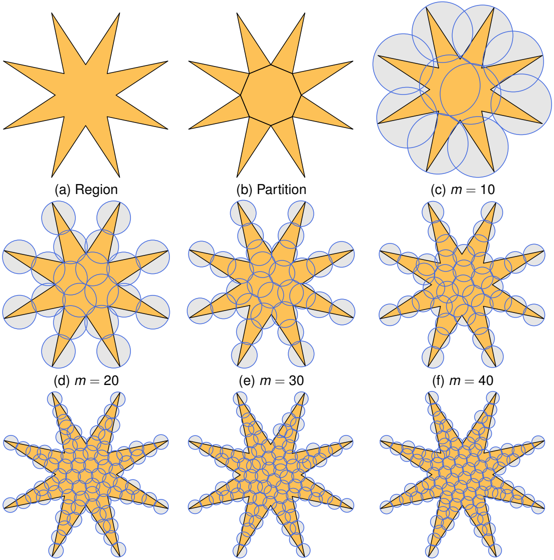
(a) Region (b) Partition (c) $m = 10$ (d) $m = 20$



(i) $m = 70$ (j) $m = 80$ (k) $m = 90$ (l) $m = 100$

Figure: (a) Sketch of America, partitioned into $p = 34$ convex polygons. Pictures from (c) to (l) display the solutions for $m \in \{10, \ldots, 100\}$.

Numerical results for Algorithm 2



(a) Region (b) Partition (c) $m = 10$

(d) $m = 20$ (e) $m = 30$ (f) $m = 40$

(j) $m = 80$ (k) $m = 90$ (l) $m = 100$

Conclusion

- Shape-Newton method in a nonsmooth setting.
- Algorithm 1 is based on first-order derivative and works for general shapes A.
- Algorithm 2 is based on first- and second-order derivatives and works for the union of non-overlapping convex polygons. Much faster and more accurate than Algorithm 1.
- It seems that the assumptions used to derive $\nabla^2 G$ cannot be weakened.
- *A shape optimization approach to the problem of covering a two-dimensional region with minimum-radius identical balls*
  E. G. Birgin, A. Laurain, R. Massambone, and A. G. Santana
  SIAM Journal on Scientific Computing
  43(3):A2047–A2078, 2021
- *A Shape-Newton approach to the problem of covering with identical balls.*
  E. G. Birgin, A. Laurain, R. Massambone, and A. G. Santana
  arXiv:2106.03641, 2021

Outlook

- Extension to PDE constraints in 2D $\rightarrow$ the construction for T_t is the same.
- Extension to 3D $\rightarrow$ a new approach needs to be found to build T_t.
- Extension to arbitrary shapes instead of balls $\rightarrow$ a new approach needs to be found to build T_t.

Outlook

- Extension to PDE constraints in 2D $\rightarrow$ the construction for T_t is the same.
- Extension to 3D $\rightarrow$ a new approach needs to be found to build T_t.
- Extension to arbitrary shapes instead of balls $\rightarrow$ a new approach needs to be found to build T_t.

THANKS FOR YOUR ATTENTION!