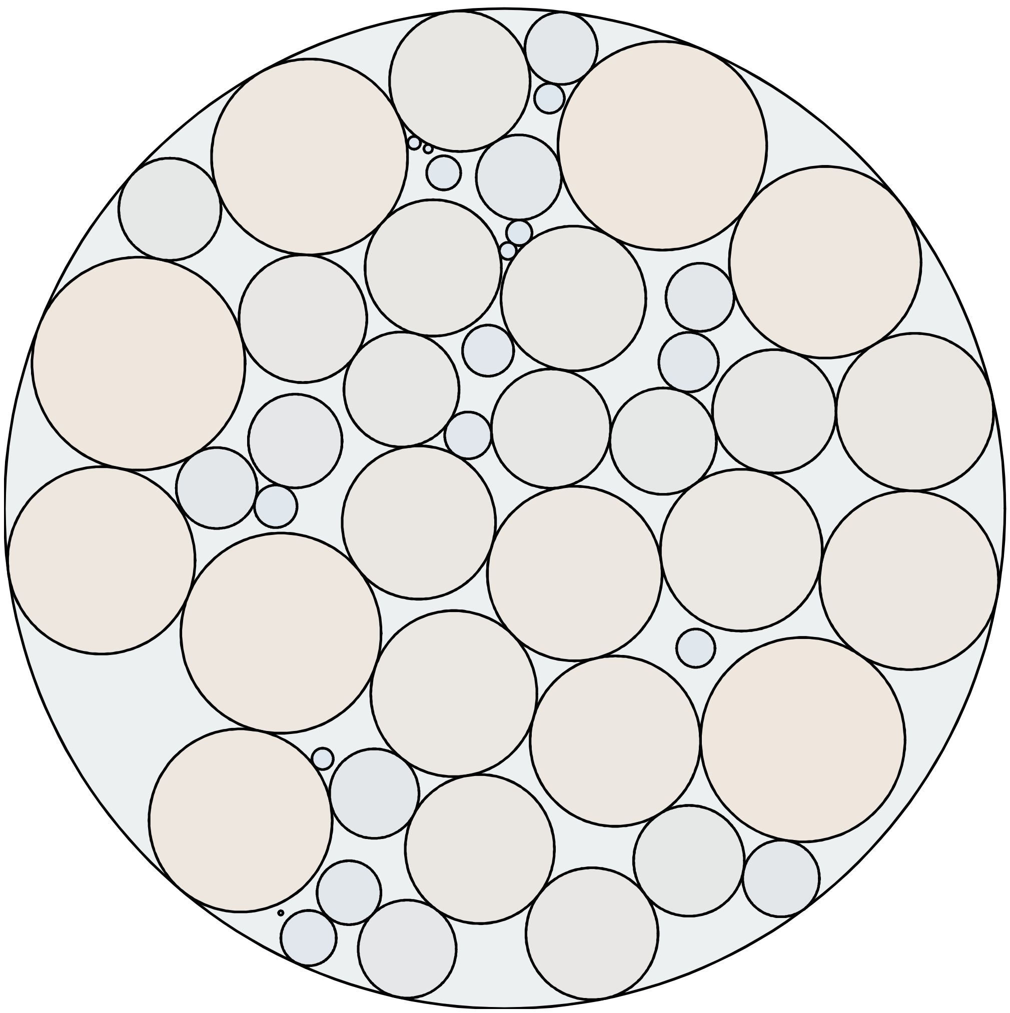
Packing Disks into a Circle

Introduction

This application finds the best packing of unequal non-overlapping disks in a larger circle, such that the radius of the container is minimized. This is a difficult global optimization problem that demands strong solvers; this application uses Maple's Global Optimization Toolbox. You must have the Global Optimization Toolbox installed to use this application

One solution for the packing of 50 disks with the radii 1 to 50 (as found by this application) is visualized below. Other solutions are documented at http://www.packomania.com.



Packing optimization is industrially important, with applications in pallet loading, the arrangement of fiber optic cables in a tube, or the placing of blocks on a circuit board.

Setup

```
> restart :
  with(GlobalOptimization) :
```

Number of circles

```
> n := 50 :
```

Radius of circle n is equal to n

> **for** i **to** n **do**
> $r_i := i$
> **end do**:

Decision Variables and Optimization Bounds

The decision variables are the coordinates (x_i, y_i) of the centers of the circles, and the radius rc of the circumscribing circle.

> $\mathrm{vars} := \left[\mathrm{seq}\left(x_i, i = 1..n\right), \mathrm{seq}\left(y_i, i = 1..n\right), rc\right]$:

> $\mathrm{bounds} := \mathrm{seq}\left(\mathrm{vars}_i = -500..500, i = 1..2\,n\right), rc = 0..500$:

Constraints

The maximum distance between the furthest point on a circle's circumference and the origin must be smaller than the radius of the circumscribing circle.

> $\mathrm{cons1} := \mathrm{seq}\left(r_i + \sqrt{x_i^2 + y_i^2} \le rc, i = 1..n\right)$:

For circles i and j not to overlap, distance between the centers of any two circles minus their radii must be greater than zero

> $\mathrm{cons2} := \mathrm{seq}\left(\mathrm{seq}\left(\sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} - r_i - r_j \ge 0, j = i + 1..n\right), i = 1..n - 1\right)$:

Hence the entire set of constraints

> $\mathrm{cons} := \{\mathrm{cons1}, \mathrm{cons2}\}$:

Optimization and Results

```
> soln := GlobalSolve(rc, cons, bounds, timelimit = 120) :
```

Hence the optimized radius of the circumscribing circle is

> *soln*[1]

232.866858569466700

> colorSpread := ColorTools:-*Gradient*([221, 231, 240]..[240, 231, 221], *number* = *n*) :

> circs := seq(plottools:-disk([rhs(select(has, soln[2], x[i])[ ]), rhs(select(has, soln[2], y[i])[ ])], r[i], color = colorSpread[i], thickness = 0), i = 1..n) :

```
> boundingCirc := plottools:-disk([0, 0], rhs(select(has, soln[2], rc)[ ]), color = RGB(236/255, 240/255, 241/255), thickness = 0) :
```

$$\text{boundingCirc} := \text{plottools:-disk}\left([0,0], \text{rhs}(\text{select}(\text{has}, \text{soln}[2], \text{rc})[\ ]), \text{color} = \text{RGB}\left(\frac{236}{255}, \frac{240}{255}, \frac{241}{255}\right), \text{thickness} = 0\right):$$

```
> plots:-display(circs, boundingCirc, scaling = constrained, size = [800, 800], axes = none)
```