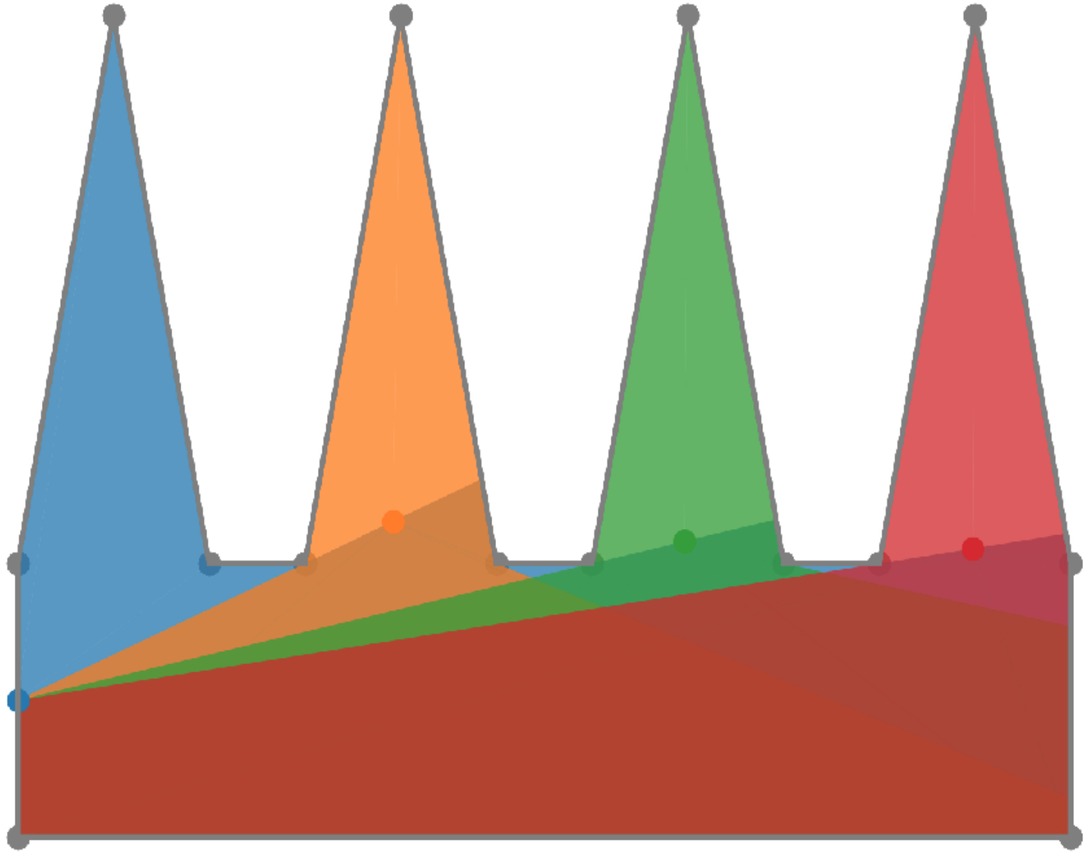
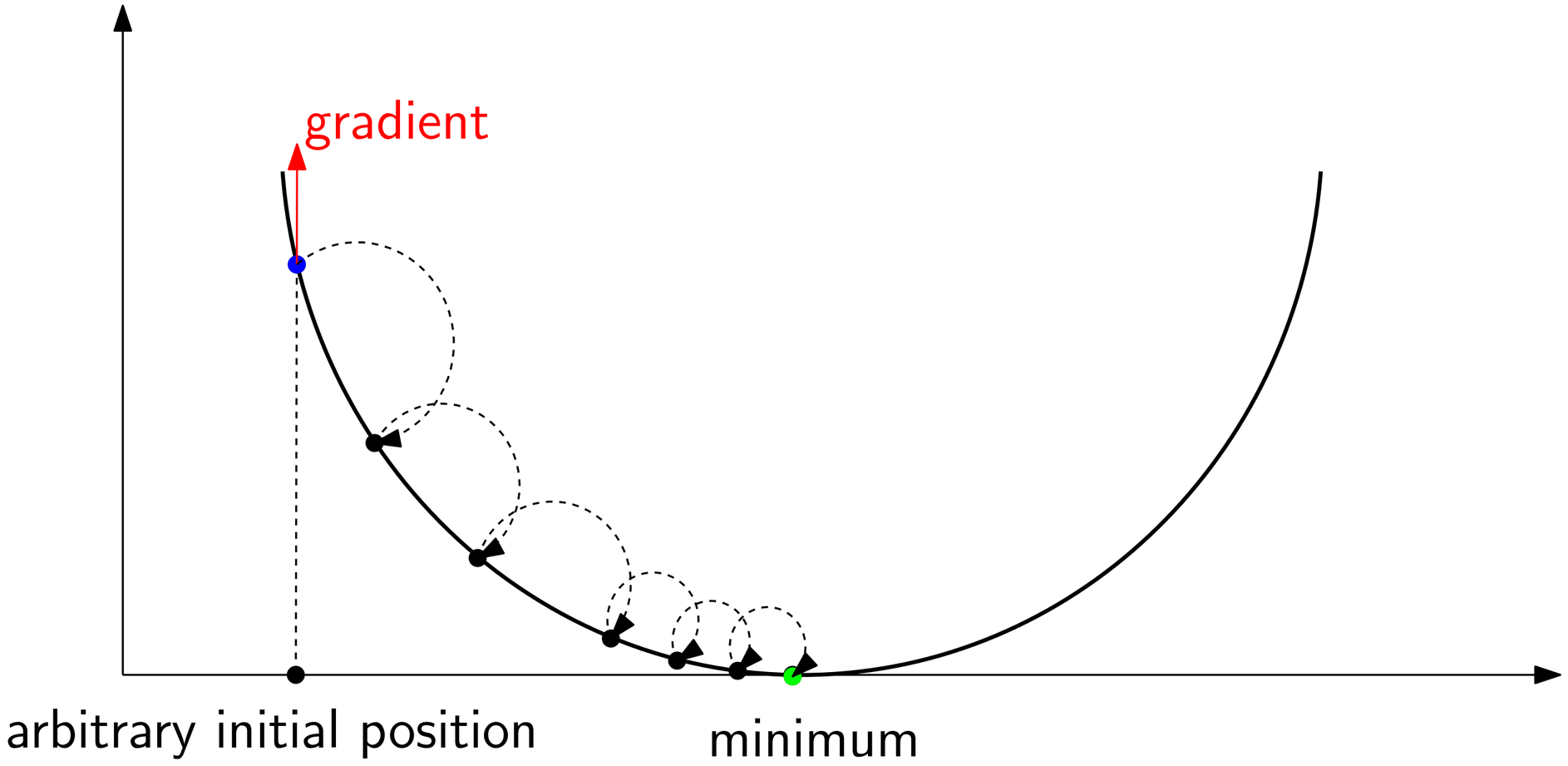
Solving the Art Gallery Problem Using Gradient Descent

Master's Thesis Defense

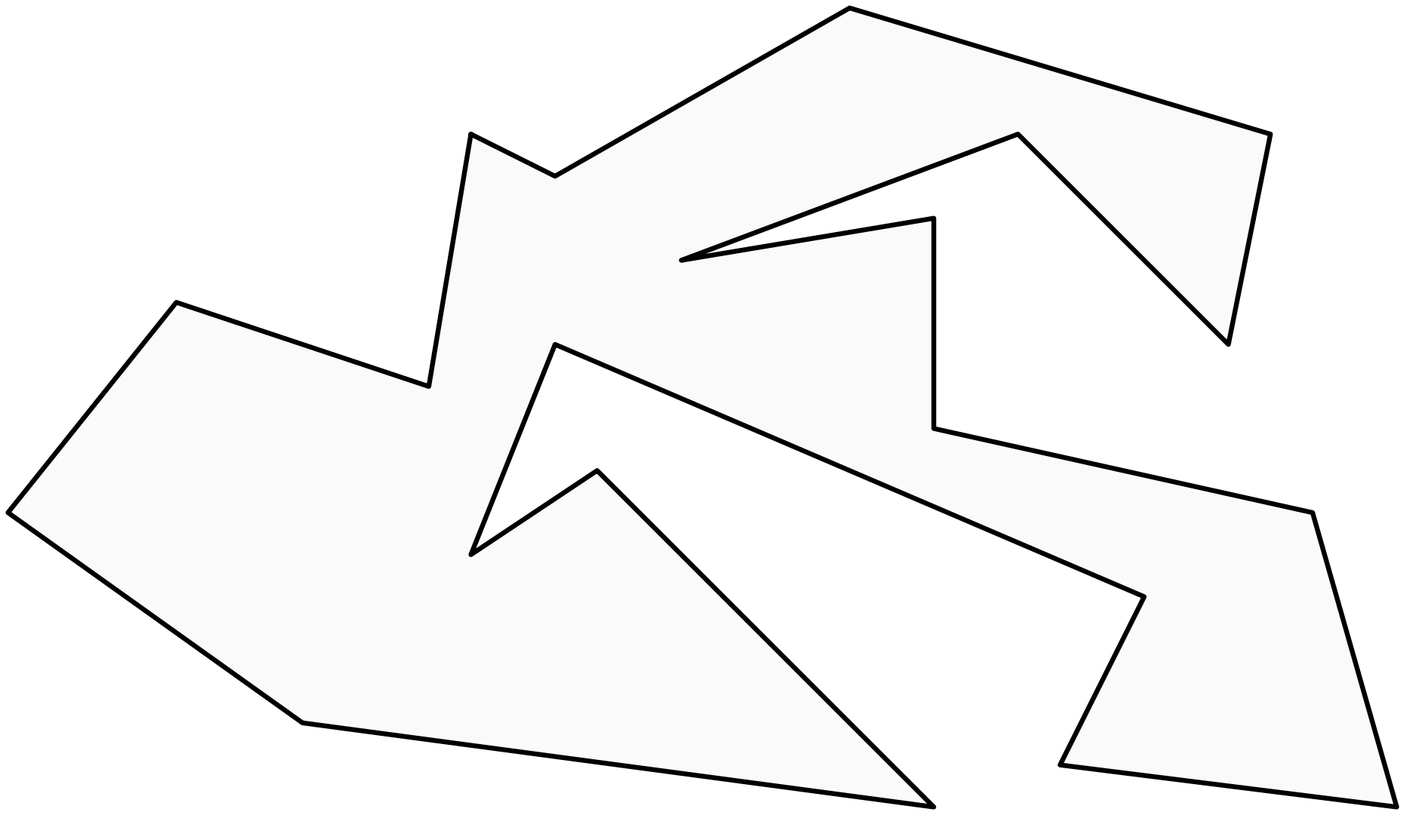
Geo Juglan



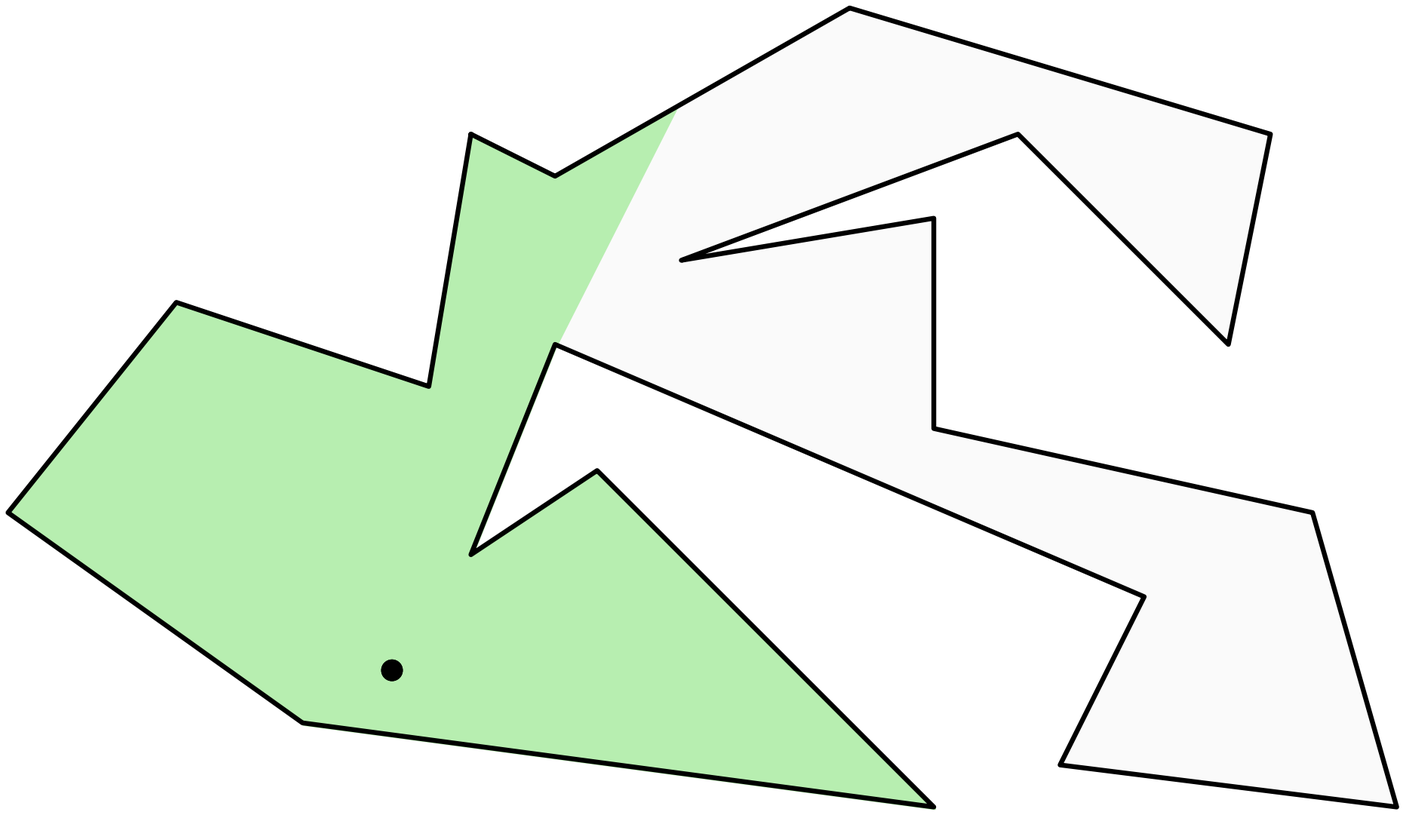
Gradient Descent



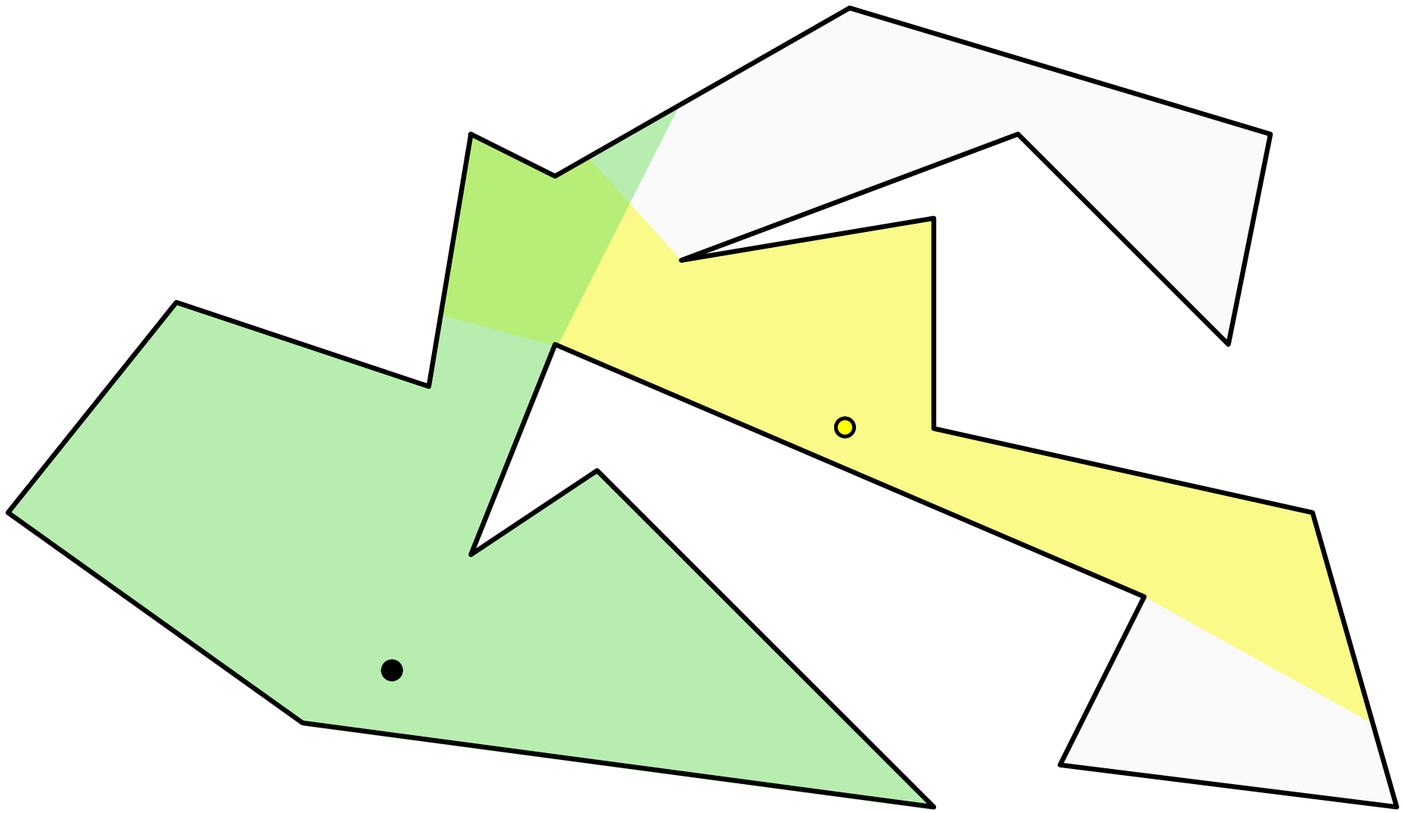
The Art Gallery Problem



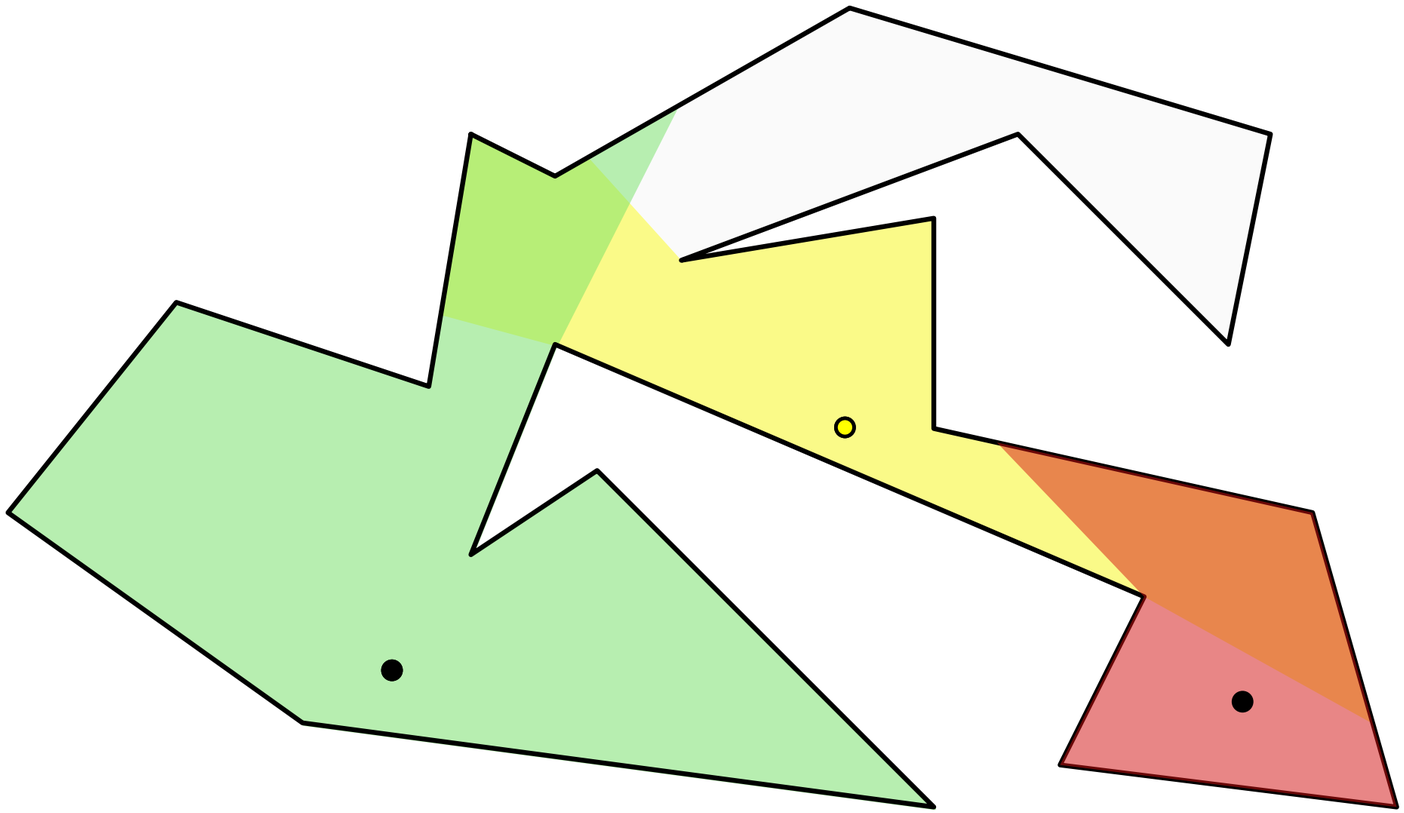
The Art Gallery Problem



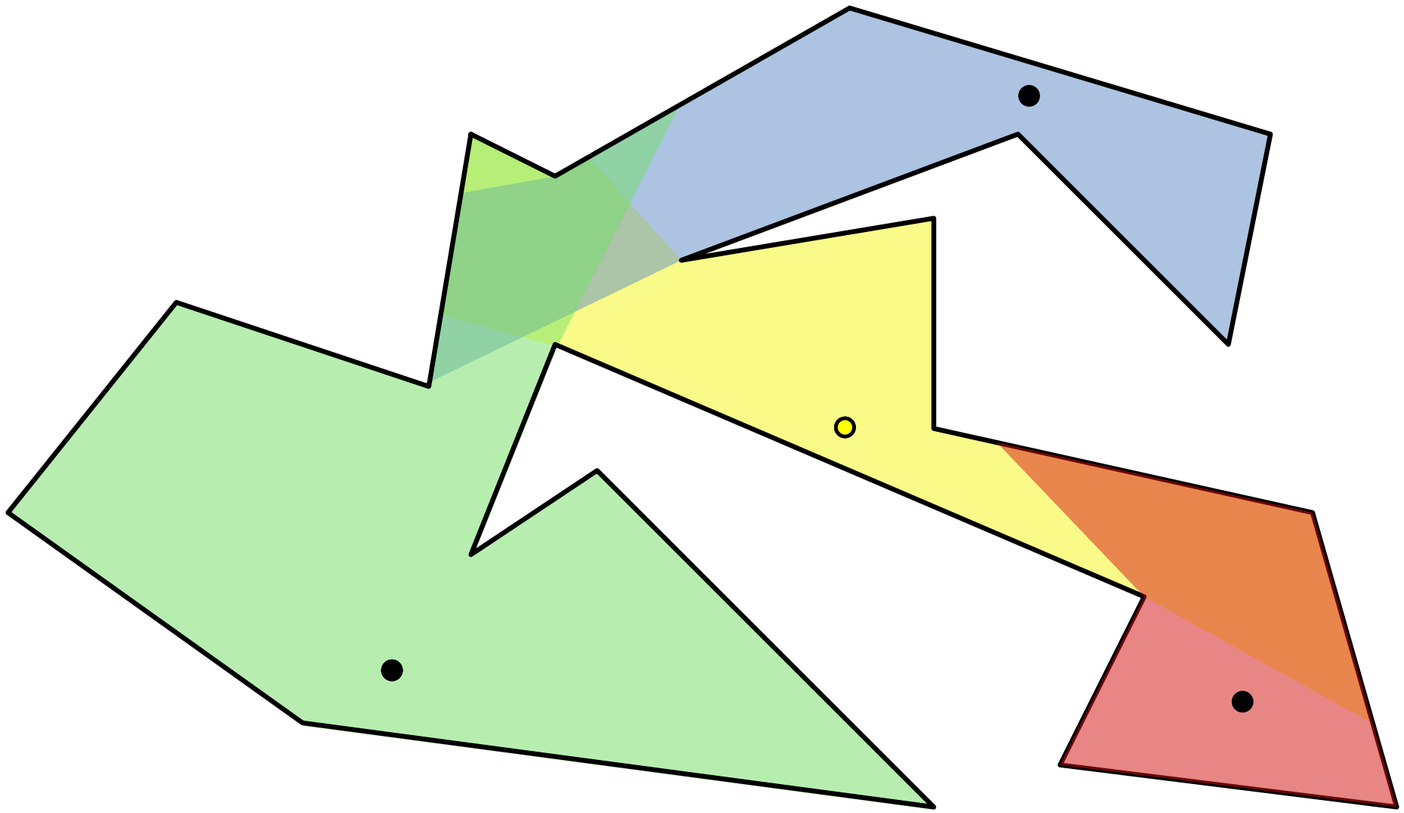
The Art Gallery Problem



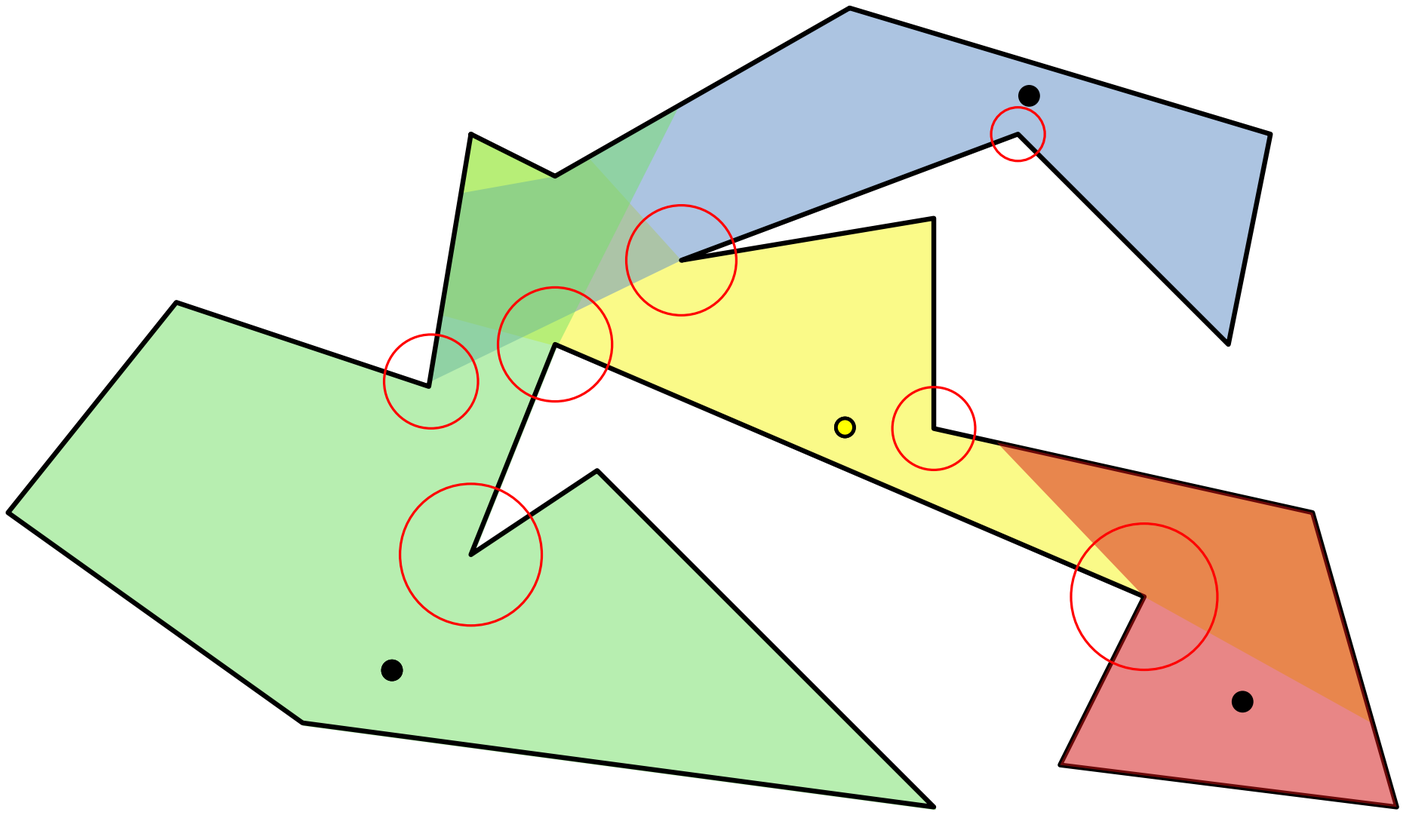
The Art Gallery Problem



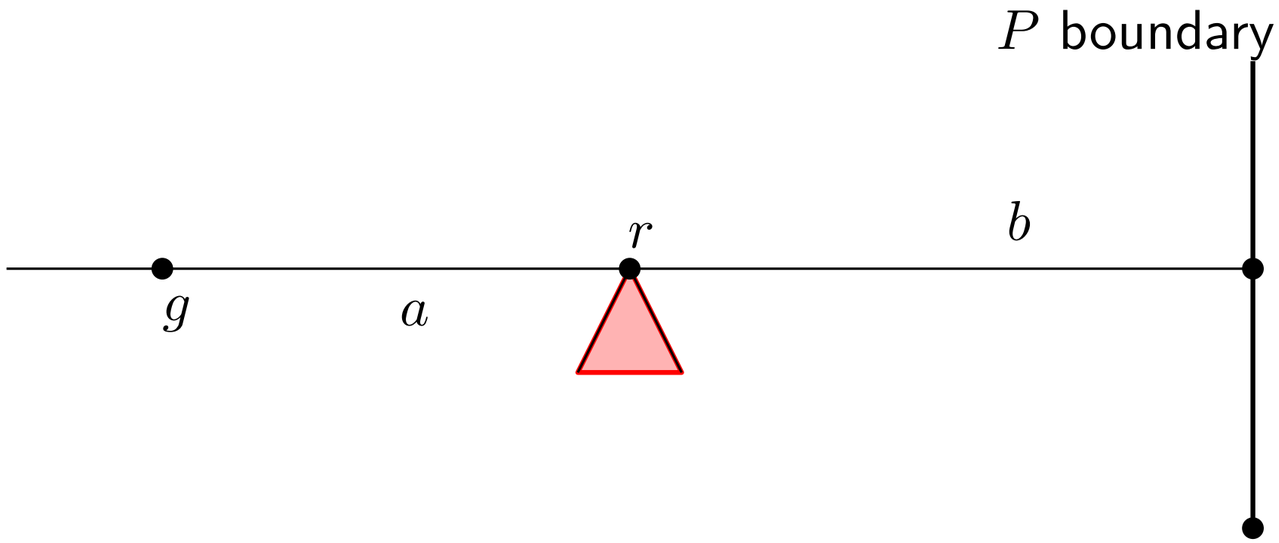
The Art Gallery Problem



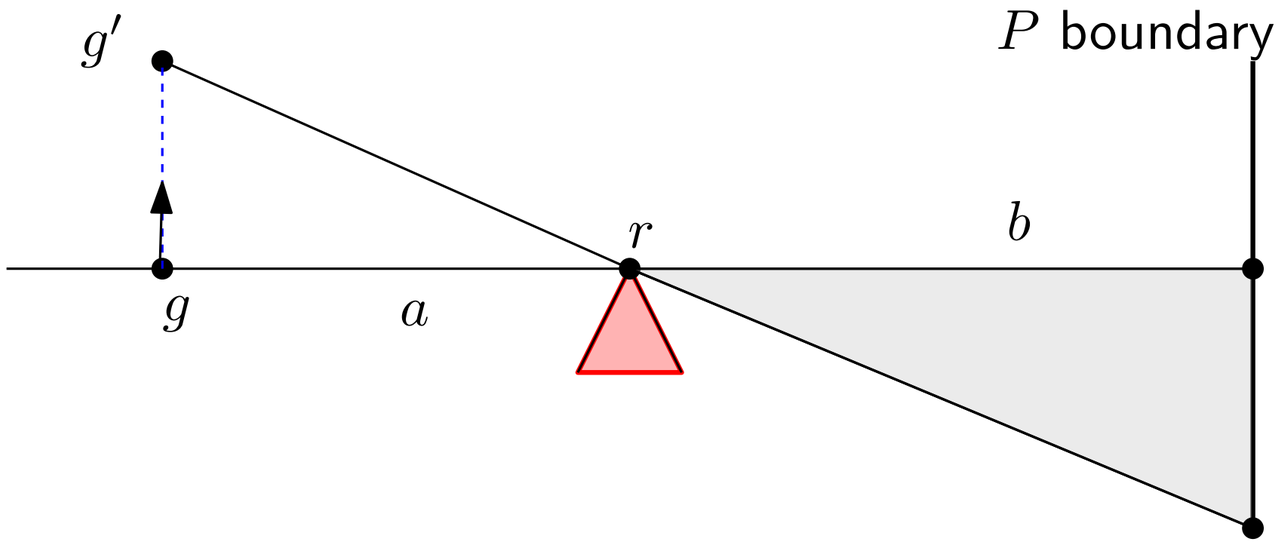
The Art Gallery Problem



Computing the gradient

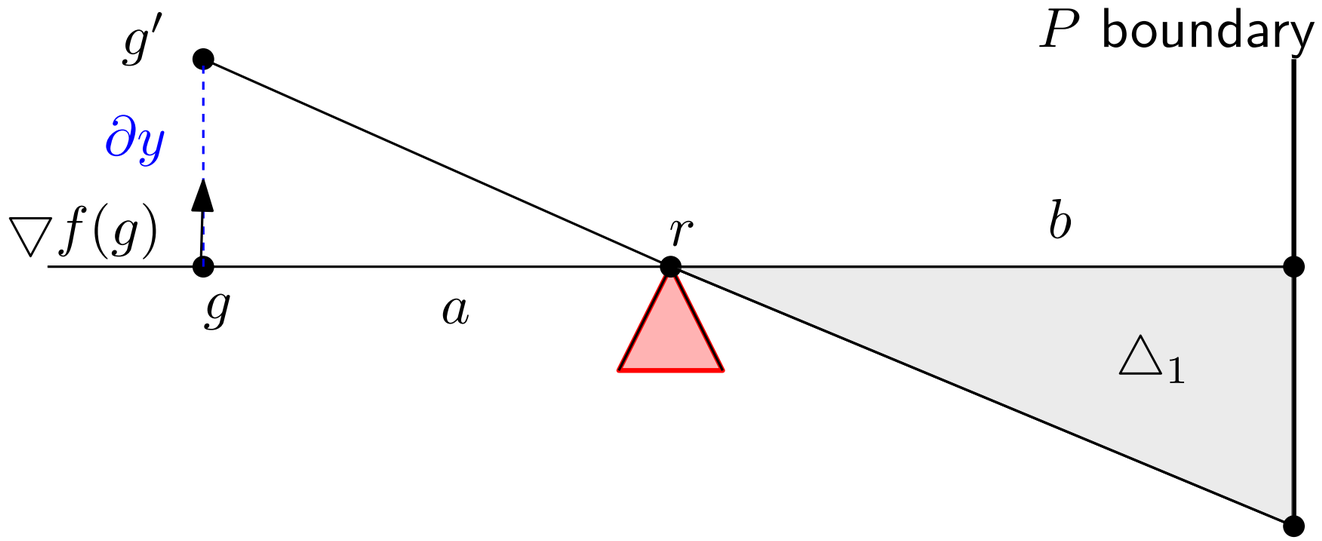


Computing the gradient

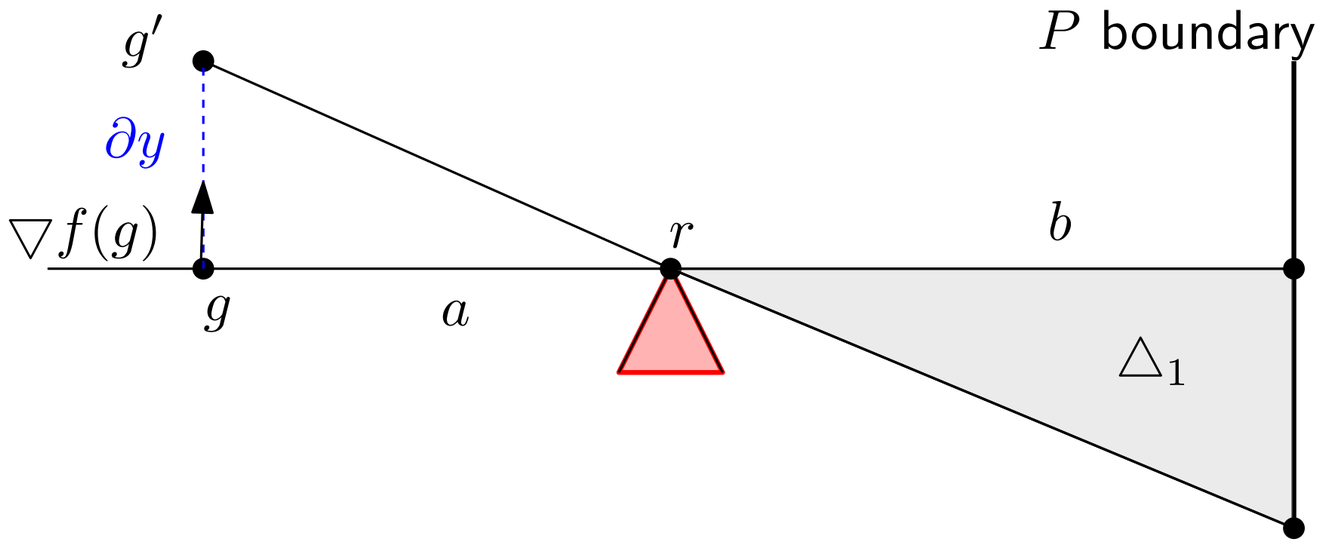


Computing the gradient

$$\bigtriangledown f(g) = \bigtriangledown \text{ Area}_{\bigtriangleup_1}(g)$$
$$\bigtriangledown f(g) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)^{\mathsf{T}}$$



Computing the gradient

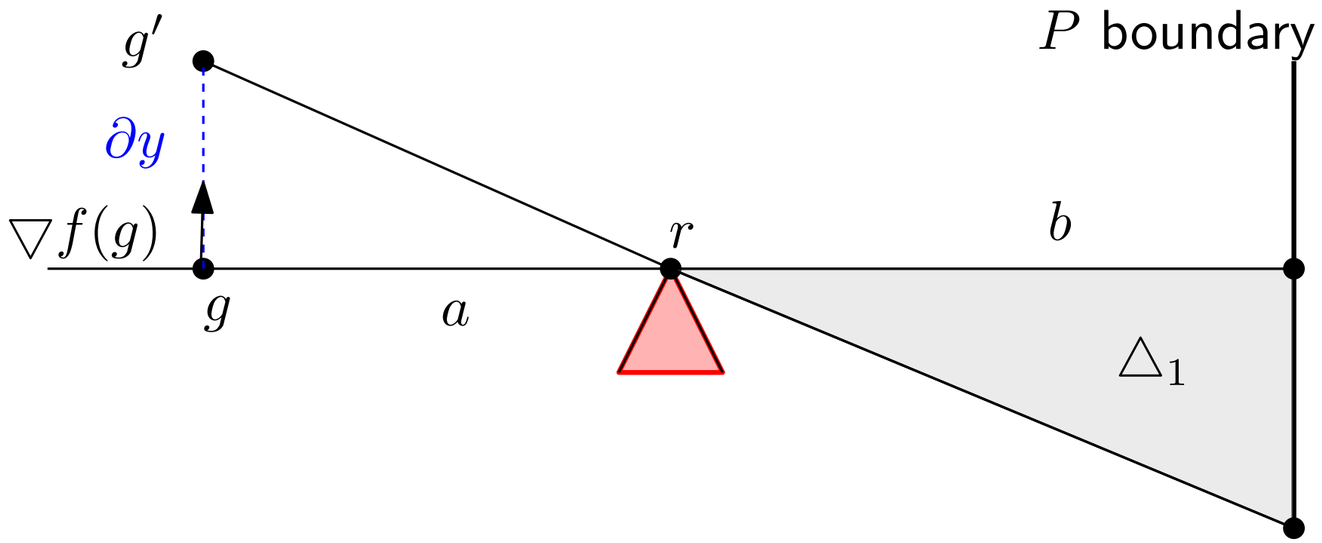


$$\nabla f(g) = \nabla \text{ Area}_{\triangle_1}(g)$$

$$\nabla f(g) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)^\intercal$$

$$\nabla f(g) = \left(0, \frac{b^2}{2a}\right)^\intercal$$

Computing the gradient



$$\nabla f(g) = \nabla \text{ Area}_{\triangle_1}(g)$$

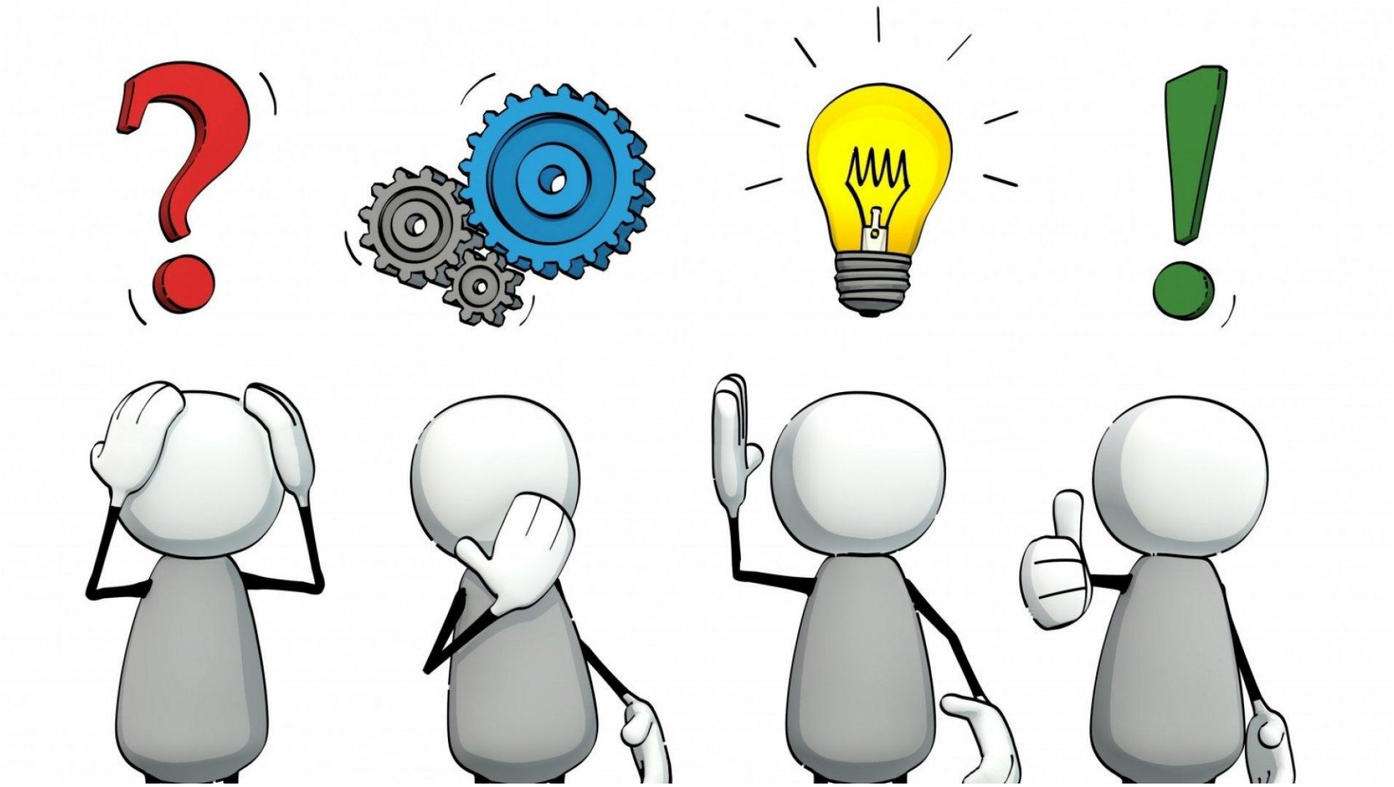
$$\nabla f(g) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)^\intercal$$

$$\nabla f(g) = \left(0, \frac{b^2}{2a}\right)^\intercal$$

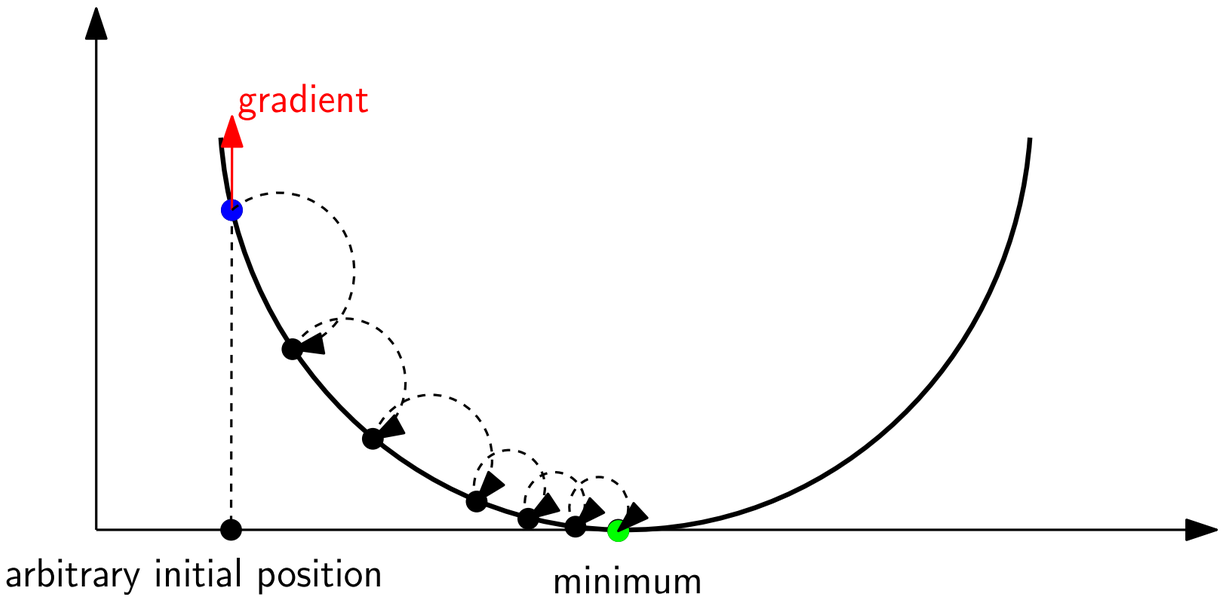
$$g' = g + \alpha \nabla f(g)$$

α - learning rate

Heuristics

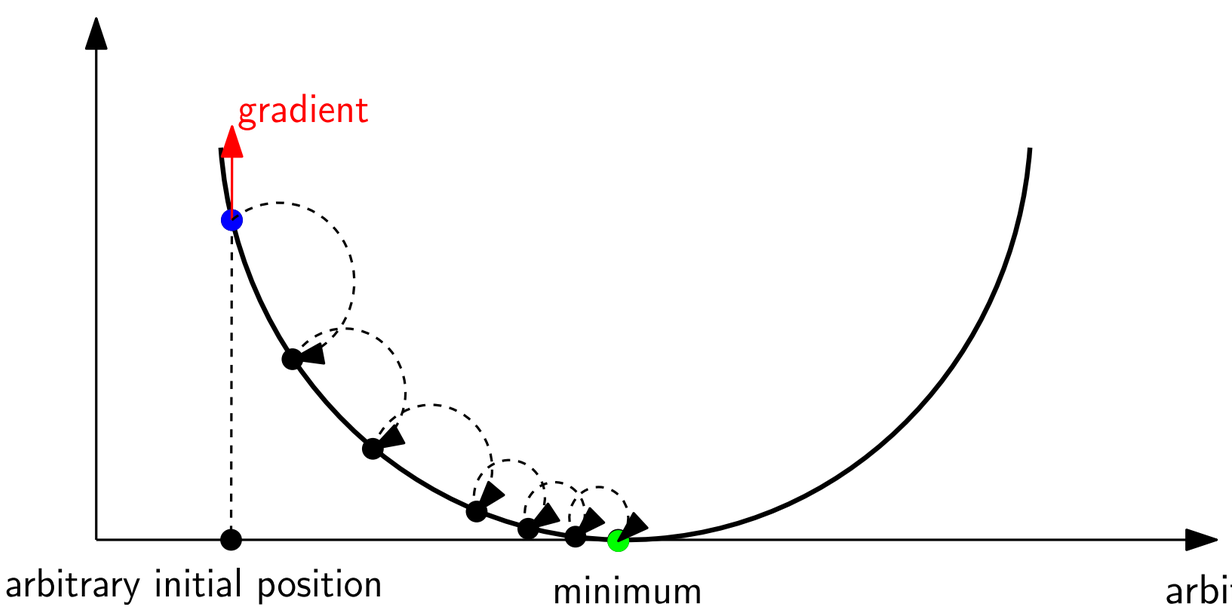


Heuristics: Momentum

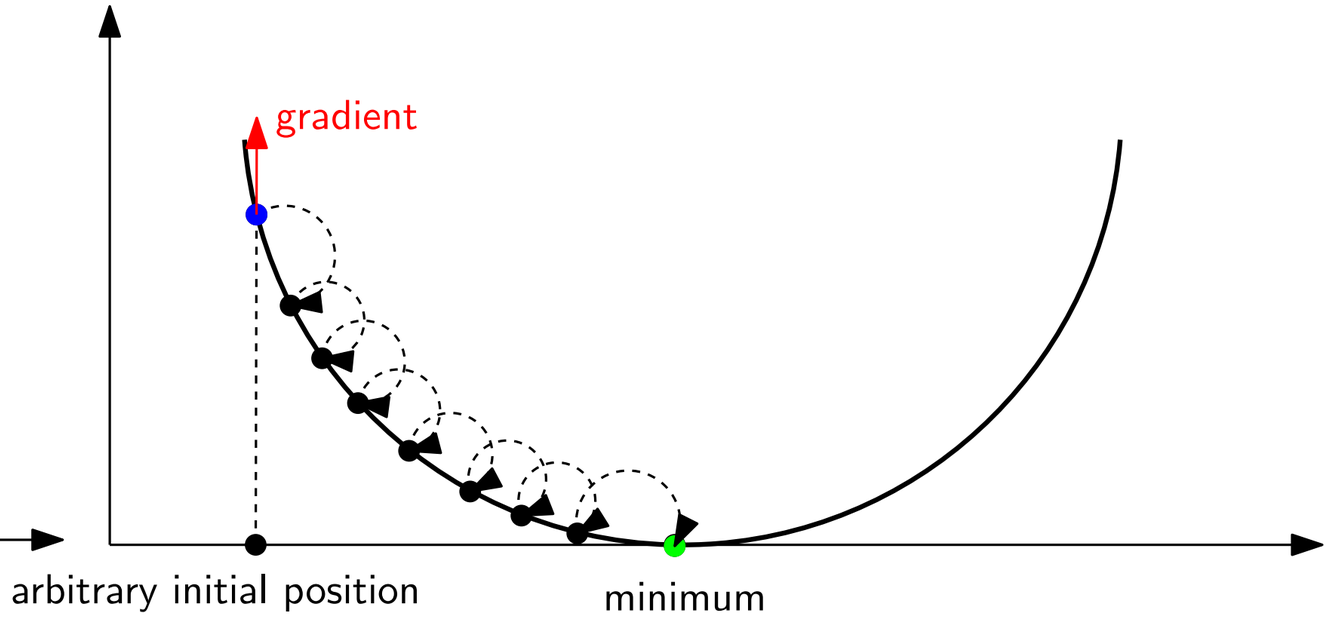


No Momentum

Heuristics: Momentum



No Momentum



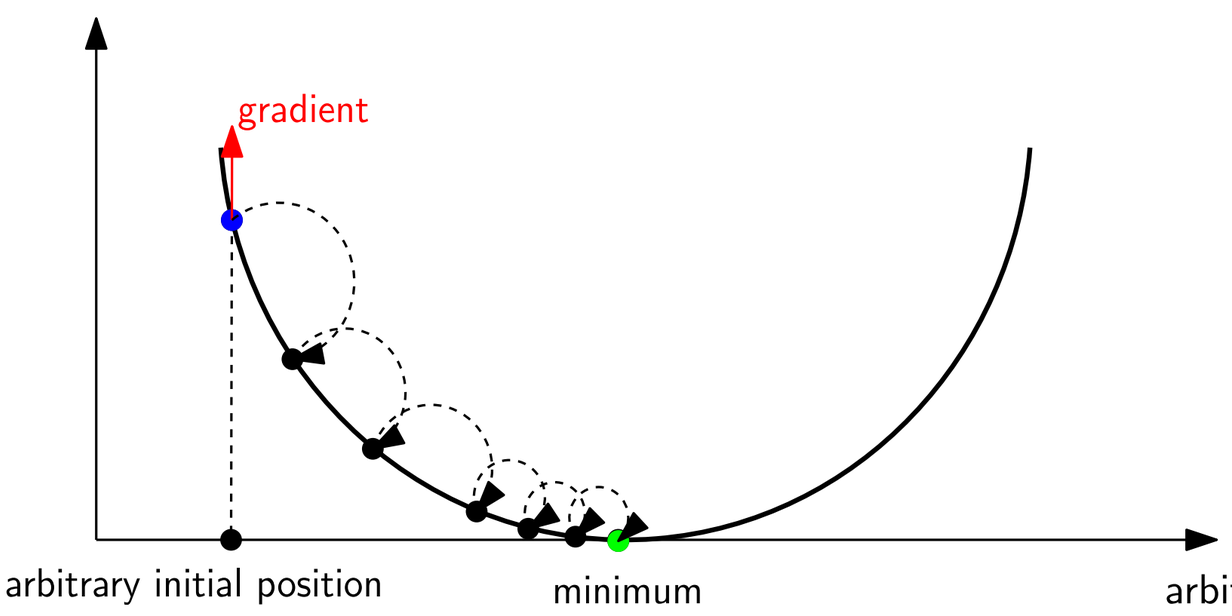
With Momentum

Heuristics: Momentum

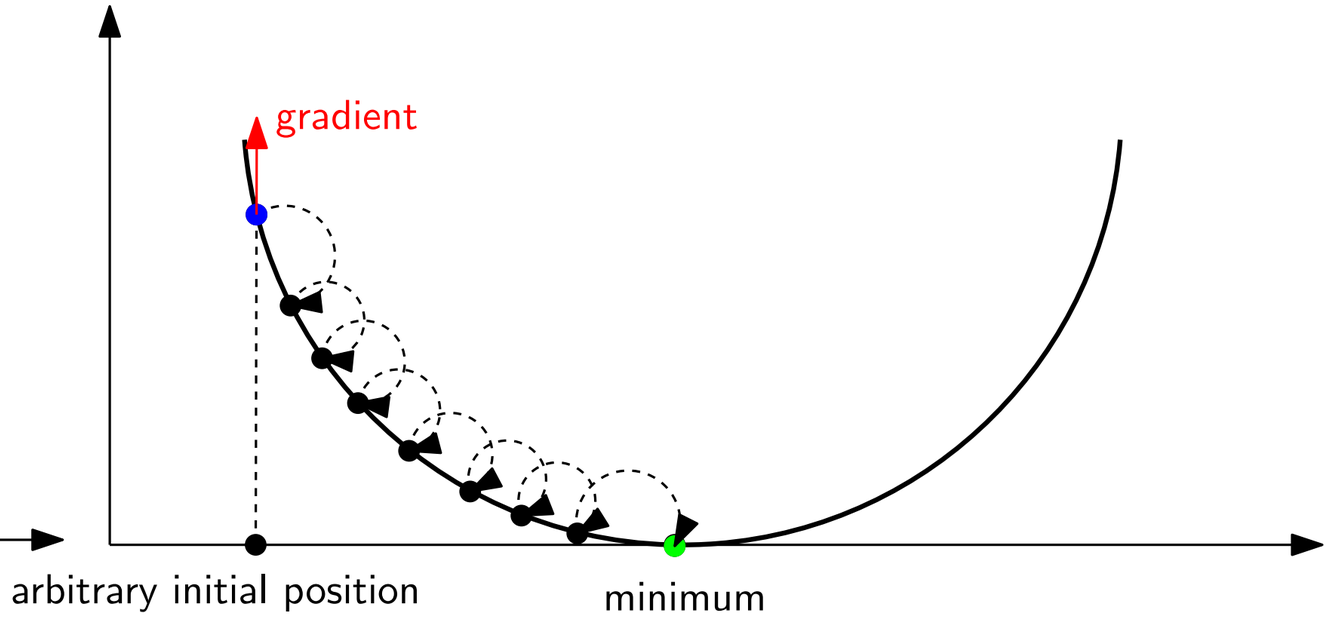
iteration i:

γ - past state weight

$$M_i(g_i) = \gamma M_{i-1}(g_{i-1}) + (1-\gamma) \bigtriangledown f_i(g_i)$$



No Momentum



With Momentum

Heuristics: Momentum

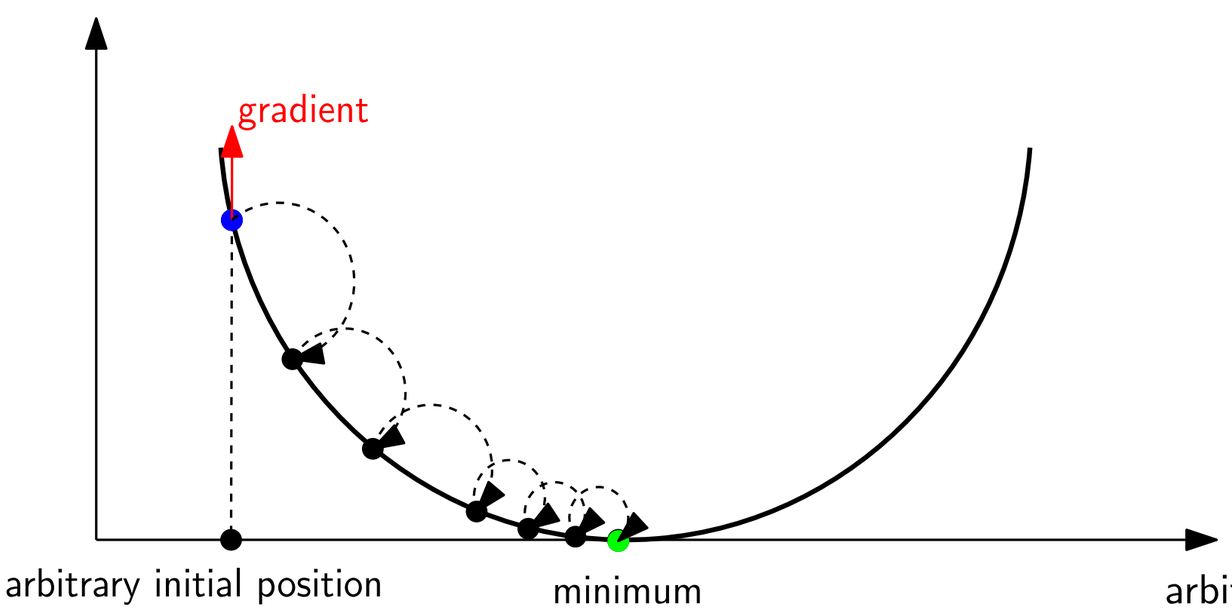
iteration i:

γ - past state weight

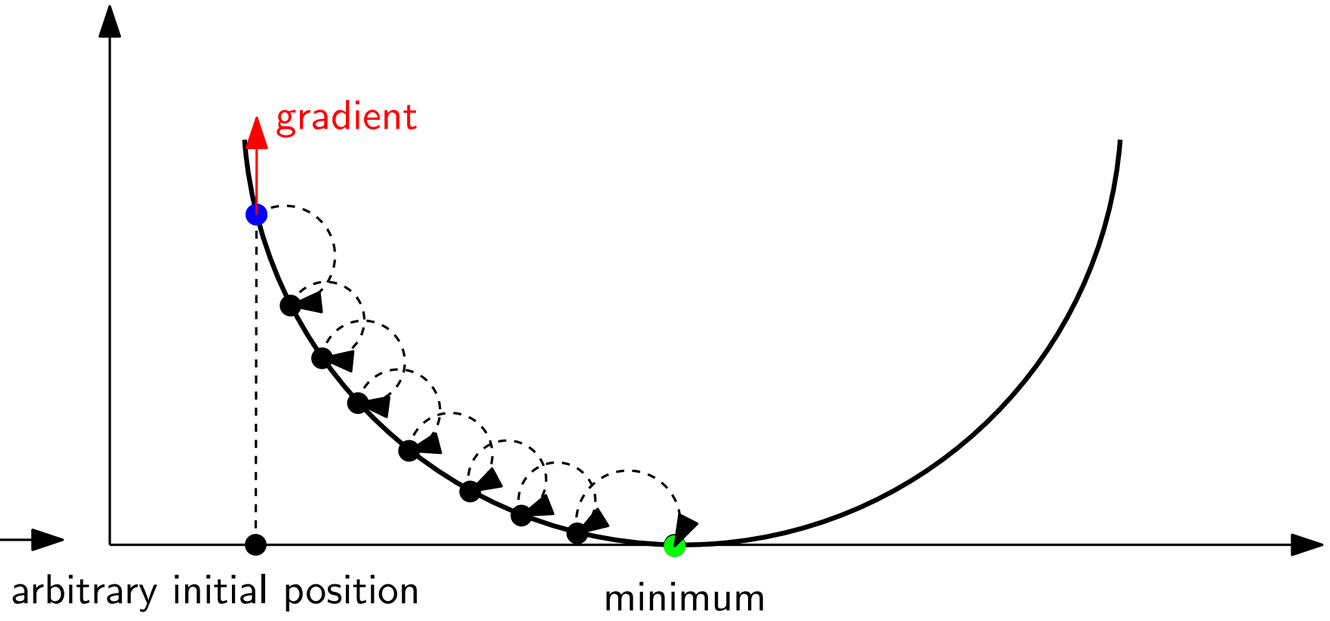
$$M_i(g_i) = \gamma M_{i-1}(g_{i-1}) + (1-\gamma) \bigtriangledown f_i(g_i)$$

$$g_i = g_{i-1} + \alpha M_i(g_{i-1})$$

$$g_i = g_{i-1} + \alpha \bigtriangledown f_i(g_i)$$

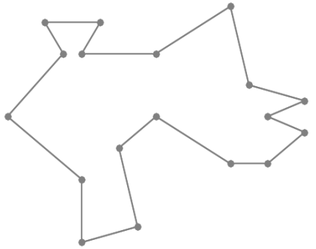


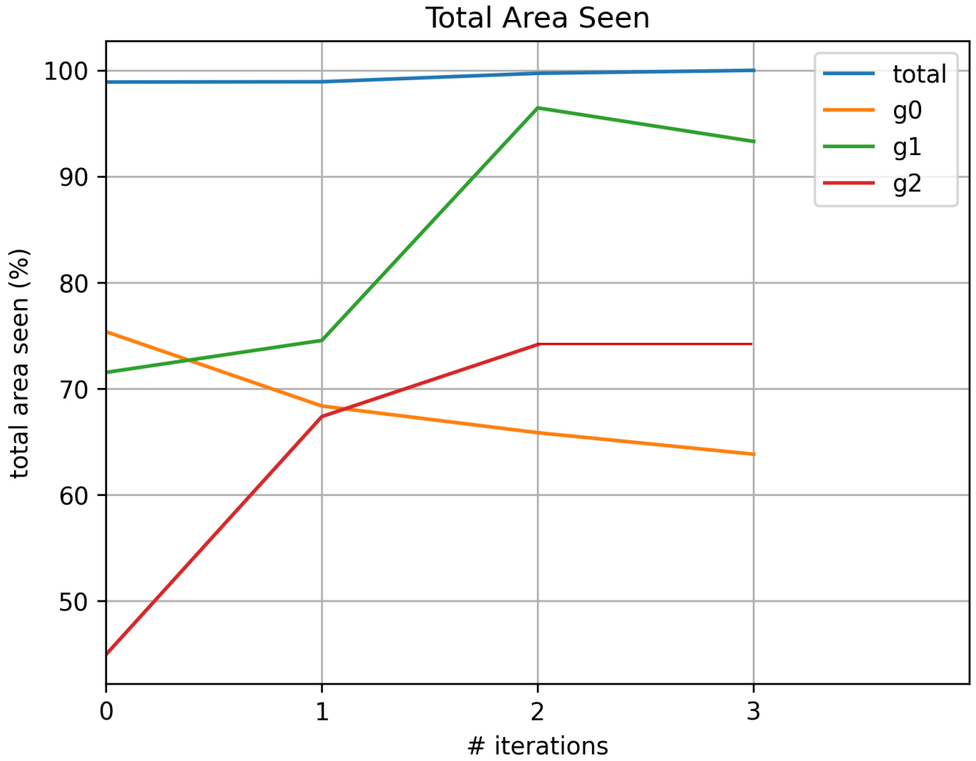
No Momentum



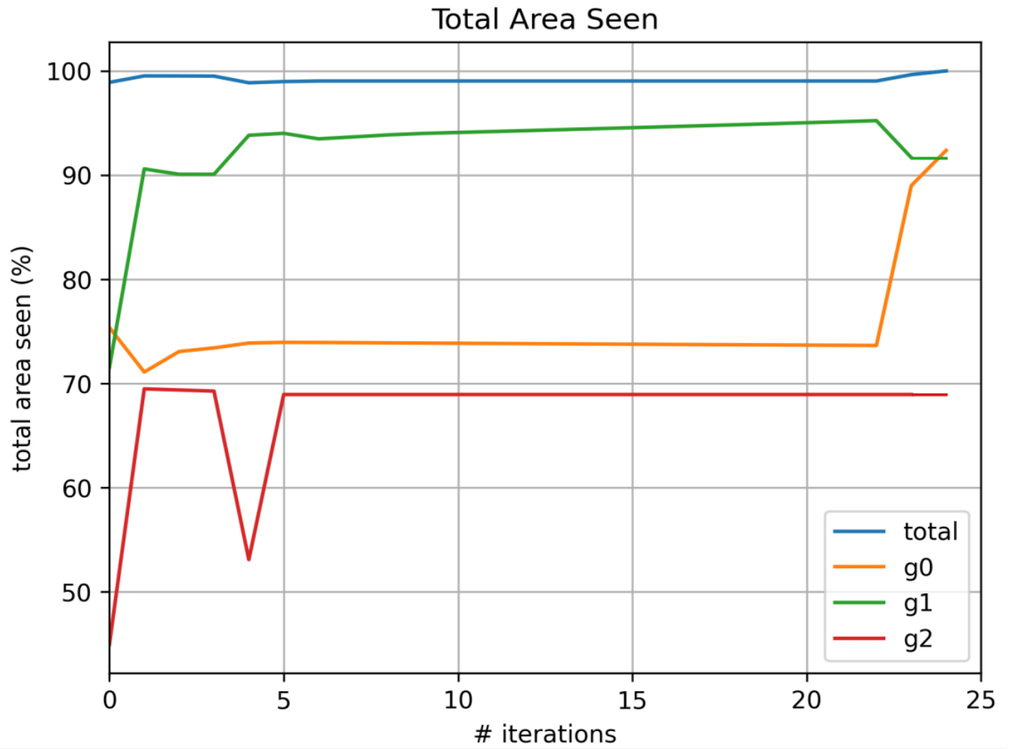
With Momentum

Heuristics: Momentum



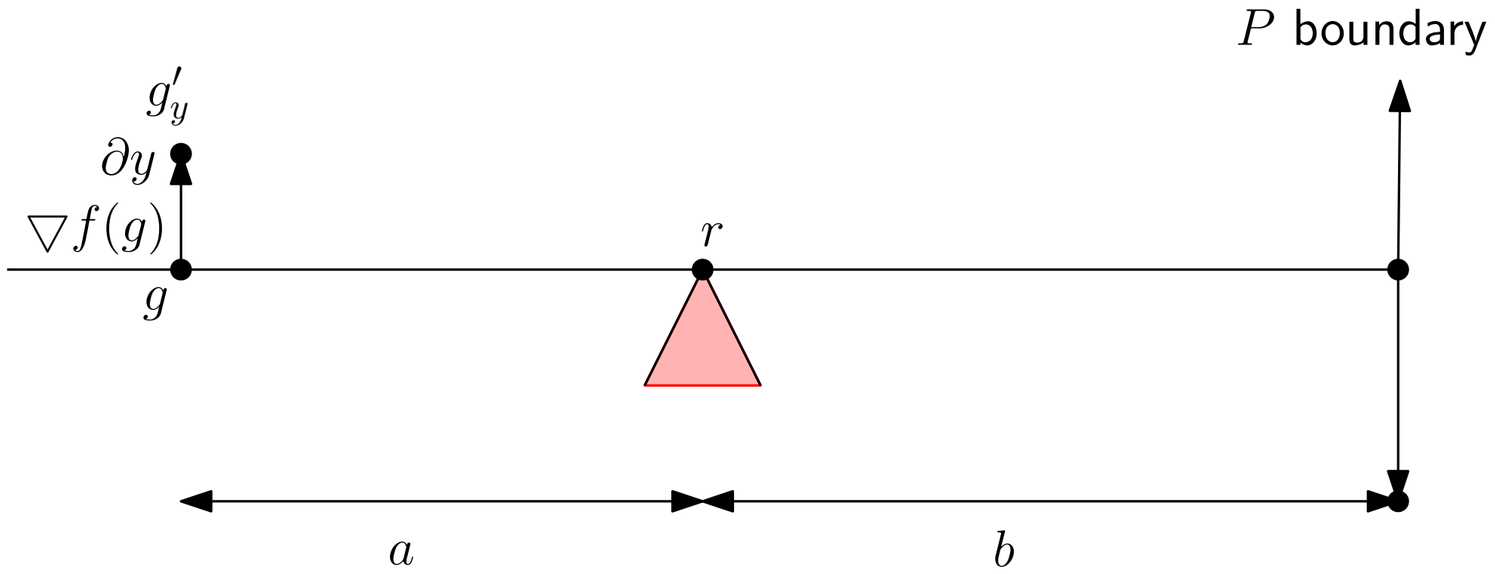


All heuristics



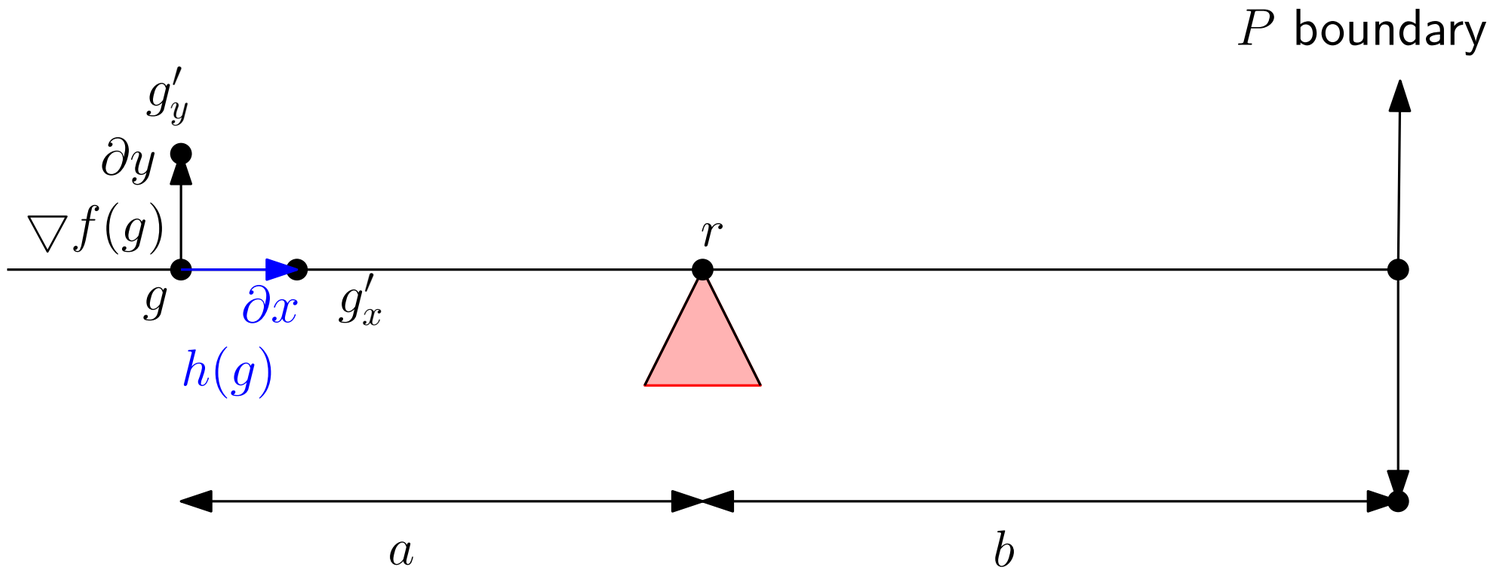
No momentum

Heuristics: Pull towards reflex vertex

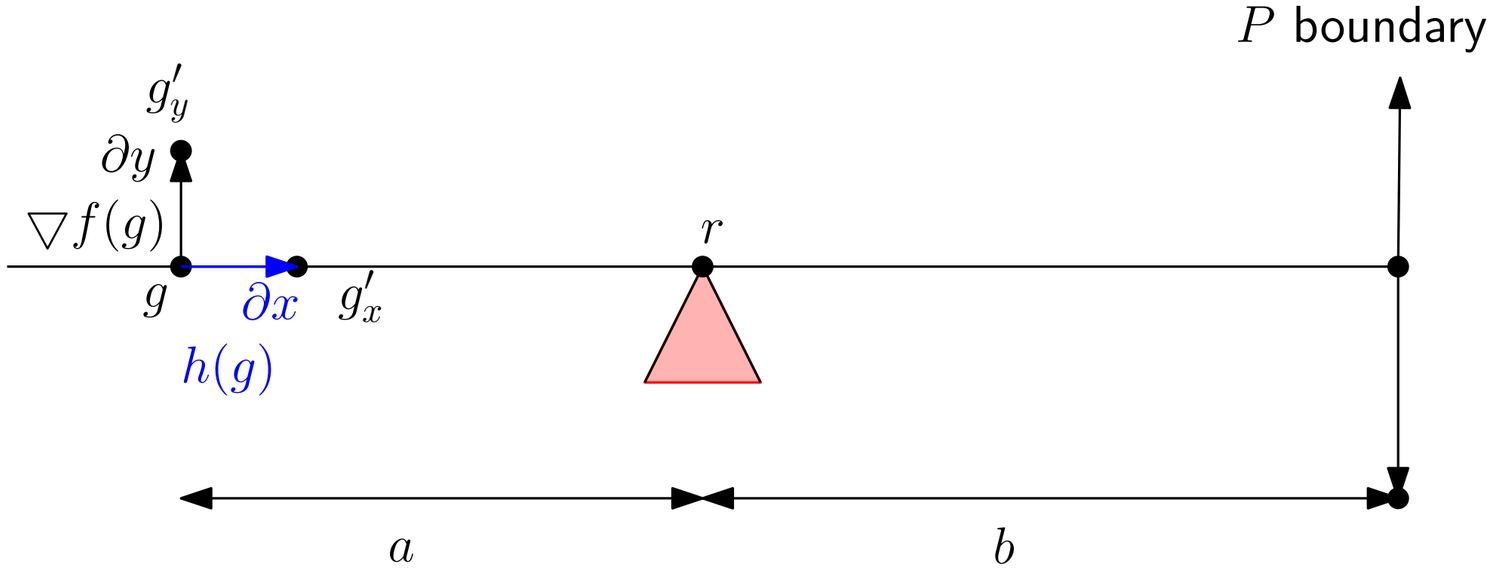


Heuristics: Pull towards reflex vertex

$$h(g) = \bigtriangledown || \bigtriangledown f(g) ||$$



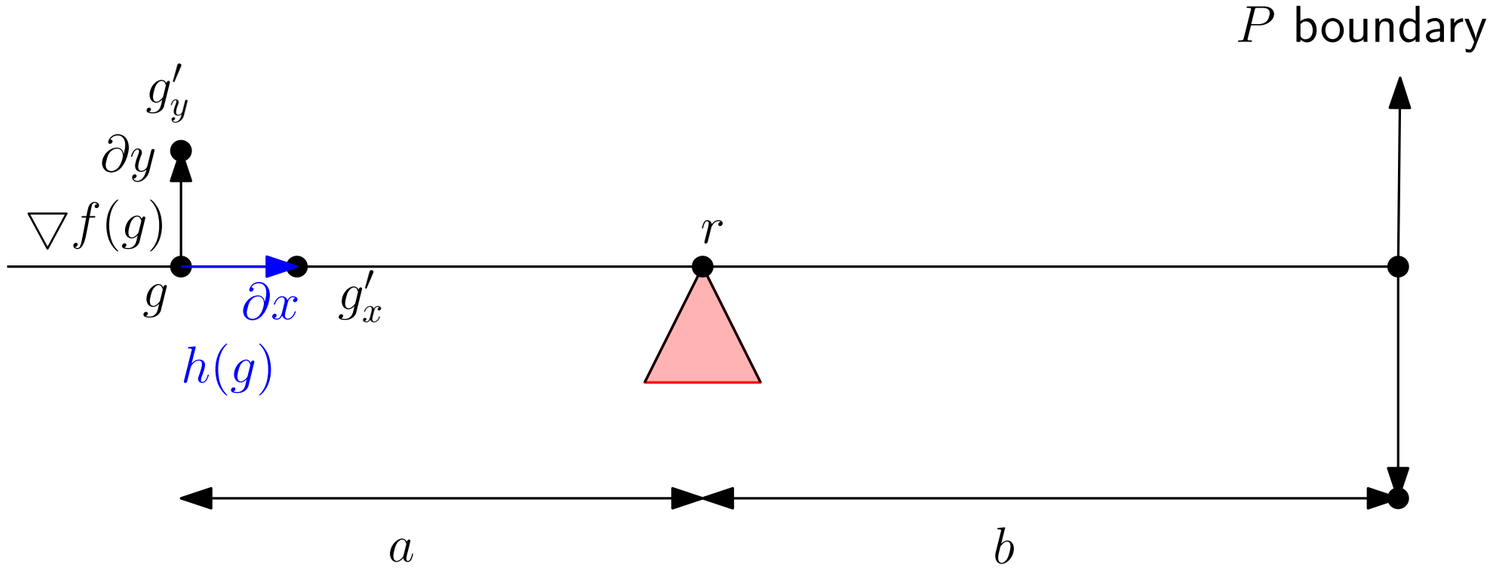
Heuristics: Pull towards reflex vertex



$$h(g) = \bigtriangledown || \bigtriangledown f(g)||$$

$$h(g) = \left(\frac{\partial \bigtriangledown f(g)}{\partial x}, \frac{\partial \bigtriangledown f(g)}{\partial y} \right)^{\intercal}$$

Heuristics: Pull towards reflex vertex

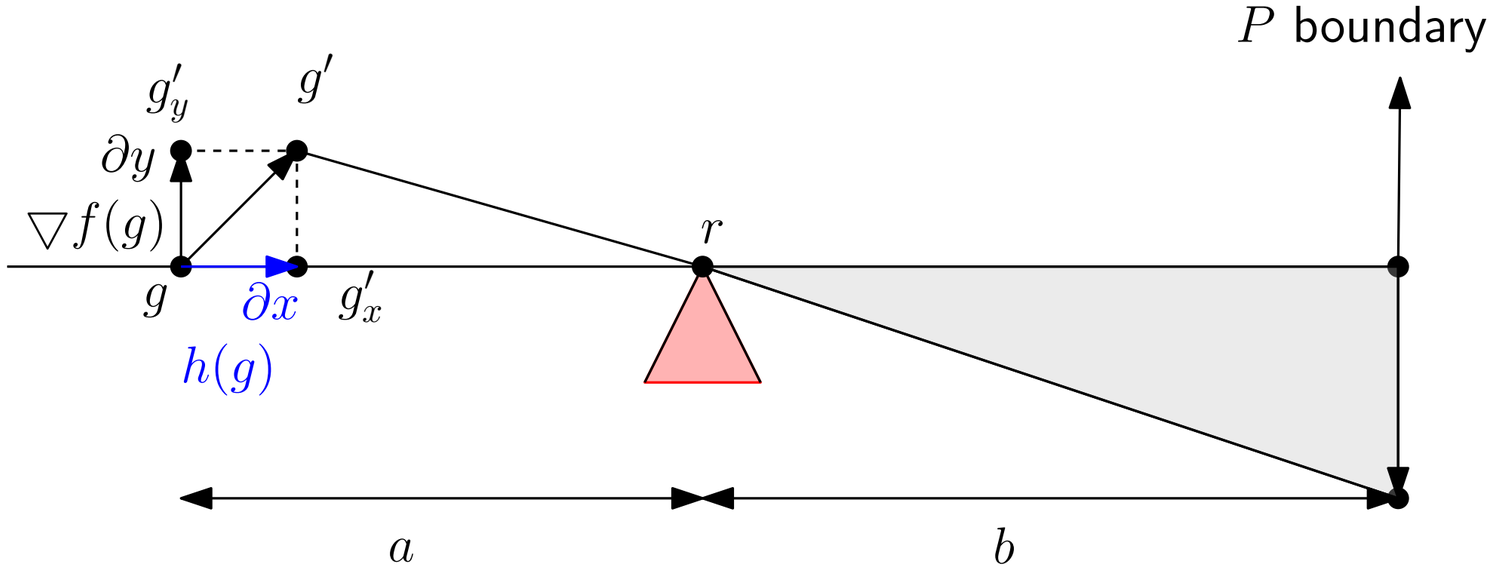


$$h(g) = \bigtriangledown || \bigtriangledown f(g)||$$

$$h(g) = \left(\frac{\partial \bigtriangledown f(g)}{\partial x}, \frac{\partial \bigtriangledown f(g)}{\partial y} \right)^{\intercal}$$

$$h(g) = \left(\frac{-b^2}{2a^3}, 0 \right)^{\intercal}$$

Heuristics: Pull towards reflex vertex



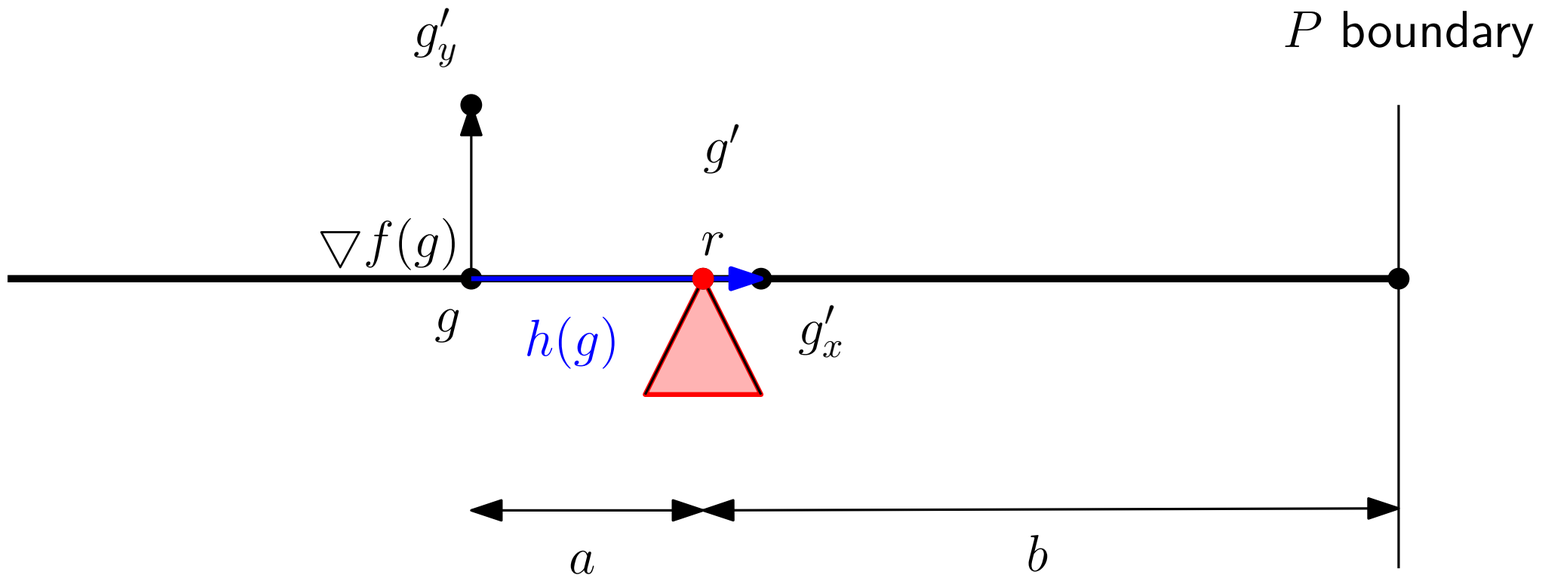
$$h(g) = \nabla || \nabla f(g) ||$$

$$h(g) = \left(\frac{\partial \nabla f(g)}{\partial x}, \frac{\partial \nabla f(g)}{\partial y} \right)^{\intercal}$$

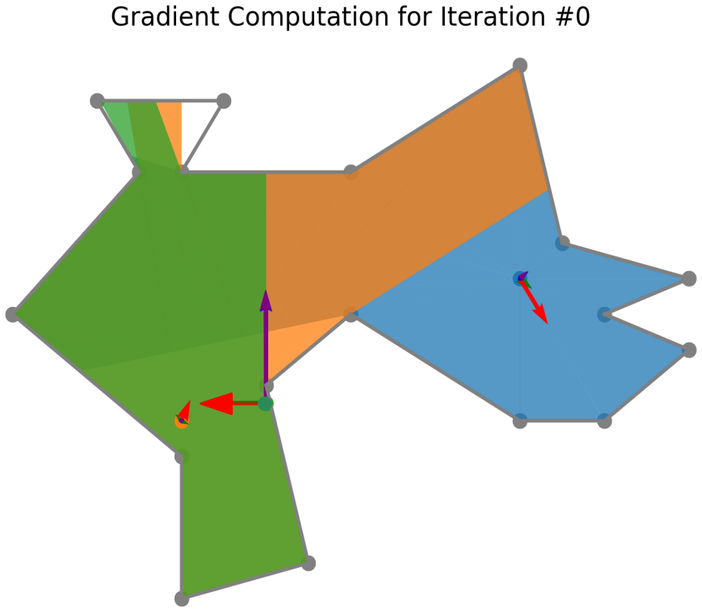
$$h(g) = \left(\frac{-b^2}{2a^3}, 0 \right)^{\intercal}$$

$$g' = g + \alpha(\nabla f(g) + h(g))$$

Heuristics: Pull towards reflex vertex

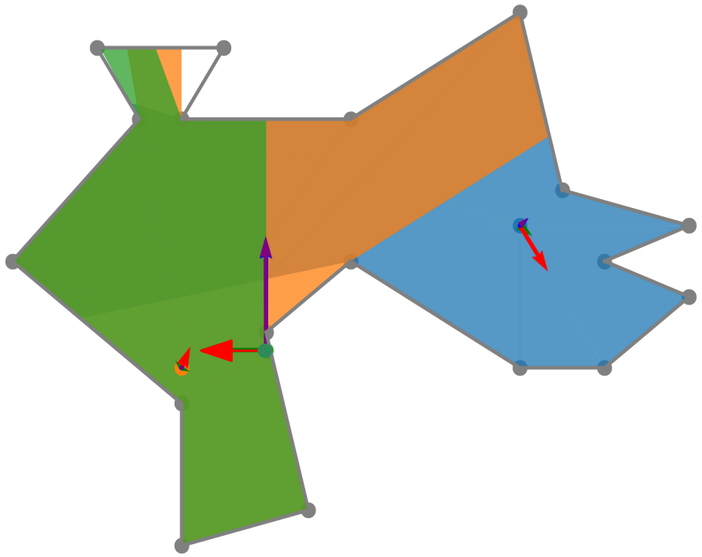


Heuristics: Pull towards reflex vertex



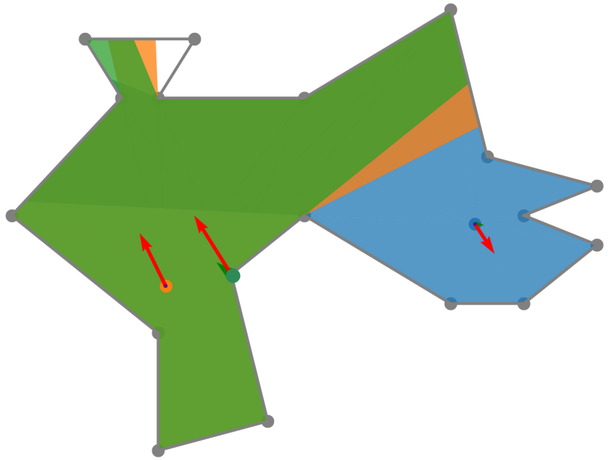
Heuristics: Pull towards reflex vertex

Gradient Computation for Iteration #0



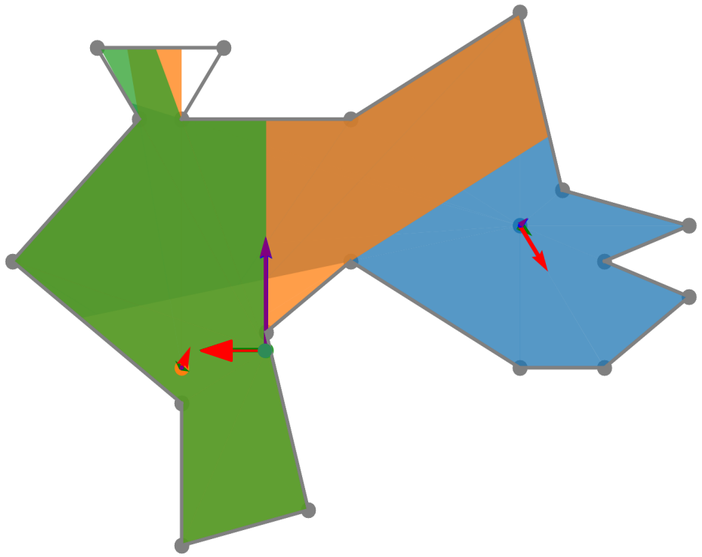
with pull

Gradient Computation for Iteration #1



Heuristics: Pull towards reflex vertex

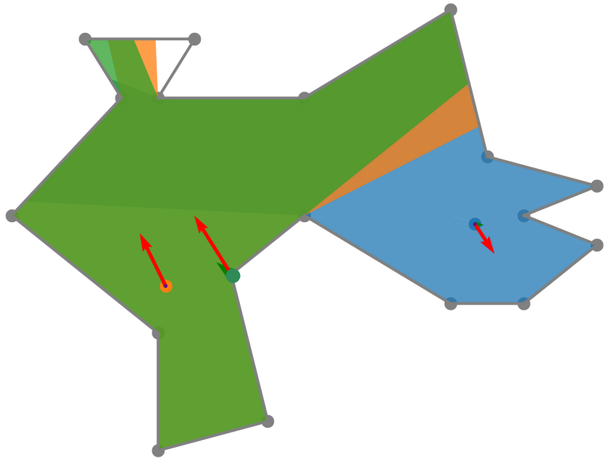
Gradient Computation for Iteration #0



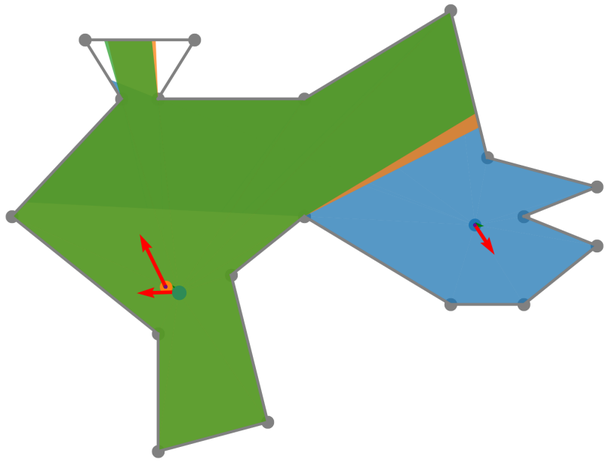
with pull

without pull

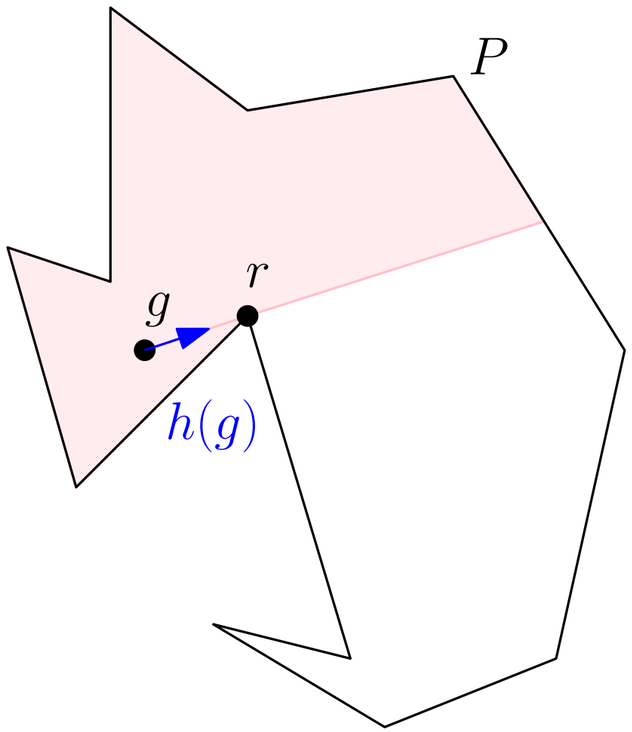
Gradient Computation for Iteration #1



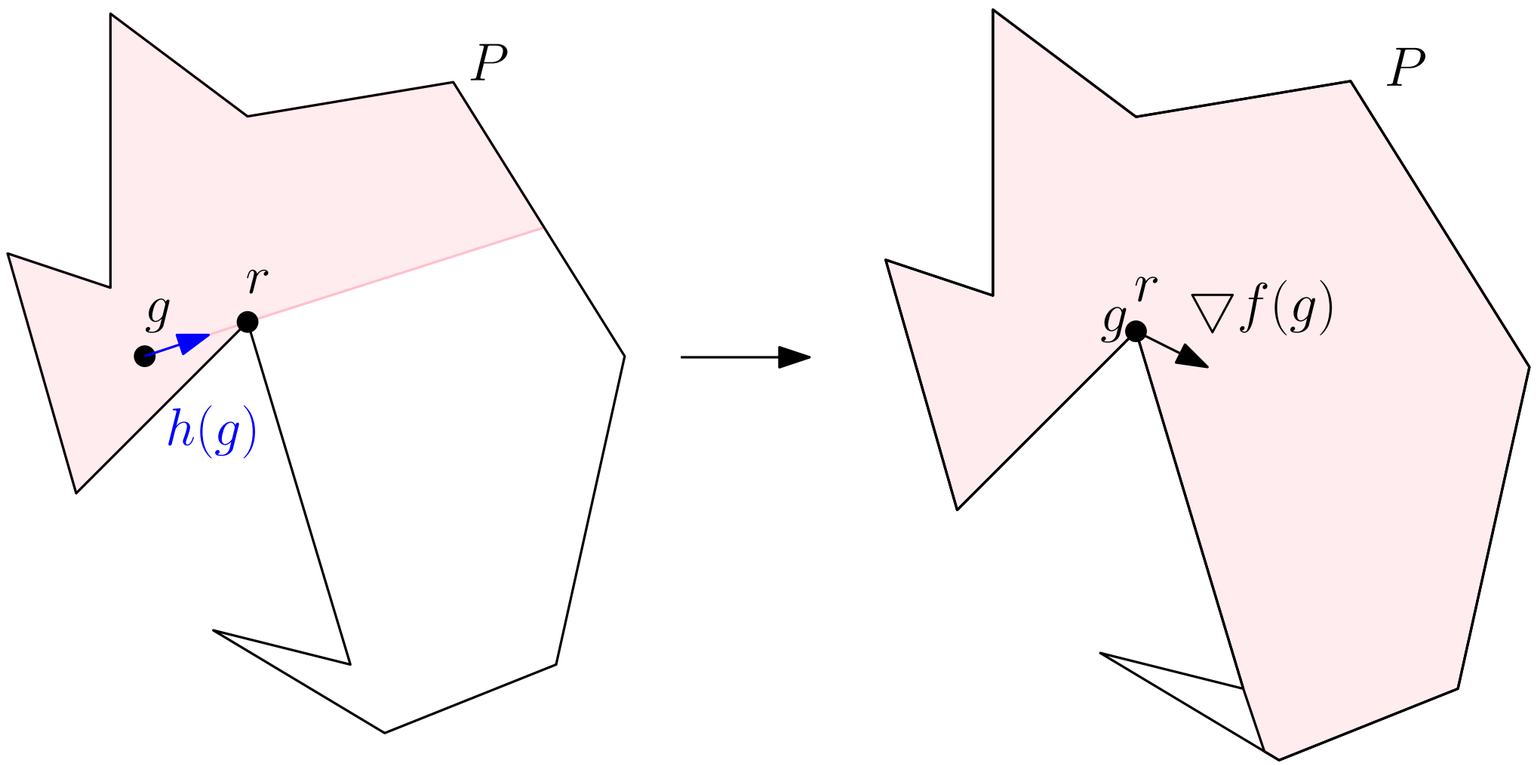
Gradient Computation for Iteration #1



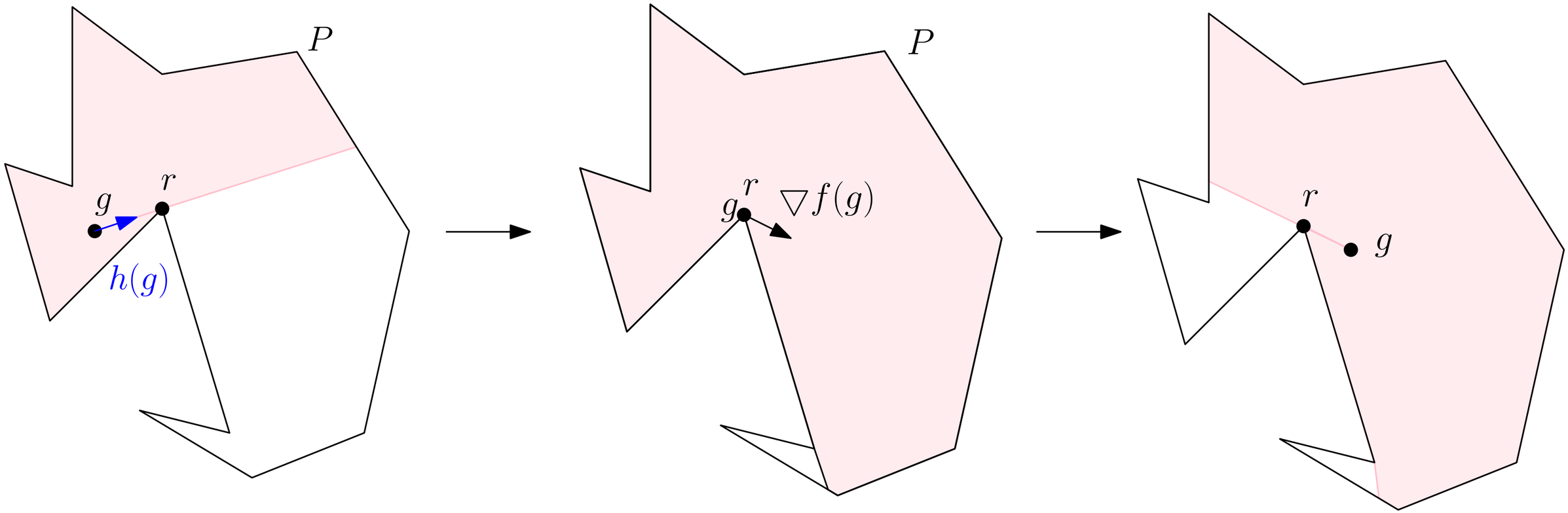
Heuristics: Reflex area



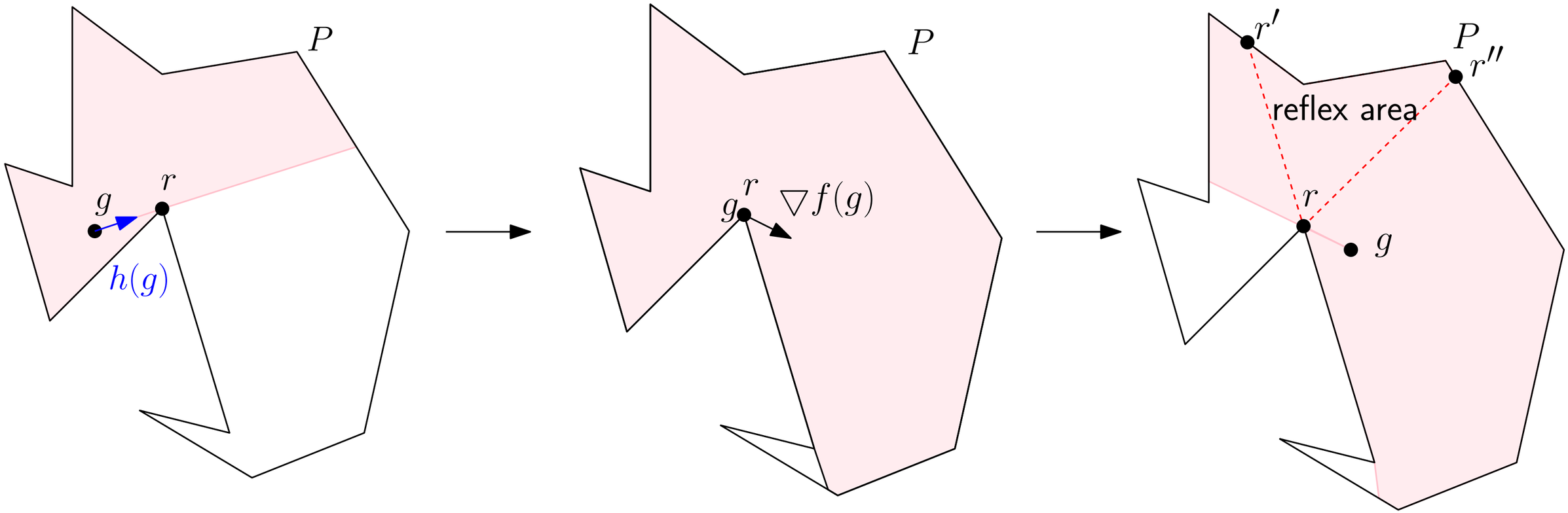
Heuristics: Reflex area



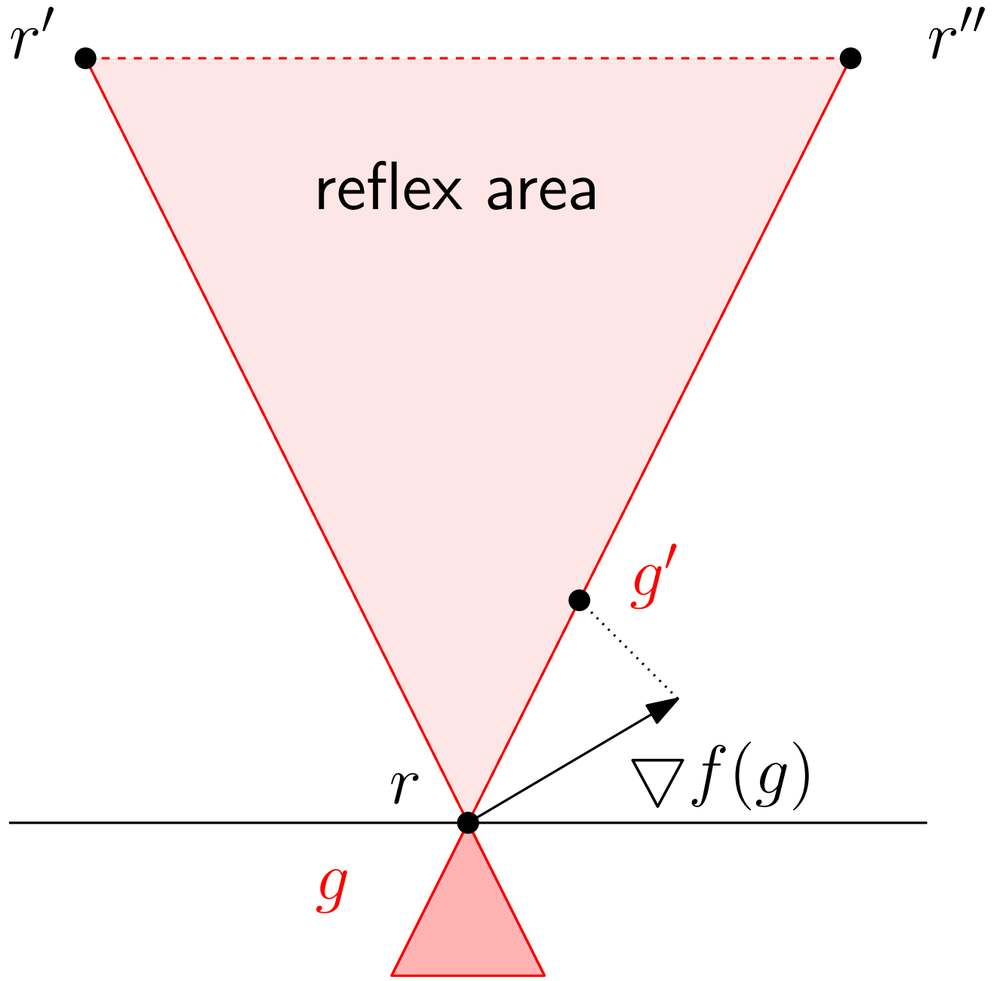
Heuristics: Reflex area



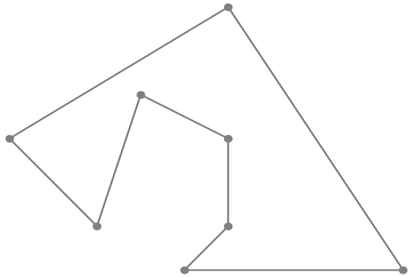
Heuristics: Reflex area

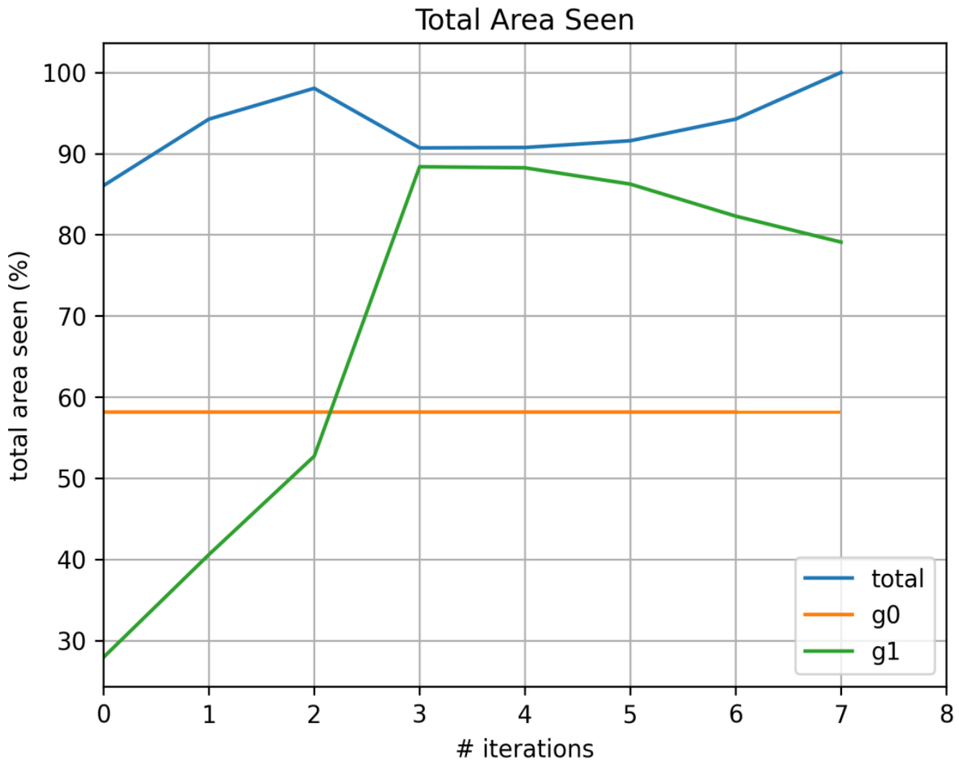


Heuristics: Reflex area

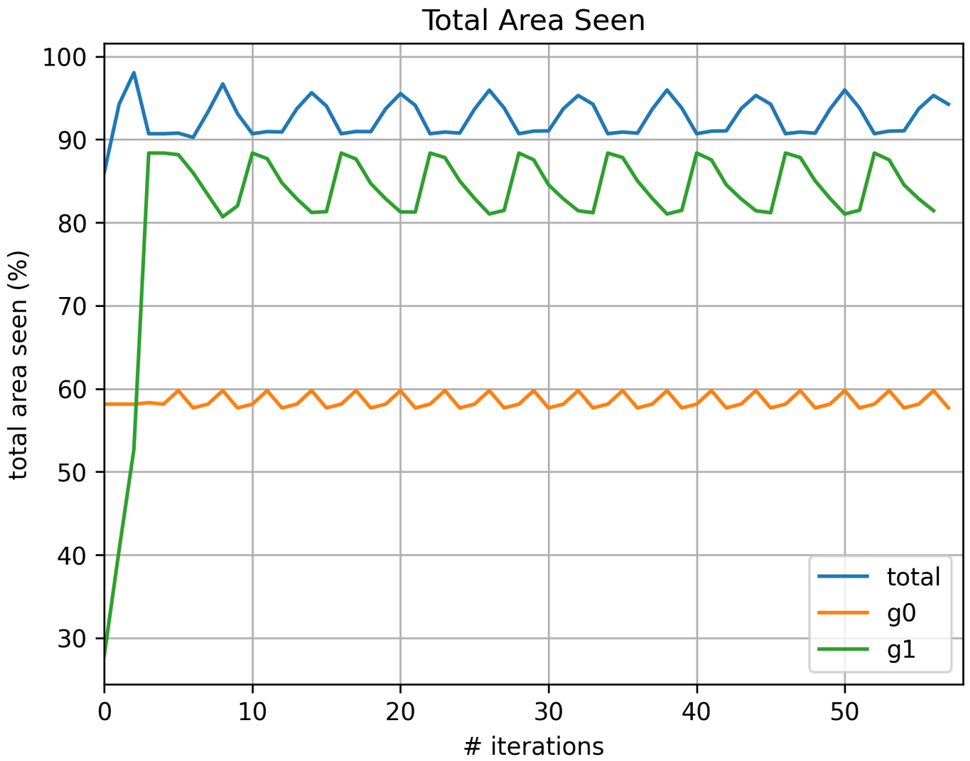


Heuristics: Reflex area





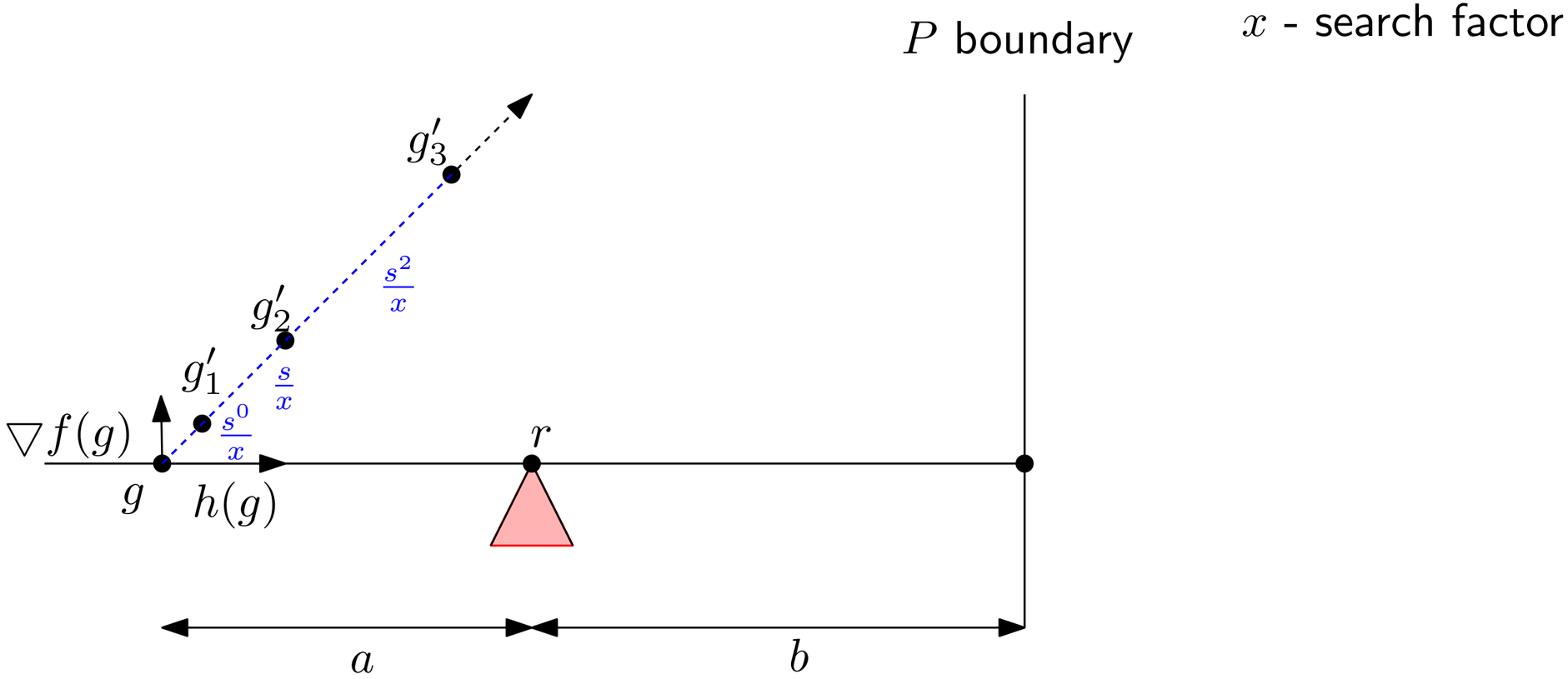
All heuristics



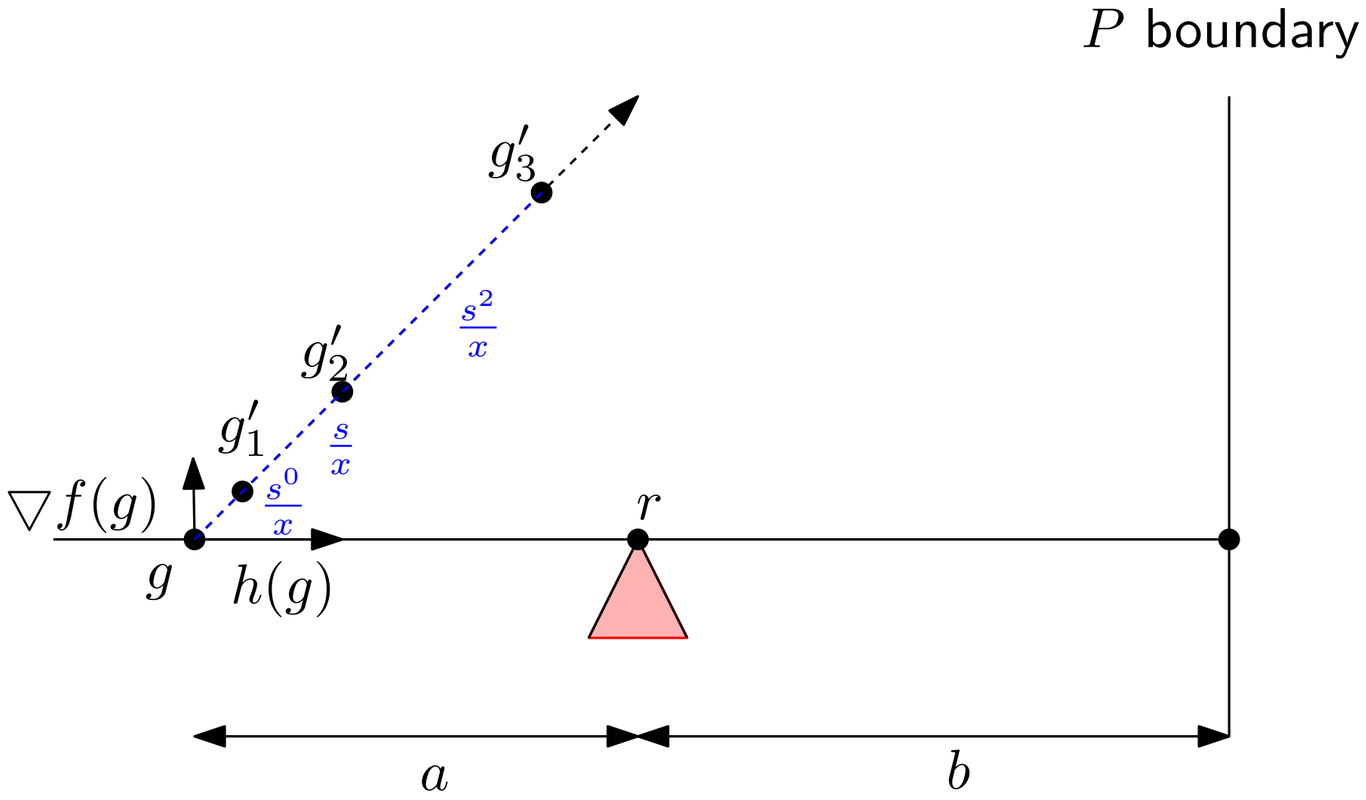
No reflex area

Heuristics: Line Search

s - step size
x - search factor



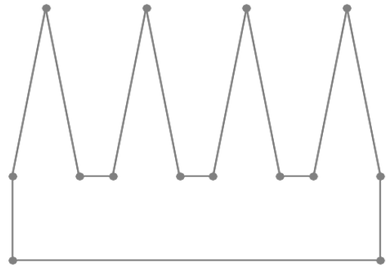
Heuristics: Line Search

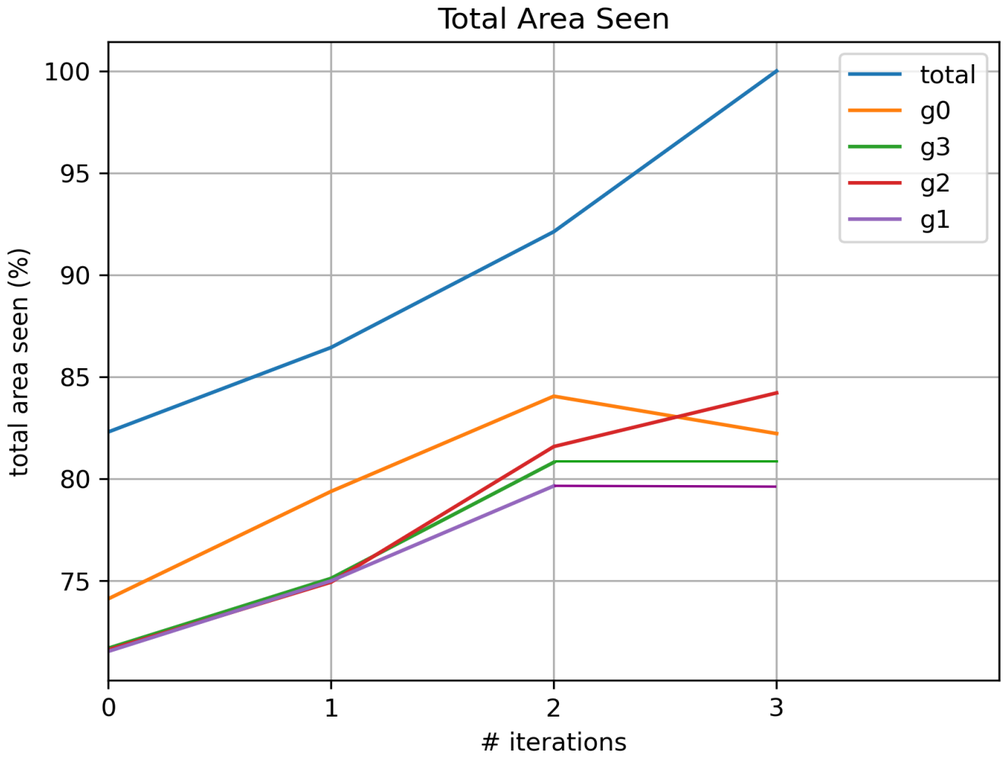


s - step size
x - search factor

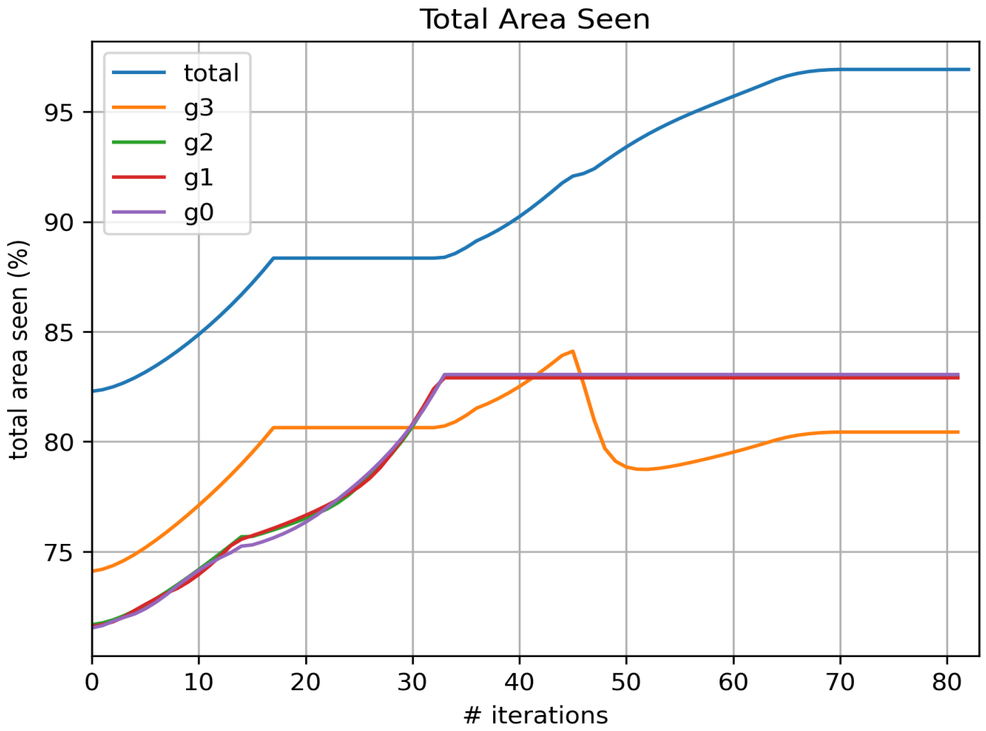
$$g'_1 = g + \frac{1}{x} M(g)$$
$$g'_2 = g + \frac{s}{x} M(g)$$
$$g'_3 = g + \frac{s^2}{x} M(g)$$

Heuristics: Line Search



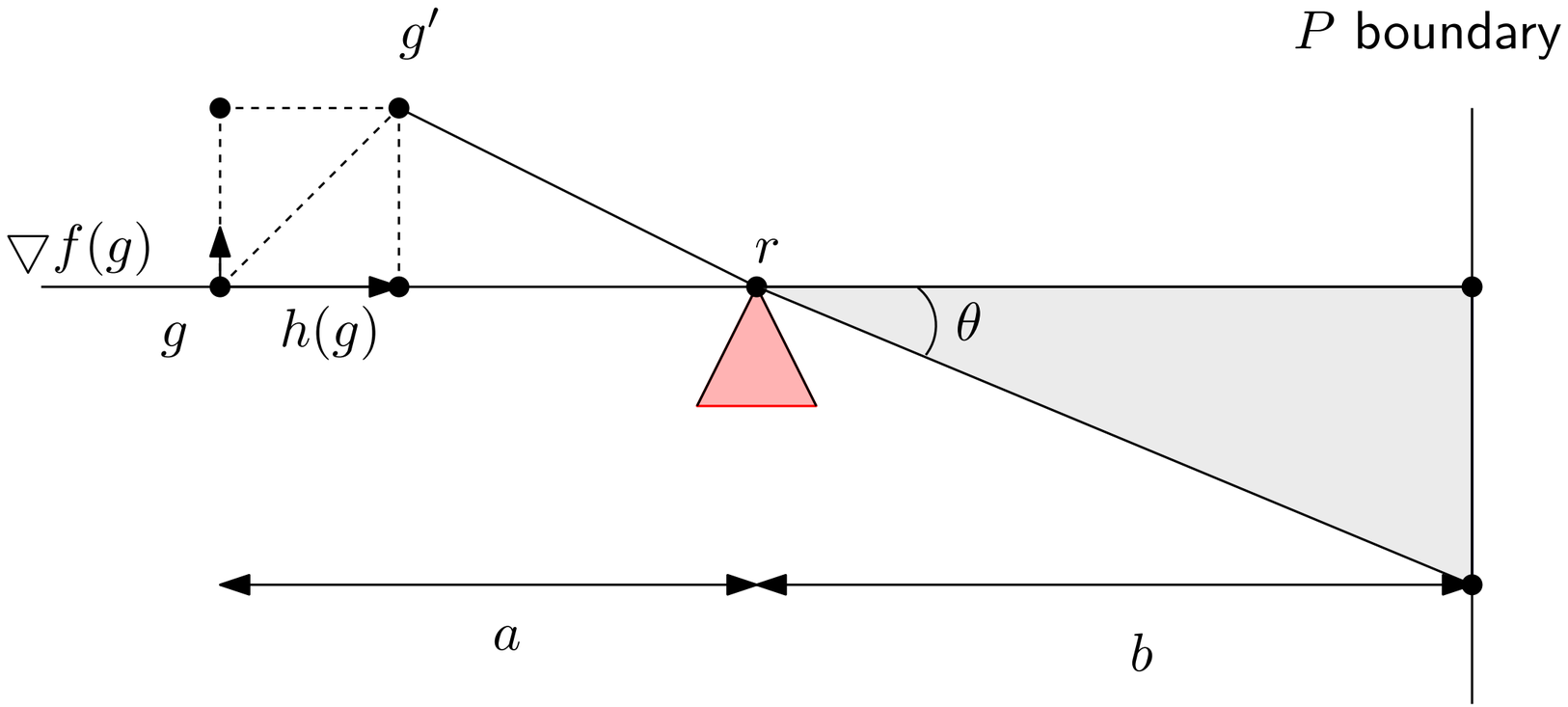


All heuristics



No line search

Heuristics: Angle behind reflex vertex

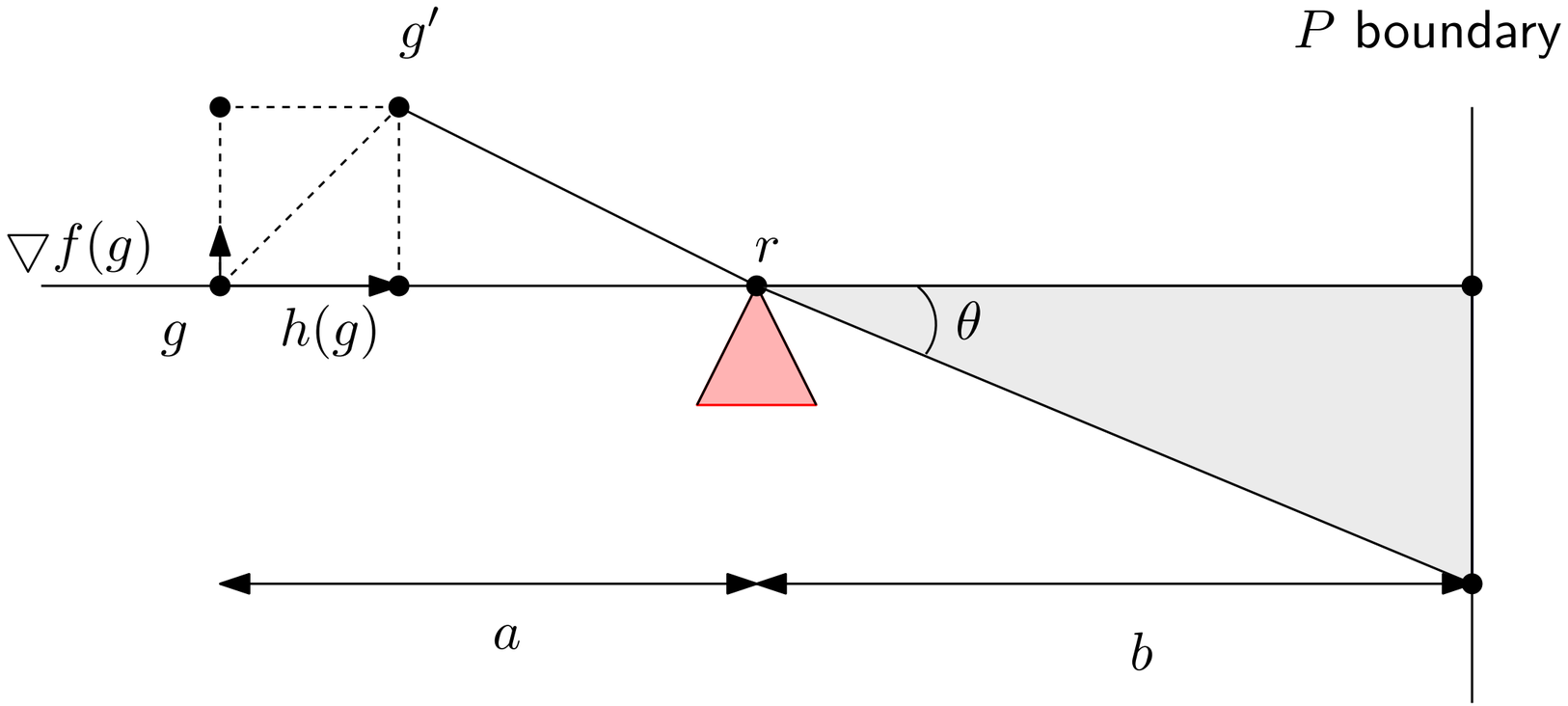


Heuristics: Angle behind reflex vertex

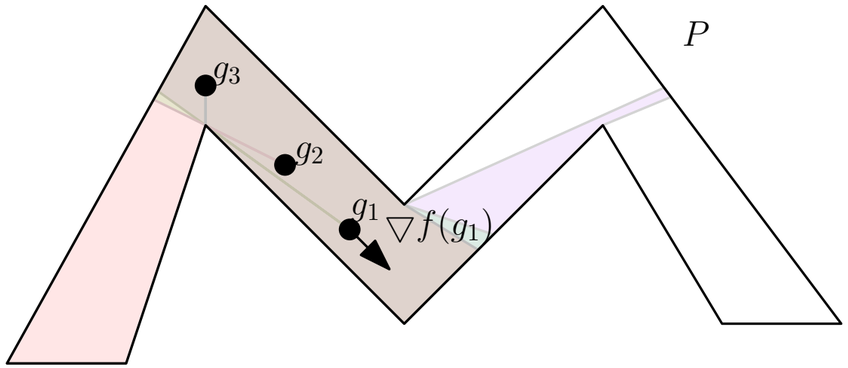
$$g' = g + \frac{s^t}{x}(\frac{\theta}{2\pi} + c)M(g)$$

c - constant

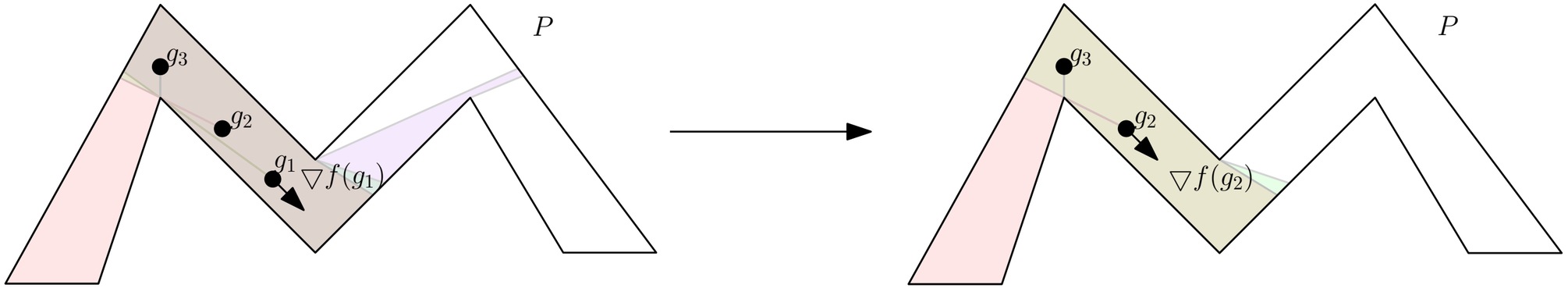
t - optimum step factor



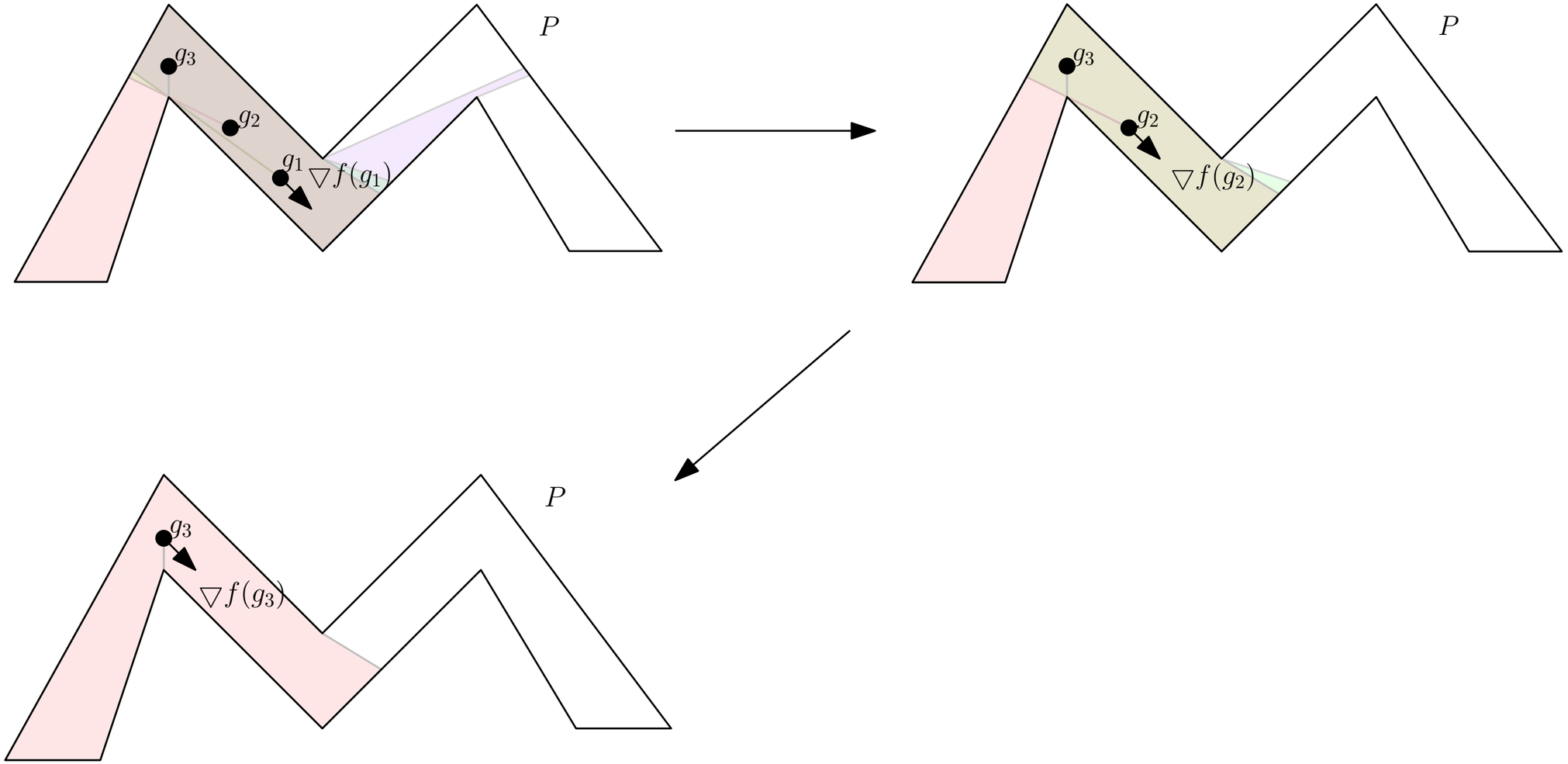
Heuristics: Hidden movement



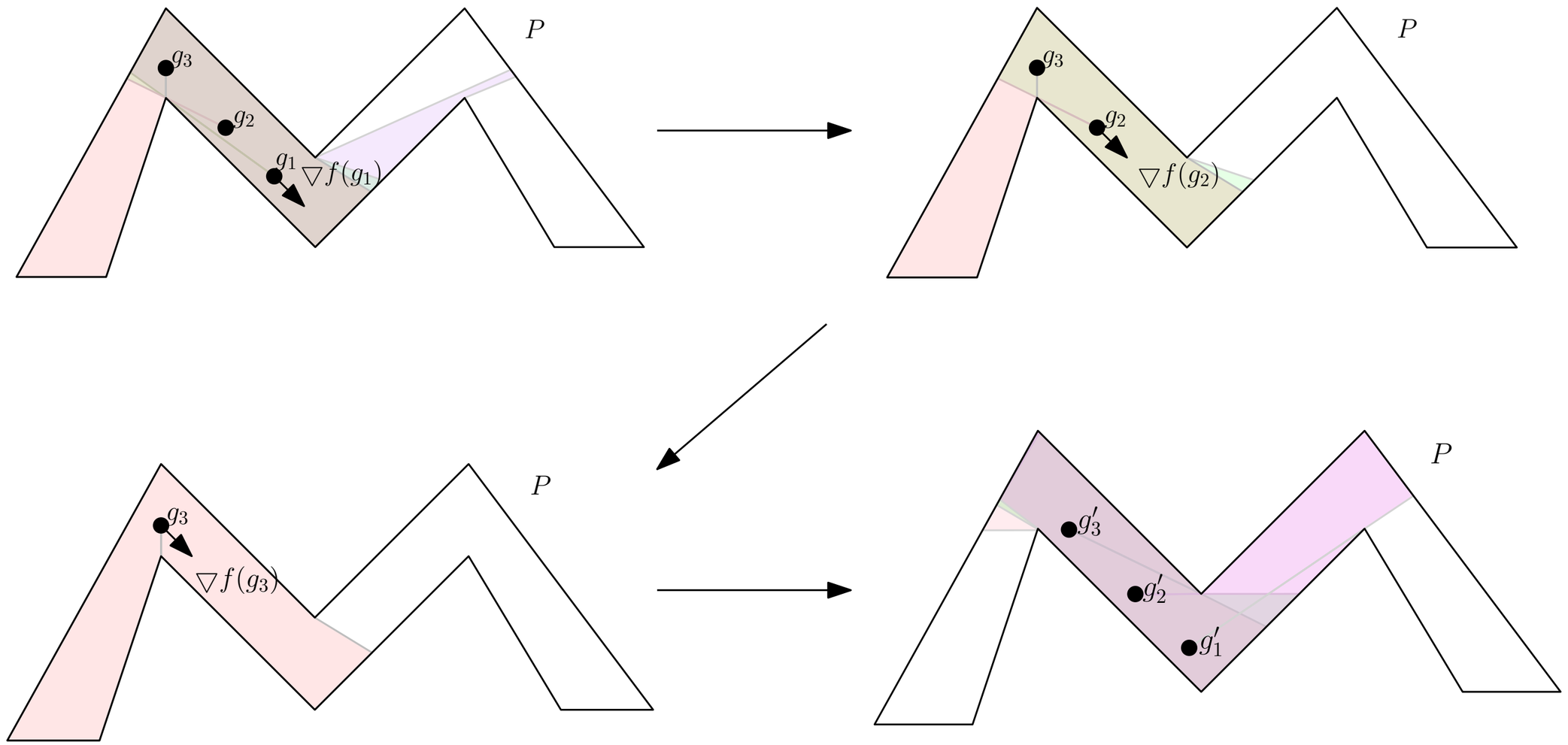
Heuristics: Hidden movement



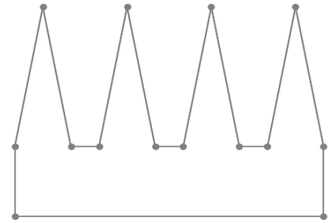
Heuristics: Hidden movement

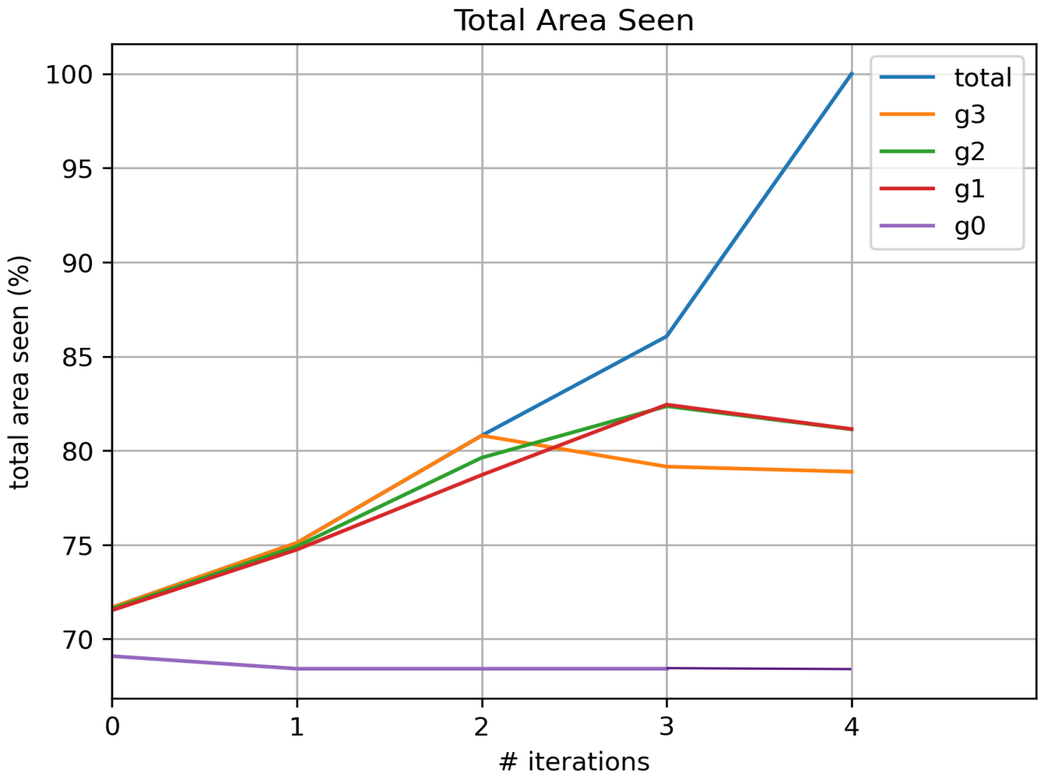


Heuristics: Hidden movement

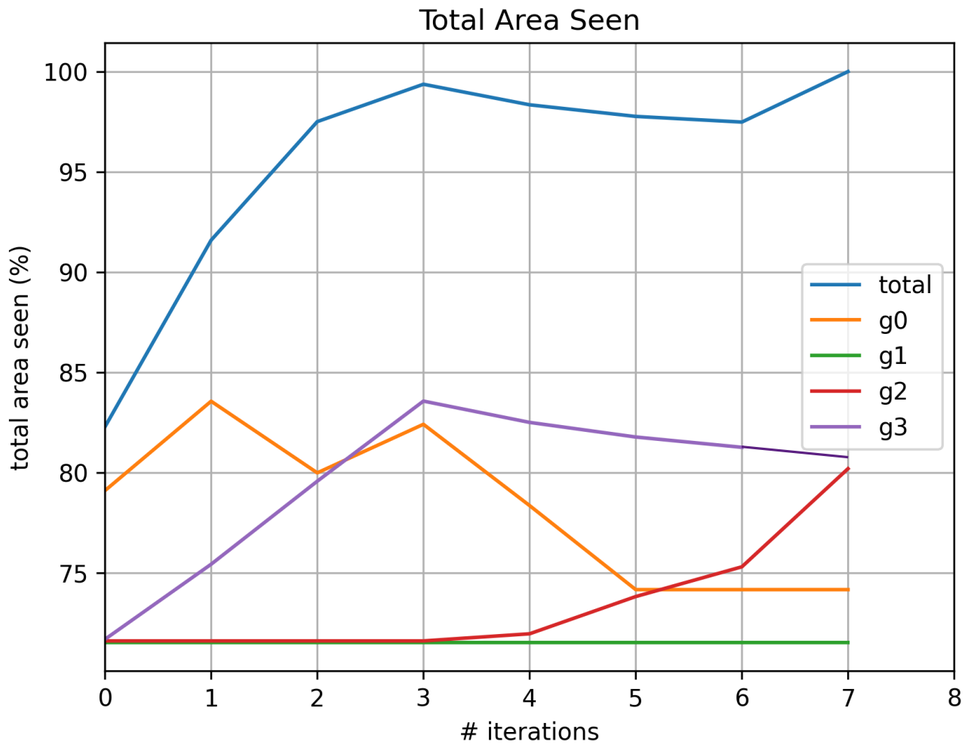


Heuristics: Hidden movement



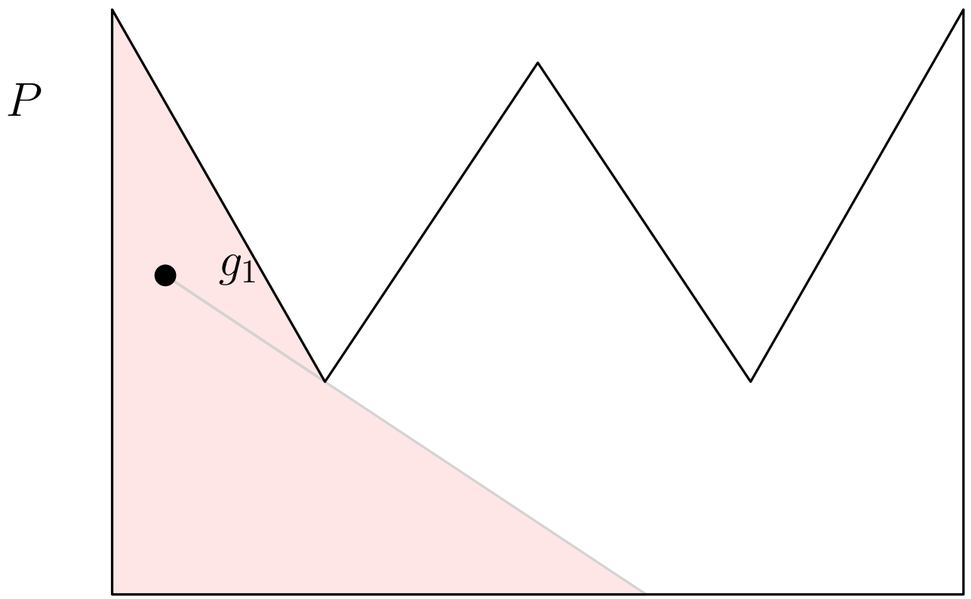


All heuristics

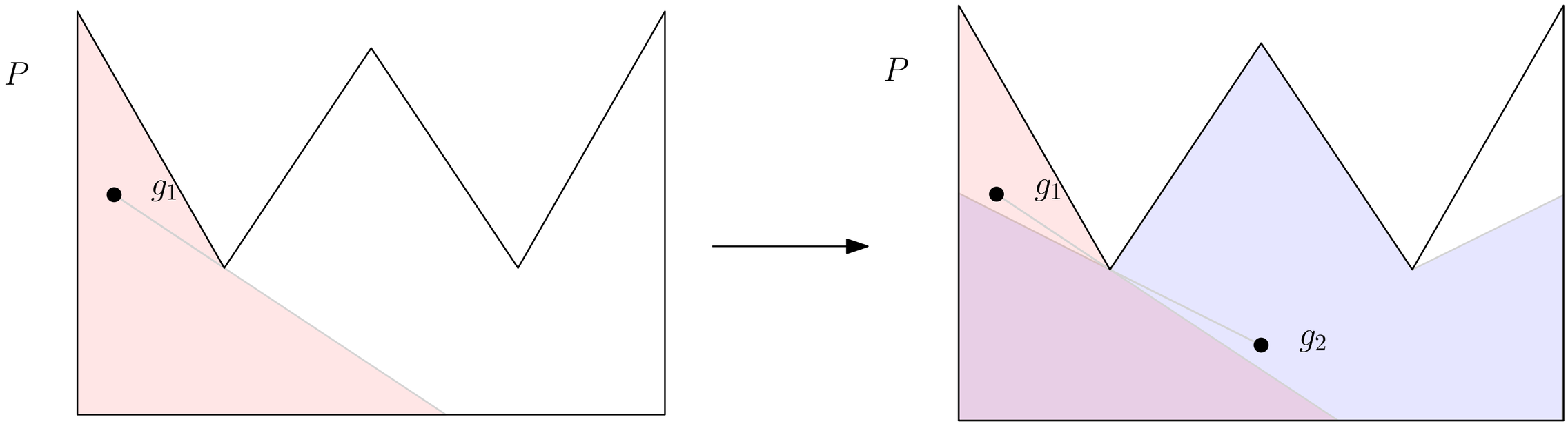


No hidden movement

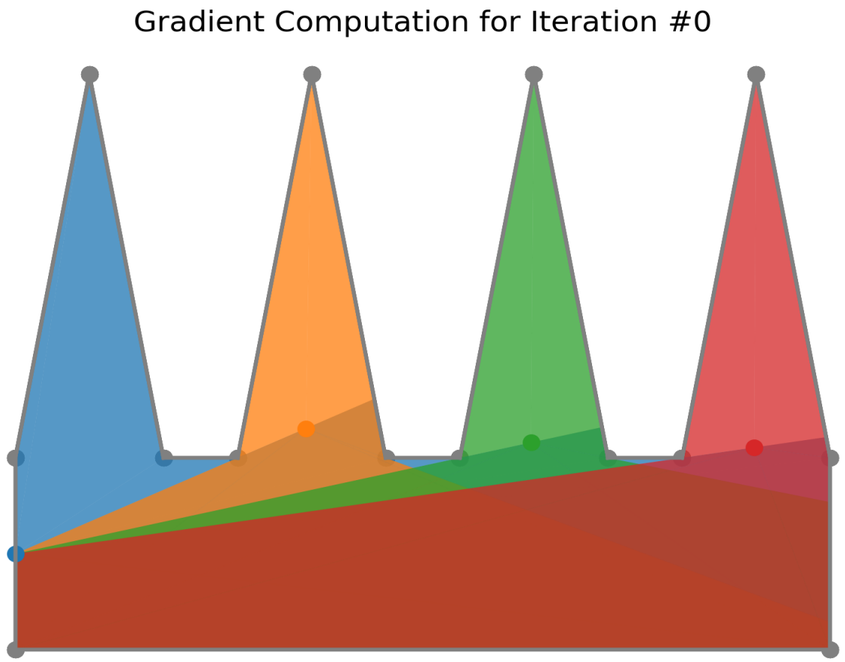
Heuristics: Greedy initialisation



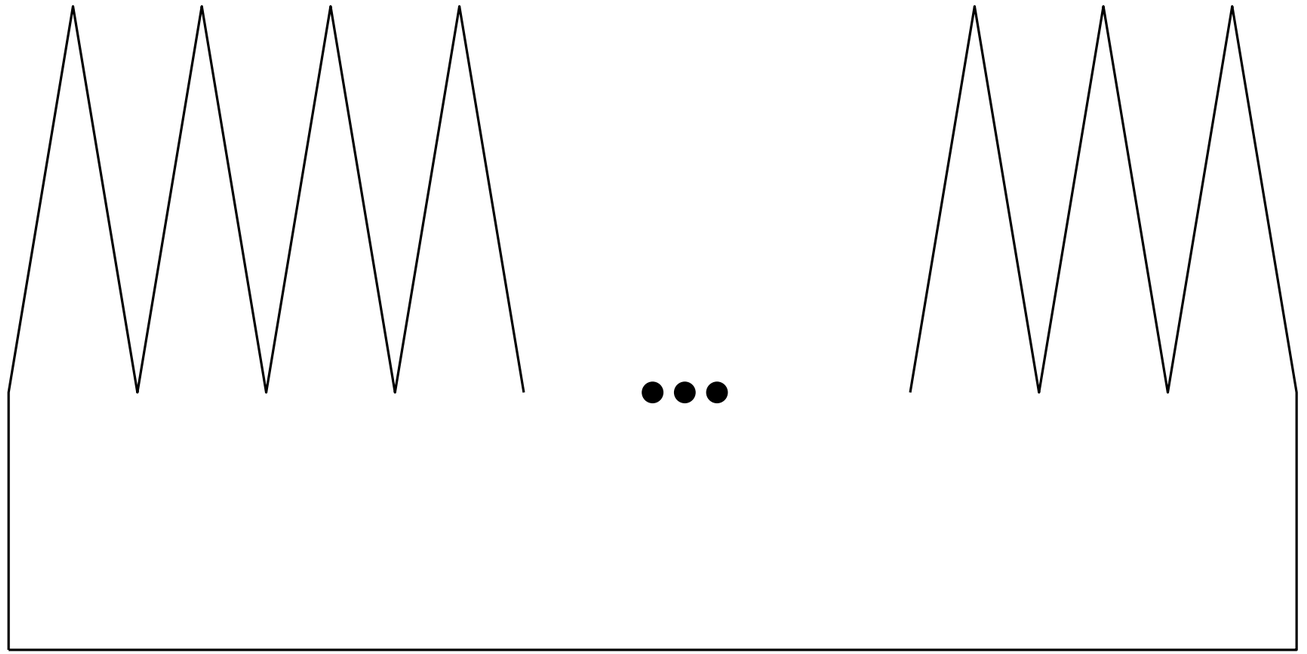
Heuristics: Greedy initialisation



Heuristics: Greedy initialisation

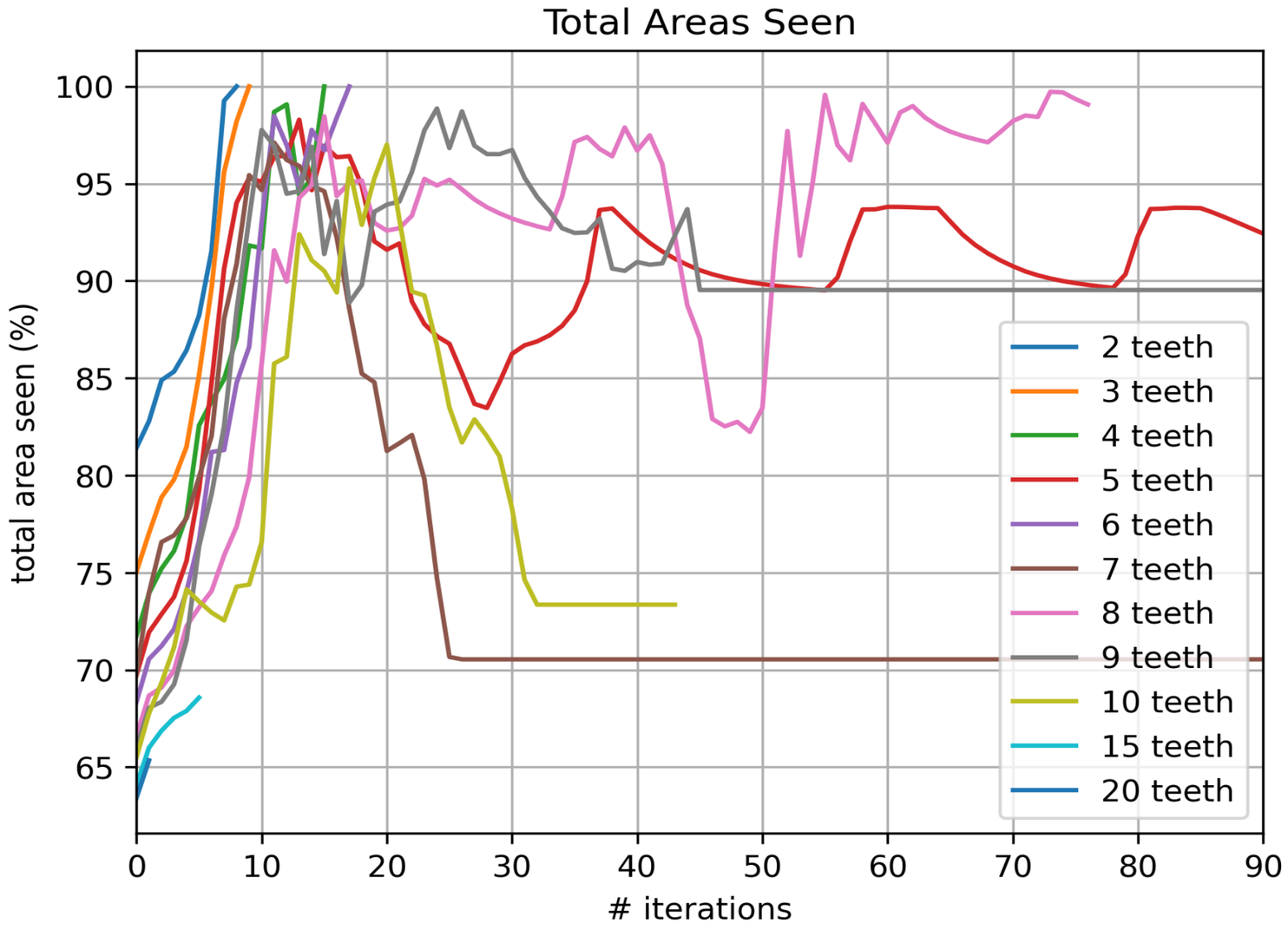


Scalability for the comb polygon

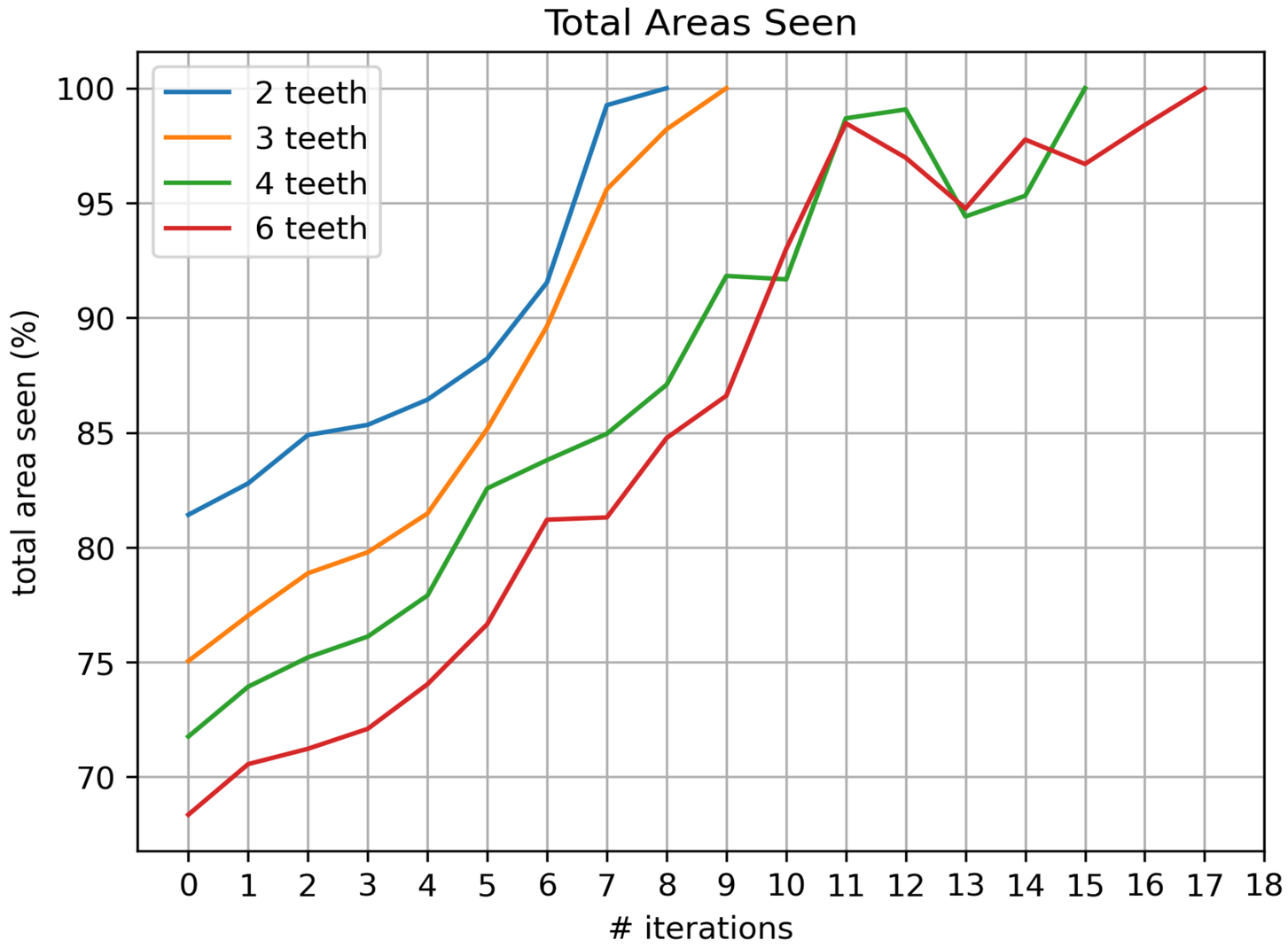


2, 3, ..., 10, 15, 20 teeth

Scalability for the comb polygon



Scalability for the comb polygon



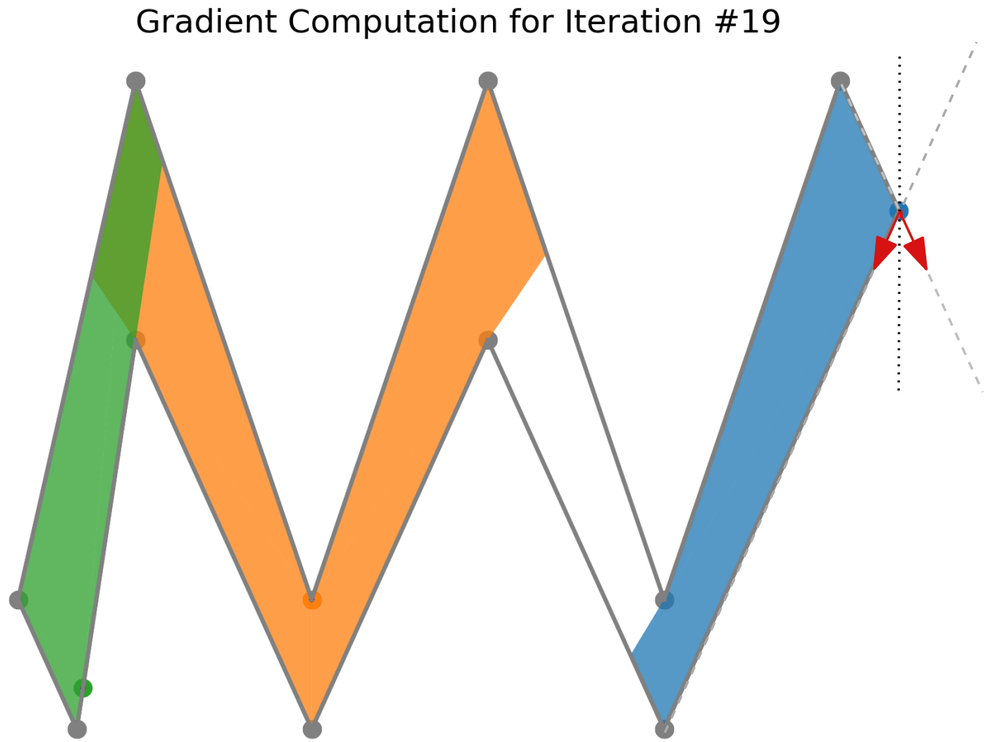
Problems encountered

hyperparameter sensitivity

Problems encountered

hyperparameter sensitivity

edge-cases



Problems encountered

hyperparameter sensitivity

edge-cases

CGAL errors

```
terminate called after throwing an instance of 'CGAL::Assertion_exception'
  what():  CGAL ERROR: assertion violation!
Expr: is_finite(d)
File: /usr/include/CGAL/Interval_nt.h
Line: 133
fish: “./../build/main < inputs/love.i…” terminated by signal SIGABRT (Abort)
```

Future work

solve existing bugs

Future work

solve existing bugs

improve the algorithm's robustness, performance and scalability

Future work

solve existing bugs

improve the algorithm's robustness, performance and scalability

implement other heuristics

Future work

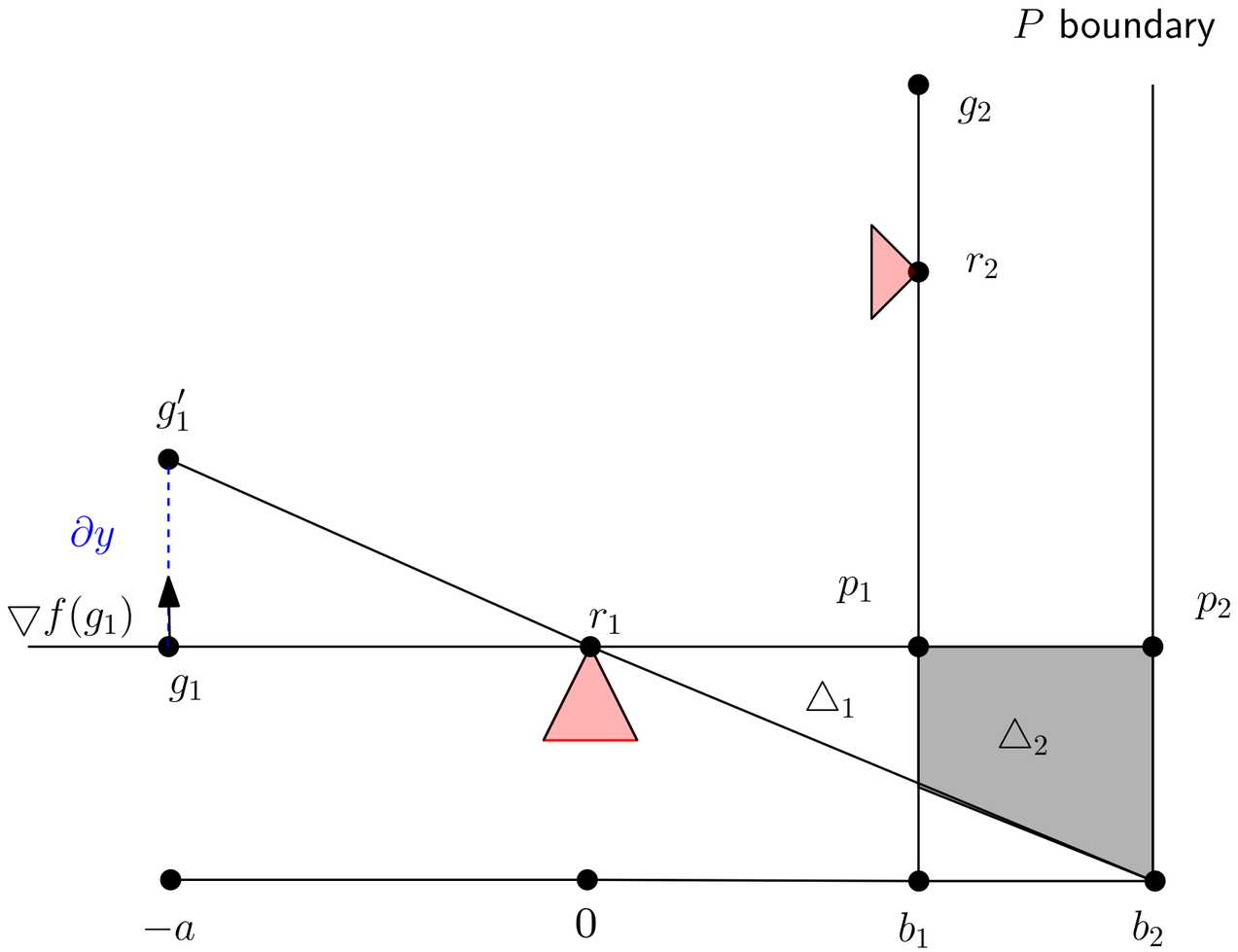
solve existing bugs

improve the algorithm's robustness, performance and scalability

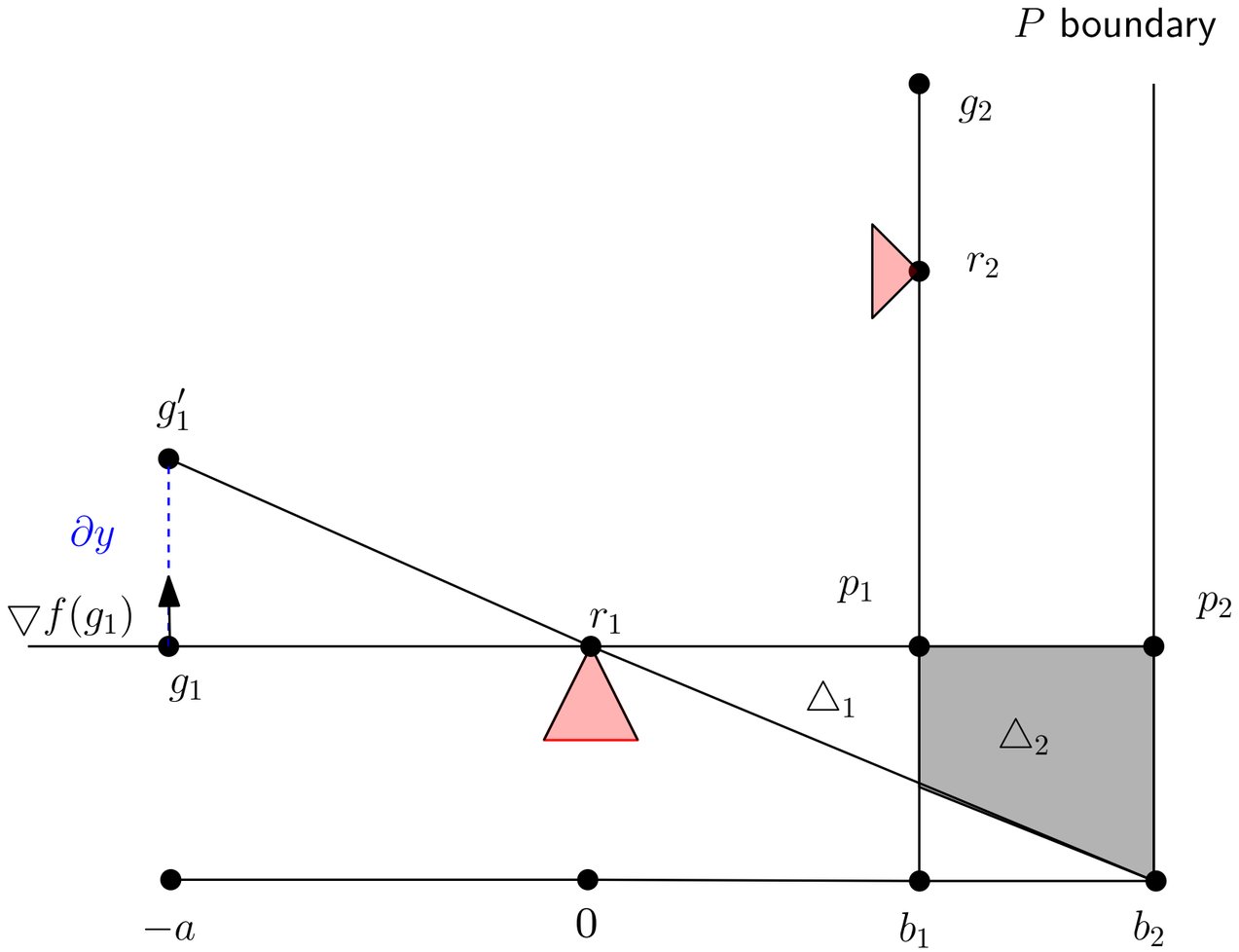
implement other heuristics

test the algorithm on larger polygons with more guards

Computing the gradient for multiple guards

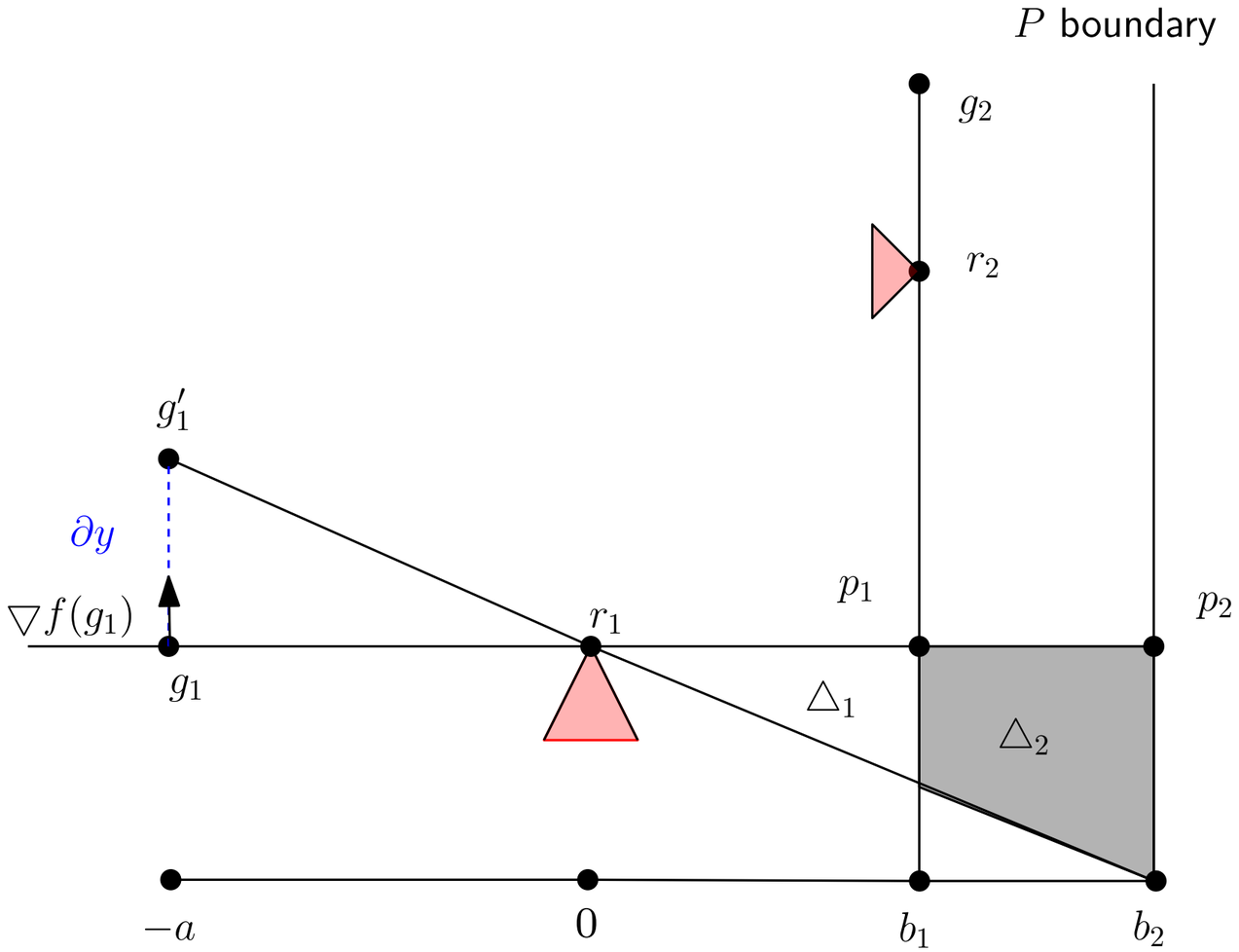


Computing the gradient for multiple guards



$$\text{Area}_{\triangle_1+\triangle_2}(g_1) = (b_1 + b_2)^2 \frac{\partial y}{2a}$$

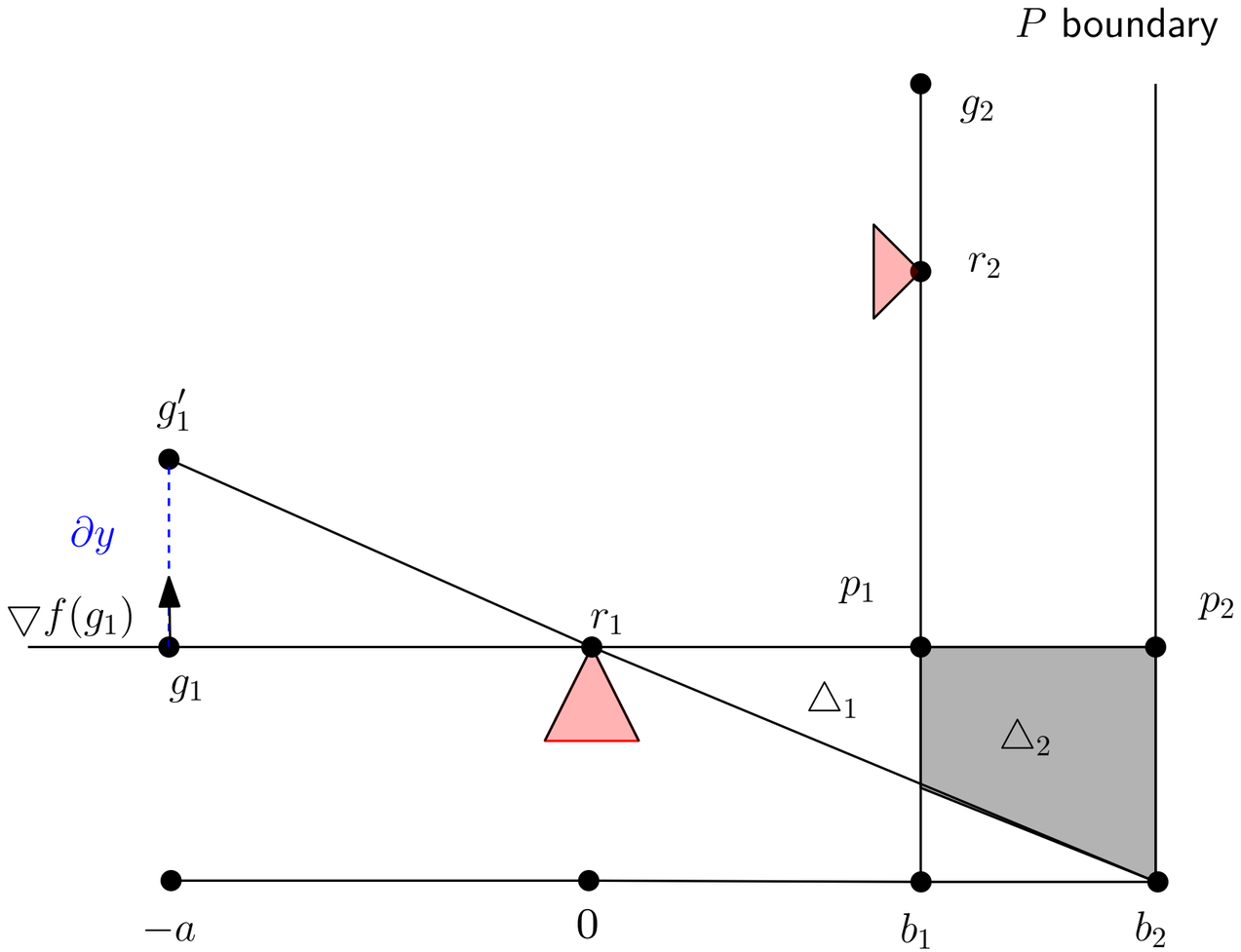
Computing the gradient for multiple guards



$$\text{Area}_{\triangle_1+\triangle_2}(g_1) = (b_1 + b_2)^2 \frac{\partial y}{2a}$$

$$\text{Area}_{\triangle_1}(g_1) = b_1^2 \frac{\partial y}{2a}$$

Computing the gradient for multiple guards

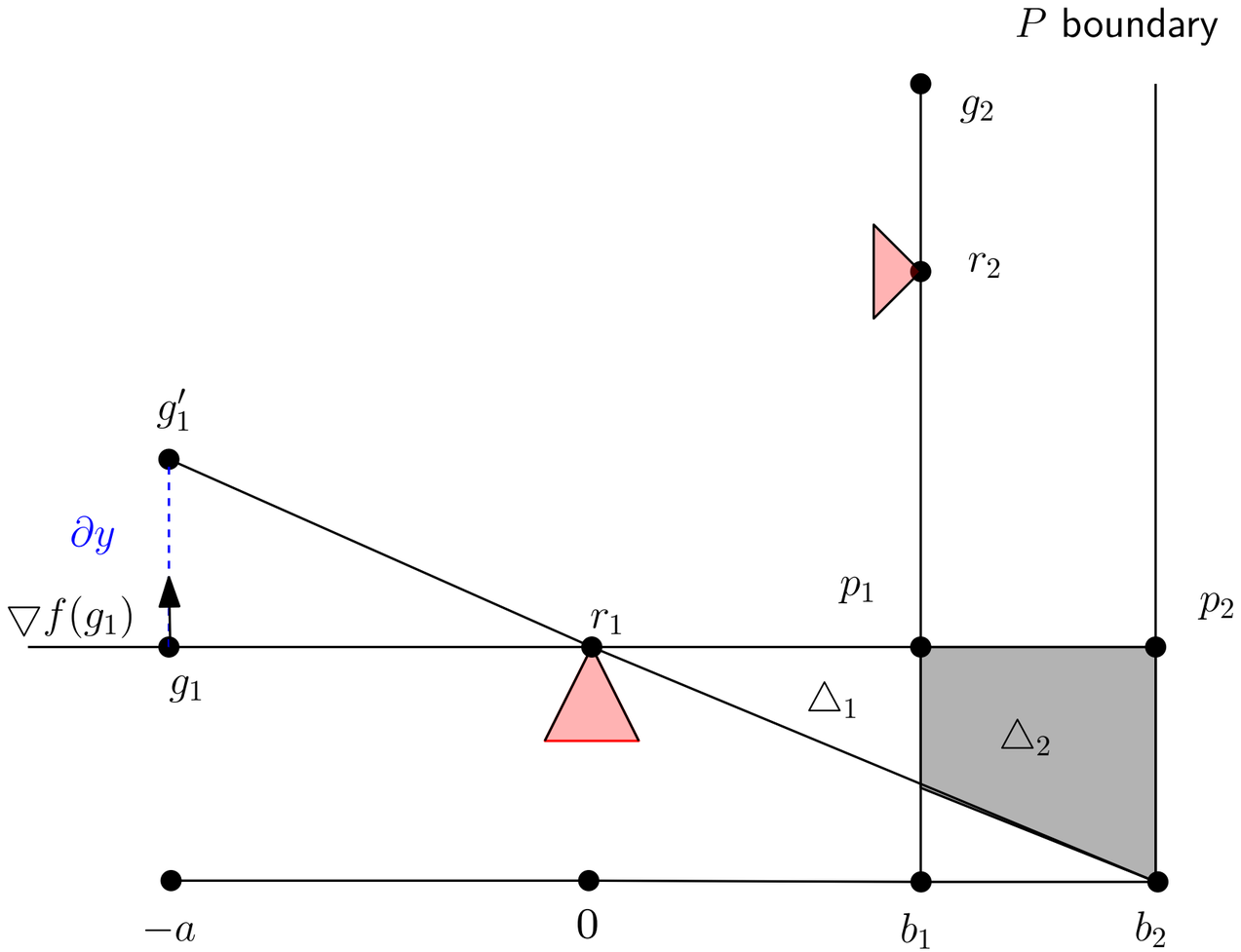


$$\text{Area}_{\triangle_1+\triangle_2}(g_1) = (b_1 + b_2)^2 \frac{\partial y}{2a}$$

$$\text{Area}_{\triangle_1}(g_1) = b_1^2 \frac{\partial y}{2a}$$

$$\text{Area}_{\triangle_2}(g_1) = \text{Area}_{\triangle_1+\triangle_2}(g_1) - \text{Area}_{\triangle_1}(g_1)$$

Computing the gradient for multiple guards

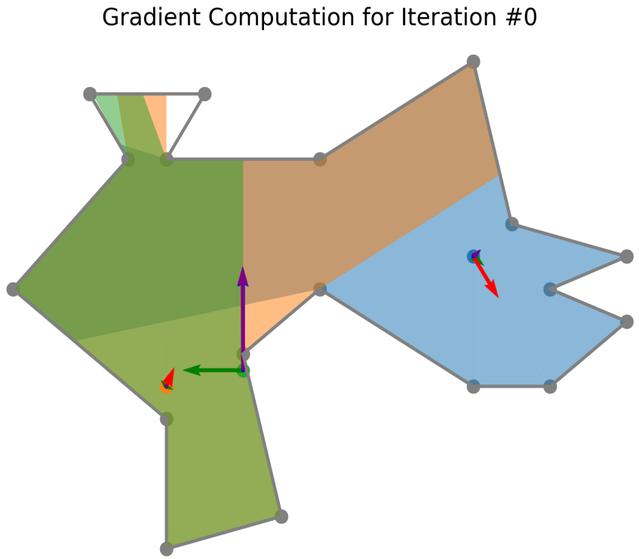


$$\text{Area}_{\triangle_1+\triangle_2}(g_1) = (b_1 + b_2)^2 \frac{\partial y}{2a}$$

$$\text{Area}_{\triangle_1}(g_1) = b_1^2 \frac{\partial y}{2a}$$

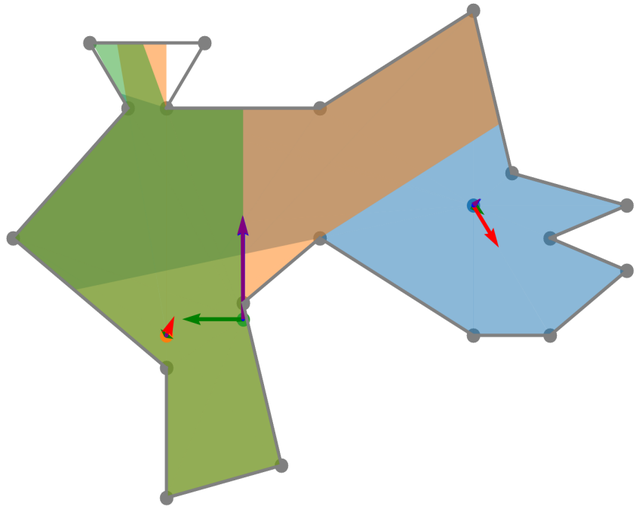
$$\begin{aligned}\text{Area}_{\triangle_2}(g_1) &= \text{Area}_{\triangle_1+\triangle_2}(g_1) - \text{Area}_{\triangle_1}(g_1) \\ &= \left[(b_1 + b_2)^2 - b_1^2\right] \frac{\partial y}{2a}\end{aligned}$$

The Art Gallery Problem

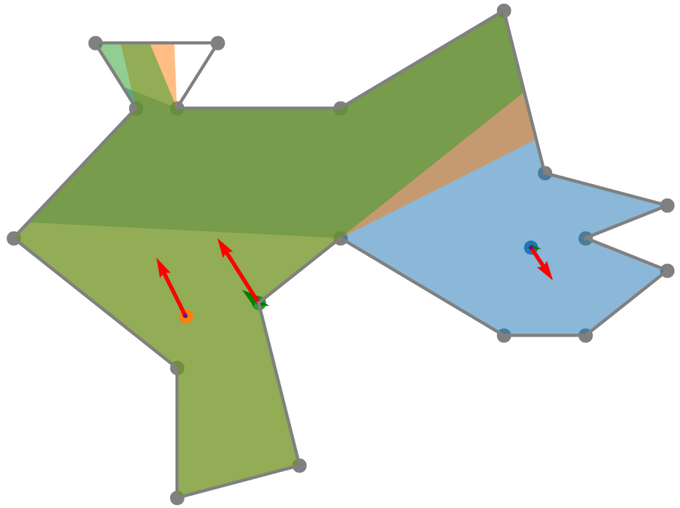


The Art Gallery Problem

Gradient Computation for Iteration #0

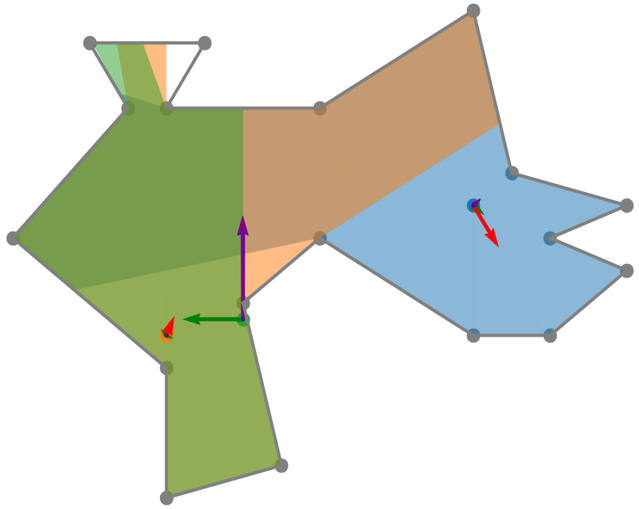


Gradient Computation for Iteration #1

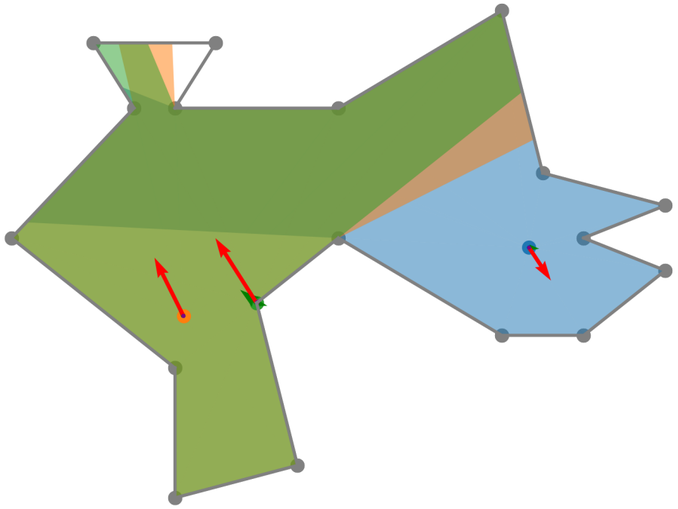


The Art Gallery Problem

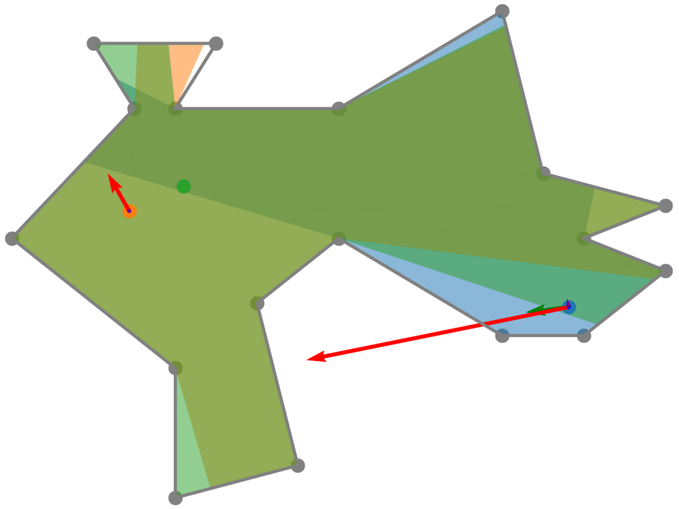
Gradient Computation for Iteration #0



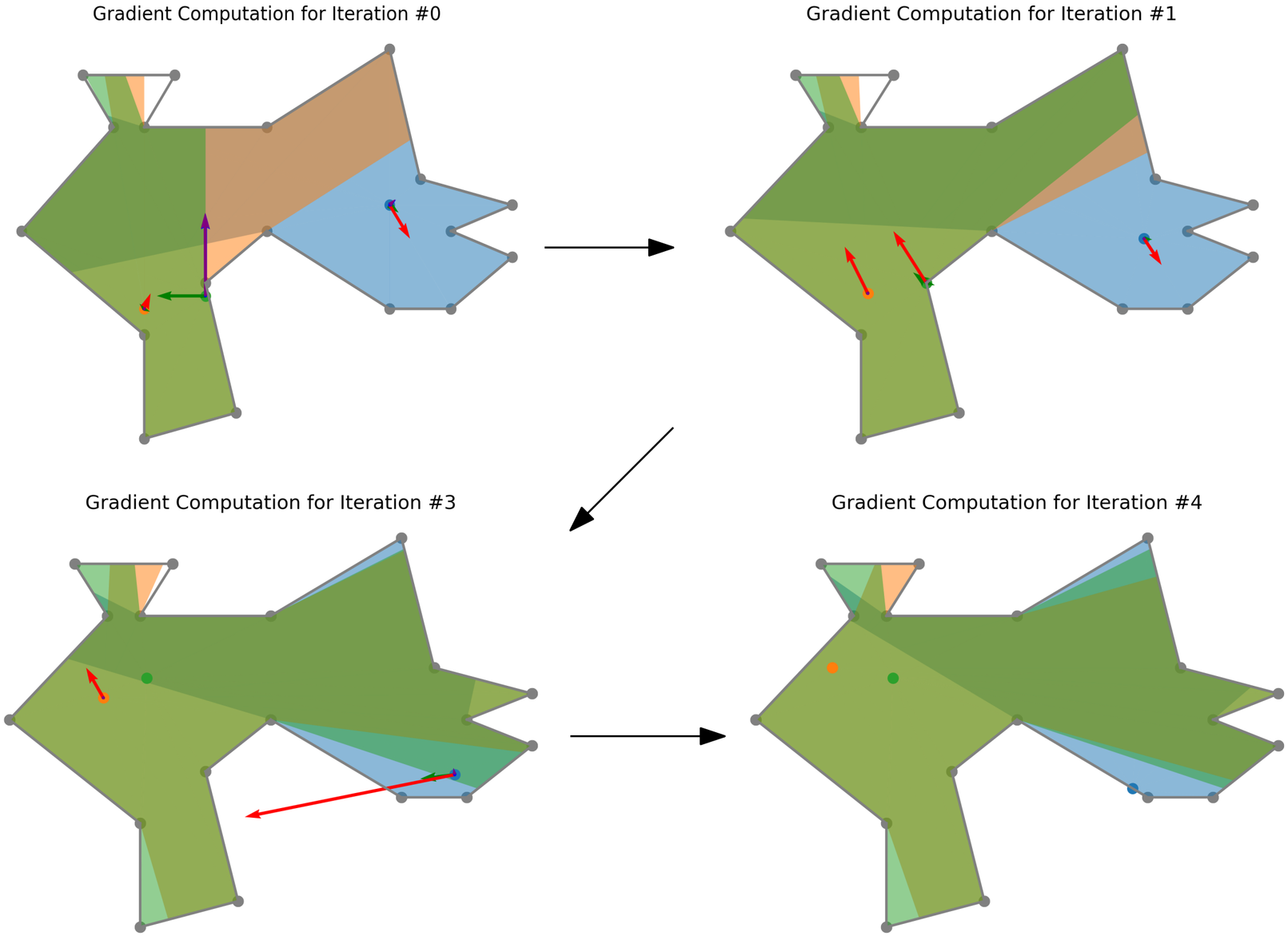
Gradient Computation for Iteration #1



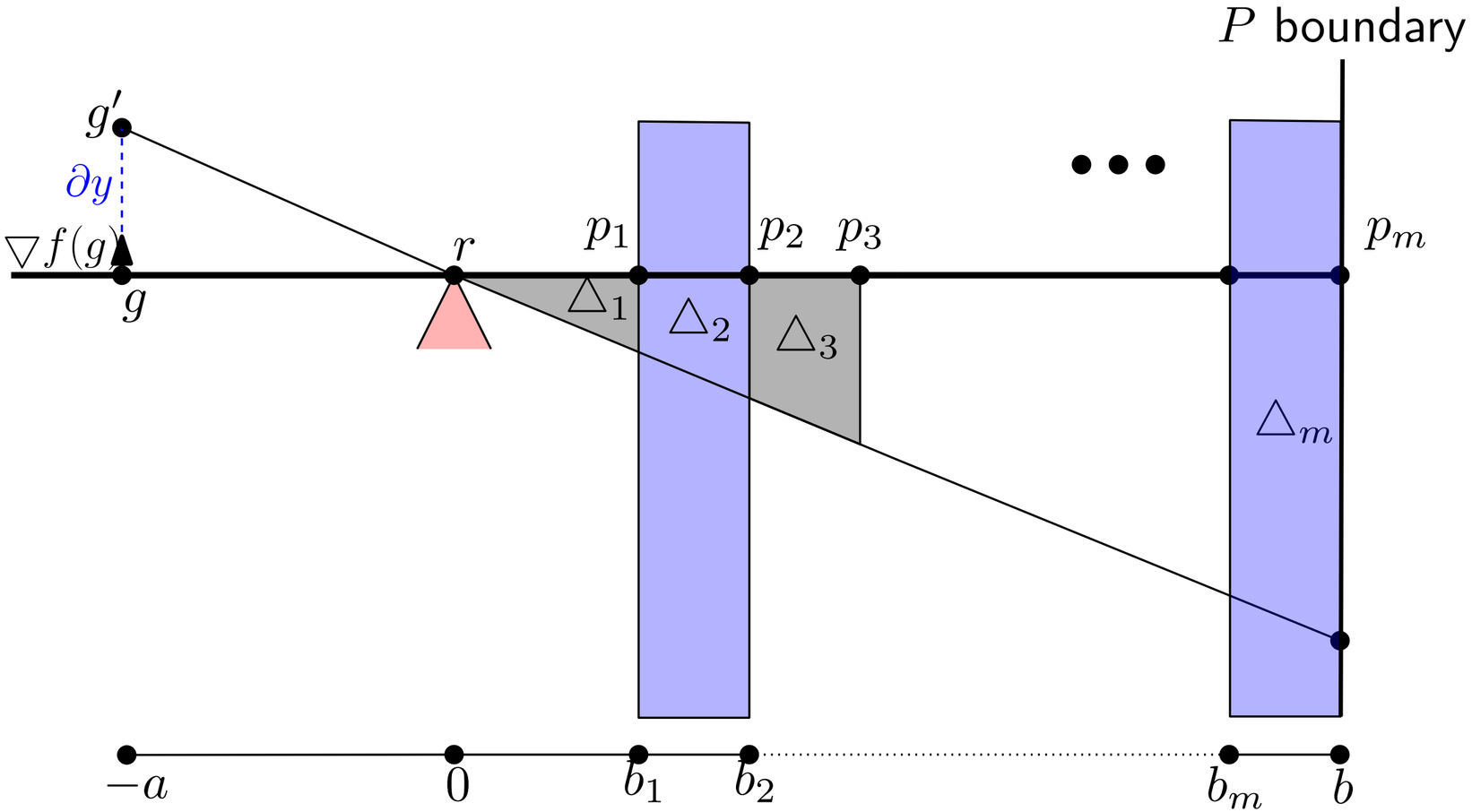
Gradient Computation for Iteration #3



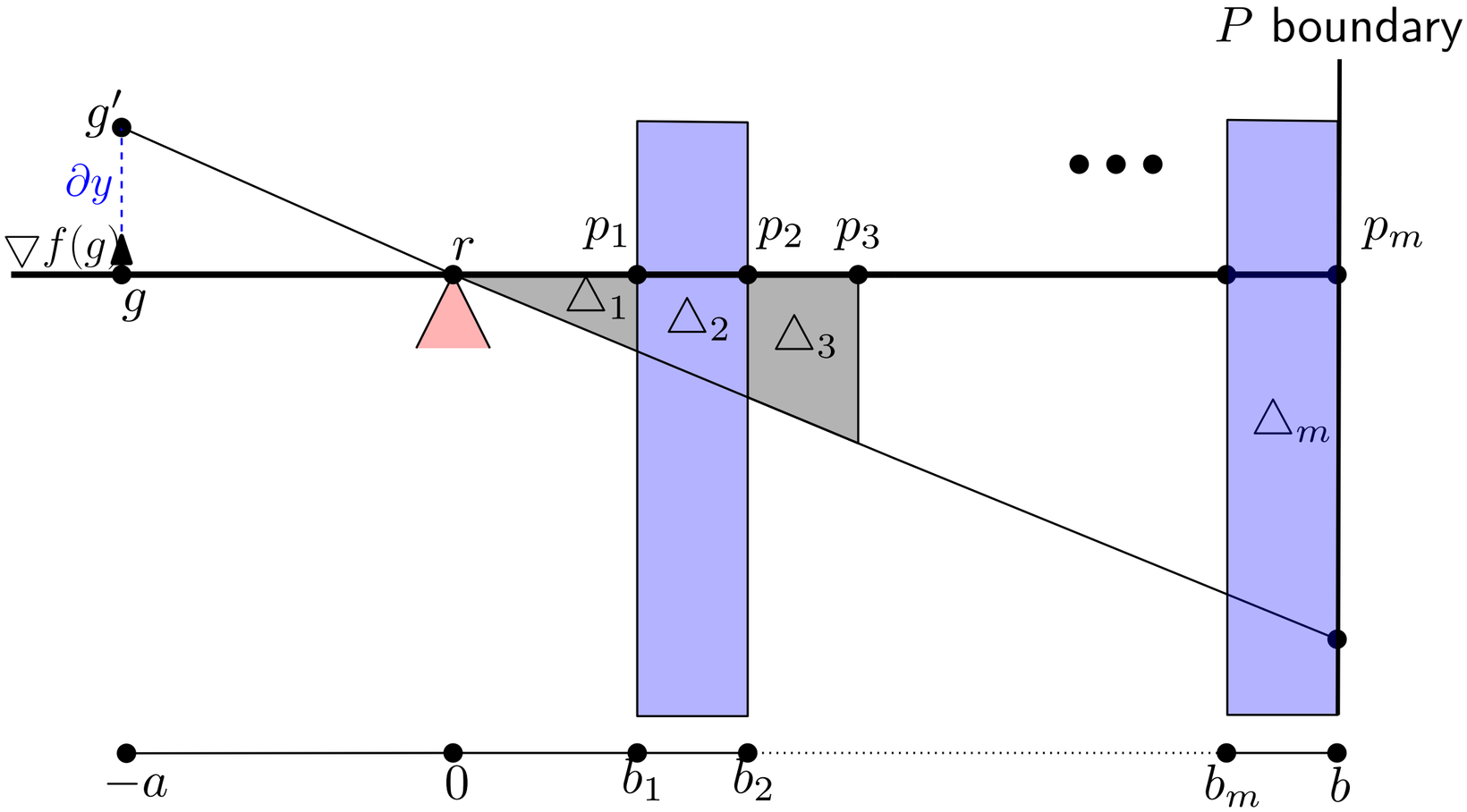
The Art Gallery Problem



Computing the gradient for multiple guards

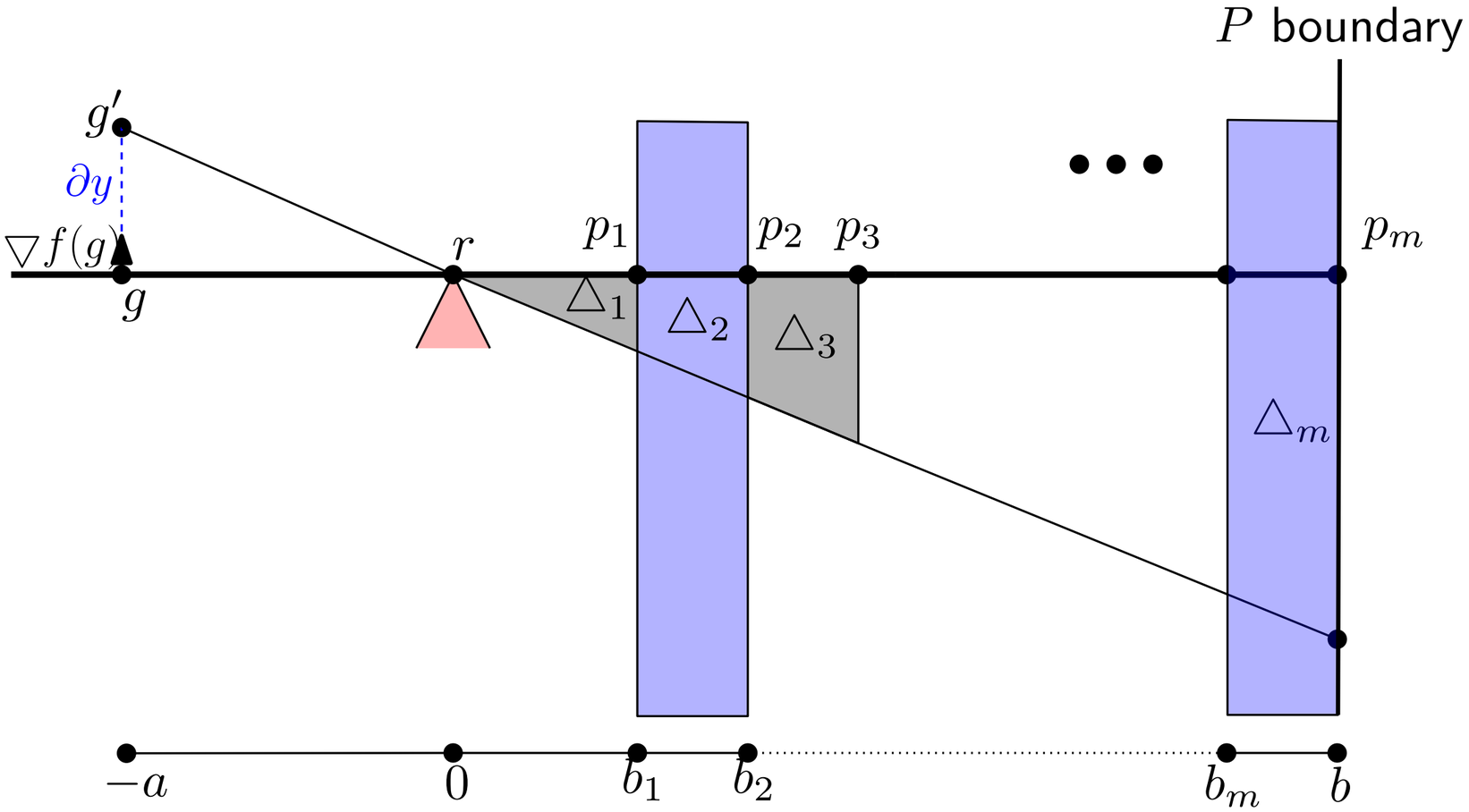


Computing the gradient for multiple guards



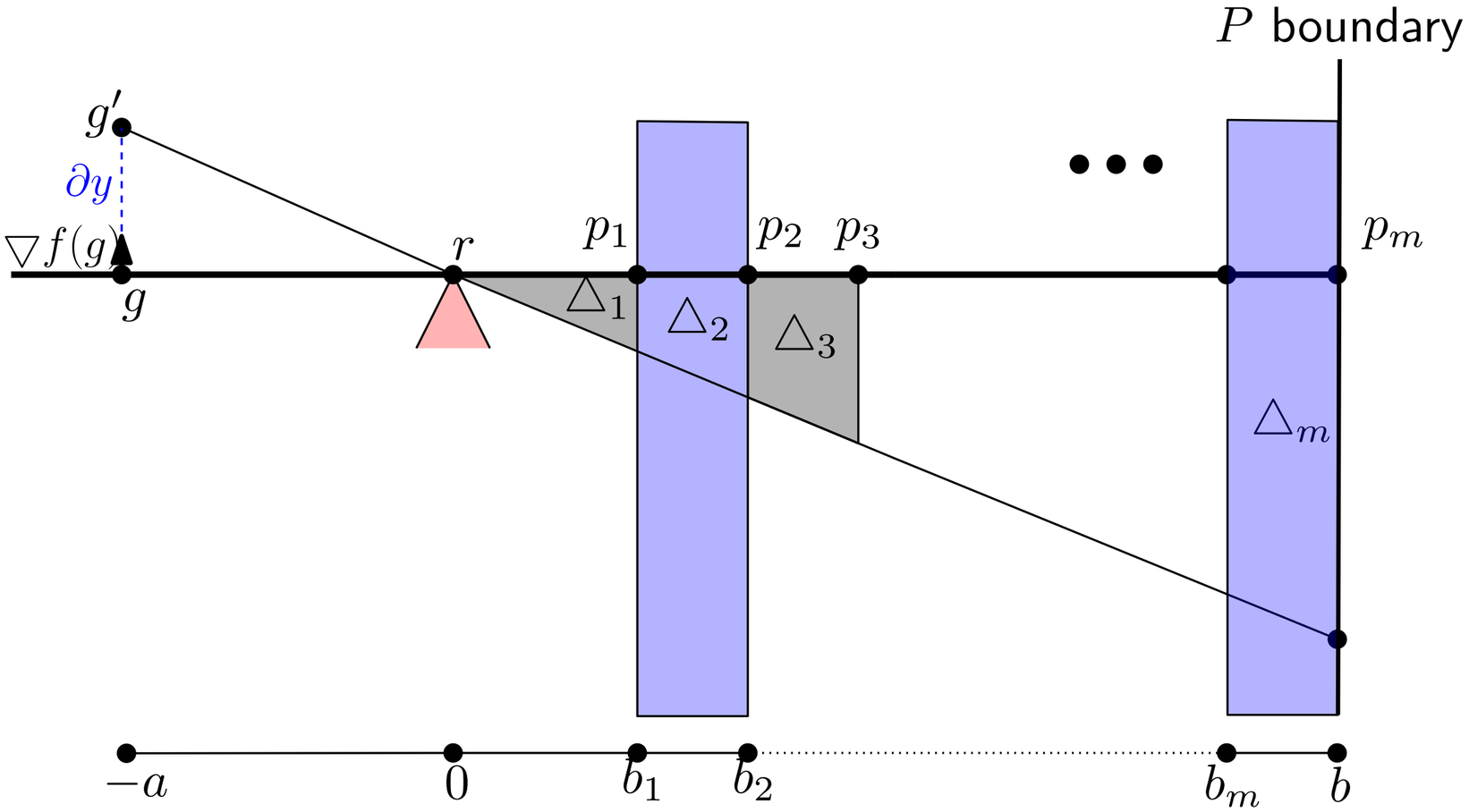
Area$_{\triangle_1+\triangle_3+\ldots+\triangle_{m-1}}(g)$

Computing the gradient for multiple guards



$$\text{Area}_{\triangle_1+\triangle_3+\ldots+\triangle_{m-1}}(g) = \text{Area}_{\triangle_1+\ldots+\triangle_m}(g) - \text{Area}_{\triangle_{m-1}}(g) + \text{Area}_{\triangle_{m-2}}(g) - \ldots - \text{Area}_{\triangle_2}(g) + \text{Area}_{\triangle_1}(g)$$

Computing the gradient for multiple guards



$$\text{Area}_{\triangle_1+\triangle_3+\ldots+\triangle_{m-1}}(g) = \text{Area}_{\triangle_1+\ldots+\triangle_m}(g) - \text{Area}_{\triangle_{m-1}}(g) + \text{Area}_{\triangle_{m-2}}(g) - \ldots - \text{Area}_{\triangle_2}(g) + \text{Area}_{\triangle_1}(g)$$

$$= \left(b^2 - b_m^2 + b_{(m-1)}^2 - \ldots - b_2^2 + b_1^2\right) \frac{\partial y}{2a}$$