Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Institut für Informatik

Optimising Active Ranges for Point Selection in Dynamic Maps

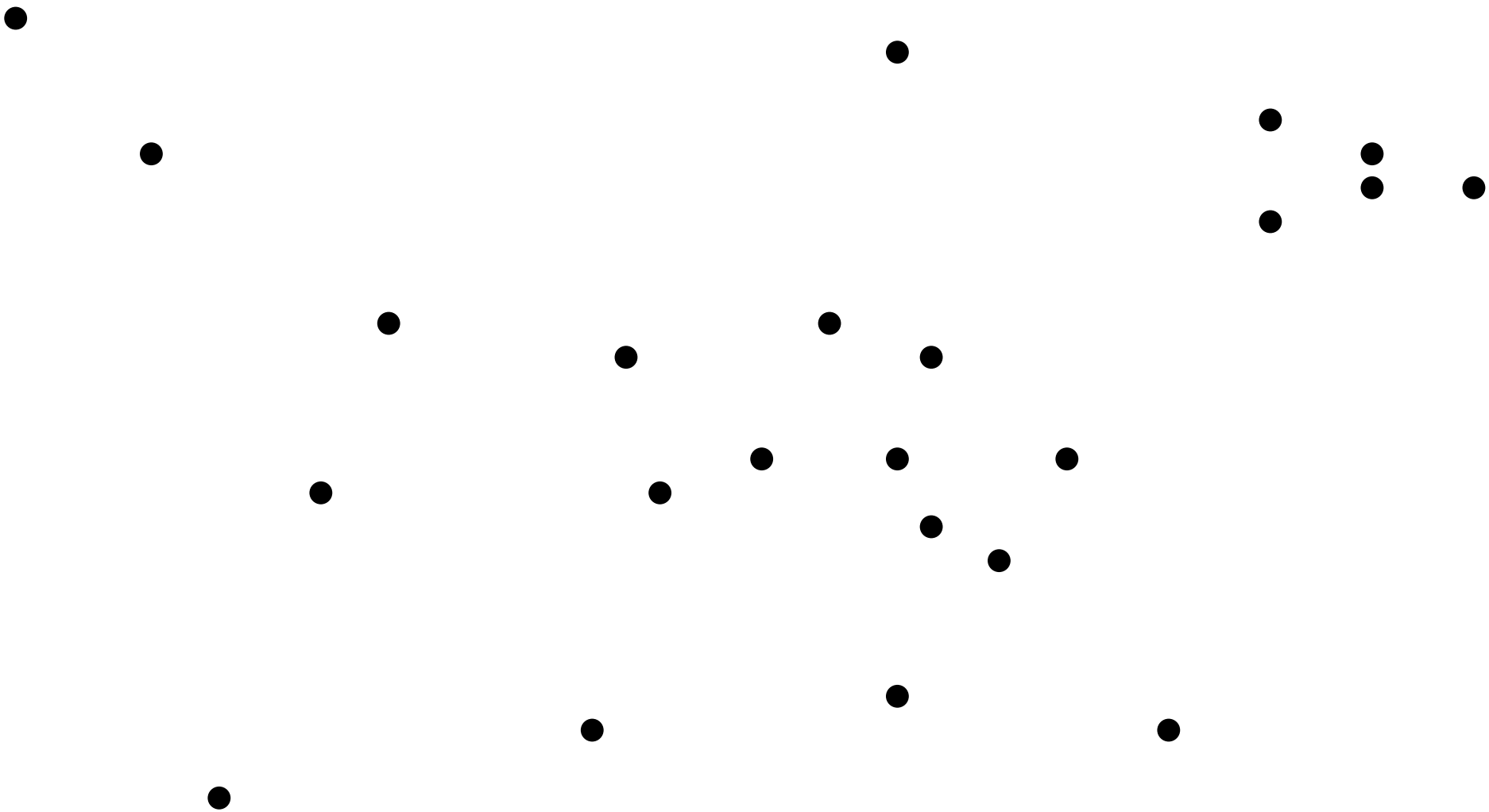
Nadine Schwartges[1] · Dennis Allerkamp[2]
Jan-Henrik Haunert[1] · Alexander Wolff[1]

ICA Generalisation Workshop '13

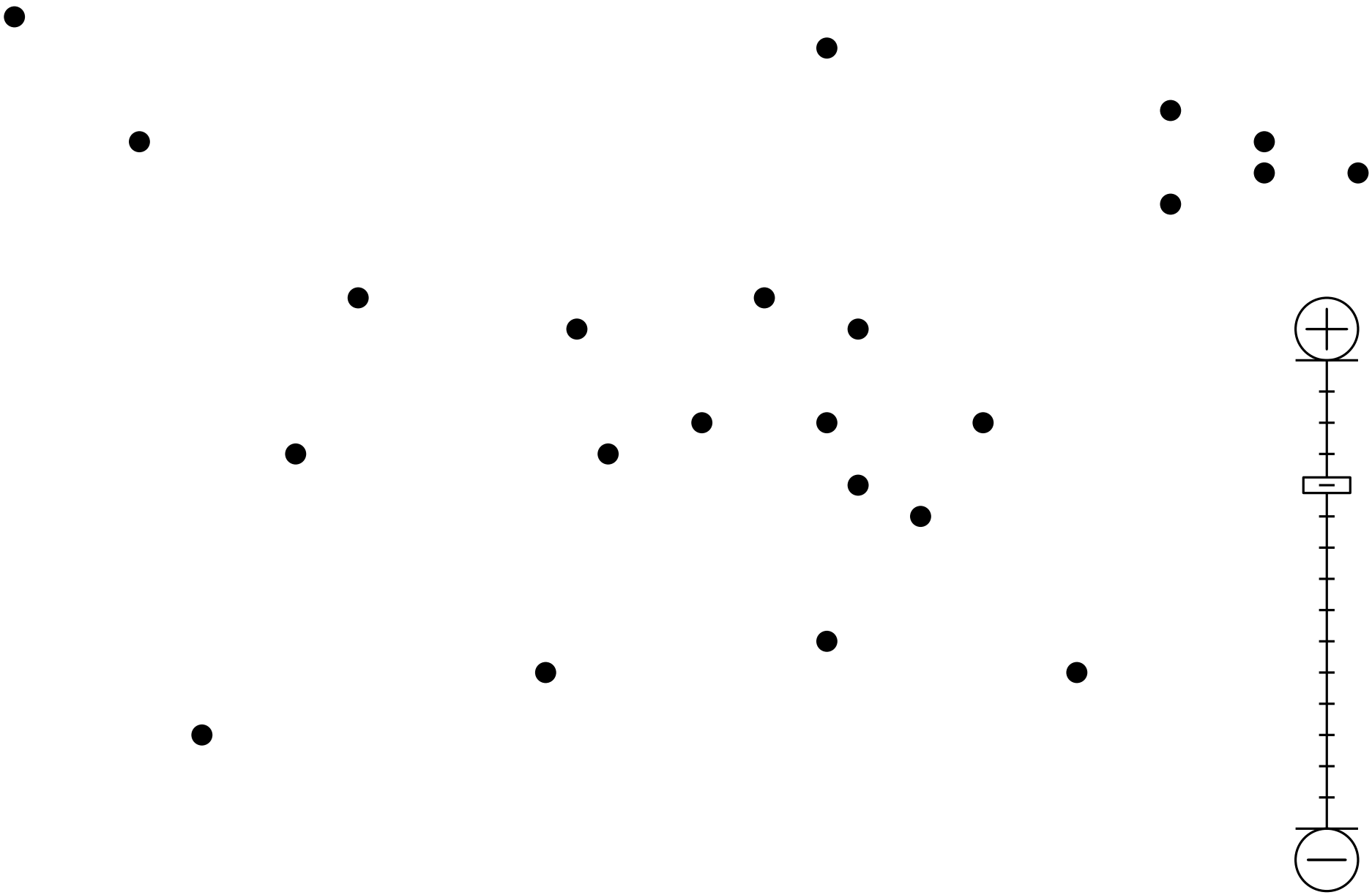
[1]Lehrstuhl für Informatik I
Universität Würzburg, Germany

[2]Robert Bosch Car Multimedia GmbH
Hildesheim, Germany

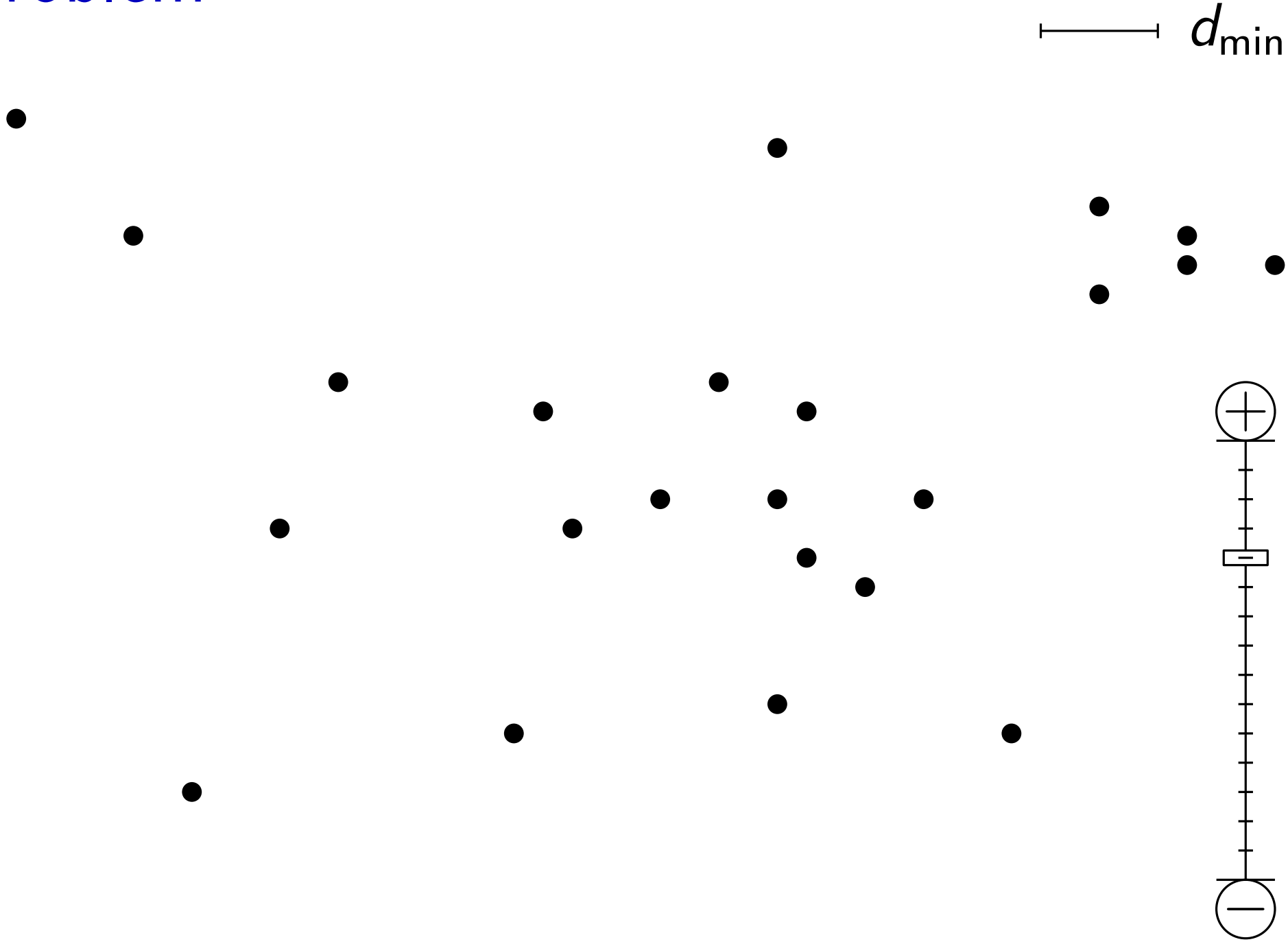
Problem



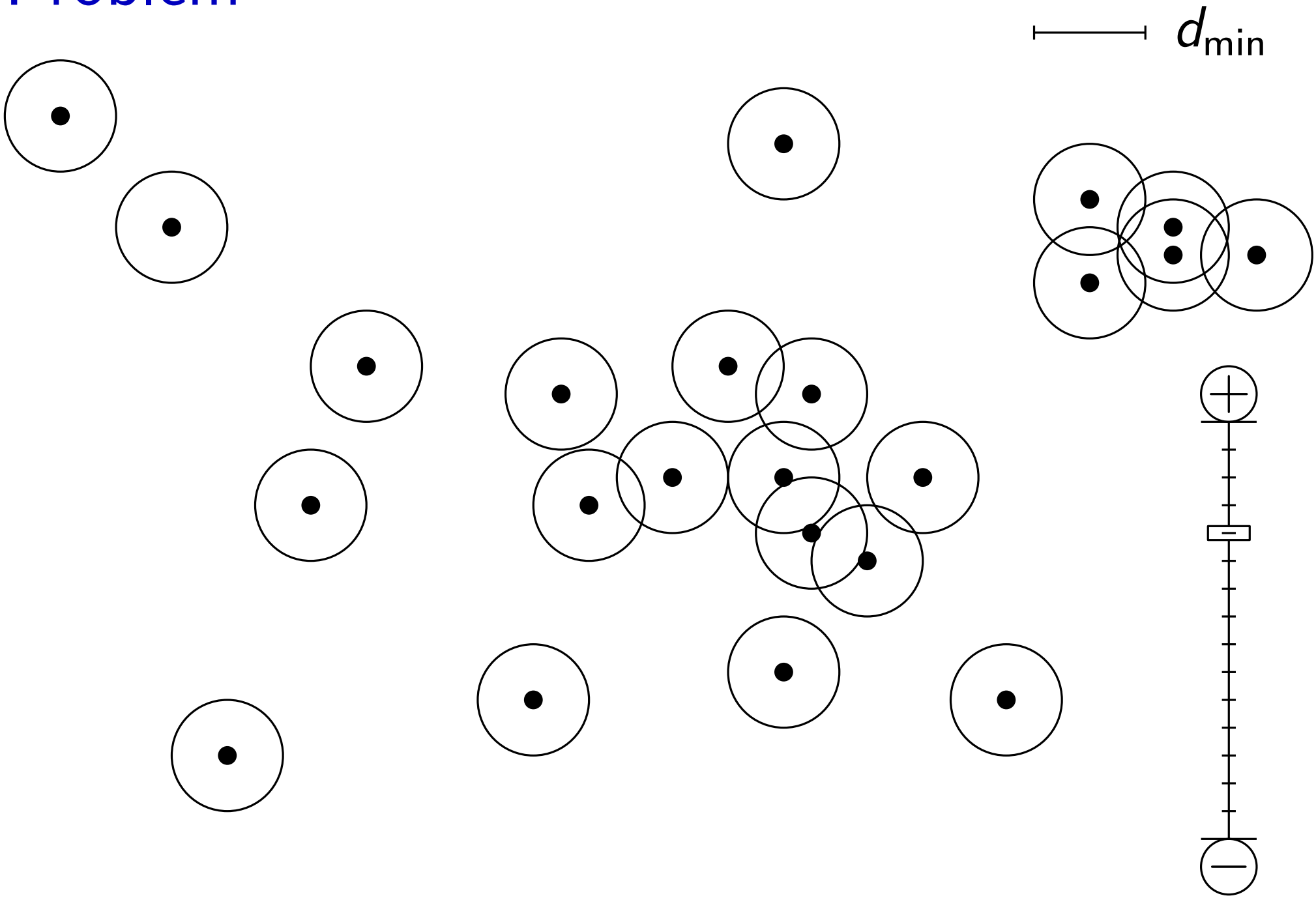
Problem



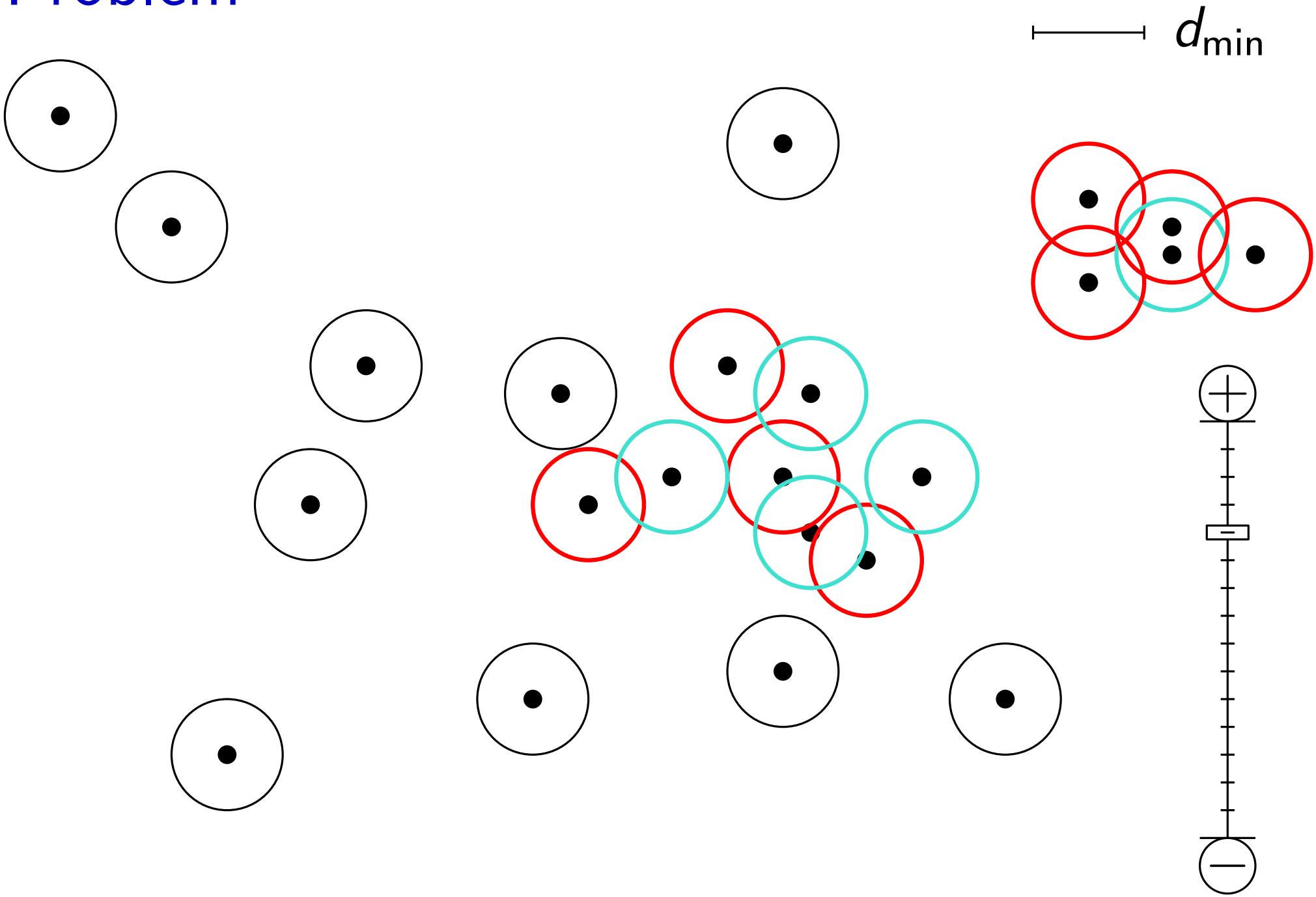
Problem



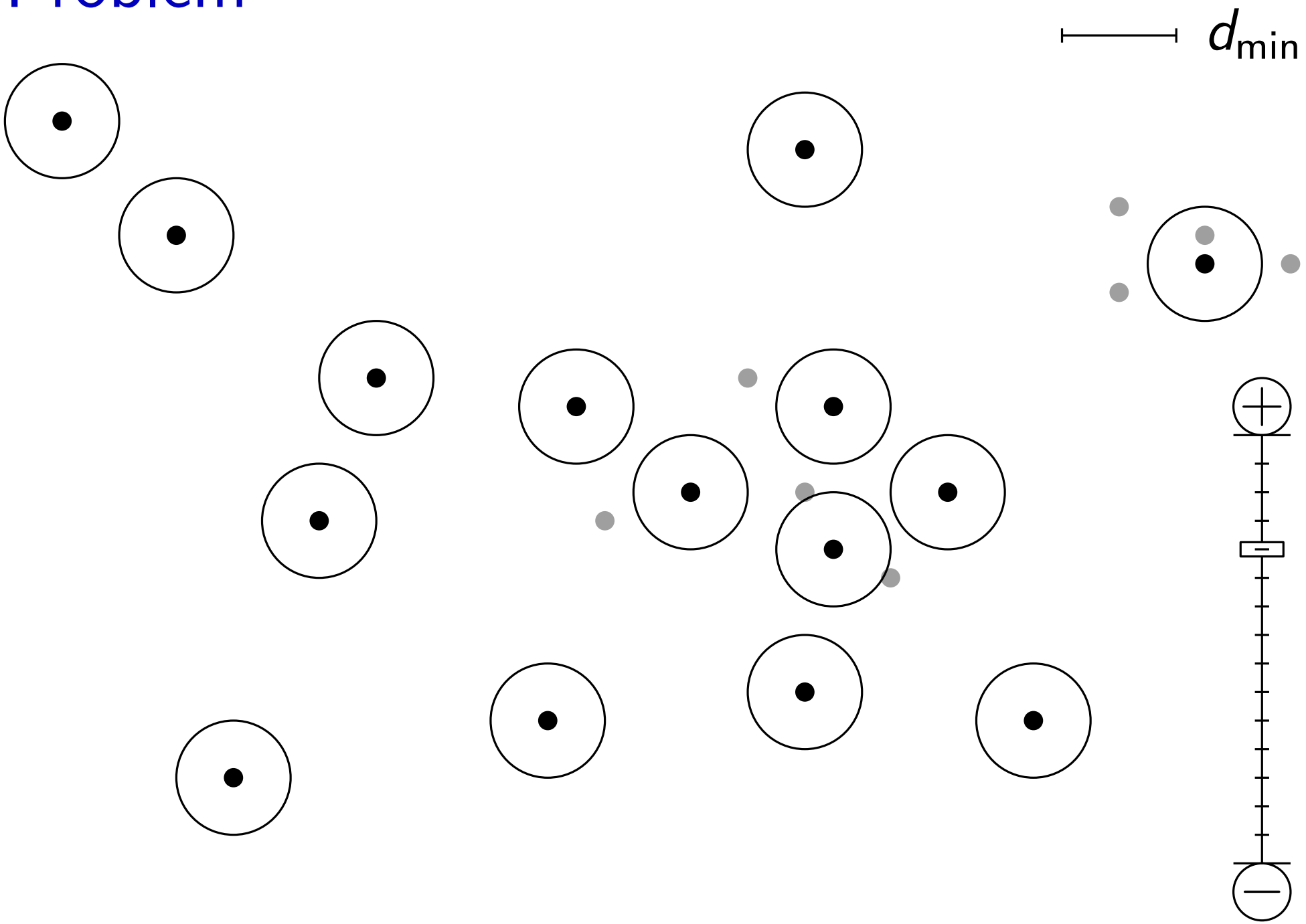
Problem



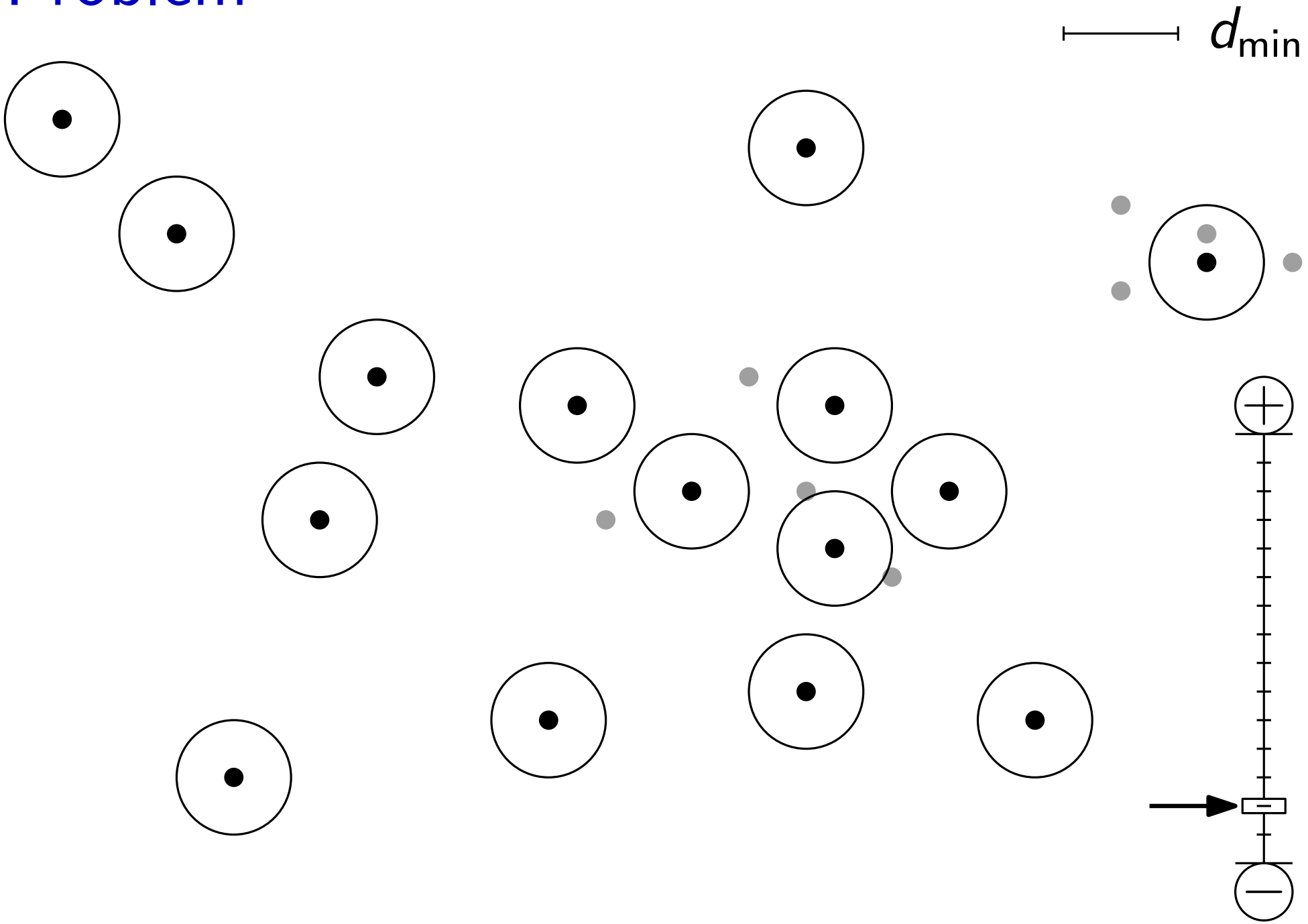
Problem



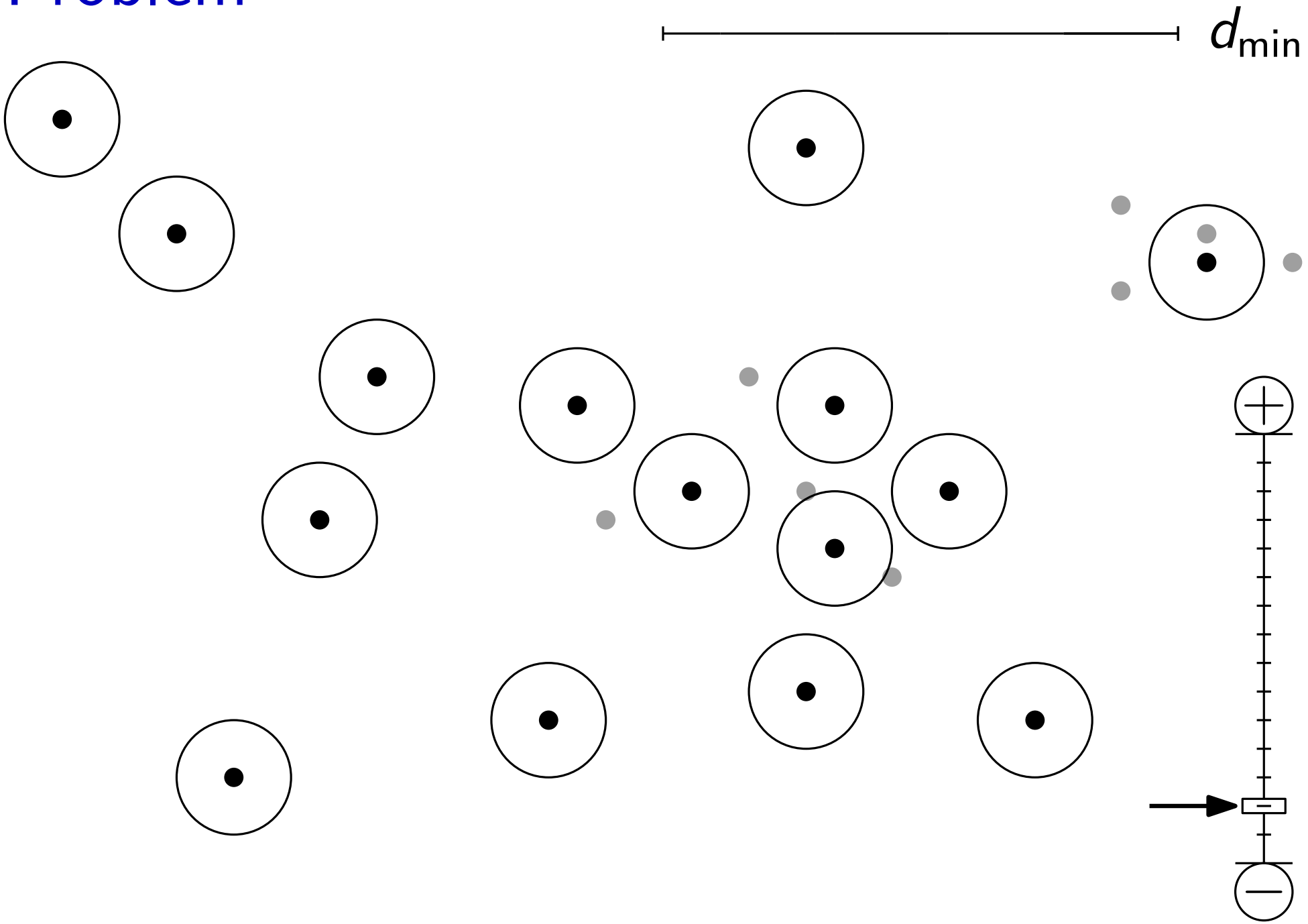
Problem



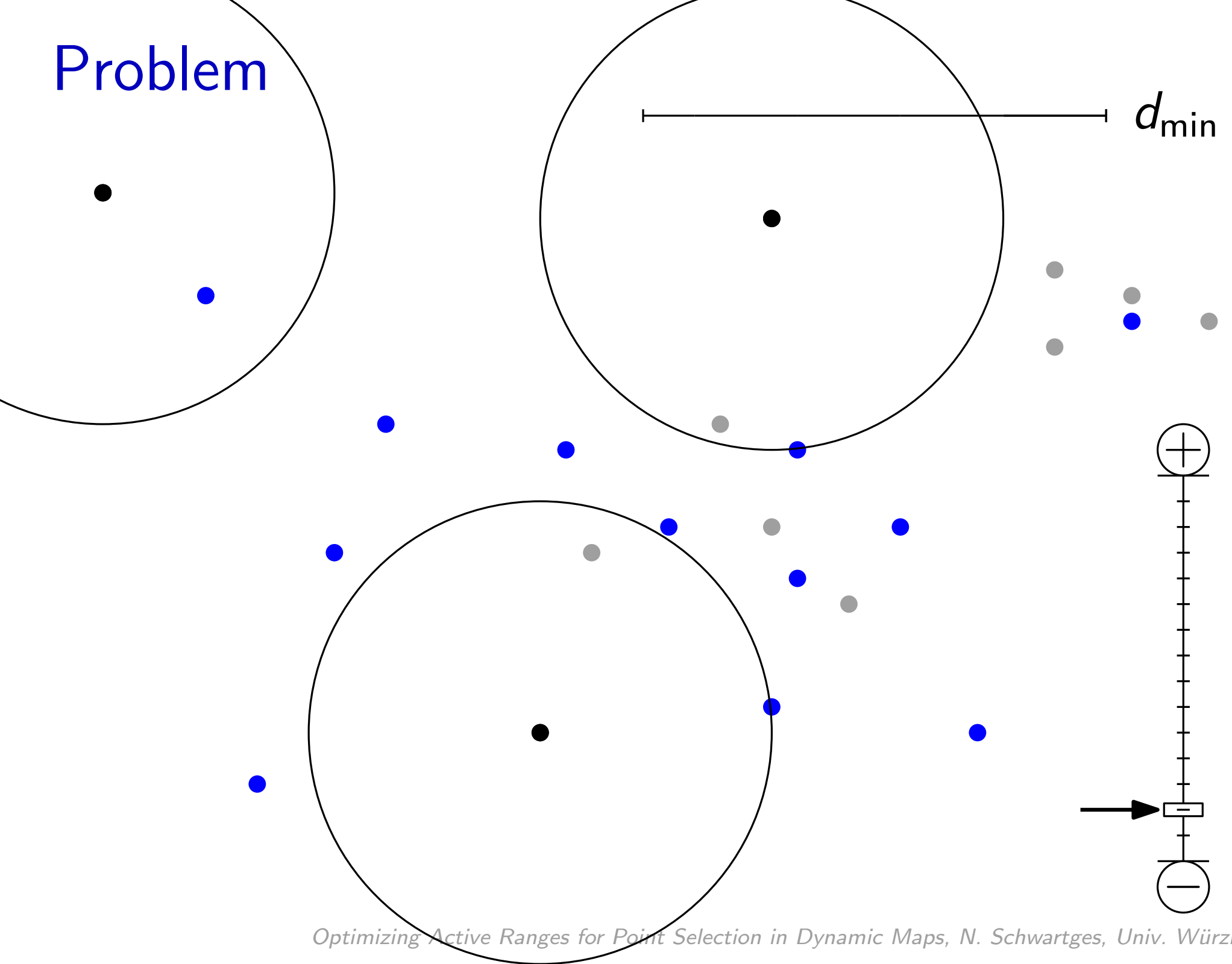
Problem



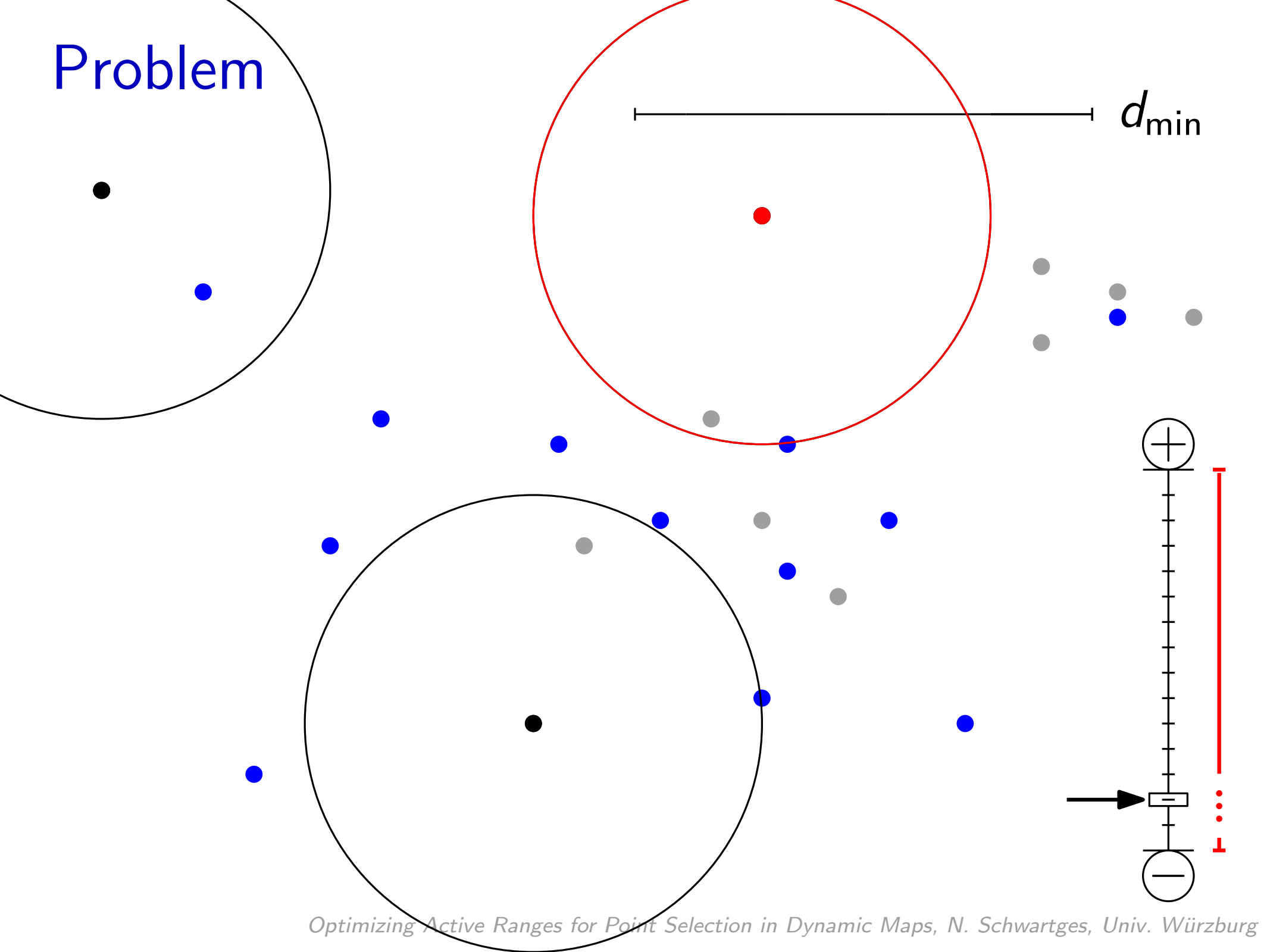
Problem



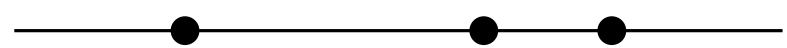
Problem



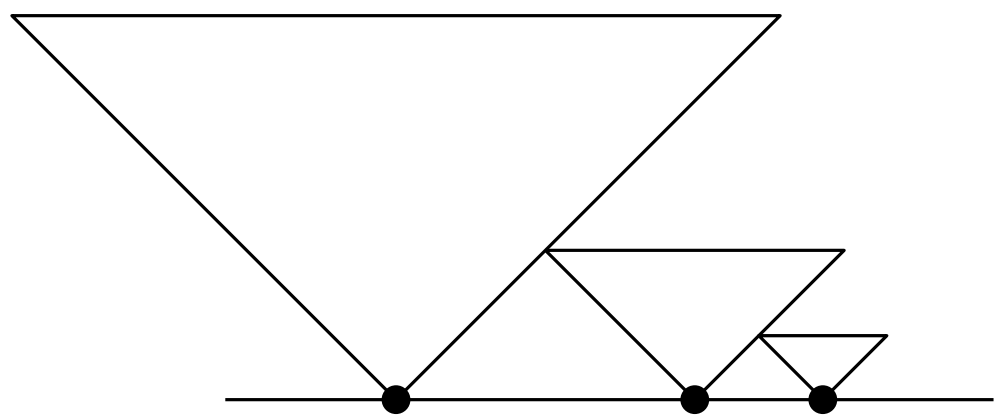
Problem



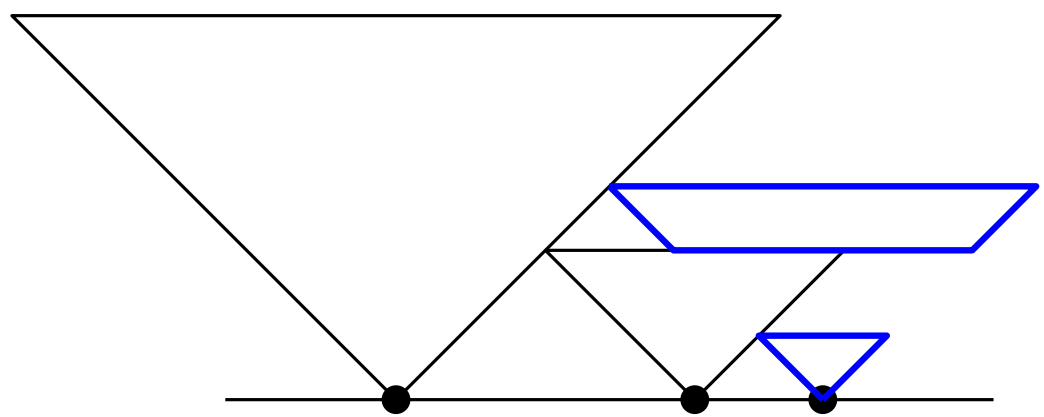
Problem



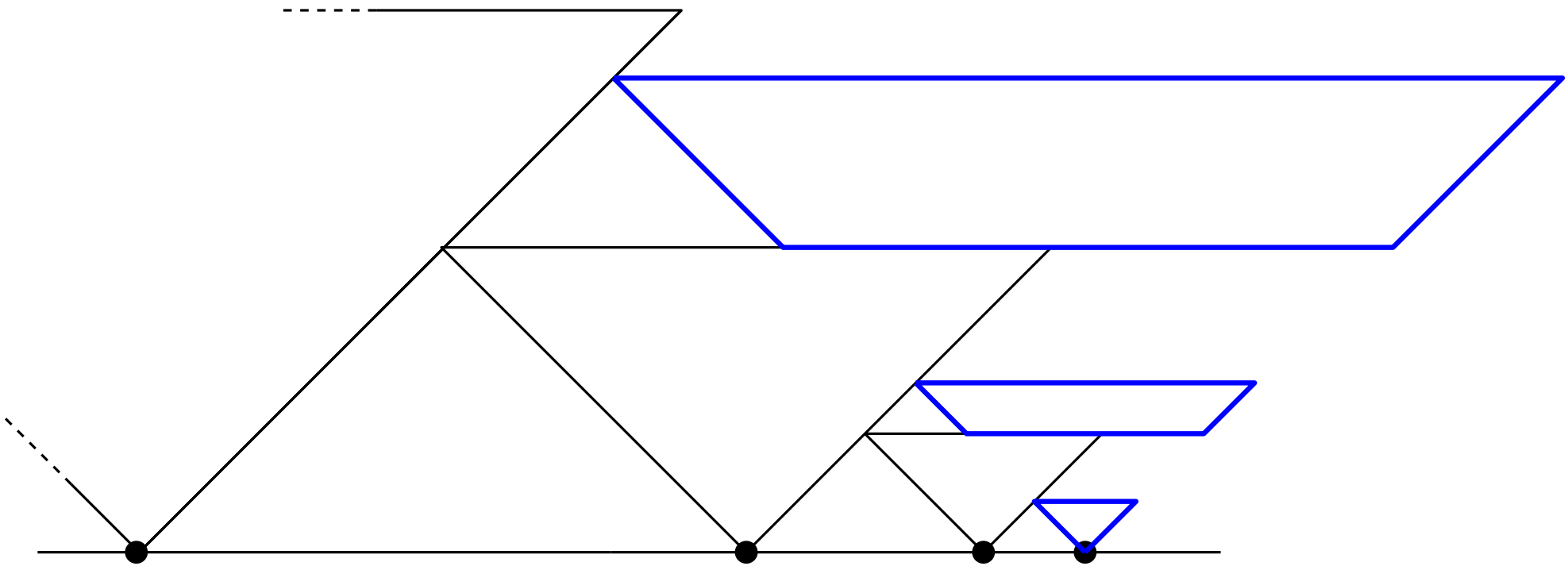
Problem



Problem



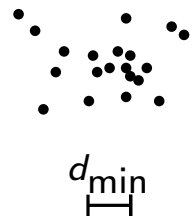
Problem



Active Range Optimisation

Given: set of points P

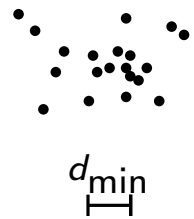
constant d_{min}



Active Range Optimisation

Given: set of points P

constant $d_{\min}$

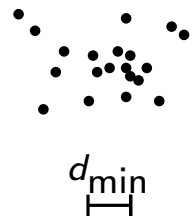


Goal: for each $p \in P$ an active range $A(p)$, such that

Active Range Optimisation

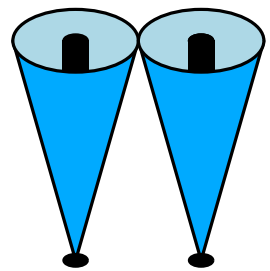
Given: set of points P

constant $d_{\min}$



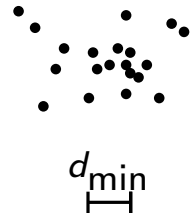
Goal: for each $p \in P$ an active range $A(p)$, such that

each pair of visible points have a distance $> d_{\min}$



Active Range Optimisation

Given: set of points P

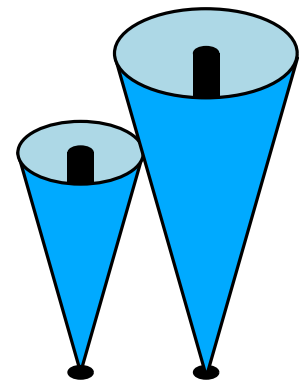


constant $d_{\min}$

Goal: for each $p \in P$ an active range $A(p)$, such that

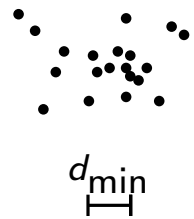
each pair of visible points have a distance $> d_{\min}$

$\max \sum_{p \in P} |A(p)|$



Active Range Optimisation

Given: set of points P

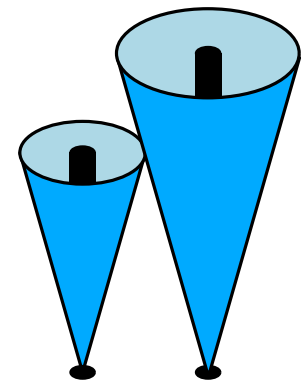


constant $d_{\min}$

Goal: for each $p \in P$ an active range $A(p)$, such that

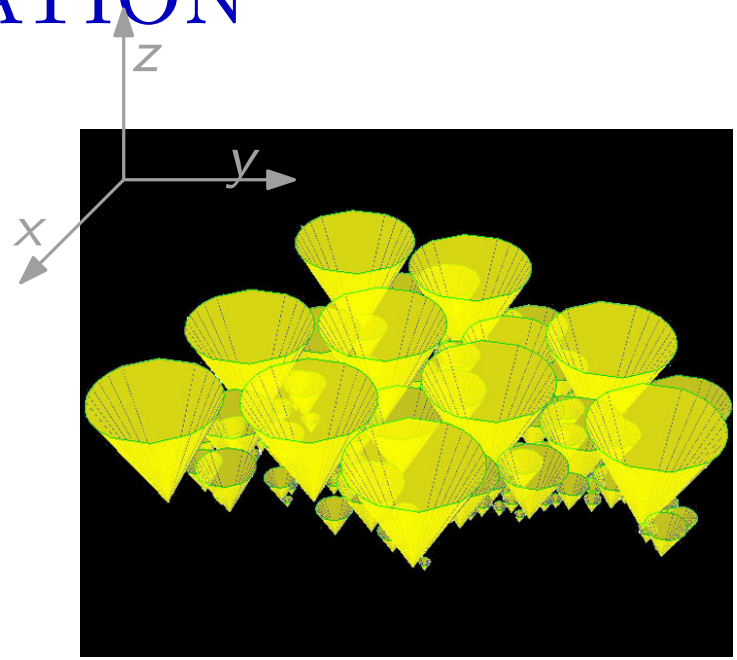
each pair of visible points have a distance $> d_{\min}$

$\max \sum_{p \in P} |A(p)|$

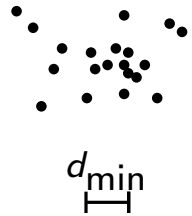


in order to query selected points at run time

Active Range Optimisation



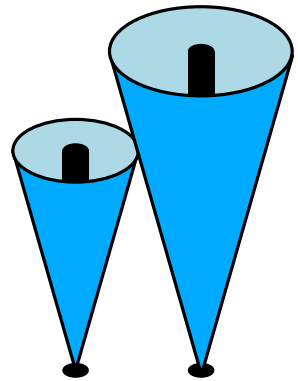
Given: set of points P



constant $d_{\min}$

Goal: for each $p \in P$ an active range $A(p)$, such that

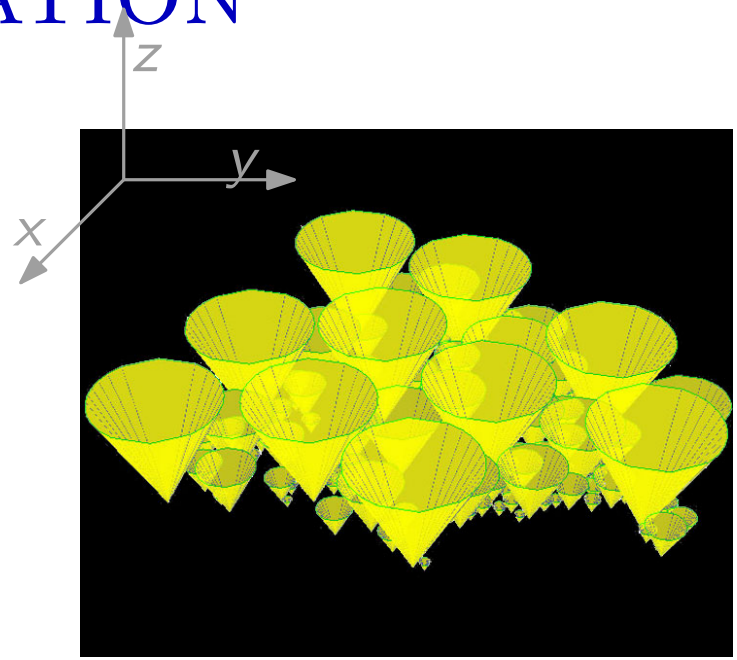
each pair of visible points have a distance $> d_{\min}$



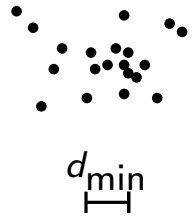
$\max \sum_{p \in P} |A(p)|$

in order to query selected points at run time

Active Range Optimisation



Given: set of points P

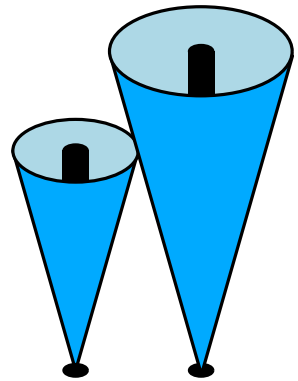


constant d_{min}

Goal: for each $p \in P$ an active range $A(p)$, such that

constrained-based map generalization
(Weibel and Dutton, 1998)

each pair of visible points have a distance $> d_{min}$

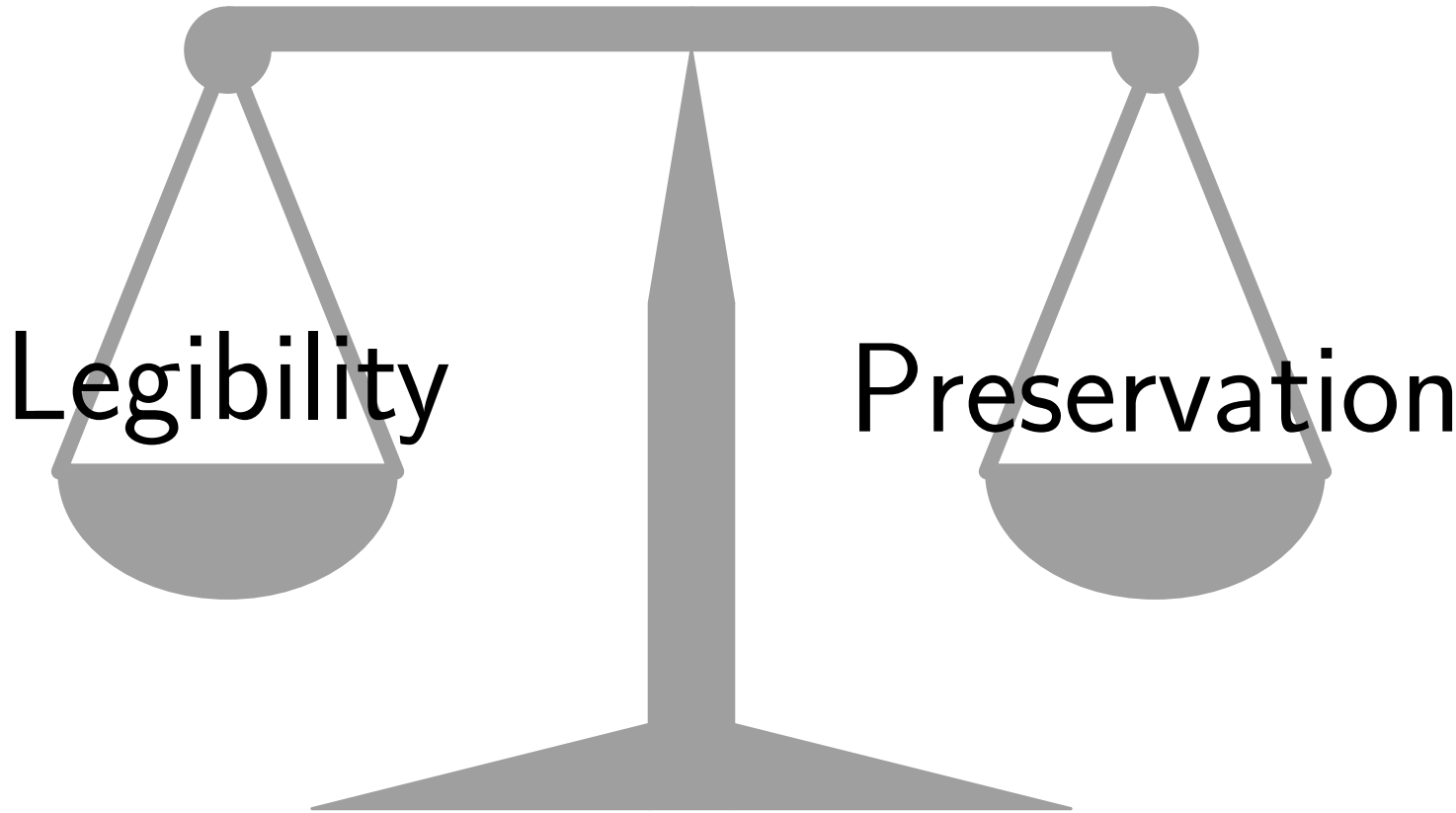


$$\max \sum_{p \in P} |A(p)|$$

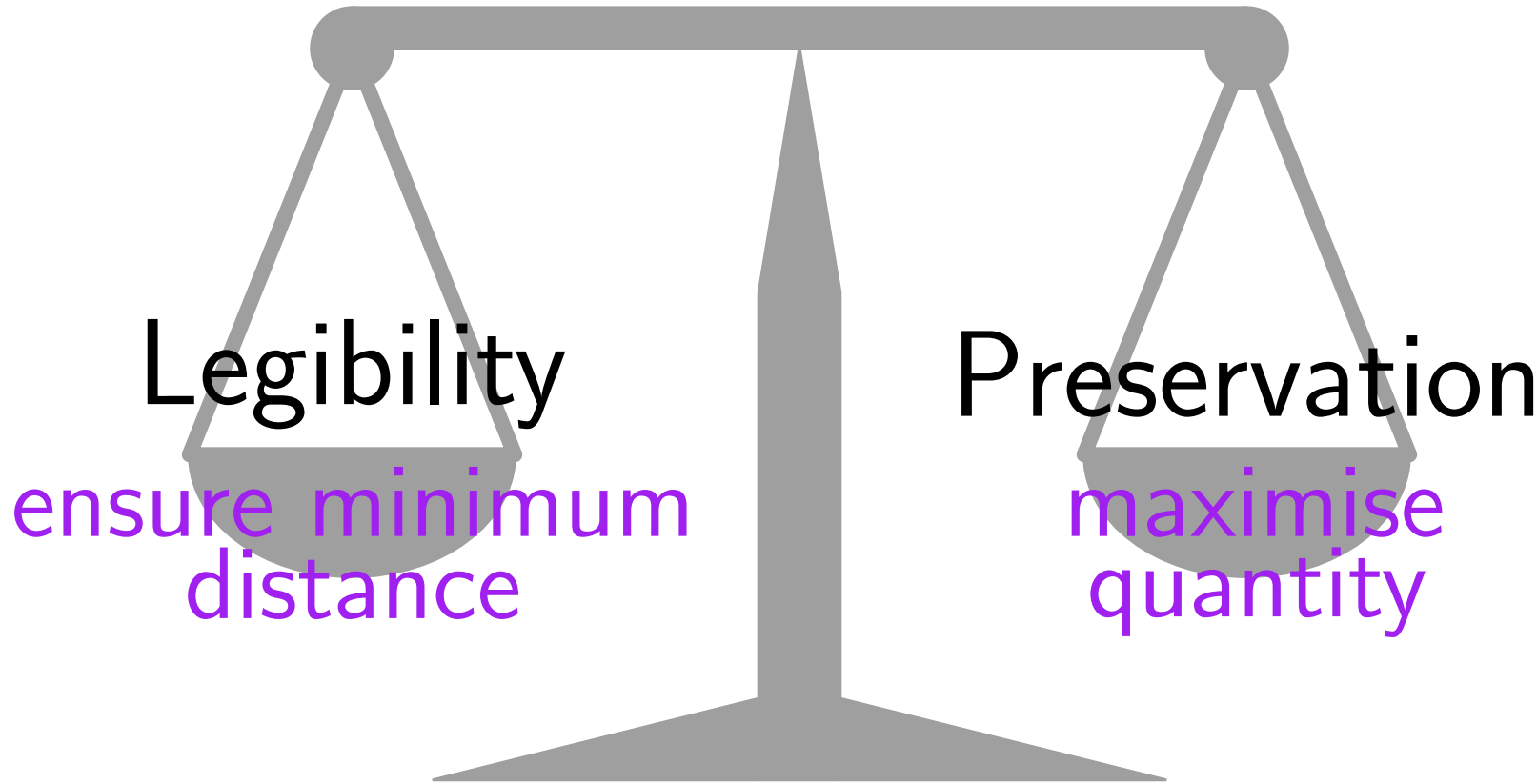
in order to query selected points at run time

Generalisation (Burghardt et al., 2007)

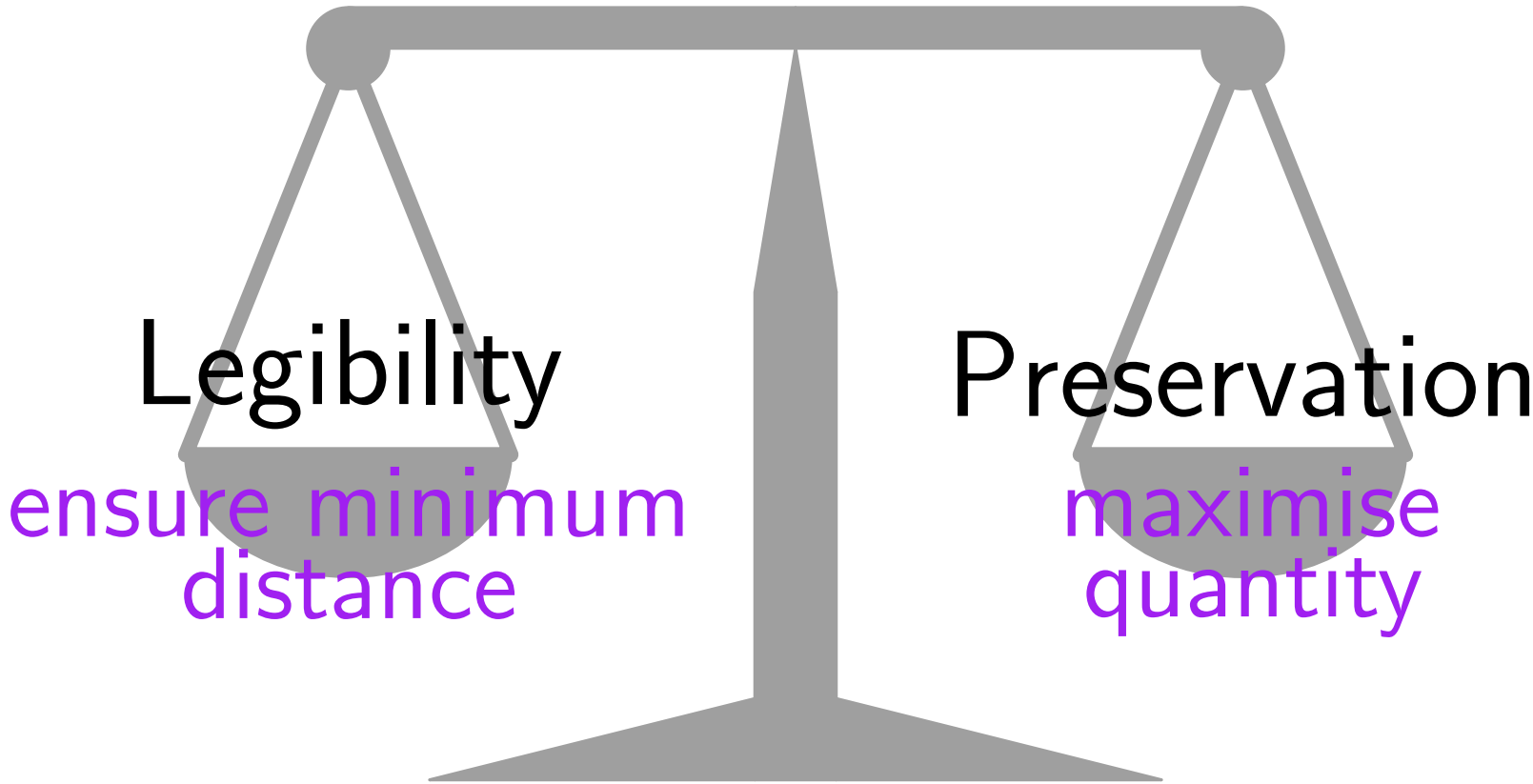
Generalisation (Burghardt et al., 2007)



Generalisation (Burghardt et al., 2007)

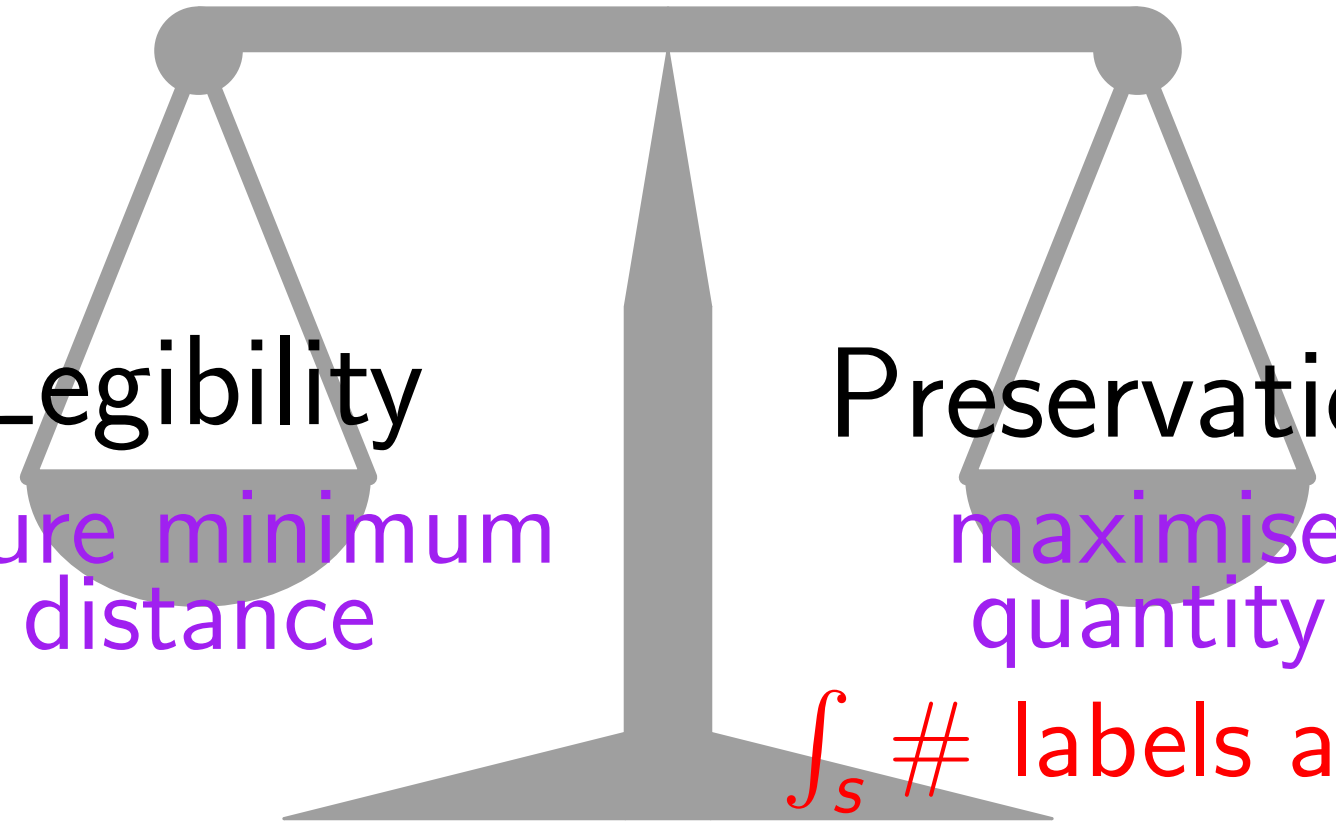


Generalisation (Burghardt et al., 2007)

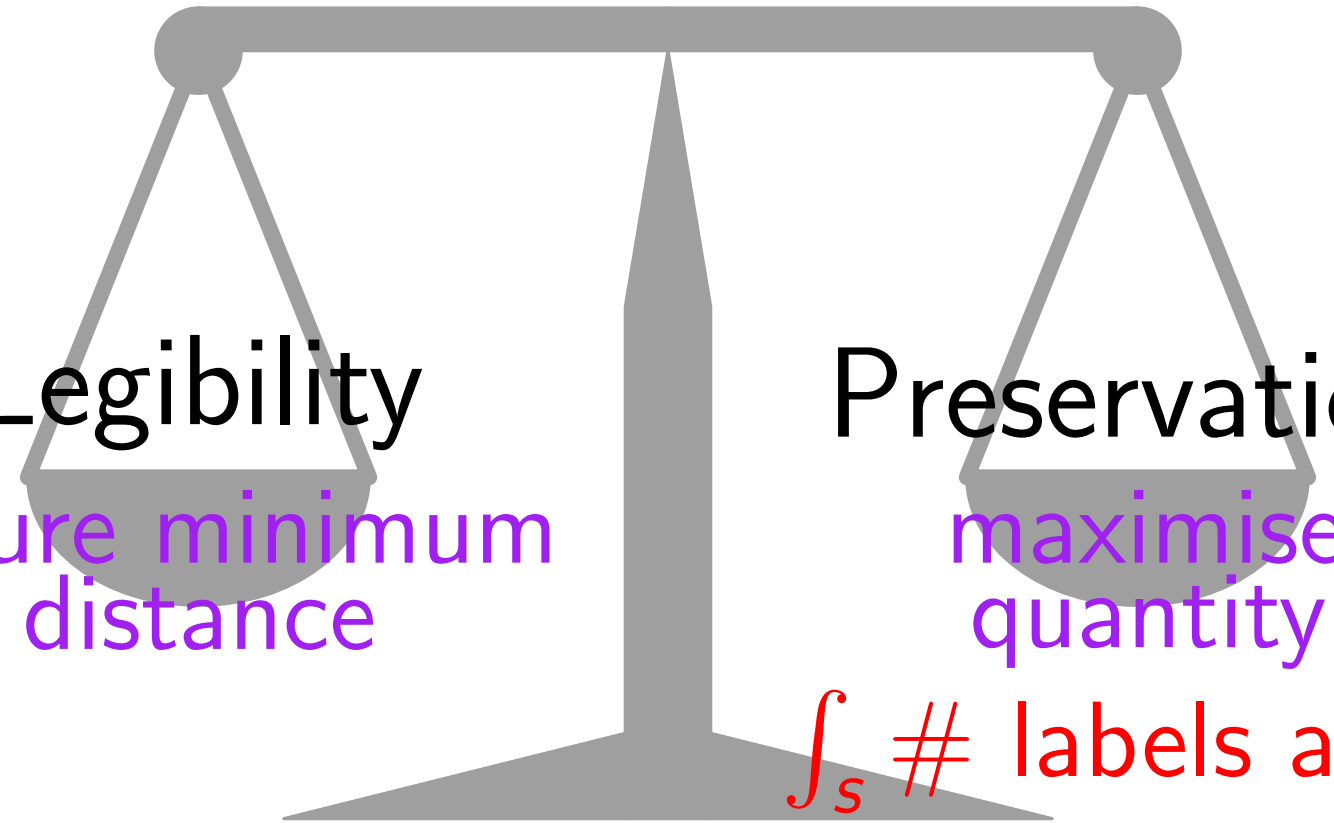


solution: optimise for each scale

Generalisation (Burghardt et al., 2007)



Generalisation (Burghardt et al., 2007)



Legibility

ensure minimum distance

Preservation

maximise quantity

$\int_s$ # labels at scale s

over all scales

solution: optimise ~~for each~~ in advance

$\Rightarrow$ **no flickering**

Optimisation

Mixed-Integer Linear Program (MIP)

Optimisation

Mixed-Integer Linear Program (MIP)

optimal solutions

Optimisation

Mixed-Integer Linear Program (MIP)

optimal solutions

high computation time

Optimisation

Mixed-Integer Linear Program (MIP)

optimal solutions

high computation time

Heuristics

Optimisation

Mixed-Integer Linear Program (MIP)

optimal solutions

high computation time

Heuristics

easy to understand

hopefully nearly-optimal solutions

near-linear running time
(under reasonable assumptions)

Optimisation

Mixed-Integer Linear Program (MIP)

optimal solutions

high computation time

Heuristics

easy to understand

hopefully nearly-optimal solutions

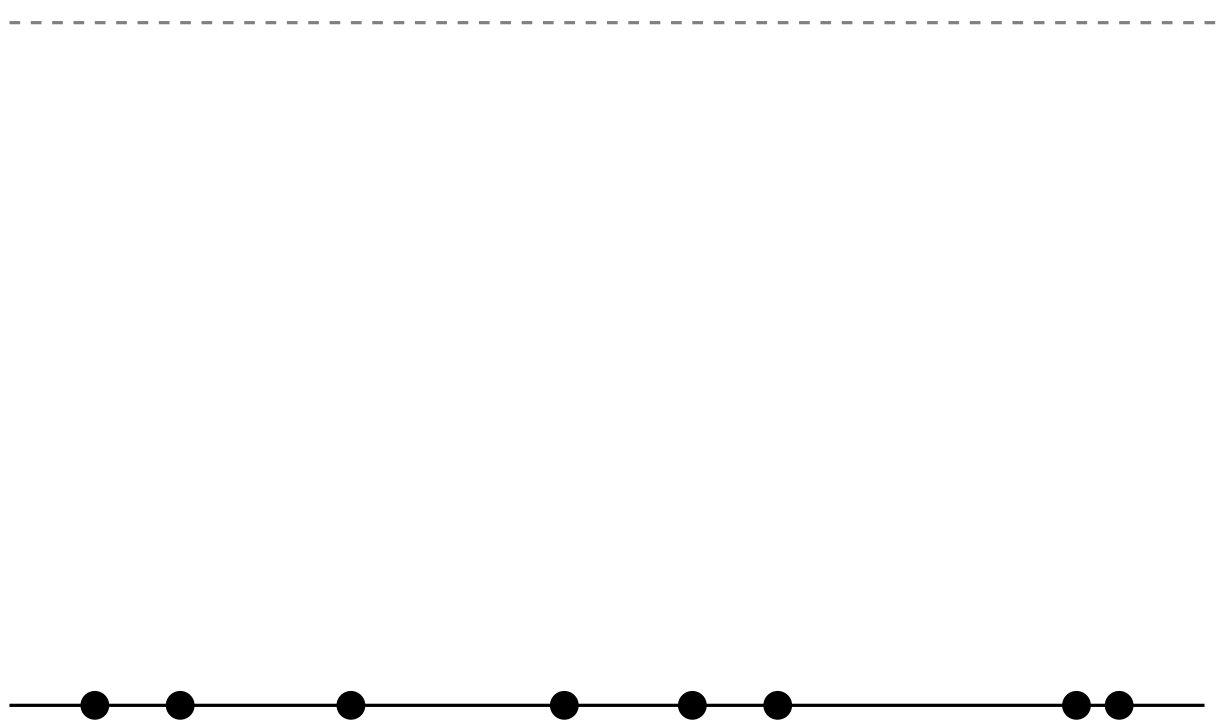
near-linear running time
(under reasonable assumptions)

Shrinking Cones

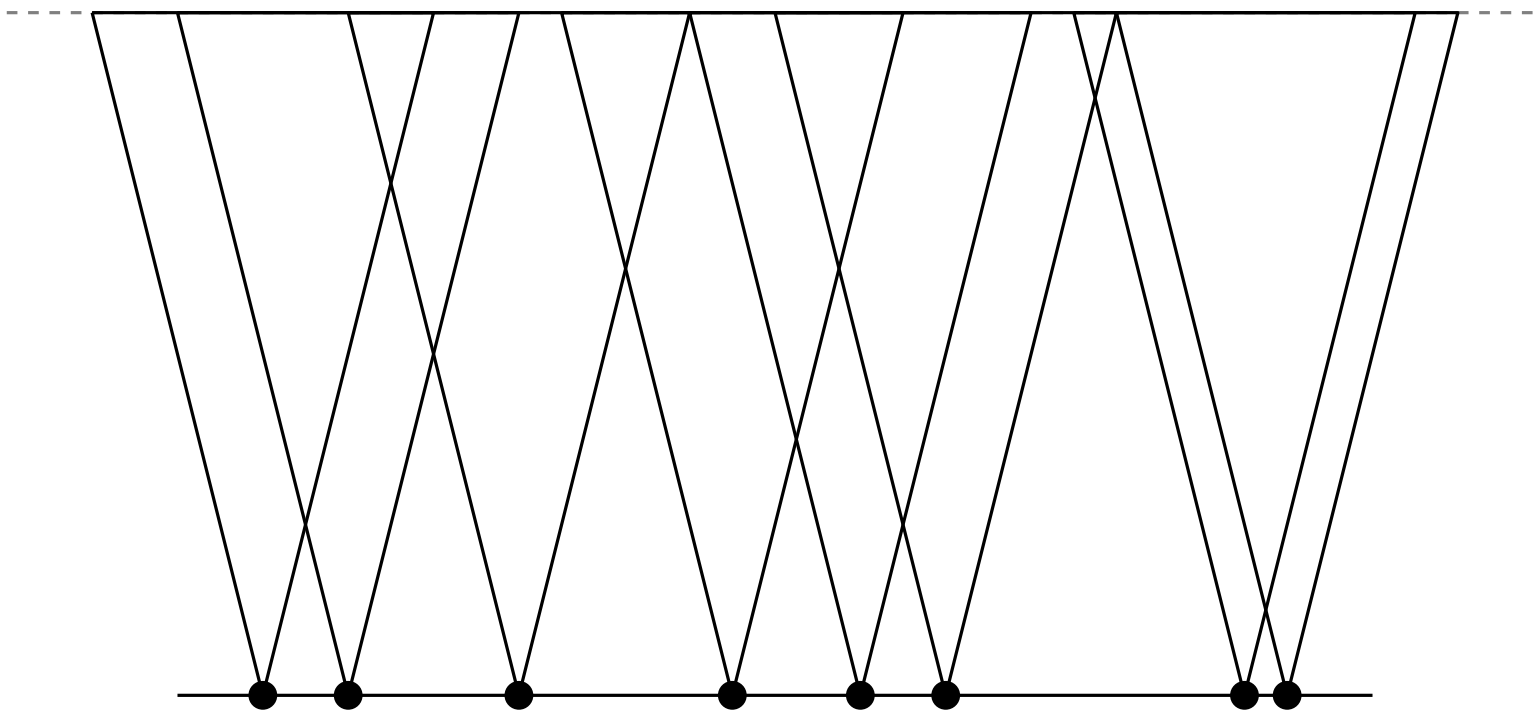
Shrinking Cones (Been et al., 2006 / Been et al., 2010)

Shrinking Cones (Been et al., 2006 / Been et al., 2010)

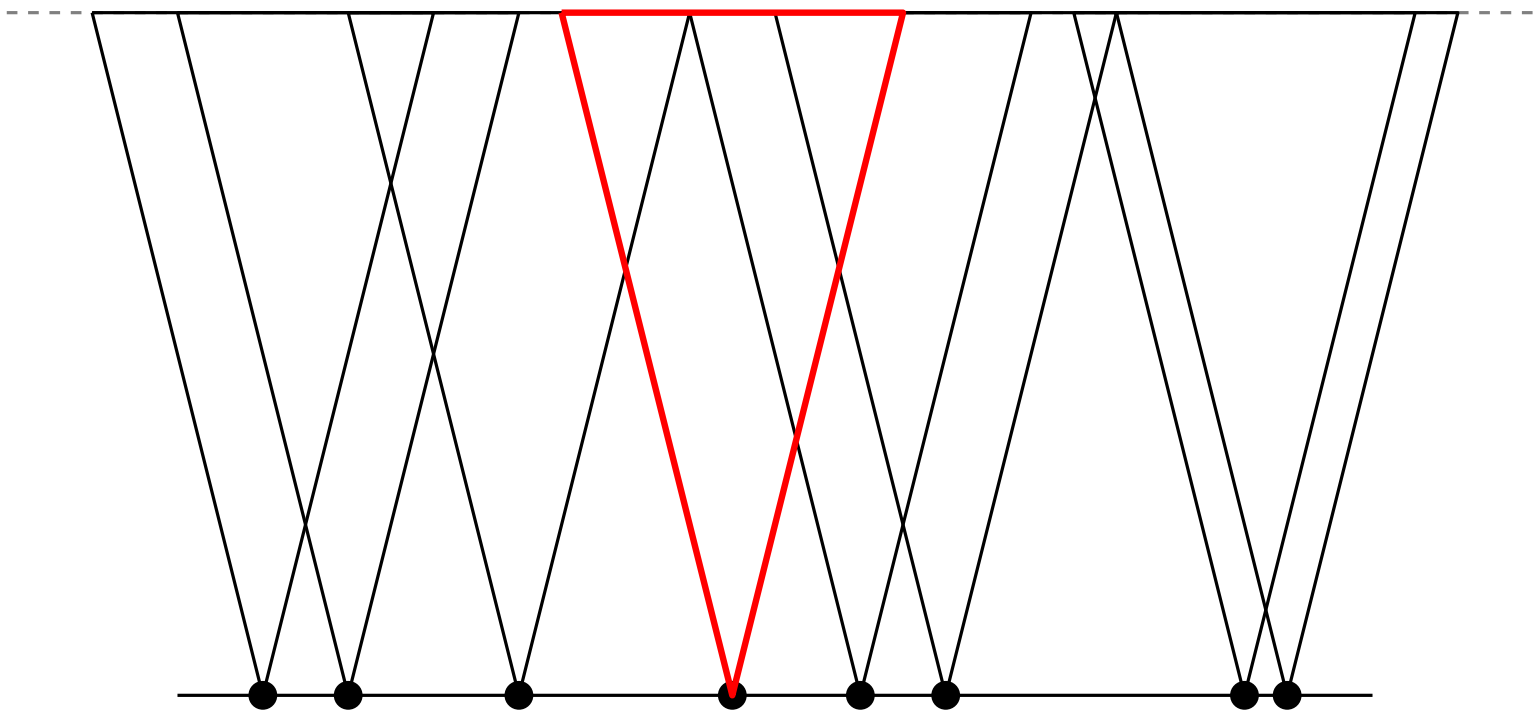
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



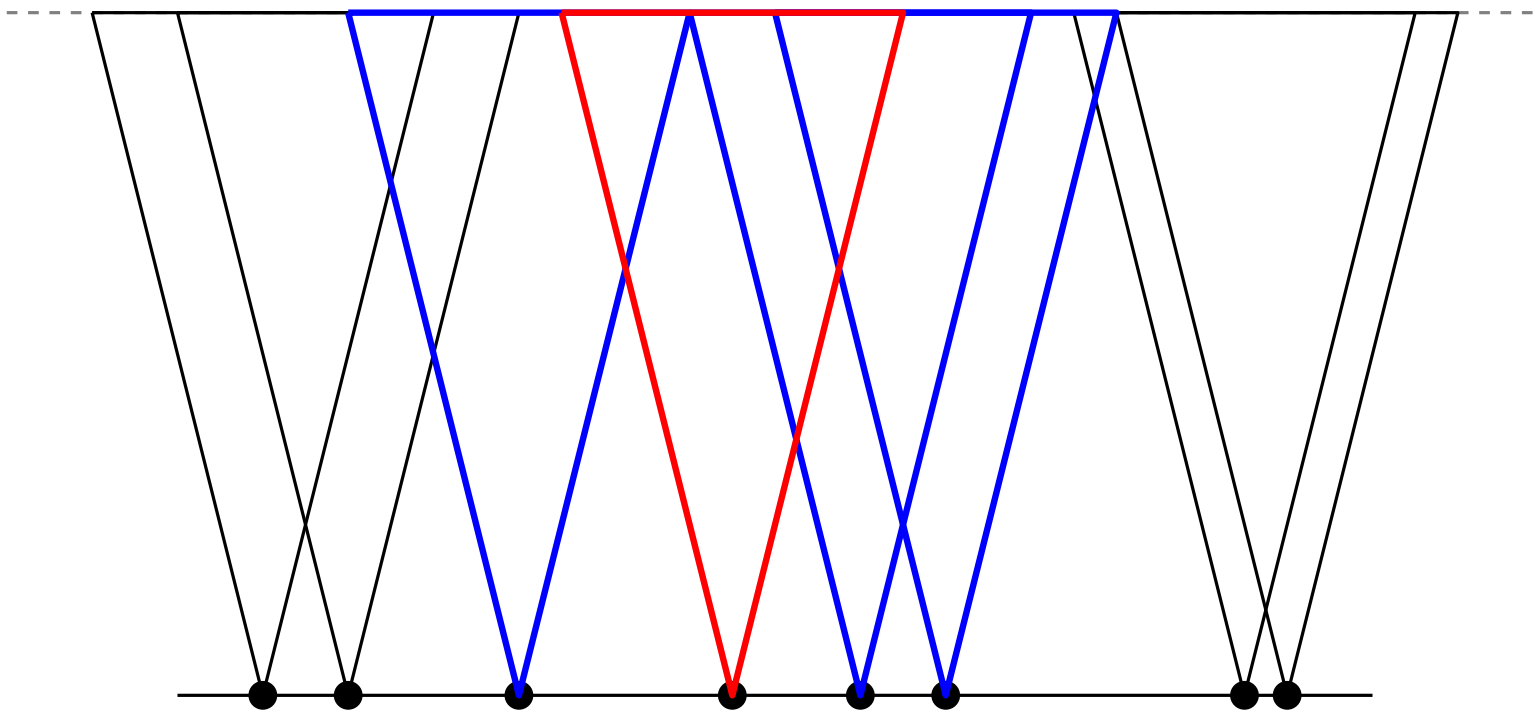
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



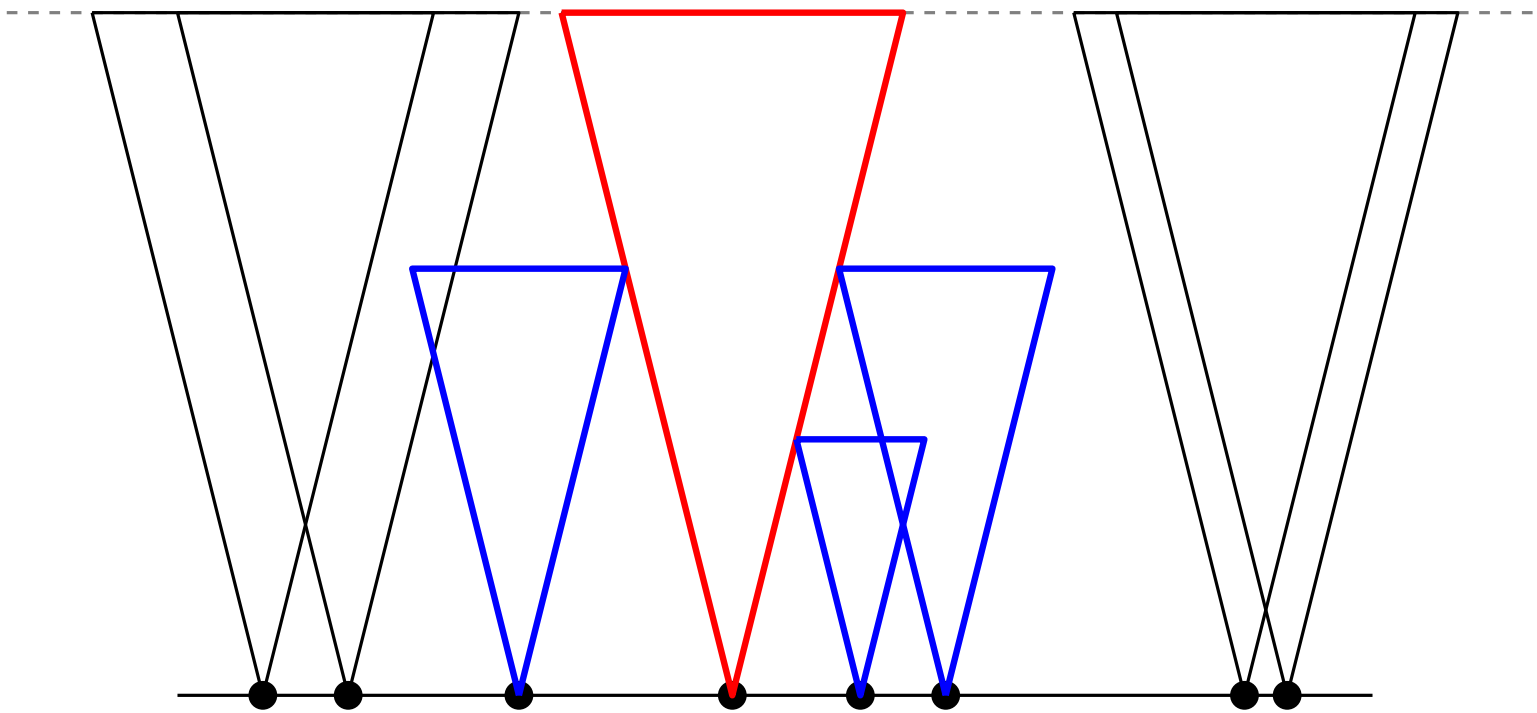
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



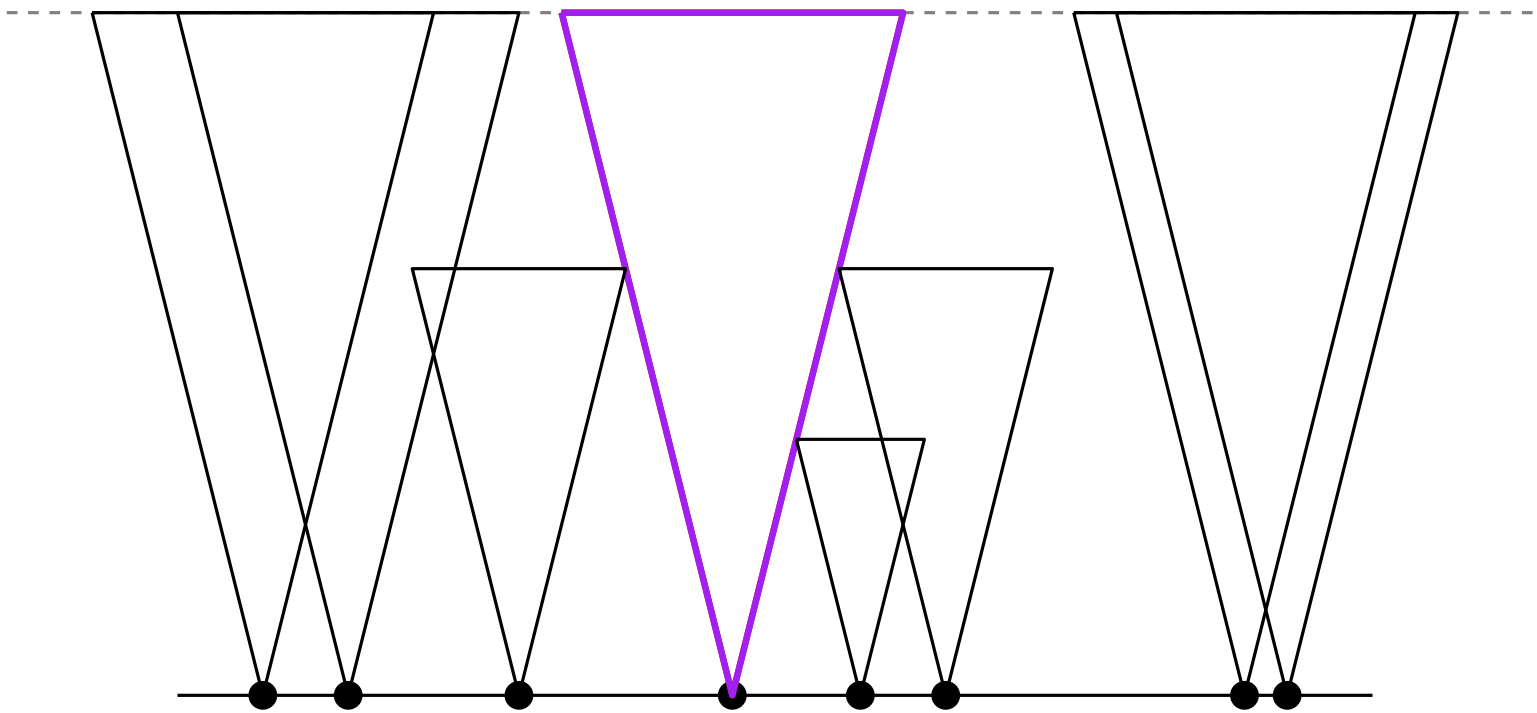
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



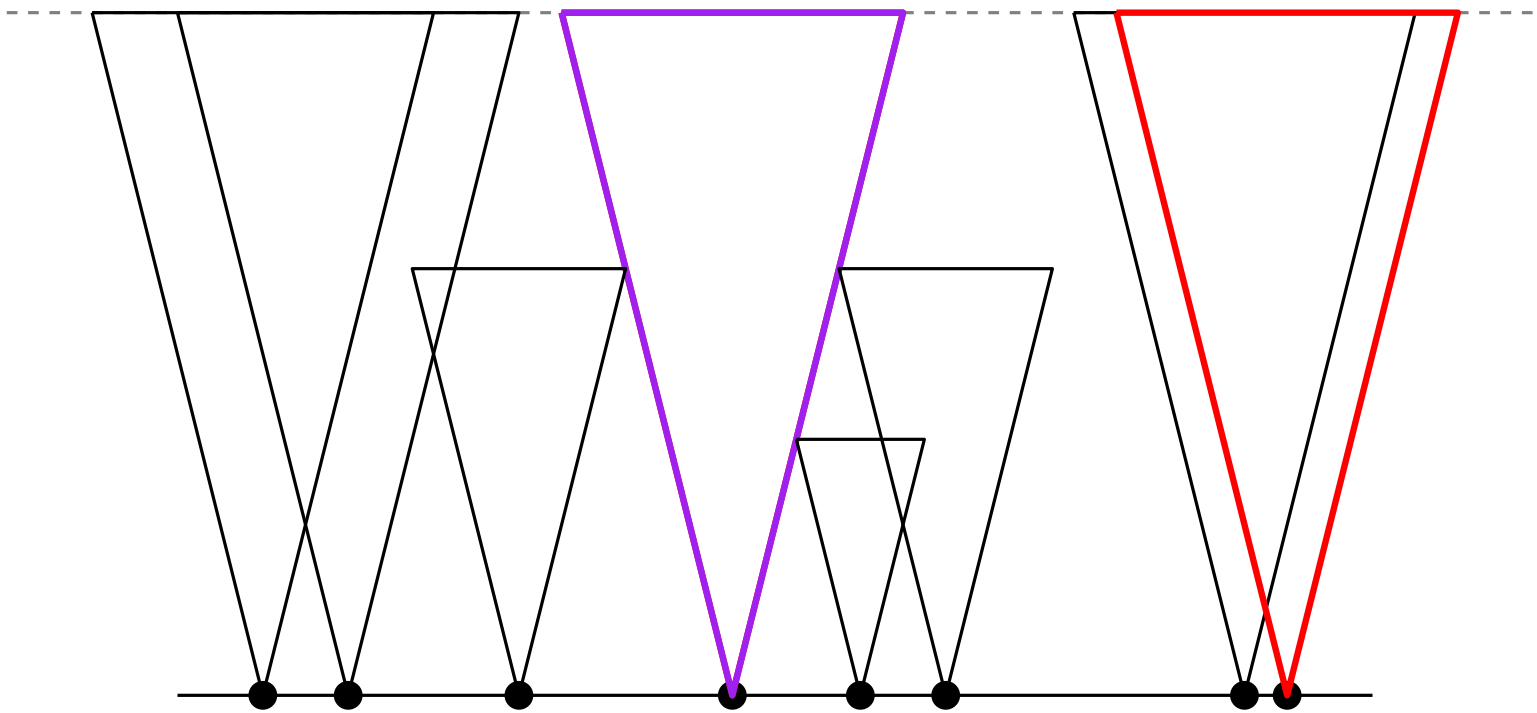
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



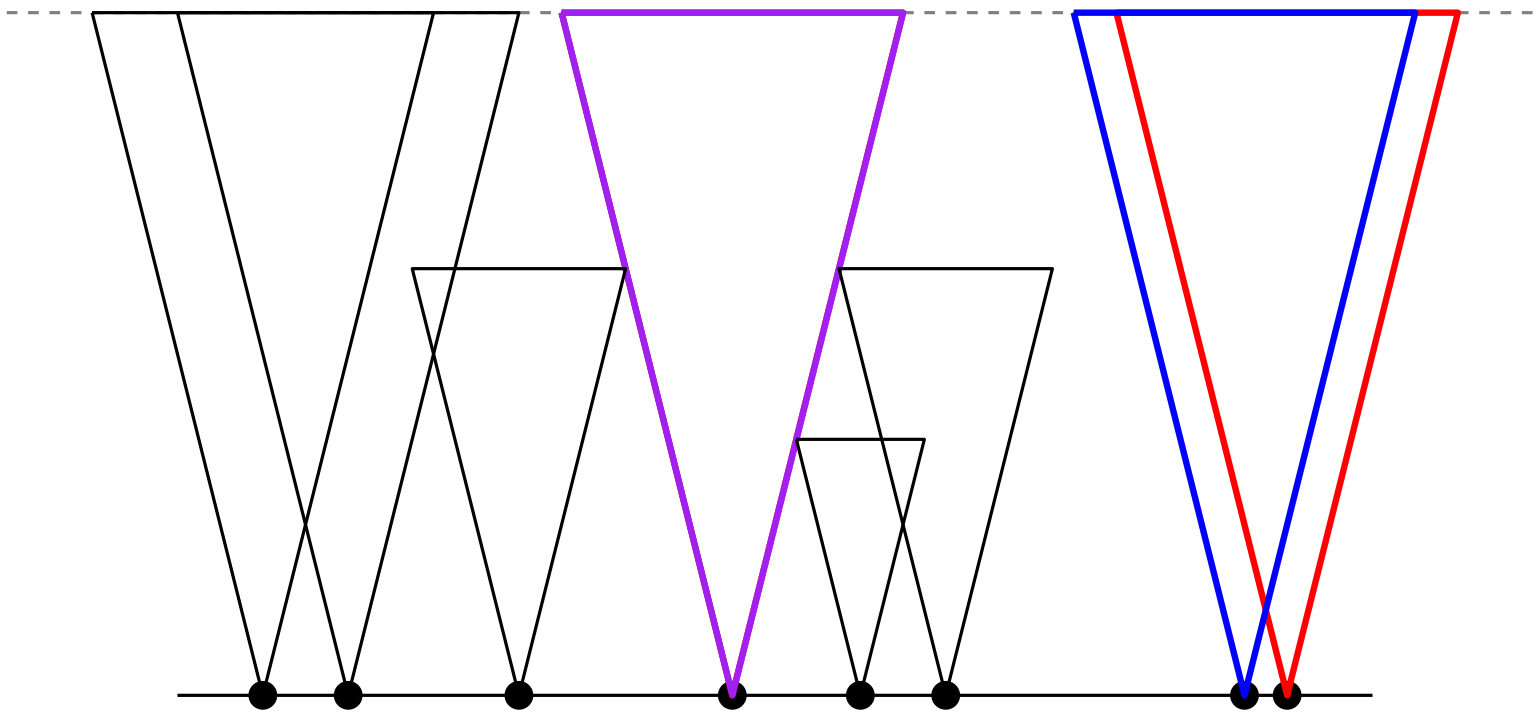
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



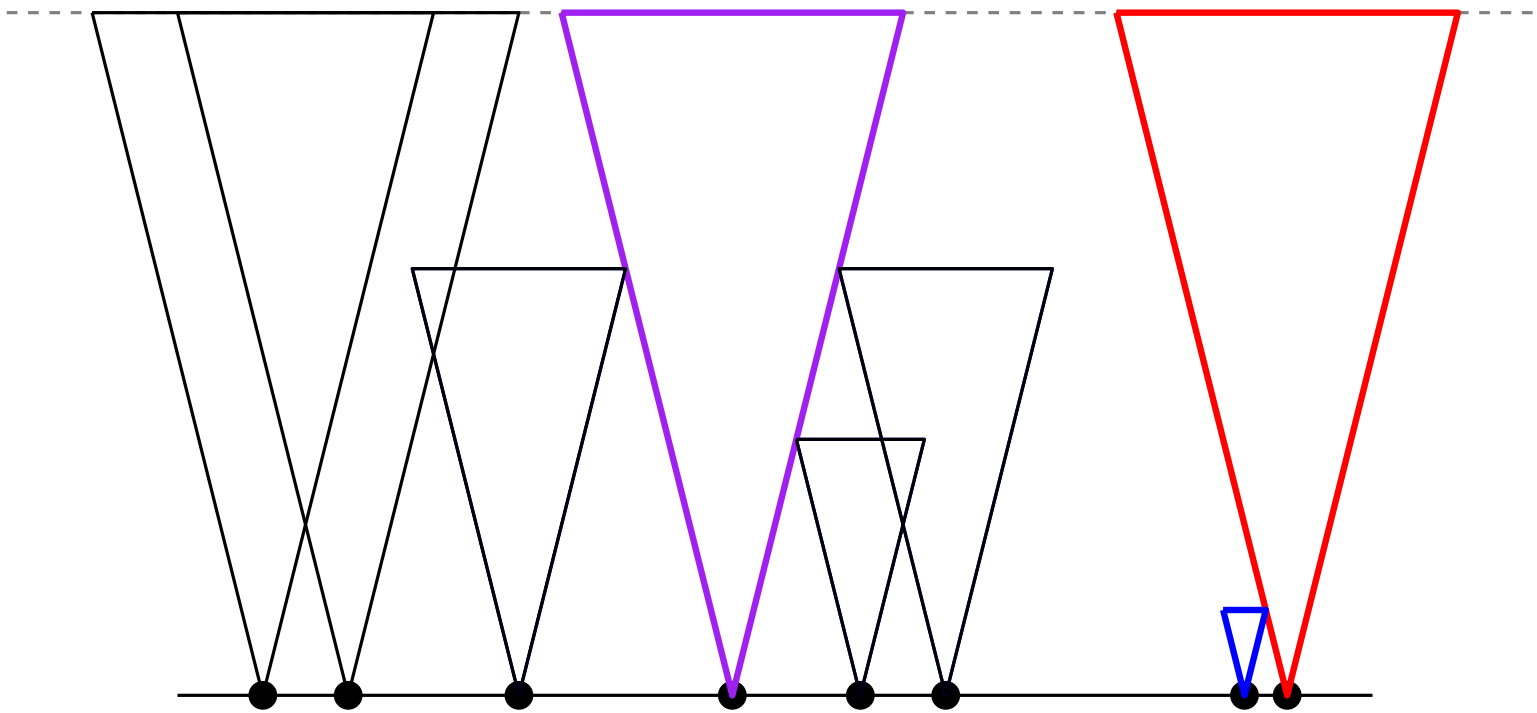
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



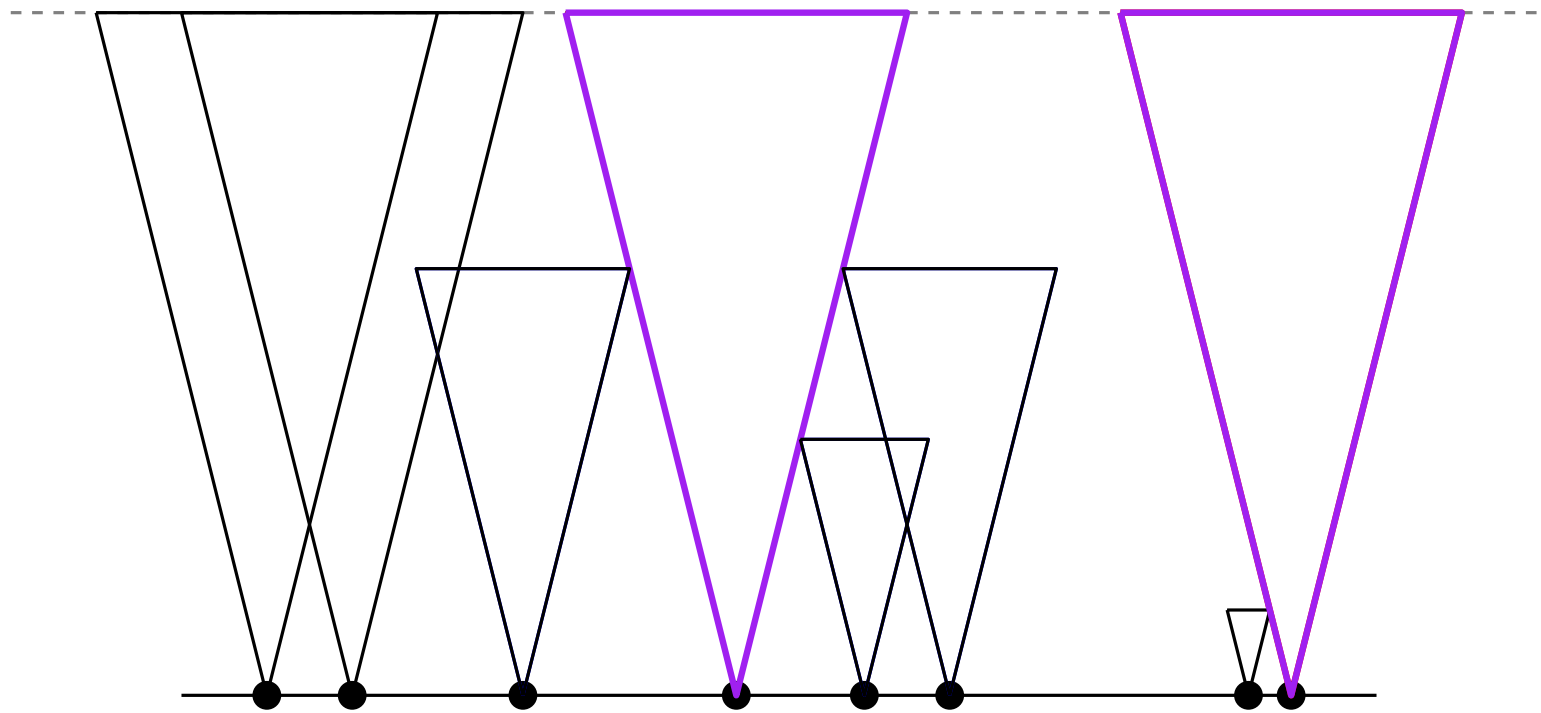
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



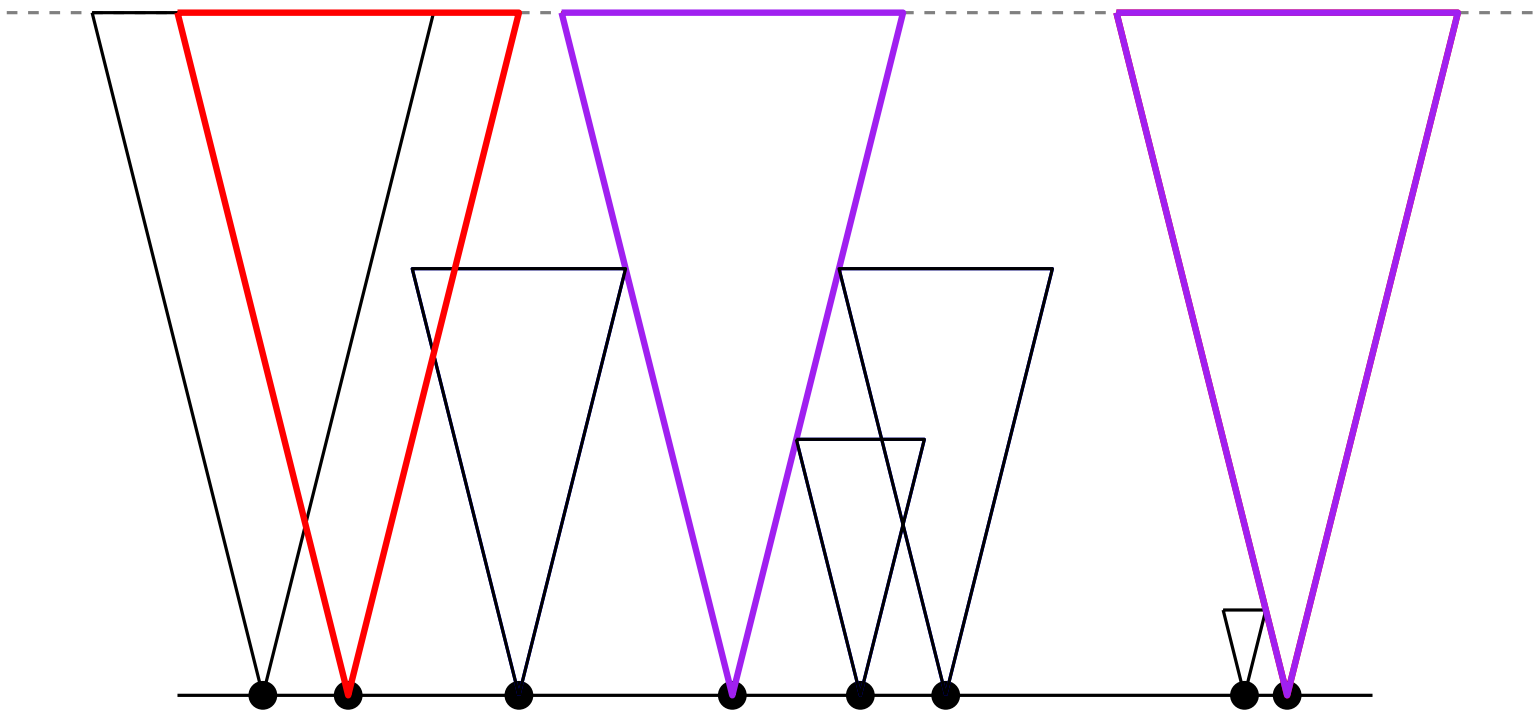
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



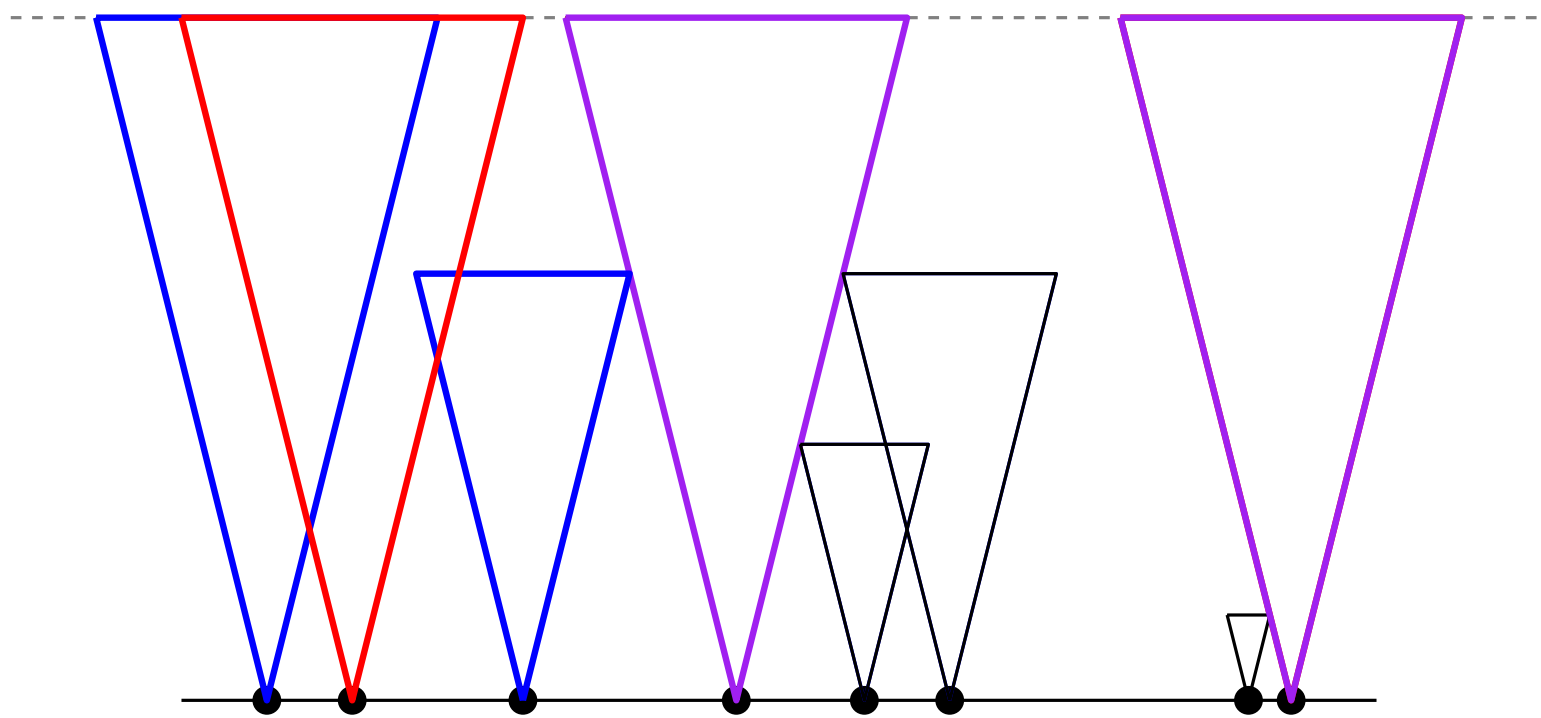
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



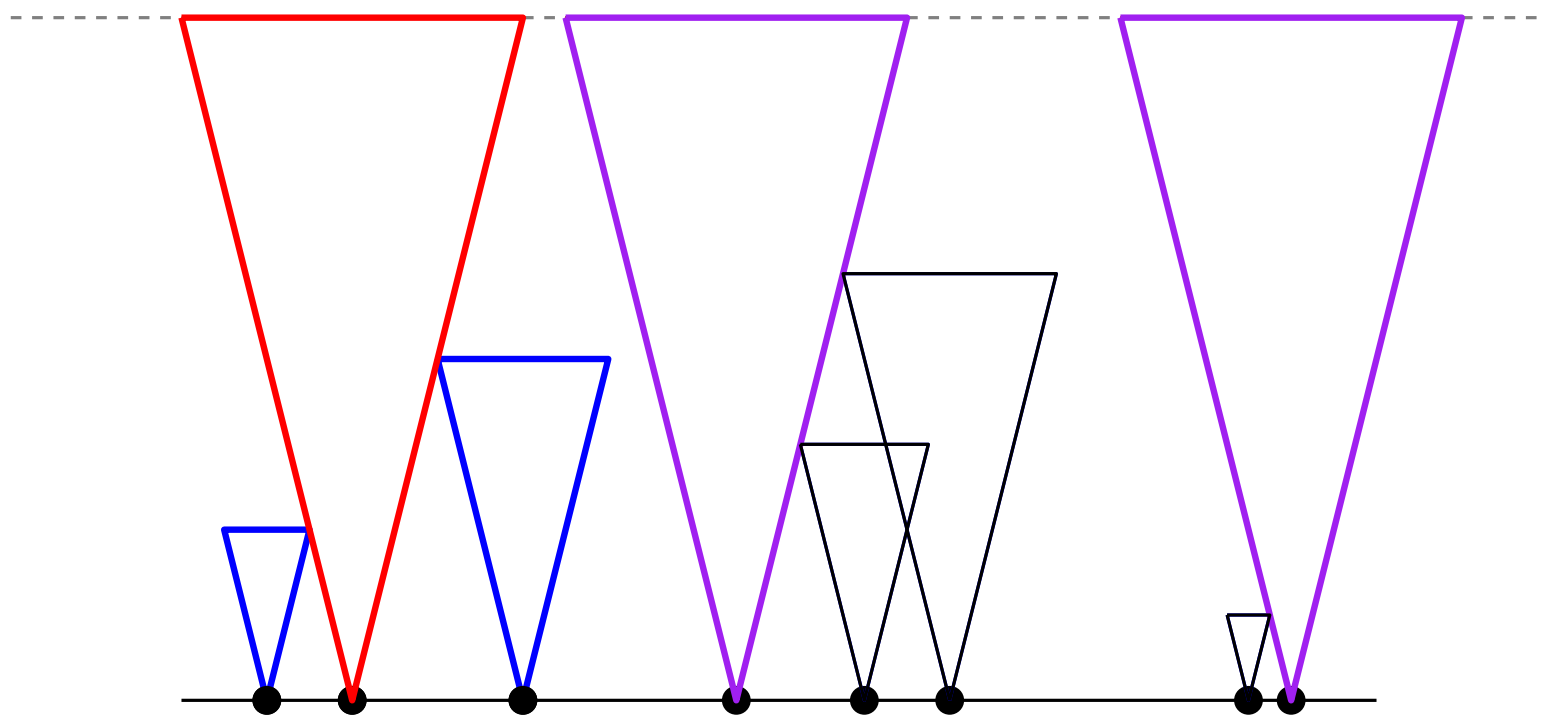
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



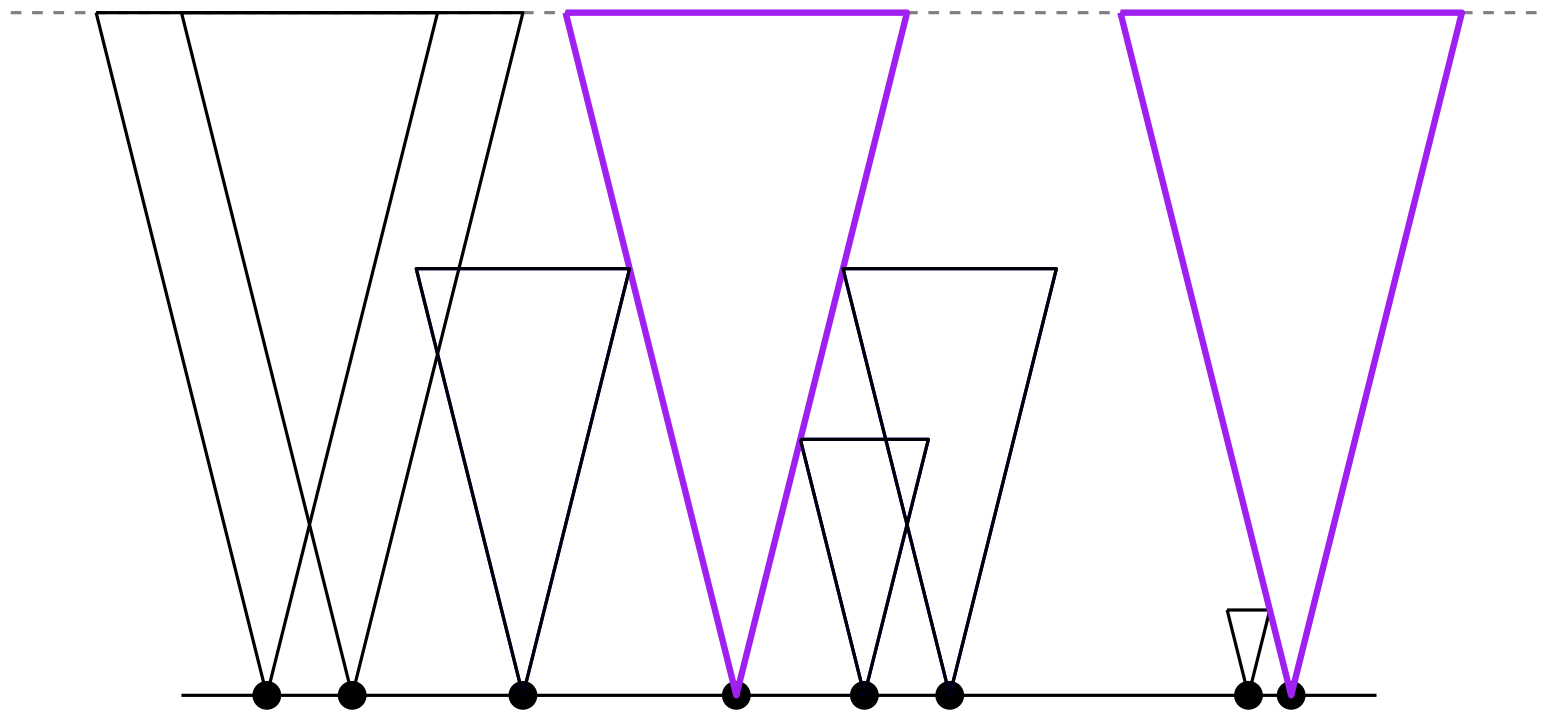
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



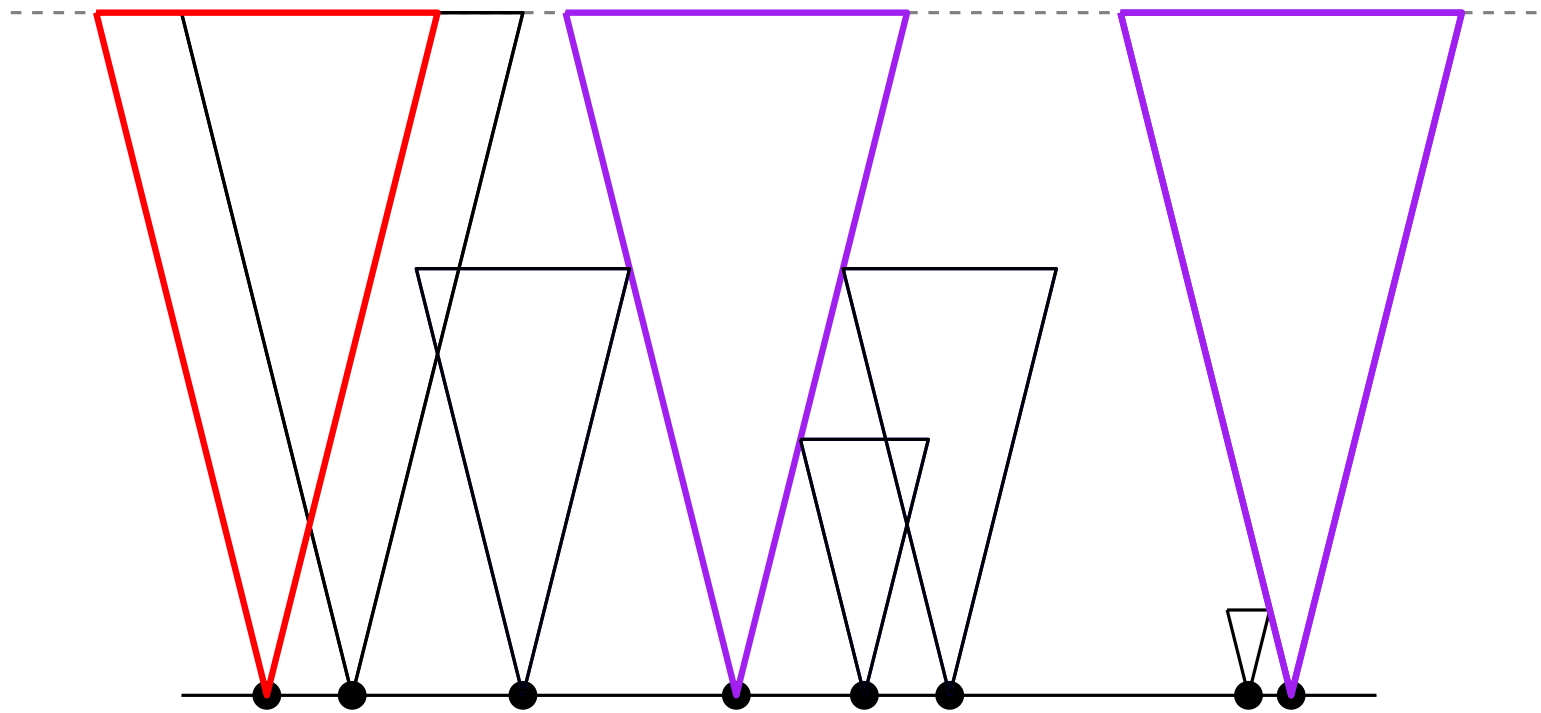
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



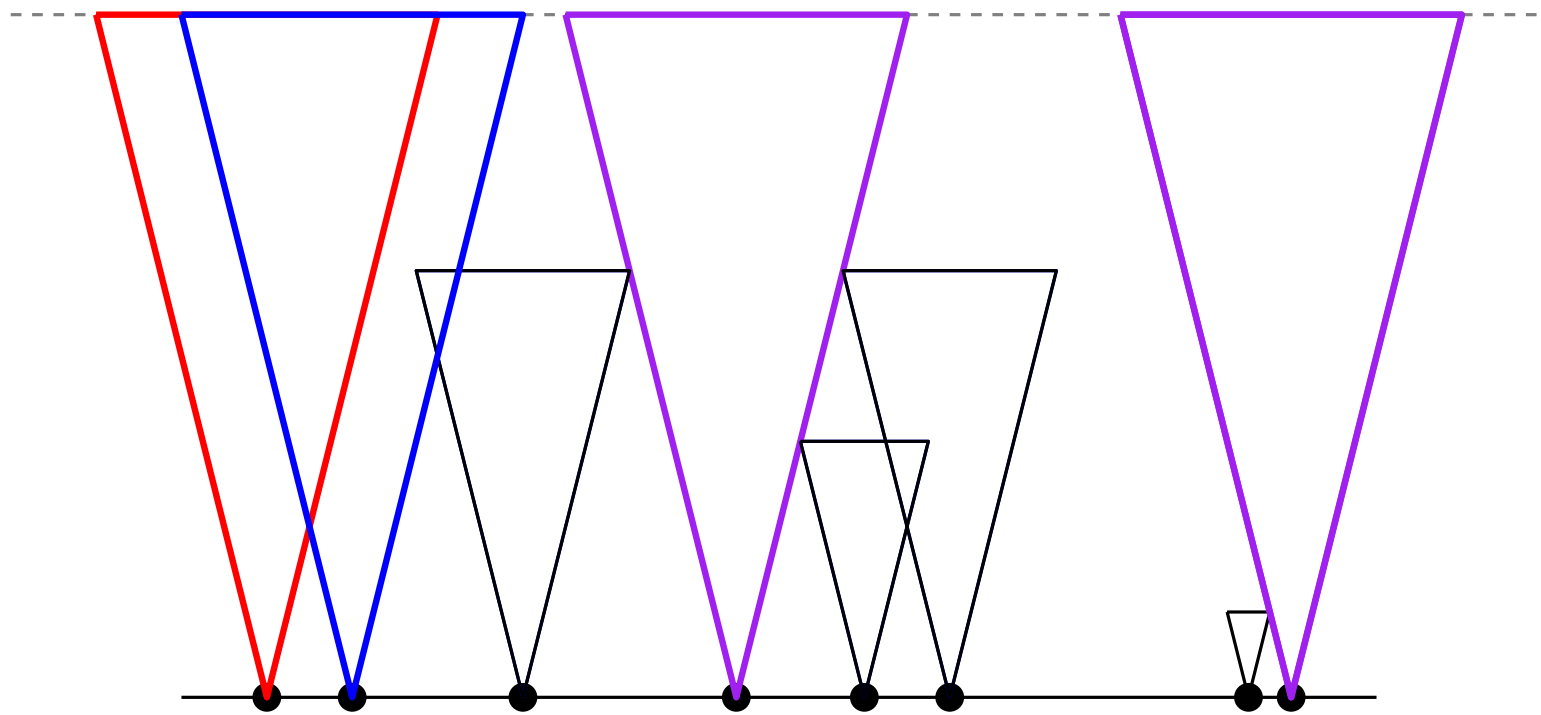
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



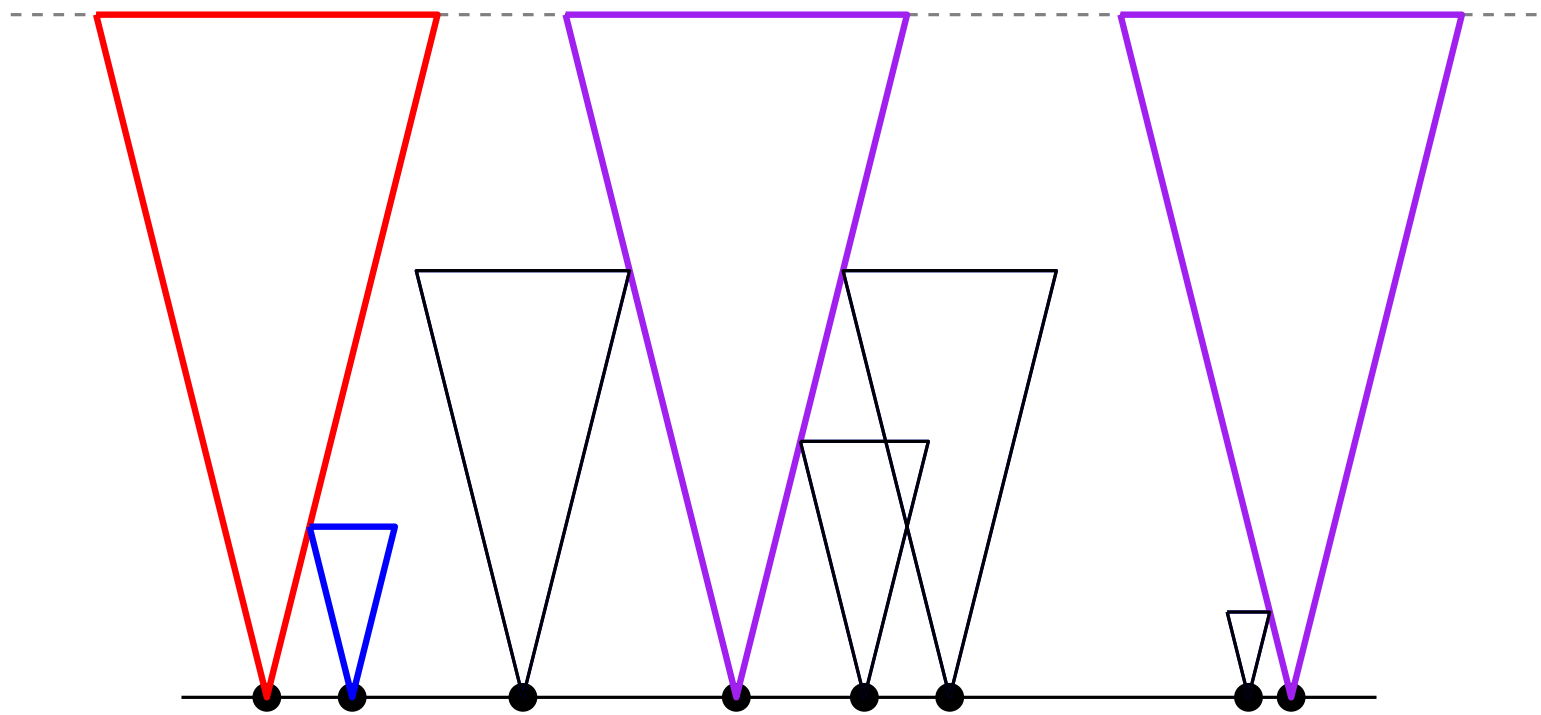
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



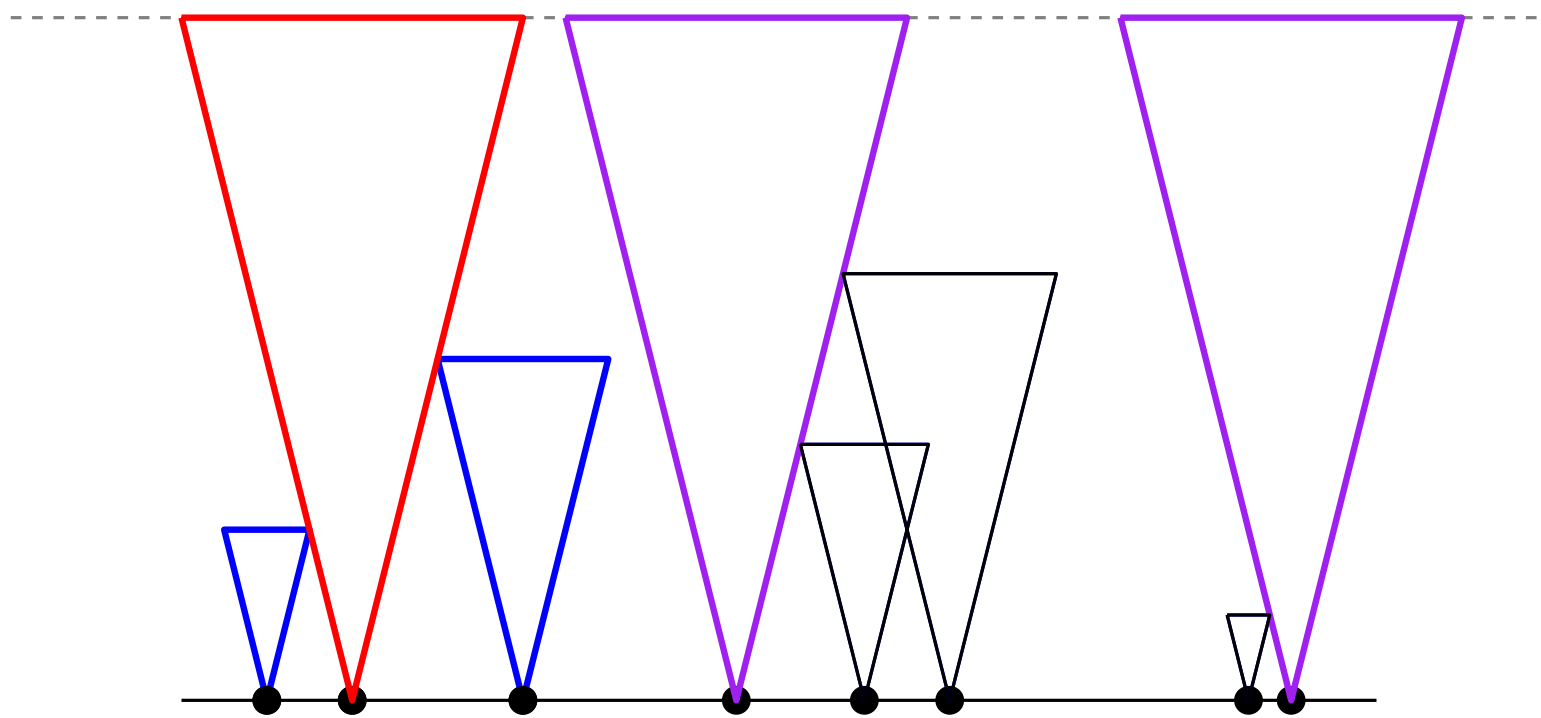
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



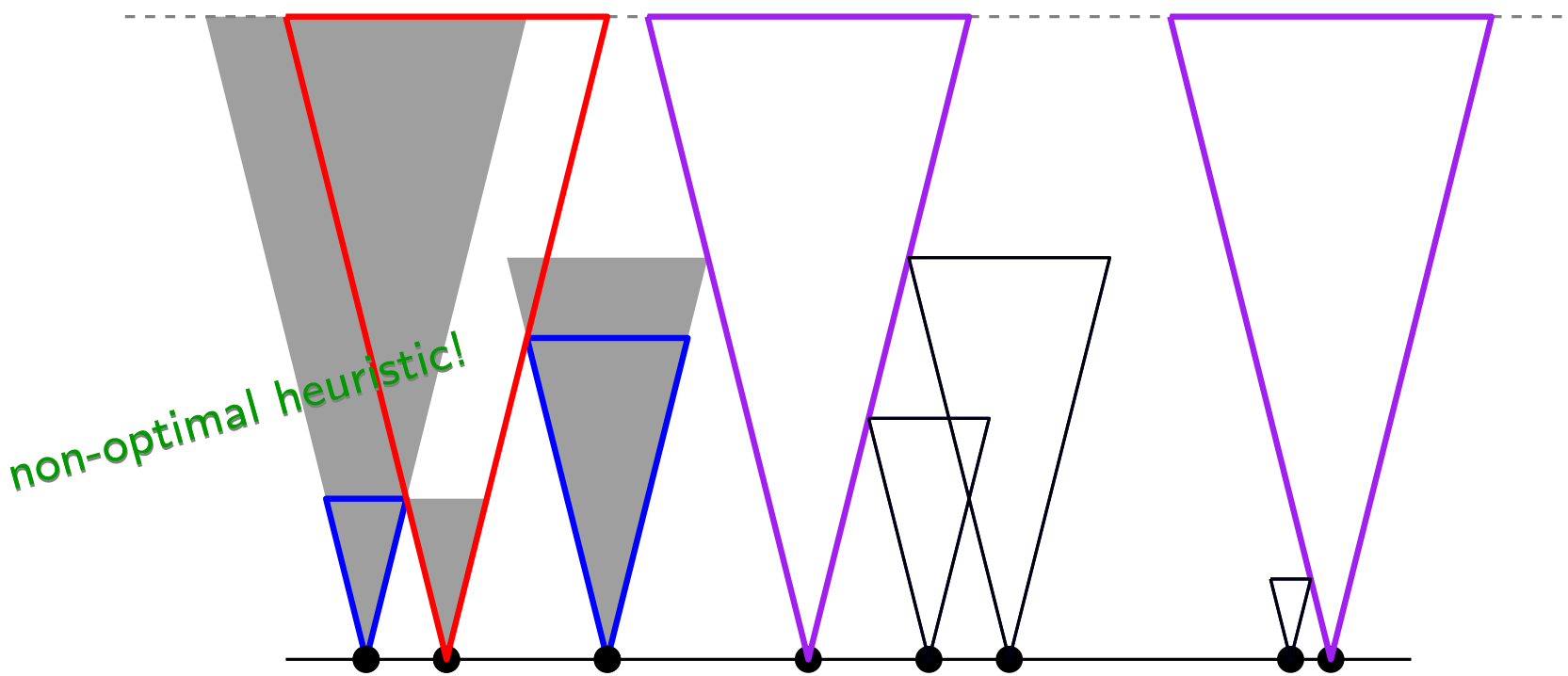
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



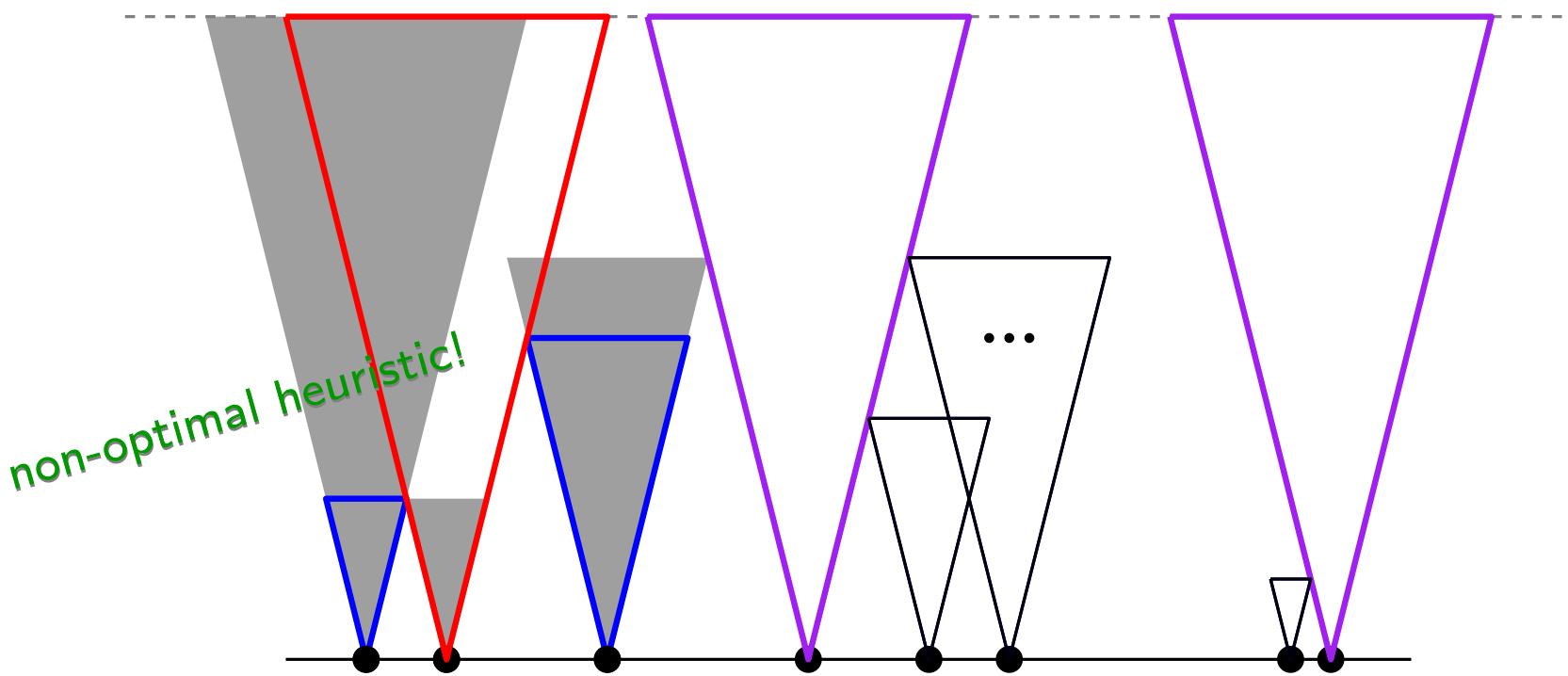
Shrinking Cones (Been et al., 2006 / Been et al., 2010)



Shrinking Cones (Been et al., 2006 / Been et al., 2010)

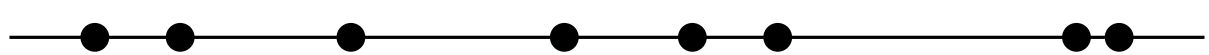


Shrinking Cones (Been et al., 2006 / Been et al., 2010)

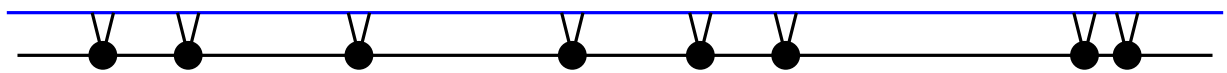


Growing Cones (M0)

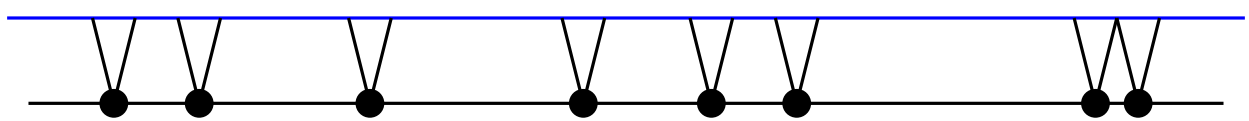
Growing Cones (M0)



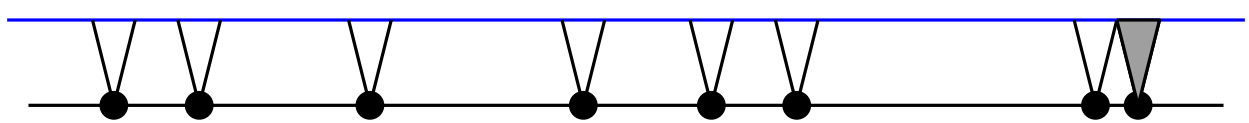
Growing Cones (M0)



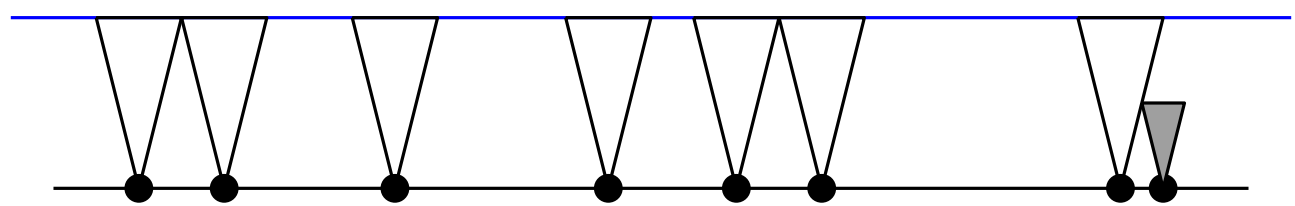
Growing Cones (M0)



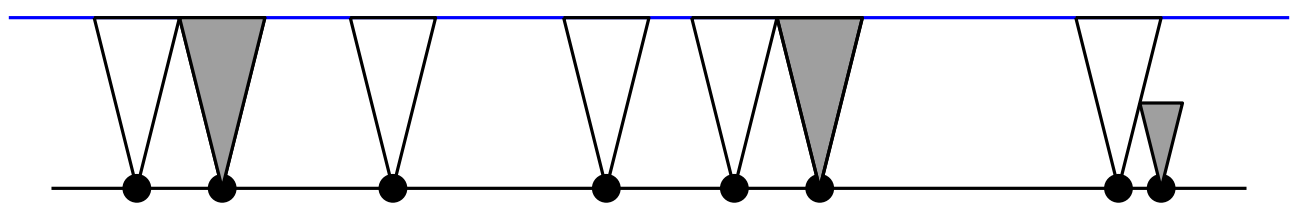
Growing Cones (M0)



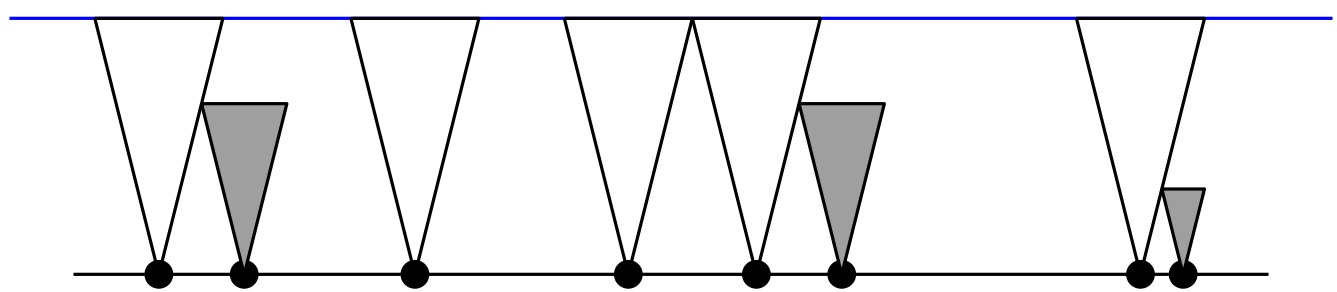
Growing Cones (M0)



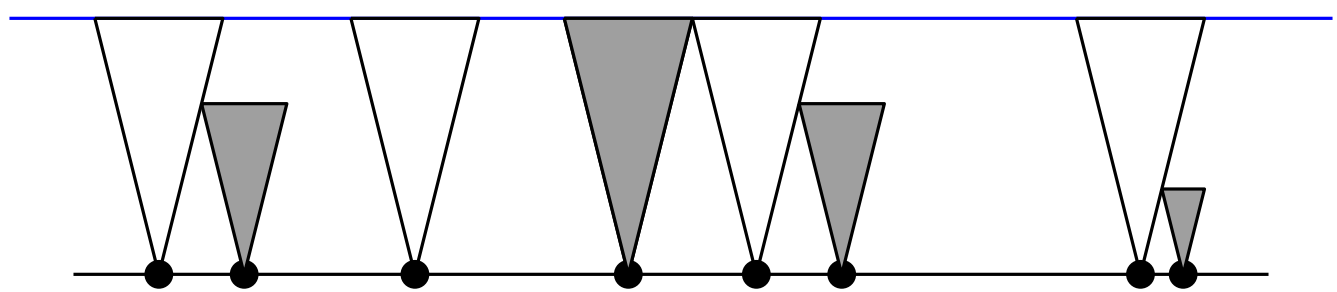
Growing Cones (M0)



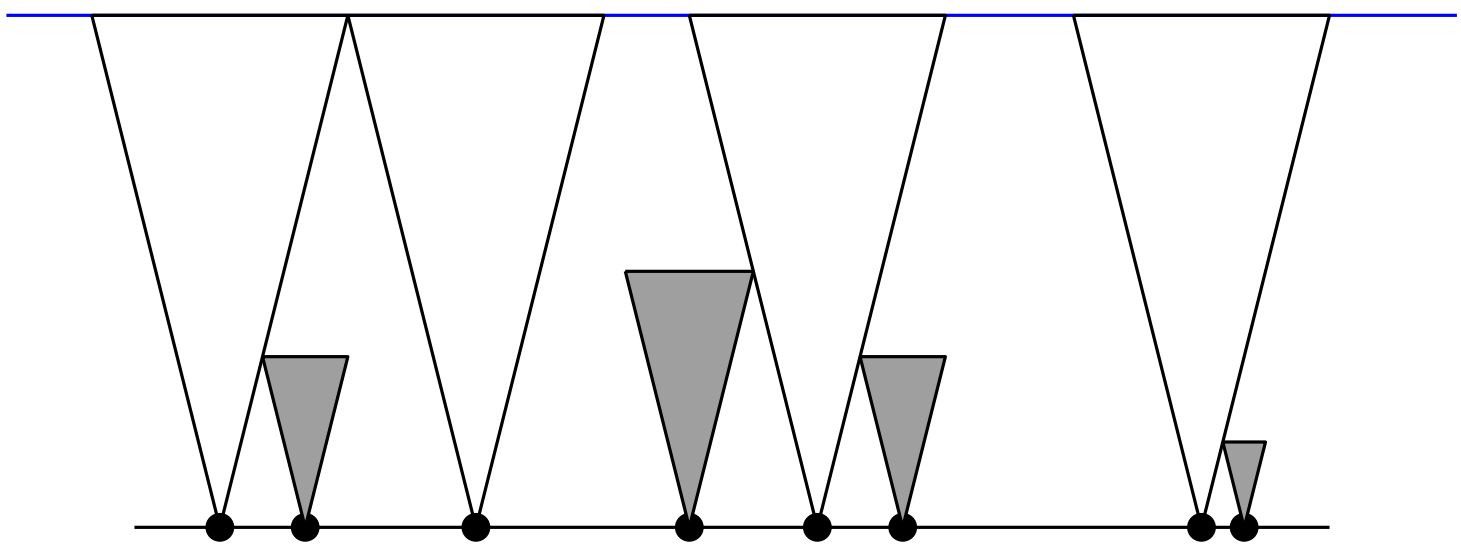
Growing Cones (M0)



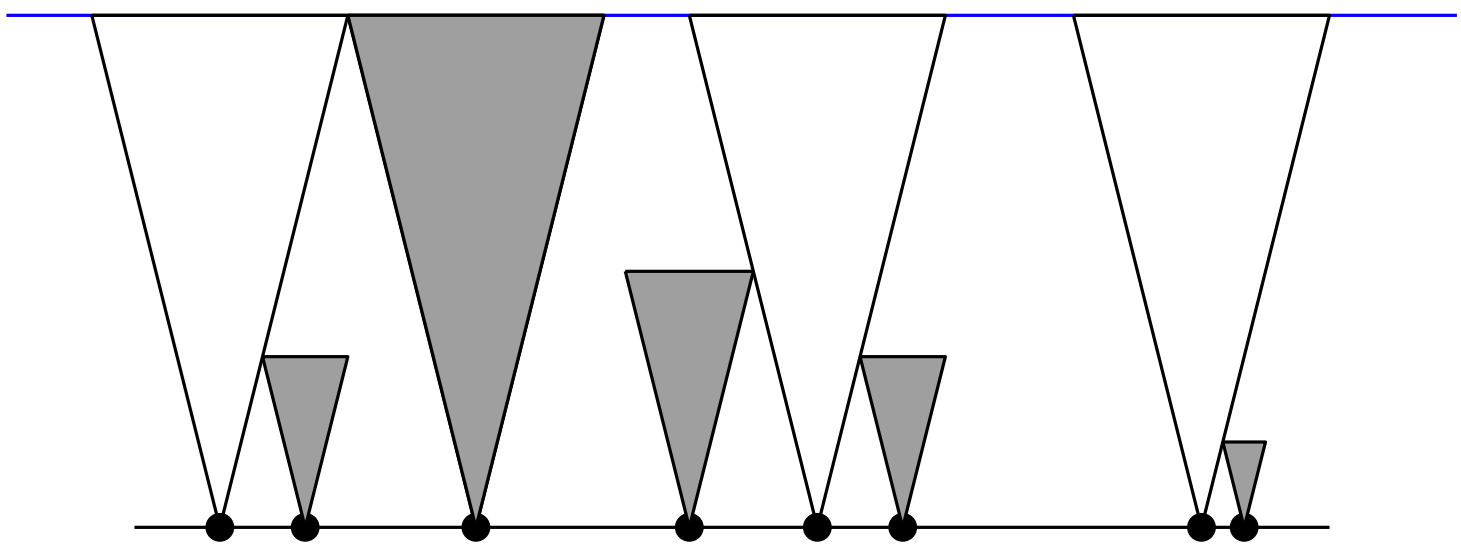
Growing Cones (M0)



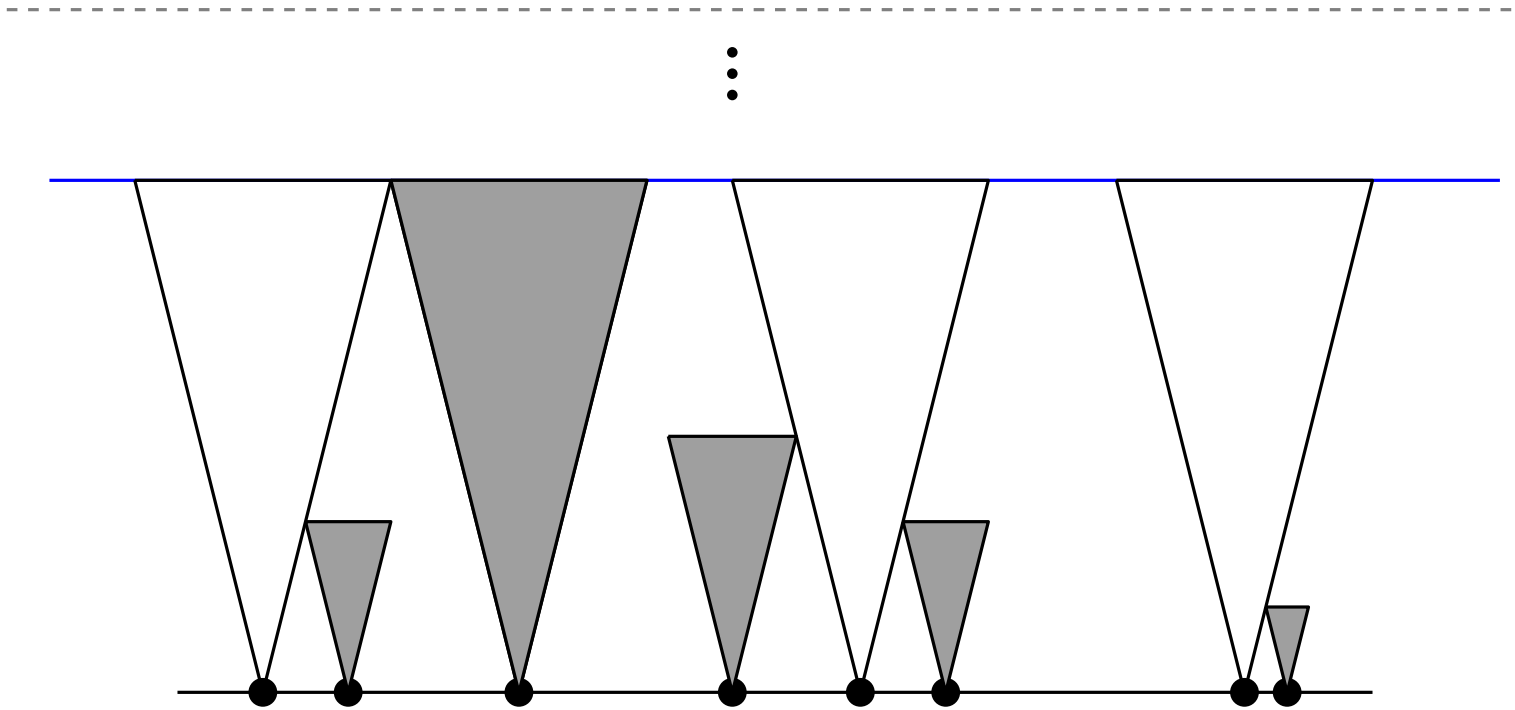
Growing Cones (M0)



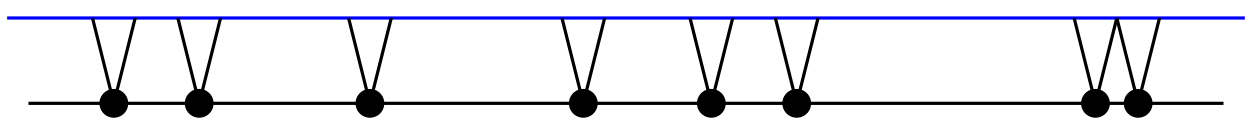
Growing Cones (M0)



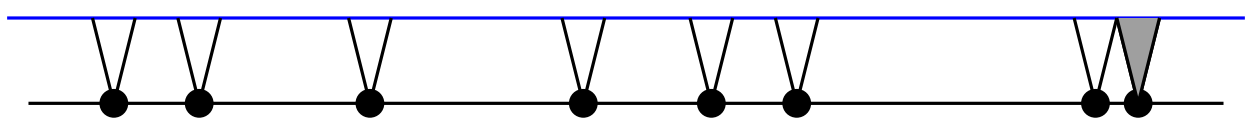
Growing Cones (M0)



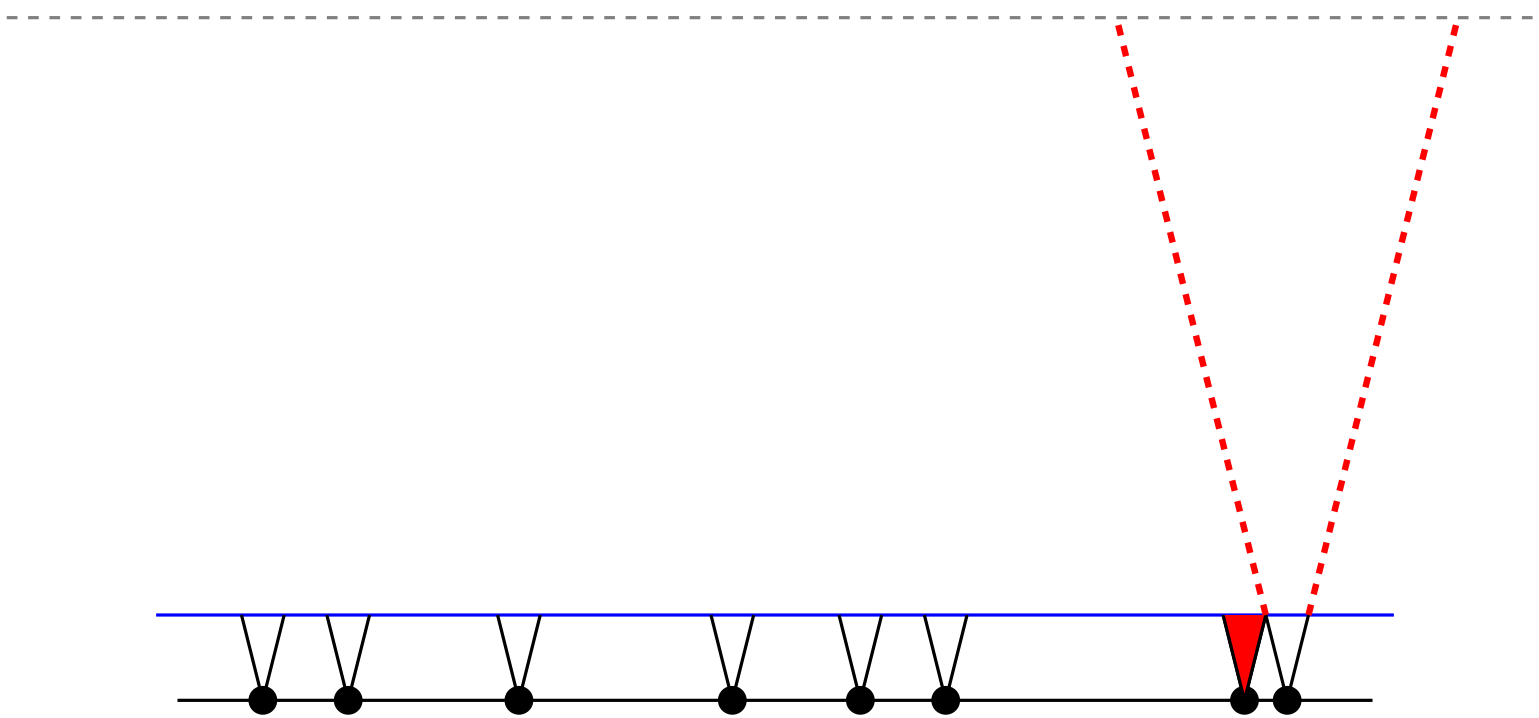
Growing Cones (M1)



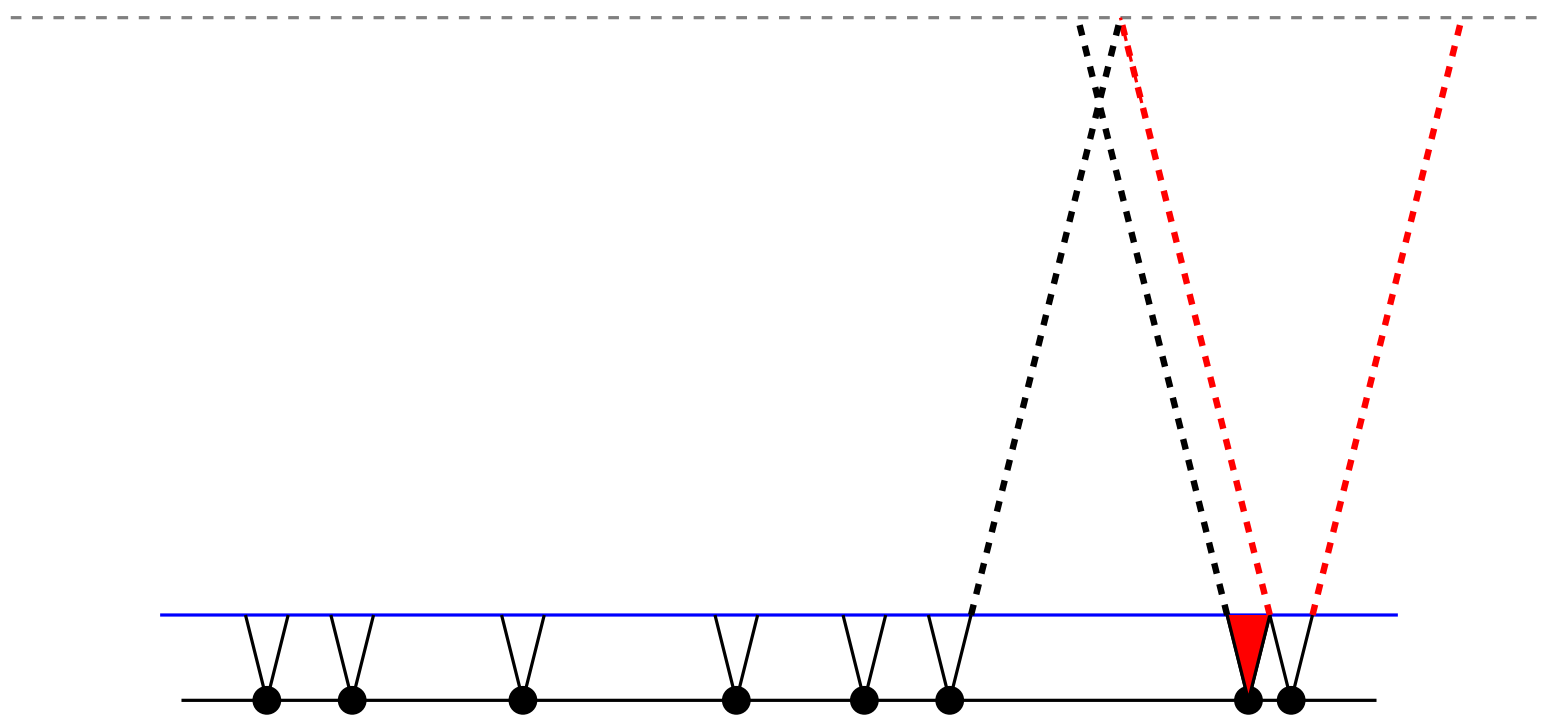
Growing Cones (M1)



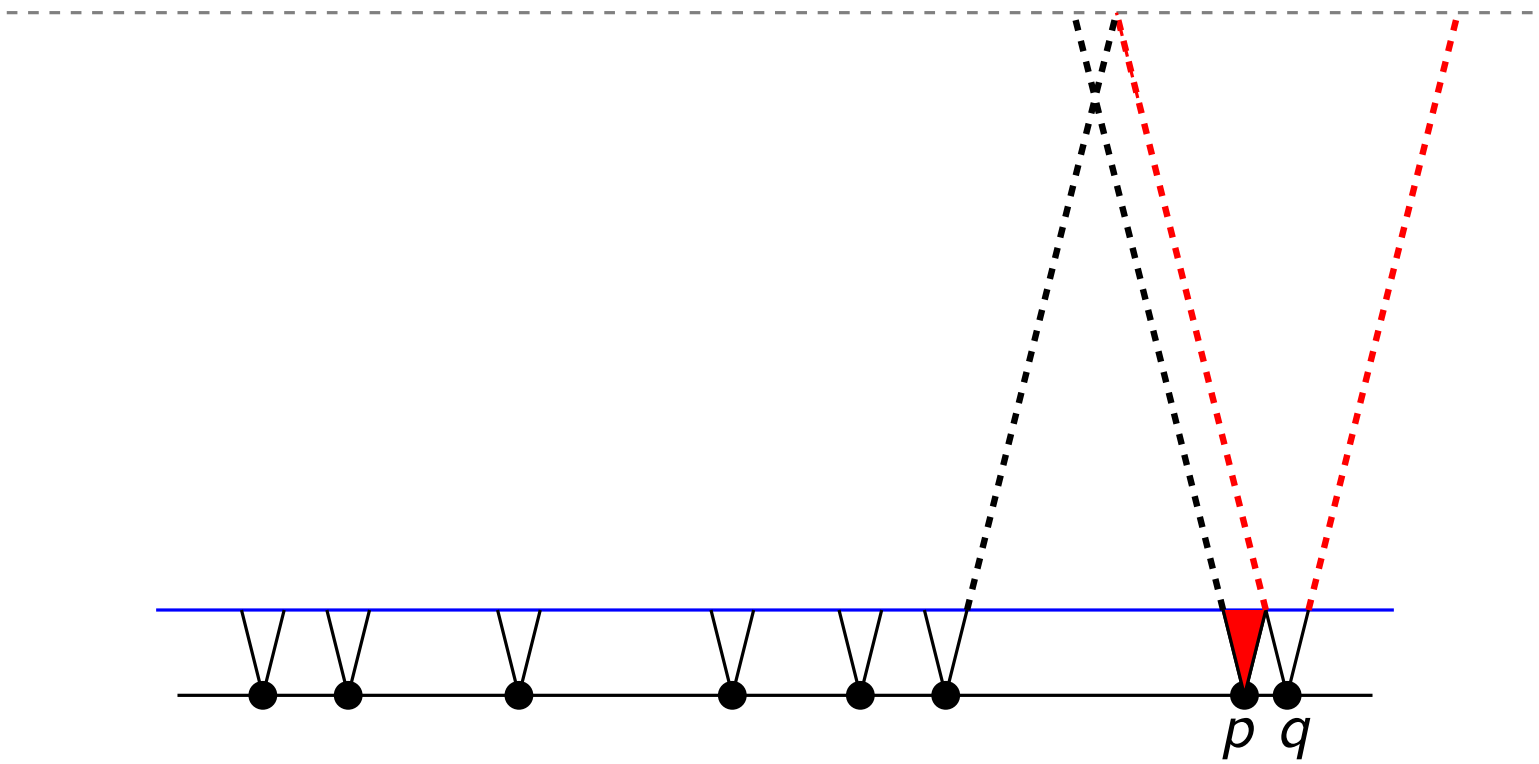
Growing Cones (M1)



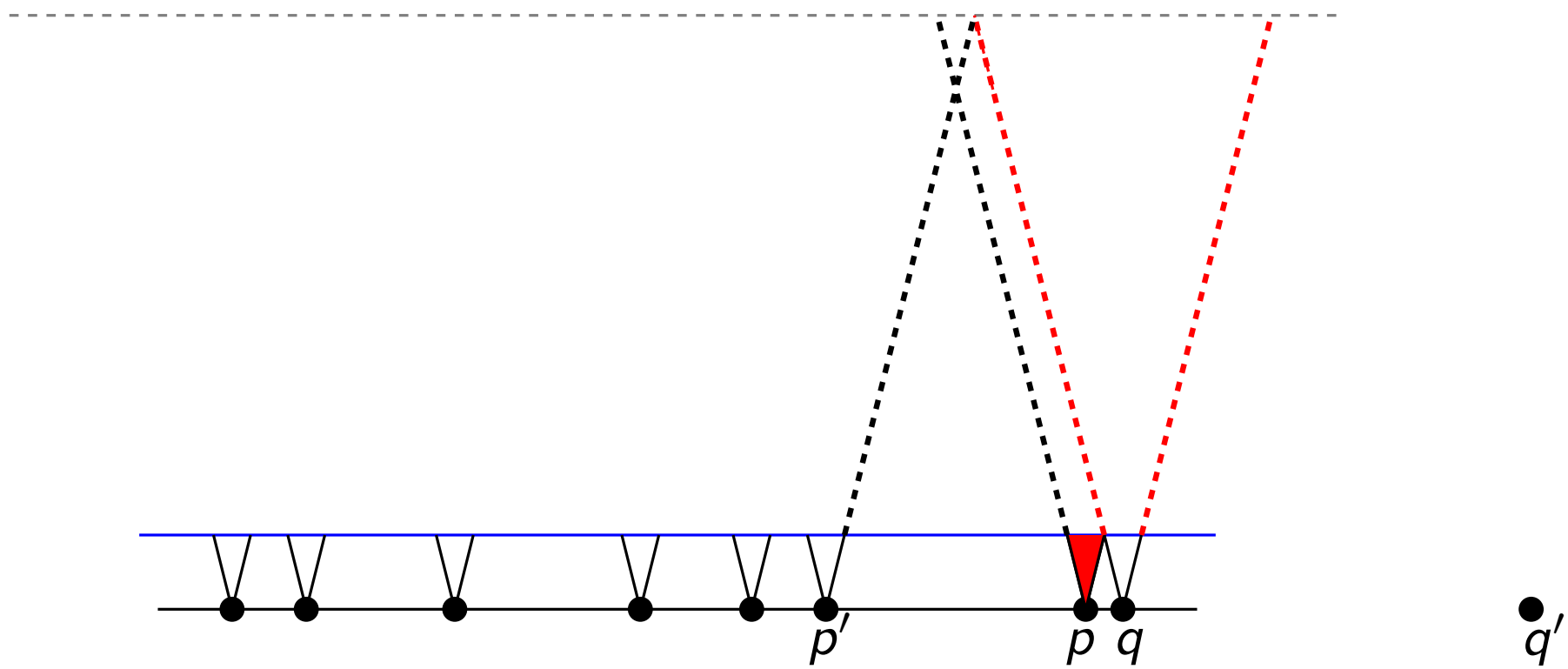
Growing Cones (M1)



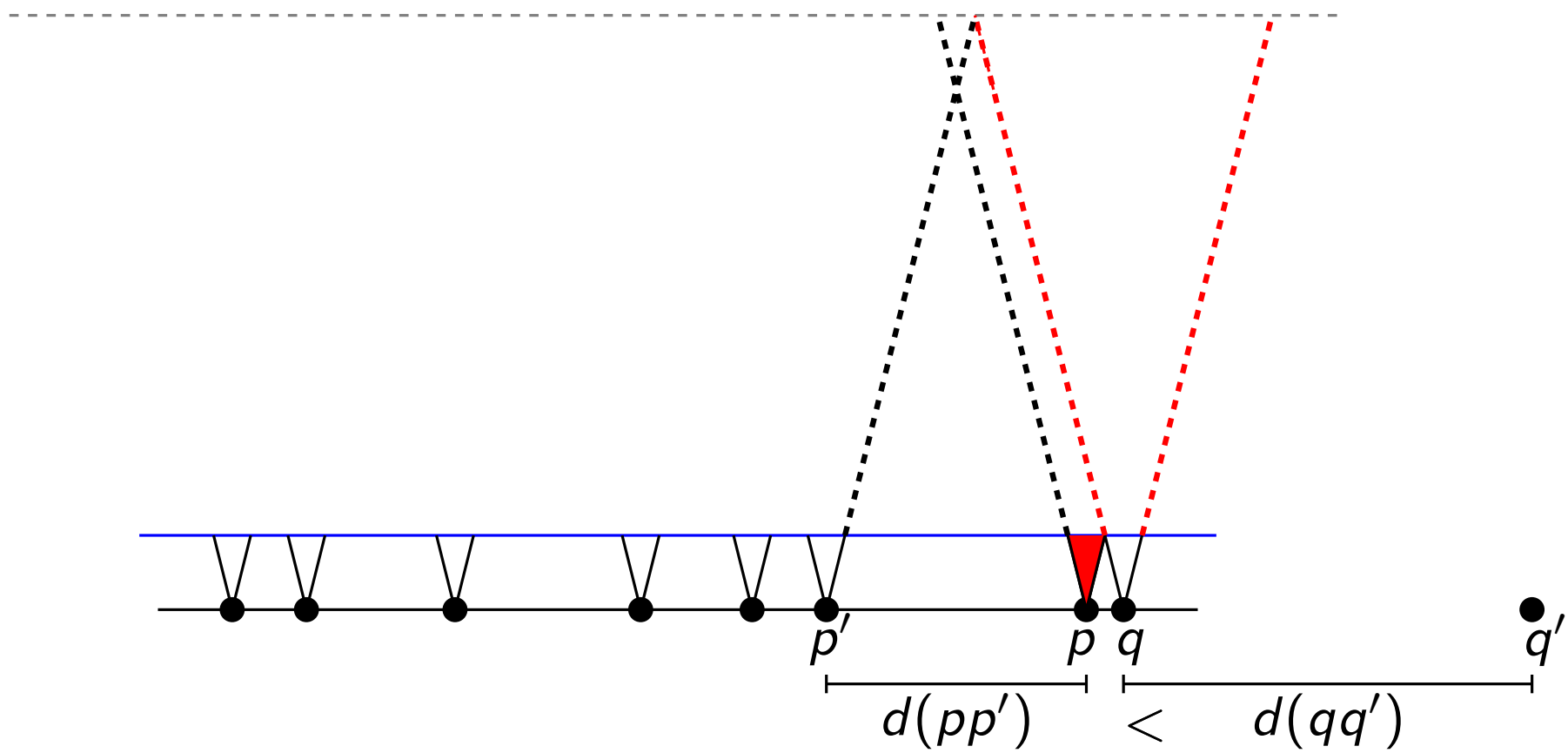
Growing Cones (M1)



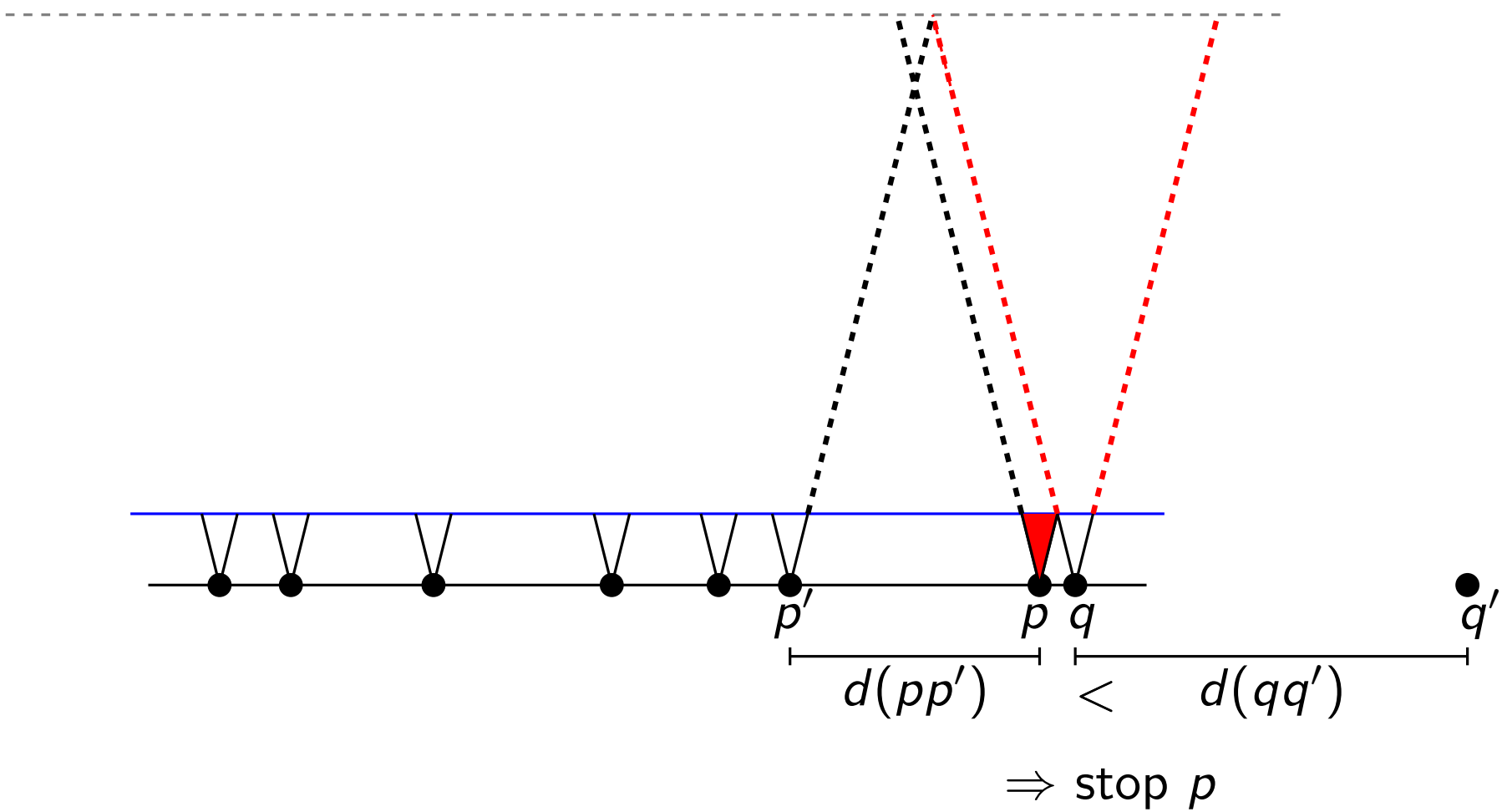
Growing Cones (M1)



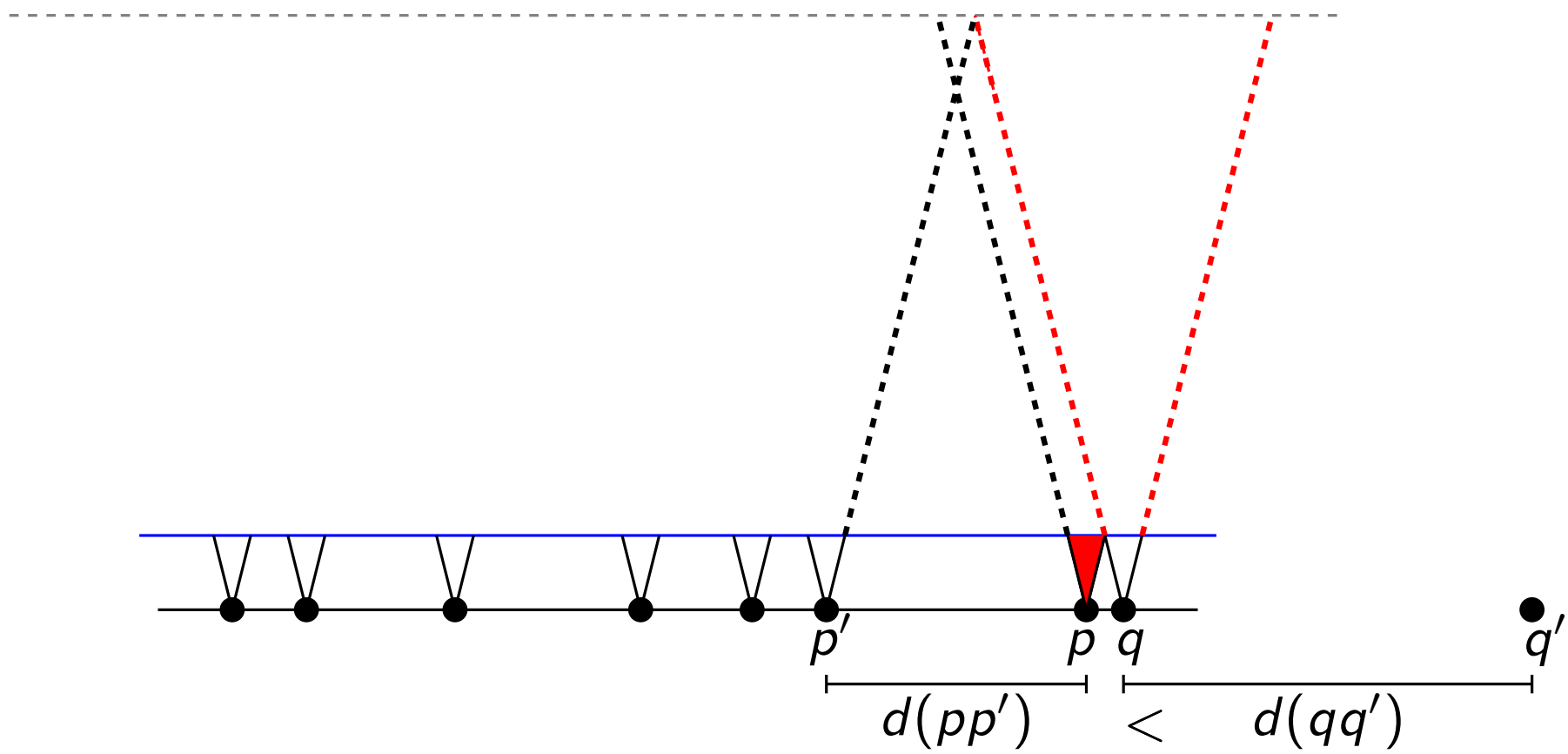
Growing Cones (M1)



Growing Cones (M1)



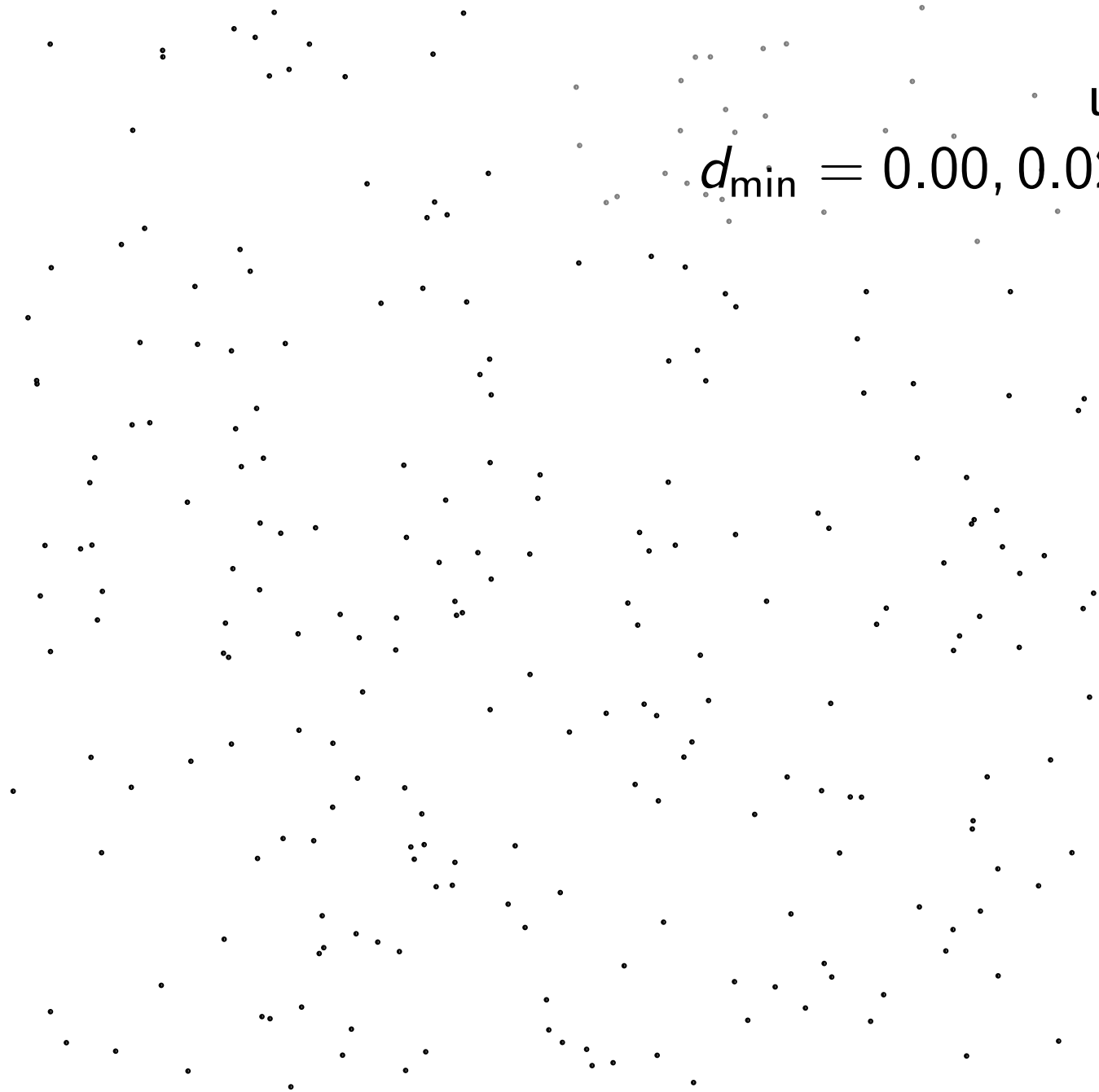
Growing Cones (M1)



means: Delaunay triangulation

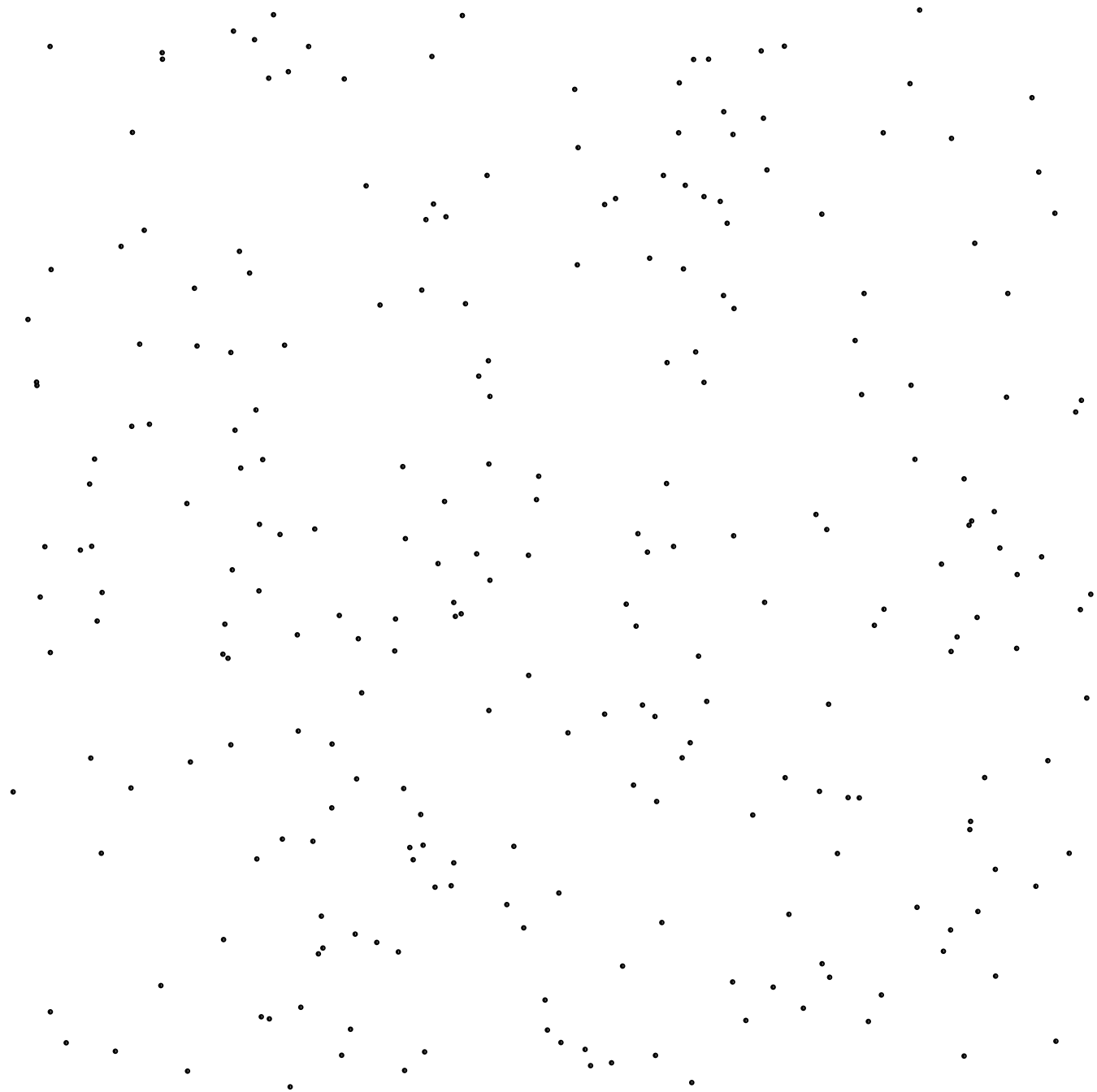
$\Rightarrow$ stop p

Example

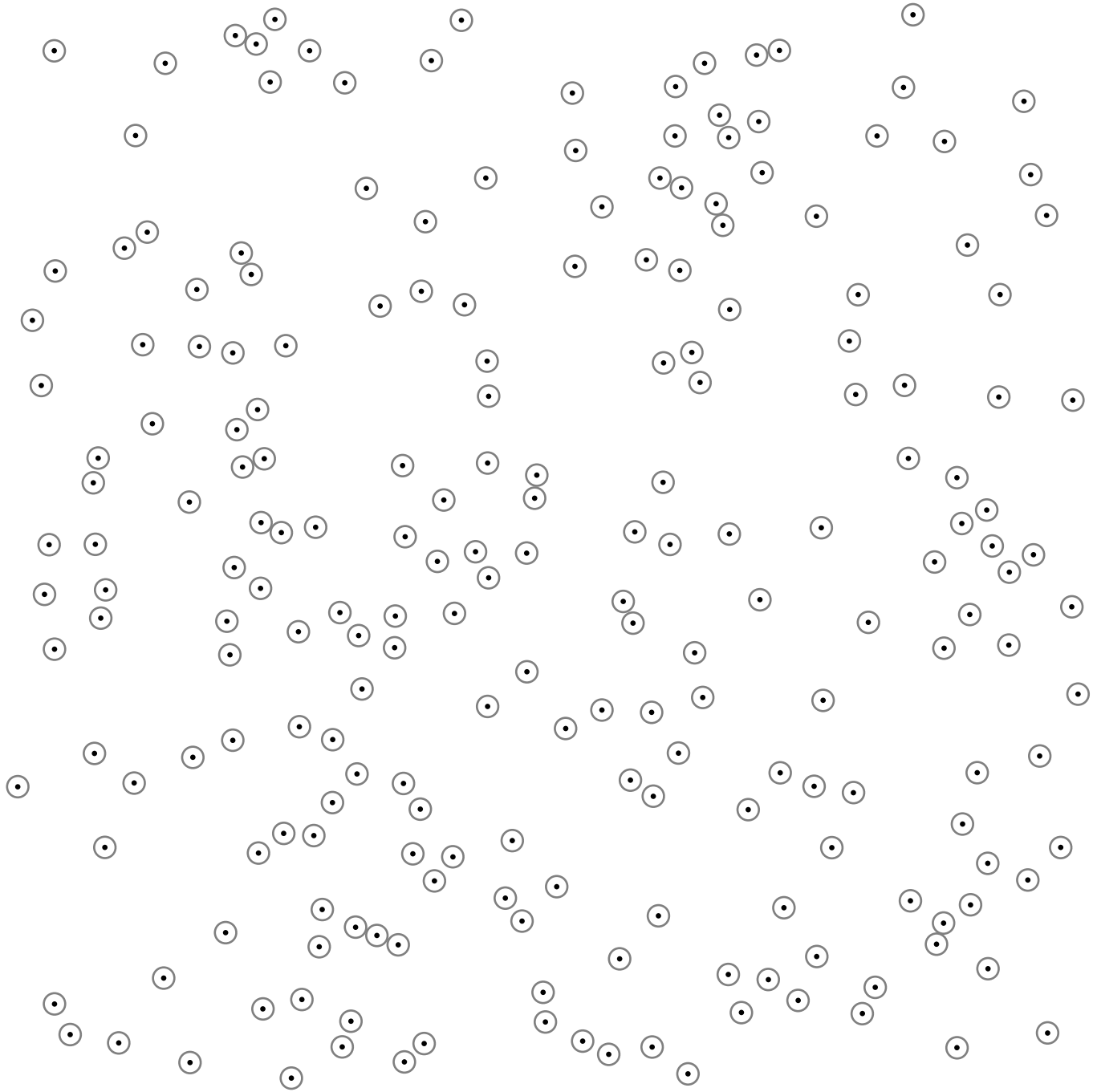


249 points
unit square
$d_{\text{min}} = 0.00, 0.02, \ldots, 0.40$

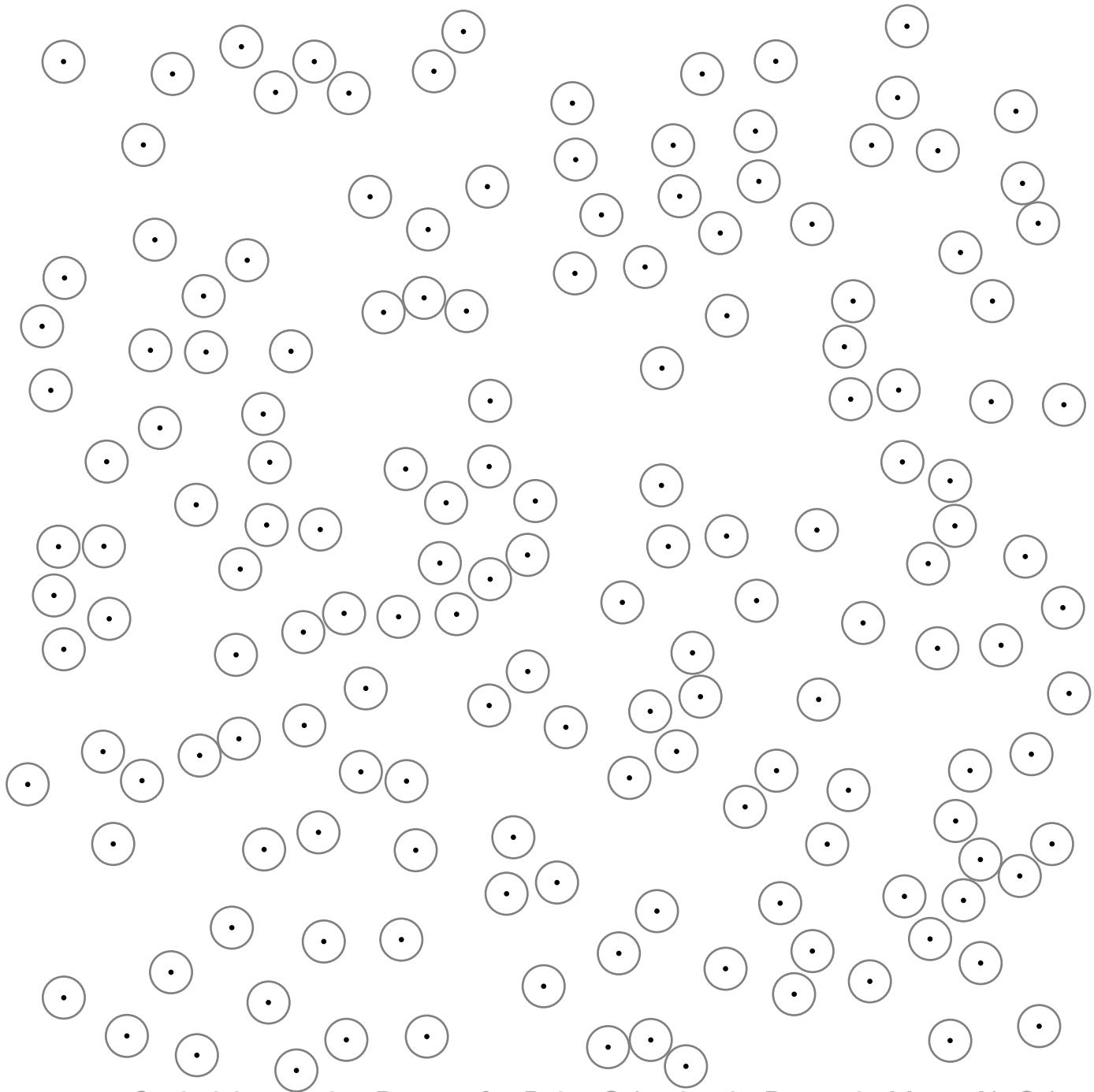
Example



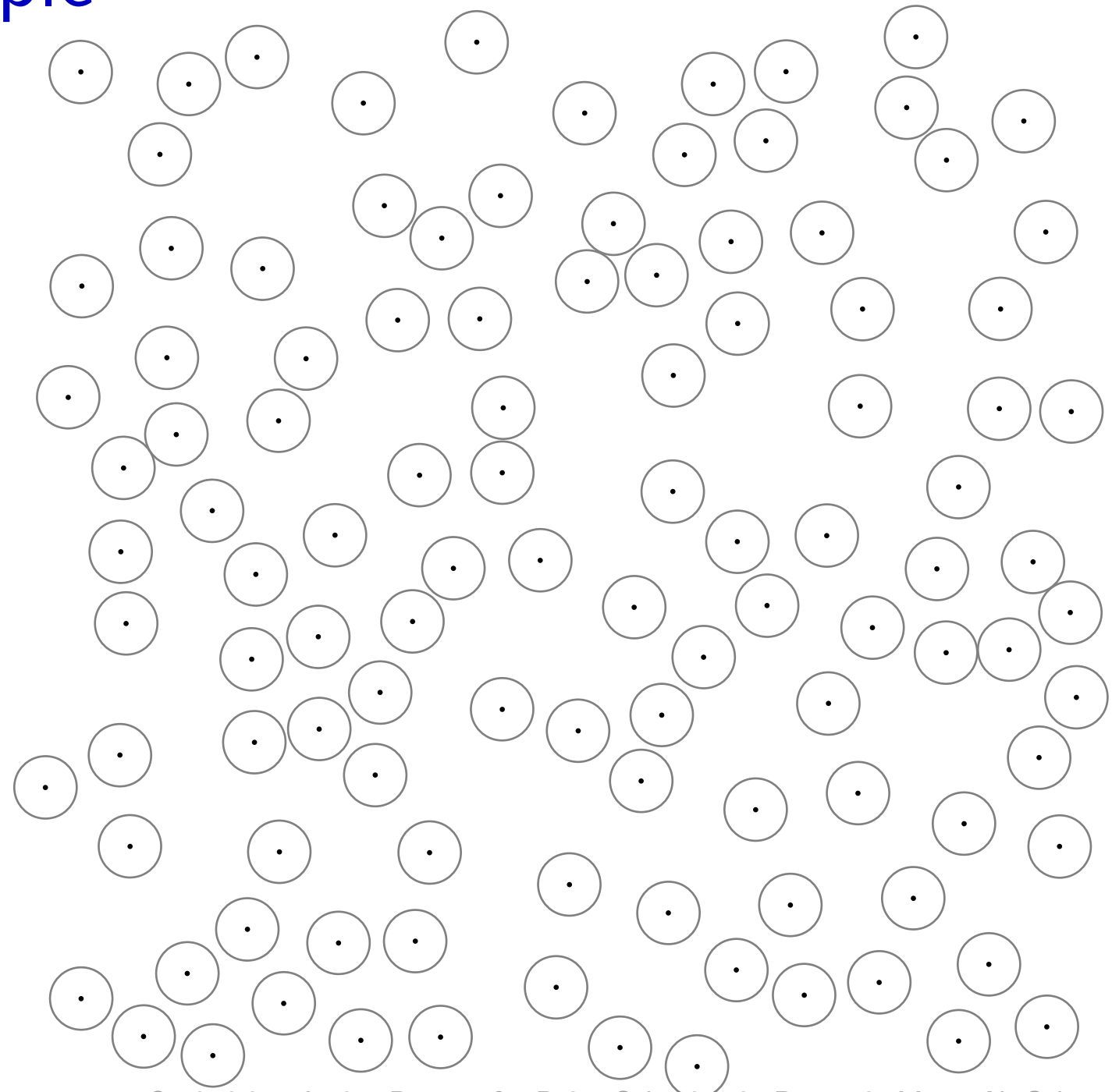
Example



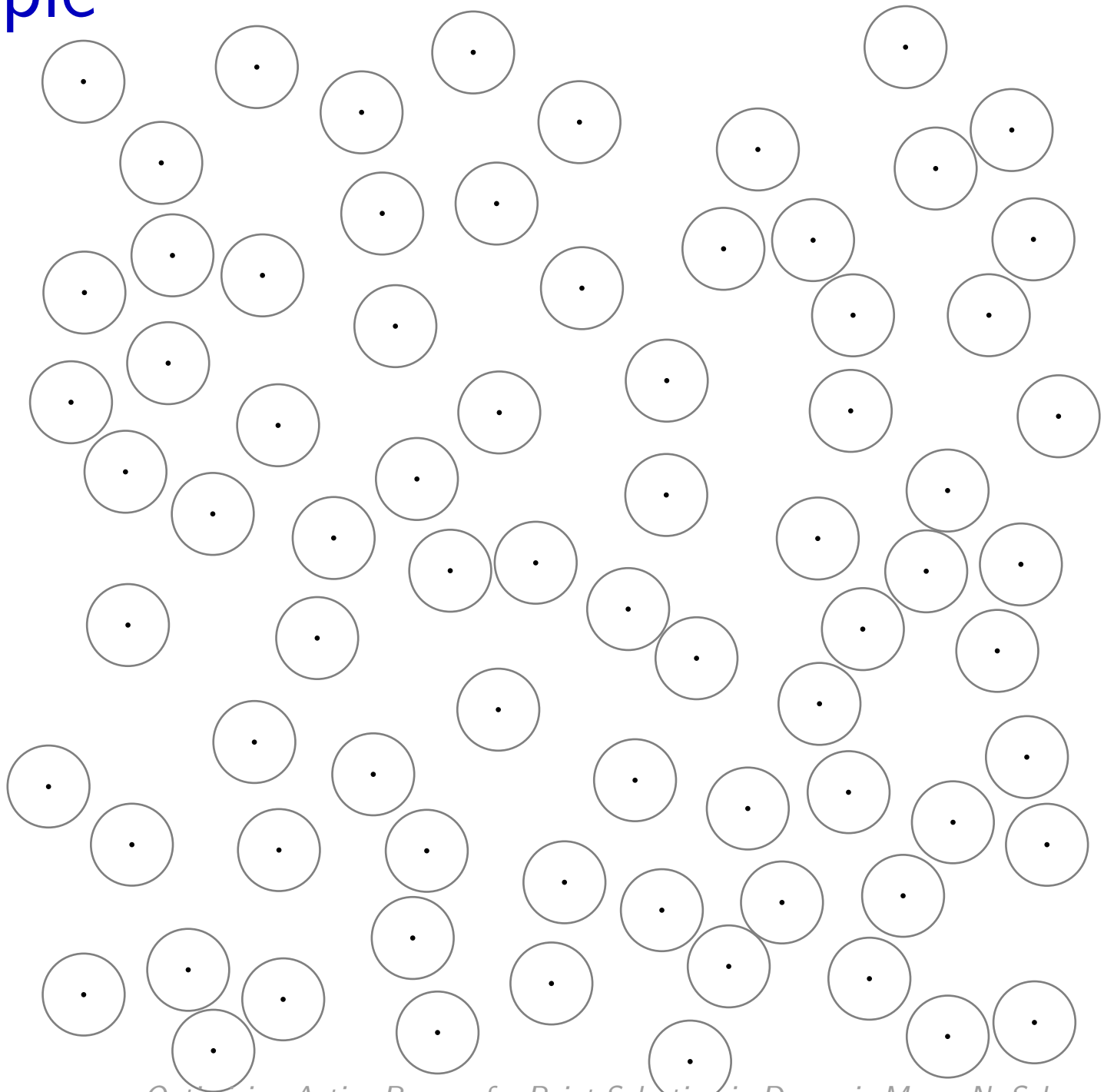
Example



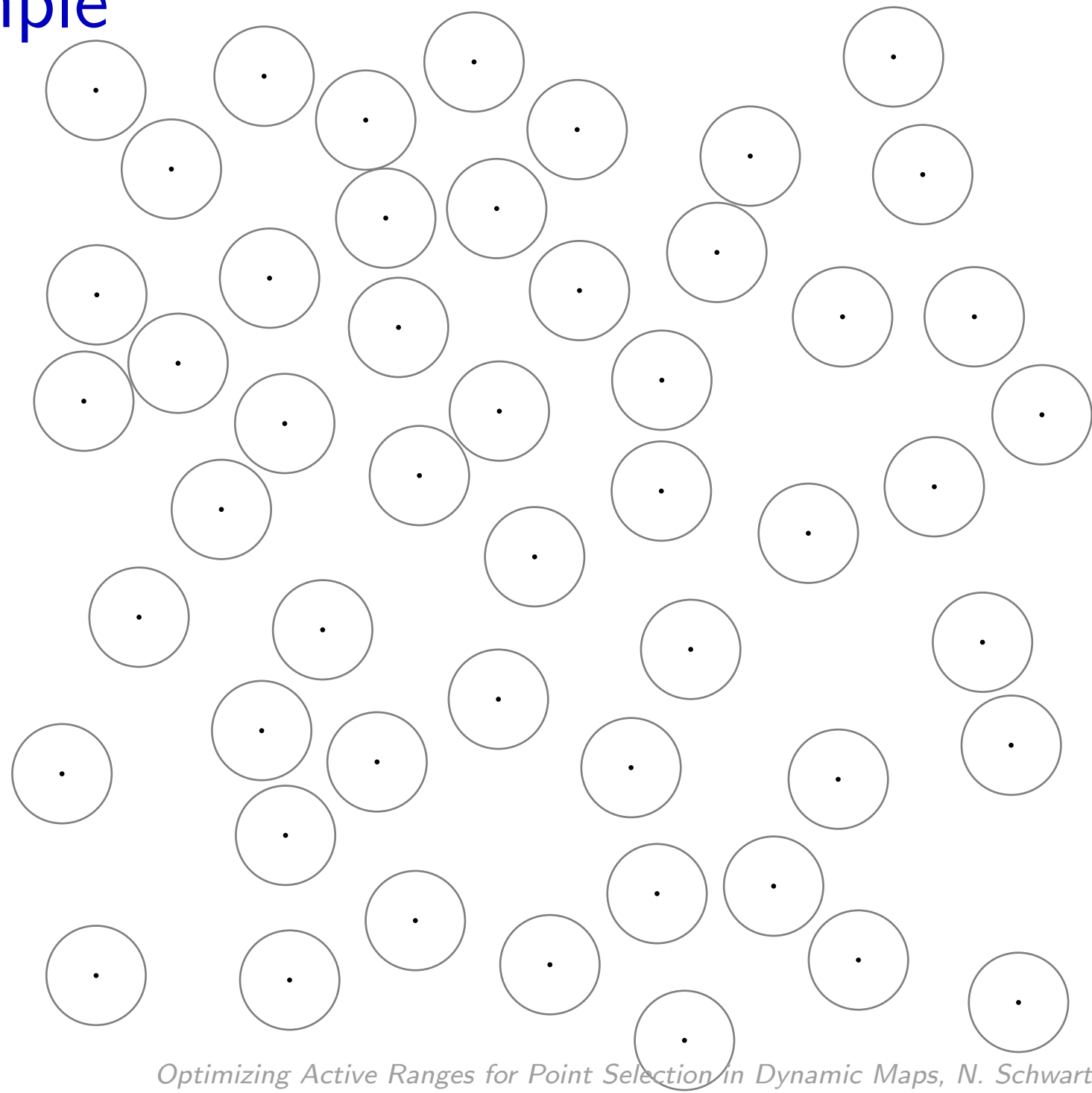
Example



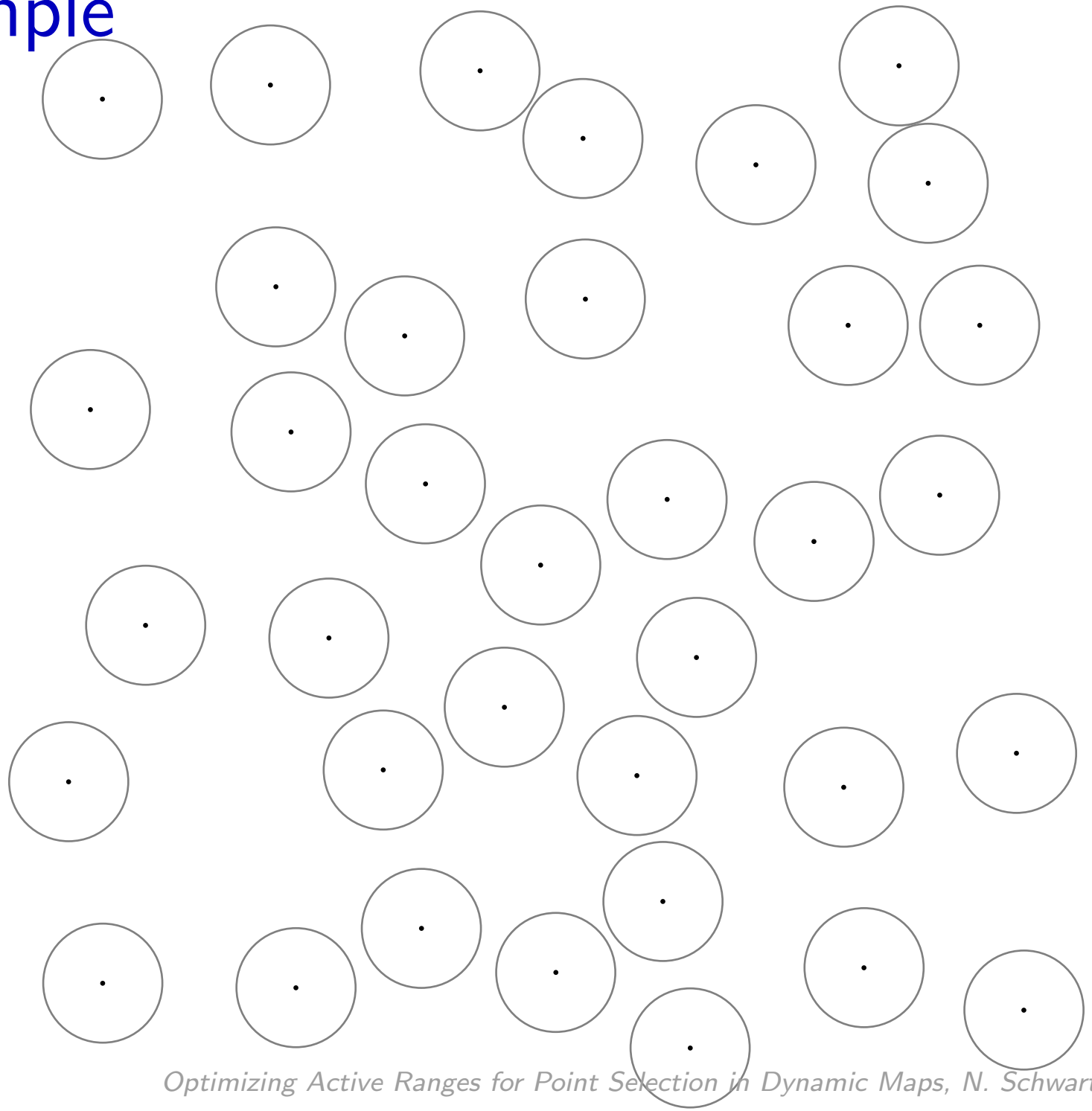
Example



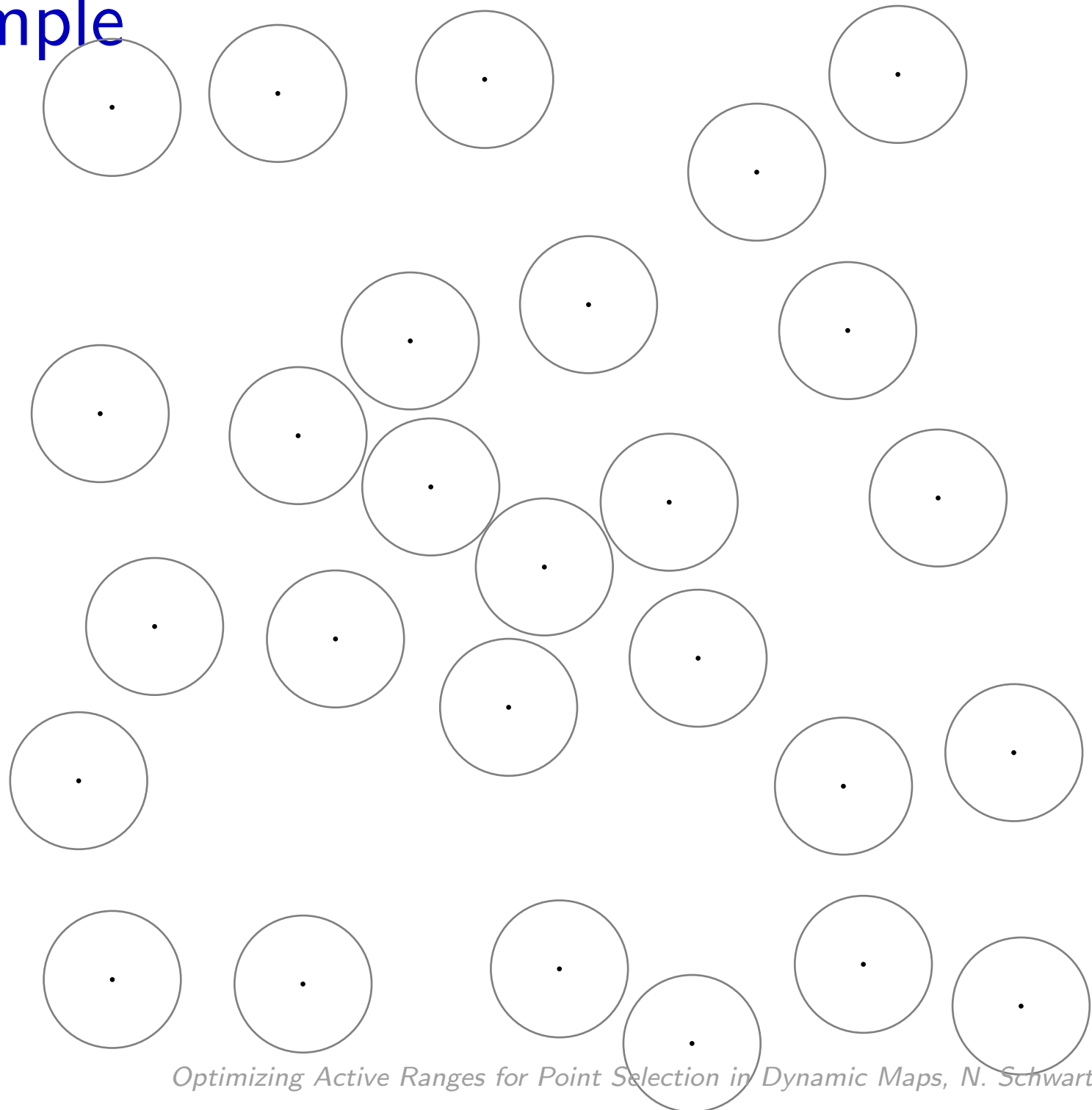
Example



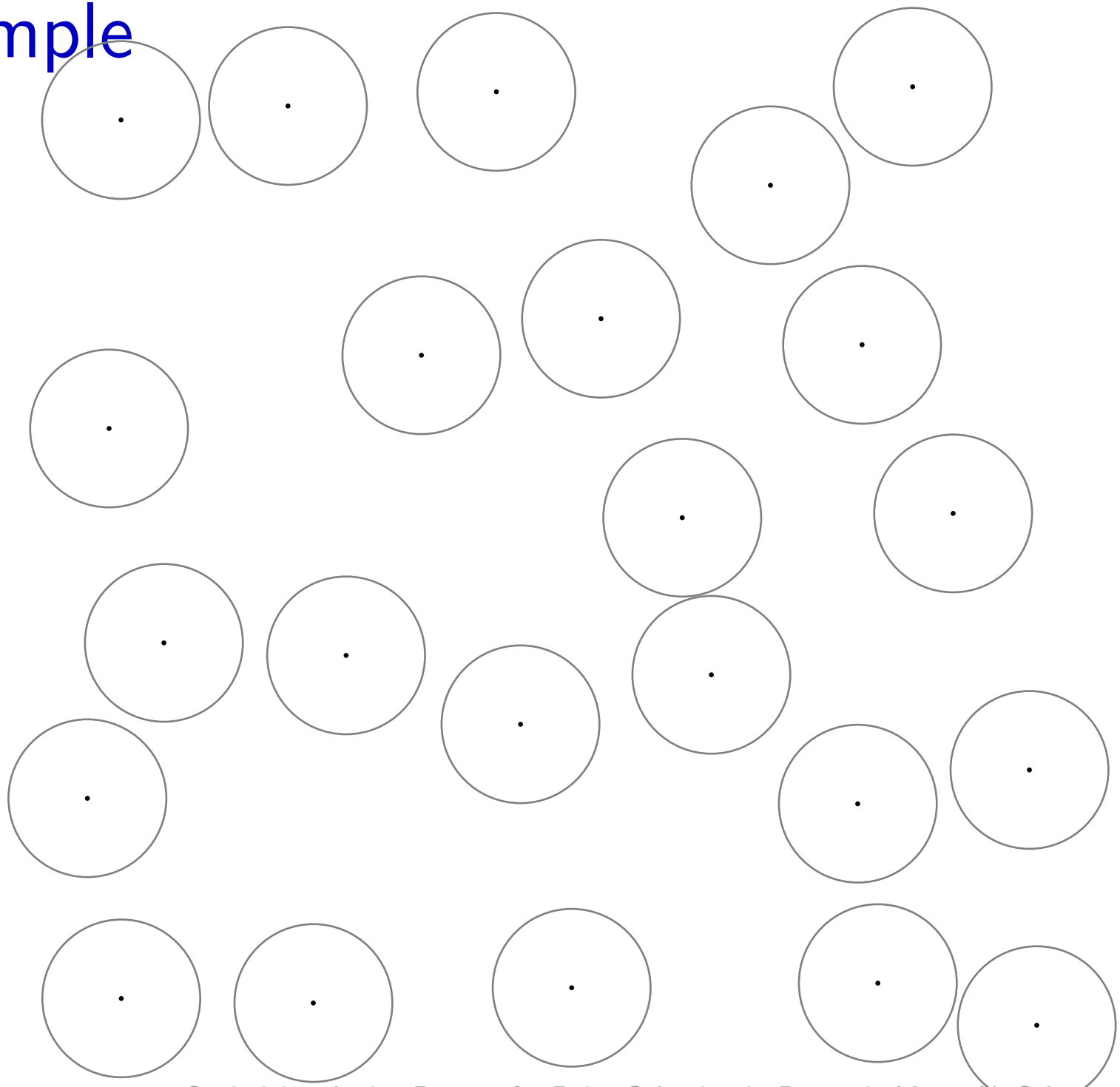
Example



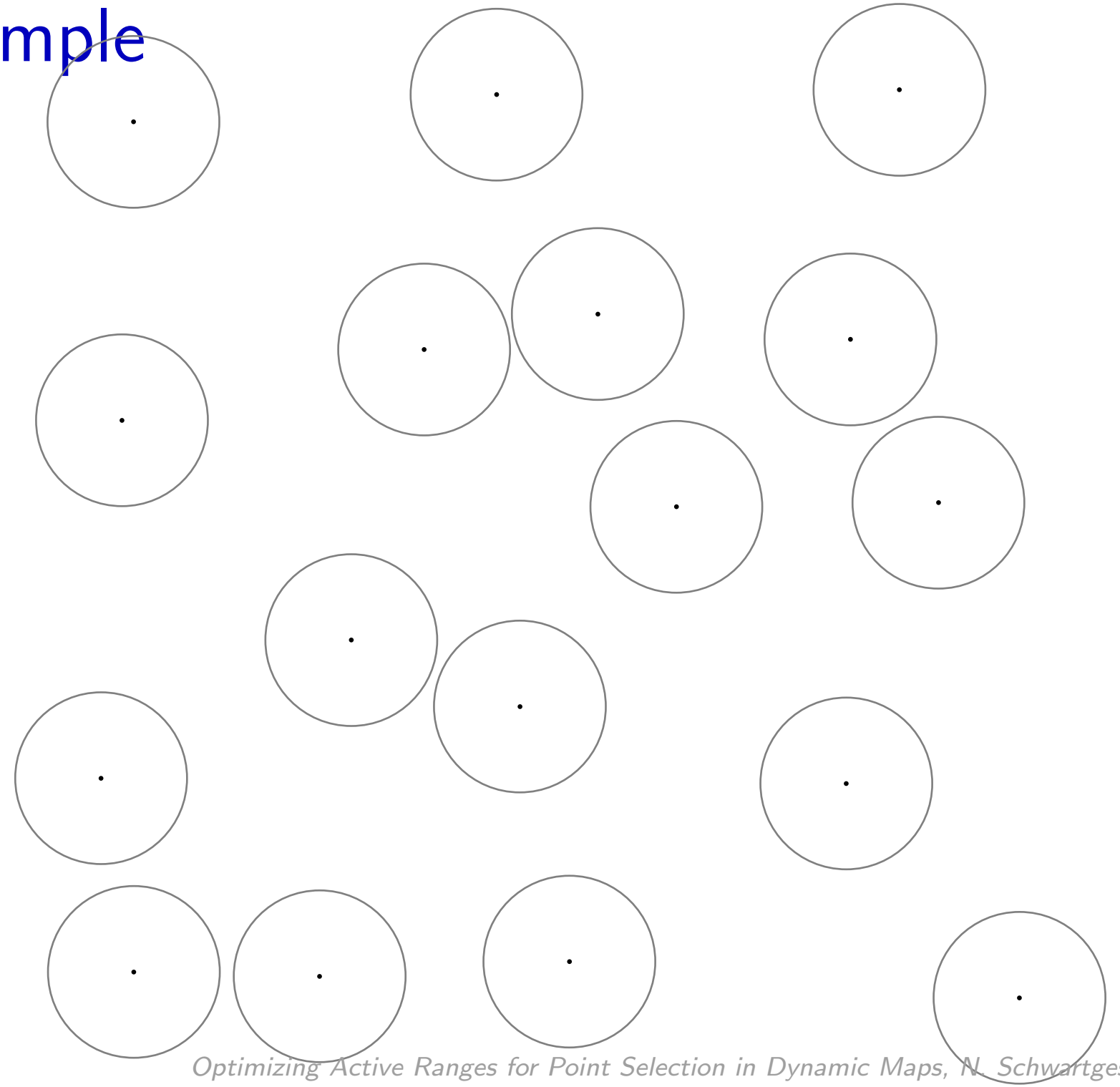
Example



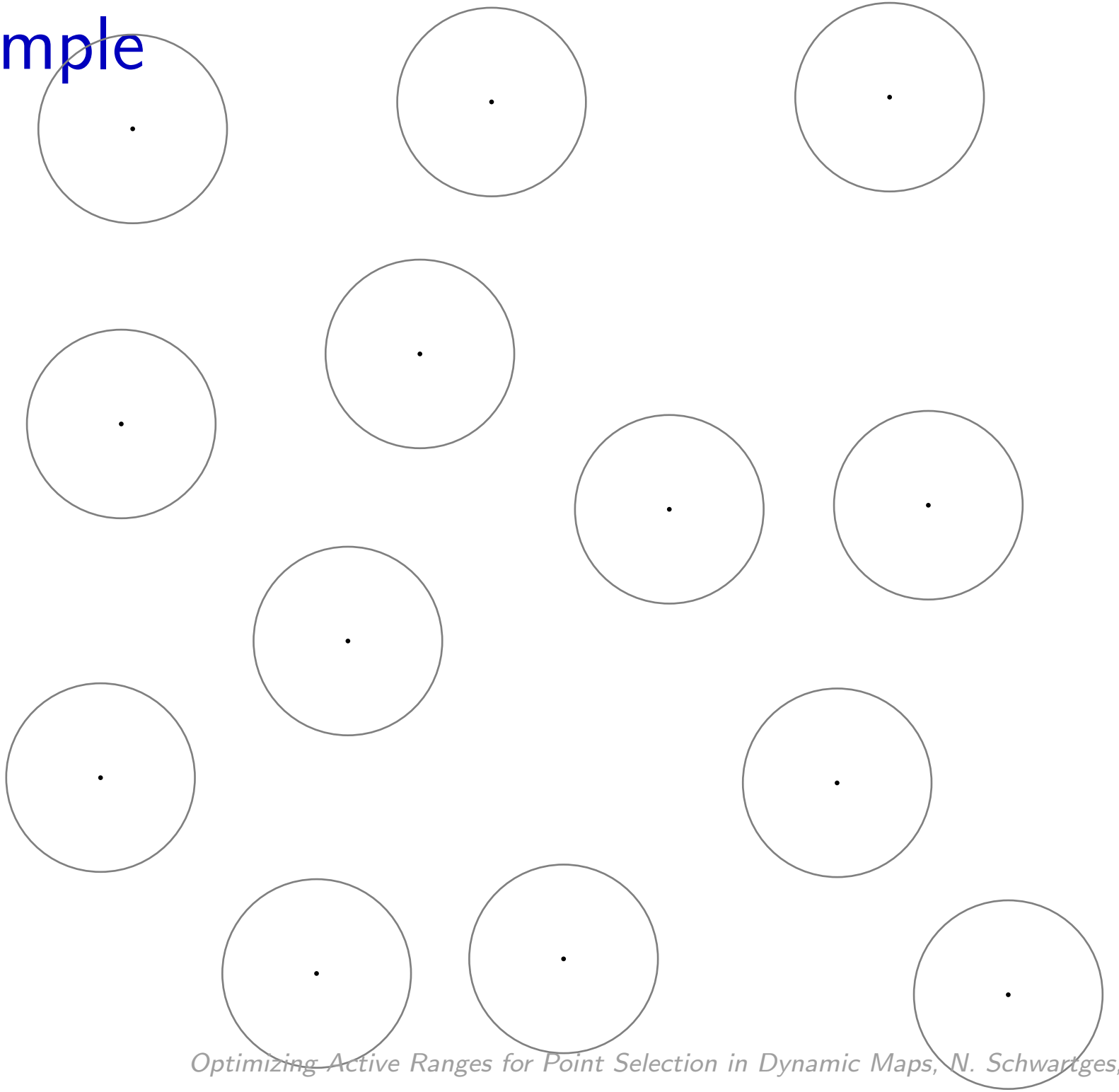
Example



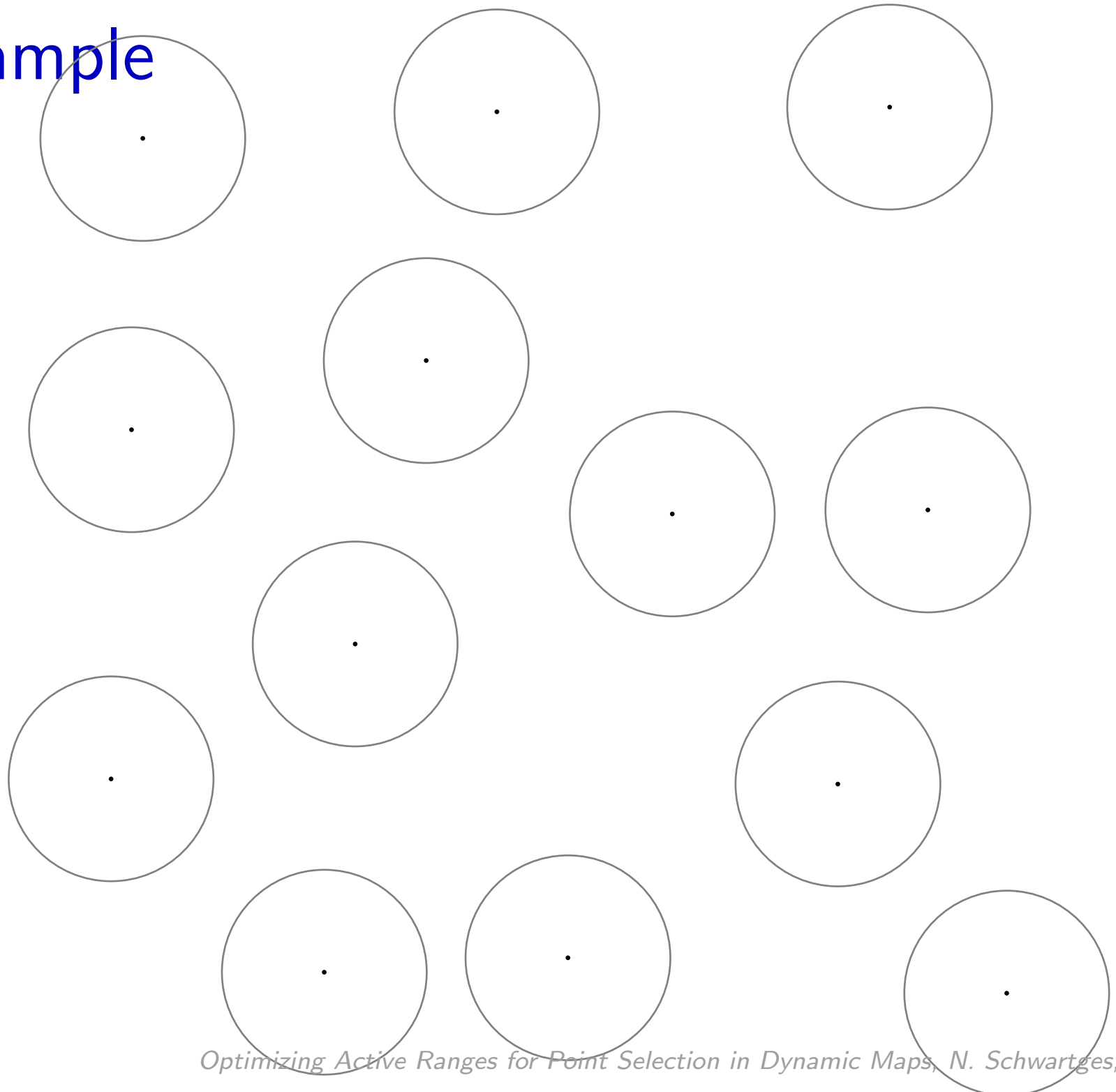
Example



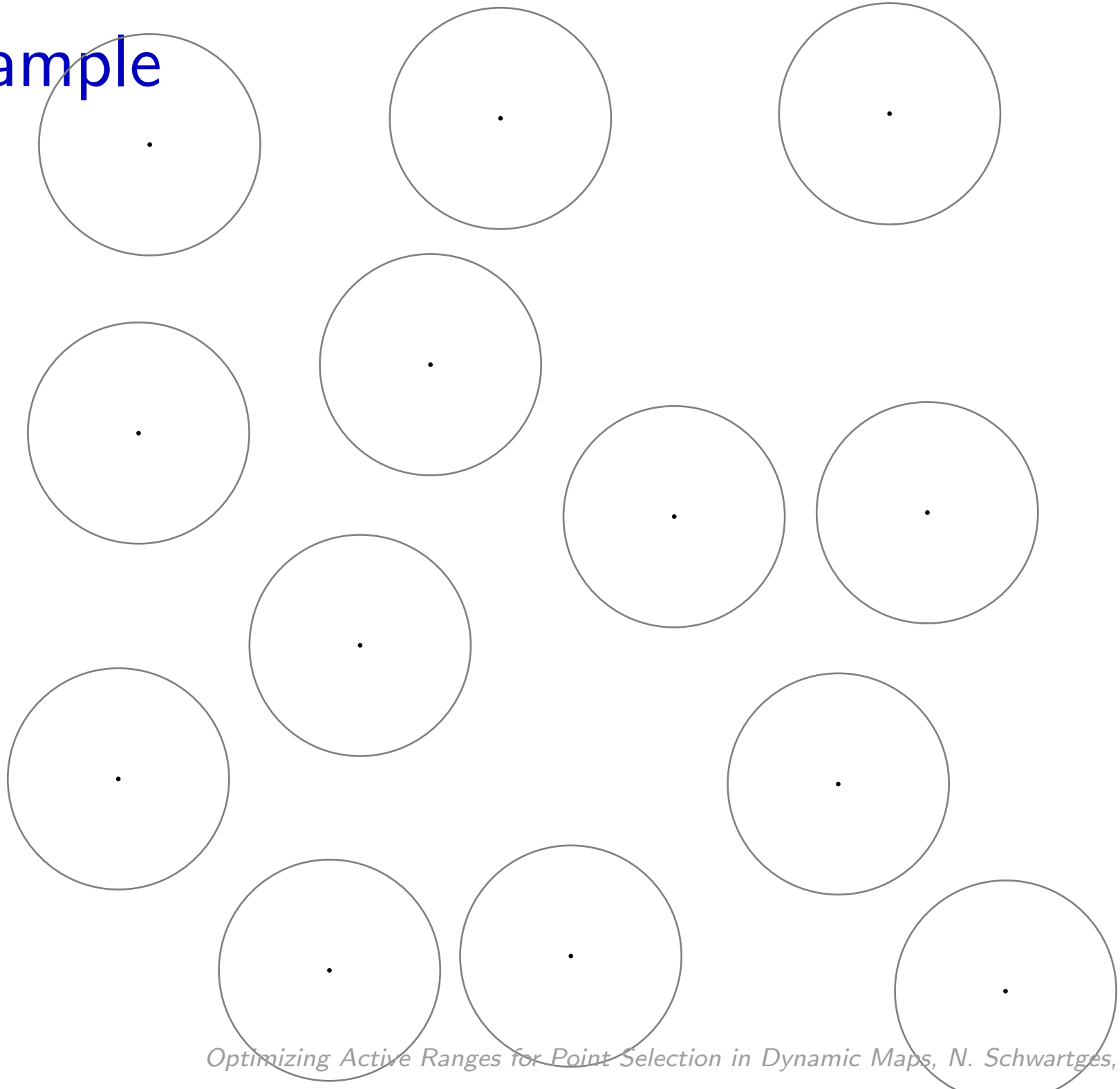
Example



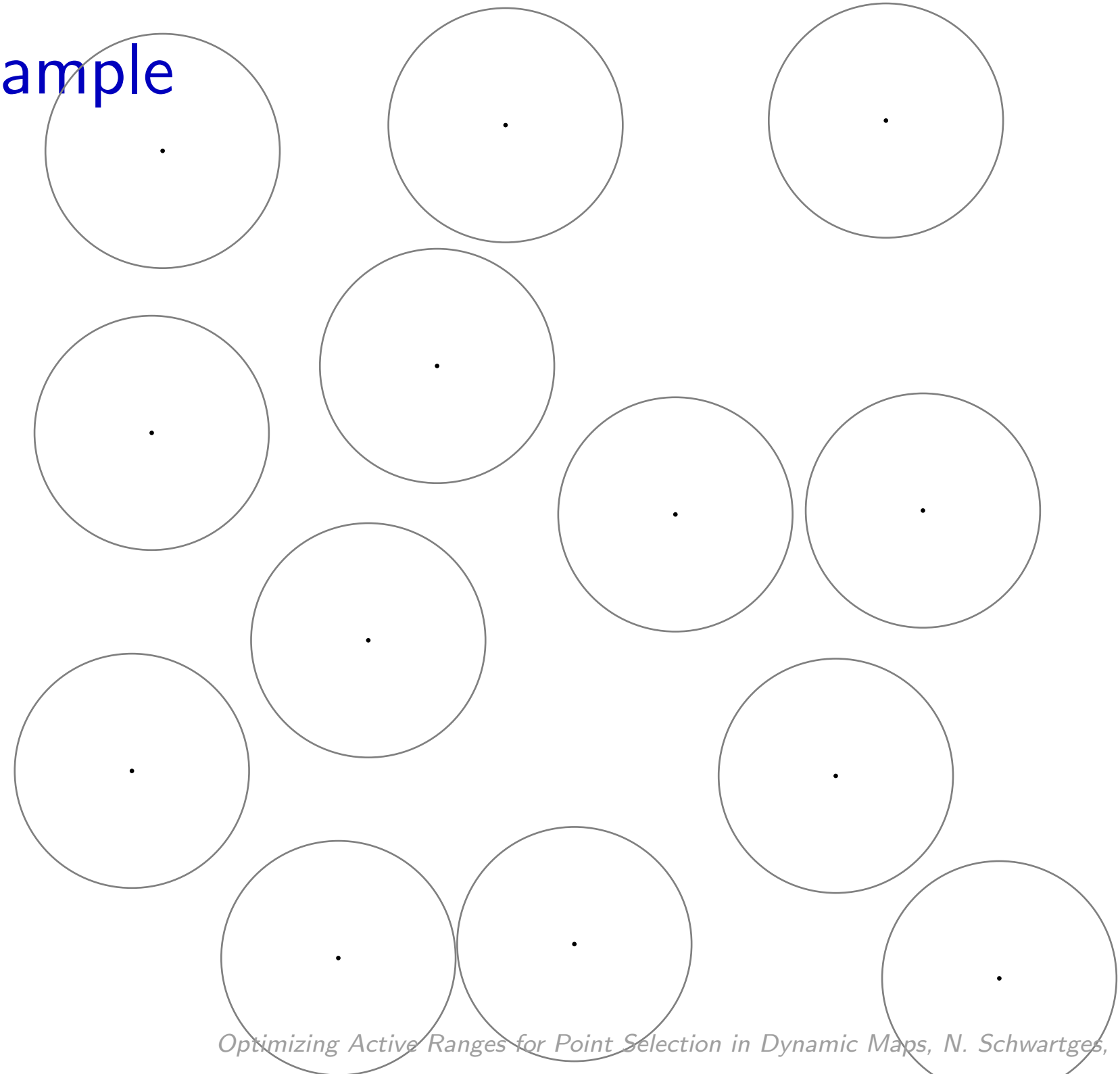
Example



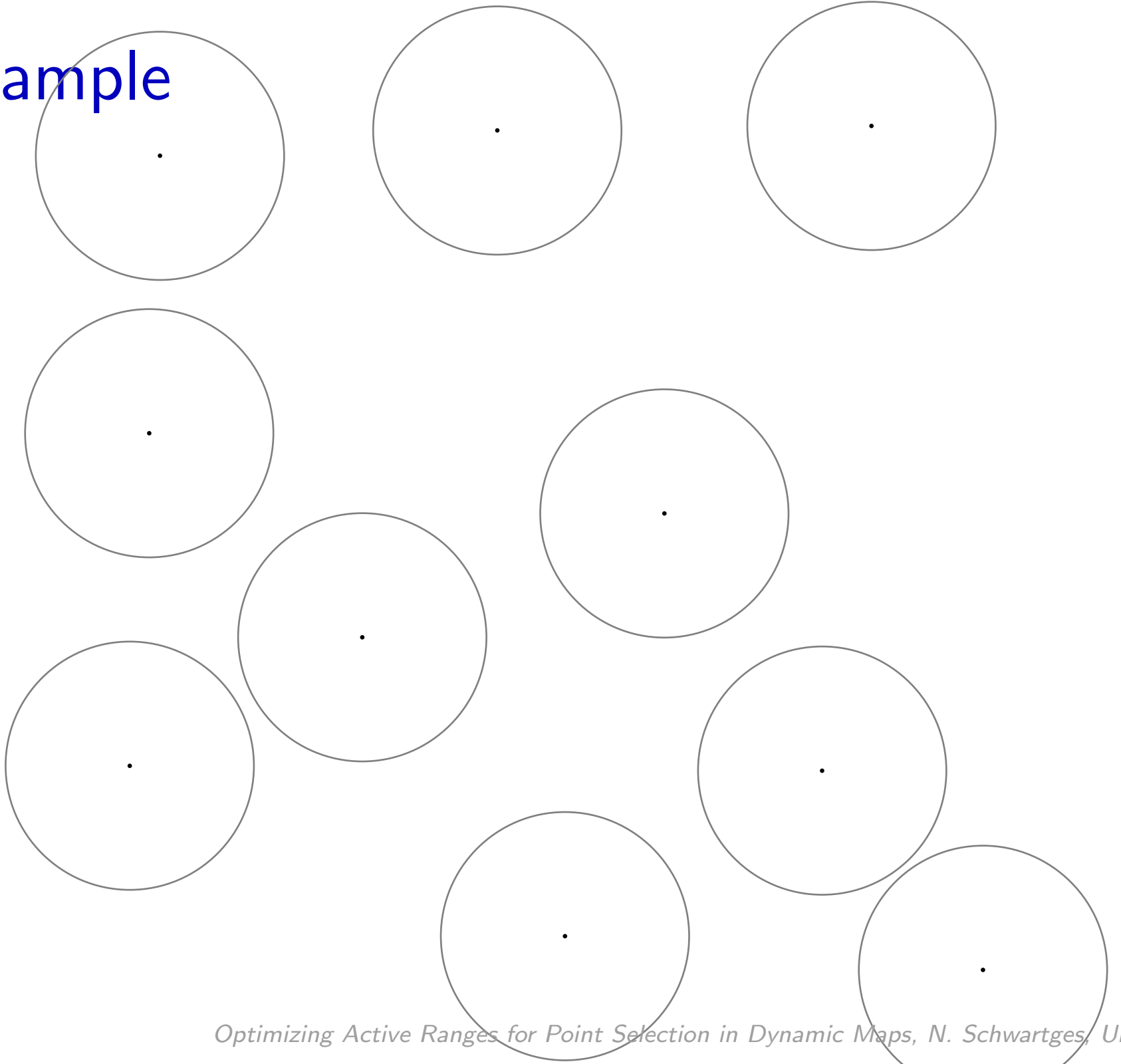
Example



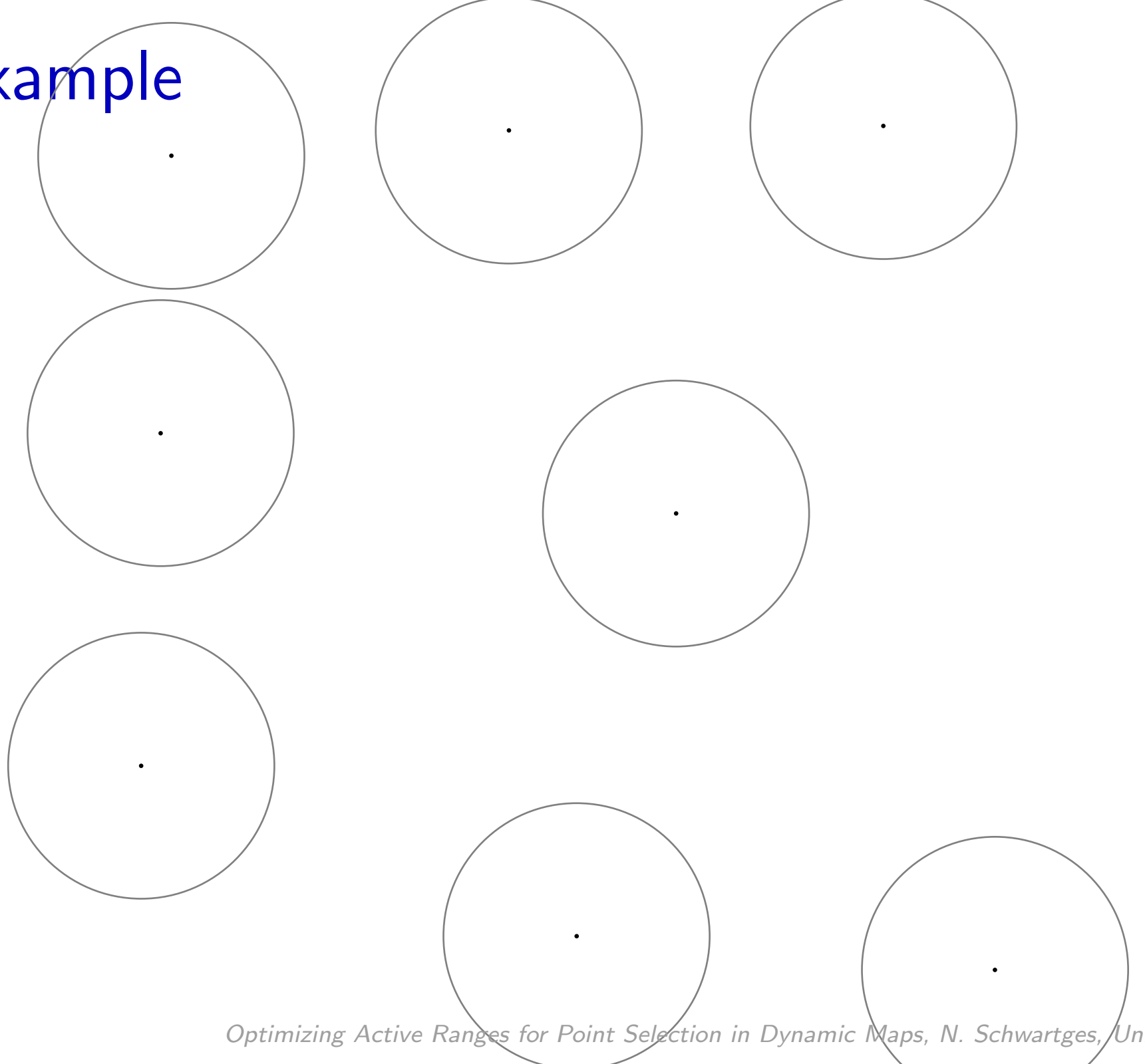
Example



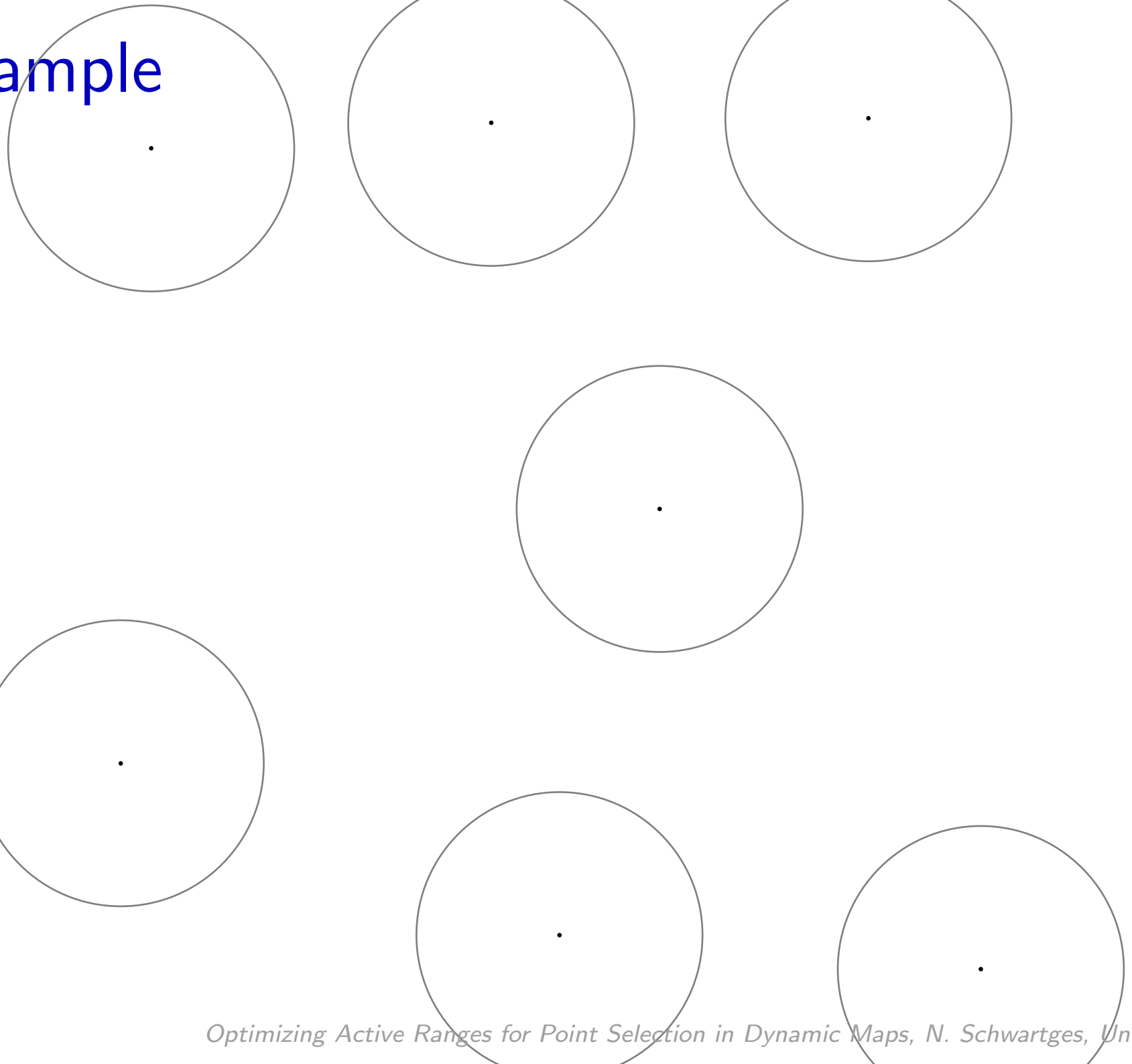
Example



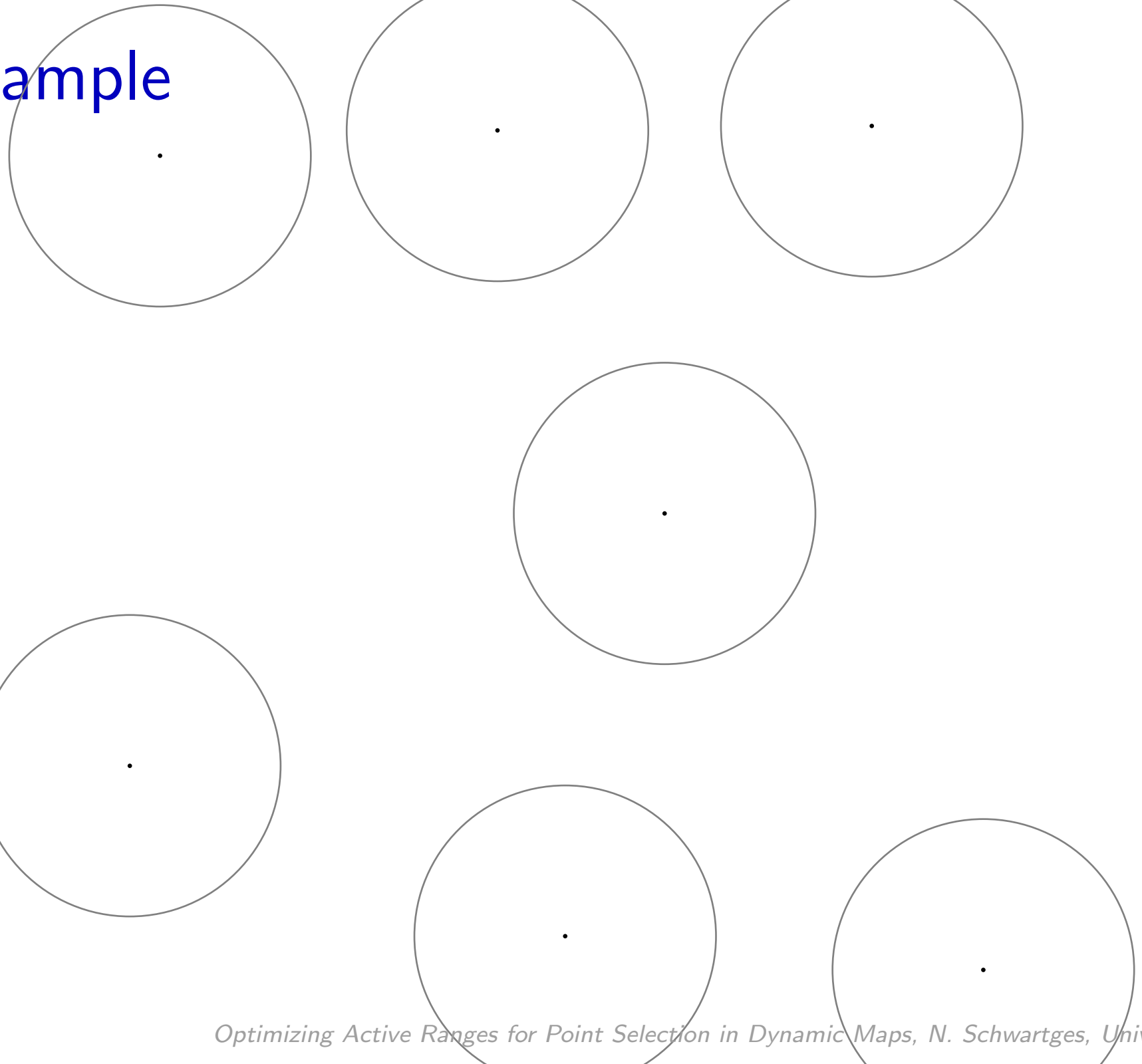
Example



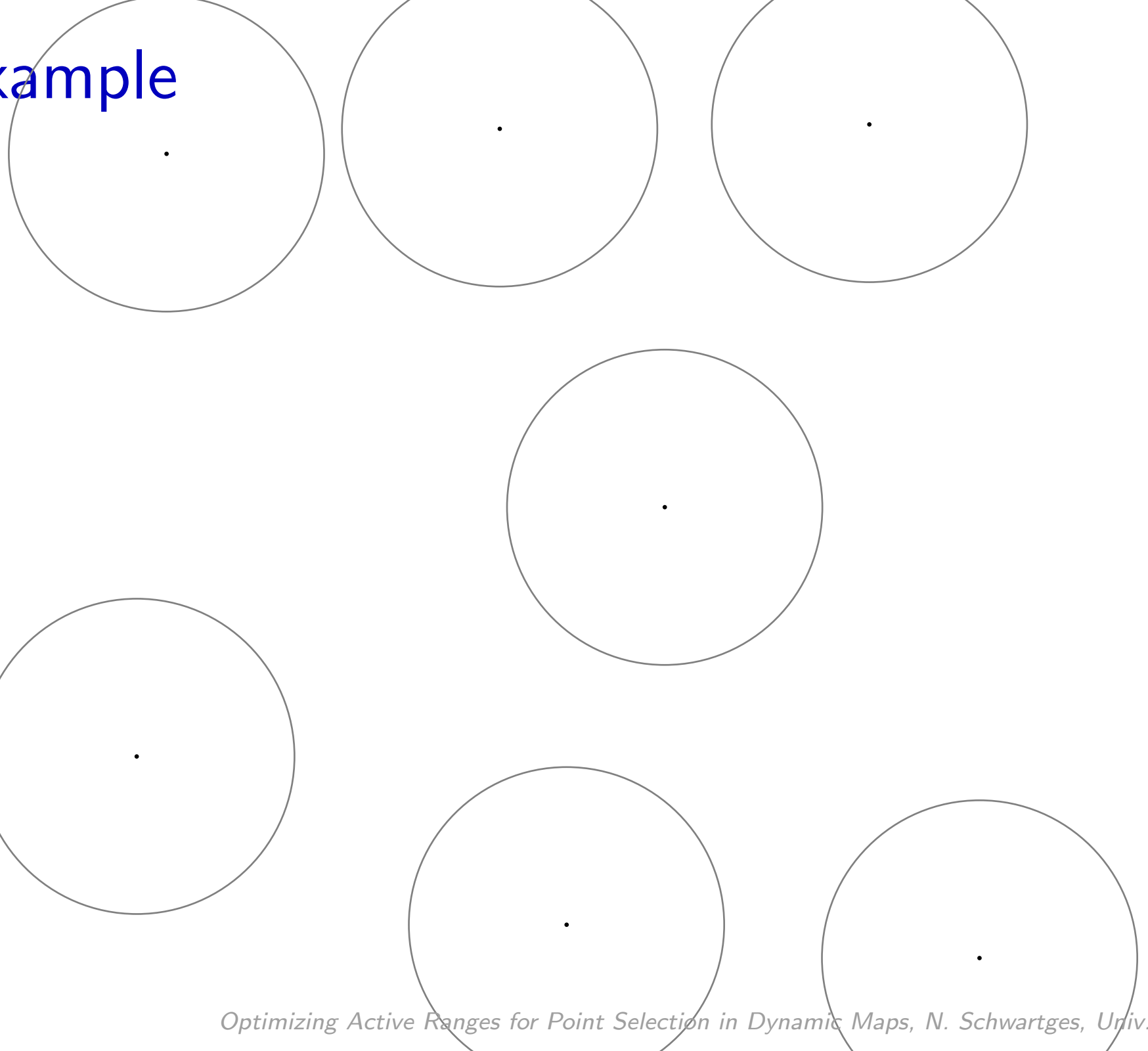
Example



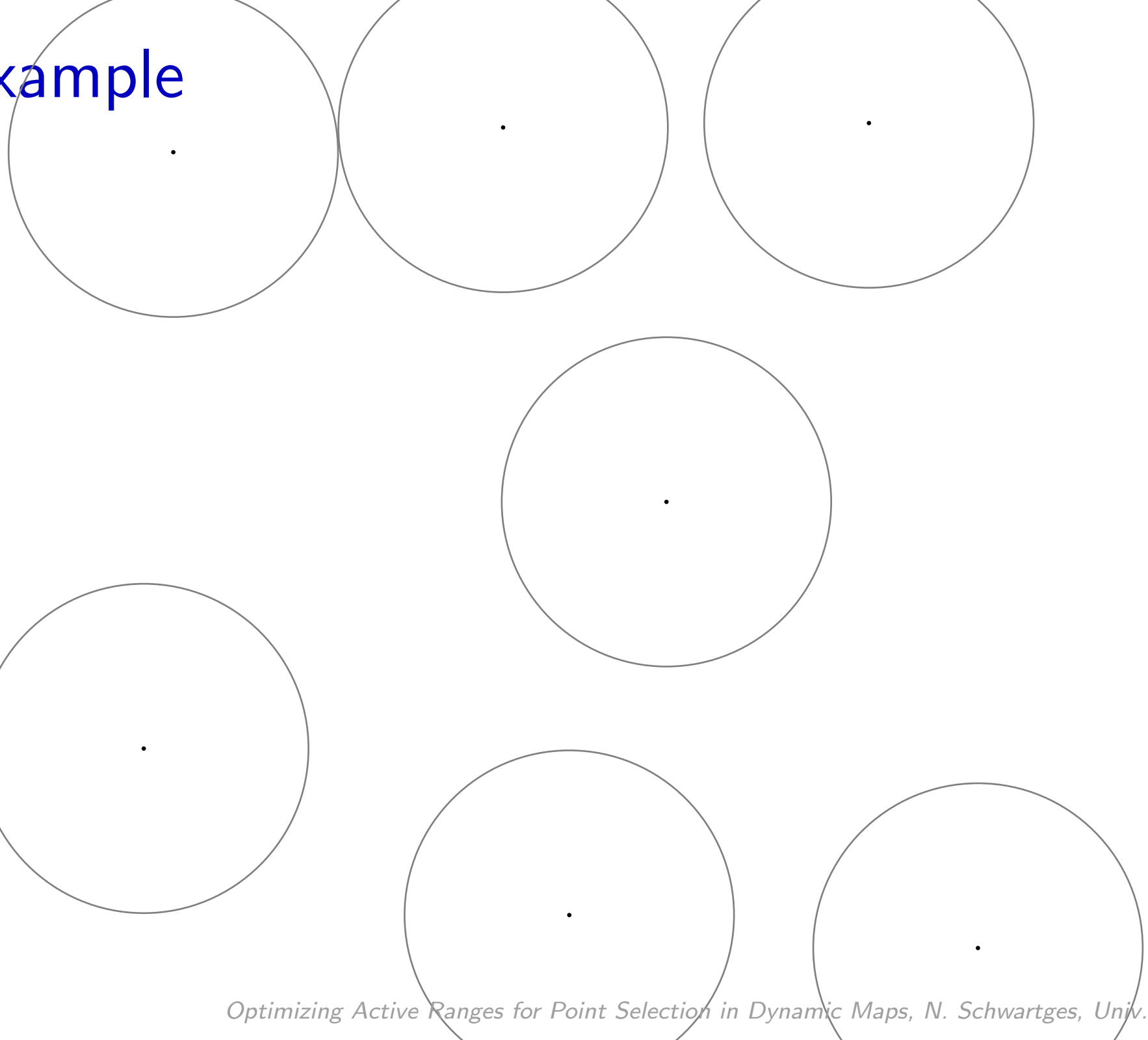
Example



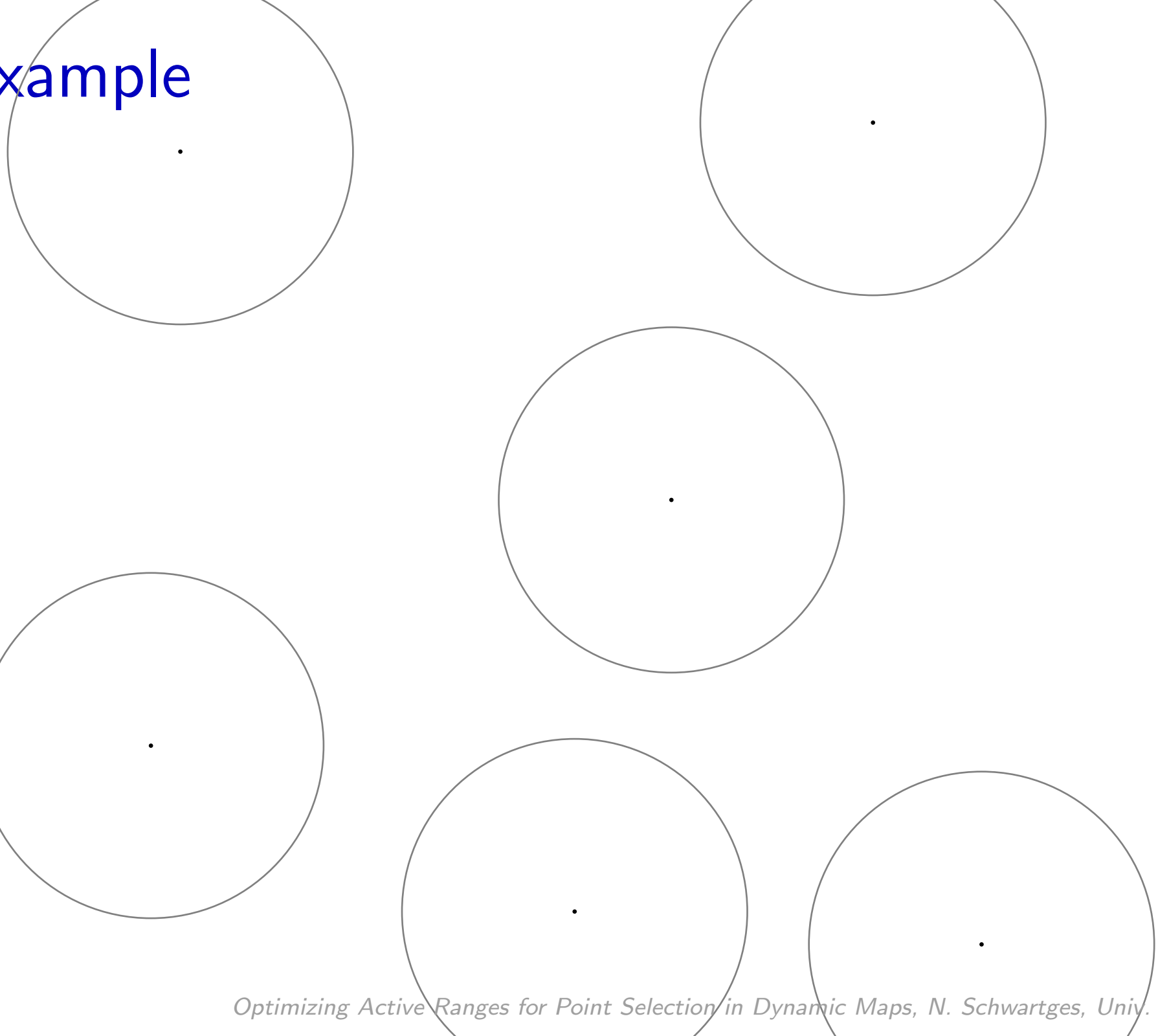
Example



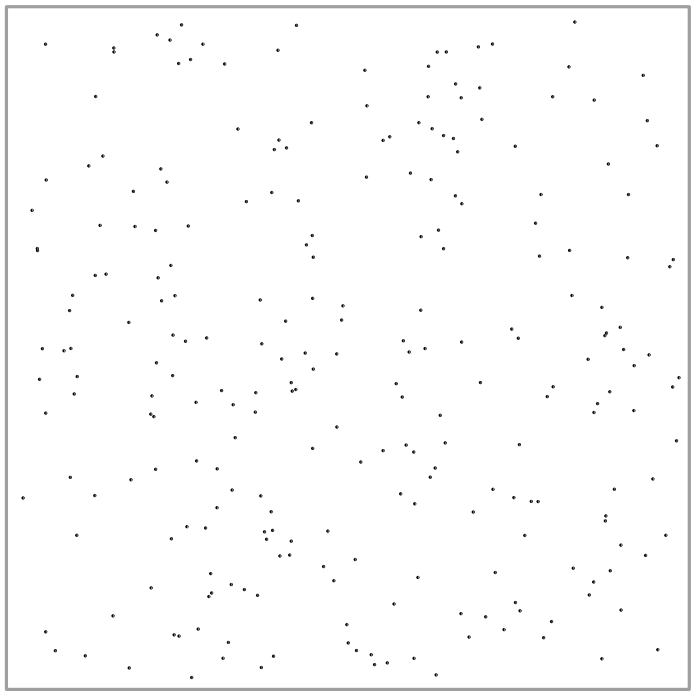
Example



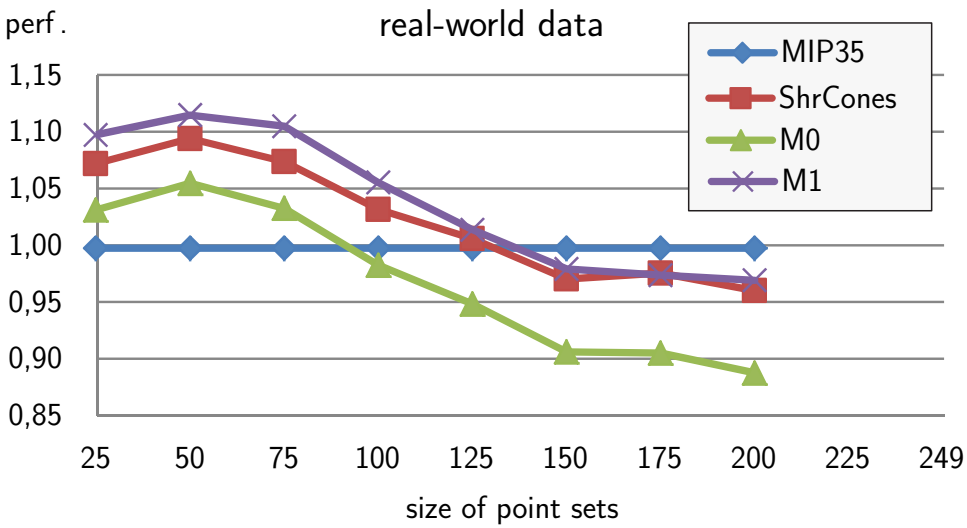
Example

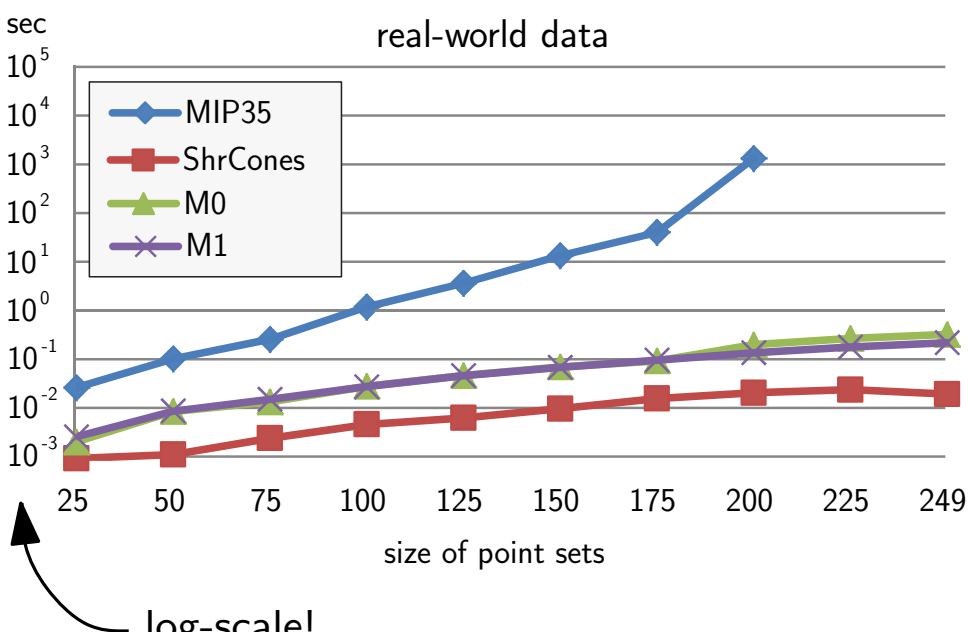


Experiments

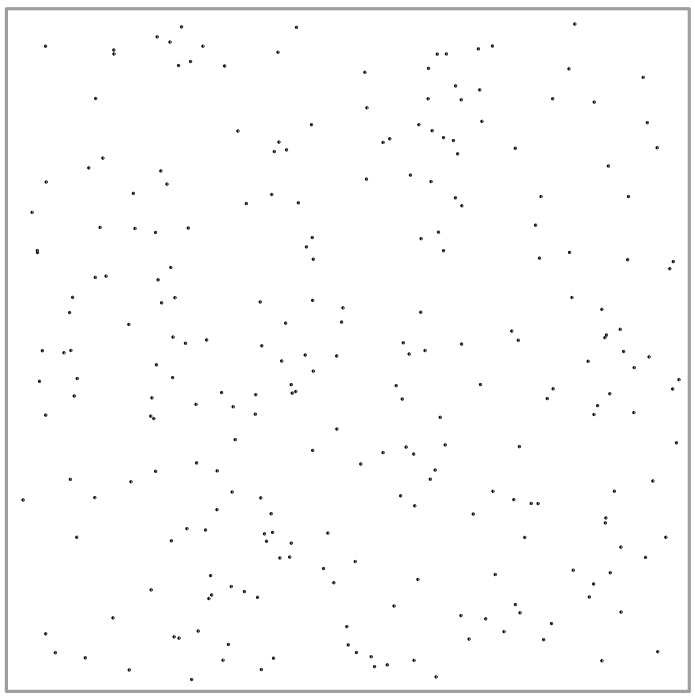


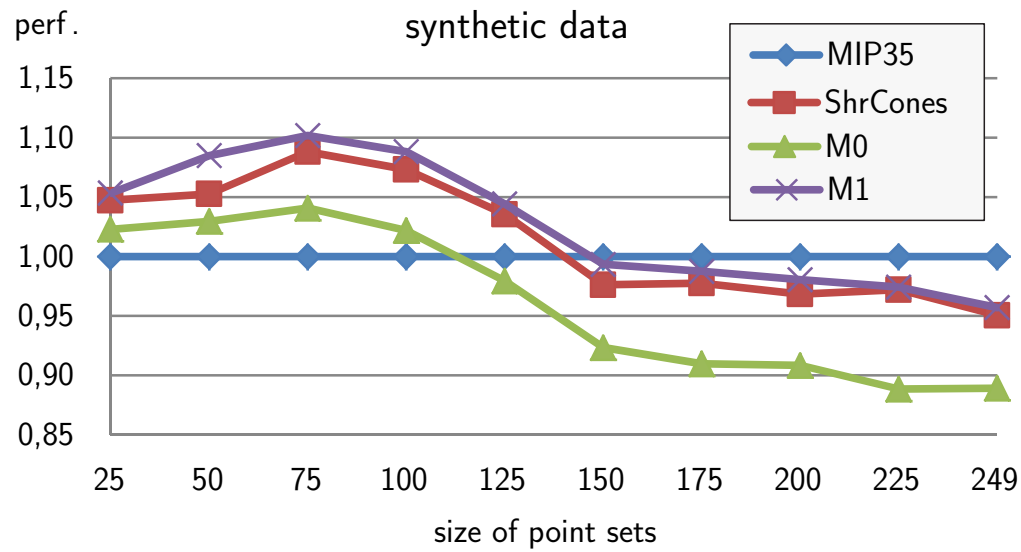
Experiments

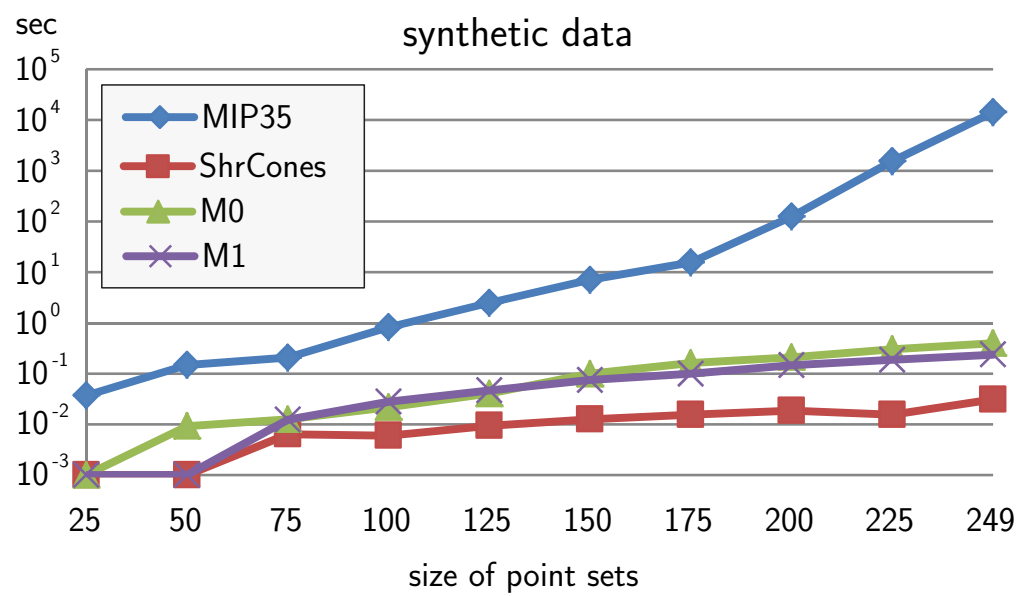




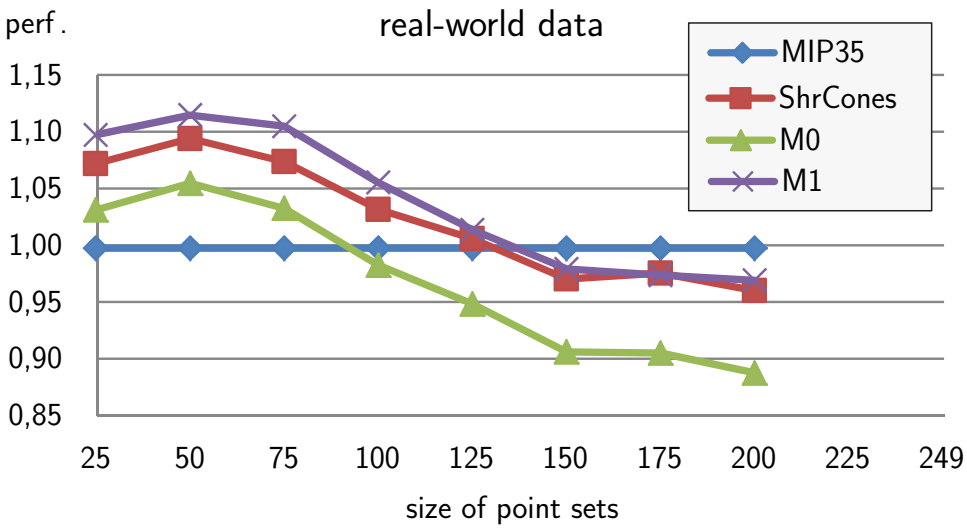
Experiments

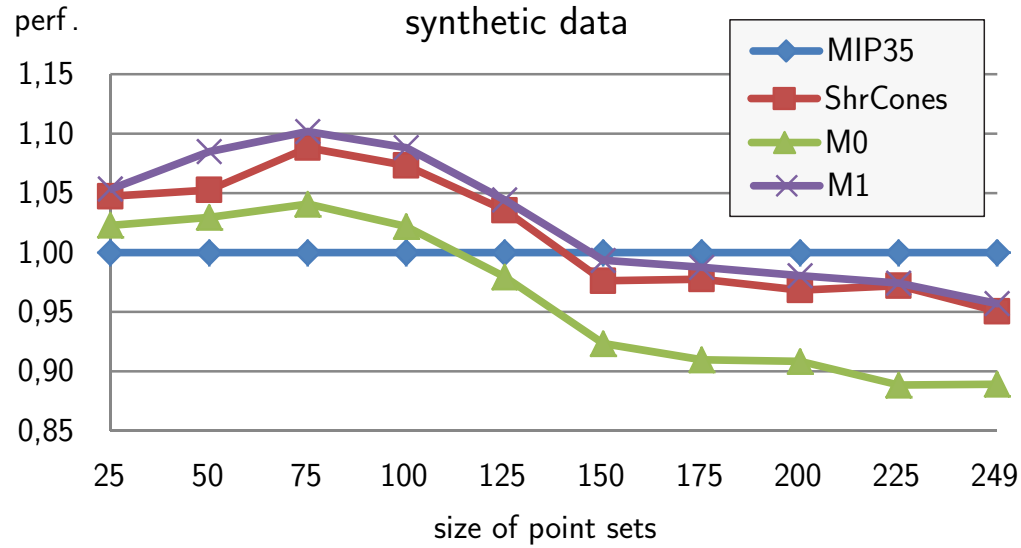


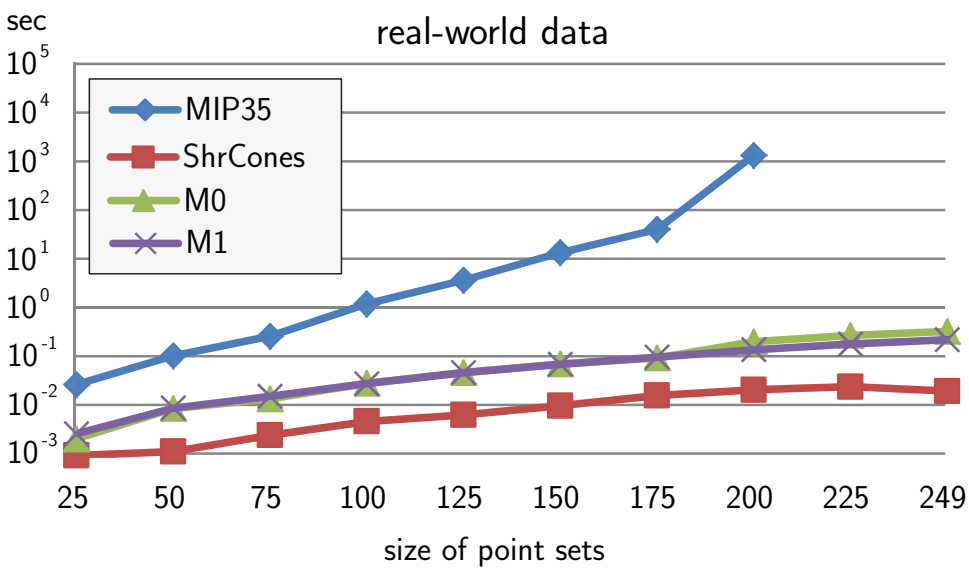


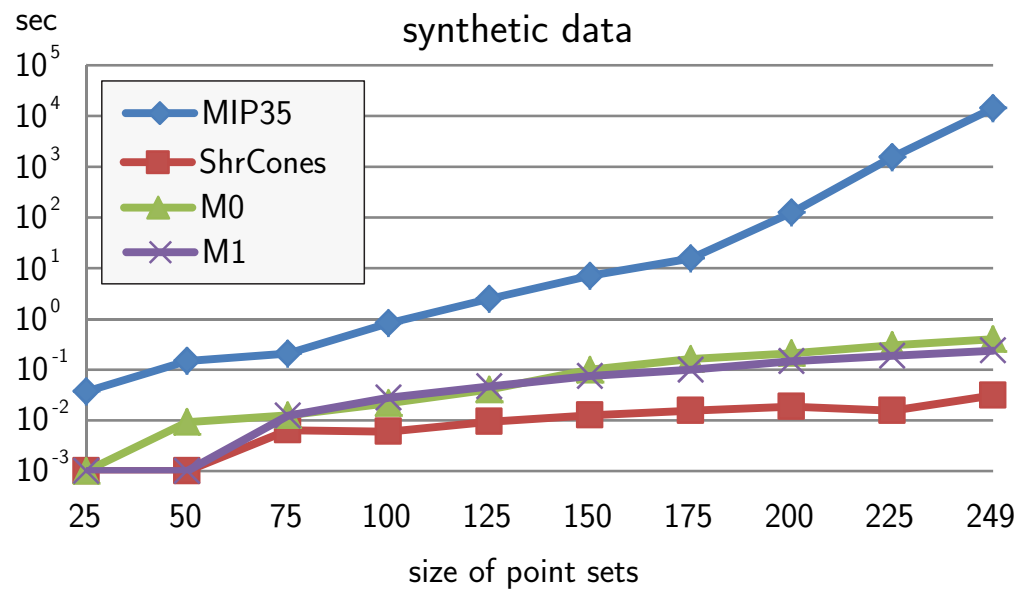


Experiments









Summary & Conclusion

Summary

- Shrinking Cones Heuristic
- Growing Cones Heuristic
  - M0
  - M1

Summary & Conclusion

Summary

- Shrinking Cones Heuristic
- Growing Cones Heuristic
  - M0
  - M1
- Experiments
  - fast
  - good performance

Summary & Conclusion

Summary

- Shrinking Cones Heuristic
- Growing Cones Heuristic
  - M0
  - M1
- Experiments ⟶ Conclusion
  - fast
  - good performance

  applicable for real-world instances

Summary & Conclusion

Summary

- Shrinking Cones Heuristic
- Growing Cones Heuristic
  - M0
  - M1
- Experiments → Conclusion
  - fast
  - good performance

applicable for real-world instances

- thinning out points

Summary & Conclusion

Summary

- Shrinking Cones Heuristic
- Growing Cones Heuristic
  - M0
  - M1
- Experiments → Conclusion
  - fast
  - good performance

applicable for real-world instances

- thinning out points
- labeling with discs (POIs)

Summary & Conclusion

Summary

- Shrinking Cones Heuristic
- Growing Cones Heuristic
  - M0
  - M1
- Experiments → Conclusion
  - fast
  - good performance

  applicable for real-world instances
  - thinning out points
  - labeling with discs (POIs)

Thank you!