Efficiently Computing Succinct Trade-off Curves

Sergei Vassilvitskii

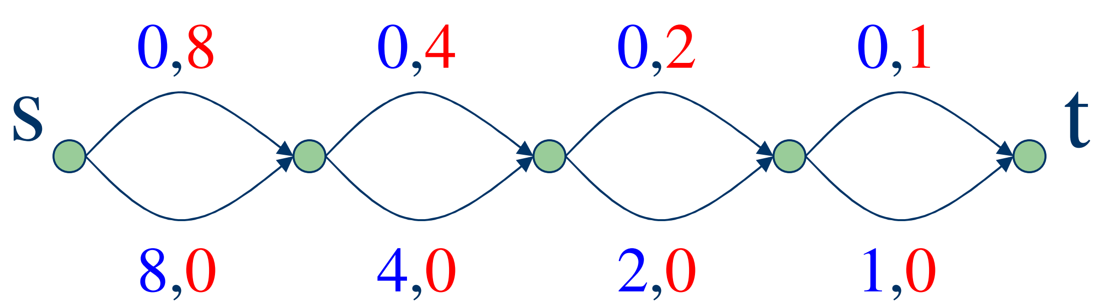
Mihalis Yannakakis

Outline

- Introduction
- Polynomial size trade off curves
- Construction of ε–Pareto curves in 2-d
- ε–Pareto curves in 3+ d
- Conclusion

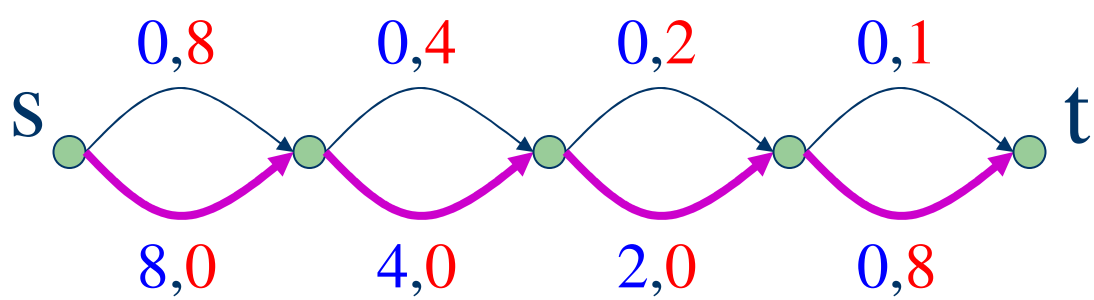
Example

- Graph G = (V,E). Each edge, e, has length l(e) and cost c(e).
- Find the shortest, cheapest s-t path.



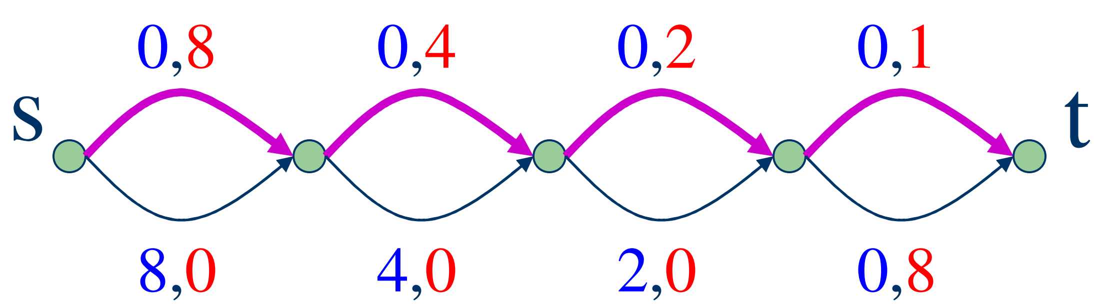
Example (2)

- Graph G = (V,E). Each edge, e, has length l(e) and cost c(e).
- Can have a cheap (0), long (15) path:



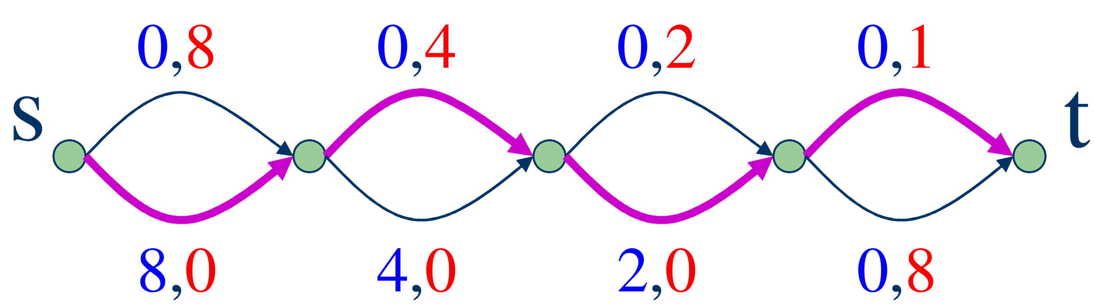
Example (3)

- Graph G = (V,E). Each edge, e, has length l(e) and cost c(e).
- Or a short (0), expensive (15) path:



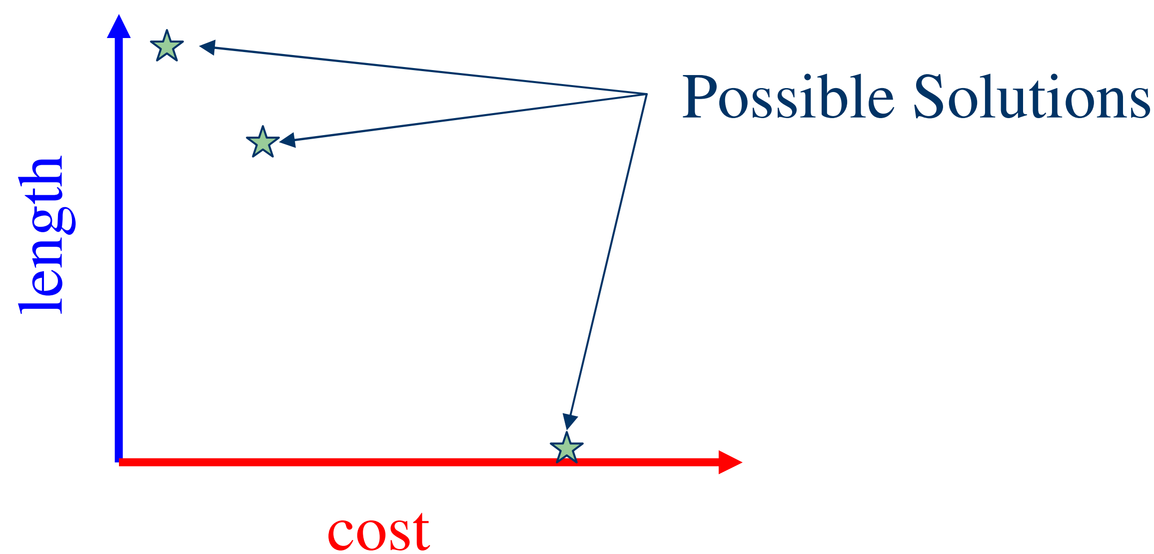
Example (4)

- Graph G = (V,E). Each edge, e, has length l(e) and cost c(e).
- Or anything in between:



Pareto/Trade-off curves

- We are looking at a trade-off (also known as Pareto) curve:



Size of the Curves

- When computing trade-off curves, we are only interested in undominated points.
- But even these trade-off curves can be exponential in size.
  - For the simple problem above, every single path defines an undominated point on the trade-off curve.

Outline

- Introduction
- Polynomial size trade-off curves
- Construction of ε–Pareto curves in 2-d
- ε–Pareto curves in 3+ d
- Conclusion

Approximate Pareto Curves

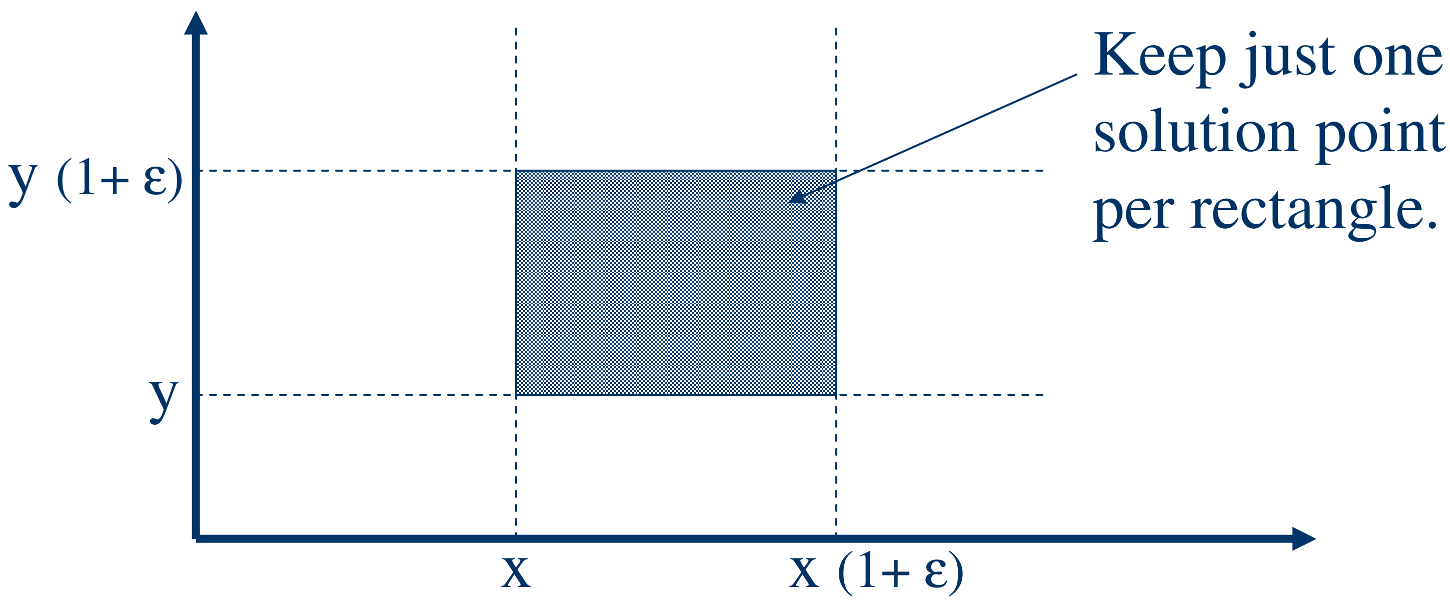
- Consider approximate trade-off curves.
  - For any solution point p, there exists a point p’ such that p’ is no worse than p by a $(1+\varepsilon)$ factor in all objectives.
  - Example: Path with length 9, cost 6 is approximated by a path with length 8, cost 7 with $\varepsilon = 0.17$

Polynomial size ε-Pareto sets (2)

- Theorem [PY ’00]: For any d-objective optimization problem, there exists an ε-approximate trade-off curve of size polynomial in $(1/\varepsilon)$ and exponential in d.

Polynomial size ε-Pareto sets

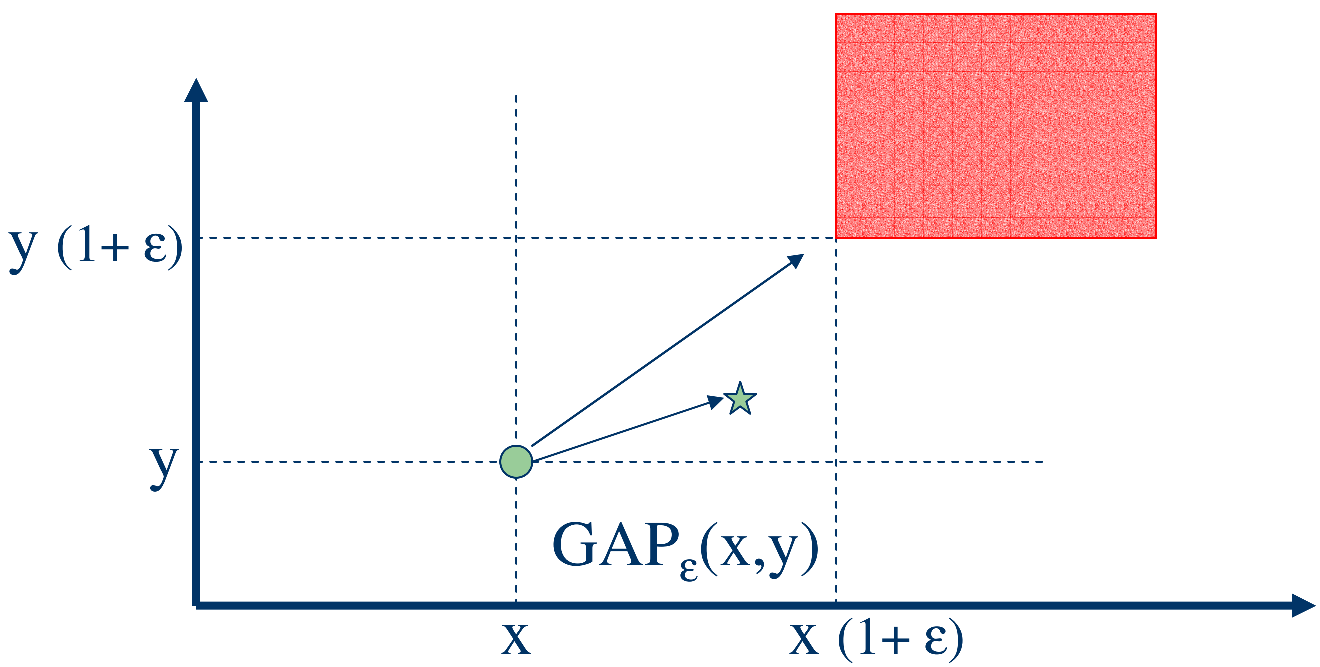
- Proof (2-objectives for simplicity)



GAP Primitive

- Further, we can find an ε-approximate trade-off curve iff we can solve the GAP primitive:
  - Given objective function values (f_1, f_2, …) either return a solution point that is better in all objectives, or assert that no solution is better by more than a $(1+\varepsilon)$ factor in all objectives.

GAP_ε Primitive (2)



Outline

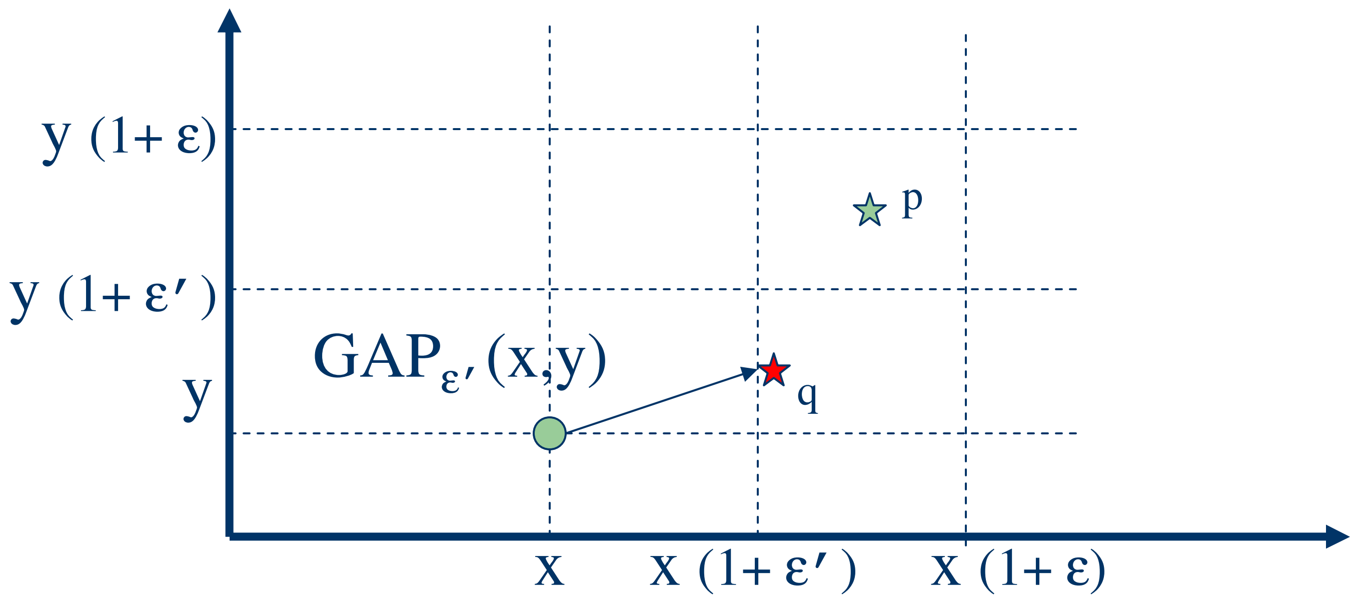
- Introduction
- Polynomial size trade-off curves
- Construction of ε–Pareto curves in 2-d
- ε–Pareto curves in 3+ d
- Conclusion

Constructing Trade-Off Curves

- [PY] give a simple algorithm for computing ε-trade-off curves:
  - Divide the space into rectangles of size $1+\varepsilon' = \sqrt{(1+\varepsilon)}$
  - Call $GAP_{\varepsilon'}$ on all corner points
  - Keep undominated solutions

Constructing Trade Off Curves (2)

- Theorem: Algorithm above produces an ε-Pareto set.



Constructing Trade-Off Curves (3)

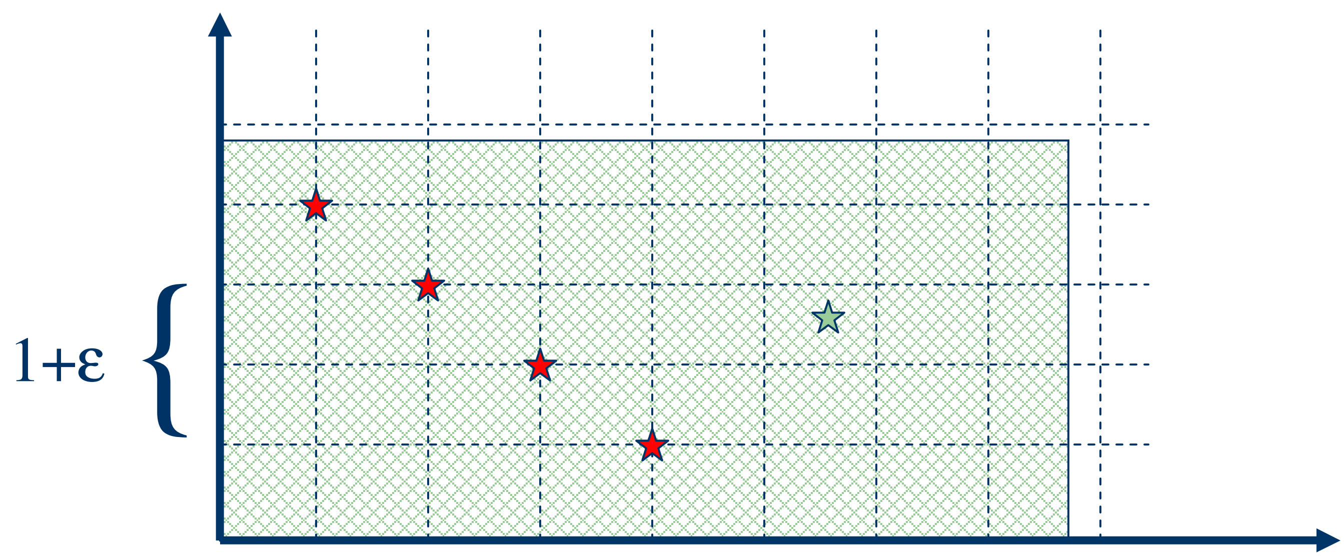
- Runtime $\sim (m/\varepsilon)^2$, m the number of bits in the objective function.
- There are no guarantees on the size of the ε-Pareto set constructed w.r.t. the optimal (smallest) ε-Pareto set for the same data.

Problem Statement

- Find an algorithm to construct small ε-Pareto sets using the GAP function as a black box.
- The algorithm should run in time proportional to the output size, and log (m/ε)

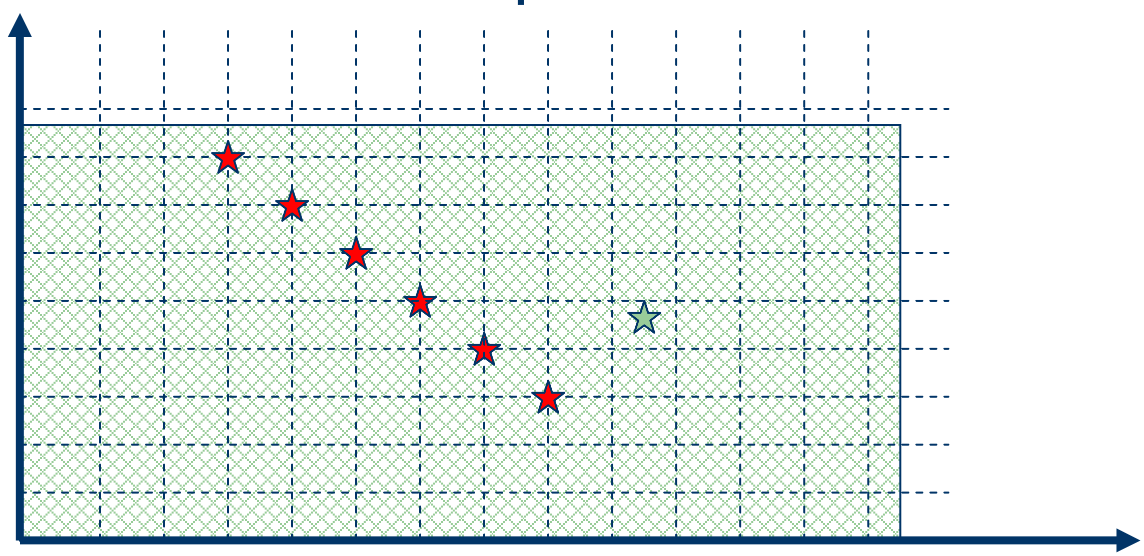
Small Trade-Off Curves (1)

- Theorem: The size of the trade-off curve produced by the PY algorithm is within 7 of opt.



Going Even Smaller

- Consider ε' : $(1+\varepsilon')^4 = 1+\varepsilon$
- The same [PY] algorithm will return an ε-Pareto size within 11 of opt, call this set Q.

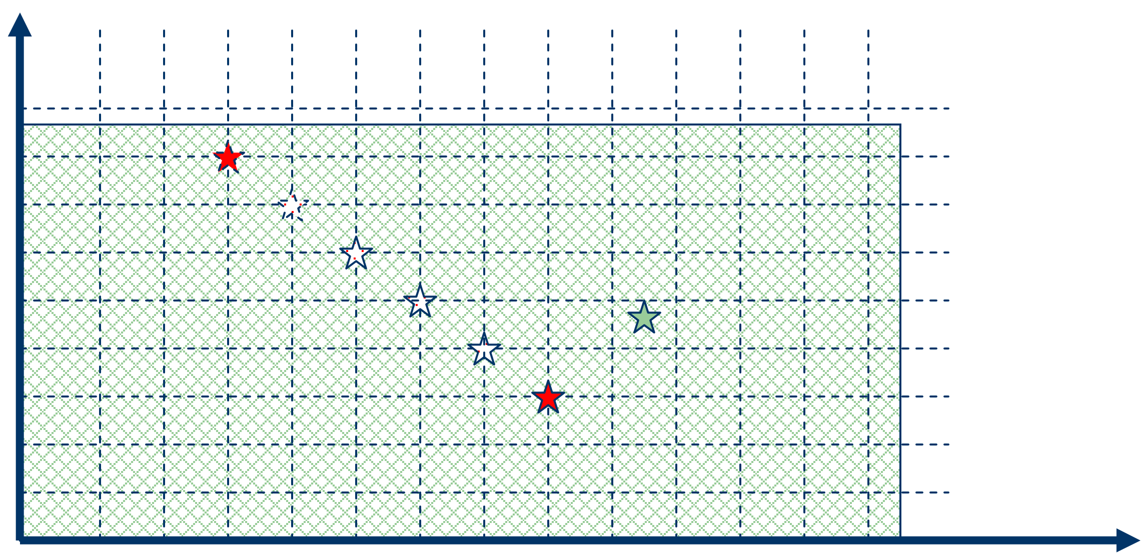


Even Smaller (2)

- But is is also a $(1+\varepsilon')^2$ Pareto set.
- Greedily make R = $(1+\varepsilon')^2$ cover of the points.
  - The result is a $(1+\varepsilon)$ Pareto set.
    - Every solution point p has a point q within $(1+\varepsilon')^2$ in the intermediate set Q.
    - q is covered within $(1+\varepsilon')^2$ in the final set R.
    - Thus p is covered within $(1+\varepsilon')^2(1+\varepsilon')^2 = 1+\varepsilon$ by some point in R.

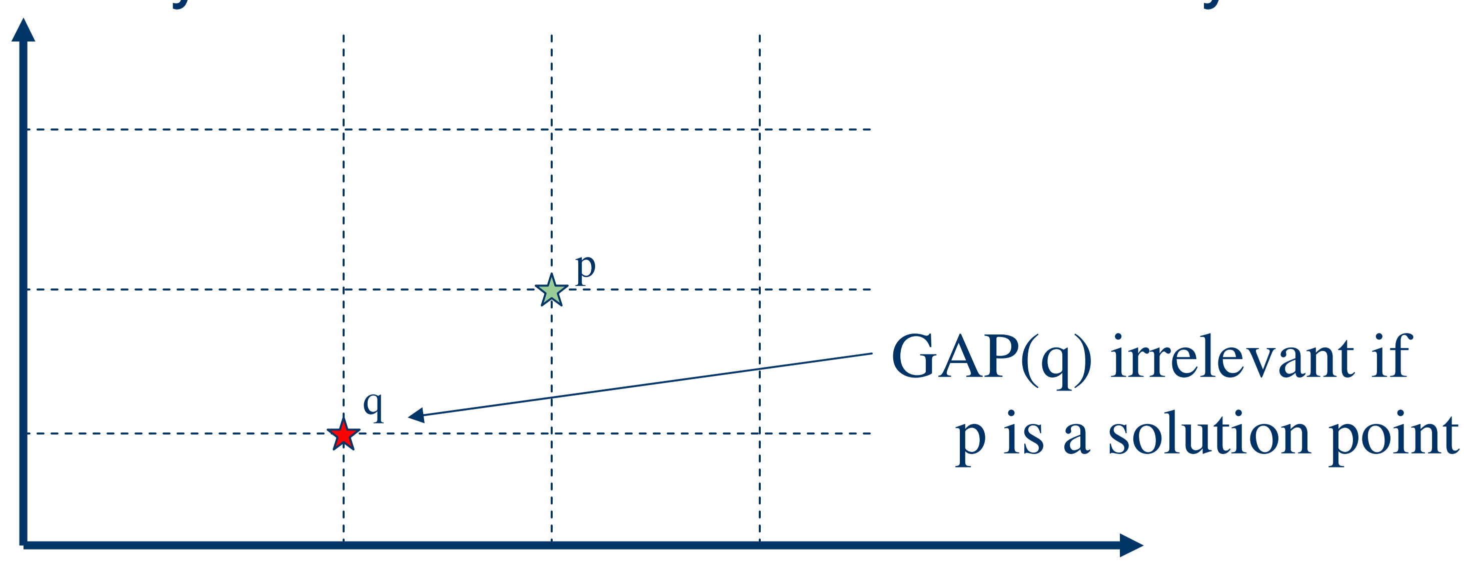
Even Smaller (3)

- Size of R is within 3 of smallest Pareto Set.
  - Need 3 points to cover the 11 present



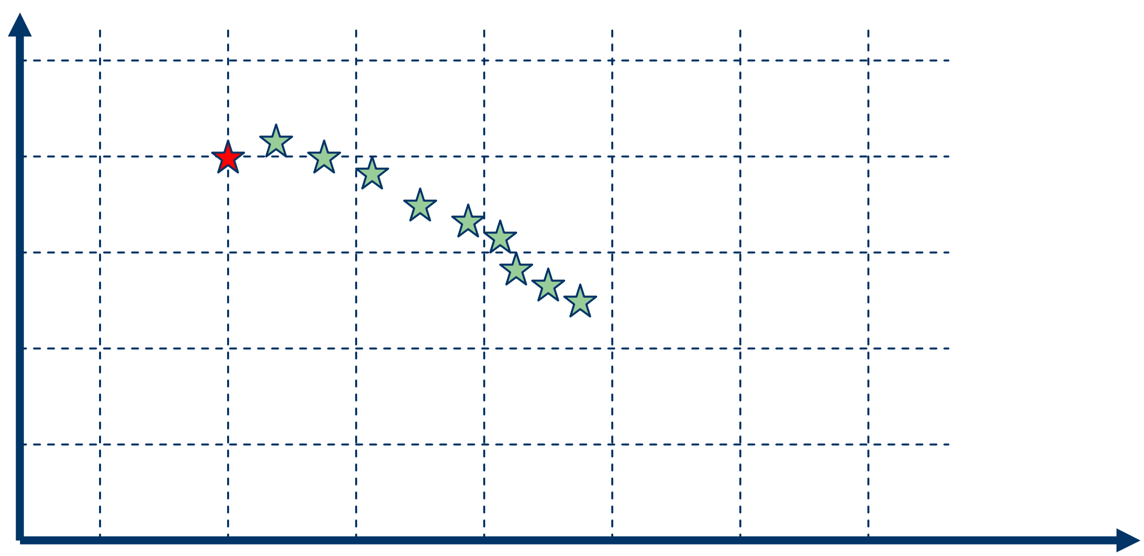
Doing it Faster

- Recall, current algorithm requires $(m/\varepsilon)^2$ number of GAP calls.
- But many of these calls are unnecessary.



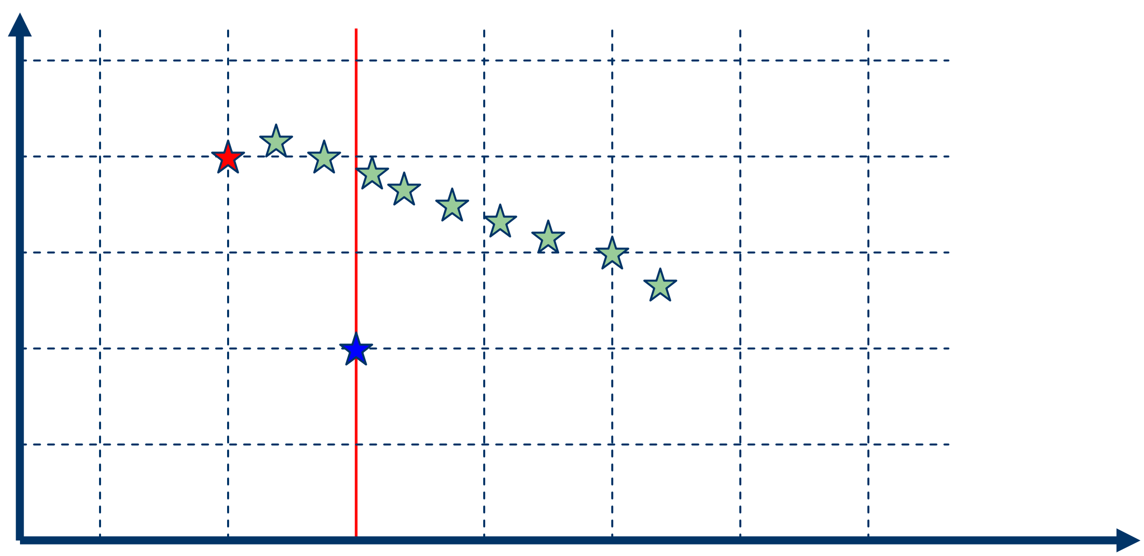
ZigZag Algorithm

- Do a search for the next point in the curve.



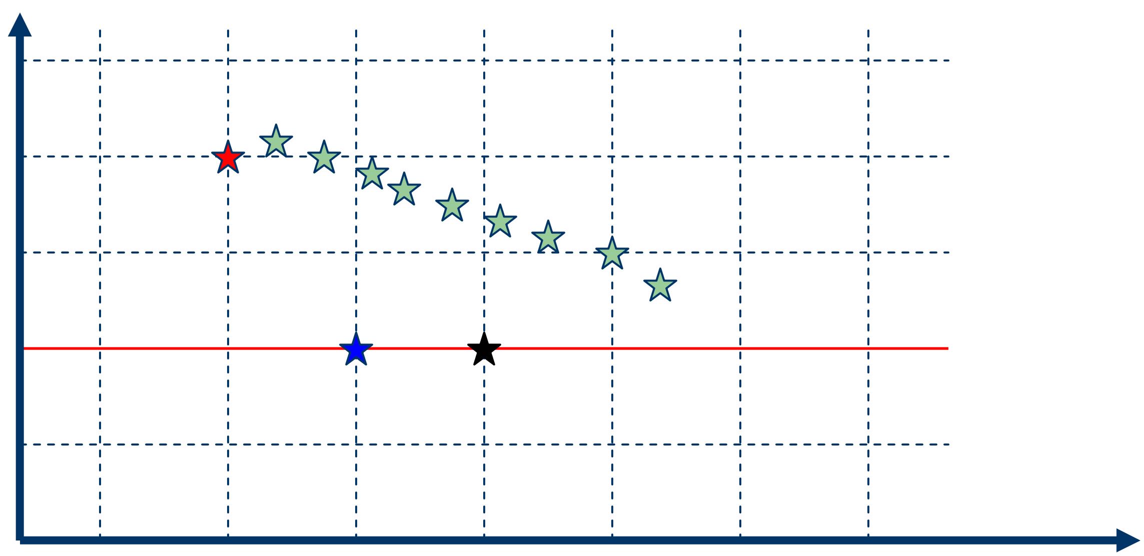
ZigZag Algorithm (2)

- Max y where GAP is yes and x is bigger.



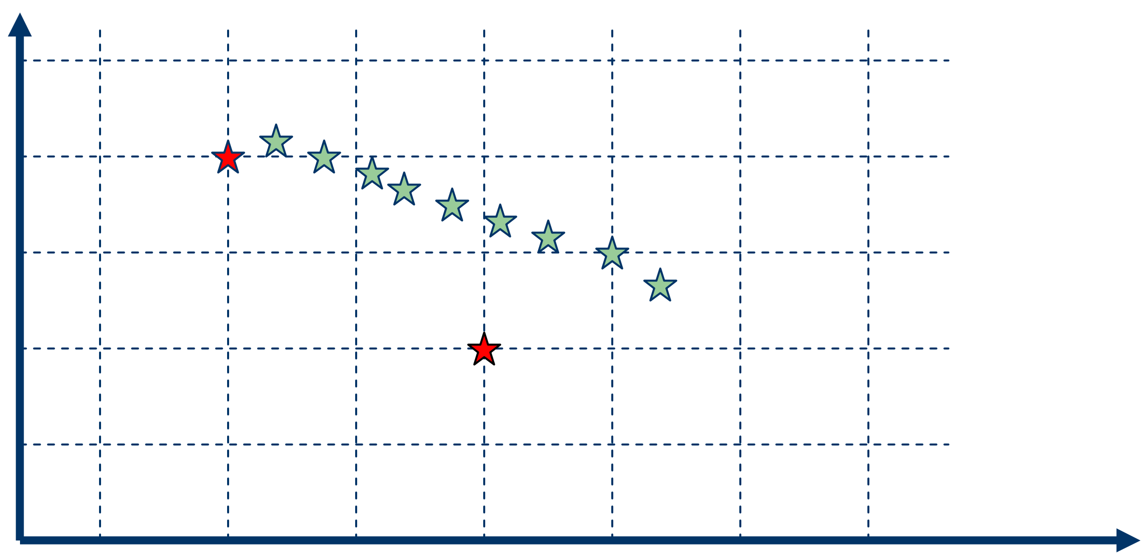
ZigZag Algorithm (2)

- Max x where GAP is yes at same y value



ZigZag Algorithm (2)

- Repeat and Continue

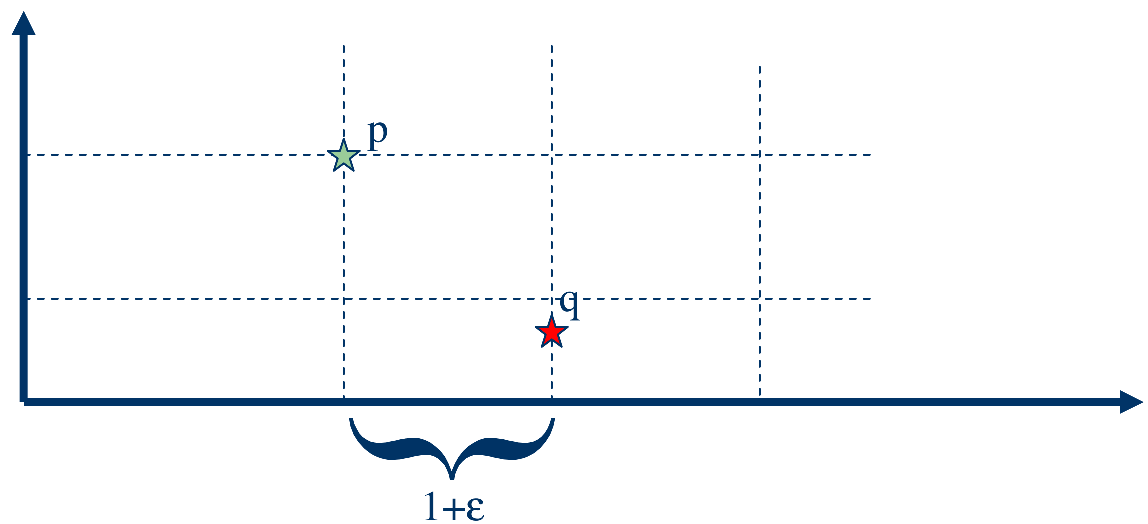


ZigZag Algorithm (3)

- Can implement the searches as a binary search.
- Thus require only $O(\log m)$ to discover a new point.
- If k is the number of points in the smallest ε-Pareto set, we will need $O(k \log m/\varepsilon)$ GAP calls total.

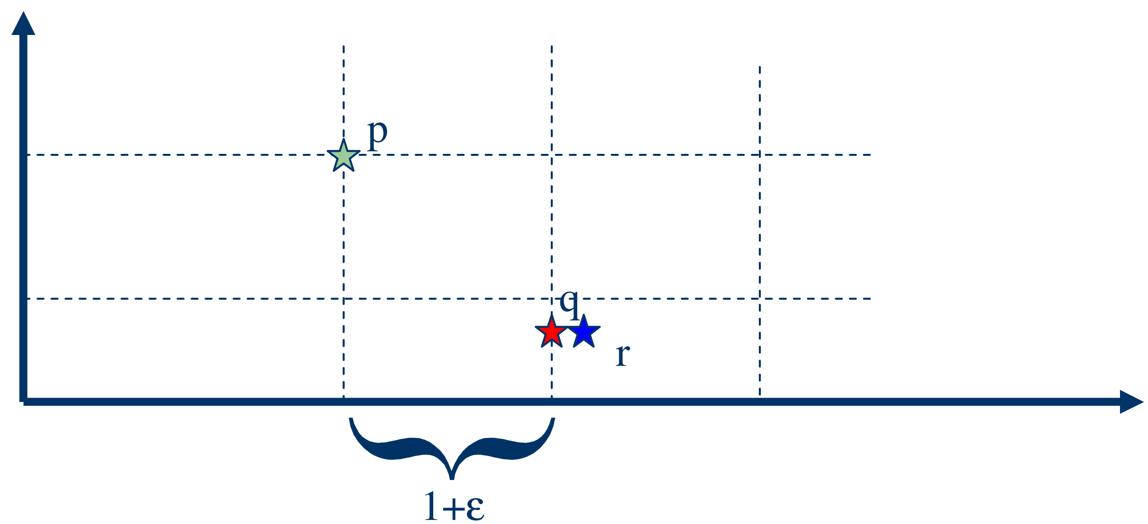
Lower Bounds

- Using the GAP Framework no algorithm can be better than 3 competitive.
- Here size of the smallest ε–Pareto set is 1.



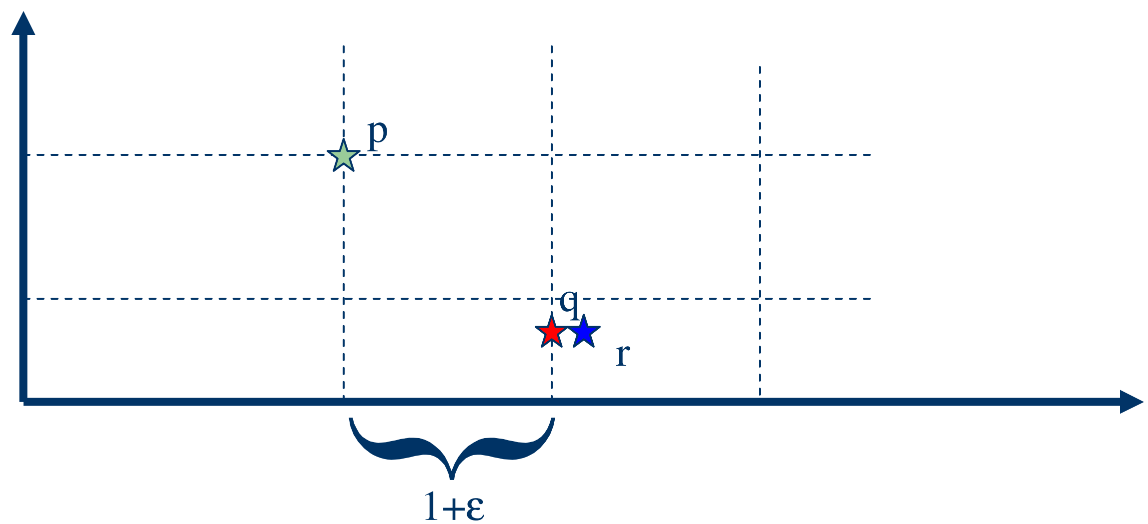
Lower Bounds

- Using the GAP Framework no algorithm can be better than 3 competitive.
- Here size of the smallest ε–Pareto set is 2



Lower Bounds (2)

- But with GAP as a black box we cannot distinguish between the two cases.



ε-Pareto on 2 objectives

- Present an algorithm:
  - ε-Pareto size $\leq 3k$ where k is optimal
  - Runtime $O(k \log m/\varepsilon)$ GAP calls.
- Lower Bound
  - Using GAP all algorithms are no better than 3 competitive in the worst case.

Outline

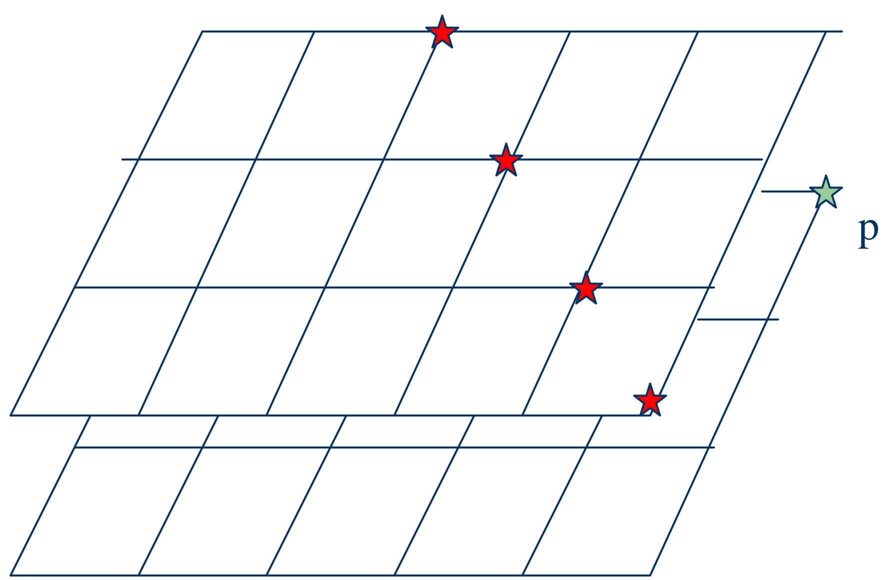
- Introduction
- Polynomial size trade-off curves
- Construction of ε–Pareto curves in 2-d
- ε-Pareto curves in 3+ d
- Conclusion

3-d Lower Bound

- For any constant c no algorithm can be c-competitive in producing an ε-Pareto set using only the GAP framework.
- We will again show two cases where the size of the smallest ε-Pareto set is different and GAP cannot distinguish between the two.

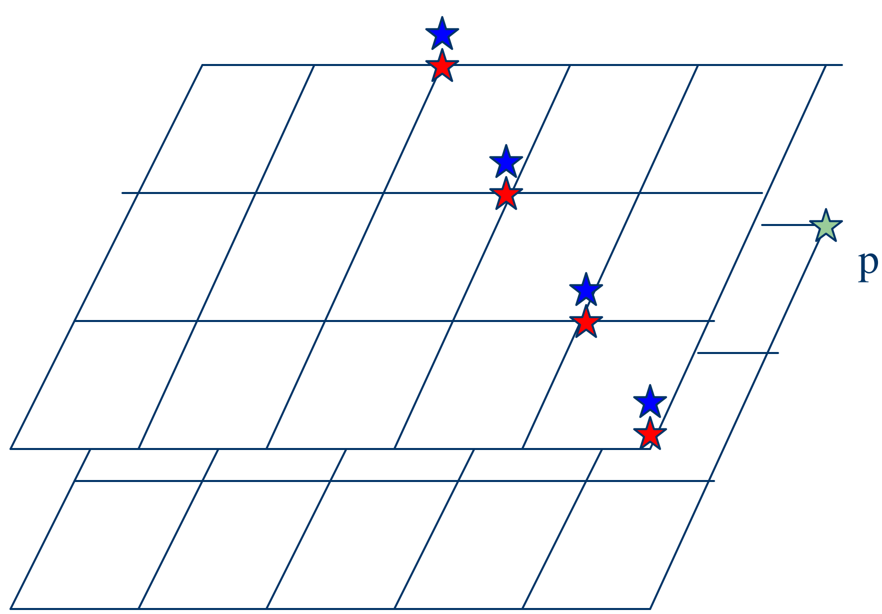
3-d Lower Bound (2)

- Size of smallest Pareto set is 1



3-d Lower Bound (3)

- Size of smallest Pareto set is > 1



3-d Results

- To get around the lower bound we look for ε'-Pareto ($\varepsilon' > \varepsilon$) sets of size comparable to the smallest ε–Pareto sets.
- In particular we give an algorithm that for $(1+\varepsilon') = (1+\varepsilon)^2$, constructs an ε'-Pareto set of size no more than 4k; k is the size of the smallest ε–Pareto set.
  - Runtime = $O(k \log m/\varepsilon)$

Higher d (More Lower Bounds)

- Even if all of the solution points are given explicitly:
  - We cannot do better than log d unless P=NP (There is a simple Set Cover Reduction)
  - Even if we look for ε'-Pareto sets with $(1+\varepsilon') < (1+\varepsilon)^{\log^* d}$, we cannot do better than a $\log^* d$ approximation to the size of the set. (Reduction from asymmetric k-center)

Outline

- Introduction
- Polynomial size trade-off curves
- Construction of ε–Pareto curves in 2-d
- ε–Pareto curves in 3+ d
- Conclusion

Conclusion (1)

- ε–Pareto sets are useful in many applications
  - Can present the user with the trade-off curve between two or more objectives
  - Can compute the ‘knee’ of the curve, and find one solution point that best approximates all of the rest
  - Can solve general versions of bicriteria problems (e.g. bicriteria shortest paths) given GAP as a black box input.

Conclusion (2)

- Presented algorithms that return
  - Almost optimal ε–Pareto sets.
- And that run in time
  - Proportional to the output size – small curves are quick to compute
  - Proportional to log (m/ε)

Open Questions

- Many questions still open
  - Better algorithms for 3+ objectives. Reducing both ε' and the approximation ratio
  - Efficiently merging two ε–Pareto sets (in 3+ d)
  - ‘Concatenating’ two ε–Pareto sets.

Thank you

Any Questions?