UNIVERSITÀ DEGLI STUDI ROMA TRE
Dipartimento di Informatica e Automazione

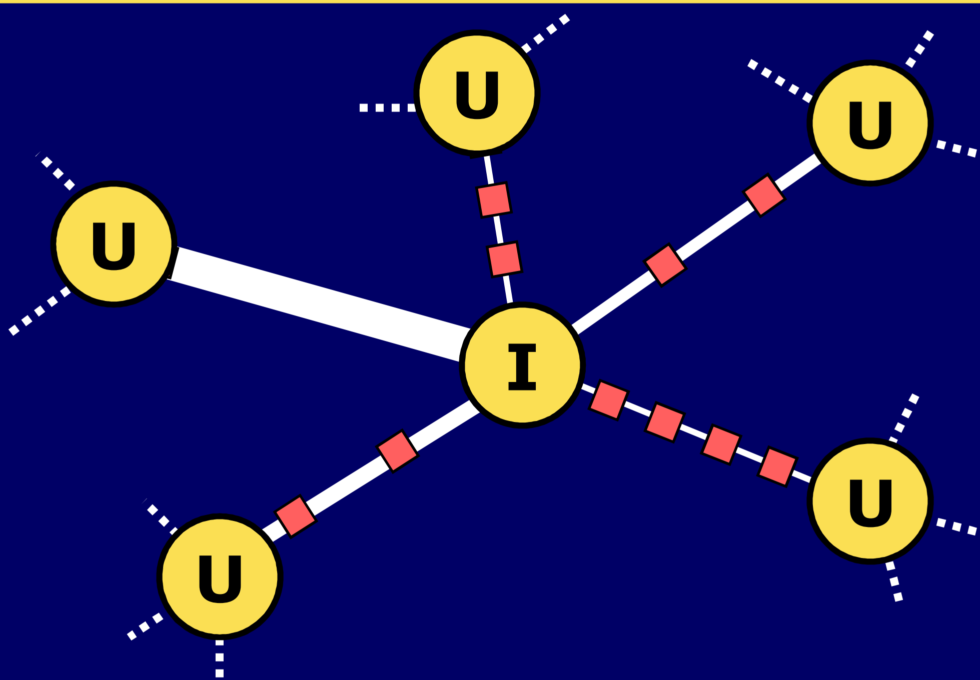
Towards Optimal Prepending for Incoming Traffic Engineering

Giuseppe Di Battista
Maurizio Patrignani
Maurizio Pizzonia
Massimo Rimondini

Roadmap

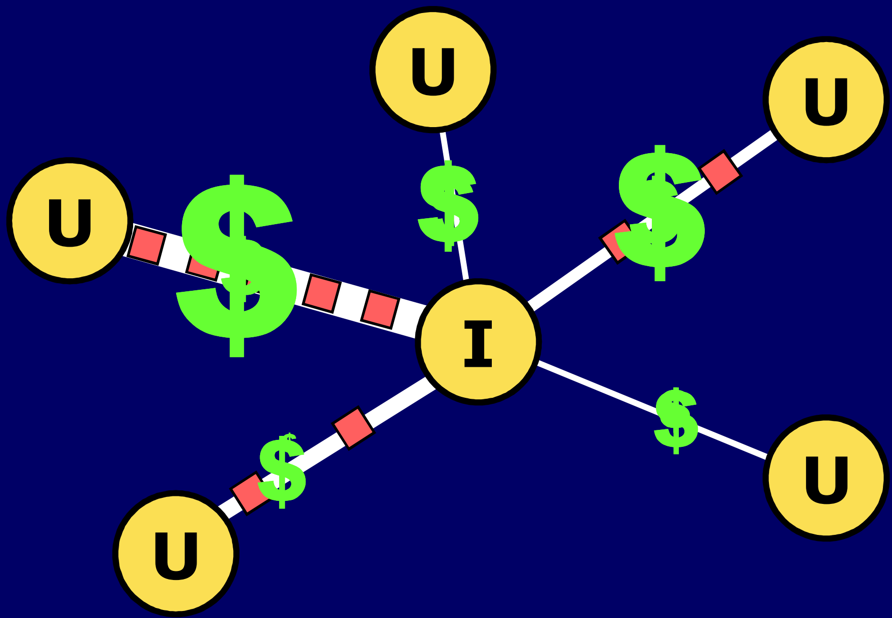
- Why Interdomain Traffic Engineering?
- TE techniques + BGP fundamentals
- Previous work
- TE by optimal prepending
  - Model
  - Integer Linear Programming Formulation
  - Computational Geometry approach
  - Computational complexity & Applicability
- Concluding Remarks

Interdomain Traffic Engineering: WHY?



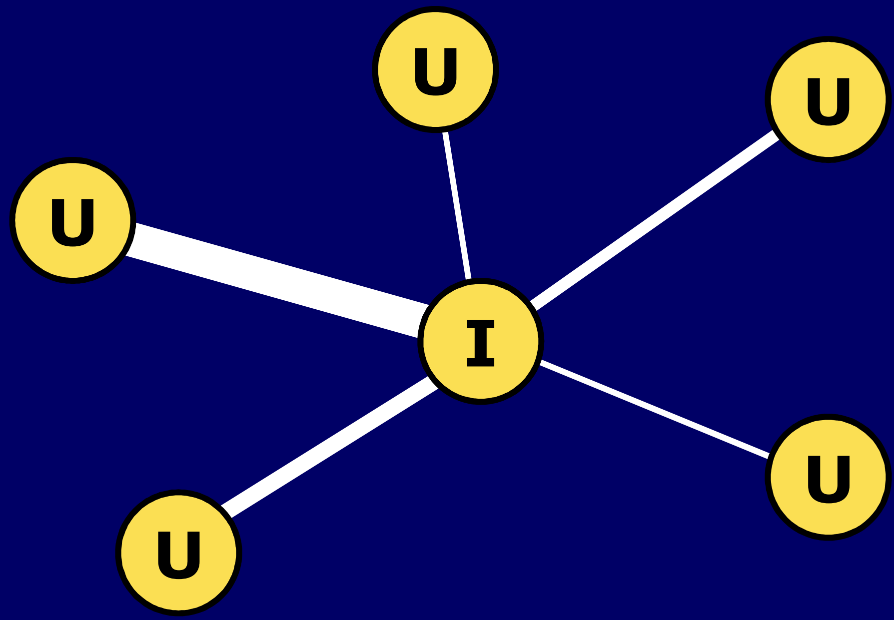
- Balance bandwidth usage

Interdomain Traffic Engineering: WHY?



- Balance bandwidth usage
- Distribute link costs evenly

Interdomain Traffic Engineering: WHY?



- Balance bandwidth usage
- Distribute link costs evenly
- Improve the distribution of internal traffic flows

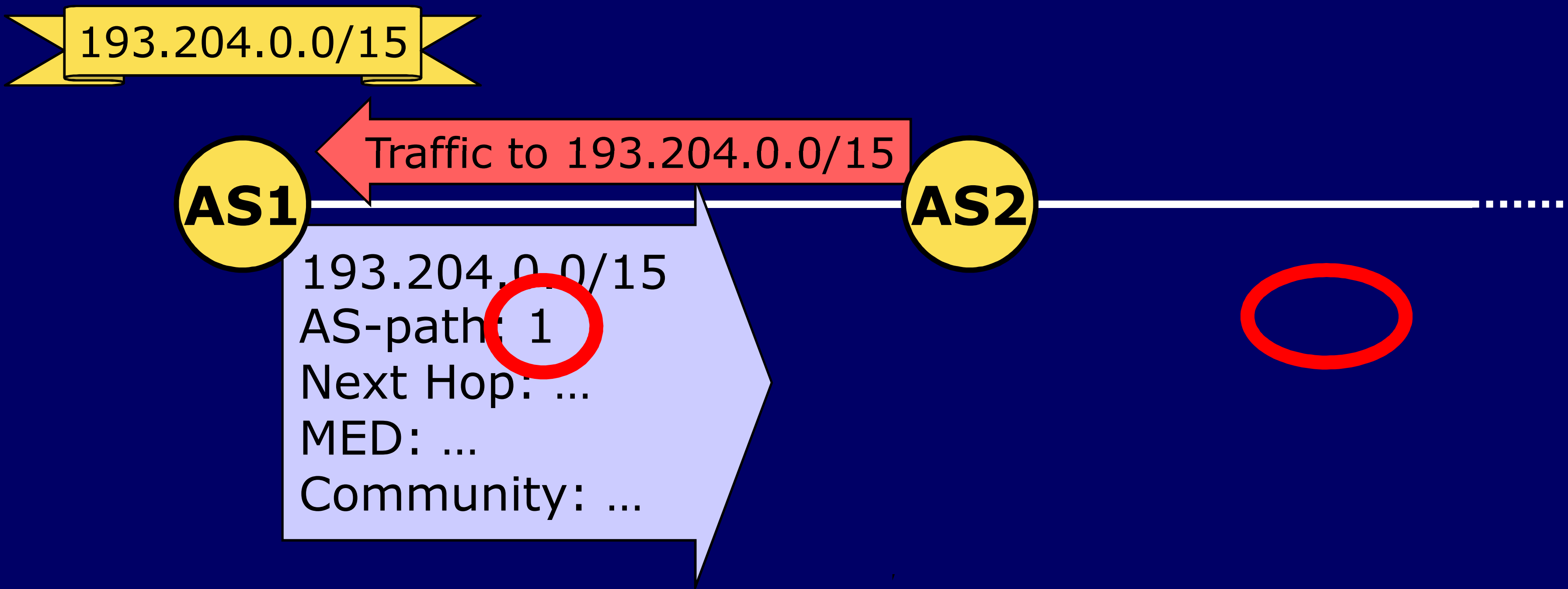
Interdomain Traffic Engineering: HOW?

- Based on BGP announcements

BGP announcement

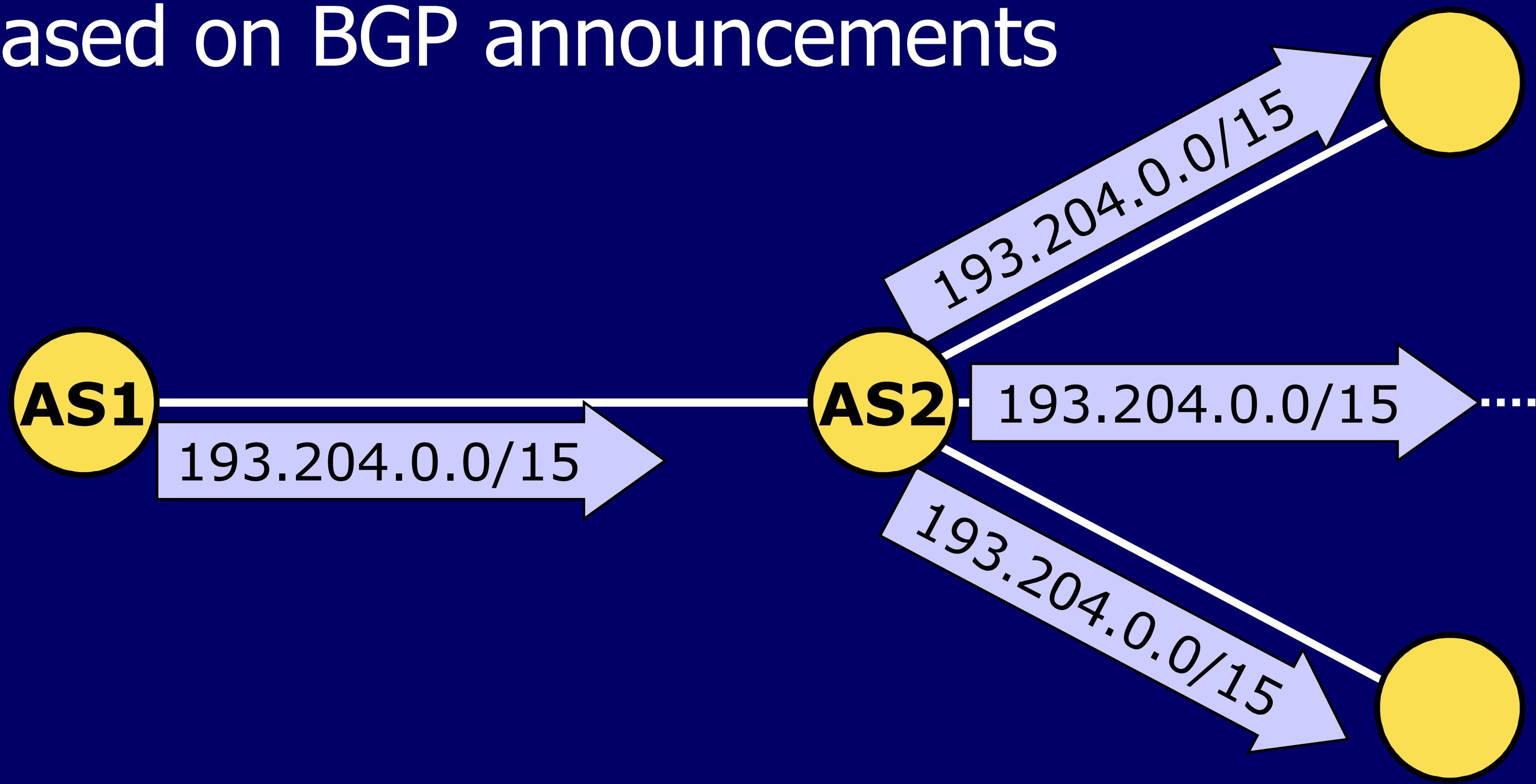
Interdomain Traffic Engineering: HOW?

- Based on BGP announcements



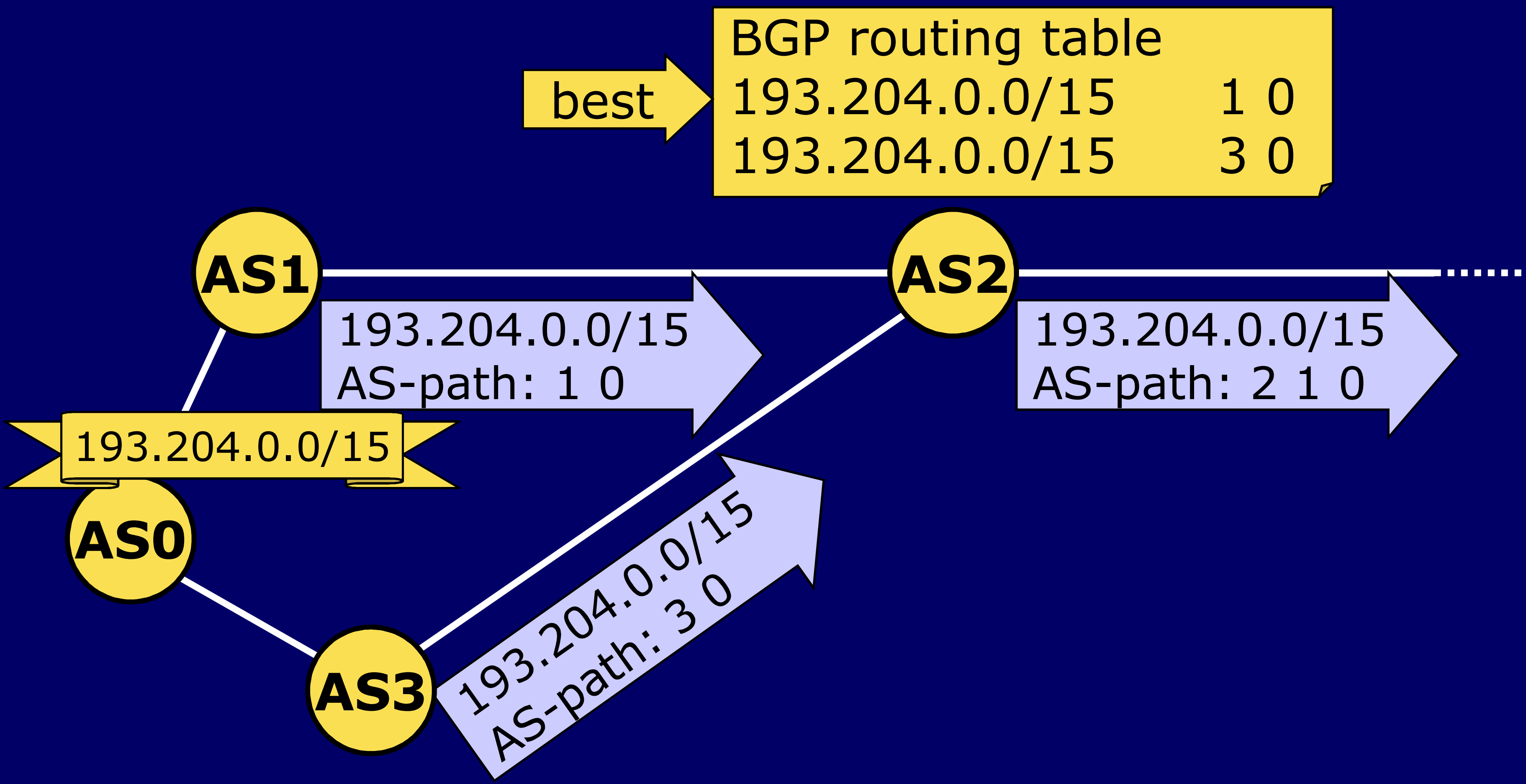
Interdomain Traffic Engineering: HOW?

- Based on BGP announcements



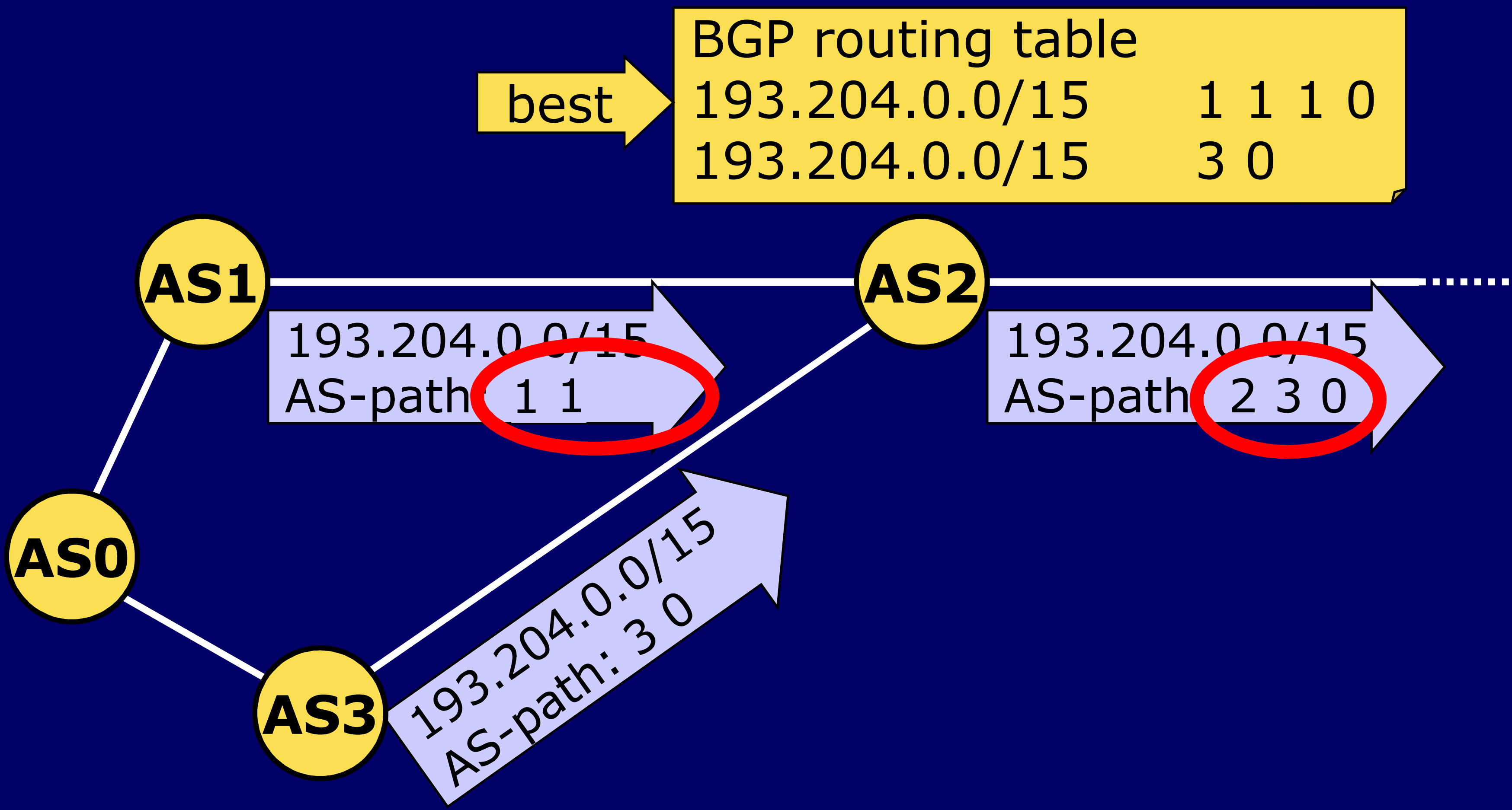
Interdomain Traffic Engineering: HOW?

- Based on BGP announcements



Interdomain Traffic Engineering: HOW?

- Based on BGP announcements

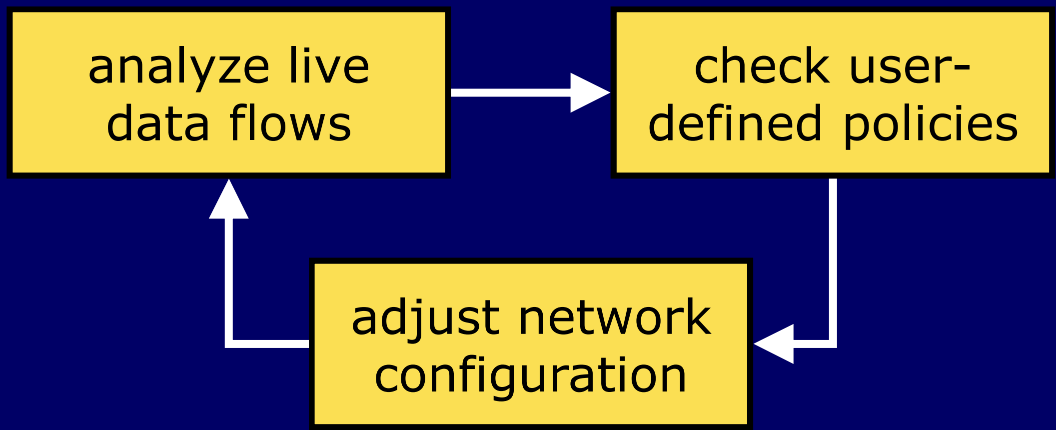


Interdomain Traffic Engineering: HOW?

- Based on BGP announcements
- Previous work:
  - Guidelines for using BGP routing tables and Netflow data to perform TE
    - N. Feamster, J. Borkenhagen, and J. Rexford, Guidelines for Interdomain Traffic Engineering, in *ACM SIGCOMM Computer Communications Review*, Oct 2003.
  - Methods for controlling incoming & outgoing traffic
    - B. Quoitin, S. Uhlig, C. Pelsser, L. Swinnen, and O. Bonaventure, Interdomain Traffic Engineering with BGP, in *IEEE Communications Magazine*, May 2003.
  - Simulations of the impact of TE techniques
    - L. Swinnen, S. Tandel, S. Uhlig, B. Quoitin, and O. Bonaventure, An Evaluation of BGP-based Traffic Engineering Techniques, *Technical Report Infonet-2002-10*, 2003.
  - Hybrid TE techniques (Netflow+ICMP probing)
  - Studies on the impact of routing policies on the AS path length
    - R. K.C. Chang and M. Lo, Inbound Traffic Engineering for Multihomed ASes using AS Path Prepending, in *Proc. NOMS 2004*.

Interdomain Traffic Engineering: HOW?

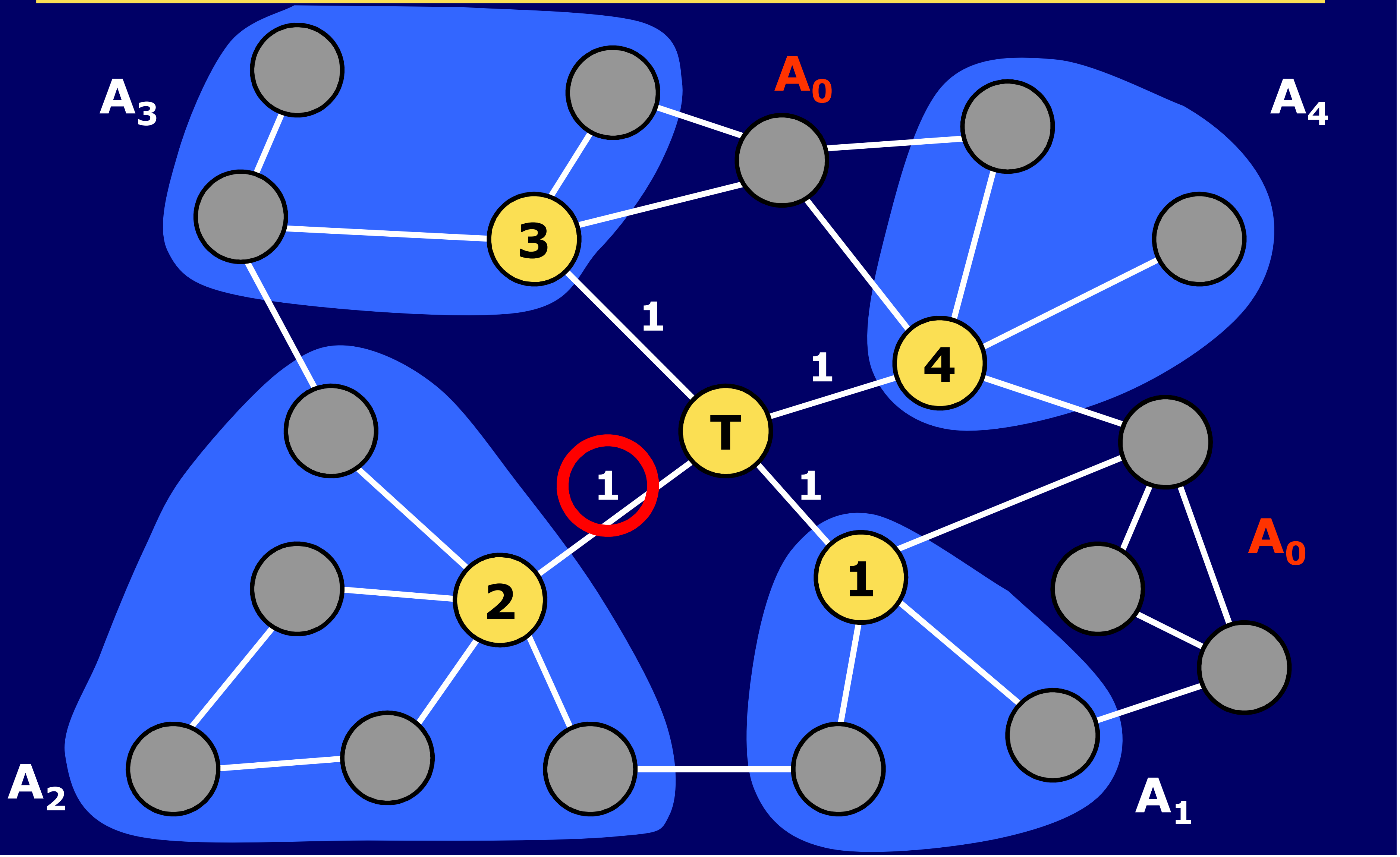
- Based on "route control" devices
  - Peer Director – Radware, Inc.
  - Flow Control Platform – Internap, Inc.
  - Adaptive Networking Software – RouteScience, Inc.



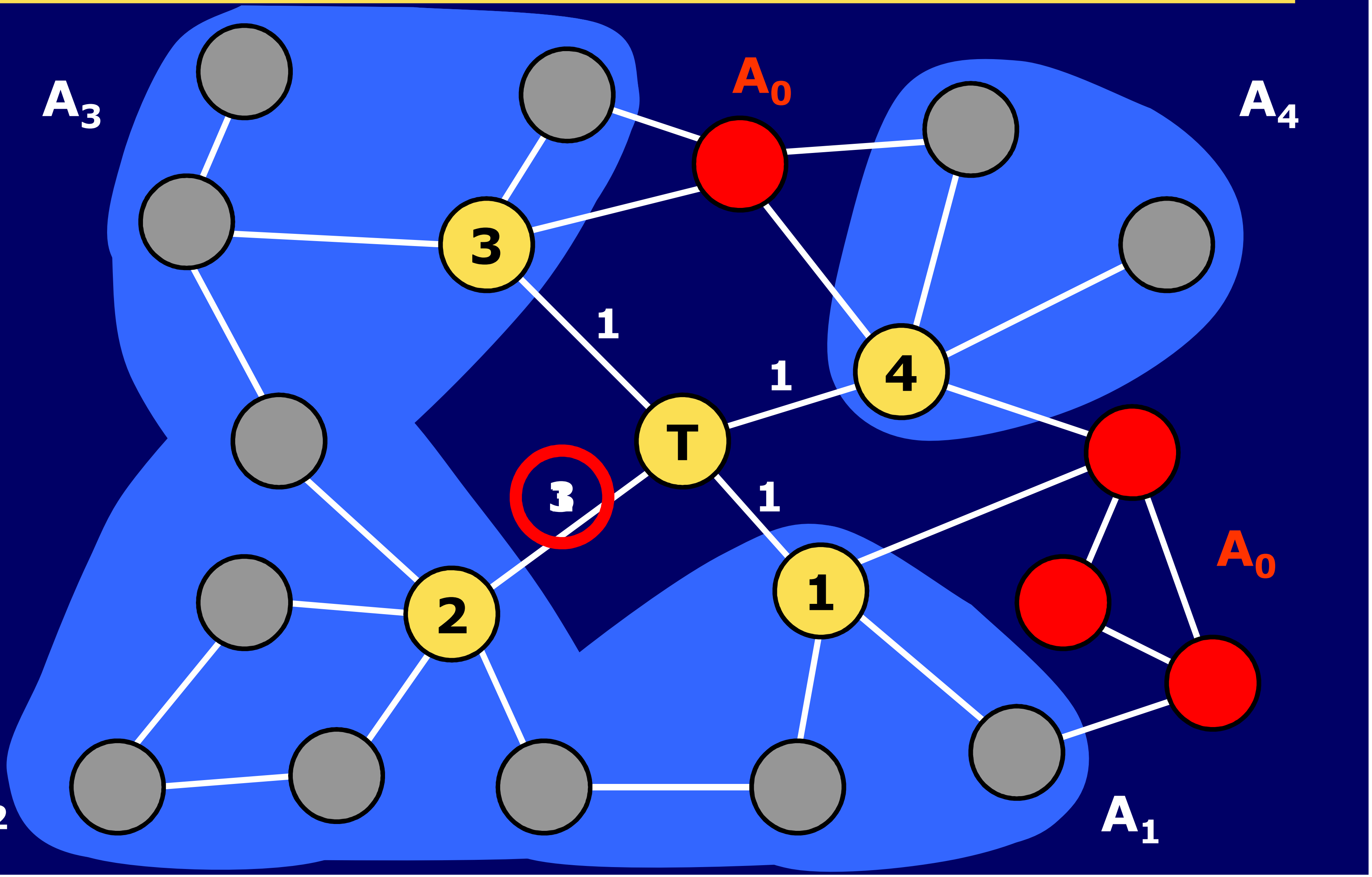
Approaching Interdomain Traffic Engineering

- We…
  - …focus on incoming TE performed by using prepending
  - …introduce a formal model to describe the prepending choices of an ISP
  - …propose approaches to compute optimal prepending:
    - Integer Linear Programming formulation
    - Computational Geometry techniques

Incoming Traffic Engineering
The Model



Incoming Traffic Engineering
The Model



Incoming Traffic Engineering
The Model

- Assumptions:
  - no policy constraints
  - ASes choose the shortest path to reach t
- Why not being naïve?
  - large number of prepending combinations
  - route flap dampening

Incoming Traffic Engineering ILP Formulation

- Input
  - d_{ai}: length of (a) shortest AS-path from *a* to *i*
- Variables
  - w_i: amount of prepending through upstream *i*
  - c_{ai}: 1 iff *a* chooses upstream *i*, 0 otherwise
  - e_{aij}: 1 iff *a* has (≥2) shortest paths through upstreams *i* and *j*

Incoming Traffic Engineering ILP Formulation

- Objective function

$$\min f(c_{ai})$$

  - Equal cardinality
  - Equal load
  - Shape bandwidth
  - Equal cost

$$f(c_{ai}) = \max_{i,j \in U} \left(\text{cost}_i - \text{cost}_j \right)$$

$$f(c_{ai}) = \max_{i \in U} \left| \sum_{a \in A-\{t\}} c_{ai} l_a - b_i \right|$$

    - Cost model:

$$\text{cost}_i = \max\left\{ \sum_{a \in A-\{t\}} c_{ai} l_a - \text{thresh}_i, 0 \right\} \text{unit}_i + \text{base}_i$$

Incoming Traffic Engineering ILP Formulation

- Constraints
  - Choice-constraints

$$\forall a \in A - \{t\}, i \in U: c_{ai} = 1 \Rightarrow \forall j \in U - \{i\}: w_j + d_{aj} + (c_{ai} - 1)M > w_i + d_{ai}$$

  - Tie-constraints

$$\forall a \in A - \{t\}, i, j \in U, i \neq j:$$

$$w_i + d_{ai} \geq w_j + d_{aj} + (e_{aij} - 1)M$$

$$w_j + d_{aj} \geq w_i + d_{ai} + (e_{aij} - 1)M$$

$$\forall a \in A - \{t\}, i, j, k \in U, i \neq j:$$

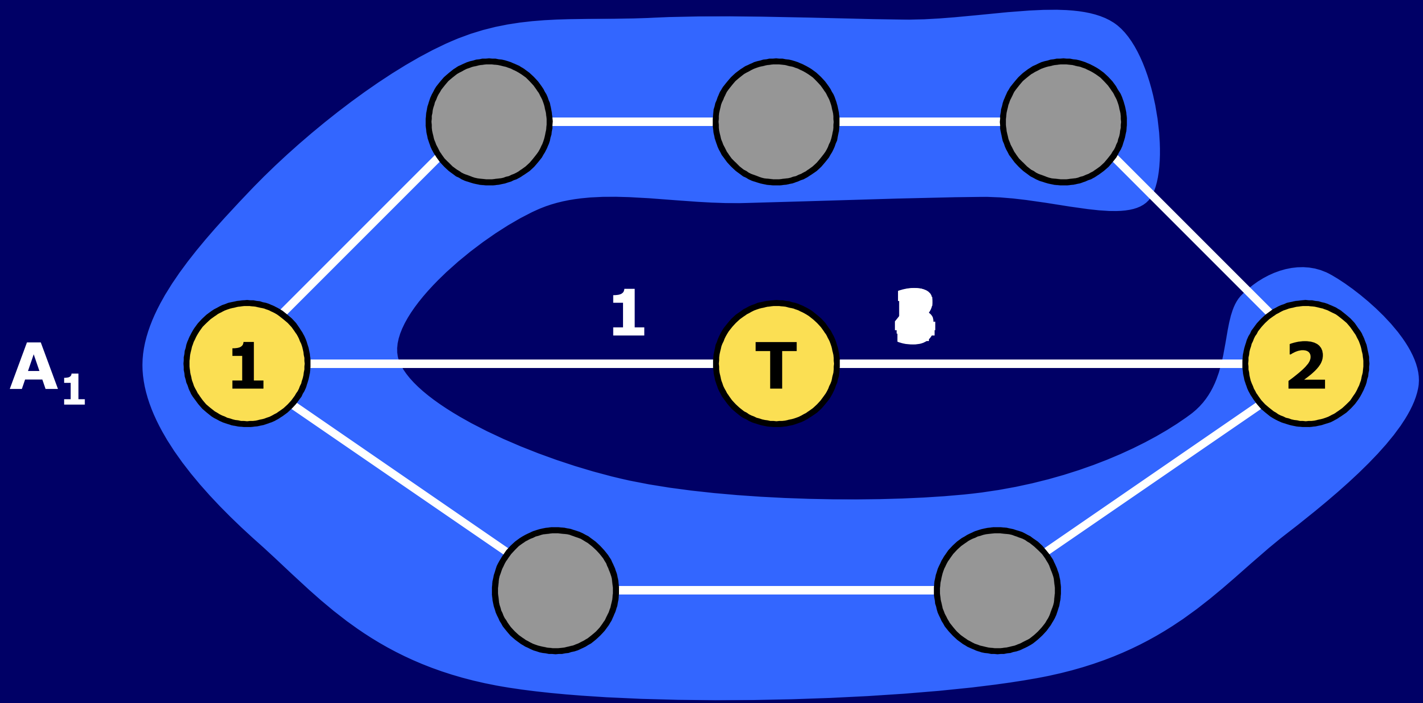
$$w_i + d_{ai} + (e_{aij} - 1)M \leq w_k + d_{ak}$$

  - # of constraints: $(n-1)m^2(m-1)$, $n = |A|, m = |U|$
- What if we consider many prefixes?
  - Variables become vectors

$$\forall i \in U: \text{cost}_i = \text{unit}_i \sum_{a \in A - \{t\}} \sum_{1 \leq k \leq p} c_{ai}[k] \mathbf{l}_a[k]$$

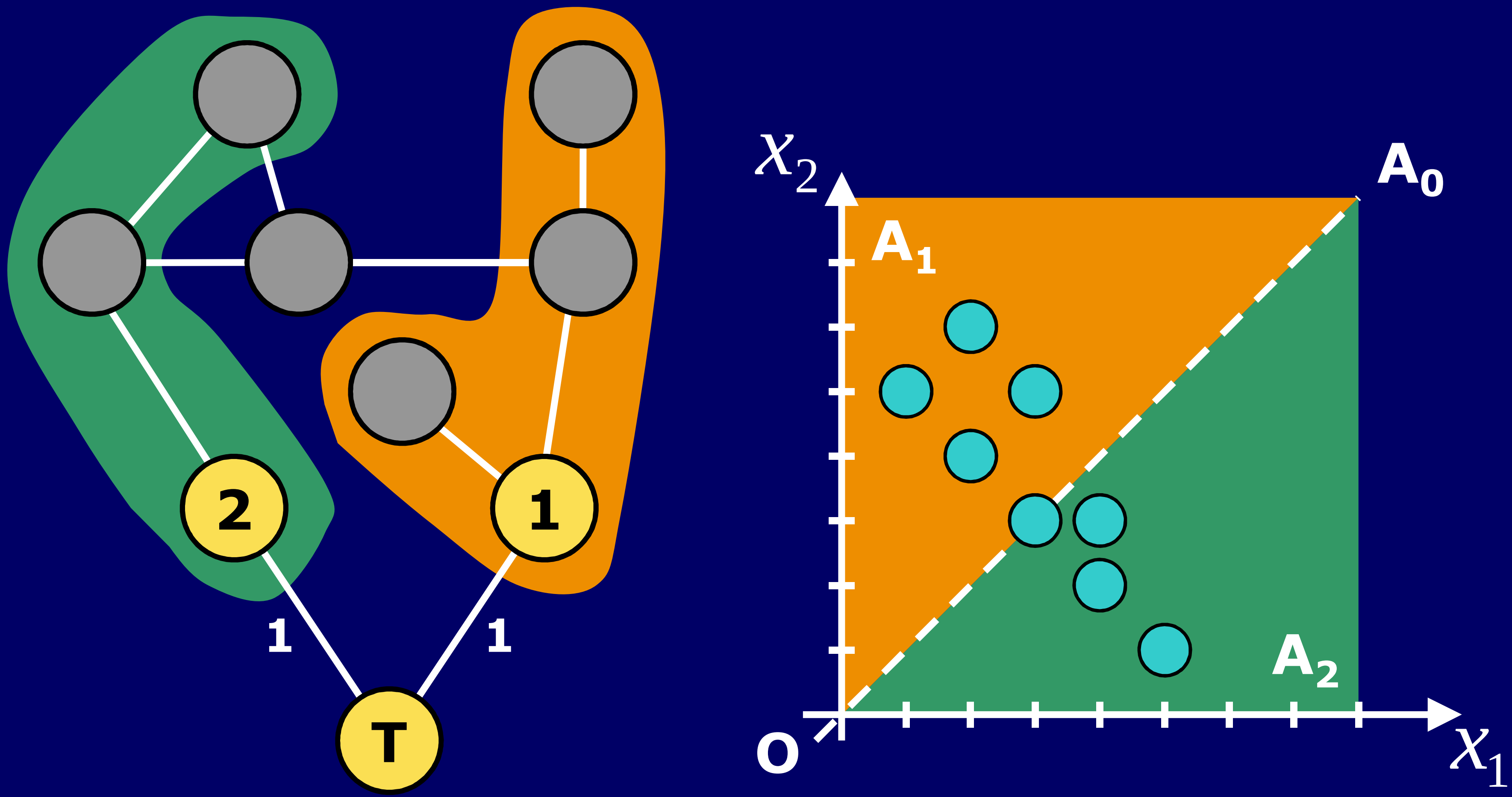
Incoming Traffic Engineering ILP Formulation

- Dealing with ties

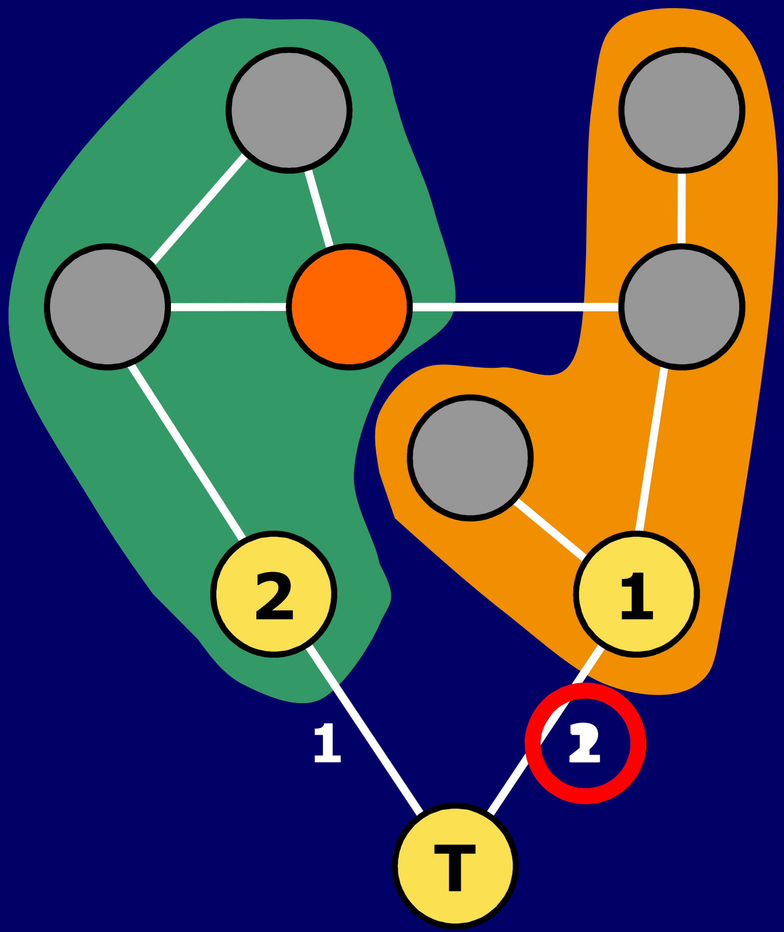


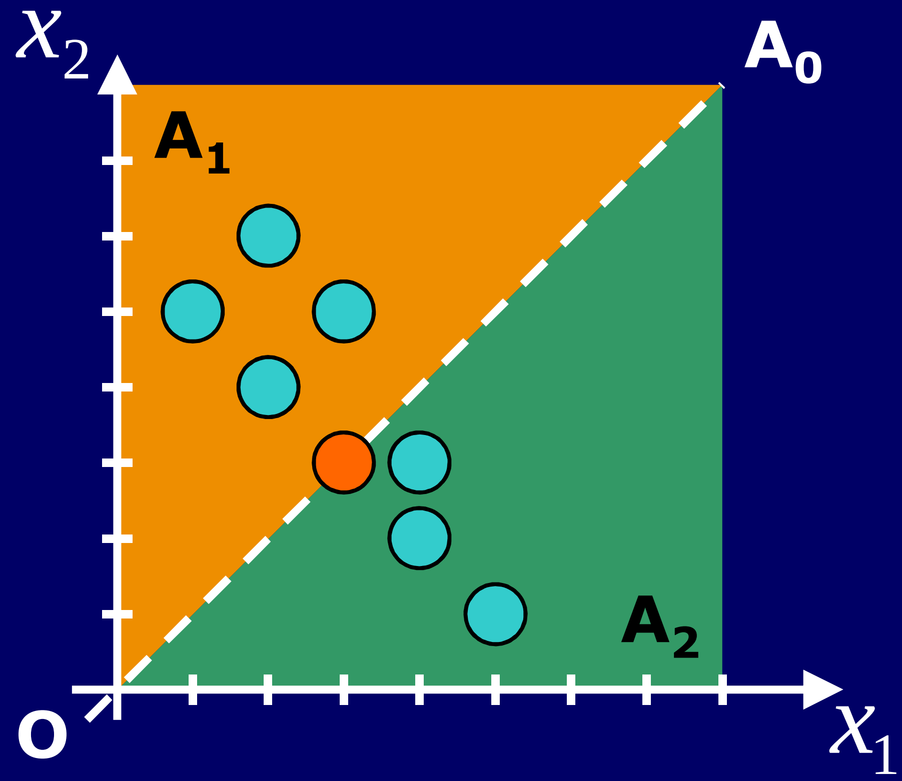
$A_2 = \varnothing$!

Incoming Traffic Engineering
Computational Geometry Approach

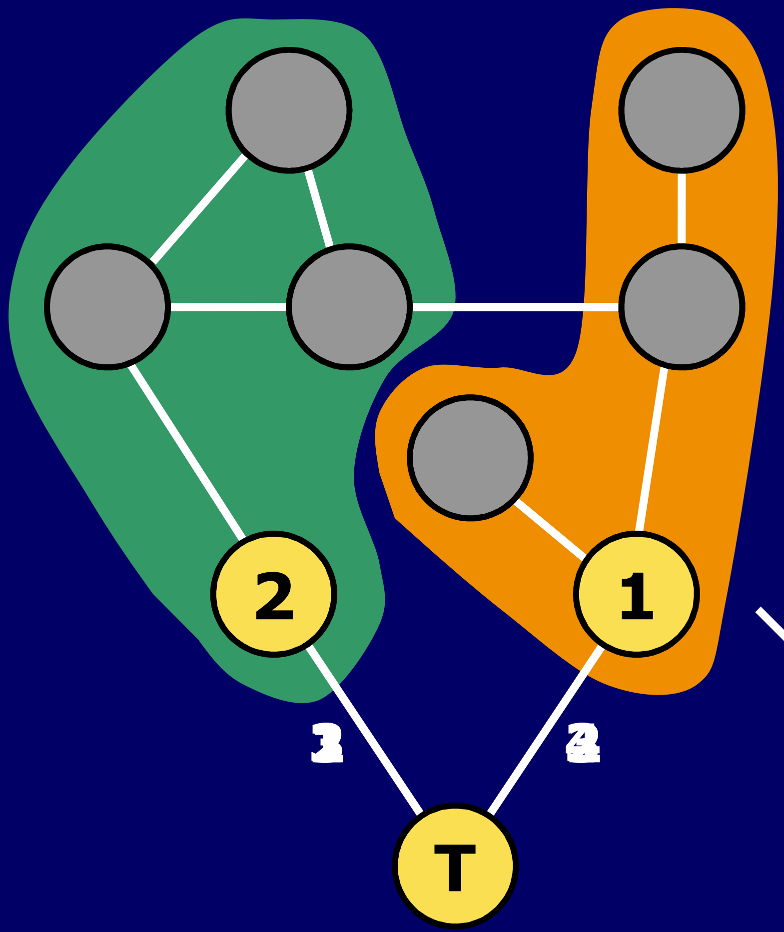


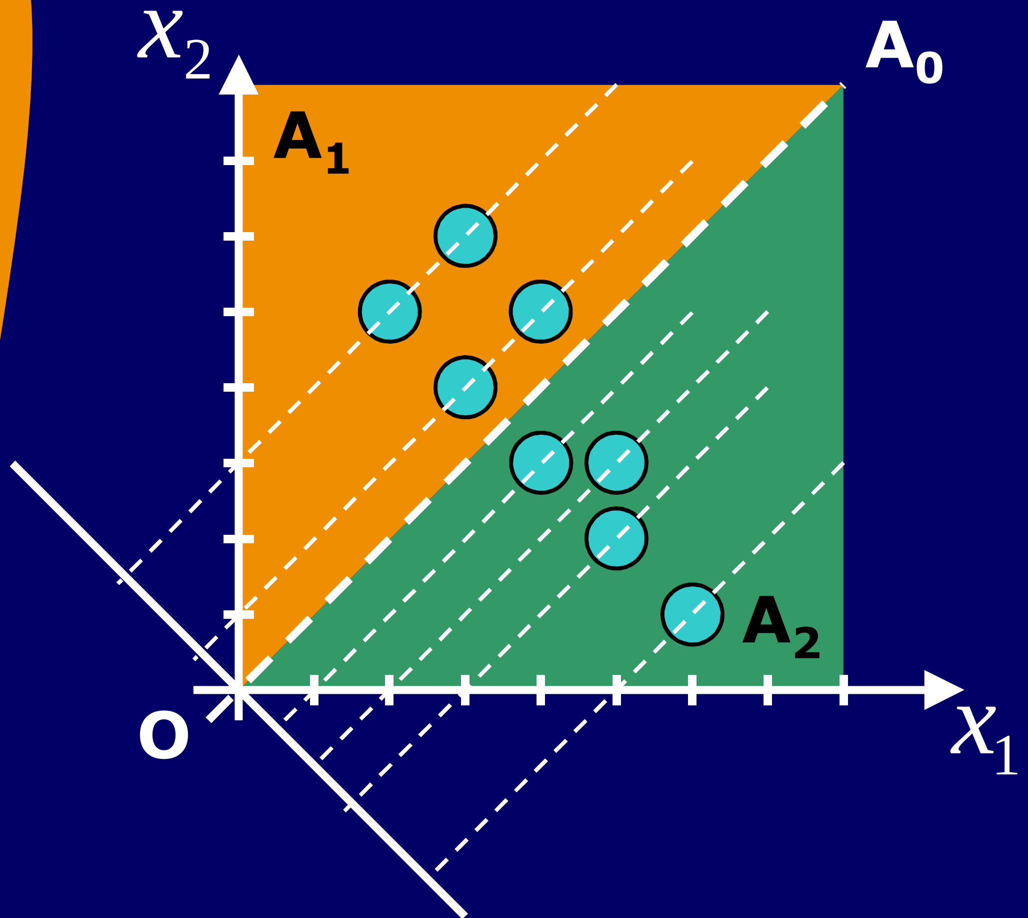
Incoming Traffic Engineering Computational Geometry Approach





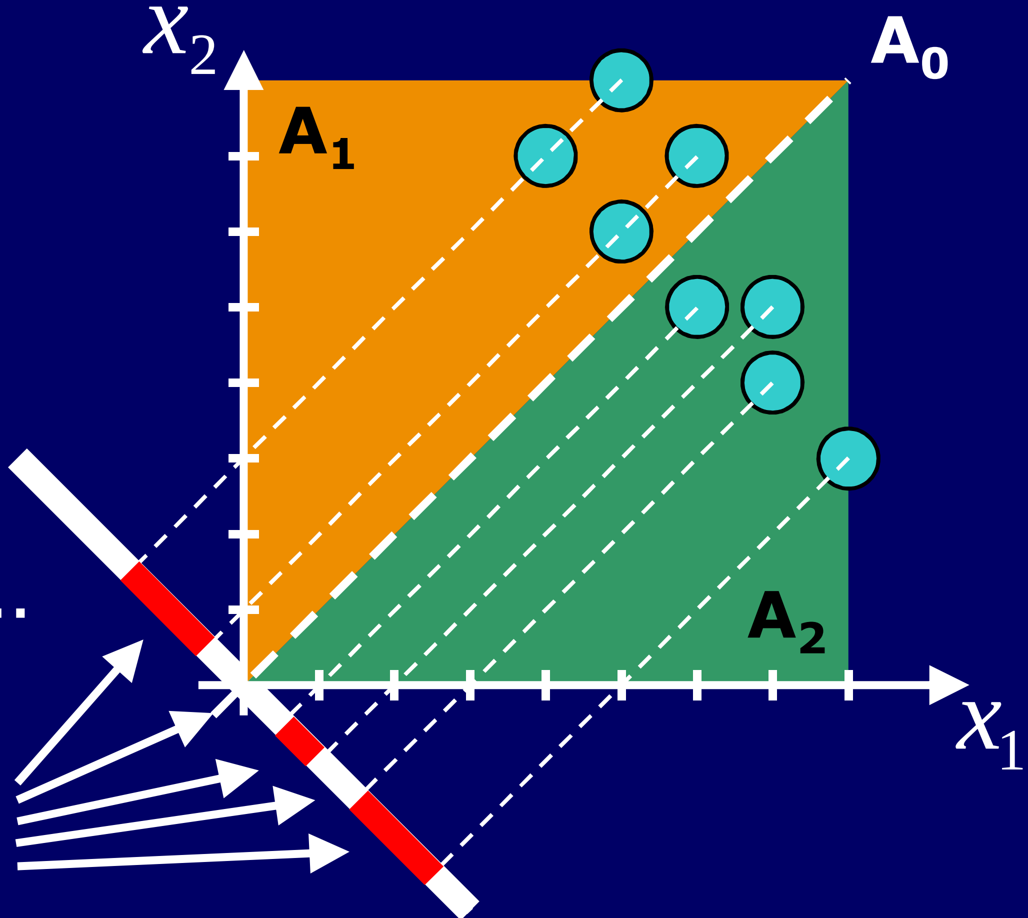
Incoming Traffic Engineering
Computational Geometry Approach





Incoming Traffic Engineering Computational Geometry Approach

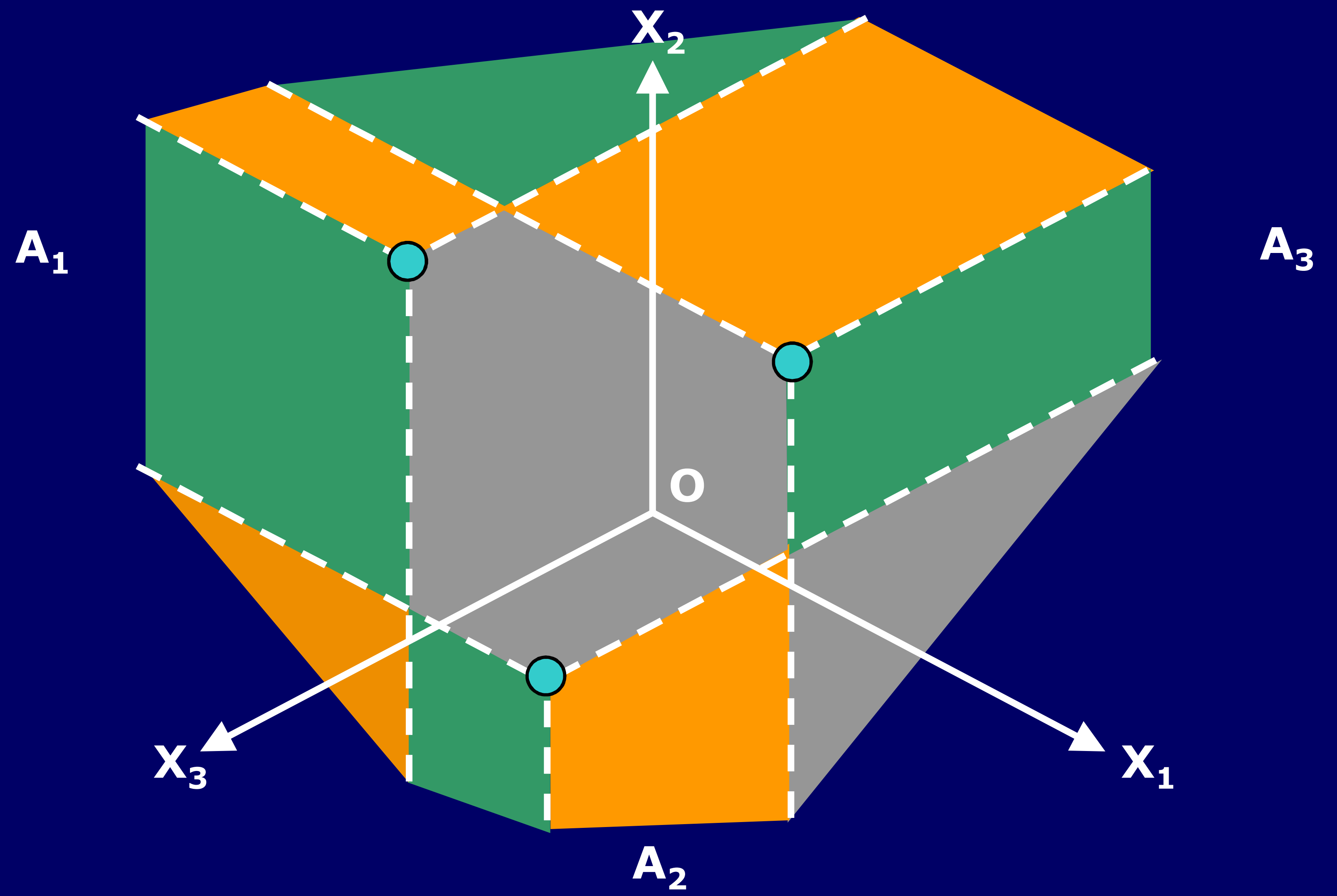
- Projection reduces computational complexity: $O(n^3) \rightarrow O(n^2)$



- Moving the origin…

Equivalence areas

Incoming Traffic Engineering Computational Geometry Approach



Incoming Traffic Engineering Computational Geometry Approach

- How to find optimal prepending?
  1. assign prep… ng 1 to every upstream
  2. compu… s from every AS to *t* through the upstreams
  3. plot ASes as points in an (hyper)space
  4. identify equivalence areas
  5. for each equivalence area A:
     5.1. place origin O in a point P$\in$A
     5.2. compute objective function
  6. pick the best prepending assignment

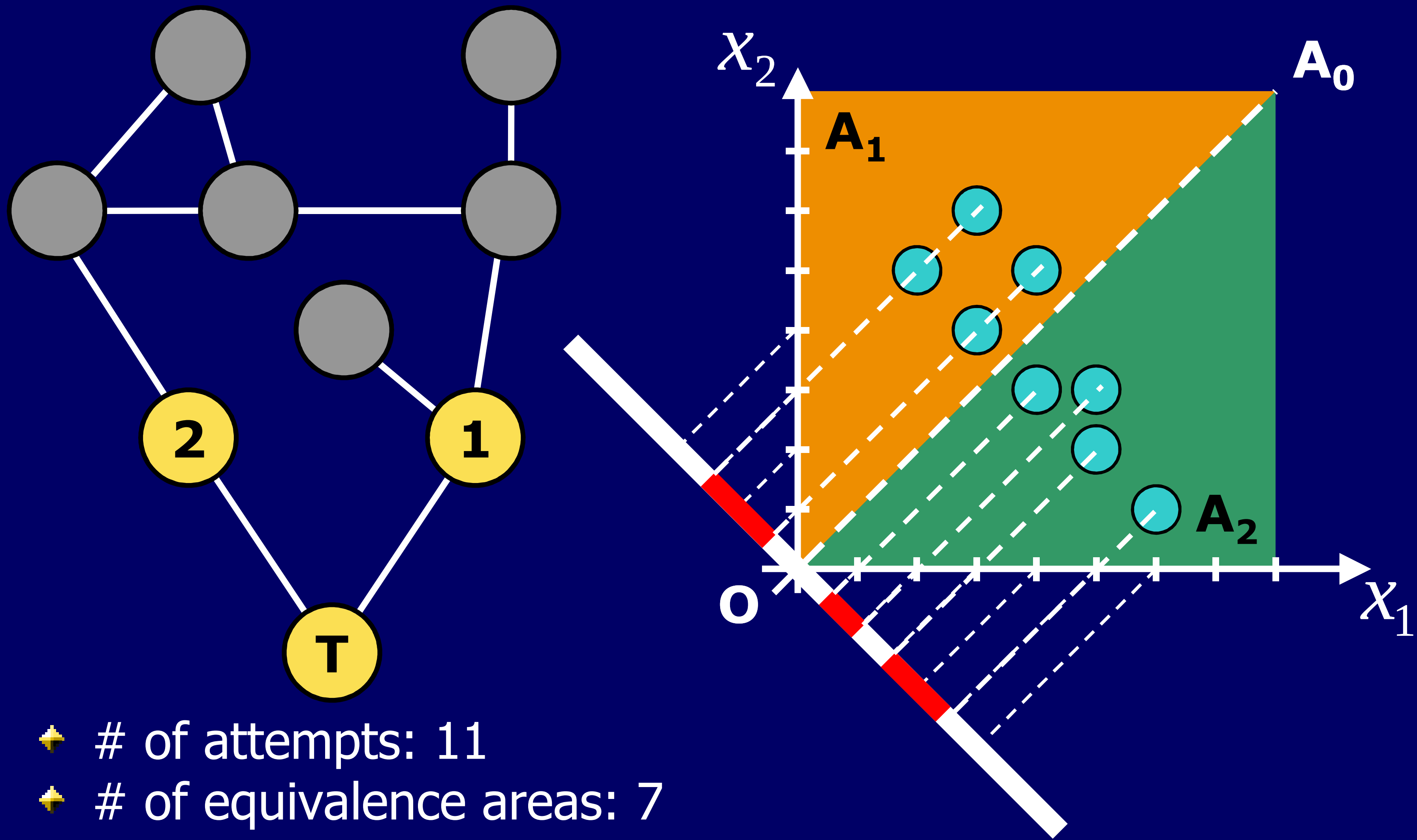
wrt the aforementioned objective functions

Incoming Traffic Engineering
Computational Geometry Approach

- What about complexity?
  - # of equivalence areas (attempts): $O(n^2)$, n=|A|
  - $\Rightarrow$ no improvement, but…
- …ASes that send most of the traffic to *t* are (usually) only a few…
  - …so what?

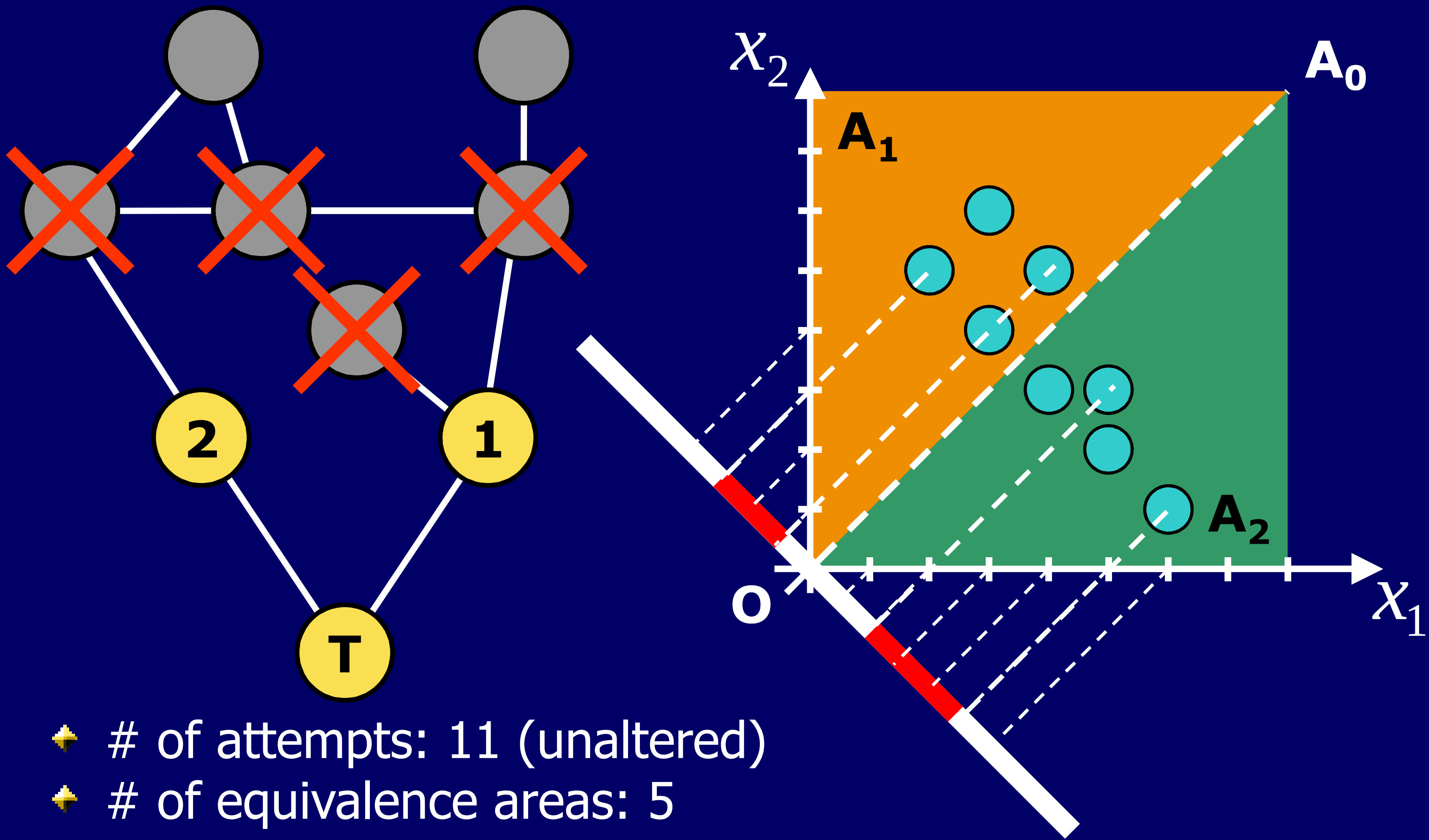
1. using a prepending > $\max d_{ai}$ is pointless
2. just consider top traffic senders $\Rightarrow$ few ASes with large d_{ai}
3. few ASes $\Rightarrow$ few equivalence areas

Incoming Traffic Engineering Computational Geometry Approach



- # of attempts: 11
- # of equivalence areas: 7

Incoming Traffic Engineering Computational Geometry Approach



- # of attempts: 11 (unaltered)
- # of equivalence areas: 5

Concluding Remarks

- A theoretical approach to TE
  - ILP formulation ⇒ many constraints
  - Computational geometry ⇒ efficient algorithms
- Applicability
  - knowing the network topology is not needed, but how to find d_{ai}?
    - announce a prefix at one upstream at a time
    - restrict to top traffic senders
    - reverse traceroute results
- Future work
  -
  - does the prepending game admit a (Nash) equilibrium?