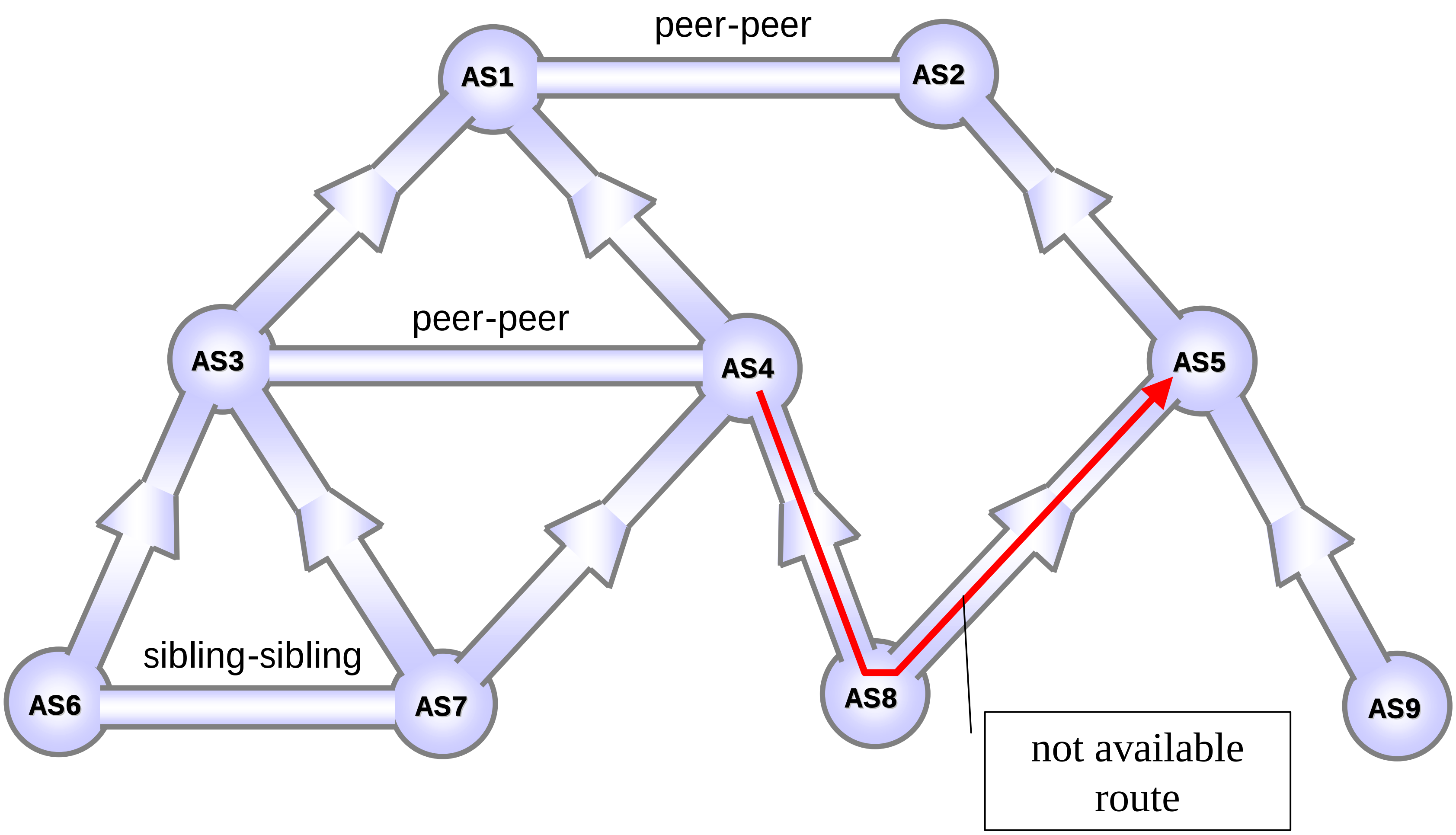
computing the relationships between autonomous systems

giuseppe di battista
maurizio patrignani
maurizio pizzonia

univ. of rome III
http://www.dia.uniroma3.it/~compunet/

the scenario

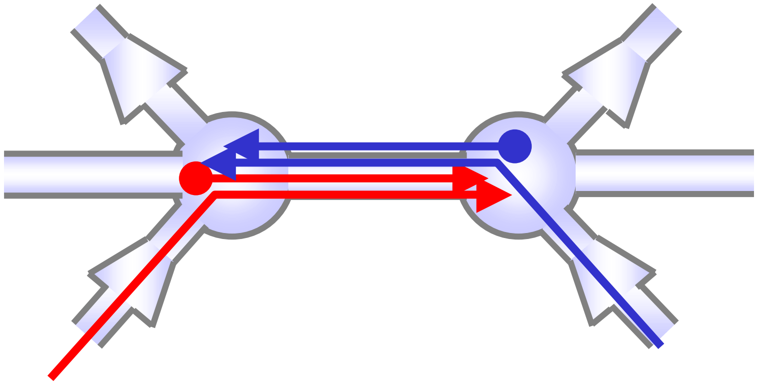


problem history

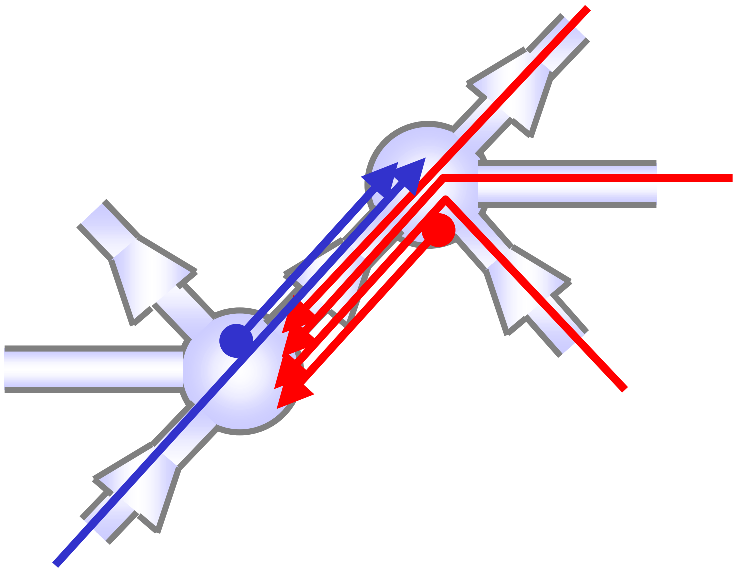
- the problem is introduced by Lixin Gao ("On Inferring Autonomous System Relationships in the Internet", IEEE Trans. Networking, 2001)
- relationships are classified into three categories: customer-provider, peer-peer, and sibling-sibling
- BGP routing tables are used as input
- heuristics are verified with information coming from other sources
- further studies on the structural properties of the labeled graph (Ge, Figueiredo, Jaiwal, Gao, "On the Hierarchical Structure of the Logical Internet Graph", Proc. SPIE ITCOM, 2001)

a simpler problem formulation

due to Subramanian, Agarwal, Rexford, and Katz ("Characterizing the Internet Hierarchy from Multiple Vantage Points" INFOCOM 2002) [SARK 2002]



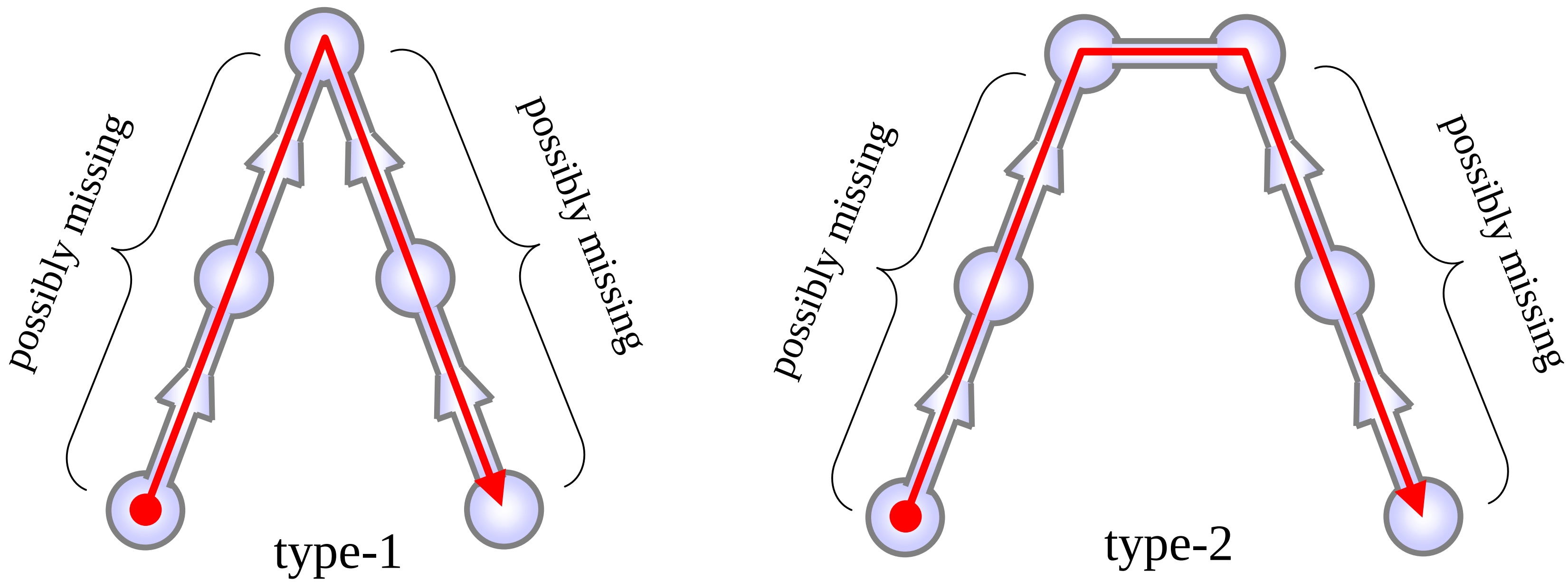
peer-peer



customer-provider

valid and invalid AS-paths

if all ASes respect the advertisement rules, AS-path would be all *valid*



type-1 valid AS-path: a (possibly missing) uphill path followed by a (possibly missing) downhill path

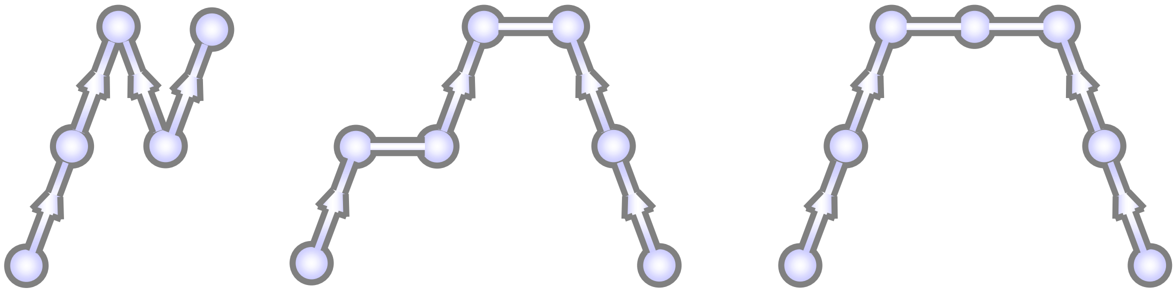
type-2 valid AS-path: a (possibly missing) uphill path followed by a peer-peer edge, followed by a (possibly missing) downhill path

the ToR (Type of Relationships) problem

ToR problem [SARK 2002]

given an undirected graph G and a set of paths P, give an orientation to **some** of the edges of G to minimize the number of invalid paths in P

invalid paths: not "valley free" or involving more than one peering



in [SARK 2002] the ToR problem is conjectured to be NP-hard

a real-life ToR problem instance

six rows extracted from the BGP routing table of Oregon RouteViews (Apr 18, 2001)

network	next hop	path
200.1.225.0	167.142.3.6	5056 701 6461 4926 4270 4387 i
200.10.112.0/23	167.142.3.6	5056 701 4926 4926 4926 6461 2914 174 174 174 174 14318 i
204.71.2.0	203.181.248.233	7660 1 5056 701 11334 i
231.172.64.0/19	167.142.3.6	5056 1239 1 1755 1755 1755 1755 3216 13099 i
200.33.121.0	167.142.3.6	5056 1 1239 8151 i
204.71.2.0	144.228.241.81	1239 5056 701 11334 i

extract AS paths and eliminate prepending

a real-life ToR problem instance

six rows extracted from the BGP routing table of Oregon RouteViews (Apr 18, 2001)

network	next hop	path
200.1.225.0	167.142.3.6	5056 701 6461 4926 4270 4387 i
200.10.112.0/23	167.142.3.6	5056 701 4926 4926 4926 6461 2914 174 174 174 174 14318 i
204.71.2.0	203.181.248.233	7660 1 5056 701 11334 i
231.172.64.0/19	167.142.3.6	5056 1239 1 1755 1755 1755 1755 3216 13099 i
200.33.121.0	167.142.3.6	5056 1 1239 8151 i
204.71.2.0	144.228.241.81	1239 5056 701 11334 i

extract AS paths and eliminate prepending

5056 701 6461 4926 4270 4387

5056 701 4926 4926 4926 6461 2914 174 1 14318

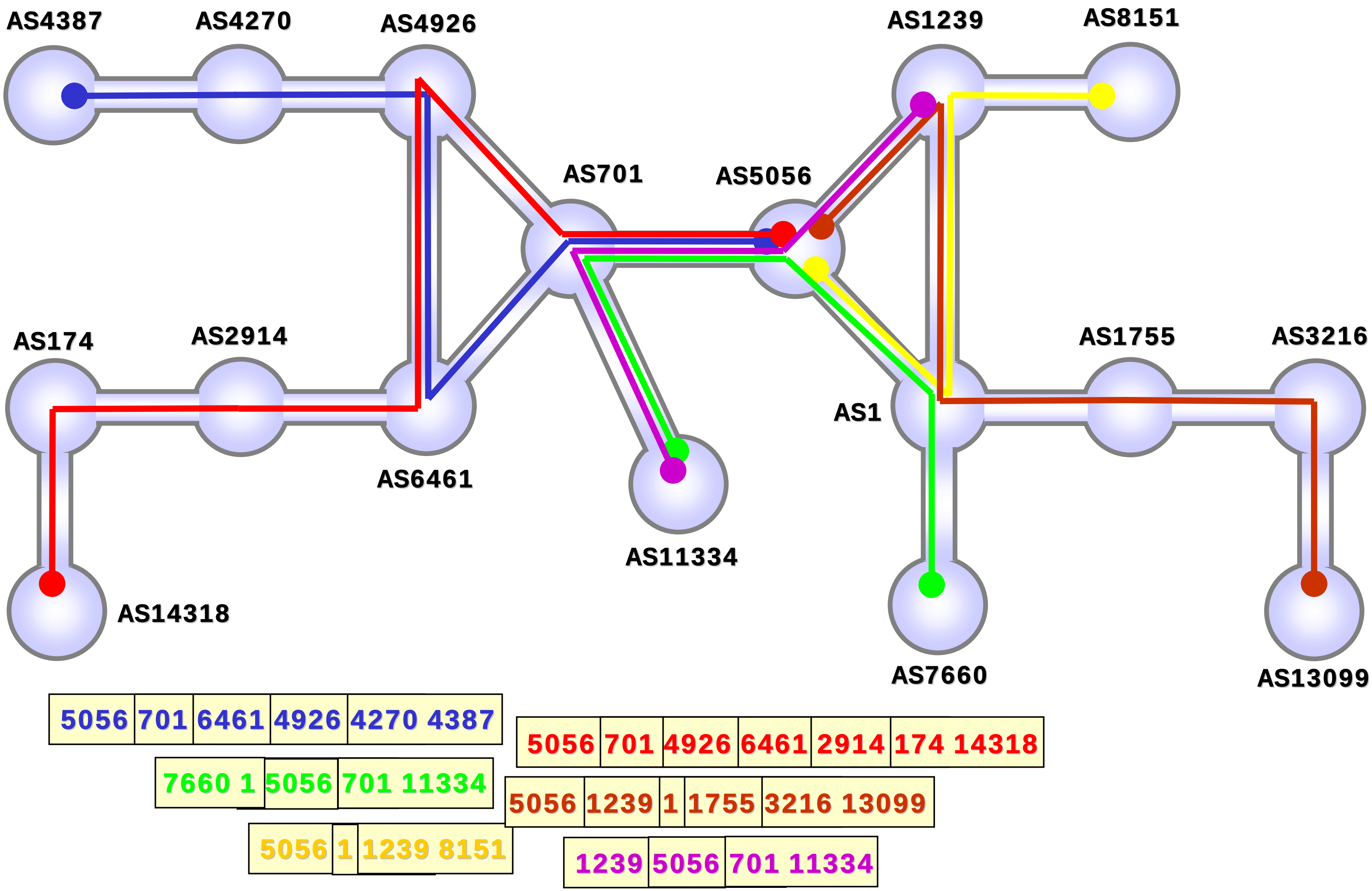
7660 1 5056 701 11334

5056 1239 1 1755 1755 1755 1755 3216 13099

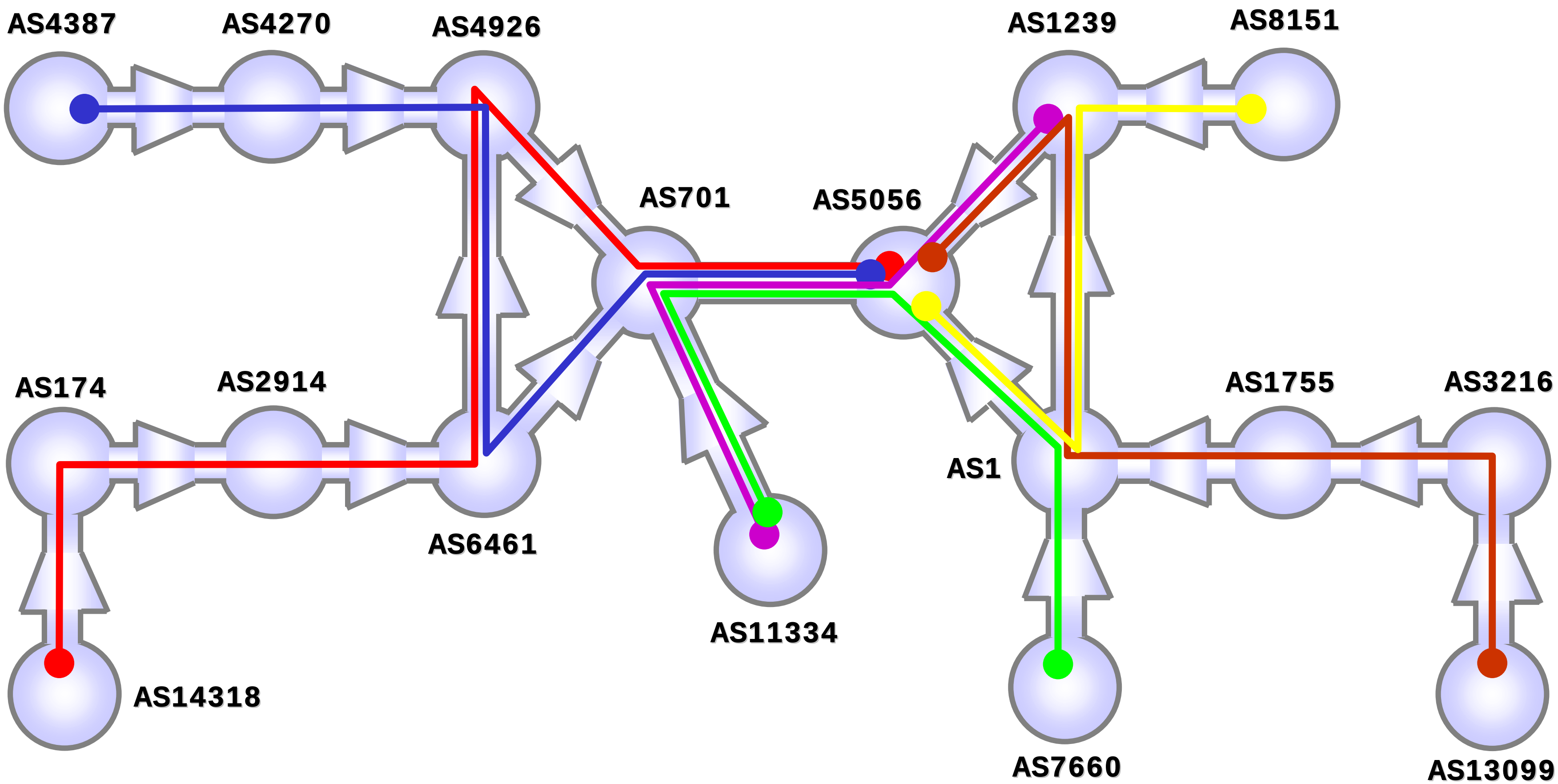
5056 1 1239 8151

1239 5056 701 11334

building the corresponding AS graph



an orientation for the AS graph



5056 701 6461 4926 4270 4387

5056 701 4926 6461 2914 174 14318

7660 1 5056 701 11334

5056 1239 1 1755 3216 13099

5056 1 1239 8151

1239 5056 701 11334

our contributions

- we show that, although the ToR-problem is NP-hard, a solution without invalid paths (if it exists) can be found in linear time
- we propose heuristics for the general problem based on a novel paradigm and show their effectiveness against publicly available data sets
- the experiments put in evidence that our heuristics performs significantly better than state of the art heuristics

a hot topic

- independently of our work, Erlebach, Hall, and Schank of the Theory of Communication Networks Group of Zurich discovered analogous results concerning the time complexity of the general problem and the linearity in the case of all valid paths
- however, while they put more emphasis on the approximability of the problem, we focus more on the engineering and the experimentation of an effective heuristic approach

formulation as a decision problem

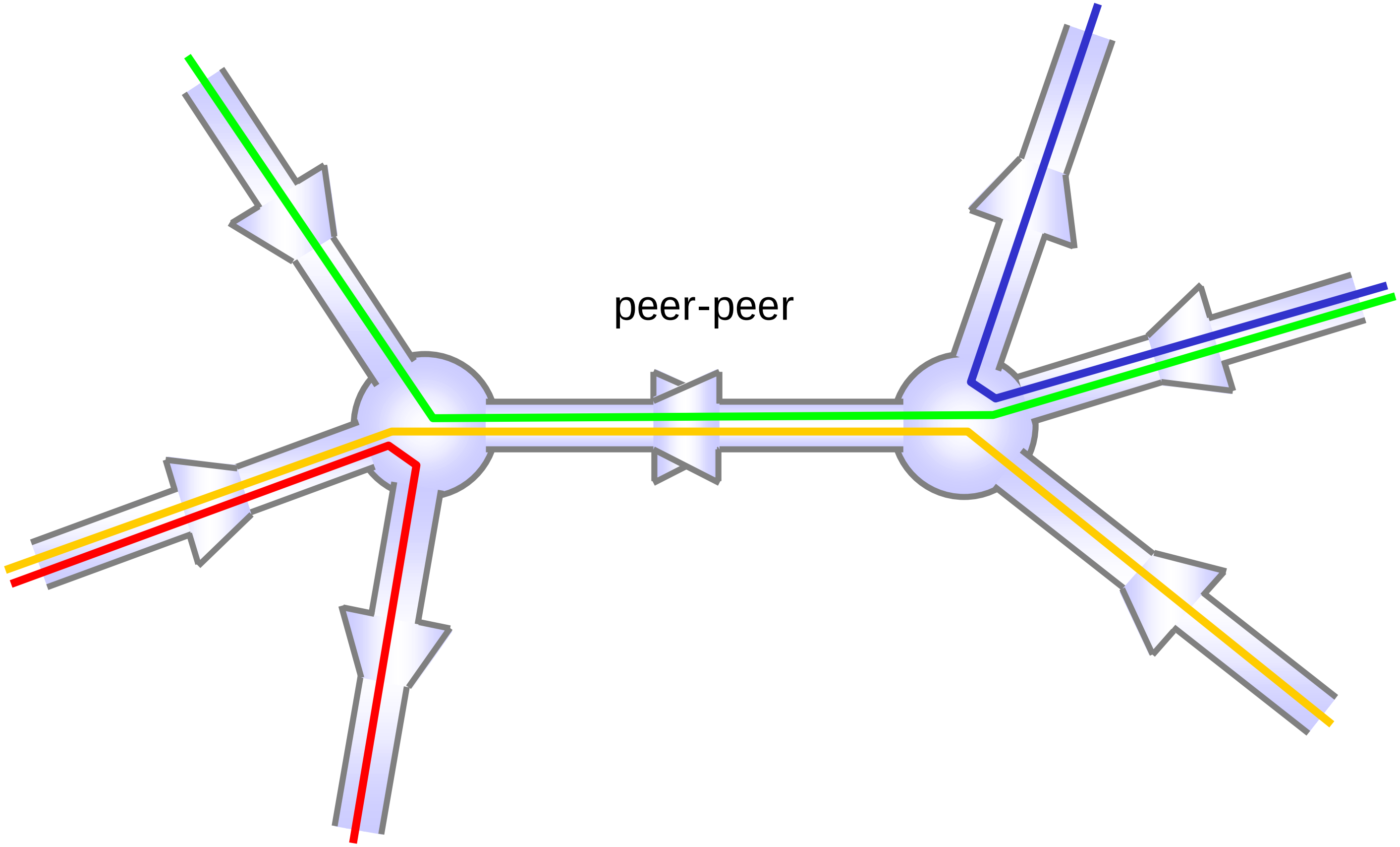
- the ToR minimization problem corresponds to the following decision problem

ToR-D problem

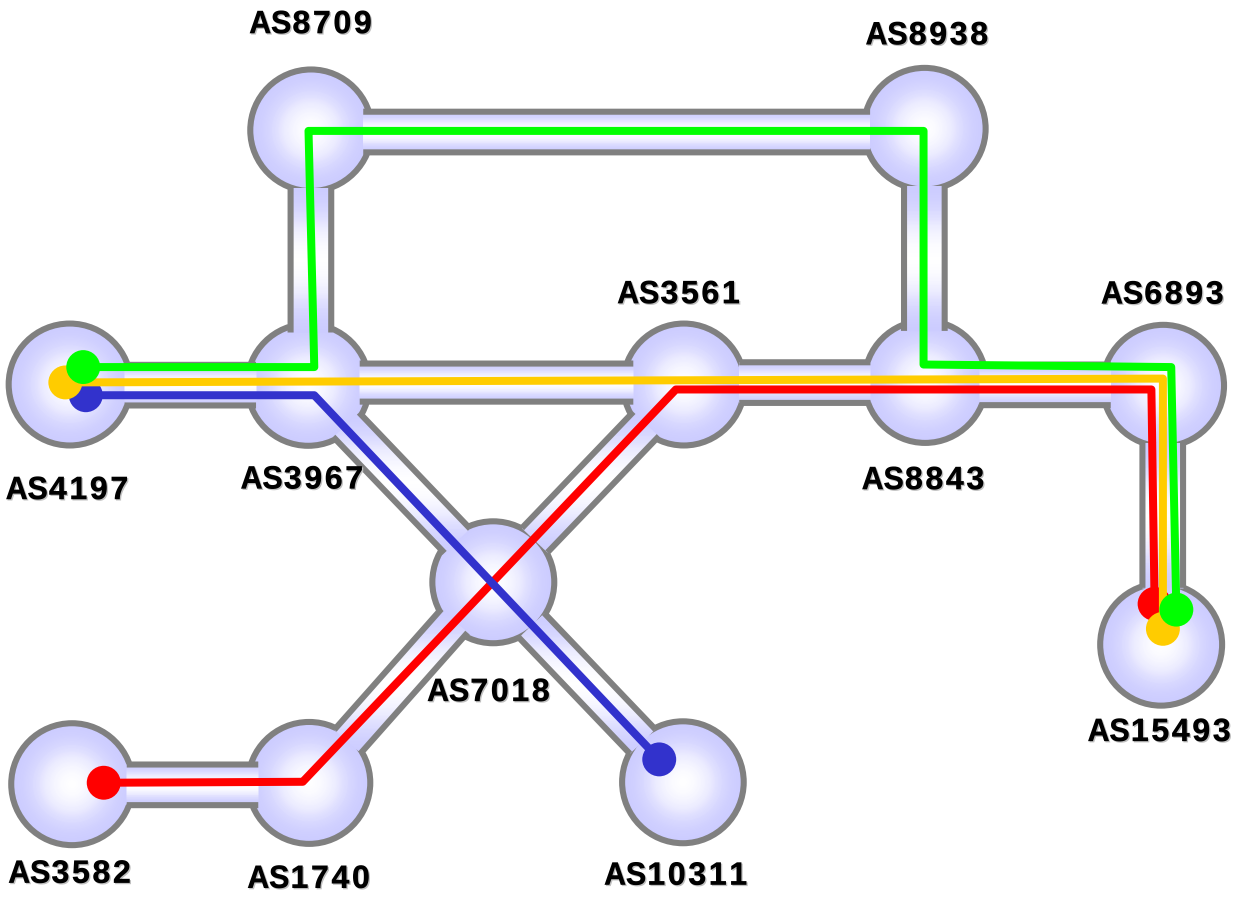
given an undirected graph G, a set of paths P, and an integer k, test if it is possible to give an orientation to **some** of the edges of G so that number of invalid paths in P is at most k

simplifying the decision problem

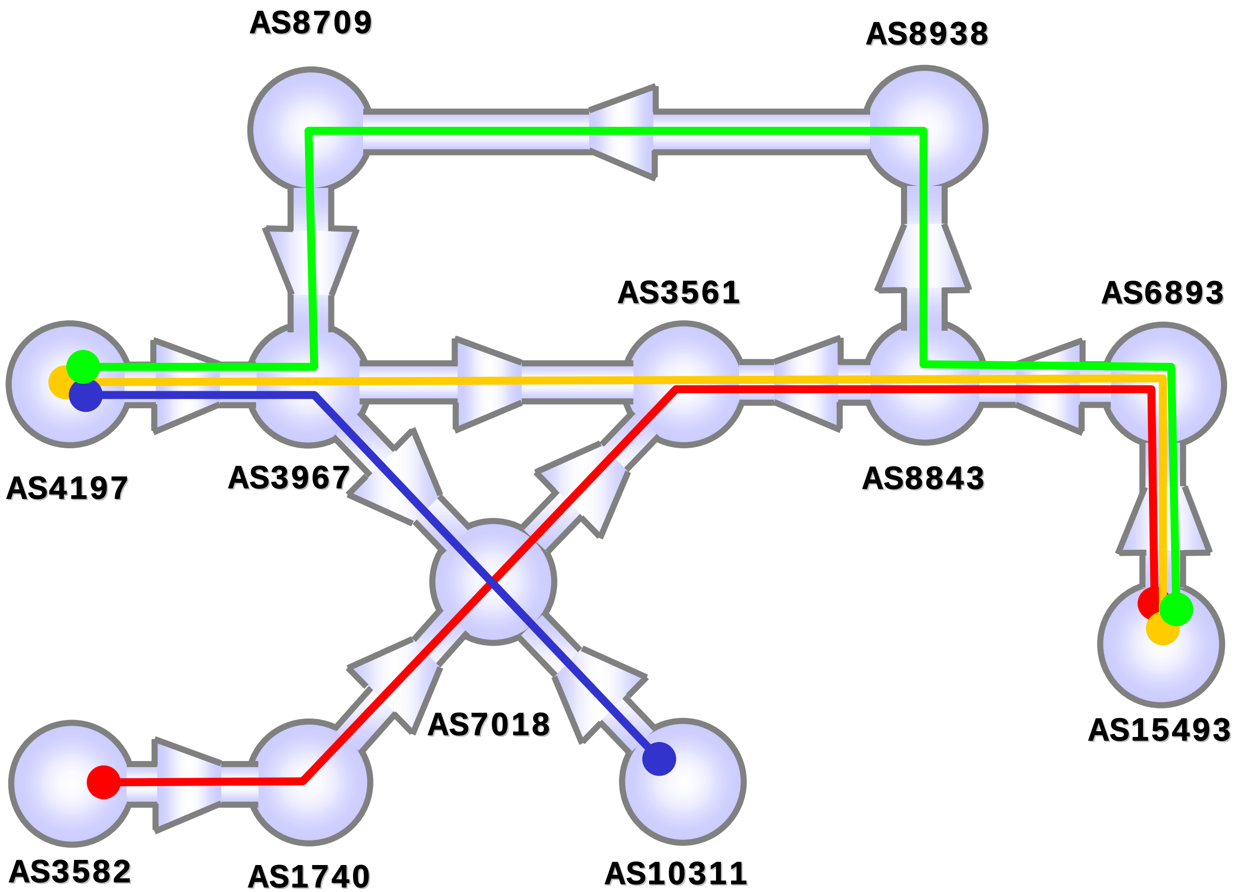
suppose to have a solution to the TOR problem with all valid paths and consider an edge labeled as a peer-peer edge



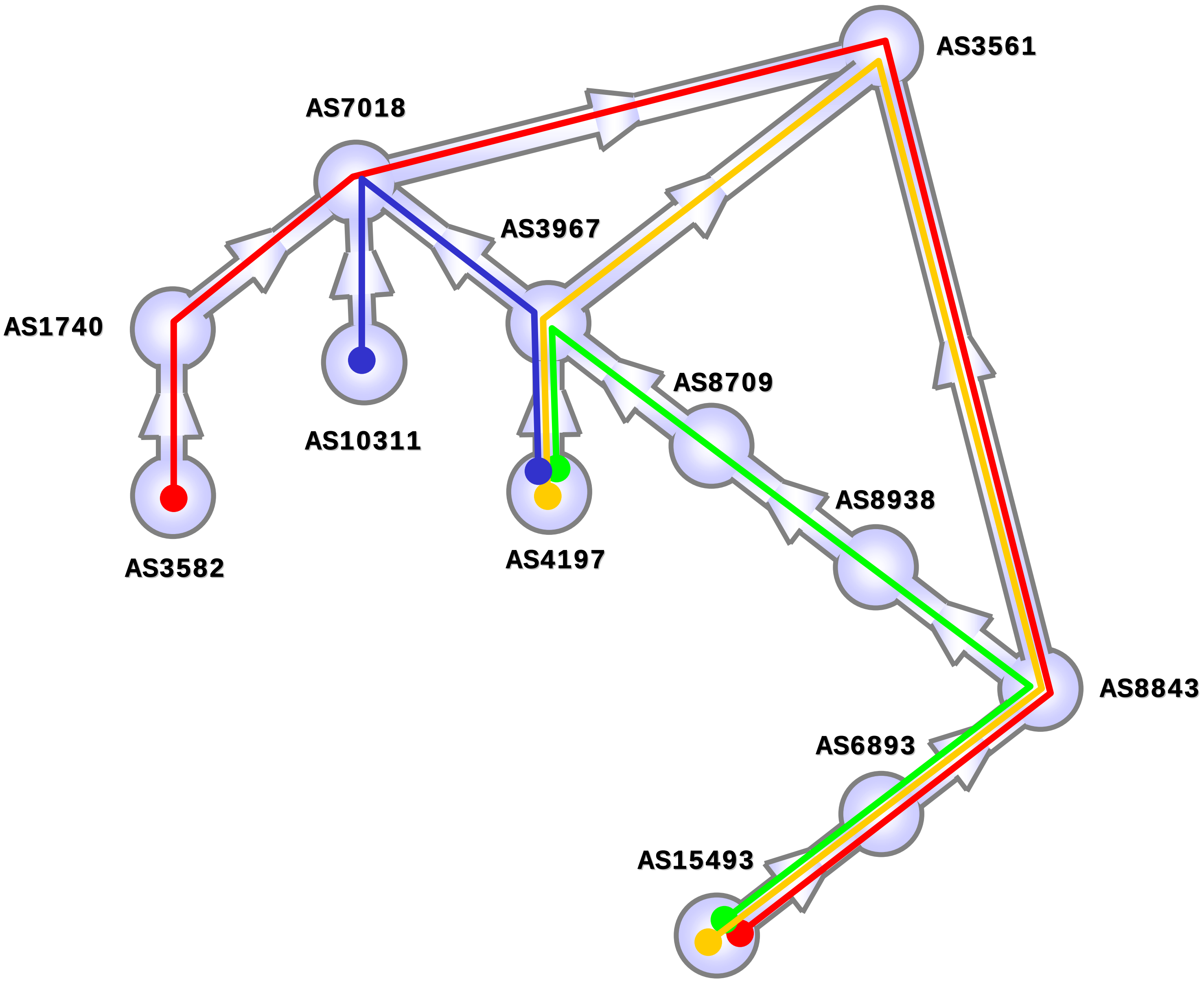
example of orientable AS graph



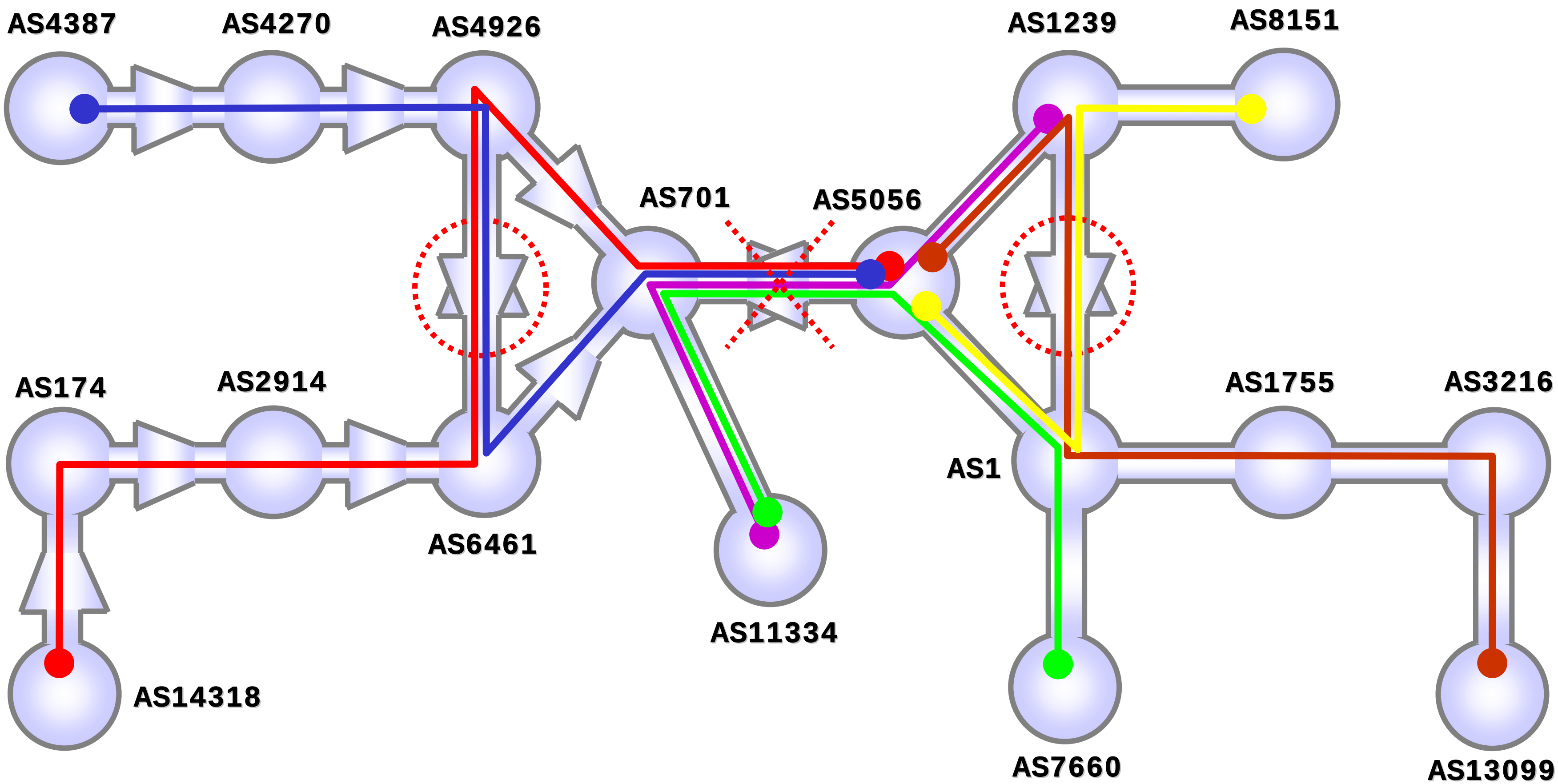
an orientation leaving all valid paths



a different representation



example of not orientable AS graph



5056 701 6461 4926 4270 4387

5056 701 4926 6461 2914 174 14318

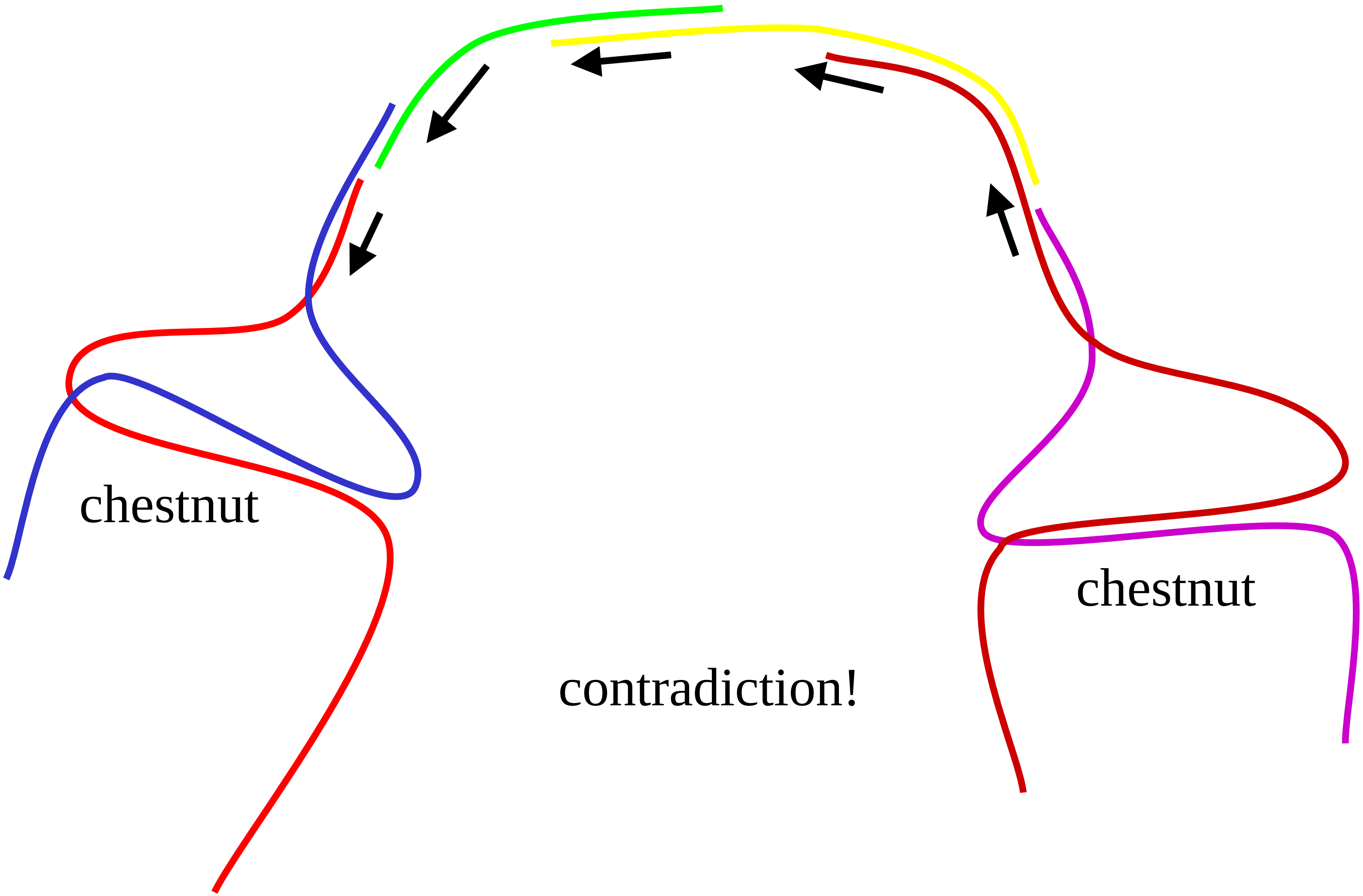
7660 1 5056 701 11334

5056 1239 1 1755 3216 13099

5056 1 1239 8151

1239 5056 701 11334

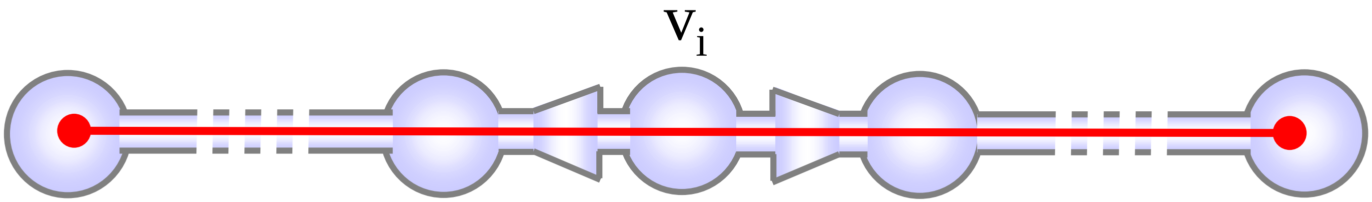
“*chestnuts*” and contradictions



testing if a solution exists with all valid paths

observation

a path $p = v_1, \ldots, v_n$ is valid if and only if it does not have a vertex v_i such that the two edges of p incident on v_i are directed away form v_i



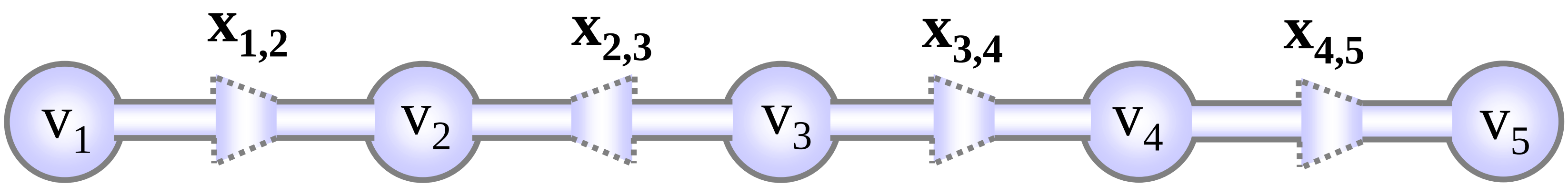
based on this observation the problem can be mapped to 2SAT:

given a set X of Boolean variables and a formula in conjunctive normal form composed by clauses of two literals, where a literal is a variable or a negated variable, find a truth assignment for the Boolean variables in X such that the formula is satisfied

example: $(x_1 \vee x_2) \;\wedge\; (\neg x_2 \vee \neg x_3) \;\wedge\; (x_3 \vee \neg x_1)$

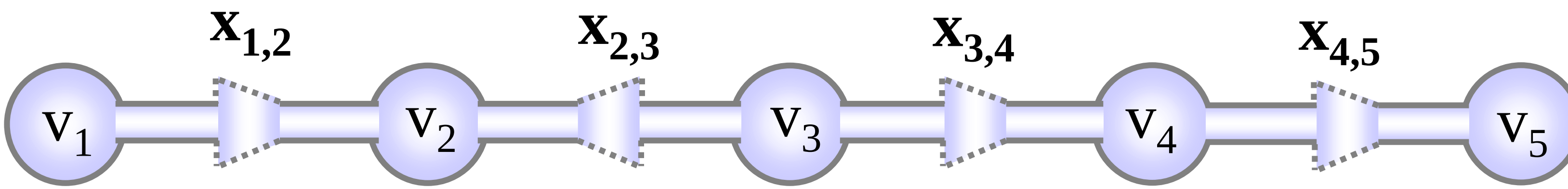
mapping the problem to 2SAT

- associate each edge with a Boolean variable
- provide each edge with an arbitrary (say random) orientation

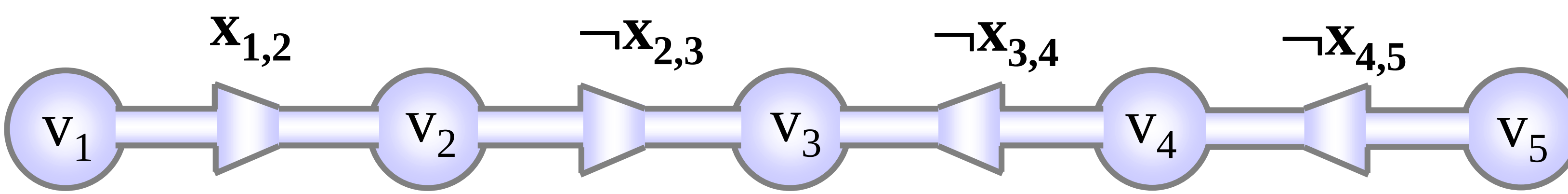


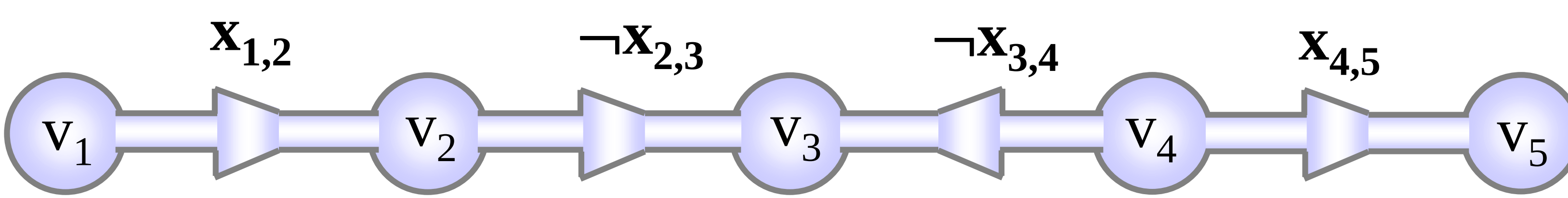
- a truth assignment for the Boolean variables corresponds to an orientation for all the edges of the AS-graph (and vice versa)
  - if the variable is true, the associated edge preserves its original direction
  - if the variable is false, the associated edge is reversed

construction of the 2SAT instance



$$(x_{1,2} \vee x_{2,3}) \quad \wedge \quad (\neg x_{2,3} \vee \neg x_{3,4}) \quad \wedge \quad (x_{3,4} \vee \neg x_{4,5})$$



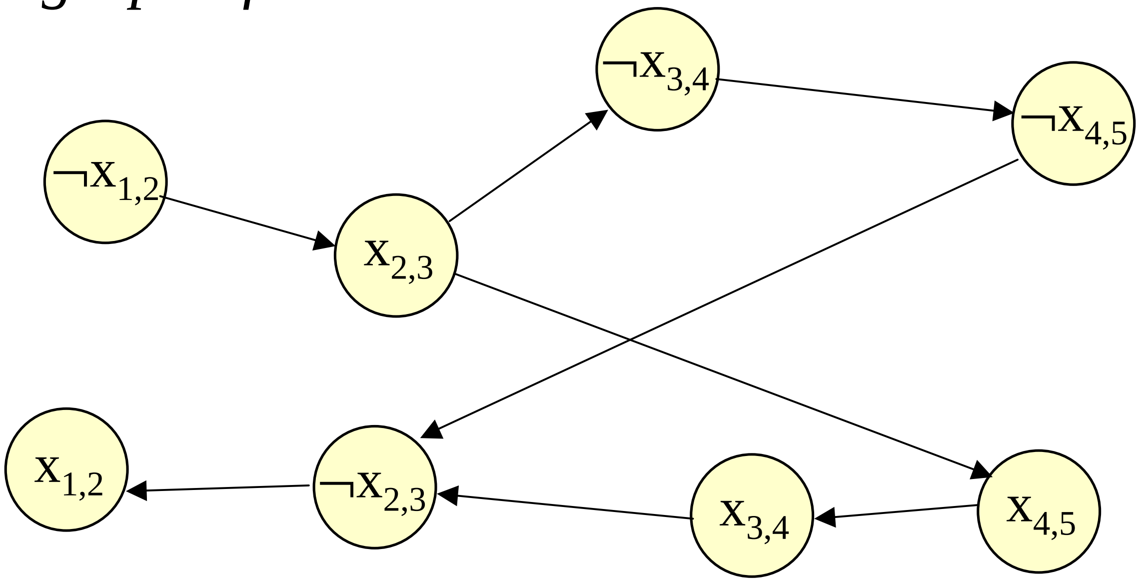


solving a huge 2SAT

given a 2SAT formula

$$(x_{1,2} \vee x_{2,3}) \wedge (\neg x_{2,3} \vee \neg x_{3,4}) \wedge (x_{3,4} \vee \neg x_{4,5}) \wedge (x_{4,5} \vee \neg x_{2,3})$$

compute the *graph of the literals*

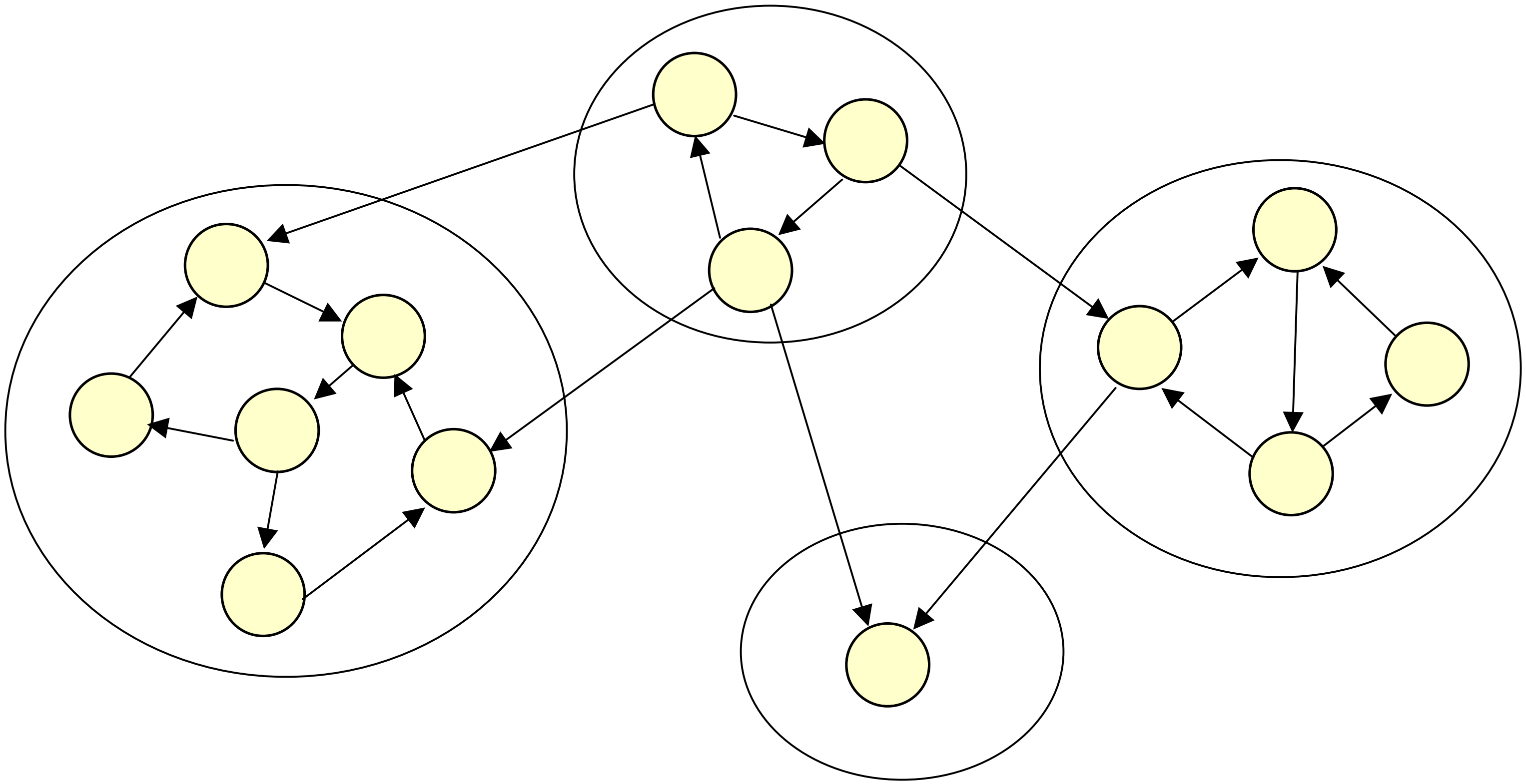


there is a direct path between a literal and its opposite and vice versa iff a solution without invalid paths does not exist (Aspvall, Plass, Tarjan, IPL '79)

strongly connected components

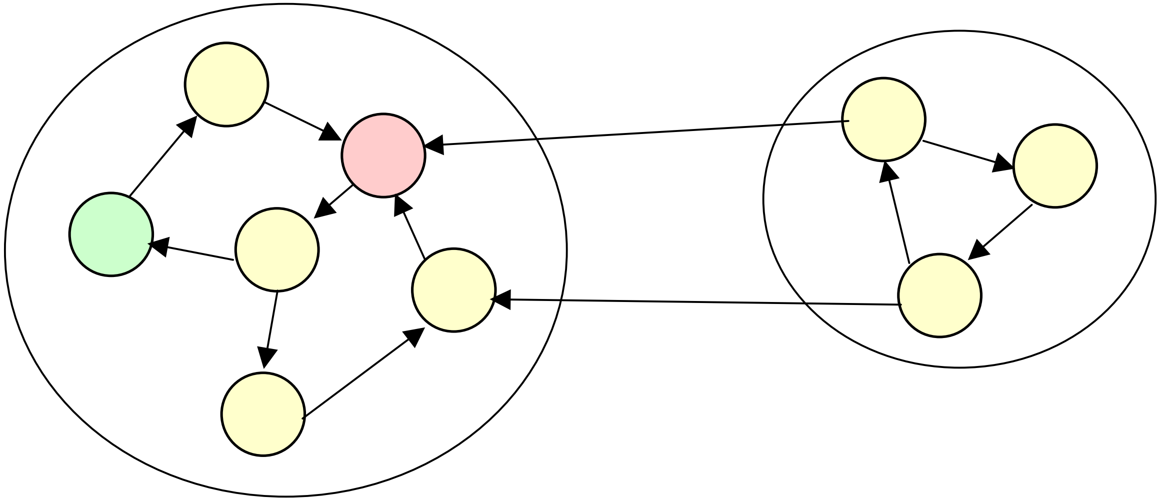
compute the strongly connected components of the graph of the literals

strongly connected component = maximal set of vertices such that for each pair *u*, *v* of vertices of the set there exists a directed path from *u* to *v* and vice versa



testing if a solution exists

- a solution for the 2SAT problem does not exists if and only if the two literals of the same Boolean variable (edge) fall into the same strongly connected component
- the test takes O(n+m+q) time
  - where n is the number of ASes, m is the number of edges, and q is the sum of the lengths of the AS-paths



the experimental setting

- same setting used by [SARK 2002]
- ten telnet looking glasses are selected as sources of data
- all the AS-paths put together are used to test the algorithms
- four web sources are used to validate the output

AS graphs from the various sources

http://www.cs.berkeley.edu/~sagarwal/research/BGP-hierarchy/

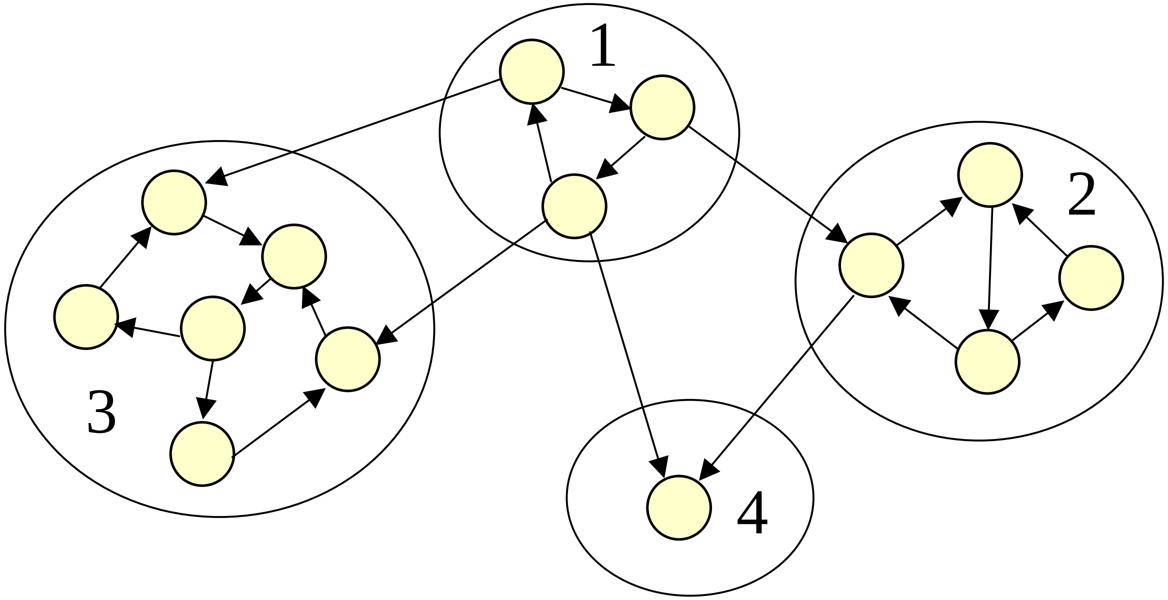
Source	AS #	AS Name	Apr 18, 2001			Apr 6, 2002		
			Paths	**Vertices**	**Edges**	**Paths**	**Vertices**	**Edges**
telnet sources	1	Genuity	58,156	10,203	13,001	63,744	12,700	15,946
	1740	CERFnet	70,830	10,007	13,416	*not available*		
	3549	Globalcrossing	60,409	10,288	13,039	76,572	12,533	16,025
	3582	Univ.of Oregon	2,584,230	10,826	22,440	4,600,981	13,055	27,277
	3967	Exodus Comm.	254,123	10,387	18,401	399,023	12,616	21,527
	4197	Global Online Japan	55,060	10,288	13,004	59,745	12,518	15,628
	5388	Energis Squared	58,832	10,411	13,259	117,003	12,659	16,822
	7018	AT&T	120,283	9,252	12,117	170,325	11,706	15,429
	8220	COLT Internet	46,606	8,376	10,932	154,855	12,660	18,421
	8709	Exodus, Europe	114,931	10,333	15,006	126,370	12,555	18,175
web sources	1755	Ebone	23,469	10,540	13,898	*not available*		
	2516	KDDI	126,414	10,790	17,735	*not available*		
	2548	MaeWest	80,549	10,583	15,249	*not available*		
	6893	CW	70,265	10,523	14,359	*not available*		

solvable without invalid paths?

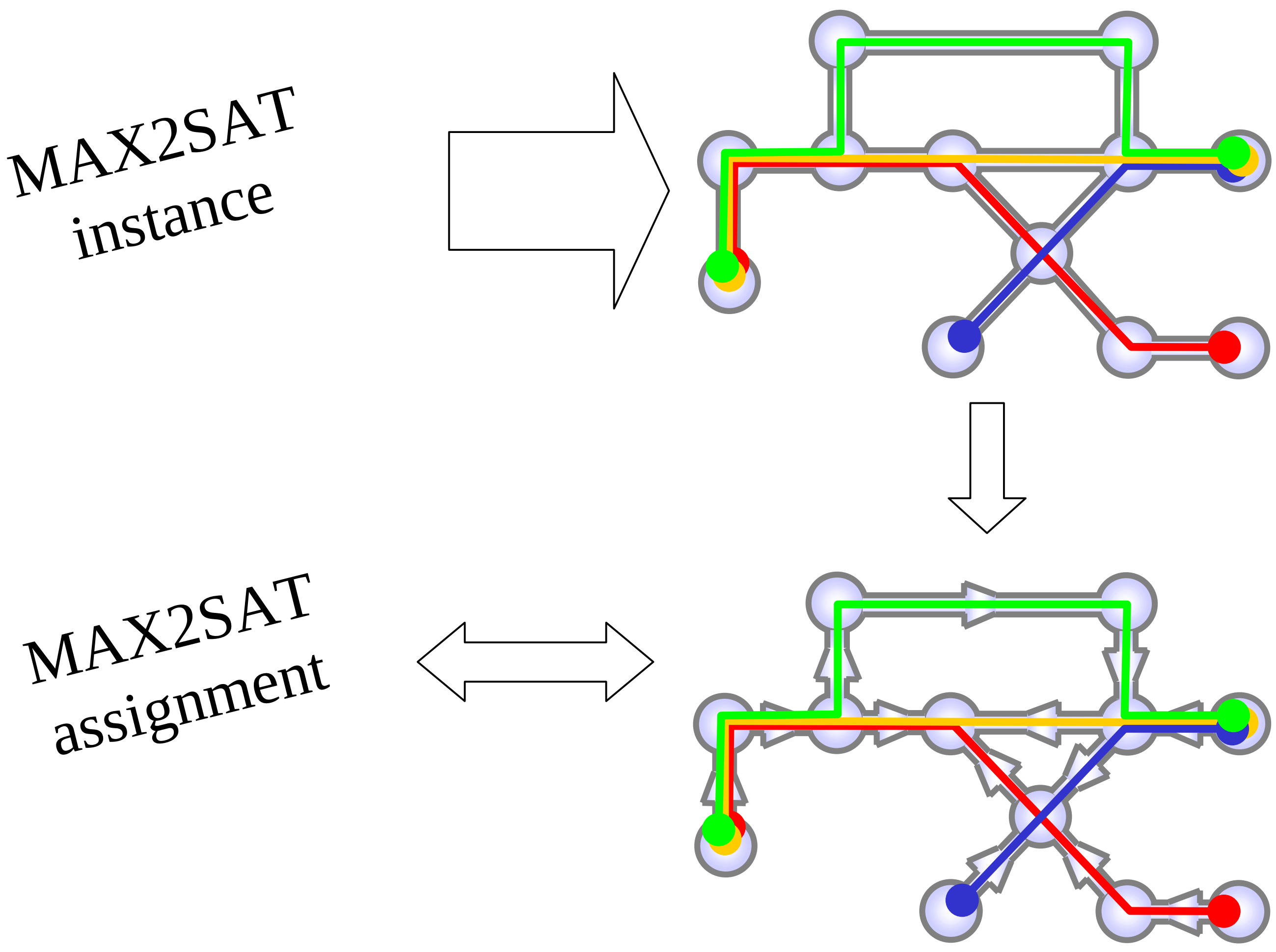
	AS #	AS name	Apr 18, 2001	Apr 6, 2002
			orientable w/o invalid paths	**orientable w/o invalid paths**
telnet sources	1	Genuity	yes	yes
	1740	CERFnet	yes	*not available*
	3549	Globalcrossing	yes	yes
	3582	Univ.of Oregon	no	no
	3967	Exodus Comm.	yes	yes
	4197	Global Online Japan	yes	yes
	5388	Energis Squared	yes	yes
	7018	AT&T	yes	yes
	8220	COLT Internet	yes	yes
	8709	Exodus, Europe	yes	yes
web sources	1755	Ebone	yes	*not available*
	2516	KDDI	no	*not available*
	2548	MaeWest	yes	*not available*
	6893	CW	yes	*not available*

computing a solution without invalid paths

- consider the directed acyclic graph of the connected components of the graph of the literals
- compute a topological sorting of the connected components and assign an integer to each component
- call f(x) the index of the component to which the literal x belongs
- a true value is assigned to variable x if $f(x) > f(\neg x)$, false otherwise

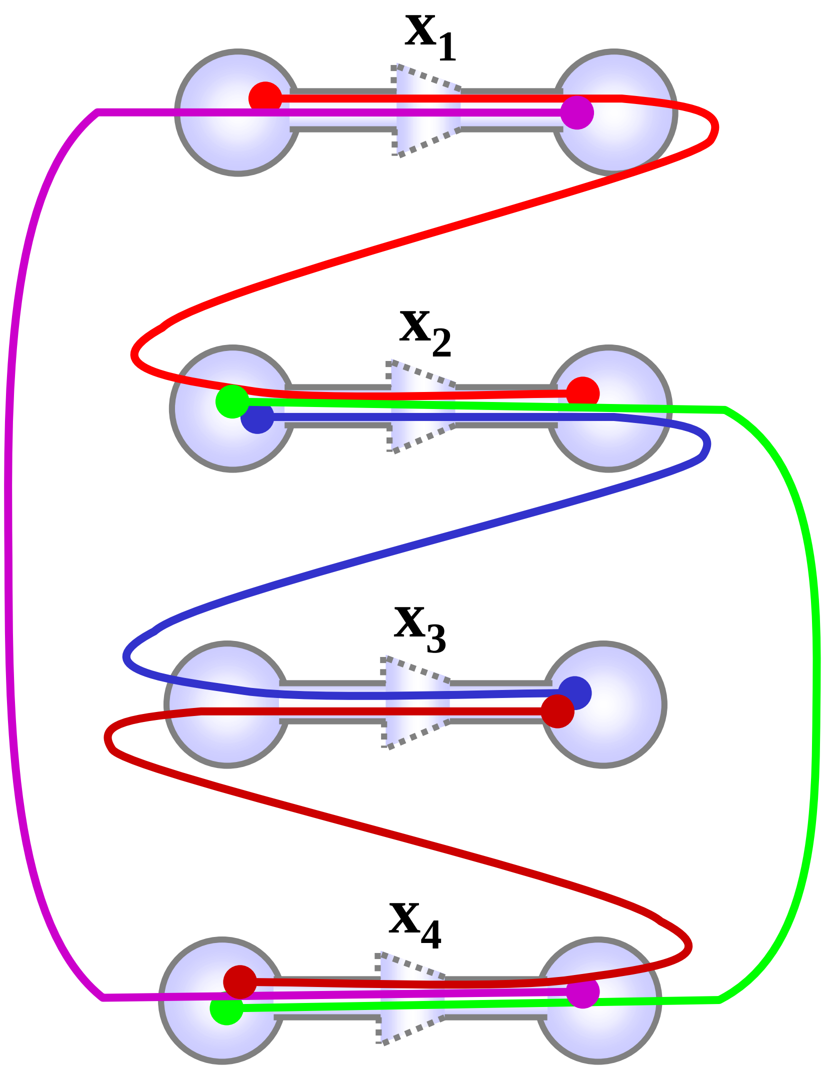


the general problem is NP-hard



example

$$(x_1 \vee \neg x_2) \wedge (x_2 \vee \neg x_3) \wedge (x_2 \vee x_4) \wedge (\neg x_1 \vee \neg x_4) \wedge (\neg x_3 \vee x_4)$$



our heuristic approach

size of the problem:

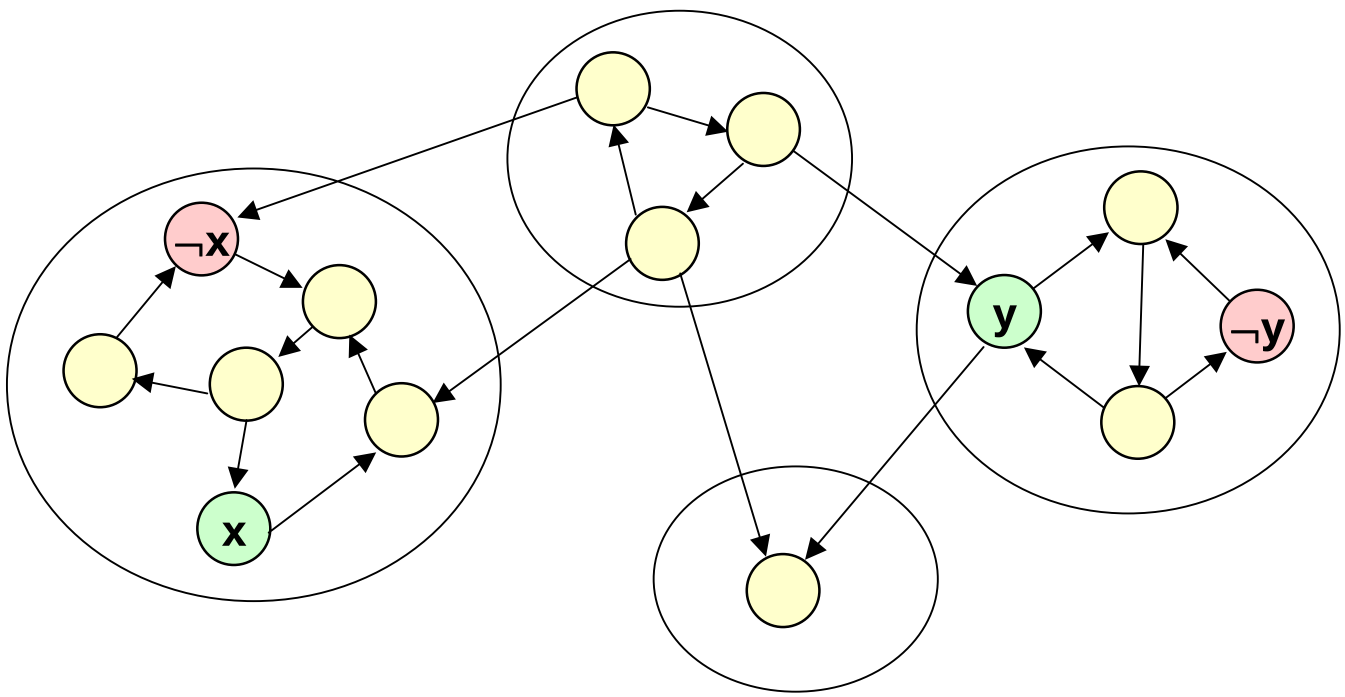
3,423,460 AS-paths
10,916 vertices
23,761 edges

very simple approach:

- find a very large set of AS-paths admitting an orientation
- try to reinsert the kept out AS-paths

1(find a very large set of paths admitting an orientation

1.a) rank the edges with respect to the path using them
1.b) construct the graph of the literals
1.c) compute the strongly connected components of it
1.d) find out all the variables whose two literals fall into the same strongly connected component
1.e) consider the corresponding edges and eliminate the one with the highest rank (and all the paths using it, too)



2) try to reinsert the kept out AS-paths

2.a) set $x = 1$
2.b) add x paths
2.c) if it is still solvable
    commit the path addition
    $x = x*2$
  else
    if x is 1
      discard the path
    $x = x/2$
2.d) repeat 2.b and 2.c until no path is left

some figures: out of 3,423,460 paths, we had 3,776,619 paths after the first step; we reinserted 222,764 with the second step, ending with 3,399,389 valid paths

comparison with the SARK paper

	AS #	AS Name	SARK Err. %	SAT Err. %	SARK recomp. Err. %
telnet sources	1	Genuity	0.65	0.45	0.86
	1740	CERFnet	n.a.	0.36	0.96
	3549	Globalcrossing	n.a.	0.13	3.44
	3582	Univ.of Oregon	n.a.	0.57	19.20
	3967	Exodus Comm.	n.a.	0.42	5.95
	4197	Global Online Japan	n.a.	0.46	2.83
	5388	Energis Squared	n.a.	0.46	53.63
	7018	AT&T	0.63	0.21	0.73
	8220	COLT Internet	n.a.	0.22	6.32
	8709	Exodus, Europe	n.a.	0.21	3.98
web sources	1755	Ebone	2.89	1.52	3.33
	2516	KDDI	8.97	4.95	20.80
	2548	MaeWest	1.49	0.19	0.99 ?
	6893	CW	2.92	0.64	11.84

a deeper comparison

SARK algorithm	
number of nodes	10923
number of edges	23757
peering edges	1136
percentage of total	5%
path covering distribution	minimum: 0 average: 50 maximum: 22327
strongly connected components	10302
scc's to edges ratio	0.4336406
size distribution:	10096 scc's of size 1 176 scc's of size 2 21 scc's of size 3 4 scc's of size 4 1 scc of size 6 1 scc of size 7 1 scc of size 19 1 scc of size 86 1 scc of size 278

SAT algorithm	
number of nodes	10897
number of edges	23690
peering edges	0
percentage of total	% 0
path covering distribution	minimum: 1 average: 80 maximum: 19918
strongly connected components	10312
scc's to edges ratio	0.4352892
size distribution:	10311 scc's of size 1 1 scc of size 586

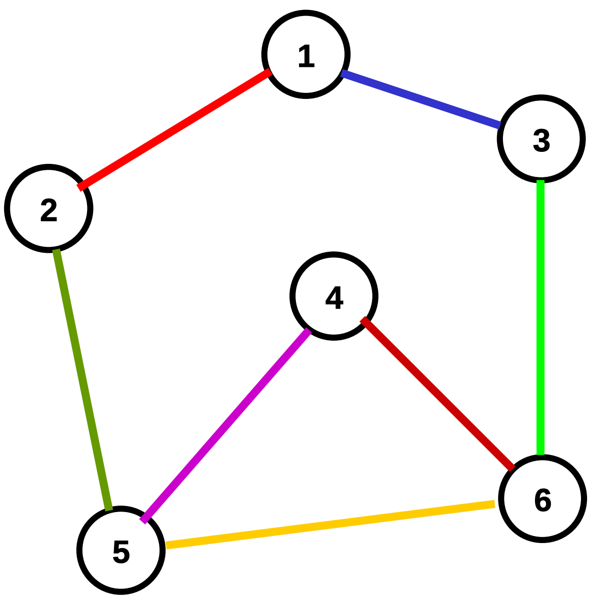
“graph of the differences”

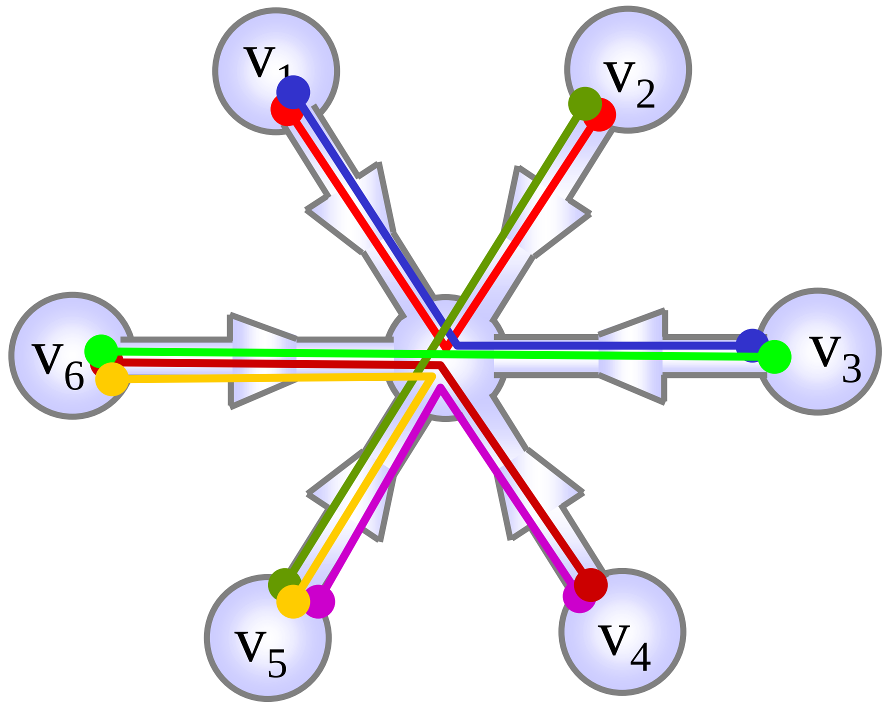
we inserted one undirected edge (and the two end vertices if needed) into the graph of the differences for every oppositely directed edge

graph of the differences	
number of nodes	489
number of edges	1148
path covering distribution	minimum: 1 average: 37 maximum: 3420
connected components	19
cc's to edges ratio	0.0165505
size distribution:	18 cc's of size 2 1 cc of size 453

discovering the peer-peer relationships

given an oriented graph, discovering peer-peer relationships is a hard problem if you want to maximize them (MAX-INDEPENDENT-SET can be mapped to it)





degrees of freedom in determining the peering relationships

AS #	AS Name	edges	candidates	peer-peer	% peer-peer
1	Genuity	13,001	11,395	8,573	65.94
1740	CERFnet	13,416	10,741	7,931	59.12
3549	Globalcrossing	13,039	10,851	8,250	63.28
3582	Univ.of Oregon				
3967	Exodus Comm.	18,401	14,562	11,626	63.18
4197	Global Online Japan	13,004	9,794	7,573	58.24
5388	Energis Squared	13,259	8,969	7,383	55.68
7018	AT&T	12,117	9,664	7,672	63.32
8220	COLT Internet	10,932	7,970	6,616	60.52
8709	Exodus, Europe	15,006	10,079	8,477	56.49
	all the ten together	23,761	4,148	3,936	16.56

need for more semantic information for determining peer-peer relationships

conclusions

- we show that the ToR-problem is NP-hard but…
- if a solution without invalid paths exists, it can be found in linear time by mapping the problem to 2SAT
- we propose heuristics for the general problem based on the 2SAT mapping
- we show the effectiveness of the heuristics against publicly available data sets
- the experiments put in evidence that our heuristics performs significantly better than state of the art heuristics
- we show that discovering peer-peer relationships is a hard problem if one wants to maximize them

issues

- how to determine peer-peer relationships once the graph is oriented?
- each AS-path is weighted one, irrespectively of the size of its prefix. To what extent is this correct?
- could snapshots of the same BGP table taken at different dates help in better understanding the relationships between ASes?
- what if we knew in advance the orientation of some of the edges?

links

- Lixin Gao, “On Inferring Autonomous System Relationships in the Internet”
  http://www-unix.ecs.umass.edu/~lgao/
- Subramanian, Agarwal, Rexford, and Katz “Characterizing the Internet Hierarchy from Multiple Vantage Points”
  http://www.cs.berkeley.edu/~sagarwal/research/BGP-hierarchy/
- Erlebach, Hall, and Schank, “Classifying Customer-Provider Relationships in the Internet”
  http://www.tik.ee.ethz.ch/~the/
- website dedicated to the algorithms described in this presentation
  http://www.dia.uniroma3.it/~compunet/relationships/

acknowledgements

- Subramanian, Agarwal, Rexford, and Katz for sharing their data and explanations
- Thomas Erlebach, Alexander Hall, and Thomas Schank for their insight
- Debora Donato, Andrea Vitaletti for useful discussions
- Massimo Rimondini for computing statistics on the graphs

questions?