Graph editing: algorithms and experimental results

Christophe Crespelle

Université Côte d'Azur

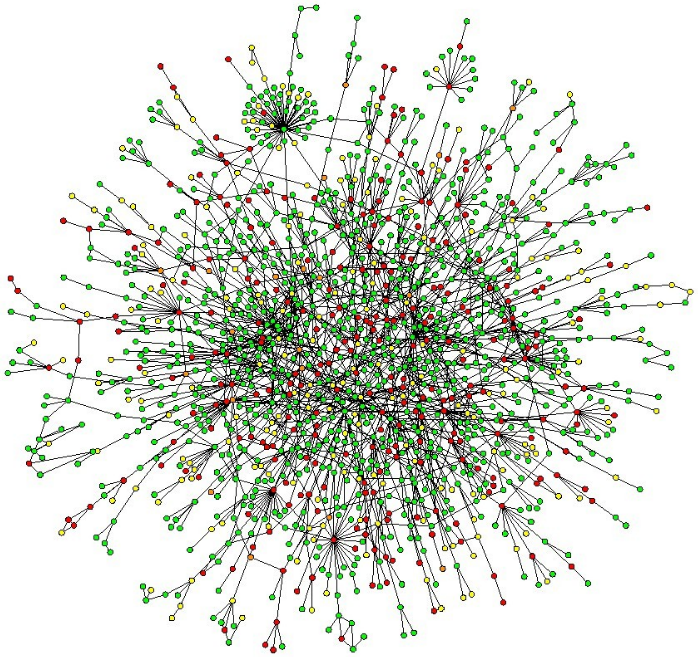
with Jean Blair, Anne-Aymone Bourguin, Benjamin Gras,
Daniel Lokshtanov, Remi Pellerin, Anthony Perez, Thi Ha Duong Phan,
Eric Thierry and Stéphan Thomassé

Complex Networks

Complex networks

- Real-world data

  Ex of contexts :
  computer science,
  social sciences,
  biology, linguistics,
  medecine,
  transportation,
  communications,
  industry, economy, ...

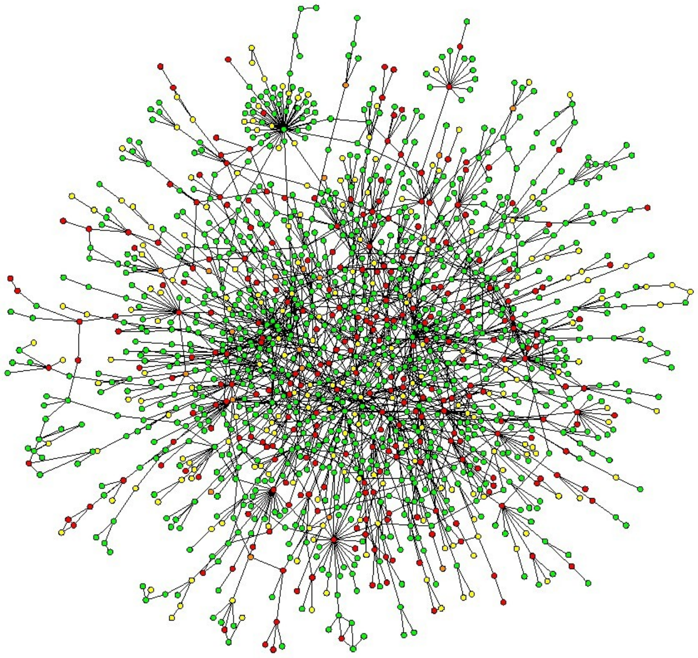


complex
=
large
+
unordered

Complex networks

- Real-world data

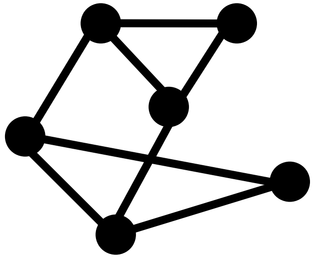
  <u>Ex of contexts :</u>
  computer science,
  social sciences,
  biology, linguistics,
  medecine,
  transportation,
  communications,
  industry, economy, ...



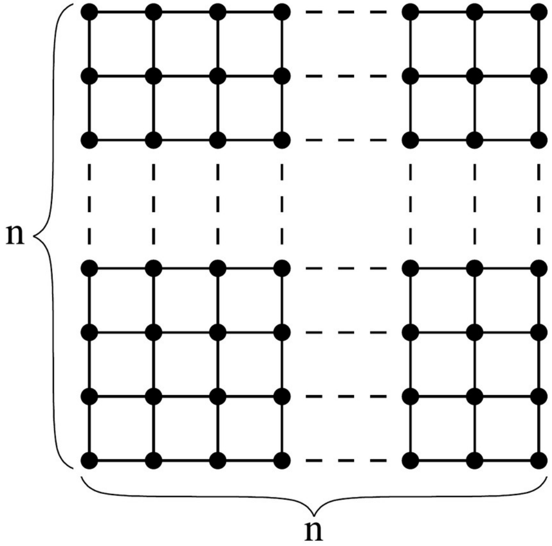
complex
=
large
+
unordered

- Not complex

small



ordered

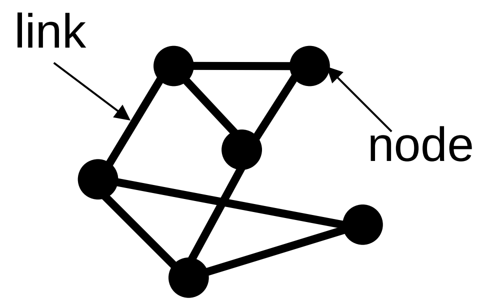


Complex networks

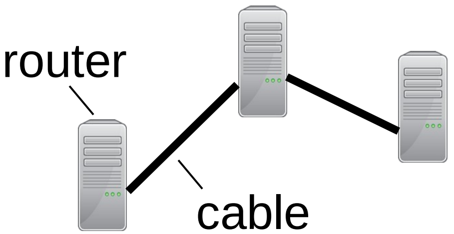
- Real-world data *(not formally defined)*

  Ex of contexts :

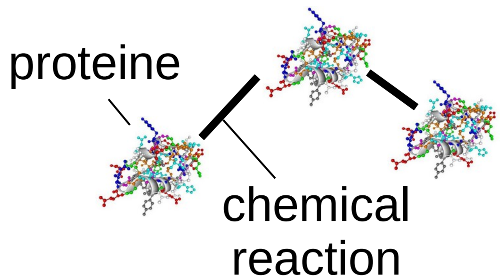
  computer science, social sciences, biology, linguistics, medecine, Transportation, communications, industry, economy, ...



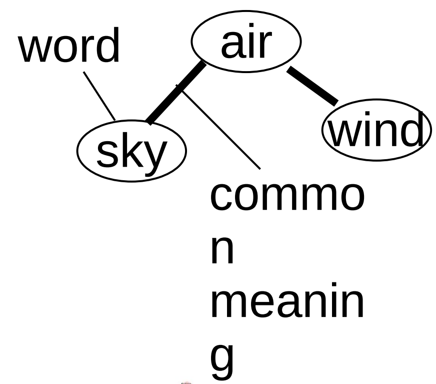
Internet

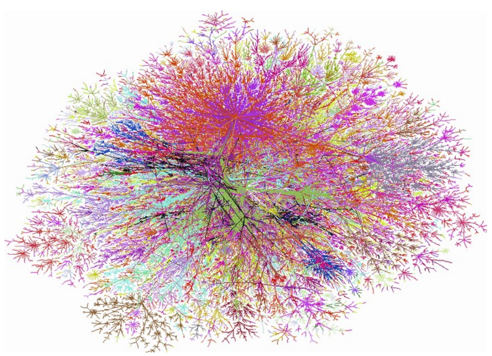


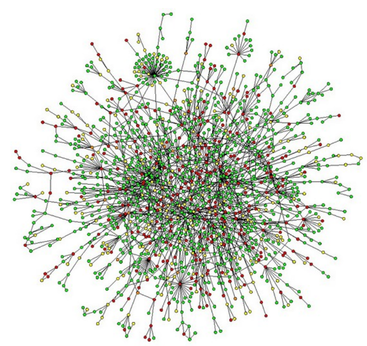
Proteine interactions

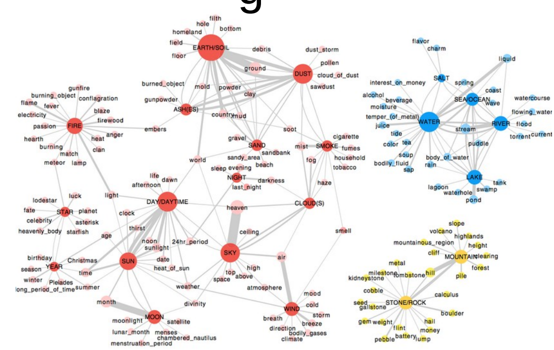


Word networks









How to carry information across the Internet?

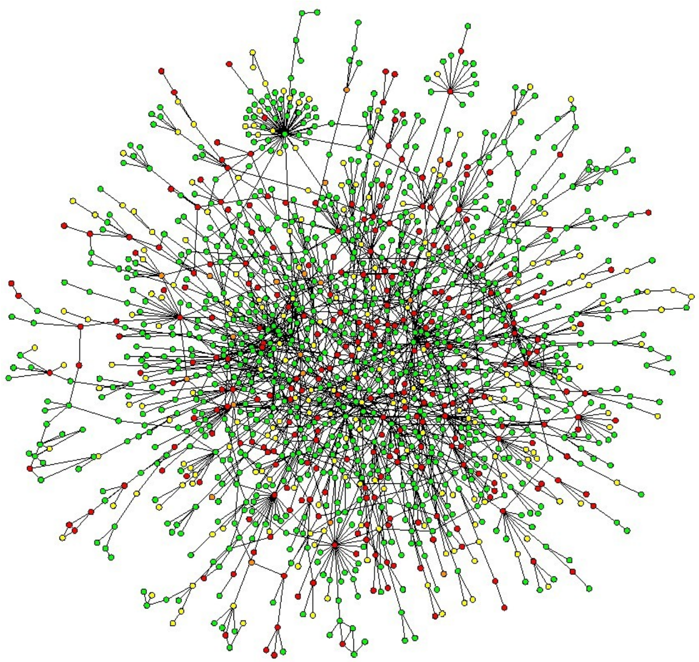
How does a living cell work?

How does a language evolve?

Complex networks

- Real-world data

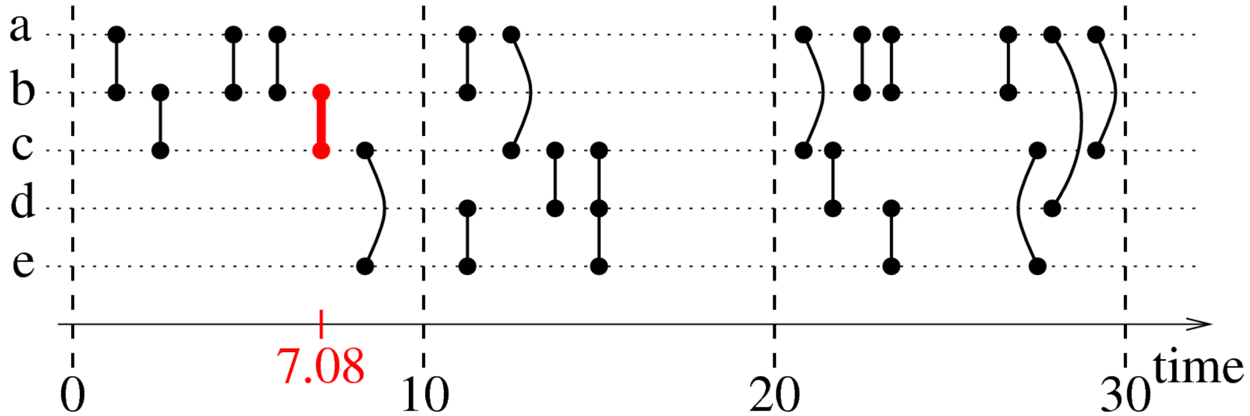
  Ex of contexts :
  computer science,
  social sciences,
  biology, linguistics,
  medecine,
  transportation,
  communications,
  industry, economy, ...



complex
=
large
+
unordered

- Links depend on time

(1.25 , a , b)
(2.50 , b , c)
(4.58 , a , b)
(5.83 , a , b)
(7.08 , b , c)
(8.33 , c , e)
. . .



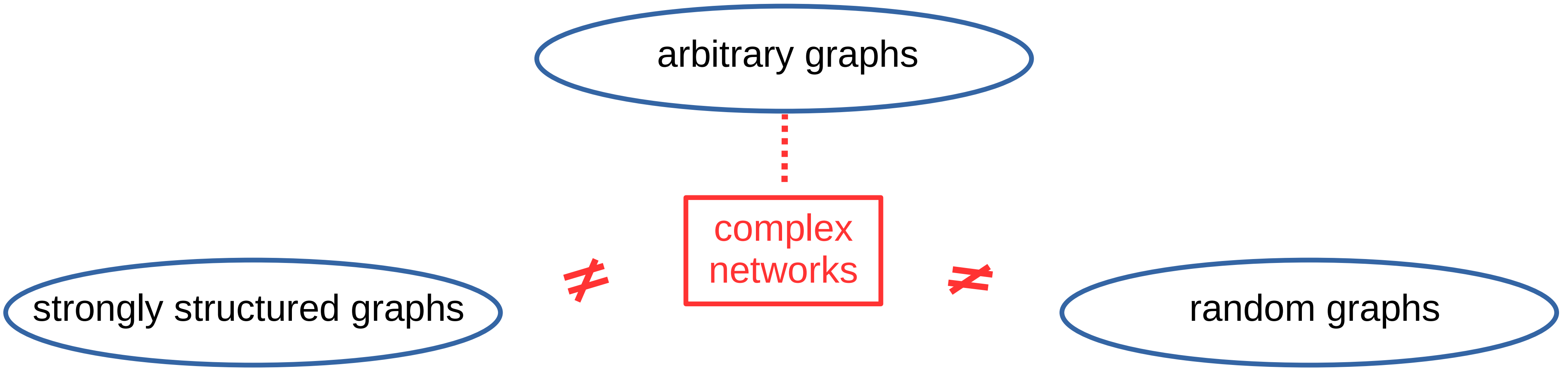
Four big classes of problems

- Measurement
- Analysis
- Modelling
- Algorithms

Four big classes of problems

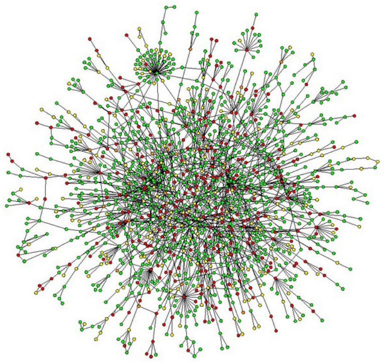
- Measurement
- Analysis
- Modelling
- Algorithms

Graph theory



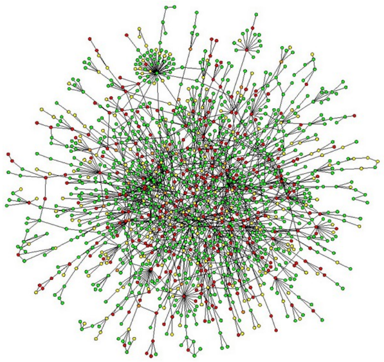
Complex networks as almost structured graphs

Almost structured graphs



- loosely constrained
  → **randomness**
- strongly impacted by their context
  → **structure**

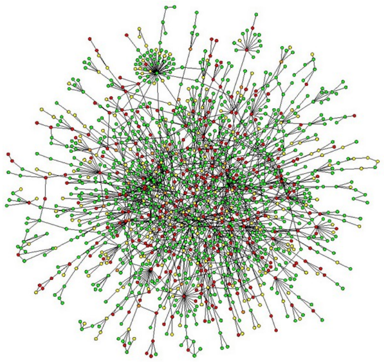
Almost structured graphs



- loosely constrained
  → **randomness**
- strongly impacted by their context
  → **structure**

Complex networks	**=**	**structure**	**+**	**randomness**
[Watts & Strogatz 1998]		High local density		Short distances

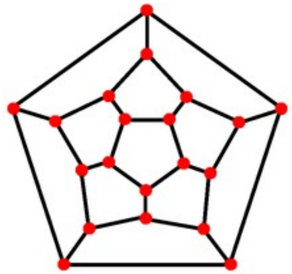
Almost structured graphs



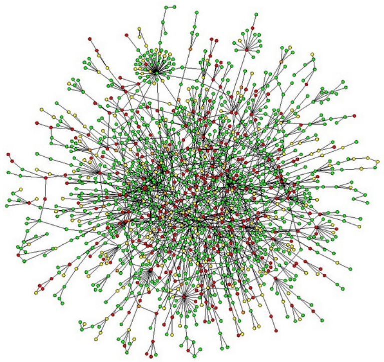
- loosely constrained
  → **randomness**
- strongly impacted by their context
  → **structure**

Complex networks = structure + randomness

1 strongly structured



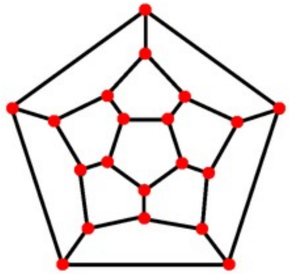
Almost structured graphs



- loosely constrained
  ➡ **randomness**
- strongly impacted by their context
  ➡ **structure**

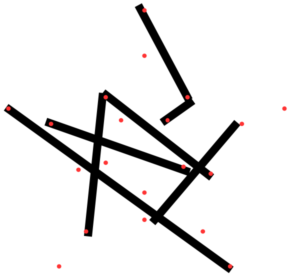
Complex networks = structure + randomness

1 strongly structured

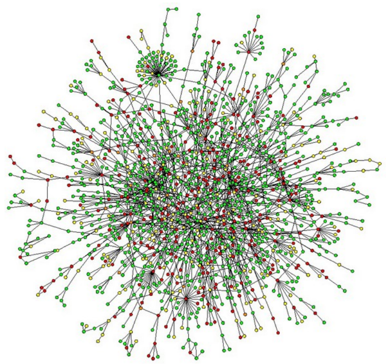


+

2 random modifications



Almost structured graphs

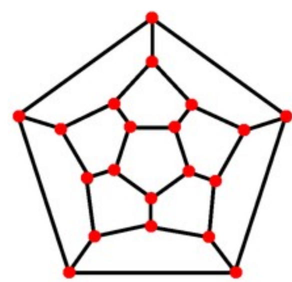


- loosely constrained
  → **randomness**
- strongly impacted by their context
  → **structure**

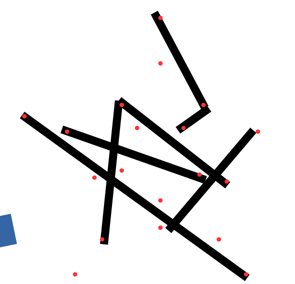
Complex networks = structure + randomness

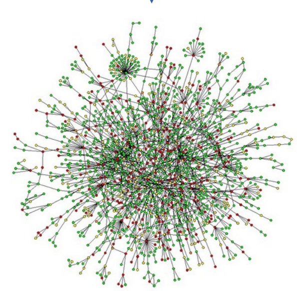
1 strongly structured

2 random modifications

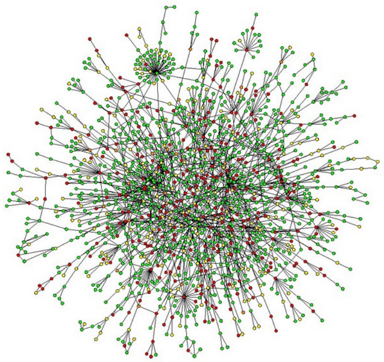


+





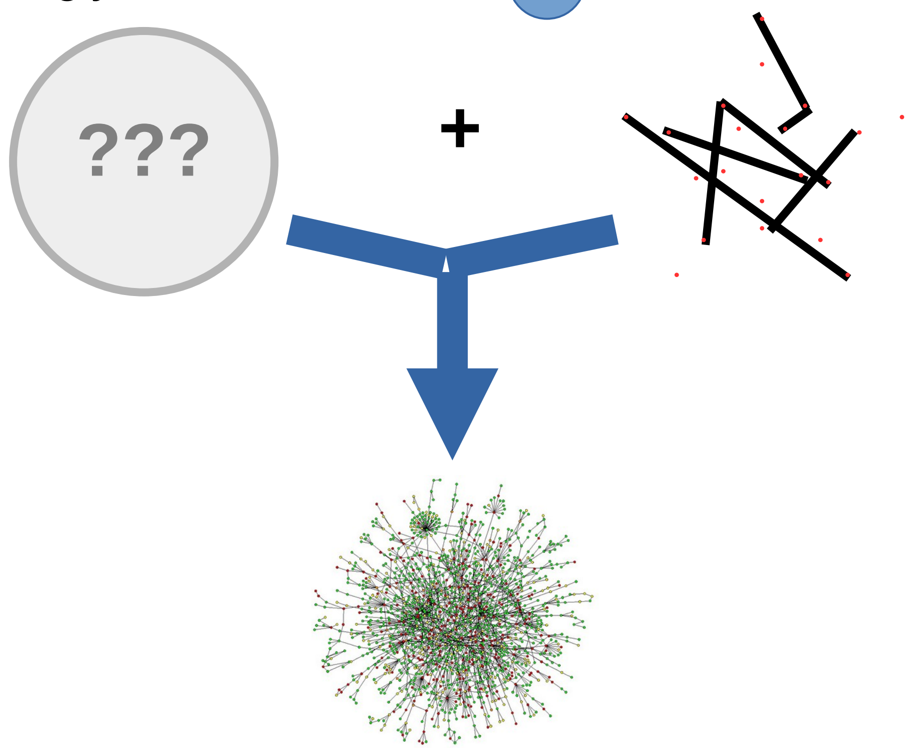
Almost structured graphs



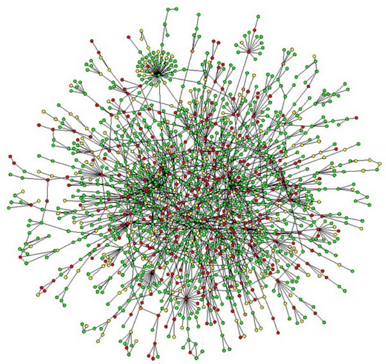
- loosely constrained
  → **randomness**
- strongly impacted by their context
  → **structure**

Complex networks = structure + randomness

1 strongly structured   2 random modifications



Almost structured graphs

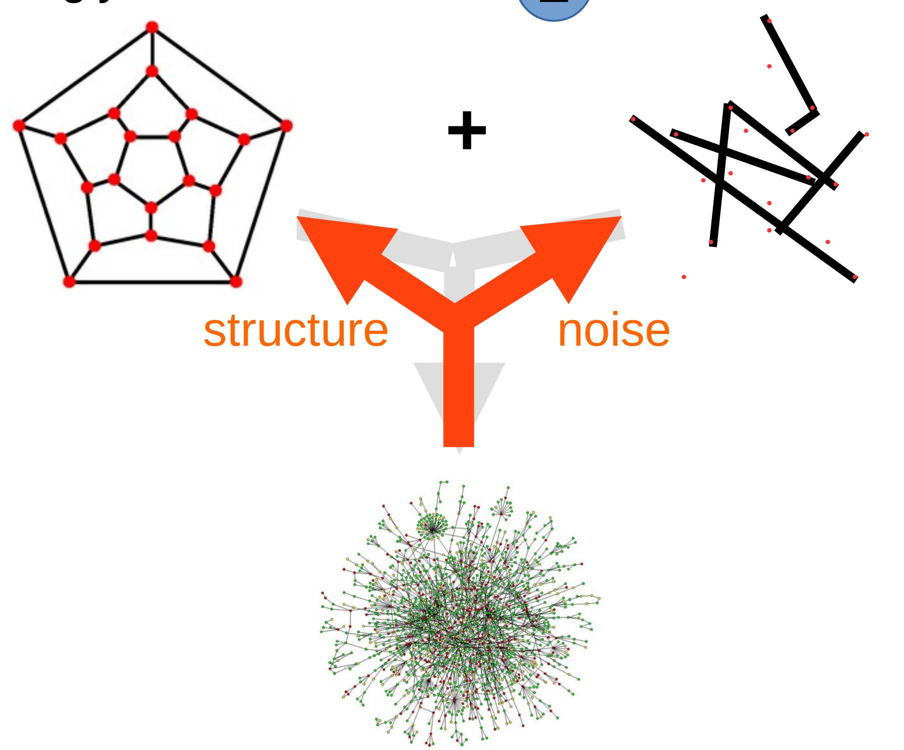


- loosely constrained
  → **randomness**
- strongly impacted by their context
  → **structure**

Complex networks = structure + randomness

1 strongly structured

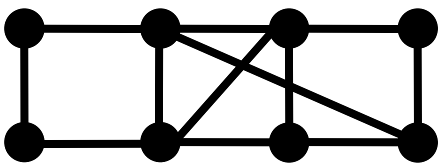
2 random modifications



Graph editing algorithms

Graph editing algorithms

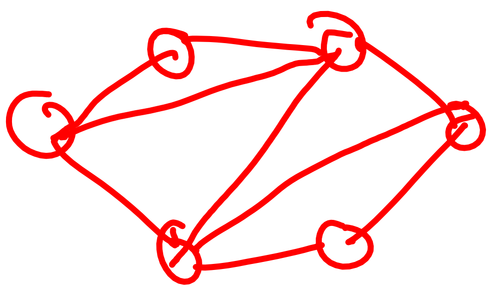
INPUT



arbitrary graph

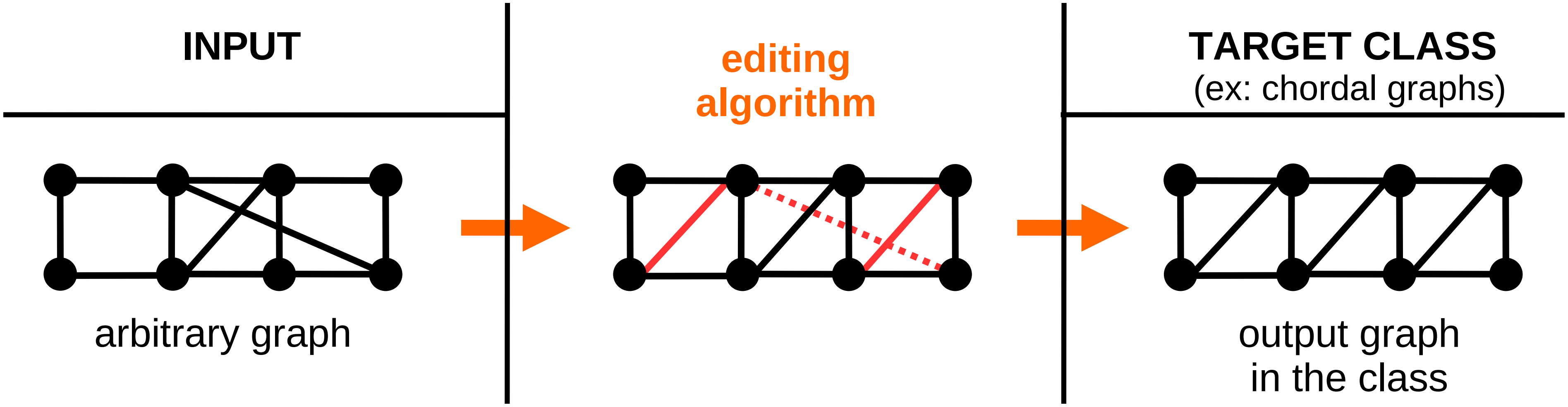
TARGET CLASS
(ex: chordal graphs)

Definition:
Chordal graphs = graphs without induced cycle on at least 4 vertices



triangulated

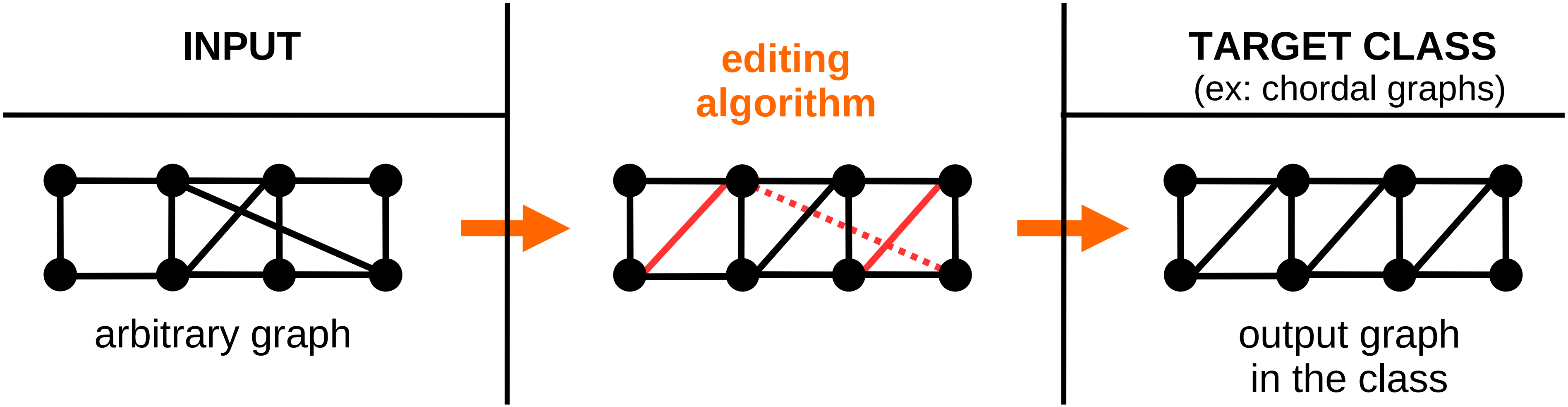
Graph editing algorithms



Definition:

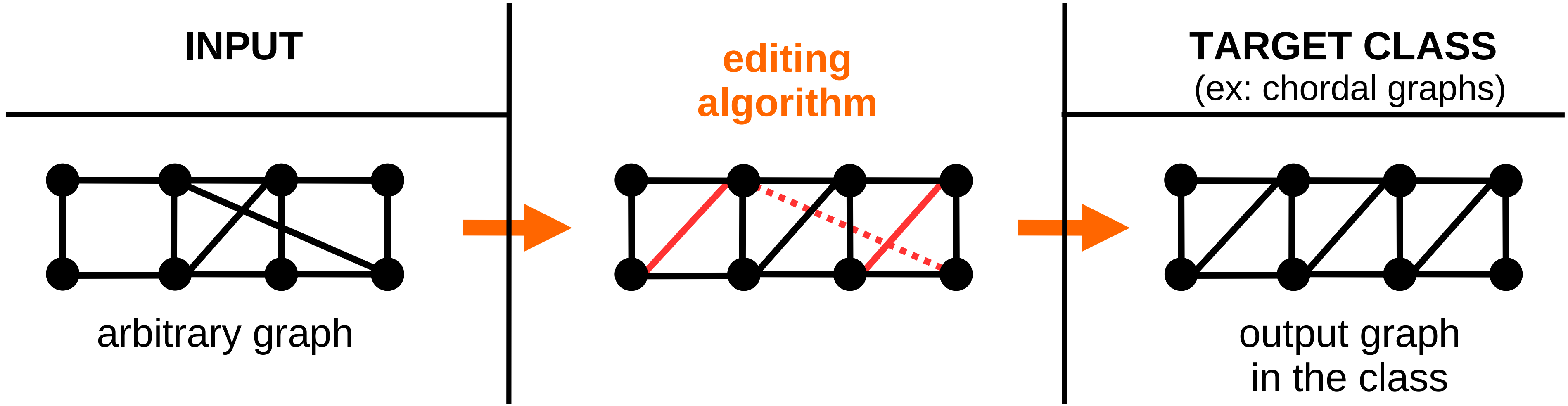
Chordal graphs = graphs without induced cycle on at least 4 vertices

Graph editing algorithms



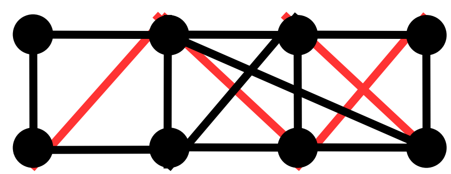
GOAL: perform as few modifications as possible

Graph editing algorithms



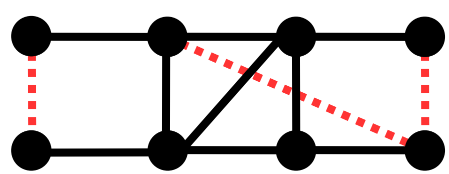
Two constrained versions of the problem:

Only additions allowed



completion algorithm

Only deletions allowed



deletion algorithm

Motivations

- **Mathematics**
  Distance to and projection on a class of graphs.
  How far is a graph from having a certain property?

- **Computation**
  Natural extension of the recognition problem of graph classes.
  When the recognition fail, how to minimally correct the graph?

- **Data science**
  Remove noise in graph data.
  - Measurement errors
  - Randomness (non-constrained part of the data)
  - Anything deviating from the main structure

Editing real-world networks

Cograph edition of real-world graphs

35 real-world graphs

+

8 random graphs

Context	Network	n	m	d°	%mod
WWW	in-2004	1 148 875	12 281 937	21.4	12 %
WWW	cnr-2000	227 058	2 187 201	19.3	19 %
PROTEIN	reactome	5 973	145 778	48.8	22 %
SOFTWARE	jdk	6 434	53 658	16.7	29 %
SOFTWARE	jung-j	6 120	50 290	16.4	29 %
WWW	eu-2005	835 044	15 718 784	37.7	29 %
CO-AUTHOR	ca-GrQc	4 158	13 422	6.5	34 %
CO-AUTHOR	ca-HepPh	11 204	117 619	21.0	34 %
SPECIES	foodweb	183	2 434	26.6	43 %
CO-AUTHOR	dblp	317 080	1 049 866	6.6	45 %
WORD-REL.	wordnet	145 145	656 230	9.0	48 %
COMMUNIC.	wiki-Talk	2 388 953	4 656 682	3.9	49 %
CO-SOLD	amazon	334 863	925 872	5.5	49 %
CO-AUTHOR	ca-CondMat	21 363	91 286	8.6	52 %
RANDOM	ER-Gnm_1M-2	796 208	958 827	2.4	52 %
CO-AUTHOR	ca-HepTh	8 638	24 806	5.7	54 %
INTERNET	as2000	6 474	12 572	3.9	54 %
ROAD	roadNet-TX	1 351 137	1 879 201	2.8	54 %
INTERNET	as-caida2007	26 475	53 381	4.0	55 %
CO-AUTHOR	ca-AstroPh	17 903	196 972	22.0	59 %
INTERNET	topology	34 761	107 720	6.2	61 %
RANDOM	ER-Gnm_1M-3	940 987	1 494 643	3.2	63 %
INTERNET	as-skitter	1 694 616	11 094 209	13.1	64 %
CO-OCCUR	bible-names	1 707	9 059	10.6	67 %
PROTEIN	figeys	2 217	6 418	5.8	67 %
CITATION-SCI.	cora	23 166	89 157	7.7	68 %
SOCIAL	youtube	1 134 890	2 987 624	5.3	69 %
CO-ACTOR	actor-col.	374 511	15 014 839	80.2	71 %
P2P-CONNECT.	p2p-Gnutella	62 561	147 878	4.7	71 %
RANDOM	ER-Gnm_1M-4	980 191	1 999 203	4.1	71 %
CITATION-SCI.	citeseer	365 154	1 721 981	9.4	75 %
CITATION-PAT.	cit-Patents	3 764 117	16 511 740	8.8	76 %
SOFTWARE	linux	30 817	213 208	13.8	77 %
SOCIAL	LiveJournal	3 997 962	34 681 189	17.4	78 %
CITATION-SCI.	cit-HepTh	27 400	352 021	25.7	79 %
RANDOM	ER-Gnm_1M-6	997 479	2 999 988	6.0	79 %
CITATION-SCI.	cit-HepPh	34 401	420 784	24.5	81 %
RANDOM	ER-Gnm_1M-8	999 684	3 999 999	8.0	84 %
RANDOM	ER-Gnm_1M-10	999 952	5 000 000	10.0	87 %
RANDOM	ER-Gnm_1M-15	1 000 000	7 500 000	15.0	91 %
SOCIAL	orkut	3 072 441	117 185 083	76.3	91 %
RANDOM	ER-Gnm_1M-20	1 000 000	10 000 000	20.0	93 %
WORD-REL.	Thesaurus	23 132	297 094	25.7	93 %

Cograph edition of real-world graphs

RESULTS

35 real-world graphs

+

8 random graphs

Context	Network	n	m	d°	%mod
WWW	in-2004	1 148 875	12 281 937	21.4	12 %
WWW	cnr-2000	227 058	2 187 201	19.3	19 %
PROTEIN	reactome	5 973	145 778	48.8	22 %
SOFTWARE	jdk	6 434	53 658	16.7	29 %
SOFTWARE	jung-j	6 120	50 290	16.4	29 %
WWW	eu-2005	835 044	15 718 784	37.7	29 %
CO-AUTHOR	ca-GrQc	4 158	13 422	6.5	34 %
CO-AUTHOR	ca-HepPh	11 204	117 619	21.0	34 %
SPECIES	foodweb	183	2 434	26.6	43 %
CO-AUTHOR	dblp	317 080	1 049 866	6.6	45 %
WORD-REL.	wordnet	145 145	656 230	9.0	48 %
COMMUNIC.	wiki-Talk	2 388 953	4 656 682	3.9	49 %
CO-SOLD	amazon	334 863	925 872	5.5	49 %
CO-AUTHOR	ca-CondMat	21 363	91 286	8.6	52 %
RANDOM	ER-Gnm_1M-2	796 208	958 827	2.4	52 %
CO-AUTHOR	ca-HepTh	8 638	24 806	5.7	54 %
INTERNET	as2000	6 474	12 572	3.9	54 %
ROAD	roadNet-TX	1 351 137	1 879 201	2.8	54 %
INTERNET	as-caida2007	26 475	53 381	4.0	55 %
CO-AUTHOR	ca-AstroPh	17 903	196 972	22.0	59 %
INTERNET	topology	34 761	107 720	6.2	61 %
RANDOM	ER-Gnm_1M-3	940 987	1 494 643	3.2	63 %
INTERNET	as-skitter	1 694 616	11 094 209	13.1	64 %
CO-OCCUR	bible-names	1 707	9 059	10.6	67 %
PROTEIN	figeys	2 217	6 418	5.8	67 %
CITATION-SCI.	cora	23 166	89 157	7.7	68 %
SOCIAL	youtube	1 134 890	2 987 624	5.3	69 %
CO-ACTOR	actor-col.	374 511	15 014 839	80.2	71 %
P2P-CONNECT.	p2p-Gnutella	62 561	147 878	4.7	71 %
RANDOM	ER-Gnm_1M-4	980 191	1 999 203	4.1	71 %
CITATION-SCI.	citeseer	365 154	1 721 981	9.4	75 %
CITATION-PAT.	cit-Patents	3 764 117	16 511 740	8.8	76 %
SOFTWARE	linux	30 817	213 208	13.8	77 %
SOCIAL	LiveJournal	3 997 962	34 681 189	17.4	78 %
CITATION-SCI.	cit-HepTh	27 400	352 021	25.7	79 %
RANDOM	ER-Gnm_1M-6	997 479	2 999 988	6.0	79 %
CITATION-SCI.	cit-HepPh	34 401	420 784	24.5	81 %
RANDOM	ER-Gnm_1M-8	999 684	3 999 999	8.0	84 %
RANDOM	ER-Gnm_1M-10	999 952	5 000 000	10.0	87 %
RANDOM	ER-Gnm_1M-15	1 000 000	7 500 000	15.0	91 %
SOCIAL	orkut	3 072 441	117 185 083	76.3	91 %
RANDOM	ER-Gnm_1M-20	1 000 000	10 000 000	20.0	93 %
WORD-REL.	Thesaurus	23 132	297 094	25.7	93 %

- Some networks are very close from cographs

Cograph edition of real-world graphs

35 real-world graphs

+

8 random graphs

Context	Network	n	m	d°	%mod
WWW	in-2004	1 148 875	12 281 937	21.4	12 %
WWW	cnr-2000	227 058	2 187 201	19.3	19 %
PROTEIN	reactome	5 973	145 778	48.8	22 %
SOFTWARE	jdk	6 434	53 658	16.7	29 %
SOFTWARE	jung-j	6 120	50 290	16.4	29 %
WWW	eu-2005	835 044	15 718 784	37.7	29 %
CO-AUTHOR	ca-GrQc	4 158	13 422	6.5	34 %
CO-AUTHOR	ca-HepPh	11 204	117 619	21.0	34 %
SPECIES	foodweb	183	2 434	26.6	43 %
CO-AUTHOR	dblp	317 080	1 049 866	6.6	45 %
WORD-REL.	wordnet	145 145	656 230	9.0	48 %
COMMUNIC.	wiki-Talk	2 388 953	4 656 682	3.9	49 %
CO-SOLD	amazon	334 863	925 872	5.5	49 %
CO-AUTHOR	ca-CondMat	21 363	91 286	8.6	52 %
RANDOM	ER-Gnm_1M-2	796 208	958 827	2.4	52 %
CO-AUTHOR	ca-HepTh	8 638	24 806	5.7	54 %
INTERNET	as2000	6 474	12 572	3.9	54 %
ROAD	roadNet-TX	1 351 137	1 879 201	2.8	54 %
INTERNET	as-caida2007	26 475	53 381	4.0	55 %
CO-AUTHOR	ca-AstroPh	17 903	196 972	22.0	59 %
INTERNET	topology	34 761	107 720	6.2	61 %
RANDOM	ER-Gnm_1M-3	940 987	1 494 643	3.2	63 %
INTERNET	as-skitter	1 694 616	11 094 209	13.1	64 %
CO-OCCUR	bible-names	1 707	9 059	10.6	67 %
PROTEIN	figeys	2 217	6 418	5.8	67 %
CITATION-SCI.	cora	23 166	89 157	7.7	68 %
SOCIAL	youtube	1 134 890	2 987 624	5.3	69 %
CO-ACTOR	actor-col.	374 511	15 014 839	80.2	71 %
P2P-CONNECT.	p2p-Gnutella	62 561	147 878	4.7	71 %
RANDOM	ER-Gnm_1M-4	980 191	1 999 203	4.1	71 %
CITATION-SCI.	citeseer	365 154	1 721 981	9.4	75 %
CITATION-PAT.	cit-Patents	3 764 117	16 511 740	8.8	76 %
SOFTWARE	linux	30 817	213 208	13.8	77 %
SOCIAL	LiveJournal	3 997 962	34 681 189	17.4	78 %
CITATION-SCI.	cit-HepTh	27 400	352 021	25.7	79 %
RANDOM	ER-Gnm_1M-6	997 479	2 999 988	6.0	79 %
CITATION-SCI.	cit-HepPh	34 401	420 784	24.5	81 %
RANDOM	ER-Gnm_1M-8	999 684	3 999 999	8.0	84 %
RANDOM	ER-Gnm_1M-10	999 952	5 000 000	10.0	87 %
RANDOM	ER-Gnm_1M-15	1 000 000	7 500 000	15.0	91 %
SOCIAL	orkut	3 072 441	117 185 083	76.3	91 %
RANDOM	ER-Gnm_1M-20	1 000 000	10 000 000	20.0	93 %
WORD-REL.	Thesaurus	23 132	297 094	25.7	93 %

RESULTS

- Some networks are very close from cographs
- Random graphs are never

Cograph edition of real-world graphs

35 real-world graphs

+

8 random graphs

Context	Network	n	m	d°	%mod
WWW	in-2004	1 148 875	12 281 937	21.4	12 %
WWW	cnr-2000	227 058	2 187 201	19.3	19 %
PROTEIN	reactome	5 973	145 778	48.8	22 %
SOFTWARE	jdk	6 434	53 658	16.7	29 %
SOFTWARE	jung-j	6 120	50 290	16.4	29 %
WWW	eu-2005	835 044	15 718 784	37.7	29 %
CO-AUTHOR	ca-GrQc	4 158	13 422	6.5	34 %
CO-AUTHOR	ca-HepPh	11 204	117 619	21.0	34 %
SPECIES	foodweb	183	2 434	26.6	43 %
CO-AUTHOR	dblp	317 080	1 049 866	6.6	45 %
WORD-REL.	wordnet	145 145	656 230	9.0	48 %
COMMUNIC.	wiki-Talk	2 388 953	4 656 682	3.9	49 %
CO-SOLD	amazon	334 863	925 872	5.5	49 %
CO-AUTHOR	ca-CondMat	21 363	91 286	8.6	52 %
RANDOM	ER-Gnm_1M-2	796 208	958 827	2.4	52 %
CO-AUTHOR	ca-HepTh	8 638	24 806	5.7	54 %
INTERNET	as2000	6 474	12 572	3.9	54 %
ROAD	roadNet-TX	1 351 137	1 879 201	2.8	54 %
INTERNET	as-caida2007	26 475	53 381	4.0	55 %
CO-AUTHOR	ca-AstroPh	17 903	196 972	22.0	59 %
INTERNET	topology	34 761	107 720	6.2	61 %
RANDOM	ER-Gnm_1M-3	940 987	1 494 643	3.2	63 %
INTERNET	as-skitter	1 694 616	11 094 209	13.1	64 %
CO-OCCUR	bible-names	1 707	9 059	10.6	67 %
PROTEIN	figeys	2 217	6 418	5.8	67 %
CITATION-SCI.	cora	23 166	89 157	7.7	68 %
SOCIAL	youtube	1 134 890	2 987 624	5.3	69 %
CO-ACTOR	actor-col.	374 511	15 014 839	80.2	71 %
P2P-CONNECT.	p2p-Gnutella	62 561	147 878	4.7	71 %
RANDOM	ER-Gnm_1M-4	980 191	1 999 203	4.1	71 %
CITATION-SCI.	citeseer	365 154	1 721 981	9.4	75 %
CITATION-PAT.	cit-Patents	3 764 117	16 511 740	8.8	76 %
SOFTWARE	linux	30 817	213 208	13.8	77 %
SOCIAL	LiveJournal	3 997 962	34 681 189	17.4	78 %
CITATION-SCI.	cit-HepTh	27 400	352 021	25.7	79 %
RANDOM	ER-Gnm_1M-6	997 479	2 999 988	6.0	79 %
CITATION-SCI.	cit-HepPh	34 401	420 784	24.5	81 %
RANDOM	ER-Gnm_1M-8	999 684	3 999 999	8.0	84 %
RANDOM	ER-Gnm_1M-10	999 952	5 000 000	10.0	87 %
RANDOM	ER-Gnm_1M-15	1 000 000	7 500 000	15.0	91 %
SOCIAL	orkut	3 072 441	117 185 083	76.3	91 %
RANDOM	ER-Gnm_1M-20	1 000 000	10 000 000	20.0	93 %
WORD-REL.	Thesaurus	23 132	297 094	25.7	93 %

RESULTS

- Some networks are very close from cographs
- Random graphs are never
- A wide range of proximity : 12% to 93%

Cograph edition of real-world graphs

35 real-world graphs

+

8 random graphs

Context	Network	n	m	d°	%mod
WWW	in-2004	1 148 875	12 281 937	21.4	12 %
WWW	cnr-2000	227 058	2 187 201	19.3	19 %
PROTEIN	reactome	5 973	145 778	48.8	22 %
SOFTWARE	jdk	6 434	53 658	16.7	29 %
SOFTWARE	jung-j	6 120	50 290	16.4	29 %
WWW	eu-2005	835 044	15 718 784	37.7	29 %
CO-AUTHOR	ca-GrQc	4 158	13 422	6.5	34 %
CO-AUTHOR	ca-HepPh	11 204	117 619	21.0	34 %
SPECIES	foodweb	183	2 434	26.6	43 %
CO-AUTHOR	dblp	317 080	1 049 866	6.6	45 %
WORD-REL.	wordnet	145 145	656 230	9.0	48 %
COMMUNIC.	wiki-Talk	2 388 953	4 656 682	3.9	49 %
CO-SOLD	amazon	334 863	925 872	5.5	49 %
CO-AUTHOR	ca-CondMat	21 363	91 286	8.6	52 %
RANDOM	ER-Gnm_1M-2	796 208	958 827	2.4	52 %
CO-AUTHOR	ca-HepTh	8 638	24 806	5.7	54 %
INTERNET	as2000	6 474	12 572	3.9	54 %
ROAD	roadNet-TX	1 351 137	1 879 201	2.8	54 %
INTERNET	as-caida2007	26 475	53 381	4.0	55 %
CO-AUTHOR	ca-AstroPh	17 903	196 972	22.0	59 %
INTERNET	topology	34 761	107 720	6.2	61 %
RANDOM	ER-Gnm_1M-3	940 987	1 494 643	3.2	63 %
INTERNET	as-skitter	1 694 616	11 094 209	13.1	64 %
CO-OCCUR	bible-names	1 707	9 059	10.6	67 %
PROTEIN	figeys	2 217	6 418	5.8	67 %
CITATION-SCI.	cora	23 166	89 157	7.7	68 %
SOCIAL	youtube	1 134 890	2 987 624	5.3	69 %
CO-ACTOR	actor-col.	374 511	15 014 839	80.2	71 %
P2P-CONNECT.	p2p-Gnutella	62 561	147 878	4.7	71 %
RANDOM	ER-Gnm_1M-4	980 191	1 999 203	4.1	71 %
CITATION-SCI.	citeseer	365 154	1 721 981	9.4	75 %
CITATION-PAT.	cit-Patents	3 764 117	16 511 740	8.8	76 %
SOFTWARE	linux	30 817	213 208	13.8	77 %
SOCIAL	LiveJournal	3 997 962	34 681 189	17.4	78 %
CITATION-SCI.	cit-HepTh	27 400	352 021	25.7	79 %
RANDOM	ER-Gnm_1M-6	997 479	2 999 988	6.0	79 %
CITATION-SCI.	cit-HepPh	34 401	420 784	24.5	81 %
RANDOM	ER-Gnm_1M-8	999 684	3 999 999	8.0	84 %
RANDOM	ER-Gnm_1M-10	999 952	5 000 000	10.0	87 %
RANDOM	ER-Gnm_1M-15	1 000 000	7 500 000	15.0	91 %
SOCIAL	orkut	3 072 441	117 185 083	76.3	91 %
RANDOM	ER-Gnm_1M-20	1 000 000	10 000 000	20.0	93 %
WORD-REL.	Thesaurus	23 132	297 094	25.7	93 %

RESULTS

- Some networks are very close from cographs
- Random graphs are never
- A wide range of proximity : 12% to 93%
- The proximity with cographs highly depends on the real-world context

Cograph edition of real-world graphs

Close to cographs

www

software

Context	Network	n	m	d°	%mod
WWW	in-2004	1 148 875	12 281 937	21.4	12 %
WWW	cnr-2000	227 058	2 187 201	19.3	19 %
PROTEIN	reactome	5 973	145 778	48.8	22 %
SOFTWARE	jdk	6 434	53 658	16.7	29 %
SOFTWARE	jung-j	6 120	50 290	16.4	29 %
WWW	eu-2005	835 044	15 718 784	37.7	29 %
CO-AUTHOR	ca-GrQc	4 158	13 422	6.5	34 %
CO-AUTHOR	ca-HepPh	11 204	117 619	21.0	34 %
SPECIES	foodweb	183	2 434	26.6	43 %
CO-AUTHOR	dblp	317 080	1 049 866	6.6	45 %
WORD-REL.	wordnet	145 145	656 230	9.0	48 %
COMMUNIC.	wiki-Talk	2 388 953	4 656 682	3.9	49 %
CO-SOLD	amazon	334 863	925 872	5.5	49 %
CO-AUTHOR	ca-CondMat	21 363	91 286	8.6	52 %
RANDOM	ER-Gnm_1M-2	796 208	958 827	2.4	52 %
CO-AUTHOR	ca-HepTh	8 638	24 806	5.7	54 %
INTERNET	as2000	6 474	12 572	3.9	54 %
ROAD	roadNet-TX	1 351 137	1 879 201	2.8	54 %
INTERNET	as-caida2007	26 475	53 381	4.0	55 %
CO-AUTHOR	ca-AstroPh	17 903	196 972	22.0	59 %
INTERNET	topology	34 761	107 720	6.2	61 %
RANDOM	ER-Gnm_1M-3	940 987	1 494 643	3.2	63 %
INTERNET	as-skitter	1 694 616	11 094 209	13.1	64 %
CO-OCCUR	bible-names	1 707	9 059	10.6	67 %
PROTEIN	figeys	2 217	6 418	5.8	67 %
CITATION-SCI.	cora	23 166	89 157	7.7	68 %
SOCIAL	youtube	1 134 890	2 987 624	5.3	69 %
CO-ACTOR	actor-col.	374 511	15 014 839	80.2	71 %
P2P-CONNECT.	p2p-Gnutella	62 561	147 878	4.7	71 %
RANDOM	ER-Gnm_1M-4	980 191	1 999 203	4.1	71 %
CITATION-SCI.	citeseer	365 154	1 721 981	9.4	75 %
CITATION-PAT.	cit-Patents	3 764 117	16 511 740	8.8	76 %
SOFTWARE	linux	30 817	213 208	13.8	77 %
SOCIAL	LiveJournal	3 997 962	34 681 189	17.4	78 %
CITATION-SCI.	cit-HepTh	27 400	352 021	25.7	79 %
RANDOM	ER-Gnm_1M-6	997 479	2 999 988	6.0	79 %
CITATION-SCI.	cit-HepPh	34 401	420 784	24.5	81 %
RANDOM	ER-Gnm_1M-8	999 684	3 999 999	8.0	84 %
RANDOM	ER-Gnm_1M-10	999 952	5 000 000	10.0	87 %
RANDOM	ER-Gnm_1M-15	1 000 000	7 500 000	15.0	91 %
SOCIAL	orkut	3 072 441	117 185 083	76.3	91 %
RANDOM	ER-Gnm_1M-20	1 000 000	10 000 000	20.0	93 %
WORD-REL.	Thesaurus	23 132	297 094	25.7	93 %

- The proximity with cographs highly depends on the real-world context

Cograph edition of real-world graphs

Not close not far

internet

road

Context	Network	n	m	d°	%mod
WWW	in-2004	1 148 875	12 281 937	21.4	12 %
WWW	cnr-2000	227 058	2 187 201	19.3	19 %
PROTEIN	reactome	5 973	145 778	48.8	22 %
SOFTWARE	jdk	6 434	53 658	16.7	29 %
SOFTWARE	jung-j	6 120	50 290	16.4	29 %
WWW	eu-2005	835 044	15 718 784	37.7	29 %
CO-AUTHOR	ca-GrQc	4 158	13 422	6.5	34 %
CO-AUTHOR	ca-HepPh	11 204	117 619	21.0	34 %
SPECIES	foodweb	183	2 434	26.6	43 %
CO-AUTHOR	dblp	317 080	1 049 866	6.6	45 %
WORD-REL.	wordnet	145 145	656 230	9.0	48 %
COMMUNIC.	wiki-Talk	2 388 953	4 656 682	3.9	49 %
CO-SOLD	amazon	334 863	925 872	5.5	49 %
CO-AUTHOR	ca-CondMat	21 363	91 286	8.6	52 %
RANDOM	ER-Gnm_1M-2	796 208	958 827	2.4	52 %
CO-AUTHOR	ca-HepTh	8 638	24 806	5.7	54 %
INTERNET	as2000	6 474	12 572	3.9	54 %
ROAD	roadNet-TX	1 351 137	1 879 201	2.8	54 %
INTERNET	as-caida2007	26 475	53 381	4.0	55 %
CO-AUTHOR	ca-AstroPh	17 903	196 972	22.0	59 %
INTERNET	topology	34 761	107 720	6.2	61 %
RANDOM	ER-Gnm_1M-3	940 987	1 494 643	3.2	63 %
INTERNET	as-skitter	1 694 616	11 094 209	13.1	64 %
CO-OCCUR	bible-names	1 707	9 059	10.6	67 %
PROTEIN	figeys	2 217	6 418	5.8	67 %
CITATION-SCI.	cora	23 166	89 157	7.7	68 %
SOCIAL	youtube	1 134 890	2 987 624	5.3	69 %
CO-ACTOR	actor-col.	374 511	15 014 839	80.2	71 %
P2P-CONNECT.	p2p-Gnutella	62 561	147 878	4.7	71 %
RANDOM	ER-Gnm_1M-4	980 191	1 999 203	4.1	71 %
CITATION-SCI.	citeseer	365 154	1 721 981	9.4	75 %
CITATION-PAT.	cit-Patents	3 764 117	16 511 740	8.8	76 %
SOFTWARE	linux	30 817	213 208	13.8	77 %
SOCIAL	LiveJournal	3 997 962	34 681 189	17.4	78 %
CITATION-SCI.	cit-HepTh	27 400	352 021	25.7	79 %
RANDOM	ER-Gnm_1M-6	997 479	2 999 988	6.0	79 %
CITATION-SCI.	cit-HepPh	34 401	420 784	24.5	81 %
RANDOM	ER-Gnm_1M-8	999 684	3 999 999	8.0	84 %
RANDOM	ER-Gnm_1M-10	999 952	5 000 000	10.0	87 %
RANDOM	ER-Gnm_1M-15	1 000 000	7 500 000	15.0	91 %
SOCIAL	orkut	3 072 441	117 185 083	76.3	91 %
RANDOM	ER-Gnm_1M-20	1 000 000	10 000 000	20.0	93 %
WORD-REL.	Thesaurus	23 132	297 094	25.7	93 %

- The proximity with cographs highly depends on the real-world context

Cograph edition of real-world graphs

Context	Network	n	m	d°	%mod
WWW	in-2004	1 148 875	12 281 937	21.4	12 %
WWW	cnr-2000	227 058	2 187 201	19.3	19 %
PROTEIN	reactome	5 973	145 778	48.8	22 %
SOFTWARE	jdk	6 434	53 658	16.7	29 %
SOFTWARE	jung-j	6 120	50 290	16.4	29 %
WWW	eu-2005	835 044	15 718 784	37.7	29 %
CO-AUTHOR	ca-GrQc	4 158	13 422	6.5	34 %
CO-AUTHOR	ca-HepPh	11 204	117 619	21.0	34 %
SPECIES	foodweb	183	2 434	26.6	43 %
CO-AUTHOR	dblp	317 080	1 049 866	6.6	45 %
WORD-REL.	wordnet	145 145	656 230	9.0	48 %
COMMUNIC.	wiki-Talk	2 388 953	4 656 682	3.9	49 %
CO-SOLD	amazon	334 863	925 872	5.5	49 %
CO-AUTHOR	ca-CondMat	21 363	91 286	8.6	52 %
RANDOM	ER-Gnm_1M-2	796 208	958 827	2.4	52 %
CO-AUTHOR	ca-HepTh	8 638	24 806	5.7	54 %
INTERNET	as2000	6 474	12 572	3.9	54 %
ROAD	roadNet-TX	1 351 137	1 879 201	2.8	54 %
INTERNET	as-caida2007	26 475	53 381	4.0	55 %
CO-AUTHOR	ca-AstroPh	17 903	196 972	22.0	59 %
INTERNET	topology	34 761	107 720	6.2	61 %
RANDOM	ER-Gnm_1M-3	940 987	1 494 643	3.2	63 %
INTERNET	as-skitter	1 694 616	11 094 209	13.1	64 %
CO-OCCUR	bible-names	1 707	9 059	10.6	67 %
PROTEIN	figeys	2 217	6 418	5.8	67 %
CITATION-SCI.	cora	23 166	89 157	7.7	68 %
SOCIAL	youtube	1 134 890	2 987 624	5.3	69 %
CO-ACTOR	actor-col.	374 511	15 014 839	80.2	71 %
P2P-CONNECT.	p2p-Gnutella	62 561	147 878	4.7	71 %
RANDOM	ER-Gnm_1M-4	980 191	1 999 203	4.1	71 %
CITATION-SCI.	citeseer	365 154	1 721 981	9.4	75 %
CITATION-PAT.	cit-Patents	3 764 117	16 511 740	8.8	76 %
SOFTWARE	linux	30 817	213 208	13.8	77 %
SOCIAL	LiveJournal	3 997 962	34 681 189	17.4	78 %
CITATION-SCI.	cit-HepTh	27 400	352 021	25.7	79 %
RANDOM	ER-Gnm_1M-6	997 479	2 999 988	6.0	79 %
CITATION-SCI.	cit-HepPh	34 401	420 784	24.5	81 %
RANDOM	ER-Gnm_1M-8	999 684	3 999 999	8.0	84 %
RANDOM	ER-Gnm_1M-10	999 952	5 000 000	10.0	87 %
RANDOM	ER-Gnm_1M-15	1 000 000	7 500 000	15.0	91 %
SOCIAL	orkut	3 072 441	117 185 083	76.3	91 %
RANDOM	ER-Gnm_1M-20	1 000 000	10 000 000	20.0	93 %
WORD-REL.	Thesaurus	23 132	297 094	25.7	93 %

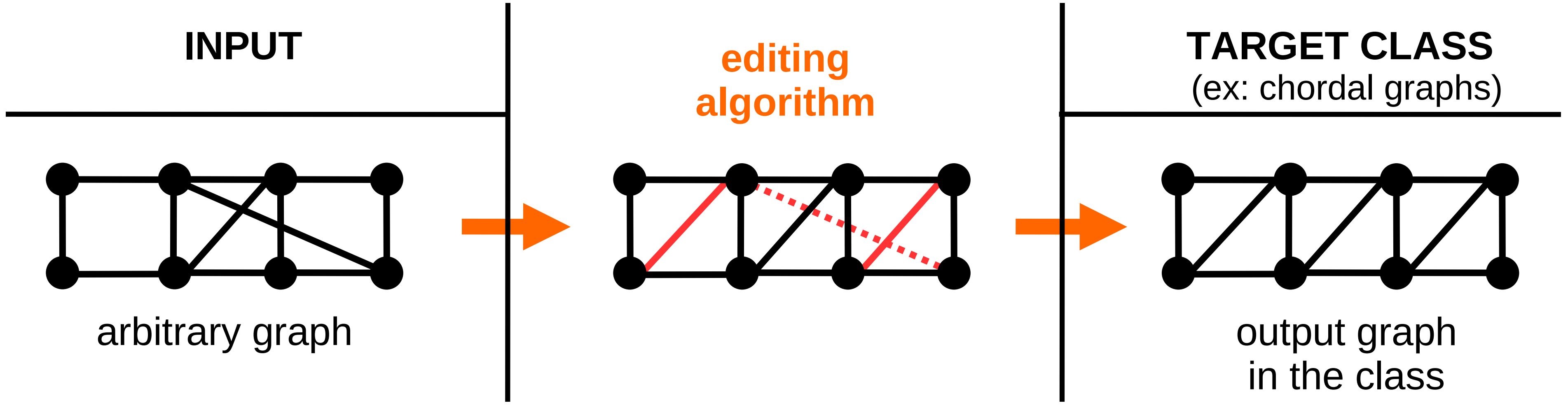
Far from cographs

- citation
- social

- The proximity with cographs highly depends on the real-world context

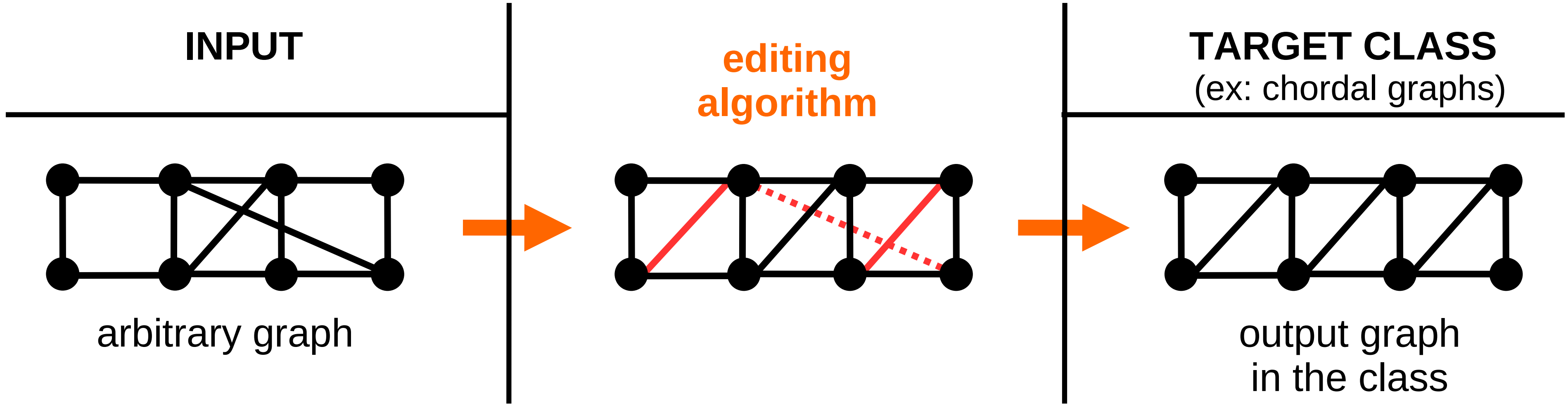
Graph editing algorithms

Graph editing algorithms



GOAL: perform as few modifications as possible

Graph editing algorithms

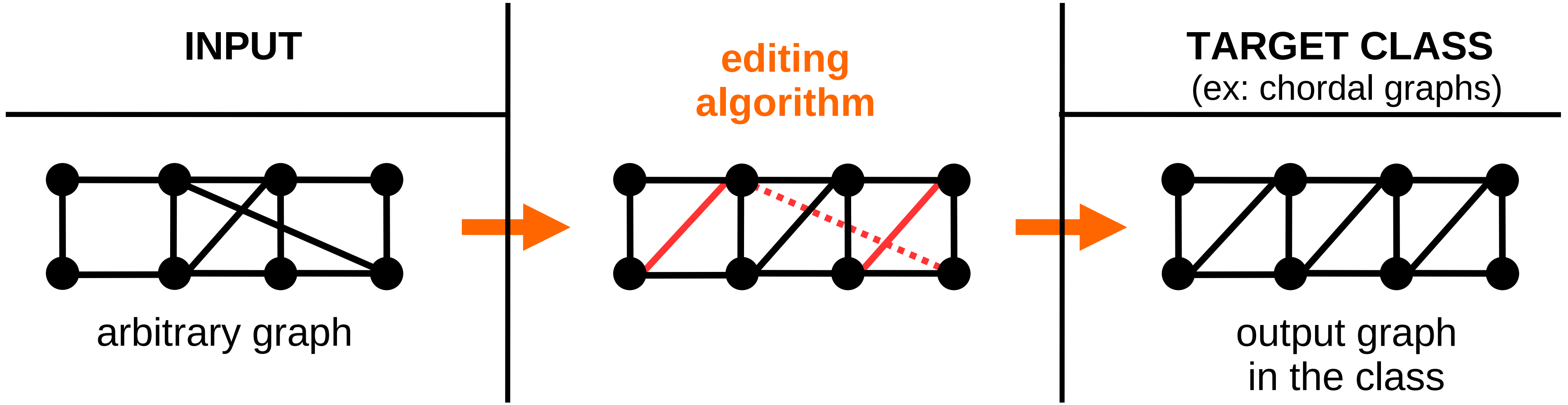


GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for most properties

Even when only one type of modifications is allowed

Graph editing algorithms



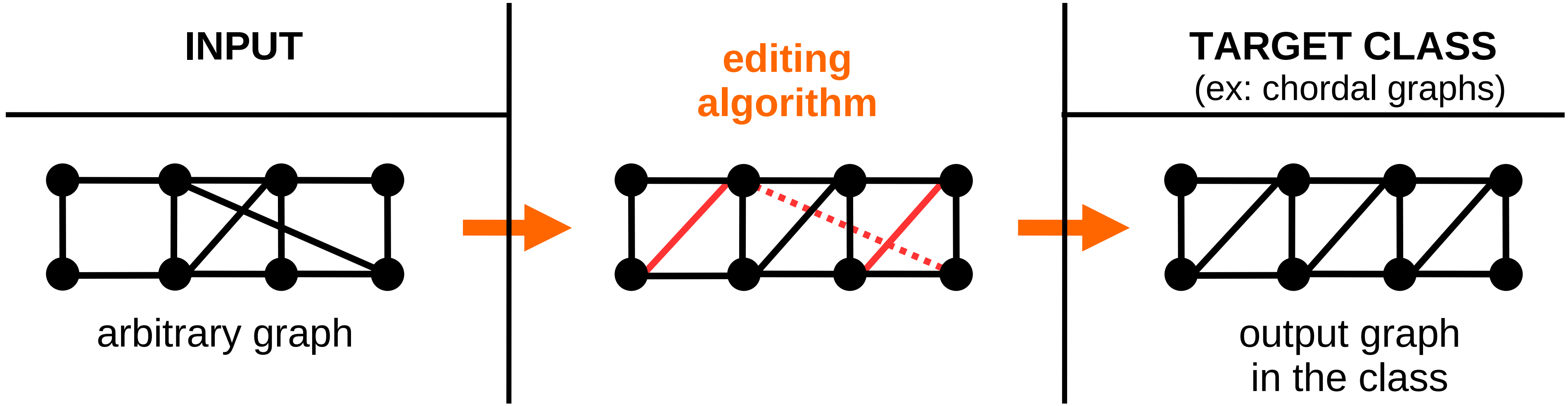
GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for most properties

  Even when only one type of modifications is allowed

- Different approaches:
  - Restricted inputs
  - Exact exponential algorithms
  - Parameterized algorithms
  - Approximation algorithms
  - Inclusion minimal modification

Graph editing algorithms



GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for most properties

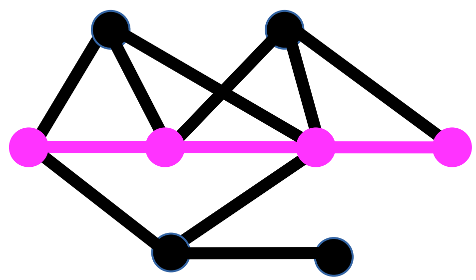
  Even when only one type of modifications is allowed

- Different approaches:
  - Restricted inputs
  - Exact exponential algorithms
  - *Parameterized algorithms (1st lecture)*
  - Approximation algorithms
  - *Inclusion minimal modification (2nd lecture)*

Cographs

Cographs

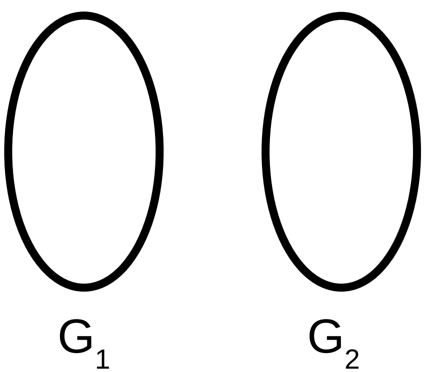
1. Characterization by forbiden subgraphs:



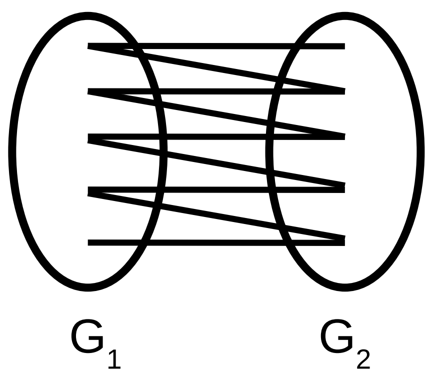
no induced P_4
(path on 4 vertices)

2. Obtained from single vertices by using two operations:

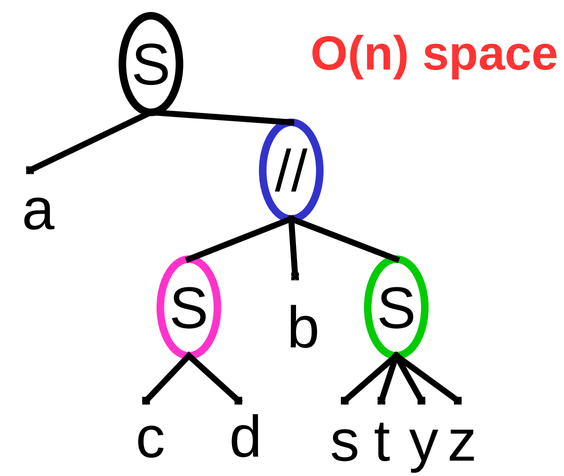
disjoint union
(//)



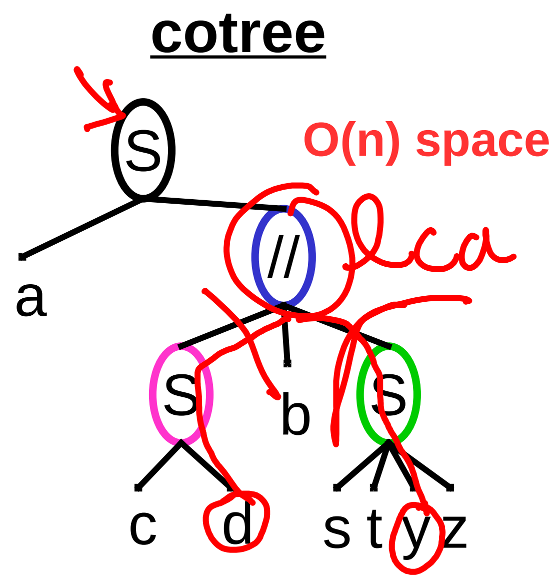
complete union
(S)



cotree



Cographs



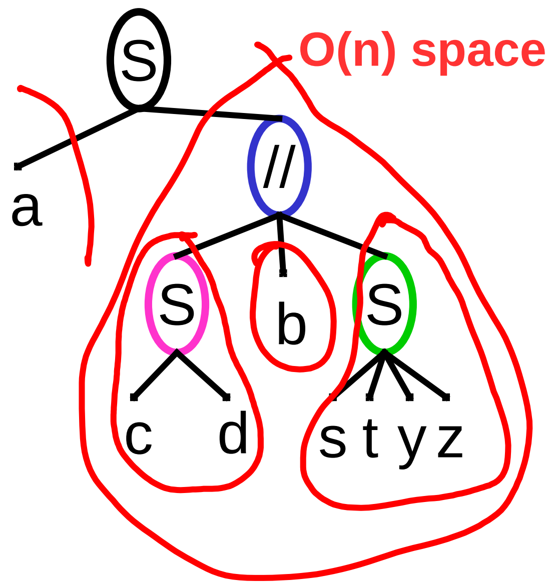
Exercise:

Is ***d*** adjacent to ***y*** ? non-adjacent.

Is ***a*** adjacent to ***t*** ? adjacent

Cographs

cotree



Exercise:

Is ***d*** adjacent to ***y*** ?

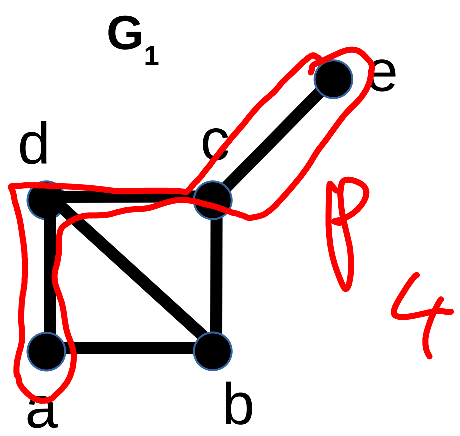
Is ***a*** adjacent to ***t*** ?

Answer:

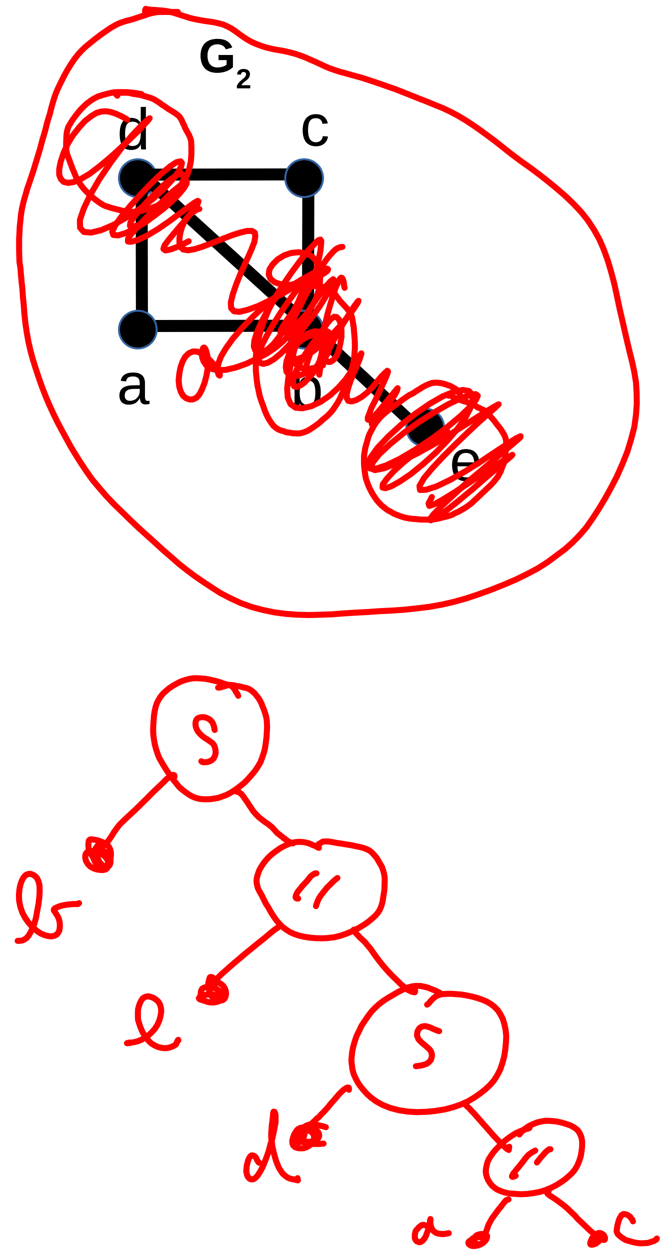
- Find the ***lowest common ancestor*** of the two leaves
- ***//*** : non-adjacent
  S : adjacent

Cographs

Exercise: Are these two graphs cographs ?

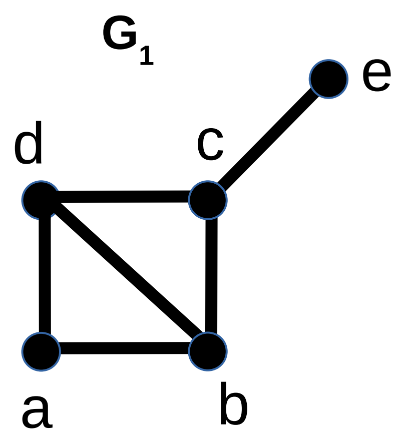


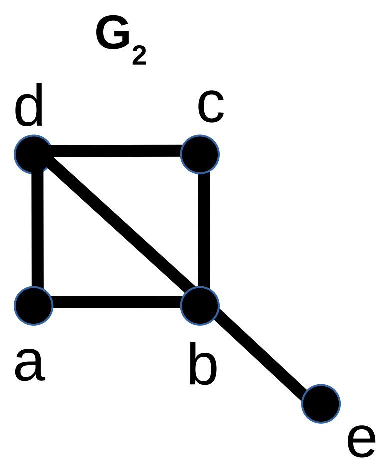
not a cograph.



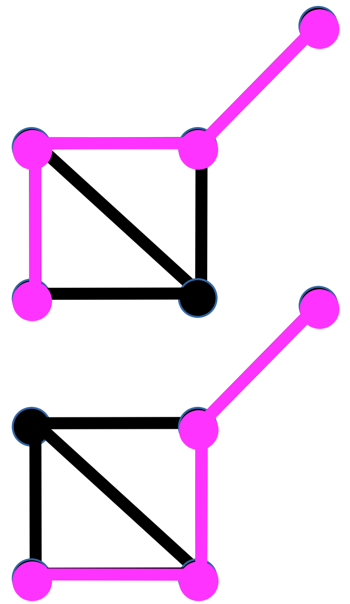
Cographs

Exercise: Are these two graphs cographs ?





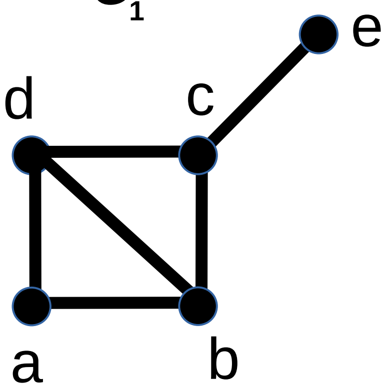
A P_4 in G_1



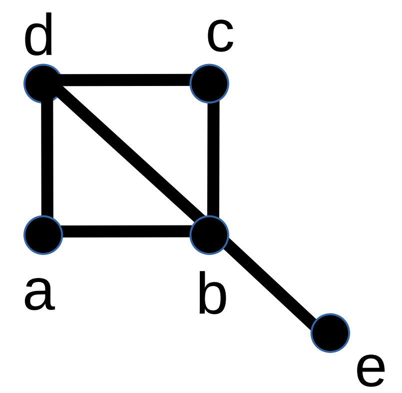
Cographs

Exercise: Are these two graphs cographs ?

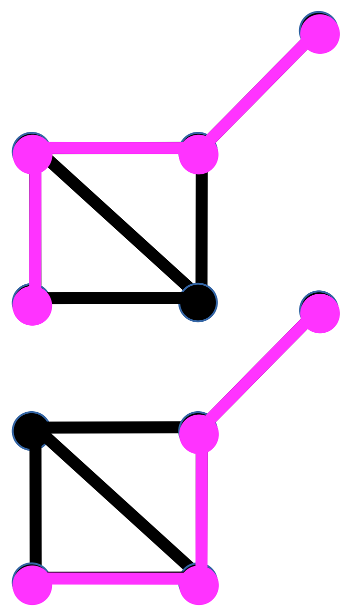
G_1



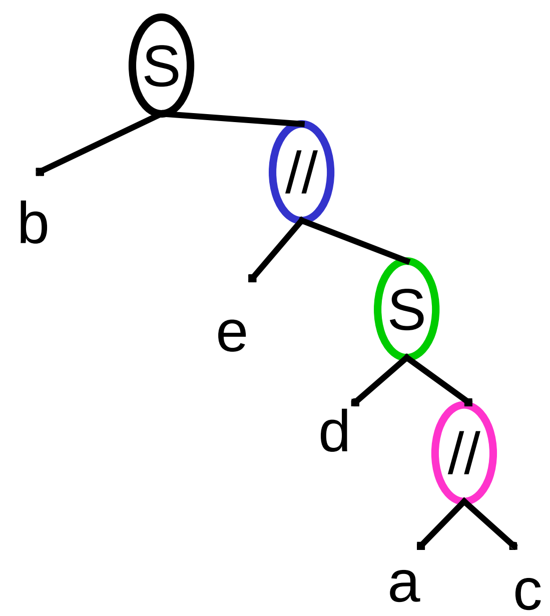
G_2



A P_4 in G_1

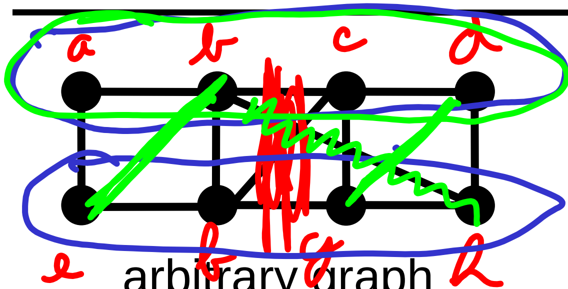


Cotree of G_2

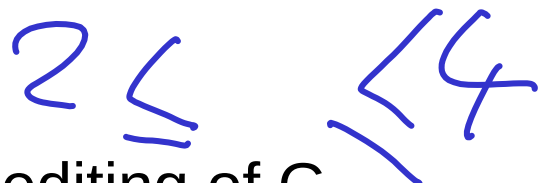


Coraph editing

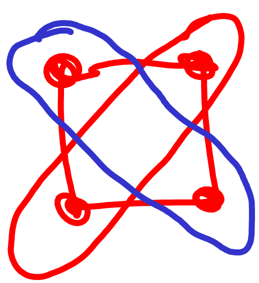
INPUT	Editing ???	TARGET CLASS: Cographs

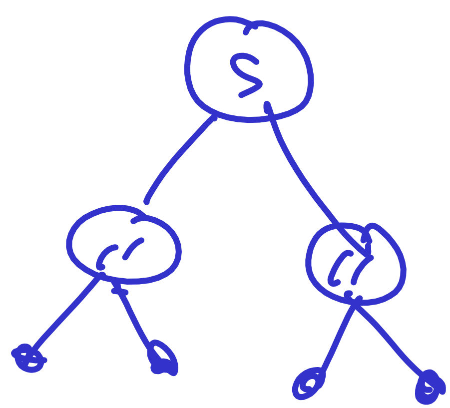


arbitrary graph
G



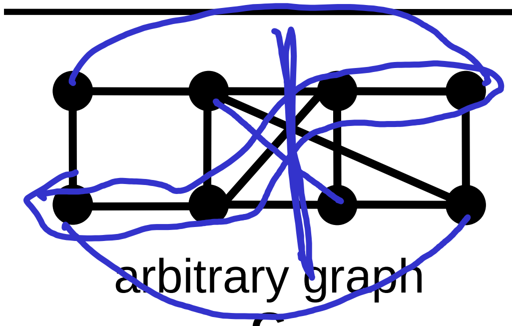
Exercise:
Give a minimum cograph editing of G





Coraph editing

INPUT	Editing ???	TARGET CLASS: Cographs

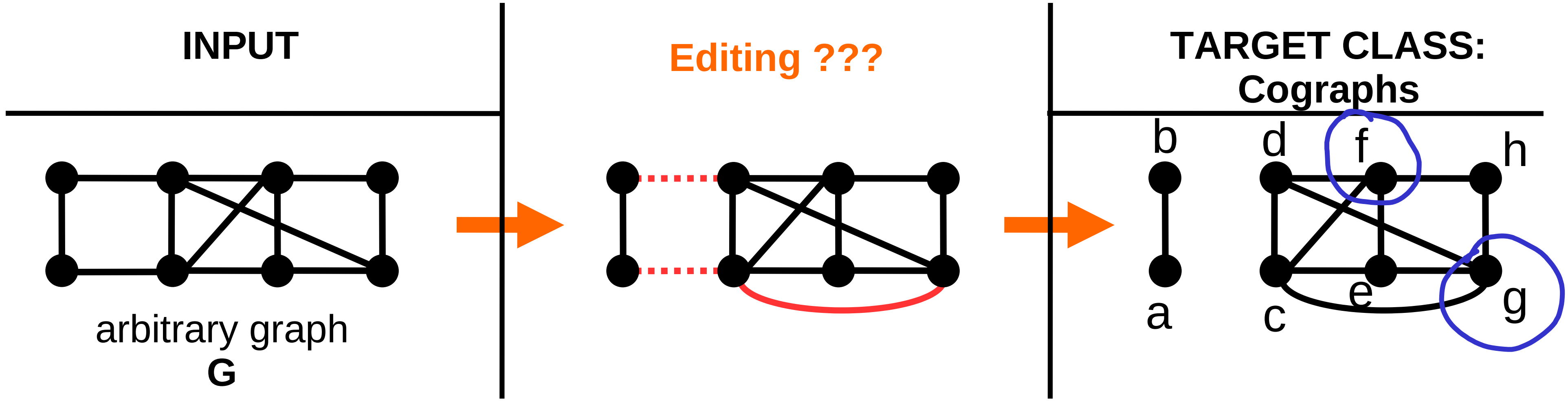


arbitrary graph **G**

Exercise:

Give a minimum cograph editing of G

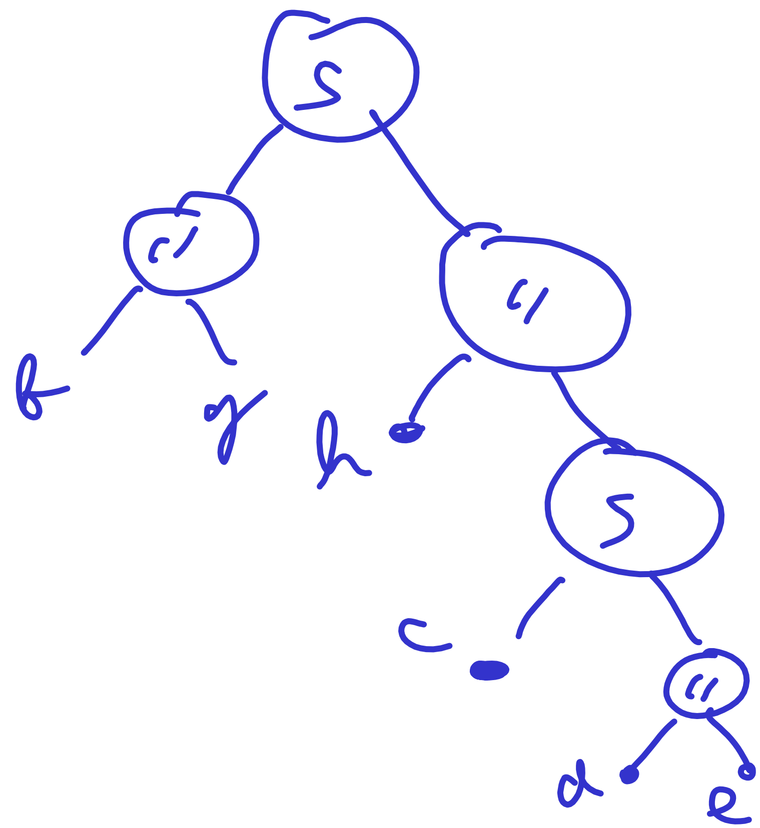
Coraph editing



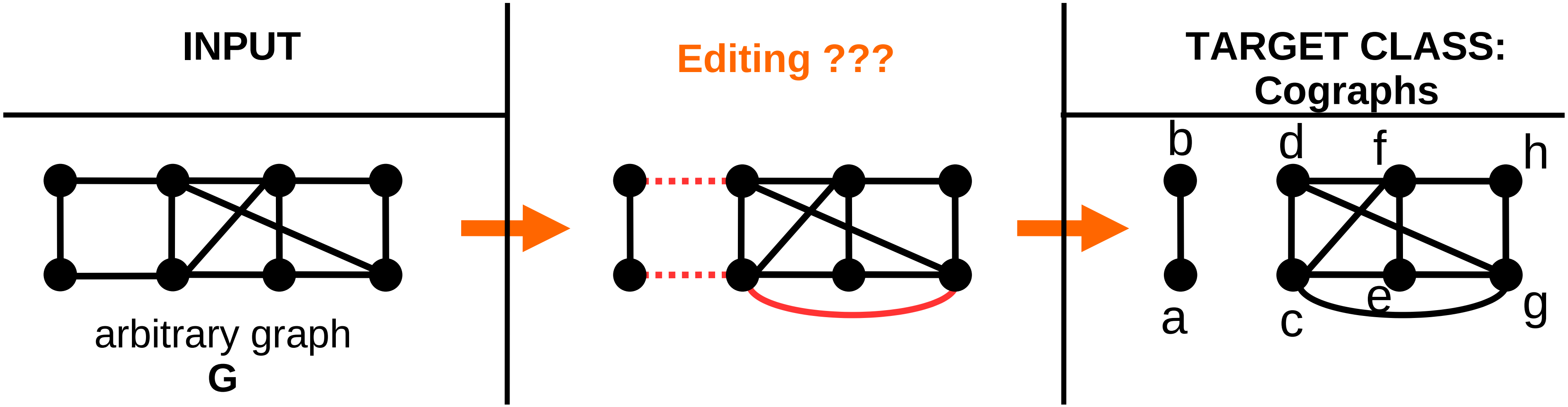
Exercise:

Give a minimum cograph editing of G

- **3 modifications are enough**



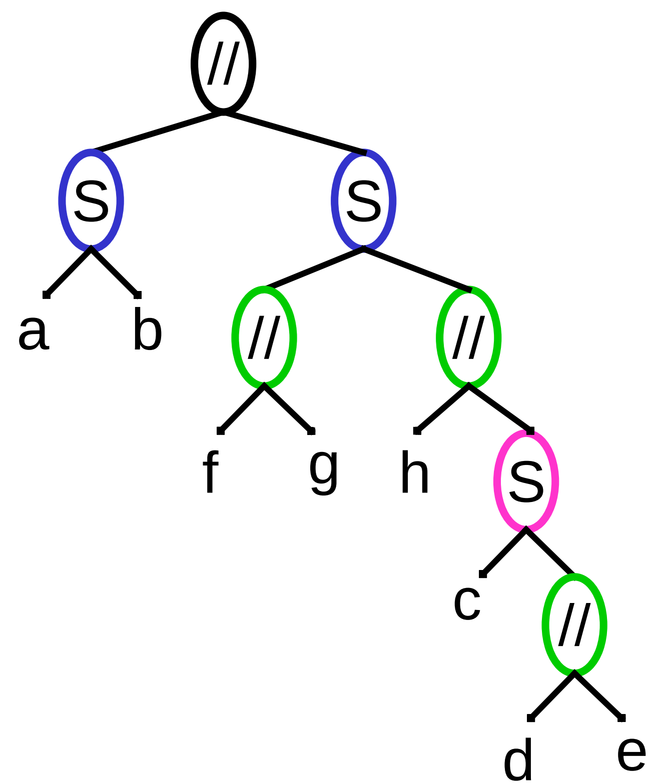
Coraph editing



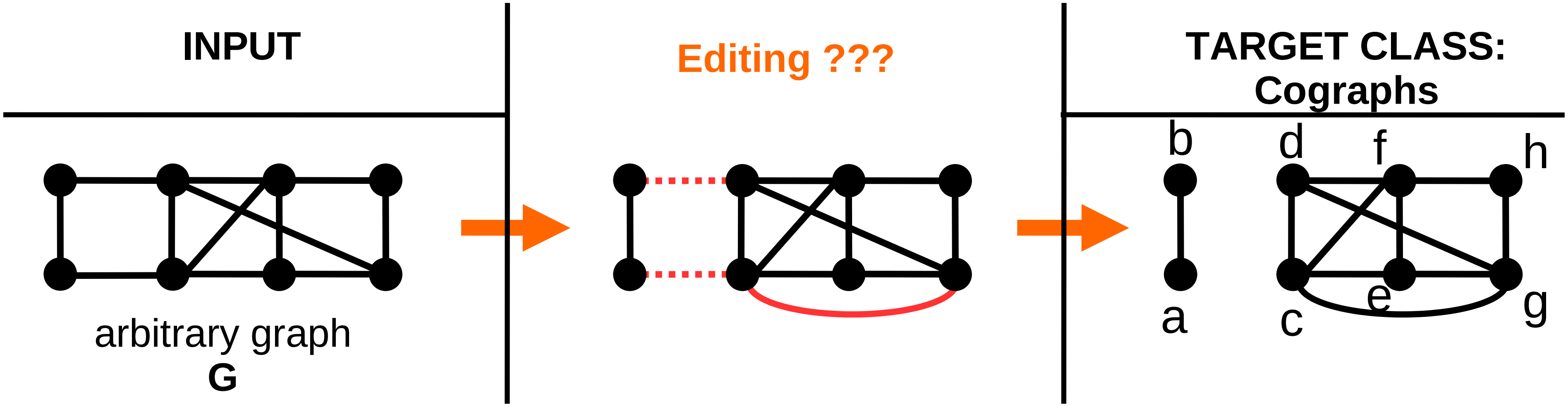
Exercise:

Give a minimum cograph editing of G

- **3 modifications are enough**



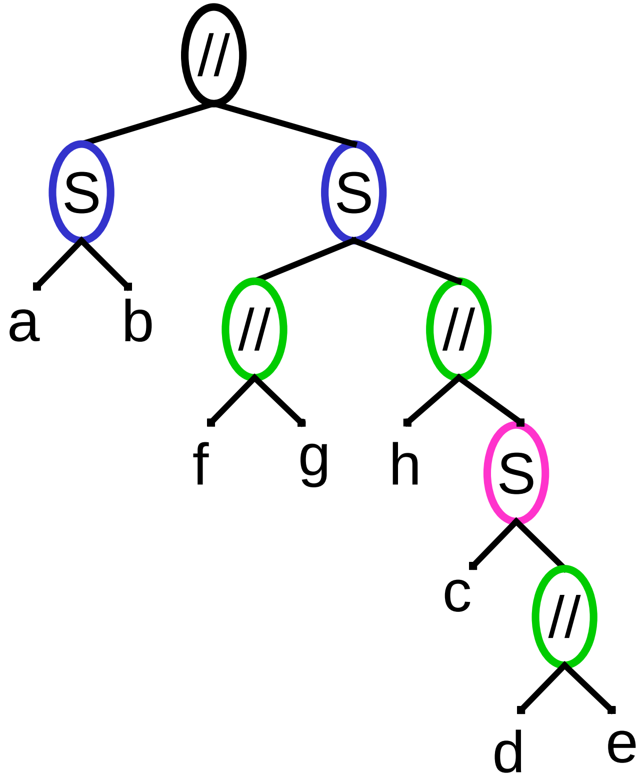
Coraph editing



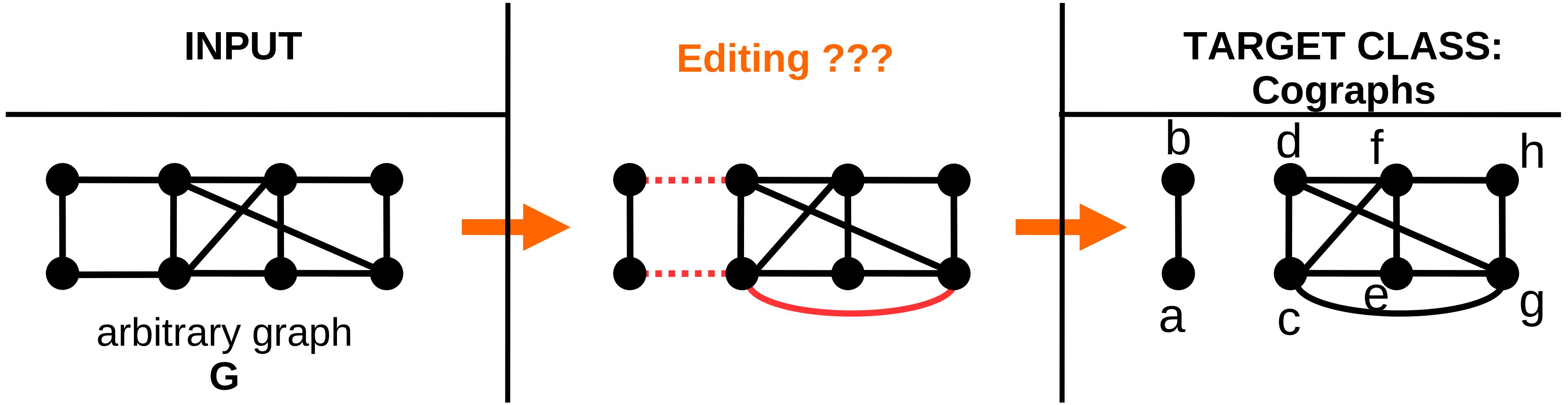
Exercise:

Give a minimum cograph editing of G

- **3 modifications are enough**
- Can you do it with 2 modifications only?



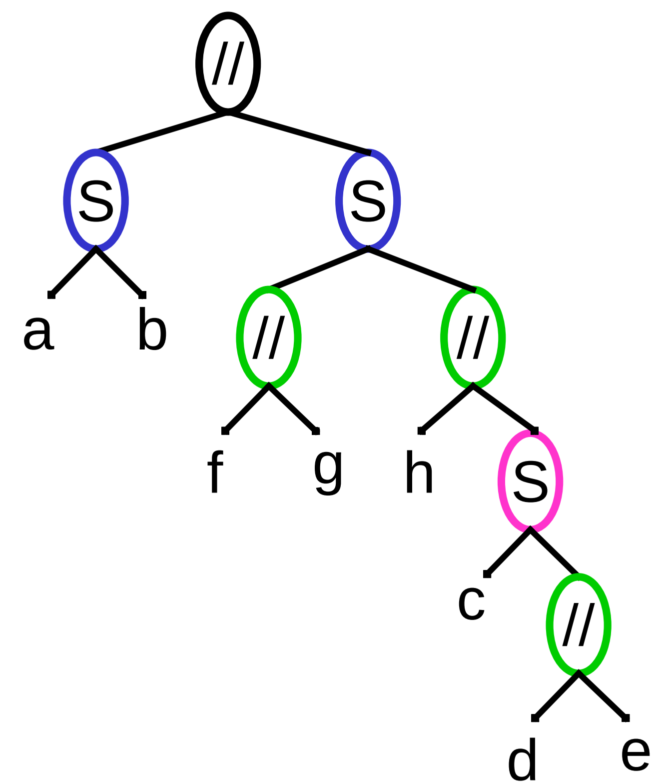
Coraph editing

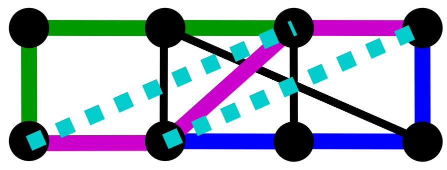


Exercise:

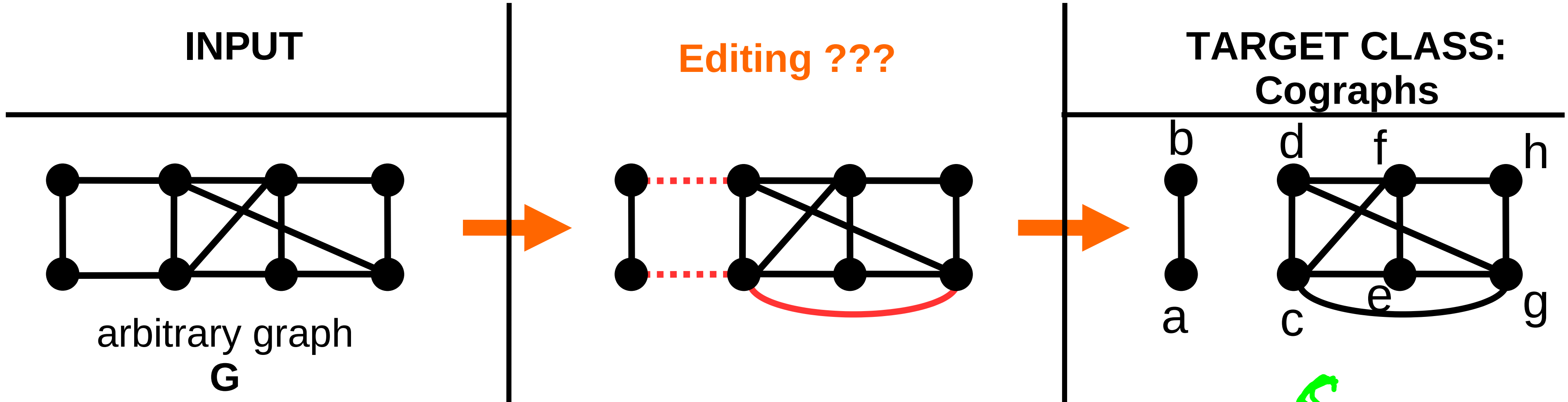
Give a minimum cograph editing of G

- **3 modifications are enough**
- Can you do it with 2 modifications only?





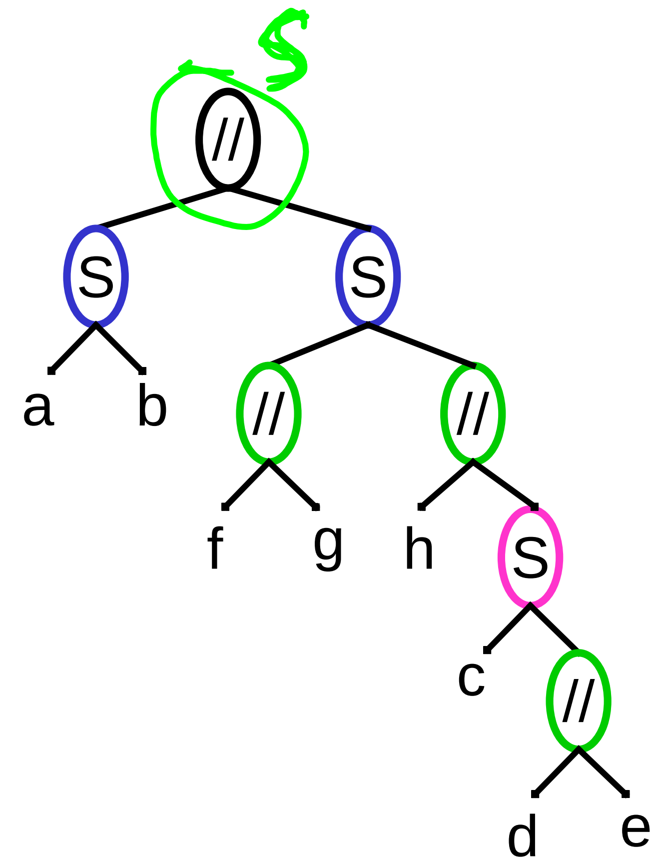
Coraph editing

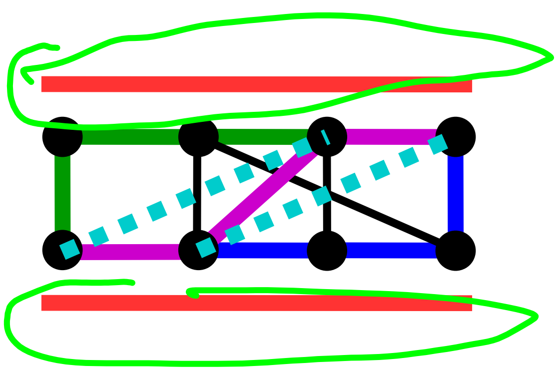


Exercise:

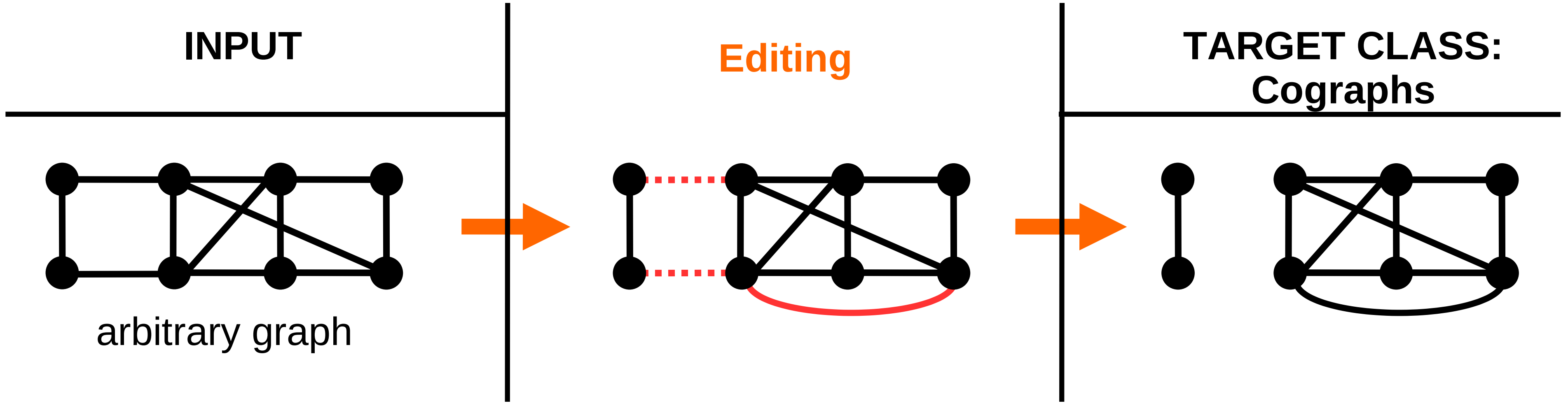
Give a minimum cograph editing of G

- **3 modifications are enough**
- Can you do it with 2 modifications only?





Coraph editing

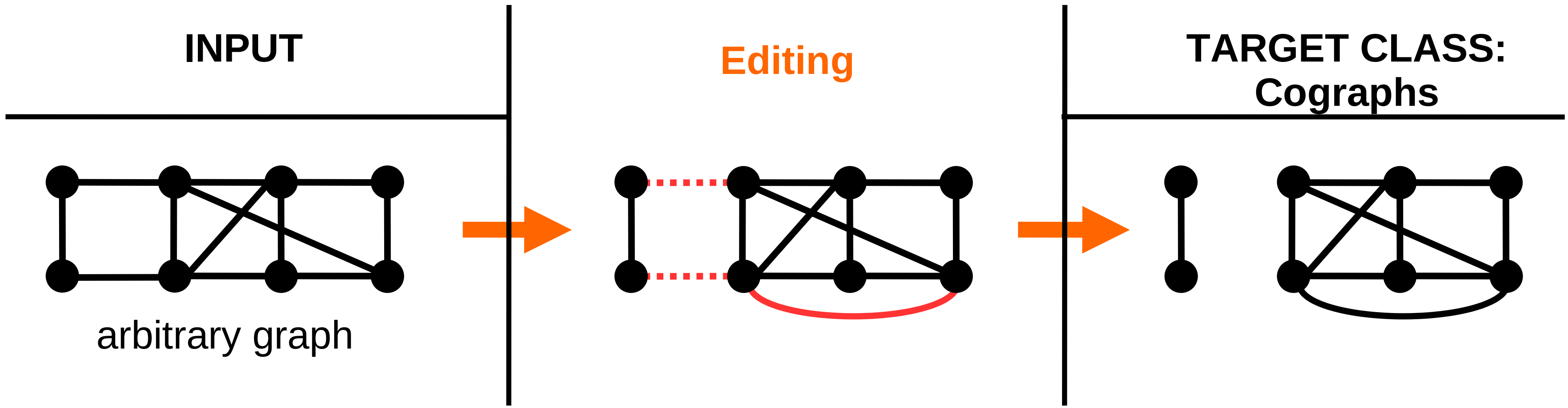


GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for cograph editing

Even when only one type of modifications is allowed

Coraph editing



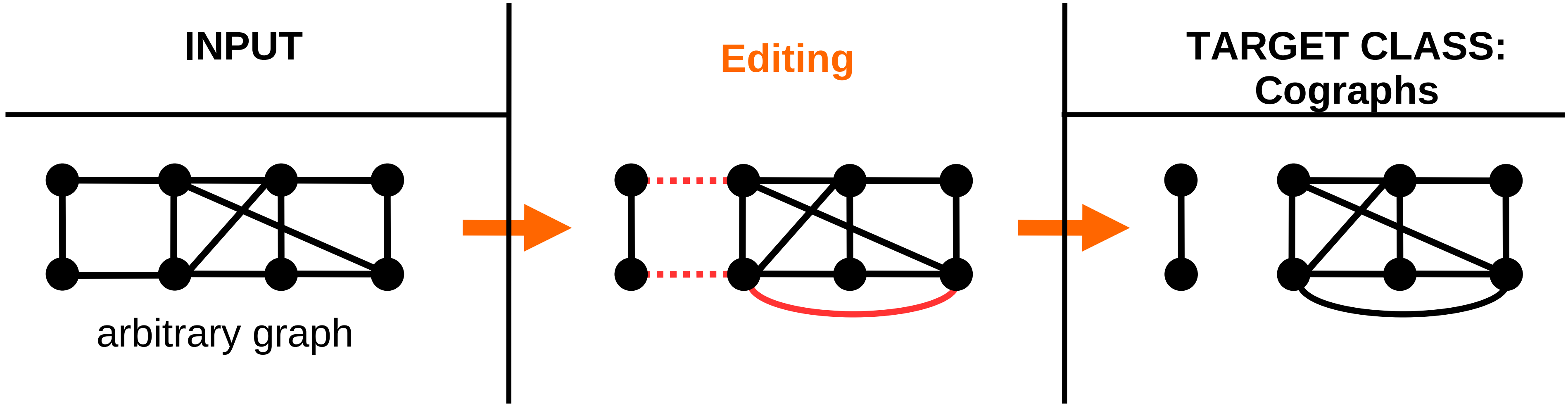
GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for cograph editing

  Even when only one type of modifications is allowed

- Are cographs a complicate class of graphs?

Coraph editing



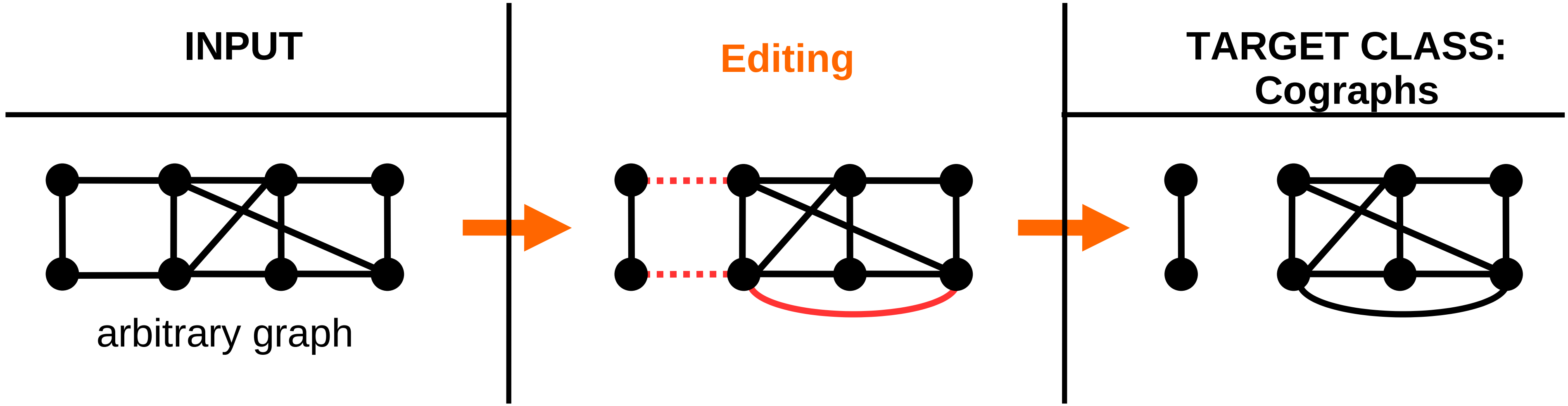
GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for cograph editing

  Even when only one type of modifications is allowed

- Are cographs a complicate class of graphs?
  - Need a criterion : propositions?

Coraph editing



GOAL: perform as few modifications as possible

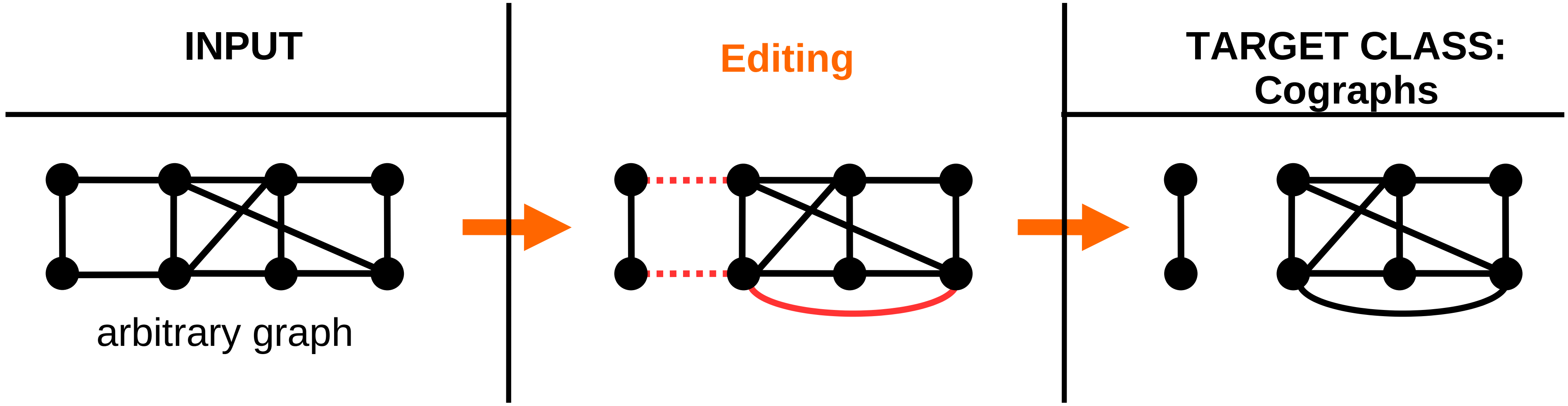
- Unfortunately: *minimum number* is ***NP-hard*** for cograph editing

  Even when only one type of modifications is allowed

- Are cographs a complicate class of graphs?
  - Need a criterion : propositions?

    Number of graphs in the class with n vertices ↔ size of the representation

Coraph editing



GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for cograph editing

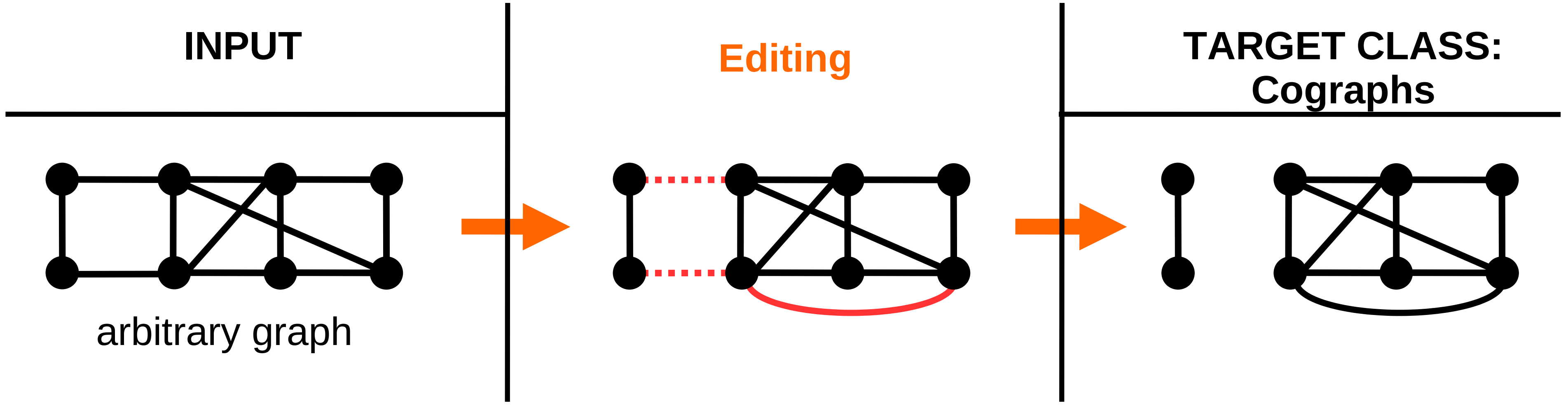
  Even when only one type of modifications is allowed

- Are cographs a complicate class of graphs?
  - Need a criterion : propositions?

    Number of graphs in the class with n vertices ↔ size of the representation

  - For labelled cographs: O(n) integers = O(n log n) bits

Coraph editing



GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for cograph editing

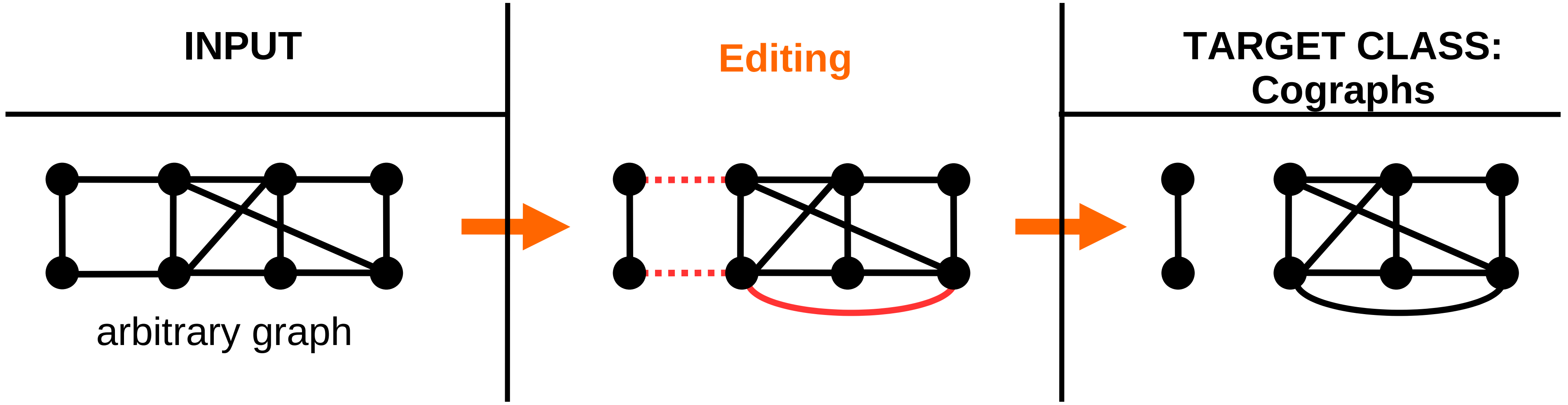
  Even when only one type of modifications is allowed

- Are cographs a complicate class of graphs?
  - Need a criterion : propositions?

    Number of graphs in the class with n vertices ↔ size of the representation

  - For labelled cographs: O(n) integers = O(n log n) bits
  - For graphs in general: $O(n^2)$ bits

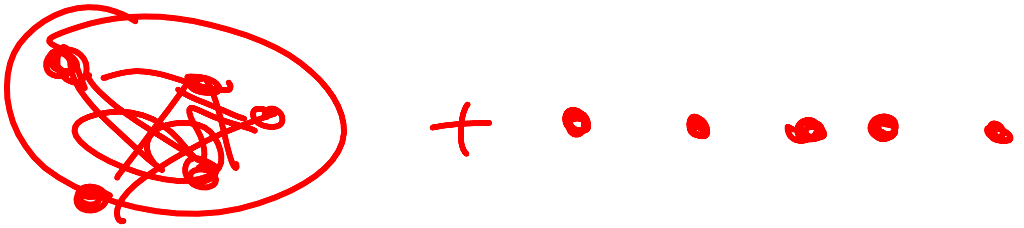
Coraph editing



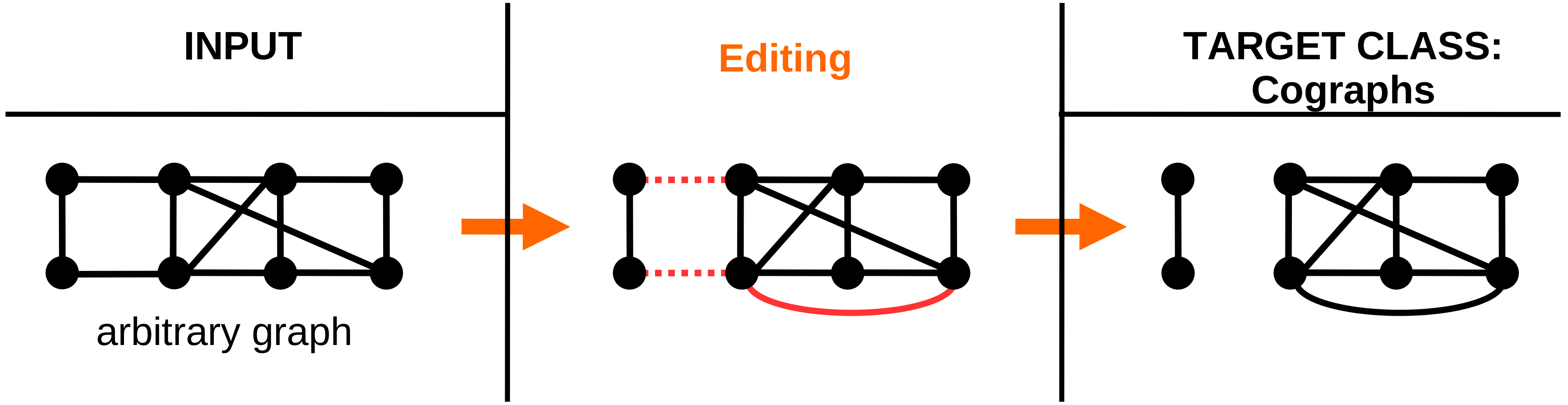
GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard***
  for clique + isolated vertices editing

- Even worse example: clique + isolated vertices



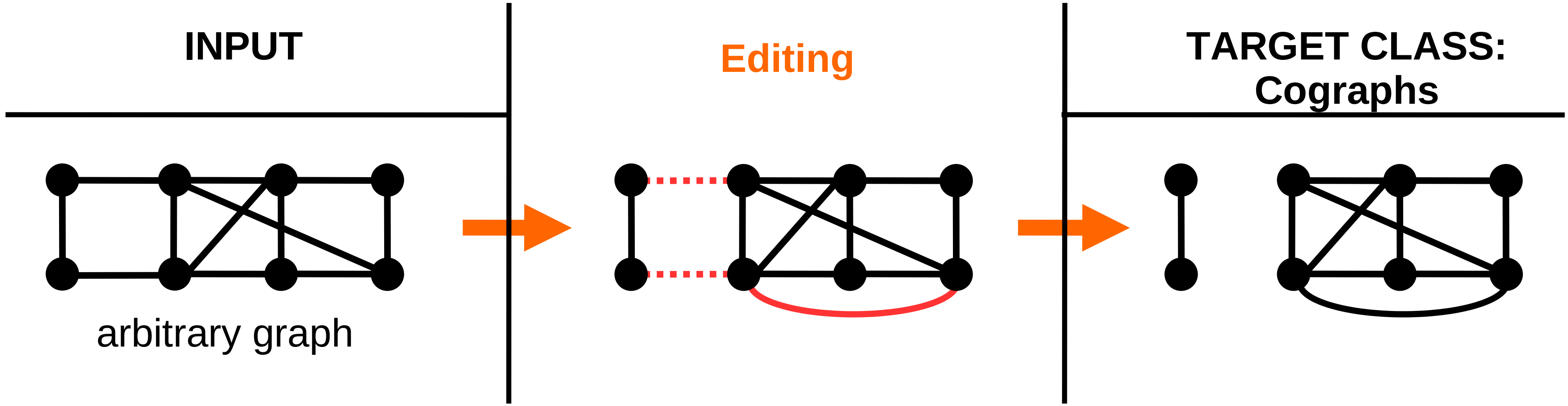
Coraph editing



GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for clique + isolated vertices editing
- Even worse example: clique + isolated vertices
  - Up to isomorphism: 1 integer = O(log n) bits

Coraph editing



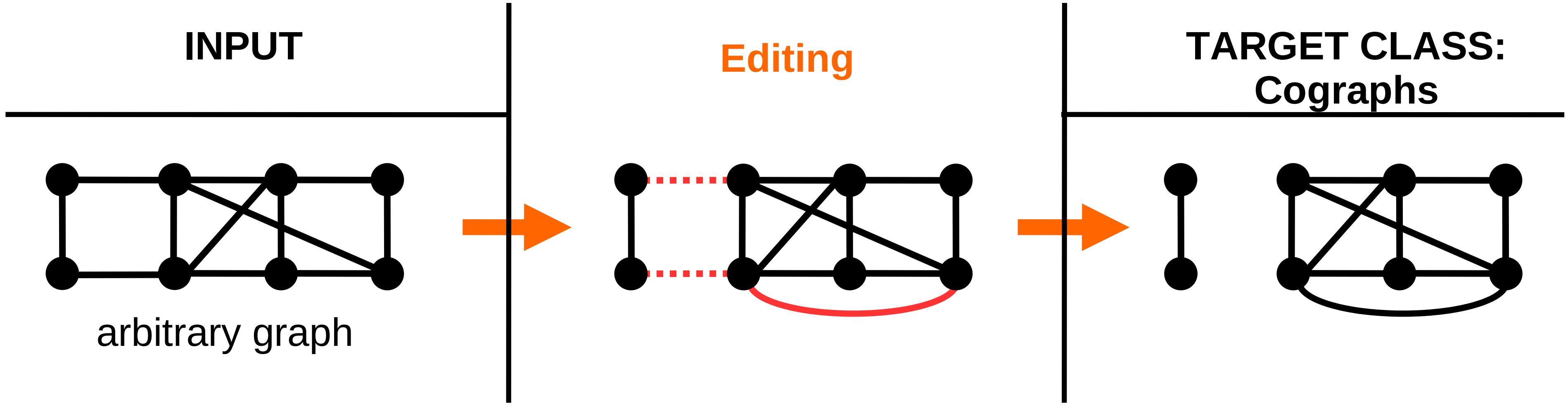
GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for clique + isolated vertices editing
- Even worse example: clique + isolated vertices
  - Up to isomorphism: 1 integer = O(log n) bits
  - For graphs in general: $O(n^2)$ bits

$\Omega(n^2)$

$\frac{2^{n^2}}{n!}$ $O(n^2 - n\log n)$

Coraph editing

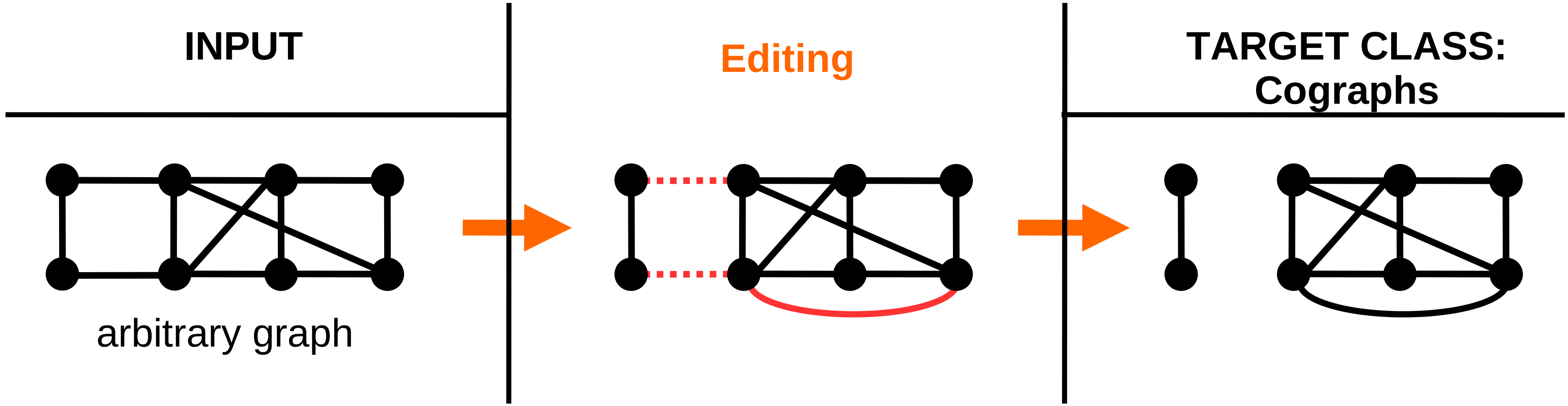


GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for clique + isolated vertices editing

Exercise:
Does it remain hard for pure completion ?
For pure deletion ?

Coraph editing

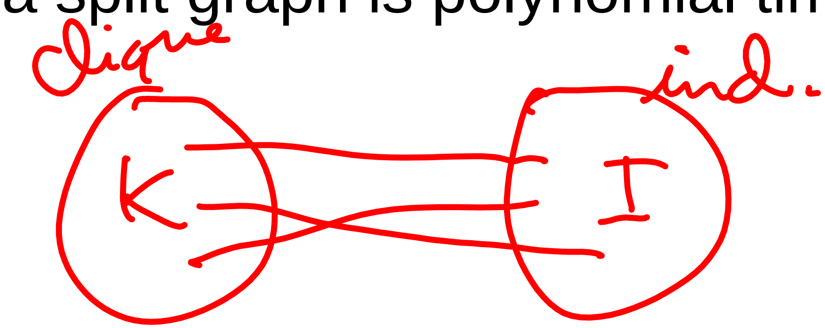


GOAL: perform as few modifications as possible

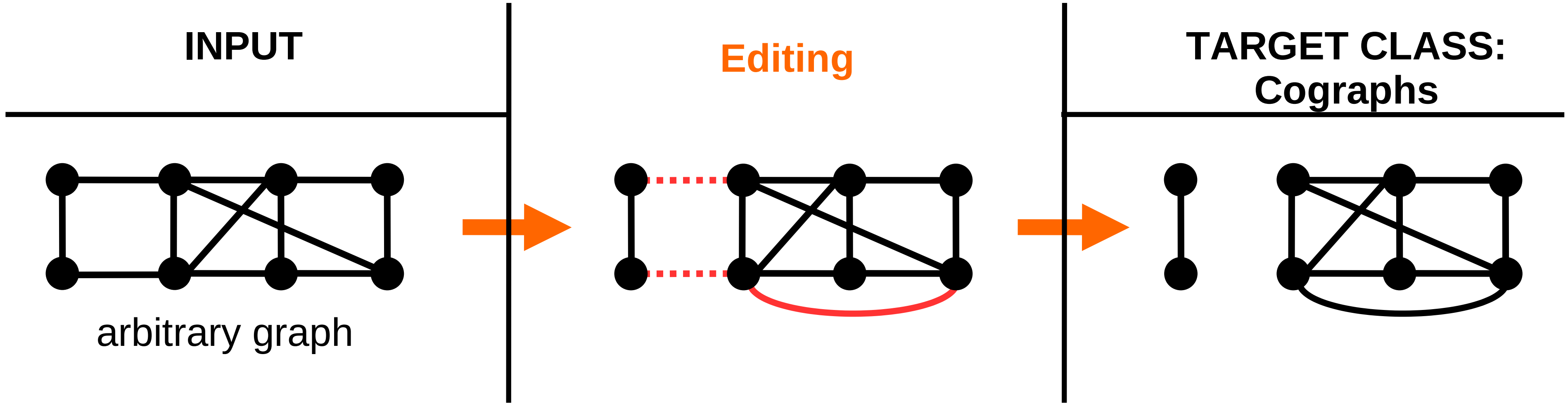
- Unfortunately: *minimum number* is ***NP-hard*** for clique + isolated vertices editing

<u>In general : no rule</u>

Minimum editing to a split graph is polynomial time solvable



Coraph editing



GOAL: perform as few modifications as possible

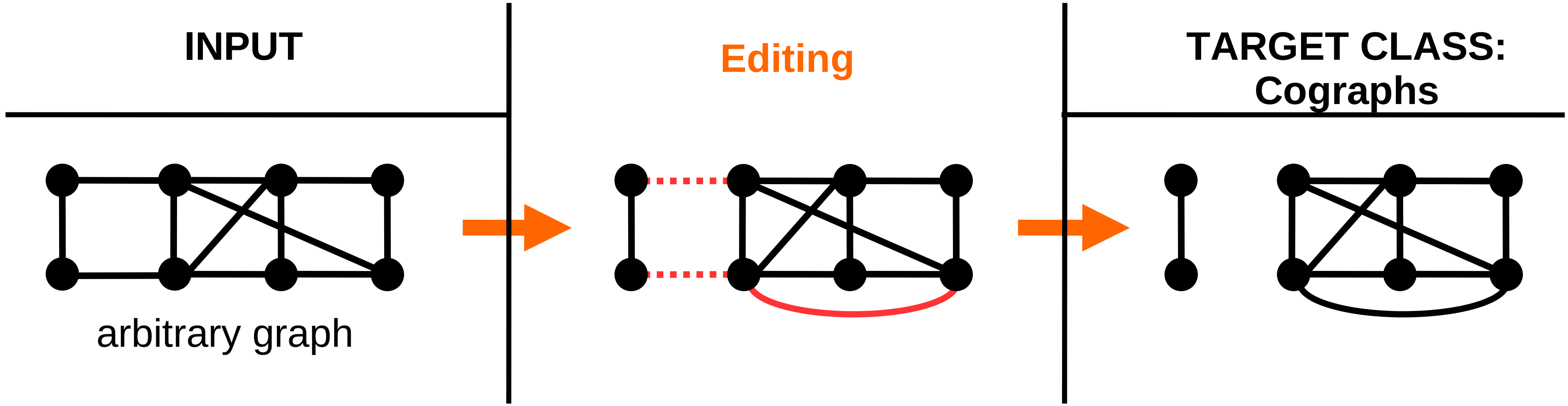
- Unfortunately: *minimum number* is ***NP-hard*** for clique + isolated vertices editing

In general : no rule

Minimum editing to a split graph is polynomial time solvable

Minimum completion and minimum deletion are NP-hard

Coraph editing



GOAL: perform as few modifications as possible

- Unfortunately: *minimum number* is ***NP-hard*** for cograph editing

  Even when only one type of modifications is allowed

- Different approaches:
  - Restricted inputs
  - Exact exponential algorithms
  - *Parameterized algorithms (1st lecture)*
  - Approximation algorithms
  - *Inclusion minimal modification (2nd lecture)*

Polynomial Kernels for Edge Modification Problems

Parameterized complexity

- <u>**Idea:**</u> the computational difficulty of treating an instance is not only due to its size: also depend on a relevant alternative **parameter k**

Parameterized complexity

- **Idea:** the computational difficulty of treating an instance is not only due to its size: also depend on a relevant alternative **parameter k**

- **Data Reduction: KERNEL**

An algorithm A that reduces an instance (I,k) to an instance (I',k') s.t.

- A runs in polynomial time (wrt. |I|)
- (I',k') is a YES-instance iff (I,k) is a YES-instance
- |I'| ≤ g(k) and k' ≤ k ➡ **|I'| depends only on k (not on |I|)**

Parameterized complexity

- **Idea:** the computational difficulty of treating an instance is not only due to its size: also depend on a relevant alternative **parameter k**

- **Data Reduction: KERNEL**

An algorithm A that reduces an instance (I,k) to an instance (I',k') s.t.

- A runs in polynomial time (wrt. |I|)
- (I',k') is a YES-instance iff (I,k) is a YES-instance
- |I'| ≤ **g(**k**)** and k' ≤ k → **|I'| depends only on k (not on |I|)**

POLYNOMIAL KERNEL : g is a polynomial

Survey on edge modification

A survey of parameterized algorithms and the complexity of edge modification

Christophe Crespelle, Pål Grønås Drange, Fedor V. Fomin, Petr A. Golovach

graph class	completion		deletion		editing	
	KERNEL	TIME SUBEPT	KERNEL	TIME SUBEPT	KERNEL	TIME SUBEPT
line	OPEN	FPT by [?] OPEN	OPEN	FPT by [?] OPEN	OPEN	FPT by [?] OPEN
s-Plex Cluster	—	— —	—	— —	s^2k [?]	$(2s+\sqrt{s})^k$ [?] NOSUB [?]
$\{K_3, 2K_2, C_5\}$ chain	as deletion		k^2 [?, ?]	SUBEPT $2^{\sqrt{k \log k}}$ [?]	k^2 [?]	SUBEPT $2^{\sqrt{k \log k}}$ [?]
$\{K_3, C_4, P_4\}$ Starforest	P		$4k$ [?]	FPT by [?] NOSUB [?]	as deletion	
$\{2K_2, C_4, P_4\}$ threshold *	k^2 [?]	SUBEPT $2^{\sqrt{k \log k}}$ [?] NO $2^{k^{1/4}}$ [?]	k^2 [?]	SUBEPT $2^{\sqrt{k \log k}}$ [?] NO $2^{k^{1/4}}$ [?]	k^2 [?]	SUBEPT $2^{\sqrt{k \log k}}$ [?]
$\{2K_2, C_4, C_5\}$ split *	k [?], $5k^{1.5}$ [?]	SUBEPT $2^{\mathcal{O}(\sqrt{k})}$ [?, Exercise 5.17]	k [?], $5k^{1.5}$ [?]	SUBEPT $2^{\mathcal{O}(\sqrt{k})}$ [?, Exercise 5.17]	P [?]	
$\{P_3, 2K_2\}$ clique + isol. vert.	P		$k/\log k$ [?]	SUBEPT $1.6355^{\sqrt{k \ln k}}$ [?]	$2k$ [folkl.]	SUBEPT $2^{\sqrt{k \ln k}}$ [?]
$\{C_4, P_4\}$ trivially perfect	k^2 [?, ?]	SUBEPT $2^{\sqrt{k \log k}}$ [?] NO $2^{k^{1/4}}$ [?]	k^3 [?]	2.42^k [?] NOSUB [?]	k^3 [?]	— NOSUB [?]
$\{claw, diamond\}$	OPEN	FPT by [?] OPEN	$k^{\mathcal{O}(1)}$ [?]	OPEN NOSUB [?]	OPEN	FPT by [?] —
$\{2K_2, C_4\}$ pseudosplit *	$5k^{1.5}$ [?]	SUBEPT $2^{\mathcal{O}(\sqrt{k})}$ [?, ?]	$5k^{1.5}$ [?]	SUBEPT $2^{\mathcal{O}(\sqrt{k})}$ [?, ?]	P [?, ?]	
$\{P_3\}$ cluster	P		$2k$ [?]	1.41^k [?] NOSUB [?]	$2k$ [?, ?]	1.76^k [?] NOSUB [?]
$\{K_3\}$	P		$6k$ [?]	FPT by [?] NOSUB [?]	as deletion	
$\{P_4\}$ cograph *	k^3 [?]	2.56^k [?] NOSUB [?, ?]	k^3 [?]	2.56^k [?] NOSUB [?, ?]	k^3 [?]	4.61^k [?] NOSUB [?]
$\{paw\}$	k^3 [?]	FPT by [?] NOSUB [?]	k^3 [?]	FPT by [?] NOSUB [?]	k^6 [?]	FPT by [?] NOSUB [?]
$\{diamond\}$	P		k^3 [?, ?]	FPT by [?] NOSUB [?, ?]	k^8 [?]	FPT by [?] NOSUB [?]
$\{claw\}$	OPEN	FPT by [?] NOSUB [?]	OPEN	FPT by [?] NOSUB [?]	OPEN	FPT by [?] NOSUB [?]
$\{K_4\}$	P		k^3 [?]	FPT by [?] NOSUB [?]	as deletion	
$\{P_\ell\}$ fixed $\ell > 4$	NOKER [?]	FPT by [?] NOSUB [?]	NOKER [?]	FPT by [?] NOSUB [?]	NOKER [?]	FPT by [?] NOSUB [?]
$\{C_\ell\}$ fixed $\ell > 3$	NOKER [?]	FPT by [?] NOSUB [?]	NOKER [?]	FPT by [?] NOSUB [?]	NOKER [?]	FPT by [?] NOSUB [?]

Survey on edge modification

A survey of parameterized algorithms and the complexity of edge modification
Christophe Crespelle, Pål Grønås Drange, Fedor V. Fomin, Petr A. Golovach

graph class	completion		deletion		editing	
	KERNEL	FPT SUBEPT	KERNEL	FPT SUBEPT	KERNEL	FPT SUBEPT
Linear forest	P		$9k$ [?]	2.29^k [?] randomized NOSUB (Hamiltonicity)	as deletion	
Distance-hereditary	OPEN	FPT (from [?]) -	OPEN	FPT (from [?]) NOSUB [?, ?]	OPEN	FPT (from [?]) NOSUB [?, ?]
Planar	P		OPEN	FPT [?] (minor closed [?]) OPEN	as deletion	
H-minor-free	P		OPEN	FPT minor closed [?] OPEN	as deletion	
Bipartite	P		k^3 [?][†] randomized	2^k [?] 1.977^k [?] NOSUB (folk.)	as deletion	
3-leaf power	k^3 [?]	FPT [?] OPEN	k^3 [?]	FPT [?] NOSUB (Clustering)	k^3 [?]	FPT [?] NOSUB (Clustering)
4-leaf power	OPEN	FPT [?, ?] -	OPEN	FPT [?, ?] -	OPEN	FPT [?, ?] -
proper interval	k^3 [?]	SUBEPT $2^{O(k^{2/3})\log k}$ [?] NO $2^{k^{1/4}}$ [?]	OPEN	FPT [?] OPEN	OPEN	FPT [?] OPEN
interval	OPEN	SUBEPT $2^{\sqrt{k}\log k}$ [?] NO $2^{k^{1/4}}$ [?]	OPEN	$2^{O(k)\log k}$ [?] OPEN	OPEN	OPEN OPEN
strongly chordal	OPEN	64^k [?] OPEN	OPEN	OPEN OPEN	OPEN	OPEN OPEN
chordal	k^2 [?]	SUBEPT $2^{\sqrt{k}\log k}$ [?] NO $2^{\sqrt{k}}$ [?]	OPEN	$2^{O(k\log k)}$ [?] OPEN	OPEN	$2^{O(k\log k)}$ [?] OPEN

Polynomial kernel algorithms

- A set of reduction rules: (I,k) → (I',k')

  Rule 1: **if** *condition 1* **then** *transformation 1*
  Rule 2: **if** *condition 2* **then** *transformation 2*
  ...

- All rules are:

  - Sound : (I',k') is a YES-instance *iff* (I,k) is a YES-instance
  - Computable in polynomial time, wrt. |I|
  - number of times the rules are applied is polynomial.

- A YES-instance (I,k) reduced under these rules always satisfies:

  |I| ≤ P(k) (with P a polynomial)

Remarks:

- Reduced = no rule applies
- If after reduction |I| > P(k) then output a constant-size NO-instance

Kernels for edge modification

Two kinds of rules

- For **forced modifications (that must be made)**

- For **removing irrelevant parts of the input graph**
  - That do not need to be modified ***and***
  - That do not influence modifications in the rest of the graph

k^3

$O(n^2)$-vertex kernel for cograph editing

Guillemot, Havet, Paul and Perez, 2010

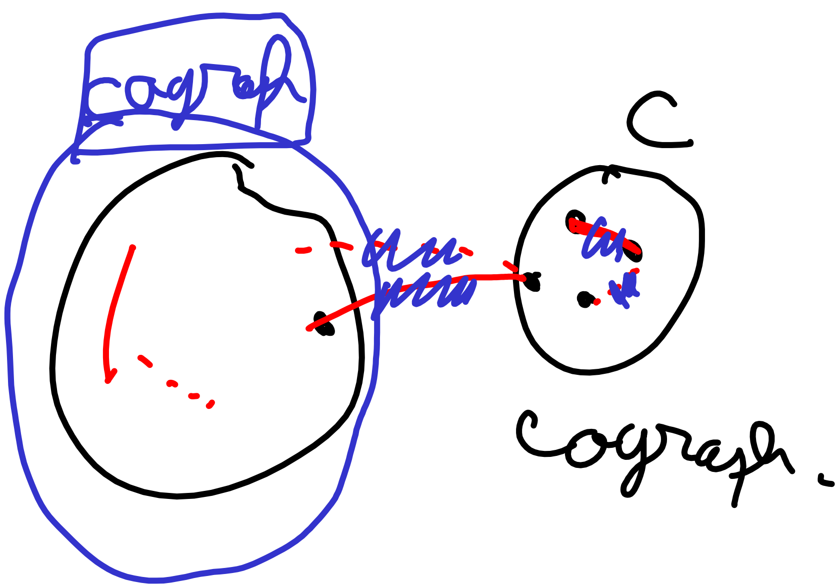
O(k³) vertex kernel for cograph editing

On the (Non-)Existence of Polynomial Kernels for P_l-Free Edge Modification Problems. Guillemot, Havet, Paul & Perez, 2010.

Rules for removing the irrelevant parts :

- Rule 1 (cograph component):

Remove the connected components of G that are cographs.



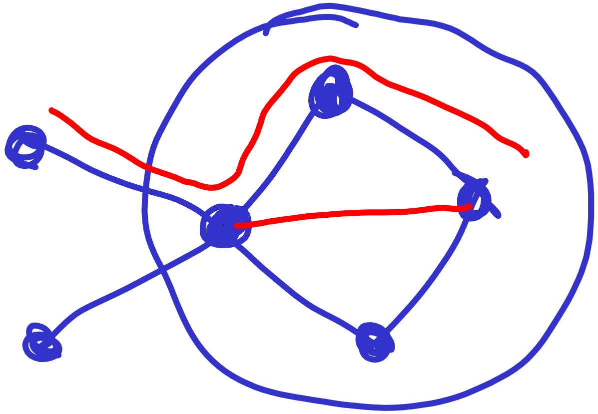
$O(k^3)$ vertex kernel for cograph editing

On the (Non-)Existence of Polynomial Kernels for P_l-Free Edge Modification Problems. Guillemot, Havet, Paul & Perez, 2010.

<u>Rules for removing the irrelevant parts :</u>

- <u>Rule 1</u> (cograph component):

Remove the connected components of G that are cographs.



! It works because it is a connected component

$O(k^3)$ vertex kernel for cograph editing

On the (Non-)Existence of Polynomial Kernels for P_l-Free Edge Modification Problems. Guillemot, Havet, Paul & Perez, 2010.

Rules for removing the irrelevant parts :

- Rule 1 (cograph component):

Remove the connected components of G that are cographs.

- Rule 2 (modules):

If **M is a non-trivial module** of G which is strictly contained in a connected component and **is not an independent set of size at most k + 1**, then return the graph G’ ⊕ G[M] where G’ is obtained from G by replacing M by an independent set module of size min{|M|, k+1}.

O(k³) vertex kernel for cograph editing

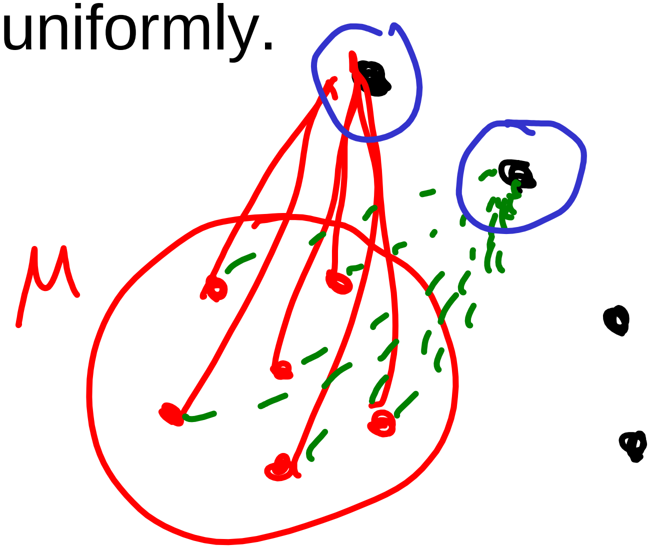
- <u>Rule 2 (modules):</u>

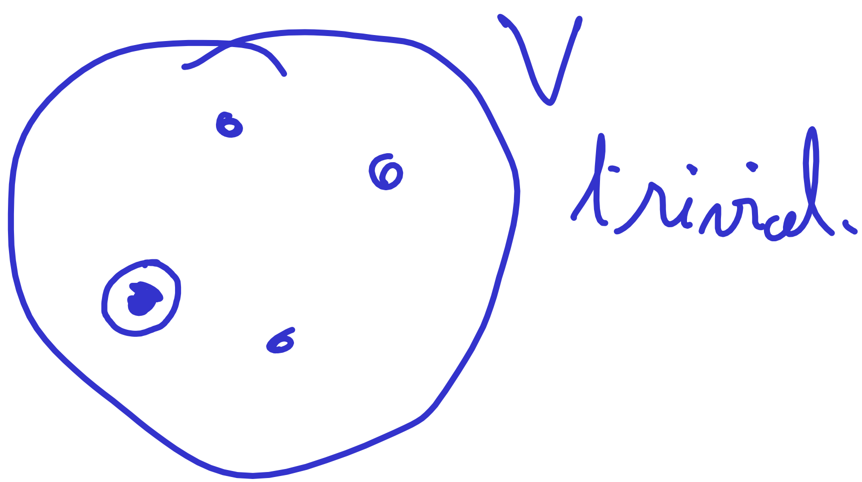
If **M is a non-trivial module** of G which is strictly contained in a connected component and **is not an independent set of size at most k + 1**, then return the graph G' ⊕ G[M] where G' is obtained from G by replacing M by an independent set module of size min{|M|, k+1}.

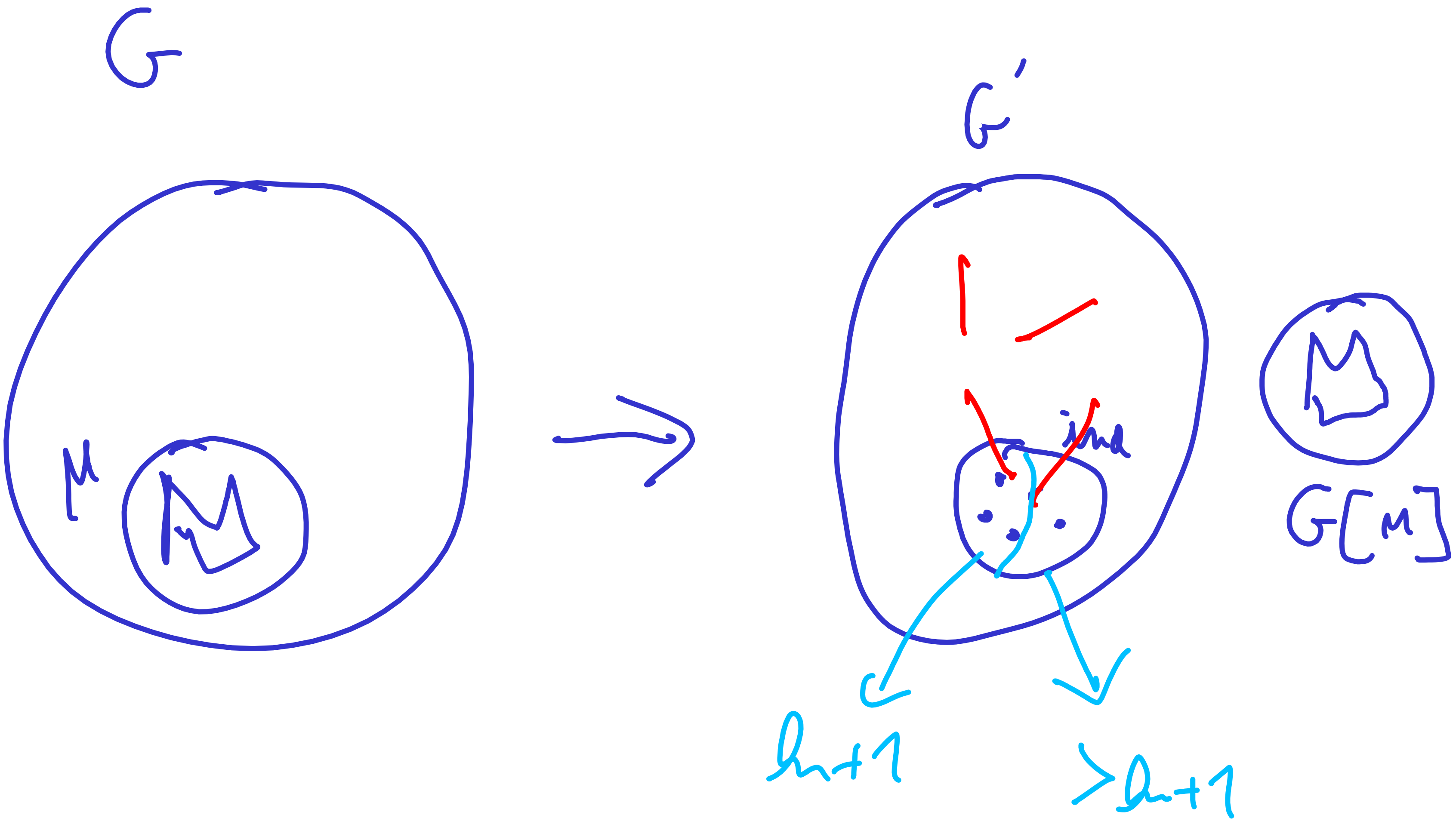
<u>Definition (module)</u>

M is a module if all the vertices of M have the same neighbours outside of M.

Or equivalently, M is a module if each vertex outside of M sees M uniformly.



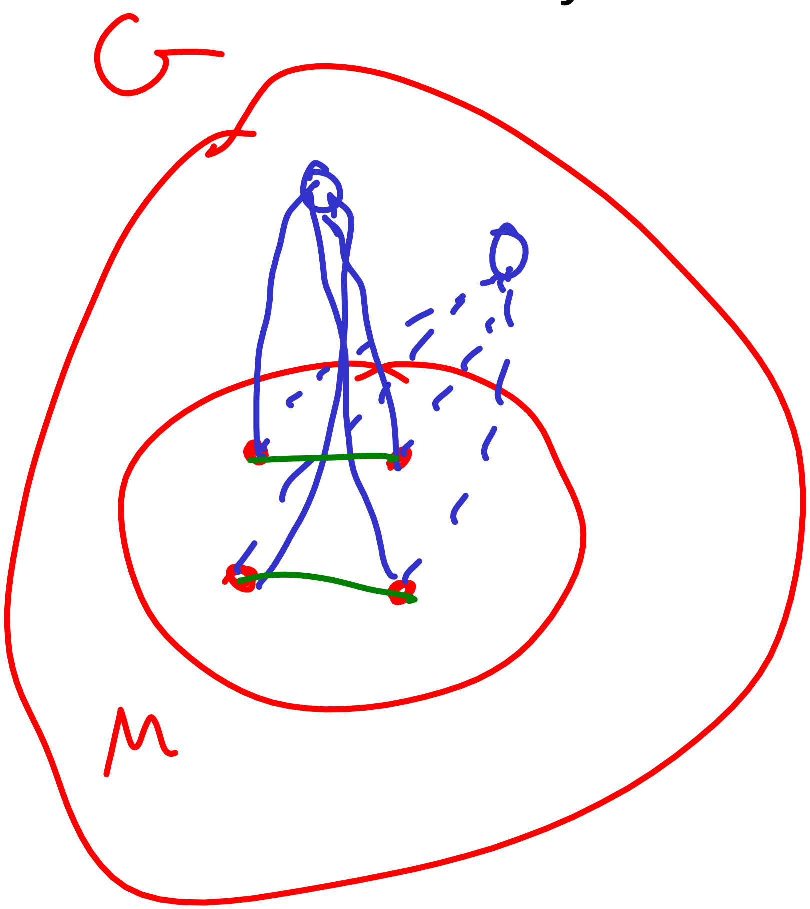


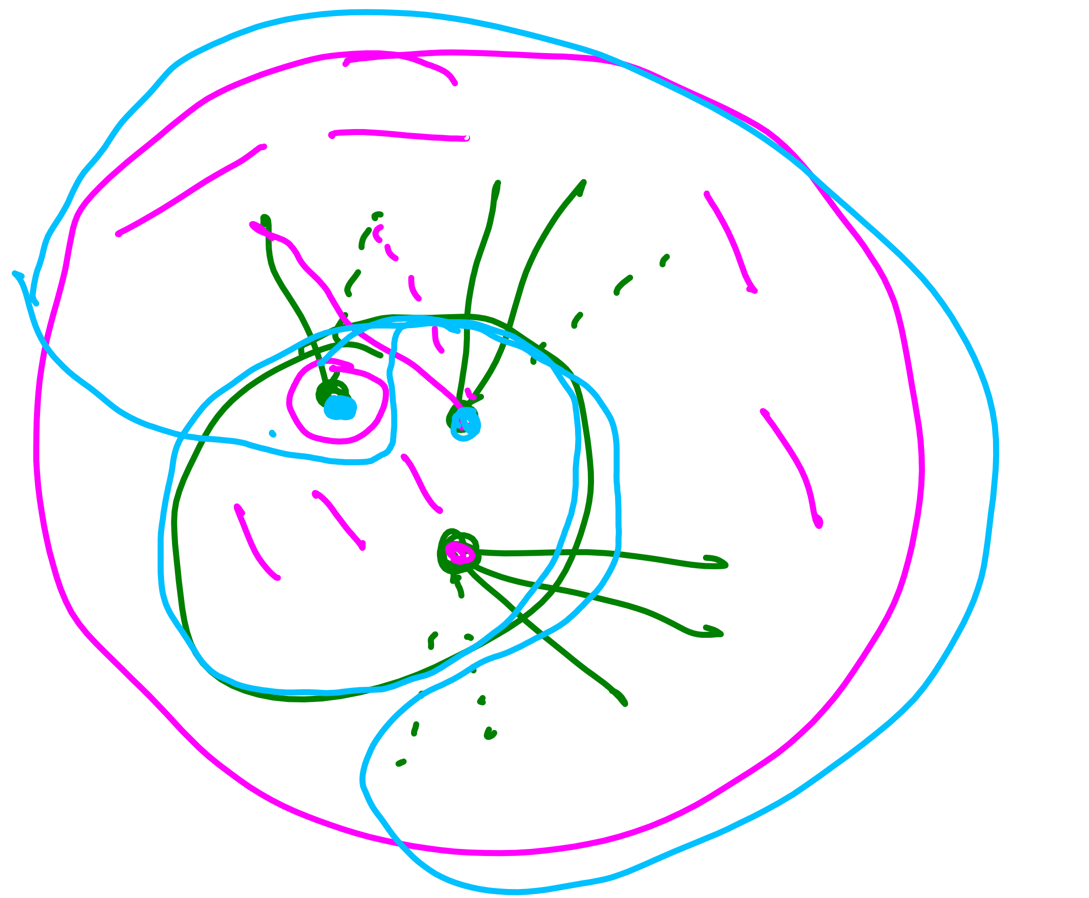

G
M
M
G'
ind
M
G[M]
$h+1$
$>h+1$

O(k³) vertex kernel for cograph editing

Exercise

Prove that if M is a module of G, there exists a minimum editing of G that edit the adjacencies between any vertex x∈M and vertices of V \ M in the same way for all x∈M.

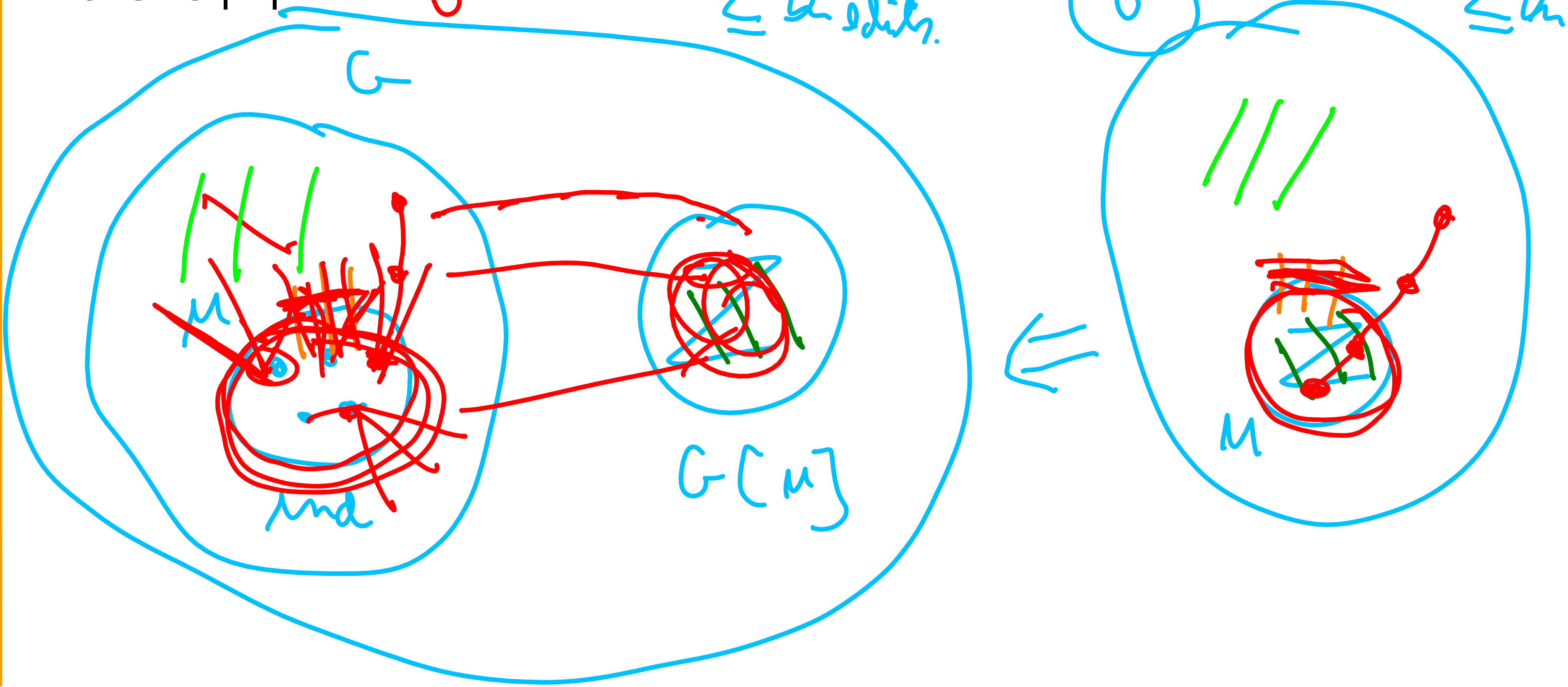




O(k^3) vertex kernel for cograph editing

Exercise

Prove that if M is a module of G, then G’’ ⊕ G[M] admits a cograph editing of size at most k iff G admits an editing of size at most k, where G’’ is obtained from G by replacing M by an independent set module of size |M|.



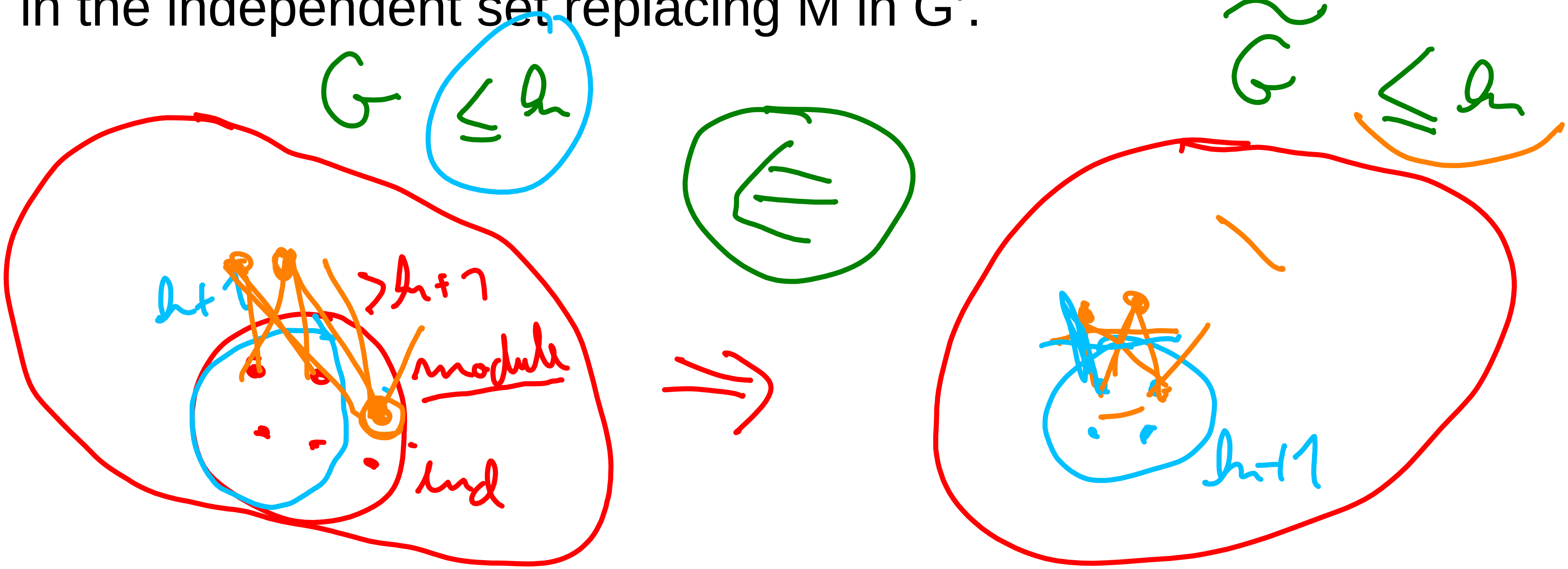
O(k³) vertex kernel for cograph editing

- Rule 2 (modules):

If **M is a non-trivial module** of G which is strictly contained in a connected component and **is not an independent set of size at most k + 1**, then return the graph G' ⊕ G[M] where G' is obtained from G by replacing M by an independent set module of size min{|M|, k+1}.

Soundness

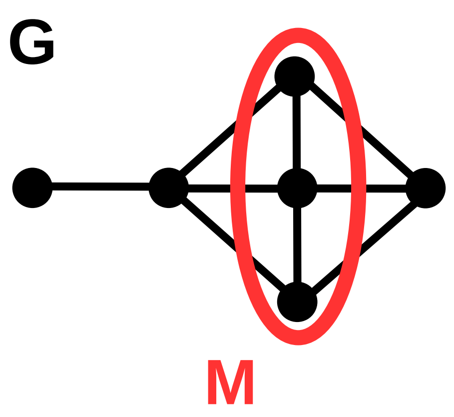
We only need to prove that if G admits a cograph editing of size k and if M has size more than k+1, then we can keep only k+1 vertices in the independent set replacing M in G'.



O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

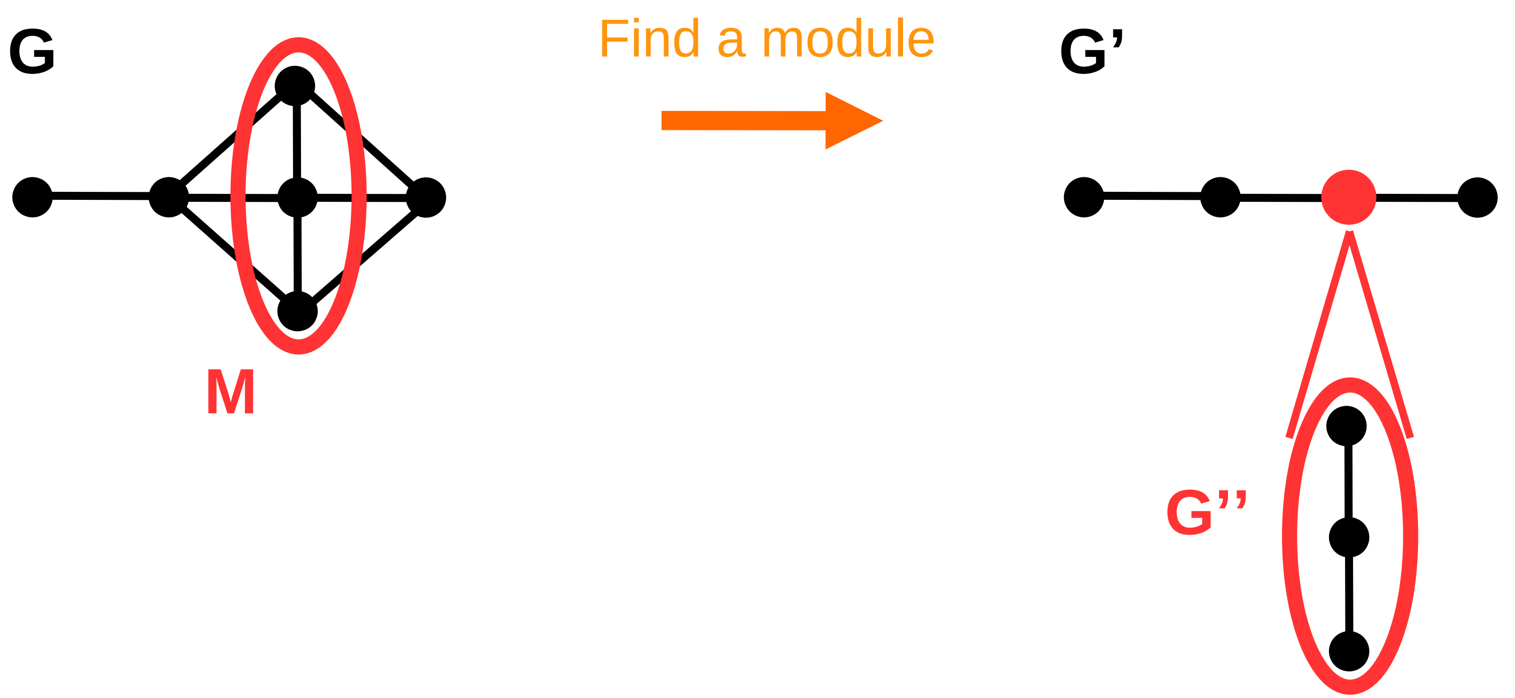
Modular decomposition tree



O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

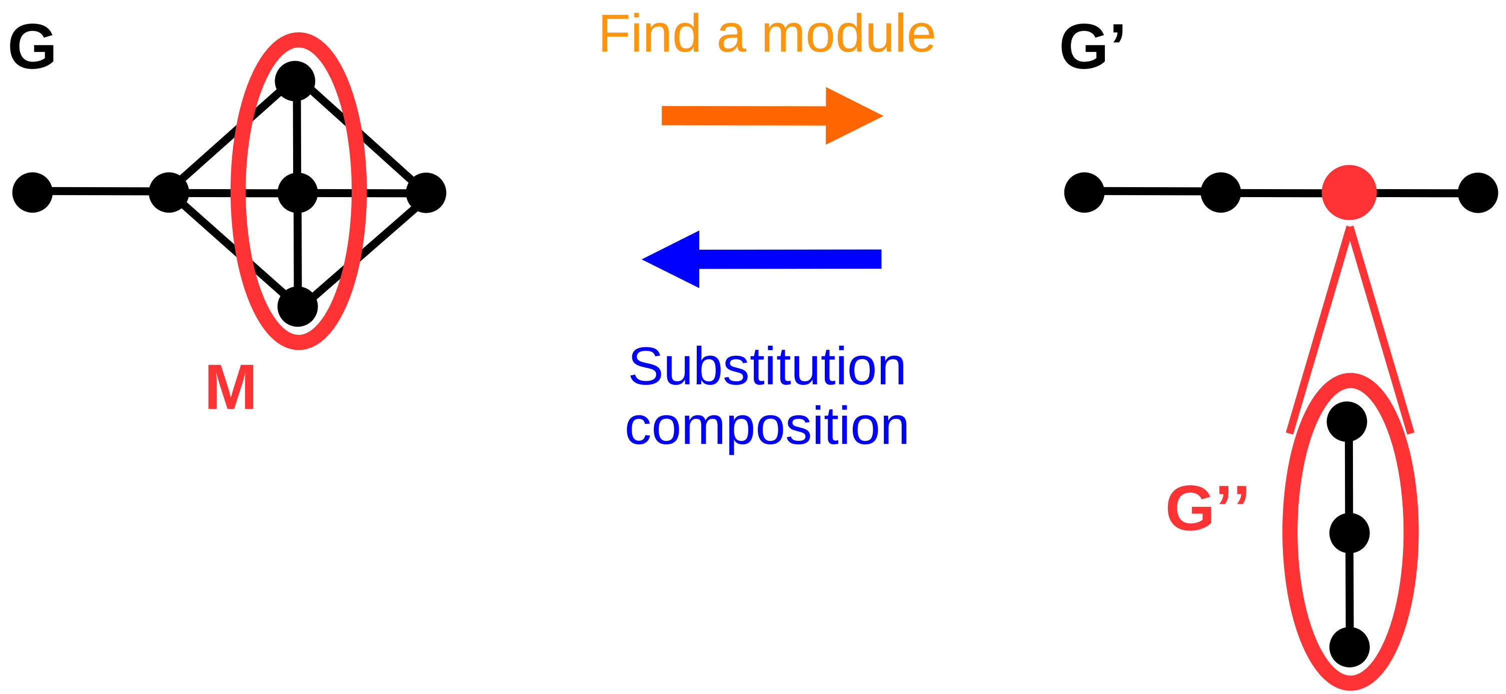
Modular decomposition tree



$O(k^3)$ vertex kernel for cograph editing

- Rules 1 and 2 work together

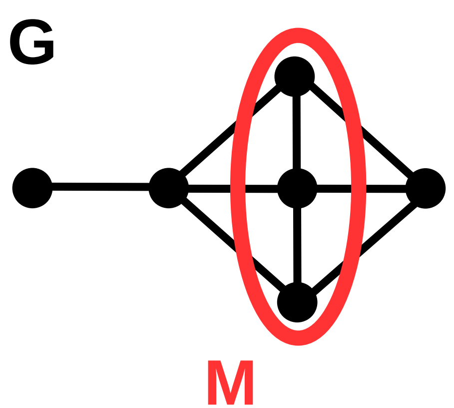
Modular decomposition tree

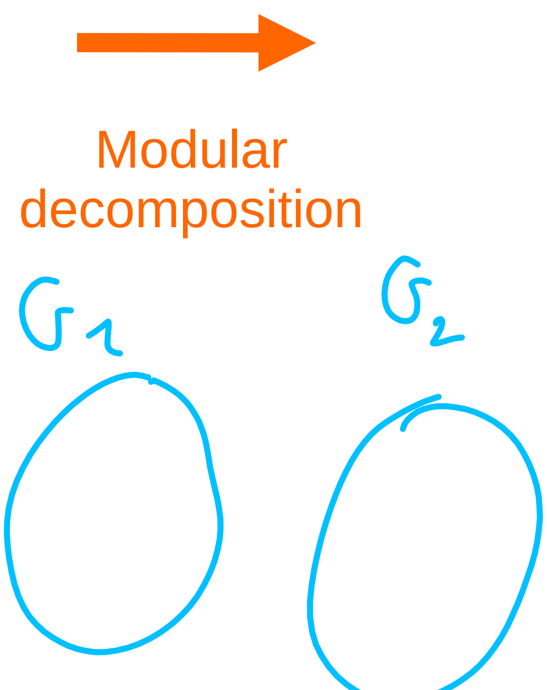


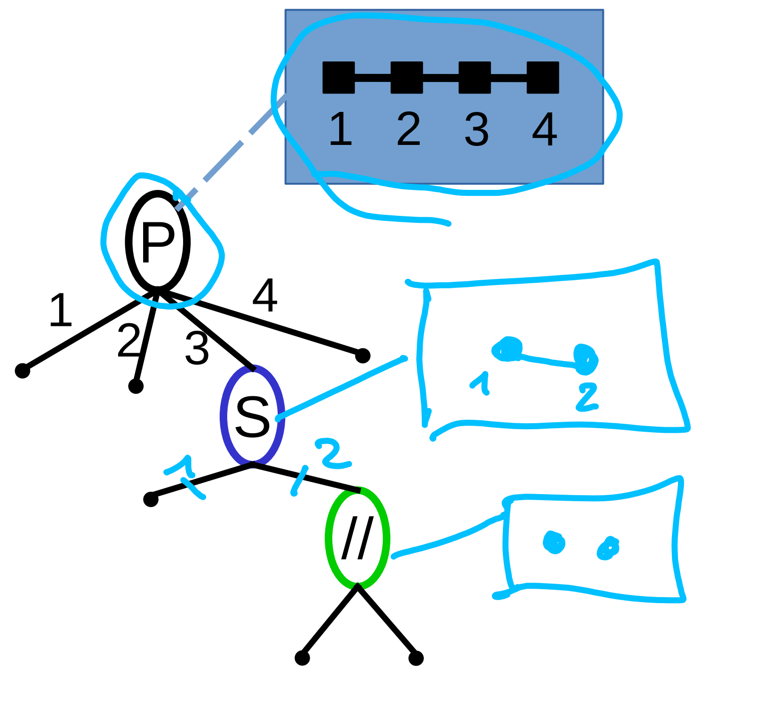
O(k³) vertex kernel for cograph editing

- <u>Rules 1 and 2 work together</u>

<u>Modular decomposition tree</u>

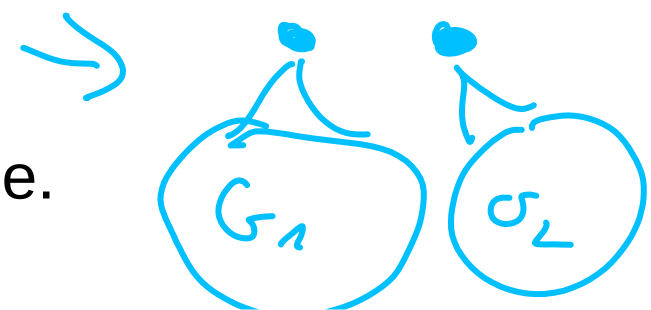






P stands for prime.

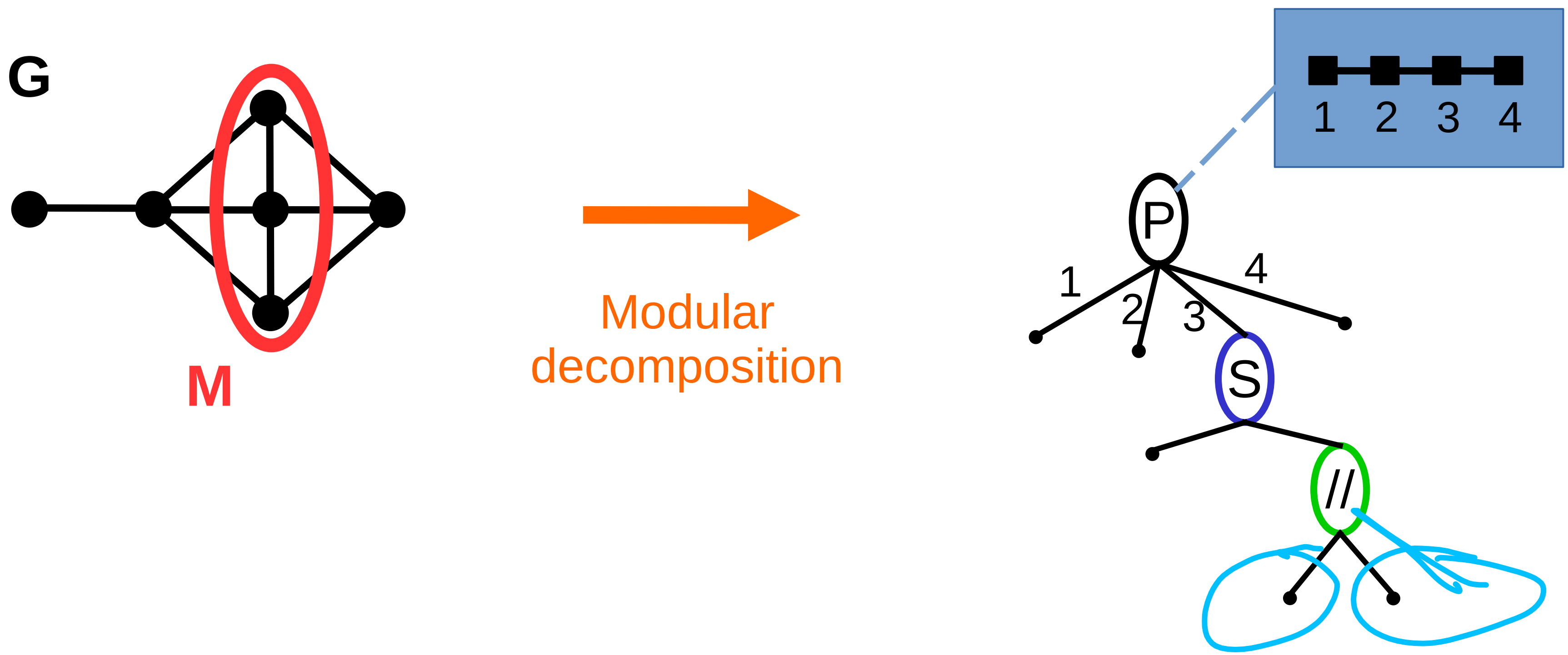
<u>Definition :</u>
A graph is ***prime*** iff it has no non-trivial module.



O(k³) vertex kernel for cograph editing

- <u>Rules 1 and 2 work together</u>

<u>Modular decomposition tree</u>



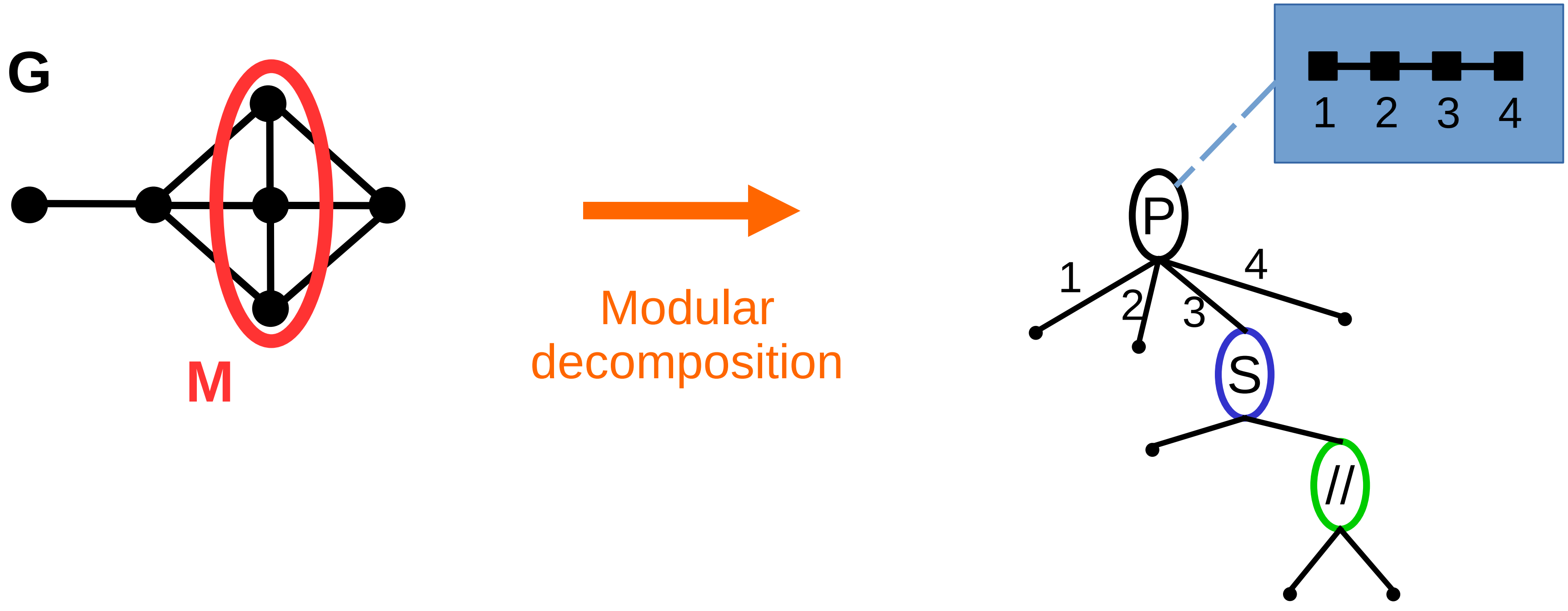
Can be computed in O(n+m) time

→ **Modular decomposition tree**

O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

Modular decomposition tree

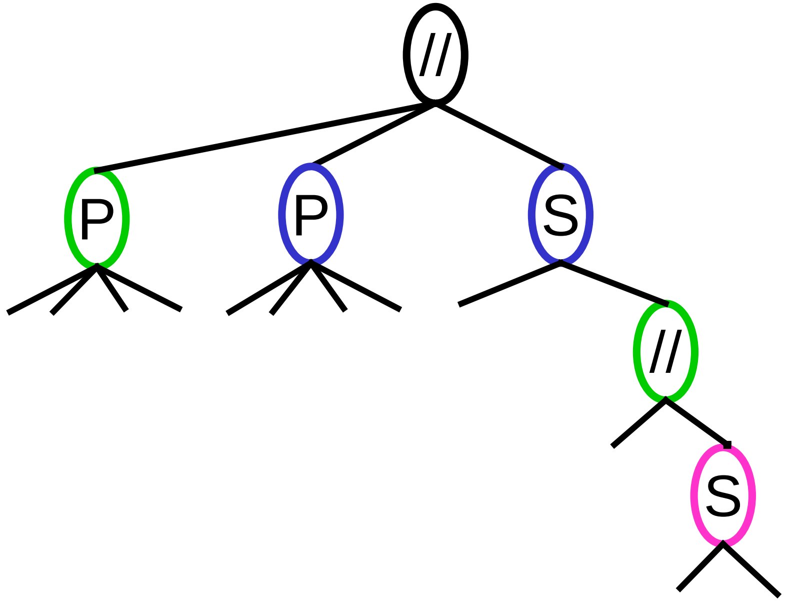


Theorem :
A graph is a cograph iff it has no P node in its modular decomposition tree.

O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

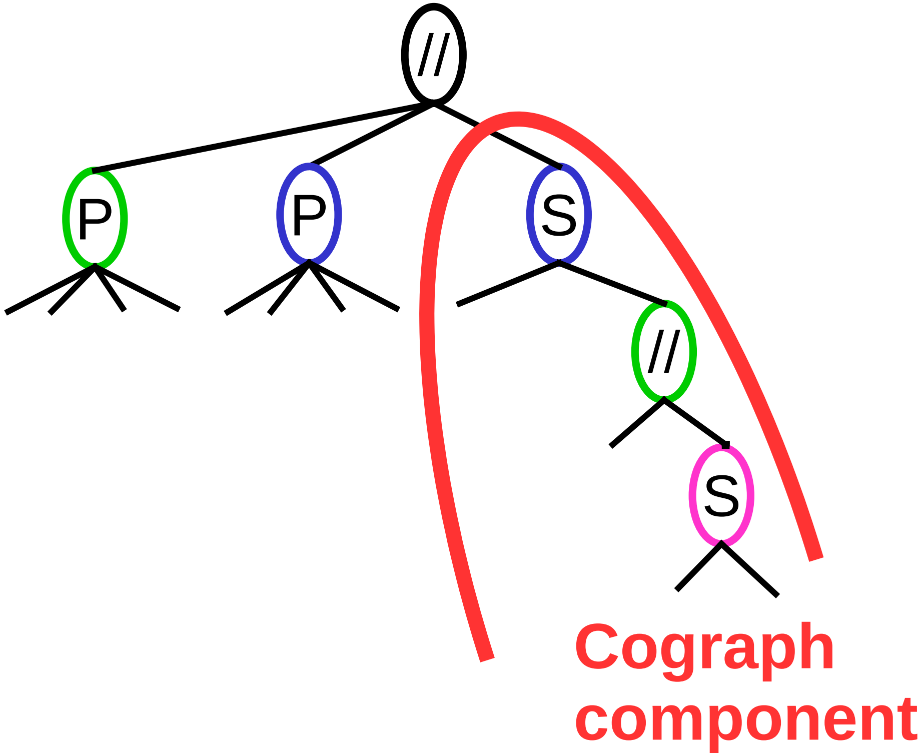
With rule 1 only :



O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

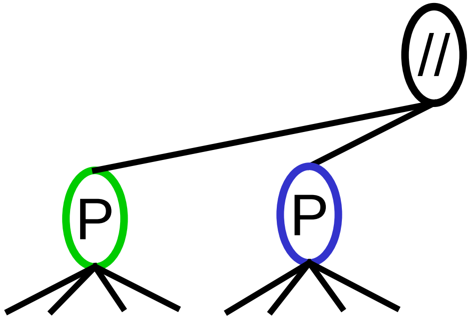
With rule 1 only :



O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

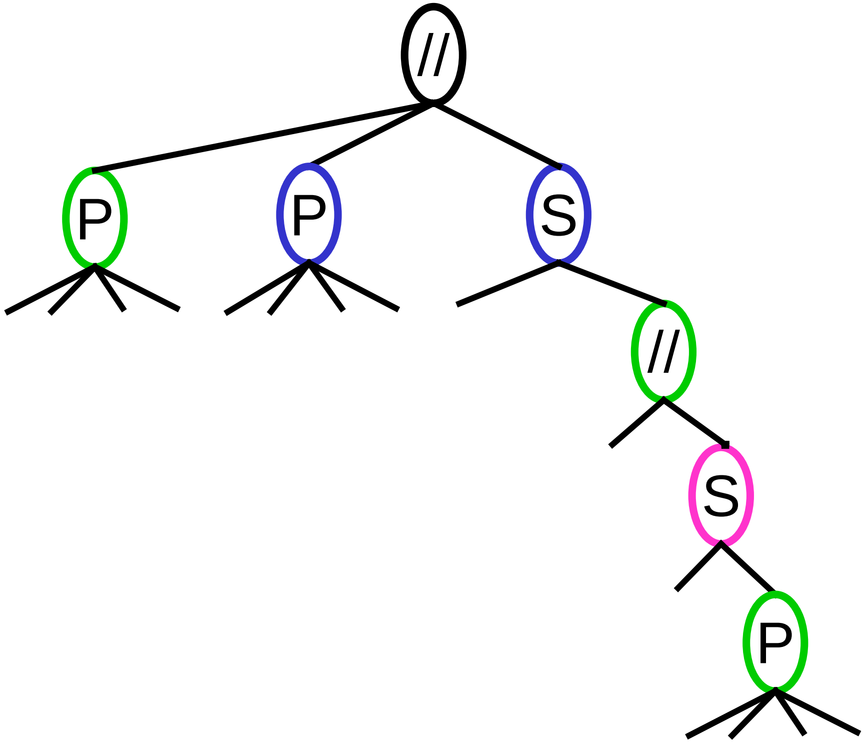
With rule 1 only :



O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

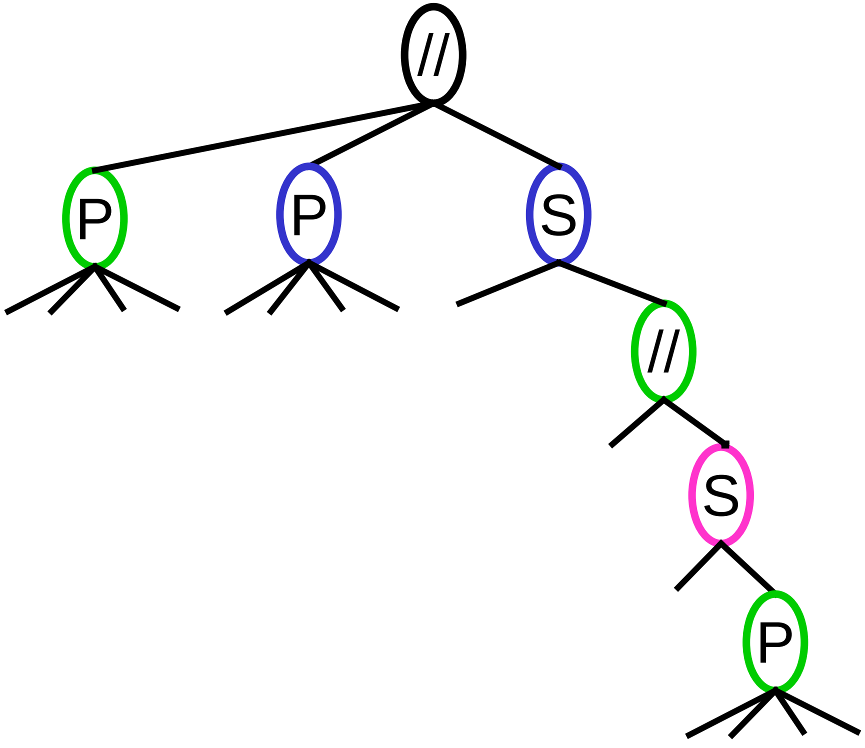
With rule 1 only :



O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

With rule 1 only : **cannot cut anything...**

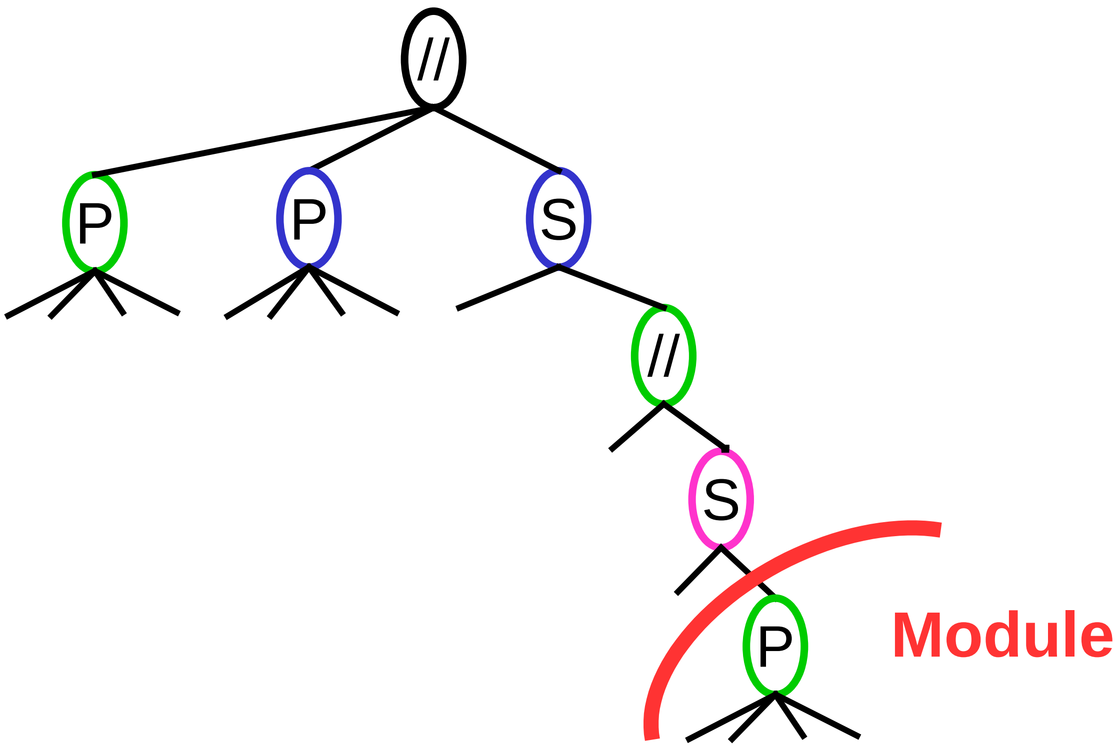


O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

With rule 1 and 2 :

Rule 2 first

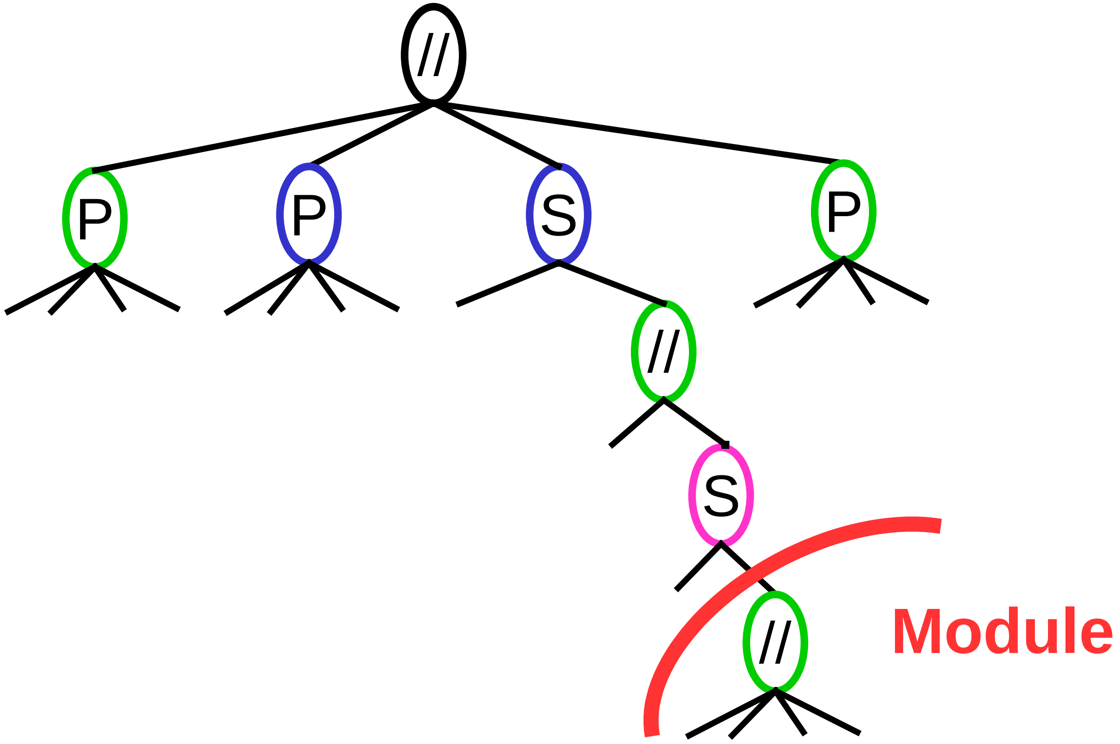


O(k³) vertex kernel for cograph editing

- Rules 1 and 2 work together

With rule 1 and 2 :

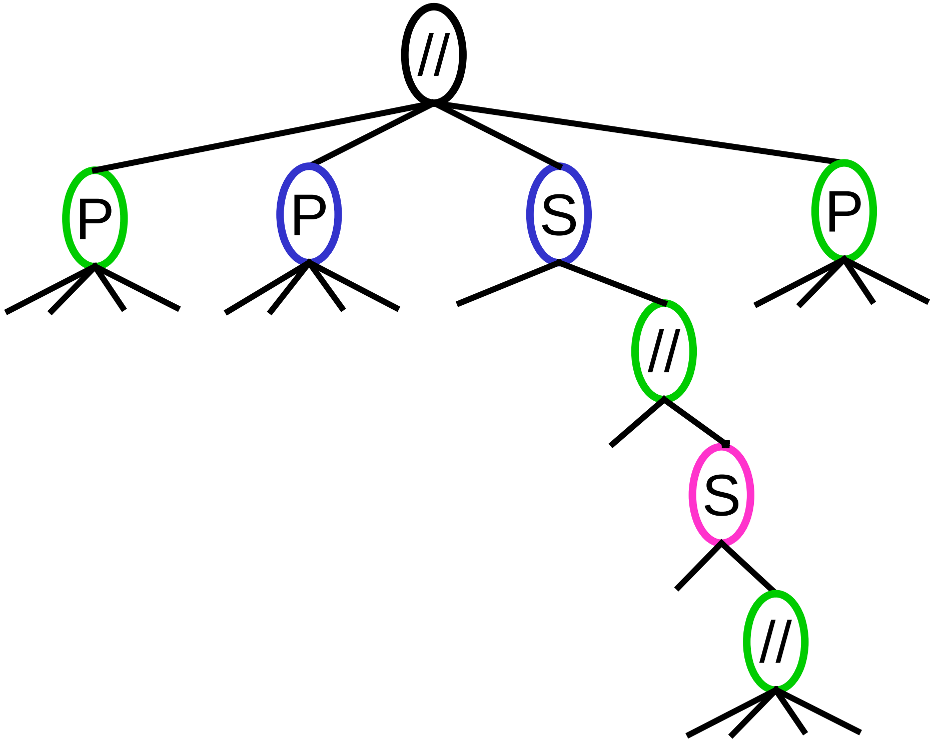
Rule 2 first



O(k^3) vertex kernel for cograph editing

- Rules 1 and 2 work together

With rule 1 and 2 : Rule 2 first



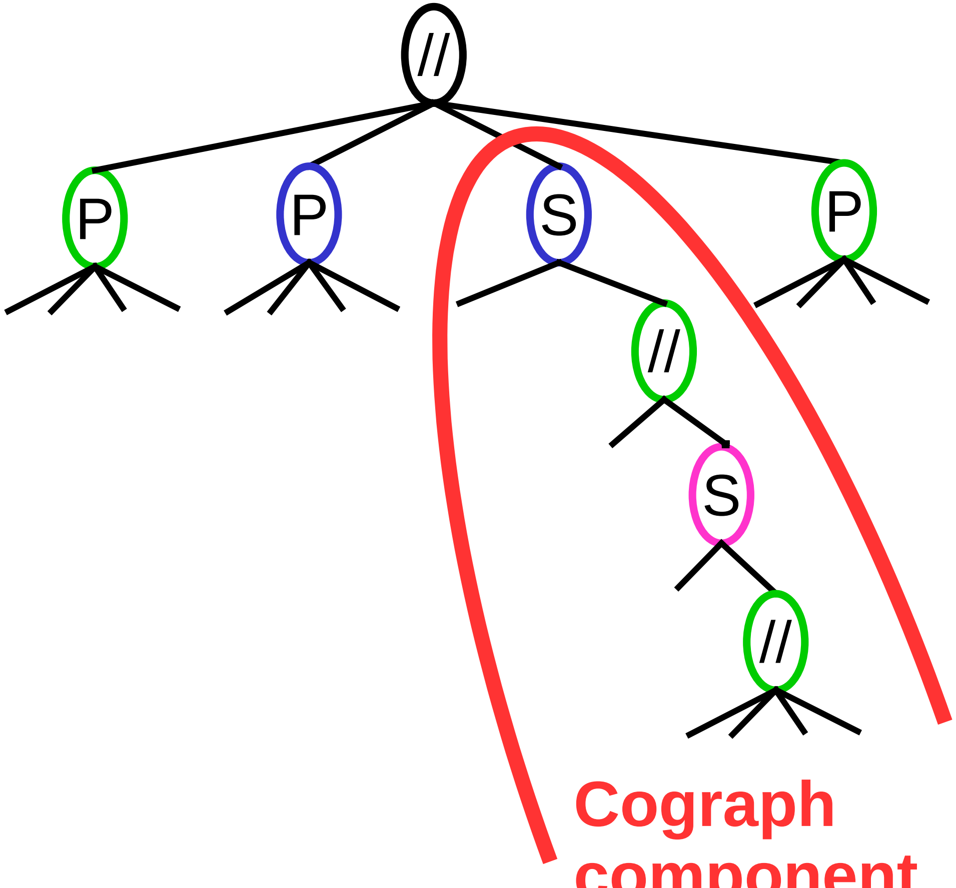
O(k^3) vertex kernel for cograph editing

- Rules 1 and 2 work together

With rule 1 and 2 :

Rule 2 first

Then Rule 1



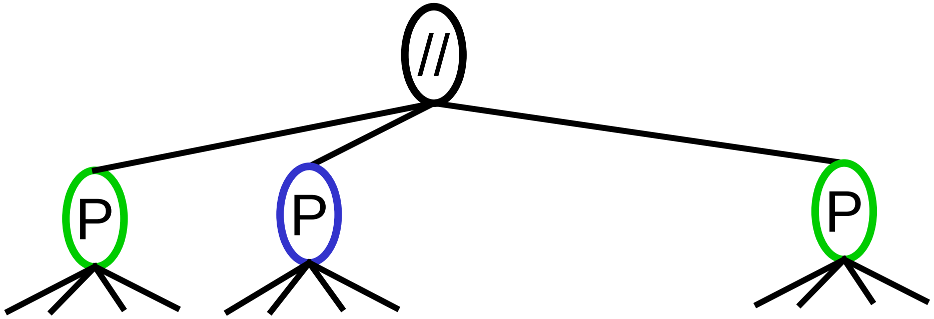
O(k^3) vertex kernel for cograph editing

- Rules 1 and 2 work together

With rule 1 and 2 :

Rule 2 first

Then Rule 1



$O(k^3)$ vertex kernel for cograph editing

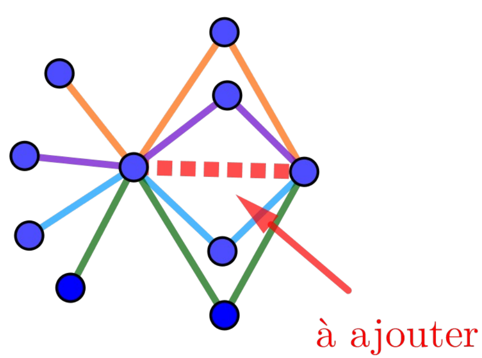
On the (Non-)Existence of Polynomial Kernels for P_l-Free Edge Modification Problems. Guillemot, Havet, Paul & Perez, 2010.

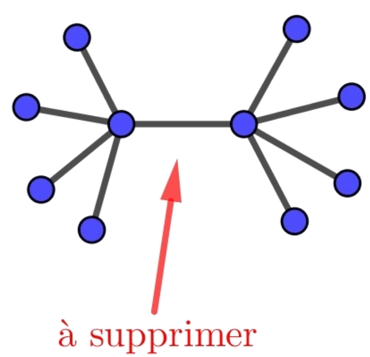
Rules for forced modifications :

- Rule 3 (P_4 sunflower):

If {x, y} is a pair of vertices of G that belongs to a set S of $t \geqslant k + 1$ quadruples $P_i = \{x, y, a_i, b_i\}$ such that for $1 \leq i \leq t$, every P_i induces a P_4 and for any $1 \leq i < j \leq t$, $P_i \cap P_j = \{x, y\}$, then edit {x,y} and decrease k by one.

$O(n^4)$ time





Proof of the size of the kernel : $O(k^3)$

Theorem (size of the kernel) :

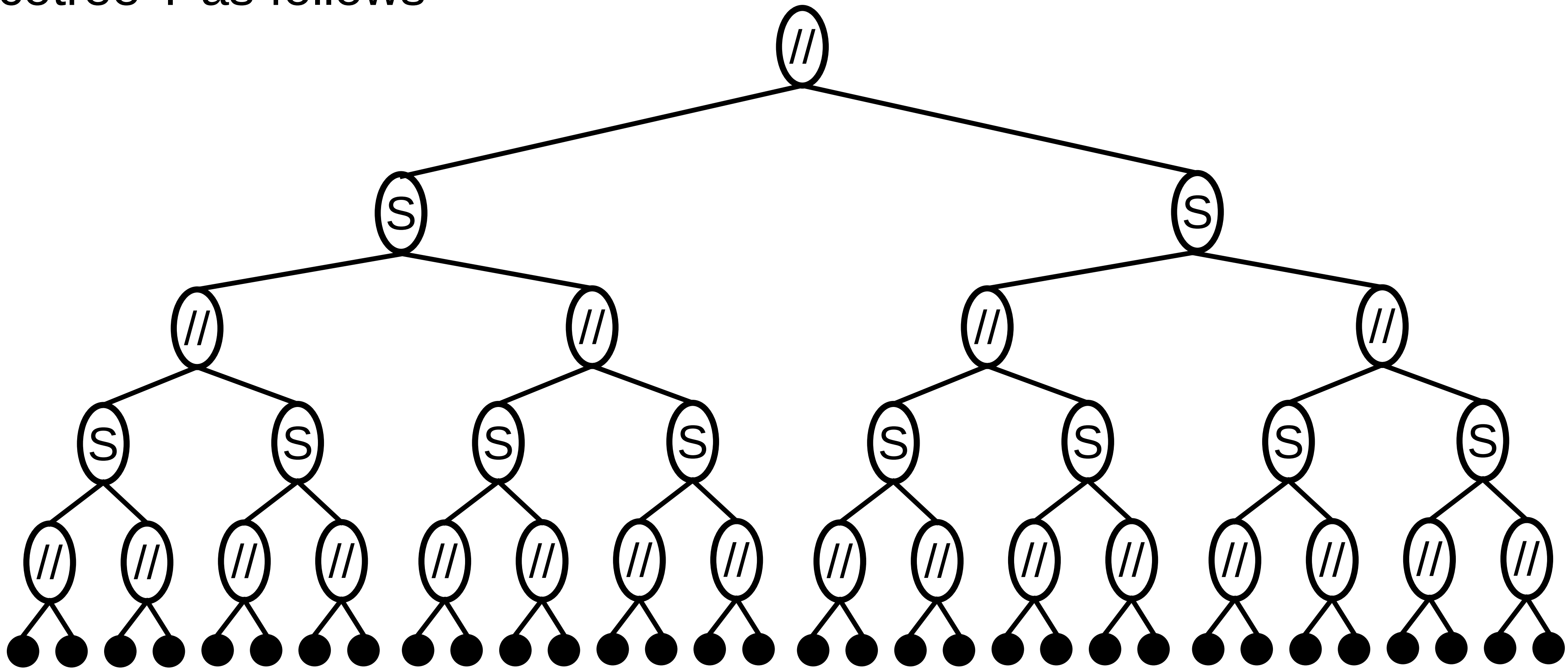
Let G be a graph ***reduced under rules 1, 2 and 3***. If G admits a cograph editing of size k, then G has $O(k^3)$ vertices.

Proof of the size of the kernel : $O(k^3)$

Theorem (size of the kernel) :

Let G be a graph ***reduced under rules 1, 2 and 3***. If G admits a cograph editing of size k, then G has $O(k^3)$ vertices.

Proof : consider a minimum modification of G into a cograph having cotree T as follows



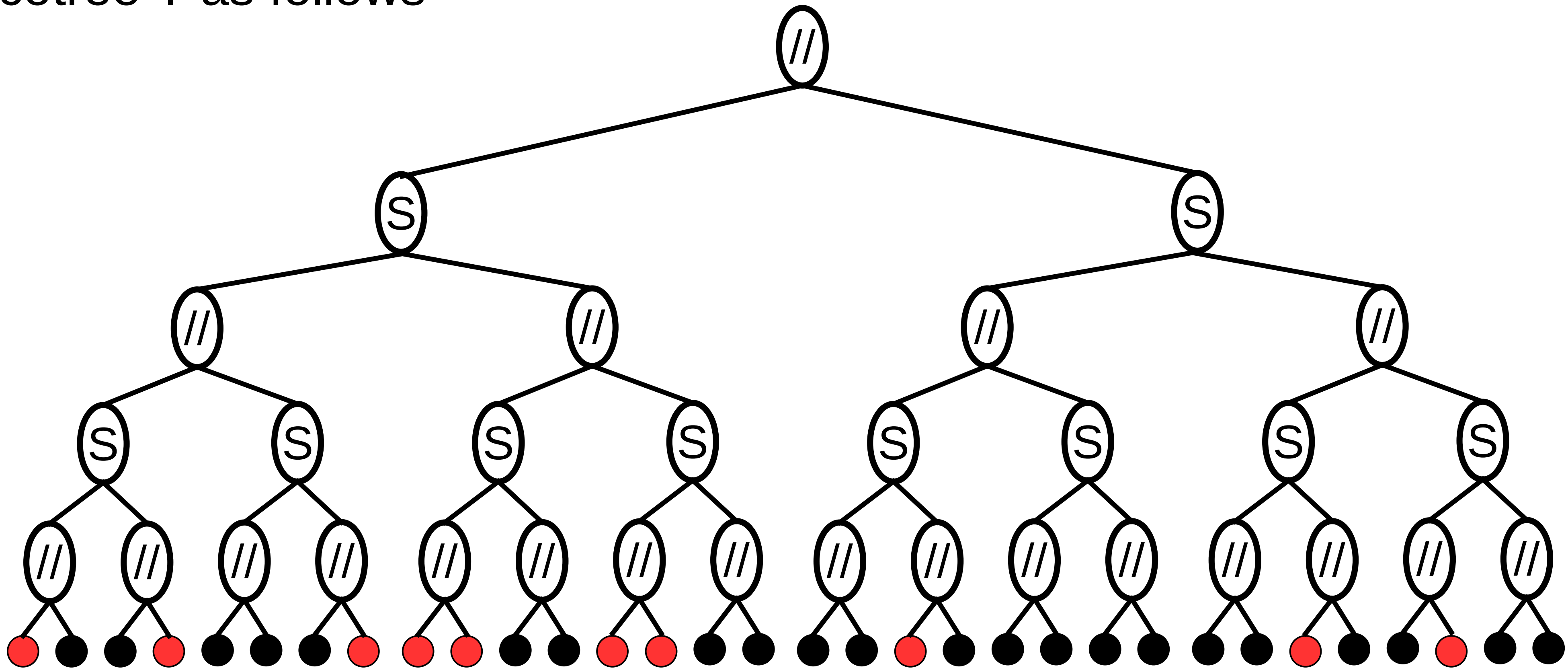
Affected vertices ≤ 2k Affected internal nodes ≤ 2k

Proof of the size of the kernel : $O(k^3)$

Theorem (size of the kernel) :

Let G be a graph ***reduced under rules 1, 2 and 3***. If G admits a cograph editing of size k, then G has $O(k^3)$ vertices.

Proof : consider a minimum modification of G into a cograph having cotree T as follows



Affected vertices ≤ 2k

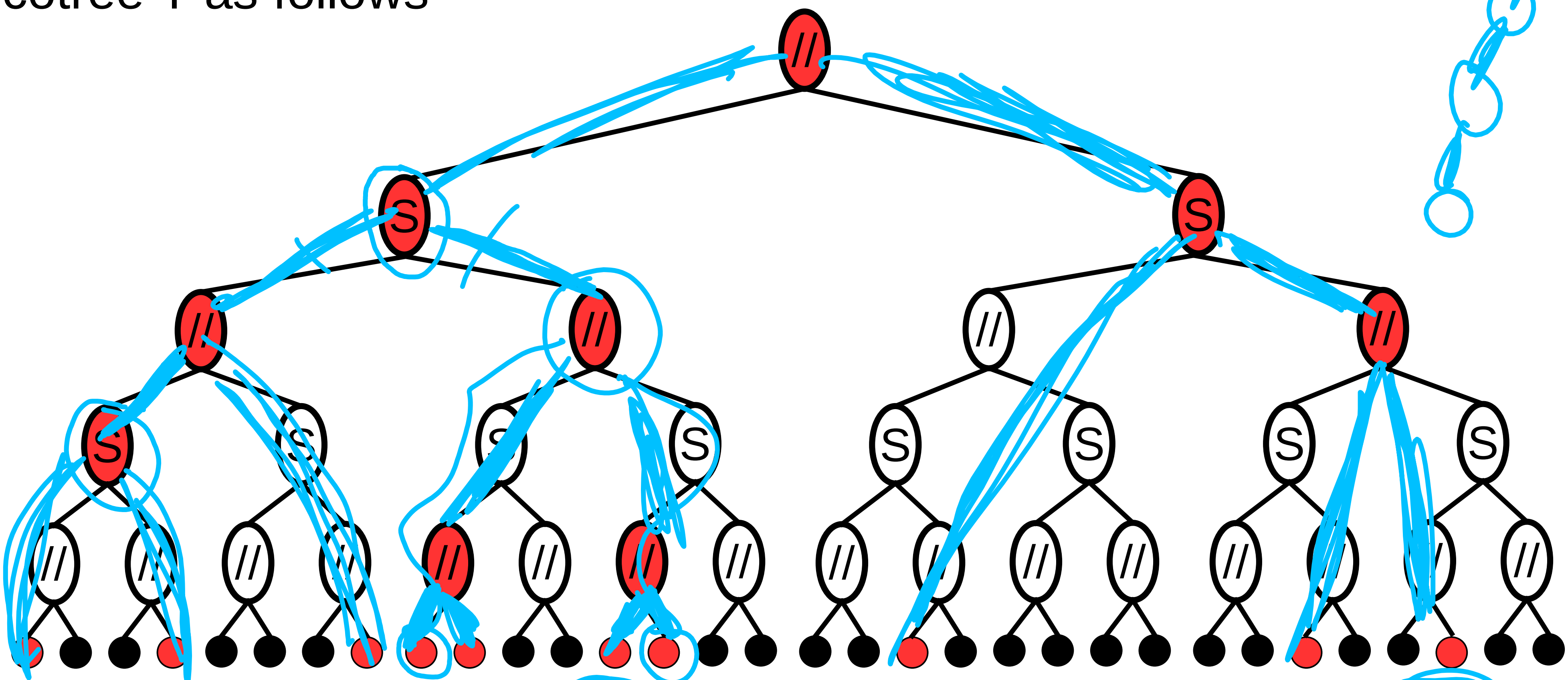
Affected internal nodes ≤ 2k

Proof of the size of the kernel : O(k^3)

Theorem (size of the kernel) :

Let G be a graph ***reduced under rules 1, 2 and 3***. If G admits a cograph editing of size k, then G has O(k^3) vertices.

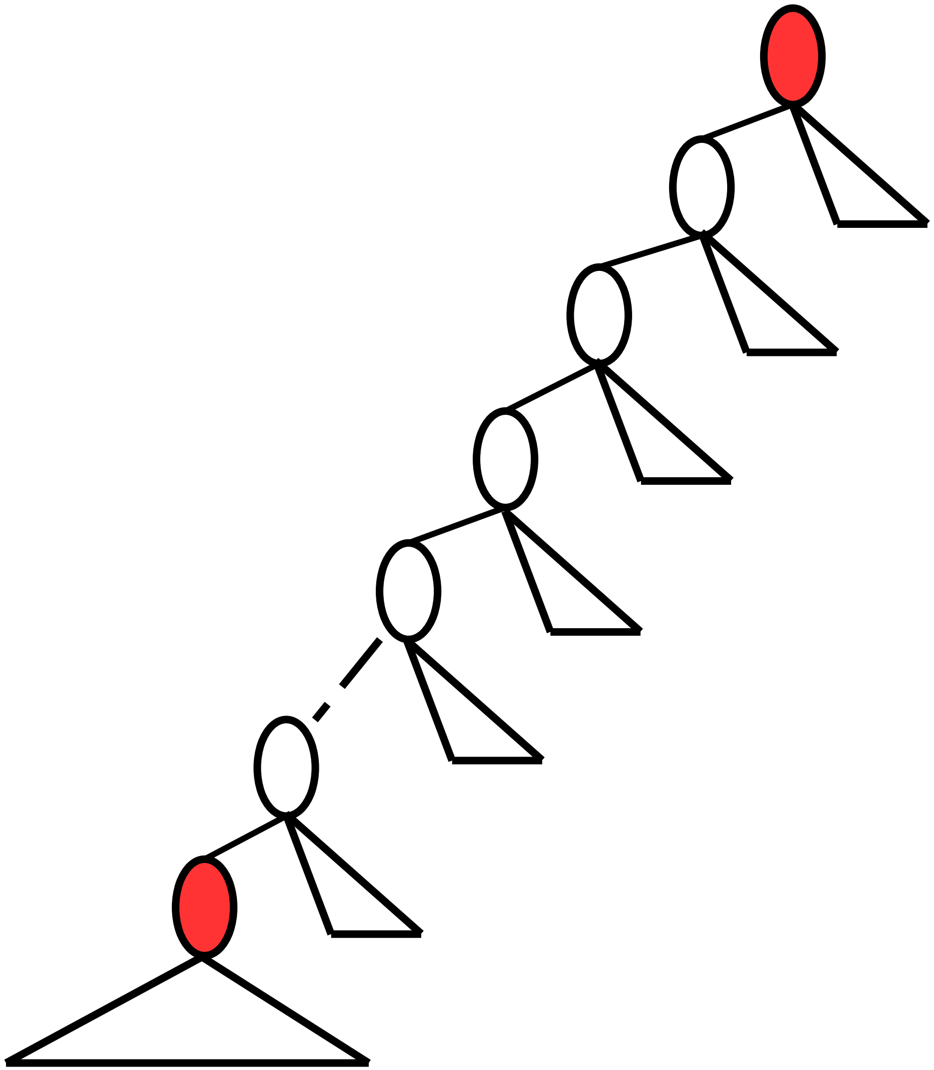
Proof : consider a minimum modification of G into a cograph having cotree T as follows



Affected vertices ≤ 2k   Affected internal nodes ≤ 2k

Proof of the size of the kernel : O(k^3)

Proof : consider a minimum modification of G into a cograph having cotree T as follows

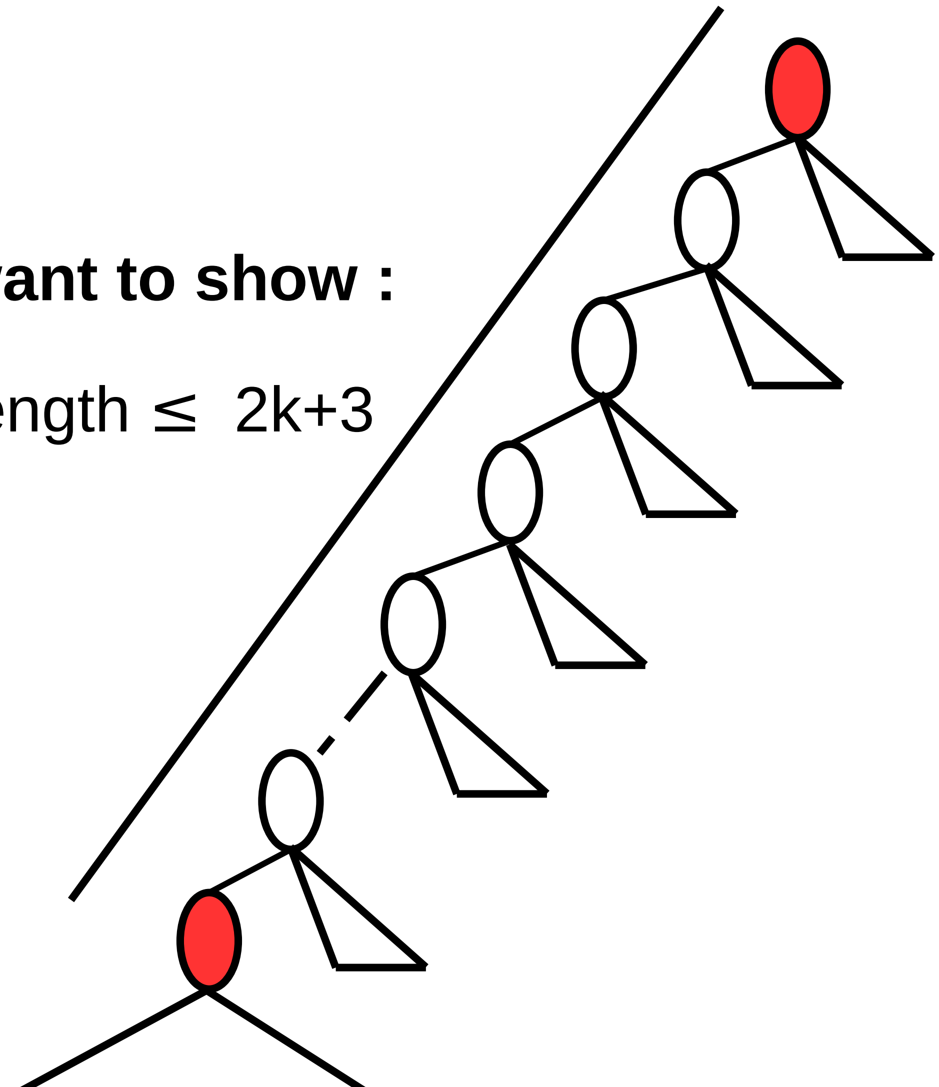


Proof of the size of the kernel : $O(k^3)$

<u>Proof :</u> consider a minimum modification of G into a cograph having cotree T as follows

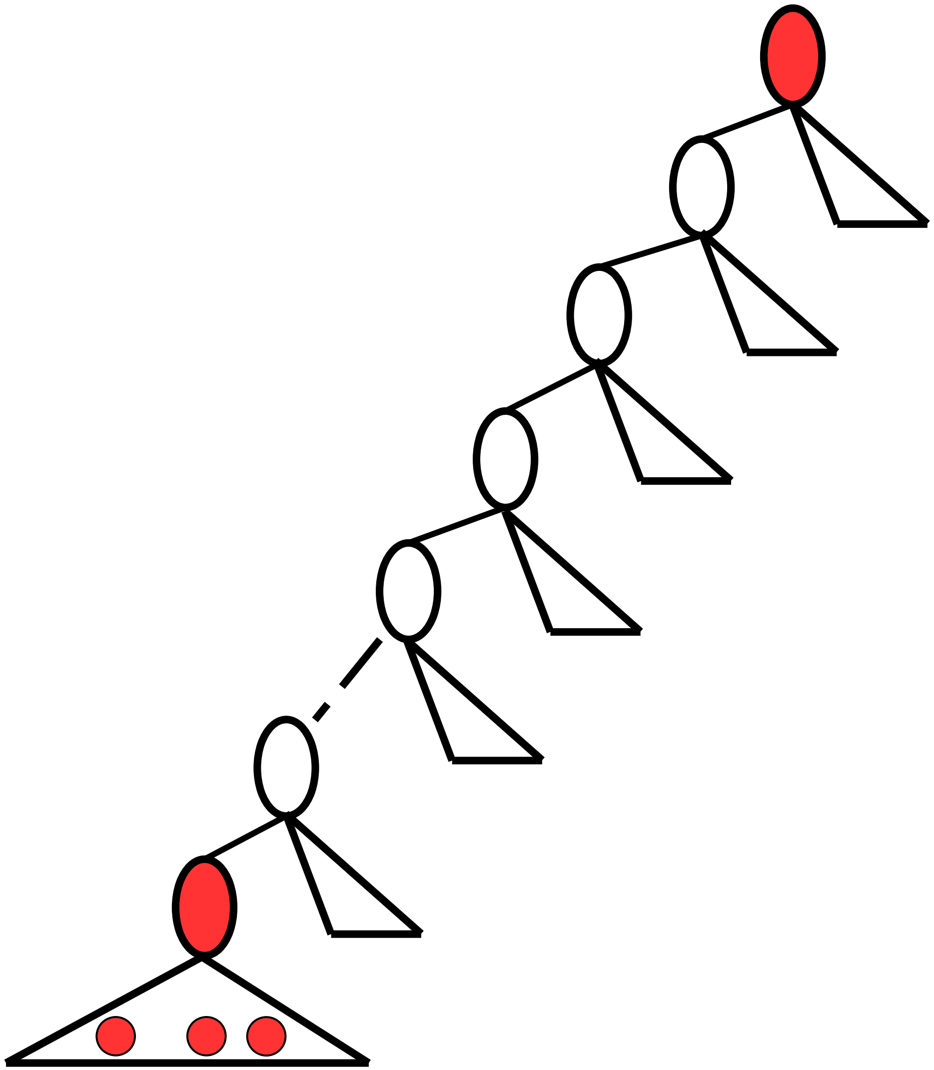
We want to show :

Length $\leq$ 2k+3



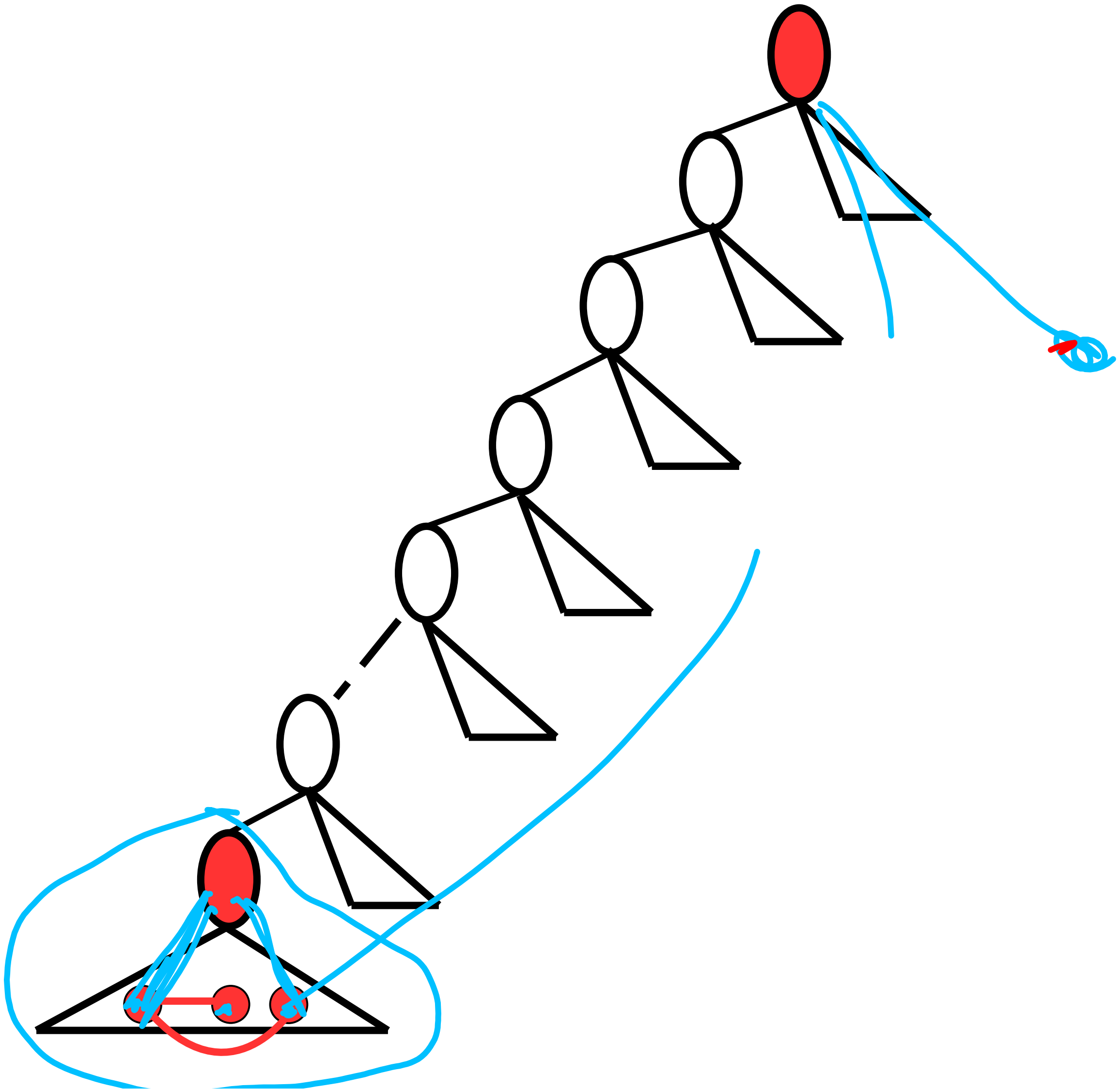
Proof of the size of the kernel : $O(k^3)$

Proof : consider a minimum modification of G into a cograph having cotree T as follows



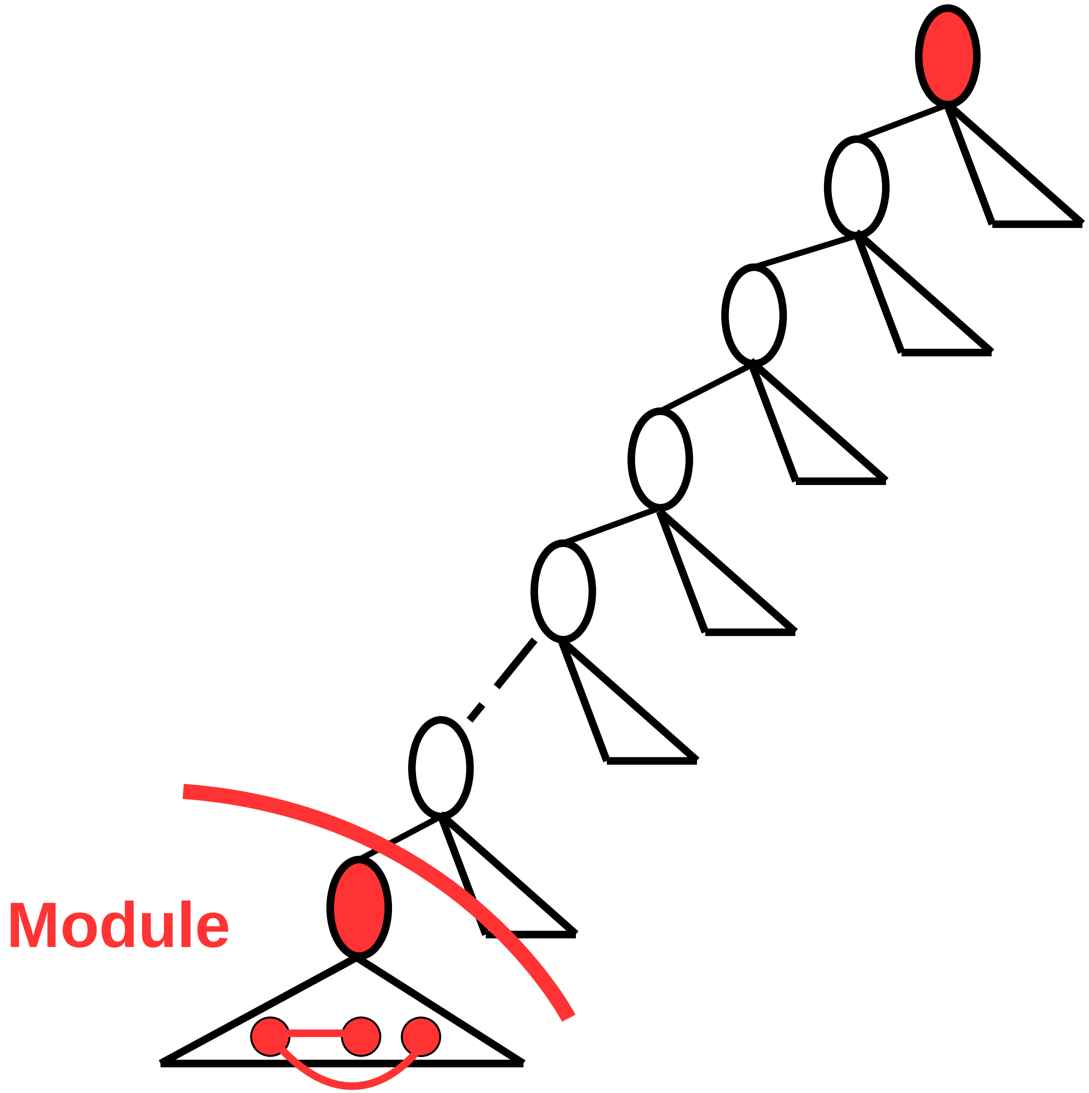
Proof of the size of the kernel : O(k^3)

Proof : consider a minimum modification of G into a cograph having cotree T as follows



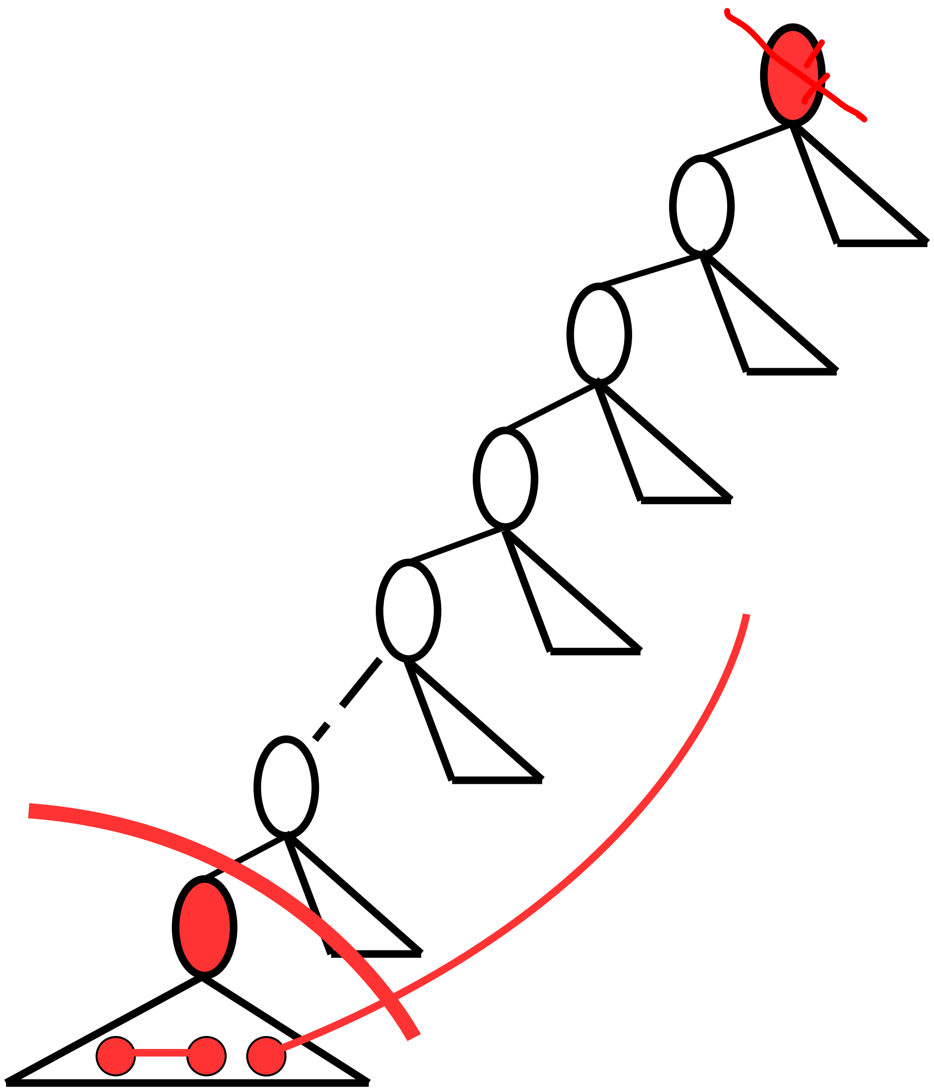
Proof of the size of the kernel : $O(k^3)$

<u>Proof :</u> consider a minimum modification of G into a cograph having cotree T as follows



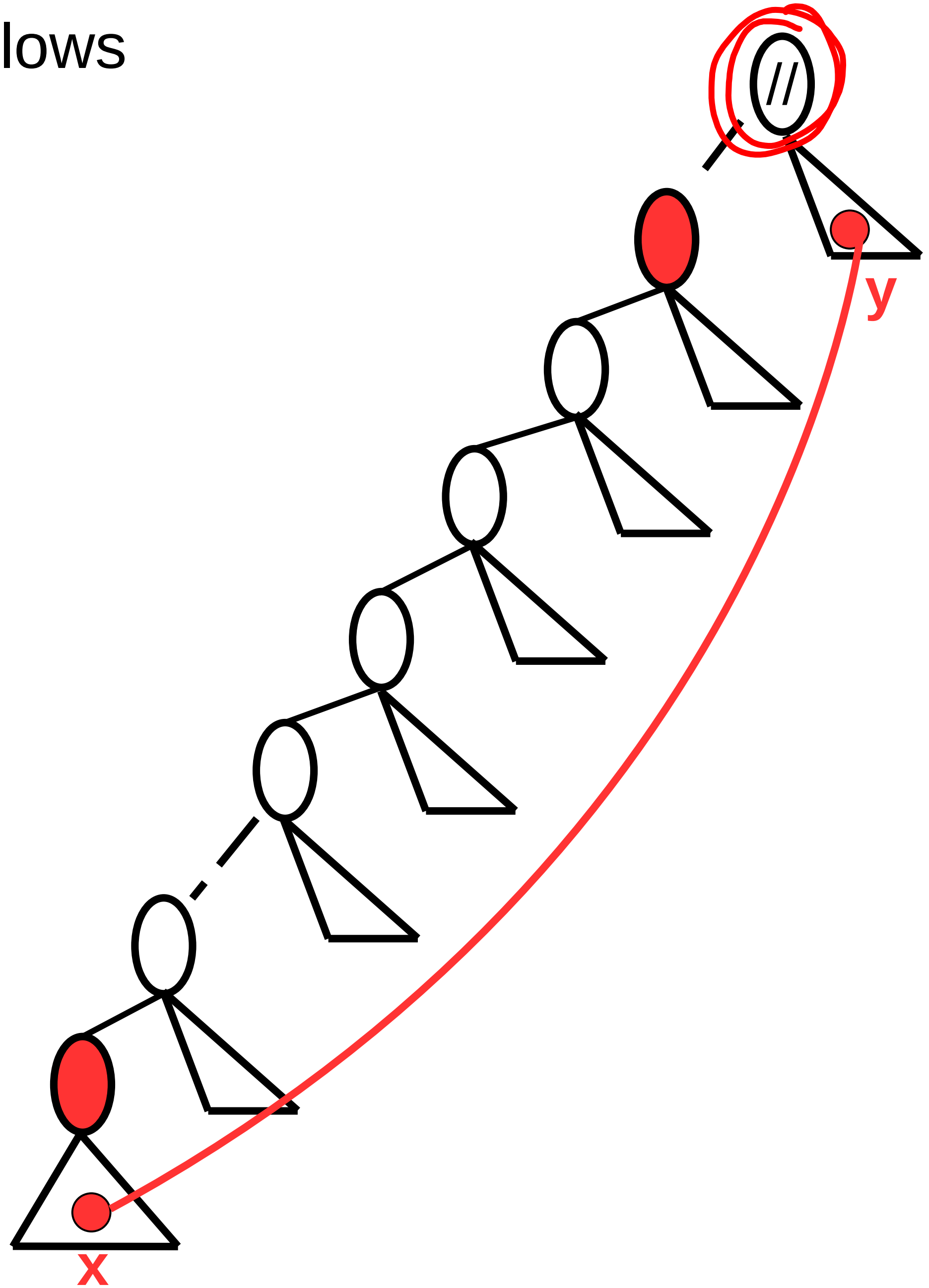
Proof of the size of the kernel : O(k^3)

Proof : consider a minimum modification of G into a cograph having cotree T as follows



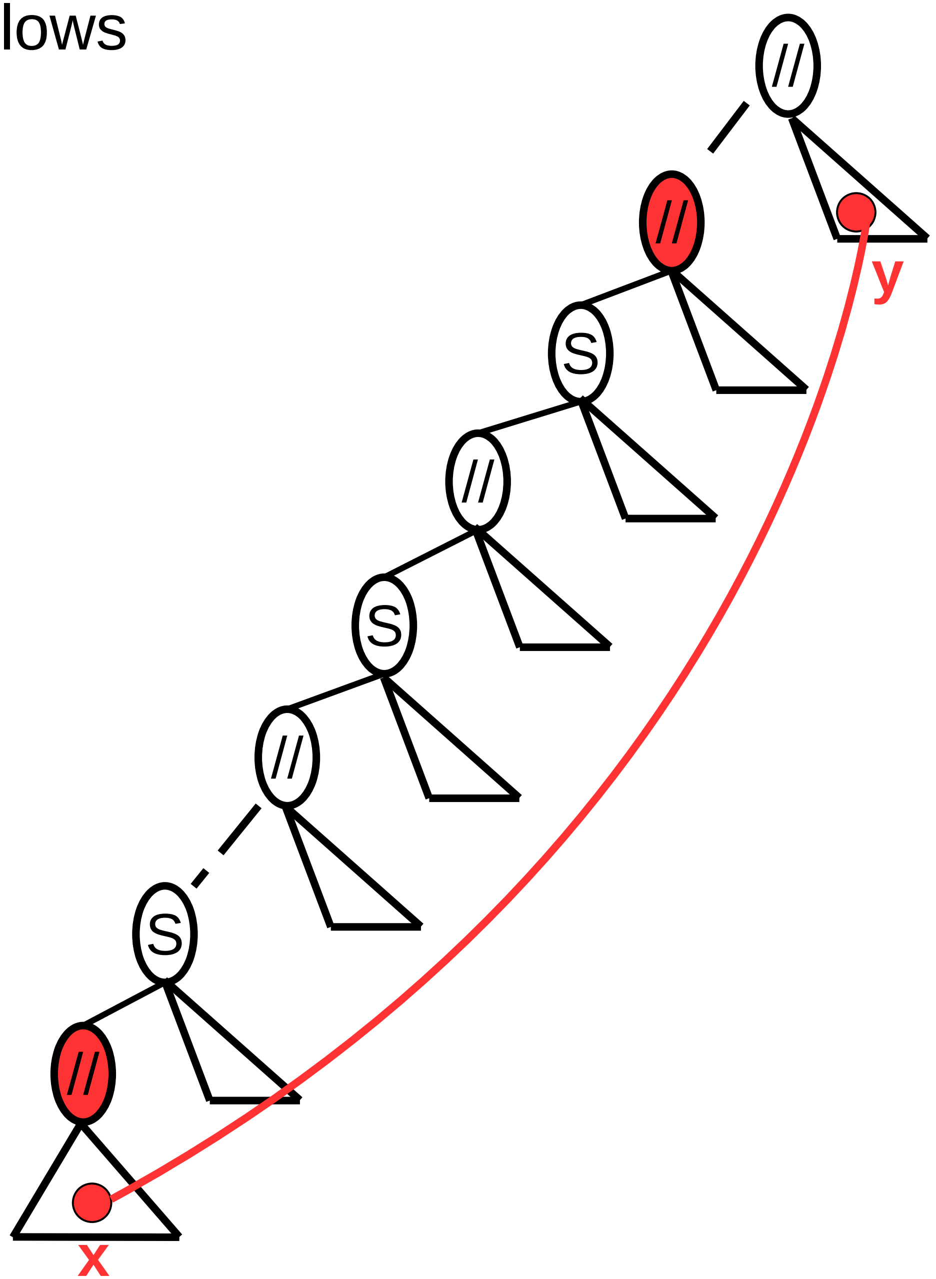
Proof of the size of the kernel : O(k^3)

Proof : consider a minimum modification of G into a cograph having cotree T as follows



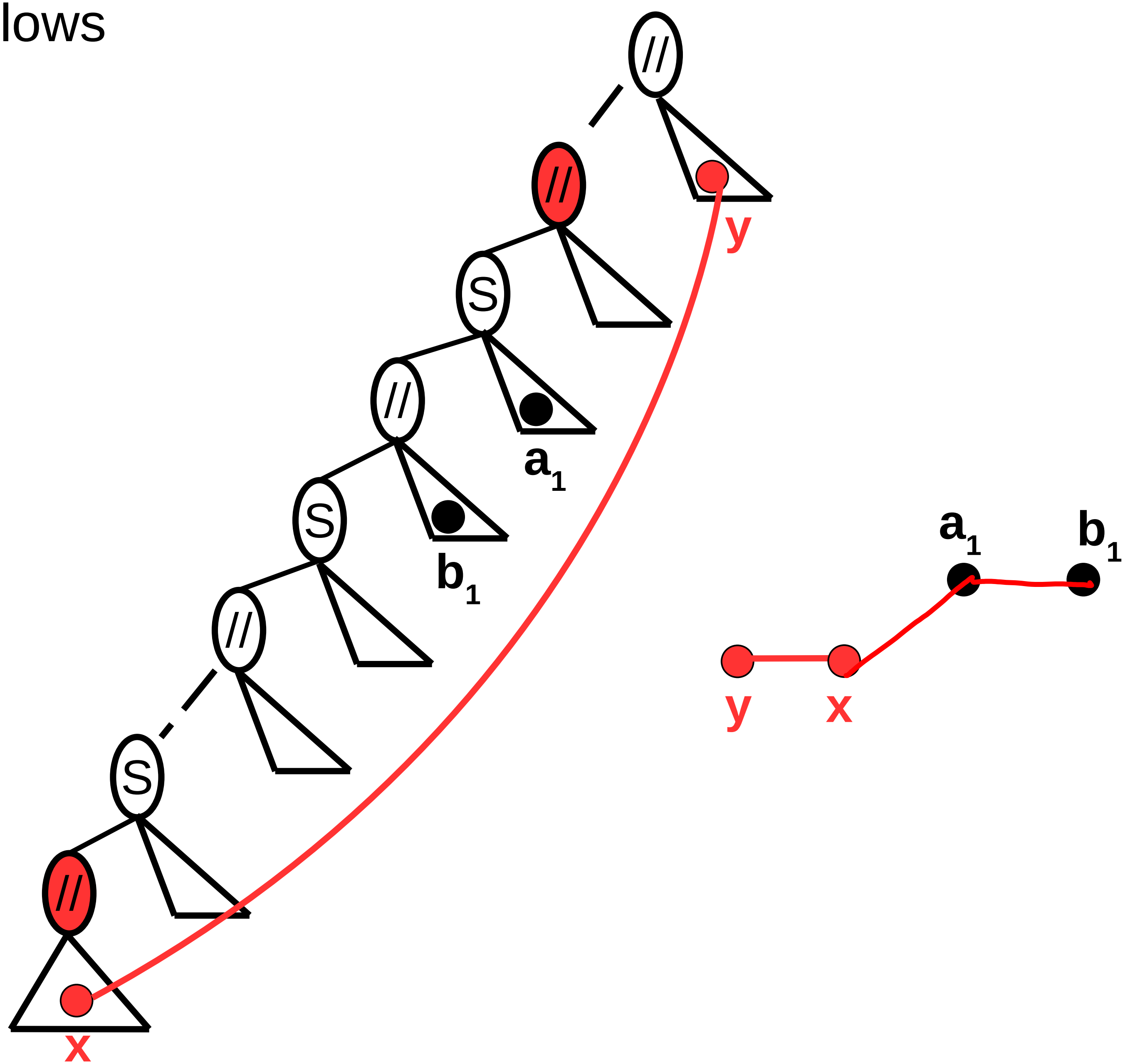
Proof of the size of the kernel : O(k^3)

Proof : consider a minimum modification of G into a cograph having cotree T as follows



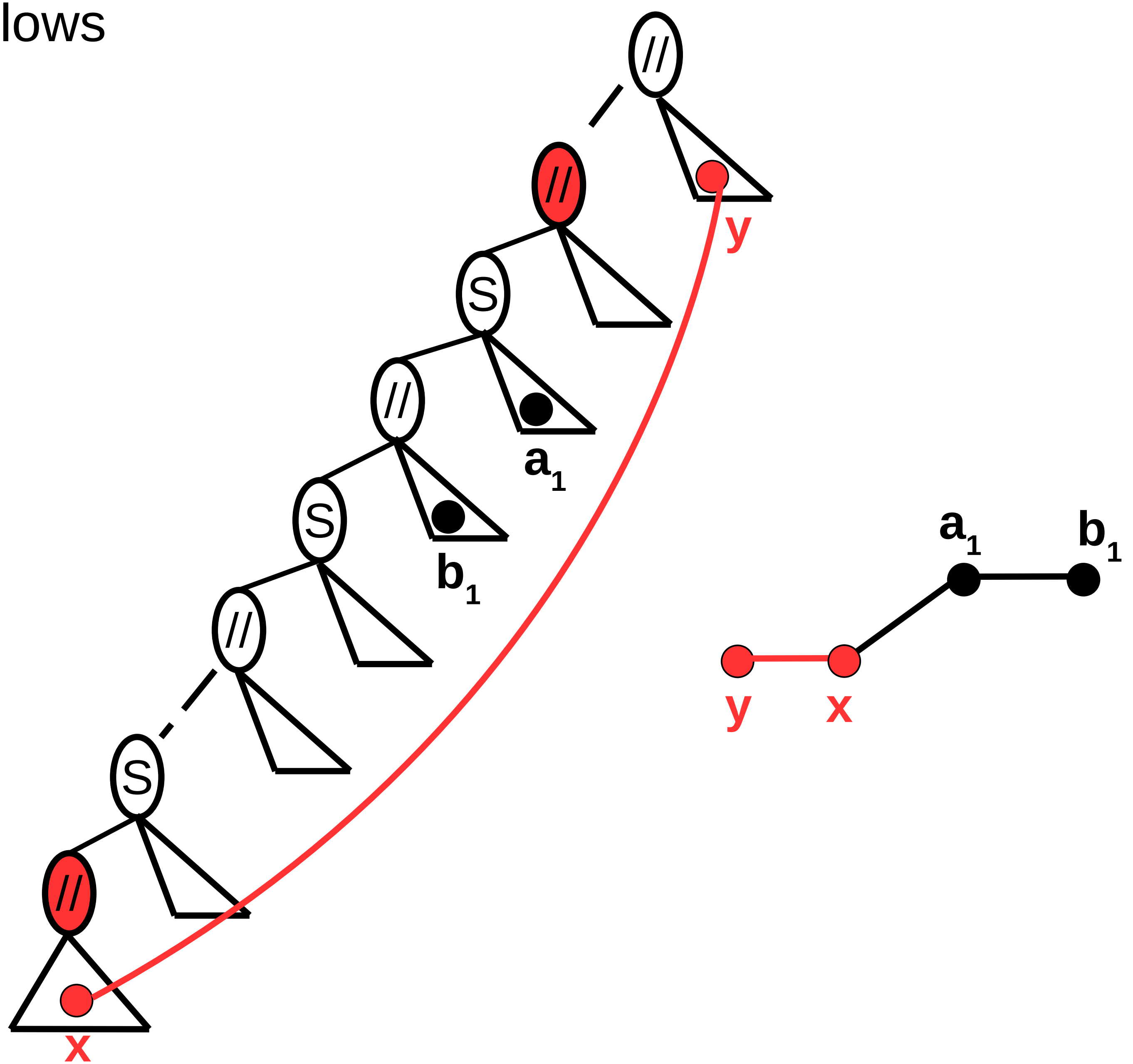
Proof of the size of the kernel : O(k^3)

Proof : consider a minimum modification of G into a cograph having cotree T as follows



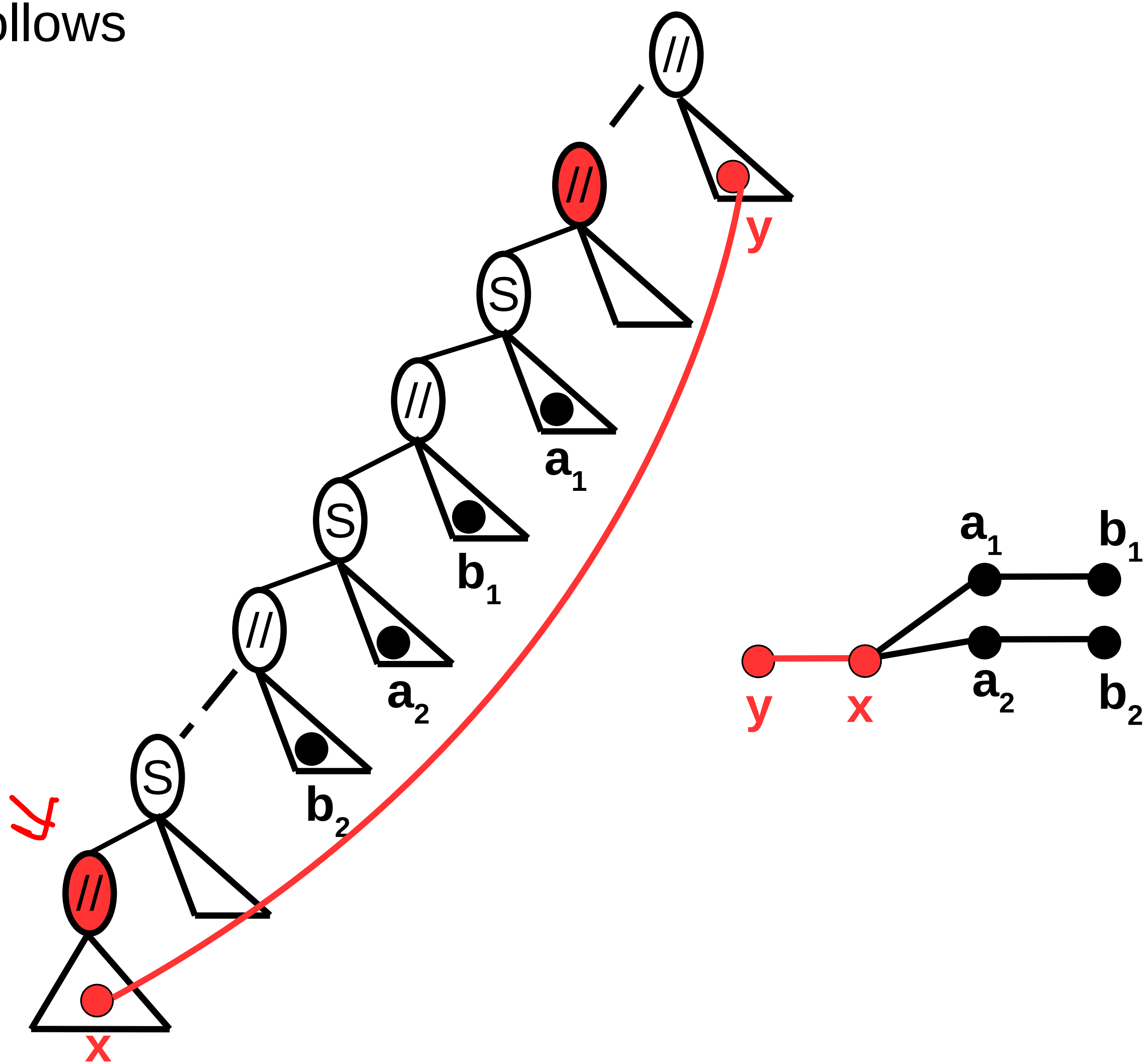
Proof of the size of the kernel : O(k^3)

Proof : consider a minimum modification of G into a cograph having cotree T as follows



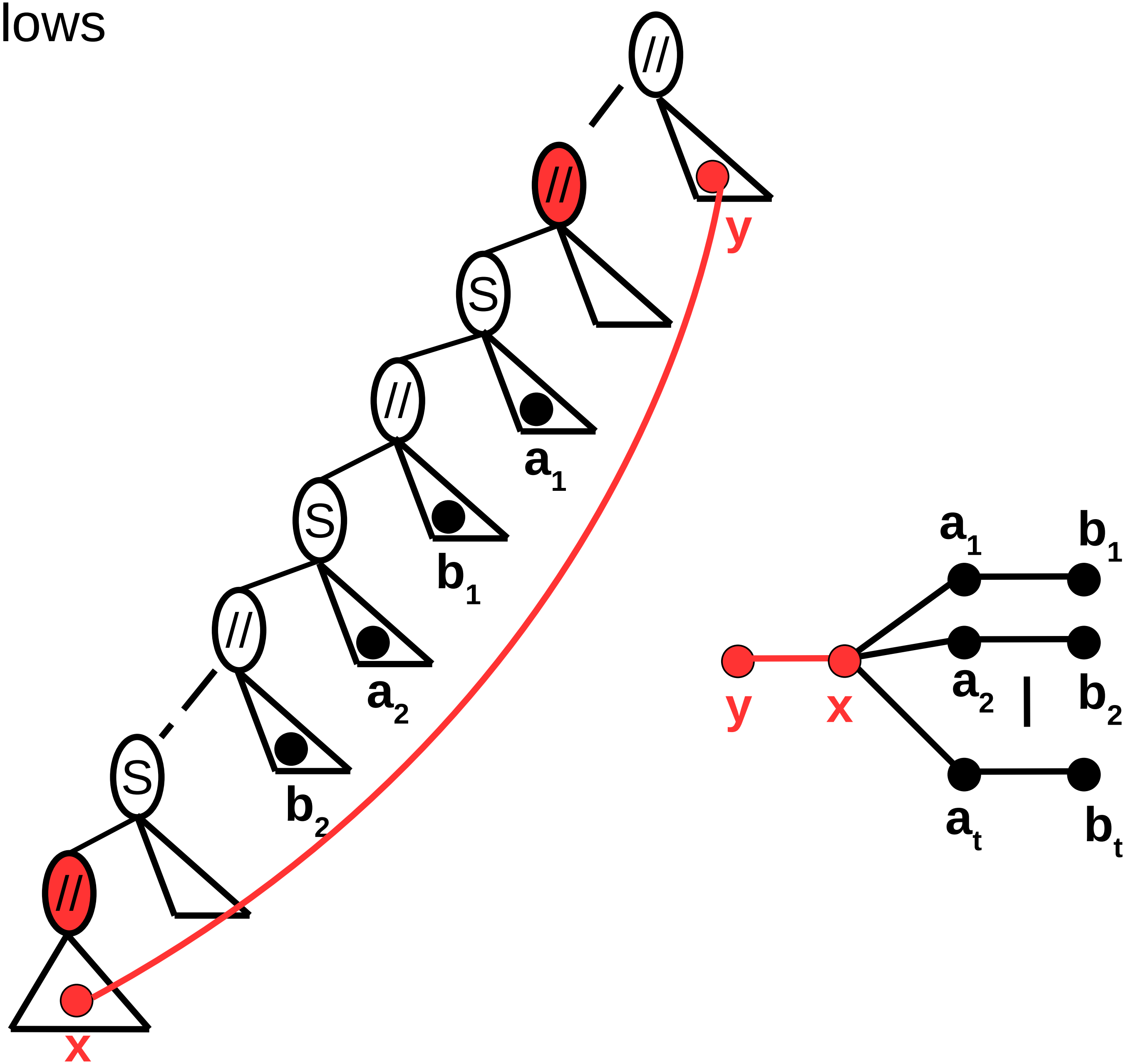
Proof of the size of the kernel : $O(k^3)$

Proof : consider a minimum modification of G into a cograph having cotree T as follows



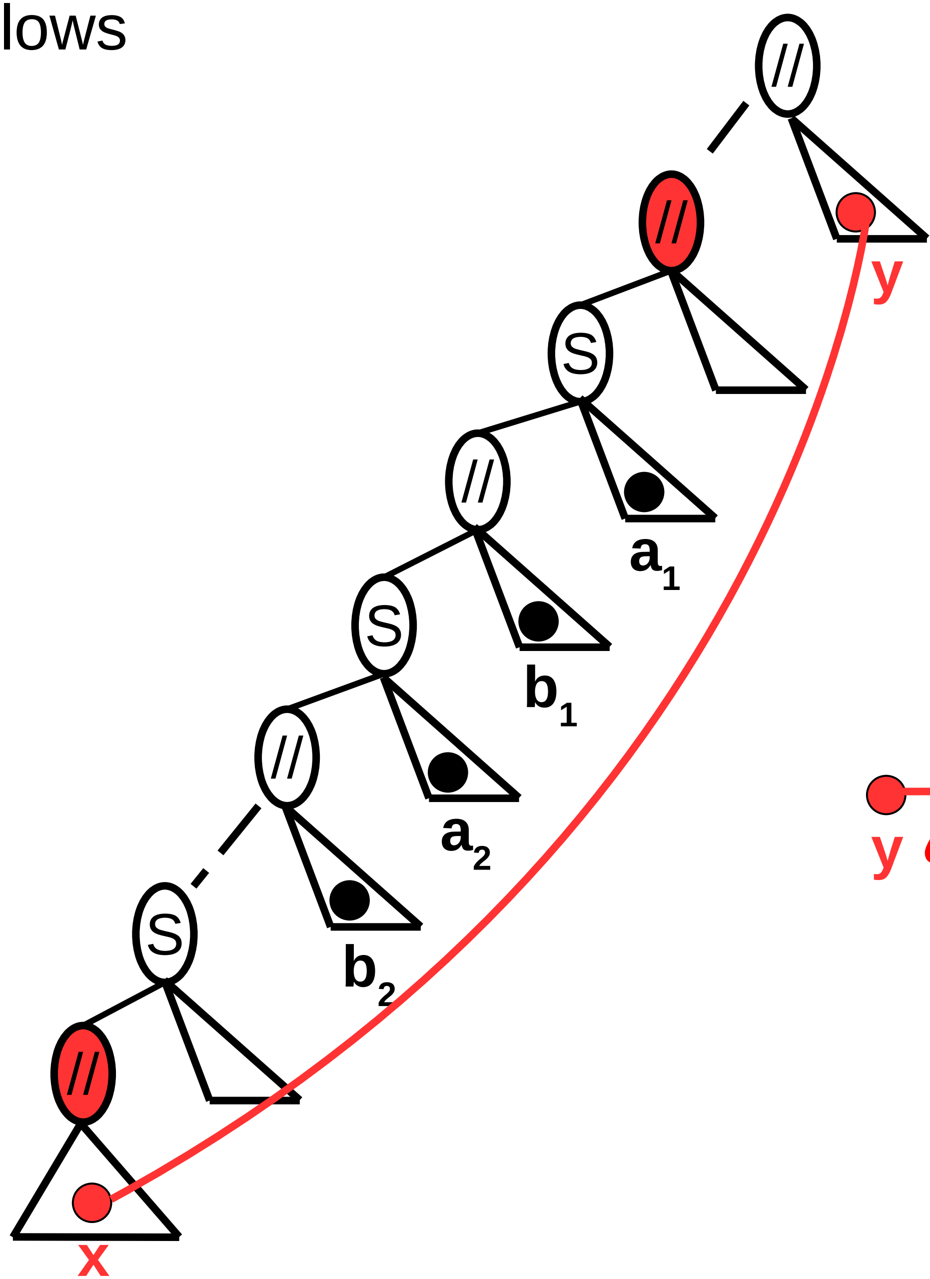
Proof of the size of the kernel : O(k^3)

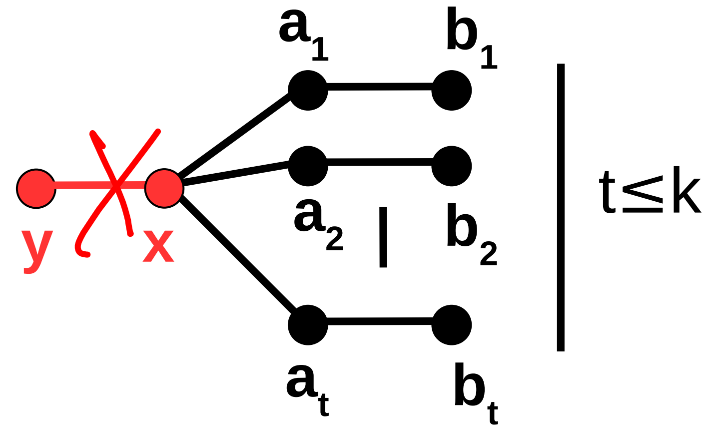
Proof : consider a minimum modification of G into a cograph having cotree T as follows



Proof of the size of the kernel : $O(k^3)$

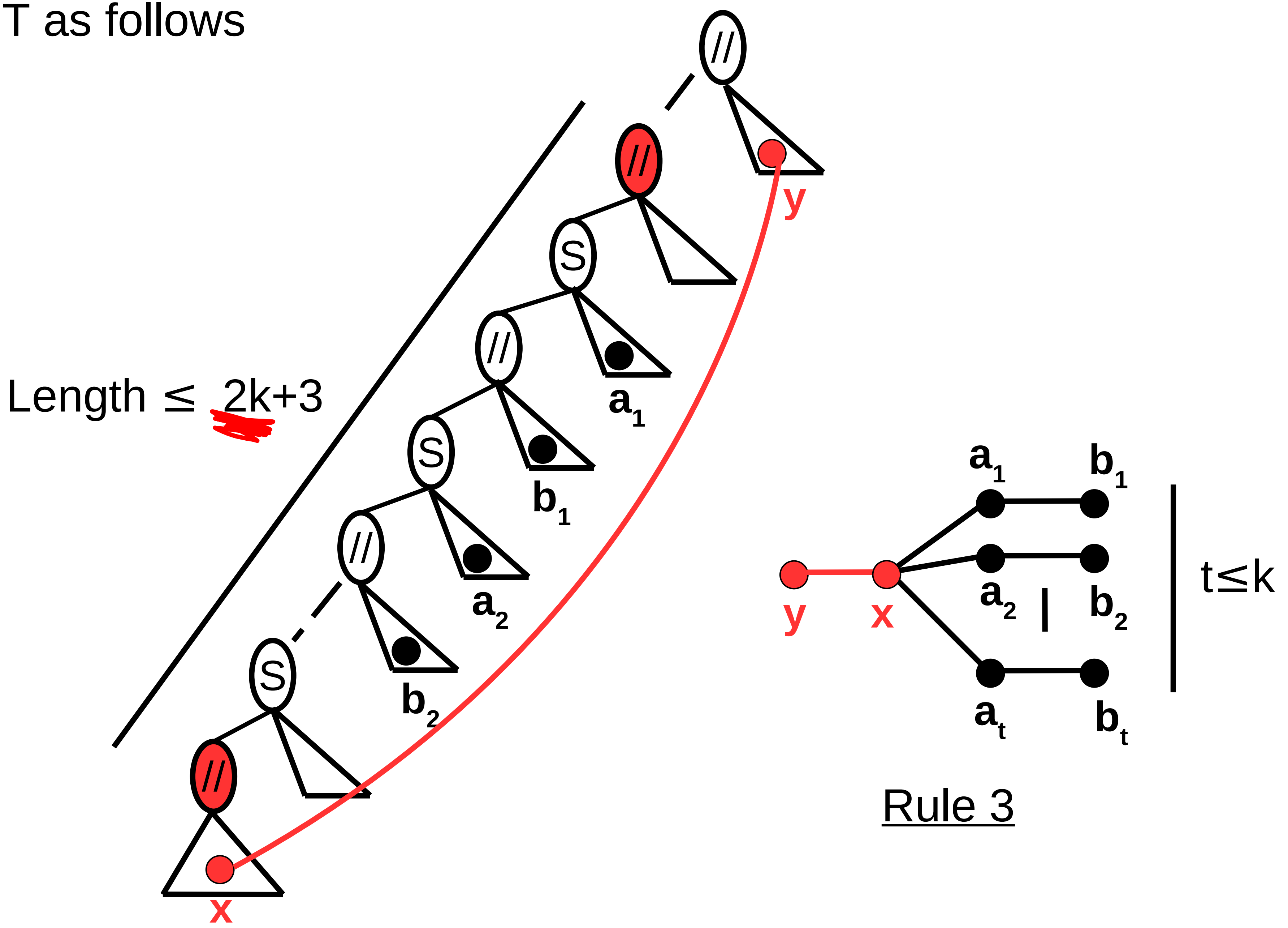
Proof : consider a minimum modification of G into a cograph having cotree T as follows



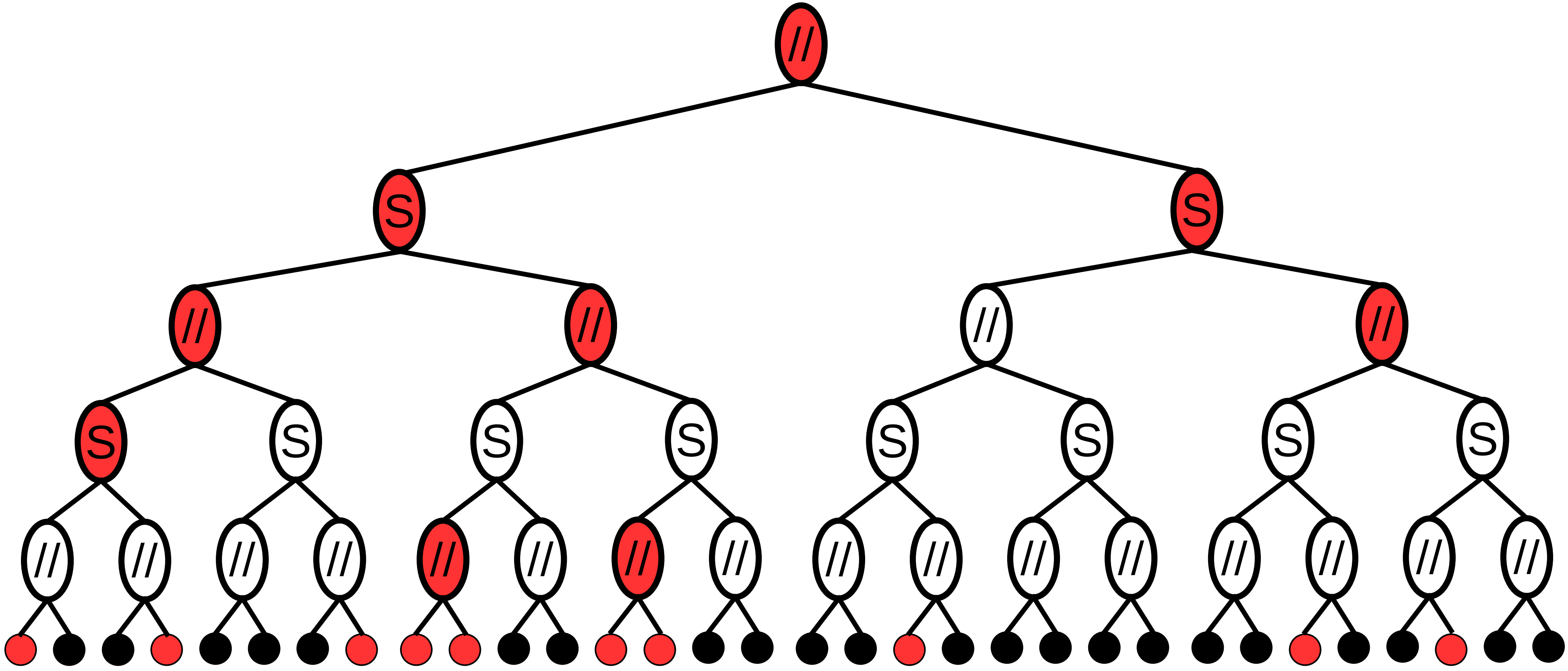


Proof of the size of the kernel : $O(k^3)$

Proof : consider a minimum modification of G into a cograph having cotree T as follows



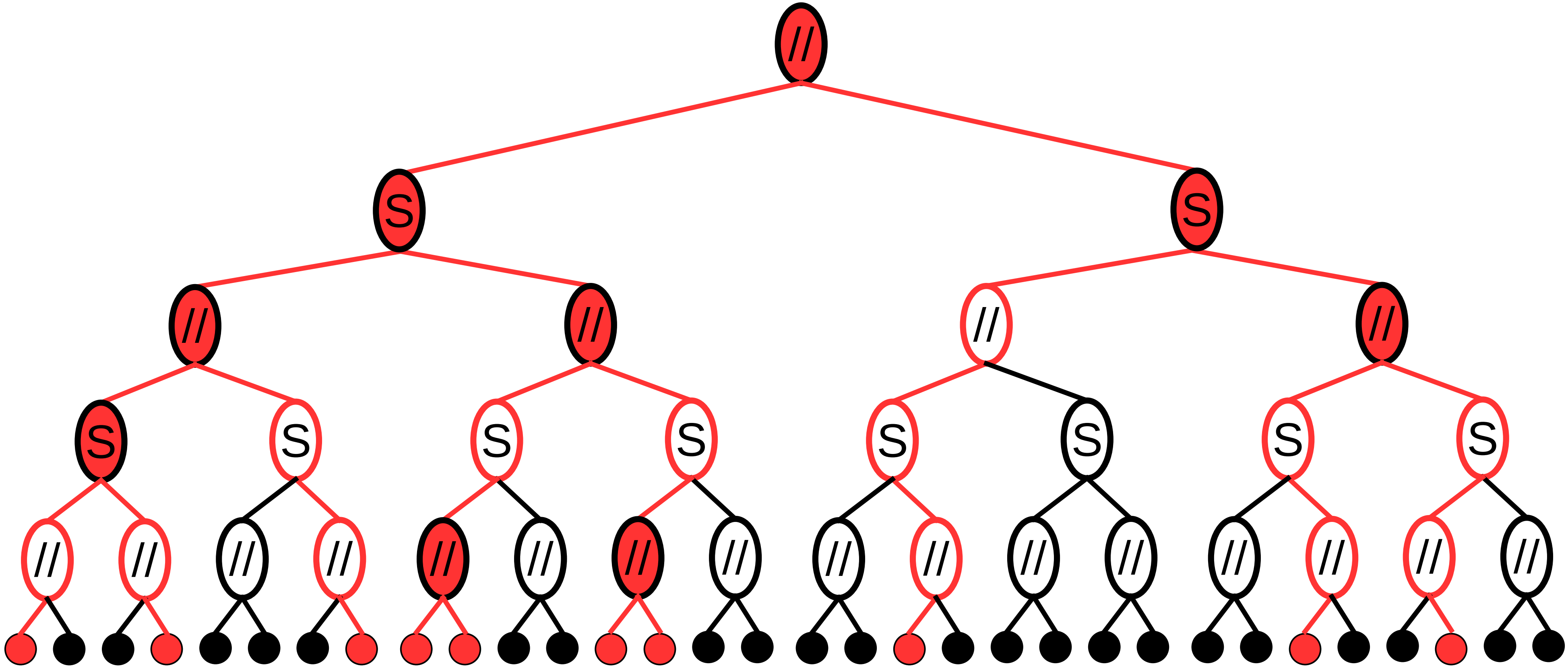
Counting the number of vertices



Affected vertices ≤ 2k

Affected internal nodes ≤ 2k

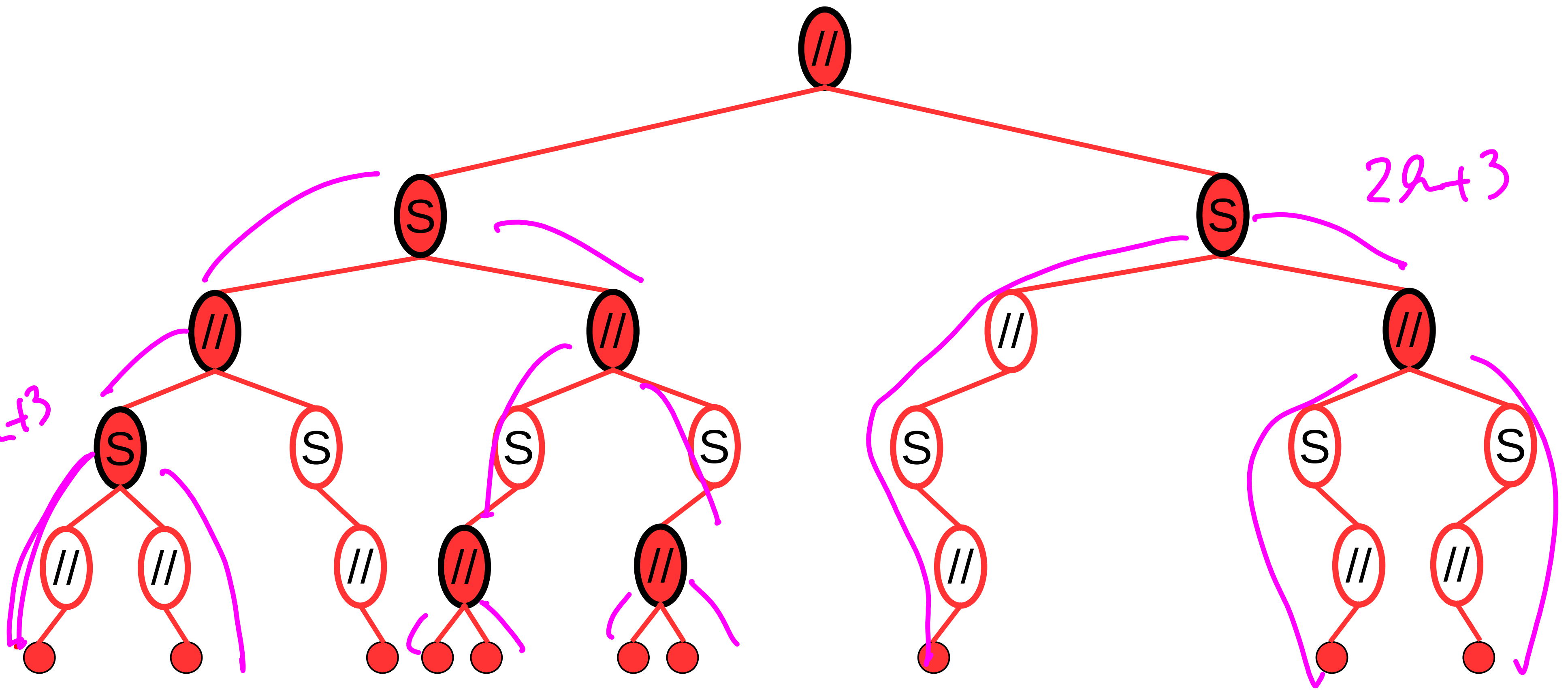
Counting the number of vertices



Affected vertices ≤ 2k

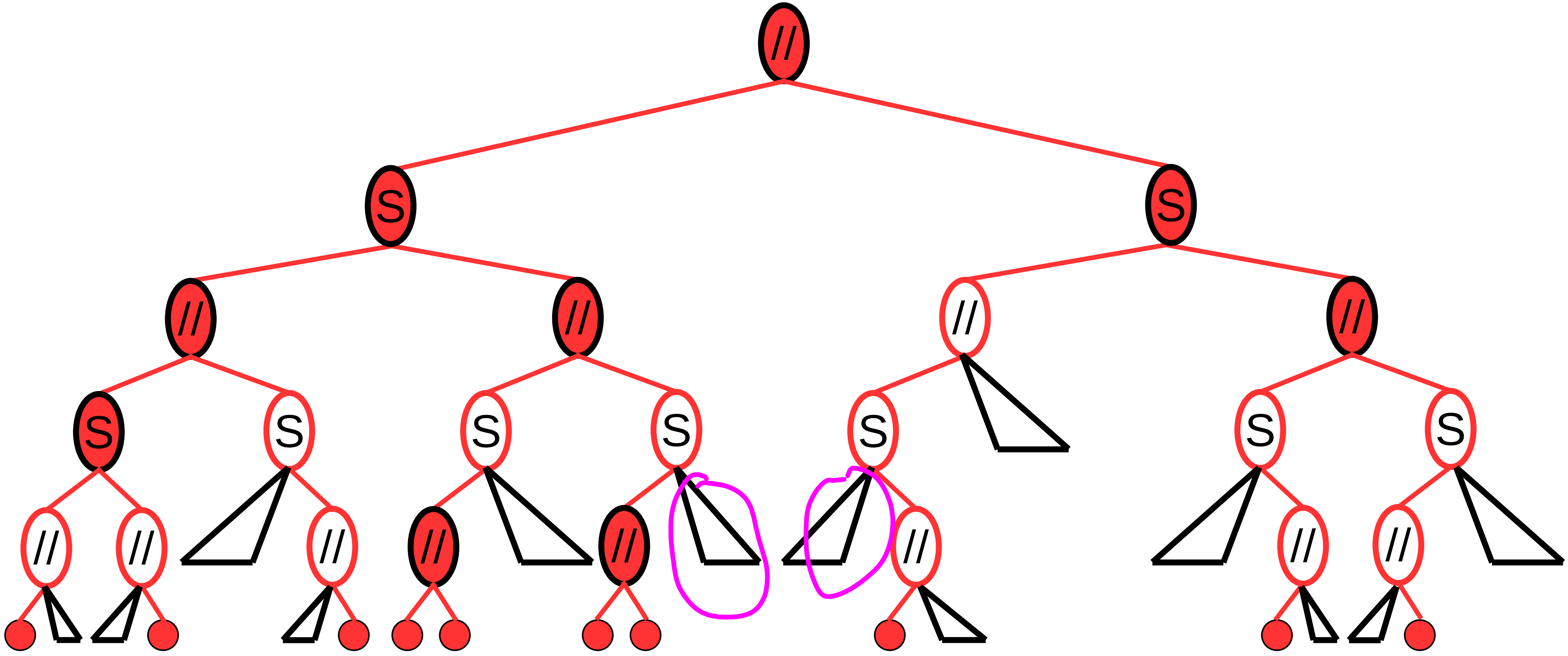
Affected internal nodes ≤ 2k

Counting the number of vertices

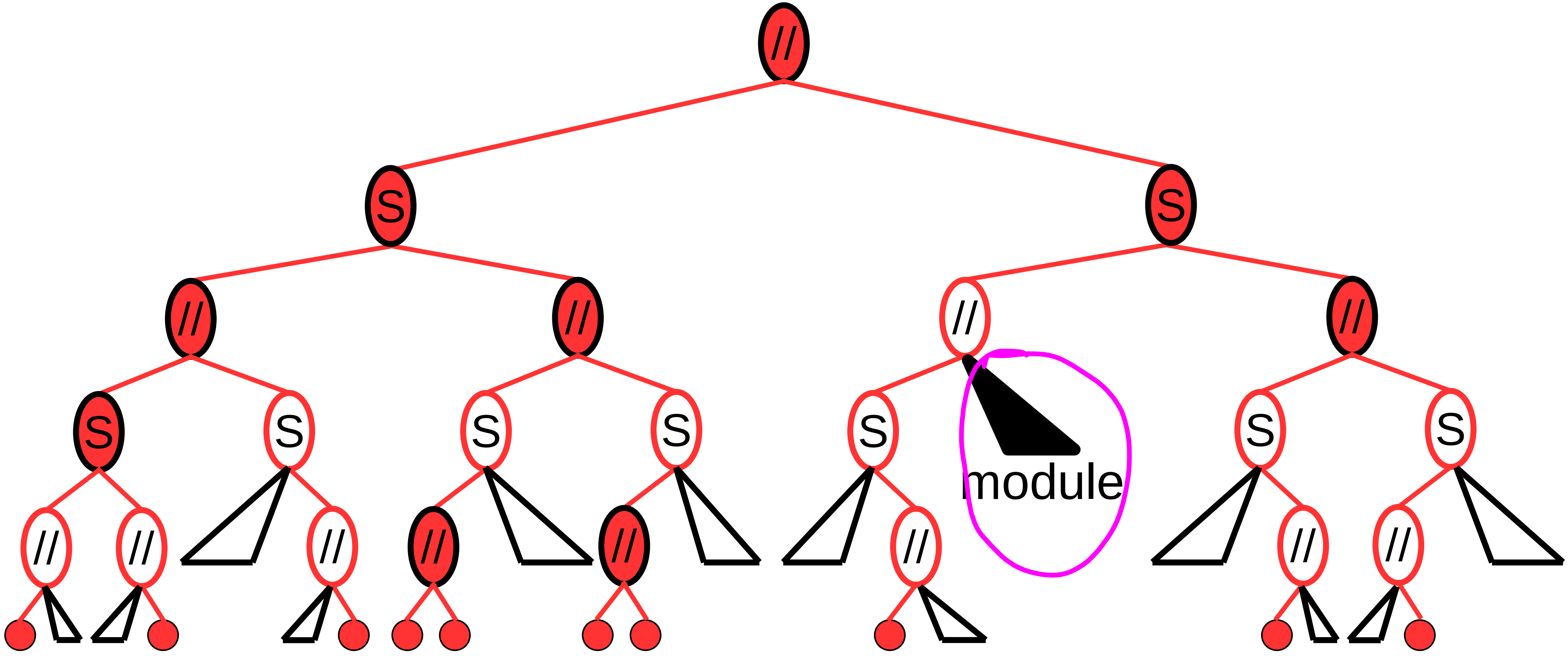


$\leq 2 \times 2n \times (2n+3) = O(n^2)$

Counting the number of vertices

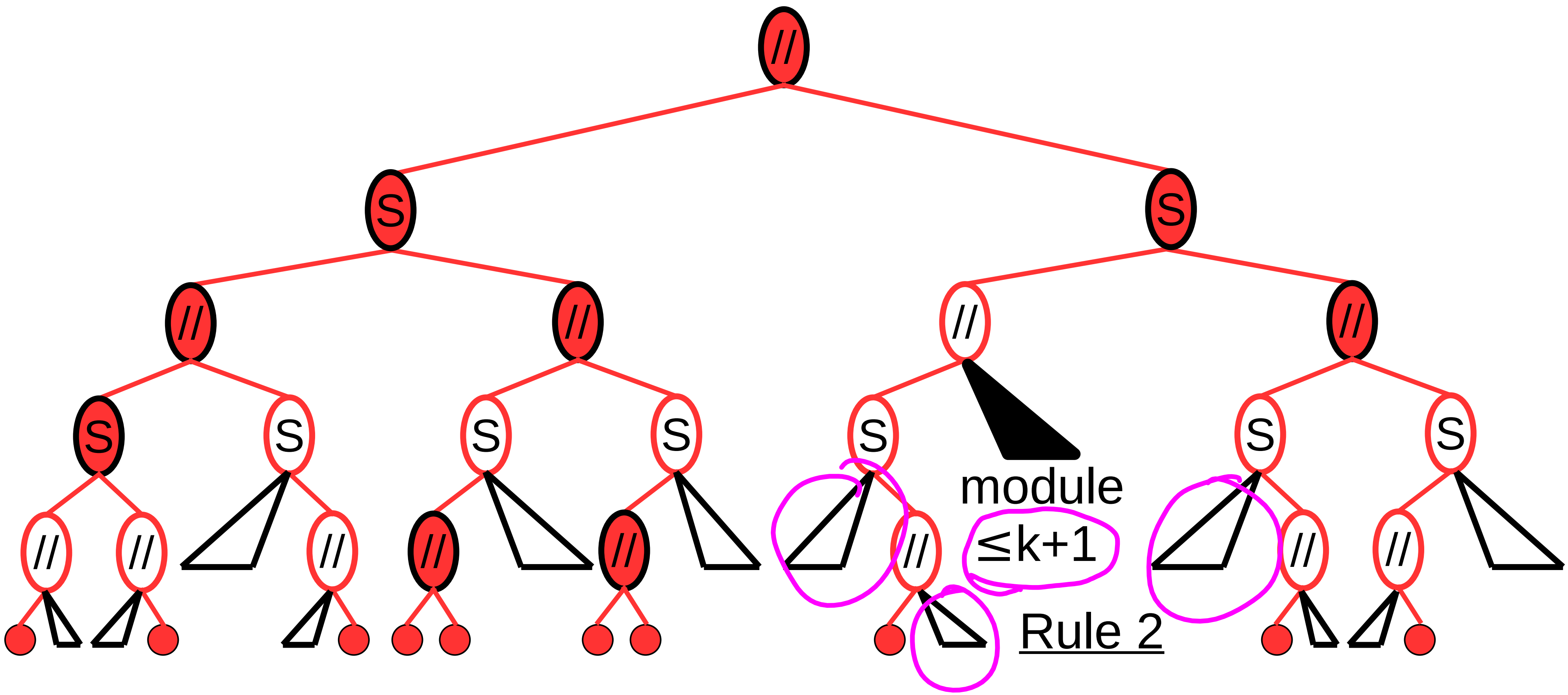


Counting the number of vertices



$O(n^2 \times (n+1)) = O(n^3)$ vertices.

Counting the number of vertices



The reduction algorithm

The generic reduction algorithm :

- While there exists some rules that applies
  - Apply an arbitrary rule among the rules that apply

The reduction algorithm

<u>The generic reduction algorithm :</u>

- While there exists some rules that applies
  - Apply an arbitrary rule among the rules that apply
- At the end: you get a reduced graph

The reduction algorithm

The generic reduction algorithm :

- While there exists some rules that applies
  - Apply an arbitrary rule among the rules that apply
- At the end: you get a reduced graph

One particular way of doing it :

- Apply rule 3 until it does not apply anymore
- Apply rule 2 until it does not apply anymore
- Apply rule 1 until it does not apply anymore

The reduction algorithm

The generic reduction algorithm :

- While there exists some rules that applies
  - Apply an arbitrary rule among the rules that apply
- At the end: you get a reduced graph

One particular way of doing it :

- Apply rule 3 until it does not apply anymore
- Apply rule 2 until it does not apply anymore
- Apply rule 1 until it does not apply anymore

Question : Is the graph obtained reduced under rules 1,2,3 ?

The reduction algorithm

Lemma :

If graph G is reduced under rule 3, then applying rule 2 to G gives a graph G' that is also reduced under rule 3.

Exercise : Prove the lemma above.

The reduction algorithm

Lemma :

If graph G is reduced under rule 3, then applying rule 2 to G gives a graph G’ that is also reduced under rule 3.

Exercise : Prove the lemma above.

Hint :

If M is a (non-trivial) module of graph G, then any P_4 of G that is not included in M has at most one vertex in M.

The reduction algorithm

<u>Lemma :</u>

If graph G is reduced under rules 2 and 3, then applying rule 1 to G gives a graph G’ that is also reduced under rules 2 and 3.

The reduction algorithm

The generic reduction algorithm :

- While there exists some rules that applies
    - Apply an arbitrary rule among the rules that apply
- At the end: you get a reduced graph

One particular way of doing it :

- Apply rule 3 until it does not apply anymore
- Apply rule 2 until it does not apply anymore
- Apply rule 1 until it does not apply anymore

Question : Is the graph obtained reduced under rules 1,2,3 ?

Question : does this algorithm run in polynomial time ?

The reduction algorithm

<u>The generic reduction algorithm :</u>

- While there exists some rules that applies
  - Apply an arbitrary rule among the rules that apply
- At the end: you get a reduced graph

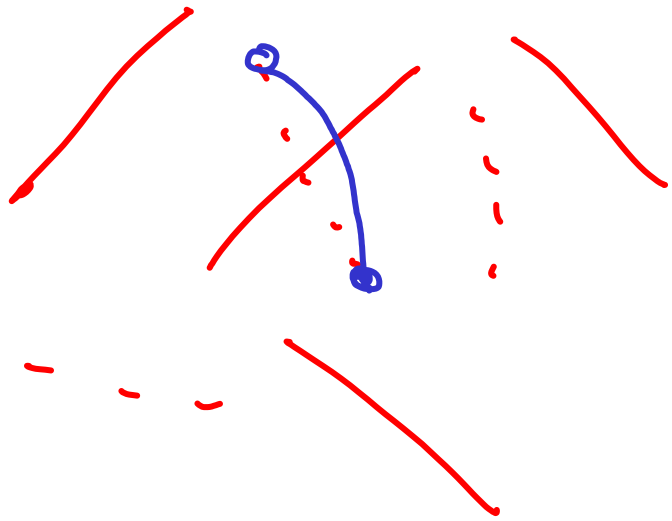
<u>One particular way of doing it :</u>

- Apply rule 3 until it does not apply anymore
- Apply rule 2 until it does not apply anymore
- Apply rule 1 until it does not apply anymore

<u>Question :</u> Is the graph obtained reduced under rules 1,2,3 ?

<u>Question :</u> does this algorithm run in polynomial time ?

<u>Subquestion :</u> does it even terminate ?



Q1: Can it really happen?

Q2: Is it a problem if it happens?

Practical limitations of kernels for edge modification problems

with Anne-Aymone Bourguin

What happens when k varies ?

- Why would k vary ?

What happens when k varies ?

- Why would k vary ?
  - We are not only interested in the decision problem

What happens when k varies ?

- Why would k vary ?
  - We are not only interested in the decision problem

- If $k_2 > k_1$ and rule 3 applies to (G,k_2) and gives (H_2,k_2') then
  - rule 3 also applies to (G,k_1) to give (H_1,k_1') and
  - $k_1' < k_2'$

What happens when k varies ?

- Why would k vary ?
  - We are not only interested in the decision problem

- If $k_2 > k_1$ and rule 3 applies to (G,k_2) and gives (H_2,k_2') then
  - rule 3 also applies to (G,k_1) to give (H_1,k_1') and
  - $k_1' < k_2'$

➡ If $k_2 > k_1$, then a series of reduction rules 3 performed from (G,k_2) can also be performed from (G,k_1) and gives a smaller graph

What happens when k varies ?

- Why would k vary ?
  - We are not only interested in the decision problem

- If $k_2 > k_1$ and rule 3 applies to (G,k_2) and gives (H_2,k_2') then
  - rule 3 also applies to (G,k_1) to give (H_1,k_1') and
  - $k_1' < k_2'$

→ If $k_2 > k_1$, then a series of reduction rules 3 performed from (G,k_2) can also be performed from (G,k_1) and gives a smaller graph

- The same should be checked for rules 2 and 1

What happens when k varies ?

- Why would k vary ?
  - We are not only interested in the decision problem

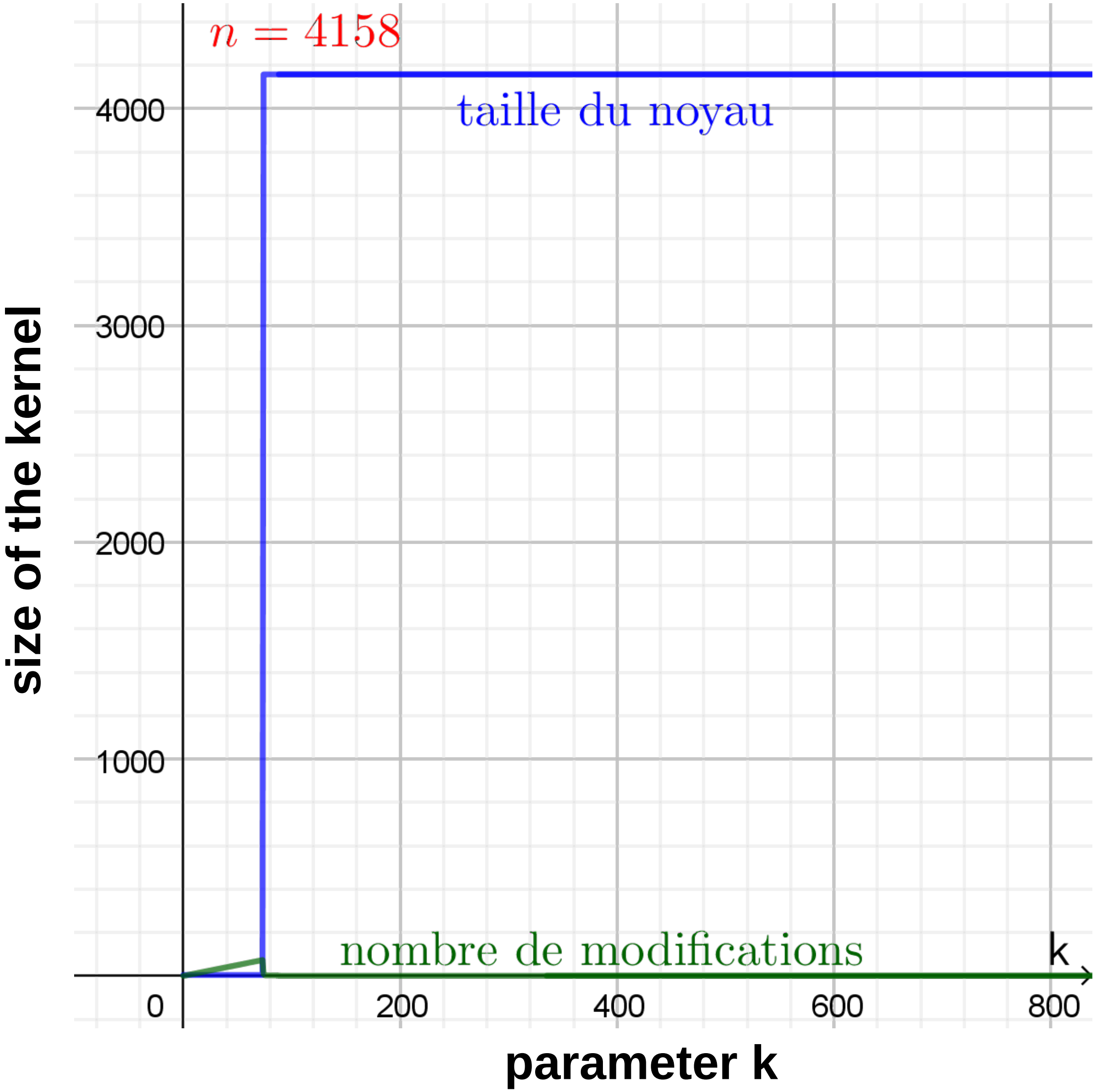
- If $k_2 > k_1$ and rule 3 applies to (G,k_2) and gives (H_2,k_2') then
  - rule 3 also applies to (G,k_1) to give (H_1,k_1') and
  - $k_1' < k_2'$

➡ If $k_2 > k_1$, then a series of reduction rules 3 performed from (G,k_2) can also be performed from (G,k_1) and gives a smaller graph

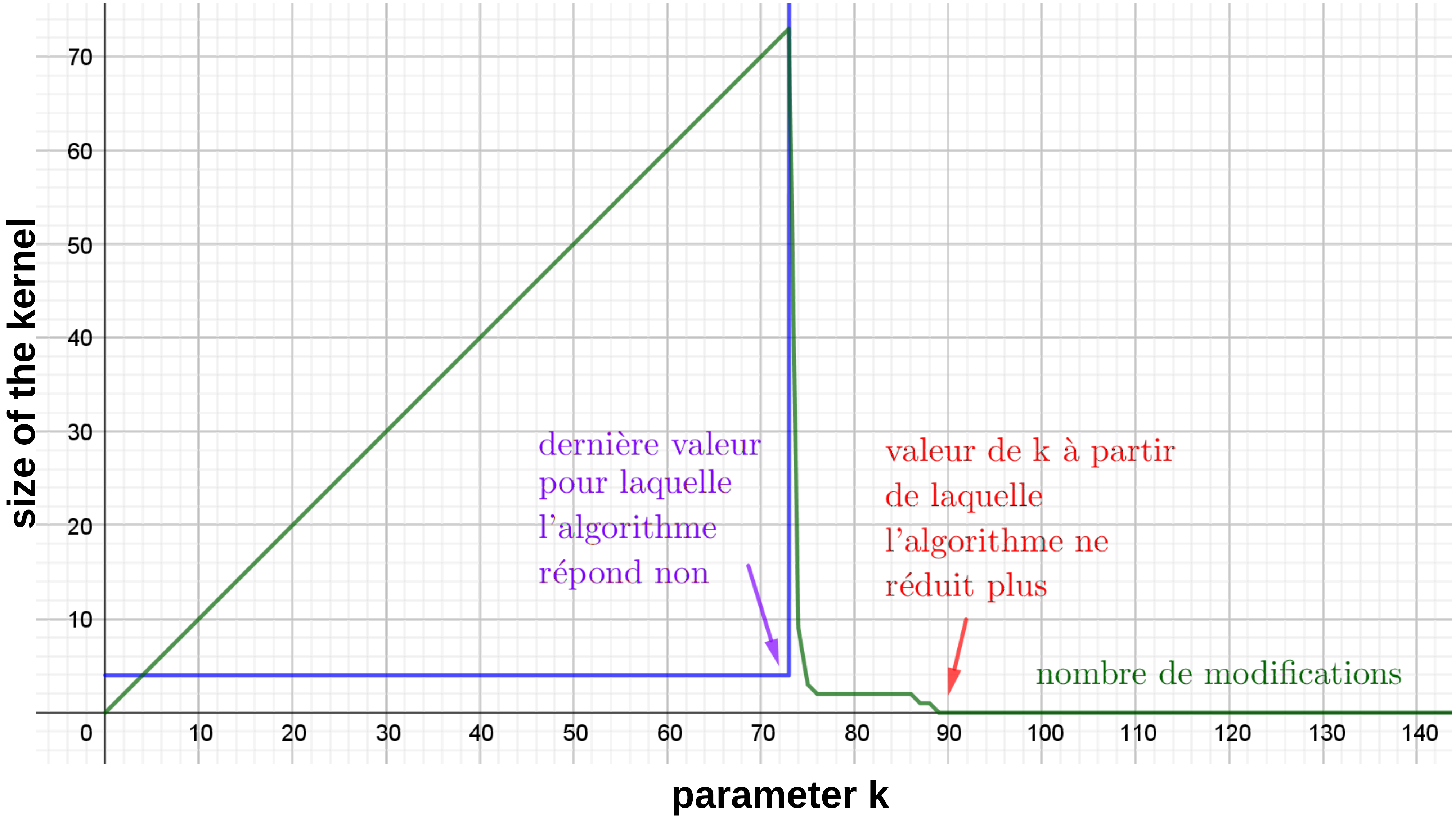
- The same should be checked for rules 2 and 1

The size of the kernel increases when k increases

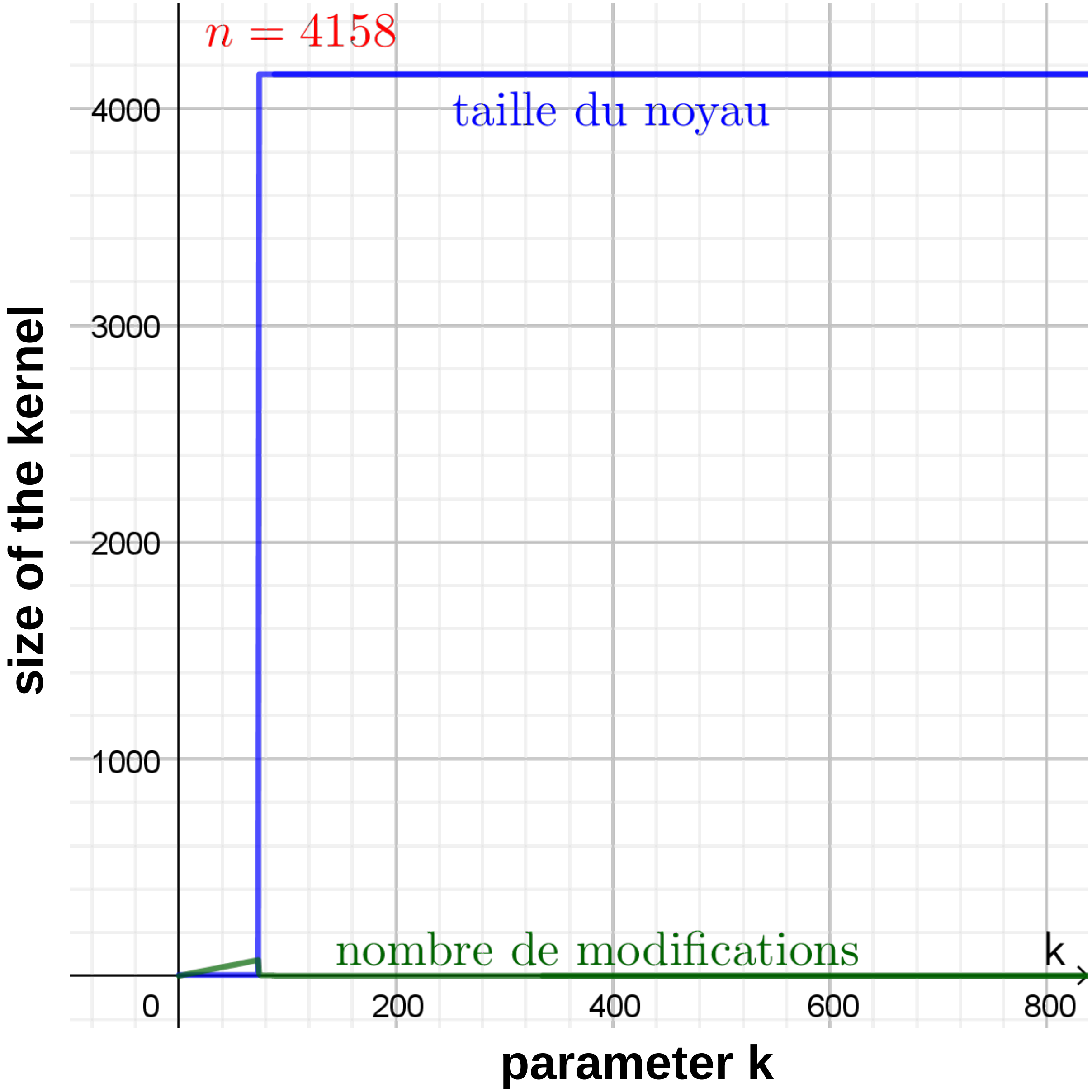
Size of reduced instance as a function of k



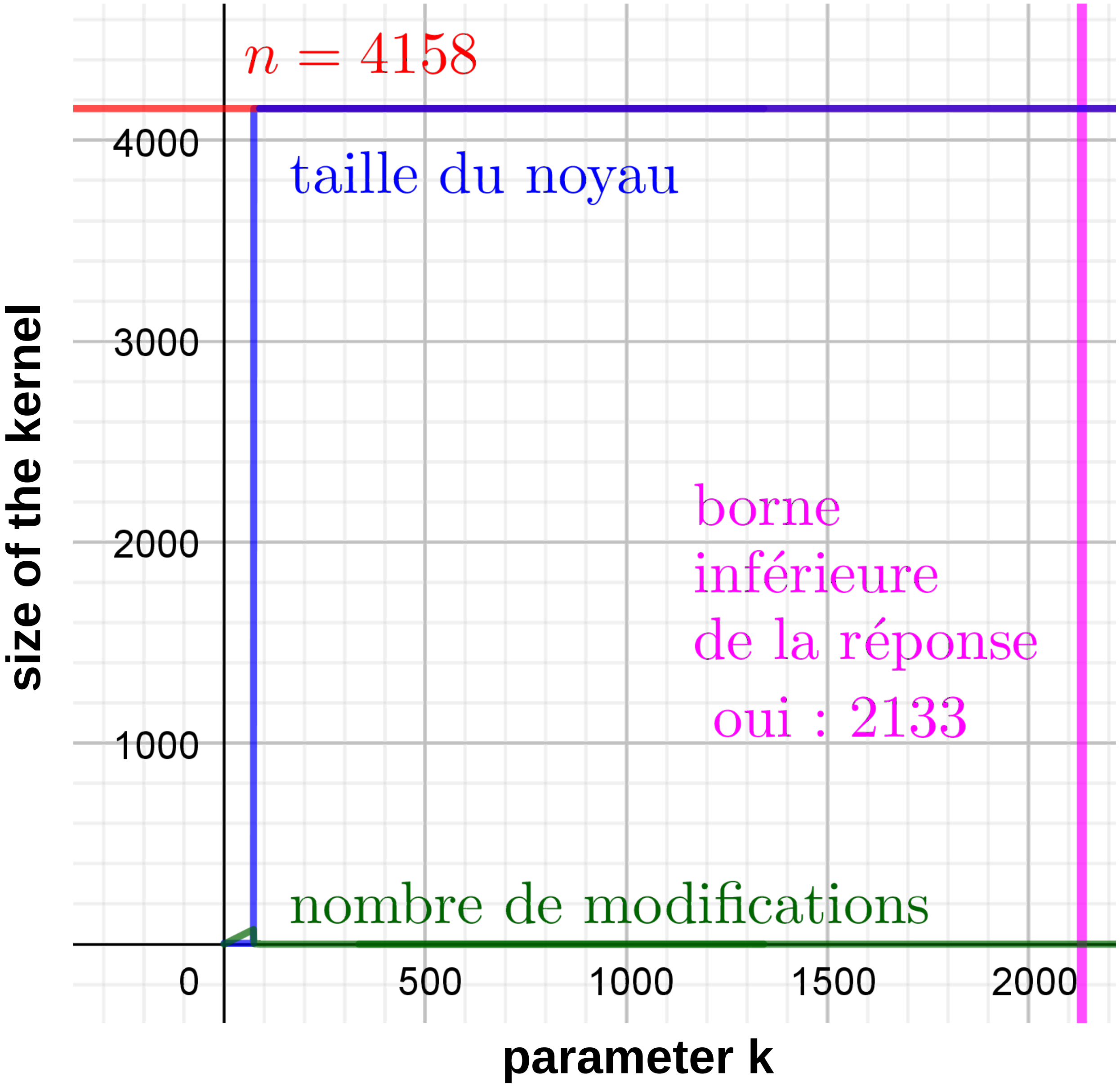
Size of reduced instance as a function of k



Size of reduced instance as a function of k



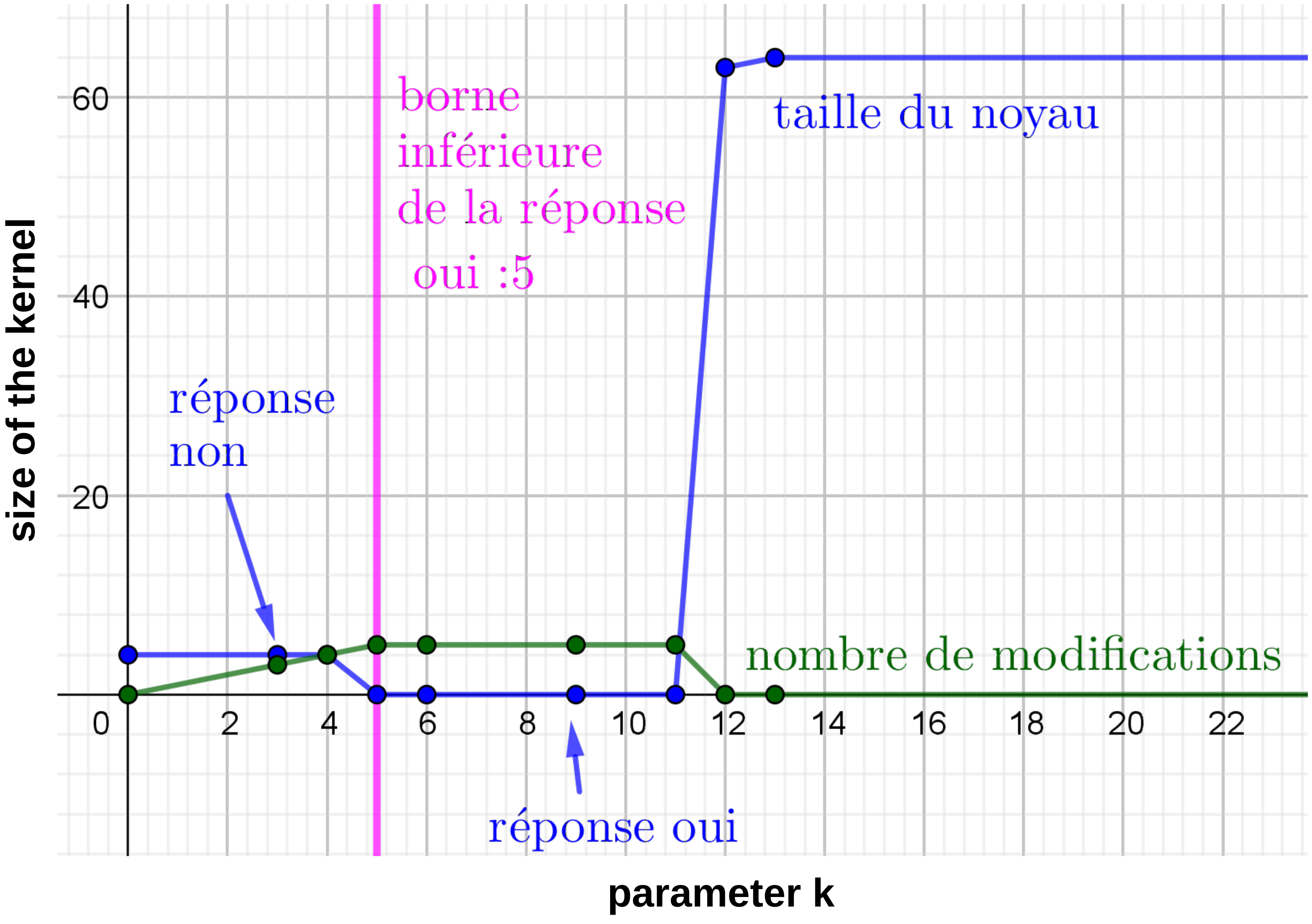
Size of reduced instance as a function of k



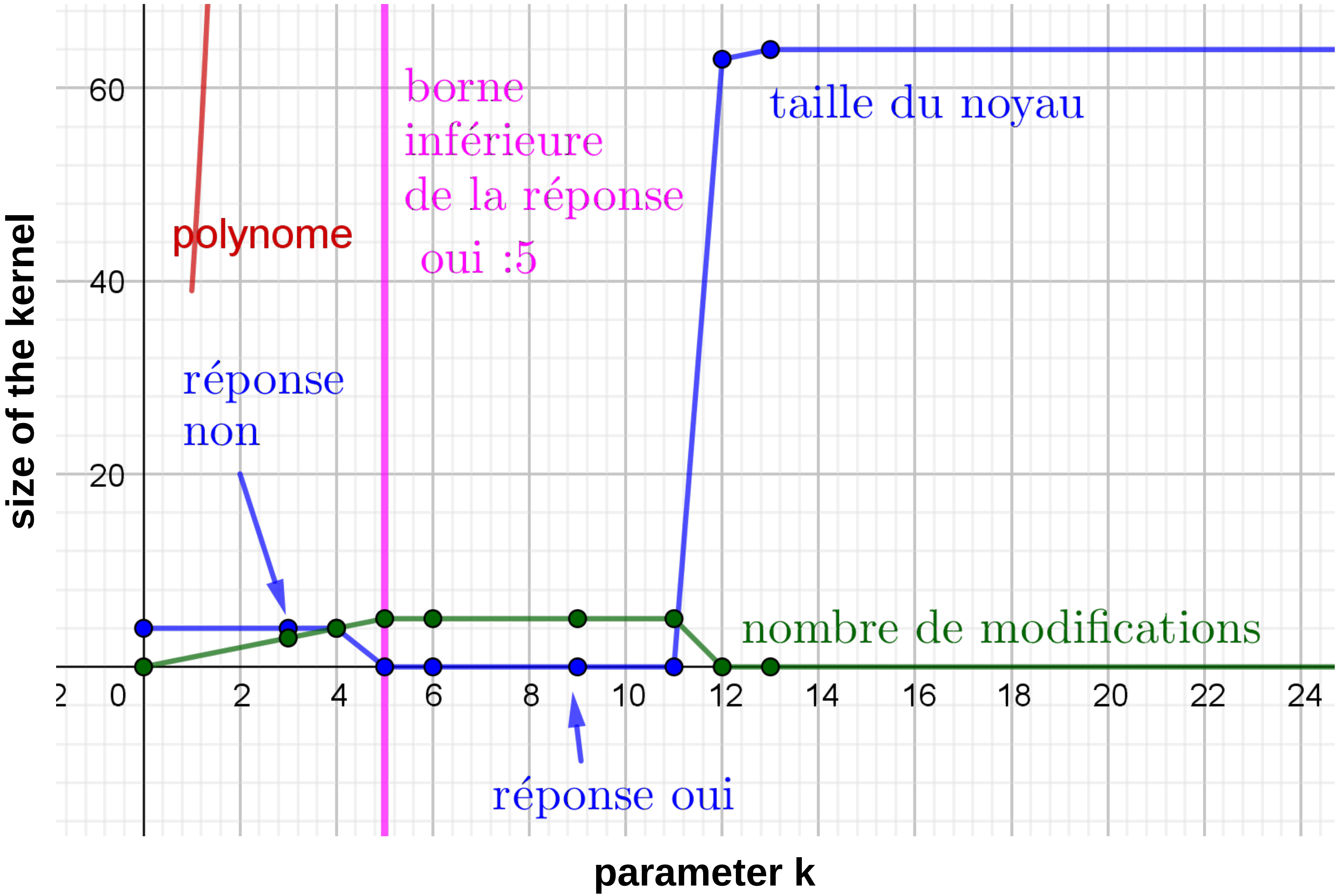
Results on real-world networks

Graphe	n	m	k_{no}: k max pour la réponse non	k max où règle 2 s'applique	k max où règle 3 s'applique	k_{ras}: k_{ras} : algo devient inefficace	k_{inf}: borne inf k_{inf}
gene_fusion	110	124	11	14	14	15	22
maayan-pdzbase	161	209	14	-	15	16	43
foodweb	183	2434	79	-	80	81	599
arenas-jazz	198	2742	85	-	86	87	698
dimacs10-netscience	379	914	19	-	23	24	118
sociopatterns-infect	410	2765	66	-	71	72	688
celegans_metabolic	453	2025	124	-	134	135	517
moreno_crime	829	1473	33	-	34	35	412
hamster-household	874	4003	153	-	158	159	1215
opsahl-ucforum	899	7019	174	-	185	186	2250
email-Eu-core	986	16064	346	-	360	361	5006
subelj_euroroad	1039	1305	11	-	11	12	341
moreno_propro	1458	1948	33	45	47	48	432
moreno_names	1707	9059	300	-	316	317	2462
figeys	2217	6418	172	-	238	239	1542
maayan-vidal	2783	6007	120	-	149	150	1658
ca-GrQC	4158	13422	73	-	88	89	2133
as2000	6474	12572	426	-	706	707	2575

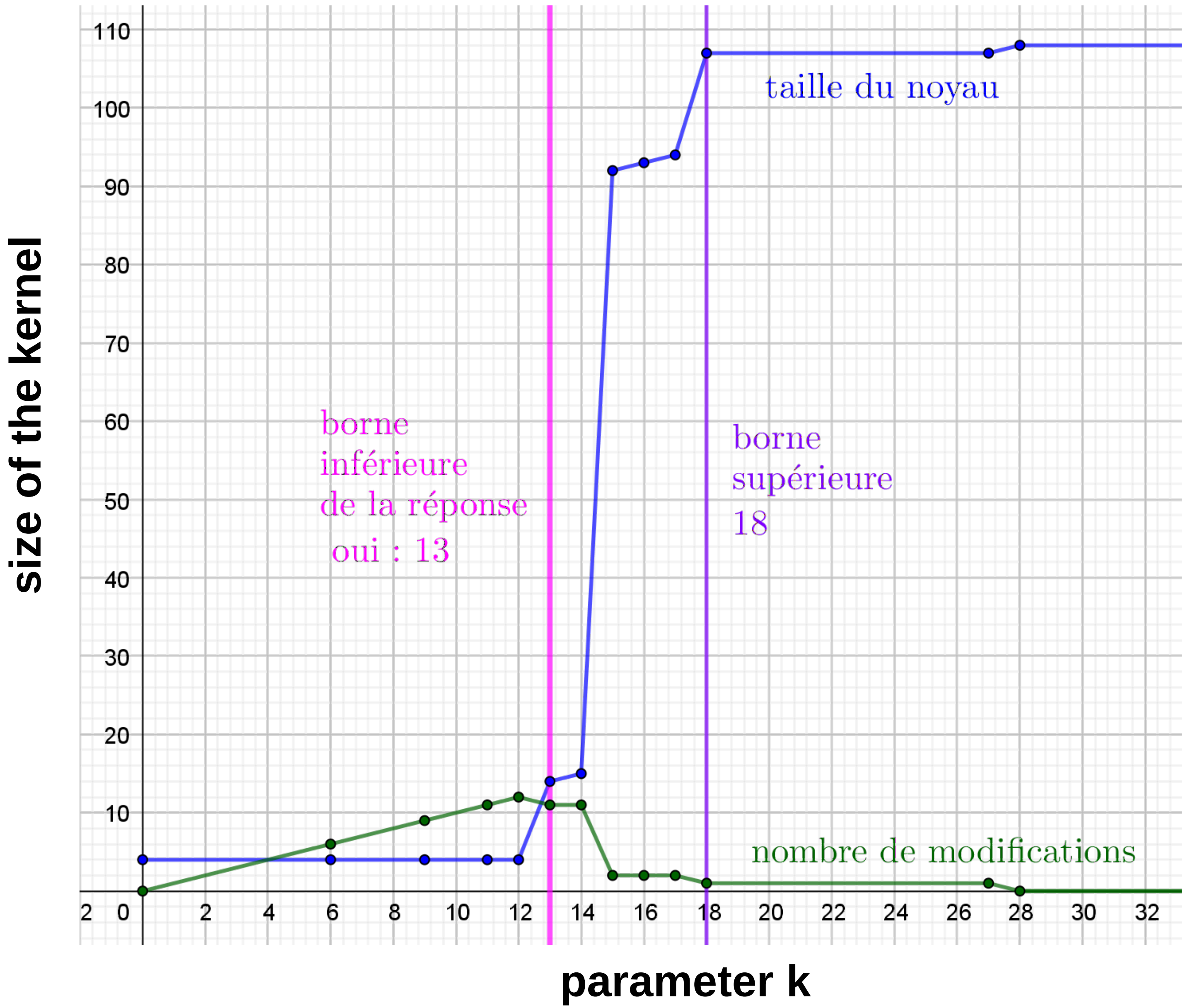
Result for an almost cograph



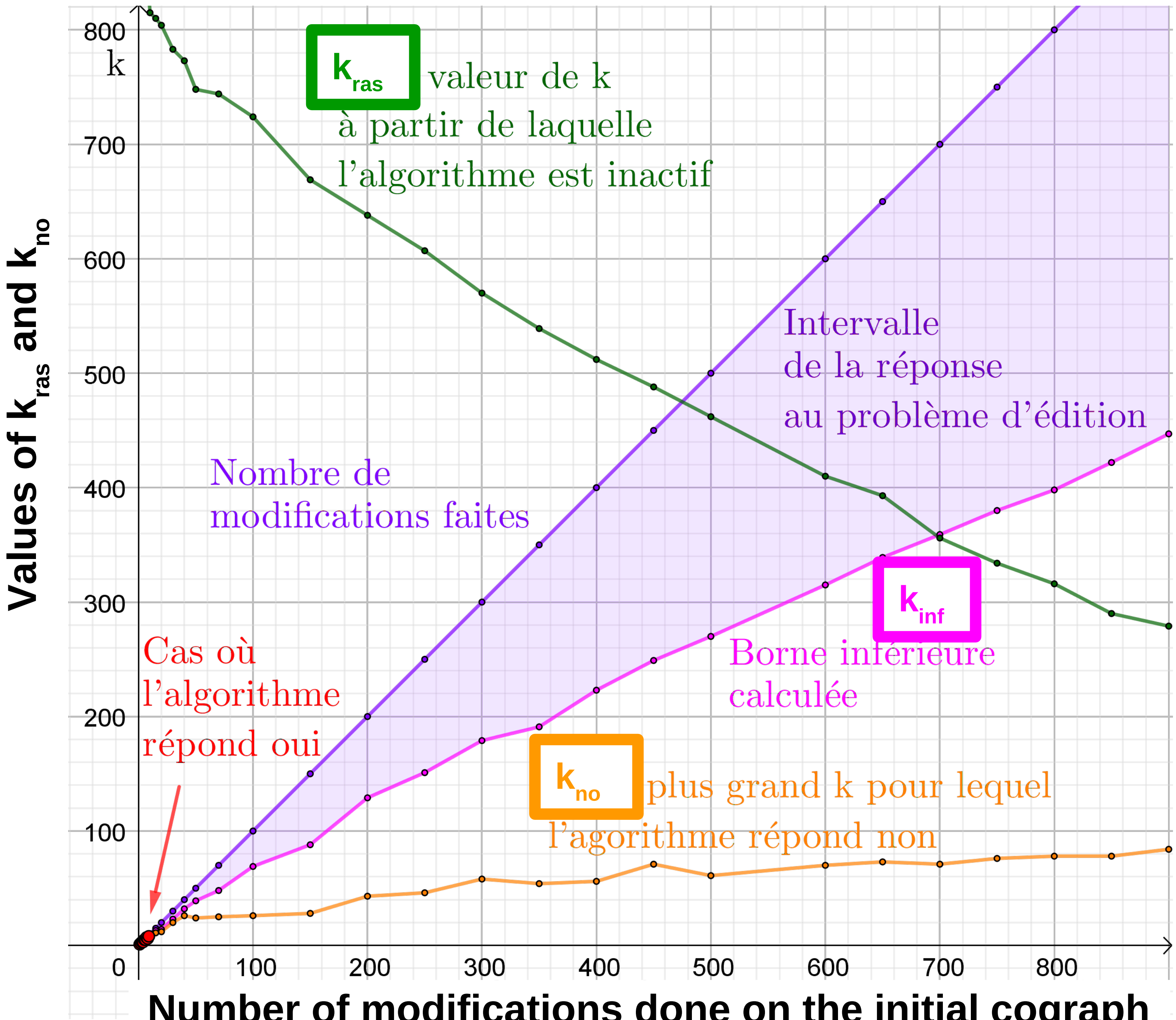
Result for an almost cograph



A less caricaturistic behaviour



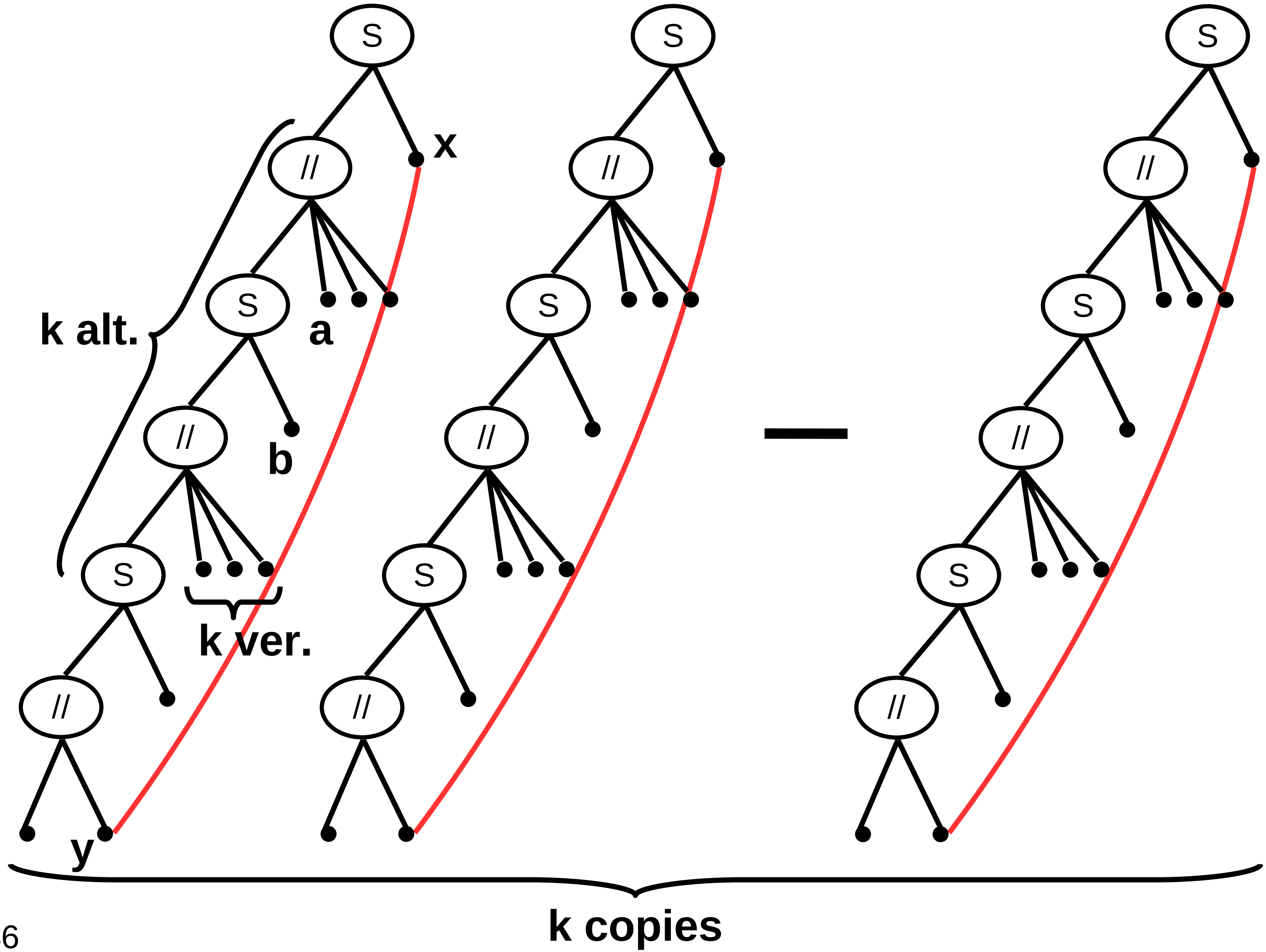
A full range of behaviours



An $O(k^2 \log k)$ Vertex kernel for cograph editing

with Remi Pellerin and Stéphan Thomassé

Guillemot et al. : O(k³) vertex



New rule : definitions

Our goal : reduce the size of the kernel to **$O(k^2 \log k)$**

New rule : definitions

<u>Our goal :</u> reduce the size of the kernel to **$O(k^2 \log k)$**

<u>Definition :</u> (t-module)

A t-module in G is a set of vertices X such that by editing a set of at most t pairs in G, we obtain G' in which X is a module.

New rule : definitions

Our goal : reduce the size of the kernel to $\mathbf{O(k^2 \log k)}$

Definition : (t-module)
A t-module in G is a set of vertices X such that by editing a set of at most t pairs in G, we obtain G' in which X is a module.

Remark : T can always be chosen in $\delta(X)$.

New rule : definitions

Our goal : reduce the size of the kernel to **$O(k^2 \log k)$**

Definition : (t-module)
A t-module in G is a set of vertices X such that by editing a set of at most t pairs in G, we obtain G' in which X is a module.

Remark : T can always be chosen in δ(X).

Definition : (budget)
The budget of a set X of G is the minimum b such that all minimum cograph editings S of G satisfy $|S \cap \delta(X)| \leq b$.

New rule : definitions

<u>Our goal :</u> reduce the size of the kernel to **O(k^2 log k)**

<u>Definition :</u> (t-module)
A t-module in G is a set of vertices X such that by editing a set of at most t pairs in G, we obtain G' in which X is a module.

*<u>Remark :</u>*T can always be chosen in δ(X).

<u>Definition :</u> (budget)
The budget of a set X of G is the minimum b such that all minimum cograph editings S of G satisfy $|S \cap \delta(X)| \leq b$.

<u>Lemma :</u>
Let X be a t-module such that $|X| > k + t$. If there exists an editing of size at most k, then the budget of X is at most t.

New rule : definitions

Lemma :

Let X be a t-module such that $|X| > k + t$. If there exists an editing of size at most k, then the budget of X is at most t.

Exercise : Prove the lemma above.

New rule : definitions

Lemma :

Let X be a t-module such that |X| > k + t. If there exists an editing of size at most k, then the budget of X is at most t.

Exercise : Prove the lemma above.

Exercise : Prove that testing if X is a t-module can be done in polynomial time.

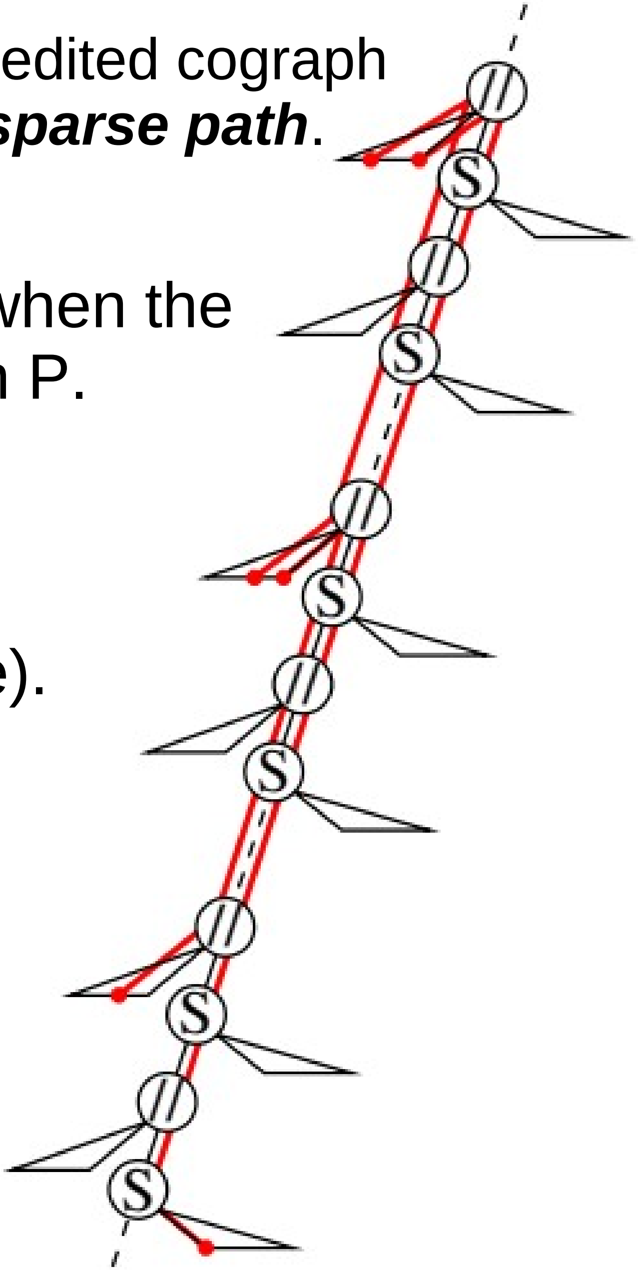
New rule : the main idea

- **<u>Purpose:</u>**

Avoid long paths (≥51.ℓ) in the cotree T of the edited cograph that ***interact*** with only few (ℓ) edited pairs: ***51-sparse path***.

<u>Definition :</u> (interact)
The edited pair xy *interacts* with path P when the path from x to y in T shares an edge with P.

<u>Lemma :</u>
If T has a 51-sparse path then the nested t-module reduction rule applies (our 4th rule).



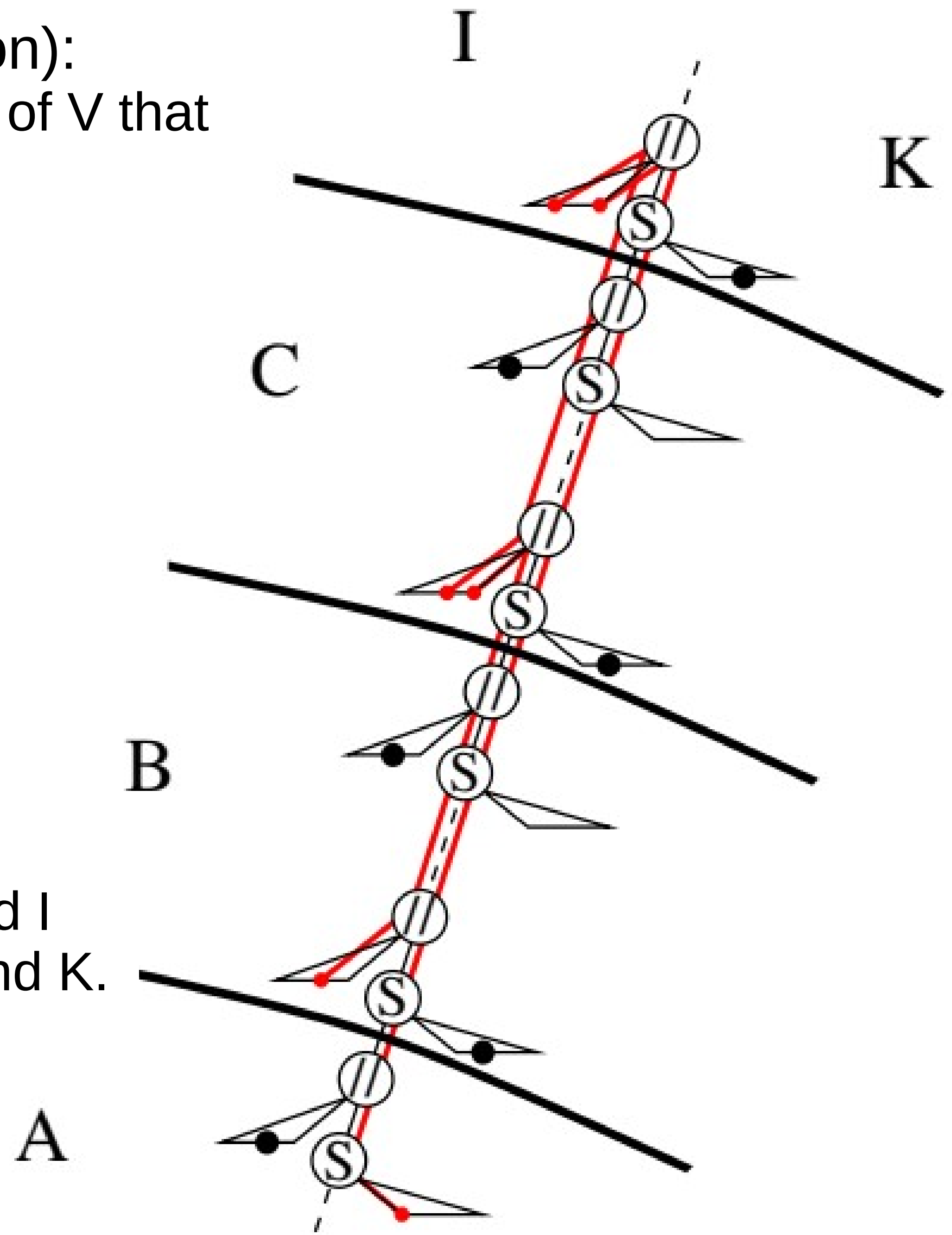
New rule : the main idea

- <u>Rule 4</u> (nested t-module reduction):

If there exists a partition A⊔B⊔C⊔I⊔K of V that satisfies the following conditions:

- A, A⊔B, A⊔B⊔C are t-modules
- |A|>k+t
- B_S, $B_{//}$, C_S, C// all have size >3t
- B_S and $B_{//}$ have the required adjacencies with A, I, K
- C_S and $C_{//}$ have the required adjacencies with A, B, I, K

Then remove all edges between A and I and add missing edges between A and K.

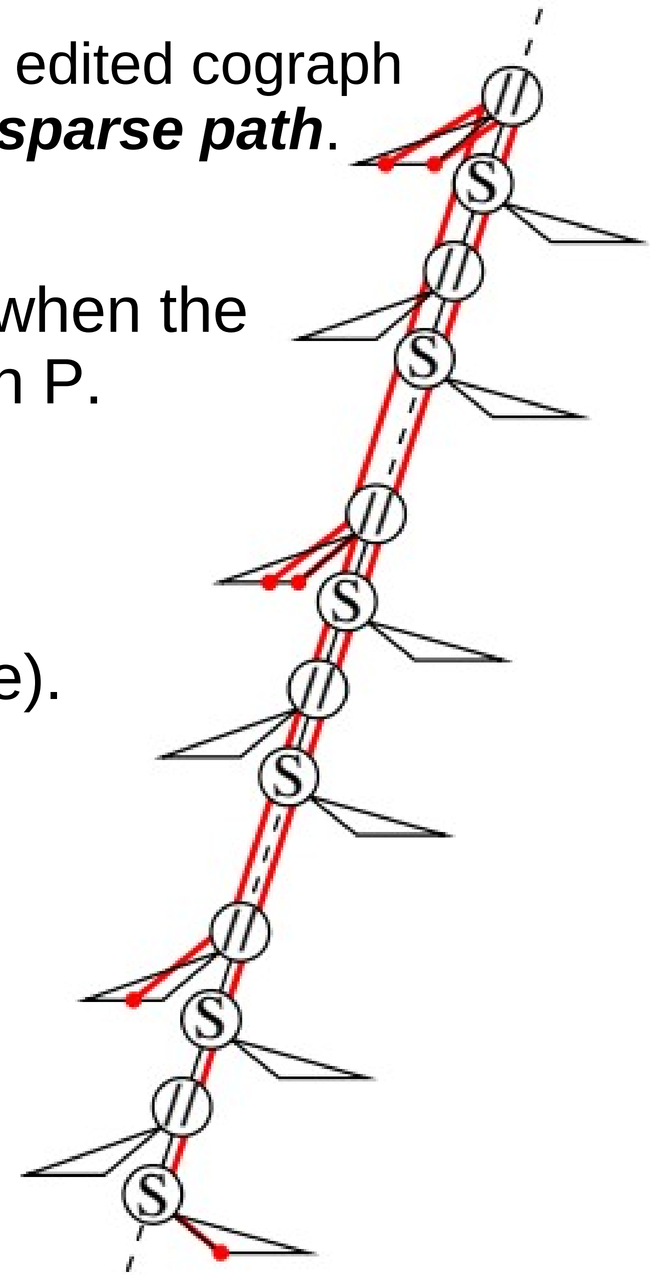


New rule : the main idea

- **<u>Purpose:</u>**

Avoid long paths (≥51.ℓ) in the cotree T of the edited cograph that ***interact*** with only few (ℓ) edited pairs: ***51-sparse path***.

<u>Definition :</u> (interact)
The edited pair xy *interacts* with path P when the path from x to y in T shares an edge with P.



<u>Lemma :</u>
If T has a 51-sparse path then the nested t-module reduction rule applies (our 4^{th} rule).

<u>Lemma :</u>
If the reduced graph H has $\Omega(k^2 \log k)$ vertices then its cotree has size $\Omega(k \log k)$ and if H is a yes-instance then T has a 51 sparse path.

Perspectives (Lecture I)

- $O(k^2)$ kernel for cograph editing?

- Reduction rules without knowing the value of the parameter k

- Kernels or FPT algorithms for edge modification problems with other (smaller) parameters
  - Local search?

Graph editing: algorithms and experimental results

Christophe Crespelle

Université Côte d'Azur

with Jean Blair, Anne-Aymone Bourguin, Benjamin Gras,
Daniel Lokshtanov, Remi Pellerin, Anthony Perez, Thi Ha Duong Phan,
Eric Thierry and Stéphan Thomassé