

Parameterized Algorithms

Summary

Thomas Bläsius

General notes

Goal of the exam

- we need to certify that you
 - know the content of the lecture
 - understand it
 - can use the learned techniques to design and analyze new algorithms

General notes

Goal of the exam

- we need to certify that you
 - know the content of the lecture
 - understand it
 - can use the learned techniques to design and analyze new algorithms
- oral exam: easiest to check first two
- focus on the second (which implies the first)
- I'll make an effort to also check the third

General notes

Goal of the exam

- we need to certify that you
 - know the content of the lecture
 - understand it
 - can use the learned techniques to design and analyze new algorithms
- oral exam: easiest to check first two
- focus on the second (which implies the first)
- I'll make an effort to also check the third

Preparation

- step 1: recap and understand the content

General notes

Goal of the exam

- we need to certify that you
 - know the content of the lecture
 - understand it
 - can use the learned techniques to design and analyze new algorithms
- oral exam: easiest to check first two
- focus on the second (which implies the first)
- I'll make an effort to also check the third

Preparation

- step 1: recap and understand the content
- step 2: explain the content to someone

General notes

Goal of the exam

- we need to certify that you
 - know the content of the lecture
 - understand it
 - can use the learned techniques to design and analyze new algorithms
- oral exam: easiest to check first two
- focus on the second (which implies the first)
- I'll make an effort to also check the third

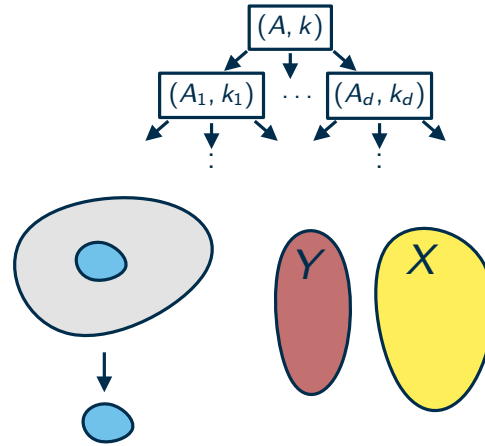
Preparation

- step 1: recap and understand the content
- step 2: explain the content to someone
- **important:** Step 2 is exactly what happens in the exam. You should train that!

Content

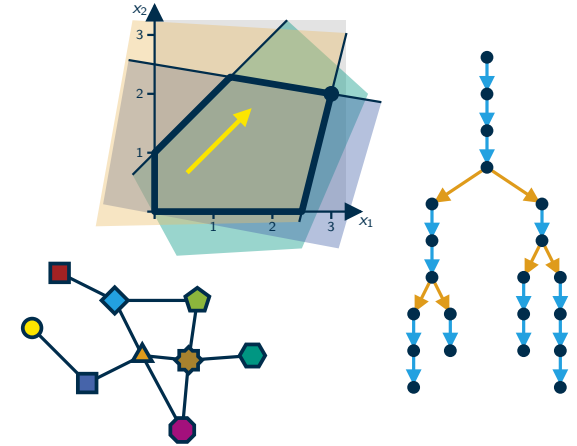
Basic toolbox

- bounded search trees
- kernelization
- iterative compression



Extended toolbox

- linear programs
- branch-and-reduce
- color coding



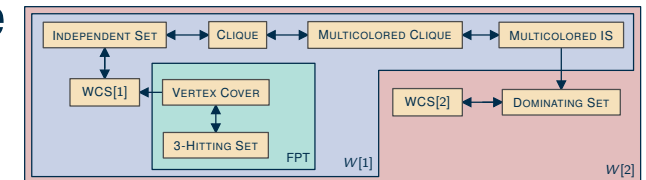
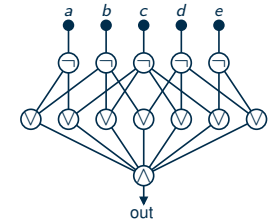
Tree width

- dynamic programming
- chordal and planar graphs
- Courcelle's theorem



Lower bounds

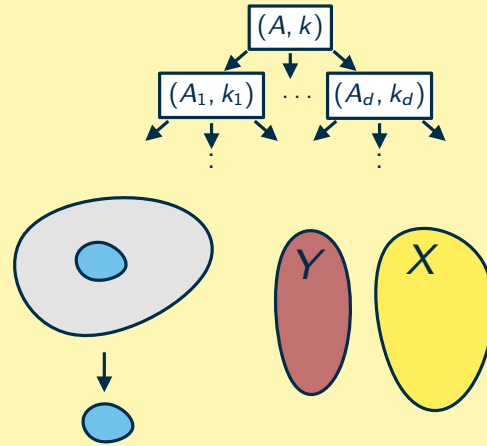
- kernel lower bounds
- parameterized reductions
- circuits and the W-hierarchy
- ETH and SETH



Content

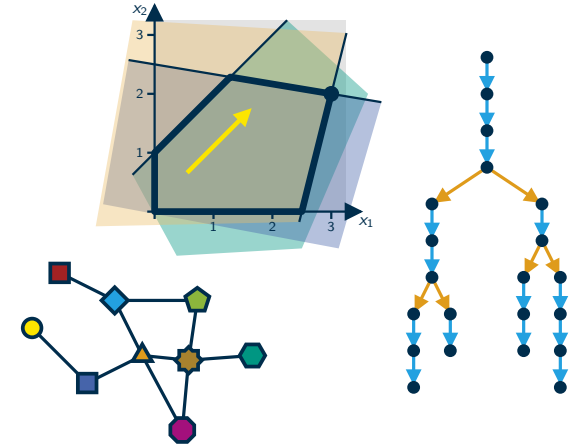
Basic toolbox

- bounded search trees
- kernelization
- iterative compression



Extended toolbox

- linear programs
- branch-and-reduce
- color coding



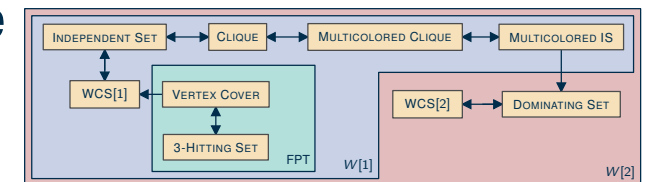
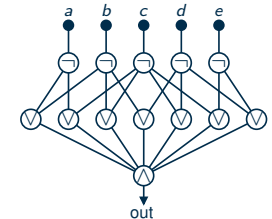
Tree width

- dynamic programming
- chordal and planar graphs
- Courcelle's theorem



Lower bounds

- kernel lower bounds
- parameterized reductions
- circuits and the W-hierarchy
- ETH and SETH



FPT basic techniques

basic terms: parameterization

- What is a parameterized problem?
- What is FPT?

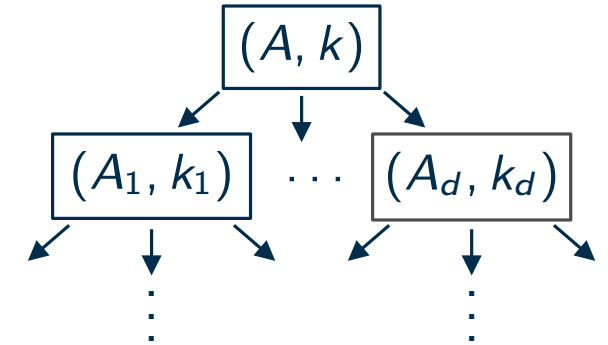
Kernelization

- What is a kernel?
- How do we compute it?
- When is kernelization possible?

How can we apply these techniques to VERTEX COVER?

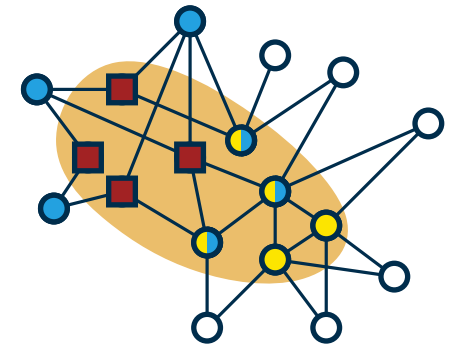
Search trees

- How do search trees work?
- How do we get FPT time?
- How do we usually bound the height?



Iterative compression

- What is the core idea of this technique?
- Where do we get the too-large solution from?
- What is the compression problem?



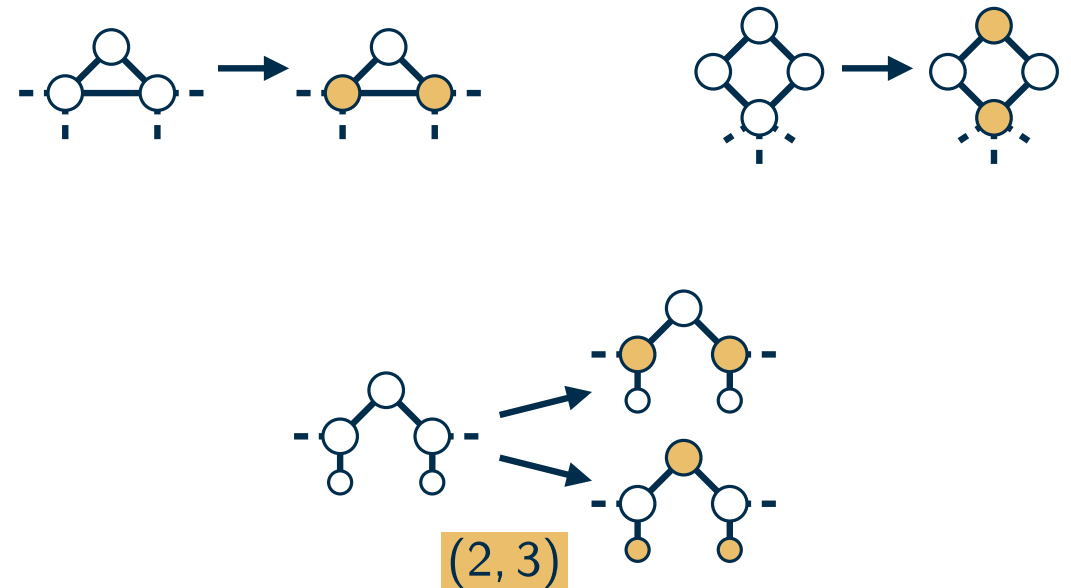
Bounded search trees

Branching in general

- What is a branching vector?
- How do we get the tree size given the branching vector?
- Why does this boil down to computing the root of a polynomial?

Better branching for VERTEX COVER

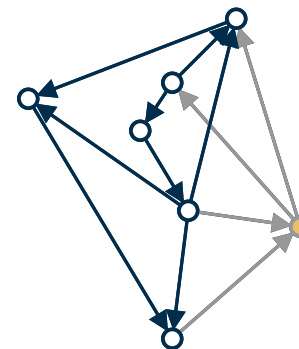
- How do vertices with high degree help?
- Is there always a vertex with degree 4?
- How do we get rid of degree-2 vertices?
- What is the idea for degree-3 vertices?
- What about degree-4 vertices?
- Why can I ignore the branching vector $(1, 4)$?



Iterative compression

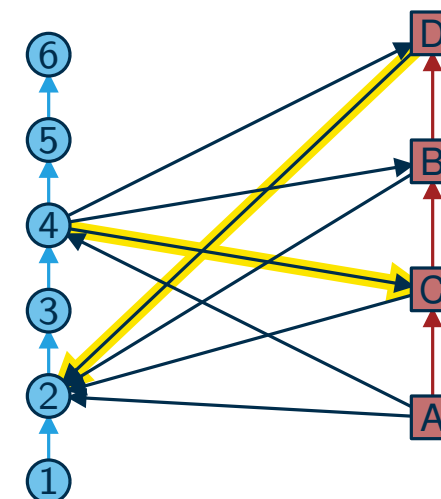
FEEDBACK VERTEX SET

- What is the problem?
- How do we reduce FVS to FVS COMPRESSION?
- How do we reduce FVS COMPRESSION to DISJOINT FVS?
- What running time costs do you accumulate with this?



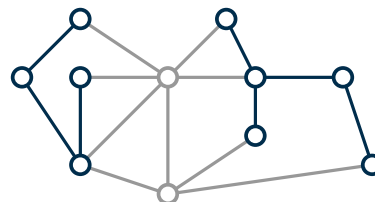
DISJOINT FEEDBACK VERTEX SET on tournament graphs

- What are tournament graphs?
- How does it reduce to LONGEST INCREASING SUBSEQUENCE?



Undirected FVS

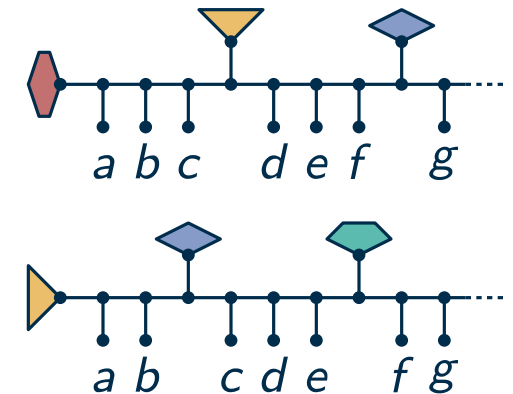
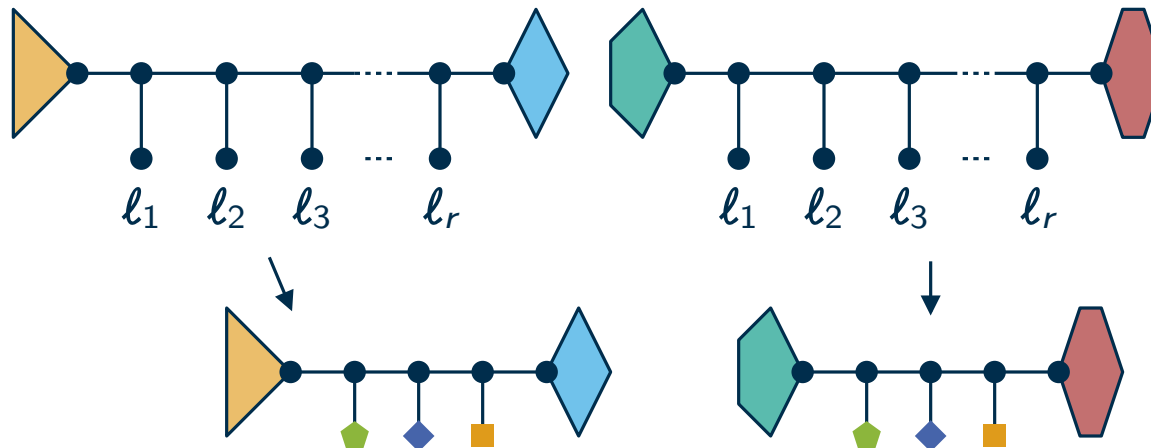
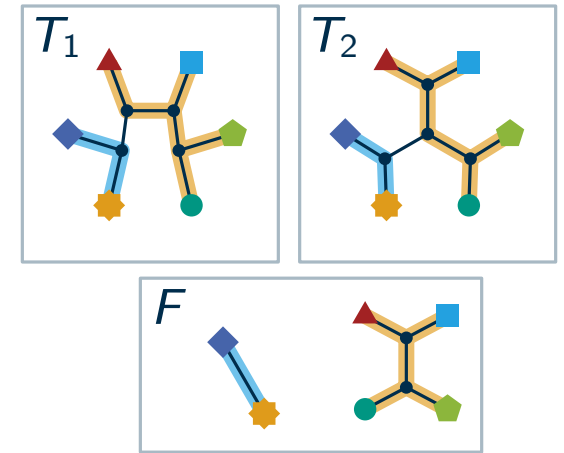
- Reduction to the disjoint variant?
- How do we solve the disjoint variant?



Kernelization: similar trees

MAXIMUM AGREEMENT FOREST

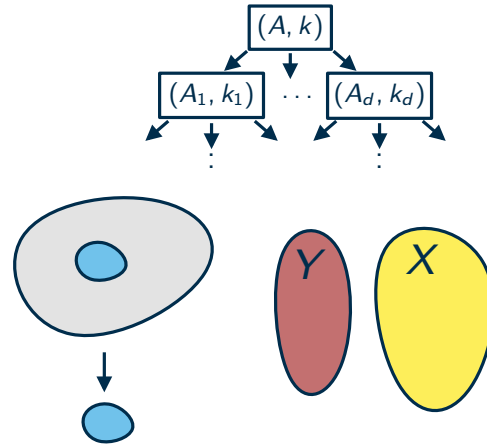
- What is the problem?
- Why would it help if every tree in the solution has few leaves?
- What counter examples for this?
- How do we get rid of them?
- How does the amortized perspective help?



Content

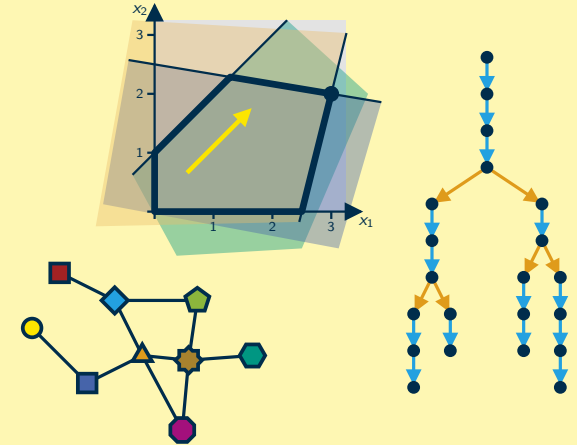
Basic toolbox

- bounded search trees
- kernelization
- iterative compression



Extended toolbox

- linear programs
- branch-and-reduce
- color coding



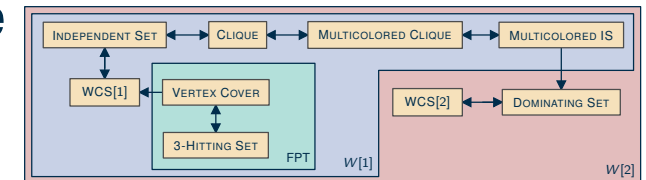
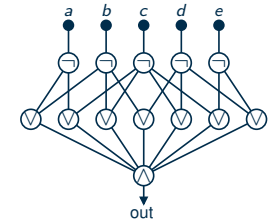
Tree width

- dynamic programming
- chordal and planar graphs
- Courcelle's theorem



Lower bounds

- kernel lower bounds
- parameterized reductions
- circuits and the W-hierarchy
- ETH and SETH



Linear programs (1)

Linear programs in general

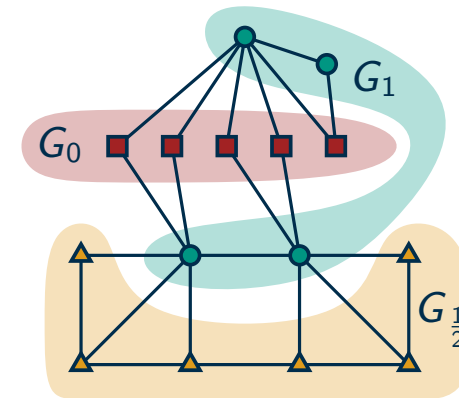
- How to deal with absolute values?
- What is the dual program linear program?

$$\begin{array}{l} \text{maximize: } 2x_1 + 3x_2 \\ \text{such that: } \end{array}$$
$$\begin{array}{l} 2x_1 + 3x_2 \leq 4x_1 + 8x_2 \leq 12 \\ 2x_1 + 3x_2 \leq \frac{4x_1 + 8x_2}{2} \leq \frac{12}{2} \\ 2x_1 + 3x_2 \leq \frac{4x_1 + 8x_2}{3} + \frac{2x_1 + 1x_2}{3} \leq \frac{12}{3} + \frac{3}{3} \end{array}$$
$$\begin{array}{l} 4x_1 + 8x_2 \leq 12 \\ 2x_1 + 1x_2 \leq 3 \\ 3x_1 + 2x_2 \leq 4 \\ x_1, x_2 \geq 0 \end{array}$$

Diagram illustrating the derivation of the dual program from the primal program. The primal program is shown on the left, and the dual program is shown on the right. The dual program is derived by combining the primal constraints using the multipliers $\frac{1}{2}$, $\frac{1}{3}$, and $\frac{1}{3}$.

VERTEX COVER and LPs

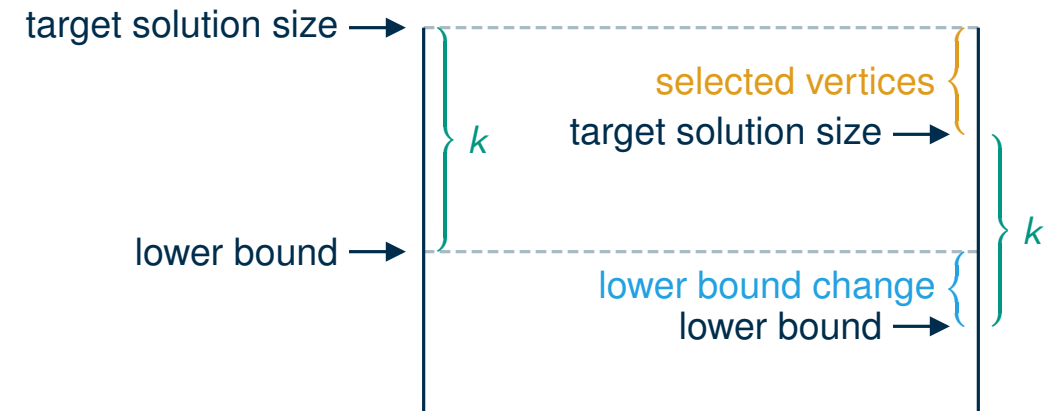
- What is special about the LP relaxation of the VC ILP?
- How can this help for kernelization?
- Do we actually need to solve an LP?



Linear programs (2)

Parameterization above lower bound

- What is it?
- Where do we have to be more careful than usual?
- How does it work for VERTEX COVER?



Lenstra's theorem

- What does it say?
- How can we use it to solve CLOSETS STRING?

s_1 A B C F G D C G E B A
 s_2 B B D A G D B G B E C
 s_3 B A E F C A C G B B F

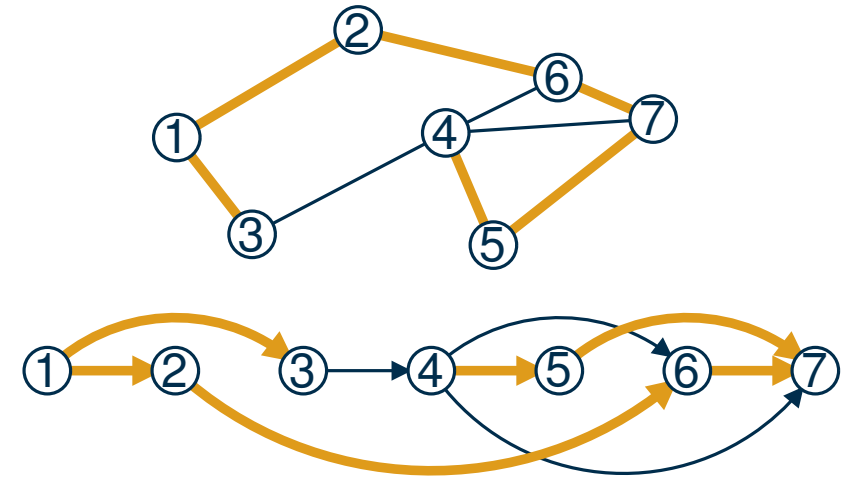
↓

s_1	1	1	1	1	1	1	1	1	1	1
s_2	2	1	2	2	1	1	2	1	2	2
s_3	2	2	3	1	2	2	1	1	2	1

Color Coding

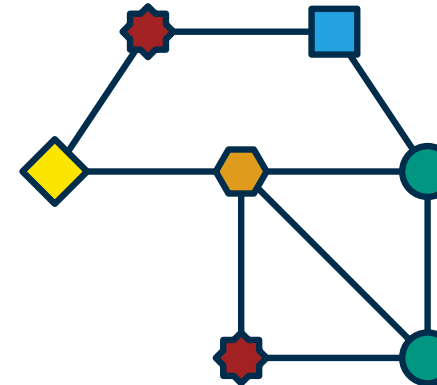
LONGEST PATH: random order

- What additional structure did we guess?
- How does this help to solve the problem?
- What do we need to show for the success probability to get FPT running time?



Color Coding

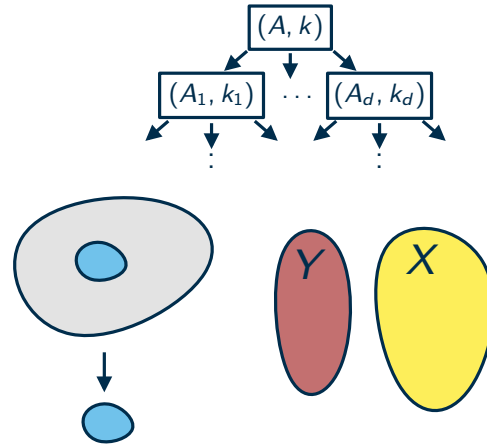
- What additional structure do we guess here?
- How does this help?
- Can we derandomize this?
- What are perfect hash families?
- What are universal sets?



Content

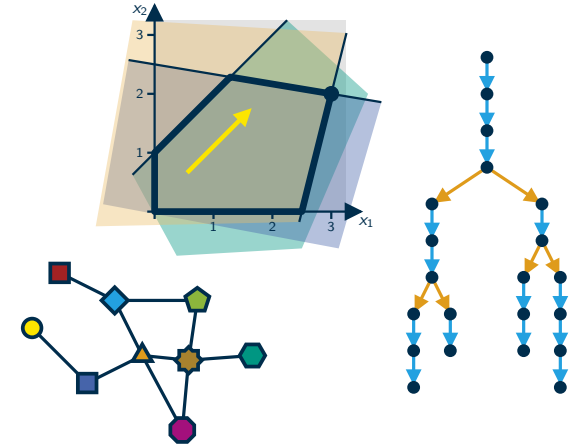
Basic toolbox

- bounded search trees
- kernelization
- iterative compression



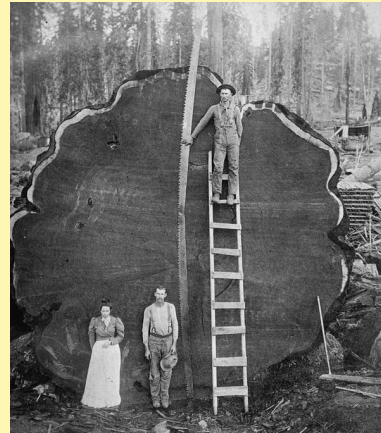
Extended toolbox

- linear programs
- branch-and-reduce
- color coding



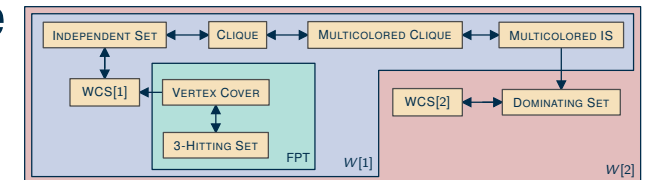
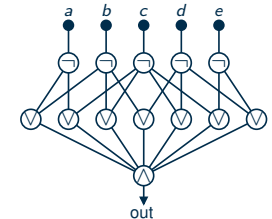
Tree width

- dynamic programming
- chordal and planar graphs
- Courcelle's theorem



Lower bounds

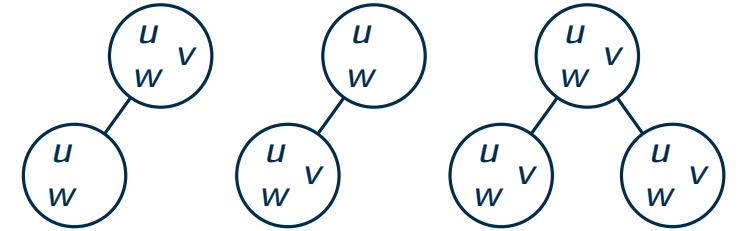
- kernel lower bounds
- parameterized reductions
- circuits and the W-hierarchy
- ETH and SETH



Treewidth

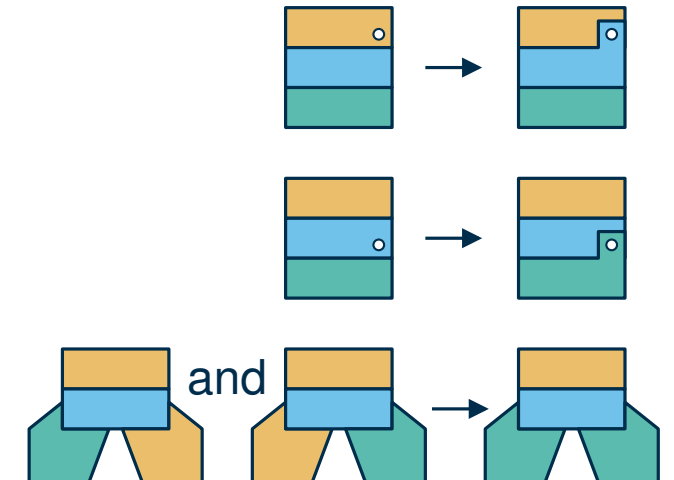
Path and tree decomposition

- What are path and tree decompositions?
- What are pathwidth and treewidth?
- When is a tree decomposition nice?
- Why can we assume to have a nice tree decomposition?



Dynamic programming

- How can we solve problems via DP on tree decompositions?
- examples: INDEPENDENT SET and HAMILTONIAN CIRCUIT
- How do we keep track of global requirements?
- Can you transfer this to other problems?



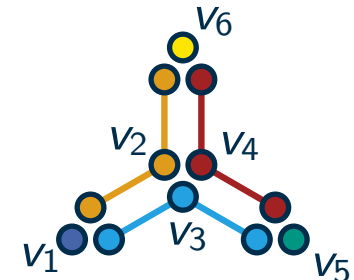
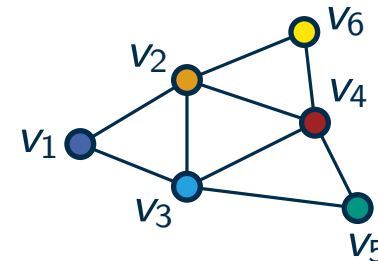
Courcelle & chordal graphs

Courcelle's theorem

- What can we do with MSO_2 ?
- How can we model connectedness?
- How to model HAMILTONIAN CIRCUIT and VERTEX COVER?
- How about other problems?
- What does Courcelle's theorem tell us?
- Why do we need the optimization variant?

Chordal graphs

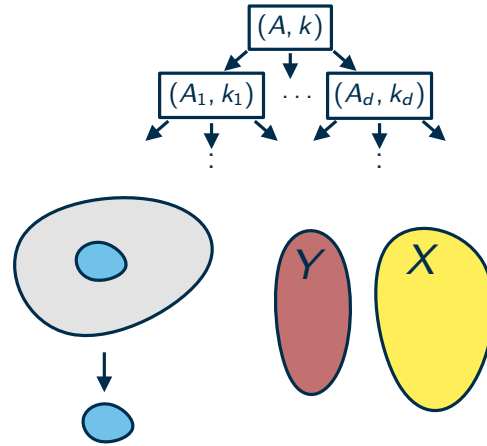
- What are interval graphs and chordal graphs?
- What is the interval/chordal width?
- What is the relation to pathwidth and treewidth?



Content

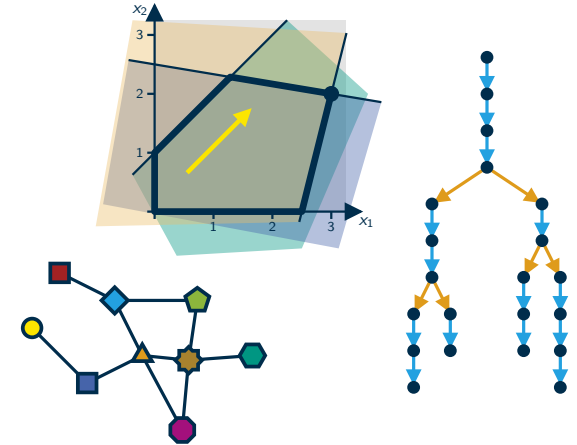
Basic toolbox

- bounded search trees
- kernelization
- iterative compression



Extended toolbox

- linear programs
- branch-and-reduce
- color coding



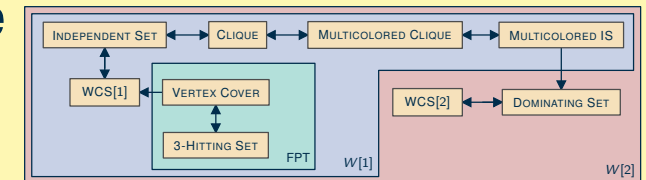
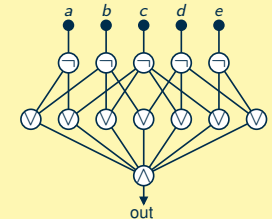
Tree width

- dynamic programming
- chordal and planar graphs
- Courcelle's theorem



Lower bounds

- kernel lower bounds
- parameterized reductions
- circuits and the W-hierarchy
- ETH and SETH



Kernel lower bounds

OR-Distillation

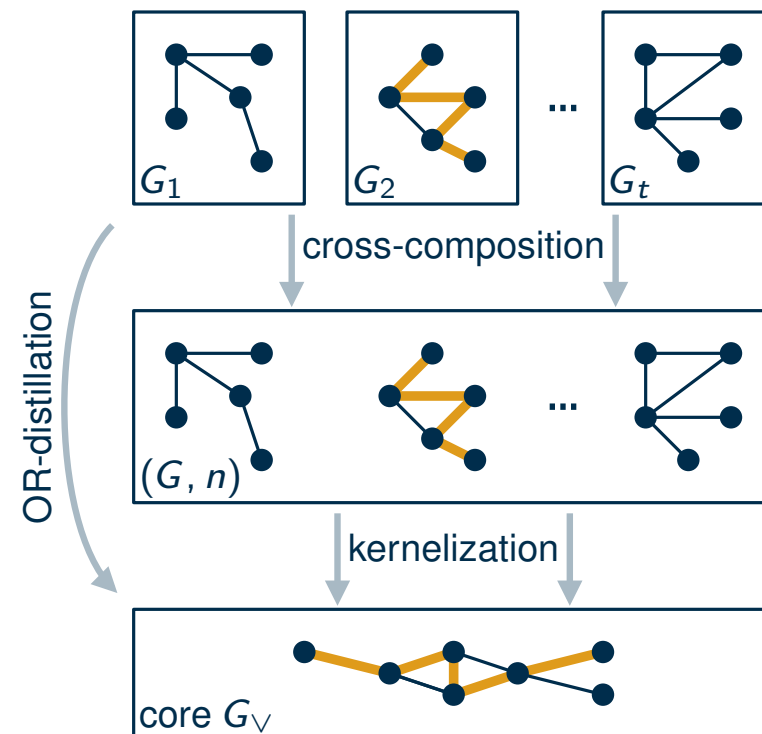
- What is an OR-distillation?
- How would a polynomial kernel for LONGEST PATH give an OR-distillation from HAMILTONIAN PATH into LONGEST PATH?

coNP/poly

- What is coNP/poly?
- What does polynomial advice mean?
- How can we show that a problem is in coNP/poly?

Kernel lower bounds

- Why would an OR-distillation of HAMP into LONGP imply $\text{HAMP} \in \text{coNP/poly}$?
- What are the no-certificates? Advice strings? How does the OR-distillation help?
- How can we get an OR-distillation of SET SPLITTING into itself?



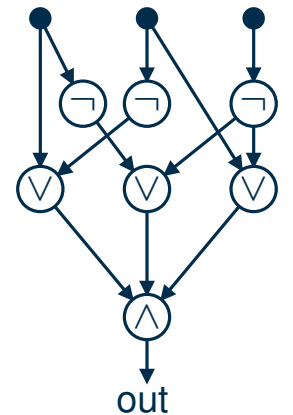
W-Hierarchie

Parameterized reductions

- What is it and why is it defined this way?
- How to reduce between MULTICOLORED CLIQUE and CLIQUE?
- How to do these reductions?
 - MULTICOLORED IS $\rightarrow$ DOMINATING SET
 - INDEPENDENT SET $\rightarrow$ WEIGHTED CIRCUIT SATISFIABILITY
 - DOMINATING SET $\rightarrow$ WEIGHTED CIRCUIT SATISFIABILITY

W-Hierarchy

- What is WCS? How does it define the classes of the W-hierarchy?
- How do we show containment/hardness for $W[t]$?
- What are complete problems $W[1]$ and $W[2]$?
- Why is FPT a subset of $W[1]$?



Relational databases

Normalized circuits

- What are they and why are they useful?

UNIQUE

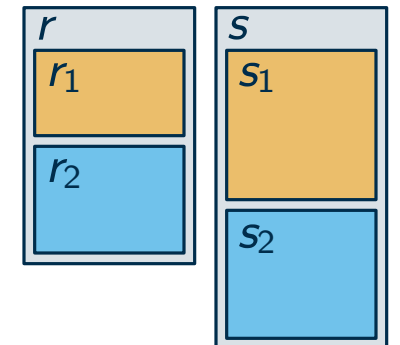
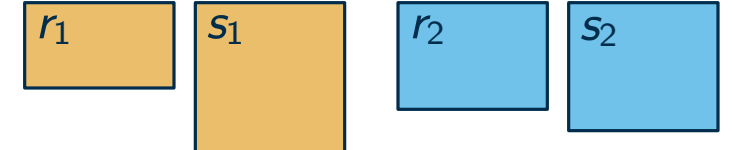
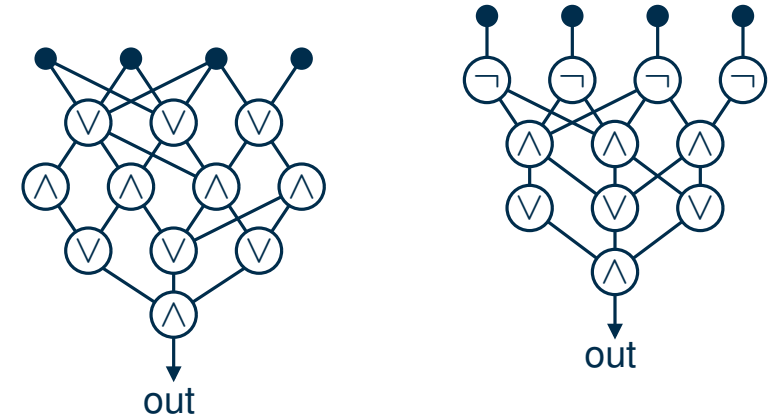
- How can we reduce from and to HITTING SET?

FD

- How can we reduce this to WCS[2]?

IND_{FIXED}

- How can we interpret instances as Boolean functions?
- How can we and them, and why does that help?
- How can we model disjunctions of conjunctions of negated literals?
- How can we reduce to a Boolean circuit?



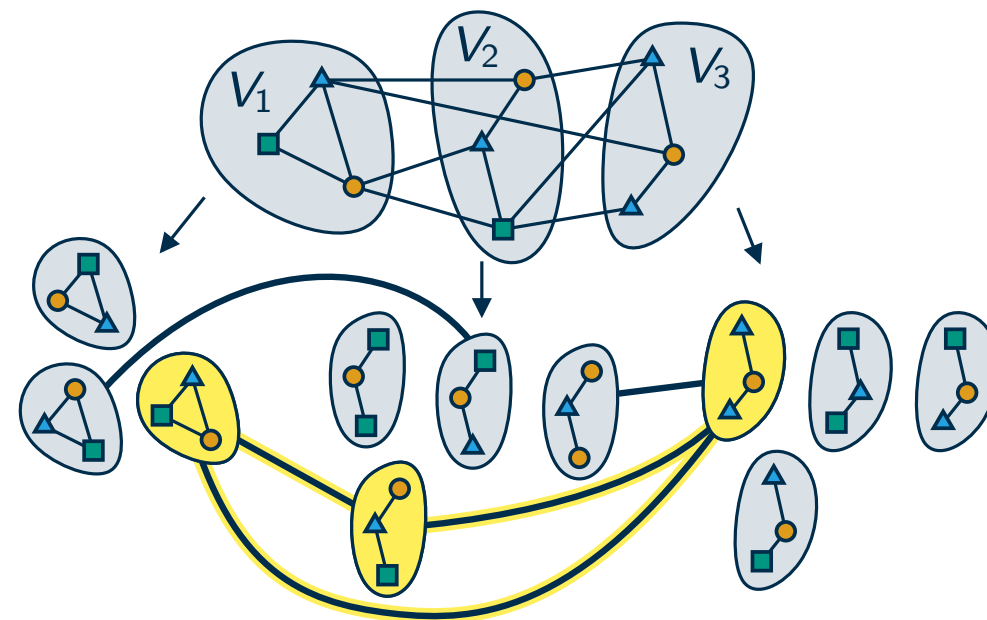
ETH and SETH

Basics

- What is ETH? What is SETH? How do they relate?
- How do we get lower bounds for other problems?
- Is a polynomial reduction enough?

So many reductions

- How do we get a lower bound for CLIQUE?
- What does that mean for $W[1]$ and FPT?
- Do we need to be careful with the parameter?
- What is GRID TILING?
- How to reduce CLIQUE to GRID TILING?
- GRID TILING to PLANAR LIST COLORING?
- GRID TILING to UNIT DISK INDEPENDENT SET?



Ad break – winter 26/27

Algorithms courses offered by us

- lecture: computational geometry
- practical course: route planning
- practical course: beating the worst-case

(we have a big room and two slots for exercises right from the start this time)

Ad break – winter 26/27

Algorithms courses offered by us

- lecture: computational geometry
- practical course: route planning
- practical course: beating the worst-case

(we have a big room and two slots for exercises right from the start this time)

Courses from other algorithms groups to check out

- Marvin Künnemann
- Henning Meyerhenke
- Peter Sanders
- Torsten Ueckerdt

Ad break – winter 26/27

Algorithms courses offered by us

- lecture: computational geometry
- practical course: route planning
- practical course: beating the worst-case

(we have a big room and two slots for exercises right from the start this time)

Courses from other algorithms groups to check out

- Marvin Künnemann
- Henning Meyerhenke
- Peter Sanders
- Torsten Ueckerdt

Also

- scientific figures and presentations in ipe

Scientific Figures and Presentations in Ipe



What?

- Learn how to use Ipe to create:
 - nice figures for thesis or scientific papers
 - presentations (like this one)
- Learn how to customize Ipe to your likings
- Learn some basic design principles

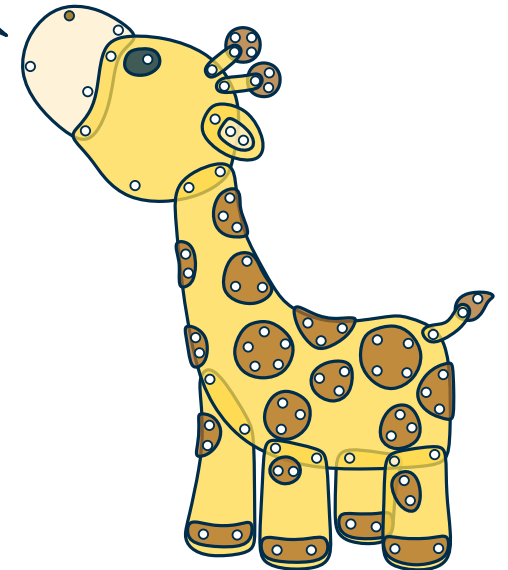
Limited capacity, apply now!

When?

- Winter semester 2026/27
- Regular meetings and exercises up to Christmas
- One final submission and presentation at the end of the lecture period

Why?

- You are tired of creating presentations with Latex Beamer or PowerPoint, or
- You have already used Ipe and want to learn some cool tricks, and
- 2 ECTS Interdisciplinary Qualifications (Schlüsselqualifikation)



Evaluation

Things to mention / discuss

- many very nice comments :-)

Evaluation

Things to mention / discuss

- many very nice comments :-)
- different activities in the exercise sessions: allocation of time

Evaluation

Things to mention / discuss

- many very nice comments :-)
- different activities in the exercise sessions: allocation of time
- script, references to book chapter, paper reading in active session

Evaluation

Things to mention / discuss

- many very nice comments :-)
- different activities in the exercise sessions: allocation of time
- script, references to book chapter, paper reading in active session
- video recordings

Evaluation

Things to mention / discuss

- many very nice comments :-)
- different activities in the exercise sessions: allocation of time
- script, references to book chapter, paper reading in active session
- video recordings
- Ilias vs. Discord, Homepage

Evaluation

Things to mention / discuss

- many very nice comments :-)
- different activities in the exercise sessions: allocation of time
- script, references to book chapter, paper reading in active session
- video recordings
- Ilias vs. Discord, Homepage
- finding a team

Evaluation

Things to mention / discuss

- many very nice comments :-)
- different activities in the exercise sessions: allocation of time
- script, references to book chapter, paper reading in active session
- video recordings
- Ilias vs. Discord, Homepage
- finding a team
- purpose of parameterized algorithms: purely theoretical vs. practically efficient algorithms
- i wish there'd be more connection to practice and real-world examples me too buddy, me too :-)

Good Luck!