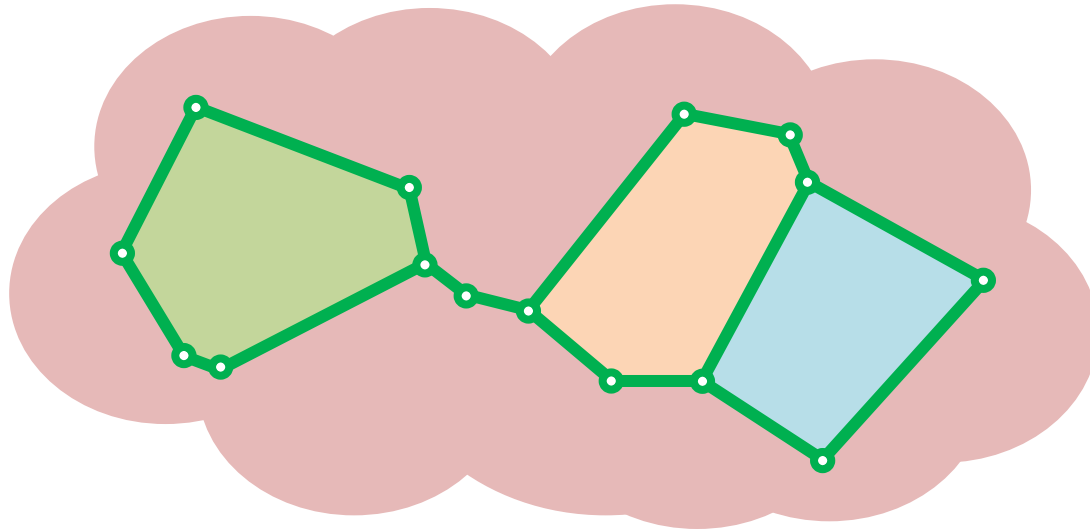


Algoritmer



Gerth Stølting Brodal

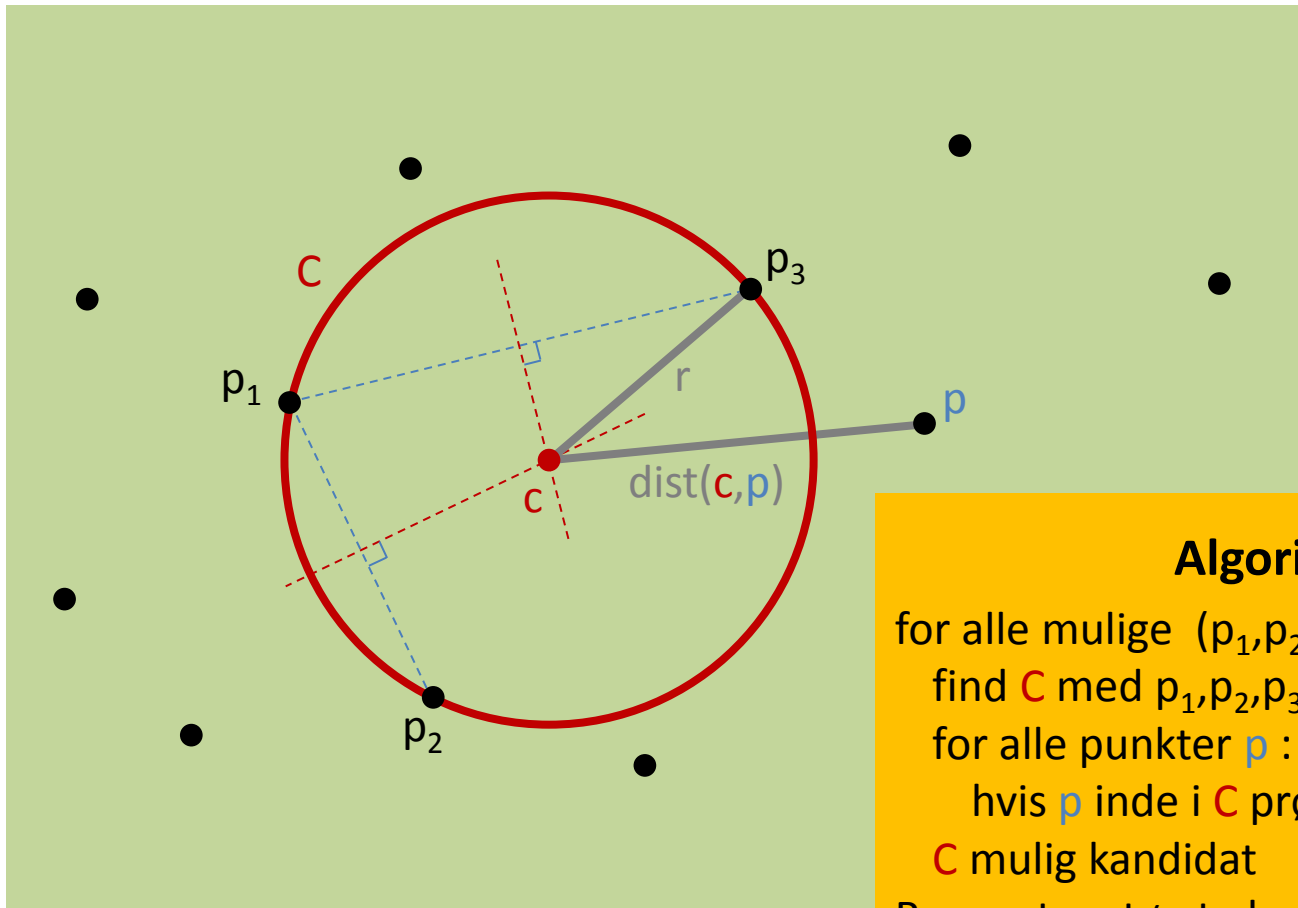
Institut for Datalogi
Aarhus Universitet

Indhold :

Grafer og Algoritmer

Fokus er på de **overordnede idéer**.
Detaljerne i algoritmerne og hvordan de omsættes til konkrete programmer kommes ikke ind på her.

Største Tomme Cirkel

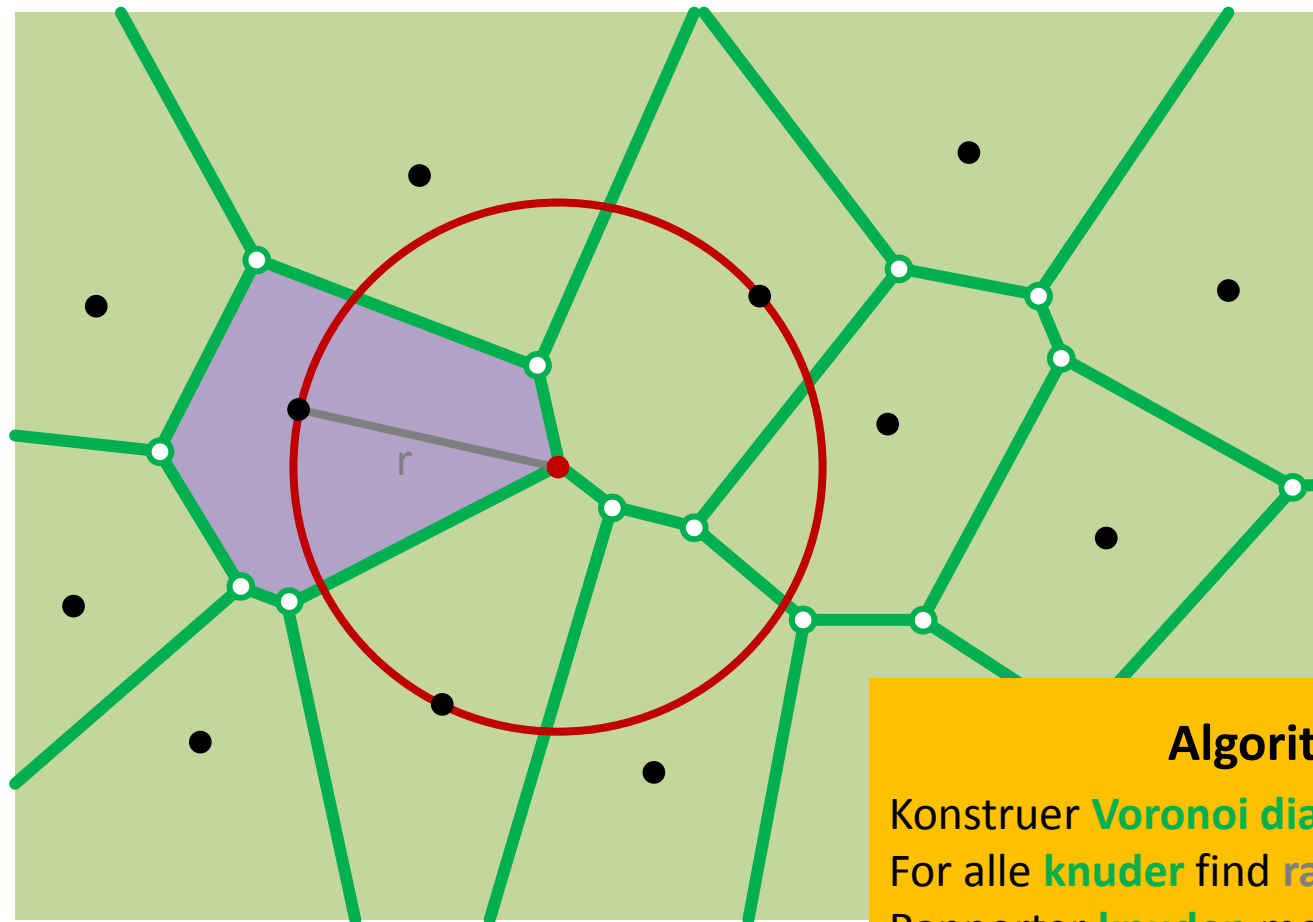


Algoritme

```
for alle mulige  $(p_1, p_2, p_3)$  :  
  find C med  $p_1, p_2, p_3$  på randen  
  for alle punkter p :  
    hvis p inde i C prøv næste  $(p_1, p_2, p_3)$   
  C mulig kandidat  
Rapporter største kandidat fundet
```

Sætning Største tomme cirkel har mindst 3 punkter på randen

Største Tomme Cirkel (II)



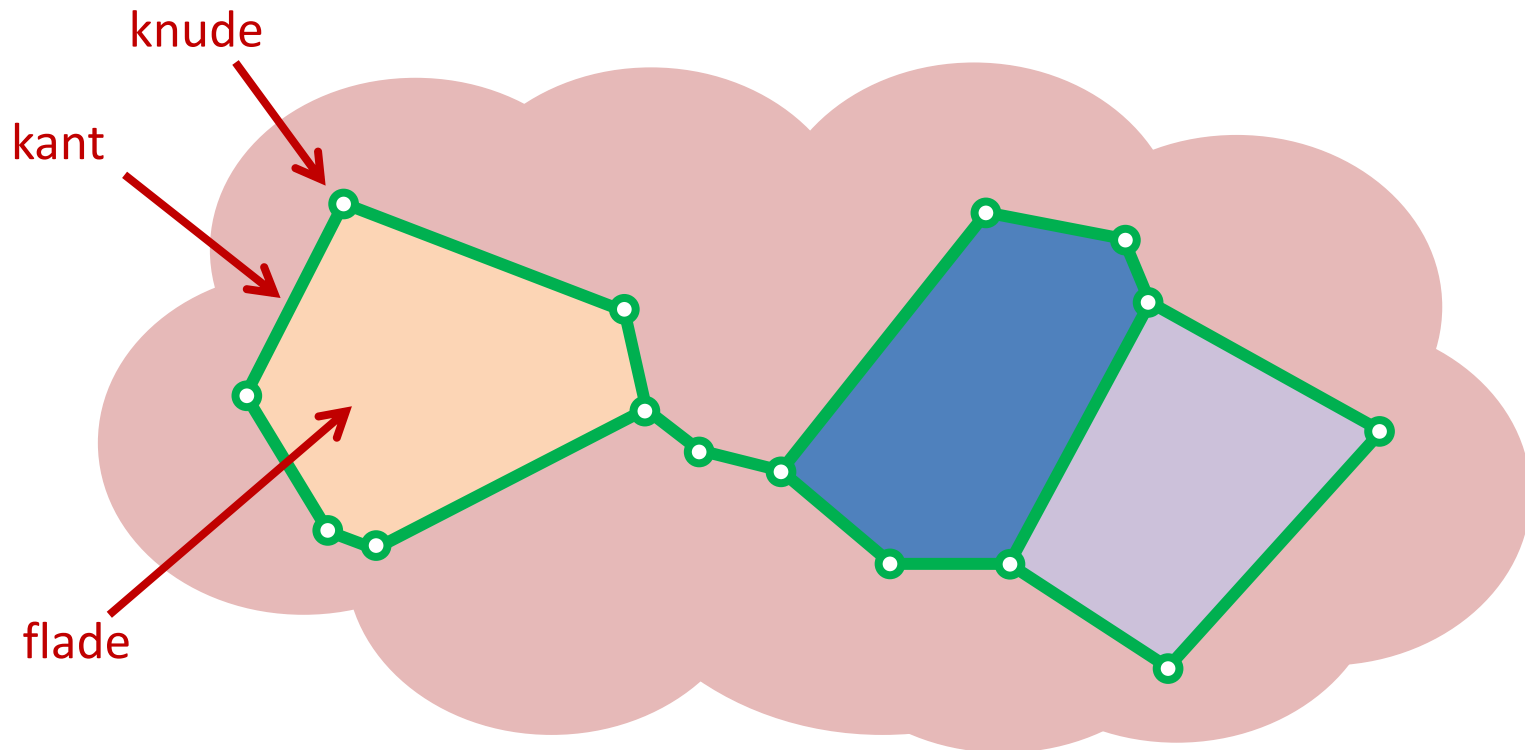
Voronoi diagram

Algoritme

Konstruer **Voronoi diagrammet**
For alle **knuder** find **radius** af **cirklen**
Rapporter **knuden** med størst **radius**

Sætning Antal Voronoi **knuder** $\leq 2 \cdot$ antal **punkter**

Graf



Euler's Sætning : # knuder + # flader - # kanter = 2

$$15 + 4 - 17 = 2$$

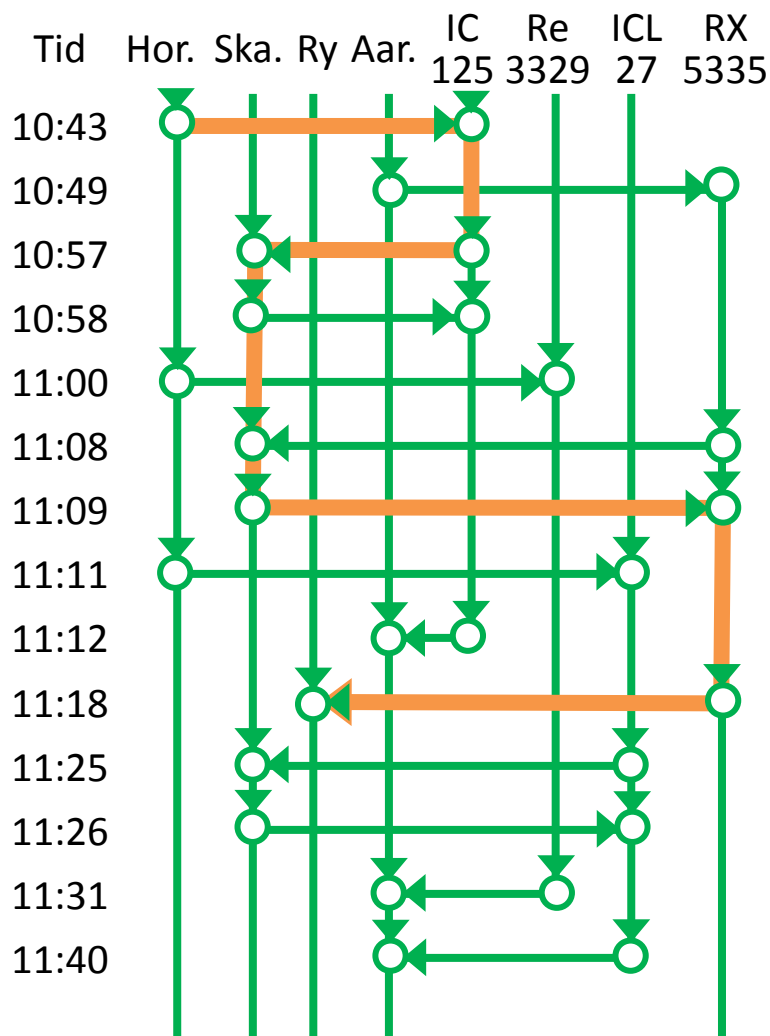
(gælder for sammenhængende grafer der kan tegnes uden krydsende kanter)

Gader
&
Stræder



Eksempel: Find (korteste) veje fra A til B i en graf
Findes der to kant-disjunkte stier fra A til B ?

Rejseplan (Horsens til Ry)



Tog	Ank	Afg	Station
		10:43	Horsens
IC125	10:57	10:58	Skanderborg St
	11:12		Aarhus H
Re3329		11:00	Horsens
	11:31		Aarhus H
		11:11	Horsens
ICL27	11:25	11:26	Skanderborg St
	11:40		Aarhus H
		10:49	Aarhus H
RX5335	11:08	11:09	Skanderborg St
	11:18		Ry St

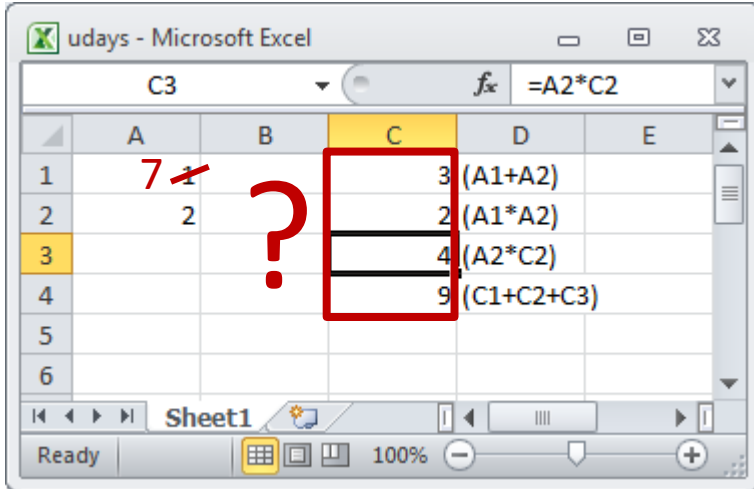


Algoritme

Find tidligste knude for **Ry** der kan nås fra en given start-knude i **Horsens**

uddrag af køreplaner

Opdatering af Regneark

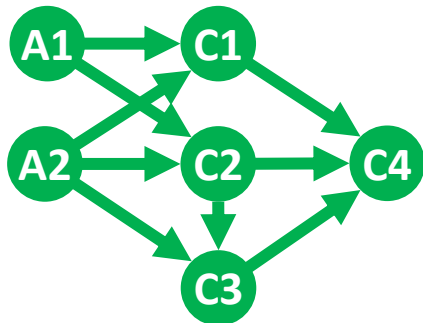


	A	B	C	D	E
1	7		3 (A1+A2)		
2	2		2 (A1*A2)		
3			4 (A2*C2)		
4			9 (C1+C2+C3)		
5					
6					

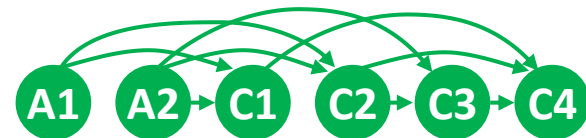
Algoritme

Så længe der findes en uberegnet celle **c**
hvor alle afhængigheder er beregnet :

Beregn **c**



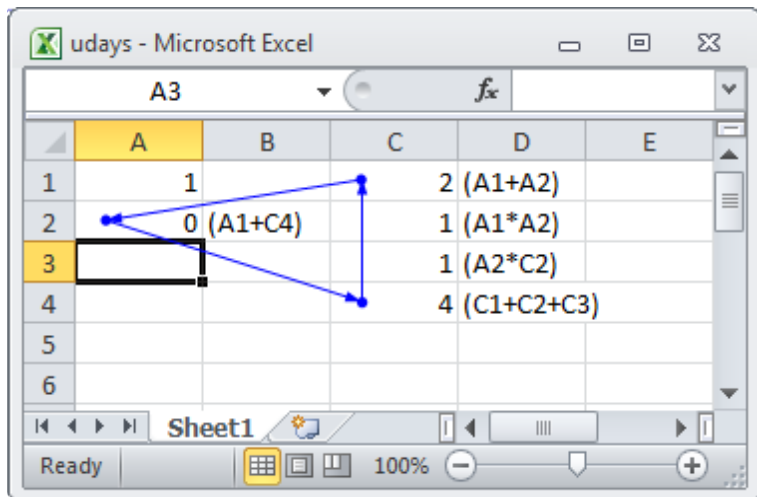
beregnings rækkefølge



topologisk sortering

alle kanter peger fra venstre mod højre

Opdatering af Regneark (II)



	A	B	C	D	E
1	1		2 (A1+A2)		
2	0 (A1+C4)		1 (A1*A2)		
3			1 (A2*C2)		
4			4 (C1+C2+C3)		
5					
6					

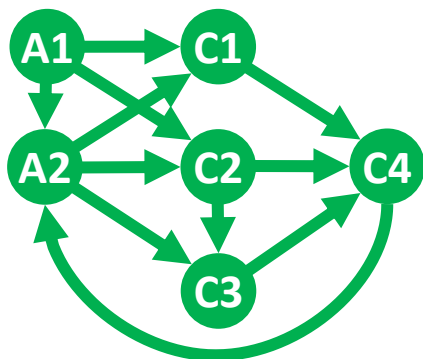
Algoritme

Så længe der findes en uberegnet celle **c**
hvor alle afhængigheder er beregnet :

Beregn **c**

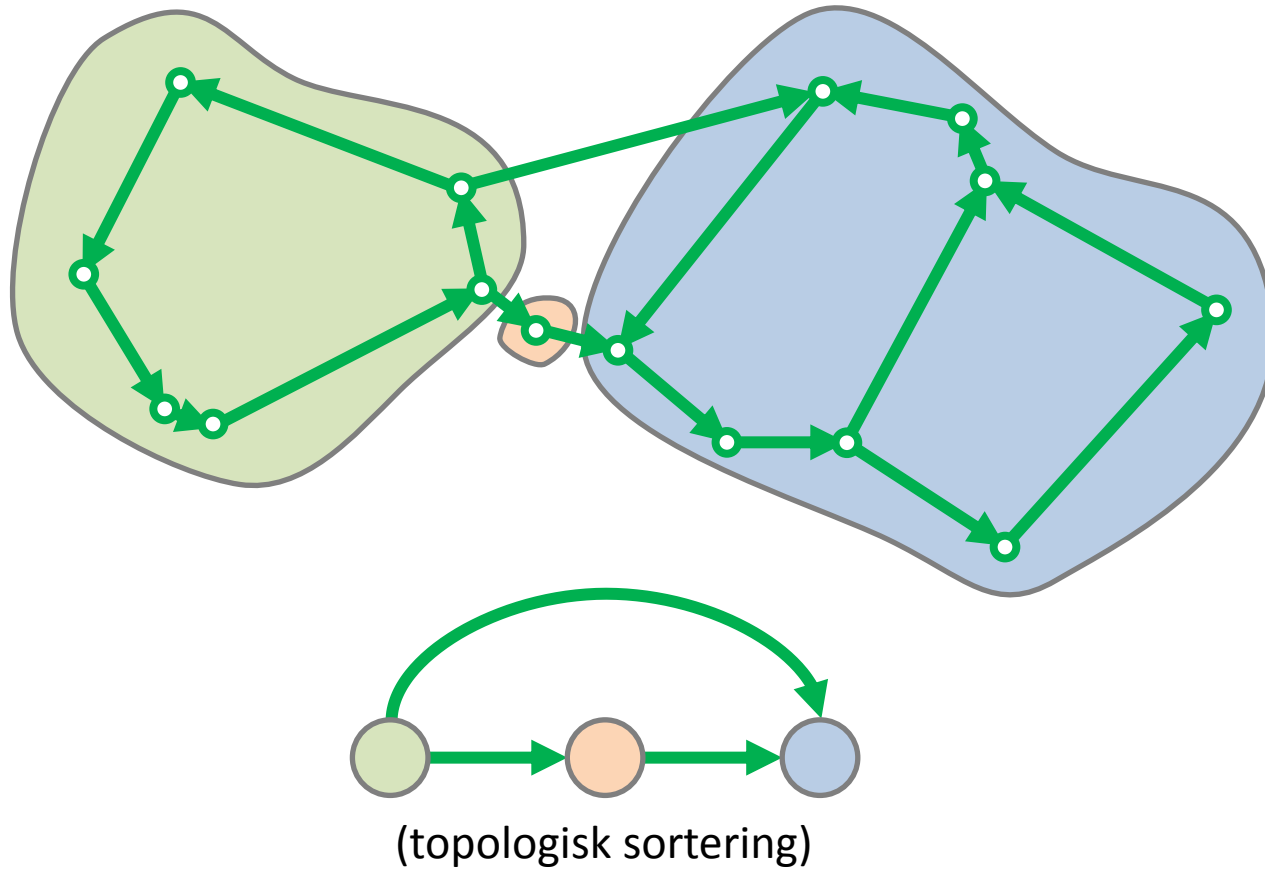
Hvis ikke alle celler beregnet :

Rapporter at der findes en **cykel**



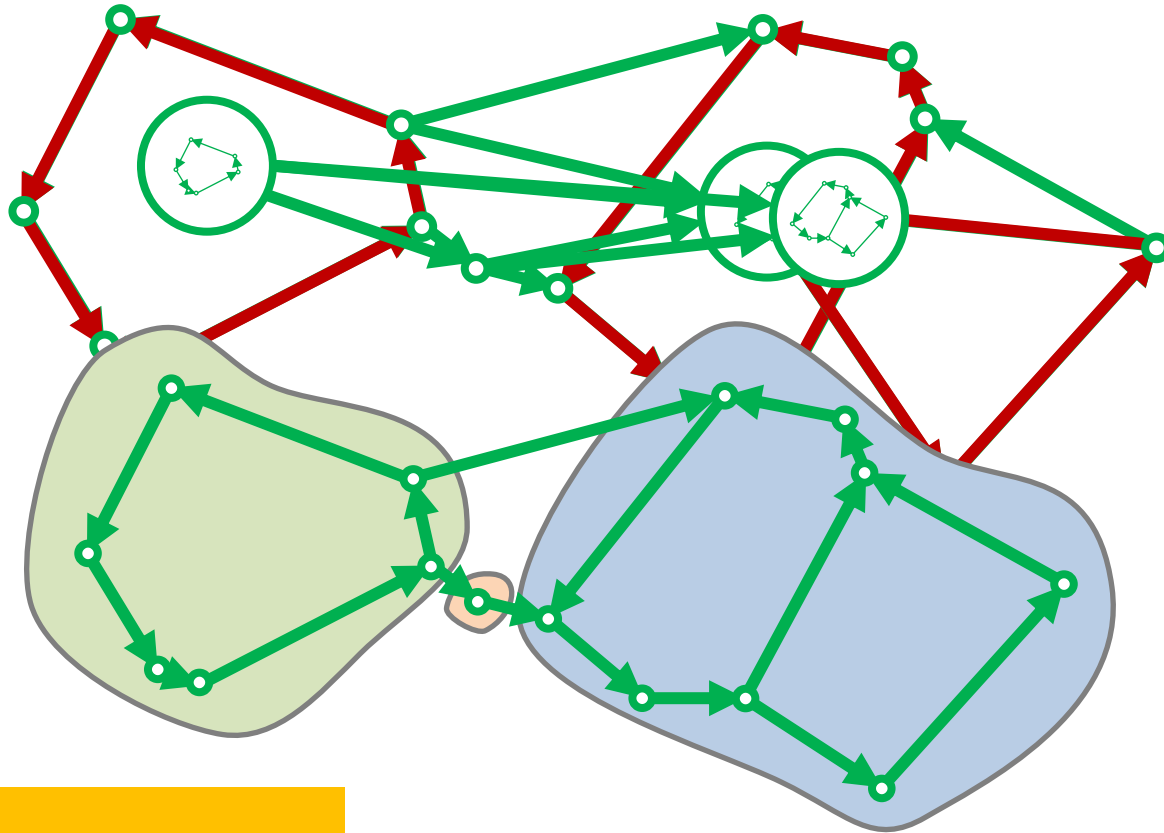
topologisk sorterer
eller
identificerer en cykel

Stærke Sammenhængskomponenter



- Kan alle par af knuder nå hinanden **begge veje** ?
(bruges f.eks. til at checke for fejl i vej-data)

Stærke Sammenhængskomponenter



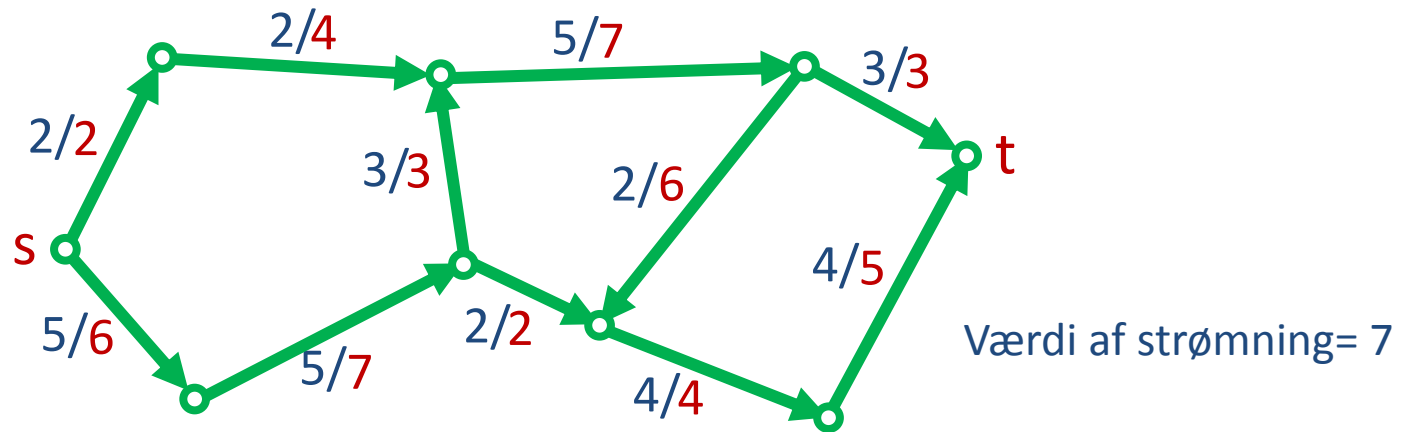
Algoritme :

Så længe der findes en cykel **C** :
Træk **C** sammen til en knude

PAGE 3

DEPARTMENT	COURSE	DESCRIPTION	PREREQS
COMPUTER SCIENCE	CPSC 432	INTERMEDIATE COMPILER DESIGN, WITH A FOCUS ON DEPENDENCY RESOLUTION.	CPSC 432

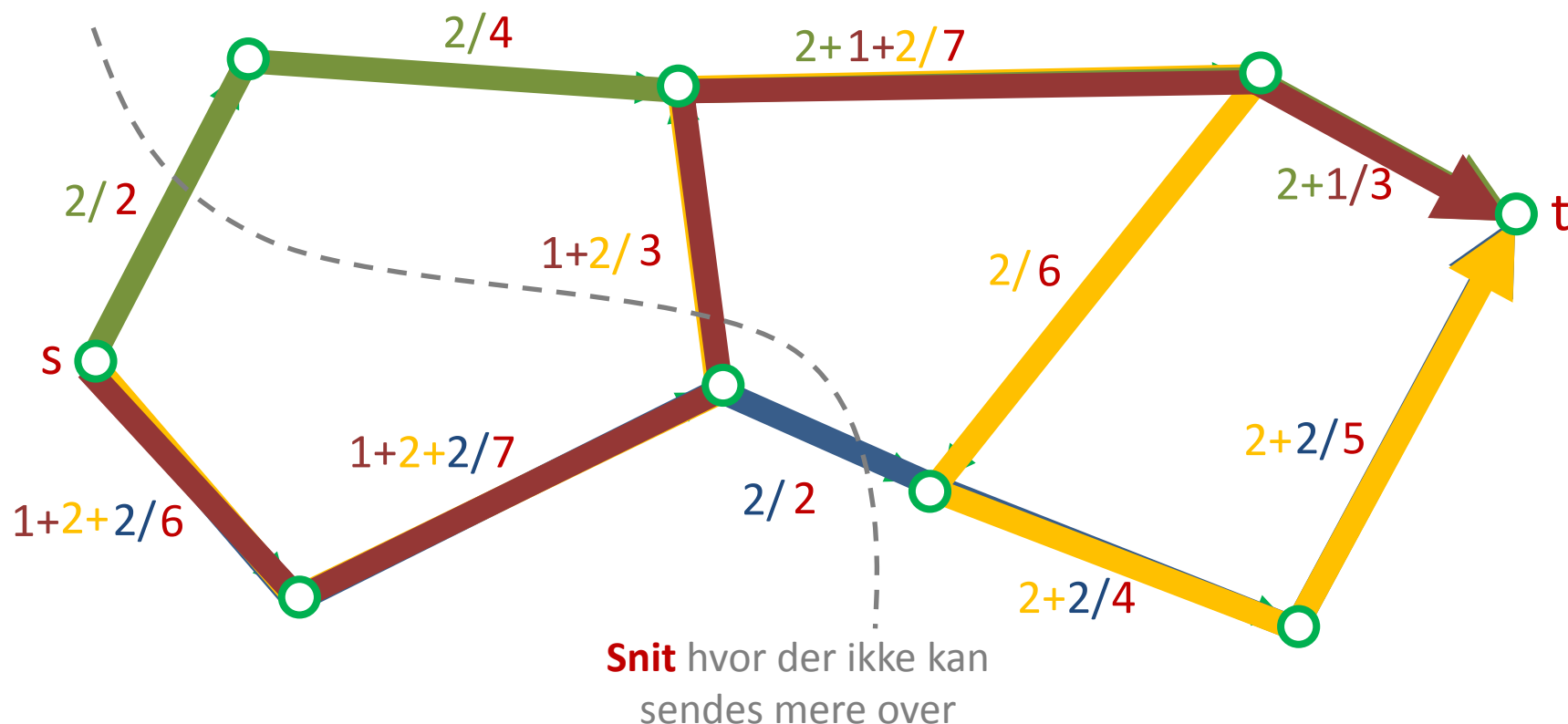
Strømninger i Netværk



- Hver kant har en **kapacitet**
- Send størst mulig **værdi** fra **s** til **t**

(f.eks. vand, kloak, fjernvarme, vejnet kapacitet, el netværk, ...)

Beregning af Strømninger i Netværk



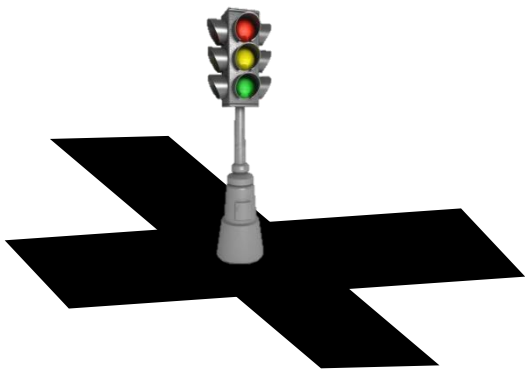
Algoritme

Så længe der findes en forbedrende sti P :
Send maksimal yderligere værdi langs stien P

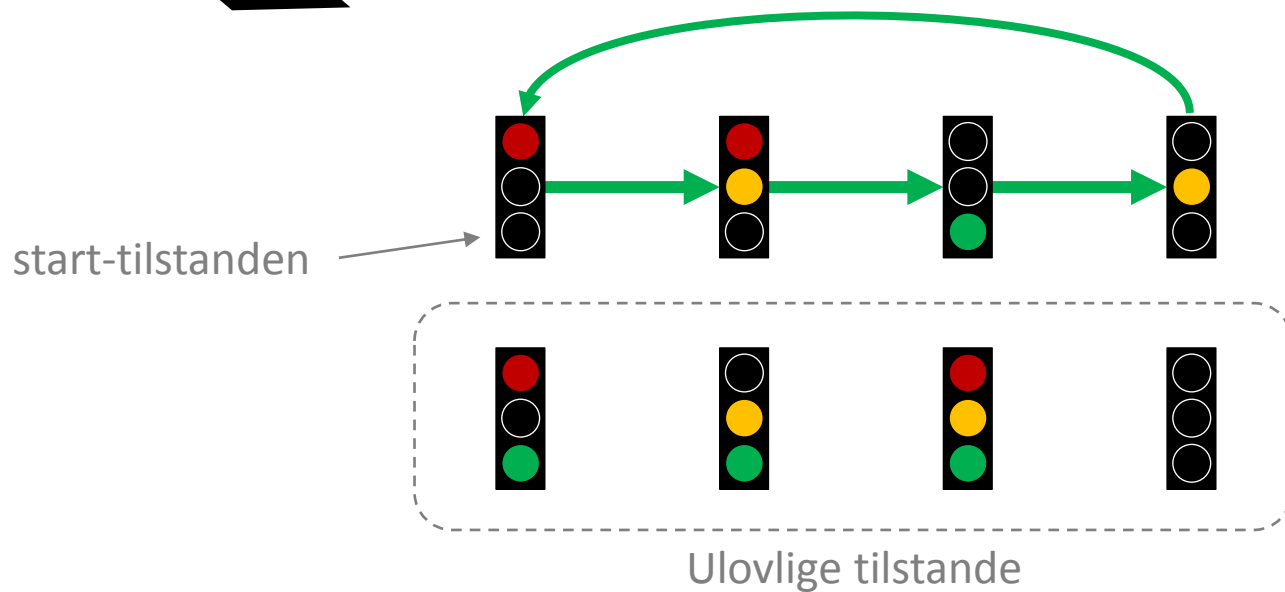
Sætning

Max strømning = min snit



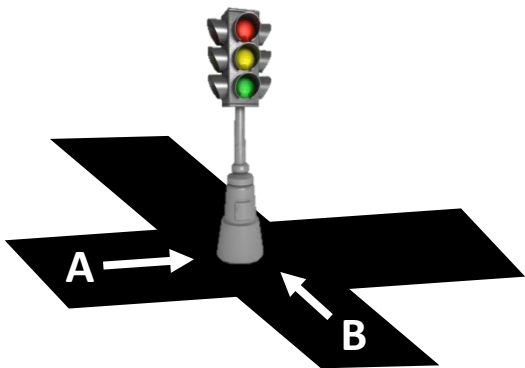


Lyskryds

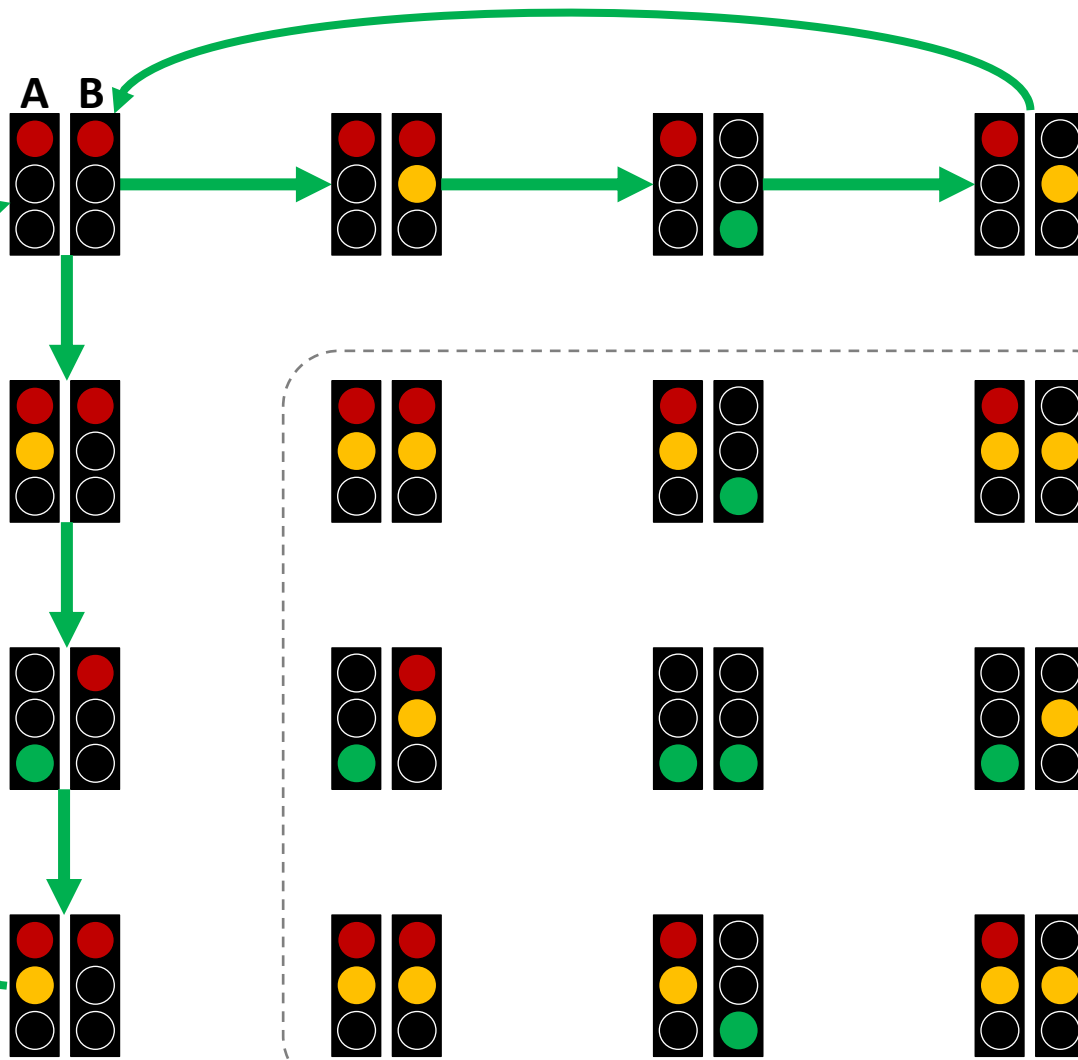


Automatisk kontrol af software til f.eks. styring af lyskryds

- Hvilke tilstande kan man nå (fra start-tilstanden) ?
- Er alle tilstande man kan nå lovlige ?
- Kan der altid blive grønt igen (liveness) ?

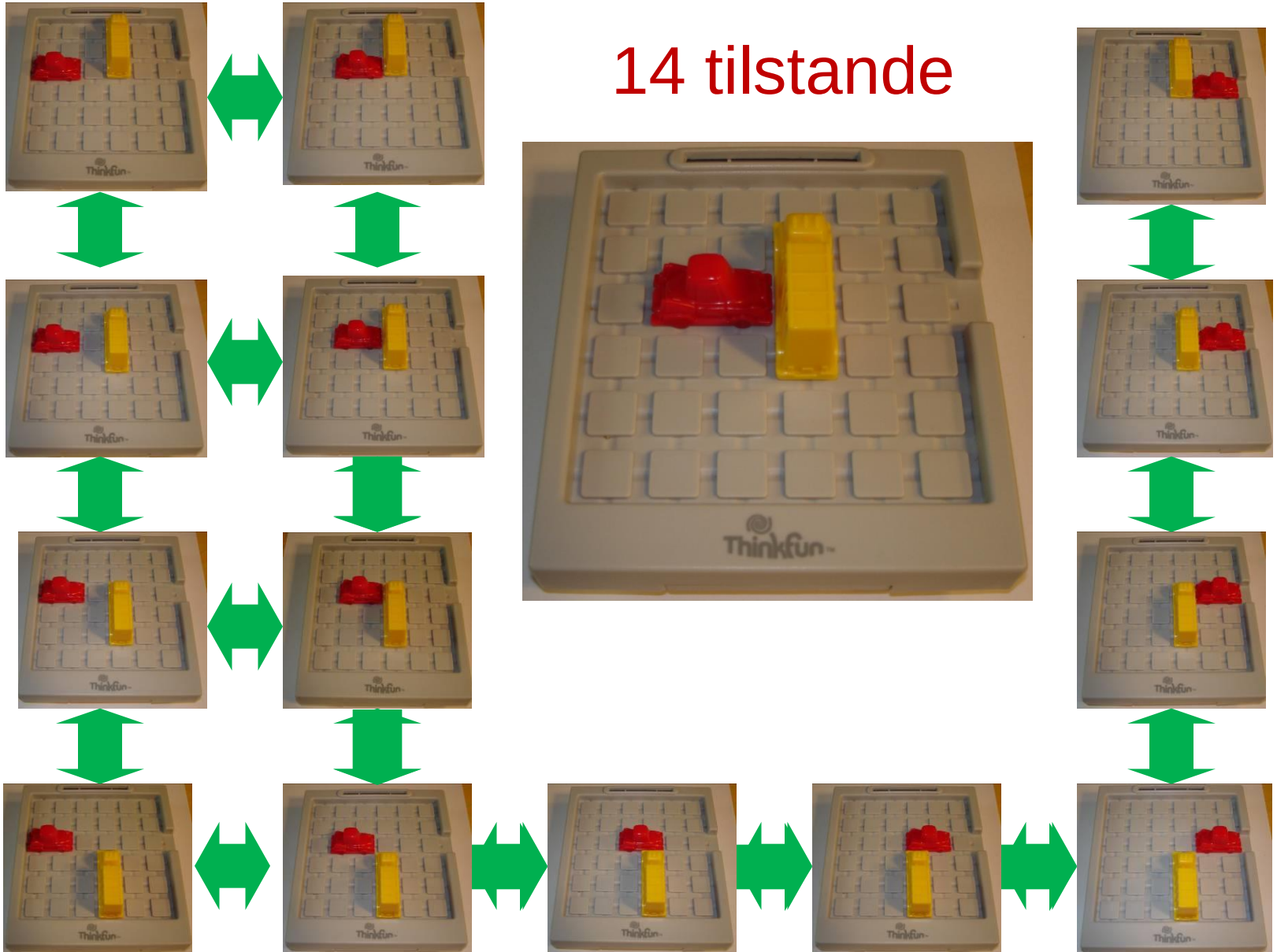


Lyskryds



Ulovlige tilstande

14 tilstande



Rushhour



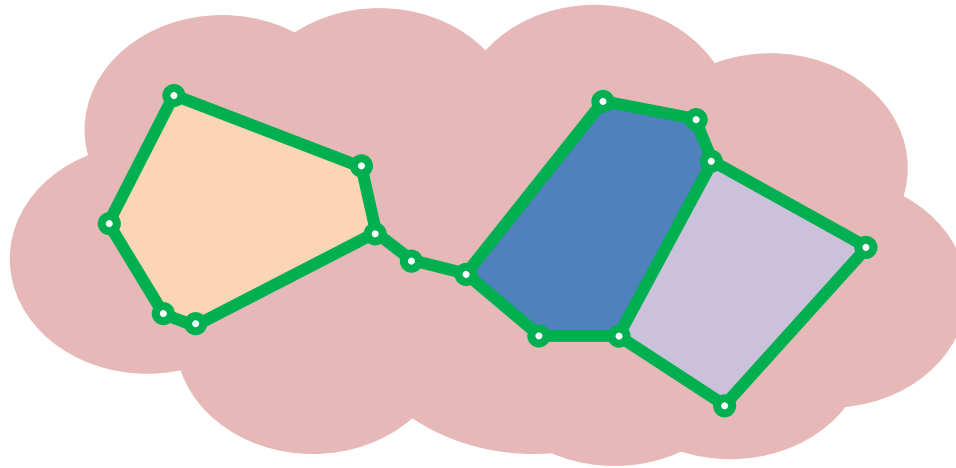
Kort	Korteste vej	Antal tilstande
1	7	1075
2	7	21055
3	13	830
4	8	451
5	8	2784
6	8	2954
7	12	7273
8	11	950
9	11	6759
10	16	4466
13	15	13119
14	16	61712
15	22	540
32	36	690
33	39	4273
37	46	1955
38	47	4405
39	49	4533
40	50	3493

Indhold :

Grafer og Algoritmer

- Planare grafer (Voronoi diagram, Euler's formel)
- Vejnet som grafer (korteste veje, disjunkte stier, stærk sammenhængende)
- Rejseplaner (modellering som graf)
- Regneark (cykler i grafer, topologisk sortering)
- Strømninger i grafer
- 2-kant sammenhængende grafer
- Tilstandsgrafer (lyskryds, spil)

Opsummering



Manger problemer kan løses med
generelle **graf algoritmer**

TAK – Spørgsmål ?