

# Kolloquium zur Bachelorarbeit

---

Referent: Benedikt Lüken-Winkels

Prüfer: Prof. Dr. Henning Fernau

Prof. Dr. Stefan Näher

07. März 2018

Universität Trier

# Knotenüberdeckungsproblem

---

## Knotenüberdeckung

EINGABE: *Graph  $G = (V, E)$ , positive Integer  $k \leq |V|$*

AUSGABE:  *$S \subseteq V$  mit  $|S| \leq k$ , sodass jede Kante aus  $E$  einen Endpunkt in  $S$  hat.*

# Graphreduktion

---



# Einfache Reduktionsregeln

---

*Graph  $G = (V, E)$ , Integer  $k$*

1.  $v \in V$  hat keine Kanten  $\Rightarrow V = V \setminus v$  (Grad<sub>0</sub>-Regel)
2.  $v \in V$  hat genau eine Kante  $\Rightarrow V = V \setminus (v \cup N(v)); k = k - 1$   
(Grad<sub>1</sub>-Regel)
3.  $v \in V$  hat mehr als  $k$  Kanten  $\Rightarrow V = V \setminus (v); k = k - 1$  (Buss-Regel)

*Graph  $G = (V, E)$ , Integer  $k$*

1.  $v \in V$  hat keine Kanten  $\Rightarrow V = V \setminus v$  (Grad<sub>0</sub>-Regel)
2.  $v \in V$  hat genau eine Kante  $\Rightarrow V = V \setminus (v \cup N(v)); k = k - 1$   
(Grad<sub>1</sub>-Regel)
3.  $v \in V$  hat mehr als  $k$  Kanten  $\Rightarrow V = V \setminus (v); k = k - 1$  (Buss-Regel)



*Graph  $G = (V, E)$ , Integer  $k$*

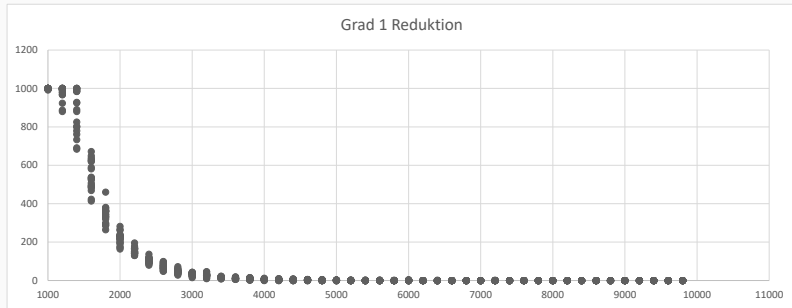
1.  $v \in V$  hat keine Kanten  $\Rightarrow V = V \setminus v$  (Grad<sub>0</sub>-Regel)
2.  $v \in V$  hat genau eine Kante  $\Rightarrow V = V \setminus (v \cup N(v)); k = k - 1$   
(Grad<sub>1</sub>-Regel)
3.  $v \in V$  hat mehr als  $k$  Kanten  $\Rightarrow V = V \setminus (v); k = k - 1$  (Buss-Regel)

*Graph  $G = (V, E)$ , Integer  $k$*

1.  $v \in V$  hat keine Kanten  $\Rightarrow V = V \setminus v$  (Grad<sub>0</sub>-Regel)
2.  $v \in V$  hat genau eine Kante  $\Rightarrow V = V \setminus (v \cup N(v)); k = k - 1$   
(Grad<sub>1</sub>-Regel)
3.  $v \in V$  hat mehr als  $k$  Kanten  $\Rightarrow V = V \setminus (v); k = k - 1$  (Buss-Regel)

*Graph  $G = (V, E)$ , Integer  $k$*

1.  $v \in V$  hat keine Kanten  $\Rightarrow V = V \setminus v$  (Grad<sub>0</sub>-Regel)
2.  $v \in V$  hat genau eine Kante  $\Rightarrow V = V \setminus (v \cup N(v)); k = k - 1$   
(Grad<sub>1</sub>-Regel)
3.  $v \in V$  hat mehr als  $k$  Kanten  $\Rightarrow V = V \setminus (v); k = k - 1$  (Buss-Regel)



# Kronenregel

---

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2     $M_1 := \emptyset$ 
3     $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11    $I' := I$ 
12    $H := N(I)$ 
13    $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```



# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```



# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Algorithmus

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

# Kronenregel - Laufzeit

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

Zeilen 1-5:  $m \cdot d$

# Kronenregel - Laufzeit

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

Zeile 6:  $n$

# Kronenregel - Laufzeit

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2     $M_1 := \emptyset$ 
3     $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv \mid u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11    $I' := I$ 
12    $H := N(I)$ 
13    $I := I \cup \{u \in O \mid \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

Zeile 7(1):  $\frac{(n-2k)d}{2}$

# Kronenregel - Laufzeit

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

Zeile 7(1):  $\sqrt{n - 2k} \cdot m$  (LEDA:mcb\_matching, Hopcroft und Karp)



# Kronenregel - Laufzeit

```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2   $M_1 := \emptyset$ 
3   $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv | u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11      $I' := I$ 
12      $H := N(I)$ 
13      $I := I \cup \{u \in O | \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

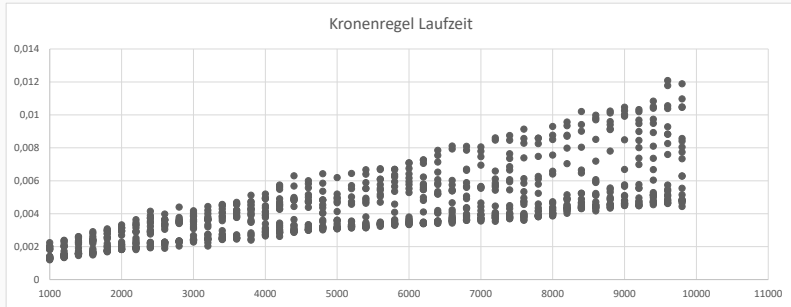
Zeile 8:

# Kronenregel - Laufzeit

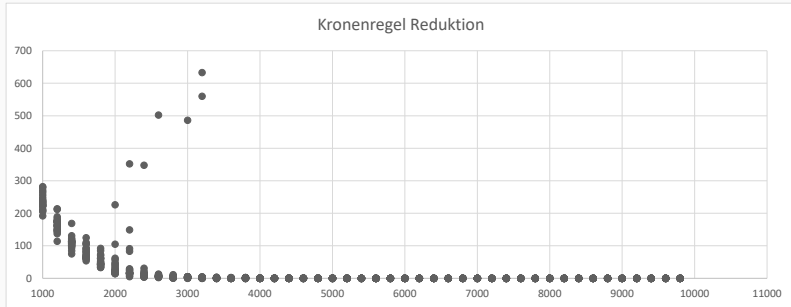
```
0   $G = (V, E)$ 
1   $M_1 :=$  Maximal Matching von  $G$ 
2     $M_1 := \emptyset$ 
3     $\forall e \in E:$ 
4       $M_1 = M_1 \cup e$ 
5      Entferne  $e$  und  $N(e)$  aus der weiteren Betrachtung
6   $O :=$  nicht gepaarte Knoten in  $M_1$ 
7   $M_2 :=$  Maximum Matching von  $B = (O, N(O), \{uv \mid u \in O \wedge v \in N(O)\})$ 
8   $I :=$  nicht gepaarte Knoten aus  $O$  in  $M_2$ 
9   $I' := \emptyset$ 
10 while  $I' \neq I$ 
11    $I' := I$ 
12    $H := N(I)$ 
13    $I := I \cup \{u \in O \mid \exists v \in H (uv \in M_2)\}$ 
14 Entferne  $N(I)$  aus  $G$ 
```

Zeilen 11-12: 1 +

# Kronenregel - Laufzeit



# Kronenregel - Ergebnisse



# Nemhauser-Trotter-Regel

---

## Nemhauser-Trotter-Theorem

Für einen Graphen  $G = (V, E)$  können zwei disjunkte Mengen  $C_0$  und  $V_0$  gefunden werden, sodass

1.  $C_0$  in einer minimalen Knotenüberdeckung von  $G$  enthalten ist,
2. der Teilgraph  $G[V_0]$  eine Knotenüberdeckung der Größe  $\leq |V_0|/2$  hat,
3. und  $VC(G) = VC(G[V_0]) \cup C_0$  gilt.

## Nemhauser-Trotter-Theorem

Für einen Graphen  $G = (V, E)$  können zwei disjunkte Mengen  $C_0$  und  $V_0$  gefunden werden, sodass

1.  $C_0$  in einer minimalen Knotenüberdeckung von  $G$  enthalten ist,
2. der Teilgraph  $G[V_0]$  eine Knotenüberdeckung der Größe  $\leq |V_0|/2$  hat,
3. und  $VC(G) = VC(G[V_0]) \cup C_0$  gilt.

## Nemhauser-Trotter-Theorem

*Für einen Graphen  $G = (V, E)$  können zwei disjunkte Mengen  $C_0$  und  $V_0$  gefunden werden, sodass*

- 1.  $C_0$  in einer minimalen Knotenüberdeckung von  $G$  enthalten ist,*
- 2. der Teilgraph  $G[V_0]$  eine Knotenüberdeckung der Größe  $\leq |V_0|/2$  hat,*
- 3. und  $VC(G) = VC(G[V_0]) \cup C_0$  gilt.*



```
0   $G = (V, E)$ 
1  Bipartiten Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

```
0   $G = (V, E)$ 
1  Bipartiten Graphen erstellen  $B = (V, V', E')$ 
2  mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

```
0   $G = (V, E)$ 
1  Bipartiden Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

```
0   $G = (V, E)$ 
1  Bipartiden Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

```
0   $G = (V, E)$ 
1  Bipartiden Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

```
0   $G = (V, E)$ 
1  Bipartiden Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

```
0   $G = (V, E)$ 
1  Bipartiten Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

Zeilen 1-2:  $n \cdot 2d$

```
0   $G = (V, E)$ 
1  Bipartiden Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

Zeilen 3-4:  $\sqrt{n} \cdot m$  (LEDA:mcb\_matching, Hopcroft und Karp)



```
0   $G = (V, E)$ 
1  Bipartiten Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

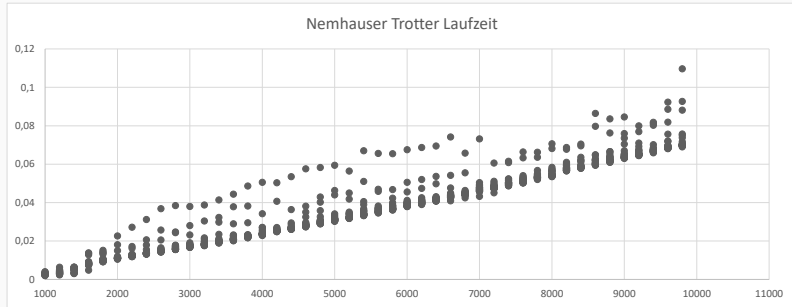
Zeilen 5-6:  $2n + k \cdot d$

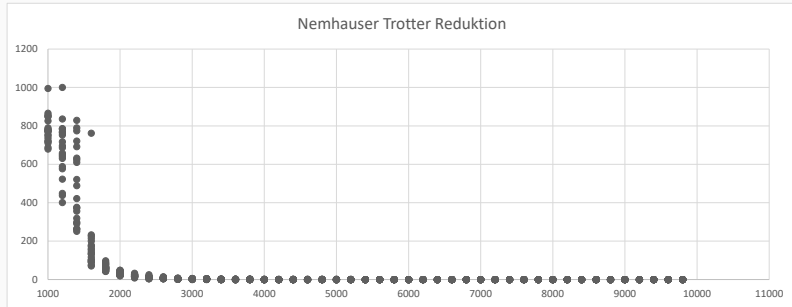
```
0   $G = (V, E)$ 
1  Bipartiten Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

$$n \cdot 2d + \sqrt{n} \cdot m + 2n + k \cdot d$$

```
0   $G = (V, E)$ 
1  Bipartiden Graphen erstellen  $B = (V, V', E')$ 
2    mit  $E' := \{\{x, y'\}, \{x', y\} \mid \{x, y\} \in E\}$ 
3  Maximum Matching  $M$  von  $B$  bestimmen
4   $C_B := VC(B)$ 
5   $C_0 := \{x \in V \mid x \in C_B \text{ und } x' \in C_B\}$ 
6   $V_0 := \{x \in V \mid \text{entweder } x \in C_B \text{ oder } x' \in C_B\}$ 
```

$$n \cdot 2d + \sqrt{n} \cdot m + 2n + k \cdot d \Rightarrow O(n + \sqrt{n} \cdot m)$$





# Vergleich

---



# Anwendung

---



**Tabelle 1:** Anwendung kombinierter Reduktionsregeln

Kombination	Anwendungen <sub>1</sub>	Anwendungen <sub>2</sub>	Anwendungen <sub>3</sub>	Reduktion
K - G <sub>1</sub>	3.63	4.3	-	331.8
G <sub>1</sub> - K	4.37	3.22	-	331.17
K - NT	0.8	0.38	-	68.28
NT - K	0.45	0.56	-	68.6
G <sub>1</sub> - NT	1.33	0.017	-	99.87
NT - G <sub>1</sub>	0.28	1.13	-	99.87
K - G <sub>1</sub> - NT	3.61	4.29	0.11	334.67
K - NT - G <sub>1</sub>	3.6	0.87	3.39	334.83
G <sub>1</sub> - NT - K	4.36	0.12	3.2	334.17
G <sub>1</sub> - K - NT	3.61	3.2	0.65	334.16
NT - K - G <sub>1</sub>	0.39	3.44	4.03	335.2
NT - G <sub>1</sub> - K	0.91	3.42	3.2	334.16

# Implementierung

---



## Fazit

---

