

Eine Variante des Sugiyama-Algorithmus für ungerichtete Graphen mit Portconstraints

Masterarbeit Julian Walter

Lehrstuhl I: Algorithmen und Komplexität

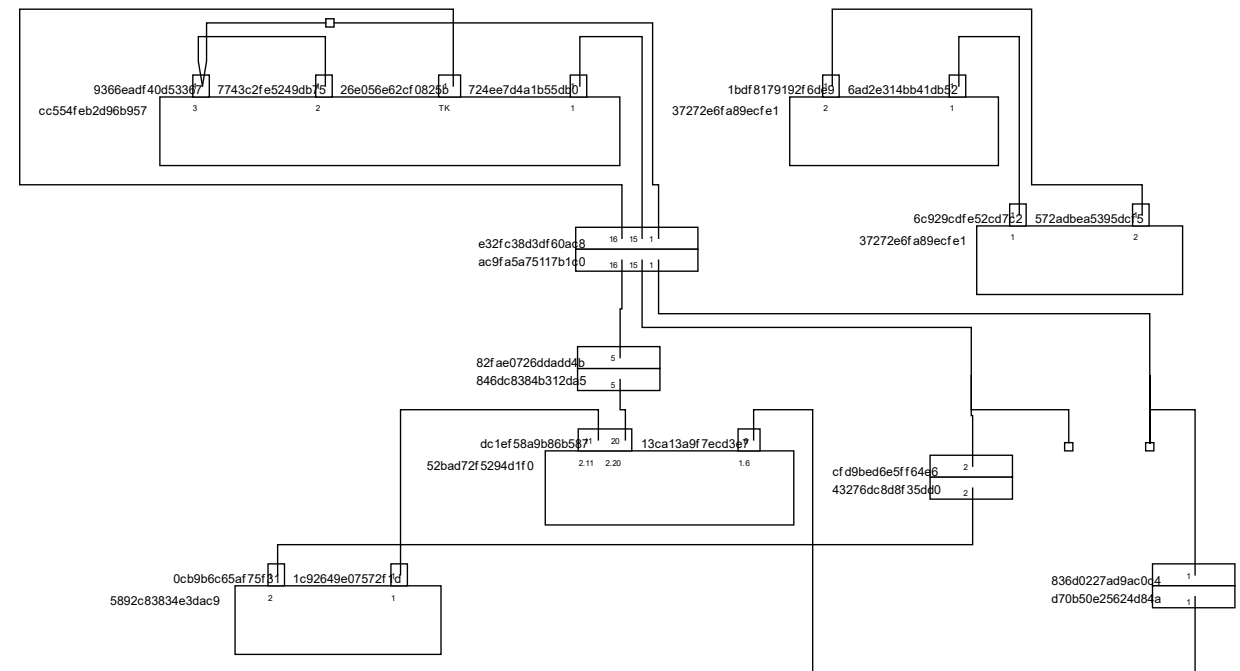
Betreuer: Johannes Zink, M.Sc.

Prof. Dr. Alexander Wolff

Motivation

- Visualisierung von elektrischen Schaltplänen
- Bisherige Methode: KLayerLayered-Algorithmus KIELER von Schulze et al.
- Verringerung Anzahl von Kreuzungen Knicken
- Durch Nutzung zusätzlicher Freiheiten

Bisherige Zeichnung:



(0s)
com.denkbare.layouting.webcable.WebCableSettings@22445d8e

Methods for Visual Understanding of Hierarchical System Structures

Kozo Sugiyama, Shojiro Tagawa, Mitsuhiko Toda

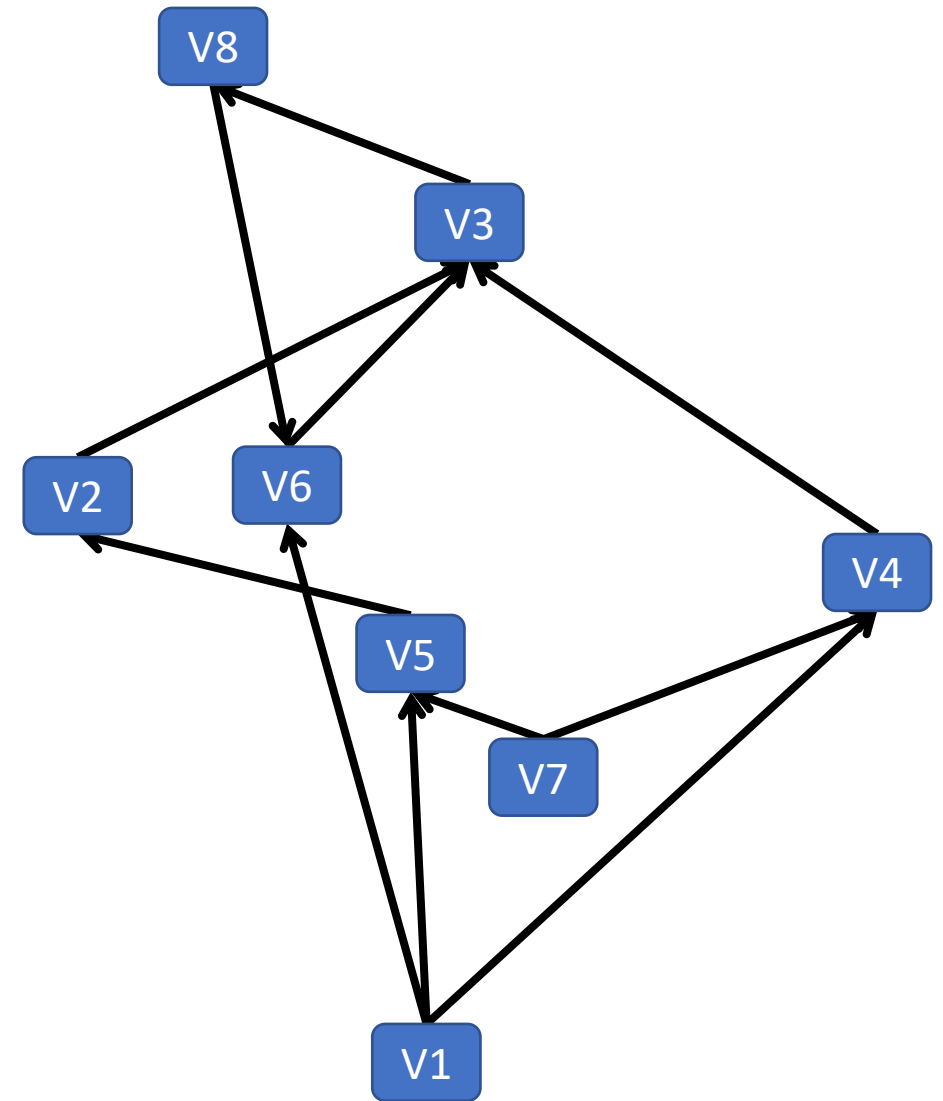
[IEEE TSMC 1981]

Sugiyama-Algorithmus

- Visualisierung von Graphen
 - gerichtet
 - hierarchisch
 - heuristisch
- Algorithmus in 4 Schritten
 - 1: Vorverarbeitung
 - 2: Kreuzungsreduzierung
 - 3: Knotenpositionierung
 - 4: Ausgabe

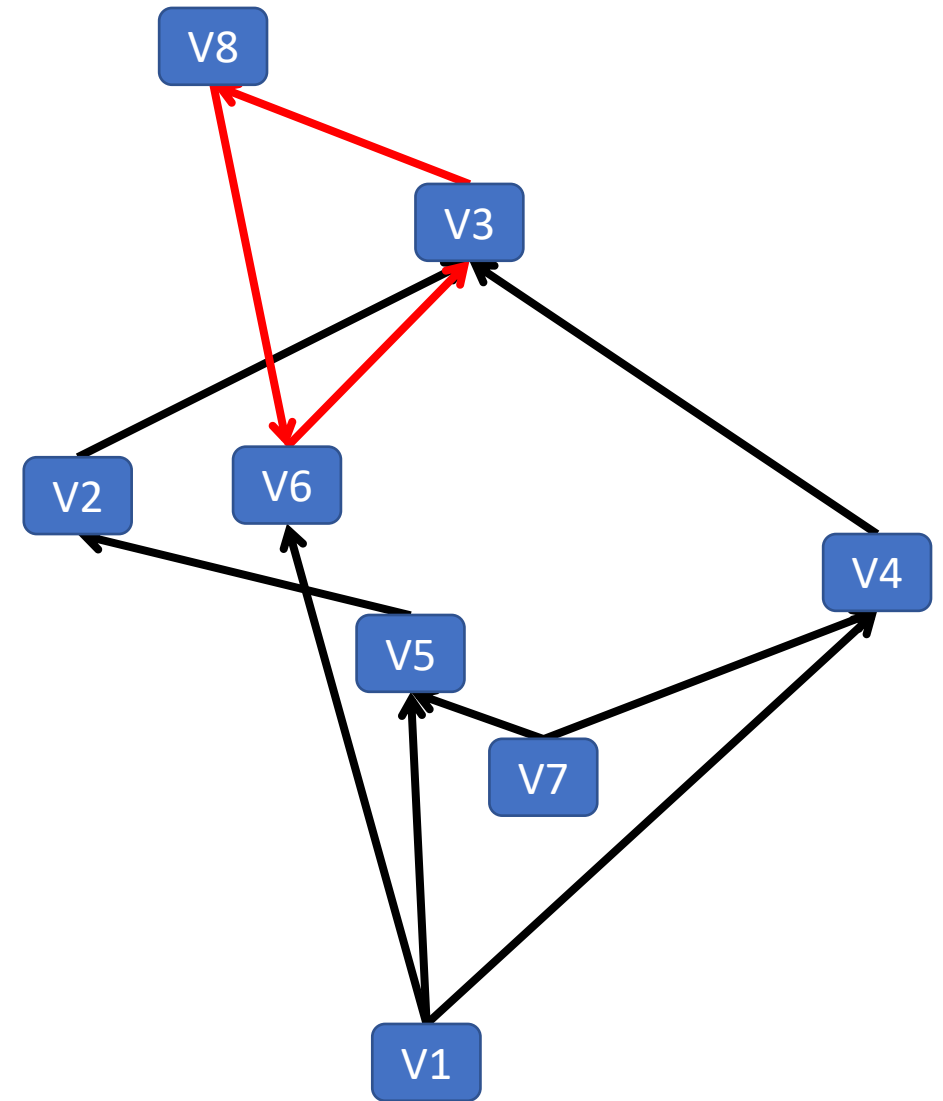
Sugiyama-Algorithmus

- 1: Vorverarbeitung
gerichtete Kreise auflösen



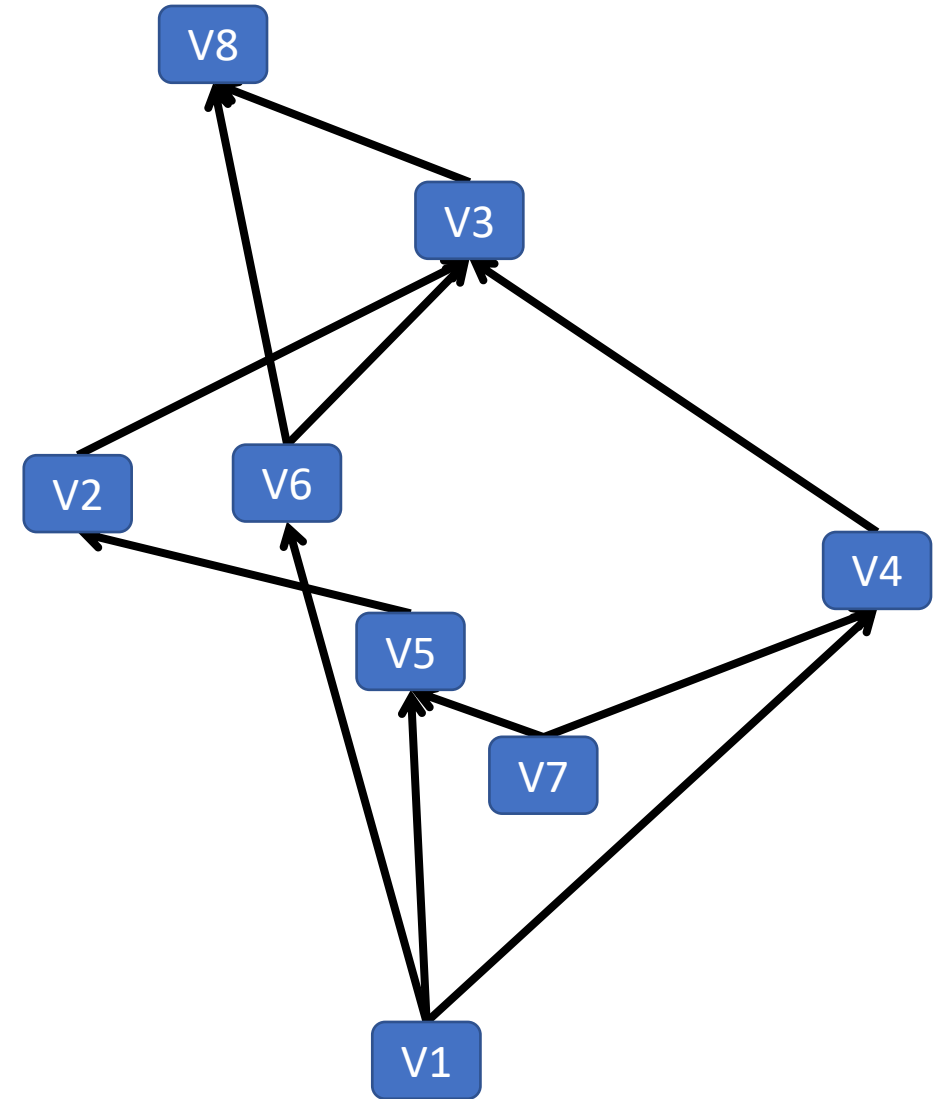
Sugiyama-Algorithmus

- 1: Vorverarbeitung
gerichtete Kreise auflösen



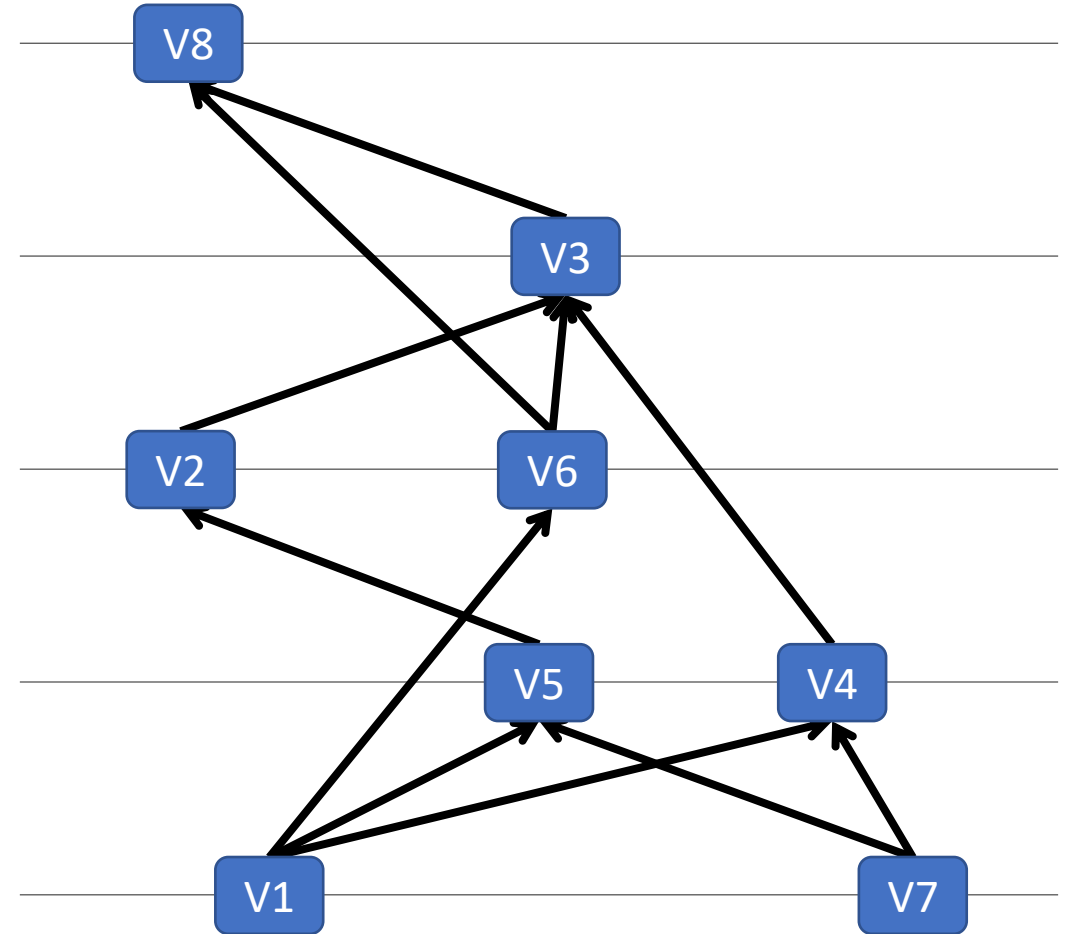
Sugiyama-Algorithmus

- 1: Vorverarbeitung
gerichtete Kreise auflösen
Knoten in Lagen einteilen



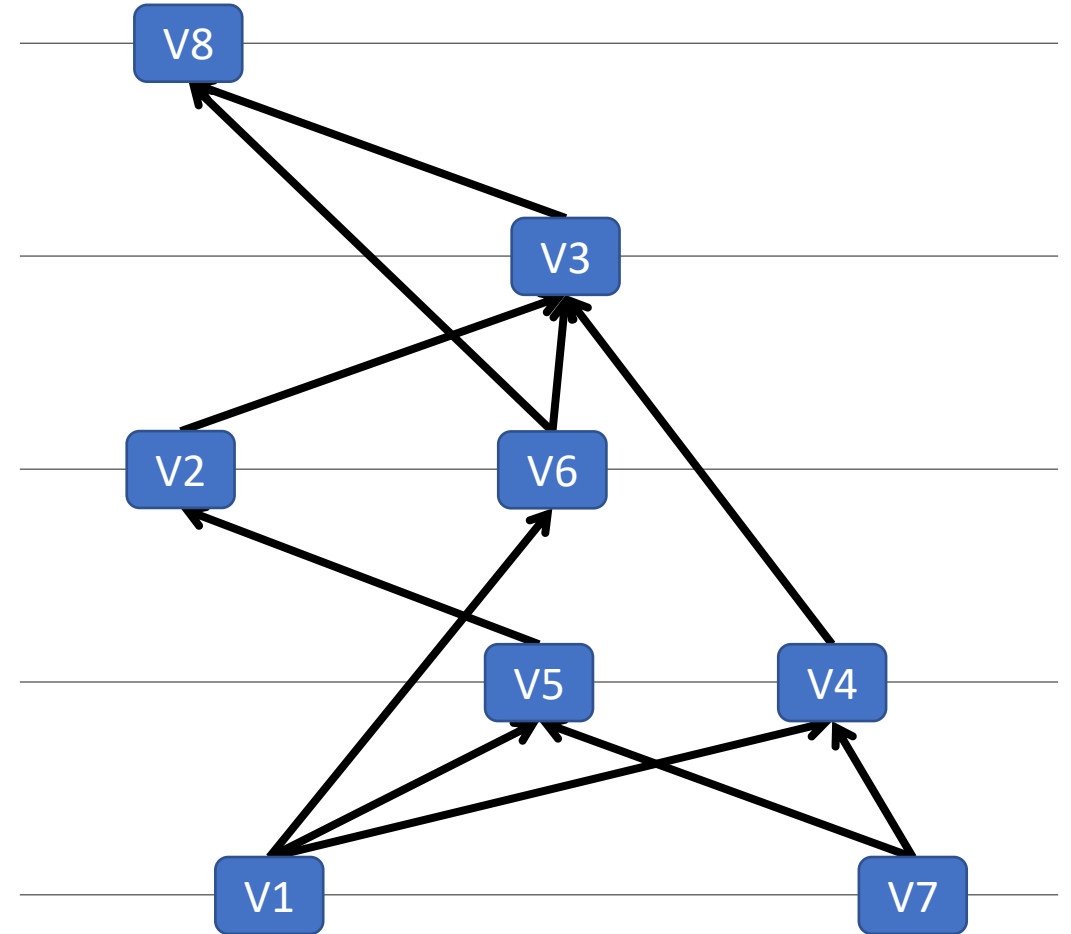
Sugiyama-Algorithmus

- 1: Vorverarbeitung
gerichtete Kreise auflösen
Knoten in Lagen einteilen



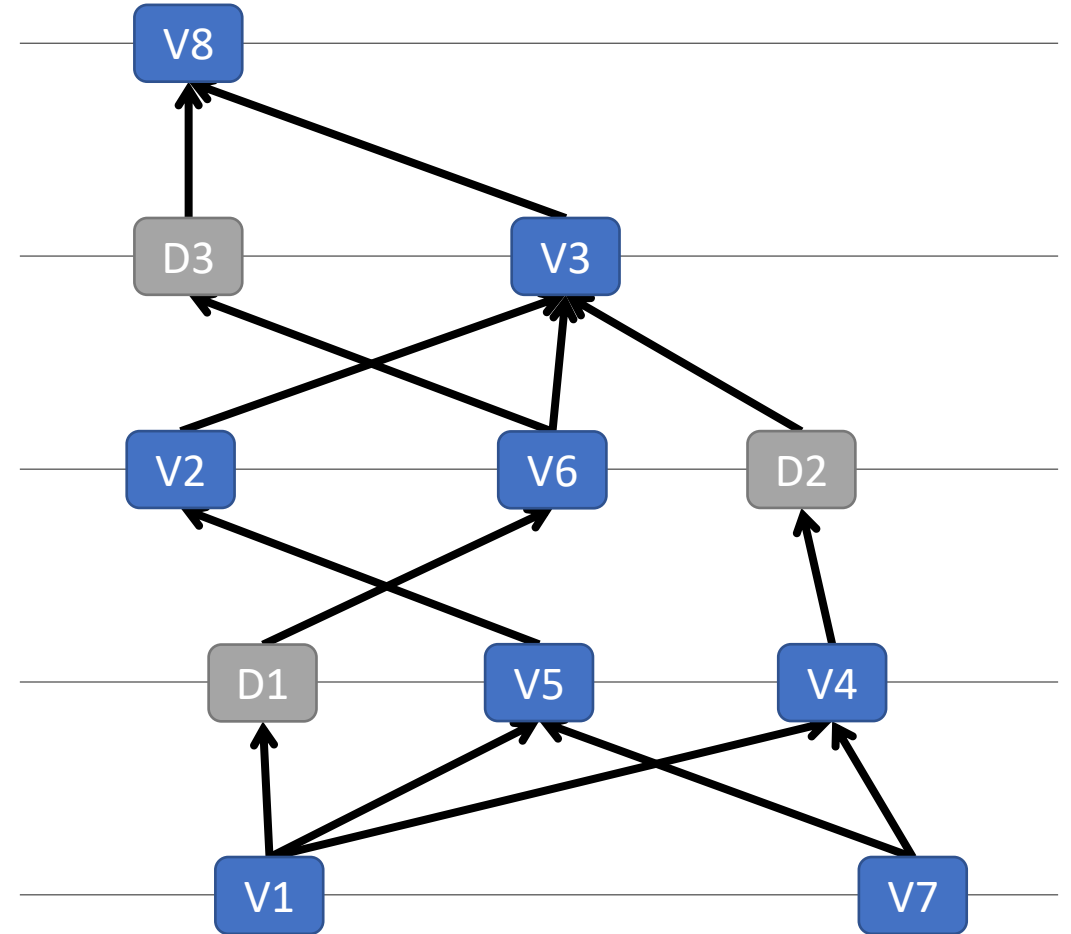
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen



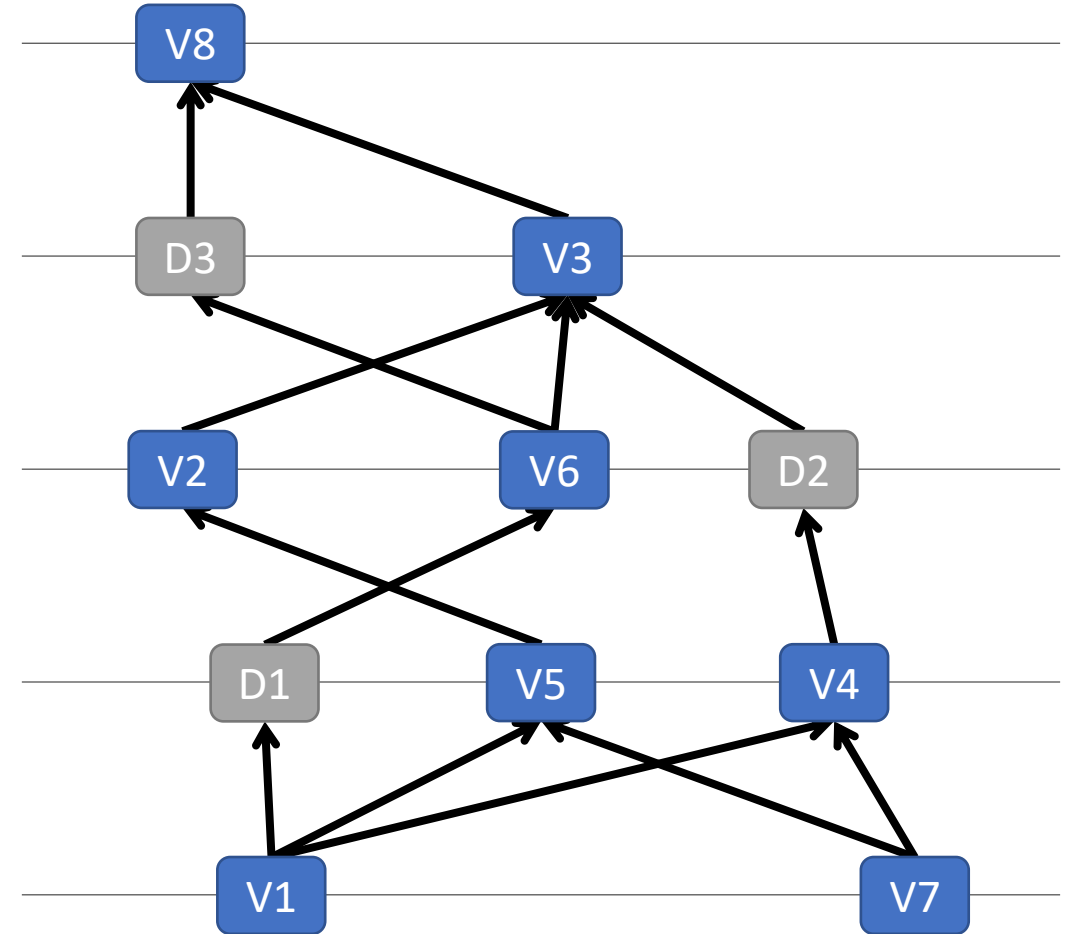
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen



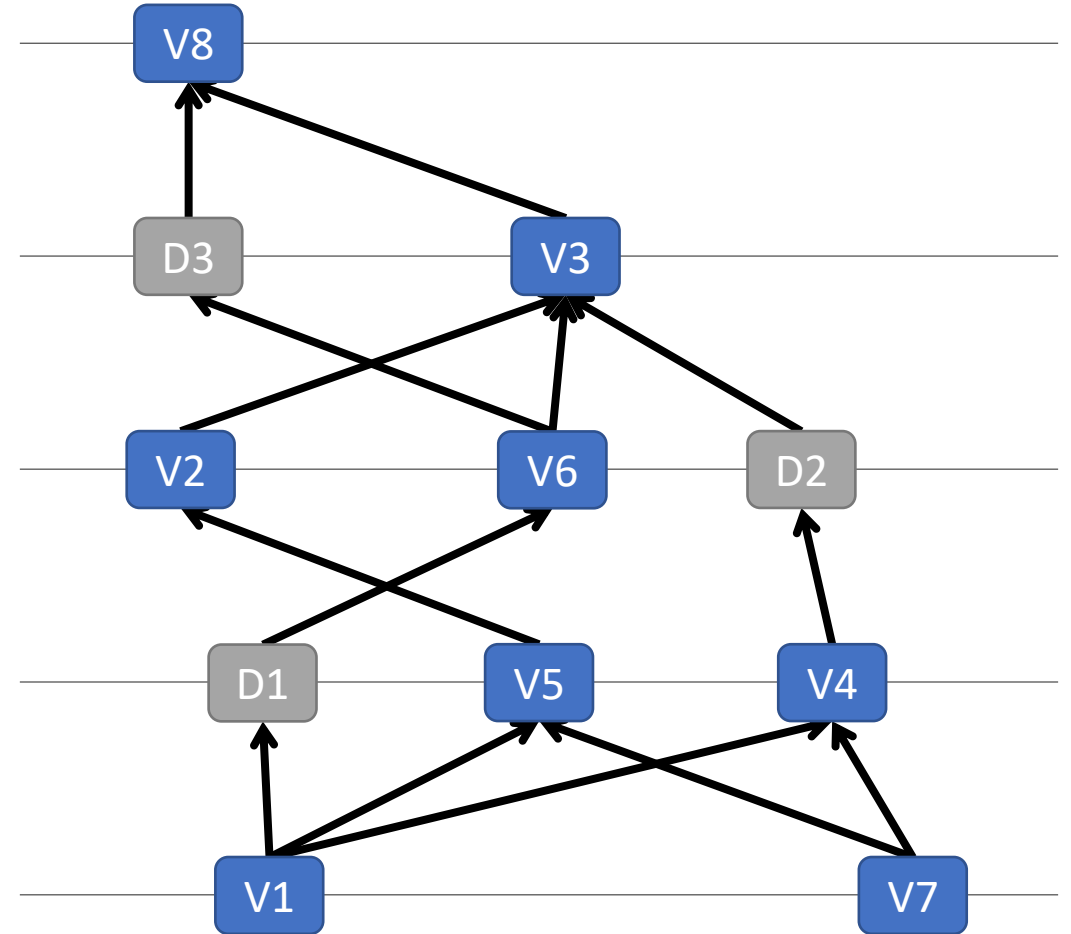
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik



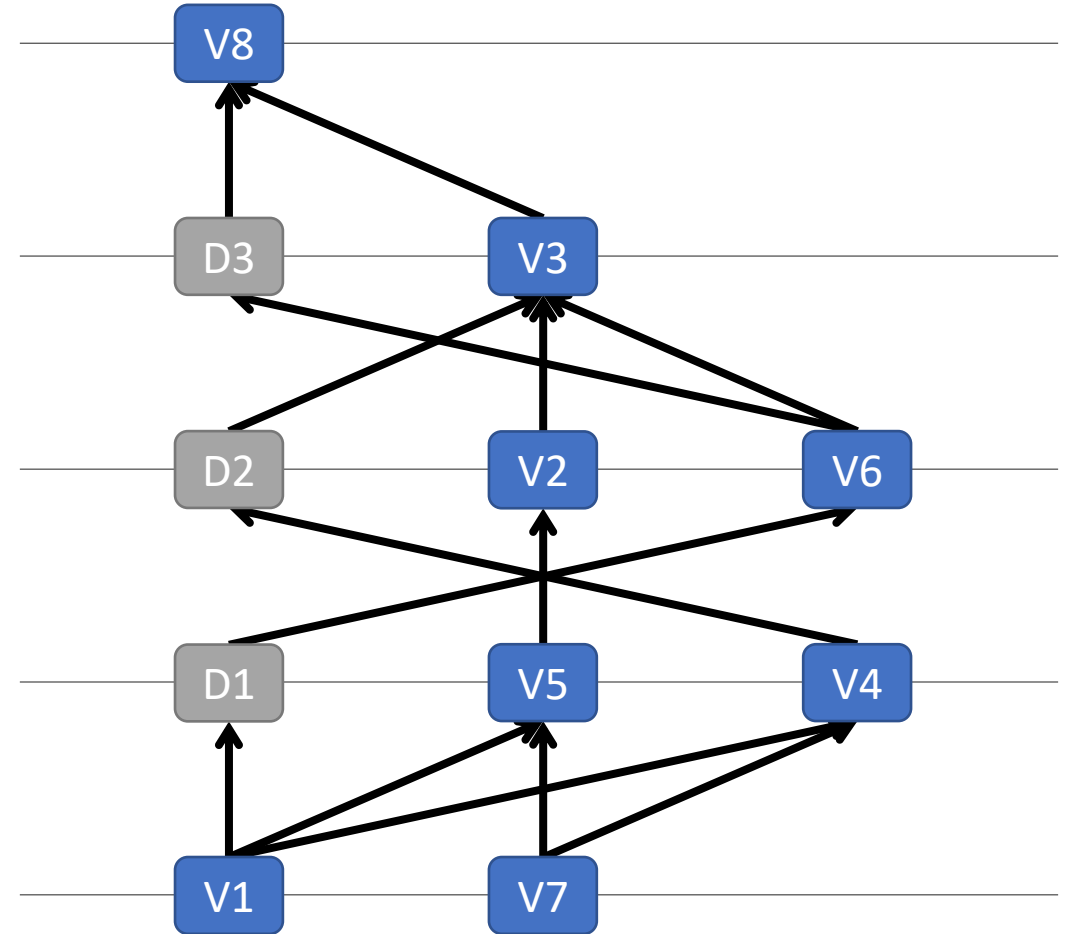
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
 - Start mit zufälliger Reihenfolge



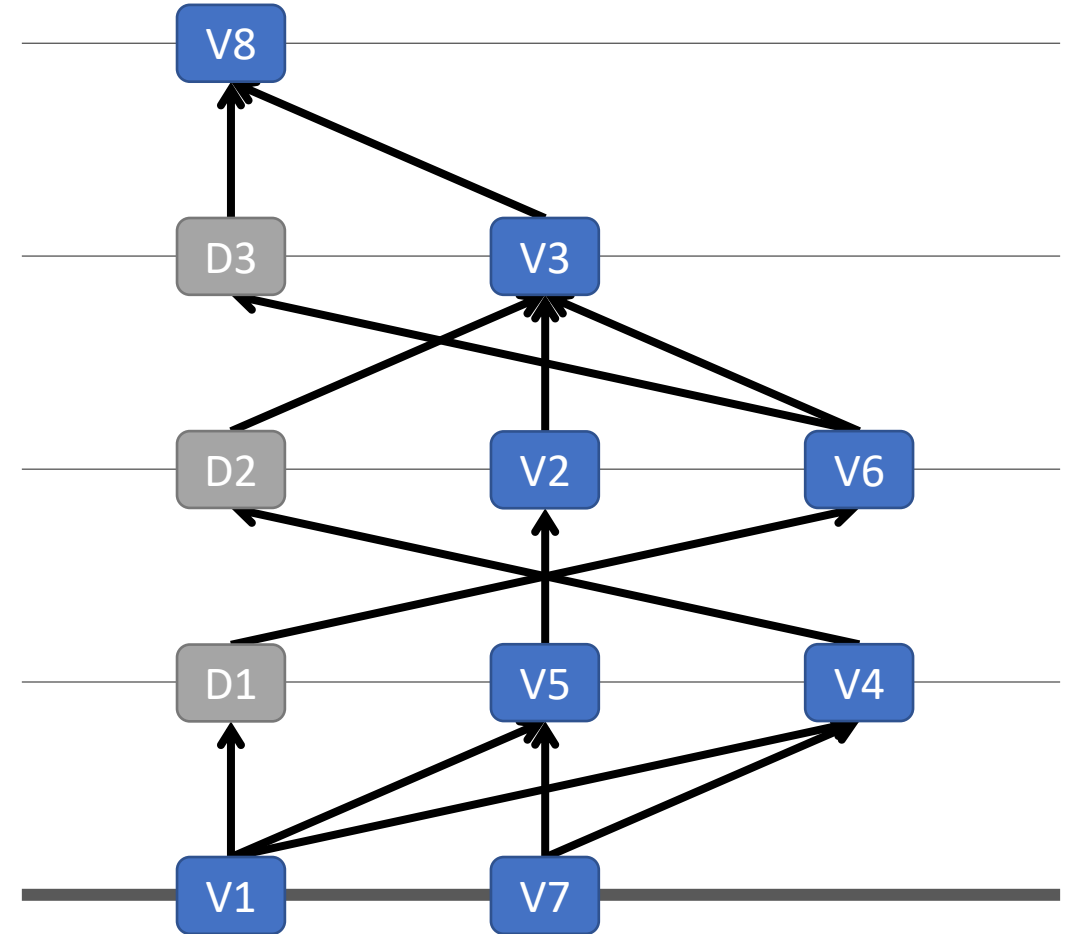
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
 - Start mit zufälliger Reihenfolge



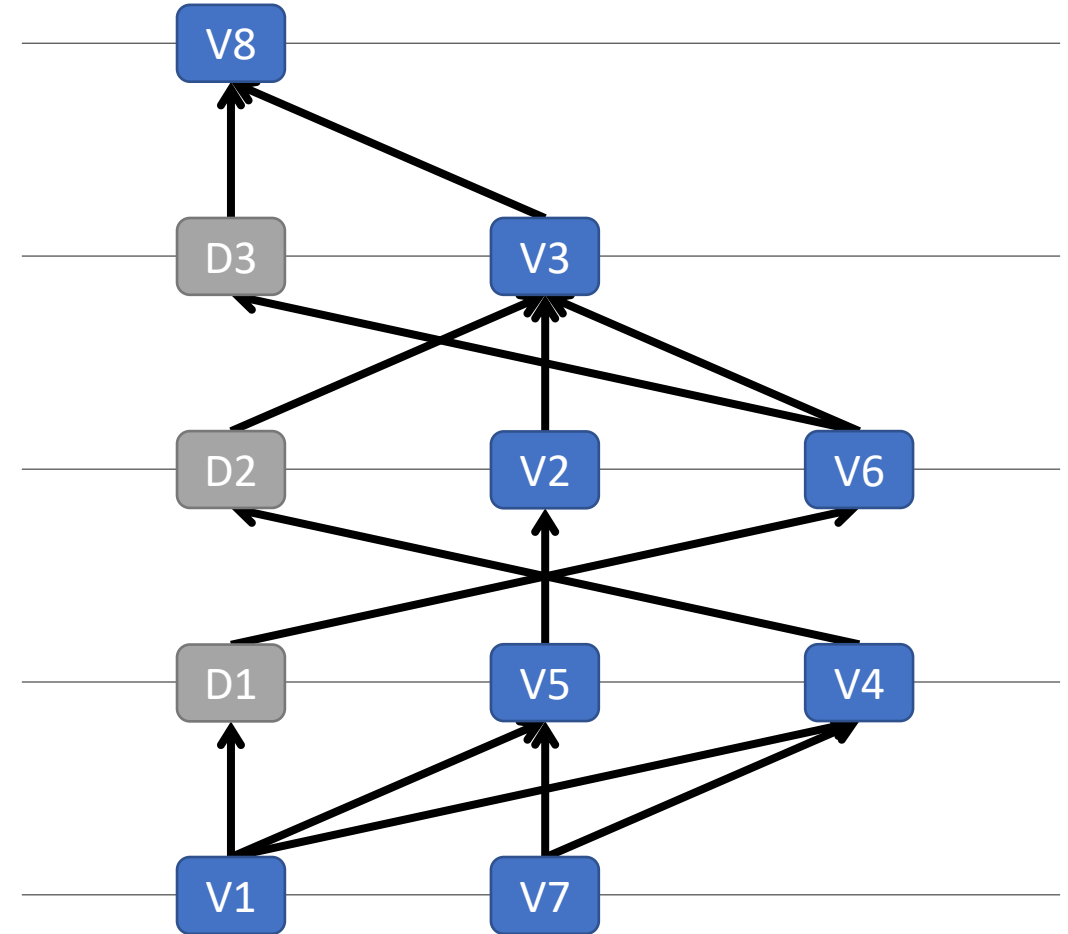
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik



Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik



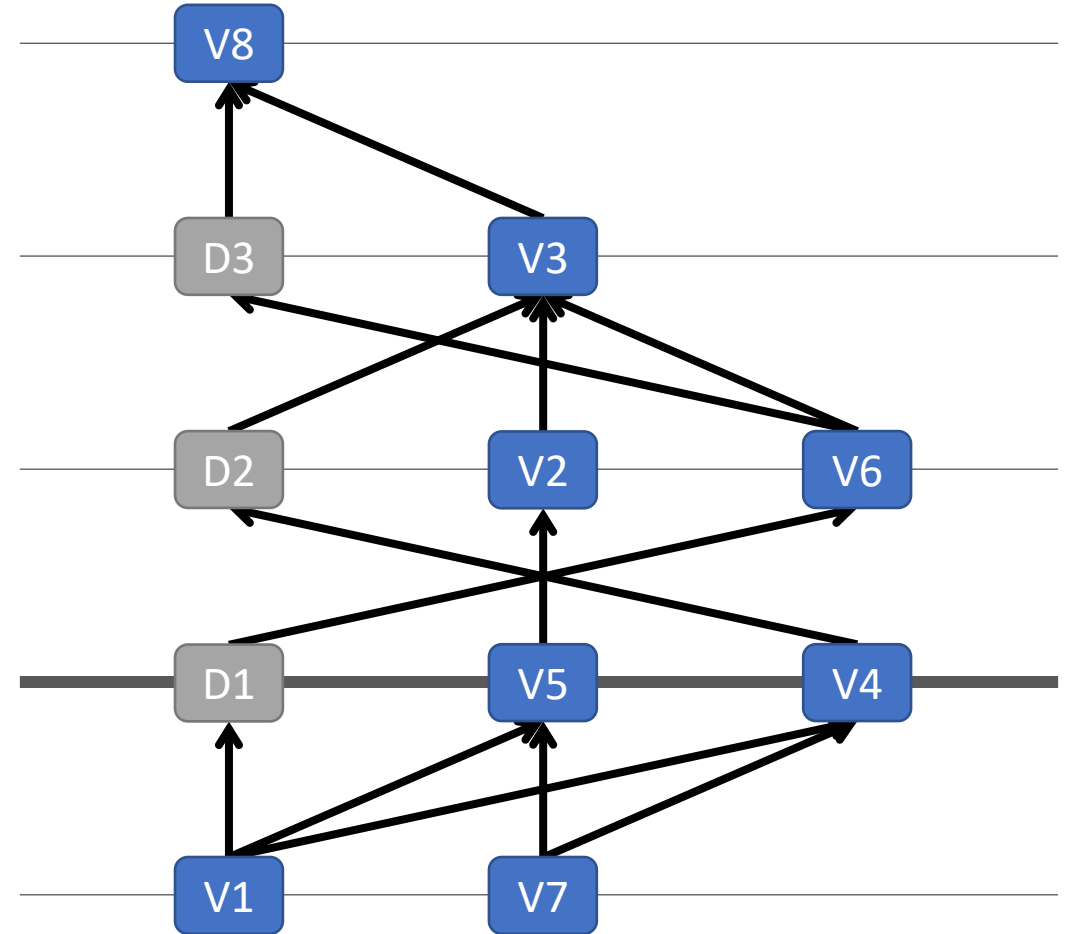
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik

$$bc(D1) = pos(V1)$$

$$bc(V5) = (pos(V1) + pos(V7)) / 2$$

$$bc(V4) = (pos(V1) + pos(V7)) / 2$$



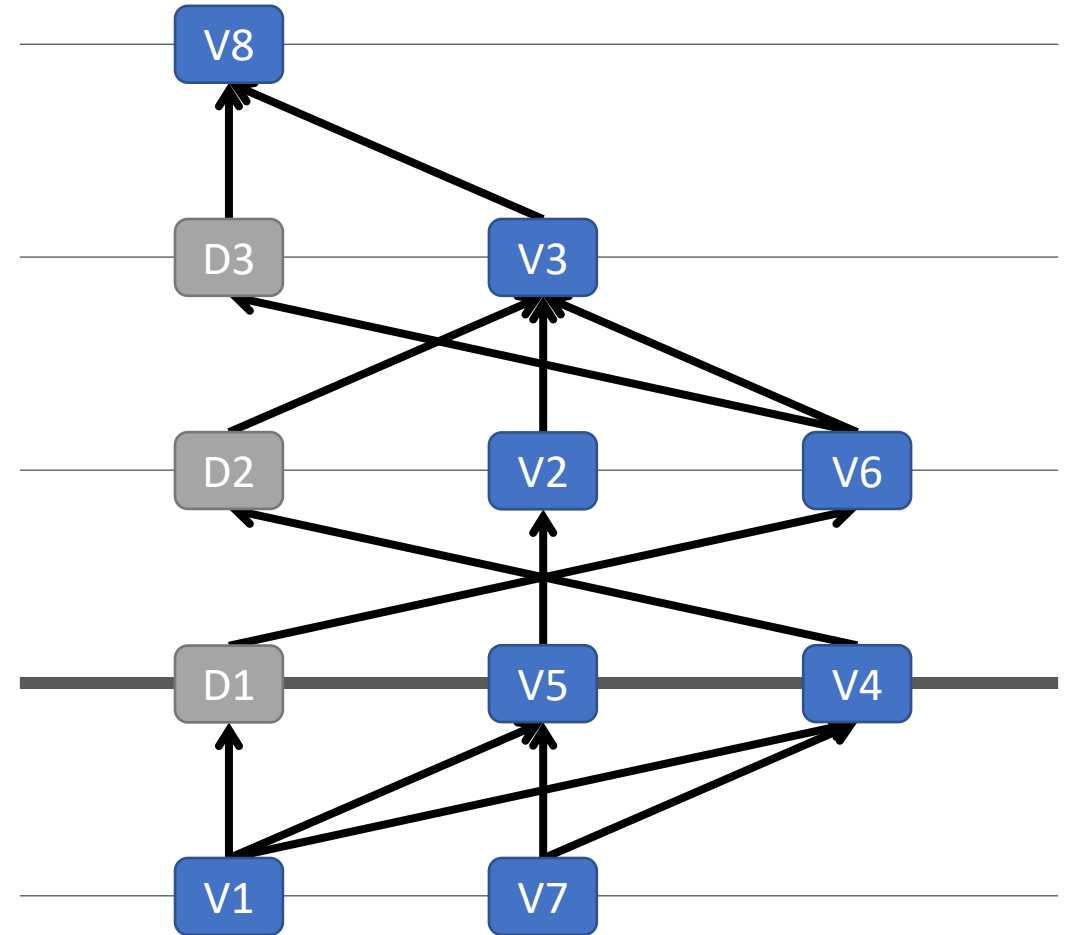
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik

$$bc(D1) = 1$$

$$bc(V5) = (1 + 2) / 2$$

$$bc(V4) = (1 + 2) / 2$$



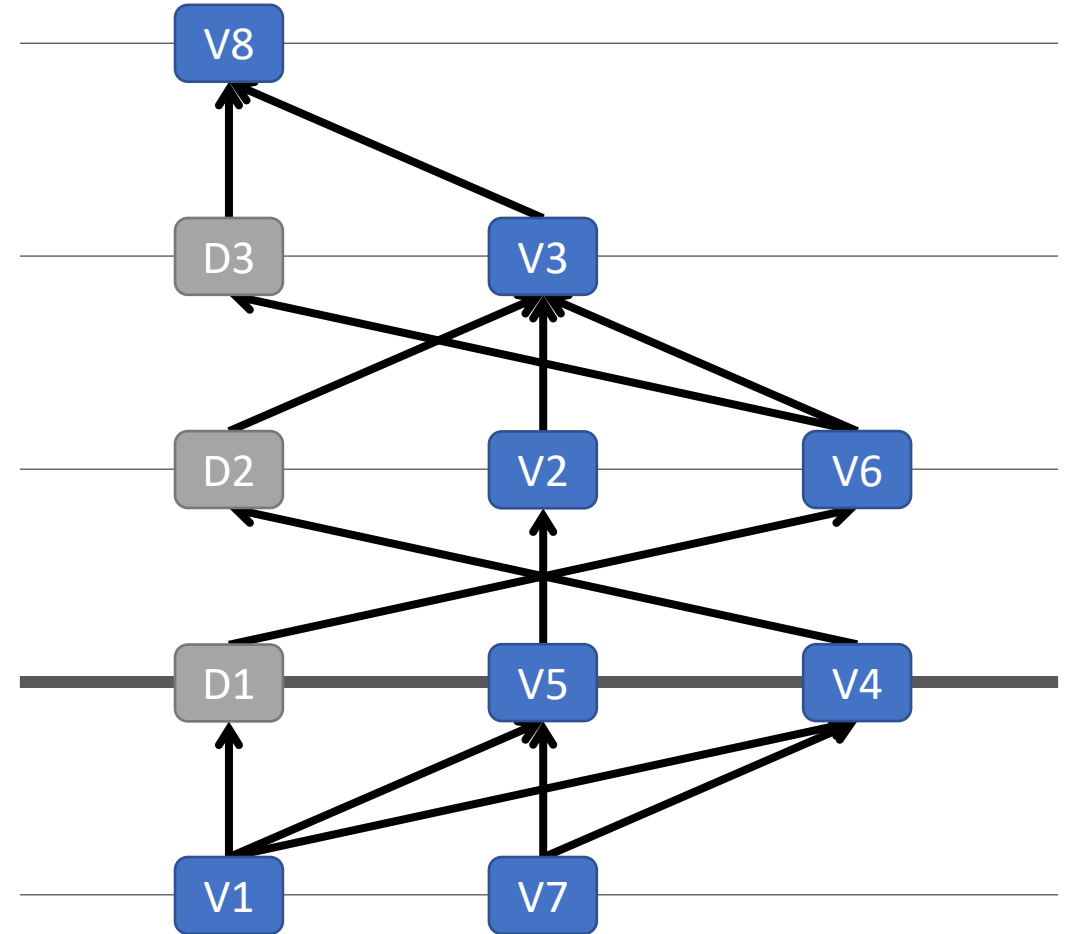
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik

$$bc(D1) = 1$$

$$bc(V5) = 1,5$$

$$bc(V4) = 1,5$$



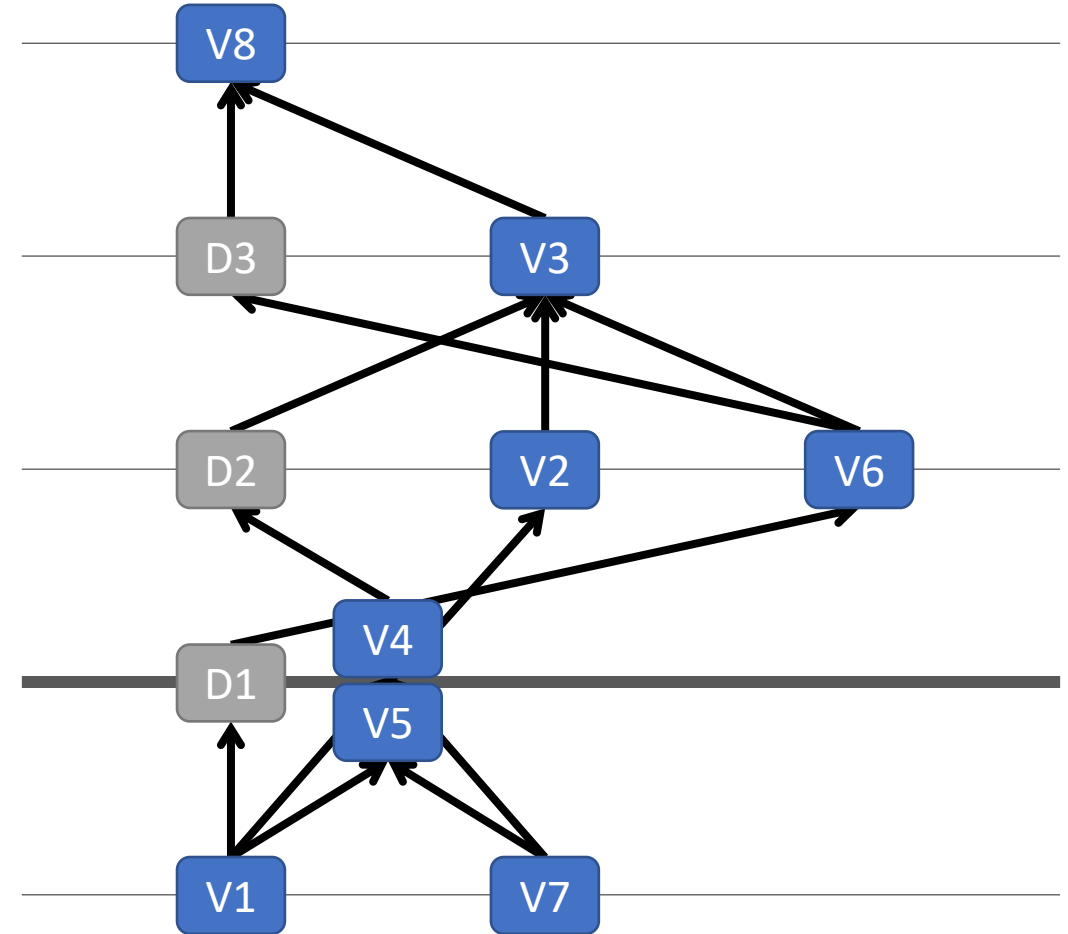
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik

$$bc(D1) = 1$$

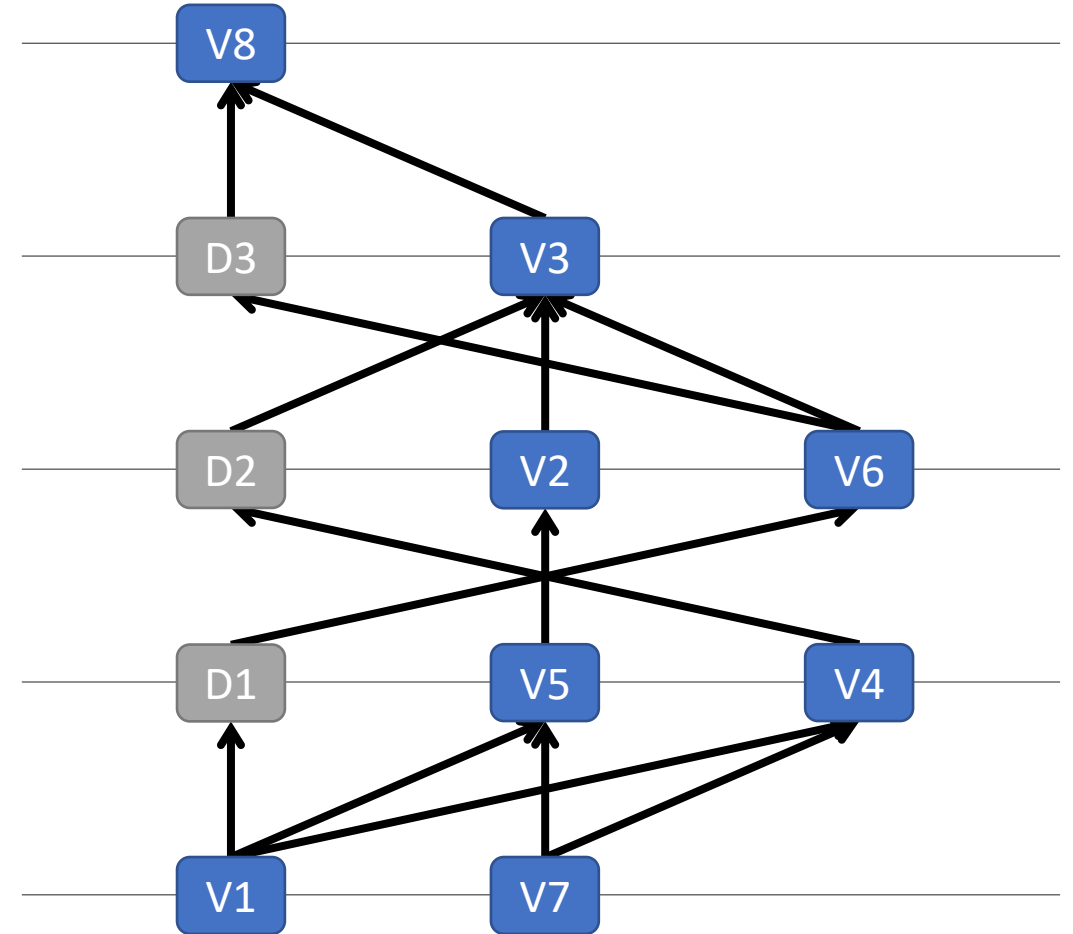
$$bc(V5) = 1,5$$

$$bc(V4) = 1,5$$



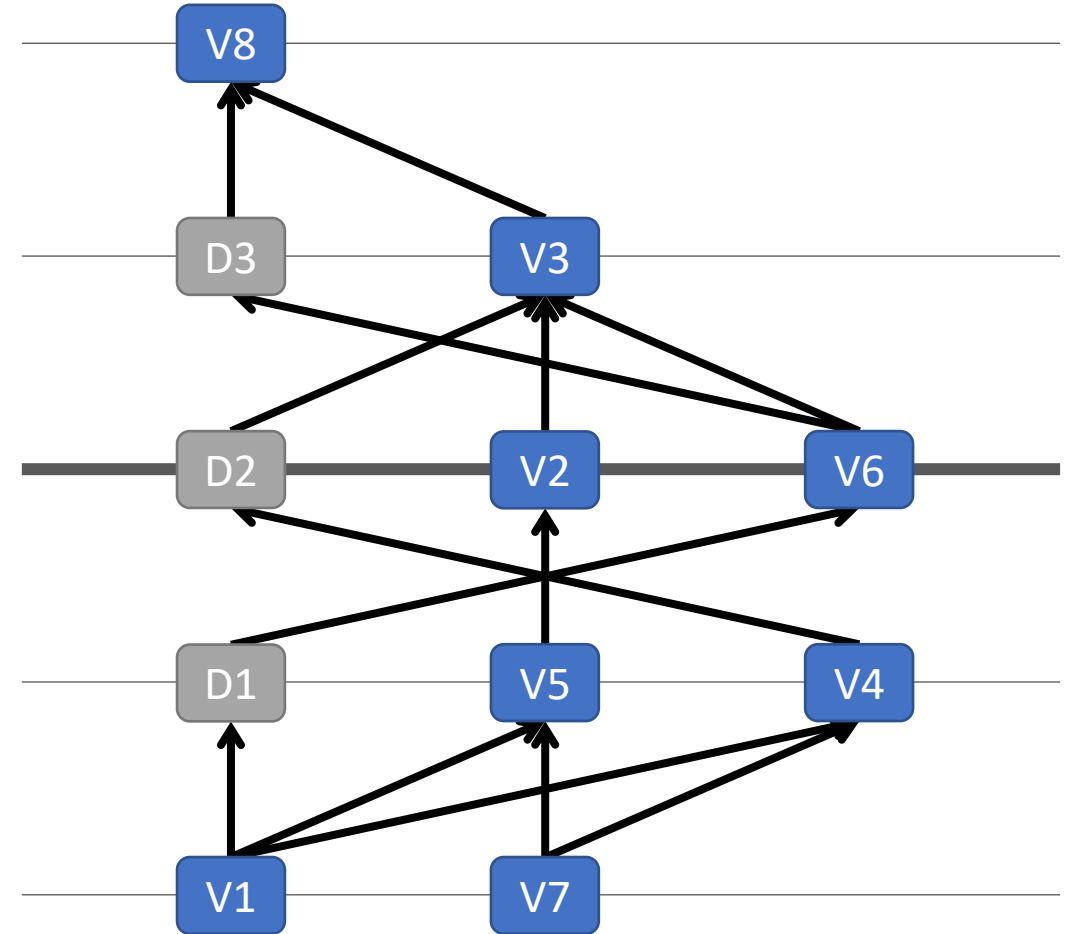
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik



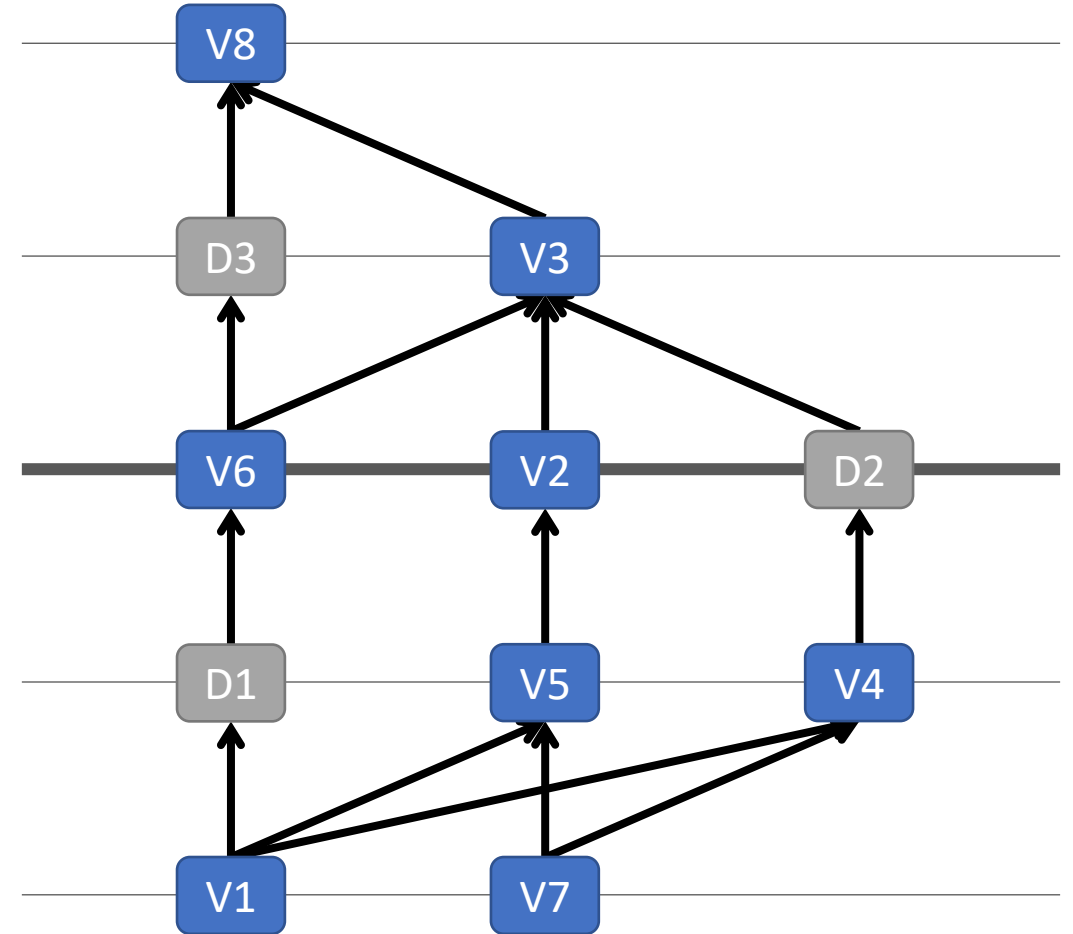
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik



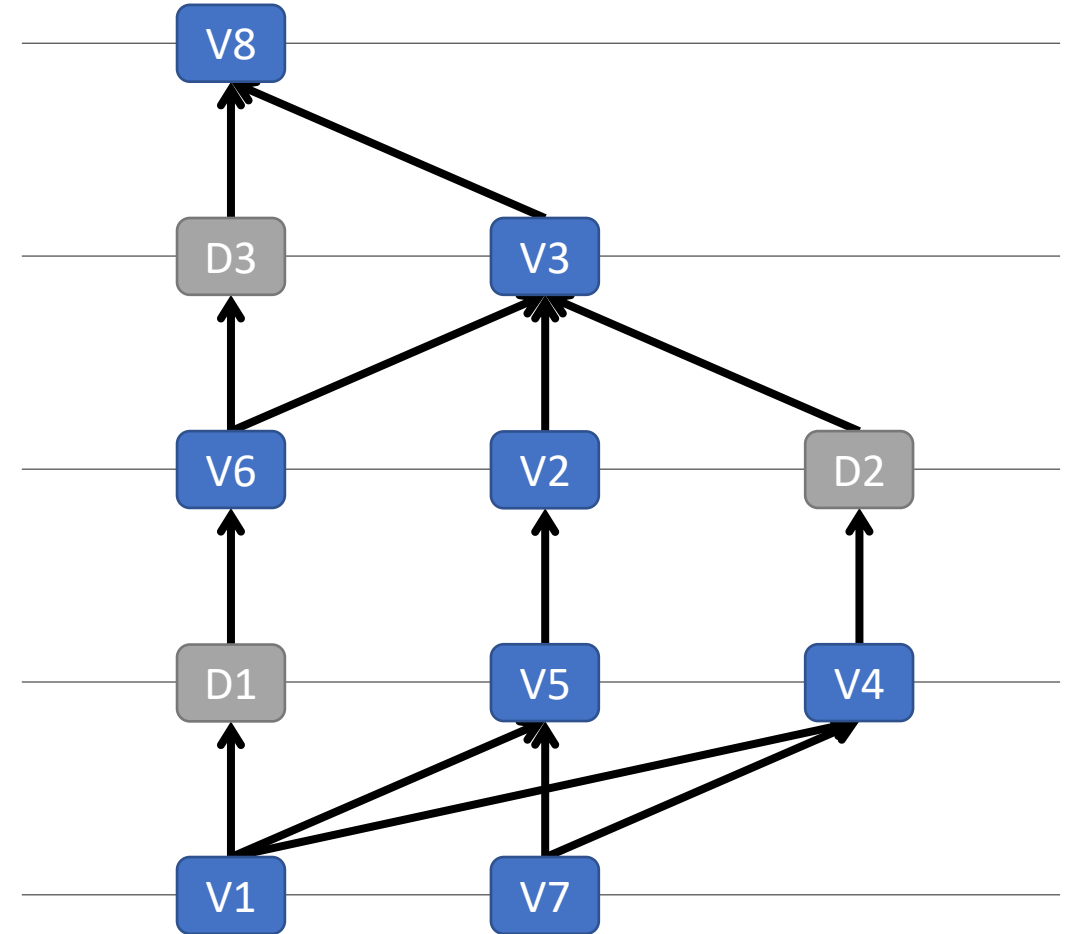
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik



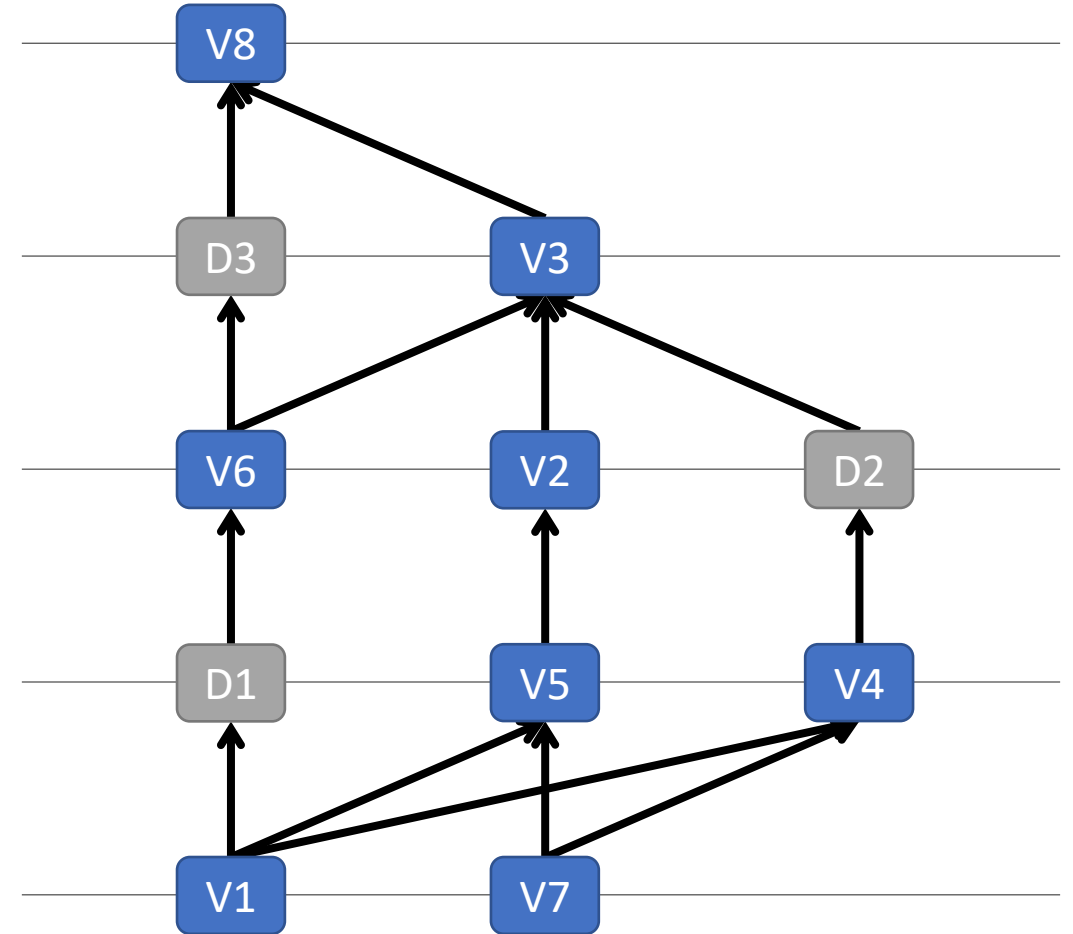
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik



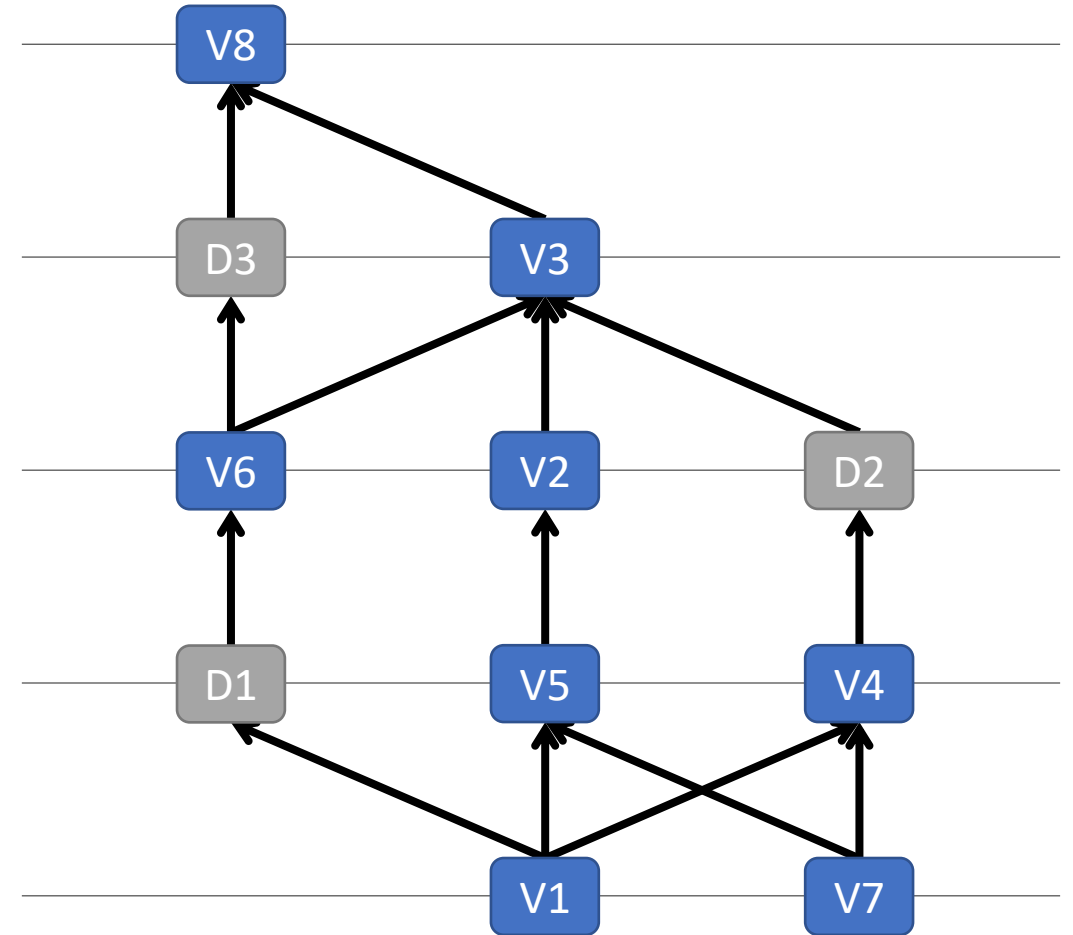
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
- 3: Knotenpositionierung
 - Knotenkoordinaten festlegen



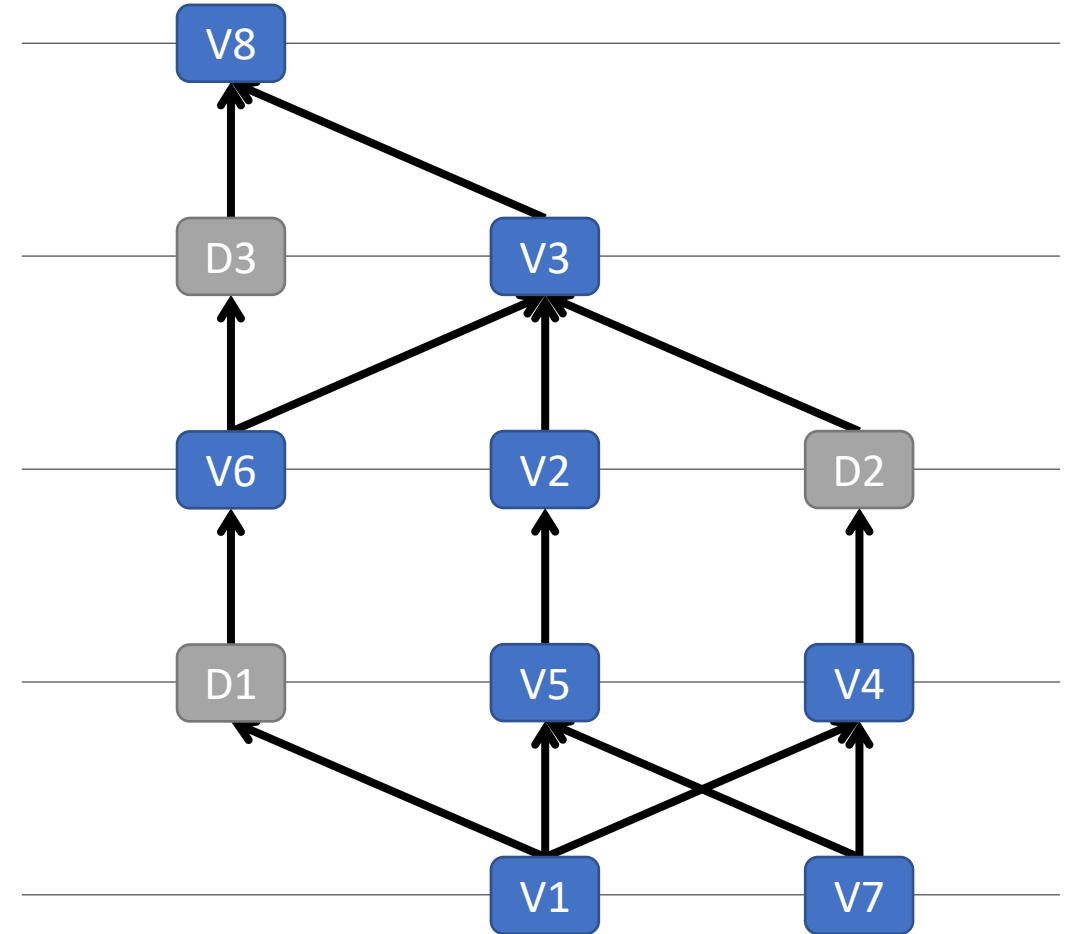
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
- 3: Knotenpositionierung
 - Knotenkoordinaten festlegen



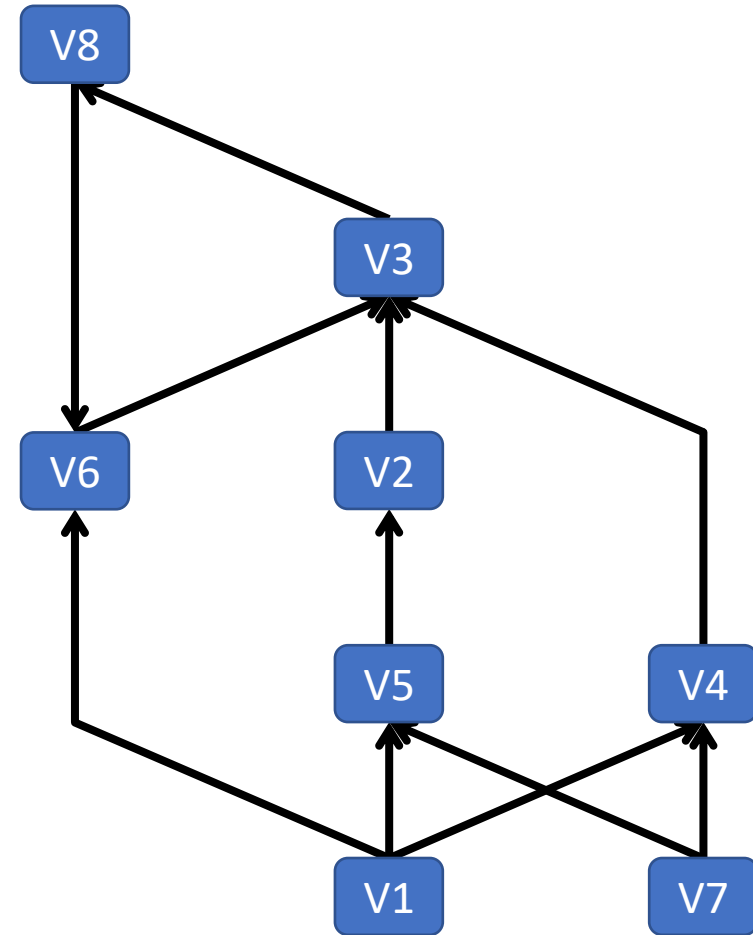
Sugiyama-Algorithmus

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
- 3: Knotenpositionierung
 - Knotenkoordinaten festlegen
- 4: Ausgabe
 - Zeichnen



Sugiyama-Algorithmus

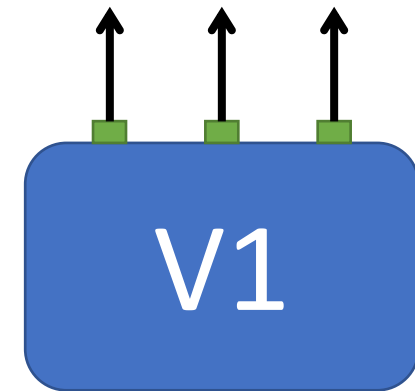
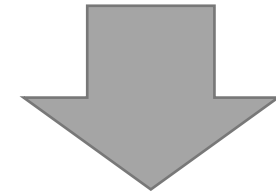
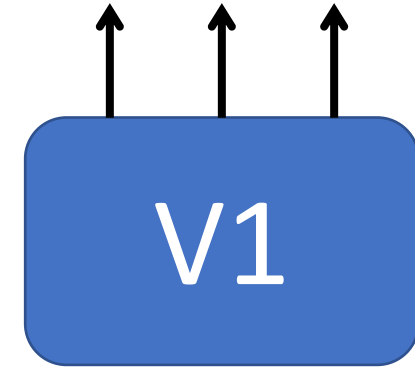
- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
- 3: Knotenpositionierung
 - Knotenkoordinaten festlegen
- 4: Ausgabe
 - Zeichnen



Drawing Layered Graphs with Port Constraints

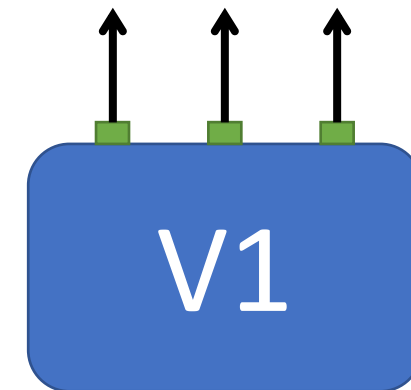
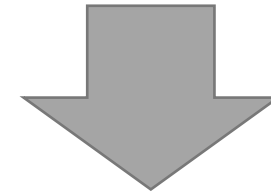
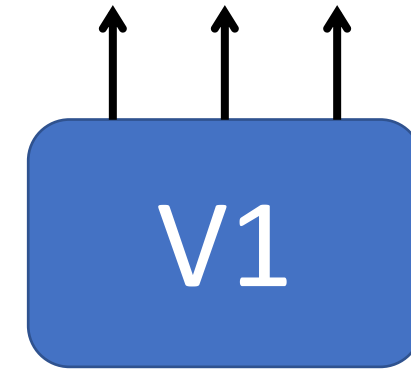
Christoph Daniel Schulze, Miro Spönemann, Reinhard von Hanxleden
[JVLC 2014]

Port Constraints



Port Constraints

- Knoten und Kanten sind über Ports verbunden
- Knoten – Rechtecke
- Port
 - Verbindungspunkt von Knoten und Kante
 - Punkte auf Rand der Knotenrechtecke
- Portreihenfolge beeinflusst Anzahl der Kreuzungen



Port Constraints

–

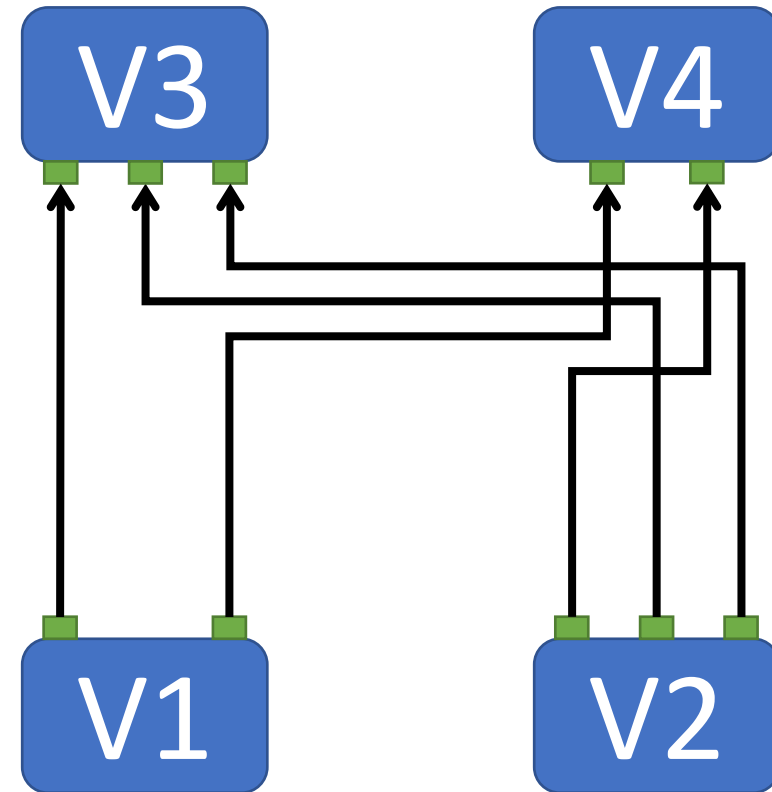
Sugiyama

- 1: Vorverarbeitung
 - 1a: Kreise auflösen
 - 1b: Lagenzuordnung & Dummyknoten
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
 - Unterstützung von Ports
- 3: Knotenpositionierung
 - Knotenkoordinaten festlegen
- 4: Kantenführung
 - orthogonale Kanten

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
- 3: Knotenpositionierung
 - Knotenkoordinaten festlegen
- 4: Ausgabe
 - zeichnen

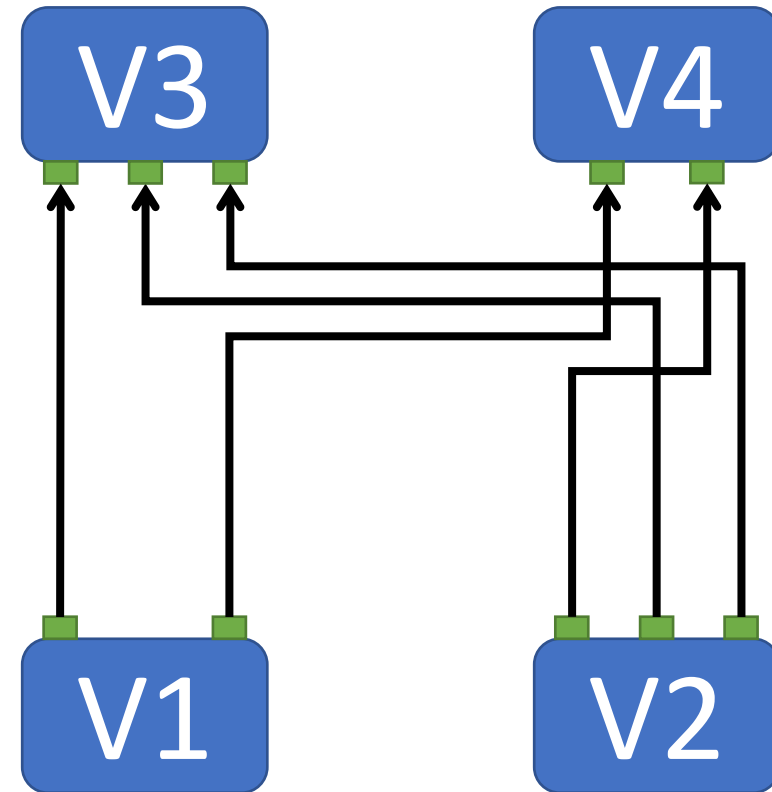
Port Constraints – Kreuzungsreduzierung

- Unterstützung von Ports



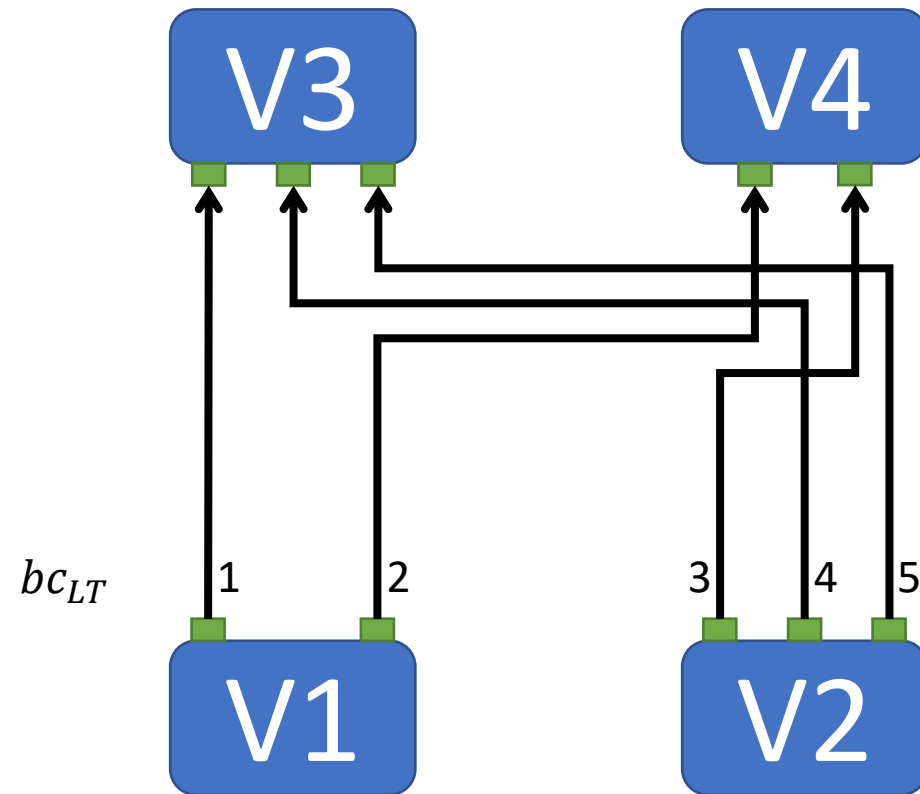
Port Constraints – Kreuzungsreduzierung

- Unterstützung von Ports
- Barycenter-Berechnung
 - *layer-total*-Methode
 - *node-relative*-Methode



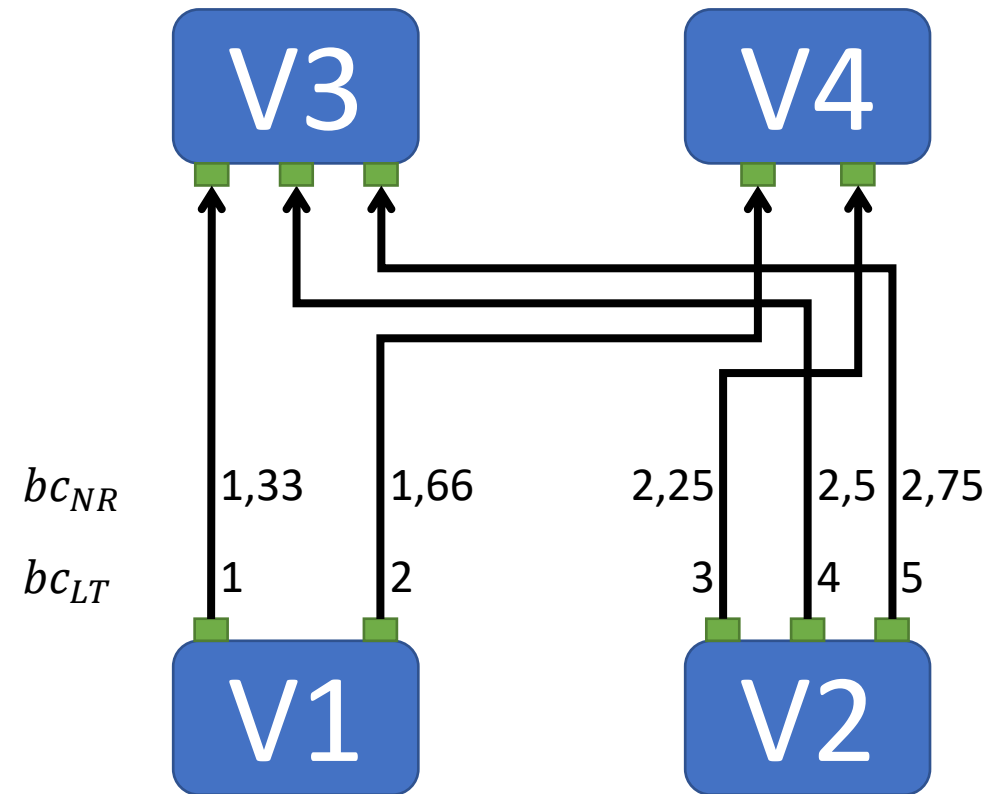
Port Constraints – Kreuzungsreduzierung

- Unterstützung von Ports
- Barycenter-Berechnung
 - *layer-total-Methode*
 - *node-relative-Methode*



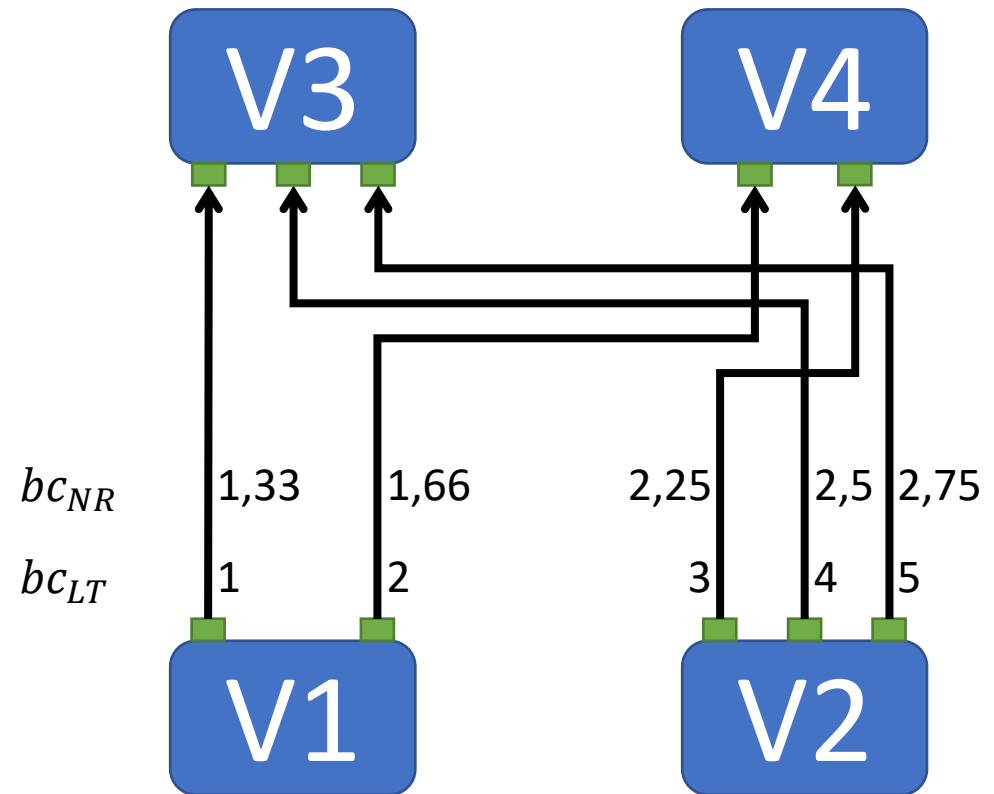
Port Constraints – Kreuzungsreduzierung

- Unterstützung von Ports
- Barycenter-Berechnung
 - *layer-total*-Methode
 - ***node-relative***-Methode



Port Constraints – Kreuzungsreduzierung

- Unterstützung von Ports
- Barycenter-Berechnung
 - *layer-total*-Methode
 - *node-relative*-Methode
- Keine Methode grundsätzlich besser
- Abhängig von Datensatz

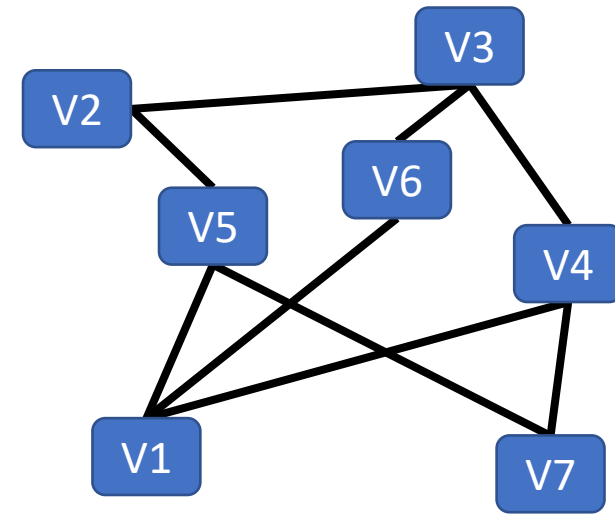


Eine Variante des Sugiyama-Algorithmus für ungerichtete Graphen mit Portconstraints

Julian Walter

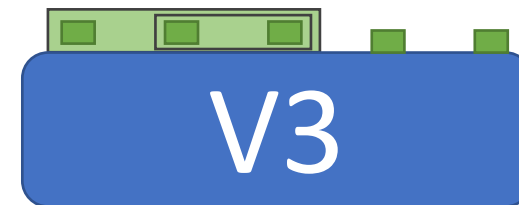
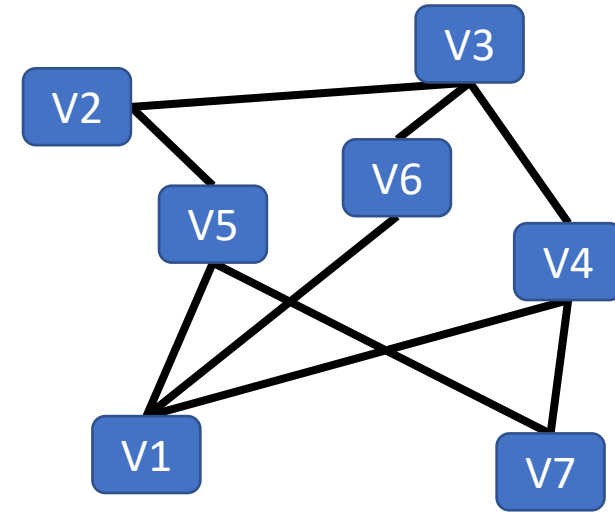
Besonderheiten

- Eingabegraph ungerichtet



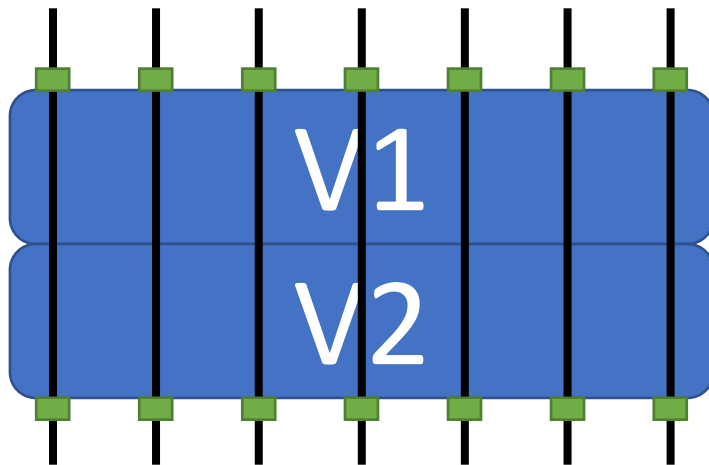
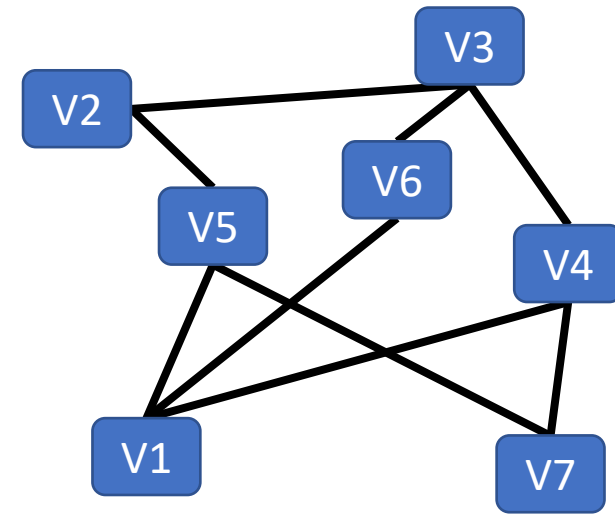
Besonderheiten

- Eingabegraph ungerichtet
- Ports und Portgruppen



Besonderheiten

- Eingabegraph ungerichtet
- Ports und Portgruppen
- Steckverbindungen



Algorithmus

–

Sugiyama

- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
- 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
- 3: Knotenpositionierung
 - Knotenkoordinaten festlegen
- 4: Ausgabe
 - zeichnen

Algorithmus

—

Sugiyama

- 0: Vorverarbeitung
 - 1: Richtungszuweisung
 - 2: Lagenzuordnung
 - 3: Dummyknotenerstellung
 - 4: Kreuzungsreduzierung
 - 5: Knotenpositionierung
 - 6: Kantenführung
- 1: Vorverarbeitung
 - gerichtete Kreise auflösen
 - Knoten in Lagen einteilen
 - Dummyknoten hinzufügen
 - 2: Kreuzungsreduzierung
 - Layer-Sweep mit Barycenter-Heuristik
 - 3: Knotenpositionierung
 - Knotenkoordinaten festlegen
 - 4: Ausgabe
 - zeichnen
-

Algorithmus

- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

Algorithmus – Vorverarbeitung

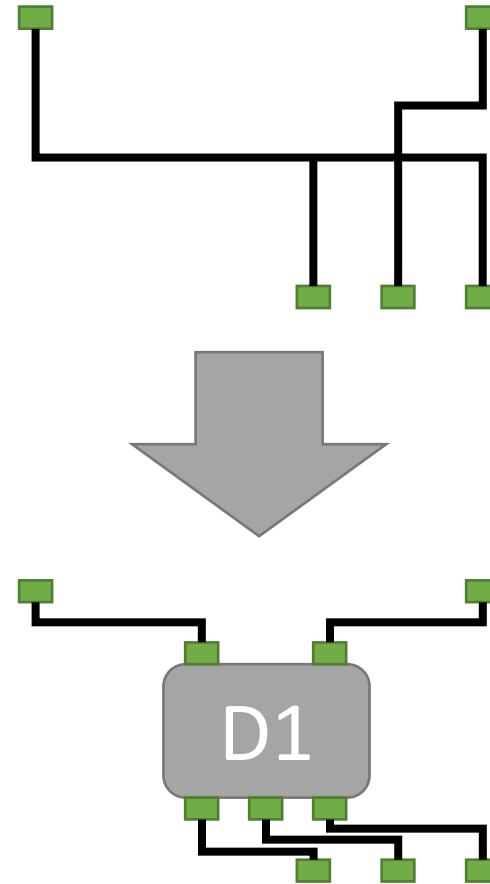
- Eingabe:
 - Kanten mit mehreren Ports
 - Knotengruppen
 - Ports mit mehreren Kanten
 - nicht zusammenhängend

Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - Knotengruppen
 - Ports mit mehreren Kanten
 - nicht zusammenhängend
- Ausgabe:
 - Graph aus Knoten, Kanten und Ports
 - Zusammenhängend
 - Genau eine Kante je Port
 - Genau zwei Ports je Kante
 - Portgruppen
 - Portpaare

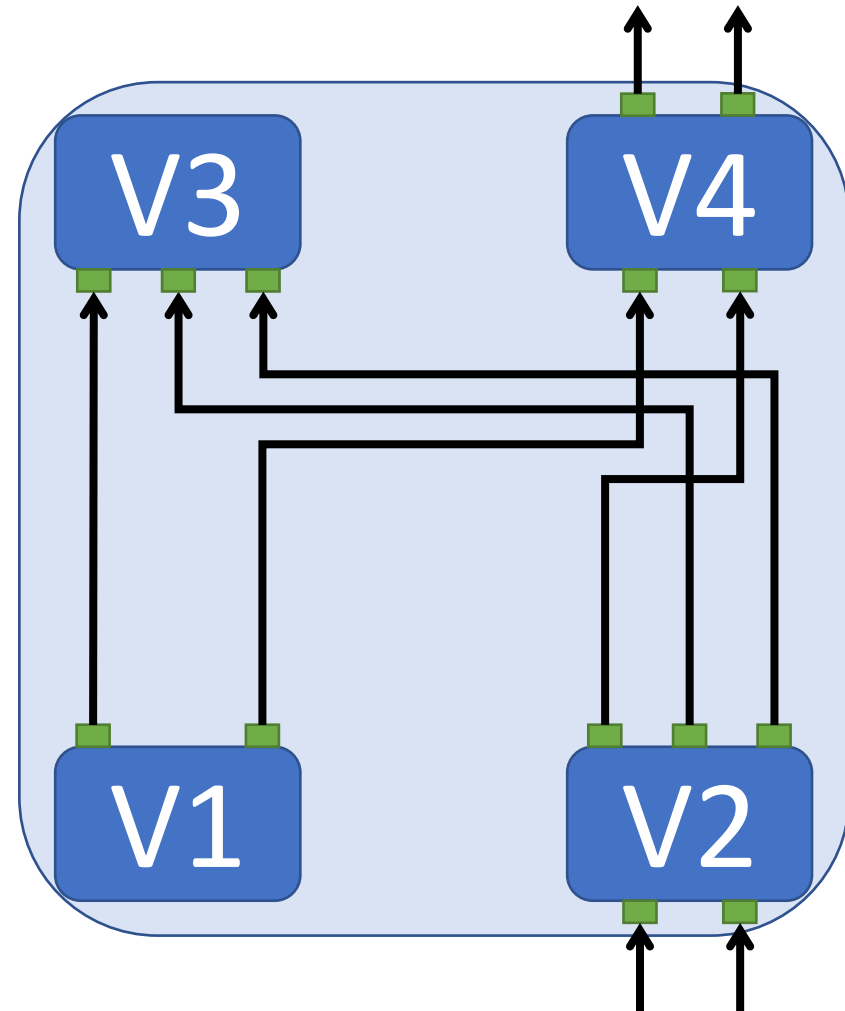
Algorithmus – Vorverarbeitung

- Eingabe:
 - **Kanten mit mehreren Ports**
 - Knotengruppen
 - Ports mit mehreren Kanten
 - nicht zusammenhängend



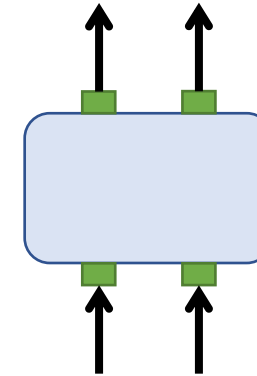
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - **Knotengruppen**
 - Ports mit mehreren Kanten
 - nicht zusammenhängend



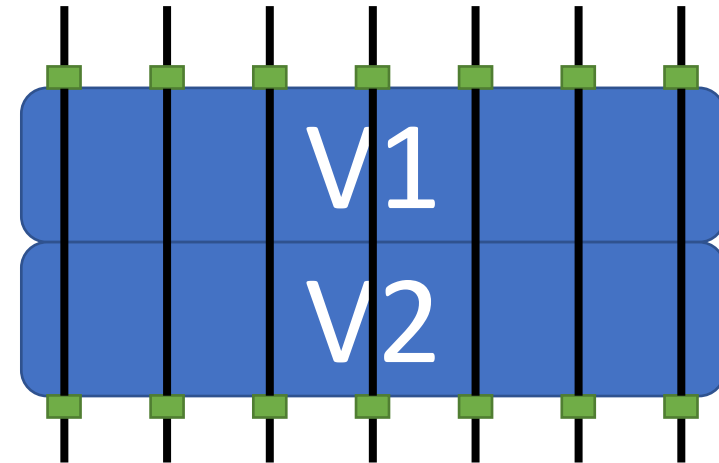
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - **Knotengruppen**
 - Ports mit mehreren Kanten
 - nicht zusammenhängend



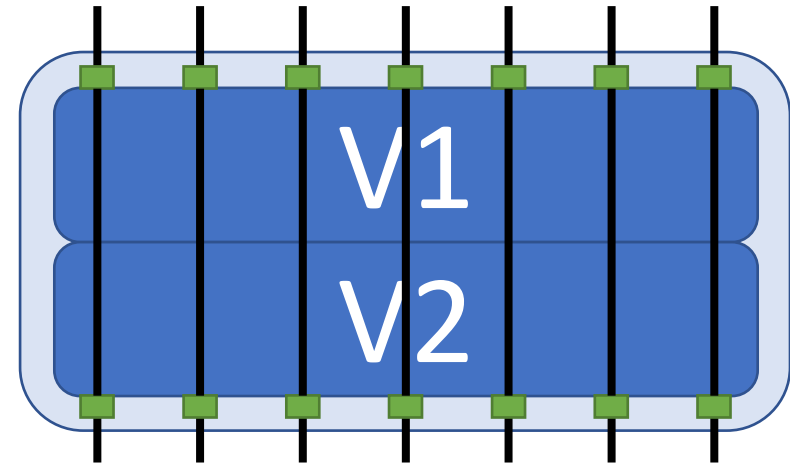
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - **Knotengruppen**
 - TouchingPair
 - PortPairing
 - Ports mit mehreren Kanten
 - nicht zusammenhängend



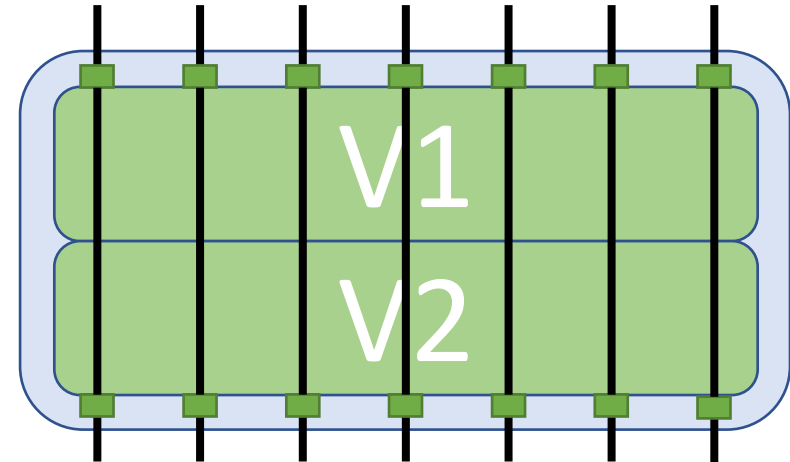
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - **Knotengruppen**
 - TouchingPair
 - PortPairing
 - Ports mit mehreren Kanten
 - nicht zusammenhängend



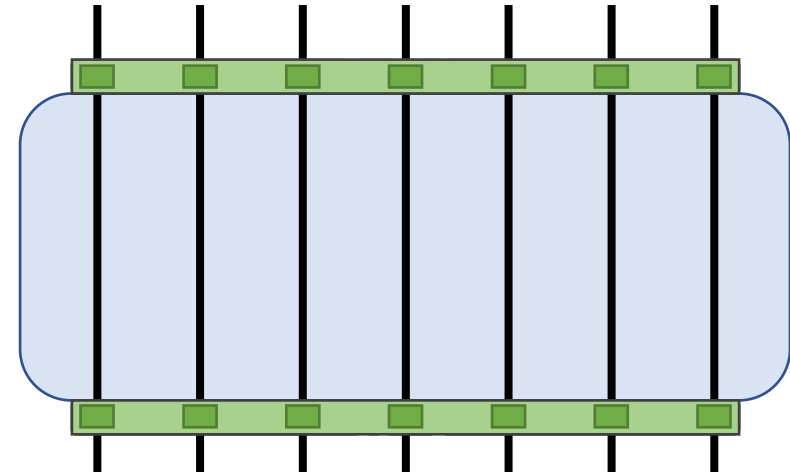
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - **Knotengruppen**
 - TouchingPair
 - PortPairing
 - Ports mit mehreren Kanten
 - nicht zusammenhängend



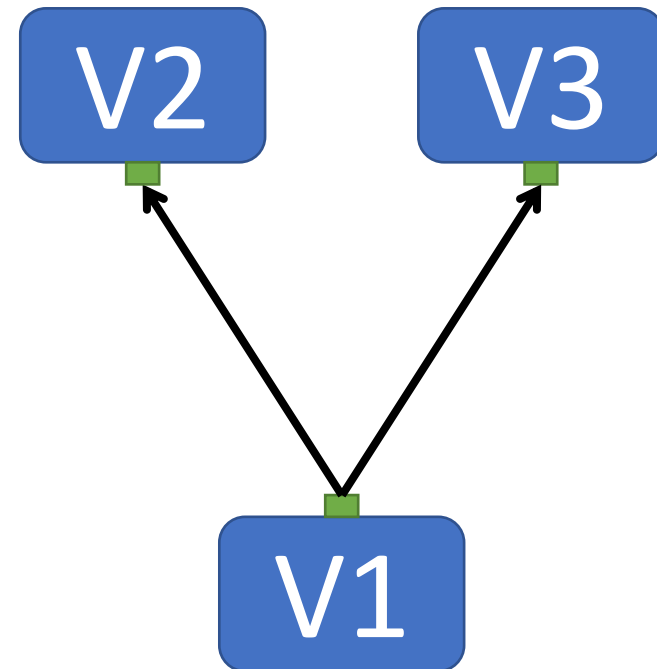
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - **Knotengruppen**
 - TouchingPair
 - PortPairing
 - Ports mit mehreren Kanten
 - nicht zusammenhängend



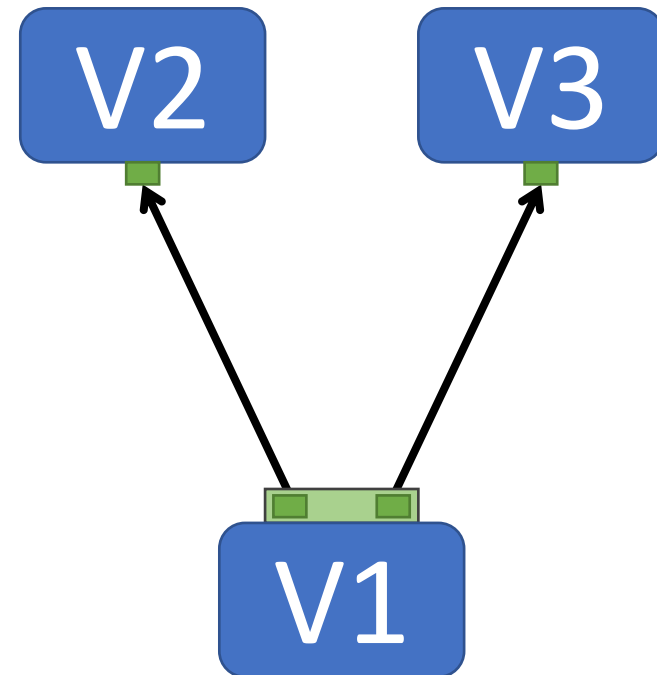
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - Knotengruppen
 - **Ports mit mehreren Kanten**
 - nicht zusammenhängend



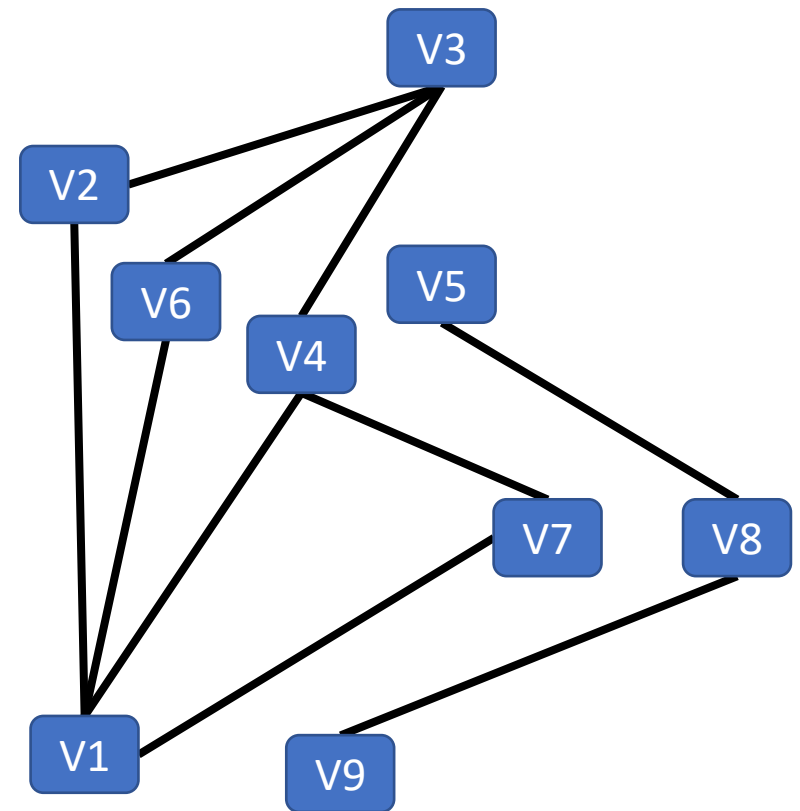
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - Knotengruppen
 - **Ports mit mehreren Kanten**
 - nicht zusammenhängend



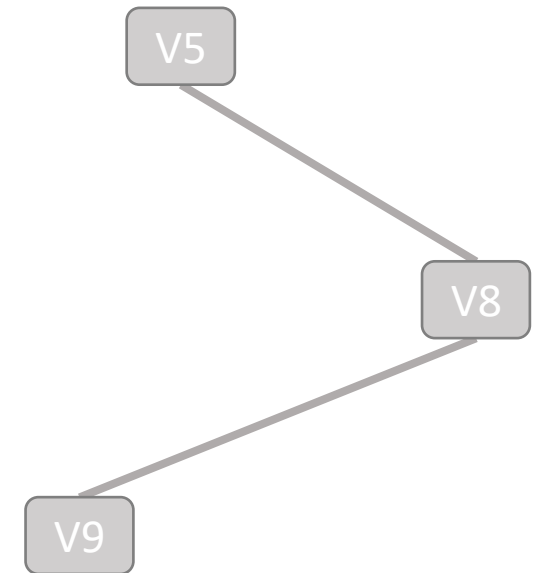
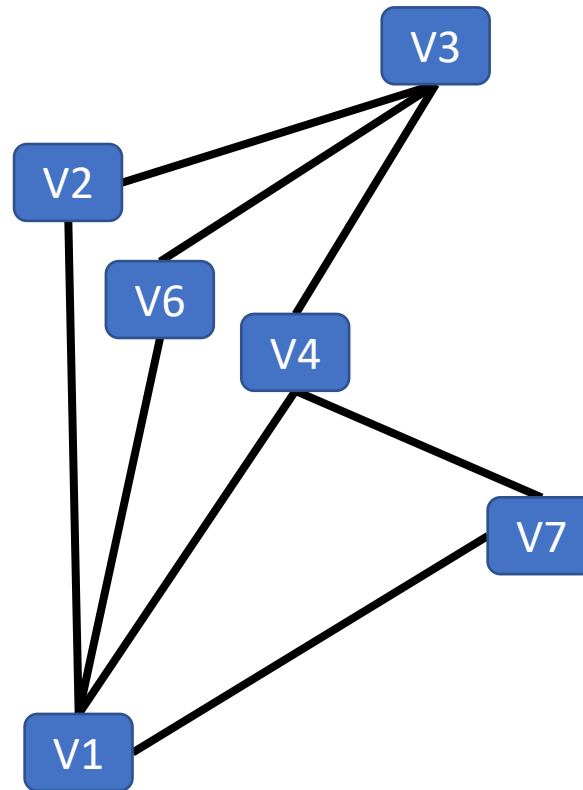
Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - Knotengruppen
 - Ports mit mehreren Kanten
 - **nicht zusammenhängend**



Algorithmus – Vorverarbeitung

- Eingabe:
 - Kanten mit mehreren Ports
 - Knotengruppen
 - Ports mit mehreren Kanten
 - **nicht zusammenhängend**

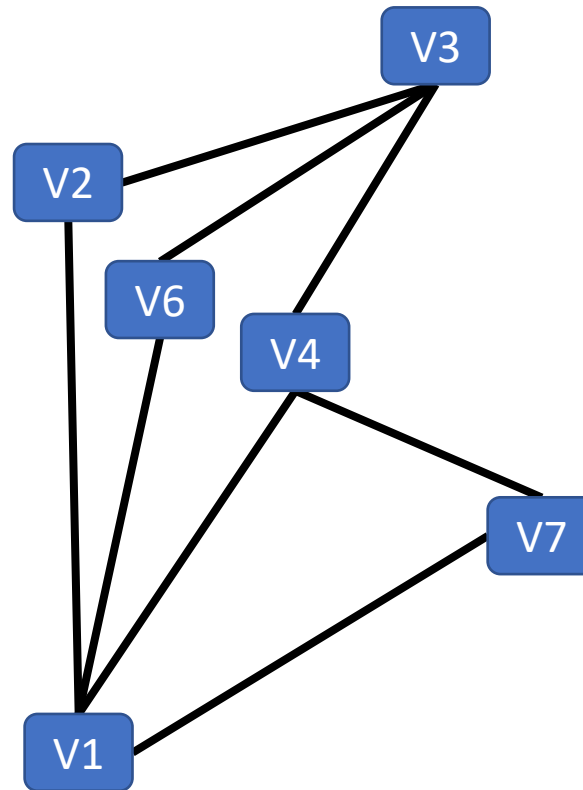


Algorithmus

- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

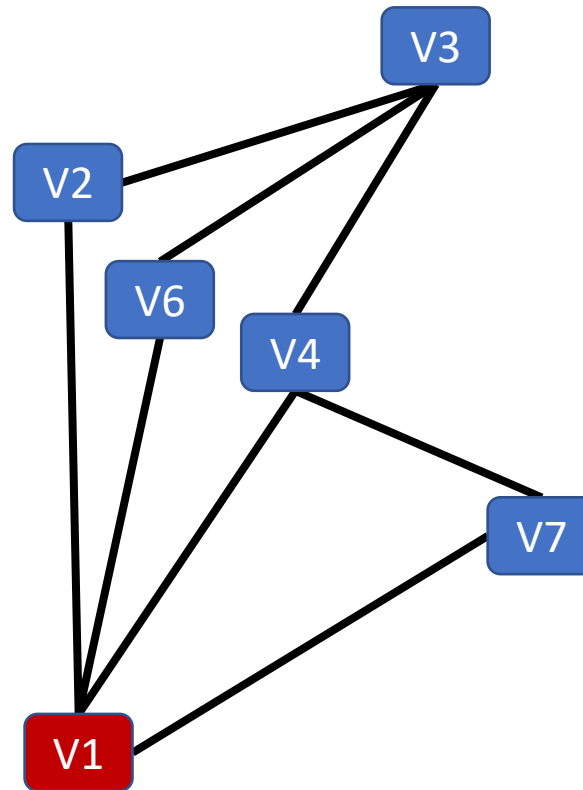
Algorithmus – Richtungszuweisung

- Breitensuche
- Kräftebasiert
- Zufällig



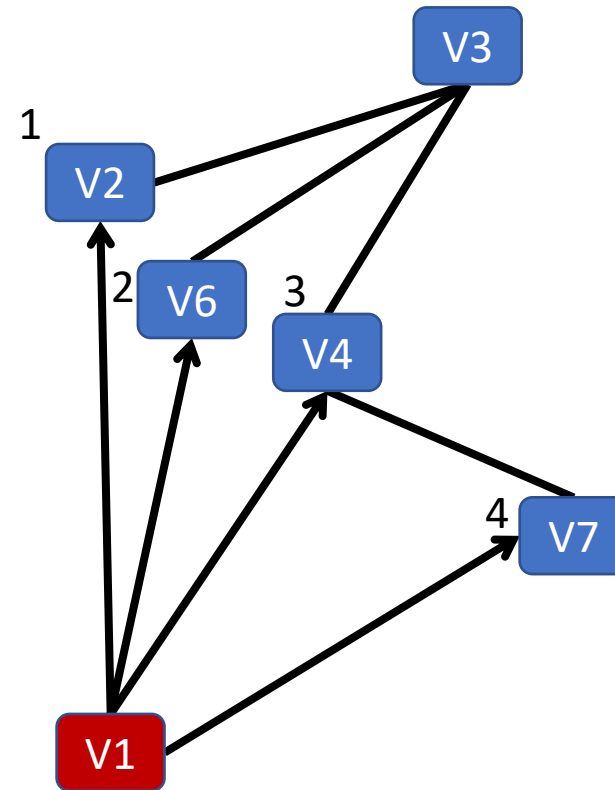
Algorithmus – Richtungszuweisung

- **Breitensuche**
- Kräftebasiert
- Zufällig



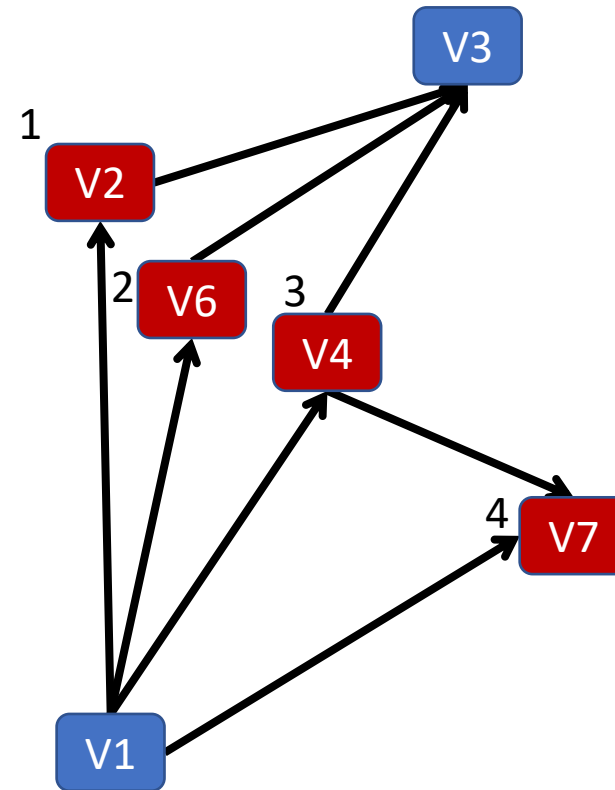
Algorithmus – Richtungszuweisung

- **Breitensuche**
- Kräftebasiert
- Zufällig



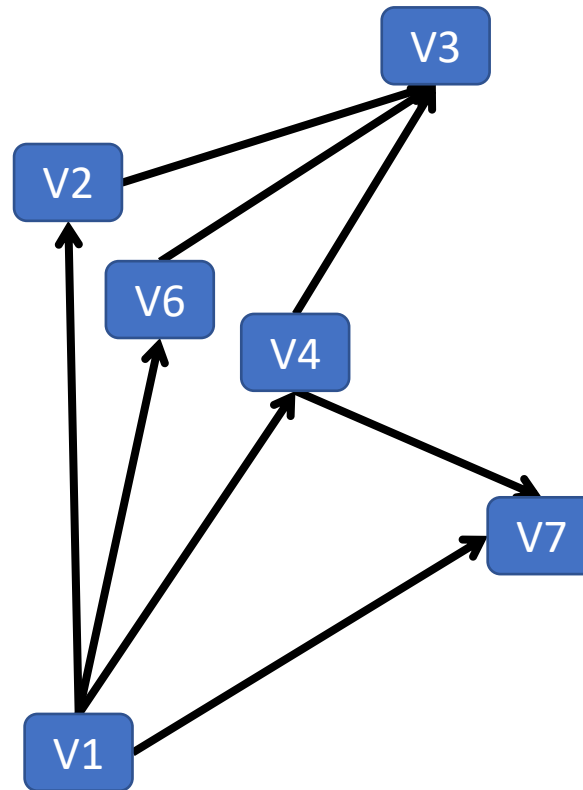
Algorithmus – Richtungszuweisung

- **Breitensuche**
- Kräftebasiert
- Zufällig



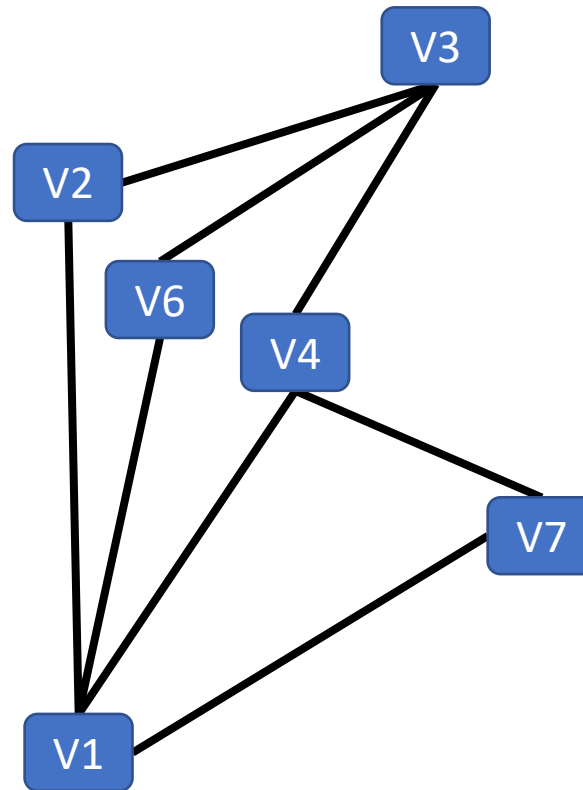
Algorithmus – Richtungszuweisung

- **Breitensuche**
- Kräftebasiert
- Zufällig



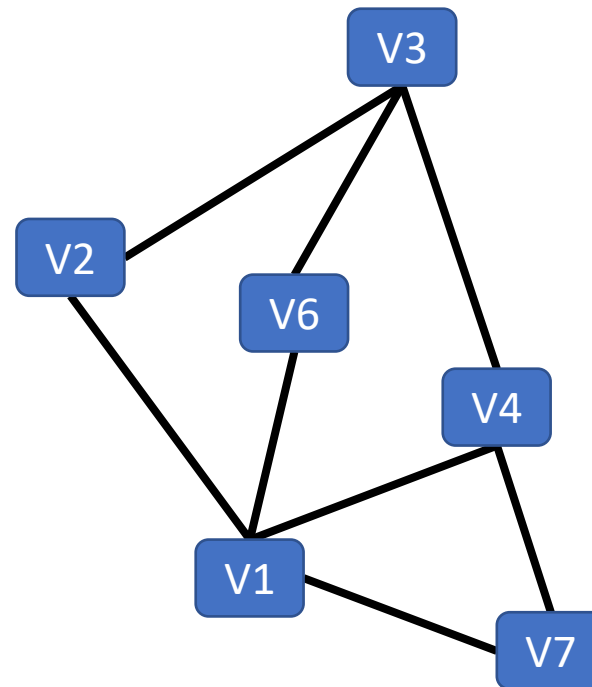
Algorithmus – Richtungszuweisung

- Breitensuche
- **Kräftebasiert**
- Zufällig



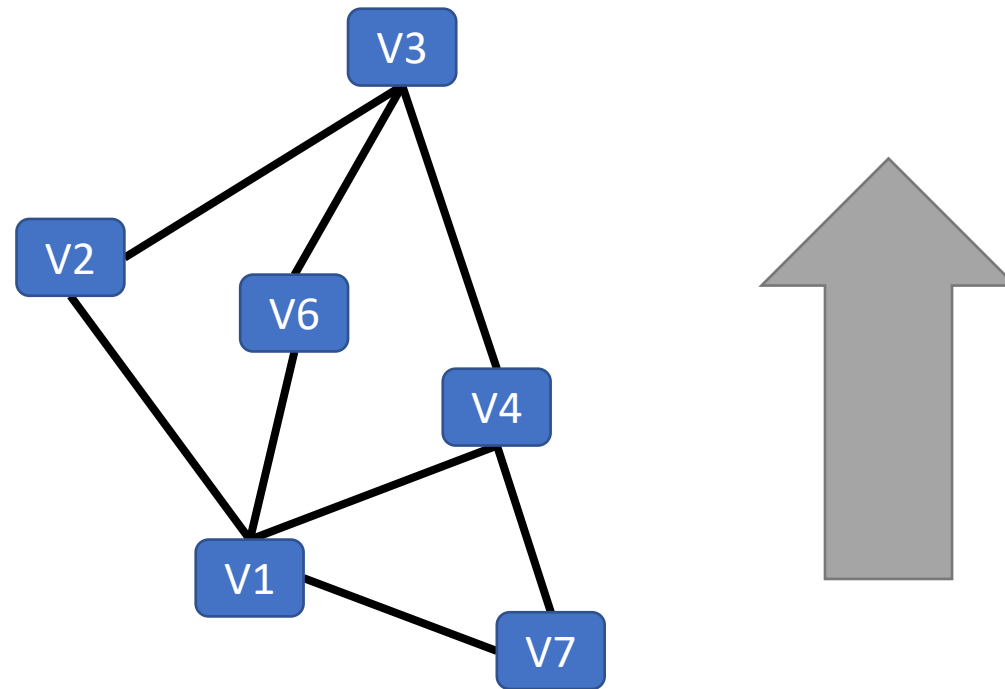
Algorithmus – Richtungszuweisung

- Breitensuche
- **Kräftebasiert**
- Zufällig



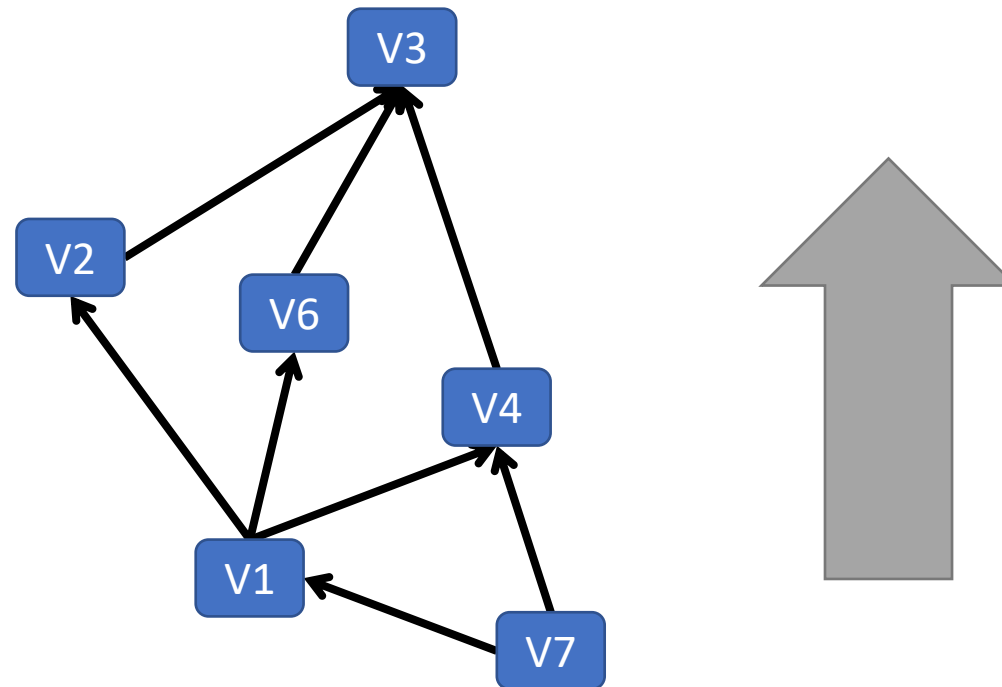
Algorithmus – Richtungszuweisung

- Breitensuche
- **Kräftebasiert**
- Zufällig



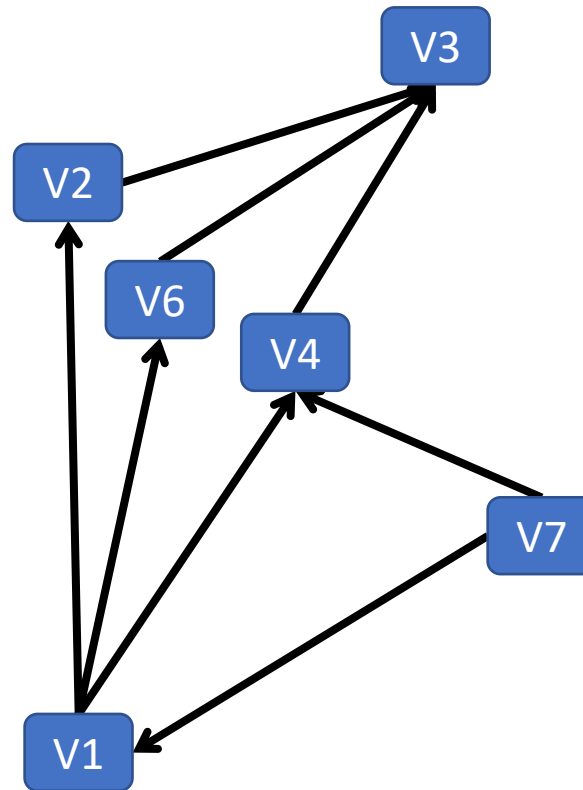
Algorithmus – Richtungszuweisung

- Breitensuche
- **Kräftebasiert**
- Zufällig



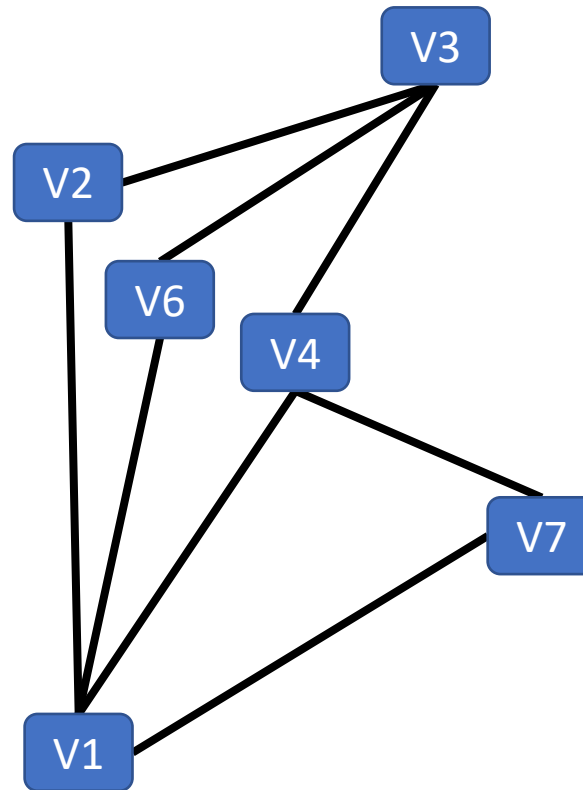
Algorithmus – Richtungszuweisung

- Breitensuche
- **Kräftebasiert**
- Zufällig



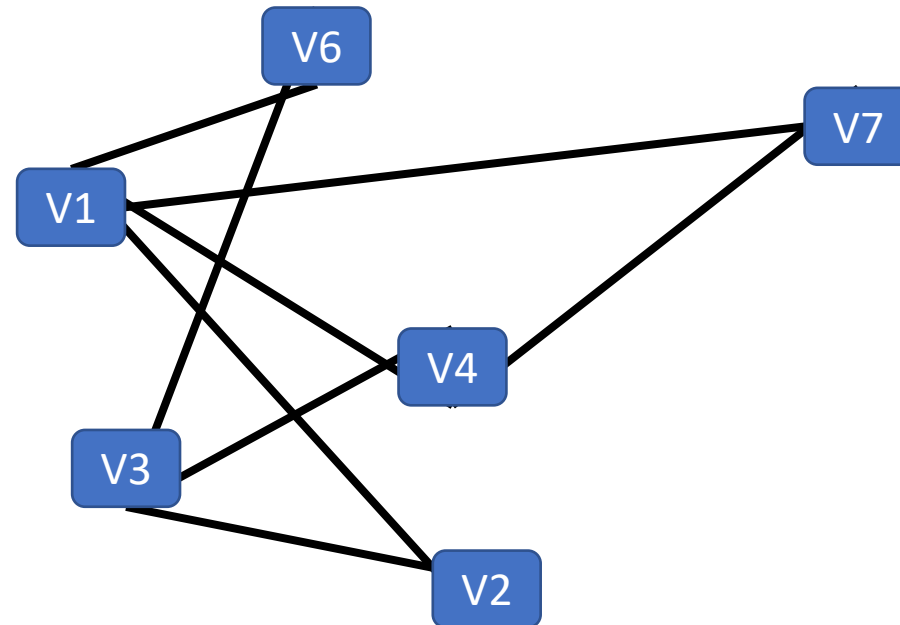
Algorithmus – Richtungszuweisung

- Breitensuche
- Kräftebasiert
- **Zufällig**



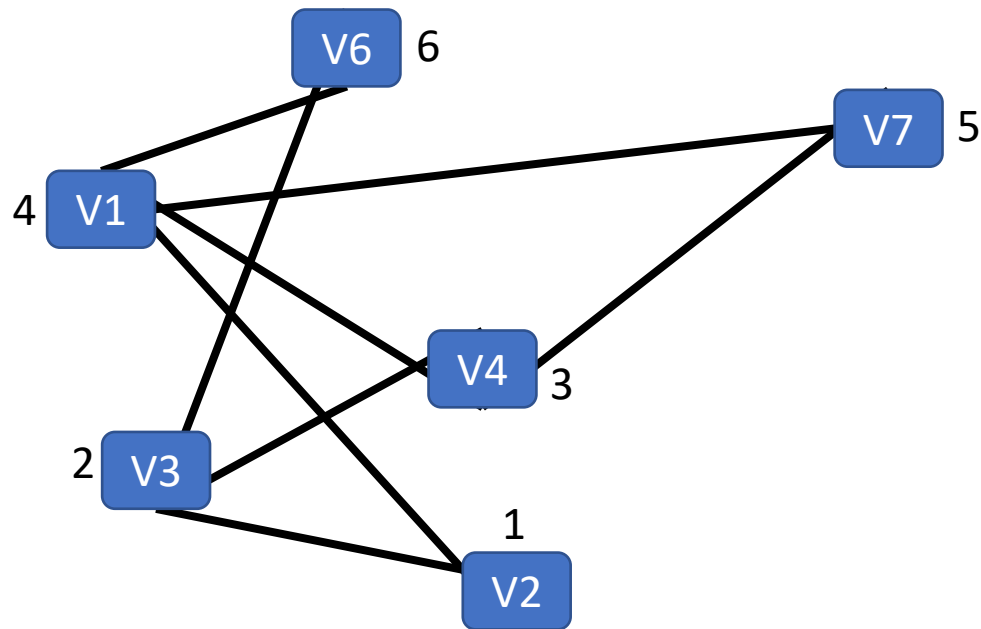
Algorithmus – Richtungszuweisung

- Breitensuche
- Kräftebasiert
- **Zufällig**



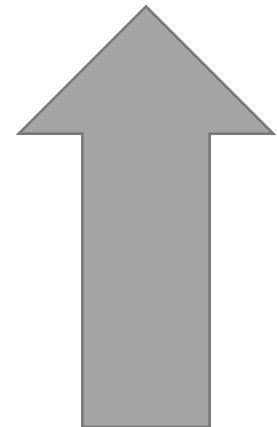
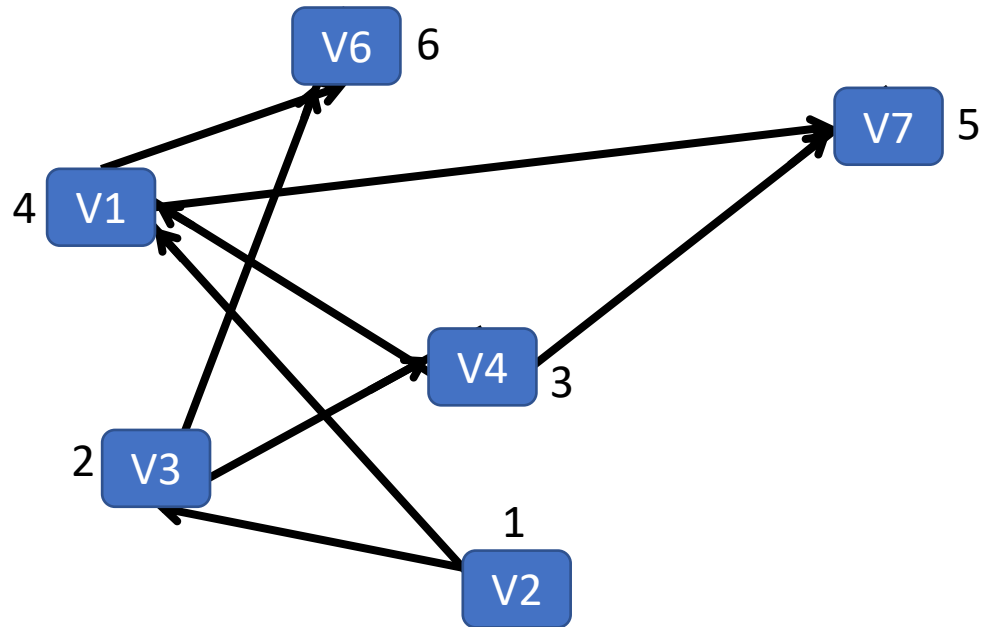
Algorithmus – Richtungszuweisung

- Breitensuche
- Kräftebasiert
- **Zufällig**



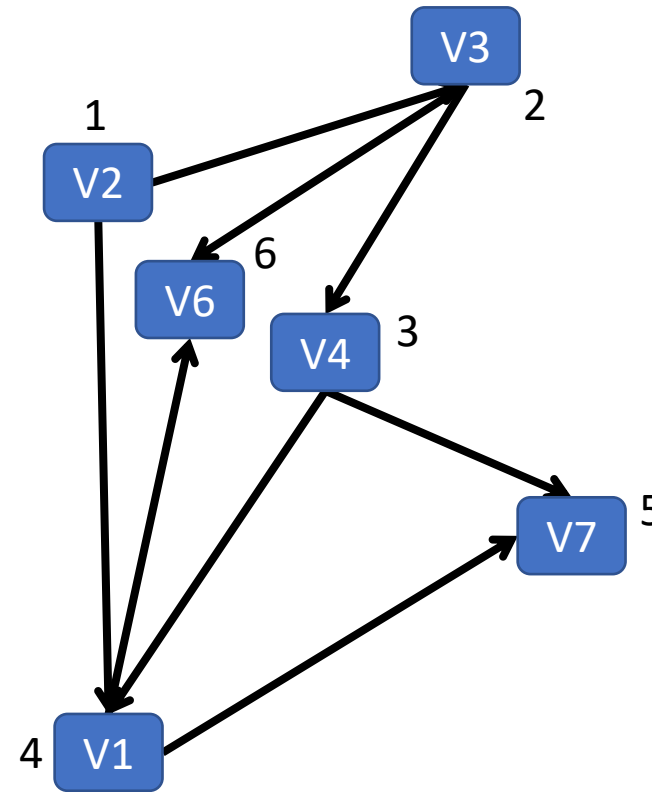
Algorithmus – Richtungszuweisung

- Breitensuche
- Kräftebasiert
- **Zufällig**



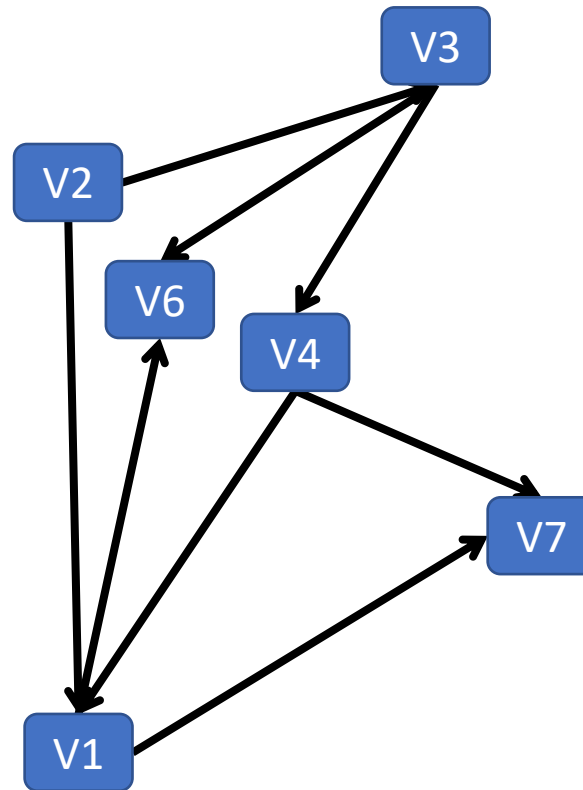
Algorithmus – Richtungszuweisung

- Breitensuche
- Kräftebasiert
- **Zufällig**



Algorithmus – Richtungszuweisung

- Breitensuche
- Kräftebasiert
- **Zufällig**



Algorithmus

- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

Algorithmus – Lagenzuordnung

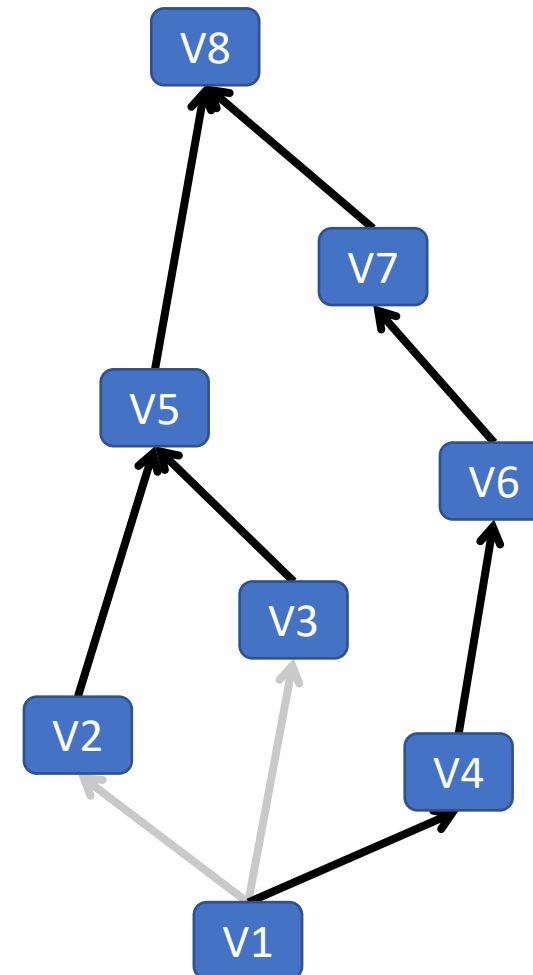
A Technique for Drawing Directed Graphs

Emden Gansner, Eleftherios Koutsofios, Stephen North, Kiem-Phong Vo

[IEEE TSE 1993]

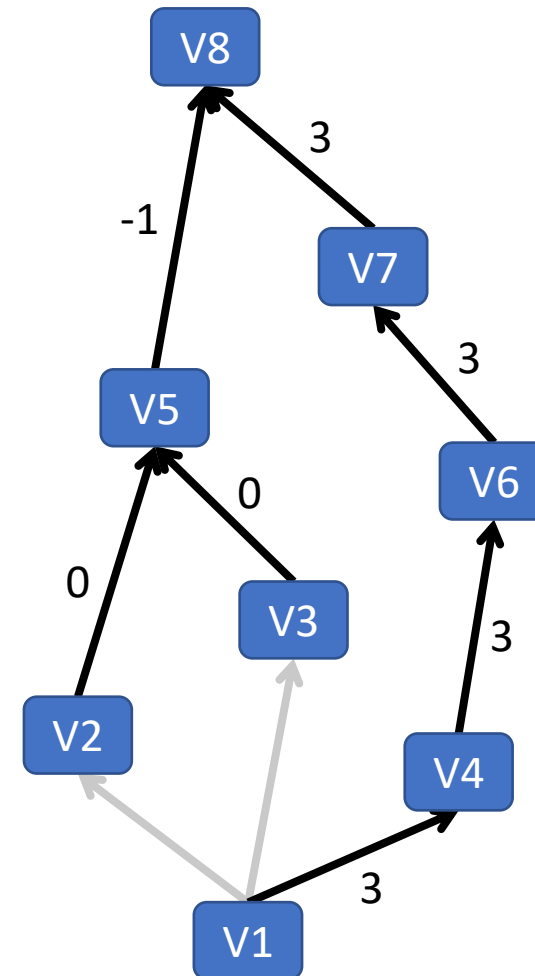
Algorithmus – Lagenzuordnung

- Nach Gansner et. al
- Network Simplex Algorithmus
 - **Spannbaum**
 - Schnittwerte der Kanten berechnen
 - Kanten mit negativem Schnittwert ersetzen
 - Terminierung wenn keine negativen Schnittwerte vorhanden sind
 - Einteilung der Knoten in Lagen entlang der Baumkanten



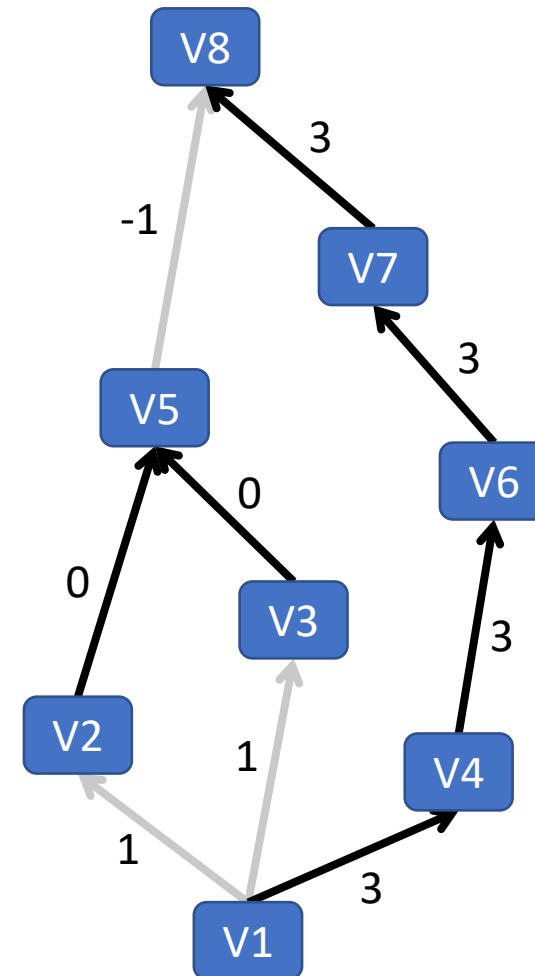
Algorithmus – Lagenzuordnung

- Nach Gansner et. al
- Network Simplex Algorithmus
 - Spannbaum
 - **Schnittwerte der Kanten berechnen**
 - Kanten mit negativem Schnittwert ersetzen
 - Terminierung wenn keine negativen Schnittwerte vorhanden sind
 - Einteilung der Knoten in Lagen entlang der Baumkanten



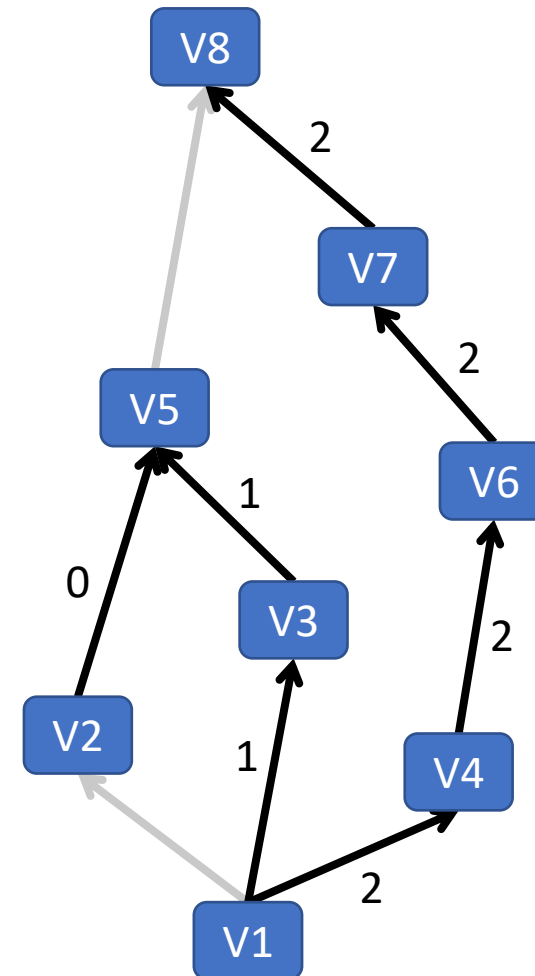
Algorithmus – Lagenzuordnung

- Nach Gansner et. al
- Network Simplex Algorithmus
 - Spannbaum
 - Schnittwerte der Kanten berechnen
 - **Kanten mit negativem Schnittwert ersetzen**
 - Terminierung wenn keine negativen Schnittwerte vorhanden sind
 - Einteilung der Knoten in Lagen entlang der Baumkanten



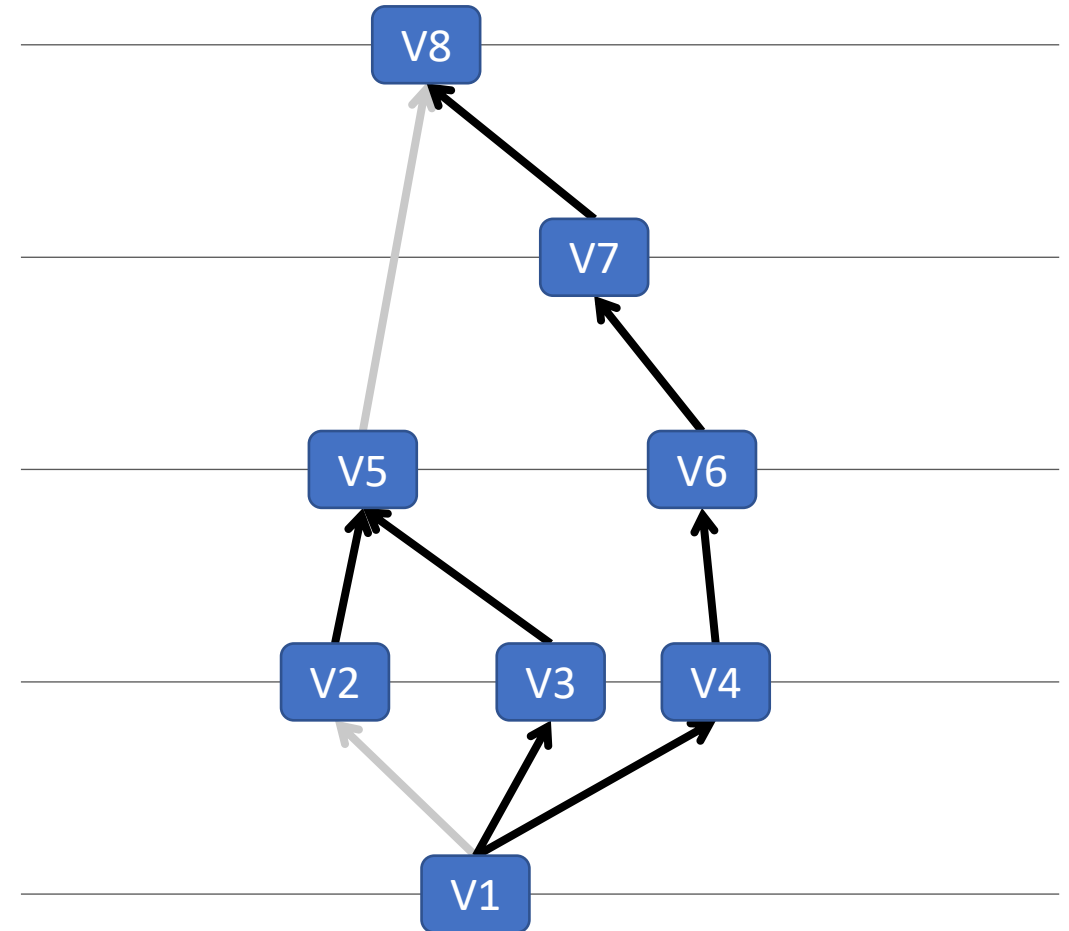
Algorithmus – Lagenzuordnung

- Nach Gansner et. al
- Network Simplex Algorithmus
 - Spannbaum
 - Schnittwerte der Kanten berechnen
 - Kanten mit negativem Schnittwert ersetzen
 - **Terminierung wenn keine negativen Schnittwerte vorhanden sind**
 - Einteilung der Knoten in Lagen entlang der Baumkanten



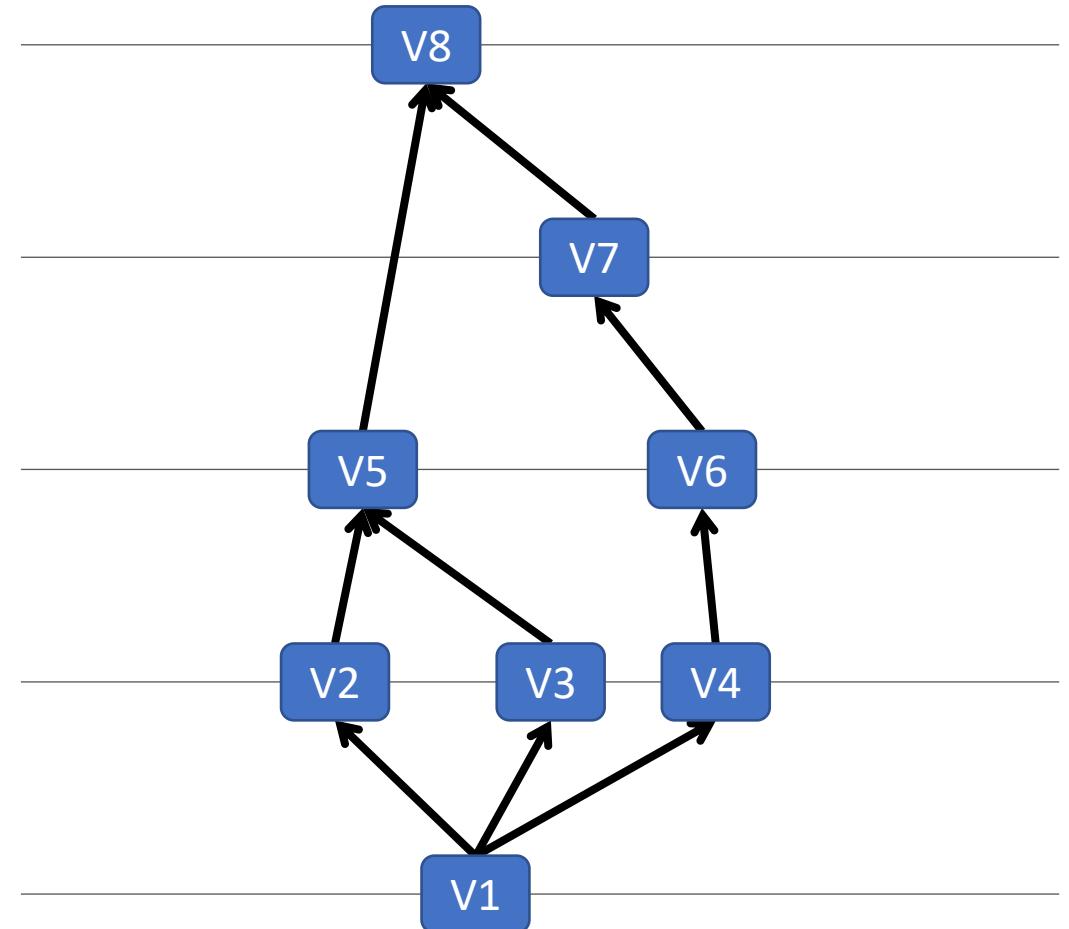
Algorithmus – Lagenzuordnung

- Nach Gansner et. al
- Network Simplex Algorithmus
 - Spannbaum
 - Schnittwerte der Kanten berechnen
 - Kanten mit negativem Schnittwert ersetzen
 - Terminierung wenn keine negativen Schnittwerte vorhanden sind
 - **Einteilung der Knoten in Lagen entlang der Baumkanten**



Algorithmus – Lagenzuordnung

- Nach Gansner et. al
- Network Simplex Algorithmus
 - Spannbaum
 - Schnittwerte der Kanten berechnen
 - Kanten mit negativem Schnittwert ersetzen
 - Terminierung wenn keine negativen Schnittwerte vorhanden sind
 - Einteilung der Knoten in Lagen entlang der Baumkanten



Algorithmus

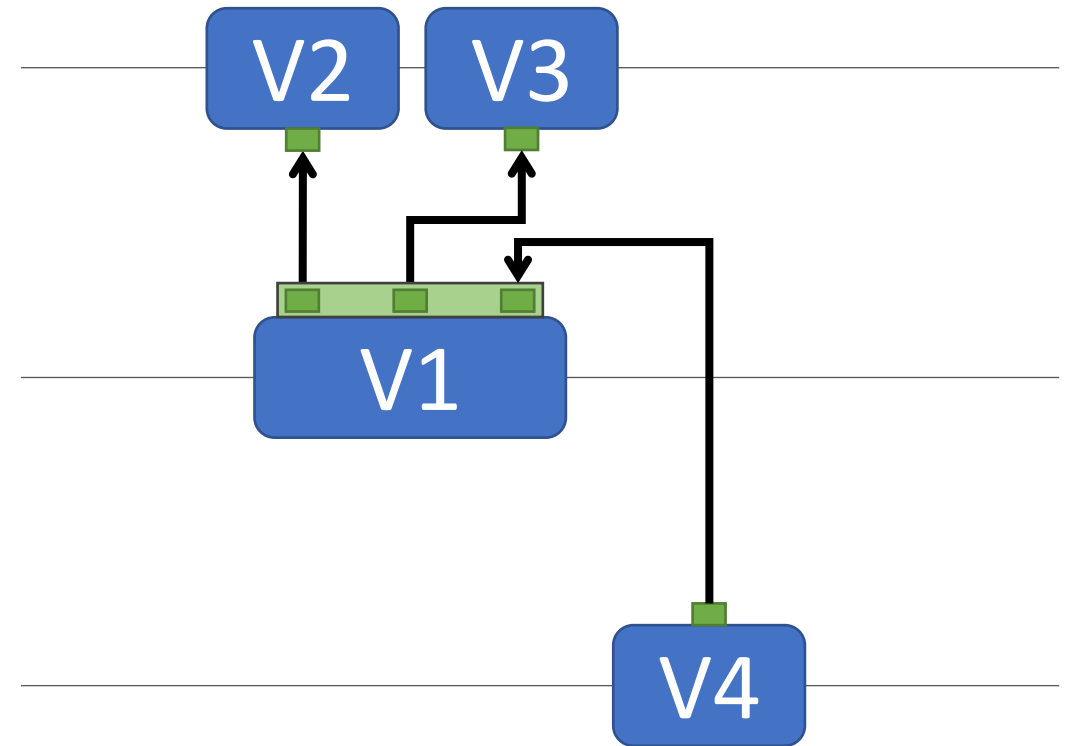
- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

Algorithmus – Dummyknotenerstellung

Algorithmus – Dummyknotenerstellung

Bei

- Ports auf der falschen Seite ihres Knotens (durch Portgruppen)
- Kanten die Ebenen überspannen



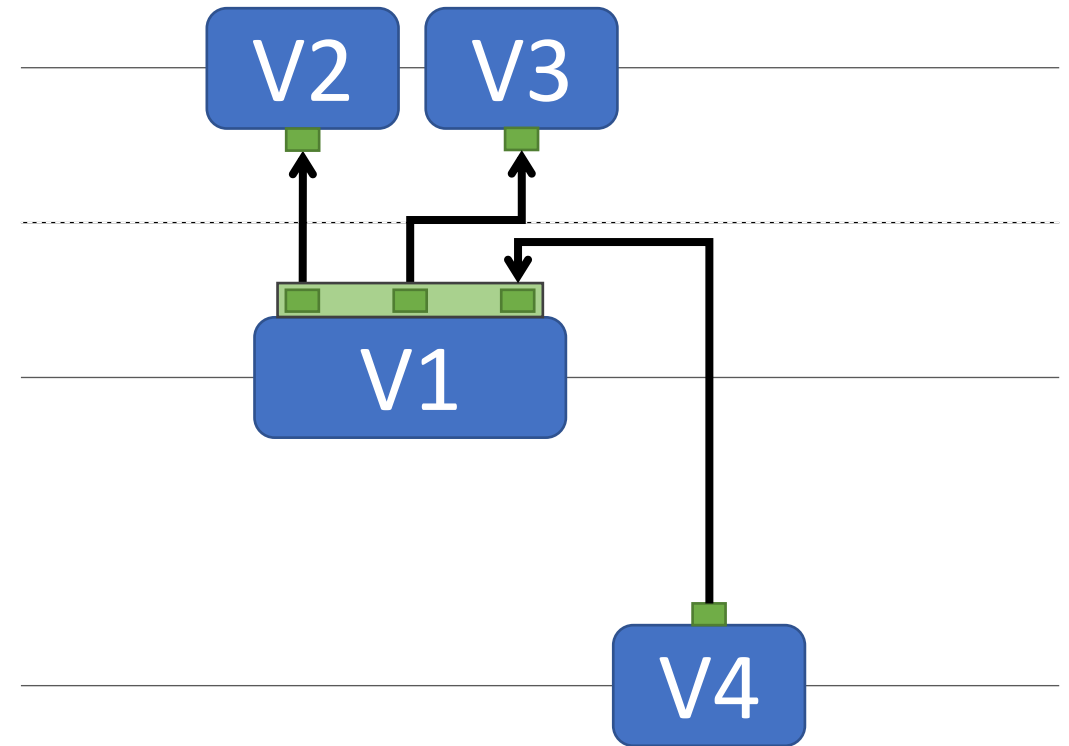
Algorithmus – Dummyknotenerstellung

Bei

- Ports auf der falschen Seite ihres Knotens (durch Portgruppen)
- Kanten die Ebenen überspannen

Vorgehen

- Zusatzebene einfügen



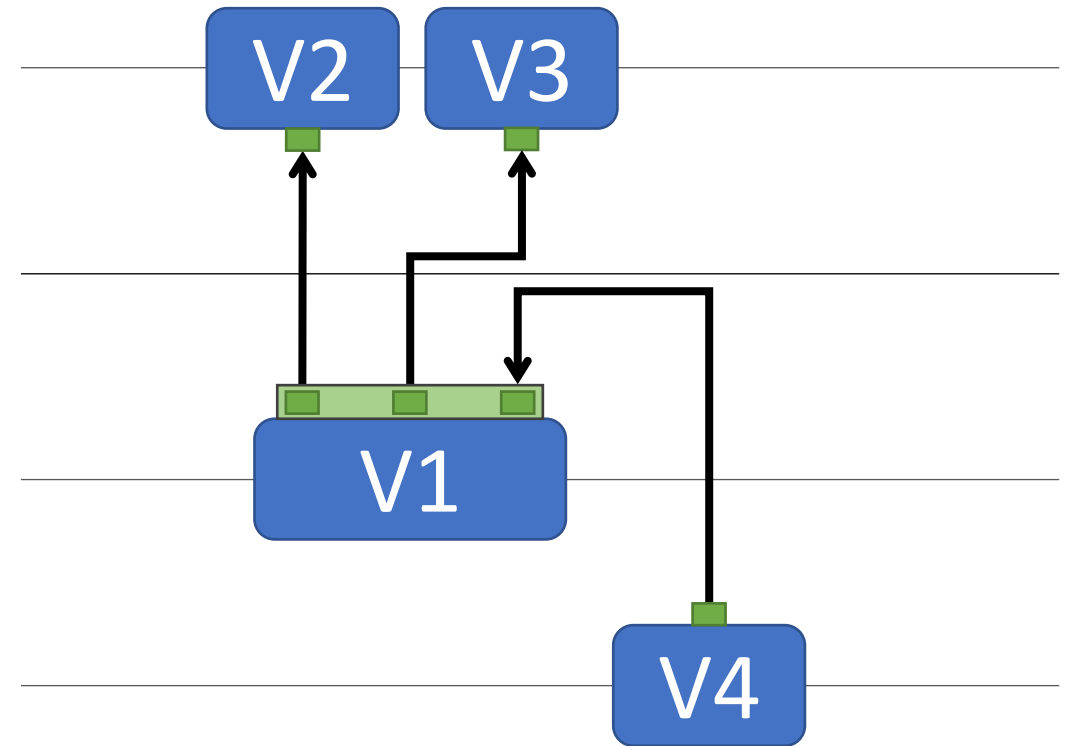
Algorithmus – Dummyknotenerstellung

Bei

- Ports auf der falschen Seite ihres Knotens (durch Portgruppen)
- Kanten die Ebenen überspannen

Vorgehen

- Zusatzebene einfügen
- Dummyknoten erstellen



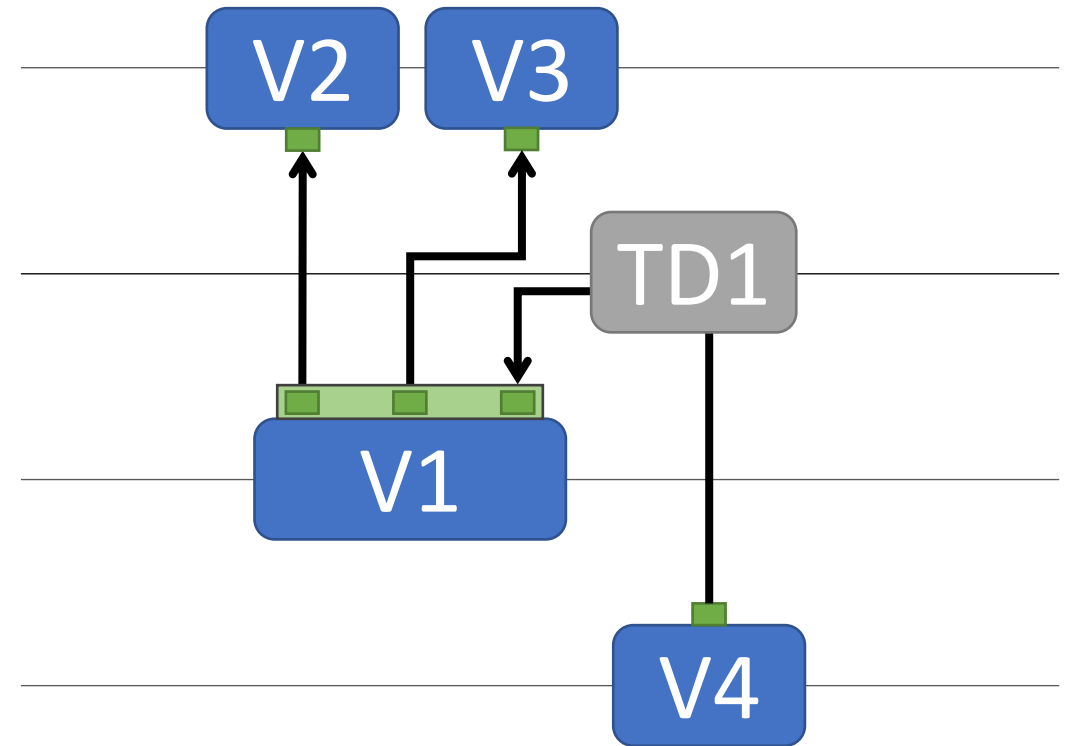
Algorithmus – Dummyknotenerstellung

Bei

- Ports auf der falschen Seite ihres Knotens (durch Portgruppen)
- Kanten die Ebenen überspannen

Vorgehen

- Zusatzebene einfügen
- Dummyknoten erstellen



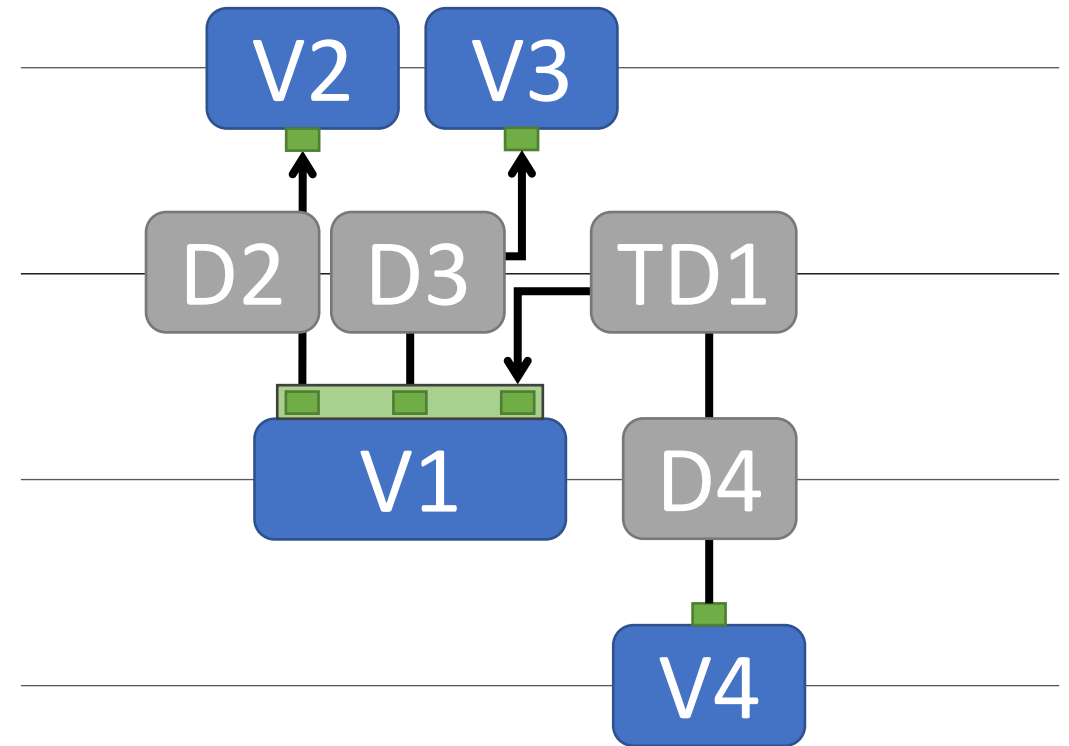
Algorithmus – Dummyknotenerstellung

Bei

- Ports auf der falschen Seite ihres Knotens (durch Portgruppen)
- Kanten die Ebenen überspannen

Vorgehen

- Zusatzebene einfügen
- Dummyknoten erstellen



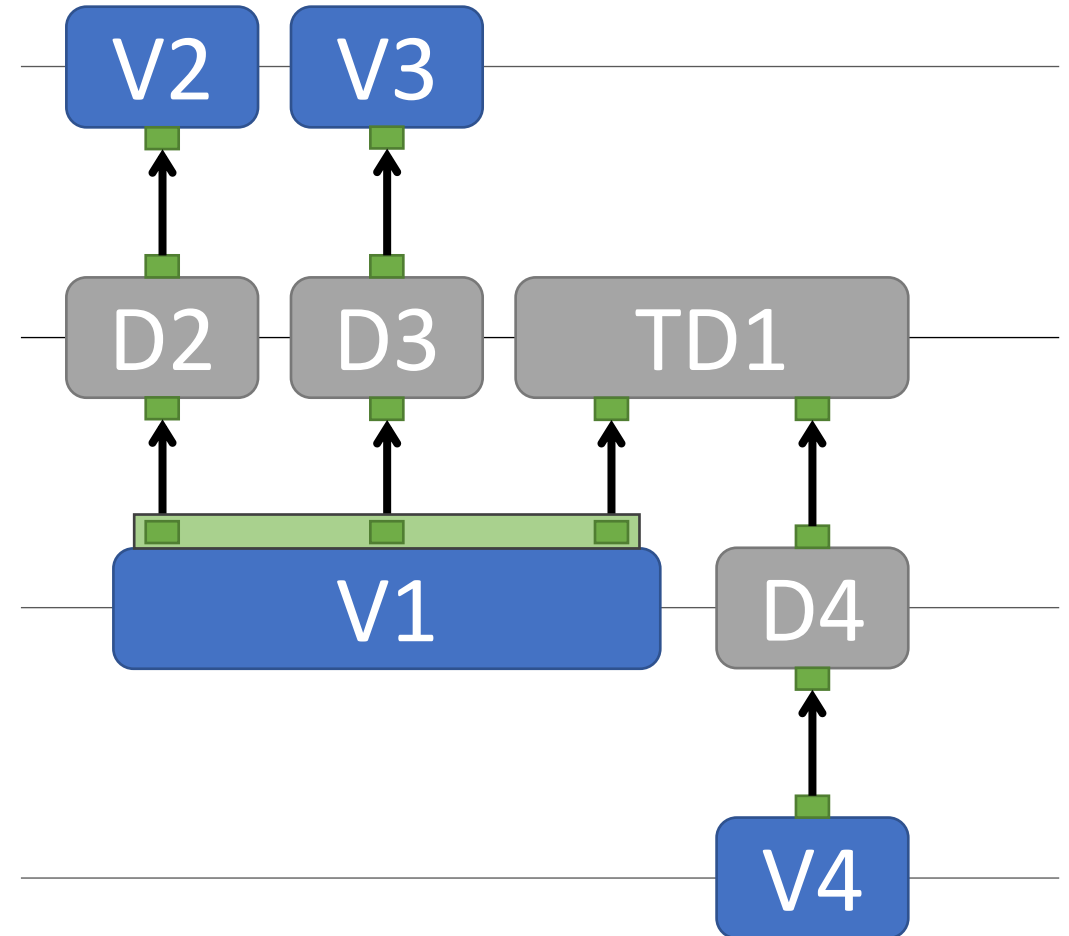
Algorithmus – Dummyknotenerstellung

Bei

- Ports auf der falschen Seite ihres Knotens (durch Portgruppen)
- Kanten die Ebenen überspannen

Vorgehen

- Zusatzebene einfügen
- Dummyknoten erstellen



Algorithmus

- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

Algorithmus – Kreuzungsreduzierung

Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung

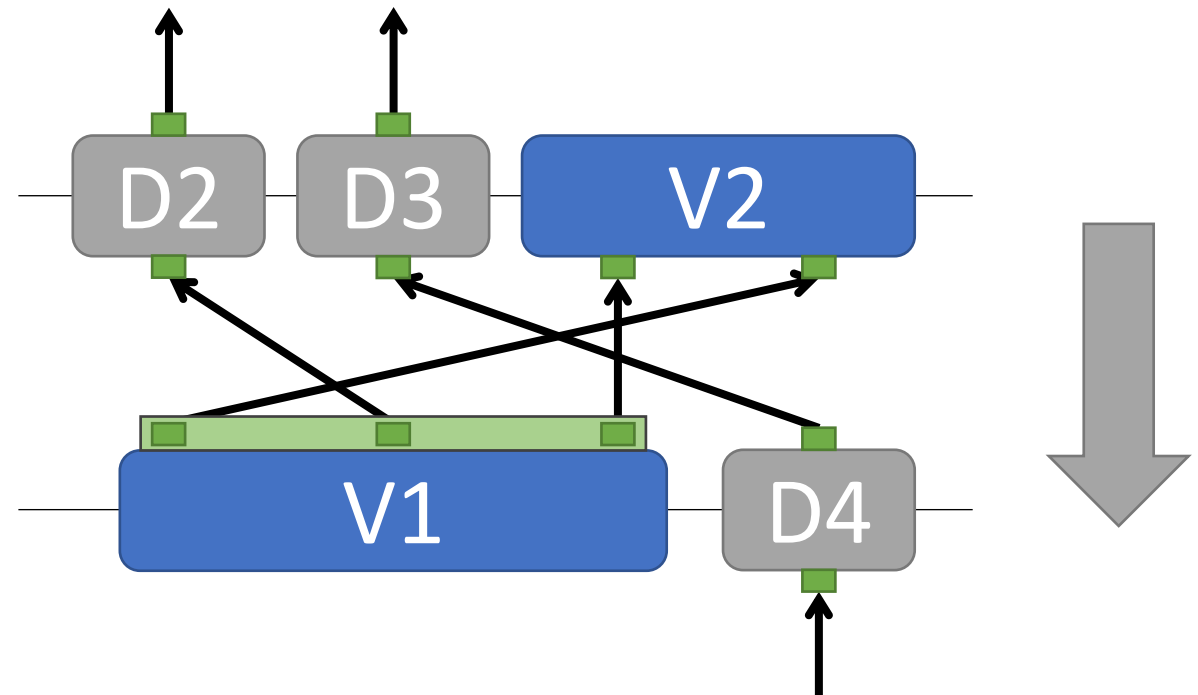
Algorithmus – Kreuzungsreduzierung

- **Behandlung von Portgruppen**

- Layer-Total-Methode
- Sortierung nach den Knoten

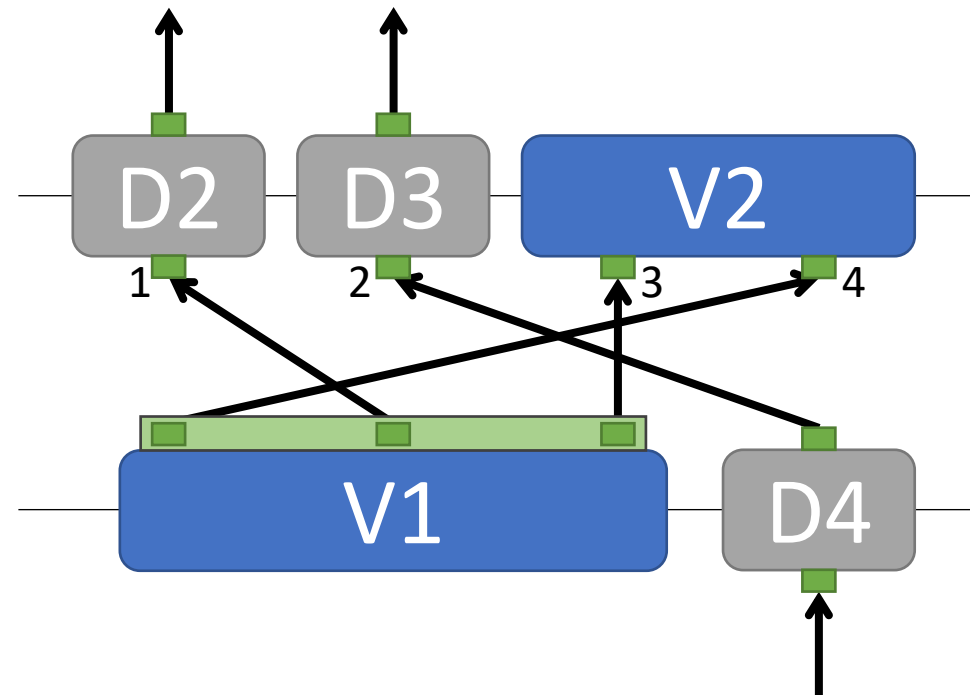
- Behandlung von Portpaaren

- Behandlung von Knoten mit Verbindung in nur eine Richtung



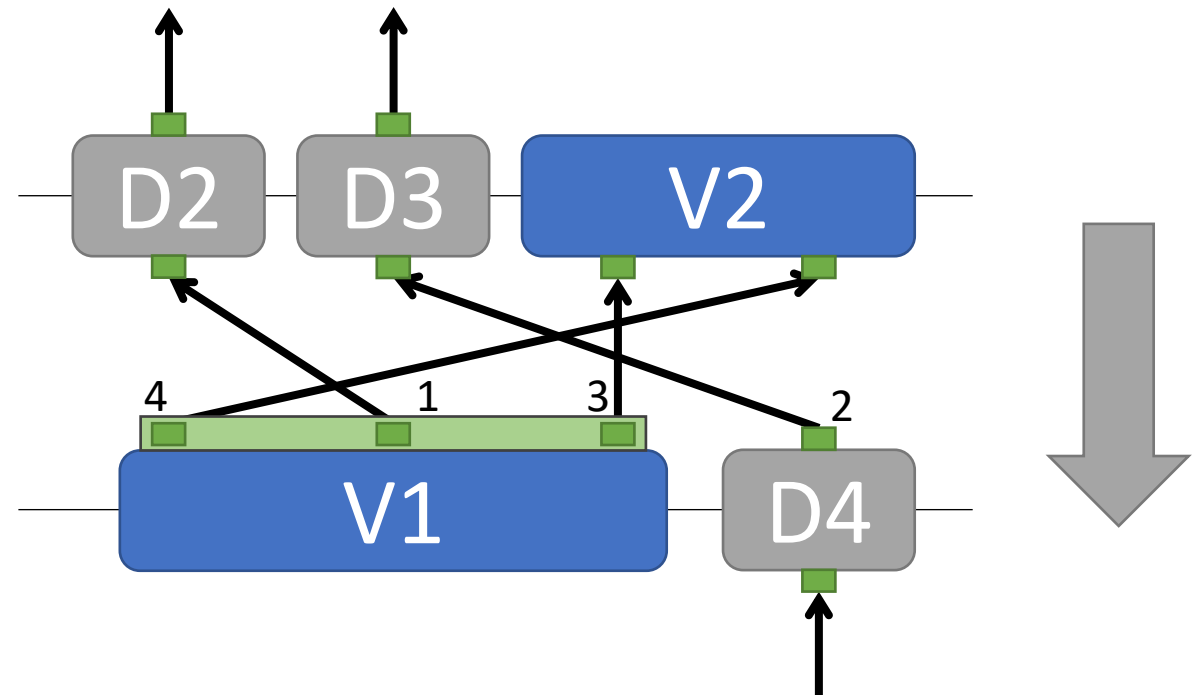
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - **Layer-Total-Methode**
 - Sortierung nach den Knoten
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung



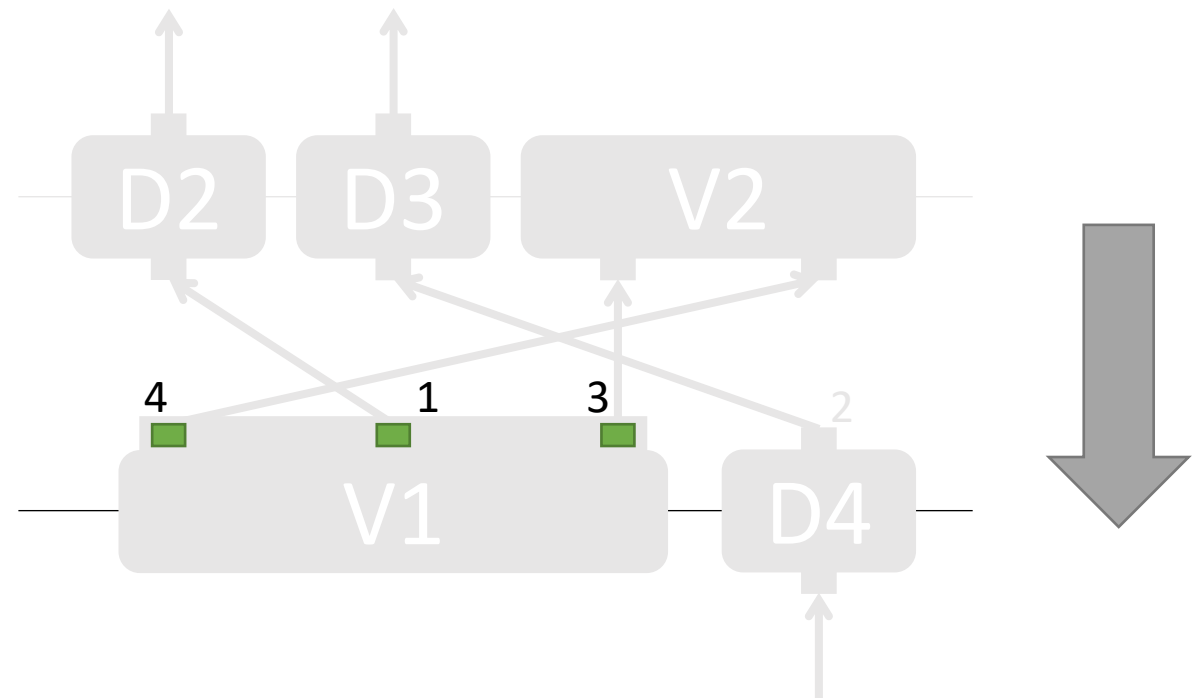
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - **Layer-Total-Methode**
 - Sortierung nach den Knoten
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung



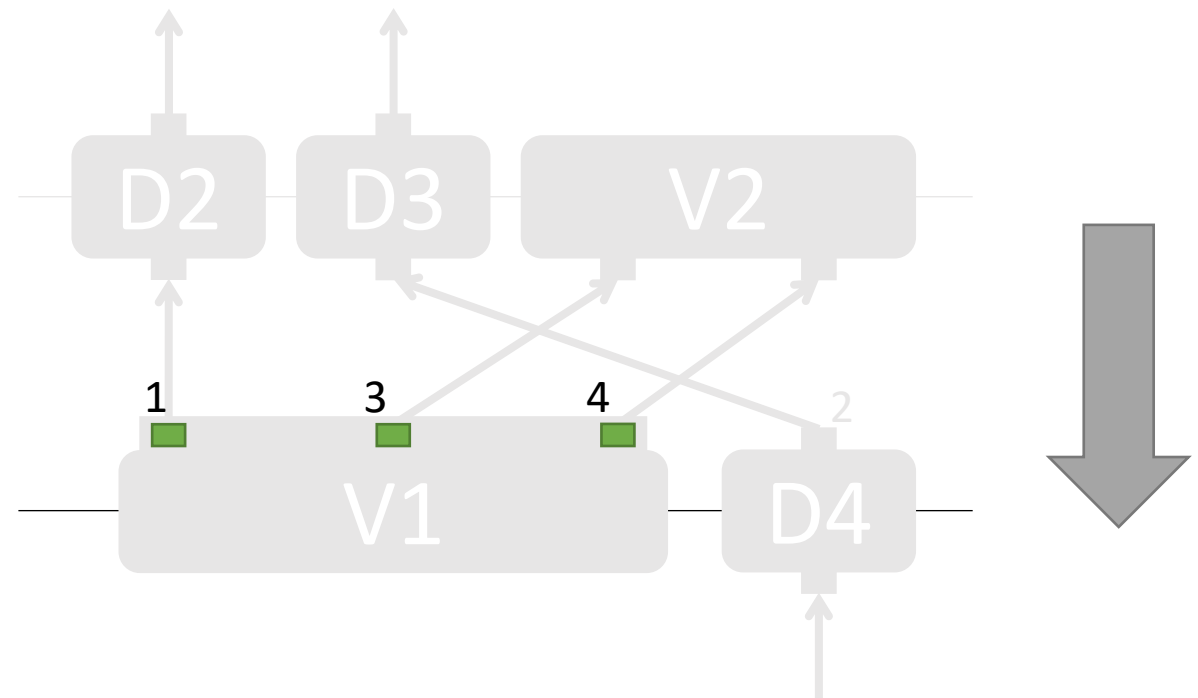
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - **Layer-Total-Methode**
 - Sortierung nach den Knoten
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung



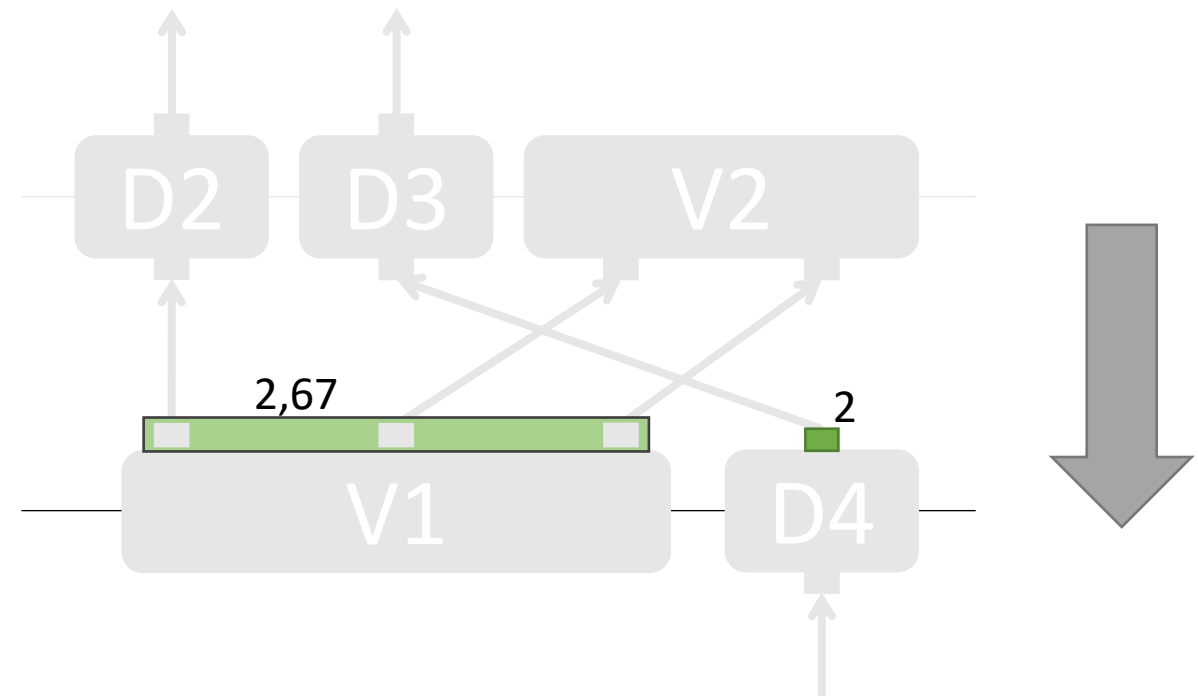
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - **Layer-Total-Methode**
 - Sortierung nach den Knoten
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung



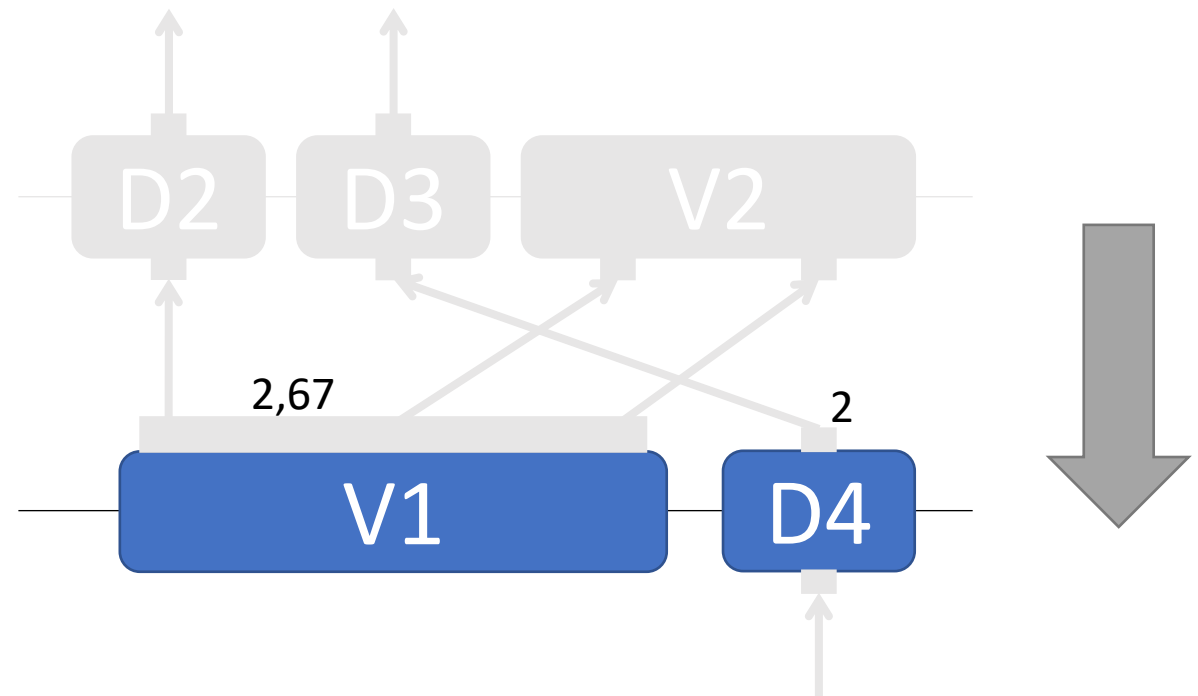
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - **Layer-Total-Methode**
 - Sortierung nach den Knoten
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung



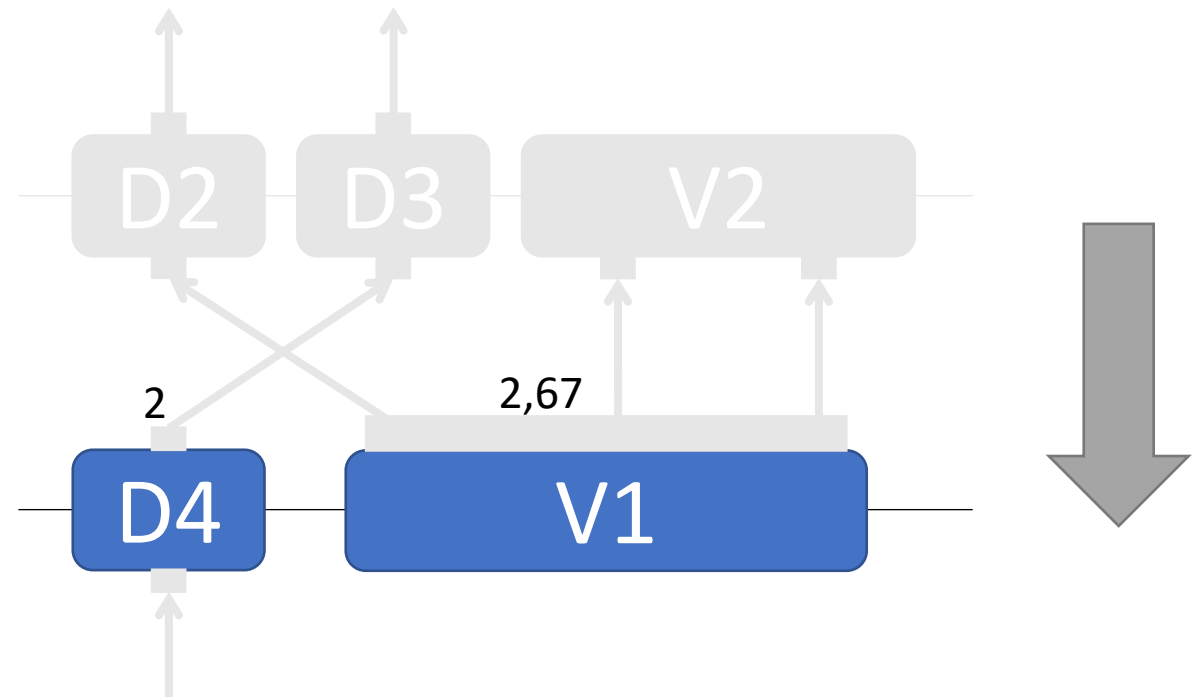
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - **Layer-Total-Methode**
 - Sortierung nach den Knoten
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung



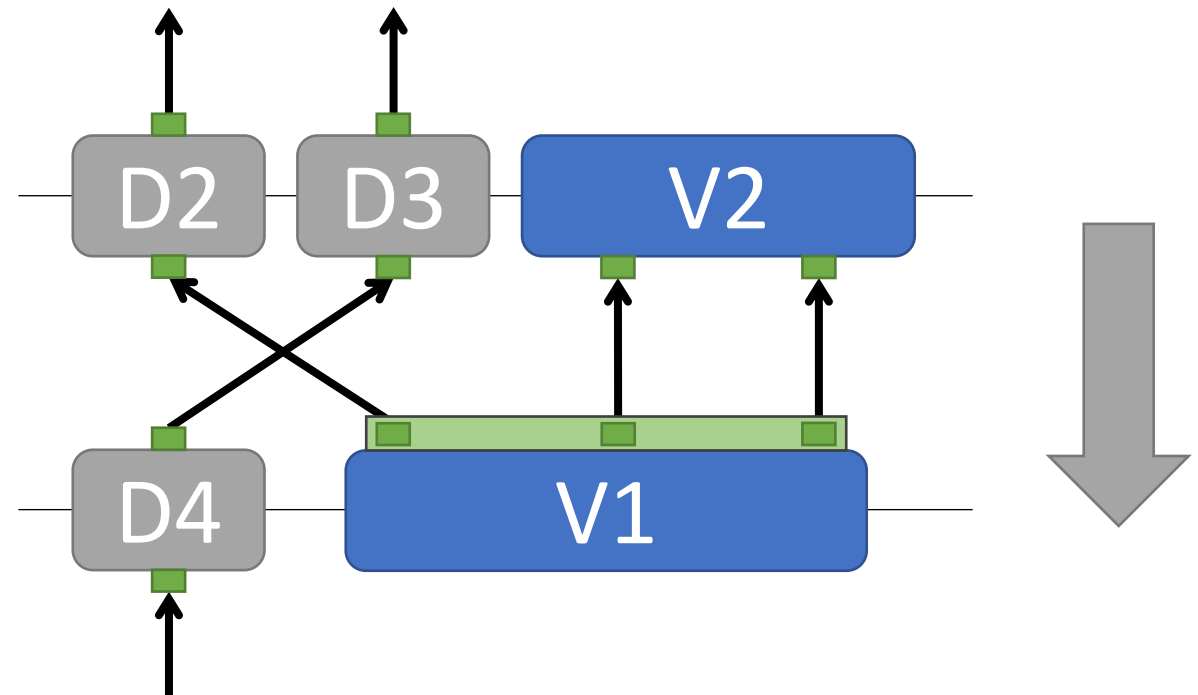
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - **Layer-Total-Methode**
 - Sortierung nach den Knoten
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung



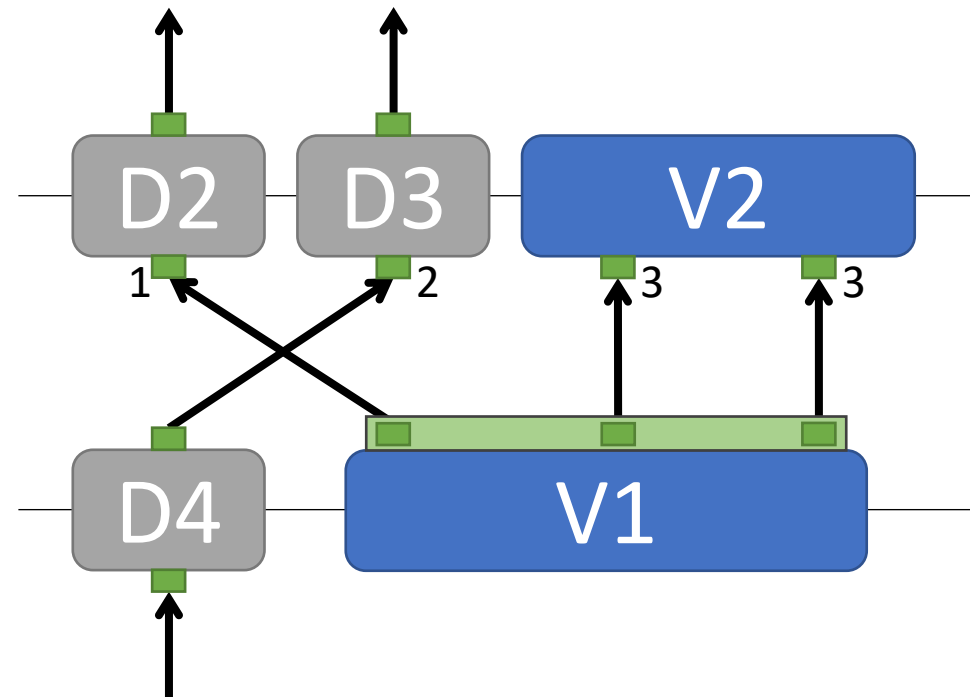
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - **Layer-Total-Methode**
 - Sortierung nach den Knoten
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung



Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
 - Layer-Total-Methode
 - **Sortierung nach den Knoten**
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung

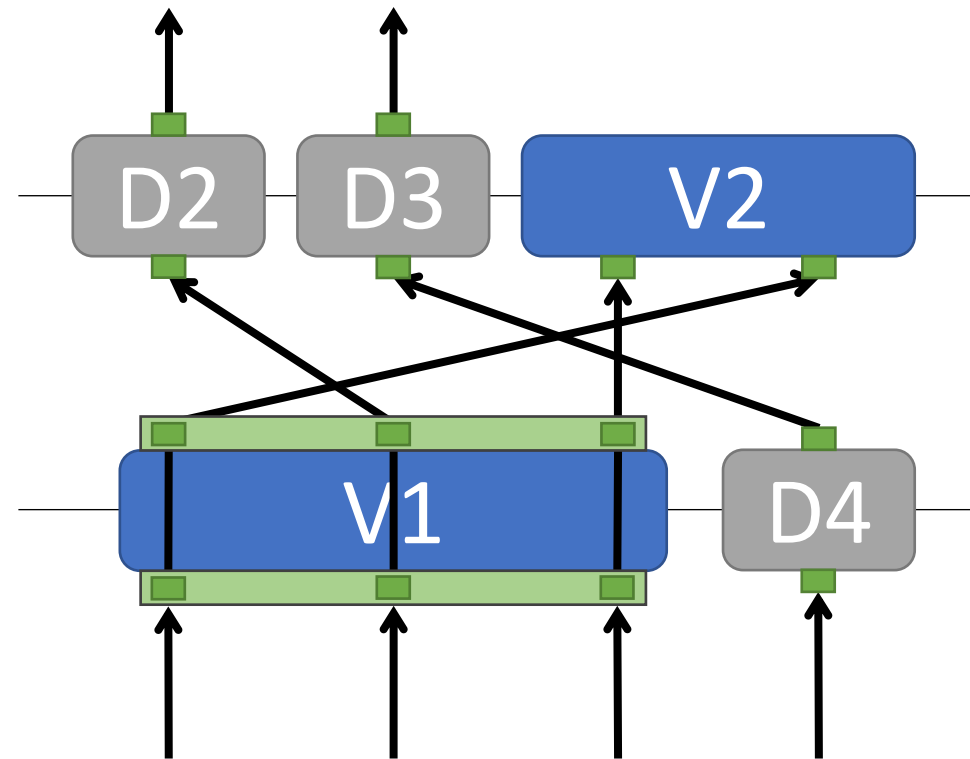


Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung

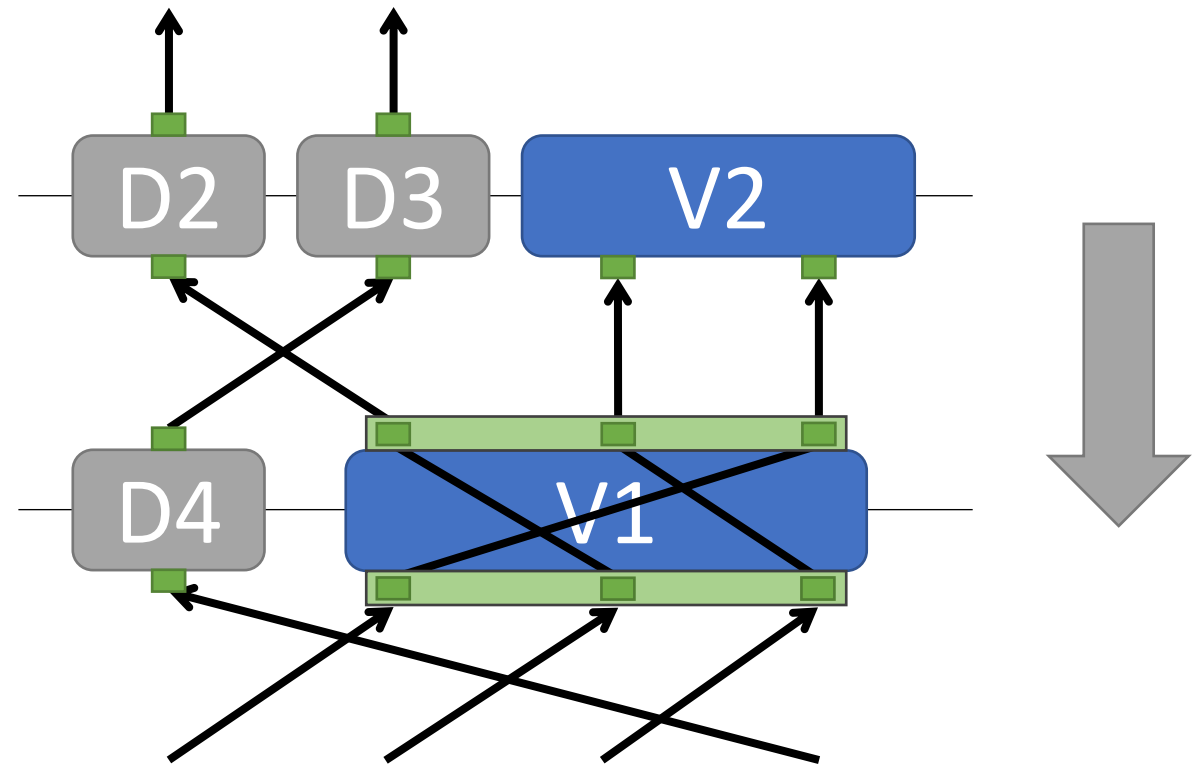
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- **Behandlung von Portpaaren**
 - Kreuzungen in Knoten vermeiden
 - Umsortieren der Partner
- Behandlung von Knoten mit Verbindung in nur eine Richtung



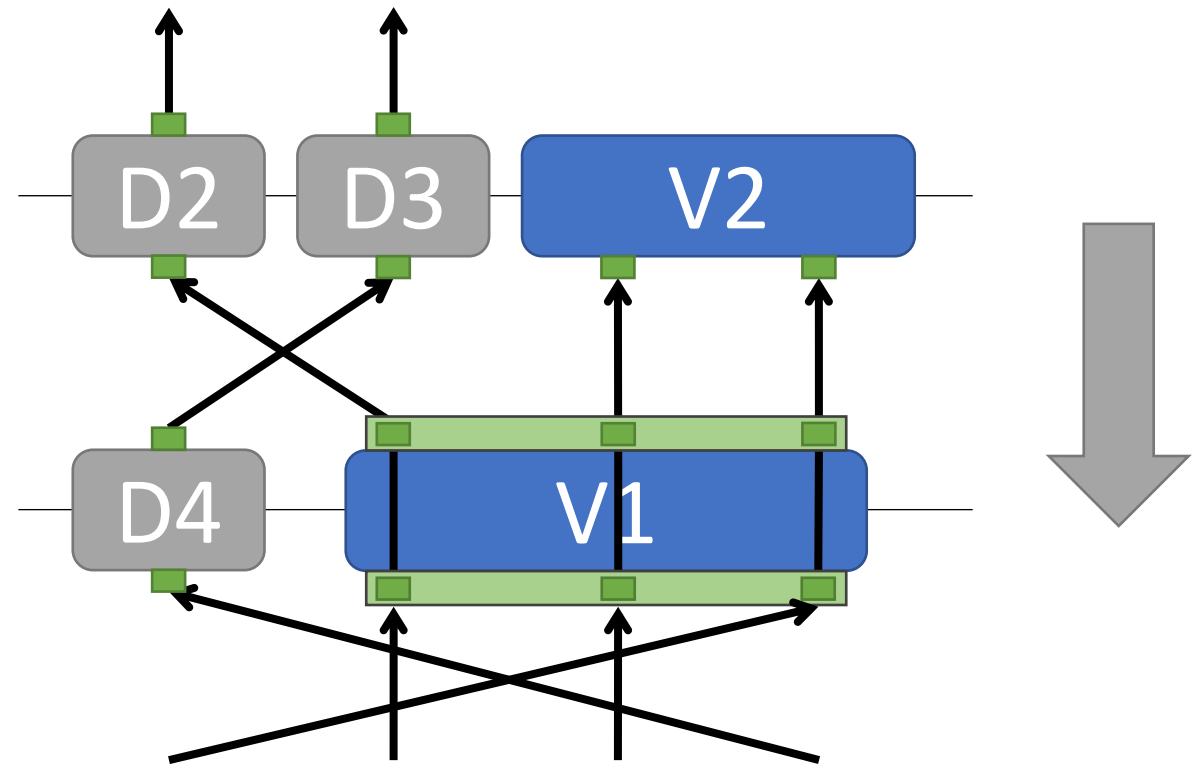
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
 - **Kreuzungen in Knoten vermeiden**
 - Umsortieren der Partner
- Behandlung von Knoten mit Verbindung in nur eine Richtung



Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
 - Kreuzungen in Knoten vermeiden
 - **Umsortieren der Partner**
- Behandlung von Knoten mit Verbindung in nur eine Richtung

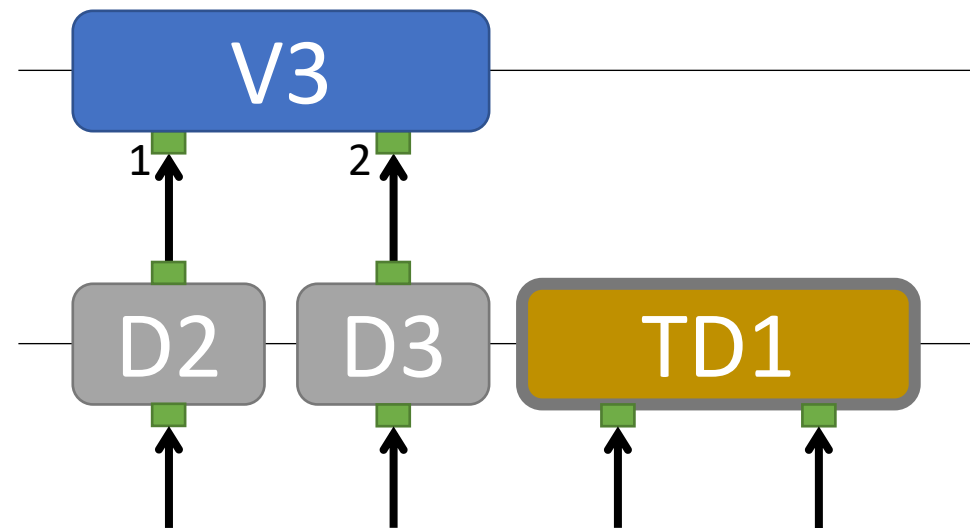


Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung

Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- **Behandlung von Knoten mit Verbindung in nur eine Richtung**
 - Ersatzwert für Barycenter entsprechend aktueller Position



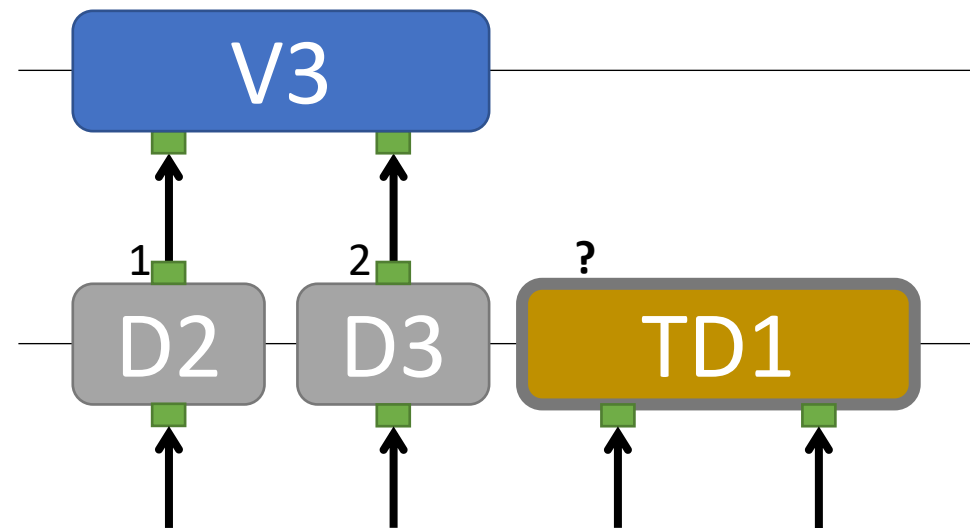
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung
 - **Ersatzwert für Barycenter entsprechend aktueller Position**

$$\text{bc}(D2) = 1$$

$$\text{bc}(D3) = 2$$

$$\text{bc}(TD1) = ?$$



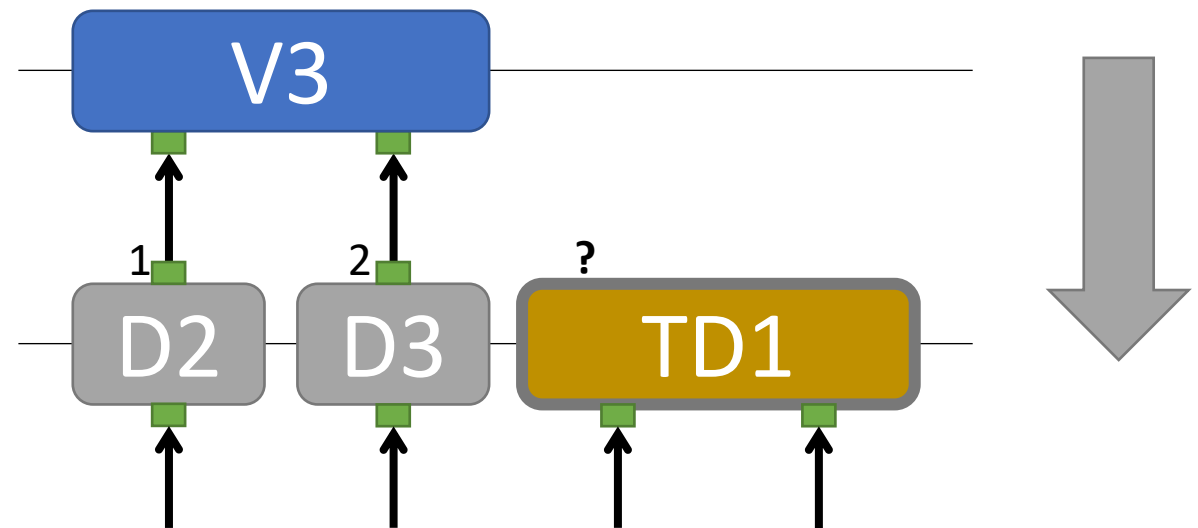
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung
 - **Ersatzwert für Barycenter entsprechend aktueller Position**

$$bc(D2) = 1$$

$$bc(D3) = 2$$

$$bc(TD1) = \left\lceil \frac{b_{max}-1}{\#n-1} \cdot [pos - 1] \right\rceil + 1$$



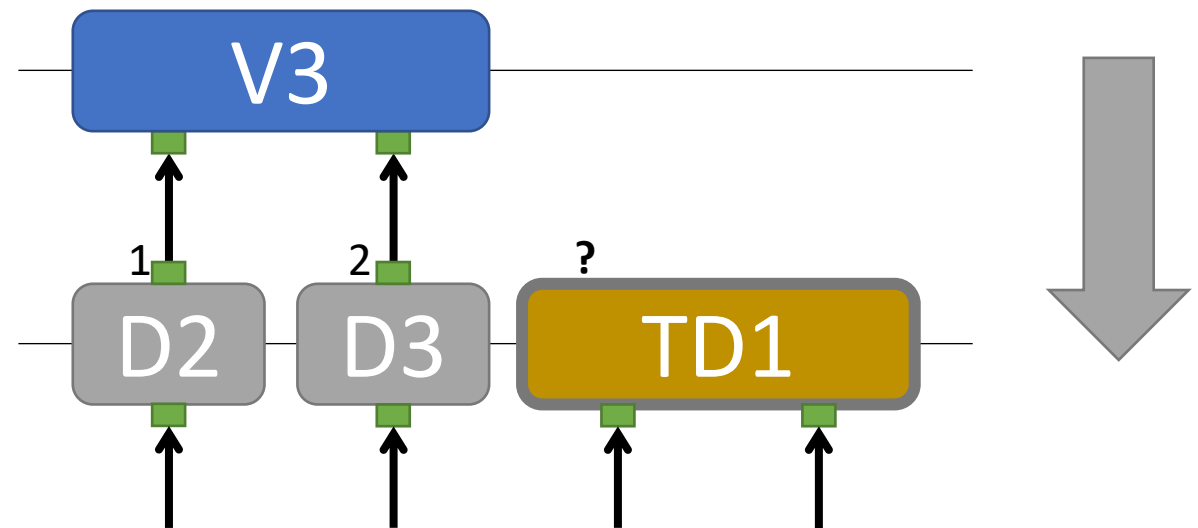
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung
 - **Ersatzwert für Barycenter entsprechend aktueller Position**

$$\text{bc}(D2) = 1$$

$$\text{bc}(D3) = 2$$

$$\text{bc}(\text{TD1}) = \left[\frac{2-1}{\#n-1} \cdot [\text{pos} - 1] \right] + 1$$



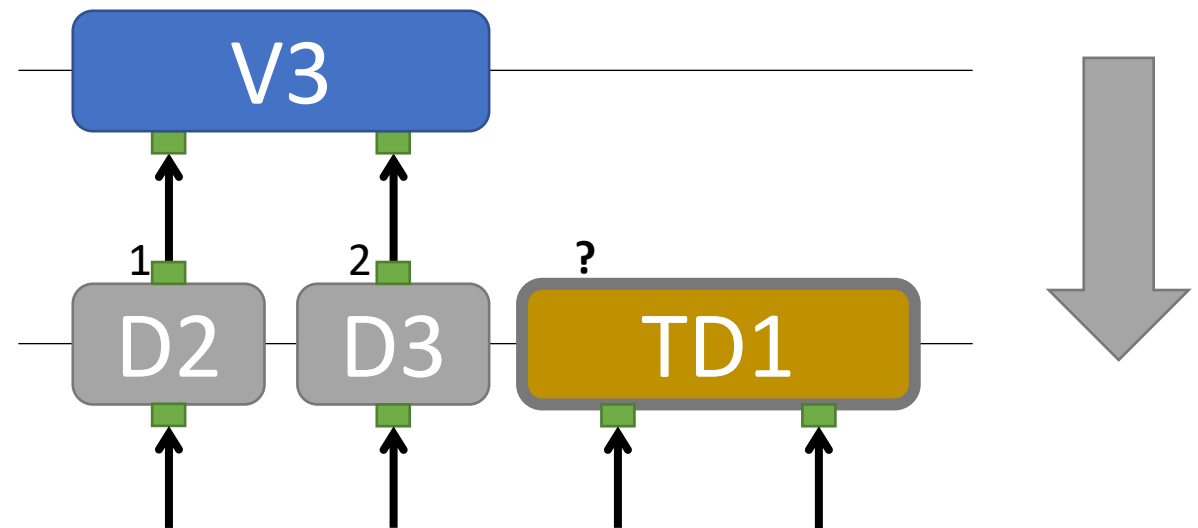
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung
 - **Ersatzwert für Barycenter entsprechend aktueller Position**

$$\text{bc}(D2) = 1$$

$$\text{bc}(D3) = 2$$

$$\text{bc}(\text{TD1}) = \left\lceil \frac{2-1}{3-1} \cdot [\text{pos} - 1] \right\rceil + 1$$



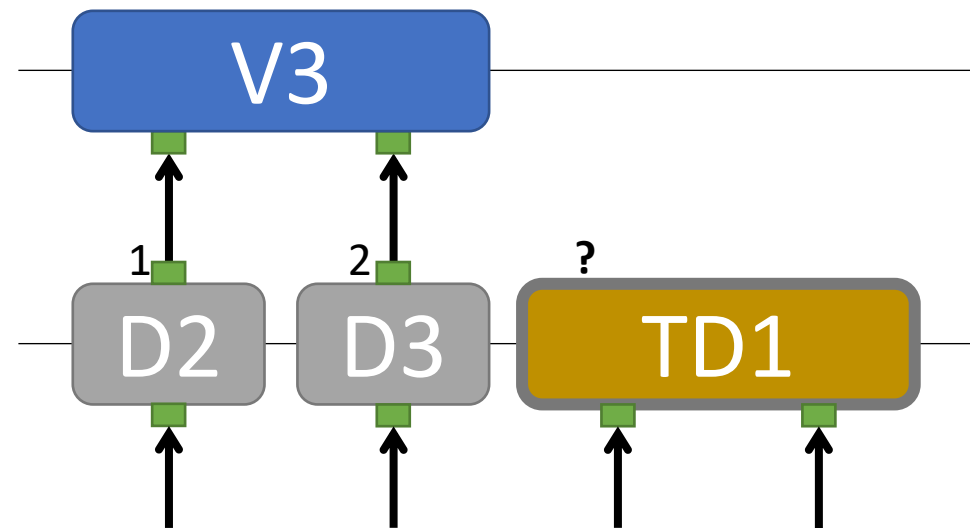
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung
 - **Ersatzwert für Barycenter entsprechend aktueller Position**

$$\text{bc}(D2) = 1$$

$$\text{bc}(D3) = 2$$

$$\text{bc}(TD1) = \left[\frac{2-1}{3-1} \cdot [3 - 1] \right] + 1$$



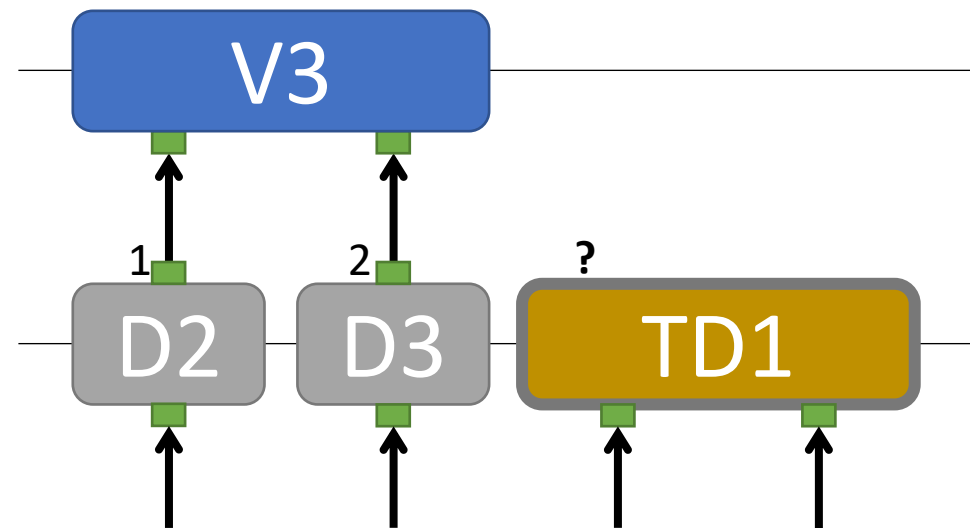
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung
 - **Ersatzwert für Barycenter entsprechend aktueller Position**

$$\text{bc}(D2) = 1$$

$$\text{bc}(D3) = 2$$

$$\text{bc}(TD1) = \left[\frac{2-1}{3-1} \cdot [3 - 1] \right] + 1 = \left[\frac{1}{2} \cdot [2] \right] + 1$$



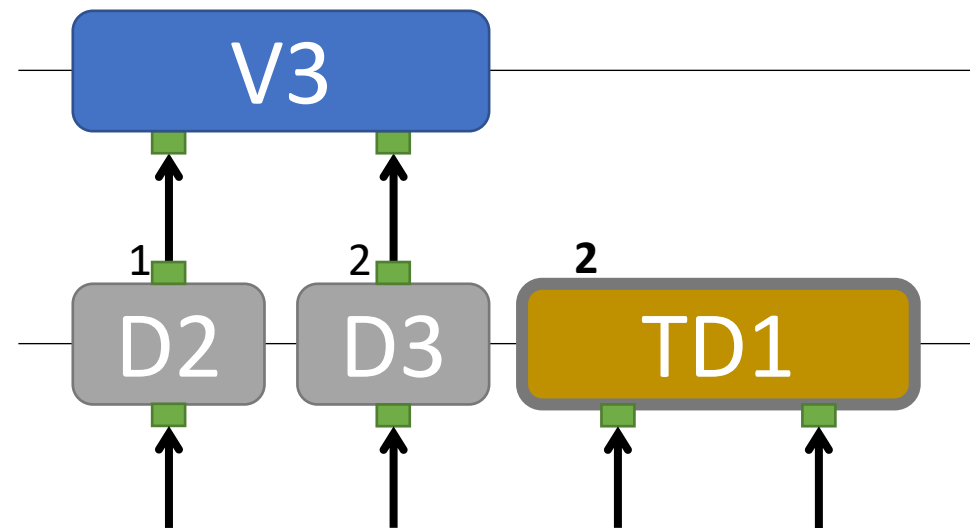
Algorithmus – Kreuzungsreduzierung

- Behandlung von Portgruppen
- Behandlung von Portpaaren
- Behandlung von Knoten mit Verbindung in nur eine Richtung
 - **Ersatzwert für Barycenter entsprechend aktueller Position**

$$\text{bc}(D2) = 1$$

$$\text{bc}(D3) = 2$$

$$\text{bc}(TD1) = \left[\frac{2-1}{3-1} \cdot [3 - 1] \right] + 1 = \left[\frac{1}{2} \cdot [2] \right] + 1 = 2$$



Algorithmus

- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- Umwandlung der Knoten

Algorithmus – Knotenpositionierung

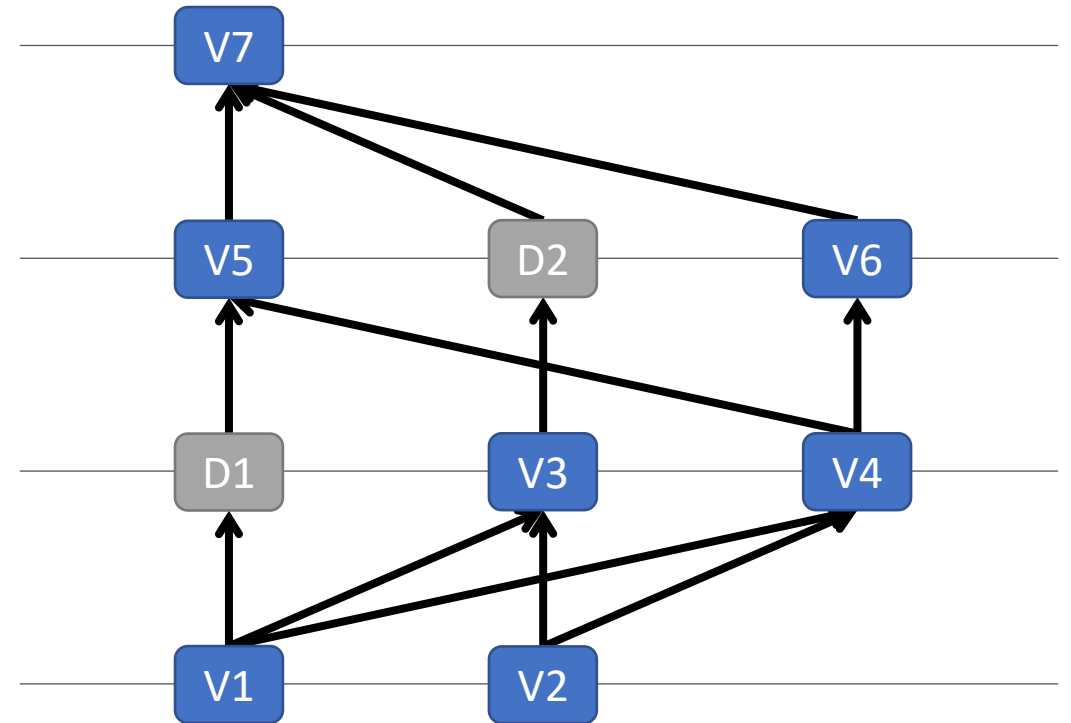
Fast and Simple Horizontal Coordinate Assignment

Ulrik Brandes, Boris Köpf

[GD 2002]

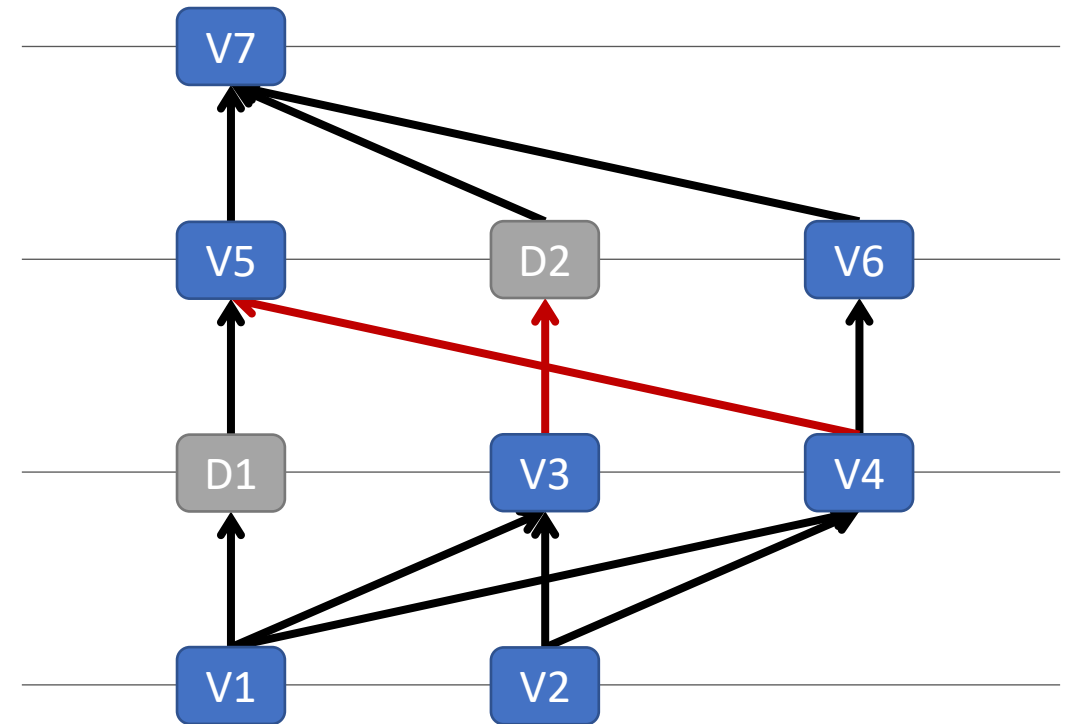
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
- Ausrichtung der Knoten entlang verbliebener Kanten
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



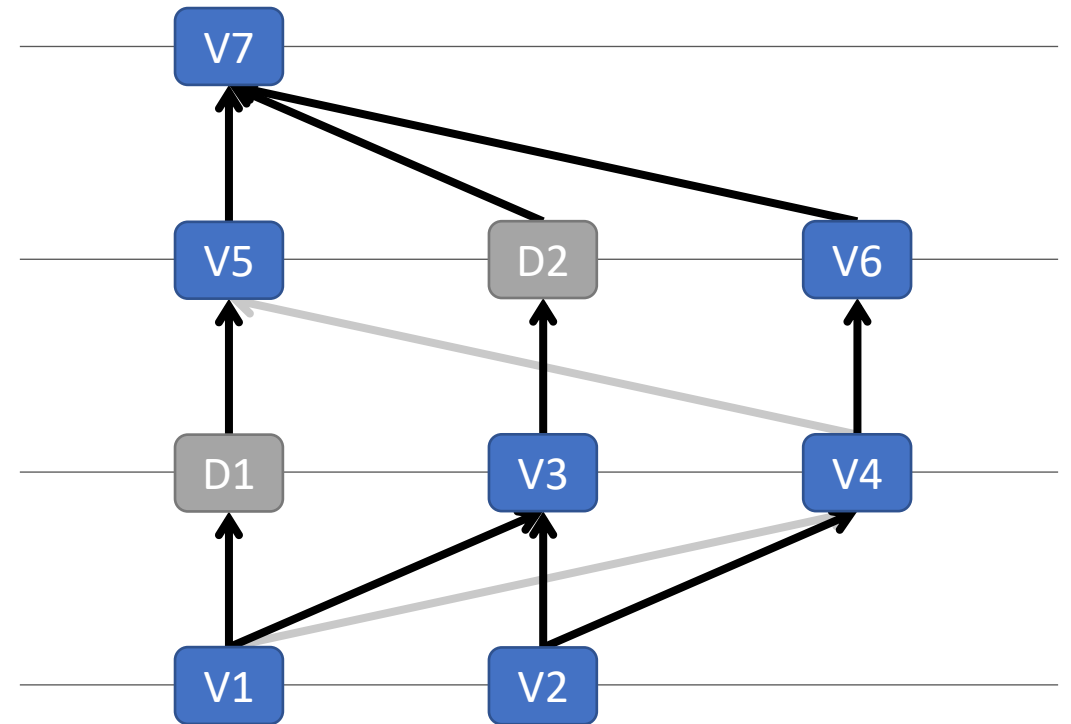
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
 - **Bei Kreuzungen**
 - Bei mehreren Kanten pro Knoten
- Ausrichtung der Knoten entlang verbliebener Kanten
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



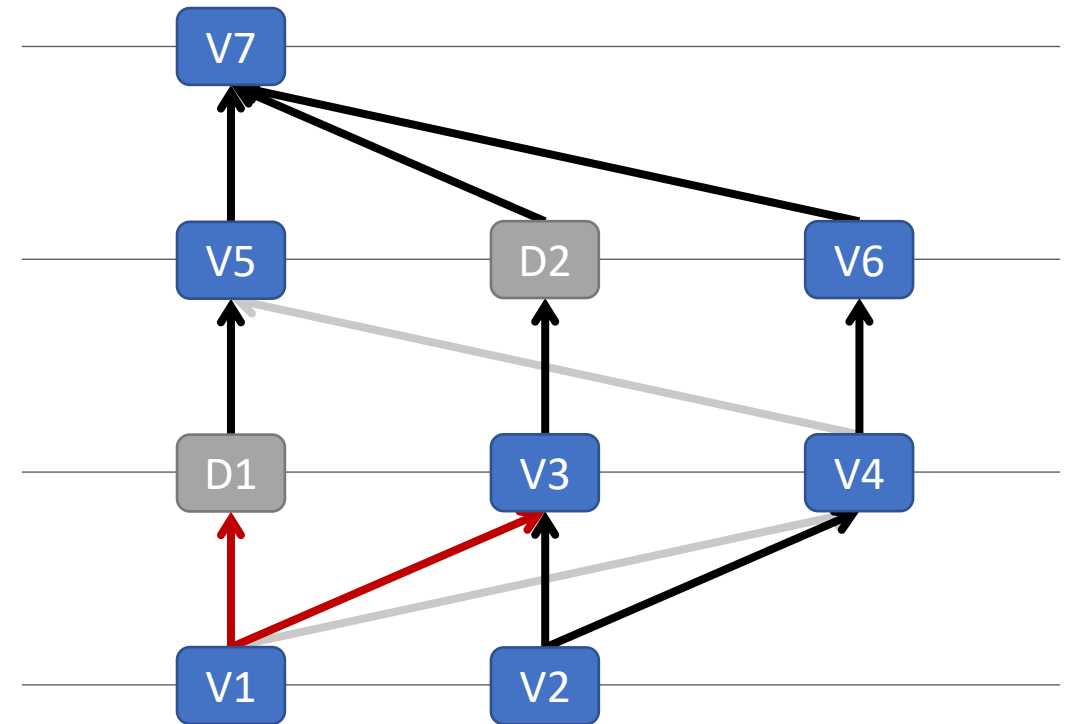
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
 - **Bei Kreuzungen**
 - Bei mehreren Kanten pro Knoten
- Ausrichtung der Knoten entlang verbliebener Kanten
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



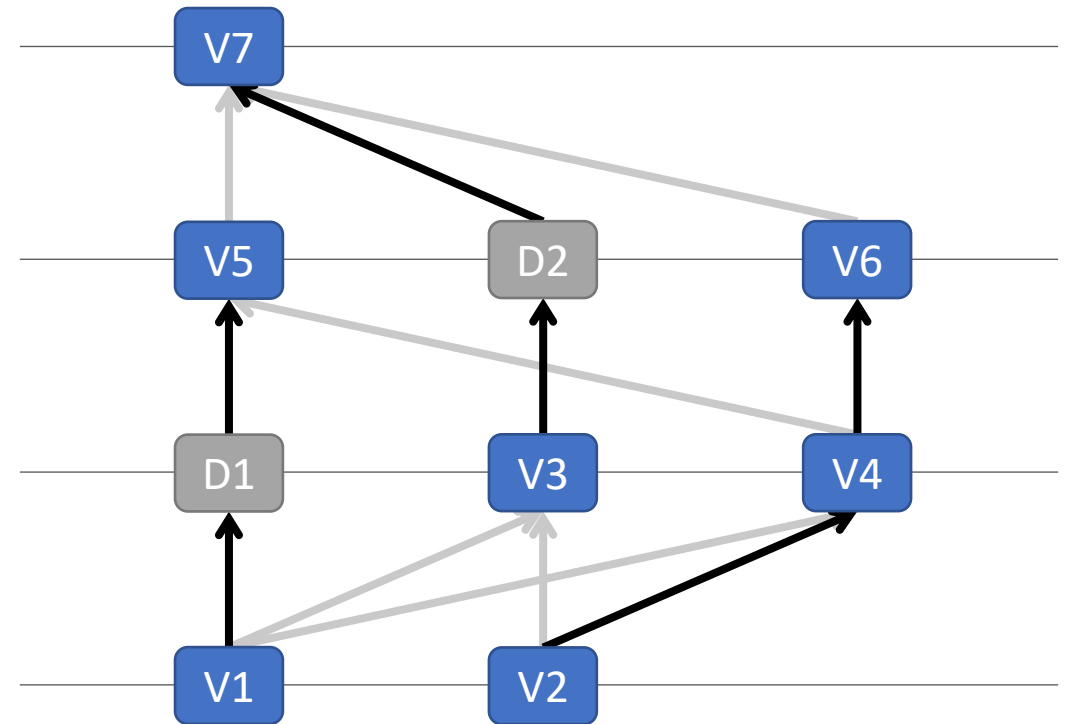
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
 - Bei Kreuzungen
 - **Bei mehreren Kanten pro Knoten**
- Ausrichtung der Knoten entlang verbliebener Kanten
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



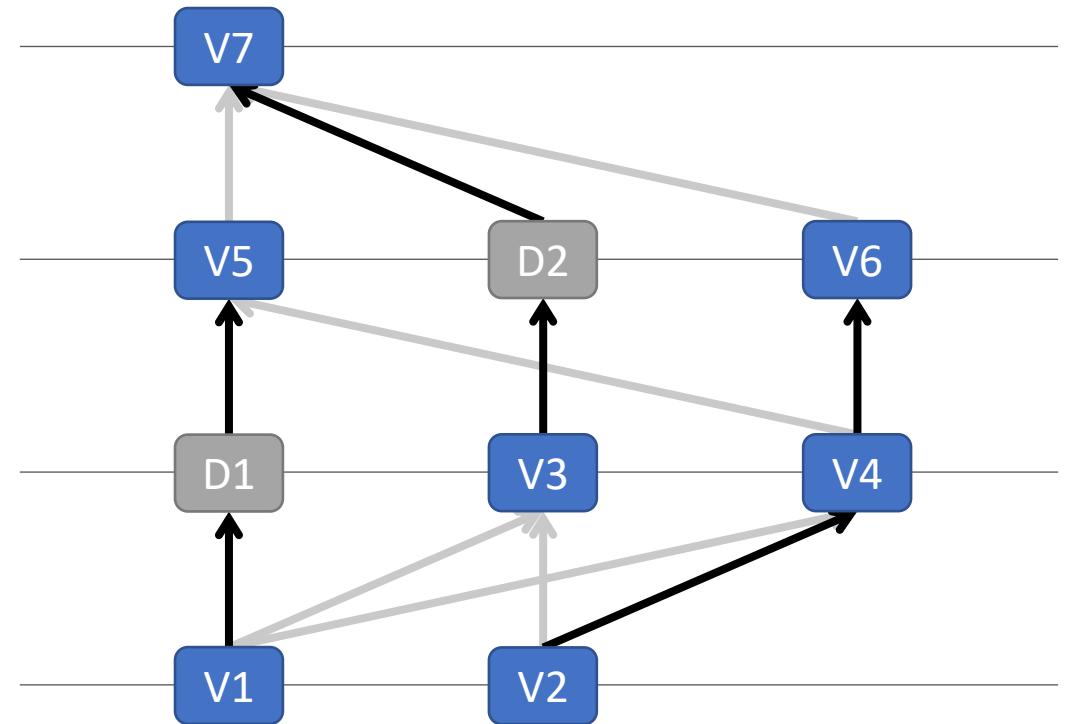
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
 - Bei Kreuzungen
 - **Bei mehreren Kanten pro Knoten**
- Ausrichtung der Knoten entlang verbliebener Kanten
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



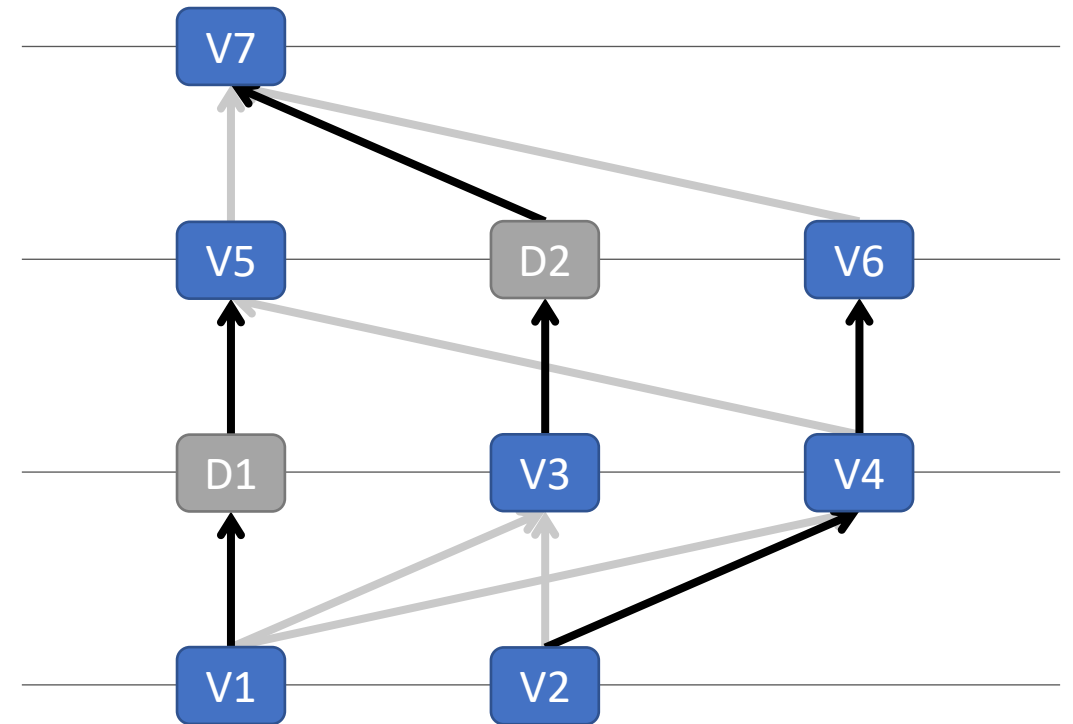
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
- Ausrichtung der Knoten entlang verbliebener Kanten
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



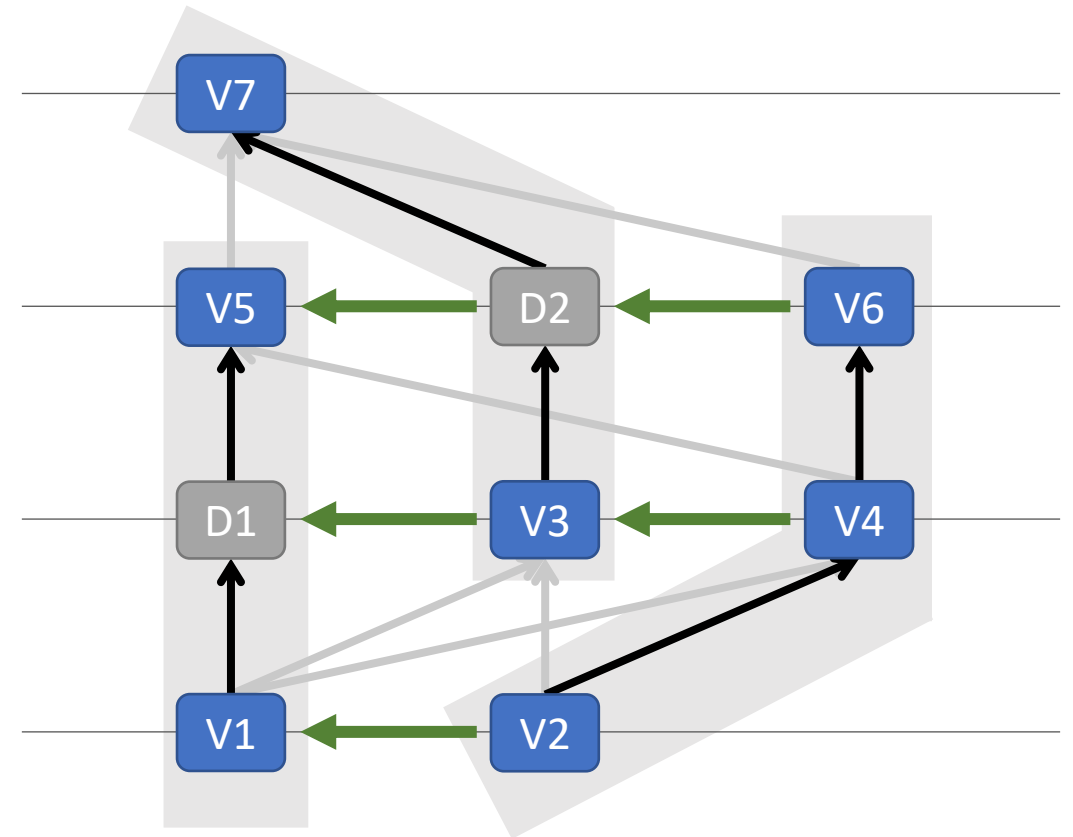
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
- **Ausrichtung der Knoten entlang verbliebener Kanten**
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



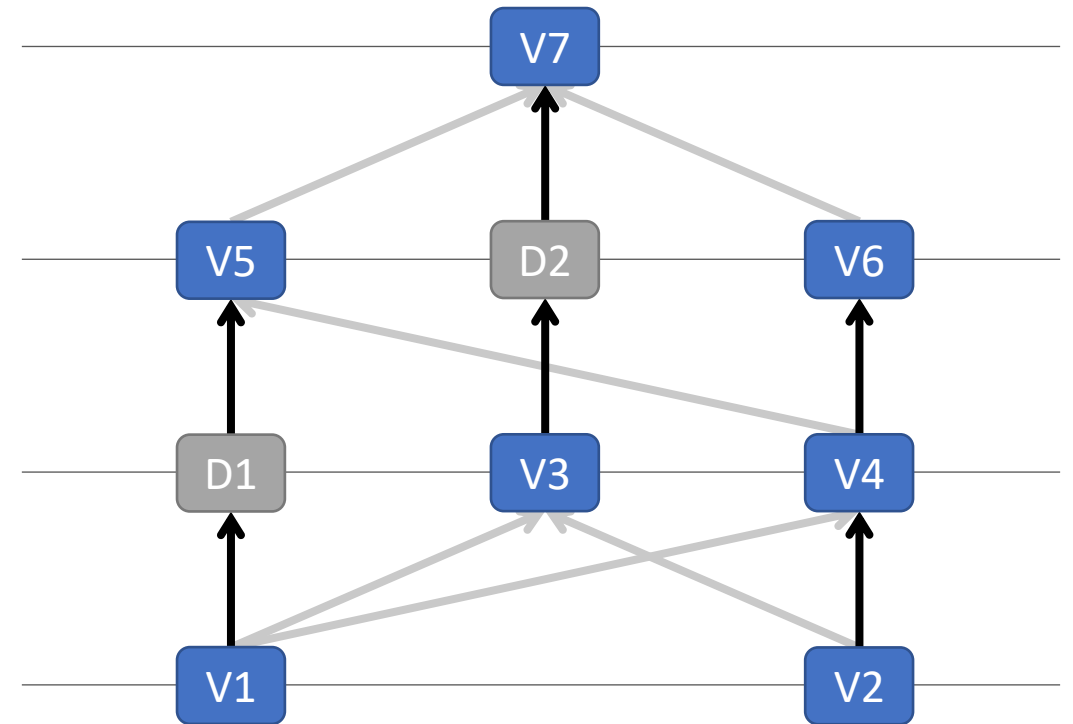
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
- **Ausrichtung der Knoten entlang verbliebener Kanten**
 - Fehler gefunden und korrigiert
 - Erratum Paper mit Ulrik Brandes
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



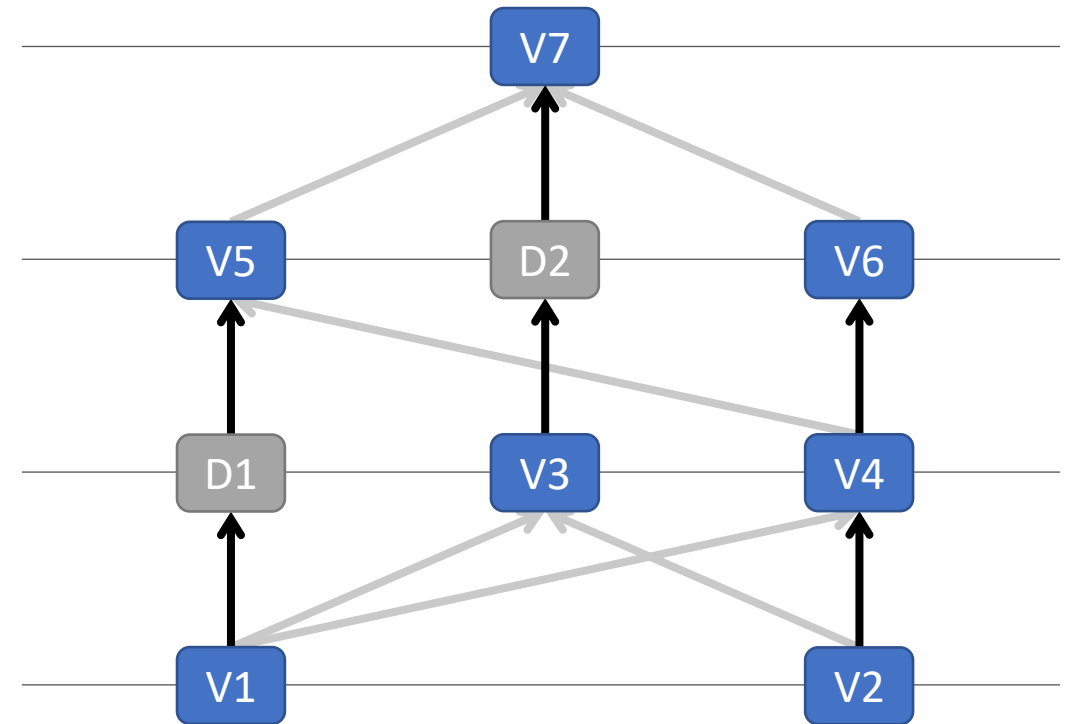
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
- **Ausrichtung der Knoten entlang verbliebener Kanten**
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



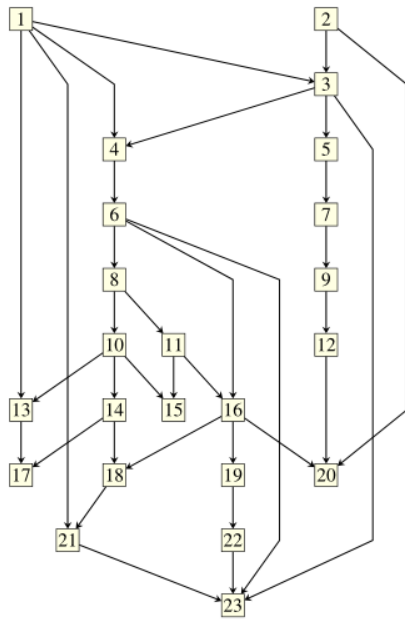
Algorithmus – Knotenpositionierung

- Entfernen von Kanten
- Ausrichtung der Knoten entlang verbliebener Kanten
- **Mehrfachausführung und Positionierung auf arithmetisches Mittel**

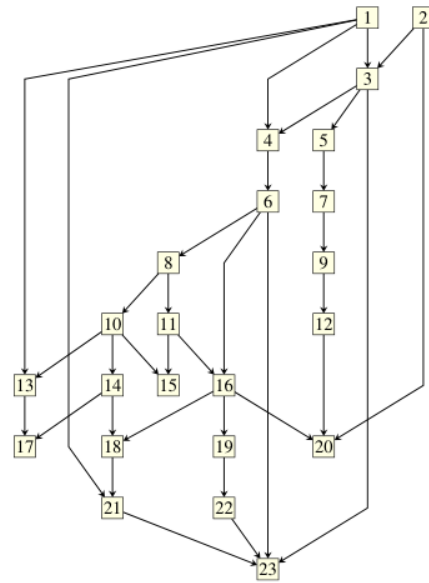


Algorithmus – Knotenpositionierung

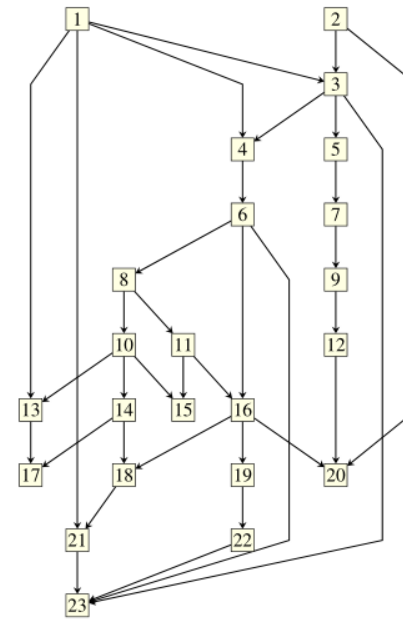
- Mehrfachausführung und Positionierung auf arithmetisches Mittel



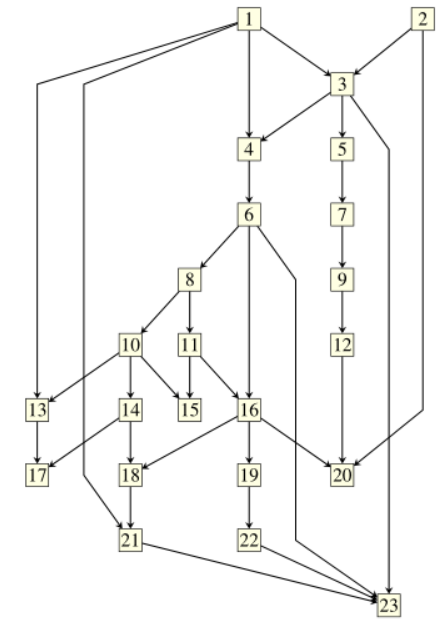
leftmost upper



rightmost upper



leftmost lower



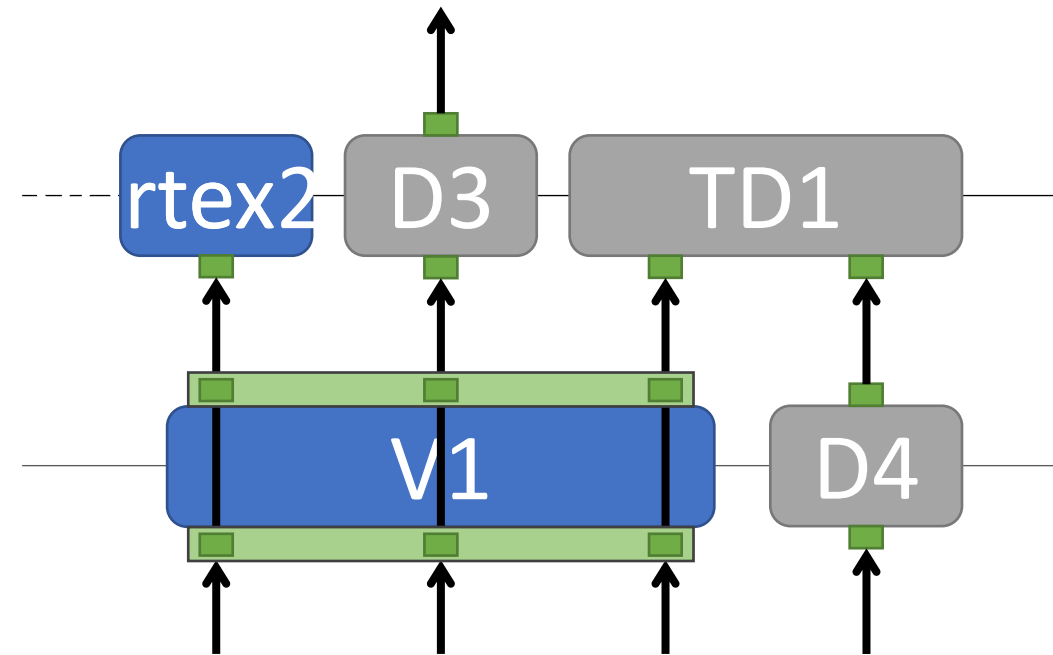
rightmost lower

Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- Umwandlung der Knoten

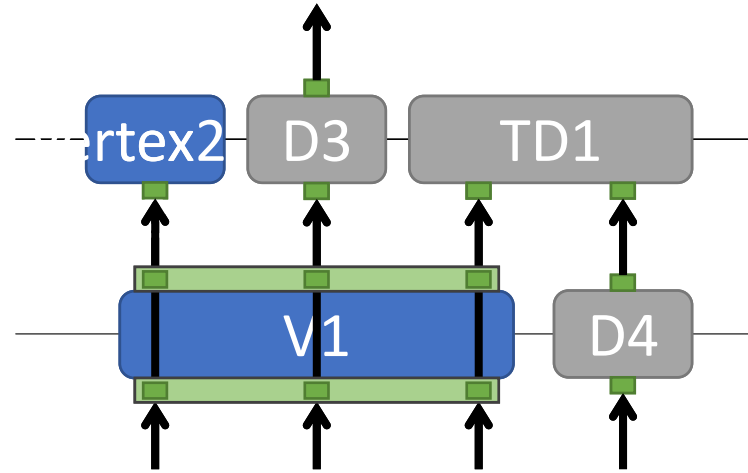
Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- **Umwandlung der Knoten**



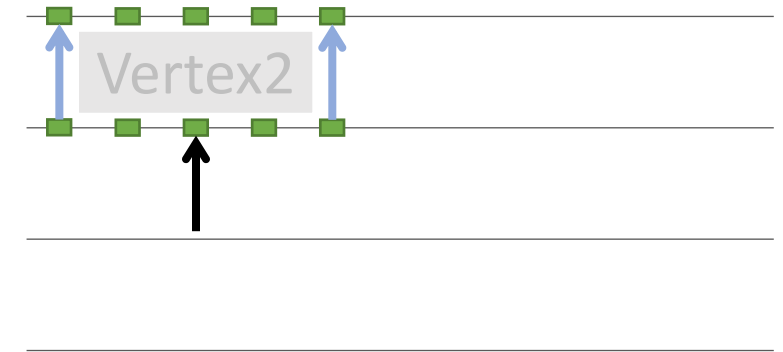
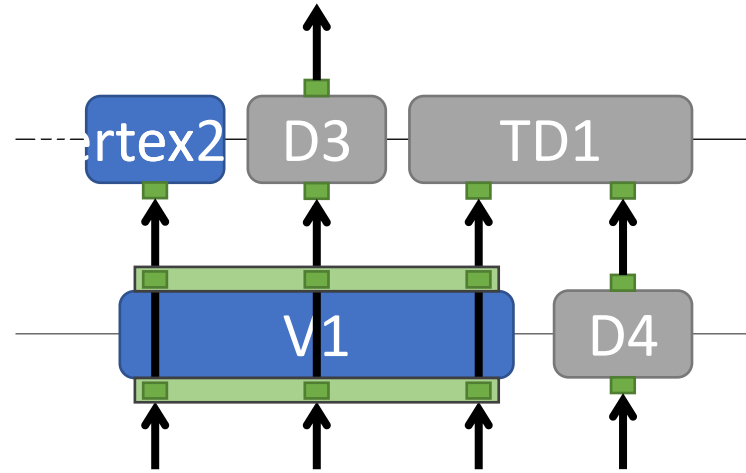
Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- **Umwandlung der Knoten**
 - Normaler Knoten
 - Berücksichtigung der Beschriftung
 - Dummyknoten
 - Knoten mit Portpaar



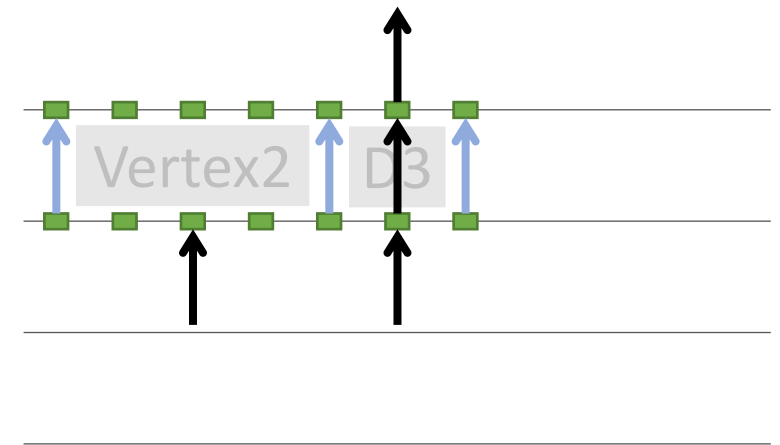
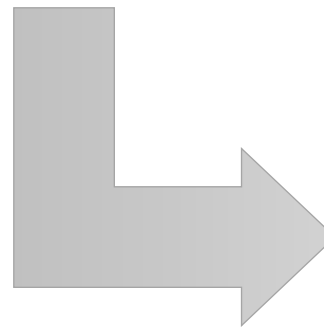
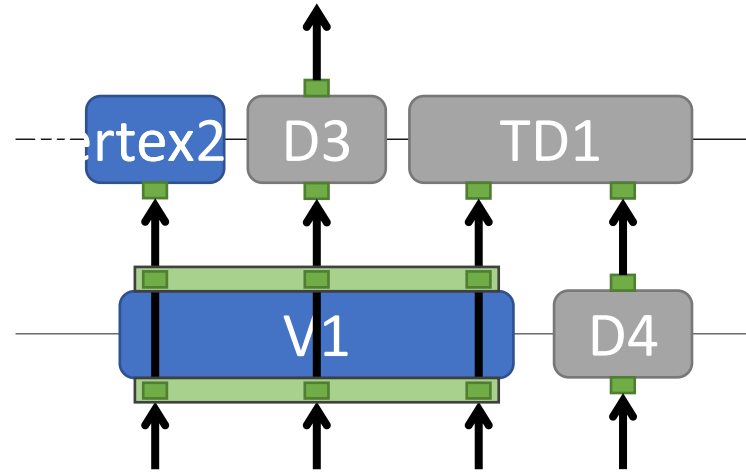
Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- Umwandlung der Knoten
 - Normaler Knoten
 - **Berücksichtigung der Beschriftung**
 - Dummyknoten
 - Knoten mit Portpaar



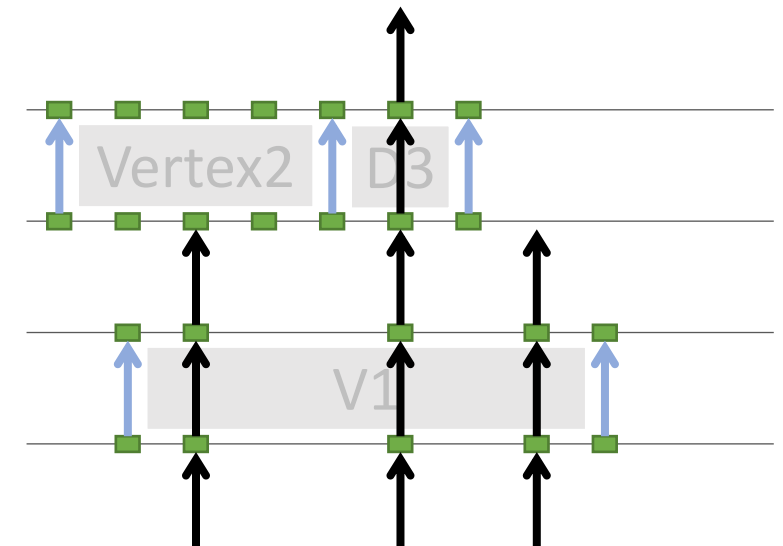
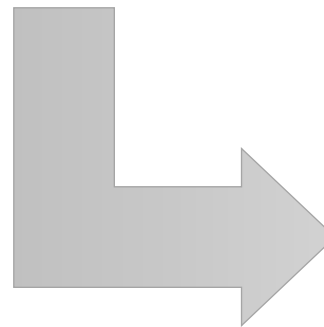
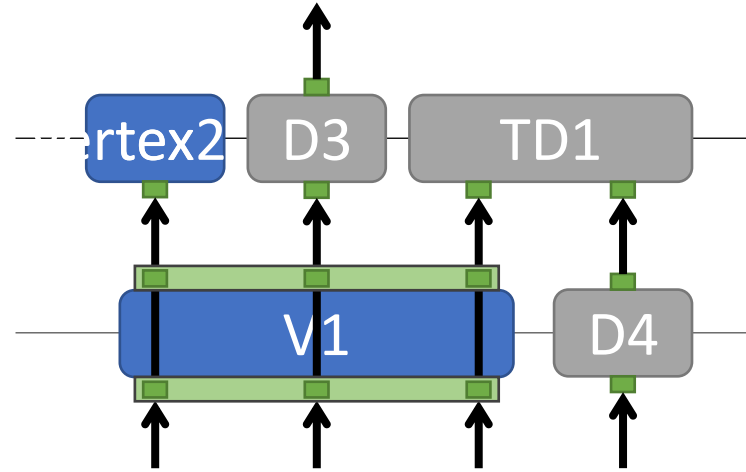
Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- Umwandlung der Knoten
 - Normaler Knoten
 - Berücksichtigung der Beschriftung
 - **Dummyknoten**
 - Knoten mit Portpaar



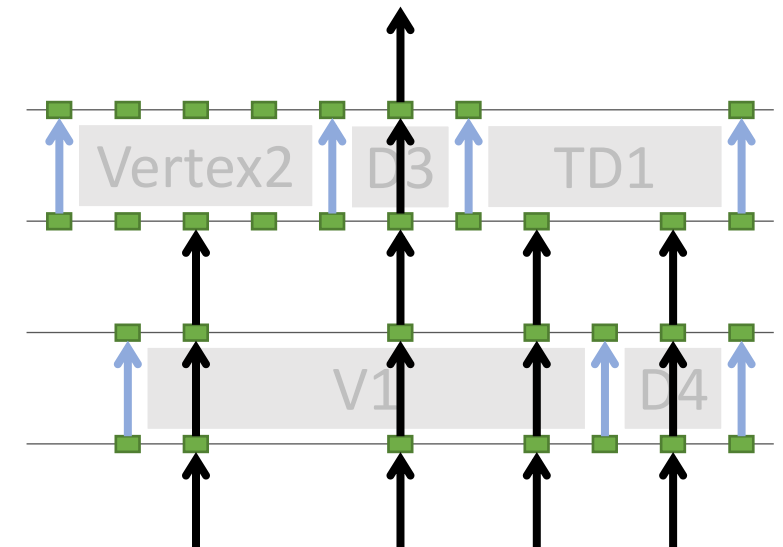
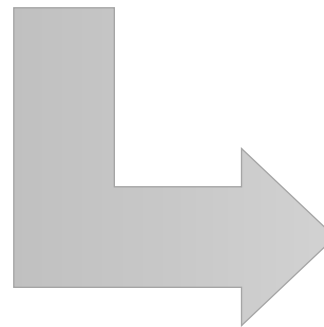
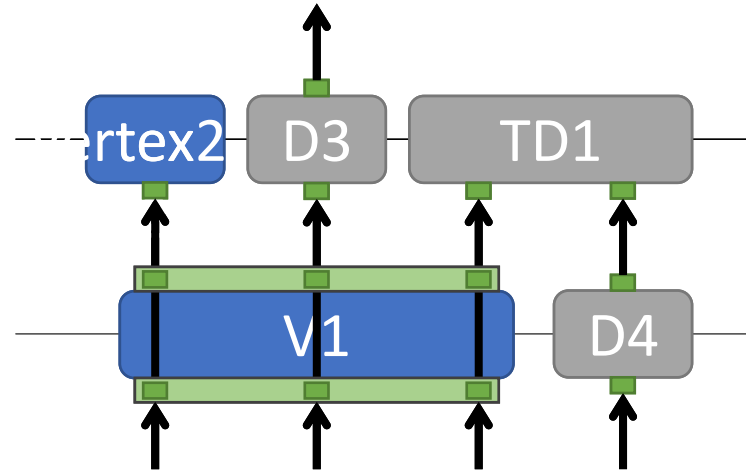
Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- Umwandlung der Knoten
 - Normaler Knoten
 - Berücksichtigung der Beschriftung
 - Dummyknoten
 - **Knoten mit Portpaar**



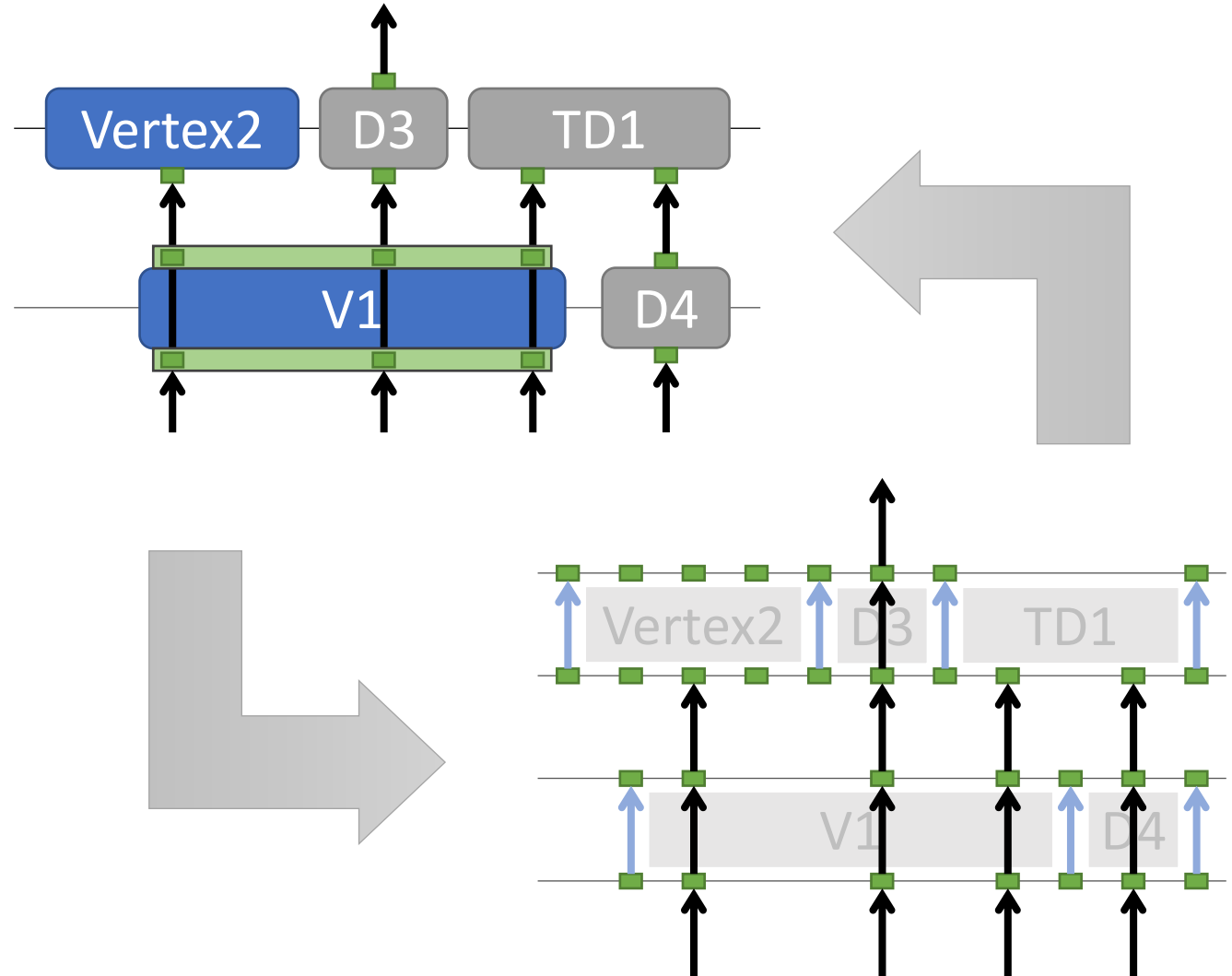
Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- Umwandlung der Knoten
 - Normaler Knoten
 - Berücksichtigung der Beschriftung
 - Dummyknoten
 - Knoten mit Portpaar



Algorithmus – Knotenpositionierung

- Nach Brandes und Köpf
- Umwandlung der Knoten
 - Normaler Knoten
 - Berücksichtigung der Beschriftung
 - Dummyknoten
 - Knoten mit Portpaar



Algorithmus

- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

Algorithmus – Kantenführung

Algorithmus – Kantenführung

- Nicht umgesetzt aus Zeitgründen
- Später umgesetzt im Rahmen des Papers

Layered Drawing of Undirected Graphs with Generalized Port Constraints

Julian Walter, Johannes Zink, Joachim Baumeister, Alexander Wolff

[GD 2020]

Algorithmus

- 0: Vorverarbeitung
- 1: Richtungszuweisung
- 2: Lagenzuordnung
- 3: Dummyknotenerstellung
- 4: Kreuzungsreduzierung
- 5: Knotenpositionierung
- 6: Kantenführung

Tests

Tests

- Richtungszuweisung
 - kräftebasiert
 - Breitensuche
 - zufällig
- Kreuzungsreduzierung
 - Knotenkoordinaten
 - Portkoordinaten
- Visueller Vergleich
 - Schulze et al.
 - dieses Verfahren

Tests

- **Richtungszuweisung**

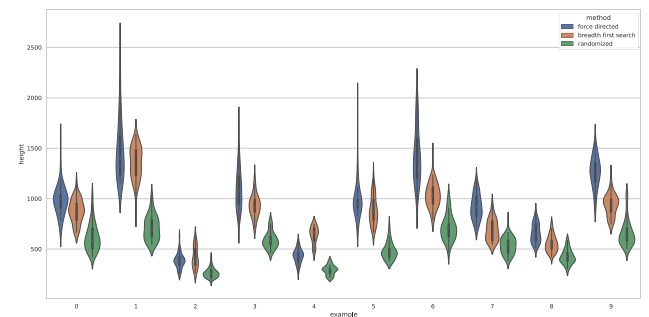
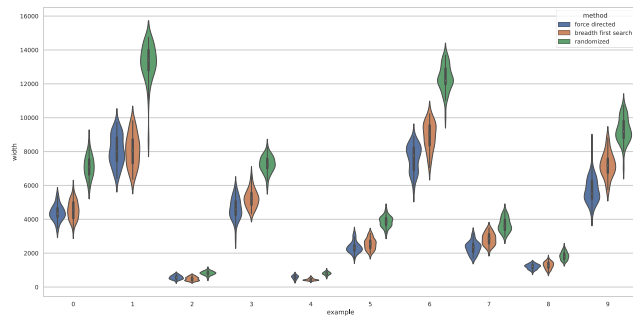
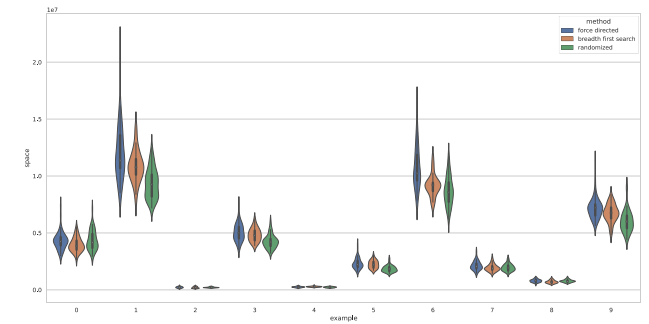
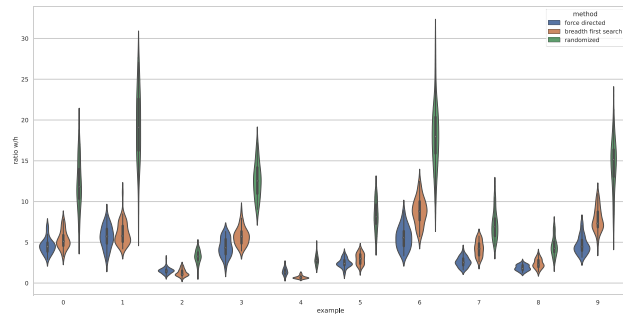
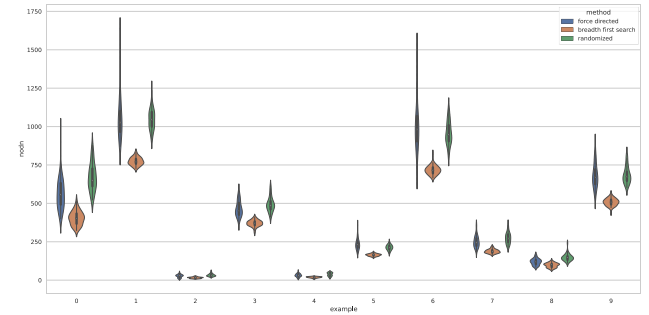
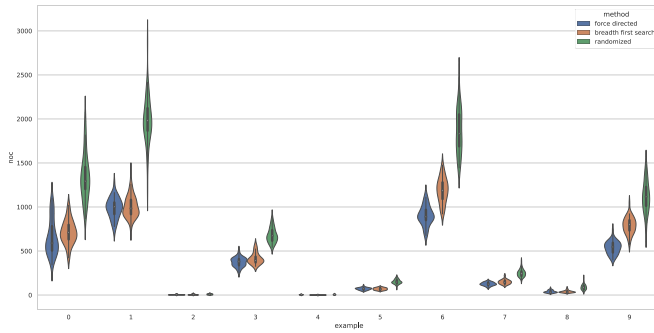
 - kräftebasiert
 - Breitensuche
 - zufällig

- **Kreuzungsreduzierung**

 - Knotenkoordinaten
 - Portkoordinaten

- **Visueller Vergleich**

 - Schulze et al.
dieses Verfahren



Tests

- **Richtungszuweisung**

kräftebasiert

Breitensuche

zufällig

- **Kreuzungsreduzierung**

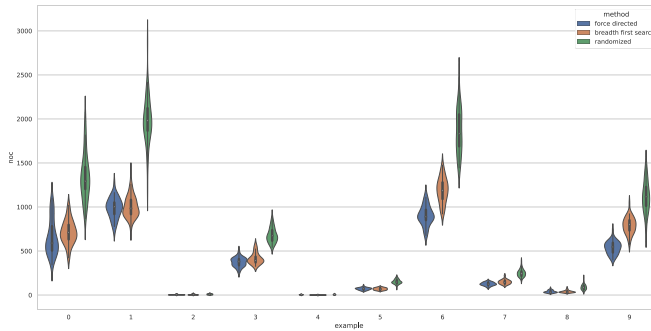
Knotenkoordinaten

Portkoordinaten

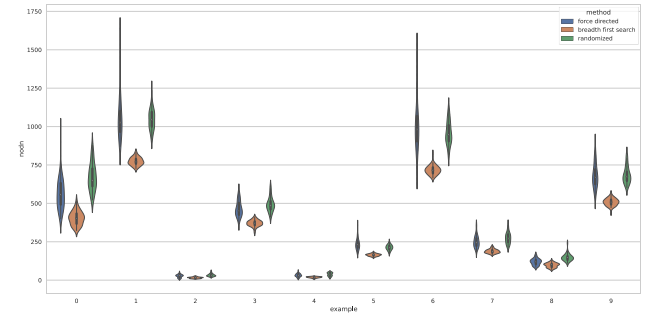
- **Visueller Vergleich**

Schulze et al.

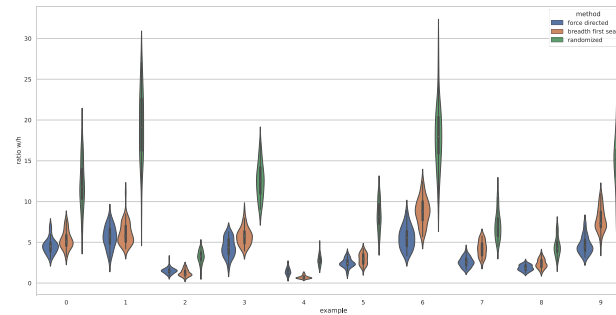
dieses Verfahren



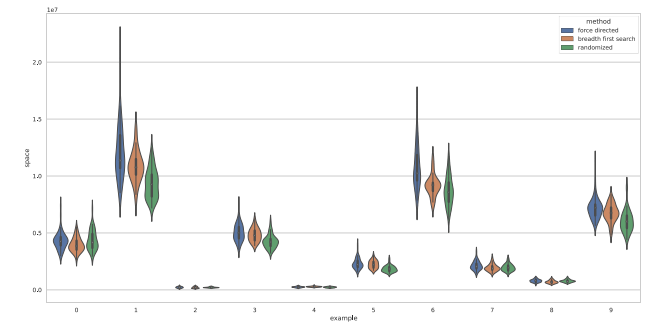
Kreuzungen



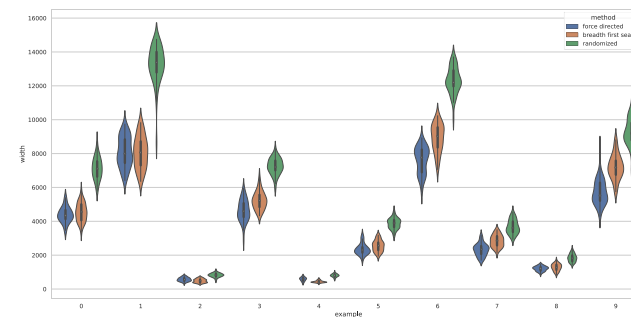
Dummyknoten



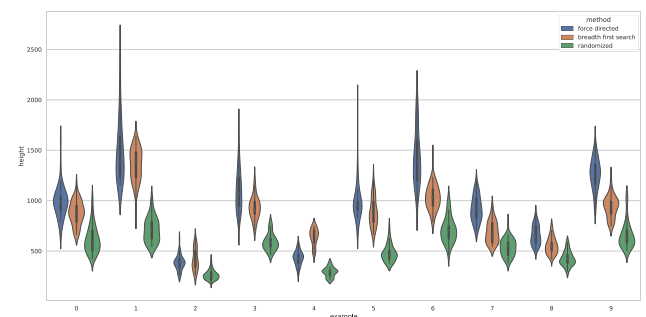
Seitenverhältnis



Fläche



Breite



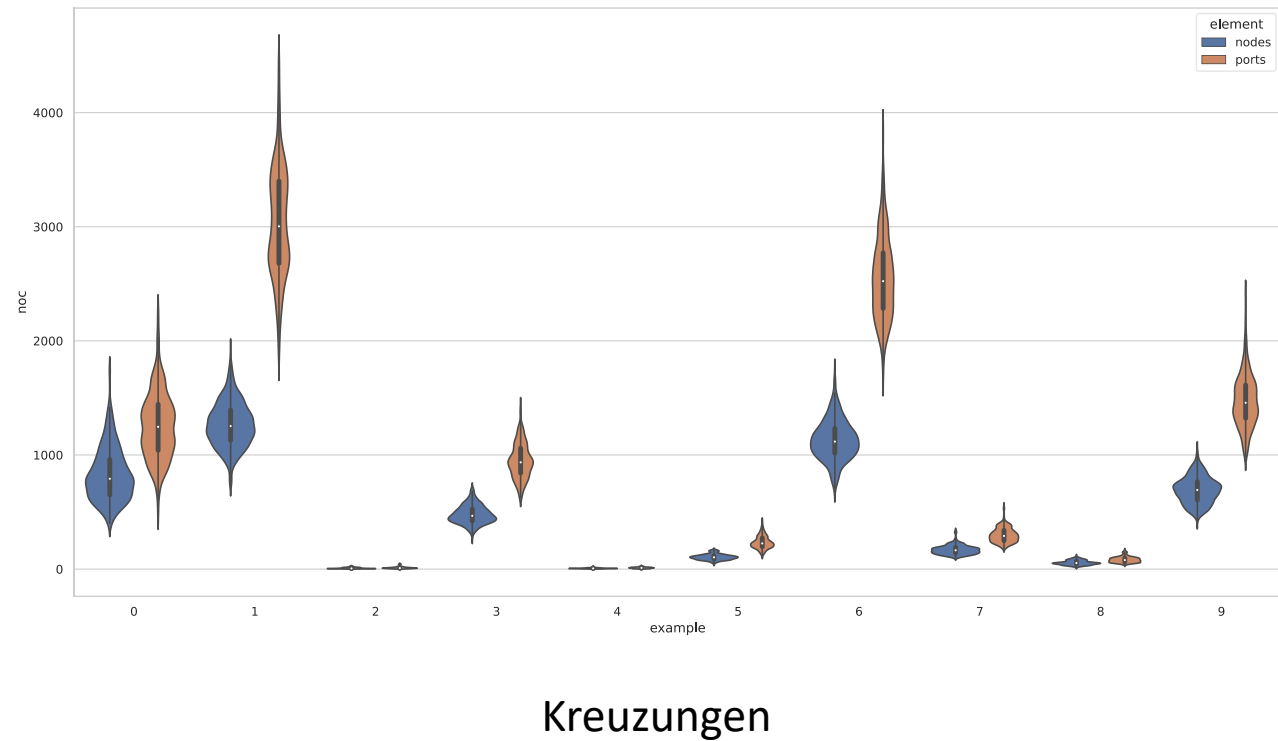
Höhe

Tests

- Richtungszuweisung
 - kräftebasiert
 - Breitensuche
 - zufällig
- Kreuzungsreduzierung
 - Knotenkoordinaten
 - Portkoordinaten
- Visueller Vergleich
 - Schulze et al.
 - dieses Verfahren

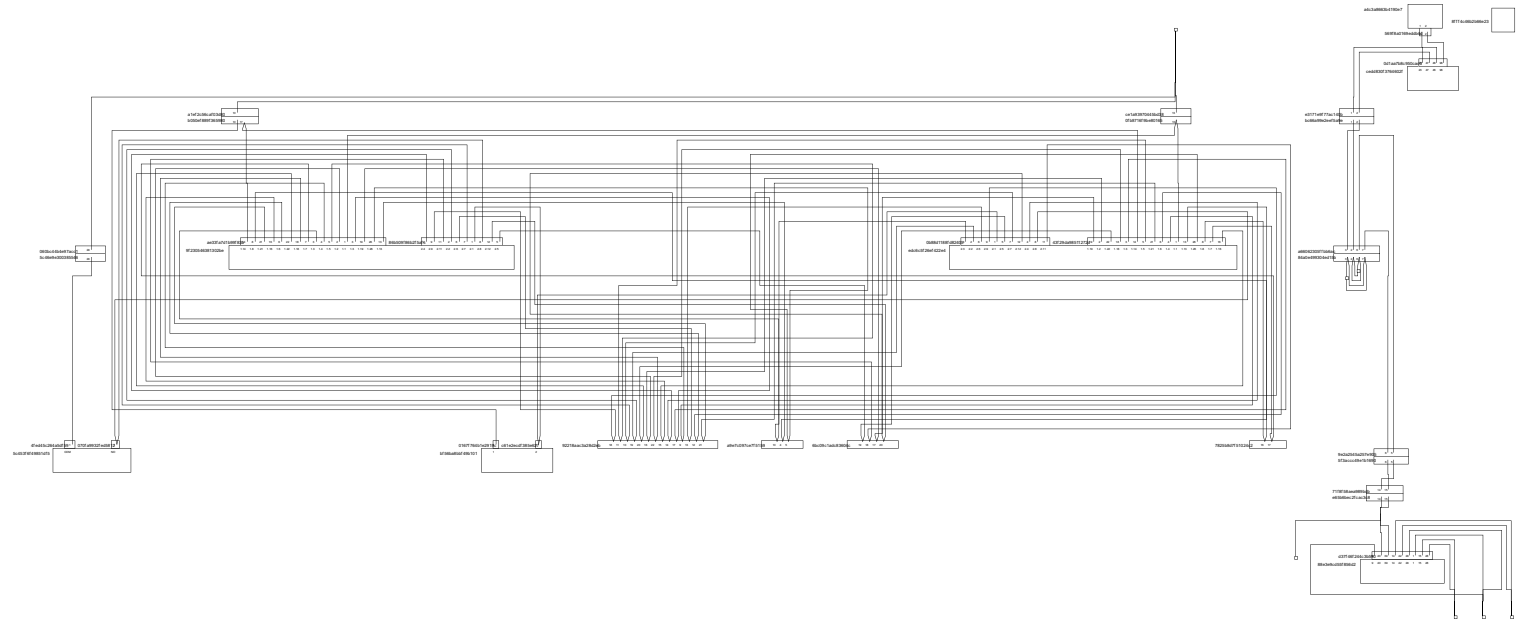
Tests

- Richtungszuweisung
 - kräftebasiert
 - Breitensuche
 - zufällig
- **Kreuzungsreduzierung**
 - Knotenkoordinaten
 - Portkoordinaten
- Visueller Vergleich
 - Schulze et al.
 - dieses Verfahren

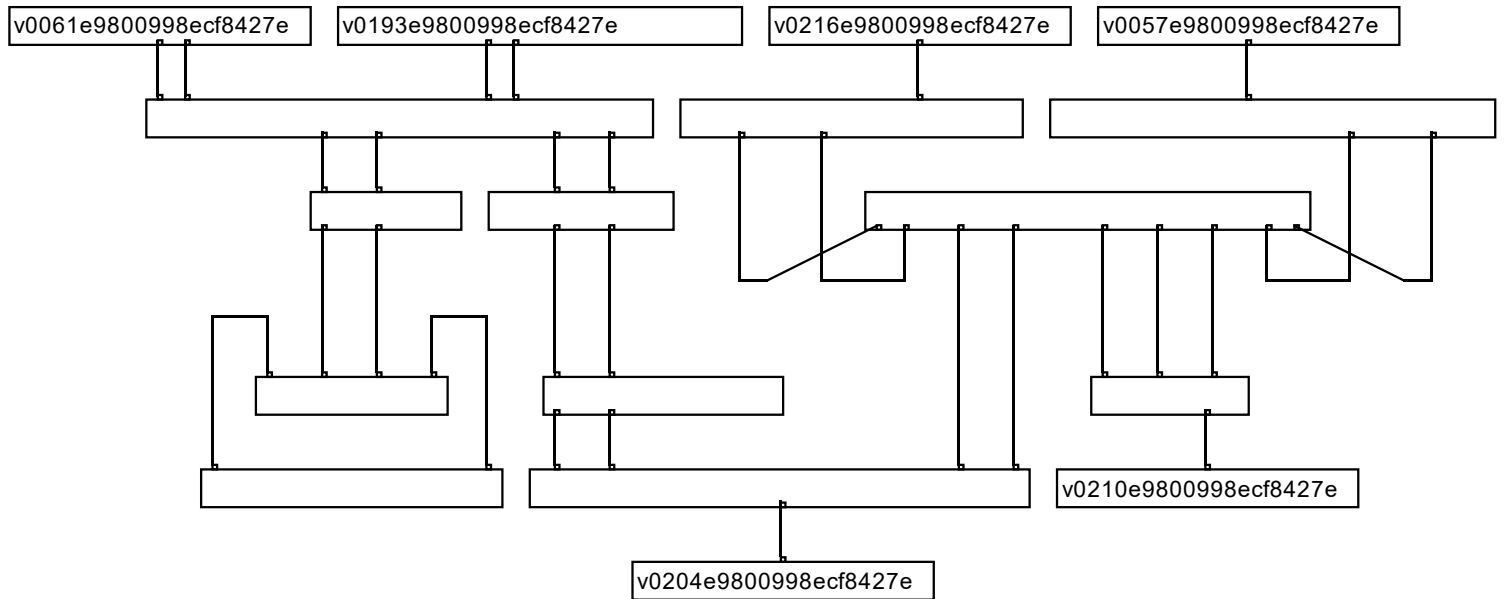


Tests

- **Richtungszuweisung**
 - kräftebasiert
 - Breitensuche
 - zufällig
- **Kreuzungsreduzierung**
 - Knotenkoordinaten
 - Portkoordinaten
- **Visueller Vergleich**
 - Schulze et al.
 - dieses Verfahren

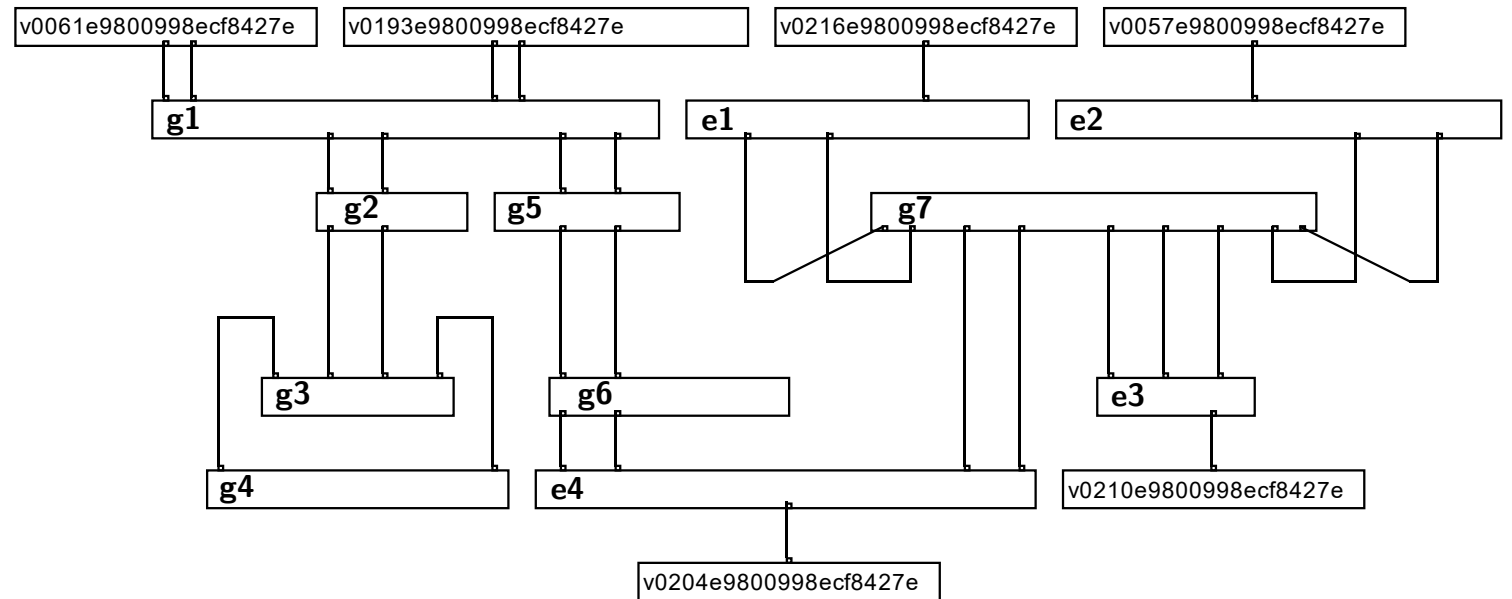
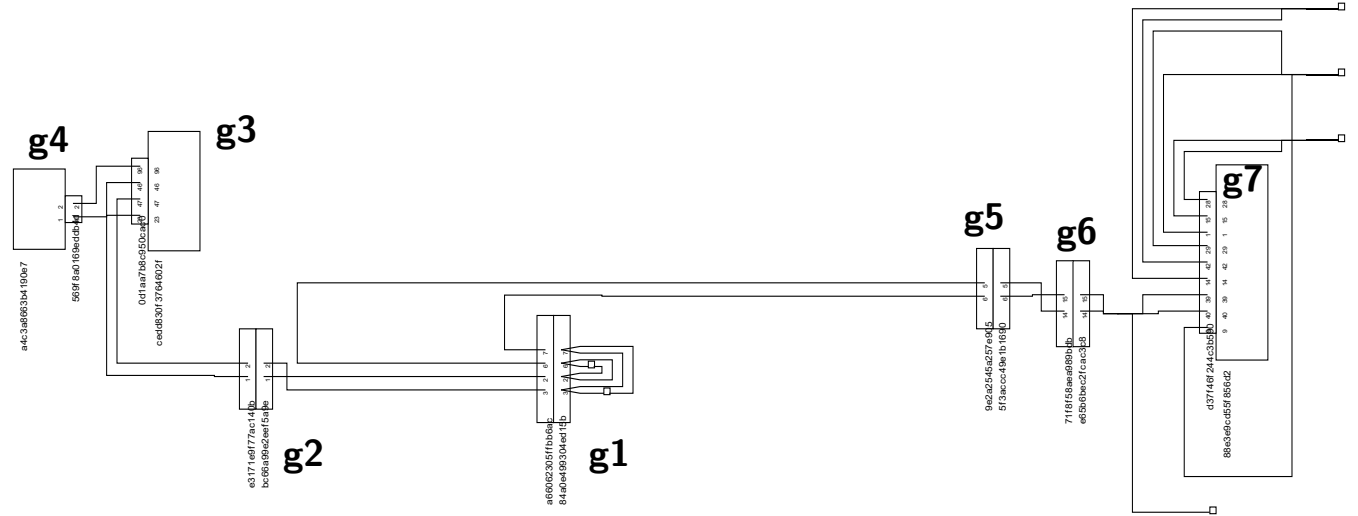


(50) vom drahtlosen layouting website: WebCableSettings@Dnet1400



Tests

- **Richtungszuweisung**
 - kräftebasiert
 - Breitensuche
 - zufällig
- **Kreuzungsreduzierung**
 - Knotenkoordinaten
 - Portkoordinaten
- **Visueller Vergleich**
 - Schulze et al.
 - dieses Verfahren



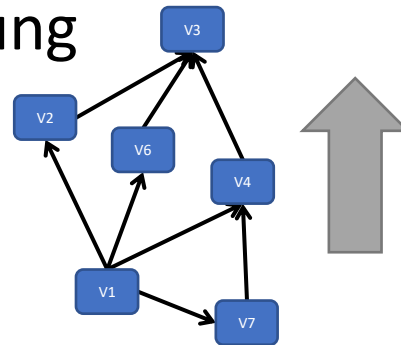
Zusammenfassung

0: Vorverarbeitung

zwei Ports je Kante
eine Kante je Port

1: Richtungszuweisung

kräftebasiert
Breitensuche
zufällig

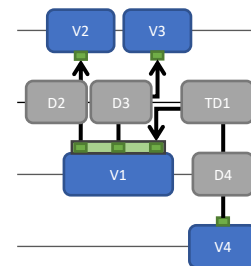


2: Lagenzuordnung

Network Simplex
nach Gansner et al.

3: Dummyknotenerstellung

Tourning Dummies

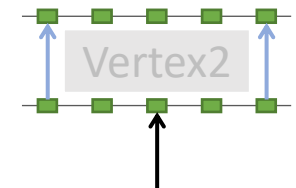


4: Kreuzungsreduzierung

Portreihenfolge
Knotenkoordinaten

5: Knotenpositionierung

Umwandlung in Ports und Kanten
nach Brandes und Köpf



6: Kantenführung

Tests:

kräftebasiert und Knotenreihenfolge
am besten

