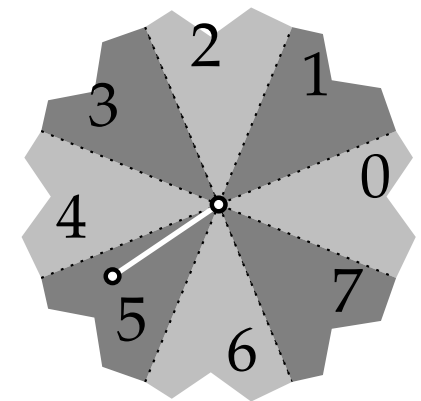
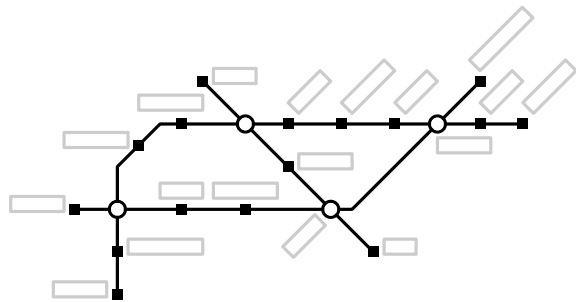
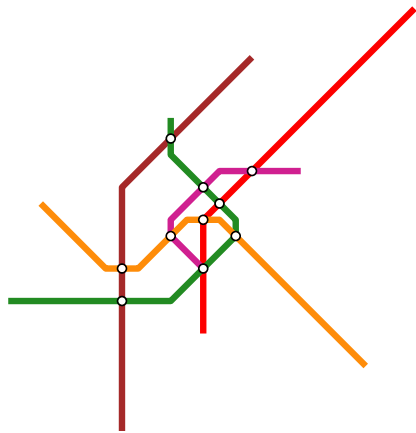


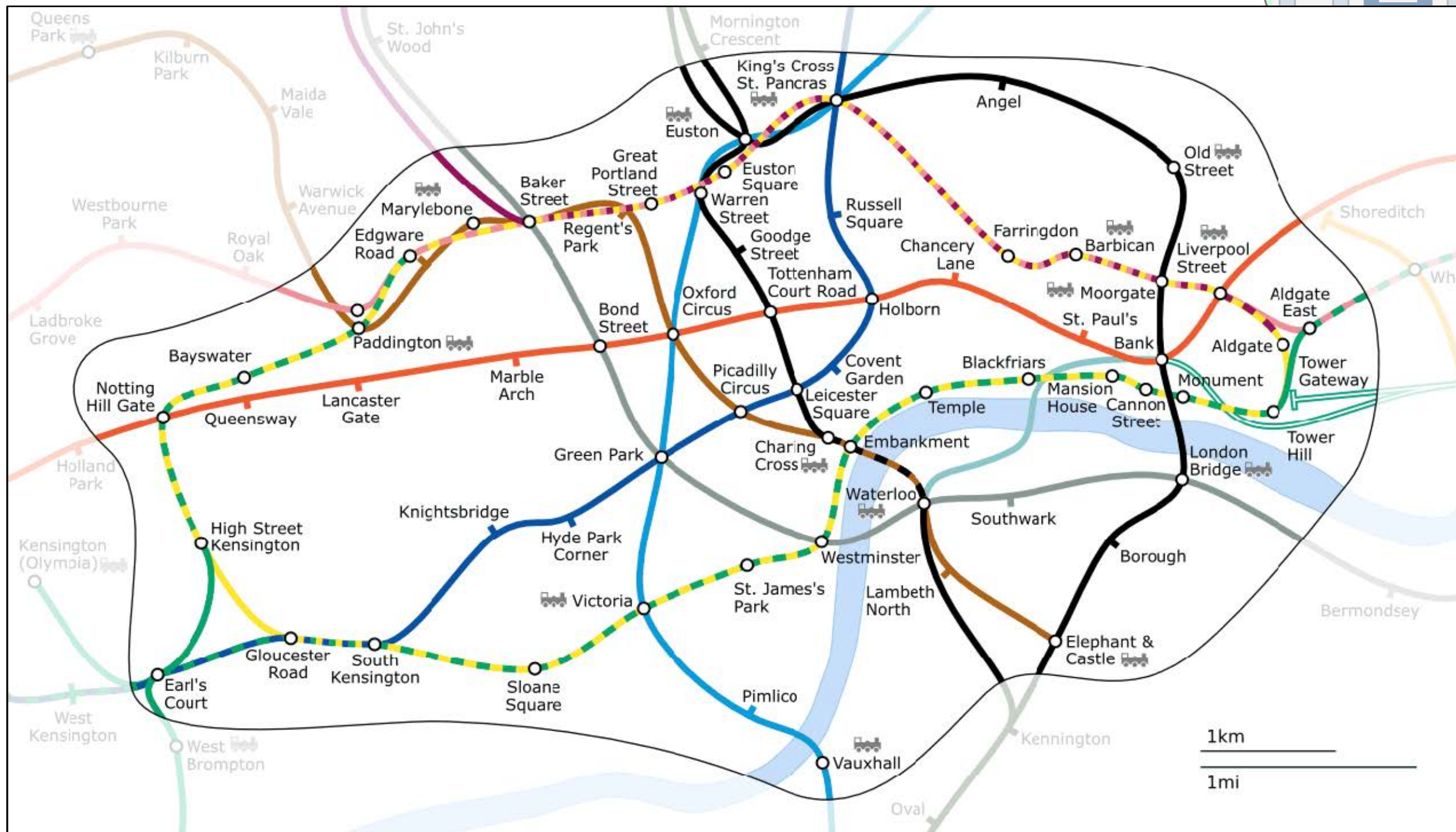
Algorithmen für geographische Informationssysteme

11. Vorlesung Netzpläne (Metro Maps)

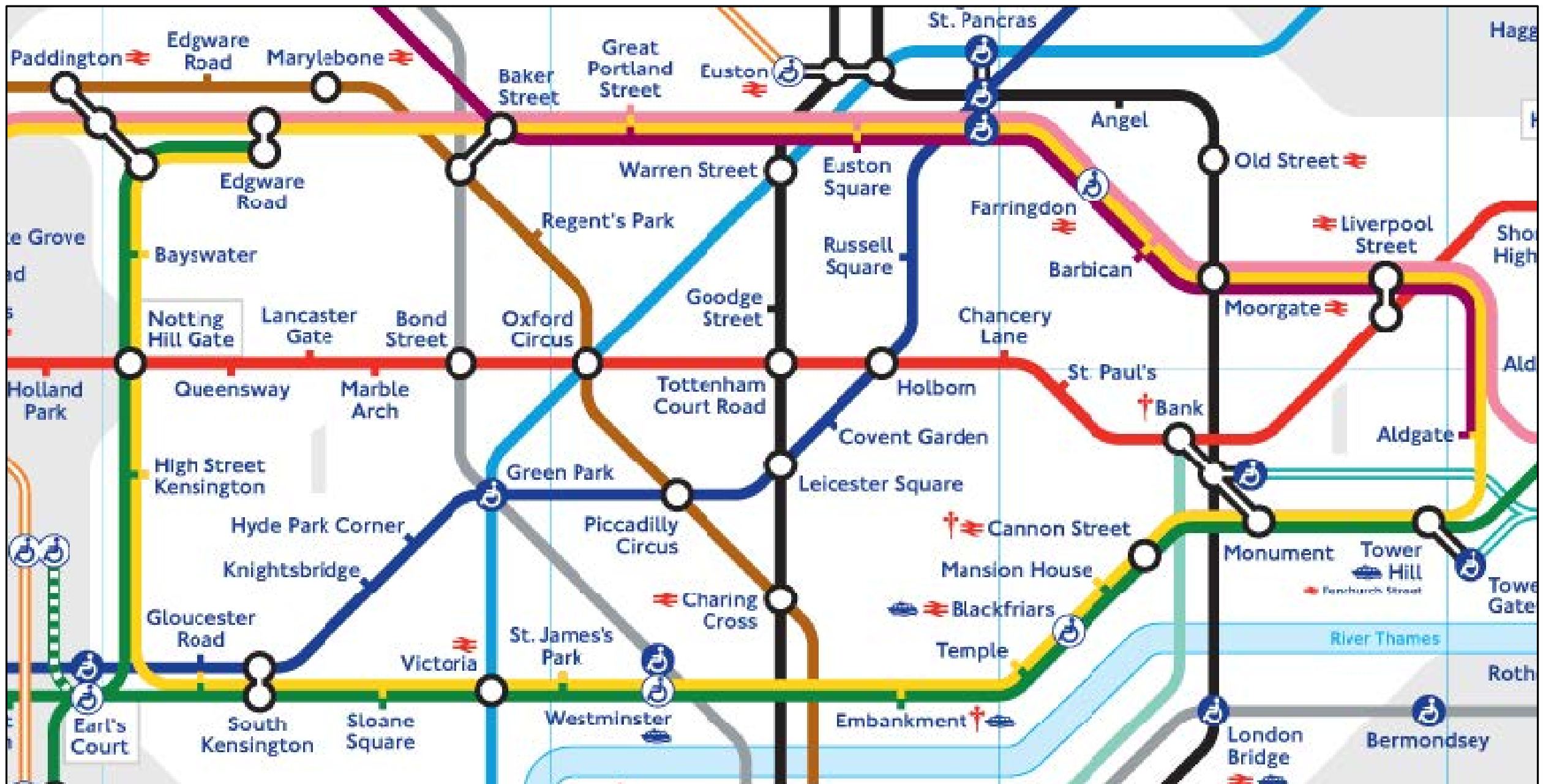
Teil I: Schematische Netzpläne



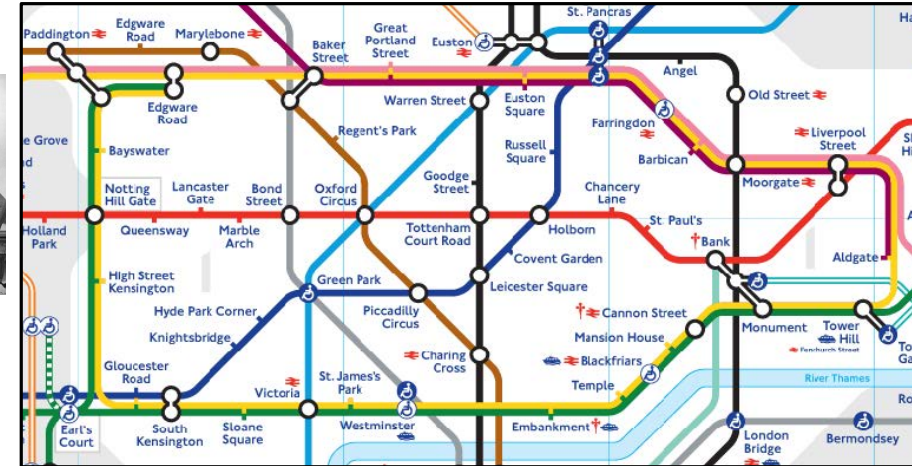
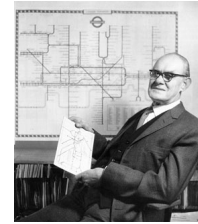
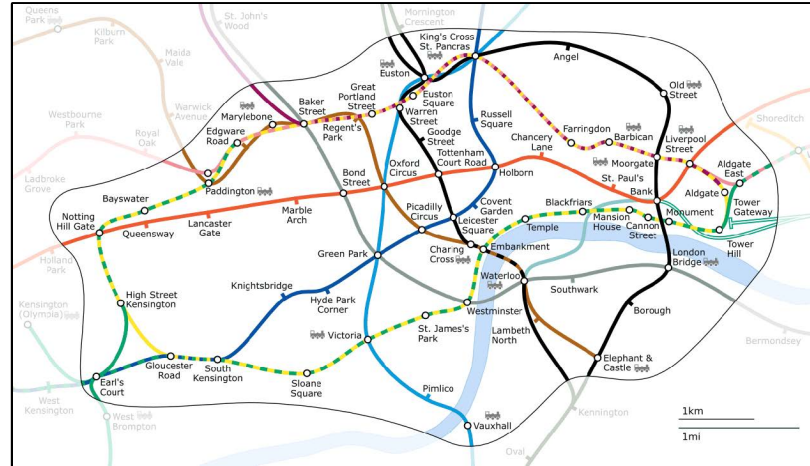
Was ist ein schematischer Netzplan?



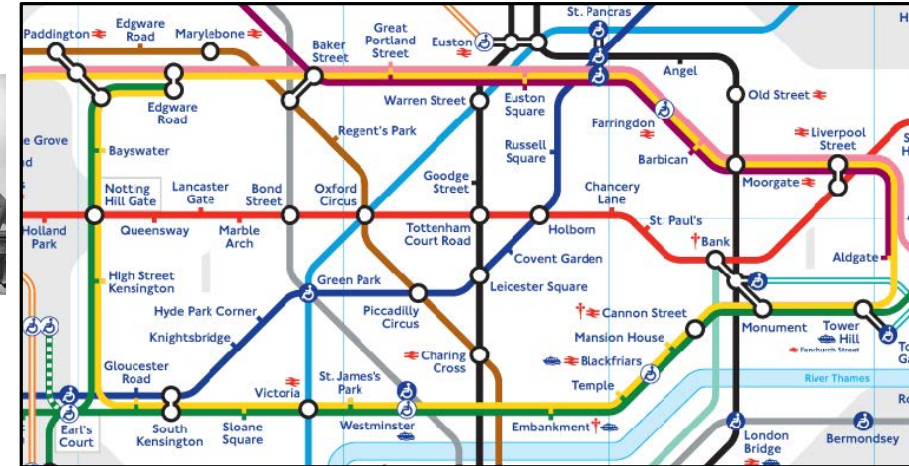
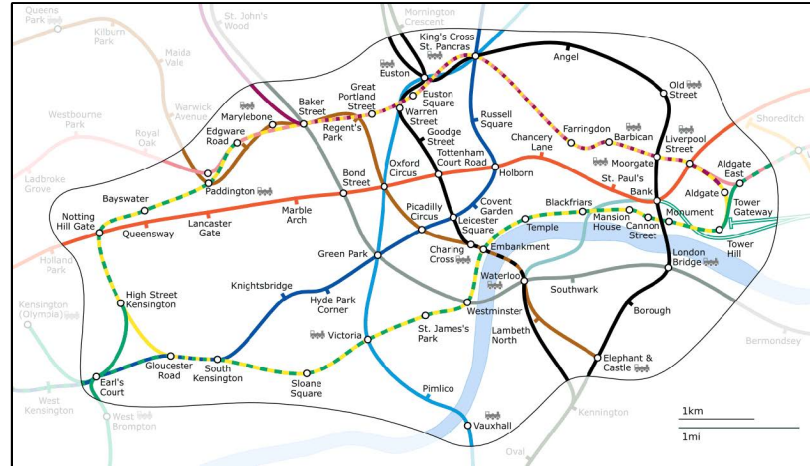
Was ist ein schematischer Netzplan?



Was ist ein schematischer Netzplan?

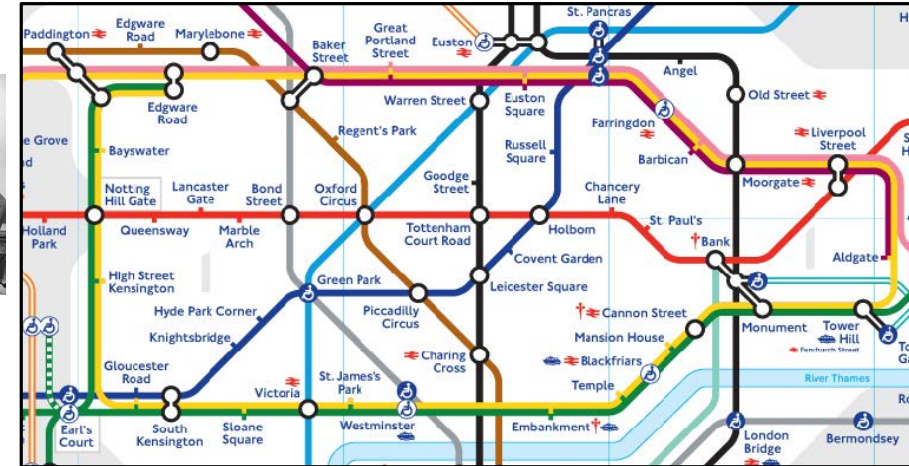
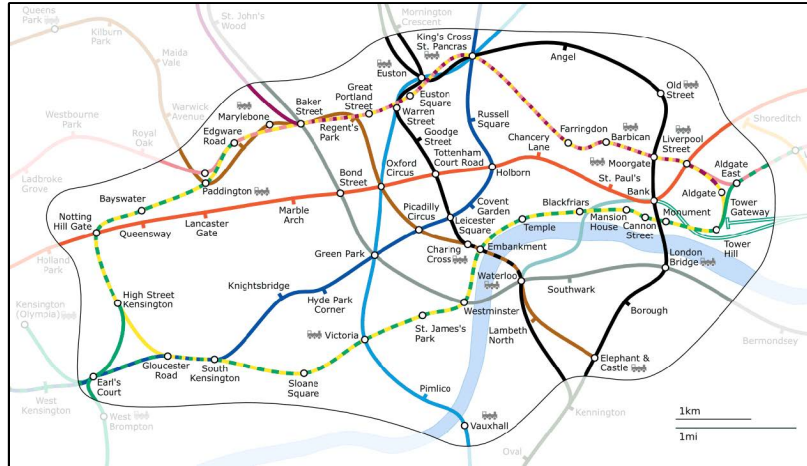


Was ist ein schematischer Netzplan?



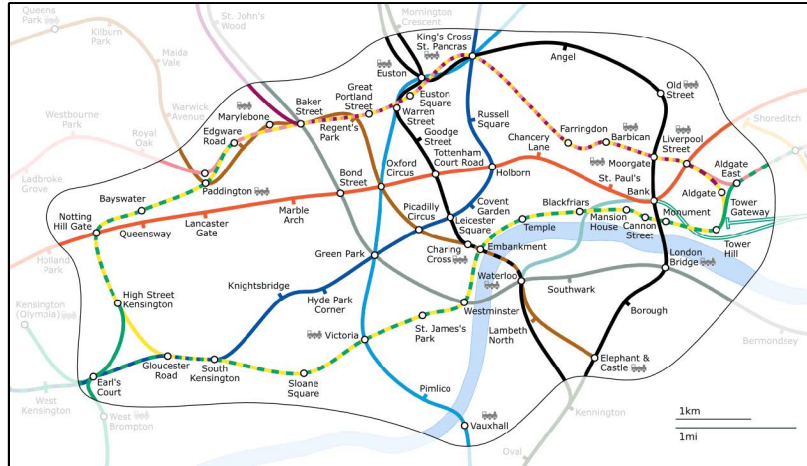
- Karte/Diagramm, das Stationen zeigt, die durch U-Bahn-Linien verbunden sind

Was ist ein schematischer Netzplan?



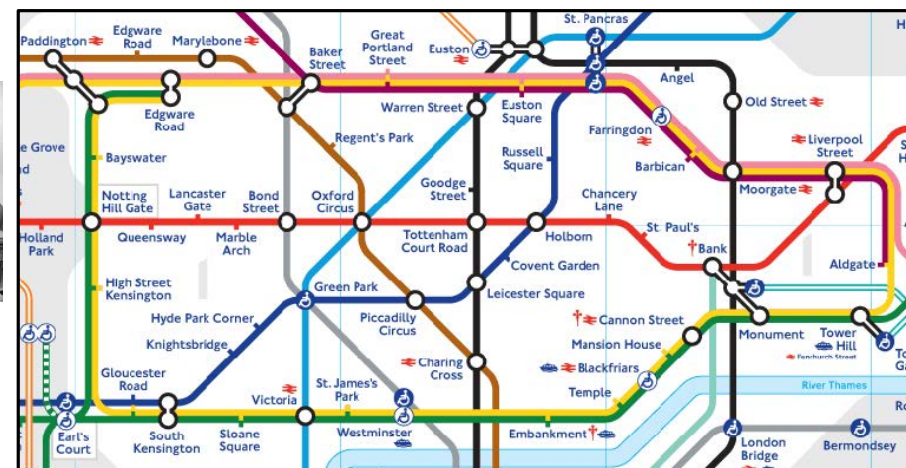
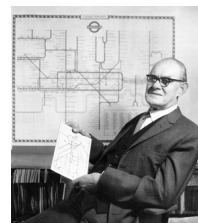
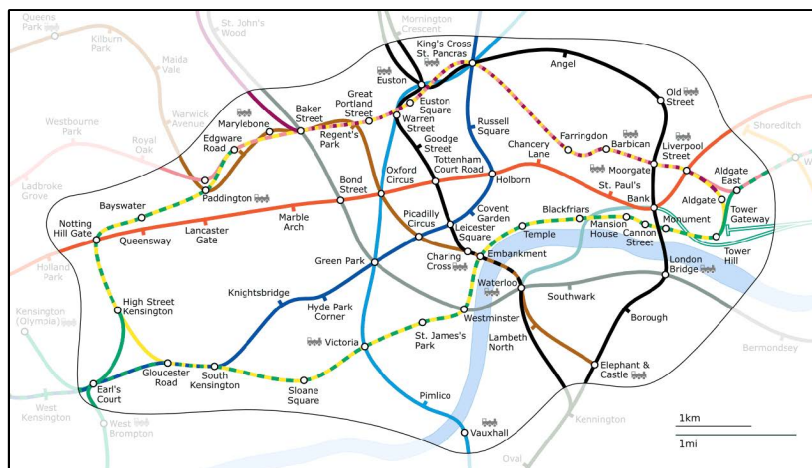
- Karte/Diagramm, das Stationen zeigt, die durch U-Bahn-Linien verbunden sind
- Fokus auf Topologie statt Topografie

Was ist ein schematischer Netzplan?



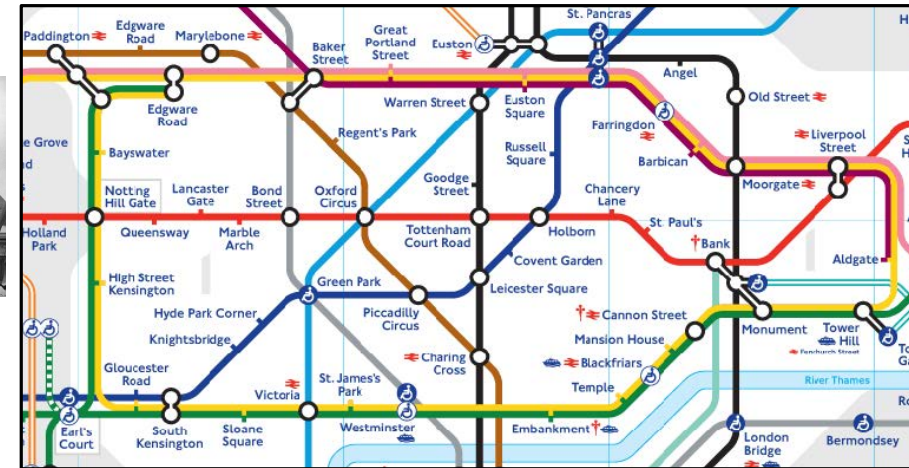
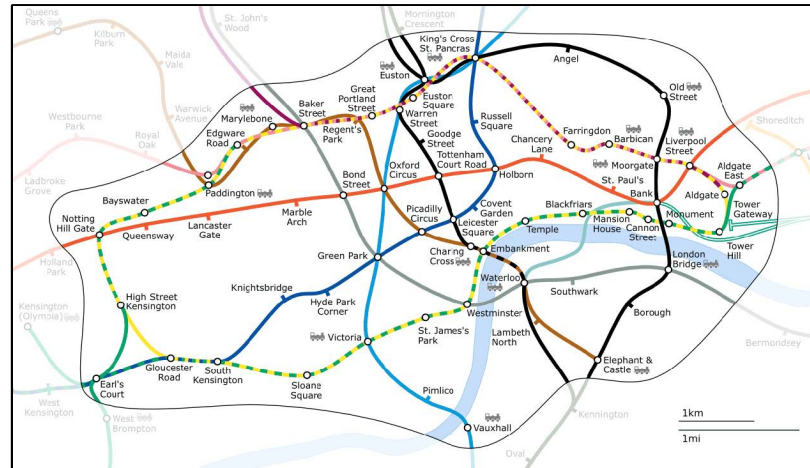
- Karte/Diagramm, das Stationen zeigt, die durch U-Bahn-Linien verbunden sind
- Fokus auf Topologie statt Topografie
- Ziel: eine benutzerfreundliche visuelle Orientierungshilfe für Fahrgäste

Was ist ein schematischer Netzplan?



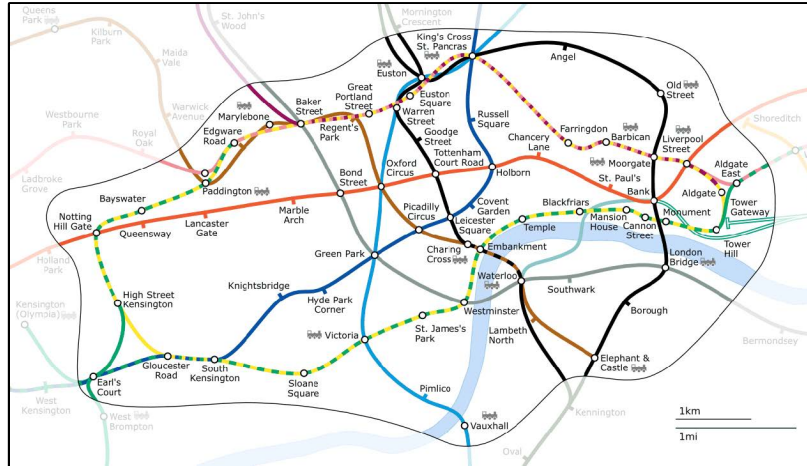
- Karte/Diagramm, das Stationen zeigt, die durch U-Bahn-Linien verbunden sind
- Fokus auf Topologie statt Topografie
- Ziel: eine benutzerfreundliche visuelle Orientierungshilfe für Fahrgäste
 - *Wie komme ich schnell von A nach B?"*

Was ist ein schematischer Netzplan?



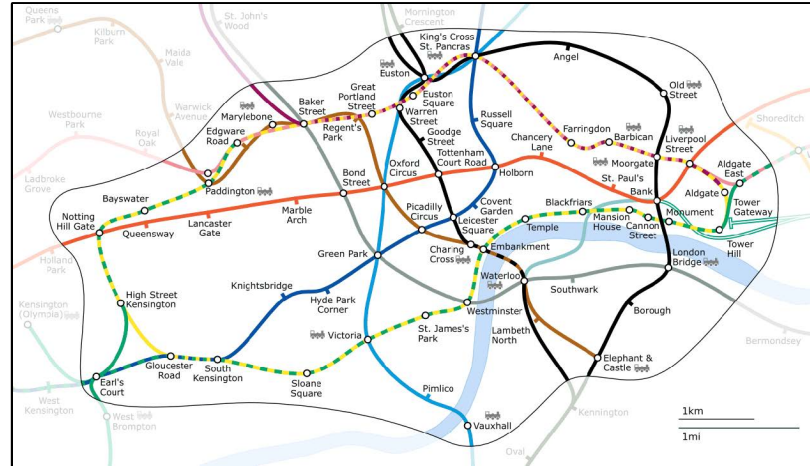
- Karte/Diagramm, das Stationen zeigt, die durch U-Bahn-Linien verbunden sind
- Fokus auf Topologie statt Topografie
- Ziel: eine benutzerfreundliche visuelle Orientierungshilfe für Fahrgäste
 - *Wie komme ich schnell von A nach B?"*
 - *Wo muss ich umsteigen?"*

Was ist ein schematischer Netzplan?



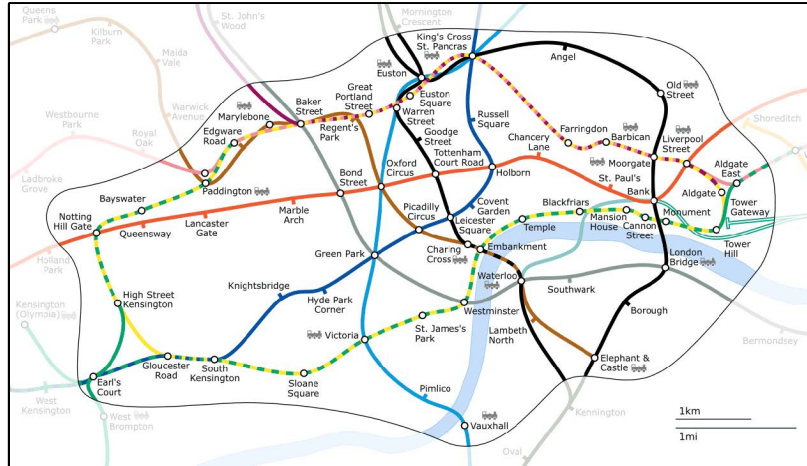
- Karte/Diagramm, das Stationen zeigt, die durch U-Bahn-Linien verbunden sind
- Fokus auf Topologie statt Topografie
- Ziel: eine benutzerfreundliche visuelle Orientierungshilfe für Fahrgäste
 - *Wie komme ich schnell von A nach B?"*
 - *Wo muss ich umsteigen?"*
- verzerrt Maßstab und Geometrie

Was ist ein schematischer Netzplan?



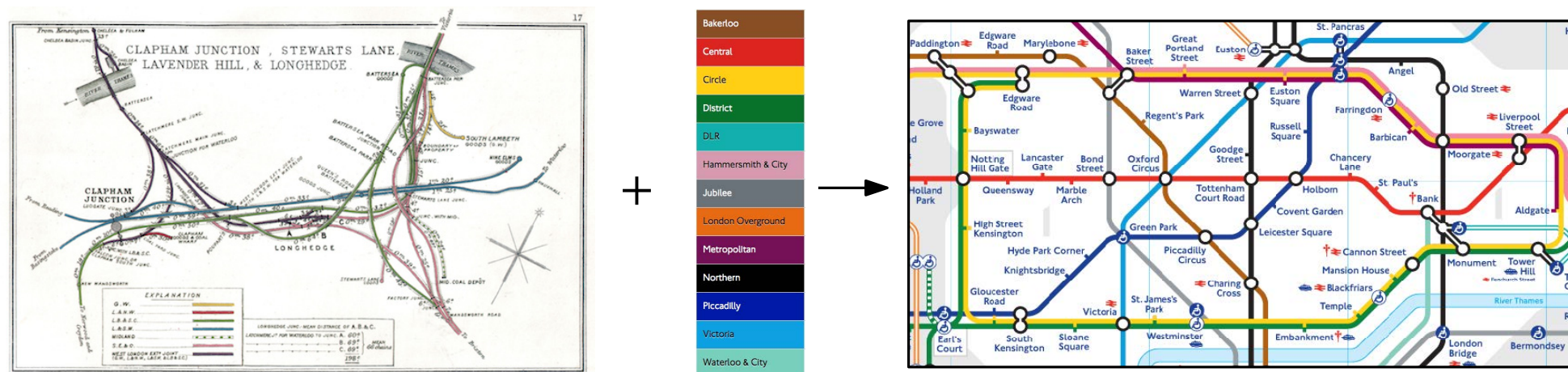
- Karte/Diagramm, das Stationen zeigt, die durch U-Bahn-Linien verbunden sind
- Fokus auf Topologie statt Topografie
- Ziel: eine benutzerfreundliche visuelle Orientierungshilfe für Fahrgäste
 - *Wie komme ich schnell von A nach B?"*
 - *Wo muss ich umsteigen?"*
- verzerrt Maßstab und Geometrie
- Gestaltung von Netzplänen ist nach wie vor ein weitgehend manueller Prozess

Was ist ein schematischer Netzplan?



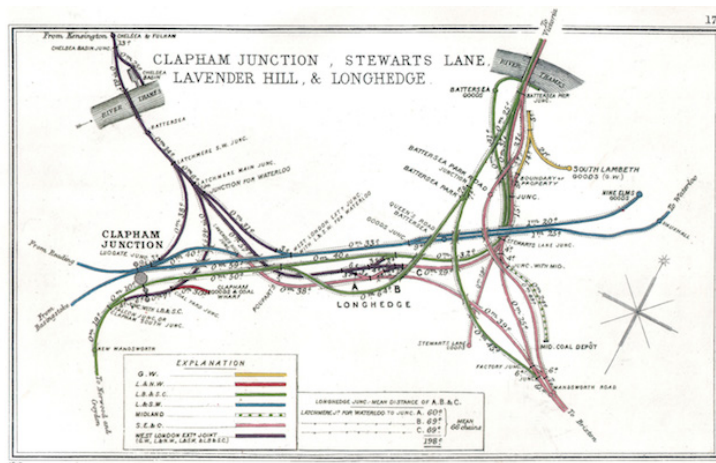
- Karte/Diagramm, das Stationen zeigt, die durch U-Bahn-Linien verbunden sind
- Fokus auf Topologie statt Topografie
- Ziel: eine benutzerfreundliche visuelle Orientierungshilfe für Fahrgäste
 - *Wie komme ich schnell von A nach B?"*
 - *Wo muss ich umsteigen?"*
- verzerrt Maßstab und Geometrie
- Gestaltung von Netzplänen ist nach wie vor ein weitgehend manueller Prozess
- Optimierung des Netzlayouts rechnerisch anspruchsvoll

Aufgaben beim Entwurf eines Netzplans

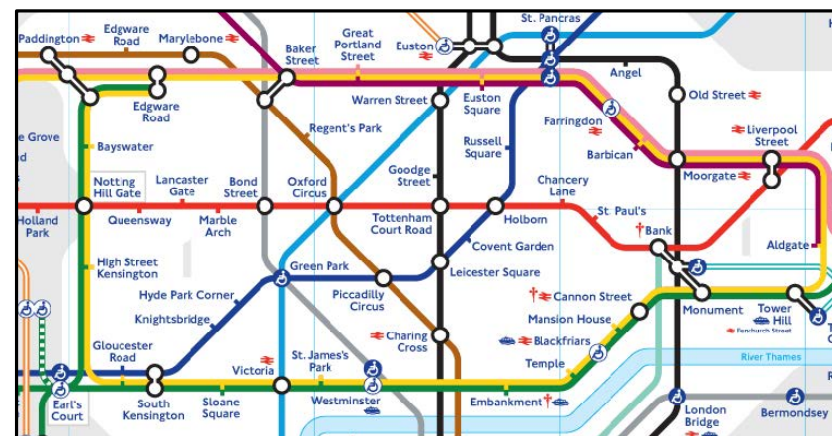
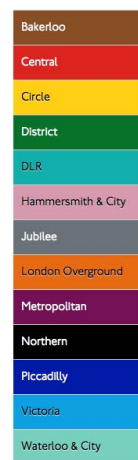


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen

Aufgaben beim Entwurf eines Netzplans



+



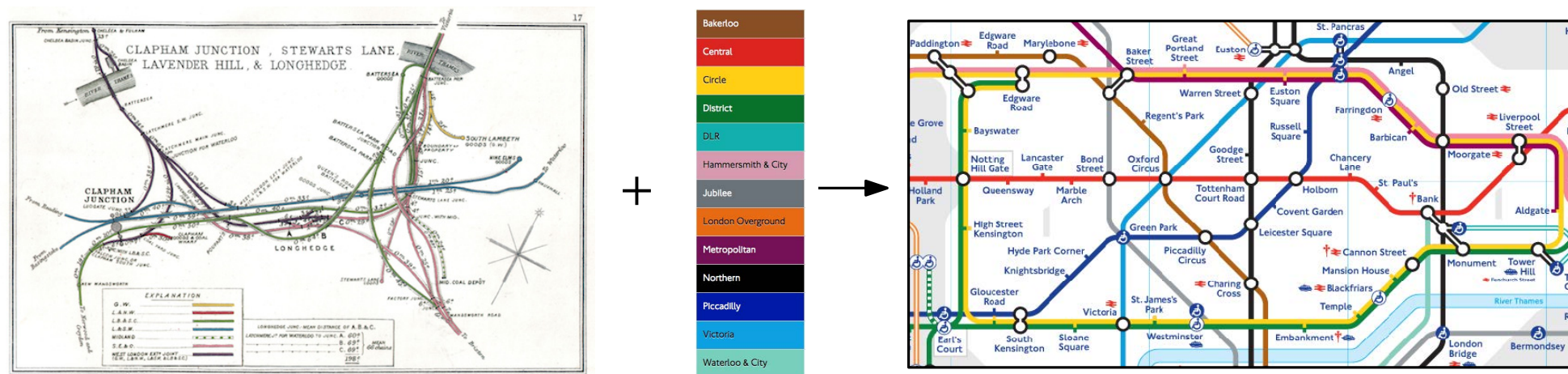
Gegeben.

- geografisch eingebettetes Bahnnetz G
- Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen

Gesucht.

- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

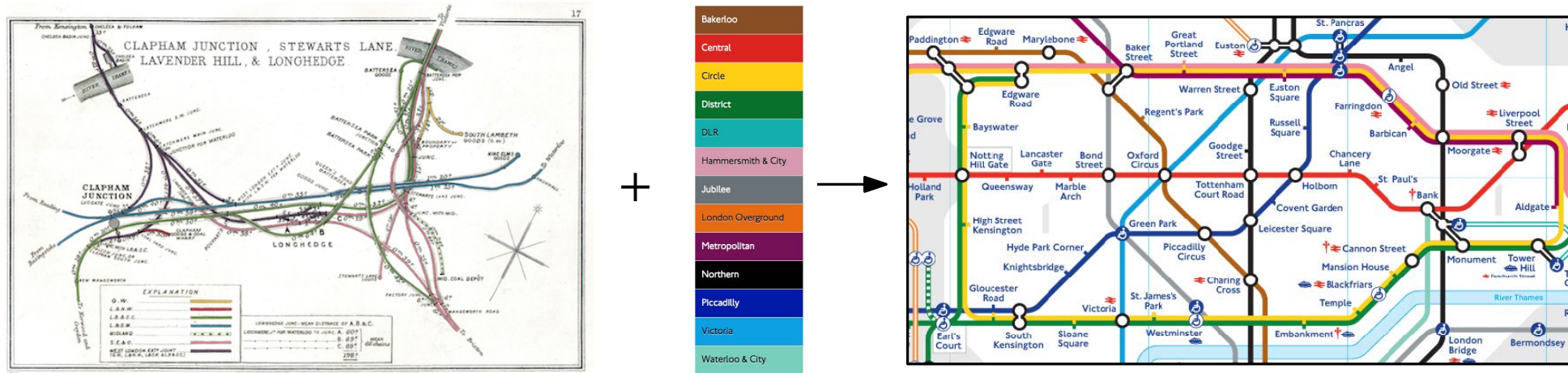
Aufgaben beim Entwurf eines Netzplans



- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.**
- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

Aufgaben beim Entwurf eines Netzplans

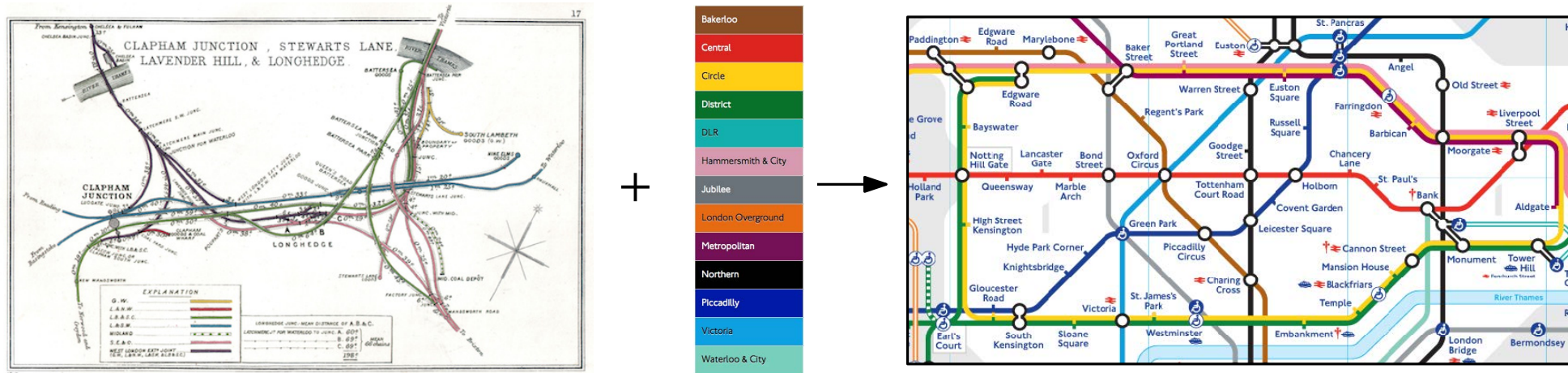


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.**
- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen

Aufgaben beim Entwurf eines Netzplans

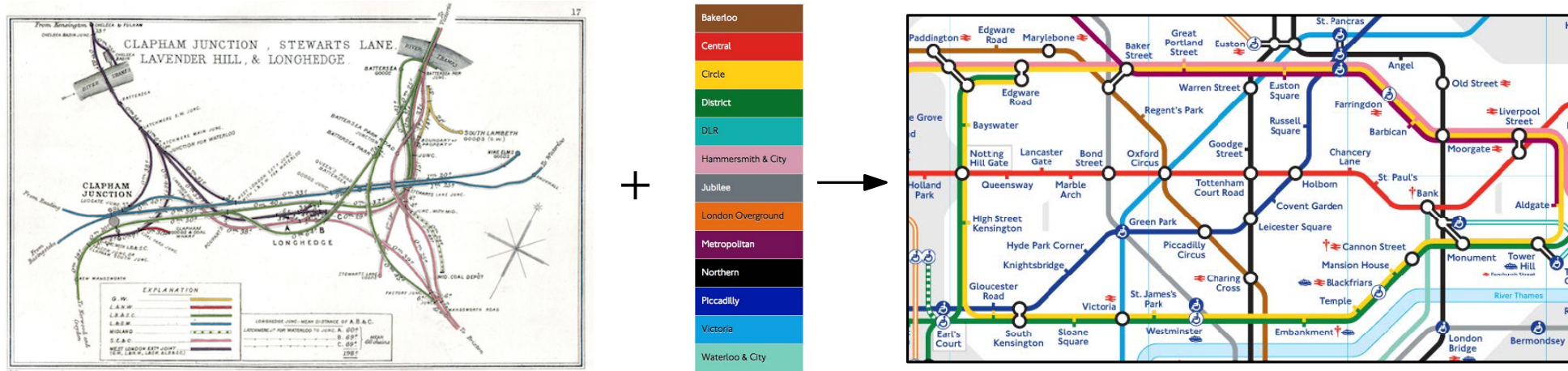


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.**
- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout

Aufgaben beim Entwurf eines Netzplans

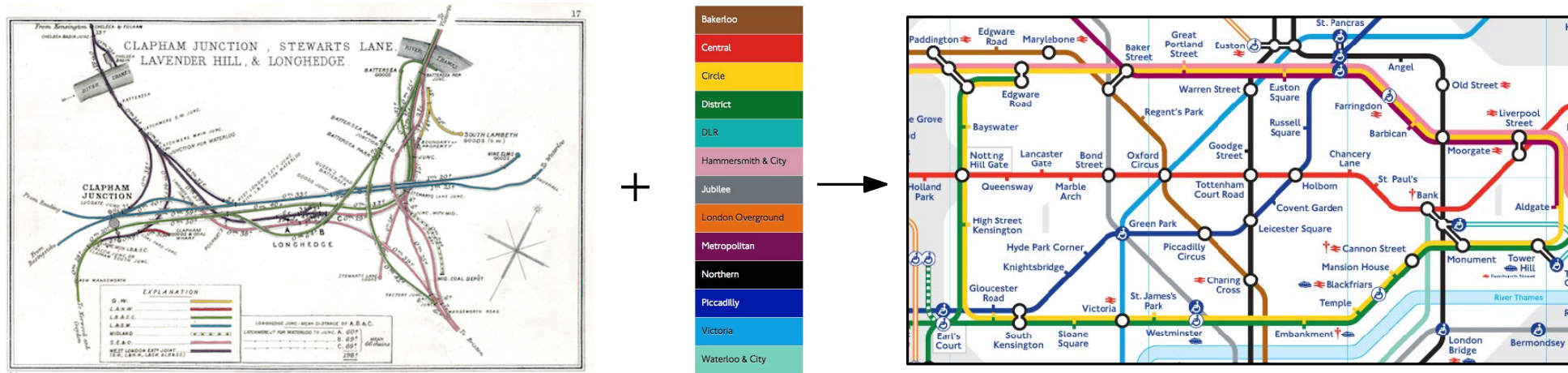


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.** ■ **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung

Aufgaben beim Entwurf eines Netzplans

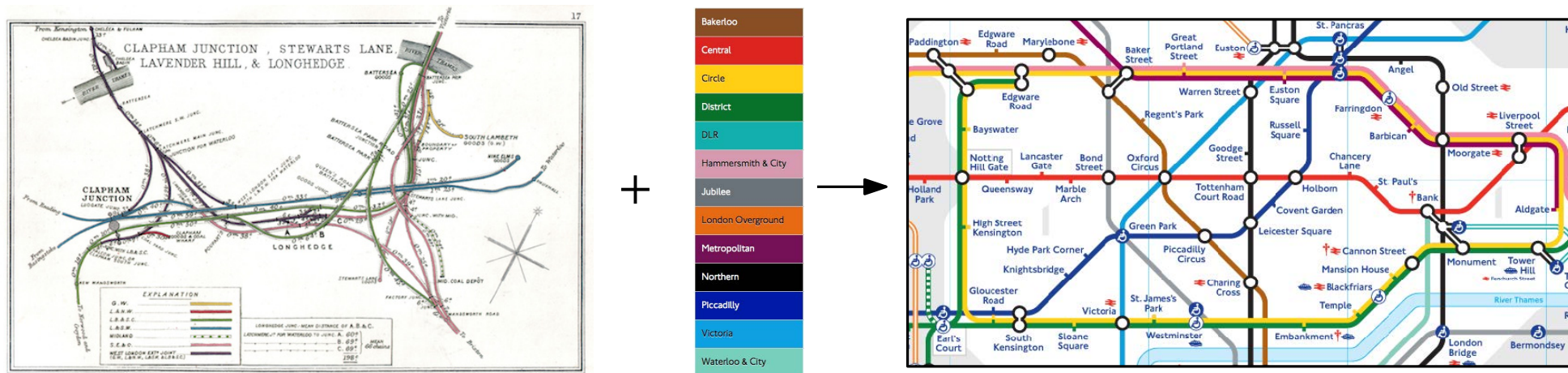


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.** ■ **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien

Aufgaben beim Entwurf eines Netzplans

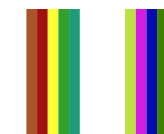


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.** ■ **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

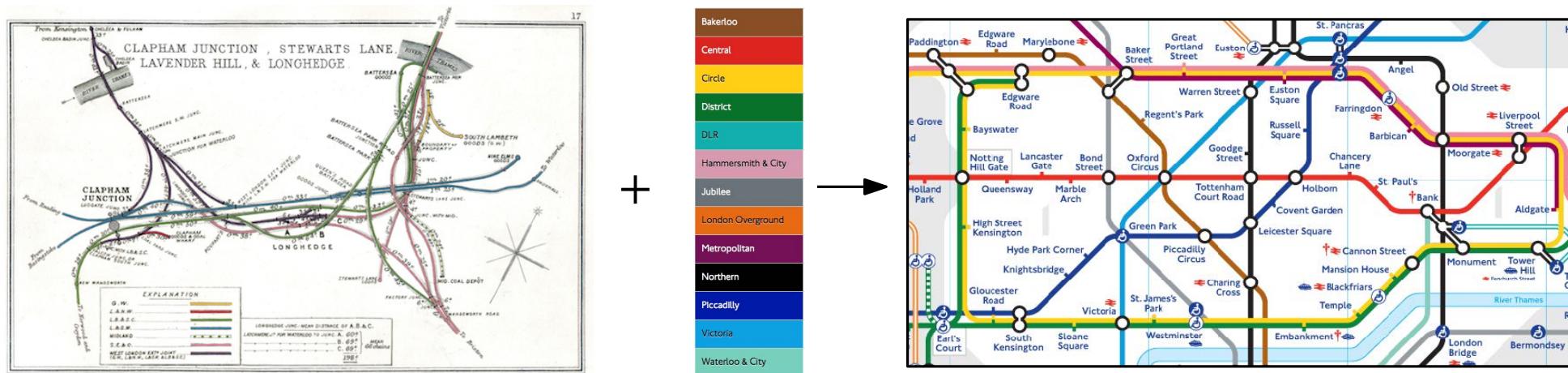
Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien

Farbe



Aufgaben beim Entwurf eines Netzplans

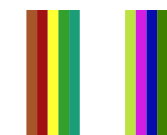


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.**
- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien

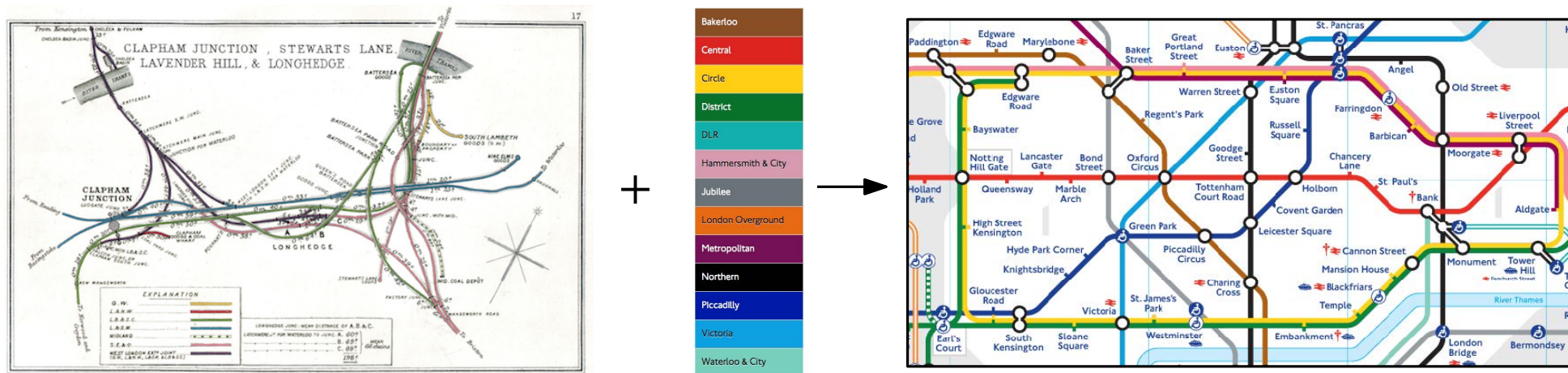
Farbe



Schriftart

Paddington
PADDINGTON
paddington

Aufgaben beim Entwurf eines Netzplans

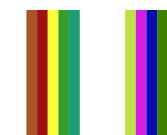


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.** ■ **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien

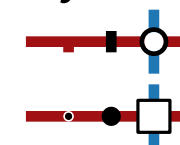
Farbe



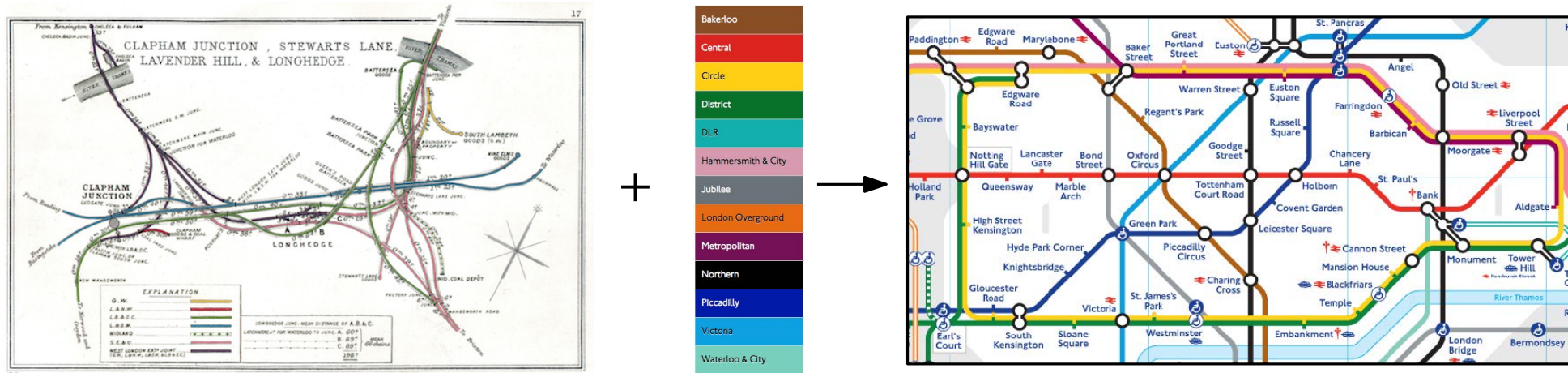
Schriftart

Paddington
PADDINGTON
Paddington

Symbol



Aufgaben beim Entwurf eines Netzplans

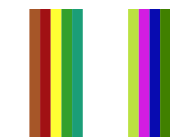


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.**
- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien

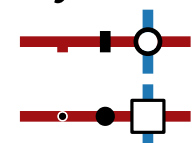
Farbe



Schriftart

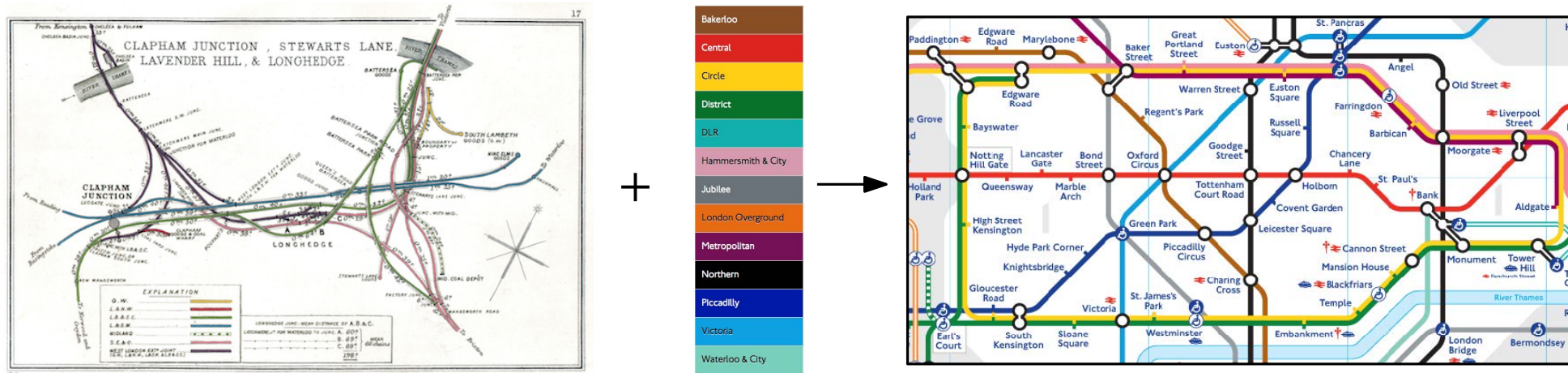
Paddington
PADDINGTON
Paddington

Symbol



➔ sehr wichtig, aber kein berechenbares Problem

Aufgaben beim Entwurf eines Netzplans

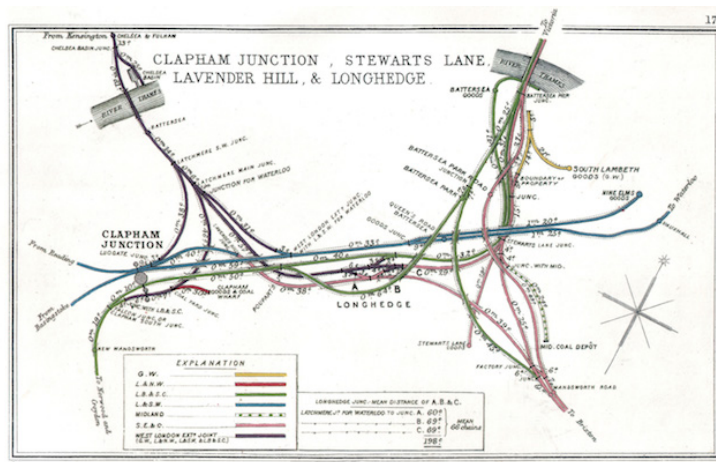


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.** ■ **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

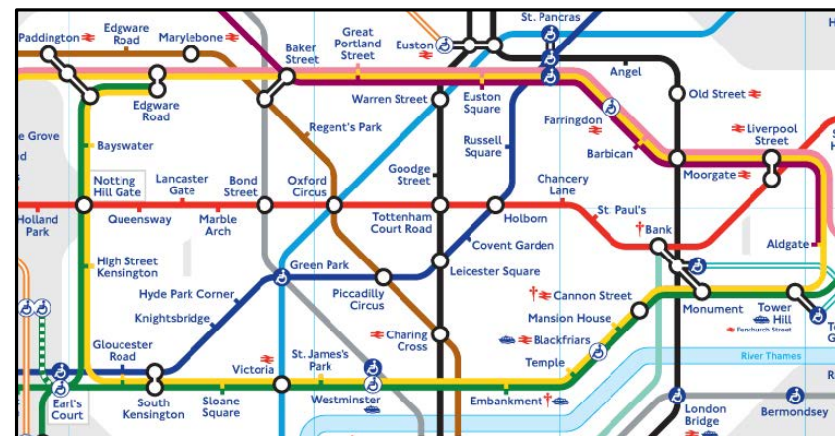
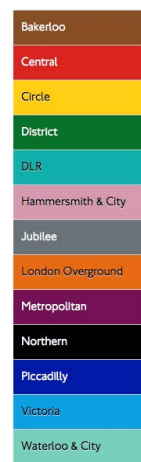
Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien

Aufgaben beim Entwurf eines Netzplans



+



Gegeben.

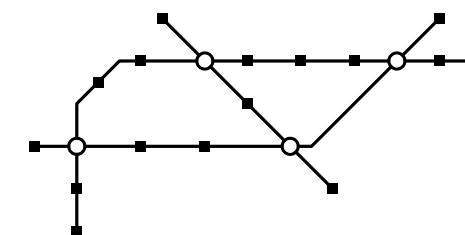
- geografisch eingebettetes Bahnnetz G
- Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen

Gesucht.

- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

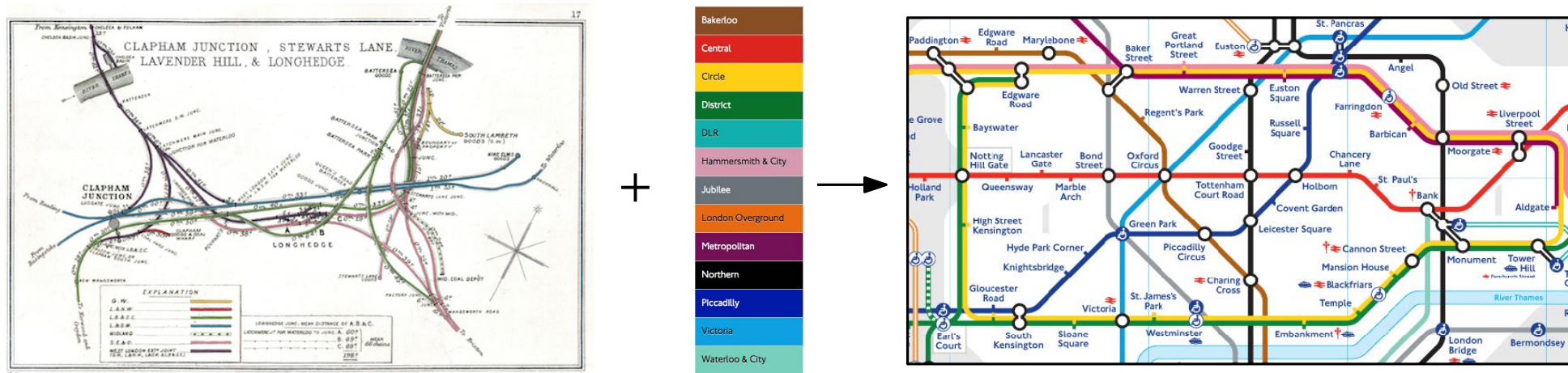
Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien



Bestimme Geometrie des Netzwerklayouts

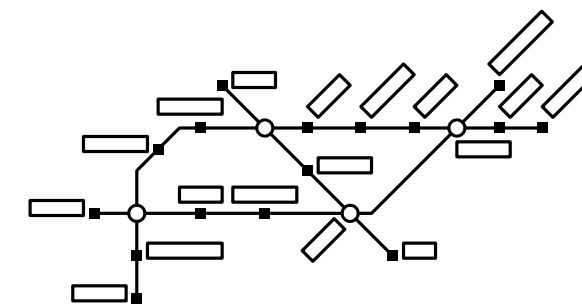
Aufgaben beim Entwurf eines Netzplans



- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.**
- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

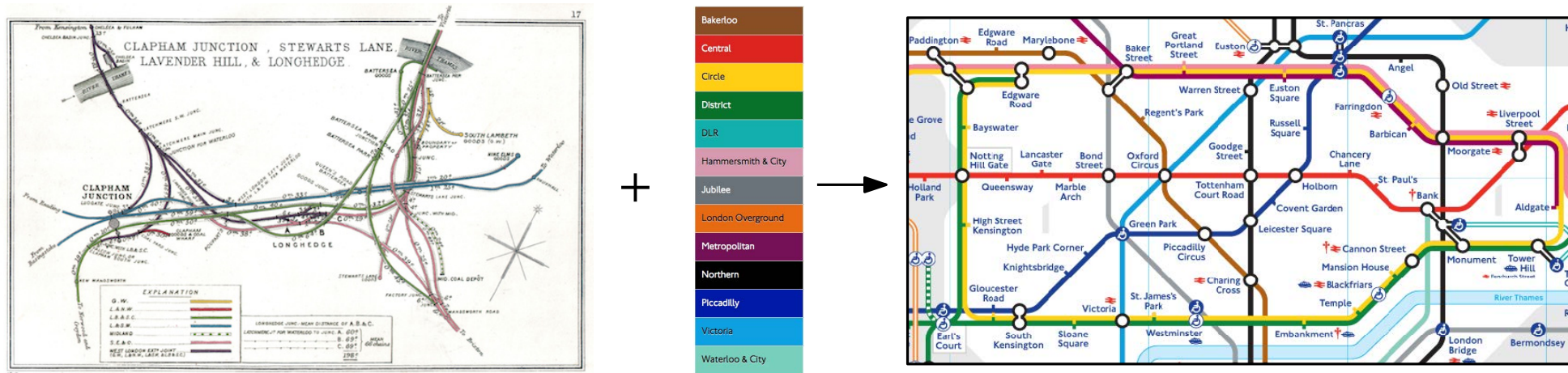
Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien



Bestimme Positionen der Stationenbeschriftungen

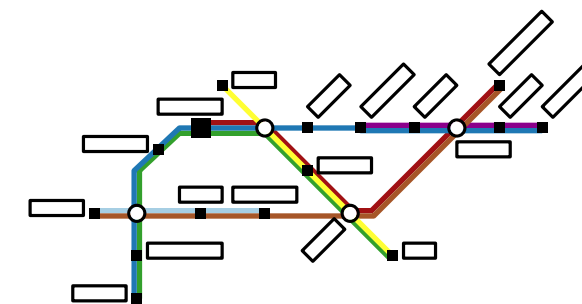
Aufgaben beim Entwurf eines Netzplans



- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.**
- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

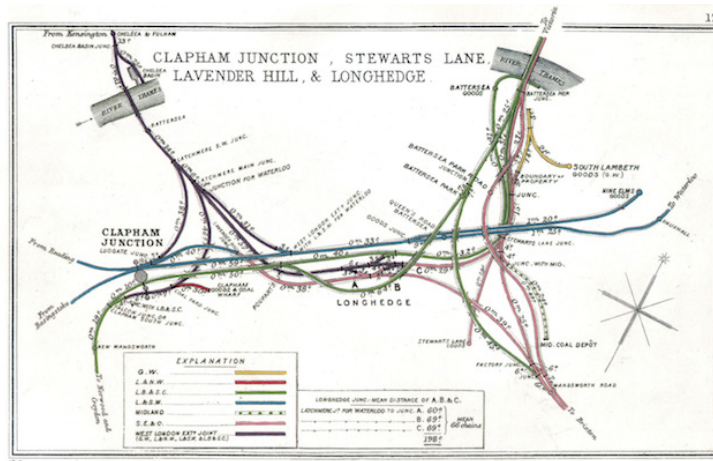
Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- Netzlayout
- Stationenbeschriftung
- Routenführung der U-Bahn-Linien

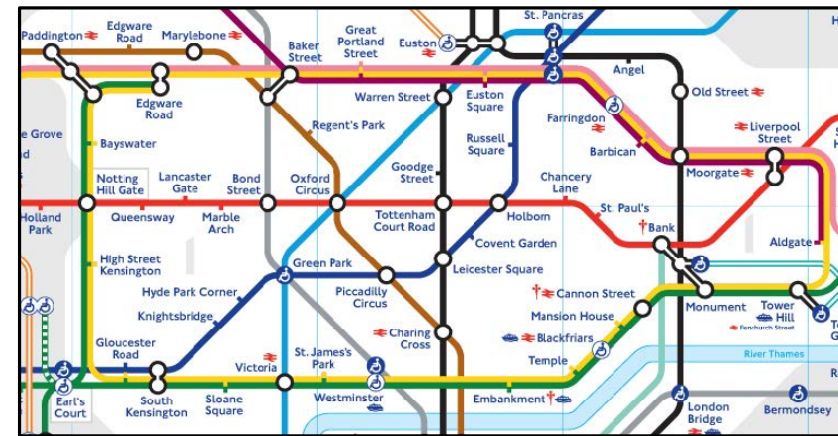
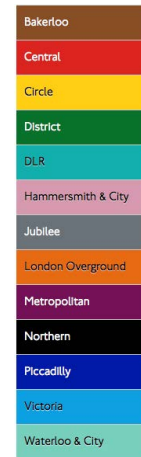


Bestimme Linienführung und Anordnung der Bündel

Aufgaben beim Entwurf eines Netzplans



+

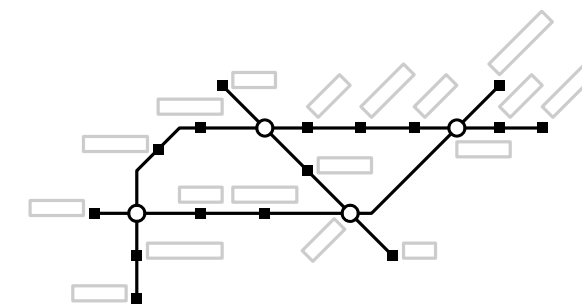


- Gegeben.**
- geografisch eingebettetes Bahnnetz G
 - Menge der U-Bahn-Linien $\mathcal{L}$, die G bedienen
- Gesucht.**
- **optimales** U-Bahn-Netzplan-Layout (was auch immer das bedeutet)

Teile die Aufgabe in mehrere Teilaufgaben auf.

- Darstellung und gestalterische Entscheidungen
- **Netzlayout**
- **Stationenbeschriftung**
- Routenführung der U-Bahn-Linien

Fokus heute



Formalisierung

Gegeben.



Formalisierung



Gegeben. ■ Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$

Formalisierung



- Gegeben.**
- Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$
 - Knotenmenge V (Stationen)

Formalisierung



- Gegeben.**
- Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$
 - Knotenmenge V (Stationen)
 - Kantenmenge E (Bahnverbindungen)

Formalisierung



- Gegeben.**
- Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$
 - Knotenmenge V (Stationen)
 - Kantenmenge E (Bahnverbindungen)
 - Menge von Pfaden $\mathcal{L}$ (U-Bahn-Linien in G)



Formalisierung

- Gegeben.**
- Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$
 - Knotenmenge V (Stationen)
 - Kantenmenge E (Bahnverbindungen)
 - Menge von Pfaden $\mathcal{L}$ (U-Bahn-Linien in G)
- Gesucht.** **Schematisches** Layout von $(G, \mathcal{L})$, das



Formalisierung

- Gegeben.**
- Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$
 - Knotenmenge V (Stationen)
 - Kantenmenge E (Bahnverbindungen)
 - Menge von Pfaden $\mathcal{L}$ (U-Bahn-Linien in G)
- Gesucht.** **Schematisches** Layout von $(G, \mathcal{L})$, das
- eine Layout-Beschränkungen erfüllt



Formalisierung

- Gegeben.**
- Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$
 - Knotenmenge V (Stationen)
 - Kantenmenge E (Bahnverbindungen)
 - Menge von Pfaden $\mathcal{L}$ (U-Bahn-Linien in G)
- Gesucht.** **Schematisches** Layout von $(G, \mathcal{L})$, das
- eine Layout-Beschränkungen erfüllt
 - eine Qualitätskriterien optimiert



Formalisierung

Gegeben. ■ Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$

■ Knotenmenge V (Stationen)

■ Kantenmenge E (Bahnverbindungen)

■ Menge von Pfaden $\mathcal{L}$ (U-Bahn-Linien in G)

Gesucht. **Schematisches** Layout von $(G, \mathcal{L})$, das

■ eine Layout-Beschränkungen erfüllt

■ eine Qualitätskriterien optimiert

Aber was sind die Beschränkungen und Kriterien?

Formalisierung

Gegeben. ■ Graph $G = (V, E)$ geografisch eingebettet in $\mathbb{R}^2$

■ Knotenmenge V (Stationen)

■ Kantenmenge E (Bahnverbindungen)

■ Menge von Pfaden $\mathcal{L}$ (U-Bahn-Linien in G)

Gesucht. **Schematisches** Layout von $(G, \mathcal{L})$, das

■ eine Layout-Beschränkungen erfüllt

■ eine Qualitätskriterien optimiert

Aber was sind die Beschränkungen und Kriterien?

➔ Extrahiere gemeinsame Prinzipien existierender manuell gestalteter U-Bahn-Netzpläne



Designregeln

(R1) Ändere nicht die Topologie



Designregeln



(R1) Ändere nicht die Topologie

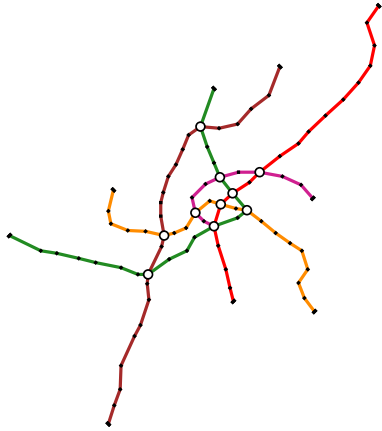
- keine neuen Kreuzungen
- keine Änderungen in kombinatorischer Einbettung

Designregeln



(R1) Ändere nicht die Topologie

- keine neuen Kreuzungen
- keine Änderungen in kombinatorischer Einbettung

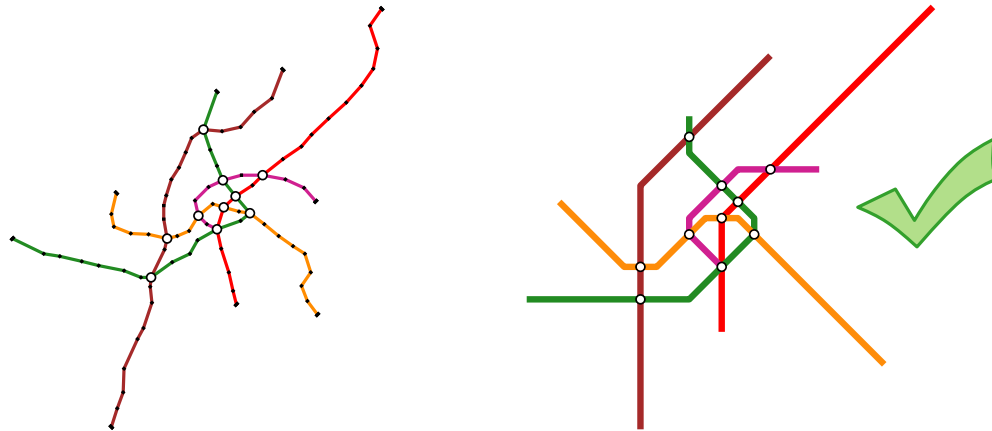


Designregeln



(R1) Ändere nicht die Topologie

- keine neuen Kreuzungen
- keine Änderungen in kombinatorischer Einbettung

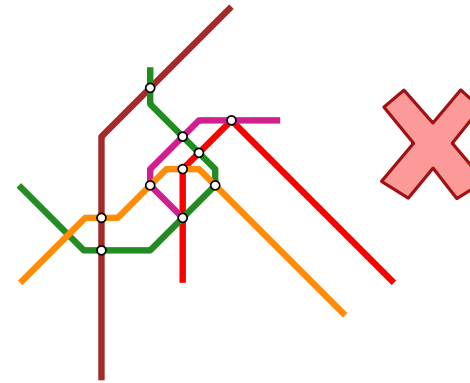
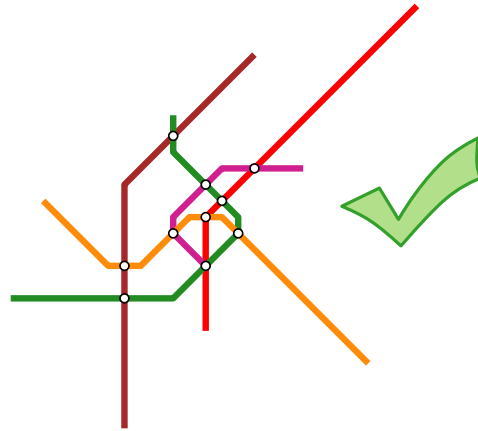
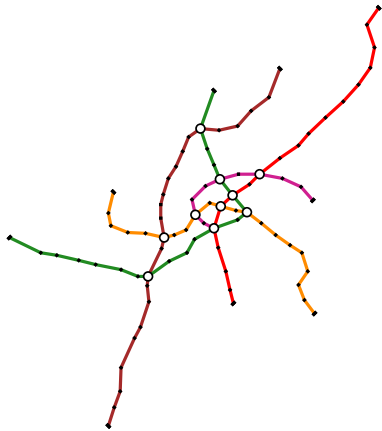


Designregeln



(R1) Ändere nicht die Topologie

- keine neuen Kreuzungen
- keine Änderungen in kombinatorischer Einbettung



Designregeln

(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen



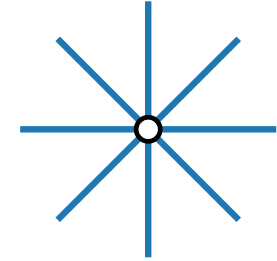
Designregeln



(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen

- meistens oktilinear
- oft abgerundet, manchmal mit Kurven



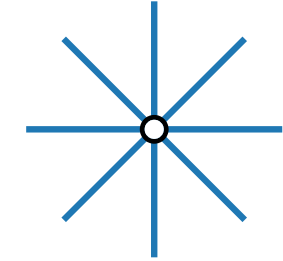
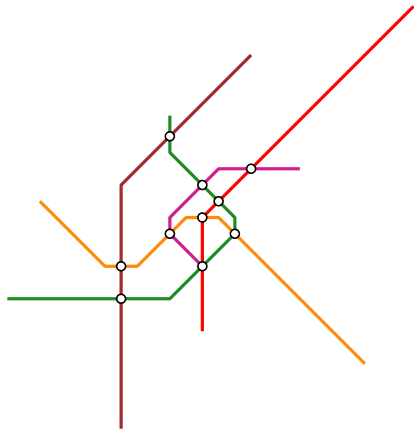
Designregeln



(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen

- meistens oktilinear
- oft abgerundet, manchmal mit Kurven



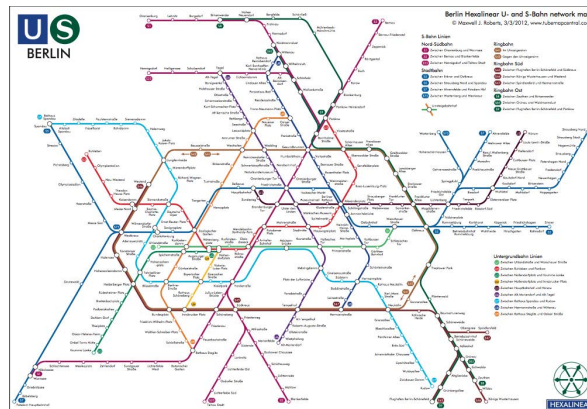
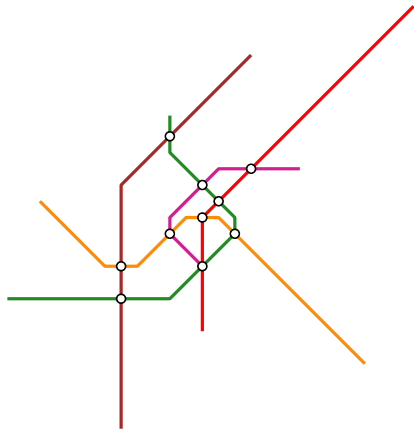
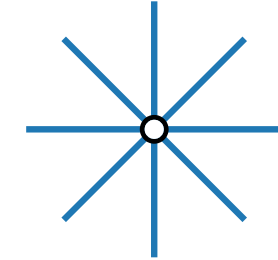
Designregeln



(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen

- meistens oktilinear
- oft abgerundet, manchmal mit Kurven



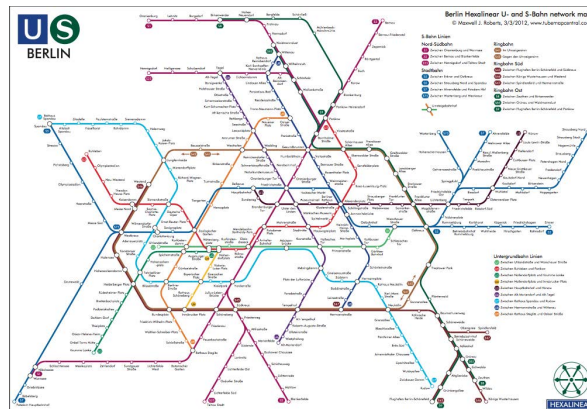
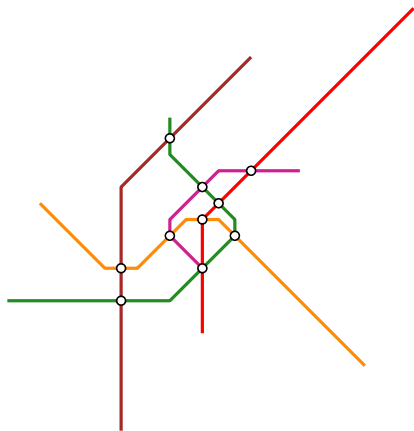
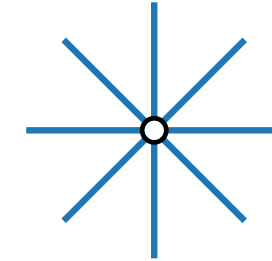
Designregeln



(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen

- meistens oktilinear
- oft abgerundet, manchmal mit Kurven



Designregeln



- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich



Designregeln

(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen

(R3) So einfach und monoton wie möglich

- vermeide Knicke
- bevorzuge große Knickwinkel
- für Kurven: bevorzuge uniforme Krümmung, wenige Wendepunkte



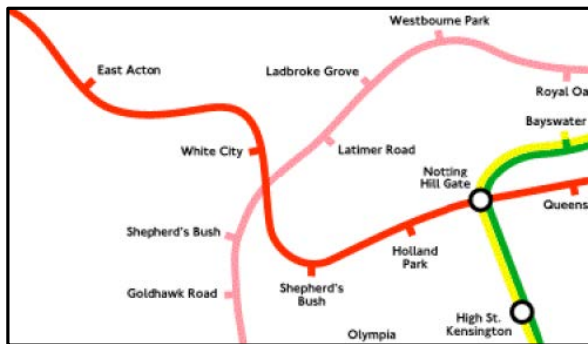
Designregeln

(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen

(R3) So einfach und monoton wie möglich

- vermeide Knicke
- bevorzuge große Knickwinkel
- für Kurven: bevorzuge uniforme Krümmung, wenige Wendepunkte





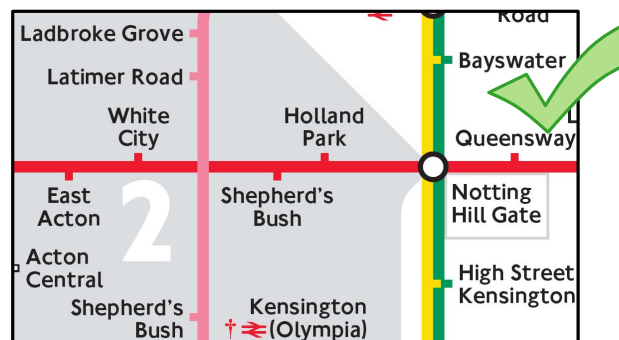
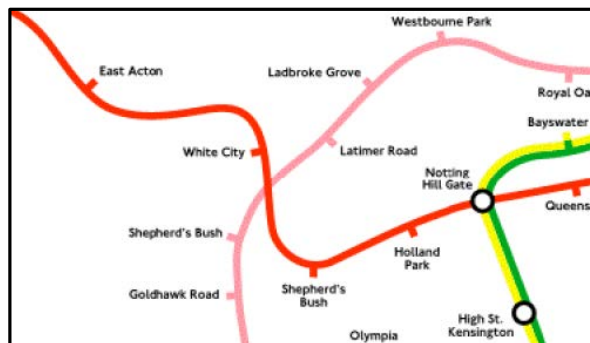
Designregeln

(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen

(R3) So einfach und monoton wie möglich

- vermeide Knicke
- bevorzuge große Knickwinkel
- für Kurven: bevorzuge uniforme Krümmung, wenige Wendepunkte



Designregeln

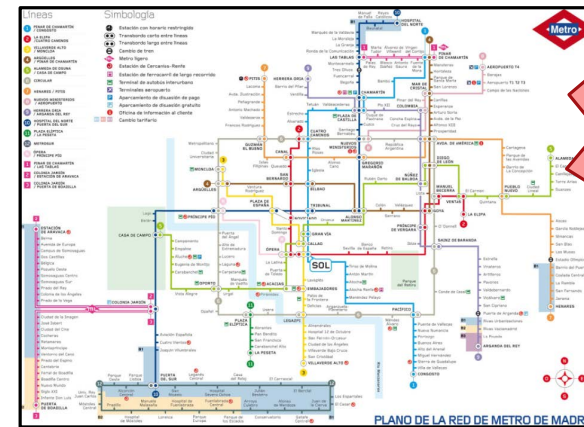
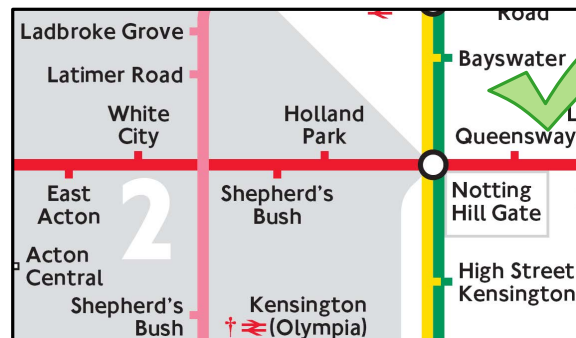
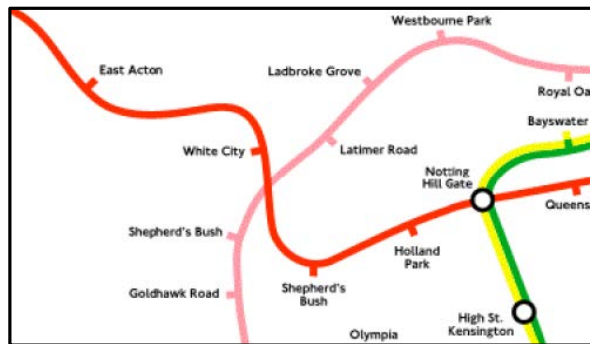


(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen

(R3) So einfach und monoton wie möglich

- vermeide Knicke
- bevorzuge große Knickwinkel
- für Kurven: bevorzuge uniforme Krümmung, wenige Wendepunkte



Designregeln



- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen

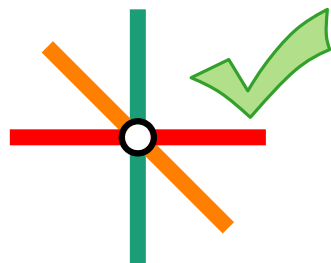


Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
 - vermeidet visuelle Unklarheiten in komplexen Stationen

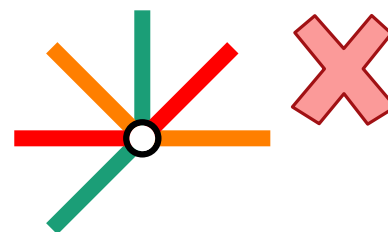
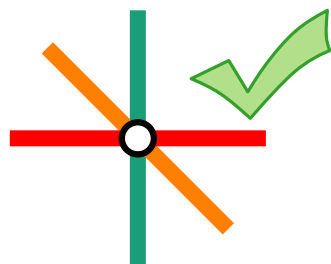
Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
 - vermeidet visuelle Unklarheiten in komplexen Stationen



Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
 - vermeidet visuelle Unklarheiten in komplexen Stationen



Designregeln



- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen

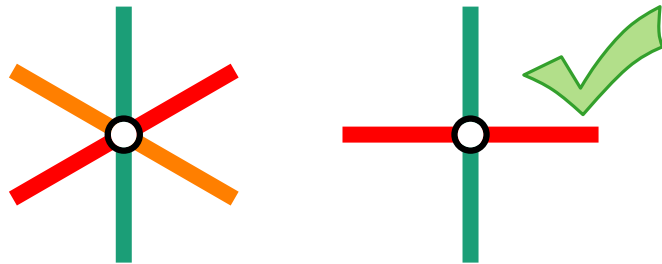


Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
 - verteilt Kanten gleichmäßig für balanciertes Aussehen

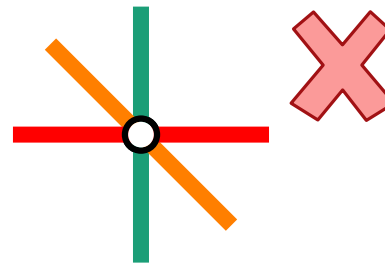
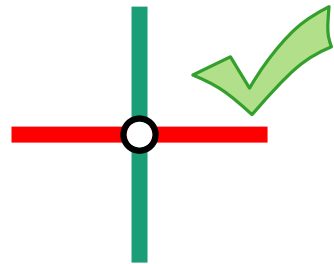
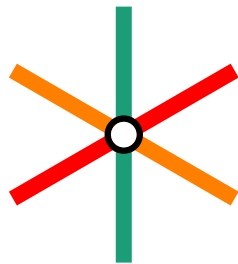
Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
 - verteilt Kanten gleichmäßig für balanciertes Aussehen



Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
 - verteilt Kanten gleichmäßig für balanciertes Aussehen





Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung



Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung
 - bewahrt die Ähnlichkeit zur Geographie
 - erhält die mentale Karte des Nutzers
 - lokal oder global anwendbar

Designregeln



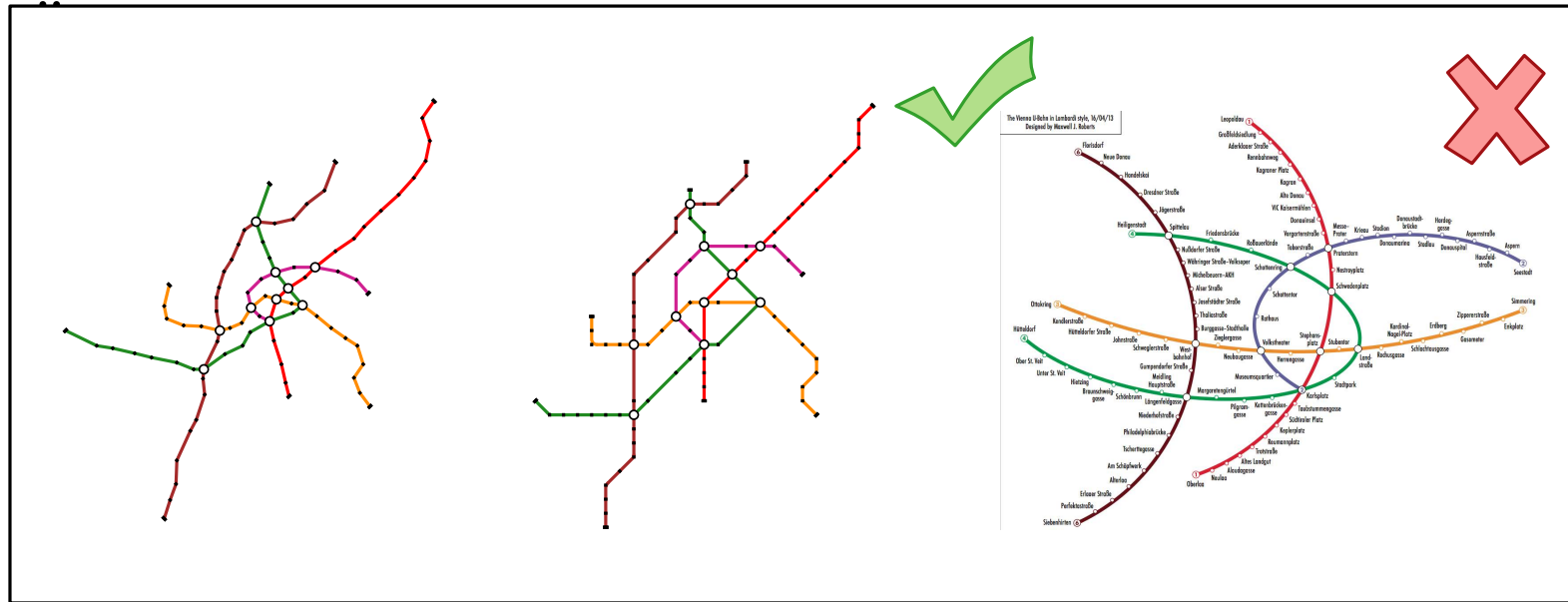
(R1)

(R2)

(R3)

(R4)

(R5)



(R6) Minimiere geometrische Verzerrung und Verschiebung

- bewahrt die Ähnlichkeit zur Geographie
- erhält die mentale Karte des Nutzers
- lokal oder global anwendbar



Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung
- (R7) Verwende gleichmäßige Kantenlängen

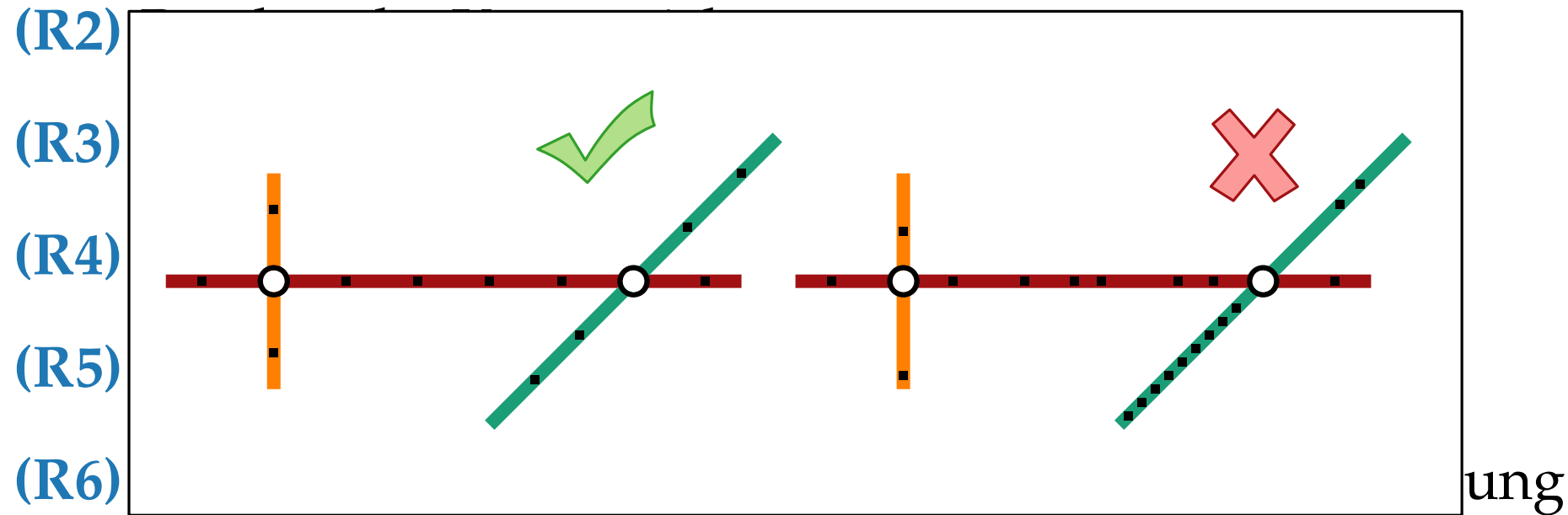


Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung
- (R7) Verwende gleichmäßige Kantenlängen
 - geografische Entfernungen sind weniger wichtig
 - Umsteigeentfernungen sind wichtiger
 - ausgewogenes Erscheinungsbild

Designregeln

(R1) Ändere nicht die Topologie



(R7) Verwende gleichmäßige Kantenlängen

- geografische Entfernungen sind weniger wichtig
- Umsteigeentfernungen sind wichtiger
- ausgewogenes Erscheinungsbild



Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung
- (R7) Verwende gleichmäßige Kantenlängen
- (R8) Trenne nicht zusammenhängende Merkmale



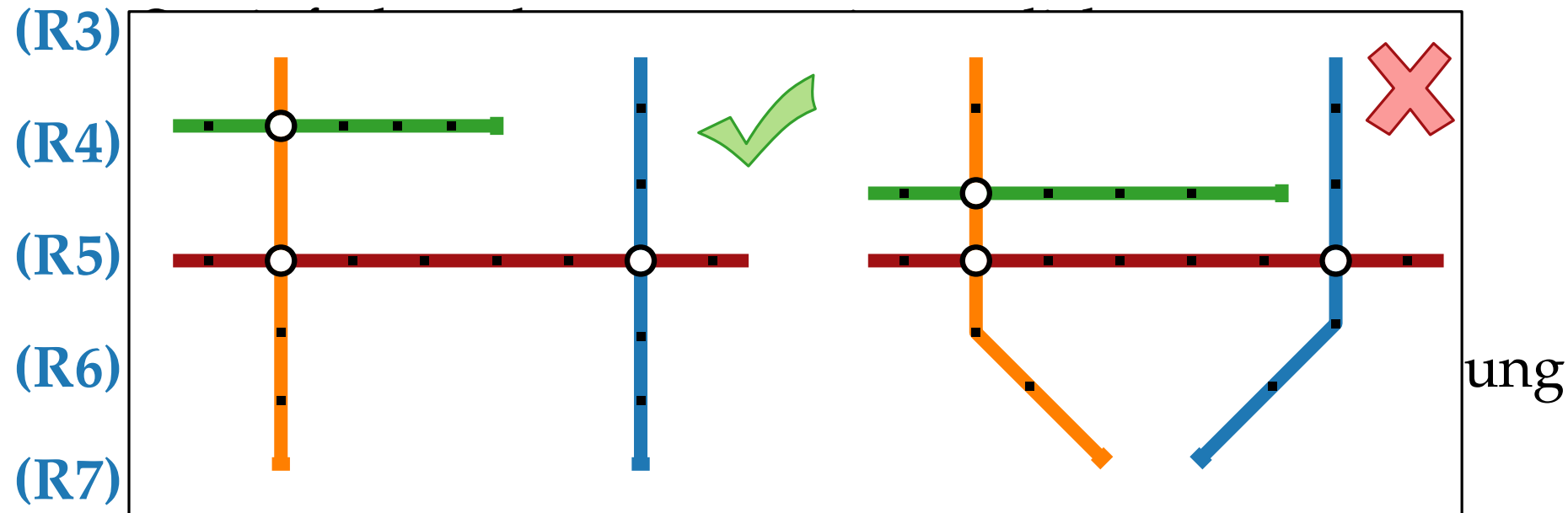
Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung
- (R7) Verwende gleichmäßige Kantenlängen
- (R8) Trenne nicht zusammenhängende Merkmale
 - garantiert minimalen Abstand zwischen Merkmalen
 - vermeidet Mehrdeutigkeiten

Designregeln

(R1) Ändere nicht die Topologie

(R2) Beschränke Kantenrichtungen



(R8) Trenne nicht zusammenhängende Merkmale

- garantiert minimalen Abstand zwischen Merkmalen
- vermeidet Mehrdeutigkeiten



Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung
- (R7) Verwende gleichmäßige Kantenlängen
- (R8) Trenne nicht zusammenhängende Merkmale
- (R9) Vermeide große leere Bereiche



Designregeln

- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung
- (R7) Verwende gleichmäßige Kantenlängen
- (R8) Trenne nicht zusammenhängende Merkmale
- (R9) Vermeide große leere Bereiche
 - balanciert lokale Dichte
 - fülle möglicherweise Lücken mit Legenden

(R1) Ändere nicht die Topologie

(R3)

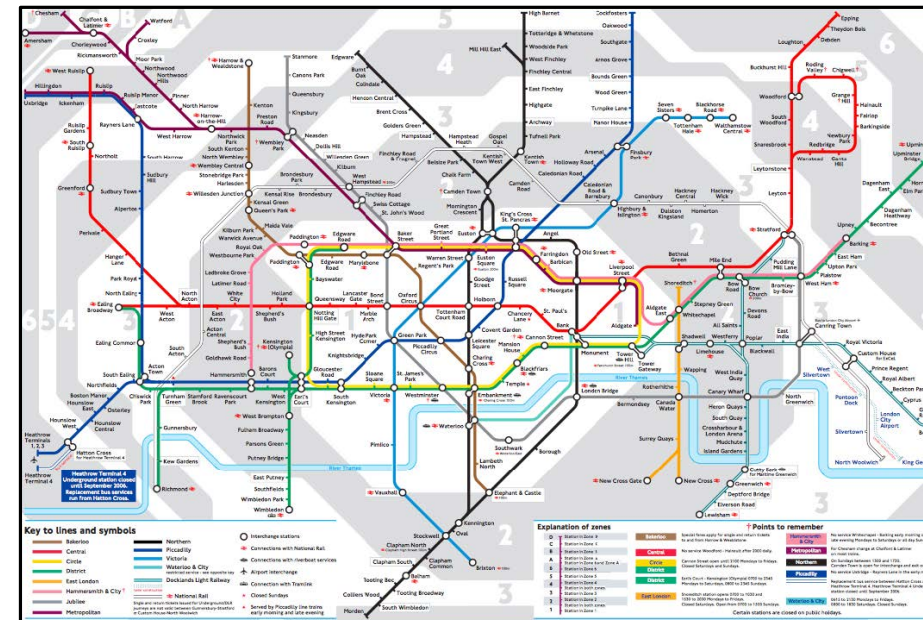
(R4)

(R5)

(R6)

(R7)

(R8)



(R9) Vermeide große leere Bereiche

- balanciert lokale Dichte
- fülle möglicherweise Lücken mit Legenden



Designregeln

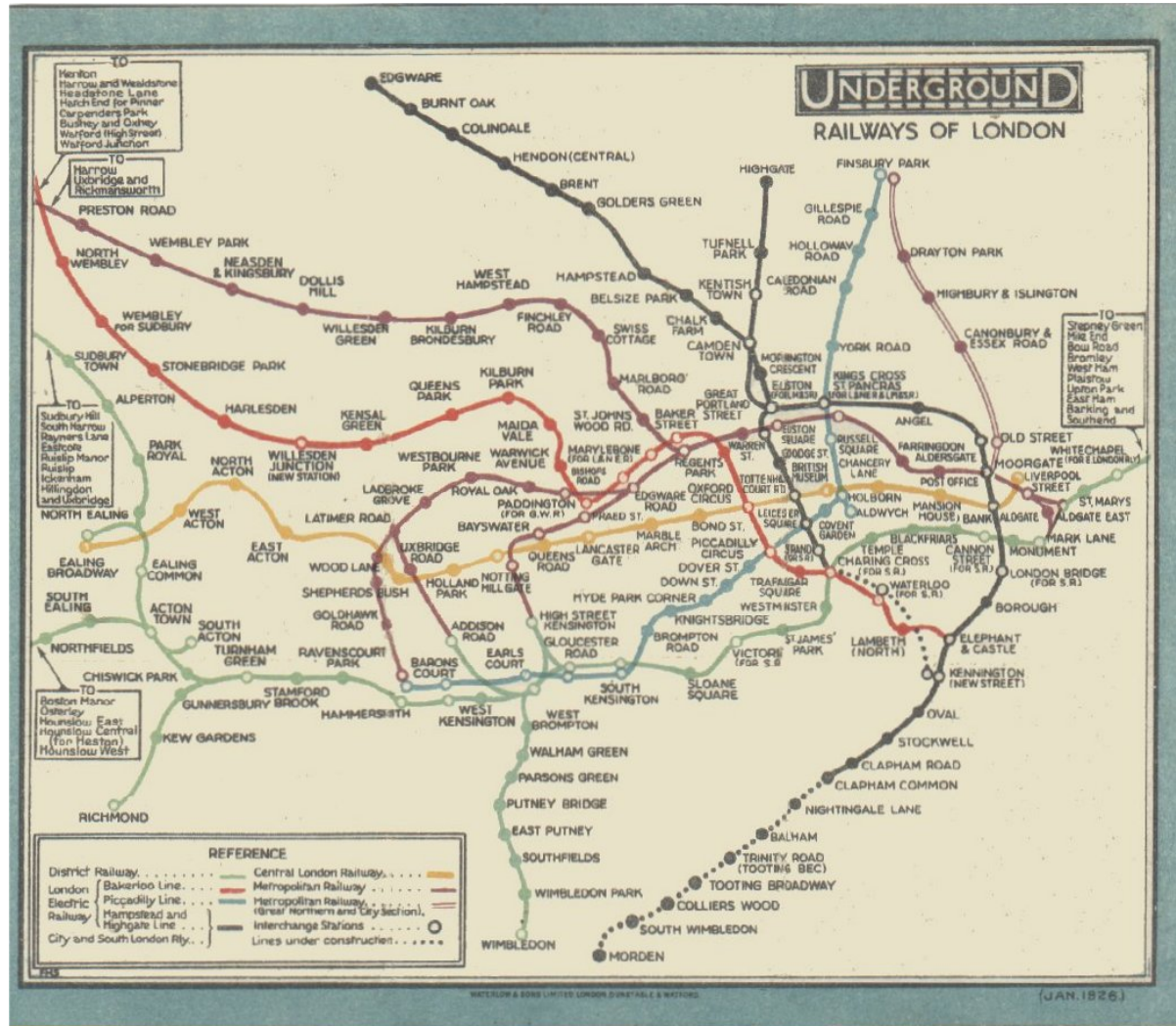
- (R1) Ändere nicht die Topologie
- (R2) Beschränke Kantenrichtungen
- (R3) So einfach und monoton wie möglich
- (R4) Linien gehen gerade durch Kreuzungen
- (R5) Benutze große Winkelauflösung an Stationen
- (R6) Minimiere geometrische Verzerrung und Verschiebung
- (R7) Verwende gleichmäßige Kantenlängen
- (R8) Trenne nicht zusammenhängende Merkmale
- (R9) Vermeide große leere Bereiche



Designregeln

- (R1) Ändere nicht die Topologie
 - (R2) Beschränke Kantenrichtungen
 - (R3) So einfach und monoton wie möglich
 - (R4) Linien gehen gerade durch Kreuzungen
 - (R5) Benutze große Winkelauflösung an Stationen
 - (R6) Minimiere geometrische Verzerrung und Verschiebung
 - (R7) Verwende gleichmäßige Kantenlängen
 - (R8) Trenne nicht zusammenhängende Merkmale
 - (R9) Vermeide große leere Bereiche
- ➔ Regeln widersprechen sich möglicherweise und benötigen Prioritäten.

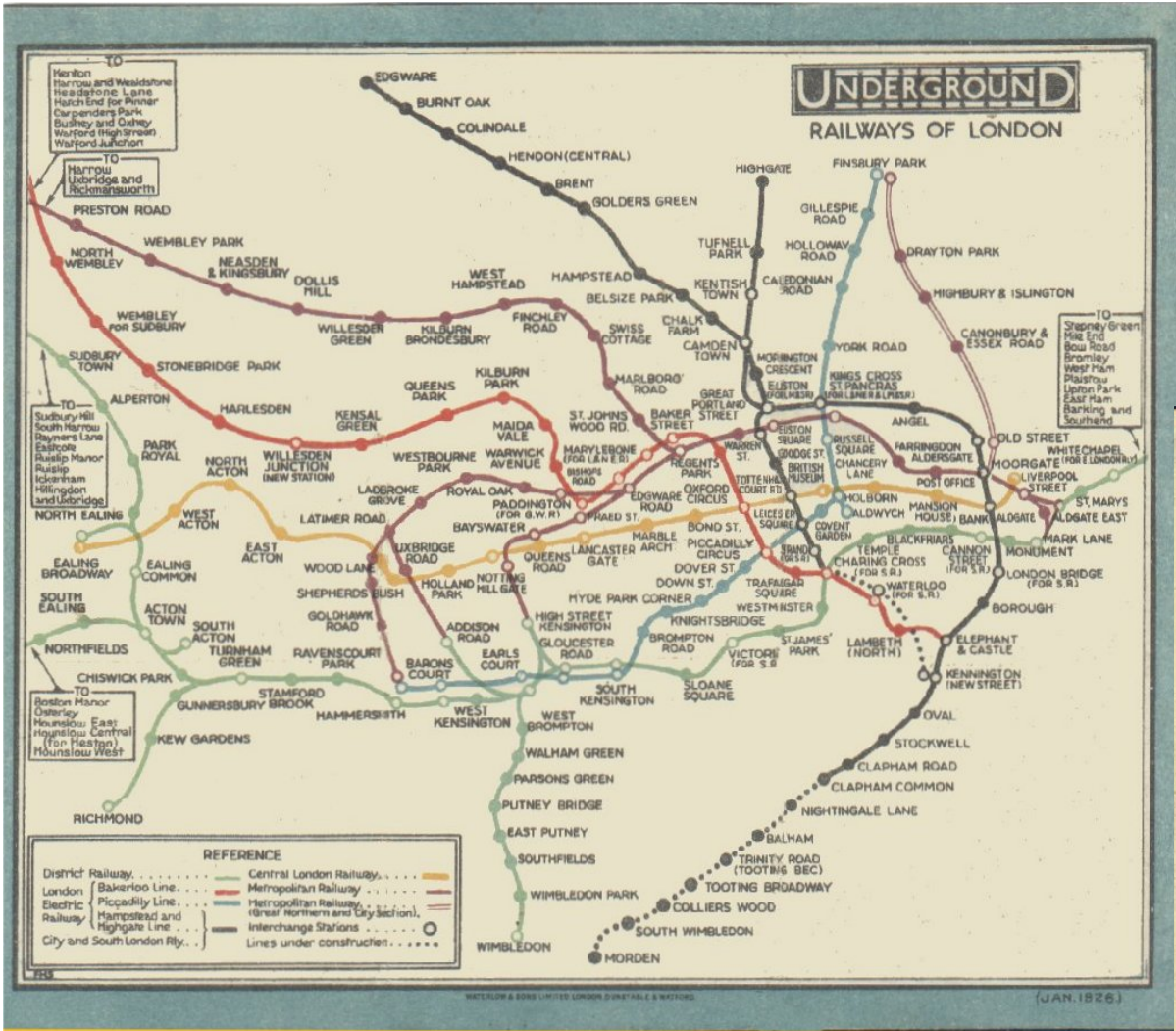
Ein bisschen Geschichte



London 1927 (Fred H. Stingemore)

(c) Mike Yashworth

Ein bisschen Geschichte



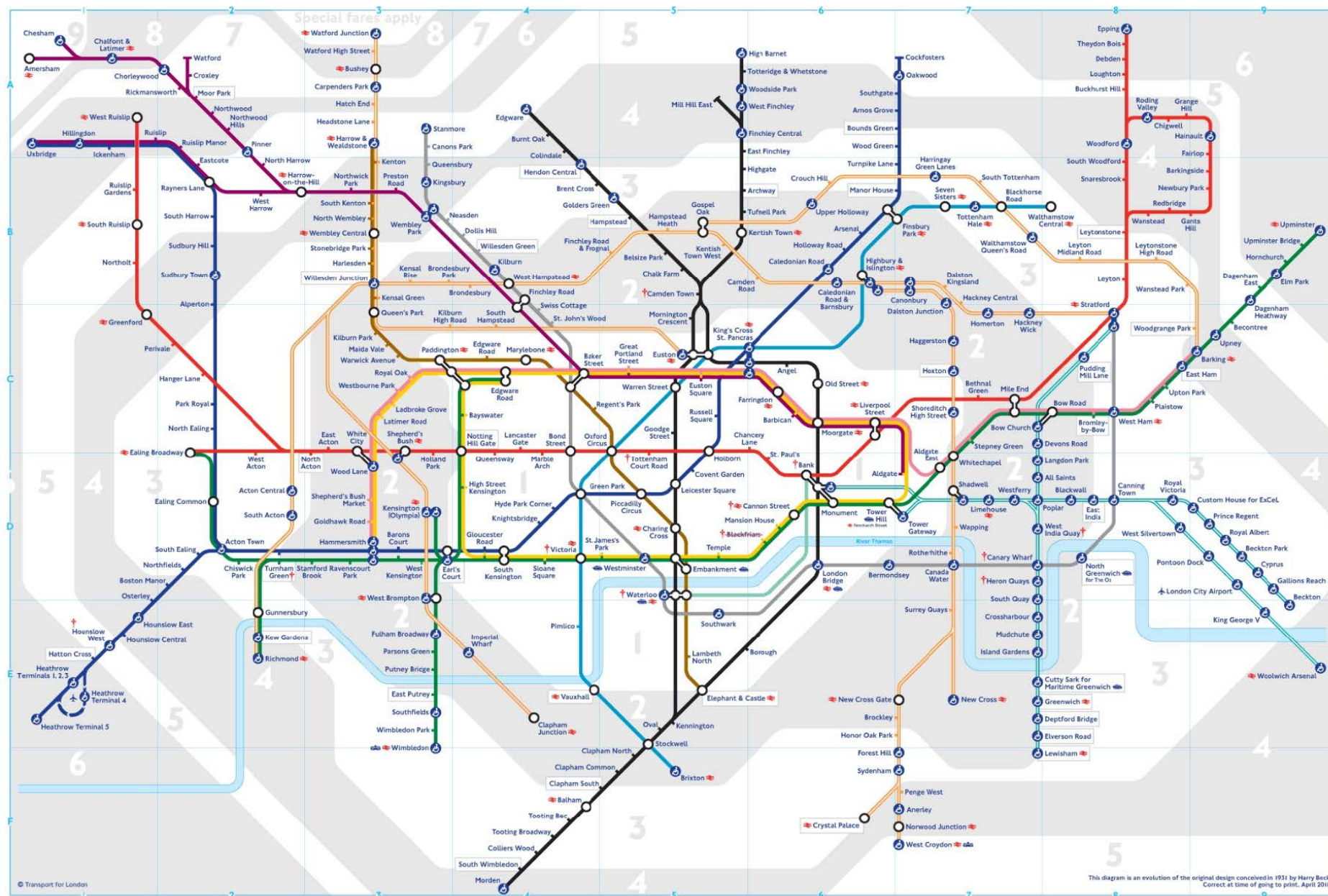
London 1927 (Fred H. Stingemore)

(c) Mike Yashworth

Henry Beck
London 1933



Ein bisschen Geschichte



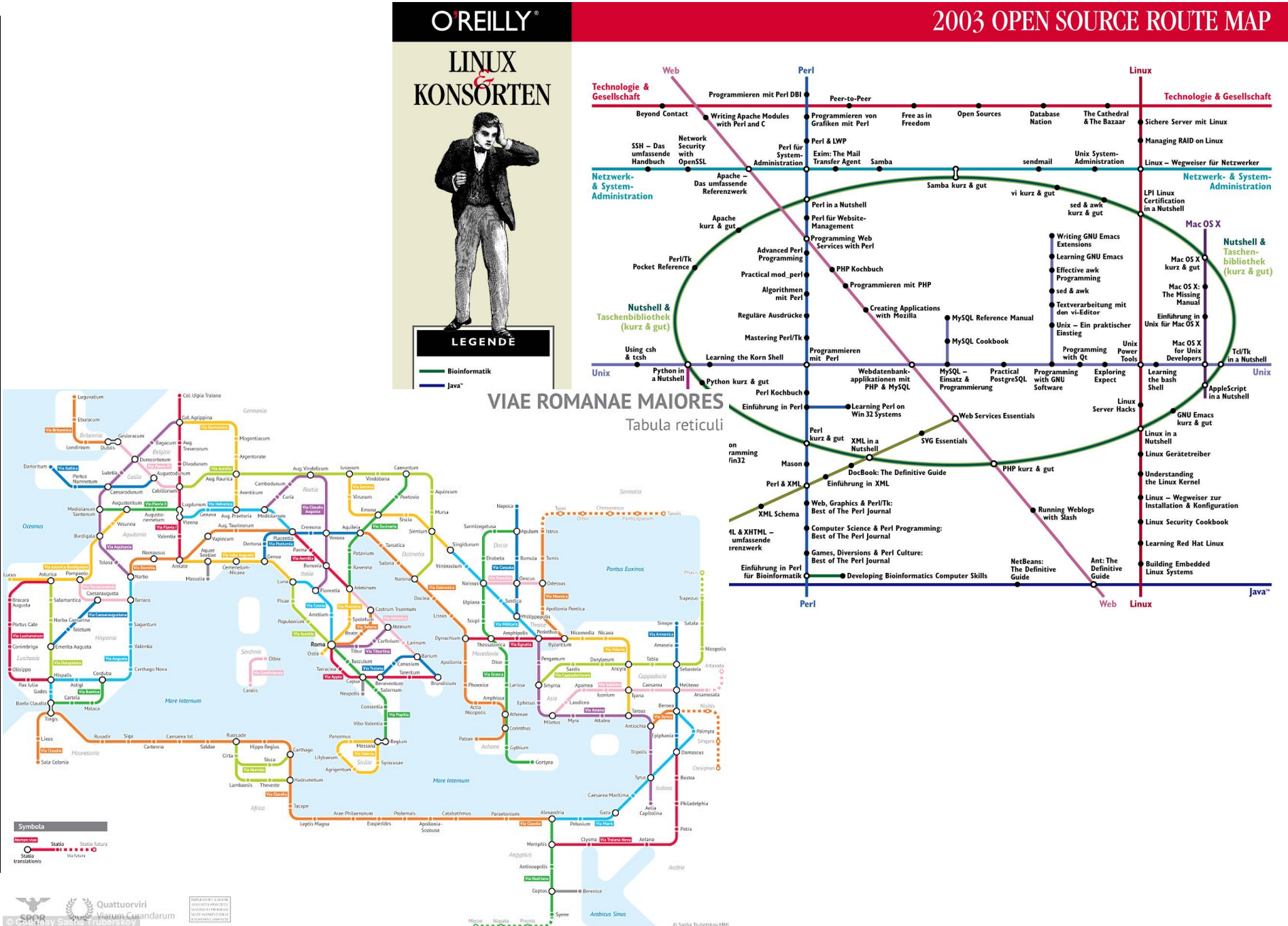
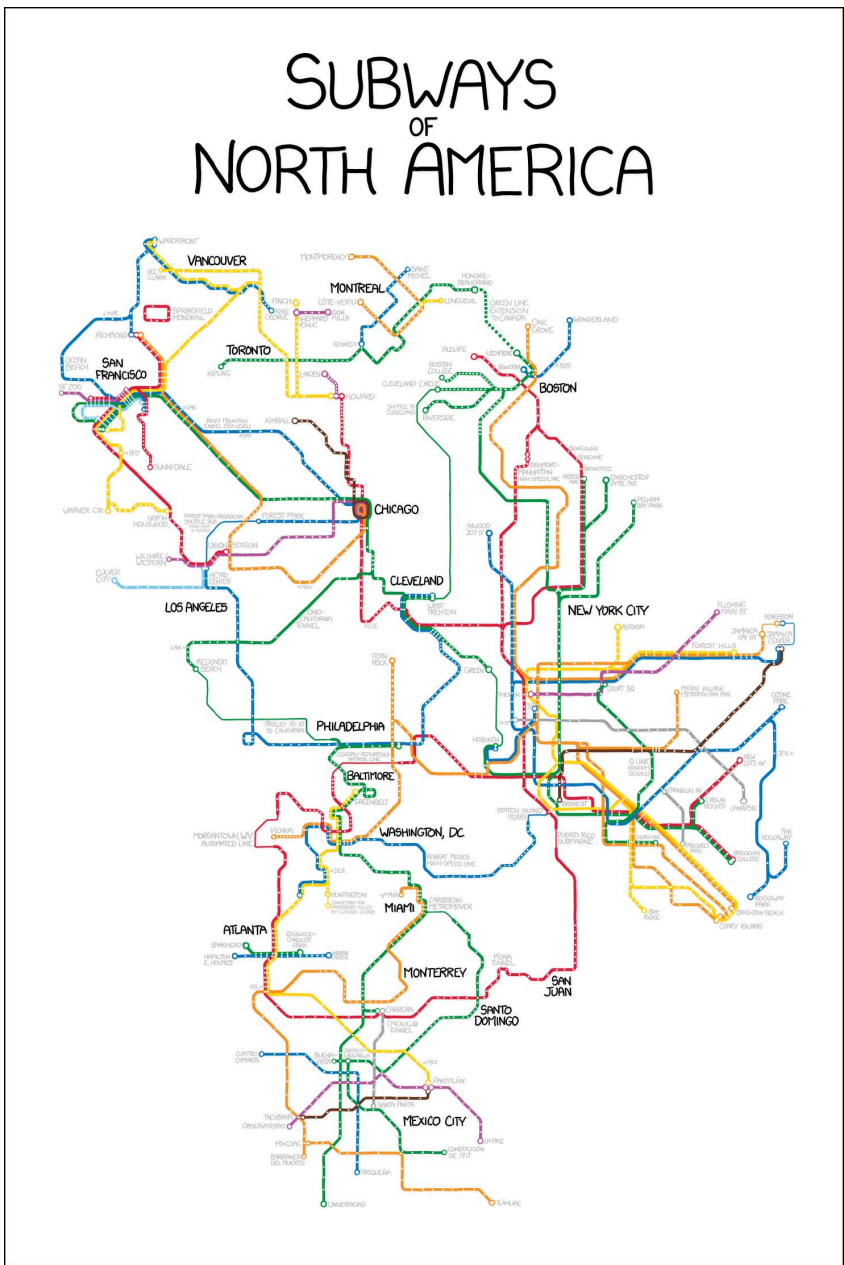
Tube Map 2006

Ein bisschen Geschichte



Berlin 1931 (by Maxwell Roberts)

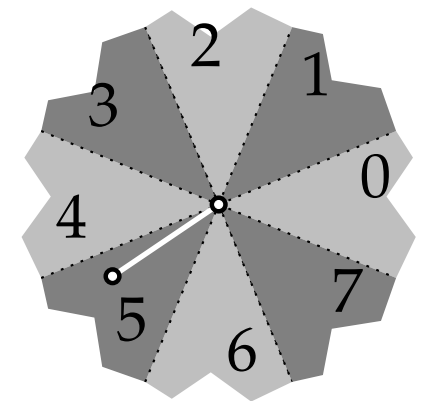
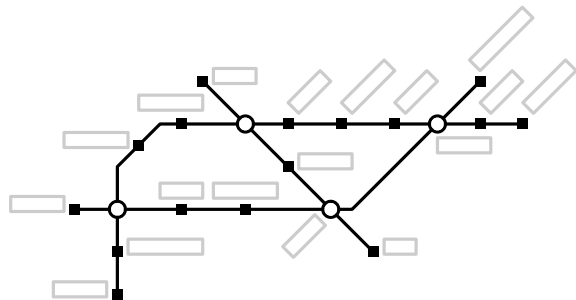
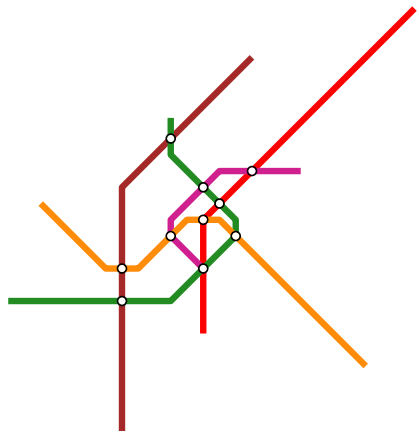
Ein bisschen Geschichte



Algorithmen für geographische Informationssysteme

11. Vorlesung Netzpläne (Metro Maps)

Teil II: Komplexität und pfadbasierte Schematisierung



Komplexität



Satz.

[Nöllenburg 2005]

Für einen eingebetteten Graphen G (Knotengrade ≤ 8) ist

Knickminimierung (R3) NP-schwer, wenn **Topologieerhaltung** (R1) und **Oktilinearität** (R2) gefordert sind.

Komplexität



Satz.

[Nöllenburg 2005]

Für einen eingebetteten Graphen G (Knotengrade ≤ 8) ist

Knickminimierung (R3) NP-schwer, wenn **Topologieerhaltung** (R1) und **Oktilinearität** (R2) gefordert sind.

Beweisidee.

Reduktion von PLANAR-3SAT mit starren mechanischen Gadgets

Komplexität



Satz.

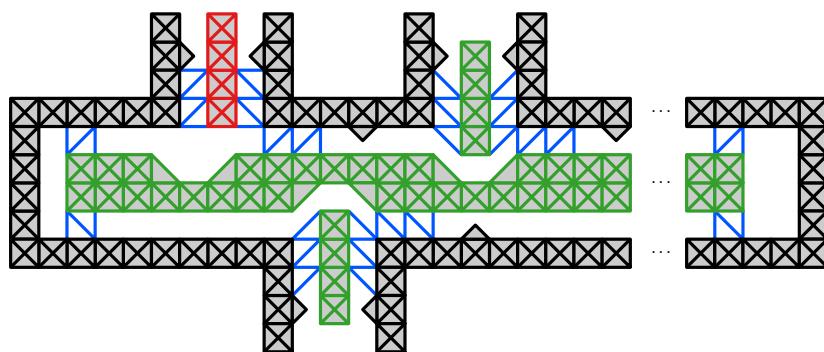
[Nöllenburg 2005]

Für einen eingebetteten Graphen G (Knotengrade ≤ 8) ist

Knickminimierung (R3) NP-schwer, wenn **Topologieerhaltung** (R1) und **Oktilinearität** (R2) gefordert sind.

Beweisidee.

Reduktion von PLANAR-3SAT mit starren mechanischen Gadgets



Komplexität



Satz.

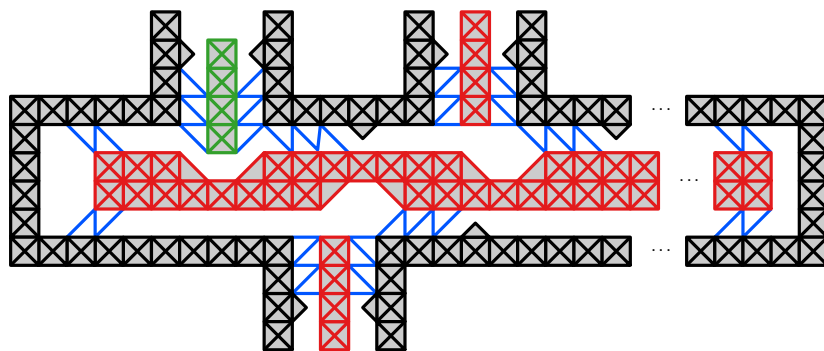
[Nöllenburg 2005]

Für einen eingebetteten Graphen G (Knotengrade ≤ 8) ist

Knickminimierung (R3) NP-schwer, wenn **Topologieerhaltung** (R1) und **Oktilinearität** (R2) gefordert sind.

Beweisidee.

Reduktion von PLANAR-3SAT mit starren mechanischen Gadgets



Komplexität



Satz.

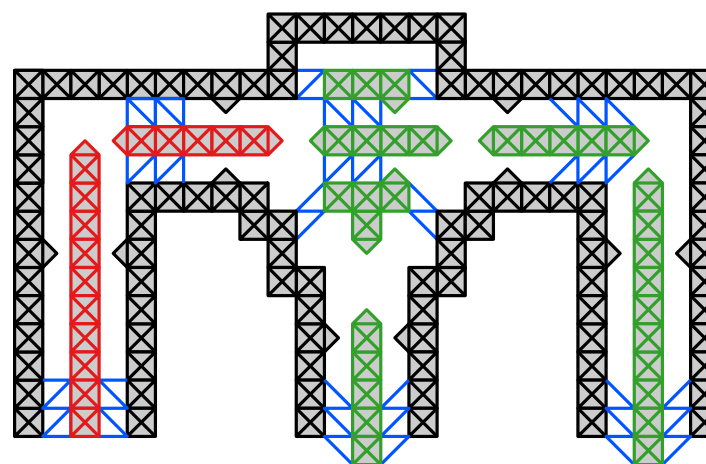
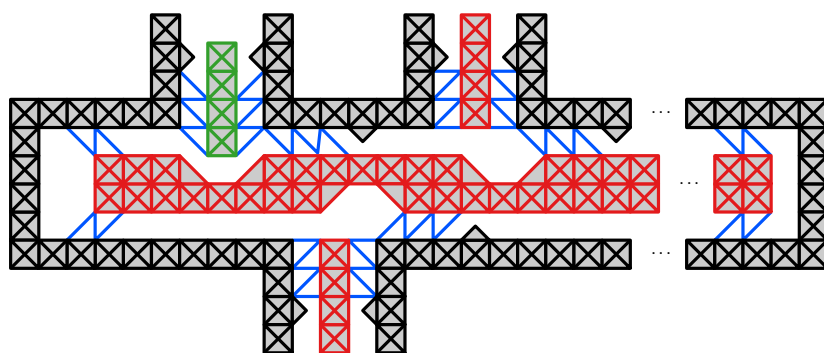
[Nöllenburg 2005]

Für einen eingebetteten Graphen G (Knotengrade ≤ 8) ist

Knickminimierung (R3) NP-schwer, wenn **Topologieerhaltung** (R1) und **Oktilinearität** (R2) gefordert sind.

Beweisidee.

Reduktion von PLANAR-3SAT mit starren mechanischen Gadgets



Komplexität



Satz.

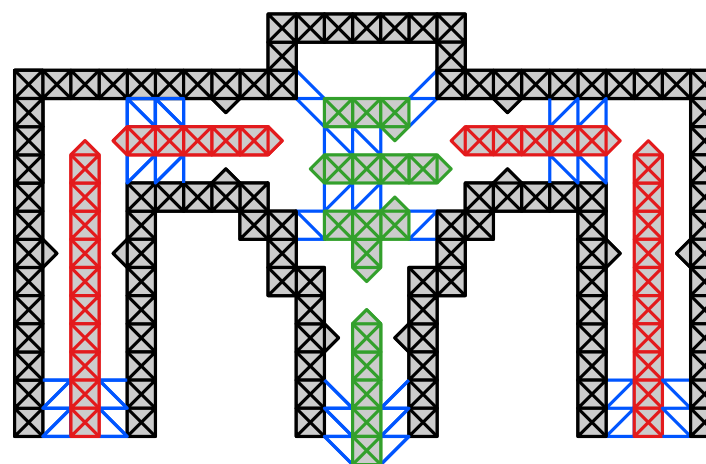
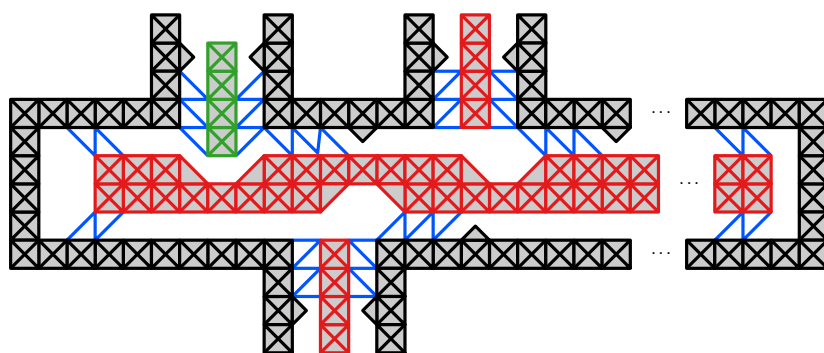
[Nöllenburg 2005]

Für einen eingebetteten Graphen G (Knotengrade ≤ 8) ist

Knickminimierung (R3) NP-schwer, wenn **Topologieerhaltung** (R1) und **Oktilinearität** (R2) gefordert sind.

Beweisidee.

Reduktion von PLANAR-3SAT mit starren mechanischen Gadgets



Komplexität



Satz.

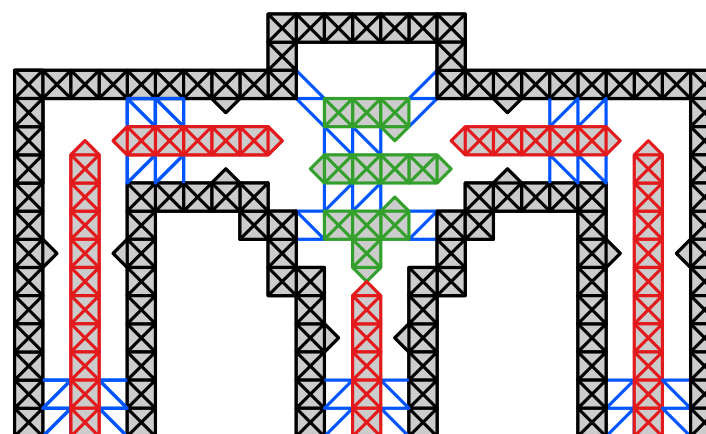
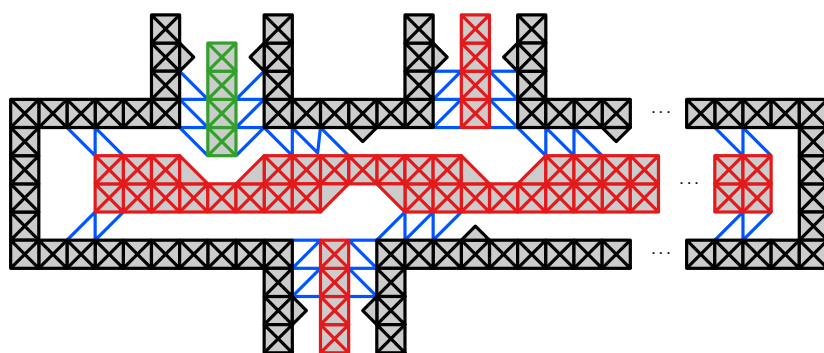
[Nöllenburg 2005]

Für einen eingebetteten Graphen G (Knotengrade ≤ 8) ist

Knickminimierung (R3) NP-schwer, wenn **Topologieerhaltung** (R1) und **Oktilinearität** (R2) gefordert sind.

Beweisidee.

Reduktion von PLANAR-3SAT mit starren mechanischen Gadgets



Komplexität



Satz.

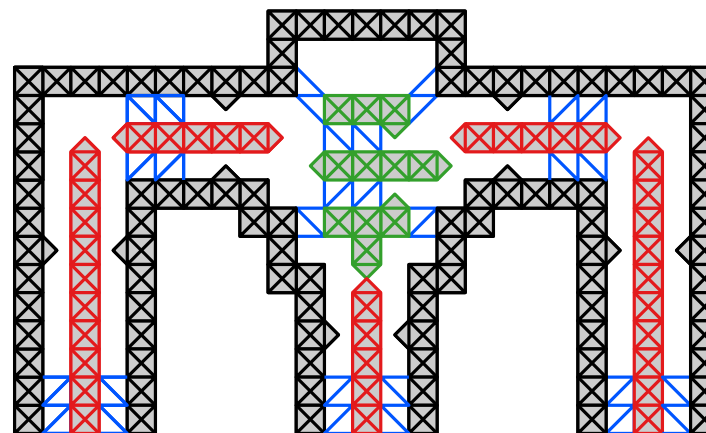
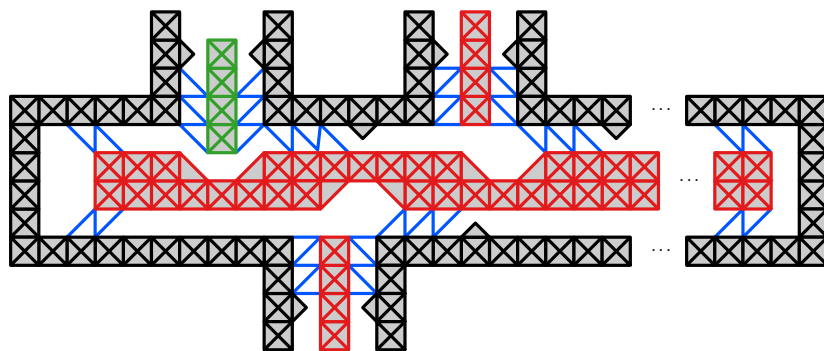
[Nöllenburg 2005]

Für einen eingebetteten Graphen G (Knotengrade ≤ 8) ist

Knickminimierung (R3) NP-schwer, wenn **Topologieerhaltung** (R1) und **Oktilinearität** (R2) gefordert sind.

Beweisidee.

Reduktion von PLANAR-3SAT mit starren mechanischen Gadgets



Anmerkung.

- wahrscheinlich keine effizienten exakten Algorithmen
- Das Problem ohne Diagonalen (orthogonal) ist effizient lösbar [Tamassia '87]

Pfadbasierte Schematisierung



Aufgabe.

Löse Spezialfall, in dem G ein Pfad (bzw. Linienzug) ist

Pfadbasierte Schematisierung



Aufgabe.

Löse Spezialfall, in dem G ein Pfad (bzw. Linienzug) ist

Beschränkungen.

- C -orientierte Kanten (z.B. oktilinear) **(R2)**
- beschränkte Verschiebung **(R6)**

Pfadbasierte Schematisierung



Aufgabe.

Löse Spezialfall, in dem G ein Pfad (bzw. Linienzug) ist

Beschränkungen.

- C -orientierte Kanten (z.B. oktilinear) **(R2)**
- beschränkte Verschiebung **(R6)**

Kriterien.

- minimiere Anzahl der Liniensegmente **(R3)**

Pfadbasierte Schematisierung

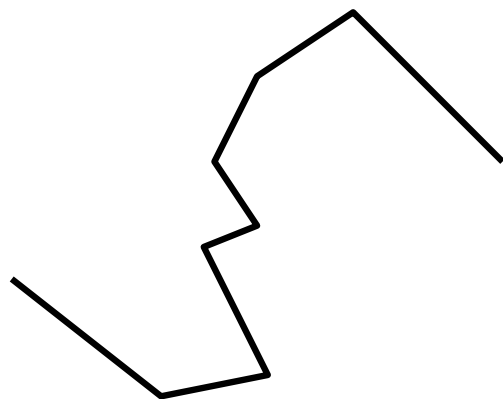


Aufgabe. Löse Spezialfall, in dem G ein Pfad (bzw. Linienzug) ist

Beschränkungen. ■ C -orientierte Kanten (z.B. oktilinear) **(R2)**

■ beschränkte Verschiebung **(R6)**

Kriterien. ■ minimiere Anzahl der Liniensegmente **(R3)**





Pfadbasierte Schematisierung

Aufgabe.

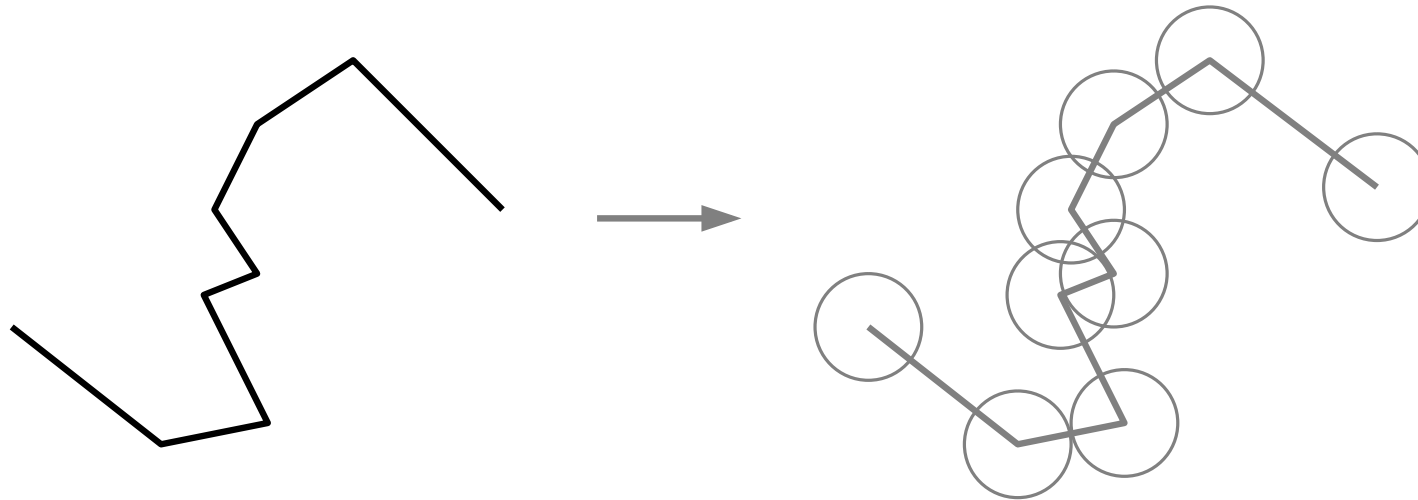
Löse Spezialfall, in dem G ein Pfad (bzw. Linienzug) ist

Beschränkungen.

- C -orientierte Kanten (z.B. oktilinear) **(R2)**
- beschränkte Verschiebung **(R6)**

Kriterien.

- minimiere Anzahl der Liniensegmente **(R3)**





Pfadbasierte Schematisierung

Aufgabe.

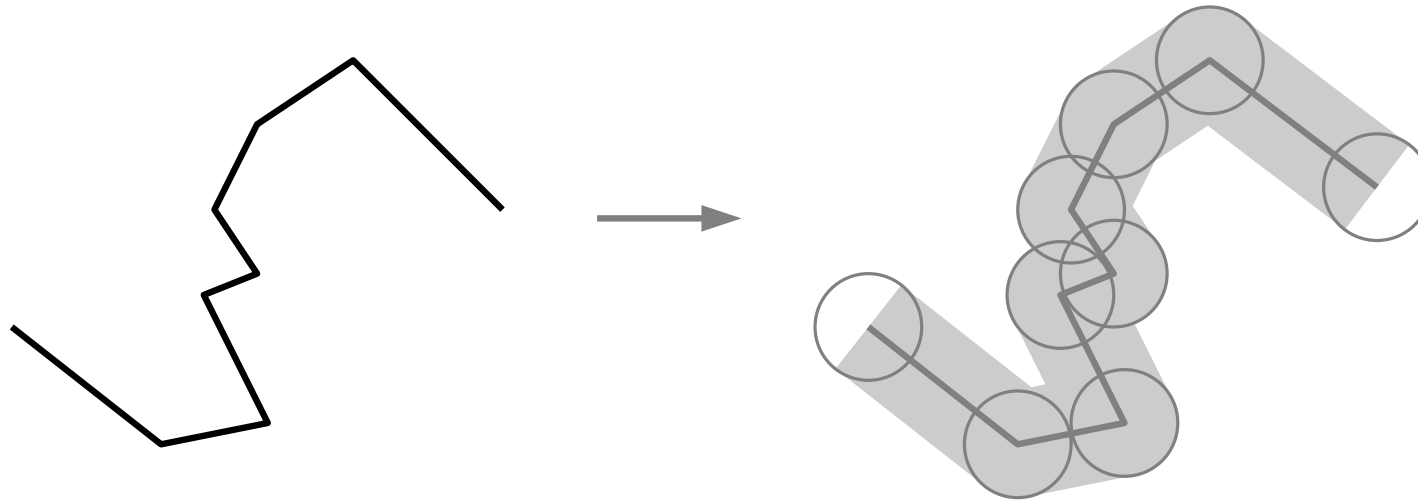
Löse Spezialfall, in dem G ein Pfad (bzw. Linienzug) ist

Beschränkungen.

- C -orientierte Kanten (z.B. oktilinear) **(R2)**
- beschränkte Verschiebung **(R6)**

Kriterien.

- minimiere Anzahl der Liniensegmente **(R3)**





Pfadbasierte Schematisierung

Aufgabe.

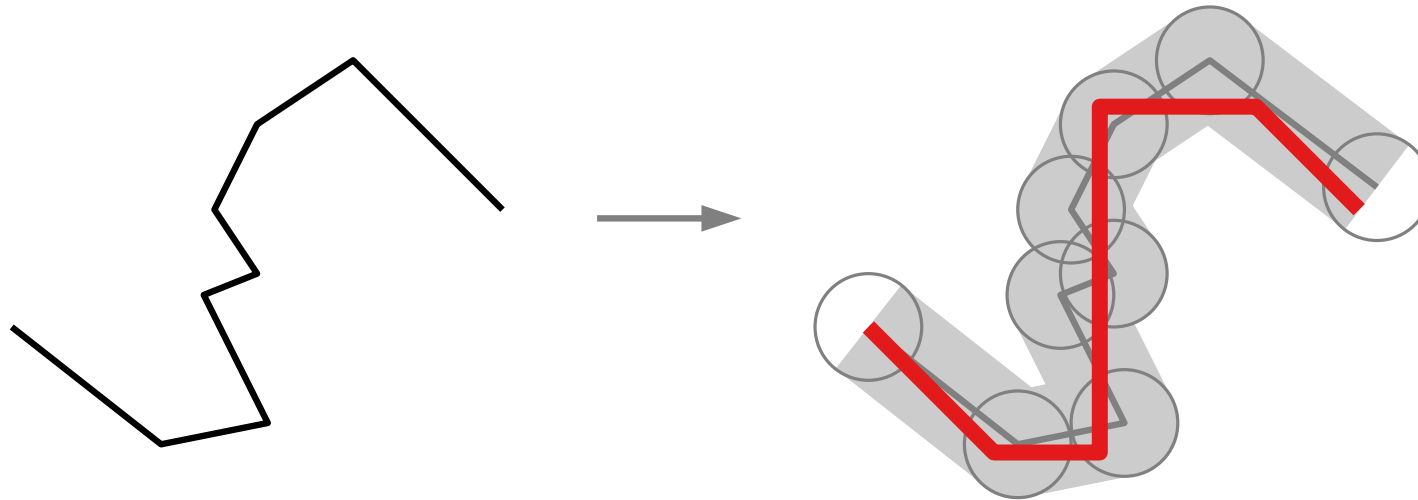
Löse Spezialfall, in dem G ein Pfad (bzw. Linienzug) ist

Beschränkungen.

- C -orientierte Kanten (z.B. oktilinear) (R2)
- beschränkte Verschiebung (R6)

Kriterien.

- minimiere Anzahl der Liniensegmente (R3)



Pfadbasierte Schematisierung – Beispiel



Satz.

[Dwyer, Hurst, Merrick 2008]

Für einen Pfad P der Länge n und eine Richtungsmenge $\mathcal{C}$ kann mit Hilfe der Methode der kleinsten Quadrate ein $\mathcal{C}$ -orientierter schematisierter Pfad in $\mathcal{O}(|\mathcal{C}|n)$ Zeit berechnet werden, der „fast“ durch die Knoten geht.

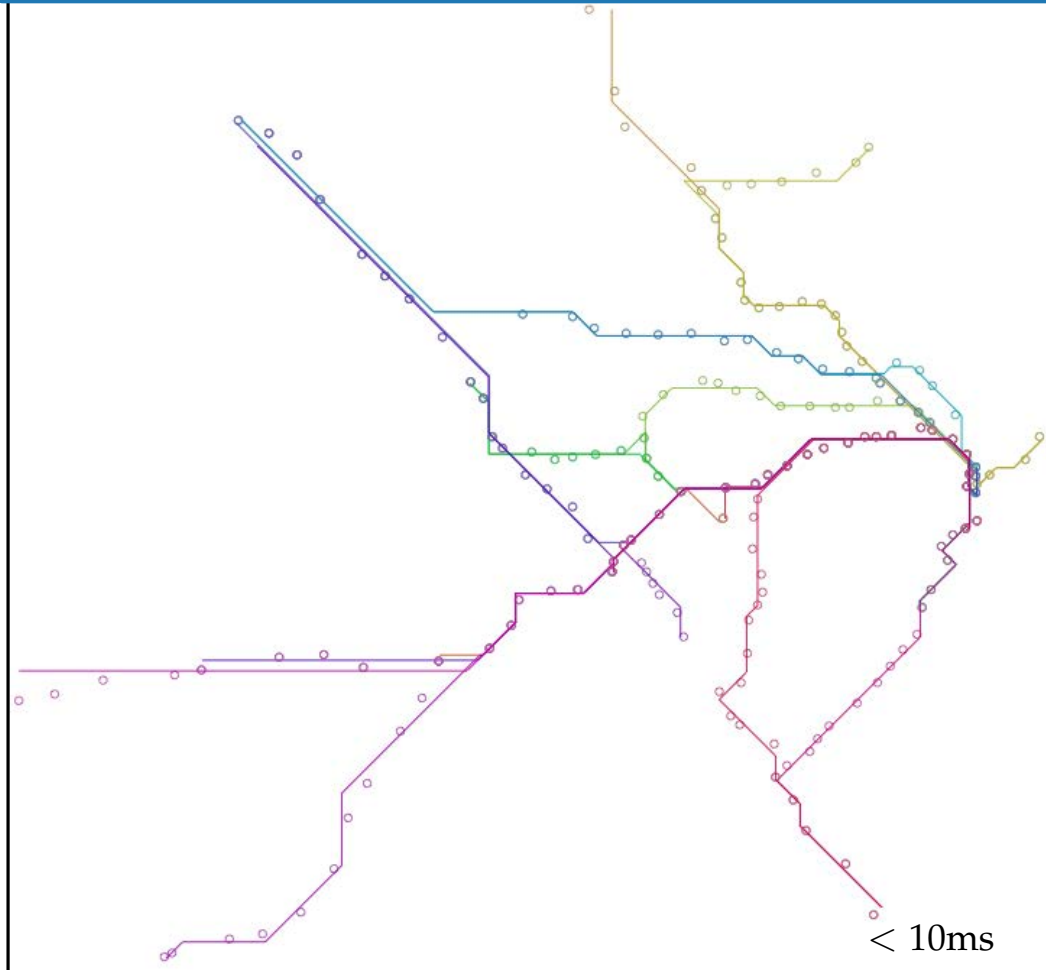
Pfadbasierte Schematisierung – Beispiel



Satz.

[Dwyer, Hurst, Merrick 2008]

Für einen Pfad P der Länge n und eine Richtungsmenge $\mathcal{C}$ kann mit Hilfe der Methode der kleinsten Quadrate ein $\mathcal{C}$ -orientierter schematisierter Pfad in $\mathcal{O}(|\mathcal{C}|n)$ Zeit berechnet werden, der „fast“ durch die Knoten geht.



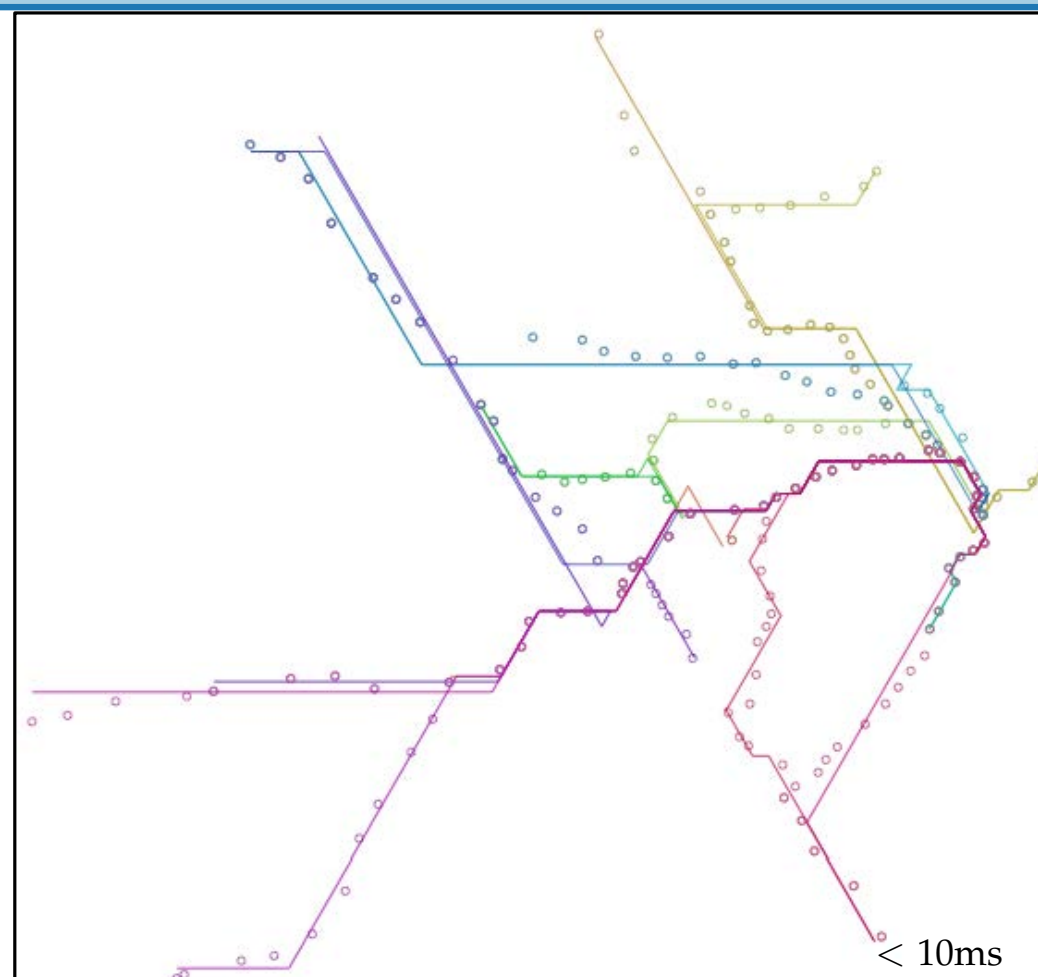
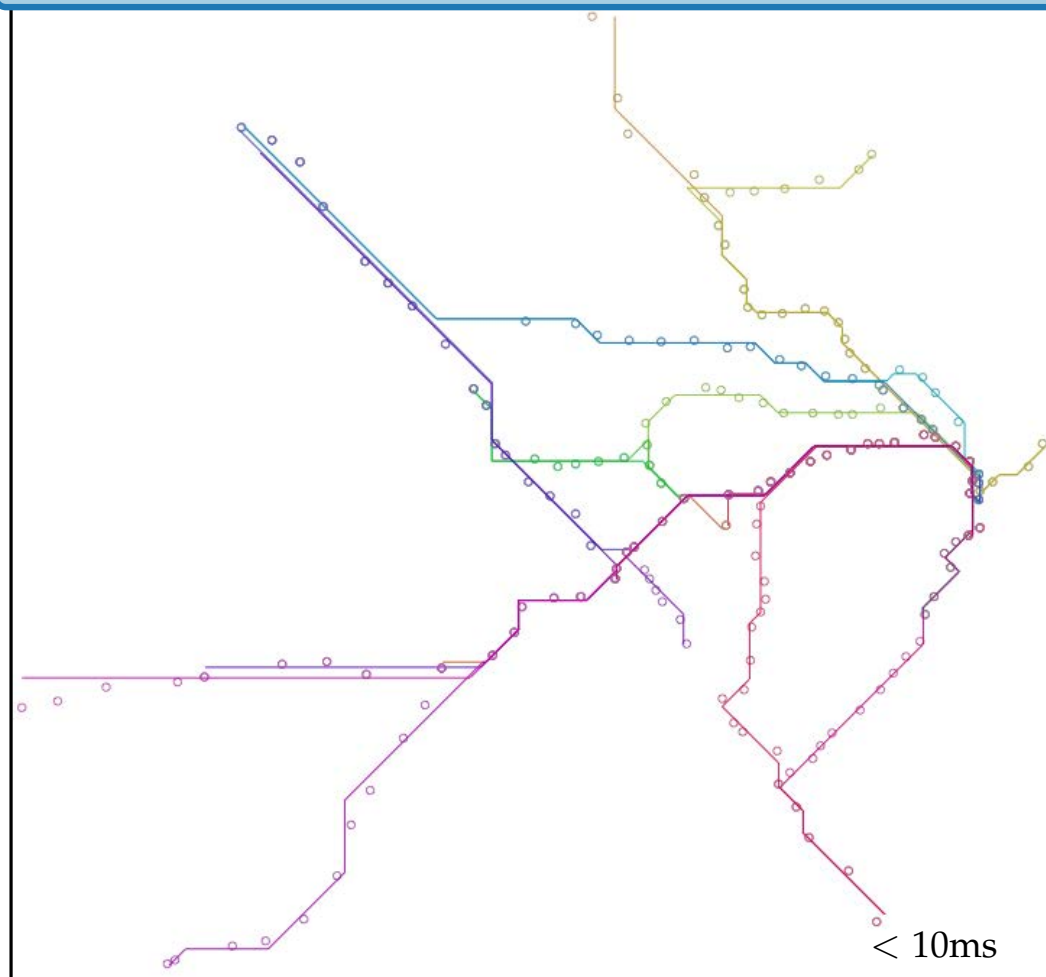
Pfadbasierte Schematisierung – Beispiel



Satz.

[Dwyer, Hurst, Merrick 2008]

Für einen Pfad P der Länge n und eine Richtungsmenge $\mathcal{C}$ kann mit Hilfe der Methode der kleinsten Quadrate ein $\mathcal{C}$ -orientierter schematisierter Pfad in $\mathcal{O}(|\mathcal{C}|n)$ Zeit berechnet werden, der „fast“ durch die Knoten geht.



$\mathcal{C}$ -orientierte Routenskizzen



Satz.

[Delling et al. 2010]

Für einen monotonen Pfad P und eine Menge $\mathcal{C}$ zulässiger Steigungen können wir in $\mathcal{O}(n^2) + \text{solve(LP)}$ -Zeit eine $\mathcal{C}$ -orientierte Schematisierung von P berechnen, die

- die orthogonale Reihenfolge von P bewahrt,
- minimale Steigungsabweichung hat,
- minimale Gesamtlänge hat.



$\mathcal{C}$ -orientierte Routenskizzen

Satz.

[Delling et al. 2010]

Für einen monotonen Pfad P und eine Menge $\mathcal{C}$ zulässiger Steigungen können wir in $\mathcal{O}(n^2) + \text{solve(LP)}$ -Zeit eine $\mathcal{C}$ -orientierte Schematisierung von P berechnen, die

- die orthogonale Reihenfolge von P bewahrt,
- minimale Steigungsabweichung hat,
- minimale Gesamtlänge hat.

relatives Nord-Süd-Öst-West
Verhältnis zwischen allen Knoten



$\mathcal{C}$ -orientierte Routenskizzen

Satz.

[Delling et al. 2010]

Für einen monotonen Pfad P und eine Menge $\mathcal{C}$ zulässiger Steigungen können wir in $\mathcal{O}(n^2) + \text{solve(LP)}$ -Zeit eine $\mathcal{C}$ -orientierte Schematisierung von P berechnen, die

- die orthogonale Reihenfolge von P bewahrt,
- minimale Steigungsabweichung hat,
- minimale Gesamtlänge hat.

relatives Nord-Süd-Öst-West
Verhältnis zwischen allen Knoten

Beweisidee.

- Dynamisches Programm für Steigungen
- LP für Kantenlängen



$\mathcal{C}$ -orientierte Routenskizzen

Satz.

[Delling et al. 2010]

Für einen monotonen Pfad P und eine Menge $\mathcal{C}$ zulässiger Steigungen können wir in $\mathcal{O}(n^2) + \text{solve(LP)}$ -Zeit eine $\mathcal{C}$ -orientierte Schematisierung von P berechnen, die

- die orthogonale Reihenfolge von P bewahrt,
- minimale Steigungsabweichung hat,
- minimale Gesamtlänge hat.

relatives Nord-Süd-Öst-West
Verhältnis zwischen allen Knoten

Beweisidee.

- Dynamisches Programm für Steigungen
- LP für Kantenlängen

Satz.

[Brandes & Pampel 2009, Gemsa et al. 2011]

Das d -reguläre (nicht-monotone) Routenskizzen Problem ist NP-schwer für jedes $d \geq 1$, wobei $\mathcal{C} = \{i \cdot 90^\circ / d \mid i \in \mathbb{Z}\}$ gilt.

C-orientierte Routenskizzen



Beispiel. Bremen nach Cuxhaven

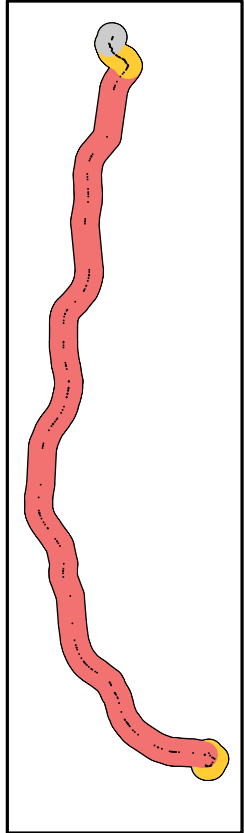
[Gemsa et al. 2011]

C-orientierte Routenskizzen



Beispiel. Bremen nach Cuxhaven

[Gemsa et al. 2011]



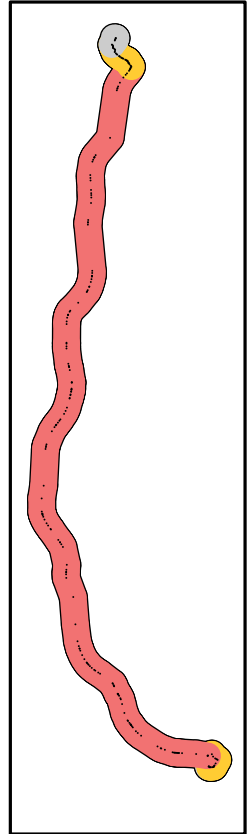
Eingabe

C -orientierte Routenskizzen

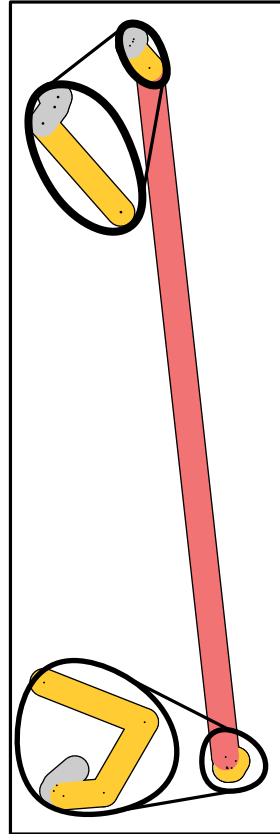


Beispiel. Bremen nach Cuxhaven

[Gemsa et al. 2011]



Eingabe

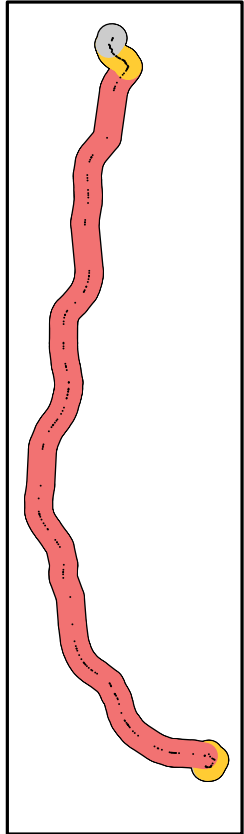


vereinfacht

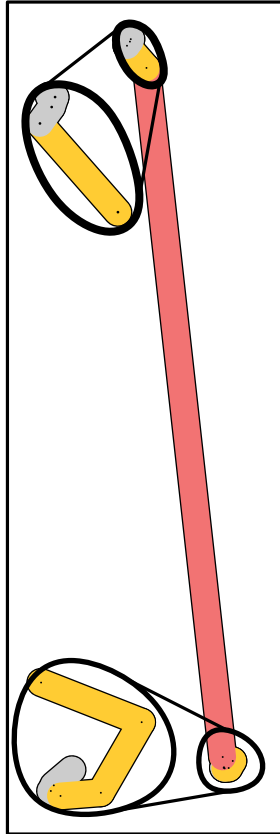
C-orientierte Routenskizzen

Beispiel. Bremen nach Cuxhaven

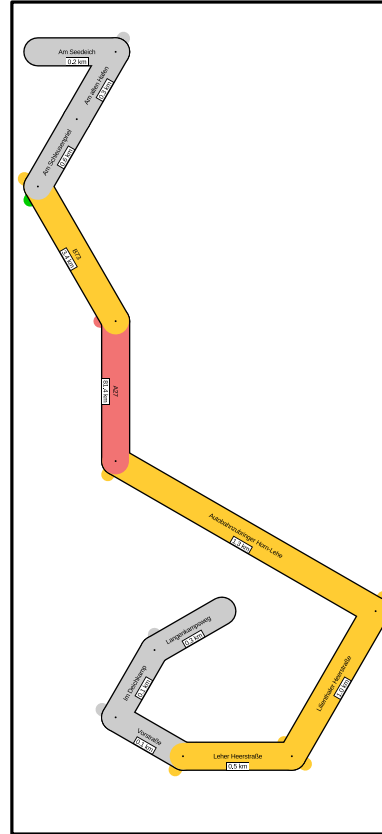
[Gemsa et al. 2011]



Eingabe



vereinfacht

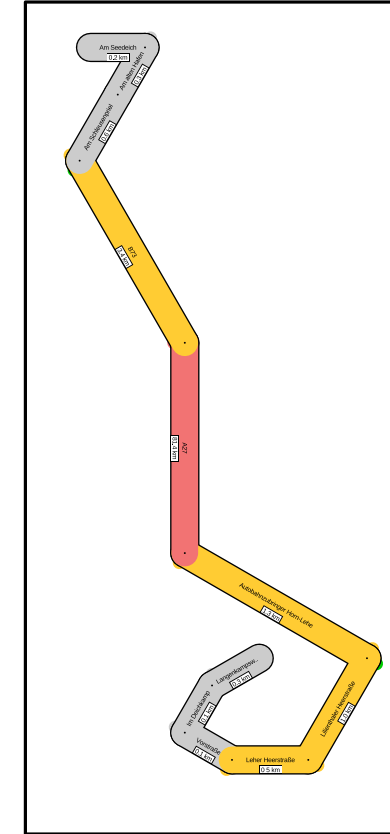
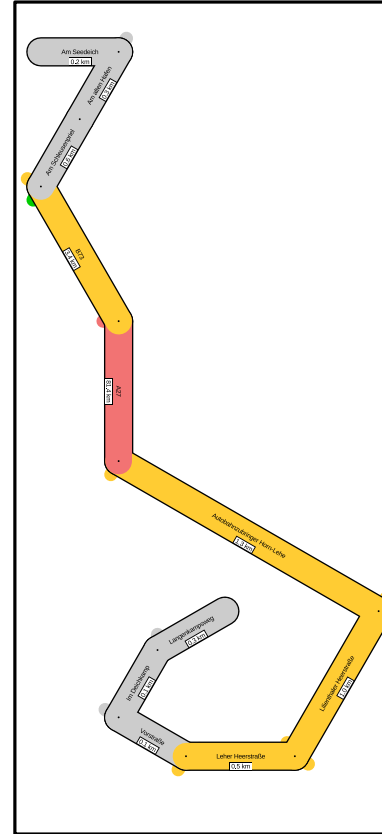
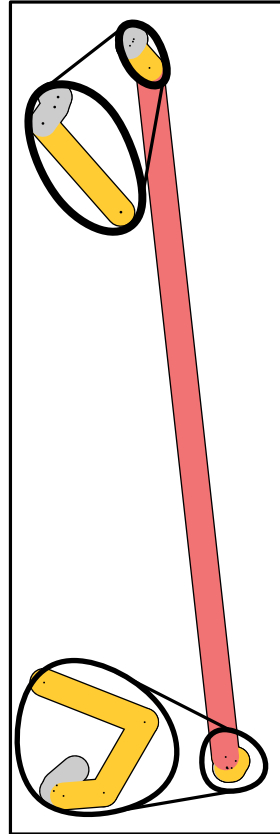
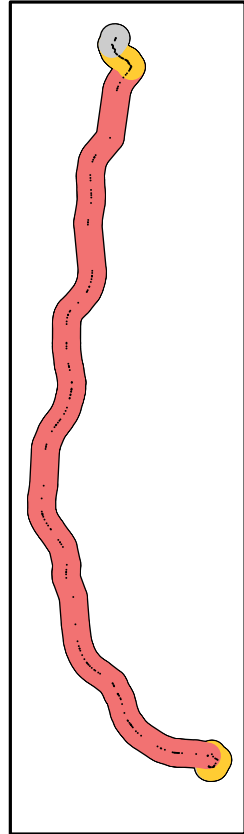


optimiert durch MIP

$\mathcal{C}$ -orientierte Routenskizzen

Beispiel. Bremen nach Cuxhaven

[Gemsa et al. 2011]



Pfadbasierte Schematisierung – Diskussion



Vorteile.

Nachteile.

Pfadbasierte Schematisierung – Diskussion



Vorteile.

- Polynomielle Laufzeiten

Nachteile.

Pfadbasierte Schematisierung – Diskussion



Vorteile.

- Polynomielle Laufzeiten
- $\mathcal{C}$ -Orientierung und garantierte begrenzte Verschiebung

Nachteile.

Pfadbasierte Schematisierung – Diskussion



Vorteile.

- Polynomielle Laufzeiten
- $\mathcal{C}$ -Orientierung und garantierte begrenzte Verschiebung
- Knickminimierung Erweiterung auf U-Bahn-Netzwerke:

Nachteile.

Pfadbasierte Schematisierung – Diskussion



Vorteile.

- Polynomielle Laufzeiten
- $\mathcal{C}$ -Orientierung und garantierte begrenzte Verschiebung
- Knickminimierung Erweiterung auf U-Bahn-Netzwerke:
 - Zerlegung des U-Bahn-Netzwerks in Pfade

Nachteile.

Pfadbasierte Schematisierung – Diskussion



Vorteile.

- Polynomielle Laufzeiten
- $\mathcal{C}$ -Orientierung und garantierte begrenzte Verschiebung
- Knickminimierung Erweiterung auf U-Bahn-Netzwerke:
 - Zerlegung des U-Bahn-Netzwerks in Pfade
 - Schematisierung einzelner Pfade

Nachteile.

Pfadbasierte Schematisierung – Diskussion



Vorteile.

- Polynomielle Laufzeiten
- $\mathcal{C}$ -Orientierung und garantierte begrenzte Verschiebung
- Knickminimierung Erweiterung auf U-Bahn-Netzwerke:
 - Zerlegung des U-Bahn-Netzwerks in Pfade
 - Schematisierung einzelner Pfade
 - Zusammenfügen der schematisierten Pfade an Stationen

Nachteile.

Pfadbasierte Schematisierung – Diskussion



Vorteile.

- Polynomielle Laufzeiten
- $\mathcal{C}$ -Orientierung und garantierte begrenzte Verschiebung
- Knickminimierung Erweiterung auf U-Bahn-Netzwerke:
 - Zerlegung des U-Bahn-Netzwerks in Pfade
 - Schematisierung einzelner Pfade
 - Zusammenfügen der schematisierten Pfade an Stationen

Nachteile.

- Keine Garantie für die Topologie **(R1)**

Pfadbasierte Schematisierung – Diskussion



Vorteile.

- Polynomielle Laufzeiten
- $\mathcal{C}$ -Orientierung und garantierte begrenzte Verschiebung
- Knickminimierung Erweiterung auf U-Bahn-Netzwerke:
 - Zerlegung des U-Bahn-Netzwerks in Pfade
 - Schematisierung einzelner Pfade
 - Zusammenfügen der schematisierten Pfade an Stationen

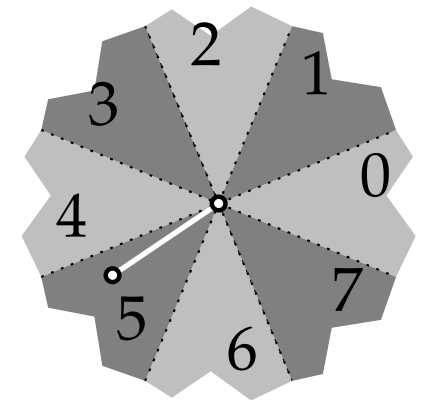
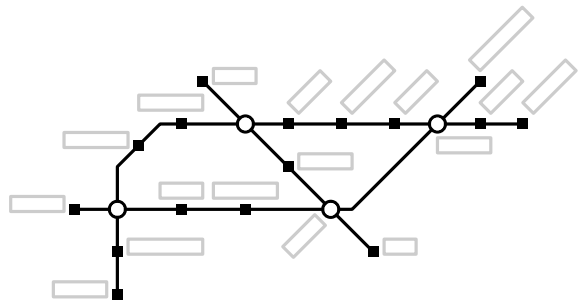
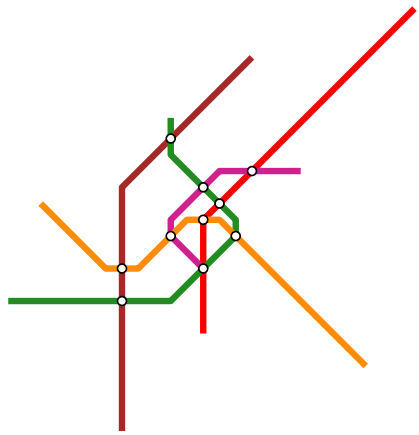
Nachteile.

- Keine Garantie für die Topologie (R1)
- Verzerrung/Verschiebung ist für U-Bahn-Netzpläne zu begrenzt

Algorithmen für geographische Informationssysteme

11. Vorlesung Netzpläne (Metro Maps)

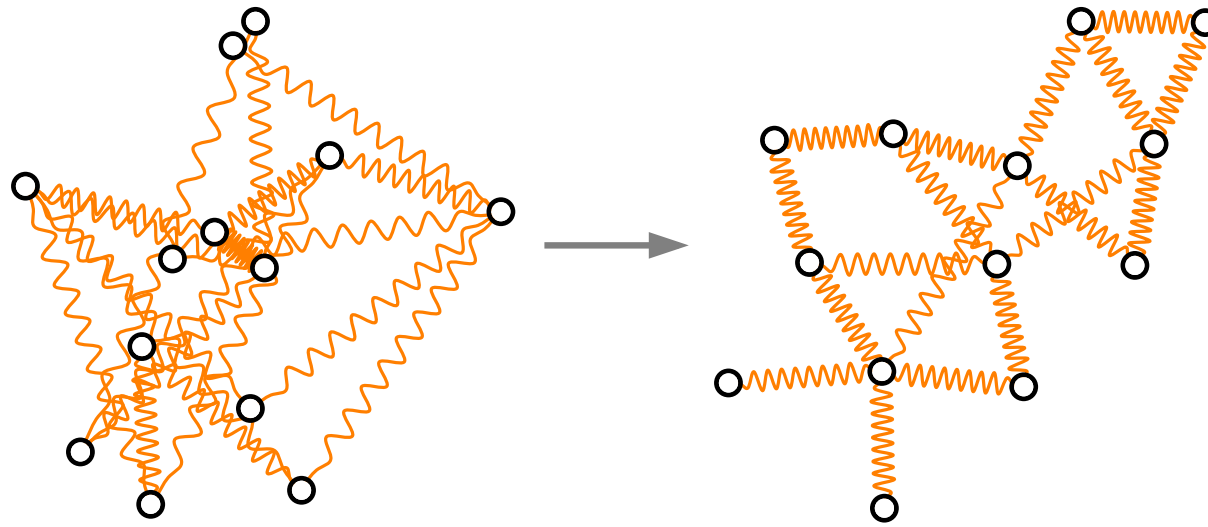
Teil III: Kräftebasierte Schematisierung



Kräftebasierte Schematisierung

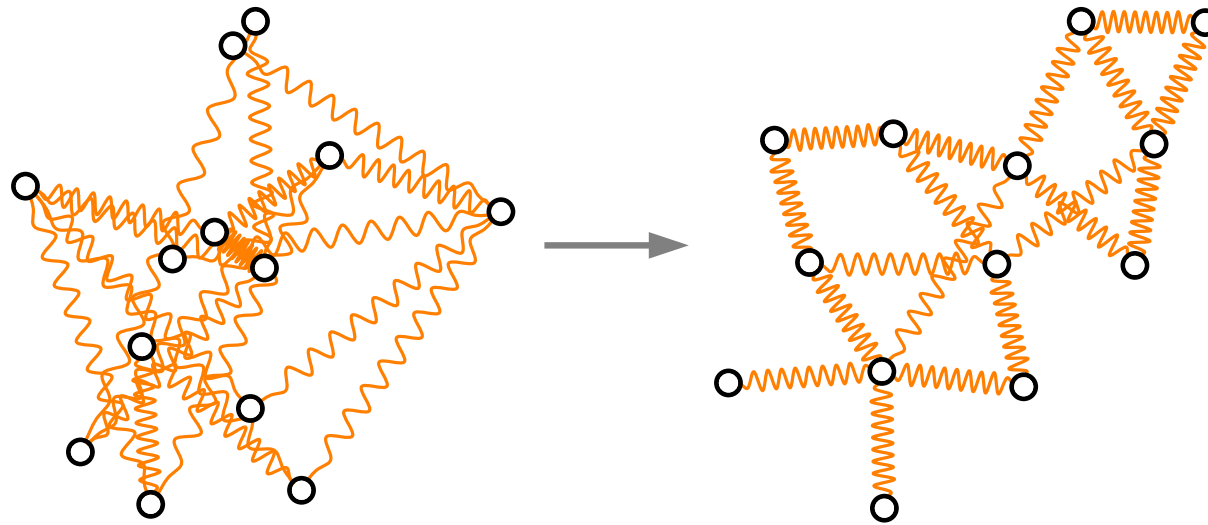


Idee. Verwende kräftebasierten Graphenzeichnen Ansatz.



Kräftebasierte Schematisierung

Idee. Verwende kräftebasierten Graphenzeichnen Ansatz.

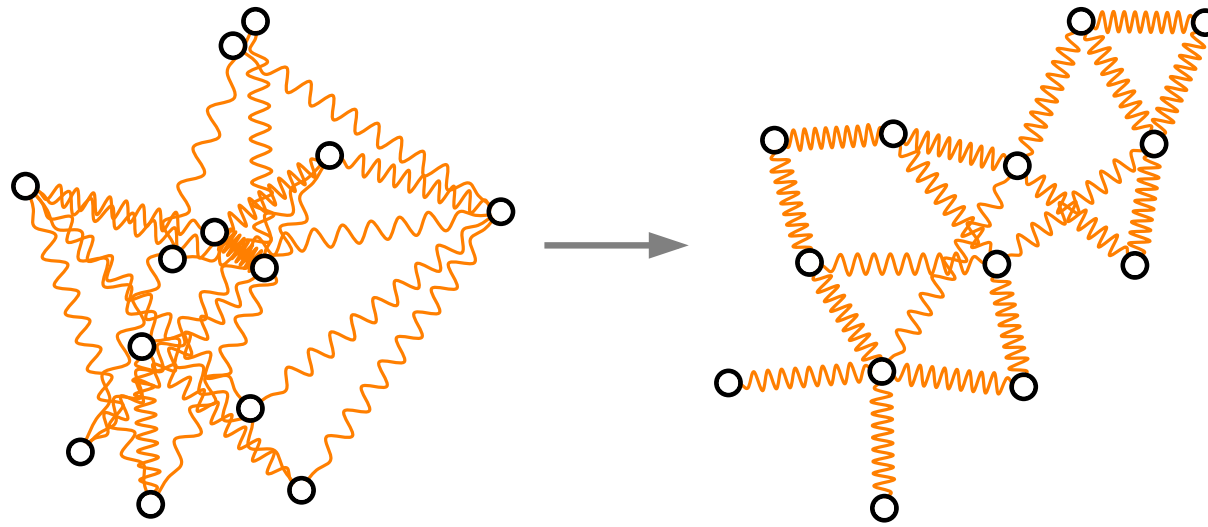


Prinzip.

- Knoten sind geladene Teilchen, die sich gegenseitig abstoßen

Kräftebasierte Schematisierung

Idee. Verwende kräftebasierten Graphenzeichnen Ansatz.

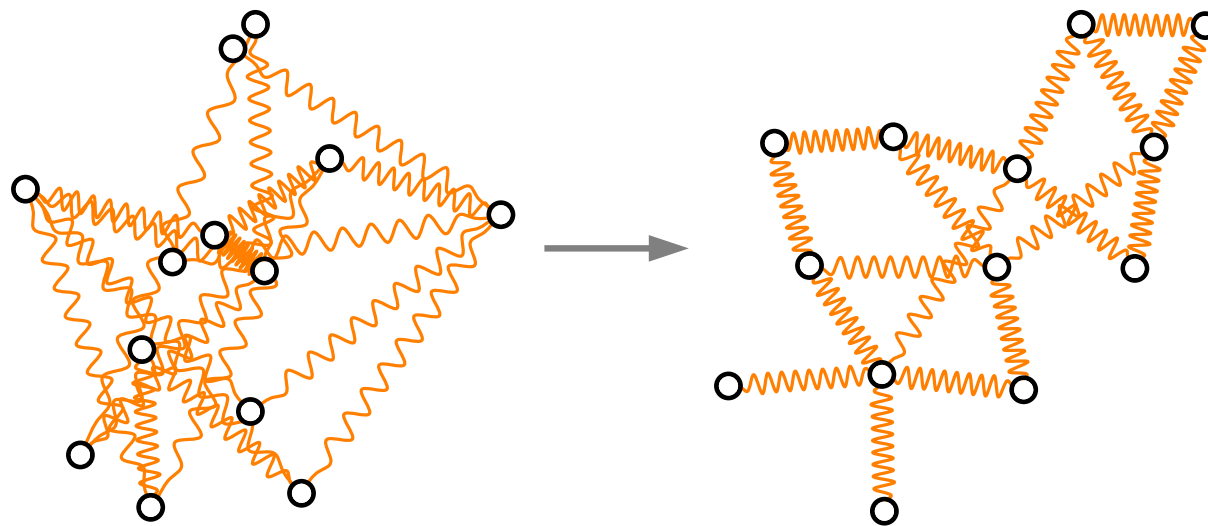


Prinzip.

- Knoten sind geladene Teilchen, die sich gegenseitig abstoßen
- Kanten sind Federn, die die Kanten auf die Ziel-Länge ziehen

Kräftebasierte Schematisierung

Idee. Verwende kräftebasierten Graphenzeichnen Ansatz.

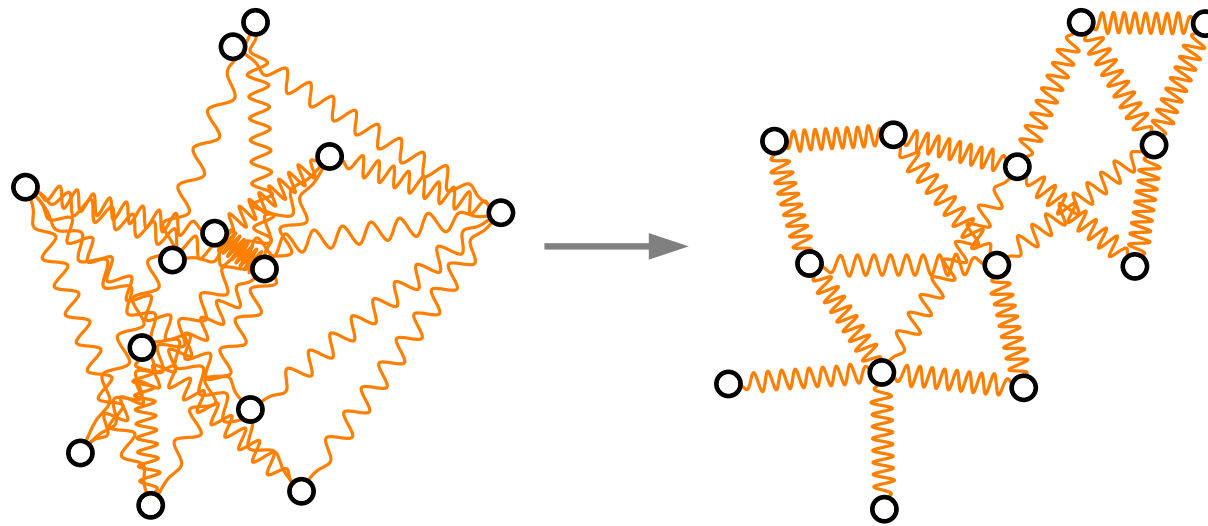


Prinzip.

- Knoten sind geladene Teilchen, die sich gegenseitig abstoßen
- Kanten sind Federn, die die Kanten auf die Ziel-Länge ziehen
- Iterative Berechnung und Anwendung der Kräfte, bis das System stabil ist

Kräftebasierte Schematisierung

Idee. Verwende kräftebasierten Graphenzeichnen Ansatz.



Prinzip.

- Knoten sind geladene Teilchen, die sich gegenseitig abstoßen
- Kanten sind Federn, die die Kanten auf die Ziel-Länge ziehen
- Iterative Berechnung und Anwendung der Kräfte, bis das System stabil ist
- ➔ definiere zusätzliche Kräfte, um Designregeln zu modellieren

Kräftebasierte Schematisierung – Oktilinear

[Hong et al. 2006]

- kontrahiere Grad-2-Knoten in gewichtete Kanten (R3)



Kräftebasierte Schematisierung – Oktilinear

[Hong et al. 2006]

- kontrahiere Grad-2-Knoten in gewichtete Kanten (R3)
- definiere ein oktilineares Magnetfeld, das Kanten anzieht (R2)



Kräftebasierte Schematisierung – Oktilinear

[Hong et al. 2006]

- kontrahiere Grad-2-Knoten in gewichtete Kanten **(R3)**
- definiere ein oktilineares Magnetfeld, das Kanten anzieht **(R2)**
- wende nur topologieerhaltende Knotenbewegungen an **(R1)**



Kräftebasierte Schematisierung – Oktilinear

[Hong et al. 2006]



- kontrahiere Grad-2-Knoten in gewichtete Kanten **(R3)**
- definiere ein oktilineares Magnetfeld, das Kanten anzieht **(R2)**
- wende nur topologieerhaltende Knotenbewegungen an **(R1)**
- Federnlängen modellieren einheitliche Kantenlängen **(R7)**

Kräftebasierte Schematisierung – Oktilinear

[Hong et al. 2006]



- kontrahiere Grad-2-Knoten in gewichtete Kanten (R3)
- definiere ein oktilineares Magnetfeld, das Kanten anzieht (R2)
- wende nur topologieerhaltende Knotenbewegungen an (R1)
- Federnlängen modellieren einheitliche Kantenlängen (R7)
- Knotenabstoßung modelliert Abstände zwischen separaten Teilen (R8)

Kräftebasierte Schematisierung – Oktilinear

[Hong et al. 2006]



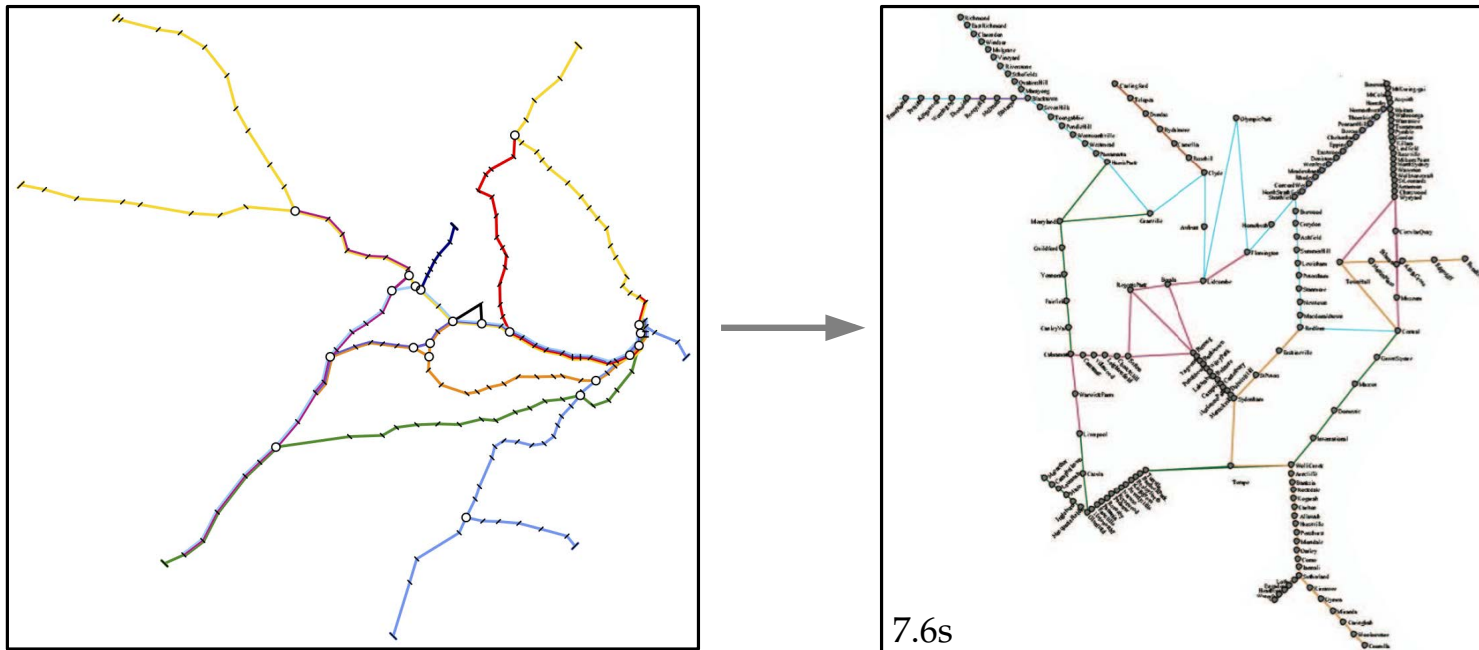
- kontrahiere Grad-2-Knoten in gewichtete Kanten (R3)
- definiere ein oktilineares Magnetfeld, das Kanten anzieht (R2)
- wende nur topologieerhaltende Knotenbewegungen an (R1)
- Federnlängen modellieren einheitliche Kantenlängen (R7)
- Knotenabstoßung modelliert Abstände zwischen separaten Teilen (R8)
- Beschriftungen werden im unabhängigen 2. Schritt platziert

Kräftebasierte Schematisierung – Oktilinear

[Hong et al. 2006]



- kontrahiere Grad-2-Knoten in gewichtete Kanten (R3)
- definiere ein oktilineares Magnetfeld, das Kanten anzieht (R2)
- wende nur topologieerhaltende Knotenbewegungen an (R1)
- Federnlängen modellieren einheitliche Kantenlängen (R7)
- Knotenabstoßung modelliert Abstände zwischen separaten Teilen (R8)
- Beschriftungen werden im unabhängigen 2. Schritt platziert

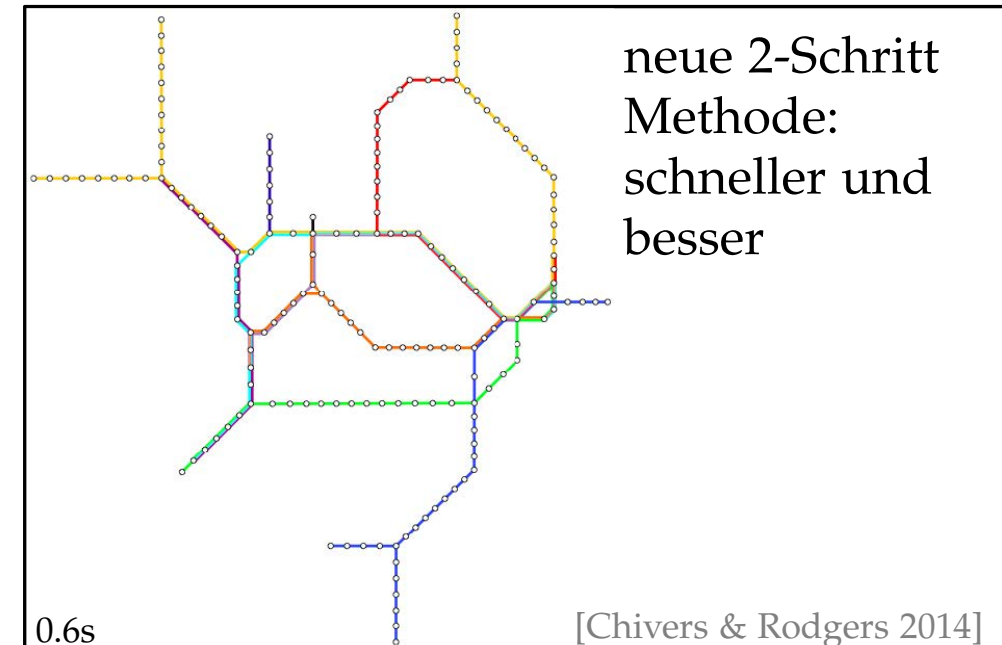
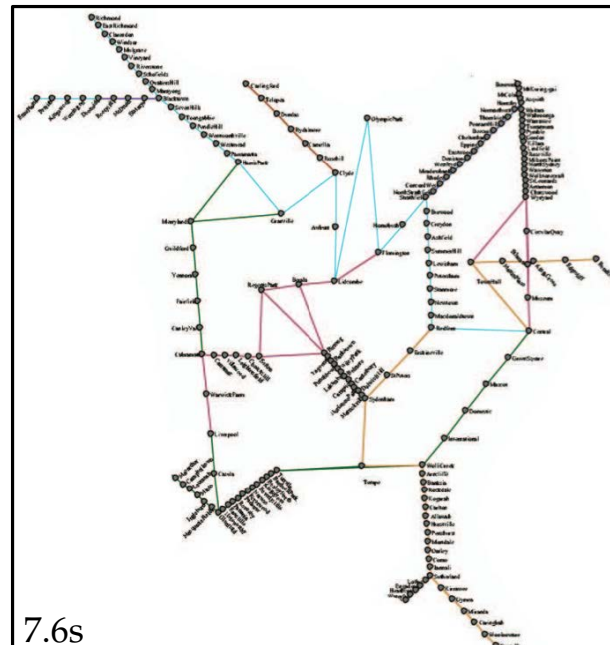
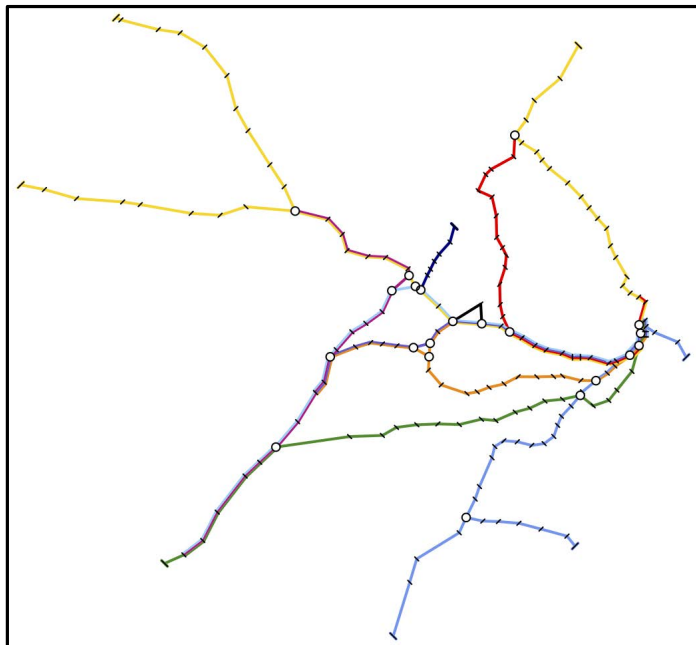


Kräftebasierte Schematisierung – Oktilinear

[Hong et al. 2006]



- kontrahiere Grad-2-Knoten in gewichtete Kanten (R3)
- definiere ein oktilineares Magnetfeld, das Kanten anzieht (R2)
- wende nur topologieerhaltende Knotenbewegungen an (R1)
- Federnlängen modellieren einheitliche Kantenlängen (R7)
- Knotenabstoßung modelliert Abstände zwischen separaten Teilen (R8)
- Beschriftungen werden im unabhängigen 2. Schritt platziert



Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]

- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
 - ➔ Knoten und Kontrollpunkte, beide unterliegen Kräften



Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]



- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
 - ➔ Knoten und Kontrollpunkte, beide unterliegen Kräften
- Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten (R4)

Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]



- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
 - ➔ Knoten und Kontrollpunkte, beide unterliegen Kräften
- Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten **(R4)**
- (Standard) anziehende und abstoßende Kräfte **(R7)+(R8)**

Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]



- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
 - ➔ Knoten und Kontrollpunkte, beide unterliegen Kräften
- Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten **(R4)**
- (Standard) anziehende und abstoßende Kräfte **(R7)+(R8)**
- (schwache) Kraft in Richtung der ursprünglichen Position **(R6)**

Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]



- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
 - ➔ Knoten und Kontrollpunkte, beide unterliegen Kräften
- Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten **(R4)**
- (Standard) anziehende und abstoßende Kräfte **(R7)+(R8)**
- (schwache) Kraft in Richtung der ursprünglichen Position **(R6)**
- Kräfte zur Verbesserung der Kurvenform (geringe Krümmung) **(R3)**

Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]



- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
→ Knoten und Kontrollpunkte, beide unterliegen Kräften
- Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten (R4)
- (Standard) anziehende und abstoßende Kräfte (R7)+(R8)
- (schwache) Kraft in Richtung der ursprünglichen Position (R6)
- Kräfte zur Verbesserung der Kurvenform (geringe Krümmung) (R3)
- Kurven vereinen, wann immer möglich (R3)

Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]



- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
→ Knoten und Kontrollpunkte, beide unterliegen Kräften
- Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten (R4)
- (Standard) anziehende und abstoßende Kräfte (R7)+(R8)
- (schwache) Kraft in Richtung der ursprünglichen Position (R6)
- Kräfte zur Verbesserung der Kurvenform (geringe Krümmung) (R3)
- Kurven vereinen, wann immer möglich (R3)
- Kräfte zur Verbesserung der Winkelauflösung (R5)

Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]

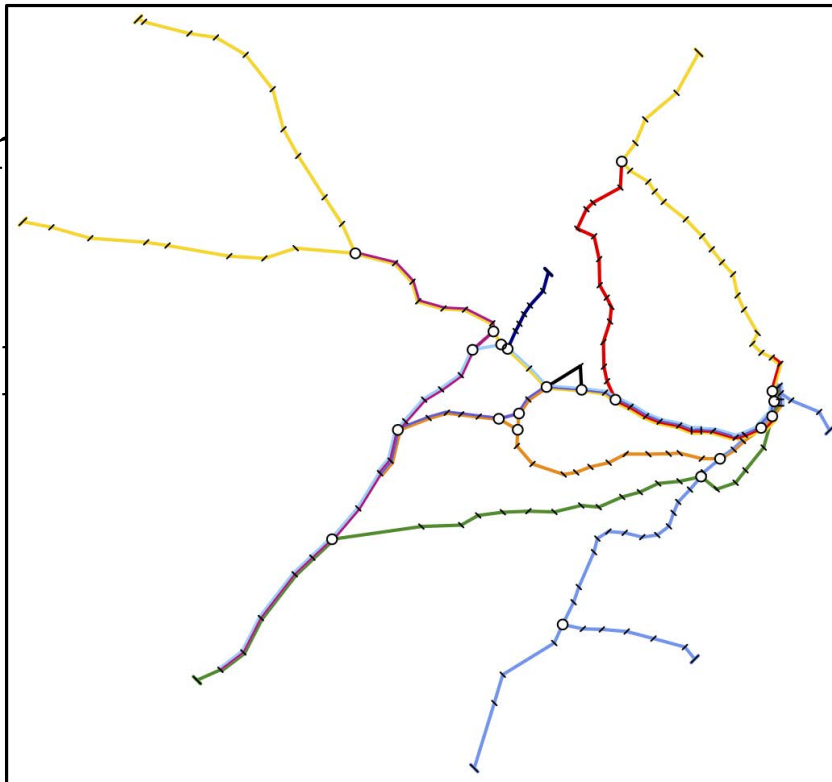


- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
→ Knoten und Kontrollpunkte, beide unterliegen Kräften
- Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten (R4)
- (Standard) anziehende und abstoßende Kräfte (R7)+(R8)
- (schwache) Kraft in Richtung der ursprünglichen Position (R6)
- Kräfte zur Verbesserung der Kurvenform (geringe Krümmung) (R3)
- Kurven vereinen, wann immer möglich (R3)
- Kräfte zur Verbesserung der Winkelauflösung (R5)
- Nur topologieerhaltende Bewegungen anwenden (R1)

Kräftebasierte Schematisierung – Bézier

[Fink et al. 2013]

- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
 - ➔ Knoten und Kontrollpunkte, beide unterliegen Kräften
 - Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten (R4)
 - (Standard) anziehende und abstoßende Kräfte (R7)+(R8)
 - (schwache) Kraft in Richtung der ursprünglichen Position (R6)
 - Kräfte
 - Kurven
 - Kräfte
 - Nur to
- enform (geringe Krümmung) (R3)
- öglich (R3)
- elsauflösung (R5)
- ngen anwenden (R1)

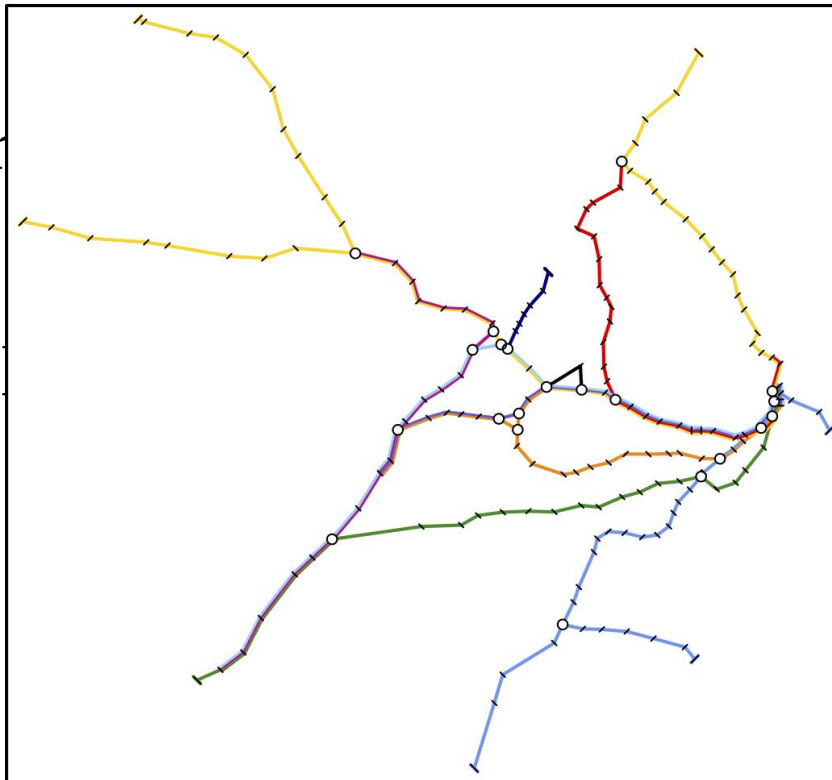


Kräftebasierte Schematisierung – Bézier

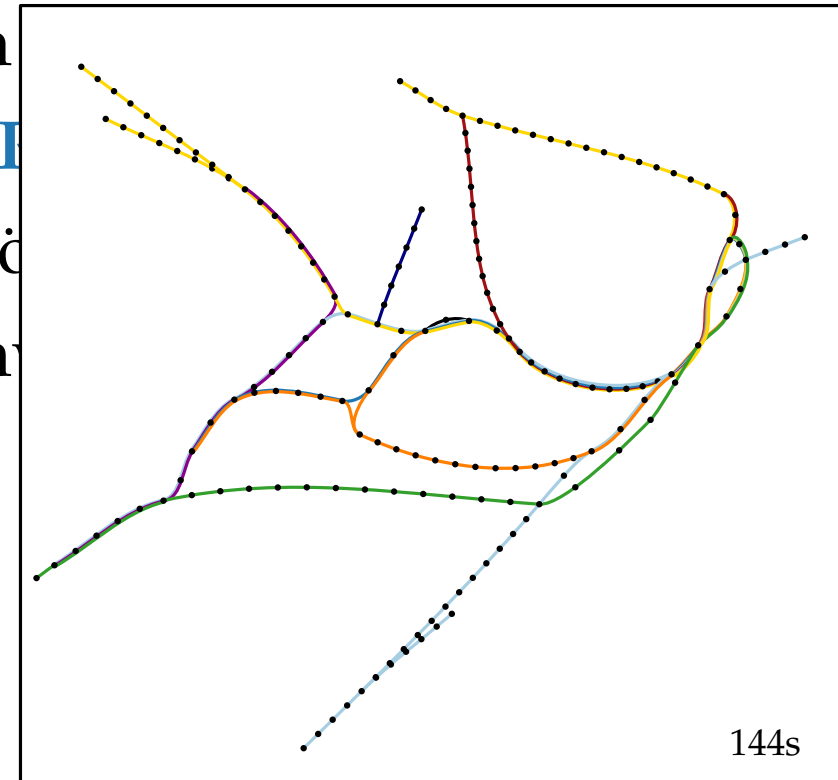
[Fink et al. 2013]

- Konvertiere (oktilineares) Eingabe-Layout in Bézier-Kurven
→ Knoten und Kontrollpunkte, beide unterliegen Kräften
- Teilkurven haben gemeinsame Tangenten im gemeinsamen Knoten **(R4)**
- (Standard) anziehende und abstoßende Kräfte **(R7)+(R8)**
- (schwache) Kraft in Richtung der ursprünglichen Position **(R6)**

- Kräfte
- Kurven
- Kräfte
- Nur to



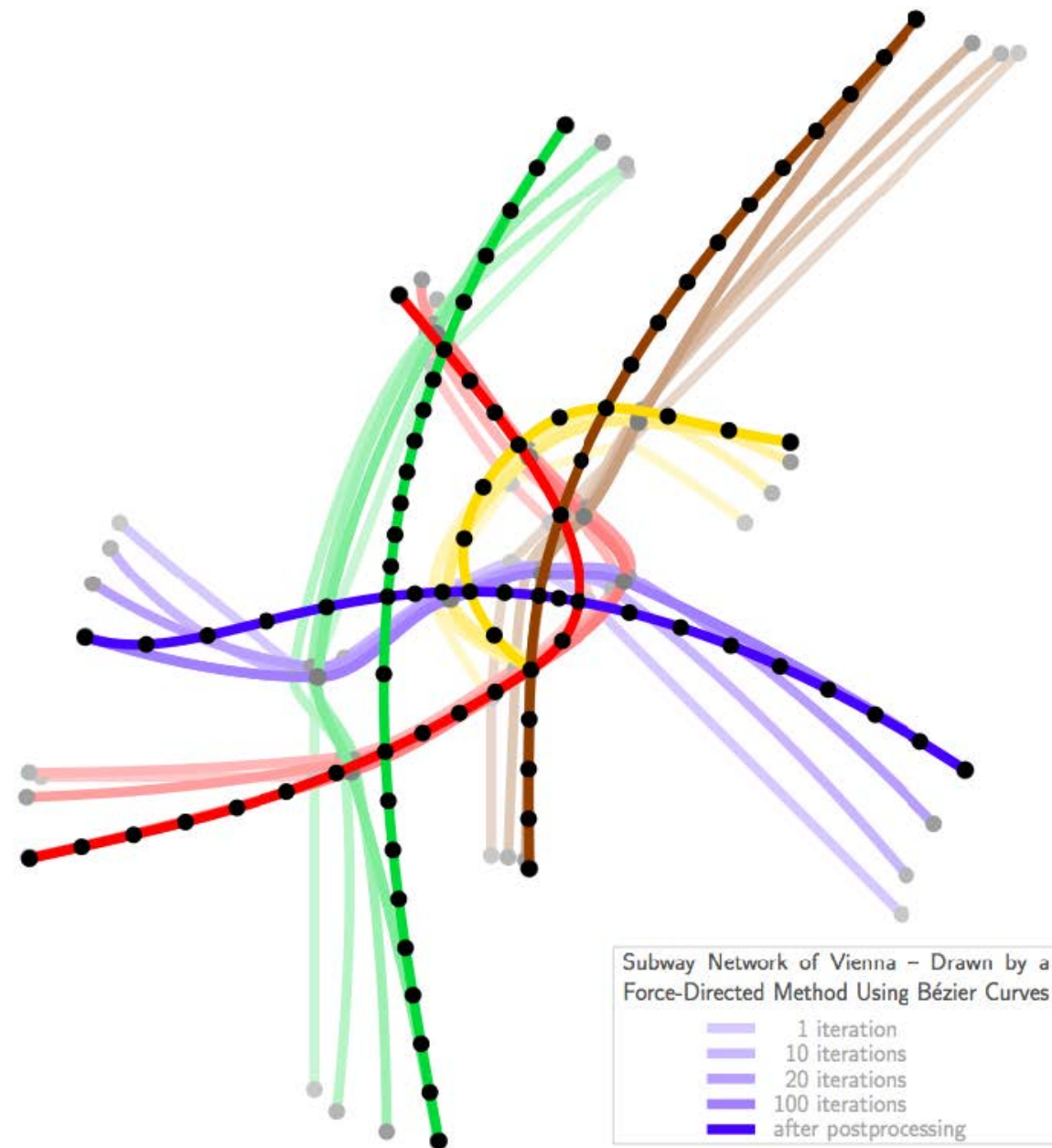
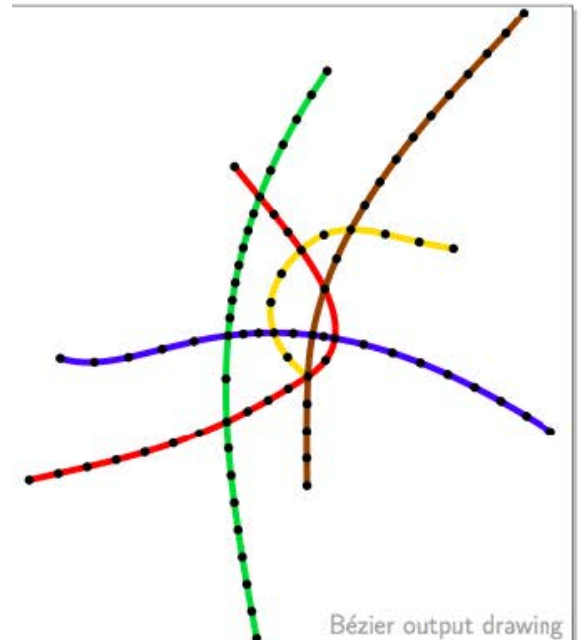
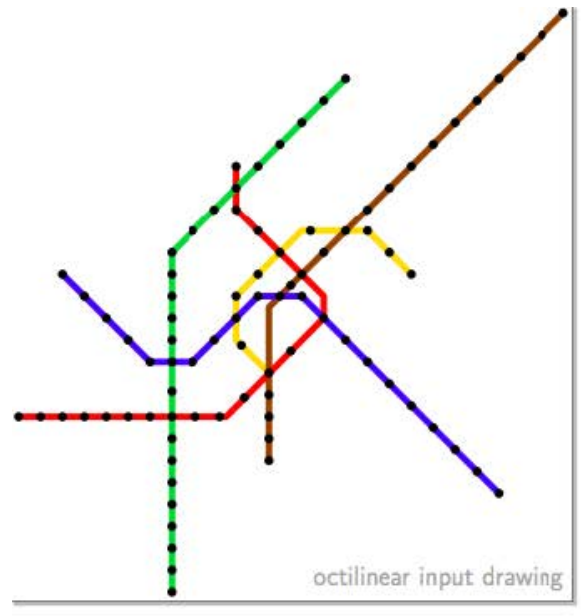
enform
möglich (I
elsauflö
ngen an



144s

Kräftebasierte Schematisierung – Bézier

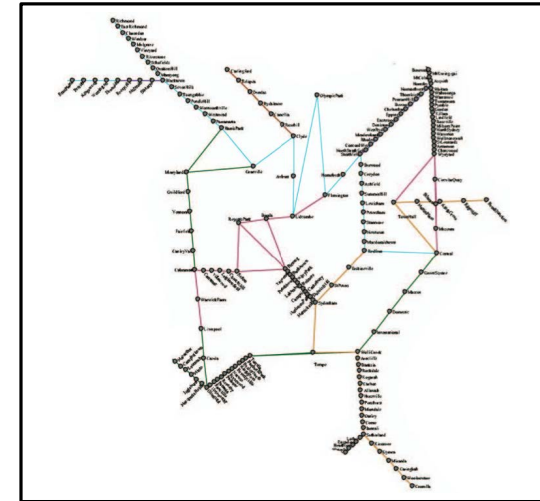
[Fink et al. 2013]



Kräftebasierte Schematisierung – Diskussion



Oktilinear.

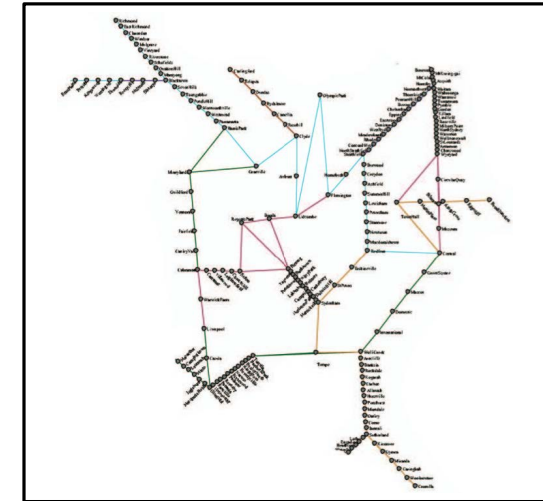


Kräftebasierte Schematisierung – Diskussion



Oktilinear.

- Garantiert Topologie (R1)

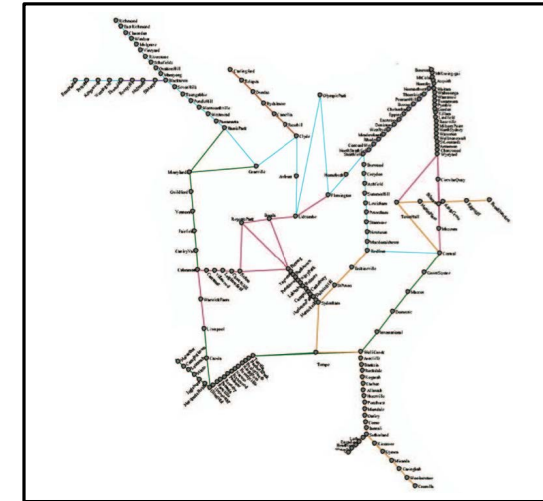


Kräftebasierte Schematisierung – Diskussion



Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug

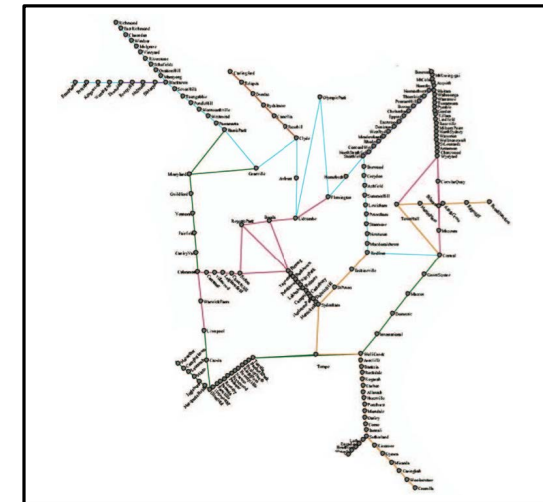


Kräftebasierte Schematisierung – Diskussion



Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität

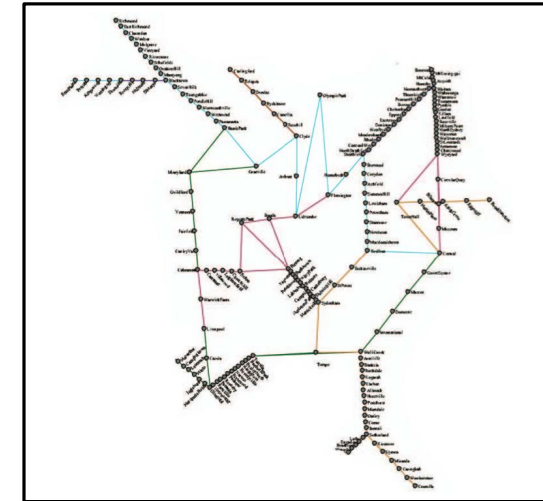


Kräftebasierte Schematisierung – Diskussion



Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen

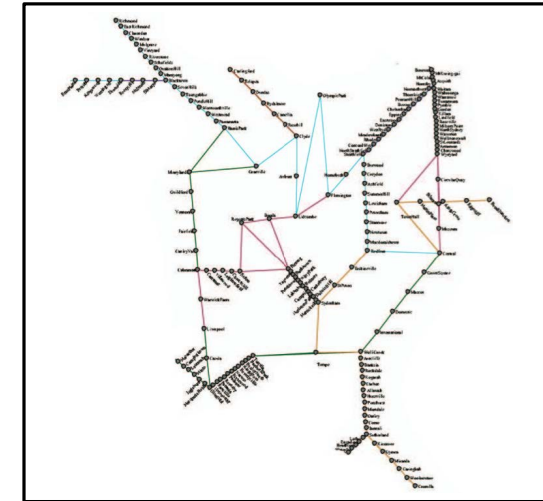


Kräftebasierte Schematisierung – Diskussion



Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen
- Knicke an Stationen

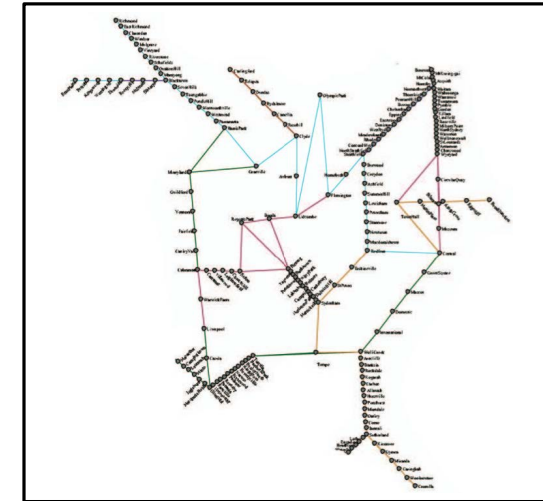


Kräftebasierte Schematisierung – Diskussion



Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen
- Knicke an Stationen
- Keine Beschränkung der Verzerrung



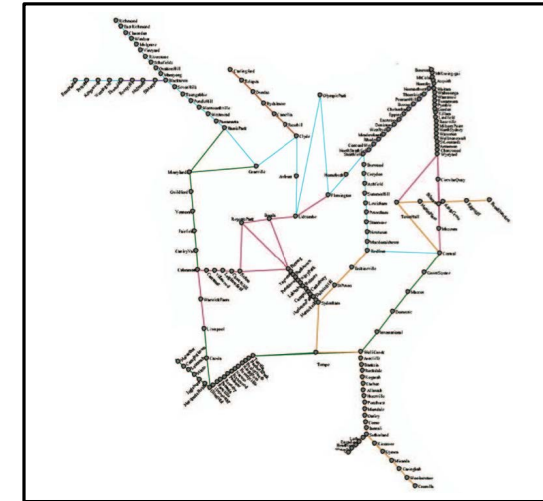
Kräftebasierte Schematisierung – Diskussion



Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen
- Knicke an Stationen
- Keine Beschränkung der Verzerrung

Bézier.

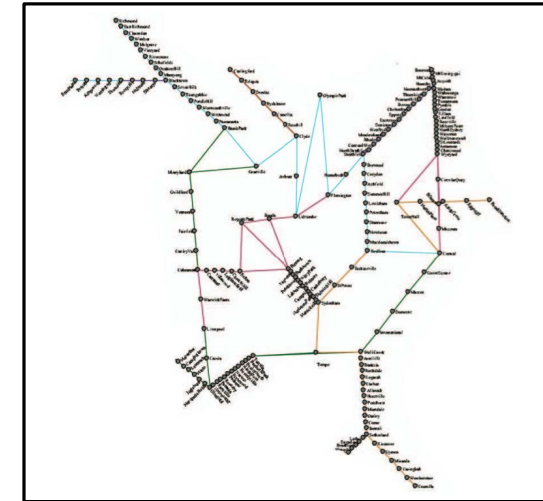


Kräftebasierte Schematisierung – Diskussion



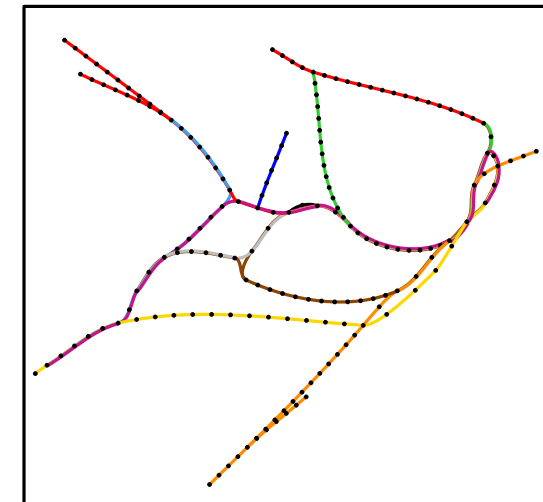
Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen
- Knicke an Stationen
- Keine Beschränkung der Verzerrung



Bézier.

- Garantiert Topologie (R1)

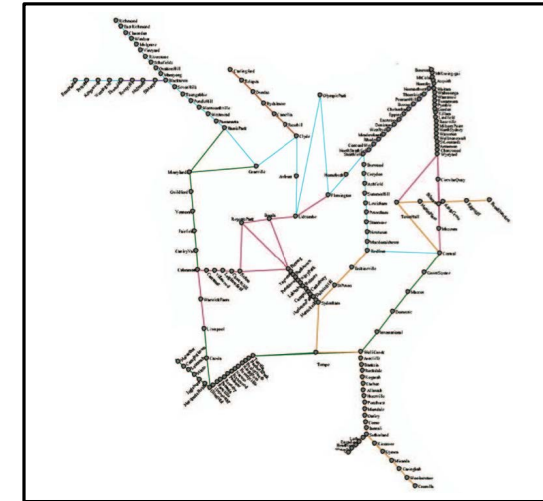


Kräftebasierte Schematisierung – Diskussion



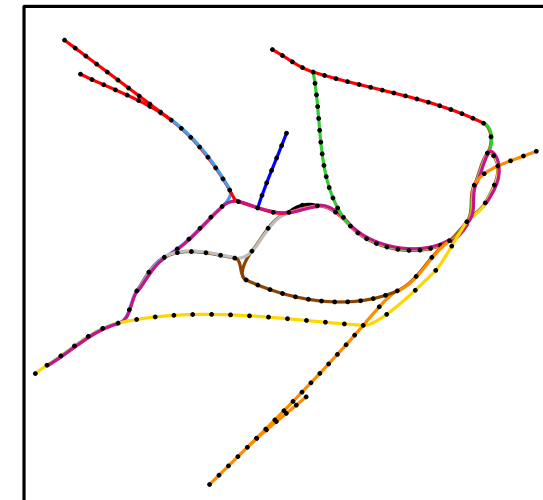
Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen
- Knicke an Stationen
- Keine Beschränkung der Verzerrung



Bézier.

- Garantiert Topologie (R1)
- Berücksichtigt fast alle Designregeln

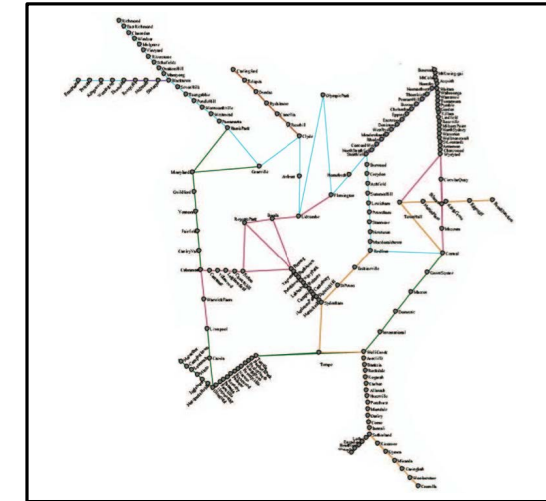


Kräftebasierte Schematisierung – Diskussion



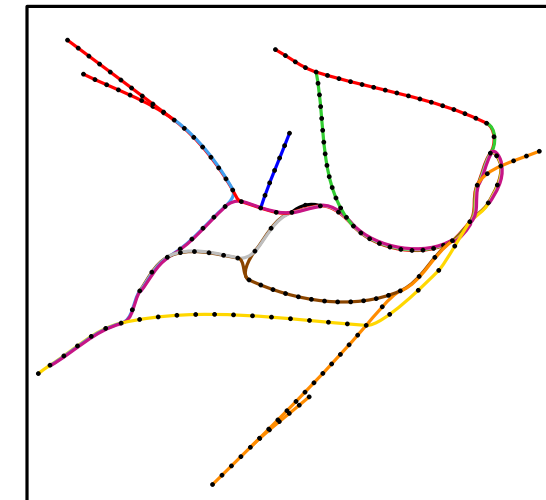
Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen
- Knicke an Stationen
- Keine Beschränkung der Verzerrung



Bézier.

- Garantiert Topologie (R1)
- Berücksichtigt fast alle Designregeln
- Erster kurviger U-Bahn-Netzplan-Algorithmus

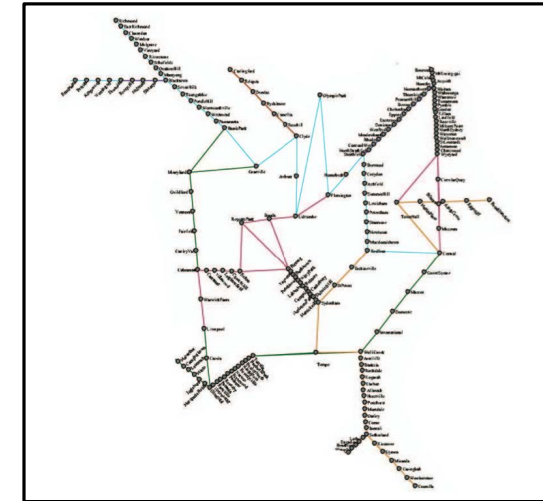


Kräftebasierte Schematisierung – Diskussion



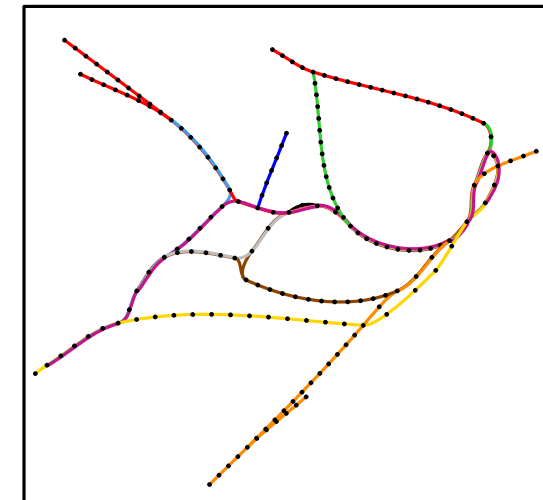
Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen
- Knicke an Stationen
- Keine Beschränkung der Verzerrung



Bézier.

- Garantiert Topologie (R1)
- Berücksichtigt fast alle Designregeln
- Erster kurviger U-Bahn-Netzplan-Algorithmus
- Funktioniert gut bei kleinen und mittleren Instanzen

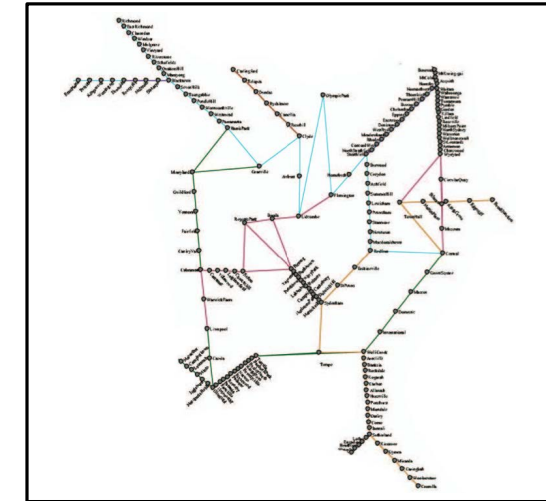


Kräftebasierte Schematisierung – Diskussion



Oktilinear.

- Garantiert Topologie (R1)
- Langsamer als pfadbasierten Algorithmen, aber immer noch schnell genug
- Keine strikte Oktilinearität
- Ziemlich unausgeglichene Kantenlängen
- Knicke an Stationen
- Keine Beschränkung der Verzerrung



Bézier.

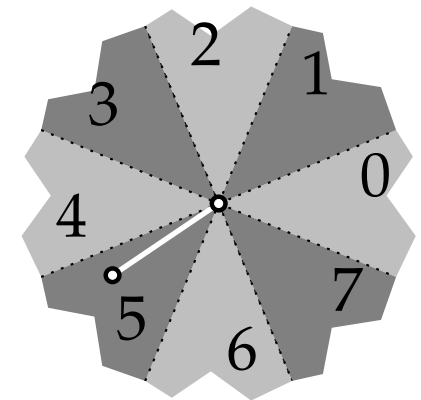
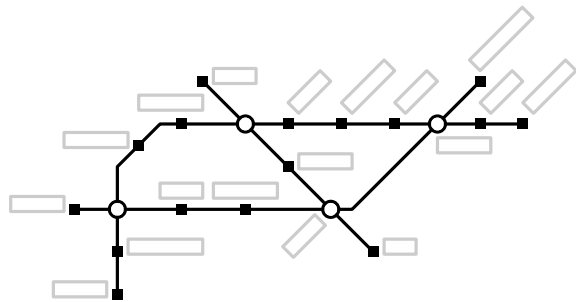
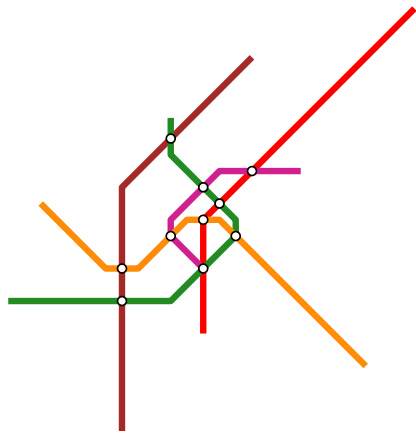
- Garantiert Topologie (R1)
- Berücksichtigt fast alle Designregeln
- Erster kurviger U-Bahn-Netzplan-Algorithmus
- Funktioniert gut bei kleinen und mittleren Instanzen
- Schwierigkeiten bei komplexeren Netzwerken



Algorithmen für geographische Informationssysteme

11. Vorlesung Netzpläne (Metro Maps)

Teil IV: Lokale Schematisierung



Lokale Schematisierung

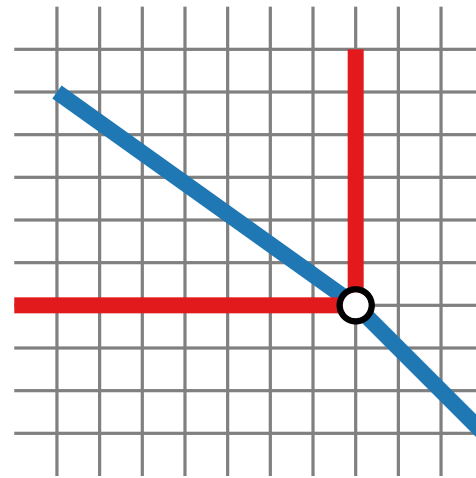
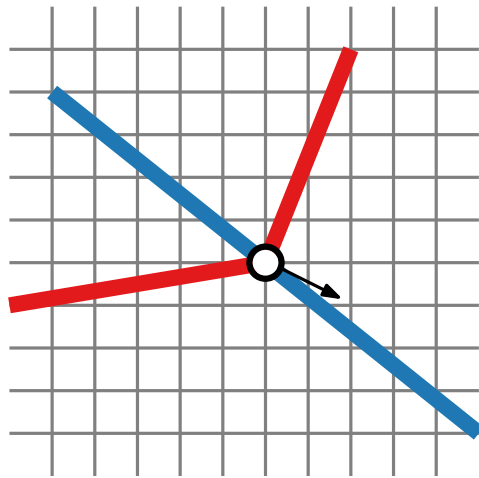


Idee. ■ definiere eine Qualitätsfunktion mit mehreren Kriterien



Lokale Schematisierung

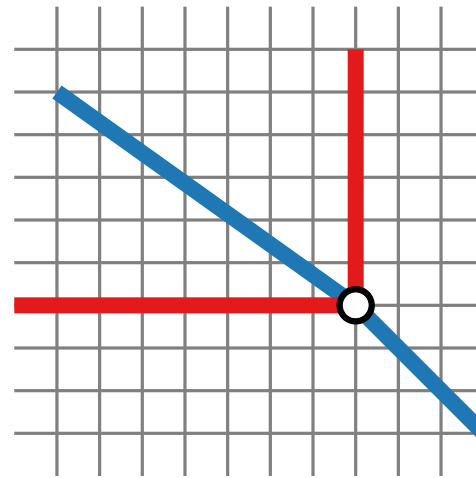
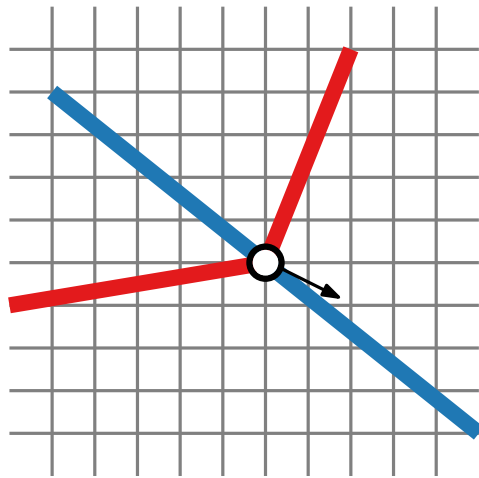
- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden





Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...





Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Avelar & Müller 2000]

- Berechne die beste Knotenposition für jedes Kriterium (Oktilinearität **(R2)**, minimale Trennung **(R8)**)



Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Avelar & Müller 2000]

- Berechne die beste Knotenposition für jedes Kriterium (Oktilinearität **(R2)**, minimale Trennung **(R8)**)
- Bewege den Knoten zum Durchschnitt der Positionen, ohne die Topologie zu verletzen **(R1)**



Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Avelar & Müller 2000]

- Berechne die beste Knotenposition für jedes Kriterium (Oktilinearität **(R2)**, minimale Trennung **(R8)**)
- Bewege den Knoten zum Durchschnitt der Positionen, ohne die Topologie zu verletzen **(R1)**

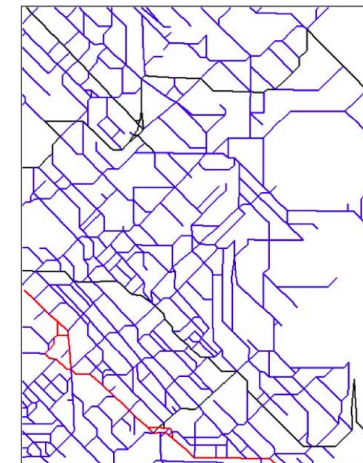


Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Avelar & Müller 2000]

- Berechne die beste Knotenposition für jedes Kriterium (Oktilinearität **(R2)**, minimale Trennung **(R8)**)
- Bewege den Knoten zum Durchschnitt der Positionen, ohne die Topologie zu verletzen **(R1)**

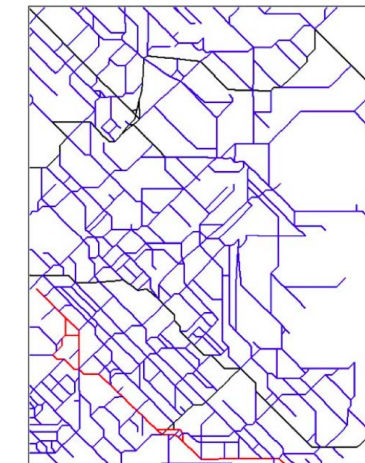
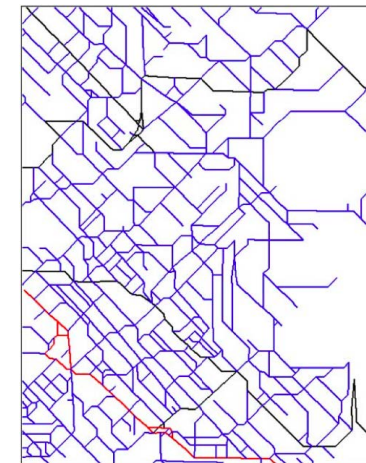


Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Avelar & Müller 2000]

- Berechne die beste Knotenposition für jedes Kriterium (Oktilinearität **(R2)**, minimale Trennung **(R8)**)
- Bewege den Knoten zum Durchschnitt der Positionen, ohne die Topologie zu verletzen **(R1)**

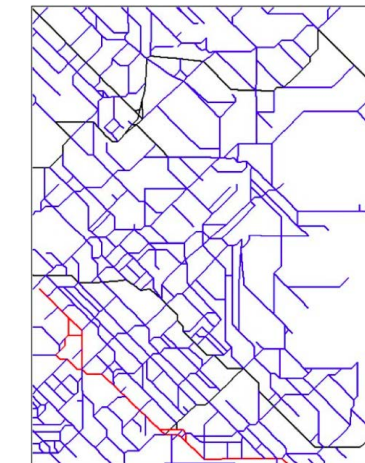
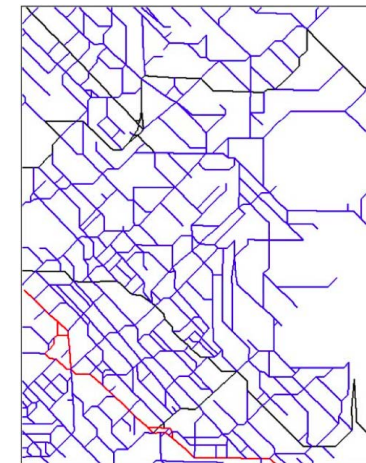


Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Avelar & Müller 2000]

- Berechne die beste Knotenposition für jedes Kriterium (Oktilinearität **(R2)**, minimale Trennung **(R8)**)
- Bewege den Knoten zum Durchschnitt der Positionen, ohne die Topologie zu verletzen **(R1)**



100

1,000

10,000

Lokale Schematisierung



- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Ware et al. 2006, Ware & Richards 2013]

- gewichtete Multikriterienfunktion



Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Ware et al. 2006, Ware & Richards 2013]

- gewichtete Multikriterienfunktion
- kontrahiere Grad-2-Knoten vor der Optimierung



Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Ware et al. 2006, Ware & Richards 2013]

- gewichtete Multikriterienfunktion
- kontrahiere Grad-2-Knoten vor der Optimierung
- mehr Designregeln implementiert (Topologie **(R1)**, Oktilinearität **(R2)**, Verschiebung **(R6)**, Kantenlängen **(R7)**, Trennung **(R8)**)



Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu besseren benachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Ware et al. 2006, Ware & Richards 2013]

- gewichtete Multikriterienfunktion
- kontrahiere Grad-2-Knoten vor der Optimierung
- mehr Designregeln implementiert (Topologie **(R1)**, Oktilinearität **(R2)**, Verschiebung **(R6)**, Kantenlängen **(R7)**, Trennung **(R8)**)
- Simulated Annealing (2006) und Ameisenkolonie-Optimierung (2013)

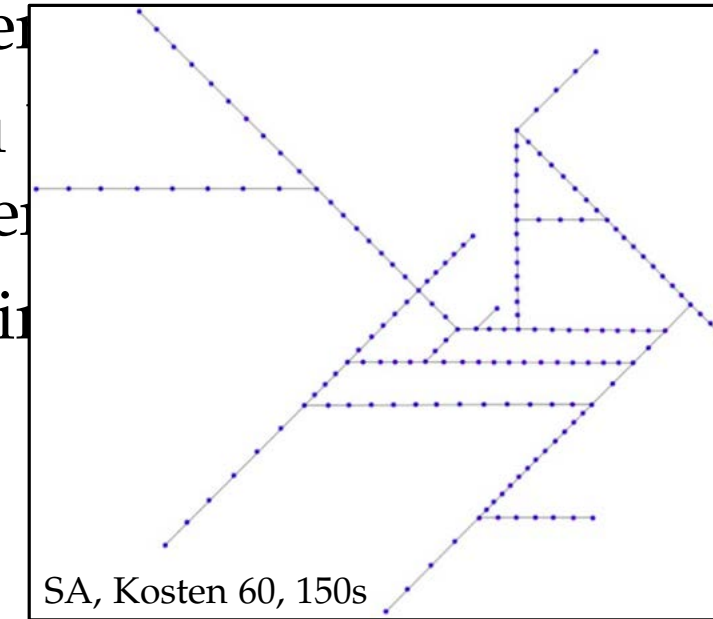


Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu nachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Ware et al. 2006, Ware & Richards 2013]

- gewichtete Multikriterienfunktion
- kontrahiere Grad-2-Knoten vor der Optimierung
- mehr Designregeln implementiert (Topologie **(R1)**, Oktilinearität **(R2)**, Verschiebung **(R6)**, Kantenlängen **(R7)**, Trennung **(R8)**)
- Simulated Annealing (2006) und Ameisenkolonie-Optimierung (2013)



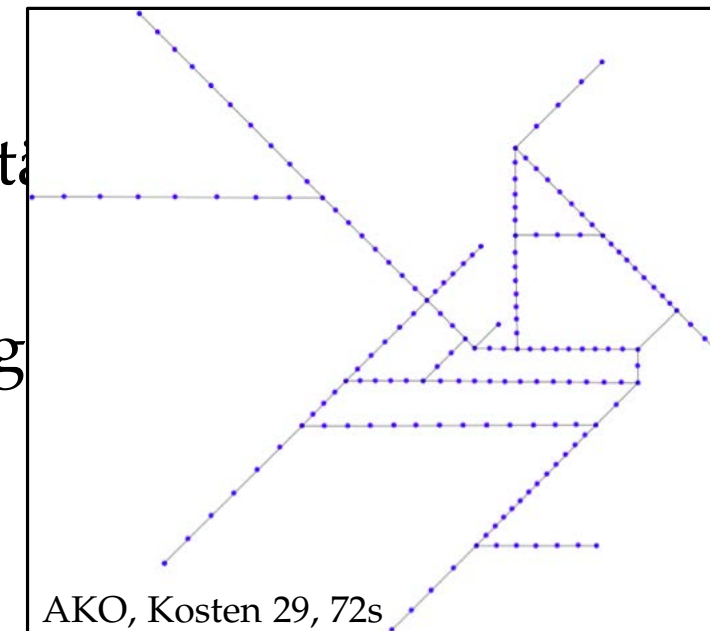
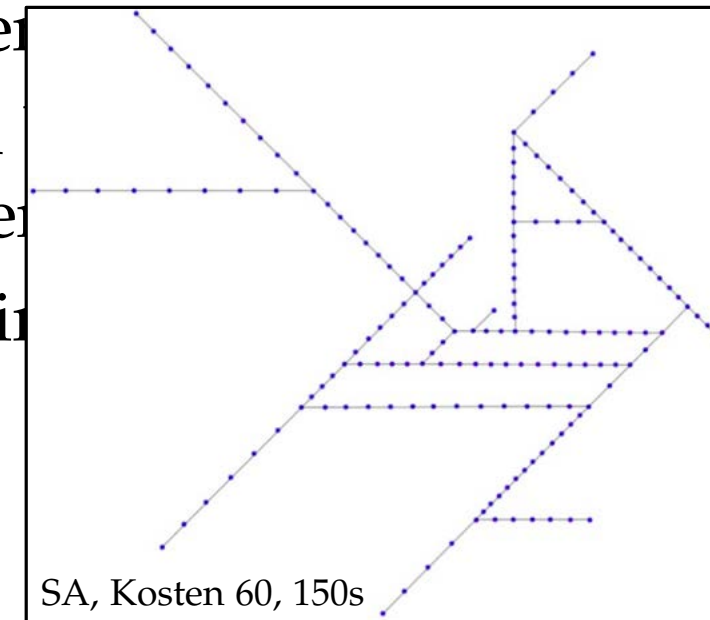


Lokale Schematisierung

- Idee.**
- definiere eine Qualitätsfunktion mit mehreren Kriterien
 - verbessere die Qualität schrittweise, indem Knoten zu nachbarten (Gitter-)Positionen lokal verschoben werden
 - anwendbare Optimierungstechniken: Hill Climbing, Simulated Annealing, Ameisenkolonie-Optimierung, ...

[Ware et al. 2006, Ware & Richards 2013]

- gewichtete Multikriterienfunktion
- kontrahiere Grad-2-Knoten vor der Optimierung
- mehr Designregeln implementiert (Topologie **(R1)**, Oktilinearität **(R2)**, Verschiebung **(R6)**, Kantenlängen **(R7)**, Trennung **(R8)**)
- Simulated Annealing (2006) und Ameisenkolonie-Optimierung



Lokale Schematisierung

[Stott et al. 2011]

Idee. ■ Designregeln wie zuvor



Lokale Schematisierung

[Stott et al. 2011]

Idee.

- Designregeln wie zuvor
- zusätzliche Kriterien für U-Bahn-Netzpläne einbeziehen (Knickminimierung **(R3)**, gerade durch Stationen **(R4)**, Winkelauflösung **(R5)**, relative Positionen **(R6)**)



Lokale Schematisierung

[Stott et al. 2011]

- Idee.**
- Designregeln wie zuvor
 - zusätzliche Kriterien für U-Bahn-Netzpläne einbeziehen (Knickminimierung **(R3)**, gerade durch Stationen **(R4)**, Winkelsauflösung **(R5)**, relative Positionen **(R6)**)
 - abwechselnd Netzwerk zeichnen und Beschriftungen platzieren



Lokale Schematisierung

[Stott et al. 2011]

Idee.

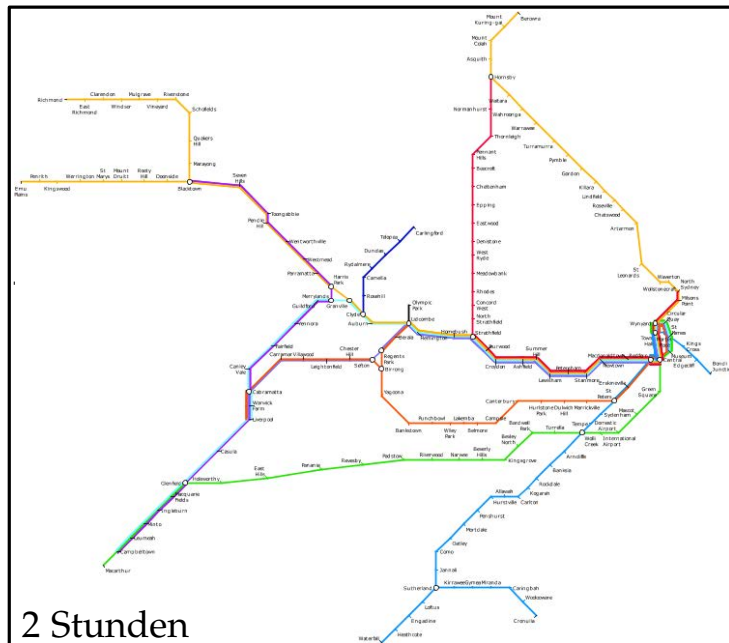
- Designregeln wie zuvor
- zusätzliche Kriterien für U-Bahn-Netzpläne einbeziehen (Knickminimierung **(R3)**, gerade durch Stationen **(R4)**, Winkelsauflösung **(R5)**, relative Positionen **(R6)**)
- abwechselnd Netzwerk zeichnen und Beschriftungen platzieren
- einige Ad-hoc-Korrekturen für Situationen mit lokalen Minima



Lokale Schematisierung

[Stott et al. 2011]

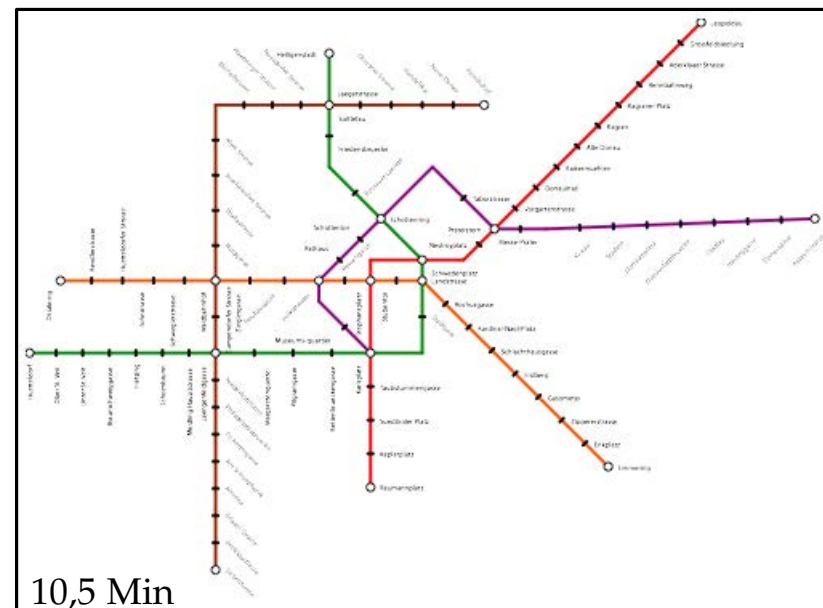
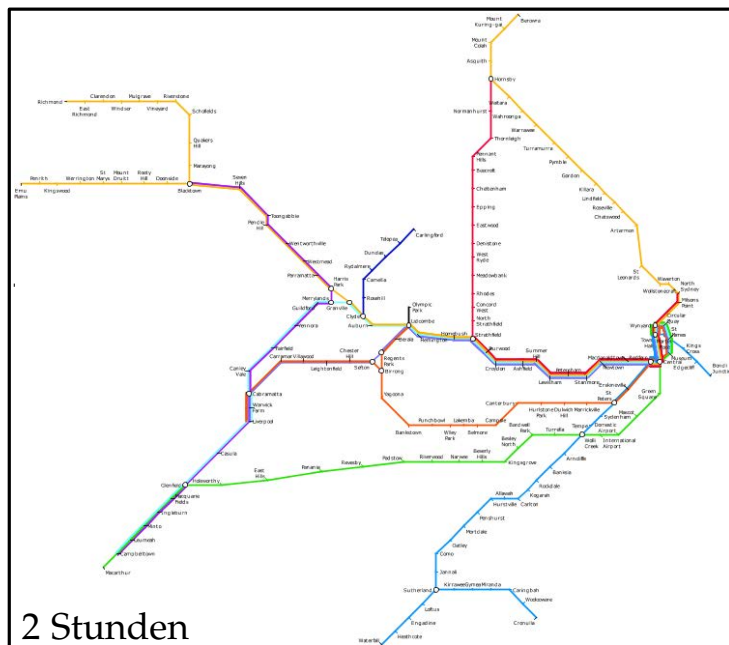
- Idee.**
- Designregeln wie zuvor
 - zusätzliche Kriterien für U-Bahn-Netzpläne einbeziehen (Knickminimierung **(R3)**, gerade durch Stationen **(R4)**, Winkelauflösung **(R5)**, relative Positionen **(R6)**)
 - abwechselnd Netzwerk zeichnen und Beschriftungen platzieren
 - einige Ad-hoc-Korrekturen für Situationen mit lokalen Minima



Lokale Schematisierung

[Stott et al. 2011]

- Idee.**
- Designregeln wie zuvor
 - zusätzliche Kriterien für U-Bahn-Netzpläne einbeziehen (Knickminimierung **(R3)**, gerade durch Stationen **(R4)**, Winkelauflösung **(R5)**, relative Positionen **(R6)**)
 - abwechselnd Netzwerk zeichnen und Beschriftungen platzieren
 - einige Ad-hoc-Korrekturen für Situationen mit lokalen Minima



Lokale Schematisierung – Diskussion



Vorteile.

Nachteile.

Lokale Schematisierung – Diskussion



Vorteile.

- flexibles Framework, einfach neue Kriterien zu integrieren

Nachteile.

Lokale Schematisierung – Diskussion



Vorteile.

- flexibles Framework, einfach neue Kriterien zu integrieren
- neuere Methoden verbesserten die visuelle Layout Qualität

Nachteile.

Lokale Schematisierung – Diskussion



Vorteile.

- flexibles Framework, einfach neue Kriterien zu integrieren
- neuere Methoden verbesserten die visuelle Layout Qualität
- Integration von Layout und Beschriftung

Nachteile.

Lokale Schematisierung – Diskussion



Vorteile.

- flexibles Framework, einfach neue Kriterien zu integrieren
- neuere Methoden verbesserten die visuelle Layout Qualität
- Integration von Layout und Beschriftung

Nachteile.

- Optimierung der Kriterien, aber keine Garantien (außer der Topologie)



Lokale Schematisierung – Diskussion

Vorteile.

- flexibles Framework, einfach neue Kriterien zu integrieren
- neuere Methoden verbesserten die visuelle Layout Qualität
- Integration von Layout und Beschriftung

Nachteile.

- Optimierung der Kriterien, aber keine Garantien (außer der Topologie)
- anfällig für lokale Minima



Lokale Schematisierung – Diskussion

Vorteile.

- flexibles Framework, einfach neue Kriterien zu integrieren
- neuere Methoden verbesserten die visuelle Layout Qualität
- Integration von Layout und Beschriftung

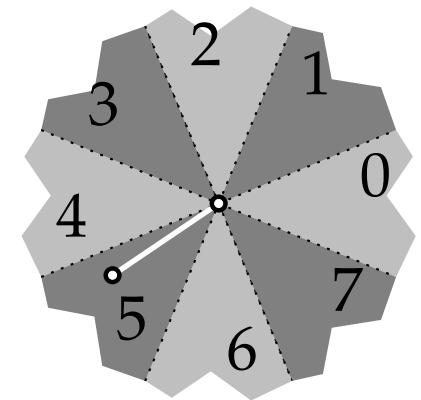
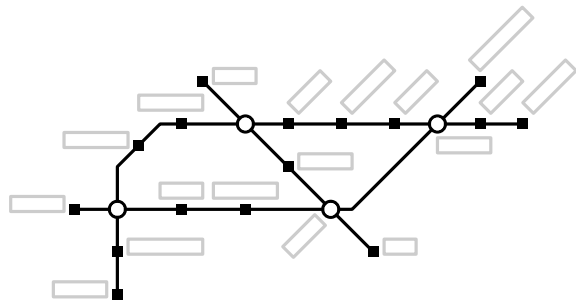
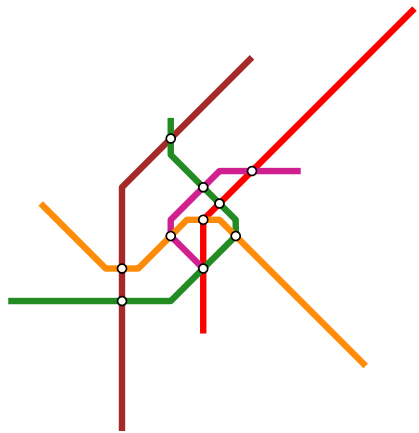
Nachteile.

- Optimierung der Kriterien, aber keine Garantien (außer der Topologie)
- anfällig für lokale Minima
- lange Laufzeiten

Algorithmen für geographische Informationssysteme

11. Vorlesung Netzpläne (Metro Maps)

Teil V: Mixed-Integer Programm



Mixed-Integer Programm

[Nöllenburg & Wolff 2011]



- finde exakte optimale Lösung mittels kombinatorischer Optimierung

Mixed-Integer Programm

[Nöllenburg & Wolff 2011]



- finde exakte optimale Lösung mittels kombinatorischer Optimierung
- teile Designregeln in **harte** und **weiche** Beschränkungen

Mixed-Integer Programm

[Nöllenburg & Wolff 2011]



- finde exakte optimale Lösung mittels kombinatorischer Optimierung
- teile Designregeln in **harte** und **weiche** Beschränkungen
- modelliere Einschränkungen als lineare (Un)Gleichungen und lineare Zielfunktion
→ Mixed-Integer Programming (**MIP**)

Mixed-Integer Programm

[Nöllenburg & Wolff 2011]



- finde exakte optimale Lösung mittels kombinatorischer Optimierung
- teile Designregeln in **harte** und **weiche** Beschränkungen
- modelliere Einschränkungen als lineare (Un)Gleichungen und lineare Zielfunktion
→ Mixed-Integer Programming (**MIP**)
- integriere überlappungsfreie Stationenbeschriftung in das Modell

Mixed-Integer Programm

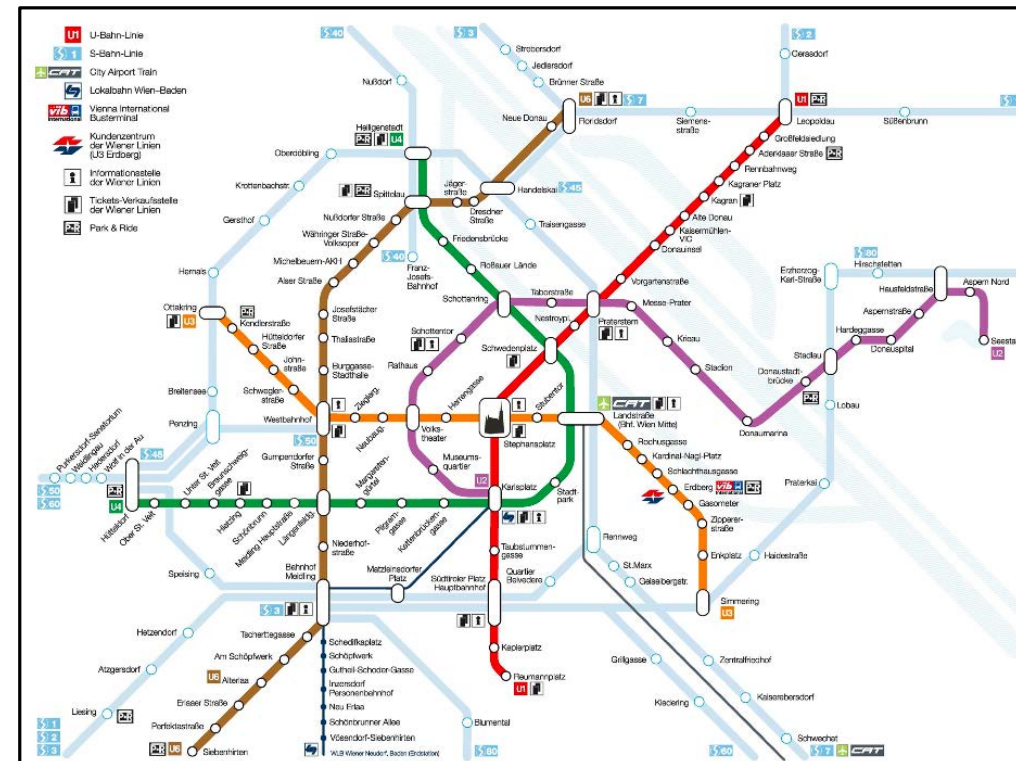
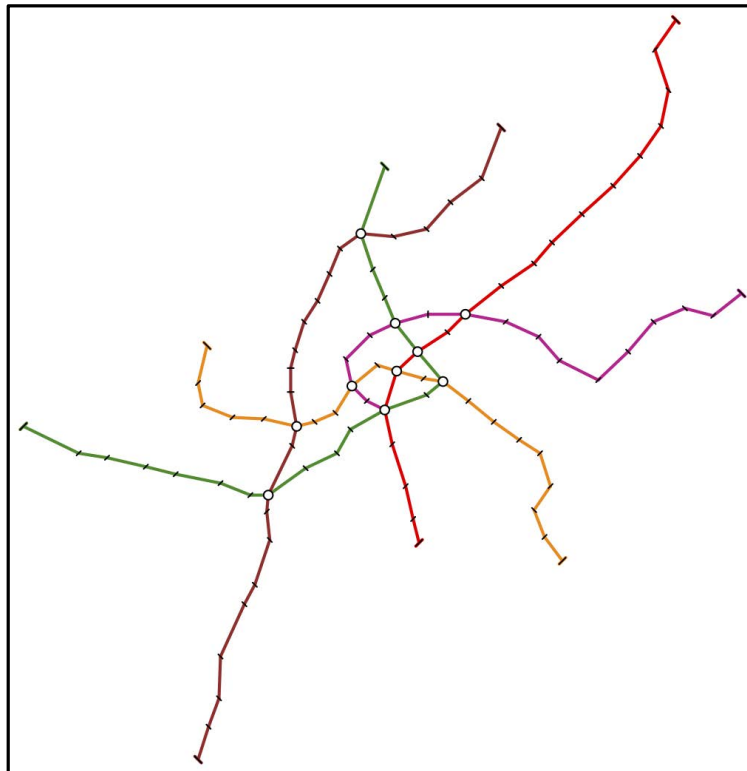
[Nöllenburg & Wolff 2011]



- finde exakte optimale Lösung mittels kombinatorischer Optimierung
- teile Designregeln in **harte** und **weiche** Beschränkungen
- modelliere Einschränkungen als lineare (Un)Gleichungen und lineare Zielfunktion
→ Mixed-Integer Programming (**MIP**)
- integriere überlappungsfreie Stationenbeschriftung in das Modell
- kann ausgeklügelte Optimierungstools als Solver nutzen (z.B. CPLEX, Gurobi)

Harte Beschränkungen

(R1) Erhalte Einbettung/**Topologie** und **Planarität**

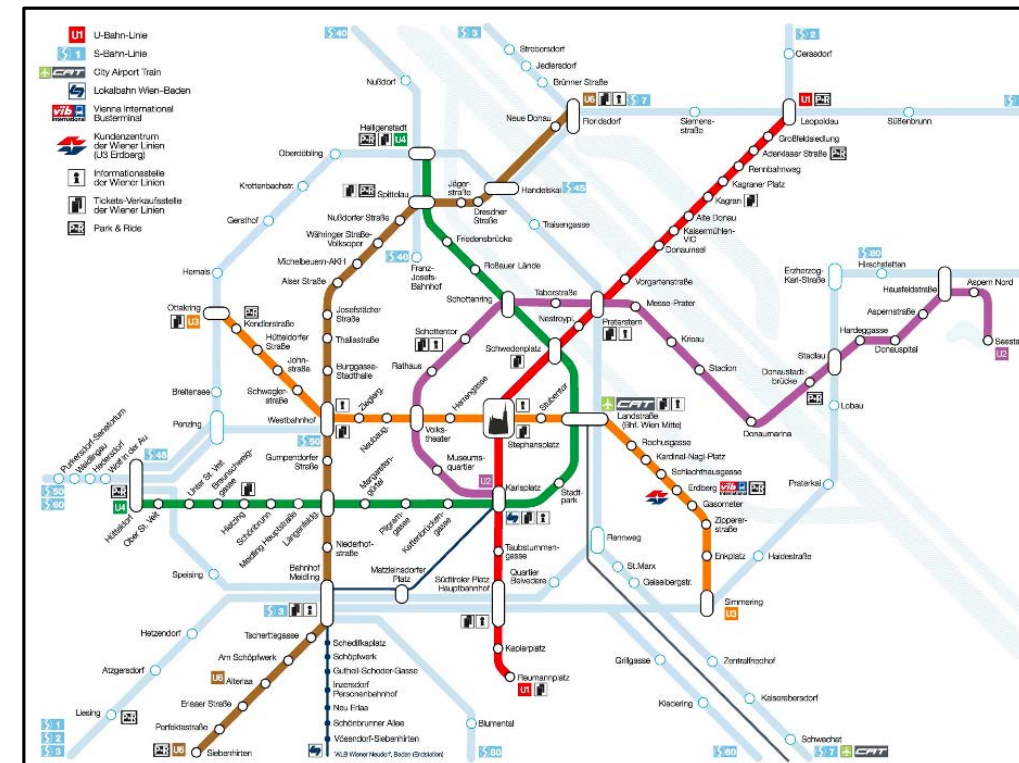
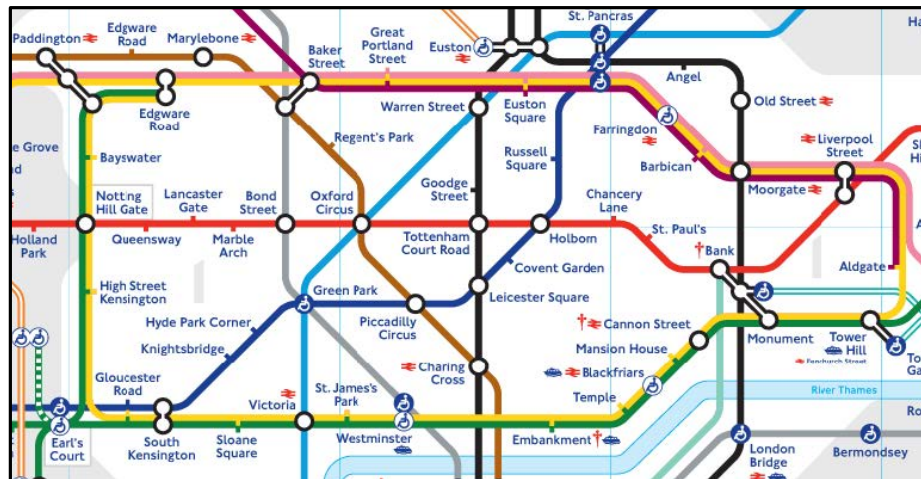
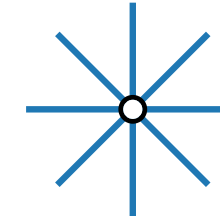


Harte Beschränkungen



(R1) Erhalte Einbettung/**Topologie** und **Planarität**

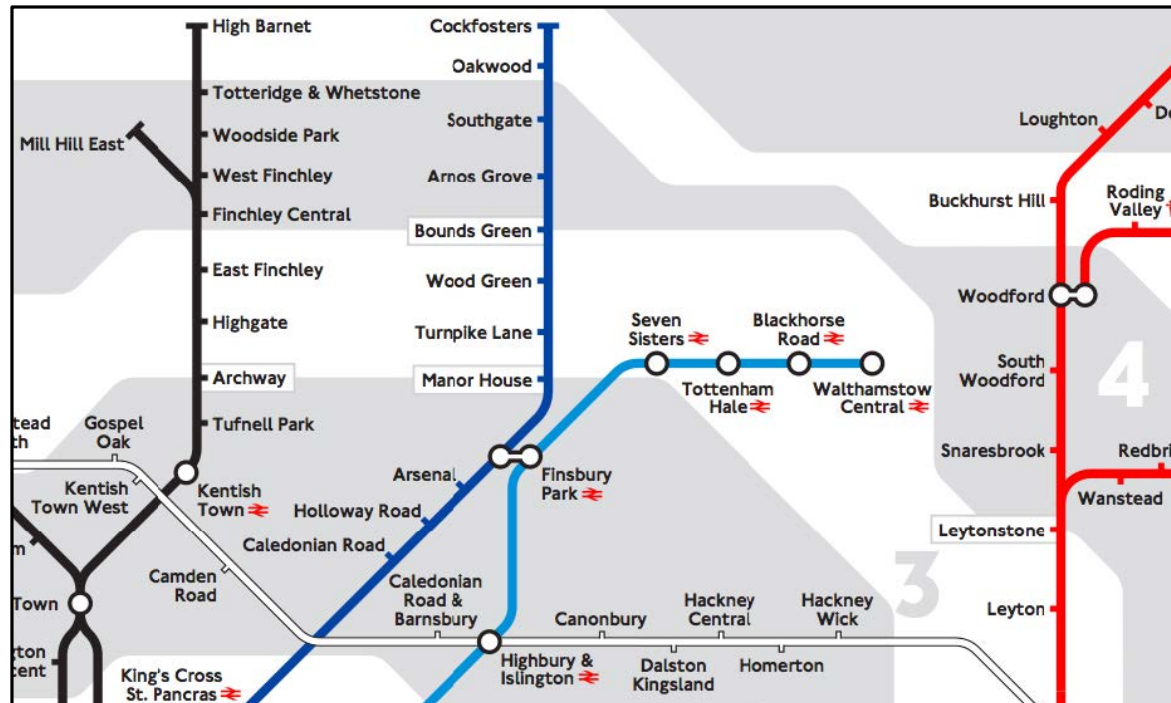
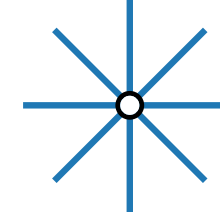
(R2) Zeichne alle Kanten **oktilinear**



Harte Beschränkungen



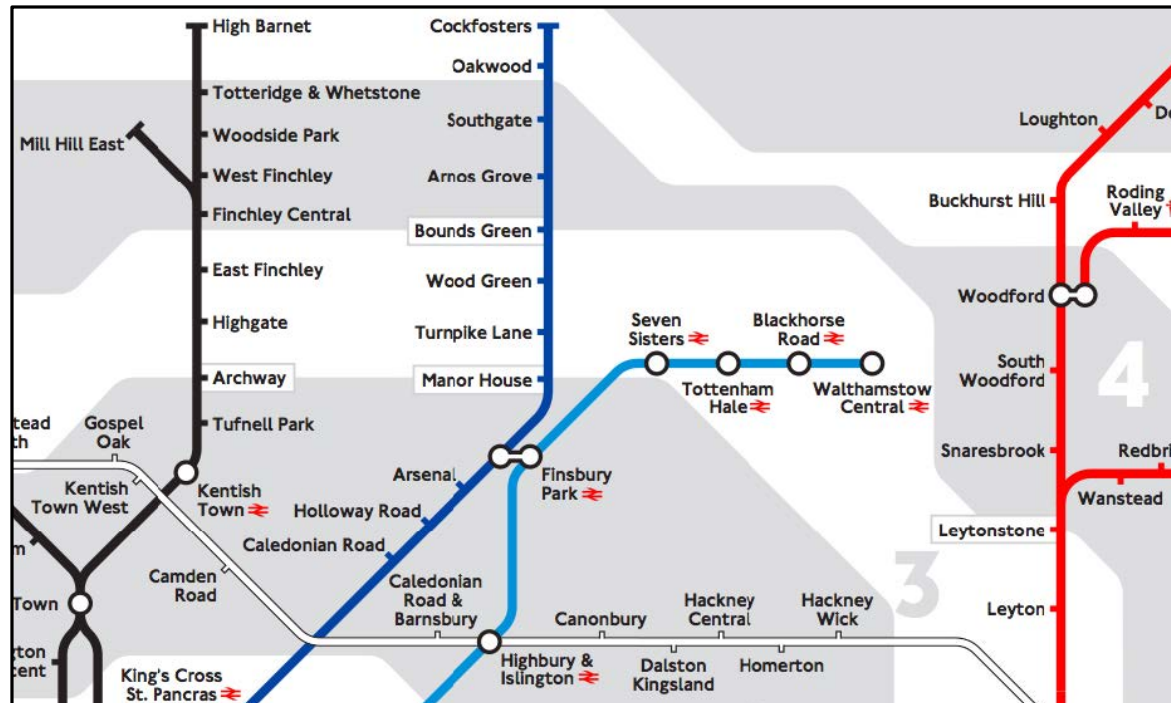
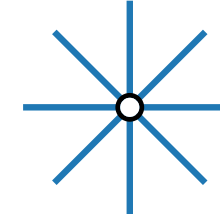
- (R1) Erhalte Einbettung/**Topologie** und **Planarität**
- (R2) Zeichne alle Kanten **oktilinear**
- (R7) Erzwingende **minimale Kantenlänge** $\ell_{\min}$



Harte Beschränkungen



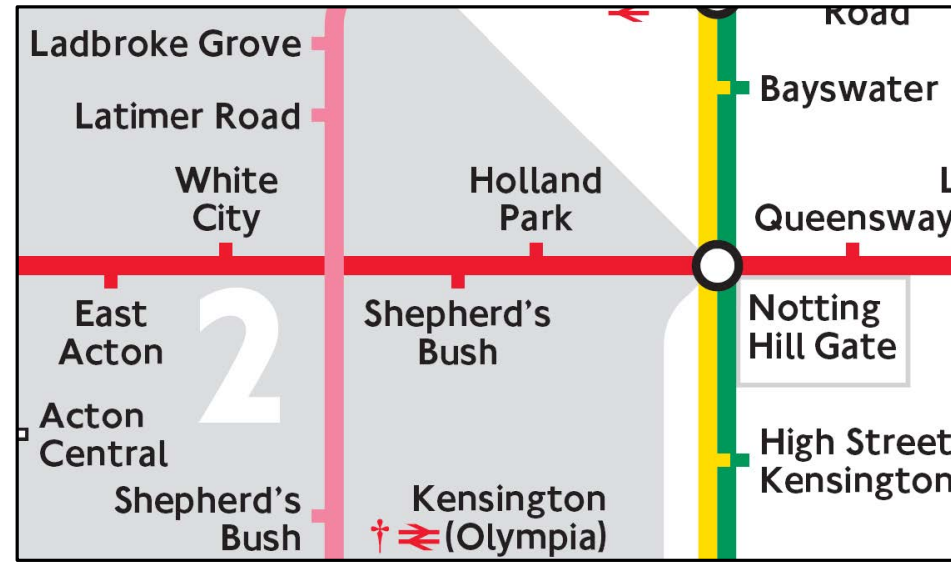
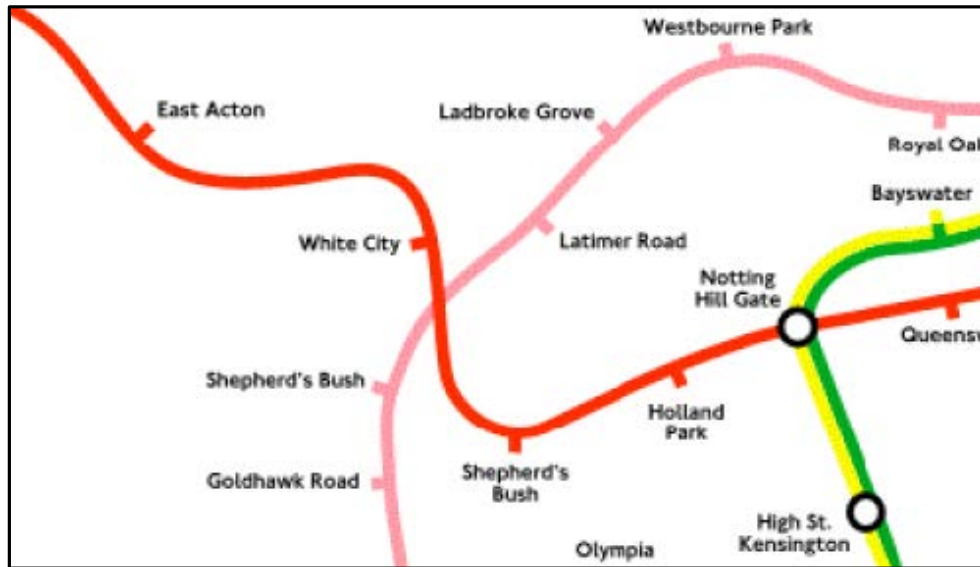
- (R1) Erhalte Einbettung/**Topologie** und **Planarität**
- (R2) Zeichne alle Kanten **oktilinear**
- (R7) Erzwingende **minimale Kantenlänge** $\ell_{\min}$
- (R8) Erzwingende **minimale Abstände** $d_{\min}$



Weiche Beschränkungen



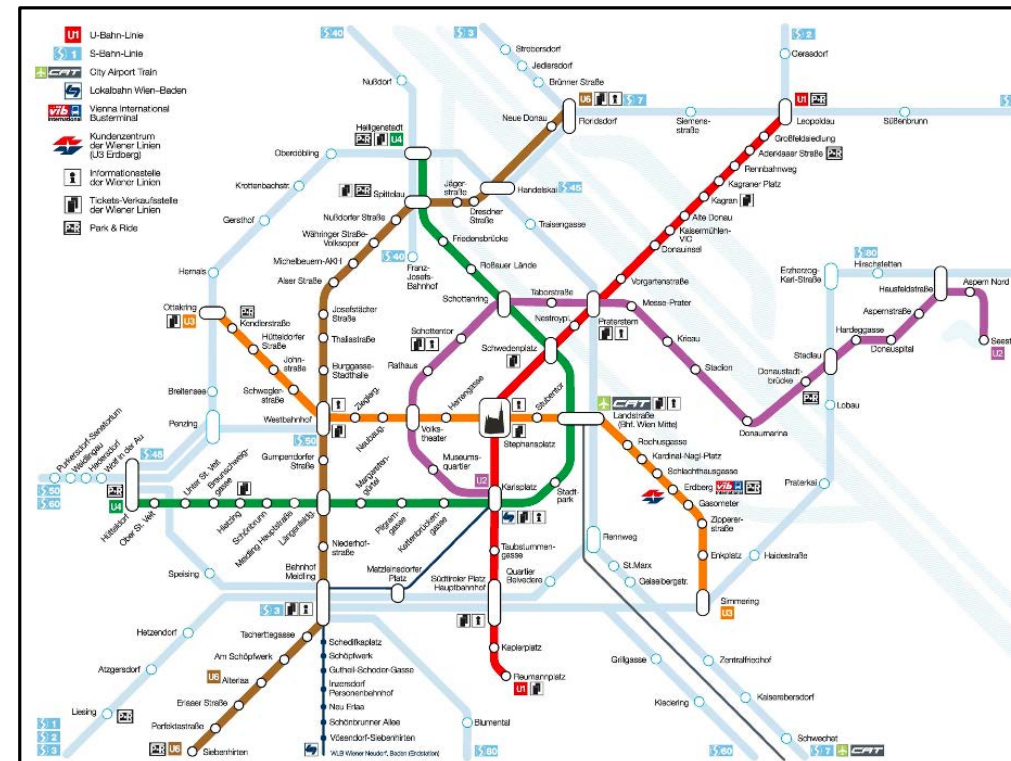
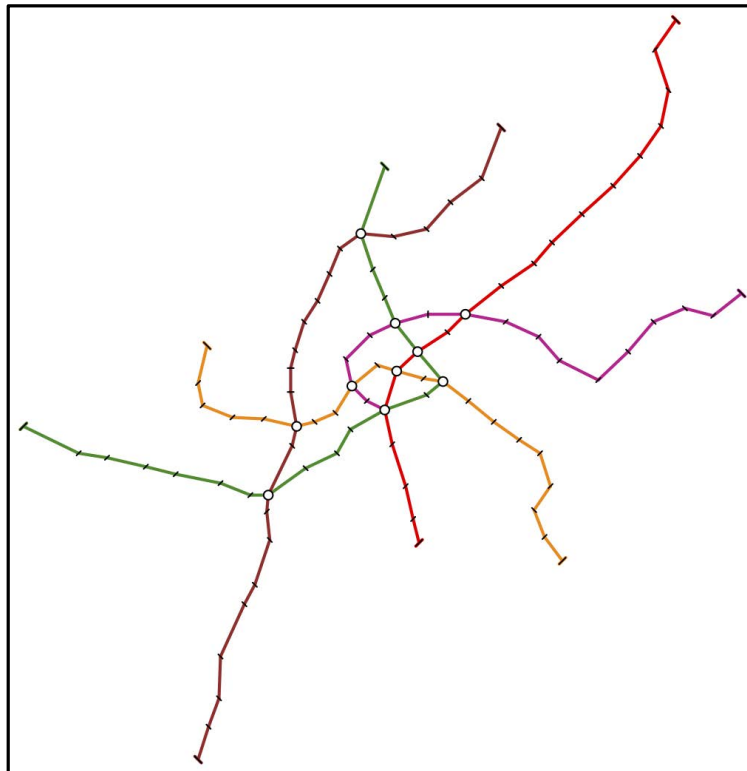
(R3+R4) Zeichne Linienzüge in $\mathcal{L}$ mit **wenig Knicken**



Weiche Beschränkungen

(R3+R4) Zeichne Linienzüge in $\mathcal{L}$ mit **wenig Knicken**

(R6) minimiere geometrische **Verzerrung**

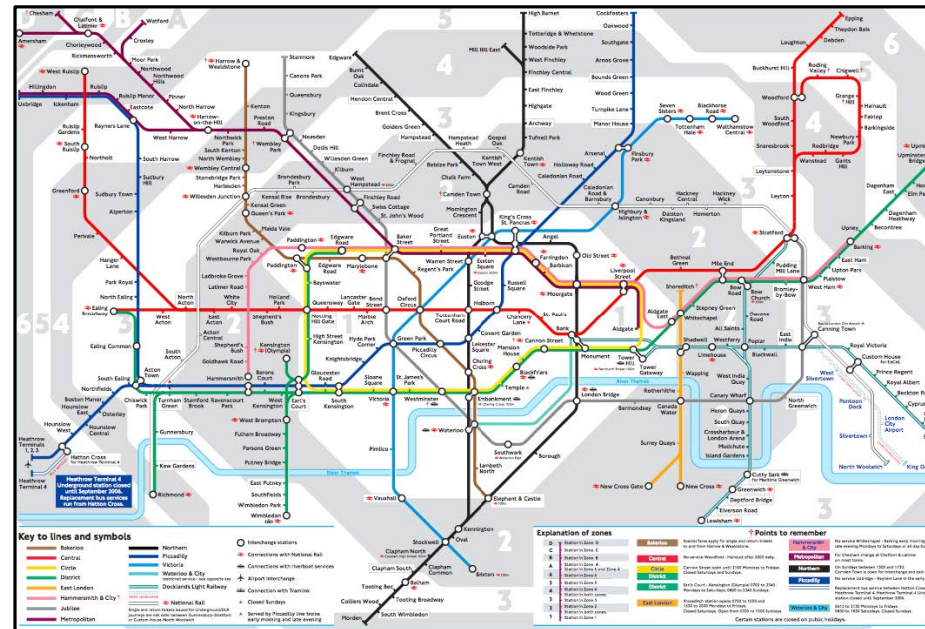
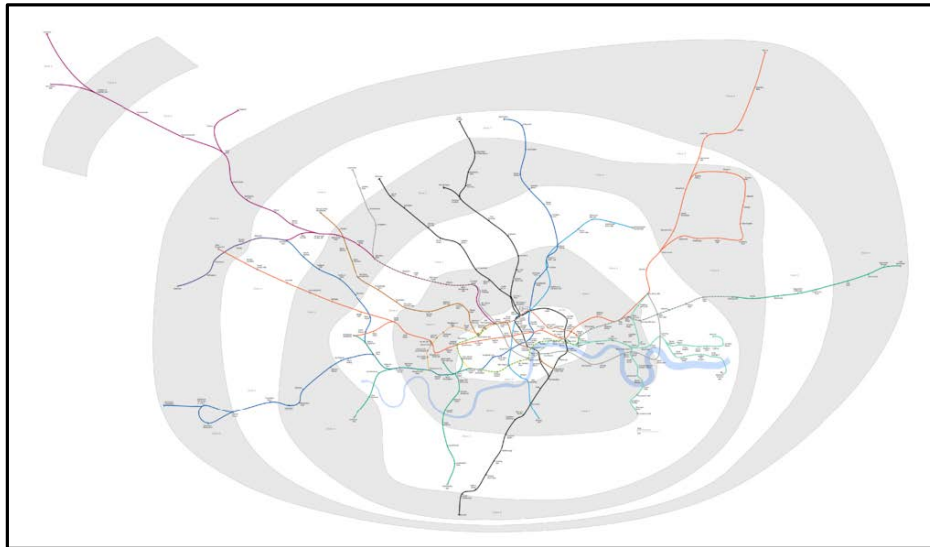


Weiche Beschränkungen

(R3+R4) Zeichne Linienzüge in $\mathcal{L}$ mit **wenig Knicken**

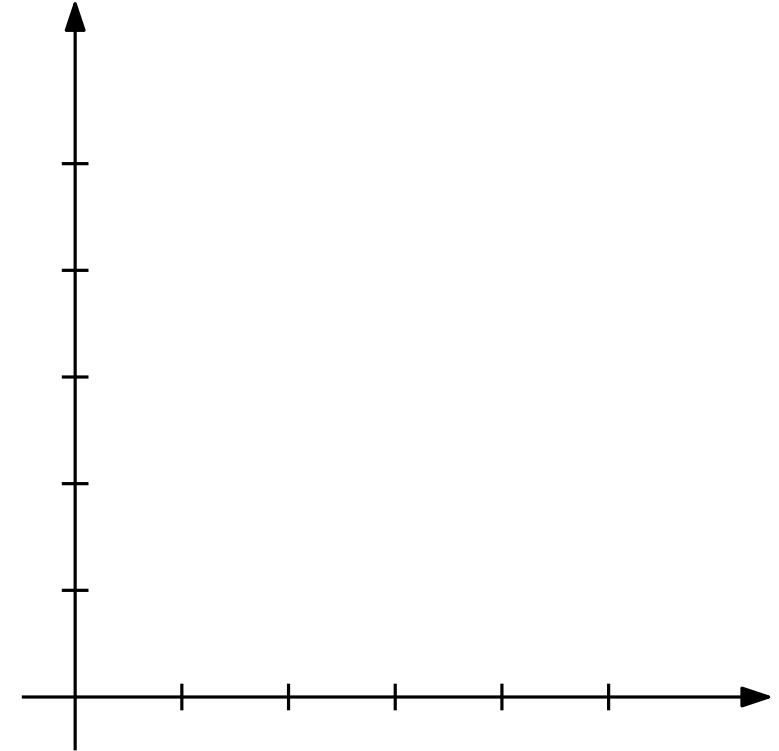
(R6) minimiere geometrische **Verzerrung**

(R7) minimiere gesamte **Kantenlänge**



Lineare Programm

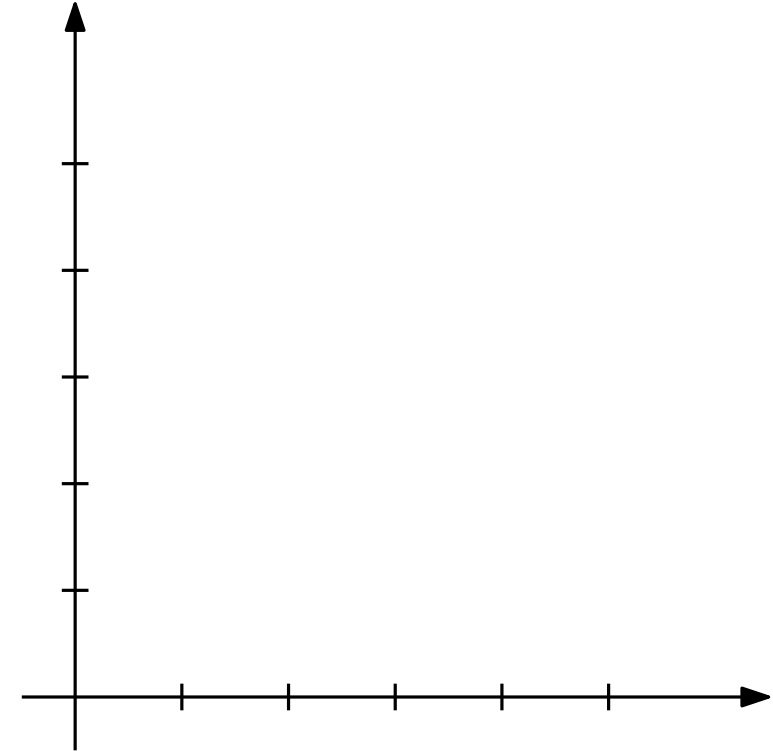
Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für



Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

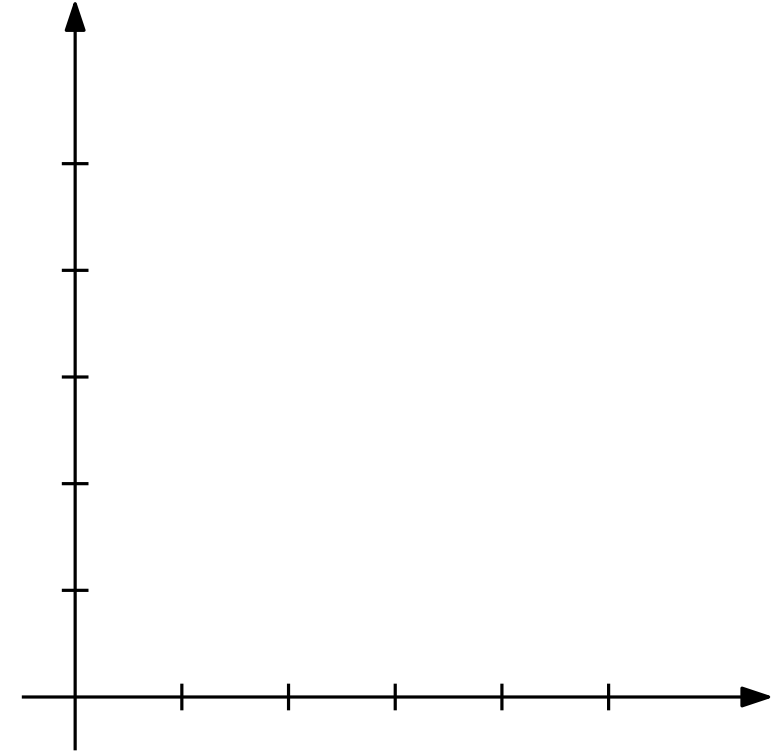
- lineare Beschränkungen



Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

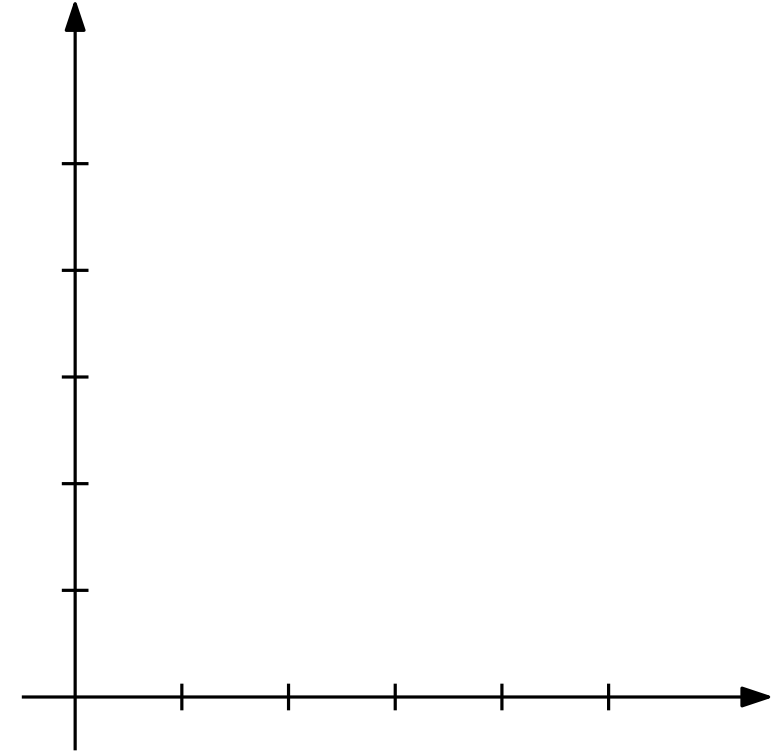
- lineare Beschränkungen
- lineare Zielfunktion



Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

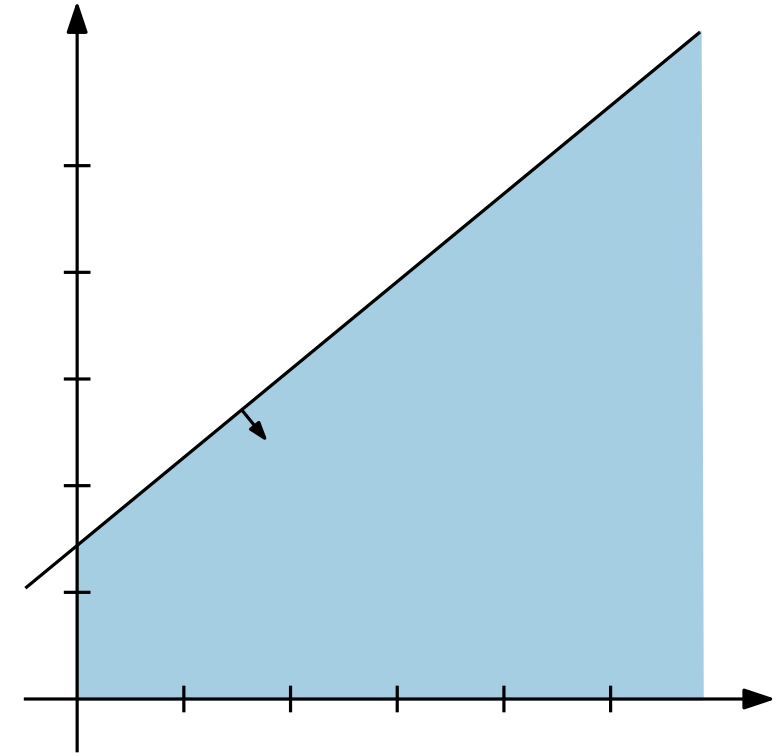
- lineare Beschränkungen
- lineare Zielfunktion
- reelle Variablen



Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

- lineare Beschränkungen
- lineare Zielfunktion
- reelle Variablen



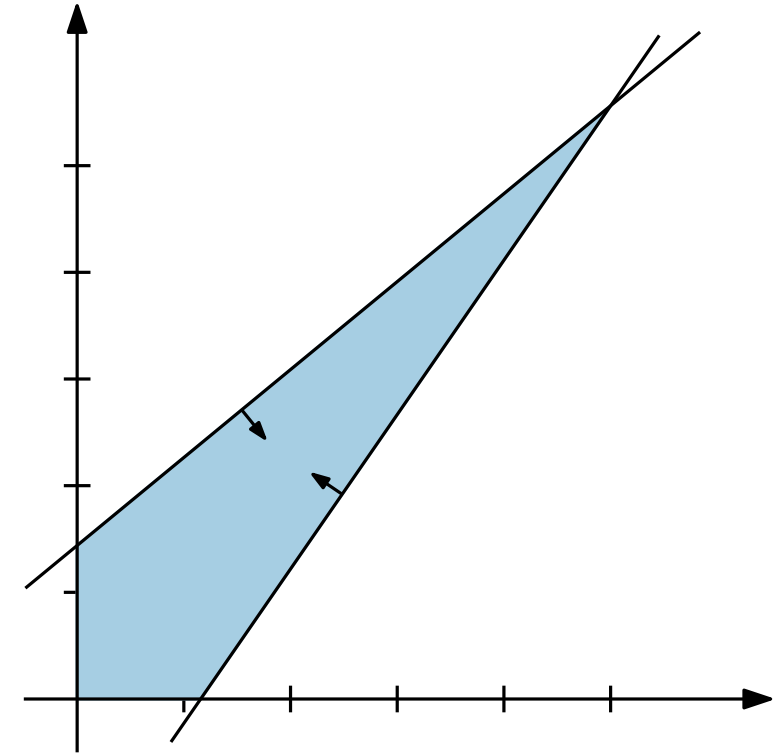
Beispiel.

$$y \leq 0.9x + 1.5$$

Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

- lineare Beschränkungen
- lineare Zielfunktion
- reelle Variablen



Beispiel.

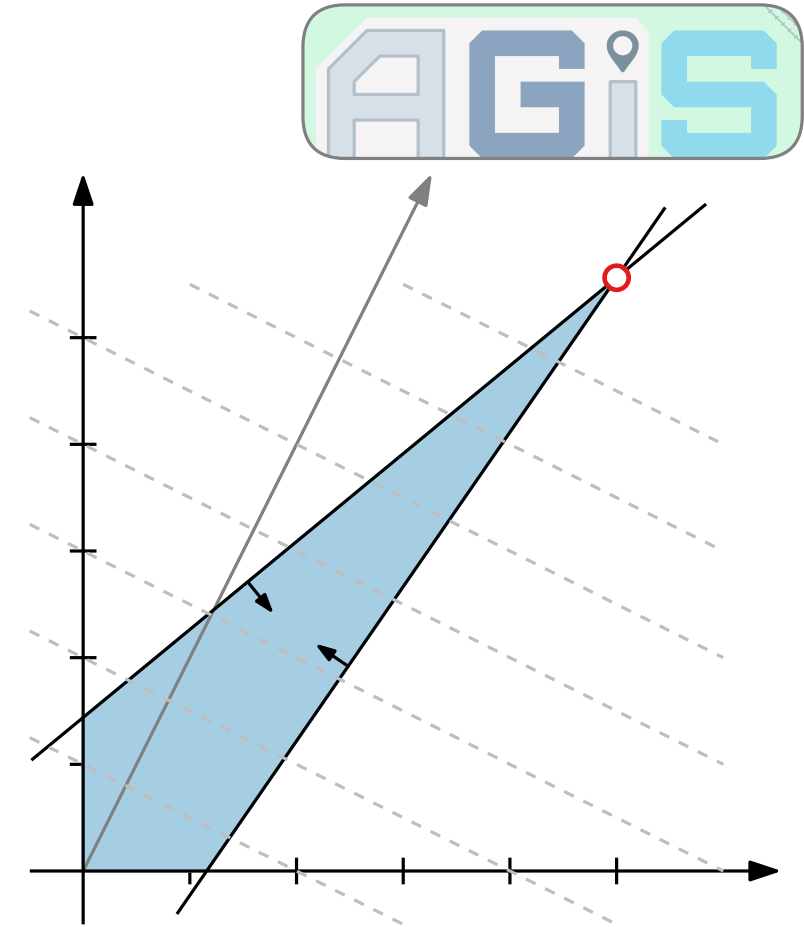
$$y \leq 0.9x + 1.5$$

$$y \geq 1.4x - 1.3$$

Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

- lineare Beschränkungen
- lineare Zielfunktion
- reelle Variablen



Beispiel.

$$y \leq 0.9x + 1.5$$

$$y \geq 1.4x - 1.3$$

$$\text{maximiere } x + 2y$$

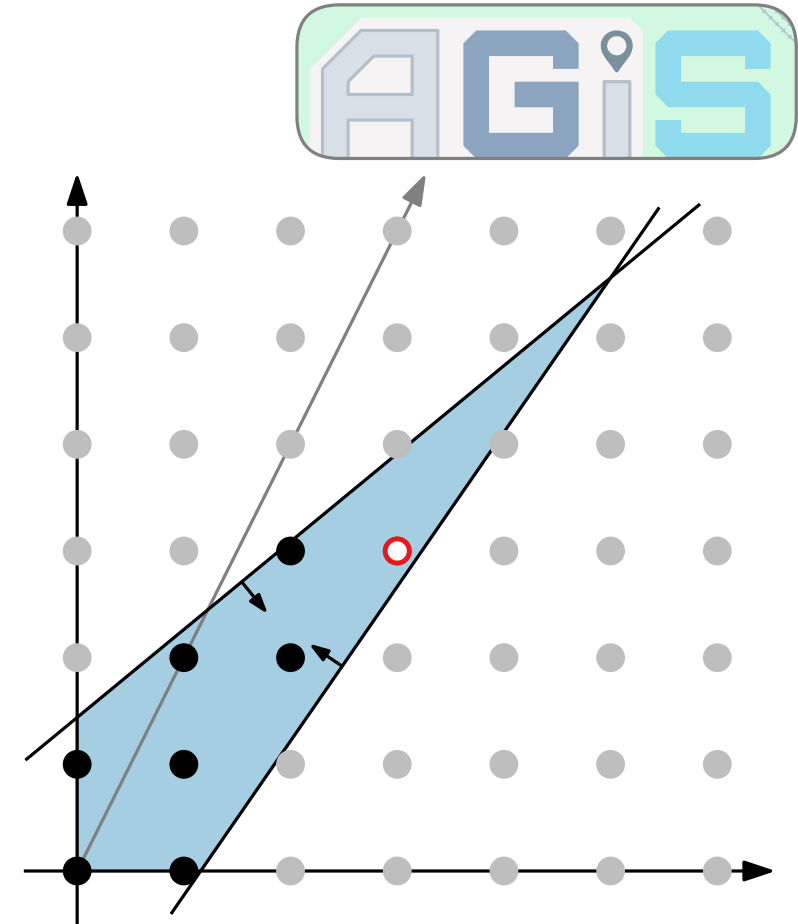
Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

- lineare Beschränkungen
- lineare Zielfunktion
- reelle Variablen

Mixed-Integer Programm (MIP)

- zusätzlich binäre und ganzzahlige Variablen



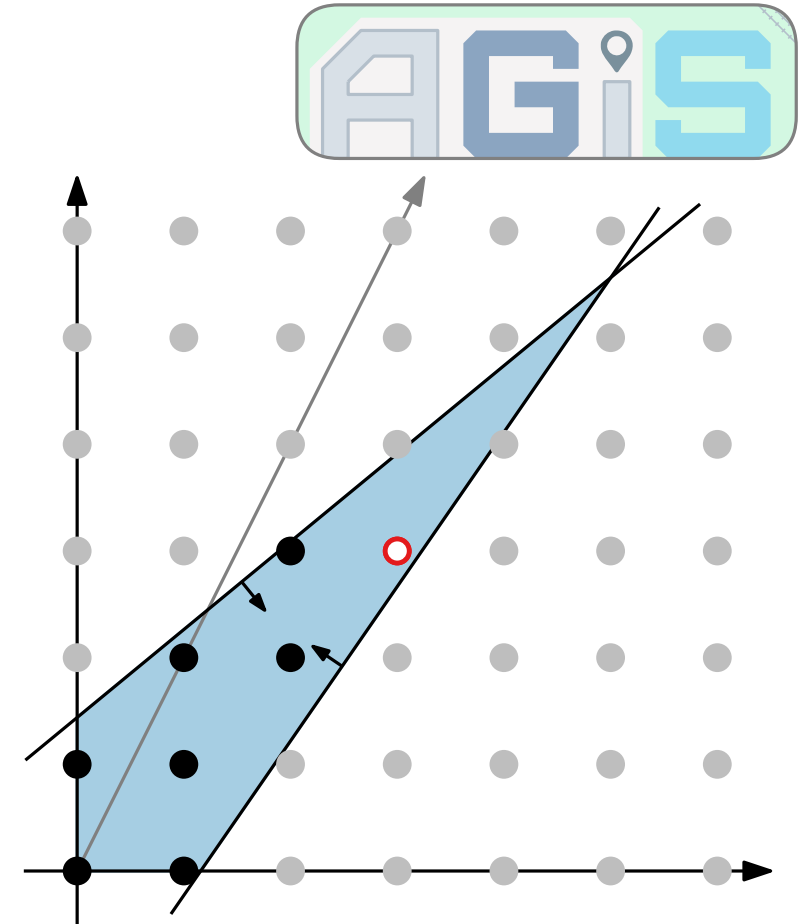
Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

- lineare Beschränkungen
- lineare Zielfunktion
- reelle Variablen

Mixed-Integer Programm (MIP)

- zusätzlich binäre und ganzzahlige Variablen
- NP-schweres Optimierungsproblem



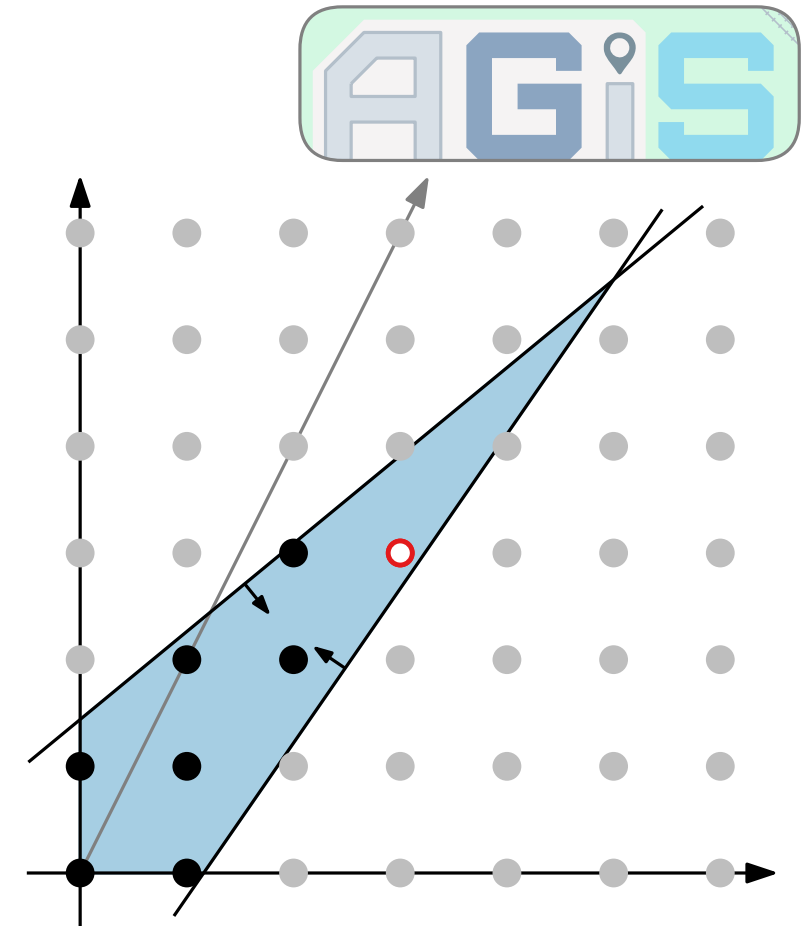
Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

- lineare Beschränkungen
- lineare Zielfunktion
- reelle Variablen

Mixed-Integer Programm (MIP)

- zusätzlich binäre und ganzzahlige Variablen
- NP-schweres Optimierungsproblem
- immer noch die bevorzugte Methode für viele praktische Optimierungsaufgaben



Lineare Programm

Lineares Programm (LP) ist eine effiziente Optimisierungsmethode für

- lineare Beschränkungen
- lineare Zielfunktion
- reelle Variablen

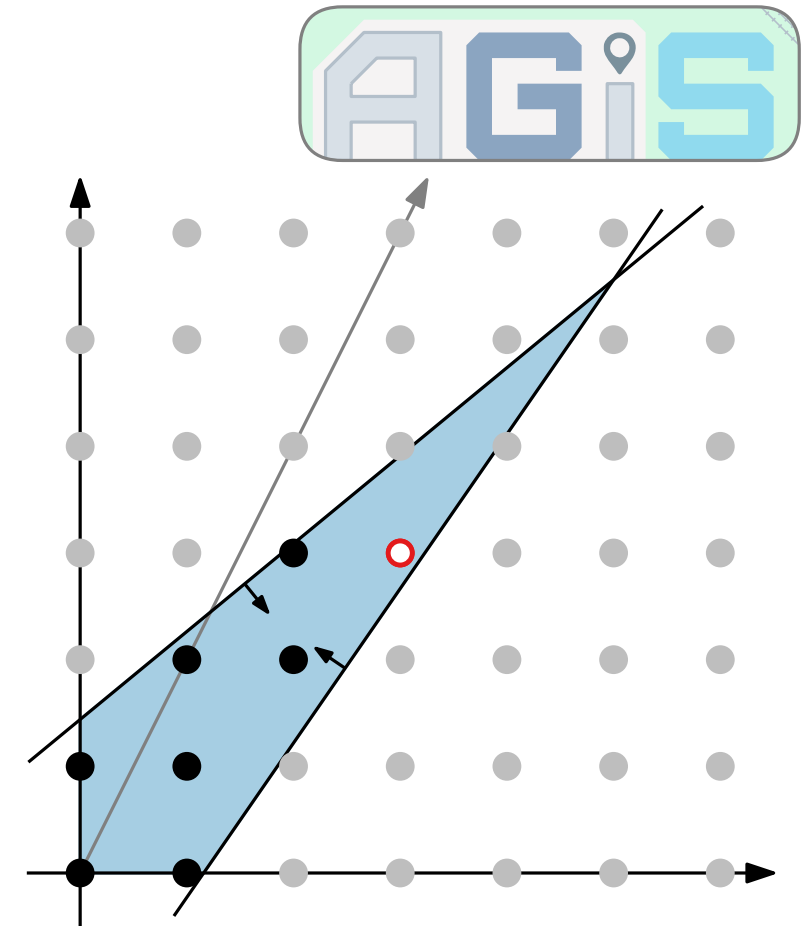
Mixed-Integer Programm (MIP)

- zusätzlich binäre und ganzzahlige Variablen
- NP-schweres Optimierungsproblem
- immer noch die bevorzugte Methode für viele praktische Optimierungsaufgaben

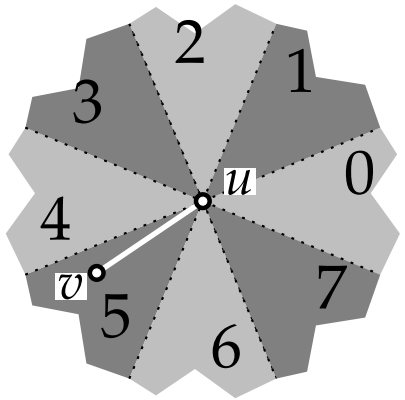
Satz.

Metro Map Layouts können als MIP modelliert werden, so dass

- harte Beschränkungen → lineare Beschränkungen
- weiche Beschränkungen → lineare Zielfunktion



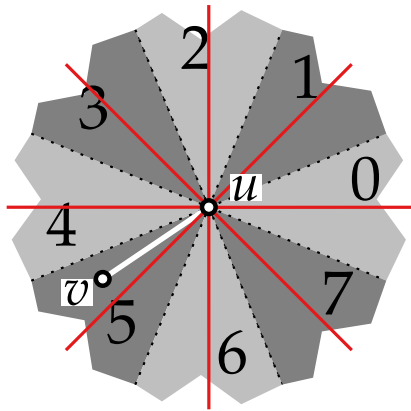
Sektoren und Koordinaten



Sektoren.

- für jeden Knoten u , unterteile die Ebene in acht Sektoren, nummeriert von 0 bis 7
hier: $\text{sec}(u, v) = 5$ in der Eingabe

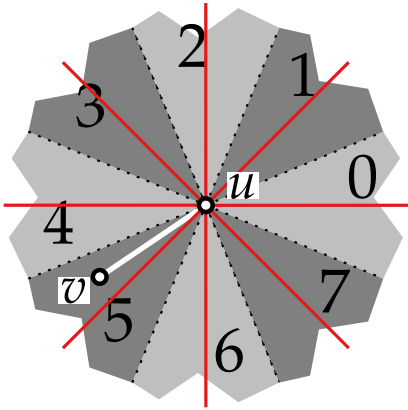
Sektoren und Koordinaten



Sektoren.

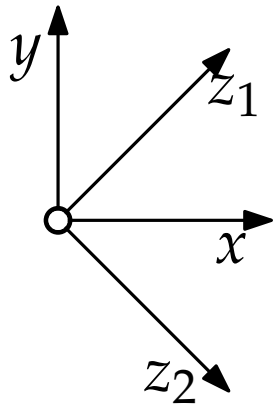
- für jeden Knoten u , unterteile die Ebene in acht Sektoren, nummeriert von 0 bis 7
hier: $\text{sec}(u, v) = 5$ in der Eingabe
- nummeriere oktilineare Kantenrichtungen entsprechend
hier, z.B., $\text{dir}(u, v) = 5$

Sektoren und Koordinaten



Sektoren.

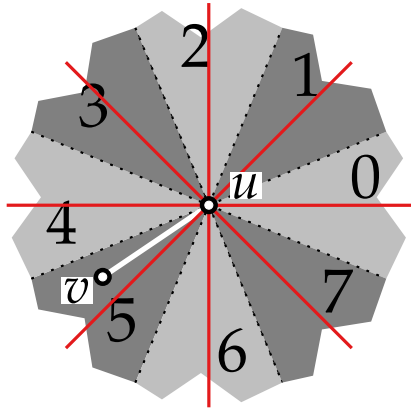
- für jeden Knoten u , unterteile die Ebene in acht Sektoren, nummeriert von 0 bis 7
hier: $\text{sec}(u, v) = 5$ in der Eingabe
- nummeriere oktilineare Kantenrichtungen entsprechend
hier, z.B., $\text{dir}(u, v) = 5$



Koordinaten.

- Weise jedem Knoten v (redundante) z_1 - und z_2 -Koordinaten zu
 - $z_1(v) = \frac{1}{2} \cdot (x(v) + y(v))$
 - $z_2(v) = \frac{1}{2} \cdot (x(v) - y(v))$

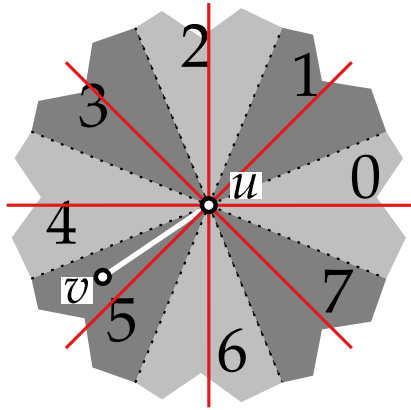
Oktilinearität und relative Positionen



Ziel.

Zeichne die Kante uv

Oktilinearität und relative Positionen



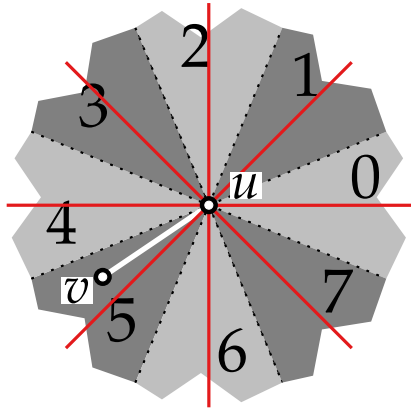
Ziel.

Zeichne die Kante uv

■ oktilinear (**R2**)



Oktilinearität und relative Positionen



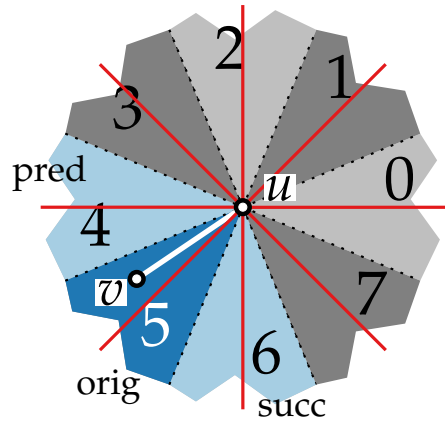
Ziel.

Zeichne die Kante uv

- oktilinear **(R2)**
- mit minimaler Länge $\ell = \ell_{uv}$ **(R7)**



Oktilinearität und relative Positionen



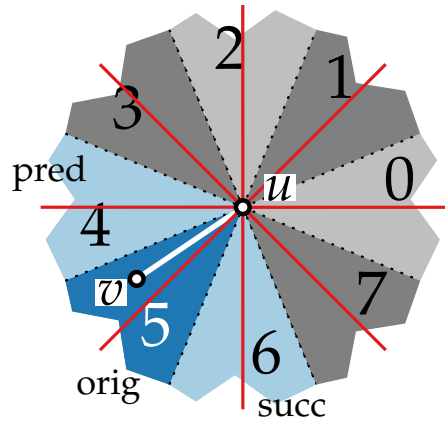
Ziel.

Zeichne die Kante uv

- oktilinear **(R2)**
- mit minimaler Länge $\ell = \ell_{uv}$ **(R7)**
- beschränkt auf die drei besten Richtungen **(R6)**



Oktilinearität und relative Positionen



Ziel.

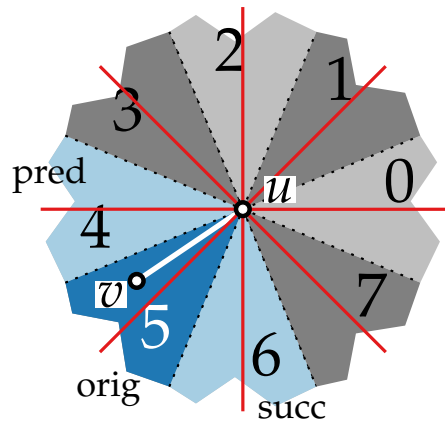
Zeichne die Kante uv

- oktilinear **(R2)**
- mit minimaler Länge $\ell = \ell_{uv}$ **(R7)**
- beschränkt auf die drei besten Richtungen **(R6)**

Wie modellieren wir das in einem MIP?



Oktilinearität und relative Positionen



Ziel.

Zeichne die Kante uv

- oktilinear **(R2)**
- mit minimaler Länge $\ell = \ell_{uv}$ **(R7)**
- beschränkt auf die drei besten Richtungen **(R6)**

Wie modellieren wir das in einem MIP?

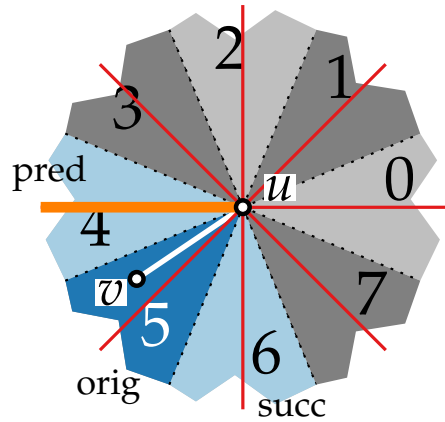
Führe **Binärvariablen** ein

$$\alpha_{\text{pred}}(u, v) + \alpha_{\text{orig}}(u, v) + \alpha_{\text{succ}}(u, v) = 1$$

um genau einen der drei Sektoren auszuwählen.



Oktilinearität und relative Positionen

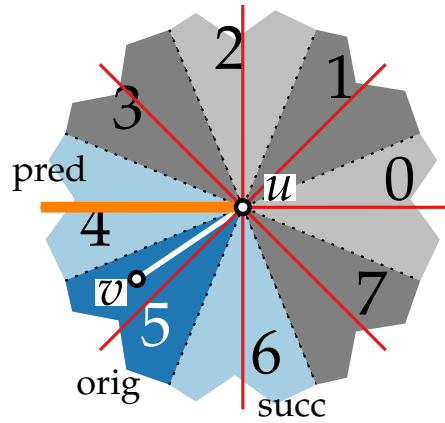


pred Sektor.

$$\begin{aligned} y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell \end{aligned}$$



Oktilinearität und relative Positionen



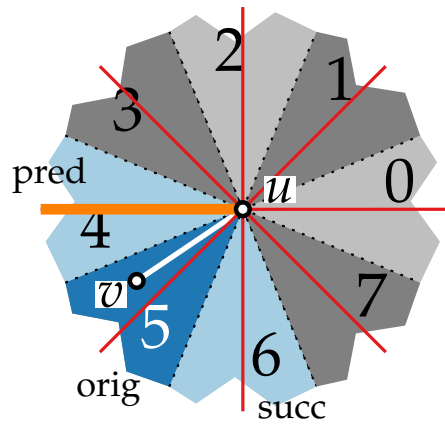
pred Sektor.

$$\begin{aligned}
 y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\
 -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\
 x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell
 \end{aligned}$$

sehr große Konstante



Oktilinearität und relative Positionen



pred Sektor.

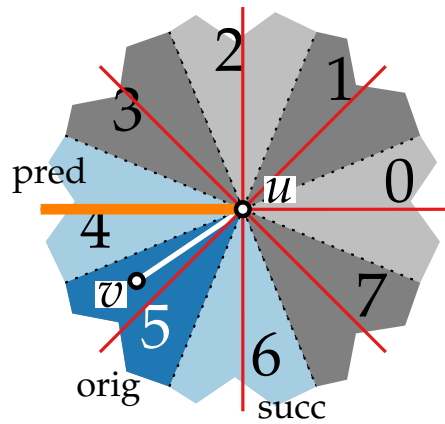
$$\begin{aligned}
 y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\
 -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\
 x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell
 \end{aligned}$$

sehr große Konstante

Wie funktioniert das?



Oktilinearität und relative Positionen



pred Sektor.

$$\begin{aligned} y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell \end{aligned}$$

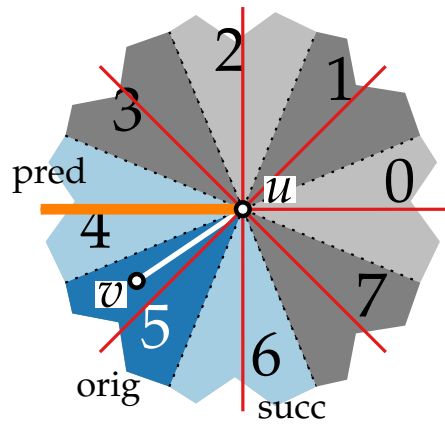
sehr große Konstante

Wie funktioniert das?

Fall 1: $\alpha_{\text{pred}}(u, v) = 1$



Oktilinearität und relative Positionen



pred Sektor.

$$\begin{aligned} y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell \end{aligned}$$

sehr große Konstante

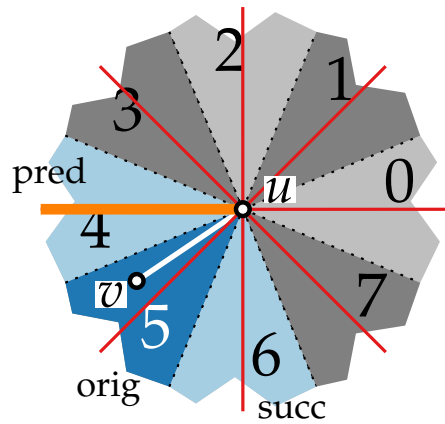
Wie funktioniert das?

Fall 1: $\alpha_{\text{pred}}(u, v) = 1$

$$\begin{aligned} y(u) - y(v) &\leq 0 \\ -y(u) + y(v) &\leq 0 \\ x(u) - x(v) &\geq \ell \end{aligned}$$



Oktilinearität und relative Positionen



pred Sektor.

$$\begin{aligned} y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell \end{aligned}$$

sehr große Konstante

Wie funktioniert das?

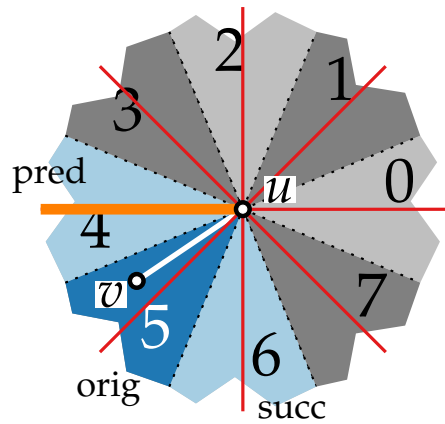
Fall 1: $\alpha_{\text{pred}}(u, v) = 1$

$$\begin{aligned} y(u) - y(v) &\leq 0 \\ -y(u) + y(v) &\leq 0 \\ x(u) - x(v) &\geq \ell \end{aligned}$$

Fall 2: $\alpha_{\text{pred}}(u, v) = 0$



Oktilinearität und relative Positionen



pred Sektor.

$$\begin{aligned} y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell \end{aligned}$$

sehr große Konstante

Wie funktioniert das?

Fall 1: $\alpha_{\text{pred}}(u, v) = 1$

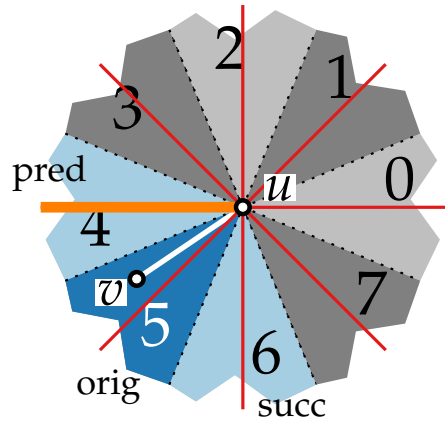
$$\begin{aligned} y(u) - y(v) &\leq 0 \\ -y(u) + y(v) &\leq 0 \\ x(u) - x(v) &\geq \ell \end{aligned}$$

Fall 2: $\alpha_{\text{pred}}(u, v) = 0$

$$\begin{aligned} y(u) - y(v) &\leq M \\ -y(u) + y(v) &\leq M \\ x(u) - x(v) &\geq -M + \ell \end{aligned}$$



Oktilinearität und relative Positionen



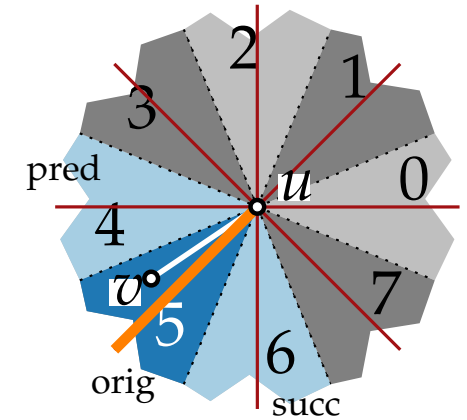
pred Sektor.

$$\begin{aligned} y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell \end{aligned}$$

sehr große Konstante

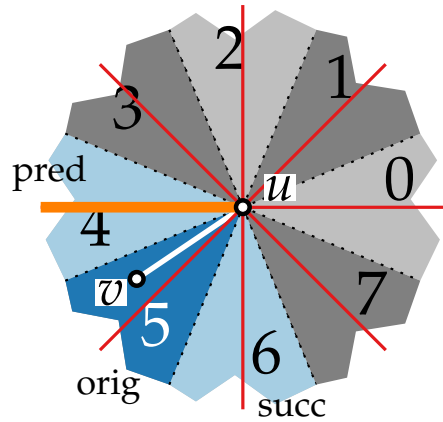
orig Sektor.

$$\begin{aligned} z_2(u) - z_2(v) &\leq M(1 - \alpha_{\text{orig}}(u, v)) \\ -z_2(u) + z_2(v) &\leq M(1 - \alpha_{\text{orig}}(u, v)) \\ z_1(u) - z_1(v) &\geq -M(1 - \alpha_{\text{orig}}(u, v)) + \ell \end{aligned}$$





Oktilinearität und relative Positionen



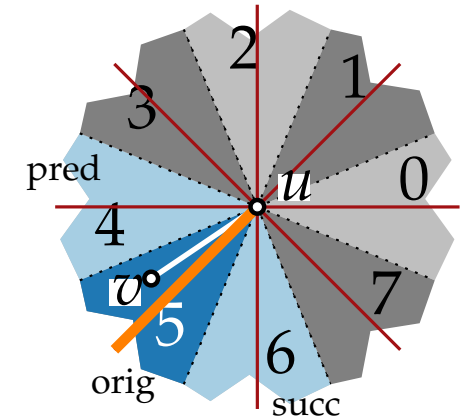
pred Sektor.

sehr große Konstante

$$\begin{aligned} y(u) - y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ -y(u) + y(v) &\leq M(1 - \alpha_{\text{pred}}(u, v)) \\ x(u) - x(v) &\geq -M(1 - \alpha_{\text{pred}}(u, v)) + \ell \end{aligned}$$

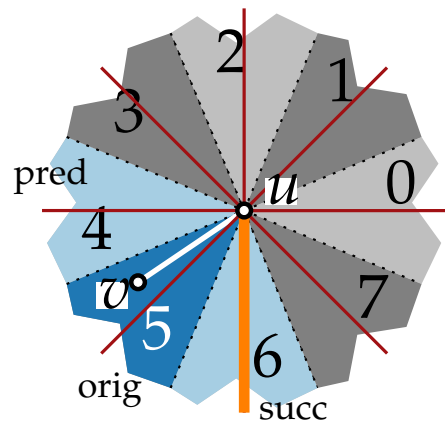
orig Sektor.

$$\begin{aligned} z_2(u) - z_2(v) &\leq M(1 - \alpha_{\text{orig}}(u, v)) \\ -z_2(u) + z_2(v) &\leq M(1 - \alpha_{\text{orig}}(u, v)) \\ z_1(u) - z_1(v) &\geq -M(1 - \alpha_{\text{orig}}(u, v)) + \ell \end{aligned}$$



succ Sektor.

$$\begin{aligned} x(u) - x(v) &\leq M(1 - \alpha_{\text{succ}}(u, v)) \\ -x(u) + x(v) &\leq M(1 - \alpha_{\text{succ}}(u, v)) \\ y(u) - y(v) &\geq -M(1 - \alpha_{\text{succ}}(u, v)) + \ell \end{aligned}$$



Zielfunktion

- modelliert die drei weichen Beschränkungen





Zielfunktion

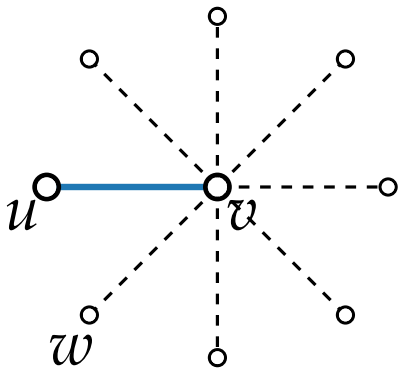
- modelliert die drei weichen Beschränkungen
 - gewichtete Summe der einzelnen Kostenfunktionen
- minimiere $\lambda_{\text{bends}} \text{cost}_{\text{bends}} + \lambda_{\text{length}} \text{cost}_{\text{length}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$



Zielfunktion

- modelliert die drei weichen Beschränkungen
- gewichtete Summe der einzelnen Kostenfunktionen
minimiere $\lambda_{\text{bends}} \text{cost}_{\text{bends}} + \lambda_{\text{length}} \text{cost}_{\text{length}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$

Beispiel: Knicke (R3/R4)



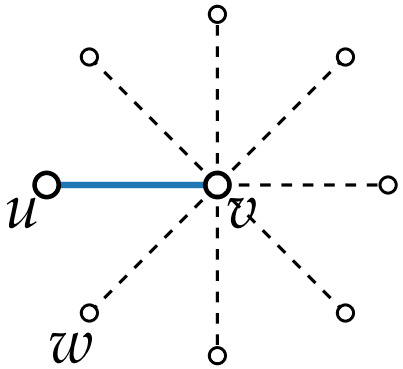


Zielfunktion

- modelliert die drei weichen Beschränkungen
- gewichtete Summe der einzelnen Kostenfunktionen
minimiere $\lambda_{\text{bends}} \text{cost}_{\text{bends}} + \lambda_{\text{length}} \text{cost}_{\text{length}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$

Beispiel: Knicke (R3/R4)

Kanten uv und vw auf einer Linie $L \in \mathcal{L}$





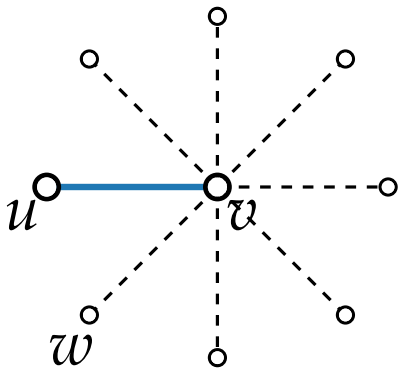
Zielfunktion

- modelliert die drei weichen Beschränkungen
- gewichtete Summe der einzelnen Kostenfunktionen
minimiere $\lambda_{\text{bends}} \text{cost}_{\text{bends}} + \lambda_{\text{length}} \text{cost}_{\text{length}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$

Beispiel: Knicke (R3/R4)

Kanten uv und vw auf einer Linie $L \in \mathcal{L}$

- so gerade wie möglich zeichnen

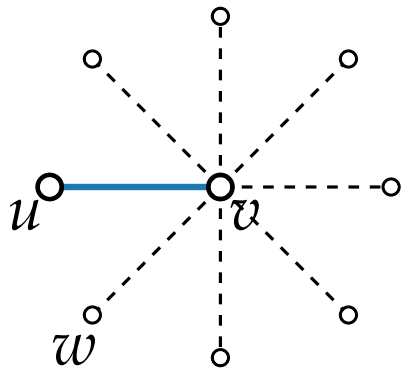




Zielfunktion

- modelliert die drei weichen Beschränkungen
- gewichtete Summe der einzelnen Kostenfunktionen
minimiere $\lambda_{\text{bends}} \text{cost}_{\text{bends}} + \lambda_{\text{length}} \text{cost}_{\text{length}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$

Beispiel: Knicke (R3/R4)



Kanten uv und vw auf einer Linie $L \in \mathcal{L}$

- so gerade wie möglich zeichnen
- zunehmende Kosten $\text{bend}(u, v, w)$ für spitzeren Winkel $\angle(\overline{uv}, \overline{vw})$

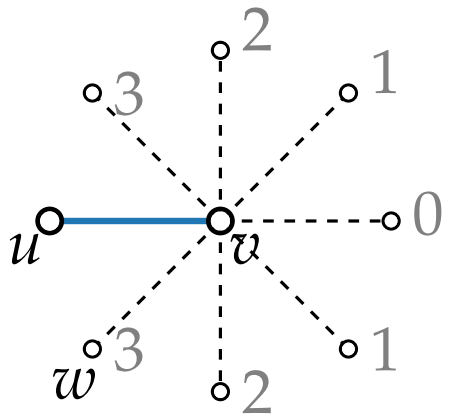
$$\text{cost}_{\text{bends}} = \sum_{L \in \mathcal{L}} \sum_{uv, vw \in L} \text{bend}(u, v, w)$$



Zielfunktion

- modelliert die drei weichen Beschränkungen
- gewichtete Summe der einzelnen Kostenfunktionen
minimiere $\lambda_{\text{bends}} \text{cost}_{\text{bends}} + \lambda_{\text{length}} \text{cost}_{\text{length}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$

Beispiel: Knicke (R3/R4)



Kanten uv und vw auf einer Linie $L \in \mathcal{L}$

- so gerade wie möglich zeichnen
- zunehmende Kosten $\text{bend}(u, v, w)$ für spitzere Winkel $\angle(\overline{uv}, \overline{vw})$

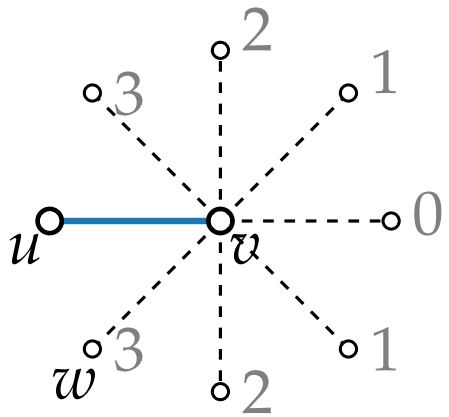
$$\text{cost}_{\text{bends}} = \sum_{L \in \mathcal{L}} \sum_{uv, vw \in L} \text{bend}(u, v, w)$$



Zielfunktion

- modelliert die drei weichen Beschränkungen
- gewichtete Summe der einzelnen Kostenfunktionen
minimiere $\lambda_{\text{bends}} \text{cost}_{\text{bends}} + \lambda_{\text{length}} \text{cost}_{\text{length}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$

Beispiel: Knicke (R3/R4)



Kanten uv und $v\omega$ auf einer Linie $L \in \mathcal{L}$

- so gerade wie möglich zeichnen
- zunehmende Kosten $\text{bend}(u, v, w)$ für spitzeren Winkel $\angle(\overline{uv}, \overline{vw})$

$$\text{cost}_{\text{bends}} = \sum_{L \in \mathcal{L}} \sum_{uv, v\omega \in L} \text{bend}(u, v, w)$$

Um $\text{bend}(u, v, w)$ korrekt zuzuweisen, müssen wir einige lineare Beschränkungen basierend auf den Richtungsvariablen $\text{dir}(u, v)$ und $\text{dir}(v, w)$ definieren.

Übersicht MIP Modell



Beschränkungen.

- Linearisierung aller harten Beschränkungen
- $\mathcal{O}(n^2)$ Variablen und Beschränkungen (dank Planarität)

Übersicht MIP Modell



Beschränkungen.

- Linearisierung aller harten Beschränkungen
- $\mathcal{O}(n^2)$ Variablen und Beschränkungen (dank Planarität)

Zielfunktion.

- gewichtete Summe der drei weichen Beschränkungen
- minimiere $\lambda_{\text{bend}} \text{cost}_{\text{bend}} + \lambda_{\text{len}} \text{cost}_{\text{len}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$

Übersicht MIP Modell

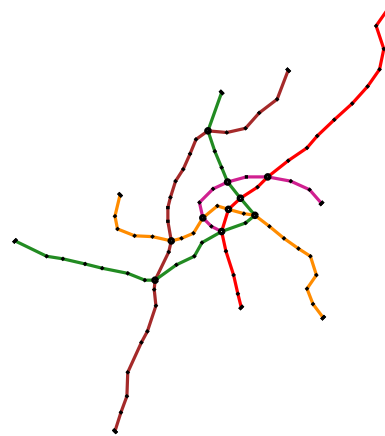


Beschränkungen.

- Linearisierung aller harten Beschränkungen
- $\mathcal{O}(n^2)$ Variablen und Beschränkungen (dank Planarität)

Zielfunktion.

- gewichtete Summe der drei weichen Beschränkungen
- minimiere $\lambda_{\text{bend}} \text{cost}_{\text{bend}} + \lambda_{\text{len}} \text{cost}_{\text{len}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$
- Effekt:





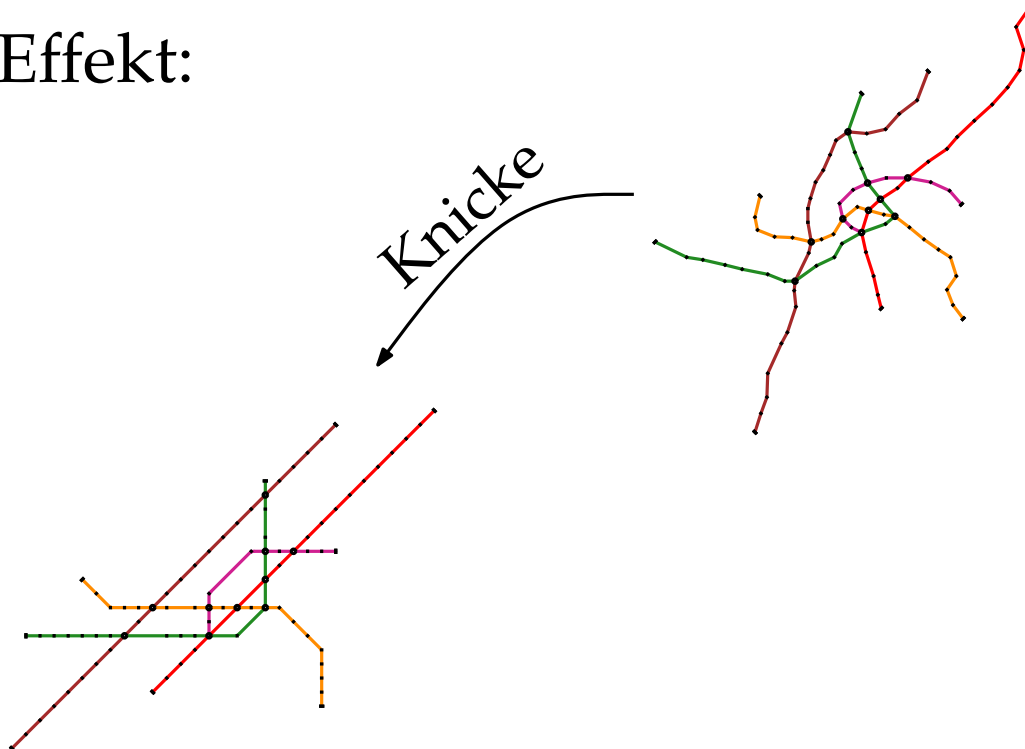
Übersicht MIP Modell

Beschränkungen.

- Linearisierung aller harten Beschränkungen
- $\mathcal{O}(n^2)$ Variablen und Beschränkungen (dank Planarität)

Zielfunktion.

- gewichtete Summe der drei weichen Beschränkungen
- minimiere $\lambda_{\text{bend}} \text{cost}_{\text{bend}} + \lambda_{\text{len}} \text{cost}_{\text{len}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$
- Effekt:



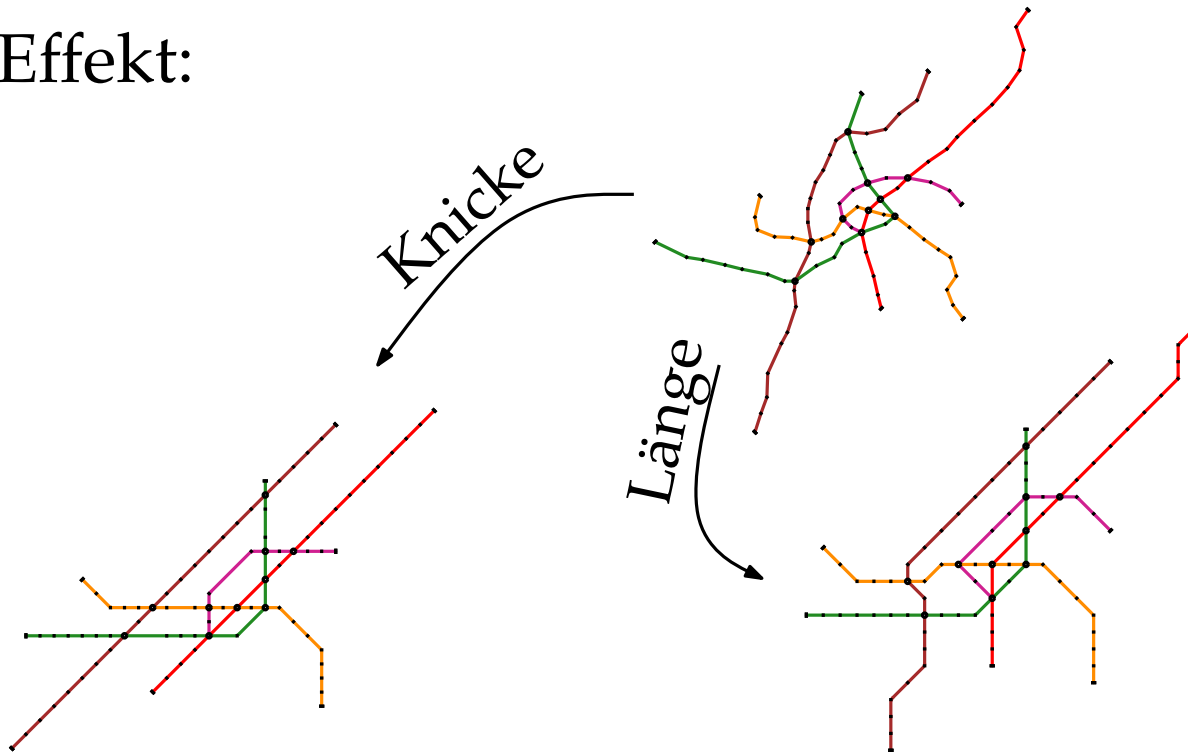
Übersicht MIP Modell

Beschränkungen.

- Linearisierung aller harten Beschränkungen
- $\mathcal{O}(n^2)$ Variablen und Beschränkungen (dank Planarität)

Zielfunktion.

- gewichtete Summe der drei weichen Beschränkungen
- minimiere $\lambda_{\text{bend}} \text{cost}_{\text{bend}} + \lambda_{\text{len}} \text{cost}_{\text{len}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$
- Effekt:



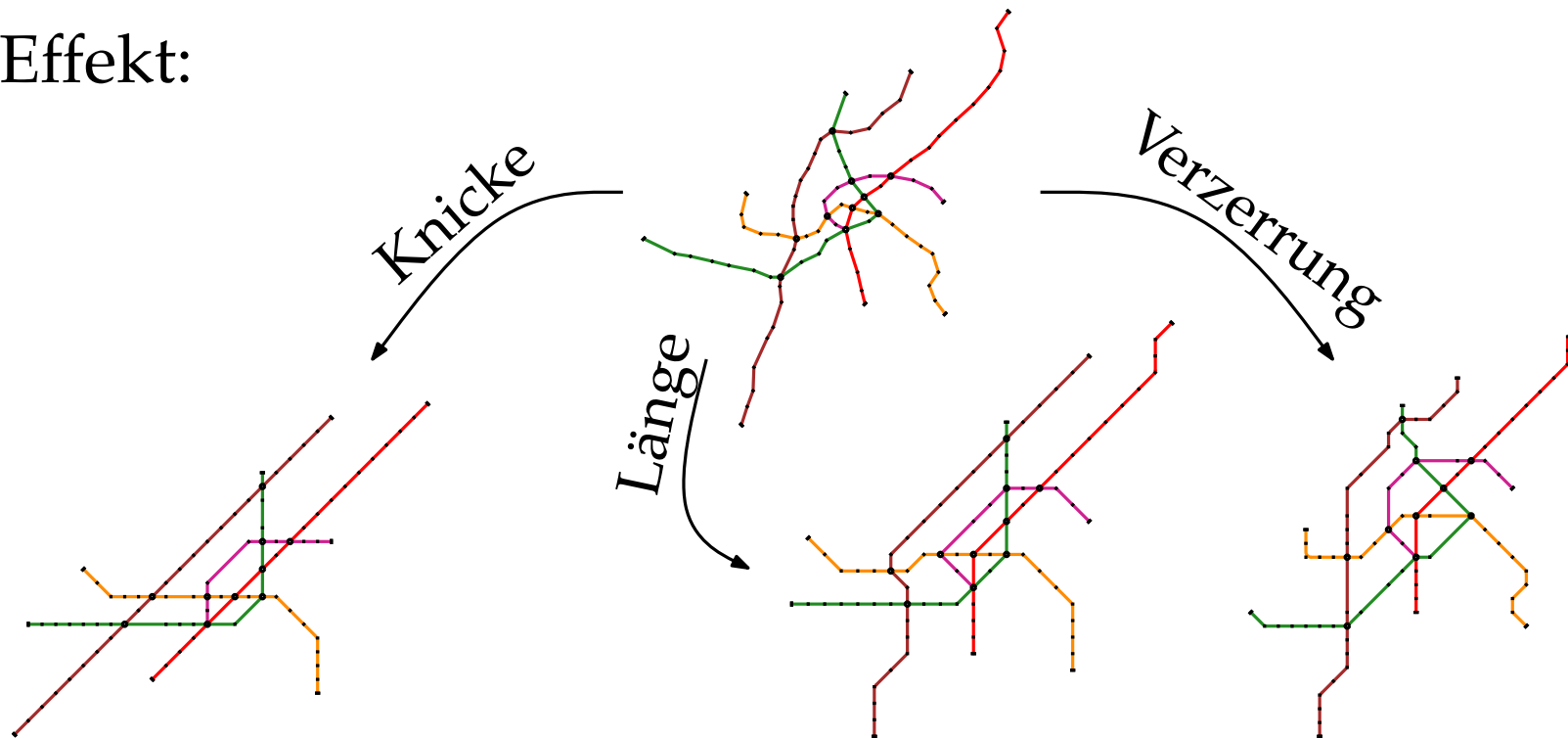
Übersicht MIP Modell

Beschränkungen.

- Linearisierung aller harten Beschränkungen
- $\mathcal{O}(n^2)$ Variablen und Beschränkungen (dank Planarität)

Zielfunktion.

- gewichtete Summe der drei weichen Beschränkungen
- minimiere $\lambda_{\text{bend}} \text{cost}_{\text{bend}} + \lambda_{\text{len}} \text{cost}_{\text{len}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$
- Effekt:



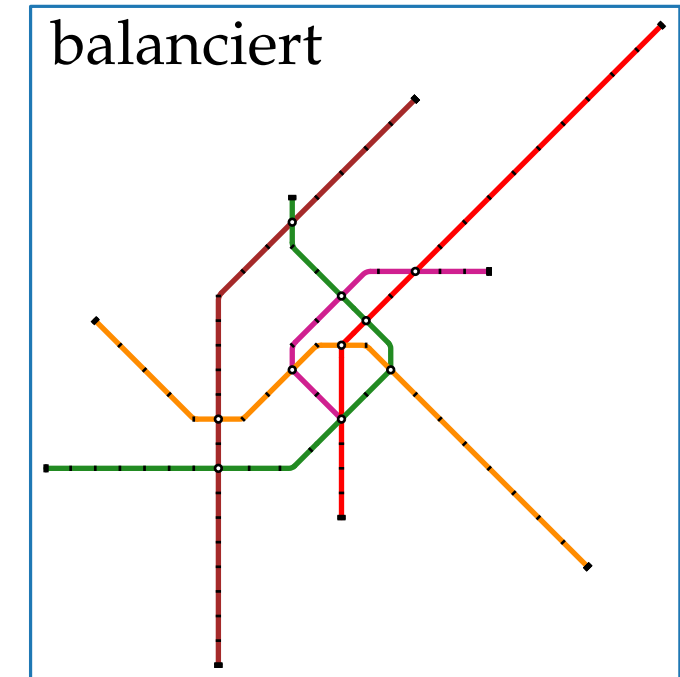
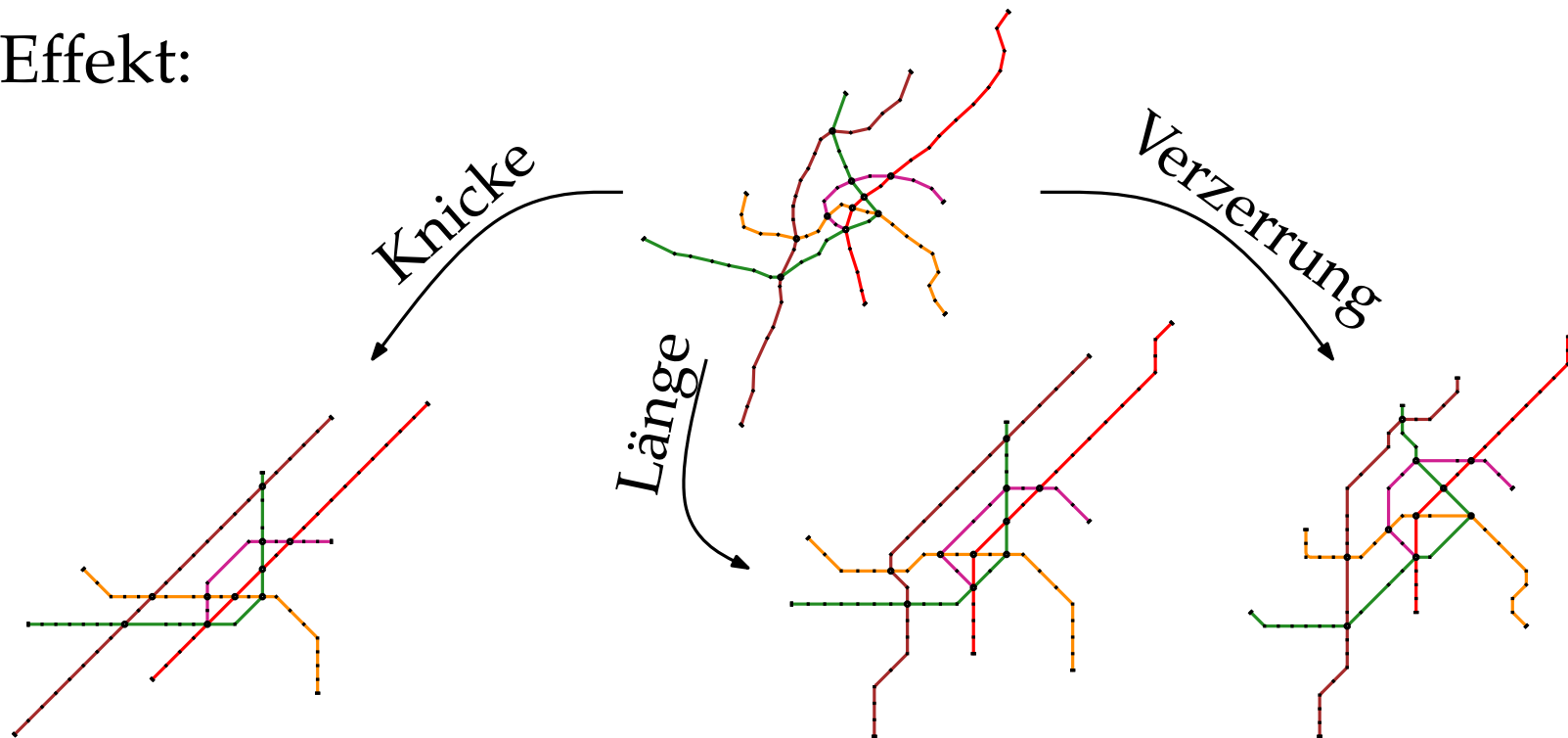
Übersicht MIP Modell

Beschränkungen.

- Linearisierung aller harten Beschränkungen
- $\mathcal{O}(n^2)$ Variablen und Beschränkungen (dank Planarität)

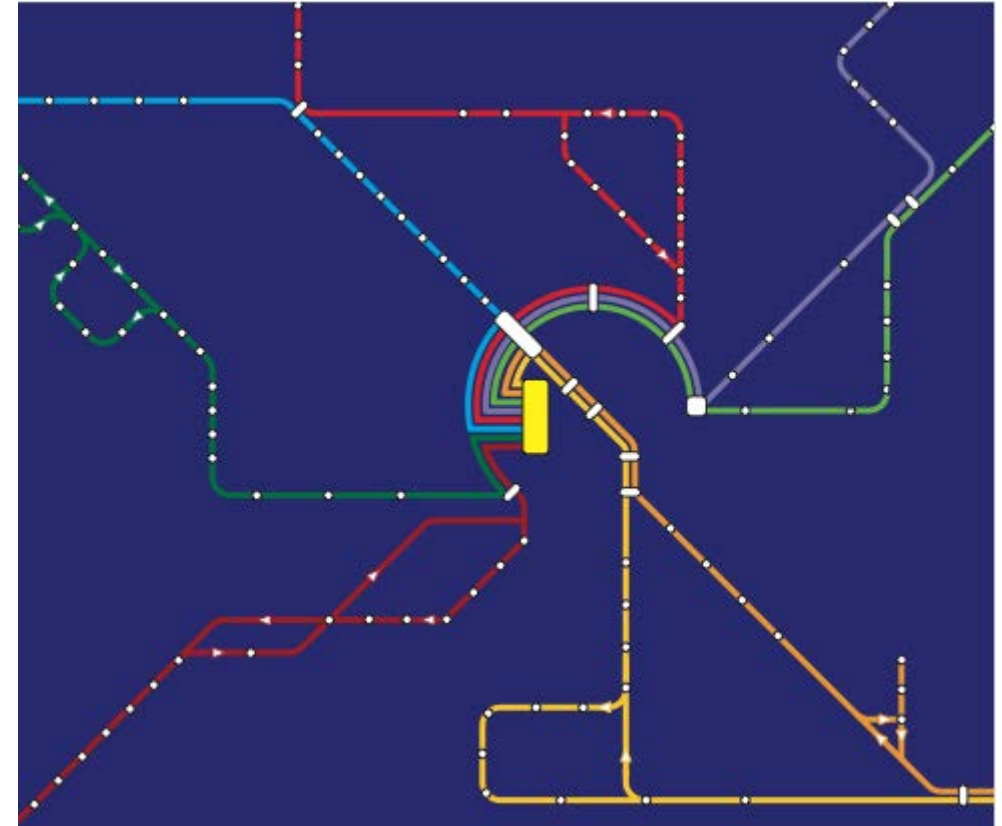
Zielfunktion.

- gewichtete Summe der drei weichen Beschränkungen
- minimiere $\lambda_{\text{bend}} \text{cost}_{\text{bend}} + \lambda_{\text{len}} \text{cost}_{\text{len}} + \lambda_{\text{dist}} \text{cost}_{\text{dist}}$
- Effekt:



Beschriftungen

- unbeschriftete Karte ziemlich nutzlos





Beschriftungen

- unbeschriftete Karte ziemlich nutzlos
- Beschriftungen brauchen Platz



Beschriftungen

- unbeschriftete Karte ziemlich nutzlos
- Beschriftungen brauchen Platz
- Beschriftungen dürfen sich nicht überlappen



Beschriftungen

- unbeschriftete Karte ziemlich nutzlos
- Beschriftungen brauchen Platz
- Beschriftungen dürfen sich nicht überlappen
- Graph Beschriftungs Problem ist NP-schwer
[Tollis & Kakoulis 2001]



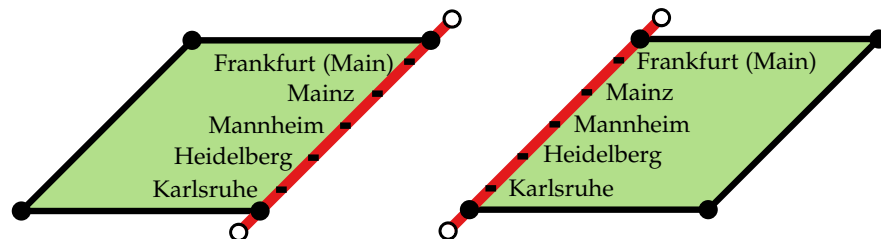
Beschriftungen



- unbeschriftete Karte ziemlich nutzlos
- Beschriftungen brauchen Platz
- Beschriftungen dürfen sich nicht überlappen
- Graph Beschriftungs Problem ist NP-schwer
[Tollis & Kakoulis 2001]

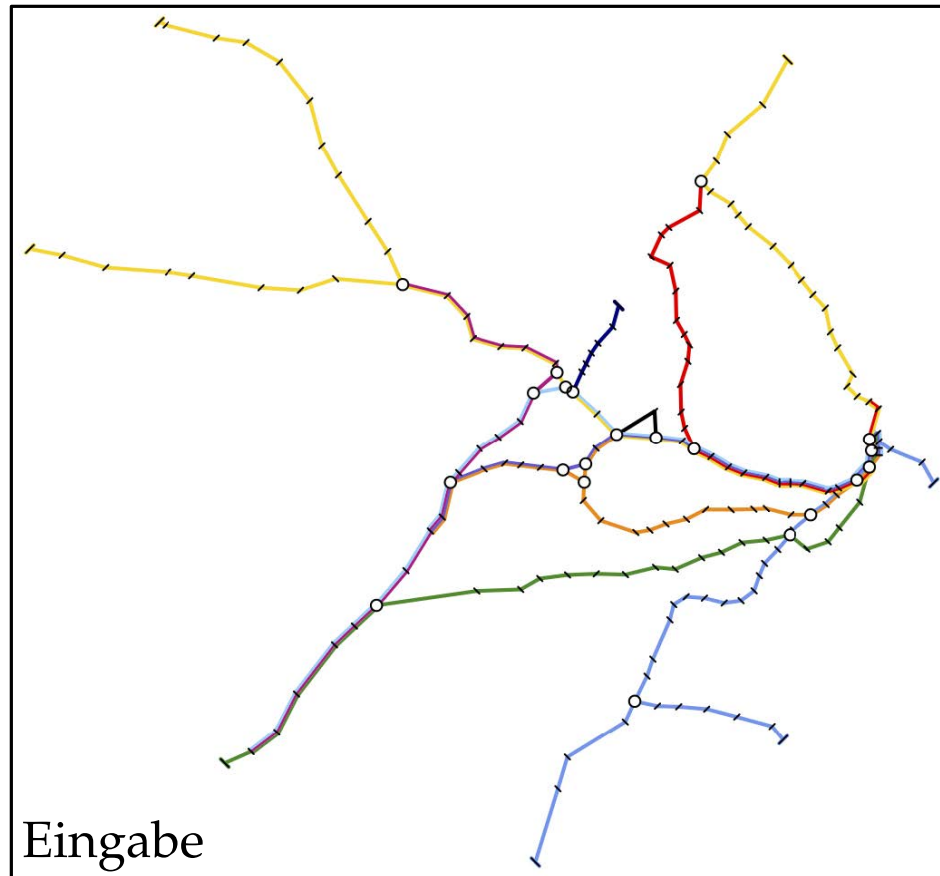


➔ kombiniere Layout und Beschriftung für optimale Ergebnisse



- Parallelogram als extra U-Bahn-Linie
- Seitenwechsel erlaubt

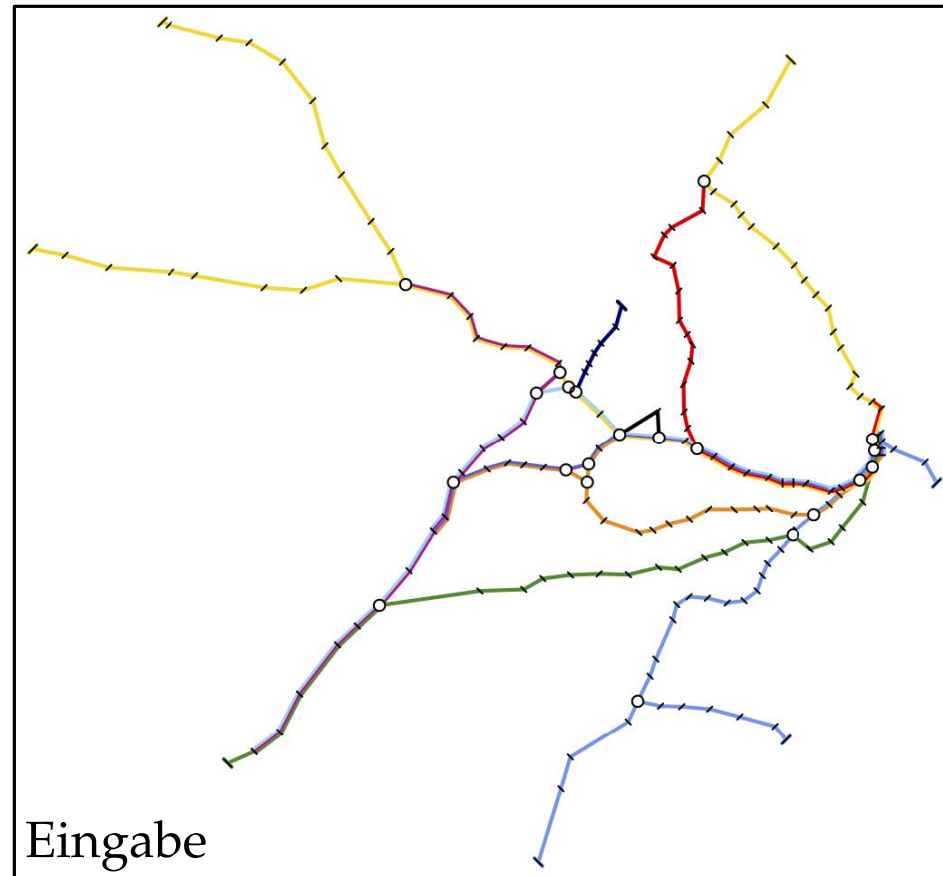
Beispiel: Sydney



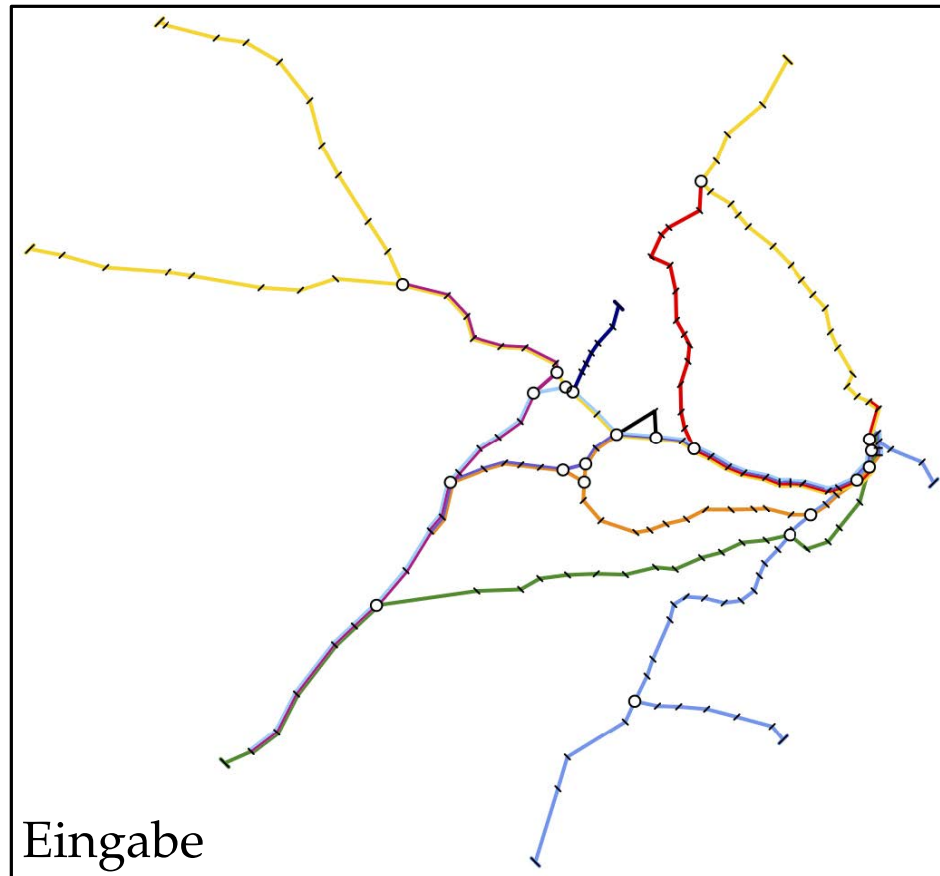
Beispiel: Sydney



Eingabe	$ V $	$ E $	$ F $	$ \mathcal{L} $
vollständig	174	183	11	10
reduziert	88	97		
beschriftet	242	270		



Beispiel: Sydney

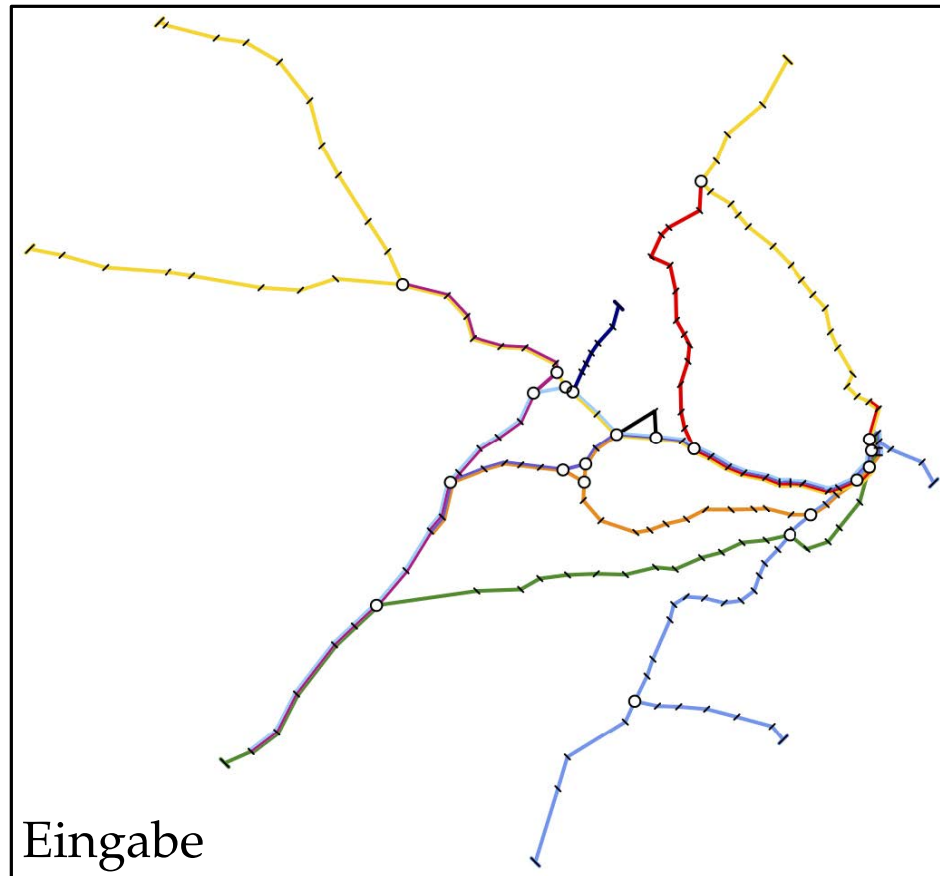


Eingabe	$ V $	$ E $	$ F $	$ \mathcal{L} $
vollständig	174	183		
reduziert	88	97	11	10
beschriftet	242	270	30	

↓

MIP	Beschränkungen	Variablen
vollständig	1,191,406	290,137
callback	21,988	92,681

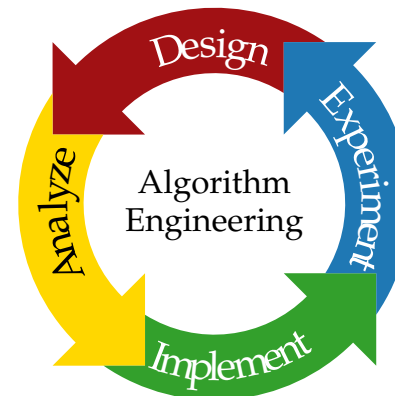
Beispiel: Sydney



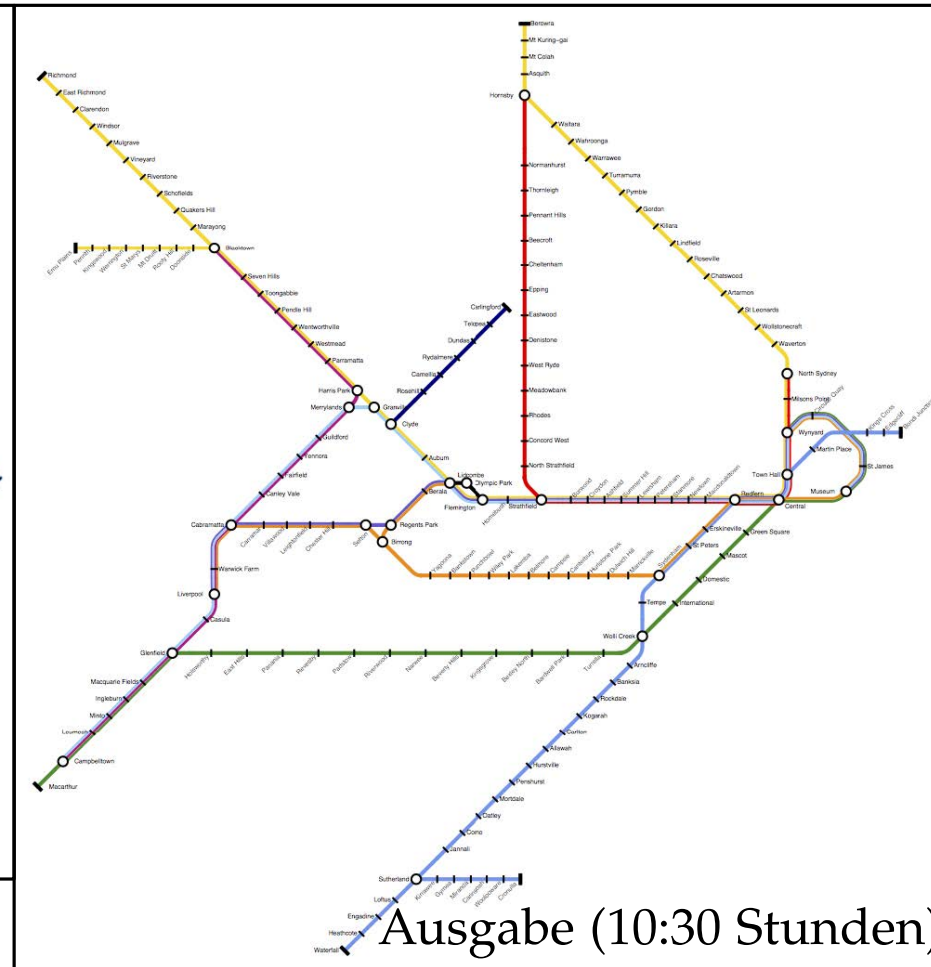
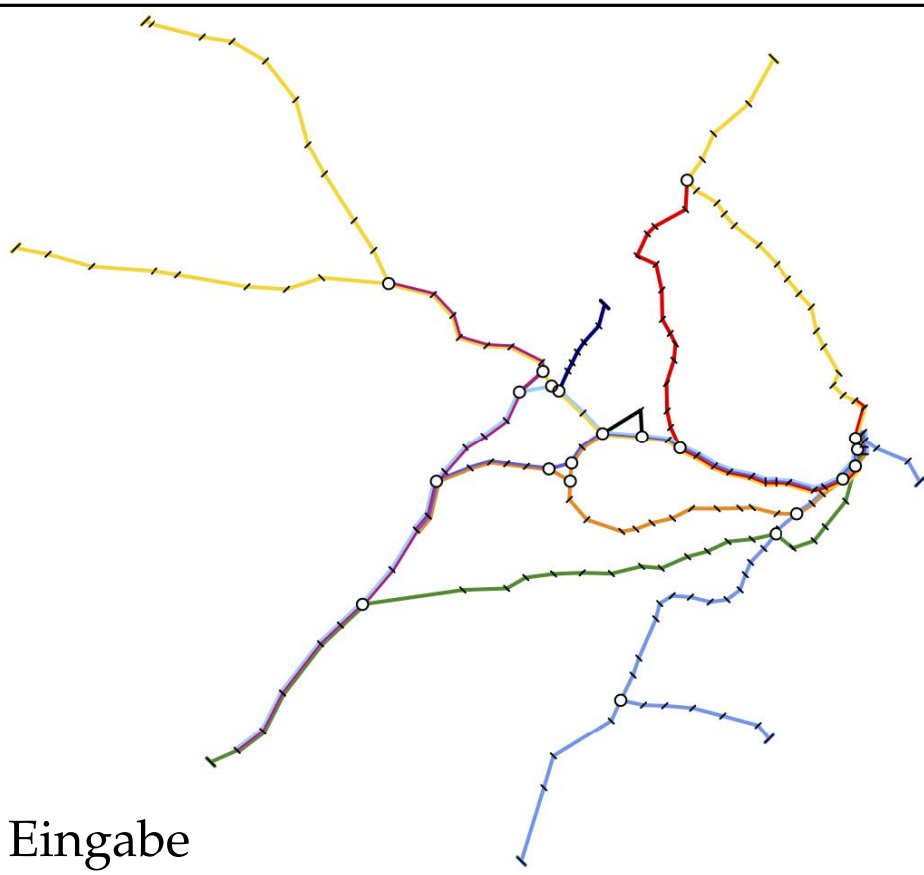
Eingabe	$ V $	$ E $	$ F $	$ \mathcal{L} $
vollständig	174	183		
reduziert	88	97	11	10
beschriftet	242	270	30	



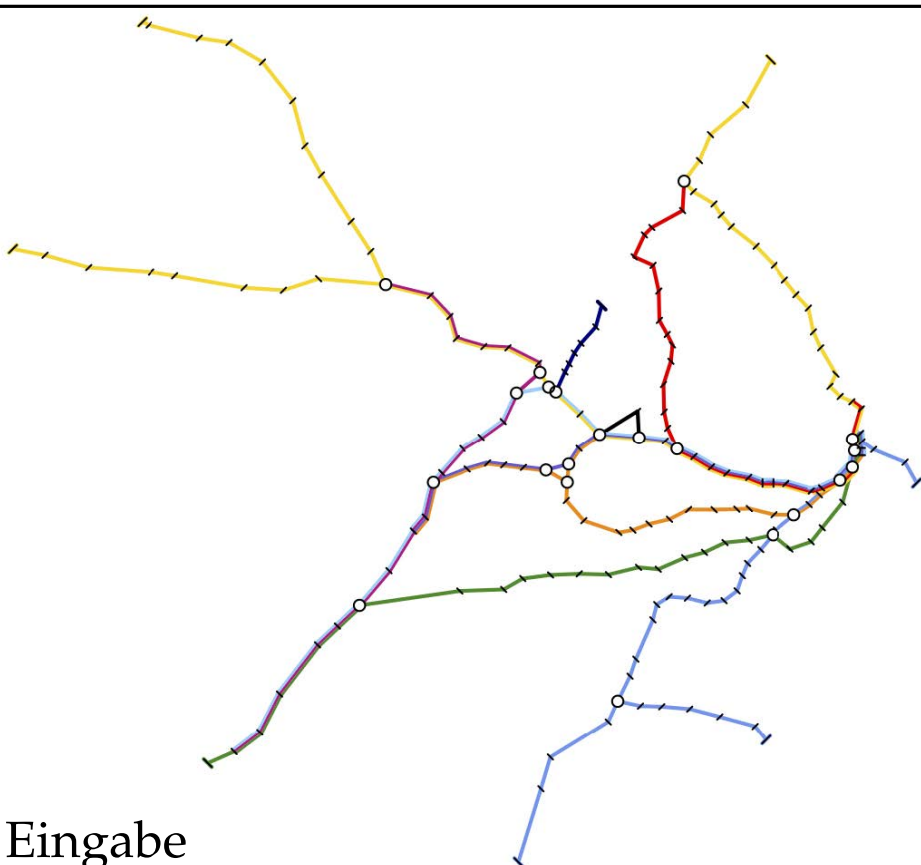
MIP	Beschränkungen	Variablen
vollständig	1,191,406	290,137
callback	21,988	92,681



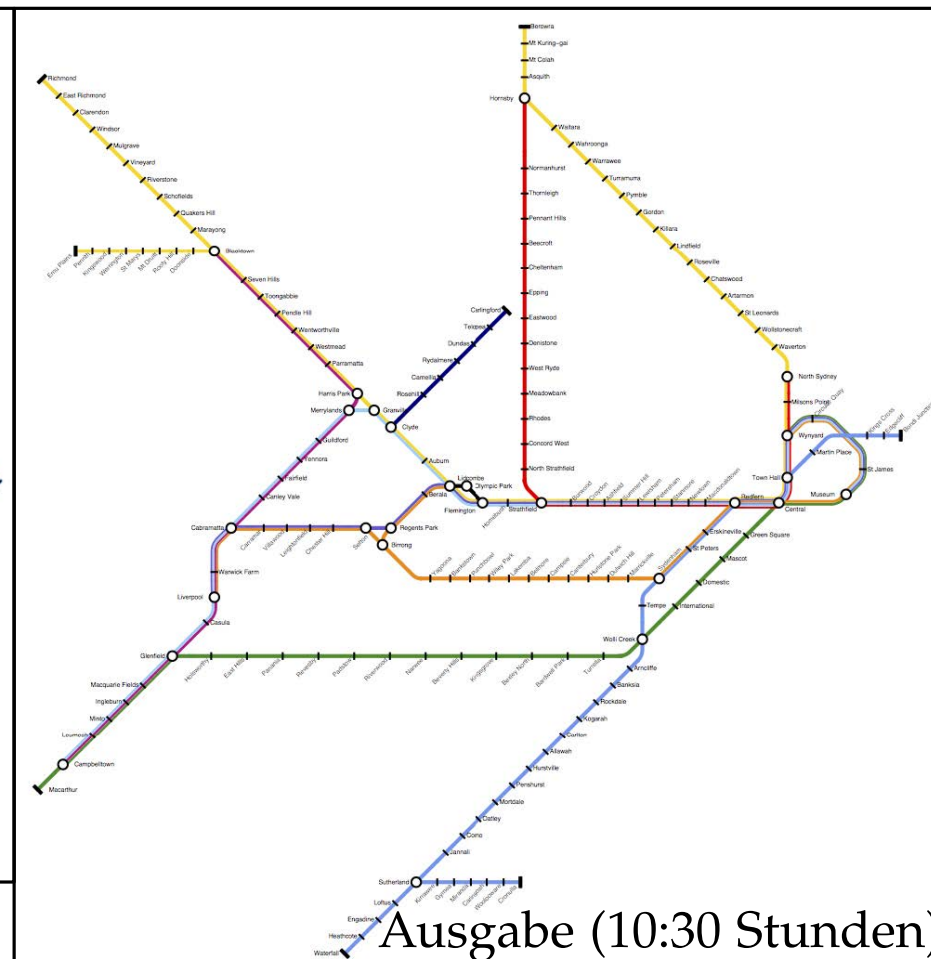
Beispiel: Sydney



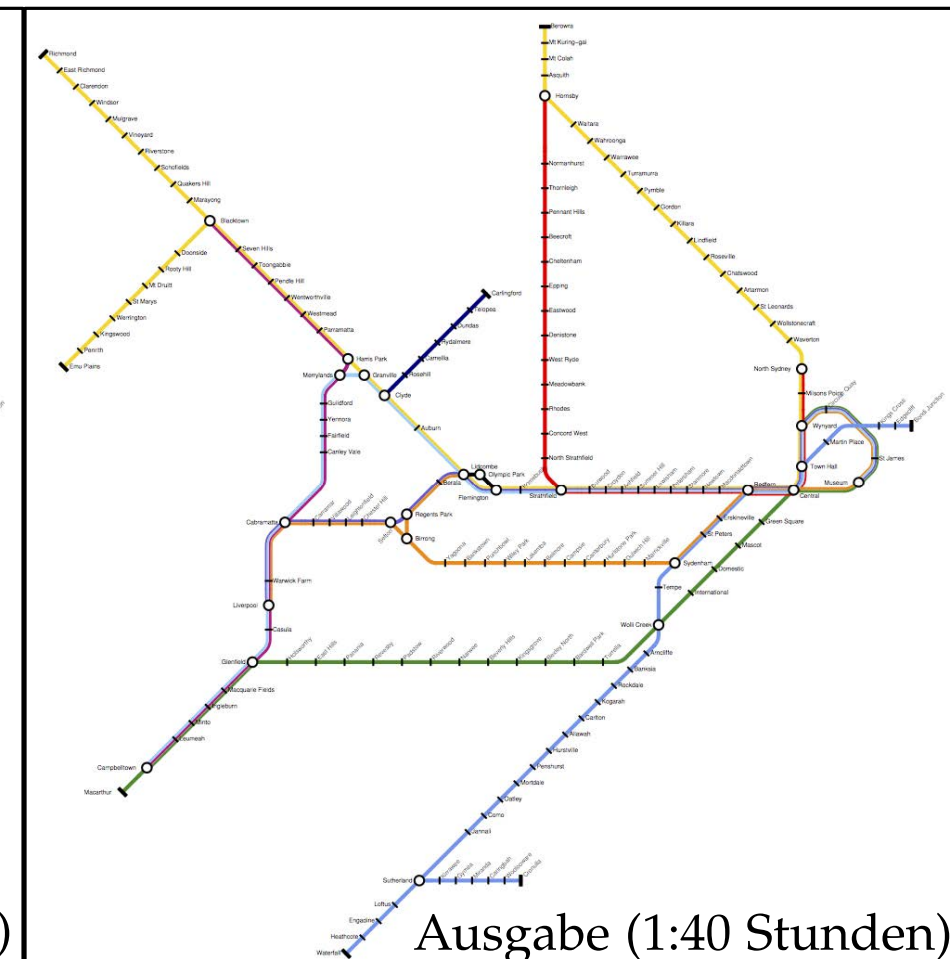
Beispiel: Sydney



Eingabe

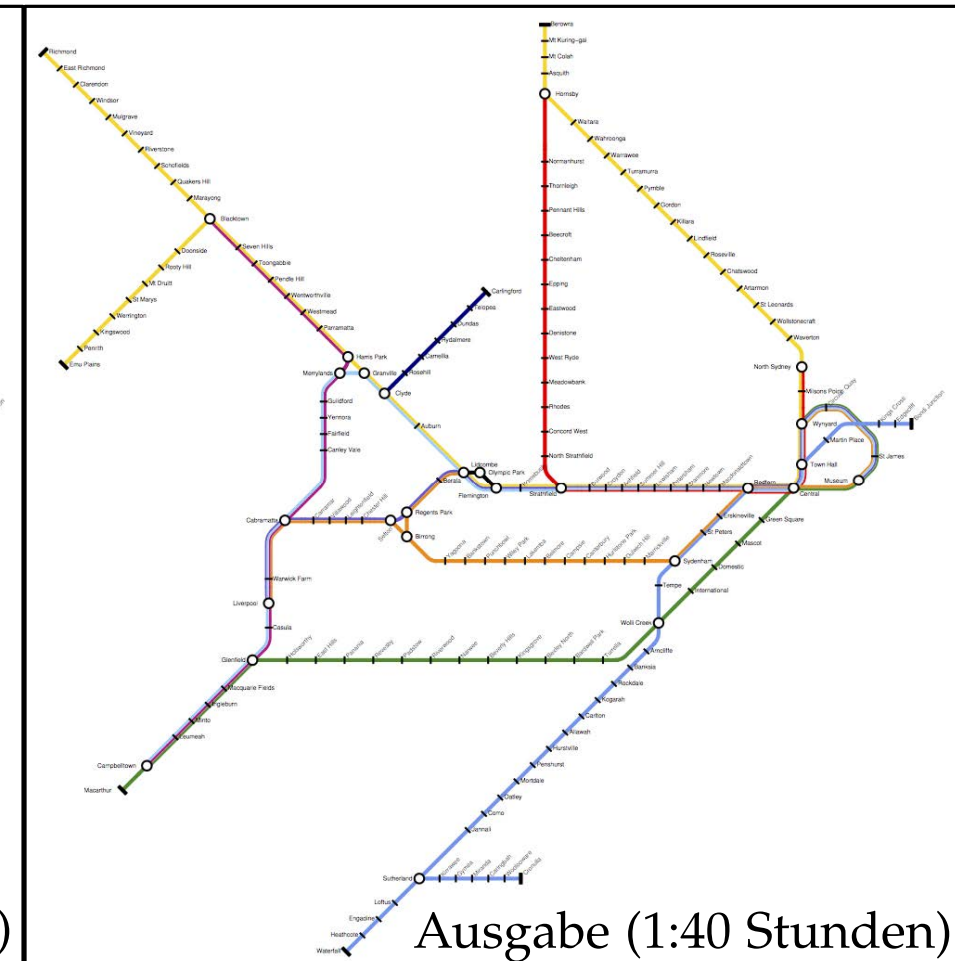
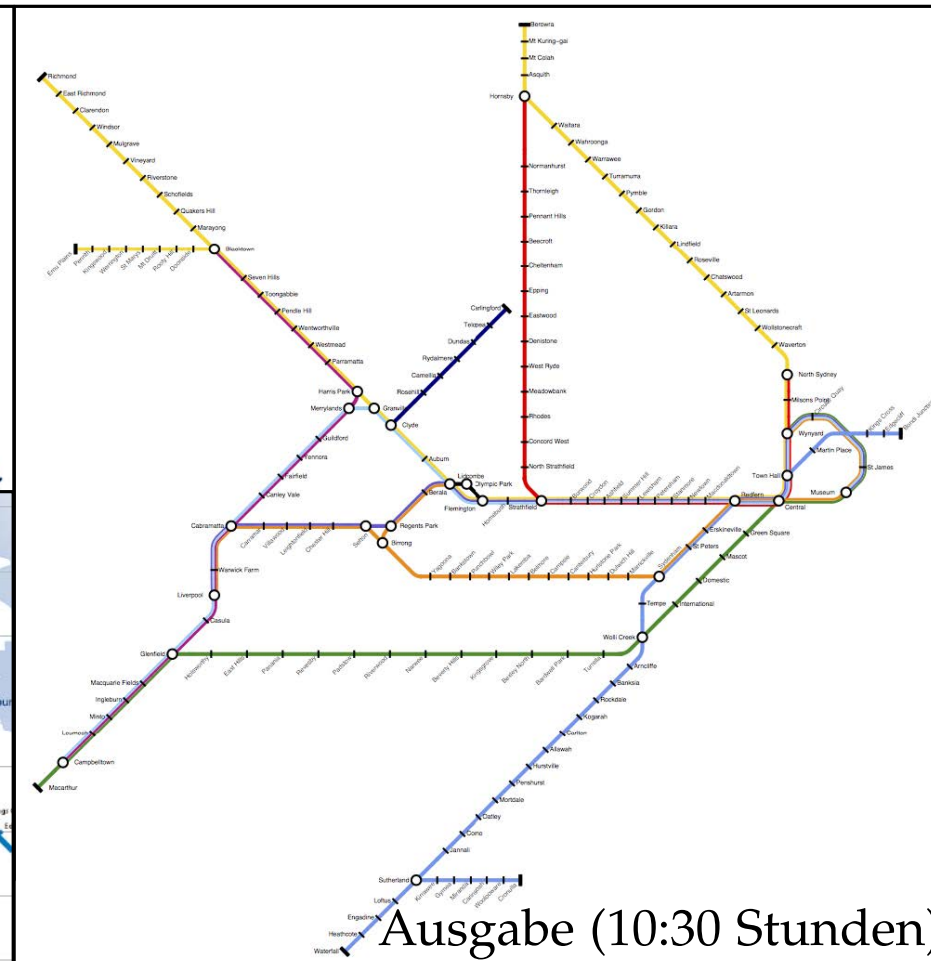
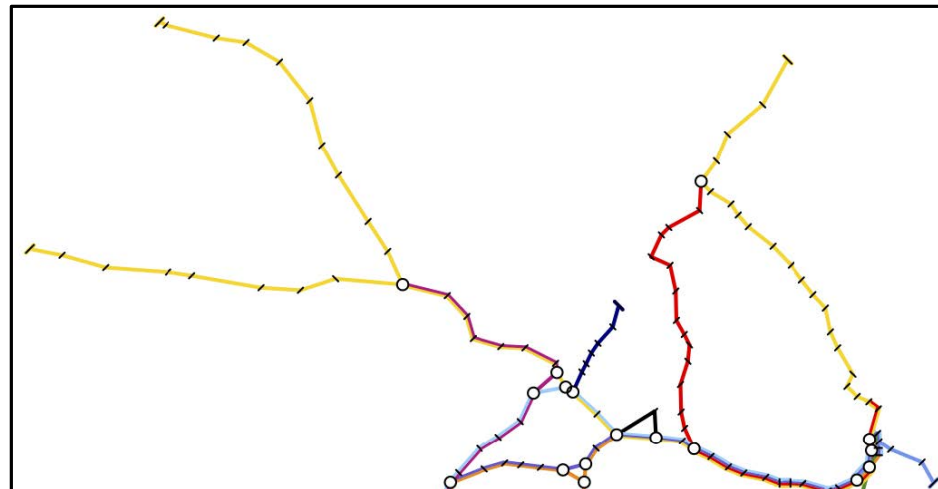


Ausgabe (10:30 Stunden)



Ausgabe (1:40 Stunden)

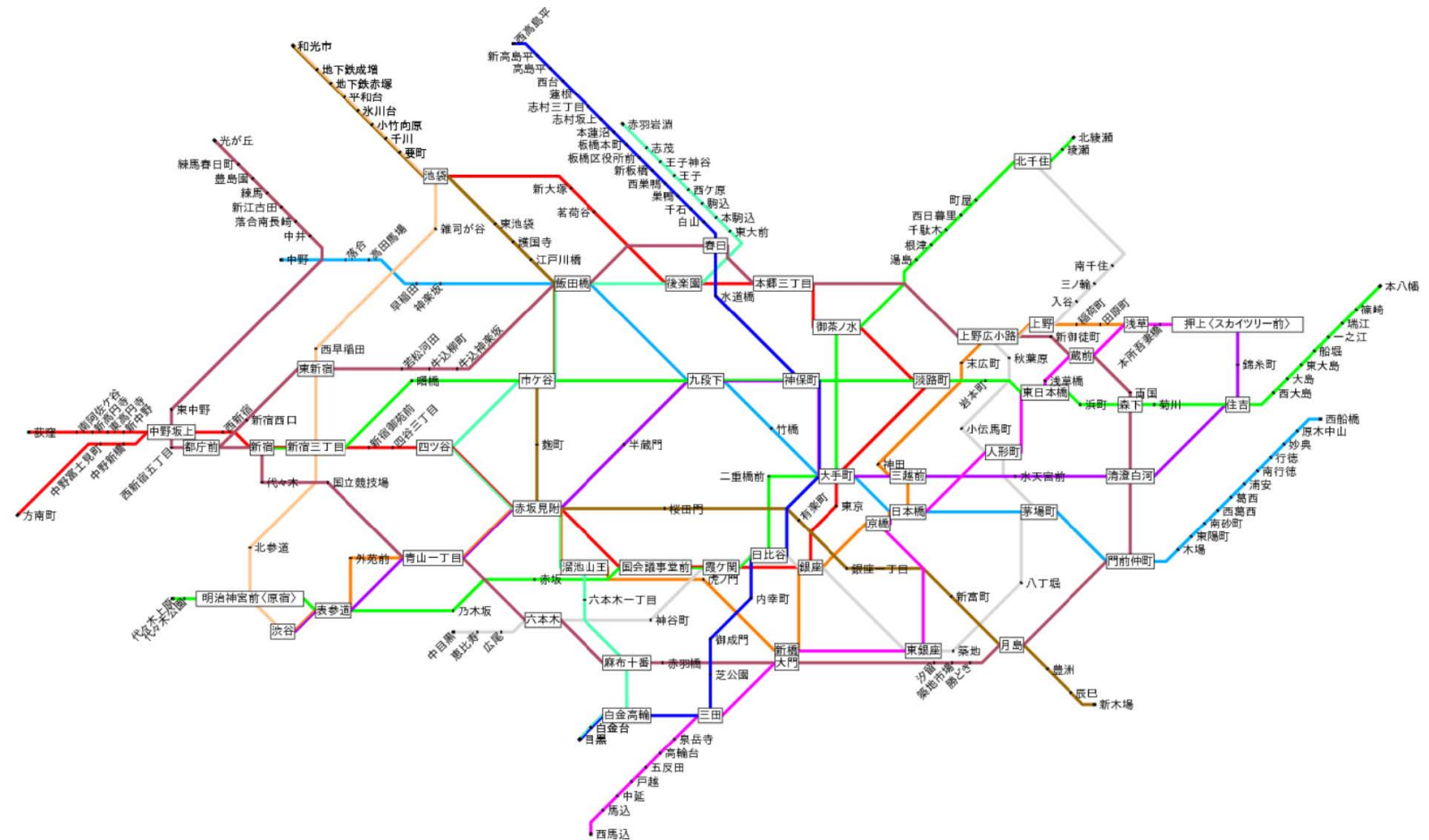
Beispiel: Sydney



Beispiel: Sydney & Tokyo



Tokyo



■ Multi-Phasen MIP Modell [Onda et al. 2018]

5 Stunden

Mixed-Integer Programm – Diskussion



Vorteile.

Nachteile.

Mixed-Integer Programm – Diskussion



Vorteile.

- flexibles Framework, aber Integration und Linearisierung neuer Kriterien erfordert viel Aufwand

Nachteile.

Mixed-Integer Programm – Diskussion



Vorteile.

- flexibles Framework, aber Integration und Linearisierung neuer Kriterien erfordert viel Aufwand
- hohe Layout- und Beschriftungsqualität

Nachteile.

Mixed-Integer Programm – Diskussion



Vorteile.

- flexibles Framework, aber Integration und Linearisierung neuer Kriterien erfordert viel Aufwand
- hohe Layout- und Beschriftungsqualität
- theoretische Garantien

Nachteile.



Mixed-Integer Programm – Diskussion

Vorteile.

- flexibles Framework, aber Integration und Linearisierung neuer Kriterien erfordert viel Aufwand
- hohe Layout- und Beschriftungsqualität
- theoretische Garantien
- kann Benutzereinschränkungen dynamisch integrieren

Nachteile.



Mixed-Integer Programm – Diskussion

Vorteile.

- flexibles Framework, aber Integration und Linearisierung neuer Kriterien erfordert viel Aufwand
- hohe Layout- und Beschriftungsqualität
- theoretische Garantien
- kann Benutzereinschränkungen dynamisch integrieren

Nachteile.

- lange, manchmal unvorhersehbare Laufzeiten



Mixed-Integer Programm – Diskussion

Vorteile.

- flexibles Framework, aber Integration und Linearisierung neuer Kriterien erfordert viel Aufwand
- hohe Layout- und Beschriftungsqualität
- theoretische Garantien
- kann Benutzereinschränkungen dynamisch integrieren

Nachteile.

- lange, manchmal unvorhersehbare Laufzeiten
- für große, beschriftete Netzwerke keine Garantie für Optimalität



Mixed-Integer Programm – Diskussion

Vorteile.

- flexibles Framework, aber Integration und Linearisierung neuer Kriterien erfordert viel Aufwand
- hohe Layout- und Beschriftungsqualität
- theoretische Garantien
- kann Benutzereinschränkungen dynamisch integrieren

Nachteile.

- lange, manchmal unvorhersehbare Laufzeiten
- für große, beschriftete Netzwerke keine Garantie für Optimalität
- Lösungen sind nur so gut wie die Modell-Spezifikation

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.
- vViele Ansätze sind anpassbar und offen für neue Kriterien

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.
- vViele Ansätze sind anpassbar und offen für neue Kriterien
- aktueller Trend: über Oktilinearität hinaus

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.
- vViele Ansätze sind anpassbar und offen für neue Kriterien
- aktueller Trend: über Oktilinearität hinaus

Warum automatisierte Karten?

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.
- vViele Ansätze sind anpassbar und offen für neue Kriterien
- aktueller Trend: über Oktilinearität hinaus

Warum automatisierte Karten?

- Basis-Layouts für Grafikdesigner (semi-automatisierter Prozess)

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.
- vViele Ansätze sind anpassbar und offen für neue Kriterien
- aktueller Trend: über Oktilinearität hinaus

Warum automatisierte Karten?

- Basis-Layouts für Grafikdesigner (semi-automatisierter Prozess)
- große Mengen an individuellen oder spezialisierten Karten

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.
- vViele Ansätze sind anpassbar und offen für neue Kriterien
- aktueller Trend: über Oktilinearität hinaus

Warum automatisierte Karten?

- Basis-Layouts für Grafikdesigner (semi-automatisierter Prozess)
- große Mengen an individuellen oder spezialisierten Karten

Herausforderungen

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.
- vViele Ansätze sind anpassbar und offen für neue Kriterien
- aktueller Trend: über Oktilinearität hinaus

Warum automatisierte Karten?

- Basis-Layouts für Grafikdesigner (semi-automatisierter Prozess)
- große Mengen an individuellen oder spezialisierten Karten

Herausforderungen

- globale Qualitätskriterien wie Harmonie, Kohärenz, Balance

Überblick



- eine Vielzahl von Layout-Methoden hat sich in den letzten 15-20 Jahren entwickelt.
- viele gemeinsame Design Regeln
- Kompromiss zwischen Geschwindigkeit und Qualität, aber durchaus vernünftige Karten können in wenigen Sekunden bis Minuten berechnet werden.
- vViele Ansätze sind anpassbar und offen für neue Kriterien
- aktueller Trend: über Oktilinearität hinaus

Warum automatisierte Karten?

- Basis-Layouts für Grafikdesigner (semi-automatisierter Prozess)
- große Mengen an individuellen oder spezialisierten Karten

Herausforderungen

- globale Qualitätskriterien wie Harmonie, Kohärenz, Balance
- Kantenbündel und große Knoten