

Automatic Team Division for PSE

Einführung ILP-Modellierung

LEHRSTUHL PROGRAMMIERPARADIGMEN



Mathematisch

Gegeben:

$$\vec{c} \in \mathbb{R}^n, \mathbf{A} \in \mathbb{R}^{m \times n}, \vec{b} \in \mathbb{R}^m$$

Gesucht:

$$x_{sol}^{\vec{}} = \max_{\vec{x} \in \mathbb{Z}^n} \left\{ \vec{c} \cdot \vec{x} \mid \mathbf{A}\vec{x} \leq \vec{b}, \vec{x} \geq \vec{0} \right\}$$

Optimierungsfunktion $f(\vec{x}) = \vec{c} \cdot \vec{x} = c_1 x_1 + c_2 x_2 \cdots + c_n x_n$

Nebenbedingungen $\mathbf{A}\vec{x} \leq \vec{b}$

Informell

Gegeben:

Lineare Funktion f mit n Variablen, m (lineare) Ungleichungen als Nebenbedingungen

Gesucht:

Belegung der n Variablen mit Ganzzahlen, so dass f maximal ist und die Nebenbedingungen alle erfüllt sind.

Dürfen einige der n Variablen auch Werte aus \mathbb{R} annehmen, spricht man auch von einem Mixed Integer Program (MIP).

Zentrale Frage

Wie müssen **A**, b , c , sowie n und m gewählt werden, dass das „richtige“ Problem gelöst wird?

Zentrale Frage

Wie müssen **A**, b , c , sowie n und m gewählt werden, dass das „richtige“ Problem gelöst wird?

Allgemein:

Harte Nebenbedingungen in **A** und b

Beispiel: Jeder Student in maximal ein Team

Weiche Nebenbedingungen in c (aka f)

Beispiel: Lerngruppen sollen erhalten bleiben

Dimensionen n und m ergeben sich aus der jeweiligen Instanz.

Welche Variablen?

Beispiel: Student M ist in Team Y

- Assoziiere x_k mit M , Wert von x_k ist Y .

Welche Variablen?

Beispiel: Student M ist in Team Y

- Assoziiere x_k mit M , Wert von x_k ist Y .

Anzahl Studenten in Team 2?

Welche Variablen?

Beispiel: Student M ist in Team Y

- Assoziiere x_k mit M , Wert von x_k ist Y . **Schlecht!**

Anzahl Studenten in Team 2?

Beispiel: Student M ist in Team Y

- Assoziiere x_k mit M , Wert von x_k ist Y . **Schlecht!**
- Assoziiere x_k mit (M, Y) , Wert von x_k ist 0 (Nein) oder 1 (Ja).

Anzahl Studenten in Team 2?

Beispiel: Student M ist in Team Y

- Assoziiere x_k mit M , Wert von x_k ist Y . **Schlecht!**
- Assoziiere x_k mit (M, Y) , Wert von x_k ist 0 (Nein) oder 1 (Ja). **Gut!**

Anzahl Studenten in Team 2?

$$\sum_M (M, 2)$$

Beispiel: Student M ist in Team Y

- Assoziiere x_k mit M , Wert von x_k ist Y . **Schlecht!**
- Assoziiere x_k mit (M, Y) , Wert von x_k ist 0 (Nein) oder 1 (Ja). **Gut!**
Aber: #Teams mal so viele Variablen!

Anzahl Studenten in Team 2?

$$\sum_M (M, 2)$$

Beispiel: Student M ist in Team Y

- Assoziiere x_k mit M , Wert von x_k ist Y . **Schlecht!**
- Assoziiere x_k mit (M, Y) , Wert von x_k ist 0 (Nein) oder 1 (Ja). **Gut!**
Aber: #Teams mal so viele Variablen!

Anzahl Studenten in Team 2?

$$\sum_M (M, 2)$$

Nebenbedingung $x_k \leq 1$ nicht vergessen!

Beispiel: Student M ist in Team Y

- Assoziiere x_k mit M , Wert von x_k ist Y . **Schlecht!**
- Assoziiere x_k mit (M, Y) , Wert von x_k ist 0 (Nein) oder 1 (Ja). **Gut!**
Aber: #Teams mal so viele Variablen!

Anzahl Studenten in Team 2?

$$\sum_M (M, 2)$$

Nebenbedingung $x_k \leq 1$ nicht vergessen!

⇒ *Entscheidungen* mit *binären* Variablen modellieren.

Beispiel: Teamgröße als eigene Variable

- Führe für jedes Team t neue Variable x_t für die jeweilige Größe ein.
- Je zwei Nebenbedingungen:

$$\sum_M (M, t) - x_t \leq 0$$

$$x_t - \sum_M (M, t) \leq 0$$

⇒ Für den ILP-Löser eigentlich unnötig, erhöht aber die Lesbarkeit!

Beispiel: Jeder Student wird in maximal 1 Team eingeteilt

Beispiel: Jeder Student wird in maximal 1 Team eingeteilt

Füge für jeden Student M die folgende Nebenbedingung hinzu:

$$\sum_t (M, t) \leq 1$$

UND-Verknüpfung

Wie modelliert man $x_1 \wedge x_2 \wedge \dots \wedge x_n$?

ODER-Verknüpfung

Wie modelliert man $x_1 \vee x_2 \vee \dots \vee x_n$?

UND-Verknüpfung

Wie modelliert man $x_1 \wedge x_2 \wedge \dots \wedge x_n$?

- Führe neue binäre Variable y (für das Ergebnis) ein.
- Zwei Nebenbedingungen:

$$\left(\sum_{1 \leq k \leq n} x_k \right) - ny \leq n - 1$$

$$ny - \left(\sum_{1 \leq k \leq n} x_k \right) \leq 0$$

ODER-Verknüpfung

Wie modelliert man $x_1 \vee x_2 \vee \dots \vee x_n$?

UND-Verknüpfung

Wie modelliert man $x_1 \wedge x_2 \wedge \dots \wedge x_n$?

ODER-Verknüpfung

Wie modelliert man $x_1 \vee x_2 \vee \dots \vee x_n$?

- Führe neue binäre Variable y (für das Ergebnis) ein.
- Zwei Nebenbedingungen:

$$ny - \left(\sum_{1 \leq k \leq n} x_k \right) \leq n - 1$$

$$\left(\sum_{1 \leq k \leq n} x_k \right) - ny \leq 0$$

Beispiel: Studenten-Glück

- Konstante Faktoren für Bewertungen C_{++} , C_{+} , C_{\sim} , ...
- Addiere auf Optimierungsfunktion:

$$\left(\sum_{\{(M,t)|M \text{ hat } t \text{ mit } ++ \text{ bewertet}\}} C_{++}(M,t) \right) +$$
$$\left(\sum_{\{(M,t)|M \text{ hat } t \text{ mit } + \text{ bewertet}\}} C_{+}(M,t) \right) +$$

...

Negative Faktoren möglich

⇒ Student lieber in „--“-Team oder lieber nicht einteilen?

Aktuell so implementiert:

- Betrachte Lerngruppe lg
- Berechne für jedes Team t binäre Variablen

$$a_t = \bigwedge_{\{M \mid M \text{ ist in } lg\}} (M, t) \vee (M, -1)$$

$$b_t = \bigvee_{\{M \mid M \text{ ist in } lg\}} (M, t)$$

$$v_t = a_t \wedge b_t$$

- Addiere *Lerngruppen-Bonus* mal v_t auf Optimierungsfunktion

v_t ist 1, gdw. alle Mitglieder von lg in Team t oder nicht eingeteilt (-1) sind.

Wie weiche Nebenbedingungen priorisieren?

Beispiel: Lerngruppen nicht splitten wichtiger als Studentenglück.

Aktuell

Parameter „von Hand“ anpassen

Idee

- Obere (untere) Schranke für modellierte Teilterme berechnen/abschätzen
- Terme *normalisieren* \implies Alle Bedingungen gleich wichtig
- Terme entsprechend der Priorität *skalieren* (Schrittweite/Größenordnung?)
- Abbruchkriterium für Löser anpassen

Überlegung: Welche *Bedeutung* haben die jeweiligen Teilterme?

Allgemeine Hinweise

- Modellierungs-Code kommentieren!

- Modellierungs-Code kommentieren!
- Verwende „sprechende“ Variablennamen
Beispiel: s_1234567_t_10 – Student mit M.Nr. 1234567 ist in Team 10

- Modellierungs-Code kommentieren!
- Verwende „sprechende“ Variablennamen
Beispiel: `s_1234567_t_10` – Student mit M.Nr. 1234567 ist in Team 10
- Modellierungs-Code kommentieren!

- Modellierungs-Code kommentieren!
- Verwende „sprechende“ Variablennamen
Beispiel: `s_1234567_t_10` – Student mit M.Nr. 1234567 ist in Team 10
- Modellierungs-Code kommentieren!
- Team mit nicht eingeteilten Studenten verwalten

- Modellierungs-Code kommentieren!
- Verwende „sprechende“ Variablennamen
Beispiel: `s_1234567_t_10` – Student mit M.Nr. 1234567 ist in Team 10
- Modellierungs-Code kommentieren!
- Team mit nicht eingeteilten Studenten verwalten
- Modellierungs-Code kommentieren!

- Modellierungs-Code kommentieren!
- Verwende „sprechende“ Variablennamen
Beispiel: `s_1234567_t_10` – Student mit M.Nr. 1234567 ist in Team 10
- Modellierungs-Code kommentieren!
- Team mit nicht eingeteilten Studenten verwalten
- Modellierungs-Code kommentieren!
- Wie mit leeren Teams umgehen?

- Modellierungs-Code kommentieren!
- Verwende „sprechende“ Variablennamen
Beispiel: `s_1234567_t_10` – Student mit M.Nr. 1234567 ist in Team 10
- Modellierungs-Code kommentieren!
- Team mit nicht eingeteilten Studenten verwalten
- Modellierungs-Code kommentieren!
- Wie mit leeren Teams umgehen?
- Modellierungs-Code kommentieren!

- Modellierungs-Code kommentieren!
- Verwende „sprechende“ Variablennamen
Beispiel: s_1234567_t_10 – Student mit M.Nr. 1234567 ist in Team 10
- Modellierungs-Code kommentieren!
- Team mit nicht eingeteilten Studenten verwalten
- Modellierungs-Code kommentieren!
- Wie mit leeren Teams umgehen?
- Modellierungs-Code kommentieren!
- Manchmal bessere Performance durch unnötige Constraints
(Hinweise an den Solver)
Beispiel: Anzahl ungesplitteter Lerngruppen, die eingeteilt wurden, ist kleiner-gleich der Anzahl Lerngruppen, die sich angemeldet haben.

- Modellierungs-Code kommentieren!
- Verwende „sprechende“ Variablennamen
Beispiel: s_1234567_t_10 – Student mit M.Nr. 1234567 ist in Team 10
- Modellierungs-Code kommentieren!
- Team mit nicht eingeteilten Studenten verwalten
- Modellierungs-Code kommentieren!
- Wie mit leeren Teams umgehen?
- Modellierungs-Code kommentieren!
- Manchmal bessere Performance durch unnötige Constraints
(Hinweise an den Solver)
Beispiel: Anzahl ungesplitteter Lerngruppen, die eingeteilt wurden, ist kleiner-gleich der Anzahl Lerngruppen, die sich angemeldet haben.
- Modellierungs-Code kommentieren!

- Gurobi kennt binäre Variablen
⇒ Keine Nebenbedingungen ≥ 0 und ≤ 1 nötig.
- Auch beliebige upper/lower bounds schon bei der Erzeugung von Variablen möglich.
- Nebenbedingungen können auch mit $=$ oder \geq angegeben werden.
- Optimierungsfunktion und Nebenbedingungen können *nach und nach* zum Modell hinzugefügt werden.

Es gibt Java-Bibliotheken zum Modellieren von ILP Instanzen!

- Modellierung Backend unabhängig
⇒ Solver austauschbar (Gurobi, CPLEX, GLPK, ...)
- Aber vorher prüfen, ob mächtig genug
⇒ Welche Solver verfügbar? Zwischenergebnis abrufbar?
Konfigurationsmöglichkeiten? Klassenhierarchie kompatibel? etc.

ToDoS

- Gurobi/CPLEX installieren
- Einfachstes Beispiel rechnen (siehe jew. Userguide) – ohne Java?
- harte Nebenbedingung „min/max Teamgröße“ modellieren
- weiche Nebenbedingung „3. Semester und höhere nicht mischen“ modellieren