

Tunnel-basiertes Pruning für klassische Handlungsplanung

Bachelorarbeit - Abschlusspräsentation

Daniel Federau

09. Februar 2015

Klassische Handlungsplanung

- Finden einer Sequenz von Aktionen von einem Anfangszustand zu einem Ziel
- Plan: Reihe von ausführbaren Aktionen vom Anfangszustand zu einem Zielzustand
- optimale Planer liefern Pläne mit minimalen Gesamtkosten

SAS+ Formalismus

Definiert durch 5-Tupel: $\Pi = \langle \mathcal{V}, s_0, s_*, \mathcal{A}, cost \rangle$

- $\mathcal{V} = \{v_1, \dots, v_n\}$: Menge von Variablen
- s_0 : Anfangszustand
- s_* : Menge von Zielzuständen
- \mathcal{A} : Menge von Aktionen. Für jede Aktion gilt:
 $a = \langle pre(a), prevail(a), eff(a) \rangle \in \mathcal{A}$
pre-post mit (v, p, p') : Variable v kommt in $pre(a)$ und $eff(a)$ vor
- *cost*: Gibt für jede Aktion nicht negative Kosten an

effektivere Suche

- bei der Suche eines Plans müssen eventuell eine grosse Menge an Zuständen besucht werden
- kann zu langen Berechnungszeiten führen
- viele verschiedene Methoden um Planer zu verbessern, z.B. :
 - Heuristiken
 - Thema dieser Arbeit: *Pruning*

Heuristische Suche

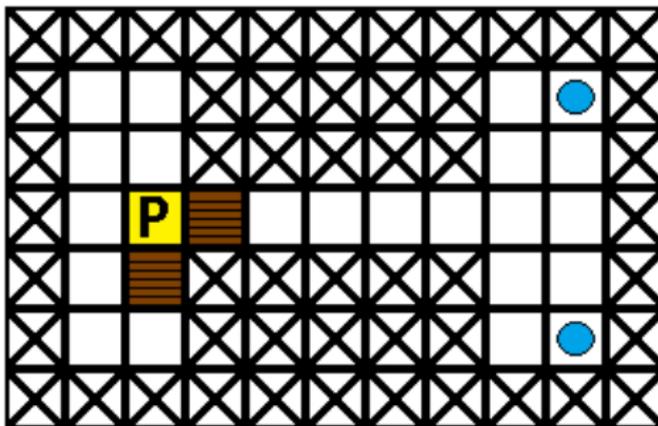
- gezielteres Suchen nach einem Zielzustand
- benutzt heuristische Funktion h um nächsten Zustand zu finden
- h schätzt für jeden Zustand die Entfernung zum Ziel
- A* ist ein berühmter Algorithmus der Heuristiken verwendet
- A* verwendet $f(s) = g(s) + h(s)$ als Kostenfunktion

Pruning

- versucht Anzahl der besuchten Zustände zu verkleinern
- zwei Methoden werden hier näher vorgestellt
- aus dem Paper “Tunneling and Decomposition-Based State Reduction for Optimal Planning“ von Raz Nissim, Udi Apsel, Ronen Brafman
- beide verwenden letzte Aktion die zum jetzigen Zustand geführt hat
- letzte Aktion muss bestimmte Kriterien erfüllen um Zustände zu reduzieren

Idee

- wurde erstmal in einem Planer für Sokoban verwendet



- Suche Aktionen, die einen Tunnel erlauben und prune alle Aktionen, die nicht im Tunnel sind
- Hier: Generalisierung dieser Methode

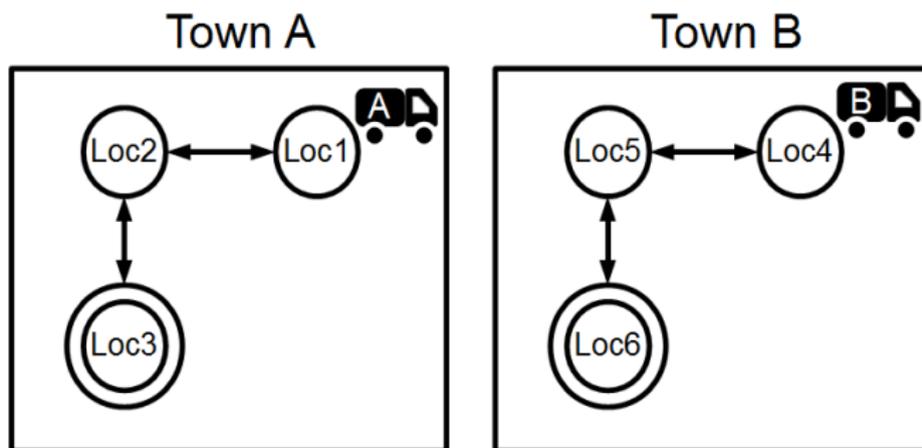
Ablauf

- Vor der Suche:
 1. Finde alle Aktionen die einen Tunnel zulassen
 2. Für jede dieser Aktionen finde alle Aktionen die zu diesem Tunnel gehören
- Während der Suche:
 3. Überprüfe, ob letzte Aktion a einen Tunnel erlaubt
 4. Wenn ja, dann prune alle Aktionen die nicht im Tunnel von a sind

1. Erlaubt Aktion a einen Tunnel?

- a muss eine Zielvariable schreiben, die noch nicht erfüllt ist
- Alle Aktionen a' die eine pre mit $(v, p) \in eff(a)$ besitzen müssen auch erfüllen:
 1. a' hat eine $pre-post$ mit (v, p, p') für die gilt: $(v, p) \in eff(a)$
 2. Es muss immer möglich sein a' nach a auszuführen.
 3. a' ändert die selben Variablen wie a
- Wenn eine Aktion a' eine dieser 3 Bedingungen nicht erfüllt, dann erlaubt a keinen Tunnel
- alle Aktionen a' , die diese Bedingungen erfüllen sind in $tunnel(a)$

Beispiel



- TruckA: $v1 = \{1, 2, 3\}$, TruckB: $v2 = \{4, 5, 6\}$
- Aktion a1: $pre(a1) = (v1, 1)$, $eff(a1) = (v1, 2)$
- Aktion a2: $pre(a2) = (v1, 2)$, $eff(a2) = (v1, 3)$

Idee

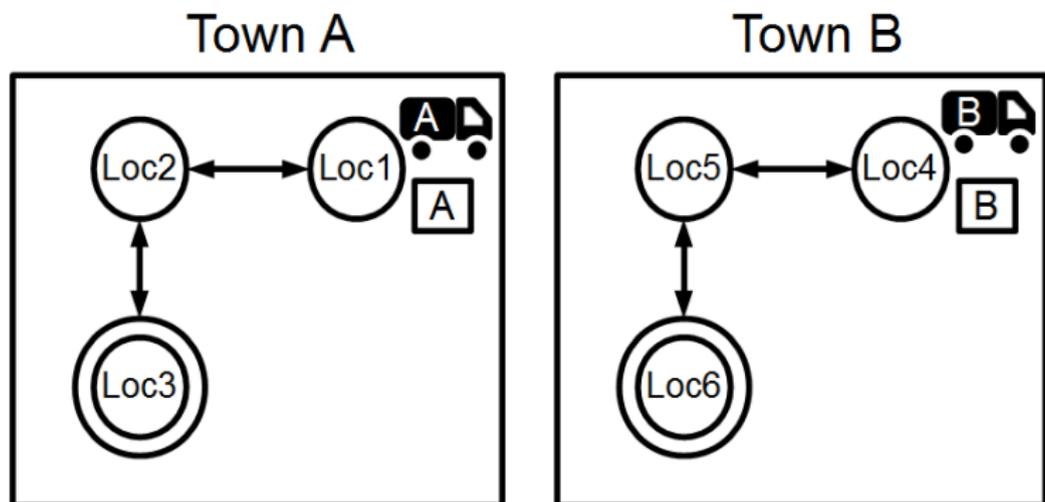
- Aufteilung des Problems in kleinere Teilprobleme
- Zuordnung aller Aktionen in Partitionen
- Aufteilung aller Aktionen in *öffentlich* und *privat*
- Eine Aktion ist öffentlich, wenn sie nicht *kommutativ* ist mit einer Variablen aus einer anderen Partition

Kommutativität

Zwei Aktionen a und a' sind *kommutativ*, wenn gilt:

1. Variable in $pre(a)$ und $pre(a')$ taucht nicht in $eff(a')$ auf
2. Variable in $pre(a)$ und $pre(a')$ taucht nicht in $eff(a)$ auf
3. Variable in $eff(a)$ taucht nicht in $eff(a')$ auf

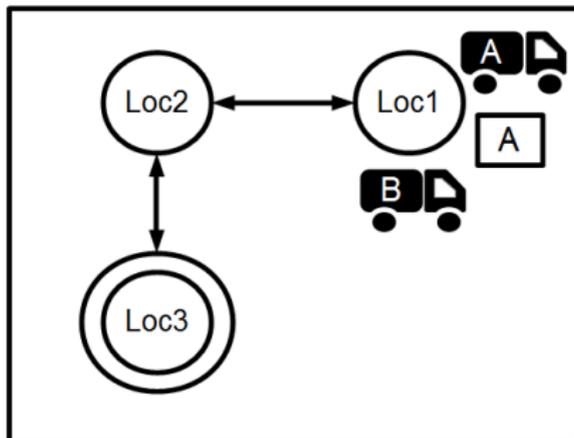
Beispiel aus der Logistik (1)



- Alle Aktionen mit einem Truck in eine Partition
- Alle Aktionen sind privat

Beispiel aus der Logistik (2)

Town A



- Alle Aktionen mit einem Truck in eine Partition
- Alle *move*-Aktionen sind privat, *load* und *unload* sind öffentlich

Ablauf

- Vor der Suche:
 1. Partitionierung aller Aktionen
 2. Finde heraus, ob Aktion *öffentlich* oder *privat* ist
- Während der Suche:
 1. Überprüfe, ob letzte Aktion *a privat* ist
 2. Wenn ja, dann prune alle Aktionen die nicht in der Partition von *a* sind

Partitionierung

- je besser die Partitionierung, desto öfter können Zustände gepruned werden
- viele verschiedene Möglichkeiten um Partitionierung festzulegen
- Implementierte Methode:
 1. zufällige Aufteilung der Aktionen
 2. Iterative Verbesserung der Partitionierung durch Vertauschen von Aktionen
- wegen Zeitmangels konnten keine besseren Partitionierungen implementiert werden

Tunnel-basiertes Pruning - Zusammenfassung

Zusammenfassung	A*-blind	A*-blind + T
Anzahl gelöster Probleme	519	524
Evaluationen - geometrisches Mittel	94520.69	89966.54
Expansionen - geometrisches Mittel	63373.28	60559.96
generierte Zustände - geometrisches Mittel	394897.81	369708.18
Suchzeit - geometrisches Mittel	0.58	0.58
gesamte Zeit - geometrisches Mittel	0.64	0.63

Tunnel-basiertes Pruning - Domänen

	Evaluationen		Expansionen	
	A*-blind	A*-blind+T	A*-blind	A*-blind+T
driverlog (7)	240401.85	220189.62	157819.23	144356.62
logistics00 (10)	44966.84	41146.45	36450.41	33842.89
logistics98 (2)	266071.07	263554.35	201674.32	200381.33
nomystery-opt11-strips (8)	75588.70	12394.16	61402.55	11518.43
psr-small (49)	1132.29	999.96	996.78	893.63
satellite (5)	36461.39	36461.34	27364.43	27364.34
zenotravel (8)	23195.52	23082.08	11321.26	11321.26

Partition-basiertes Pruning - Zusammenfassung

Zusammenfassung	A*-blind	A*-blind + PB
Anzahl gelöster Probleme	519	471
Evaluationen - geometrisches Mittel	73046.78	72626.14
Expansionen - geometrisches Mittel	49965.87	49696.35
generierte Zustände - geometrisches Mittel	317640.78	308234.08
Suchzeit - geometrisches Mittel	0.42	25.72
gesamte Zeit - geometrisches Mittel	0.46	25.73

Partition-basiertes Pruning - Domänen

generierte Zustände	A*-blind	A*-blind + PB
pathways-noneg (4)	108196.54	67089.28
rovers (5)	50138.80	33874.08
satellite (5)	408445.14	304807.57
tpp (6)	3688.45	3465.07
woodworking-opt08-strips (7)	403869.63	400930.01

Zukünftige Verbesserungen

- Verwendung von Path-Pruning A*
- Bessere Methoden zur Partitionierung, z.B. :
 - lokale Suche
 - Graphen-Partitionierung
- Kombinieren beider Methoden

Vielen Dank für ihre Aufmerksamkeit!

Fragen?