

Tiebreaking Strategies for A* Search: How to Explore the Final Frontier

浅井 政太郎 (受付番号 10073) 東京大学 総合文化研究科

発表要旨

1. これまでの研究業績：査読付き学会論文

- (a) Asai, M. and Fukunaga, A. Fully Automated Cyclic Planning for Large-Scale Manufacturing Domains. In *Proceedings of the 24th International Conference of Automated Planning and Scheduling(ICAPS)*, June 2014.
- (b) Asai, M. and Fukunaga, A. Applying Problem Decomposition to Extremely Large Planning Domains. In *Proceedings of the 5th Workshop on Knowledge Engineering for Planning and Scheduling(KEPS)*, June 2014.
- (c) Asai, M. and Fukunaga, A. Solving Large-Scale Planning Problems by Decomposition and Macro Generation. In *Proceedings of the 25th International Conference of Automated*

Planning and Scheduling(ICAPS), June 2015.

- (d) Asai, M. and Fukunaga, A. Tiebreaking Strategies for Classical Planning Using A* Search. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, February 2016.

2. 今後の研究計画

- (a) 探索グラフの対称性の検知技術と、小問題分割技術の融合

1 研究テーマ: プランニング(自動行動計画)とは?

ロボットにいかに賢く自律行動させるか? 結果を先読み(推論)して行動
行動列を求める汎用組み合わせ最適化問題



例:災害救助

T-52援竜 - テムザック(株)
現時点では人がラジコン操作
プランナが組みこまれば

目標の達成手順を自力で
思考できる救助ロボ

初期状態: 被災者が瓦礫の下敷きに!
ゴール: 患者の生存 かつ 患者の搬送
出血多量確認 → 布で縛る
→ 病院に電話 → カッターを起動
→ 鉄筋を切断 → 瓦礫を持ち上げる
→ 患者を助け出す → 救急隊に引き渡し

出力: 全てのゴールを達成する
アクション列 = プラン

ある程度長い具体的な命令列

≡ プログラム、レシピ、手順 を生成する

- × プラン ≠ 直後に実行する1ステップの行為
- × プラン ≠ 漠然とした方向性, TODOリスト
- × プラン ≠ ゴールを一つずつ達成すれば良い

状態遷移規則で様々な問題を表現

入力: 初期状態、ゴール、アクション
(状態遷移)

否定 bleeding(man), on(rock1, man),
on(robot, road), ¬cutter_on()
hold(robot, nothing) alive(man)
on(man, ambulance) stopbleeding(?X)
¬ambulance_coming() remove(?X), call(?X)
start_cutter(), cut(?X)

プランニング分野の目標:

より複雑で大規模な問題を、汎用性を失わずに解きたい

整数計画、SATと類似の基礎的な枠組み

プランニングの 応用実績

汎用性
ゆえに
様々な
応用例



Mars Exploration Rover
(Bresina et al. '05)
火星探査ロボットの
自律行動システム

Deep Space 1
(Muscettola et al. '98)
宇宙船の自律航行



Xerox Parc Printer
大規模 商業印刷システムの
自律運用



Core Security (セキュリティ企業)

企業ネットワーク脆弱性の自動診断
攻撃プランの自動生成

研究テーマのプランニングは、ロボットに、人間の助けを借りず、いかに自律して行動させるかを扱います。賢いロボットは、自分の行動の結果をよく先読みして行動する必要があります。

これをモデル化したプランニング問題は、具体的な行動の列、手順を求める 組合せ最適化問題です。

たとえば、自動プランニング機能付きのレスキューロボットというものを考えましょう。プランニング問題のソルバは、ロボットが被災者を助ける正しい手順を出力します。

プランニングは、様々な問題を状態遷移規則を用いて柔軟に表現できます。例えば、

現実の応用例では「宇宙探査機運行問題」や「企業ネットワーク脆弱性問題」も表現できます。

このように、プランニングは、難しい問題を汎用性を失わずに解くことを目指します。

仮に将来、写真のような救助ロボットにプランニングソルバを組み込めば、被災者を発見した時に適切な行動を自ら選択できるようになるかもしれません。

2 プランニング(自動行動計画)分野の位置づけ

プランニング分野は、人工知能の専門分野という位置づけで、隣接するオペレーションズ・リサーチやアルゴリズム論などの分野の技術を利用しています。特に、プランニング問題を解くにはグラフ探索の技術が用いられます。

プランニングの計算複雑性クラス:
仮定を加えてもNP困難

PLANSAT は PSPACE完全

探索空間のサイズは命題に対して指数:
探索グラフの頂点数 = $2^{(\text{命題変数の数})}$
変数100個 $\rightarrow 2^{100}$ 120個 $\rightarrow 2^{120}$

計算量理論
アルゴリズム論
グラフ理論

コンパイル \equiv プランニング
機械語1命令 \equiv アクション

情報システム

プランニングのモデル言語 PDDL
の記述力は?

コンパイラ,
プログラミング言語

AI

論理・知識表現
進化計算 機械学習

探索アルゴリズム(A*)

プランニング
スケジューリング

プランニング分野の主要国際会議:

International Conference on Automated
Planning and Scheduling (ICAPS) (1994-),
および AAI, IJCAI

主要研究グループ: MIT CSAIL, CMU, NASA/
Caltech JPL, NASA Ames, 欧州宇宙機関
(ESA)など

SAT, 制約充足

整数計画

MDP, POMDP

OR
(オペレーションズ
リサーチ)

倉庫管理問題
ロジスティクス問題
電力網最適化

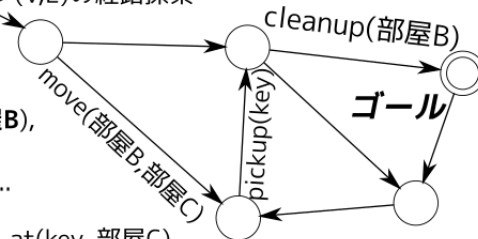
プランニングの主な手法:

頂点ノードV:状態, 辺E:アクション
である有向グラフ(V,E)の経路探索

初期状態

at(key, 部屋C),
at(ロボット, 部屋B),
 \neg clean(部屋A),
 \neg clean(部屋B)....

at(key, 部屋C),
at(ロボット, 部屋C)...



3 研究業績1：査読付き学会論文 ICAPS14

研究業績に移ります。ここでは、大規模なプランニング問題を解くために問題設定・ドメインによらず汎用に繰り返し構造を抽出する方法を開発しました。元の問題を繰り返し一周分の小問題に分割して解くことで、高速化と3割の生産時間短縮を達成しました。大切なのは汎用性です。生産問題に限らず、掃除にも宇宙船にも同じ実行バイナリが使えます。

セル生産方式: 工場生産方式の一種 (対義語: ライン生産)

ロボットアームなどを含む小規模セルが複数協調して製造

目的: 並列の製造プランを自動計画 + 製作時間を最小化
一台の組み立て問題を解くのは ある程度複雑でも可能

研究の対象

✕アクチュエータの角度制御 (例: θ : 0deg→30deg, 1deg/sec)

◎行為の計画(例: 部品2をアーム1で塗装, ネジ3を締結 etc.)

問題点: 扱う対象の数が増える場合 (製品が4個程度以上の時)

最適解を返すソルバで直接解く → ✕探索空間が広大で探索不可

1つ分の製造手順を複数つなげて後処理最適化 → ✕生産時間が長い (makespan)

手法: 問題の繰り返し構造を「工場, セル生産, 製品」など前提なしに

任意のプランニング問題の論理構造から自動検出

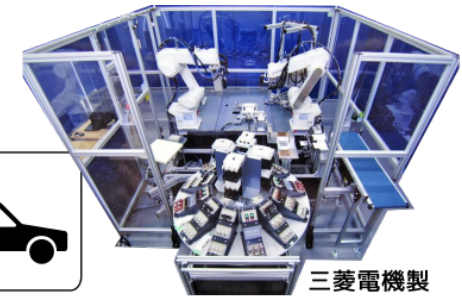
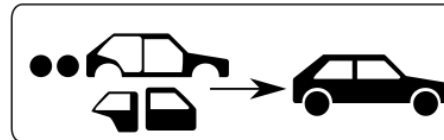
検出した繰り返し構造から効率よい (生産時間の短い) プランを生成

**結果: 1000台以上の製品を組み立てる問題を解けた
+ 生産時間を最大3割短縮(後処理最適化と比較)**

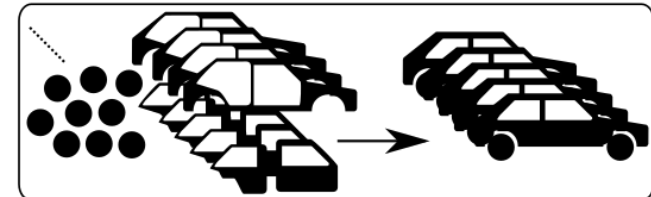
新規性: 繰り返し構造を自動で抽出するというアイデア+抽出技術

インパクト: 大きな高速化, プランニングの適用規模を広げた

株式会社IHIと共同研究(システムの先行評価として)



三菱電機製



詳細: lock/owner述語の検出+定常状態モデルの構成

0	1	2	3	4	index
(not in the system)	(b1)	(b2)	(b3)		MS ³ <0,2,3>

添字のみひとつ進んだ次の週の時点までプランを生成

探索空間のサイズは指数: 2^(命題変数の数)

変数100個 → 2¹⁰⁰ 120個 → 2¹²⁰

問題が少し大きくなるだけで百万倍難しくなる

→ しかも探索空間はその指数で増加
解ける探索空間のサイズ:

以前: 10⁶ → 提案手法: 10²⁷⁴

Asai, M. and Fukunaga, A. Fully Automated Cyclic Planning for Large-Scale Manufacturing Domains. In Proceedings of the 24th International Conference of Automated Planning and Scheduling(ICAPS), Portsmouth, NH, USA, June 21-26, 2014. (採択率33%)

4 研究業績2,3 : 査読付き学会論文 KEPS14, ICAPS15

続いて二、三本目の業績は、先ほどの手法で得られるのは1種類の小問題だけでしたが、これを複数種類の小問題に拡張しました。結果、より様々な問題で高速化を達成しました。ここまで汎用に小問題分割を適用した研究は、分野では始めてです。

プランニング分野一般の目標:

汎用性を失わずに より複雑な問題を解く! → 高度なAI

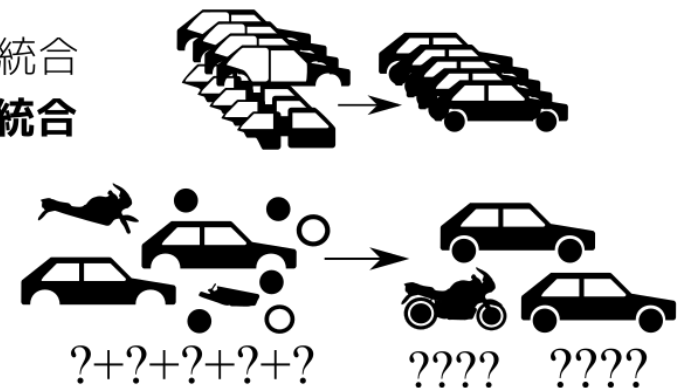
ICAPS14を一般化すればはるかに困難な問題が解けるはず

探索空間のサイズは指数: $2^{(\text{命題変数の数})}$
変数100個 → 2^{100} 120個 → 2^{120}
問題が少し大きくなるだけで百万倍難しくなる

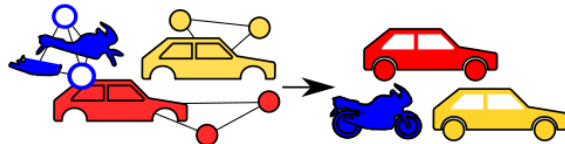
ICAPS14: 大規模な問題を同一種類の小問題に分割+処理+統合

目的: **大規模な問題を複数種類の小問題に分割+処理+統合**

課題: 1つの小問題の構成要素が不明: 自動で発見する能力が必要
分割/統合が本質的に不可能な問題でも動作する汎用性が必要
(問題分割を行うその他の既存研究は、その場合動作しない/想定していない)

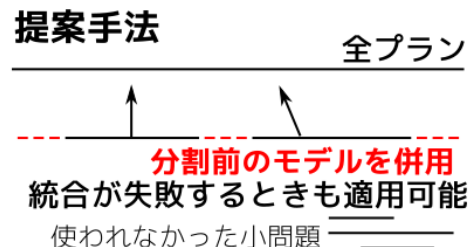
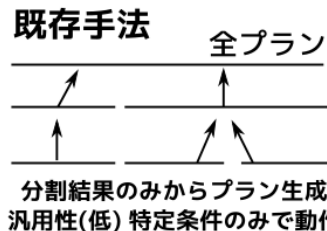


手法: **問題の論理構造から小問題を発見する新手法**



分割なしでは解けない問題
----- (探索失敗)

より柔軟な統合手法



結果: **広い問題種別で**

3-4倍変数の多い問題を解けた
解ける探索空間のサイズ:

以前: $\sim 10^7$ 提案手法: $\sim 10^{28}$

新規性: 小問題の発見手法 / 柔軟な統合手法
インパクト: 高速化, 小問題分割の汎用性を実証

Asai, M. and Fukunaga, A. Applying Problem Decomposition to Extremely Large Planning Domains. In Proceedings of the 5th Workshop on Knowledge Engineering for Planning and Scheduling(KEPS), Portsmouth, NH, June 2014.

Asai, M. and Fukunaga, A. Solving Large-Scale Planning Problems by Decomposition and Macro Generation. In Proceedings of the 25th International Conference of Automated Planning and Scheduling(ICAPS), Jerusalem, Israel, June 2015. (採択率33%)

5 研究業績4：査読付き学会論文 AAAI16 (申請後に採択)

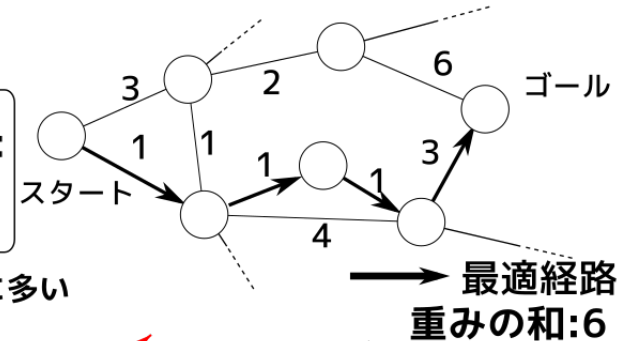
最後に、申請後に行った研究が、難関国際学会 AAAI に採択されました。研究内容は、コスト 0 の辺を含むグラフを扱うグラフ探索アルゴリズム一般に適用できる内容で、非常に大きなインパクトを持つことが考えられます。

グラフ探索一般に適用できる高速化手法の開発

グラフ探索: スタートからゴールまでの経路を探す
+ 通る辺の重みの和を最小化する

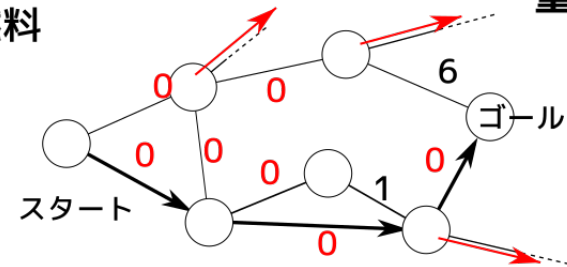
身近な例: **カーナビ** 但し、本研究の対象は
メモリには収まらない規模のグラフを扱う
例: **アミノ酸の多重配列問題(MSA)**など
プランニング問題もグラフ探索で解く

著名なグラフ探索アルゴリズム:
Dijkstra法 (1959), **A*(1968)**



複雑/実用的な探索問題には重み0の辺が大量にある 大量=指数的に多い
例: トラックの配達計画 (燃料を最小化) 辺=行為, 重み=燃料
→ 荷物の積み替えには燃料を使わない

問題点: 重み0 の辺が大量にある場合、
重みの和が同じノードが大量に存在し
探索を進める方向を見失う



オリジナリティ: A*(見積もり値を用いた最良優先探索(最良分枝限定法))にて

1. 重み和が同じノードの扱い方に関する**70年代からの定説を覆した**
2. 重み和が同じノードを更に分類する新手法を提案し性能向上

(A*のTiebreakingには
ゴールまでの距離の見積もり値 h を
流用すべきだとされてきた)

結果: 解ける問題規模が拡大 (ベンチマーク1104問:814問 → 867問 (+53問))

解ける問題の探索空間サイズ(例):

(企業ネットワーク脆弱性の診断問題にて)

以前: 10^6 提案手法: 10^{88}

今後のインパクト: 重み0の辺を持つ広範なグラフ探索問題に対して性能向上

Asai, M. and Fukunaga, A. Tiebreaking Strategies for Classical Planning Using A* Search. In Proceedings of the 30th AAAI Conference on Artificial Intelligence, Phoenix, Arizona, February 2016. (採択率26%)

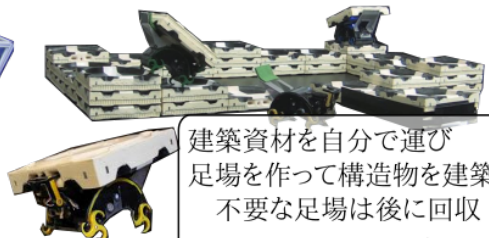
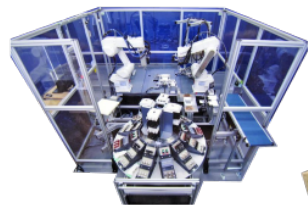
6 今後の研究計画

今後の研究計画です。今後の研究の目的は、より大規模な問題の求解です。その上で2つの課題があります。一つ目は、探索空間の対称性が問題を難しくすることです。これは、対称性を破ることで改善します。二つ目は、非対称な構造に対する扱いです。これには問題分割を適用します。ただし、これら個別の改善も重要ですが、研究の主眼は2つの融合にあります。その結果、より大規模な産業利用が可能になります。

目的: より大規模な問題を解く

大量生産問題

群ロボットによる高速な自動建築



建築資材を自分で運び
足場を作って構造物を建築
不要な足場は後に回収

MIT TERMES

課題: 非対称な問題 手法: 問題分割

課題: 問題分割と処理に時間がかかりすぎてしまう

手法: ポートフォリオ戦略

課題: 問題によっては分割が困難

手法: 本質的な構造は変えずに、

個別の
改善

課題: 分割して解いた結果を
再統合できない場合がある

手法: 解いた結果の正しさを保ったまま

再統合を妨げる要素を削除

分割しやすい論理構造に変換

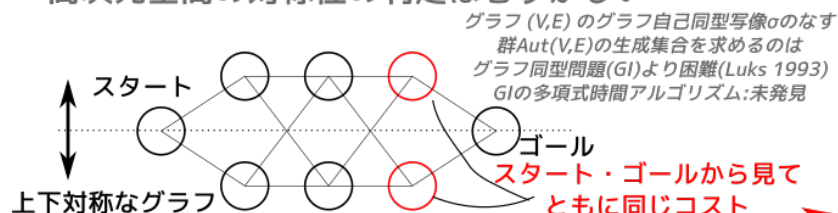
課題: 探索空間の対称性

- ・問題を著しく困難にする
- ・対称性は探索空間の指数増大を招く
- ・高次元空間の対称性の判定はむずかしい

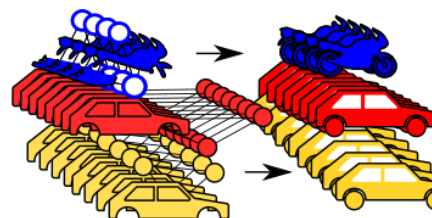
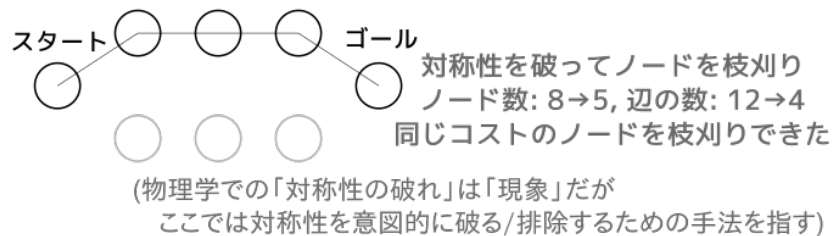
個別の
改善

課題: 対称性検知と問題分割の両立

対称性の検知 + 小問題に分割する



手法: 対称性の検知



将来のインパクト:

→ 巨大問題を求解可能に
→ 大規模な産業利用

注: 巨大問題だけが重要だとは主張しない
小規模な問題を高速に解く技術には
ロボティクスなどからの需要がある

現在の研究状況

AAAI-16: 重みゼロの辺で隣接する

同じコストのノードを効率的に探索した

→ 対称性を部分的に破っている? (今後の研究)

ICAPS-16 (査読中):

同種の小問題を検知して対称性を破壊

7 まとめ

1. これまでの研究業績：査読付き学会論文

- (a) 難関国際会議 Asai, M. and Fukunaga, A. Fully Automated Cyclic Planning for Large-Scale Manufacturing Domains. In *Proceedings of the 24th International Conference of Automated Planning and Scheduling(ICAPS)*, June 2014. 採択率 **33%**
 - i. 任意の問題から1種類の繰り返し構造を自動で検出
 - ii. 工場での製造スケジューリング (**x1000** 高速化, 探索空間 $10^6 \rightarrow 10^{274}$)
- (b) 査読付きワークショップ Asai, M. and Fukunaga, A. Applying Problem Decomposition to Extremely Large Planning Domains. In *Proceedings of the 5th Workshop on Knowledge Engineering for Planning and Scheduling(KEPS)*, June 2014.
 - i. 複数種類の繰り返し構造の検出
- (c) 難関国際会議 Asai, M. and Fukunaga, A. Solving Large-Scale Planning Problems by Decomposition and Macro Generation. In *Proceedings of the 25th International Conference of Automated Planning and Scheduling(ICAPS)*, June 2015. 採択率 **33%**

- i. 複数の繰り返し構造をより柔軟・汎用に組み合わせる手法
 - ii. ベンチマークセット全体で高速化 (**x3-4** 高速化, 探索空間 $10^7 \rightarrow 10^{28}$)
- (d) 難関国際会議 Asai, M. and Fukunaga, A. Tiebreaking Strategies for Classical Planning Using A* Search. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence(AAAI)*, February 2016. 採択率 **26%**
- i. コストゼロの辺がグラフ探索に引き起こす問題を解決
 - ii. コストゼロの辺を含むグラフ探索全てに影響 (探索空間 $10^6 \rightarrow 10^{88}$)

2. 今後の研究計画

- (a) 探索グラフの対称性の検知技術と、小問題分割技術の融合

8 付録 古典プランニング問題とは (定義)

アクション集合 A , オブジェクト集合 O , 初期状態 I , ゴール G

状態 $:=$ 真である命題の集合

アクション $a \in A : \langle \text{pre}(a), \text{add}(a), \text{del}(a), \text{cost}(a) \rangle$

ただし、 $\text{pre}(a)$: 前提条件, $\text{add}(a)$: 追加効果, $\text{del}(a)$: 削除効果, $\text{cost}(a)$: アクションの適用コスト

状態 s に対するアクション a の適用: $\text{pre}(a) \subseteq s$ の時に適用可能で、

$$a(s) = (s \cup \text{add}(a)) \setminus \text{del}(a)$$

終了判定: $s \supseteq G$ ならば ゴール達成

9 付録 古典プランニングを研究する意義は？

"古典"プランニングを研究する意義は？

古典プランニングは

様々な拡張のベースとなる **最小限** のフレームワーク

実験は 古典プランニングでやった というただそれだけ

← 古典プランニングでは評価がやりやすい

- ・ 単純なモデル
- ・ 詳細が調べつくされている
- ・ 何が性能向上の理由なのか切り分けやすい(差分が取りやすい)

直接の実用的な
意義もある



Deep Space 1
(Muscettola et al. '98)
宇宙船の自律航行



Mars Exploration Rover
(Bresina et al. '05)
火星探査ロボットの
自律行動システム

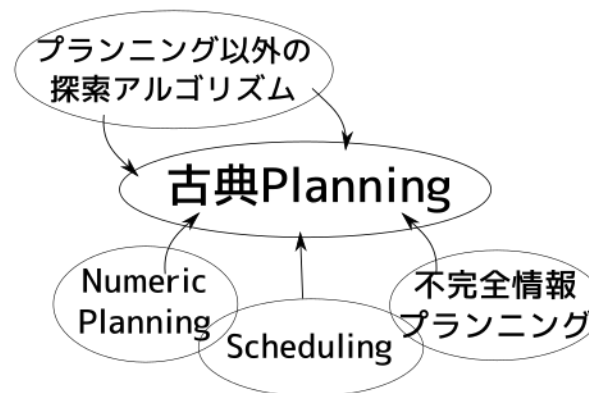
Xerox Parc Printer
大規模 商業印刷システムの
自律運用



Core Security (サービス企業)
企業ネットワーク脆弱性の自動診断
攻撃プランの自動生成

様々な拡張の「基礎」となる基盤技術

古典プランニングで成功した技術は
他の問題にも使える!



類似のシナリオ: SATソルバ

同じく基礎的なモデル



10 付録 AIの倫理について

- 研究内容は 漠然とした「AI」のうち グラフ探索 の研究
- 善悪の判断はそれ自体は行わない
- 価値判断は与えられる入力の中にある = 使用者の価値観を反映する
- 悪用の問題はある。しかし、自分としては、災害救助ロボットなど、人道的な応用を目指している

11 付録 ディープラーニングとどう違うのか

- 機械学習を埋め込むことは可能だ
- が、求められる推論の複雑さが根本的に違う → 独立した分野

ニューラルネット, **DL**, 強化学習

- 入力: 現在のデータ、過去の履歴、報酬 etc..
- 出力:
 - 次の**1ステップ**のアクション選択ポリシー (強化学習)
 - 固定長の分類結果 (画像認識)
 - ある意味 状況に応じて脊髄反射なエージェント

プランニング出力:

- **10ステップ, 100ステップ**
先の未来を先読みした
行動計画
- ICAPS14,15 の手法を使
えば 数千ステップ 先の
未来まで先読みすること
が出来る

ただしプランニングに学習機を埋め込むことは 可能 (実例複数あり)

DLから見て、プランニングは アプリケーション

プランニングから見て、DLは ツール

両者の組み合わせは機会があればやってみたい

11.1 付録 ディープラーニング関連

趣味の一環で、Common Lisp から直接使える GPGPU のライブラリを作成中 (DL を作ってみるため)

- OpenCL ベース
- Lisp の文法を直接 OpenCL C に変換し実行するトランスレータ
- OpenCL のメモリ管理を Lisp GC に埋め込み

12 付録 第五世代コンピュータとの違いは？

第五世代コンピュータ：並列推論機械 (Prolog ベース, ハードウェア, OS)

根本的なソフトウェア技術、探索技術 が未発達だった

第五世代

現在

後方全探索+バックトラック	前方ヒューリスティック探索
---------------	---------------

Prolog ベース	C/C++で高度に最適化されたプログラム
------------	----------------------

State packing, 決定木, mutex...

今はベンチマーク問題 **1104** 問 のうち 5 分で **800** 問 前後解ける

仮に 当時のソフトウェア を 現在のハードウェア で動かしたとしても、 **100** 問
も解けないだろう

13 付録 Explicit Graph と Implicit Graph との違い

カーナビ、ソーシャル
グラフなど : Explicit
Graph Search

グラフ全体がメモリ
(~数ペタバイト) また
は二次記憶 (~数ゼタ
バイト) に収まる

参考: 2012 年の全世
界のデジタルデータ:
数ゼタバイト (1ZB =
 10^{21} バイト)

AI and Web の分野な
ど

プランニングにおける探索グラフ : Implicit
Graph Search

地球に存在する全計算資源を集めても二次記憶
に入らない

グラフのノード数は状態変数に対して 指数的に
増加

動的に必要な量のみメモリ確保をしないと問題
が解けない

探索空間サイズの例:

3x3x3 のルービックキューブ: $4.32 \times 10^{19} = 4$ エ
クサバイト

4x4x4 のルービックキューブ: $7.40 \times 10^{45} \sim 10^{24}$
ゼタバイト

5x5x5 のルービックキューブ: 2.83×10^{74}

14 付録 似たような研究は誰がやっていますか どこでやられていますか

国内ですか、国外ですか？

YES NO

15 付録 プランニングはマイナーで大したことのない分野?

日本にプランニングの研究室がない \neq 世界で研究室がない

ICAPS, SoCS : 例年 150 人-200 人の参加者を集めており大変盛況,

AAAI, IJCAI : プランニングに関する論文は例年数十本採択 Proceedings の一つの章をなす

JAIR, AIJ : 論文誌でもプランニングの論文は多い (JAIR Volume 54: 12 本中 2 本 がプランニング論文)

主な研究室:

MIT CSAIL (Brian Williams), Carnegie-Mellon,

NASA (NASA Ames および NASA/Caltech JPL のそれぞれに 20 名以上の研究者), 欧州宇宙機関 (ESA)

指導教員は NASA JPL AI Lab の元メンバー

16 付録 汎用性を失わずに解く？

No Free Lunch 定理: 最適化アルゴリズムの性能は 全問題の平均を取れば 全て同じ

Q. NFL定理のもとで「汎用性を失わずに高速に解く」というのは不可能？

A. NFL定理は確かにそのように主張するが、プランニング分野の意味する「汎用性」は

人間にとって有意義な問題の集合 における汎用性である。

全プランニング問題の集合 \supseteq 人間にとって有意義な問題の集合

従って、 全問題の平均を取れば という前提が成り立たない。

17 付録 その研究は...

	重要度	評価	オリジナリティ	過去のインパクト
ACP		発表した 難関学会	ループの概念を検出	はるかに巨大な問題
CAP		難関学会 発表した いくつか質問された メールやり取り	問題分割手法 柔軟な統合手法	大規模な問題 それまでの手法では 広範囲に分類
AAAI16		三人中 二人の査読者に絶賛された	同コストのノードの分類	70年代から 通常と異なり コスト0は算

過去全体

17.1 国内で誰が似たようなことをやっているか

この専門分野をやっている人は少ないんですが、少し離れているが最も似ている研究というと

SAT ソルバの研究の人はいる – 田中先生

推論系 – logic and reasoning

Lemma Reusing for SAT based Planning and Scheduling ICAPS 10年ぐらい前
NIIの井上勝海 CSP でプランニング

神戸大学田沼先生 SAT

九大 横尾誠先生 AAMAS マルチエージェント CSP

自分の指導教官がアメリカから飛んできた研究者なので、もともと日本にいなかった研究者だった

スライド2を見せる

ERATOの人は同じくグラフ探索をやっていますが、explicit/implicit の違いがあります。瓦林先生秋葉さん

Katsutoshi Hirayama, Kobe university (constraint optimization, CSP, distributed CSP)

Toru Ishida (Kyoto U. stopped working in search a long time ago, but for a while, he was the leader in A*-related search methods in Japan; coauthored some papers with Korf).

18 付録 学会論文の位置づけ

19 HTN と ICAPS-15 の違い

HTNは人間が問題分割を行う人件費を考えると非常にコストパフォーマンスが悪い→ 自動で問題分割

20 付録 Q&A

- Q. 探索空間の比較について、なぜ「以前」の数字がスライドによって変わるの？
- A. 論文の中で使った実験設定が異なるからです。

21 付録 今後のキャリアパスは？

研究者押しで押しまくる大学あるいは企業の研究者迷うことなくパツと答える

22 付録 今後の研究計画が、インパクトの少ない今までの延長のように見えるんだけど …

23 付録 アドホックな解法に見えるんだけど …

24 付録 失敗しそうなんだけど …

25 付録?? とにかくまとはずれな質問

引き出しを探るタイプの質問

上席研究員主任研究員

senior engineer

4段階

下 associate XX senior principal 上

PI principal investigator

time sink