

自律ブロックエージェントによる ブロック取り出し問題の基礎研究

木下正博 嘉数侑昇
北海道大学工学部

1. はじめに

すでにコンテナ内部に存在しているブロック群の移動プランを獲得する問題は空間的制約を伴う。いくつかの工業分野ではこの種の空間的制約を伴う問題に対し、様々なアプローチがなされてきた。特に「カッティングとパッキング」の分野では、移動プランを伴う様々な問題が存在しており、衣料、靴業界あるいは自動車産業等では多くの応用的な問題が発生している。干渉回避という観点から2種類の分析が存在する。1つは「カッティングとパッキング」におけるレイアウト問題で、この場合、移動プランが実行されている間形状の干渉を考慮する必要がない。なぜなら、最終的な配置において重なりがなければ良く、移動処理中は形状同士すり抜けることが許されるからである。もう1つは「コンパクションとセパレーション」[1]に見られるような移動プランに干渉を考慮しなければならない問題である。一般にこのような移動プランを構築することは容易ではないと言われており、2次元の干渉を考慮した移動プランは長方形状であってもNP-hardである[1]。また、形状を移動の初期状態からゴール状態まで扱う倉庫人問題ではPSPACE-hardであると言われている[2]。本研究のブロック取り出し問題は同様の構造を持っている。本報告では大域的最適化により、各ブロックが自律エージェントとして振る舞うような移動プランを獲得するための方法論を提案し、その有用性を示す。

2. ブロック取り出し問題

ブロック取り出し問題とは出口をもつコンテナからいくつかのブロックを取り出す問題である。出口形状は任意とし、すべてのブロックを出すことができなくても良いとする。また、出口の個数も1つに限定しない。たとえば、図1に示すようなコンテナとブロックとそれを運搬するフォークリフトを考える。フォークリフトにはコンテナの壁、他のブロックを持ったフォークリフトとの距離を判定できるセンサー及び接触センサー、また前進、後退、回転が可能なアクチュエータ、センサーからの情報を処理する情報処理系を装備していると仮定する(図2)。

各フォークリフトを自律的なエージェントとした場合、経路計画において以下のような困難な点が存在する。

- (1) フォークリフト間のインタラクションの発生。
- (2) 自律的振る舞いによるデッドロックの発生。
- (3) 優先順序のない同時移動。

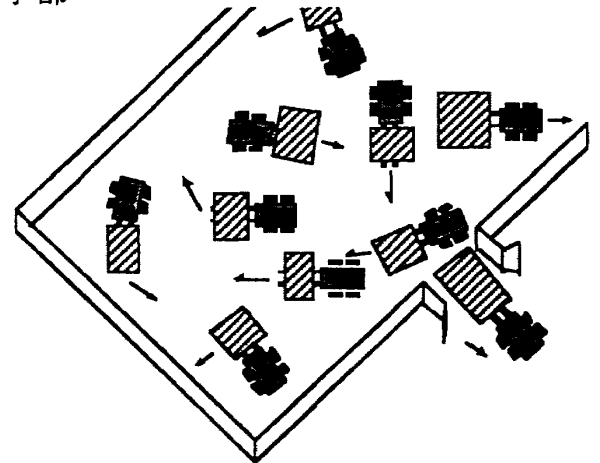


図1 ブロック取り出し問題の具体例

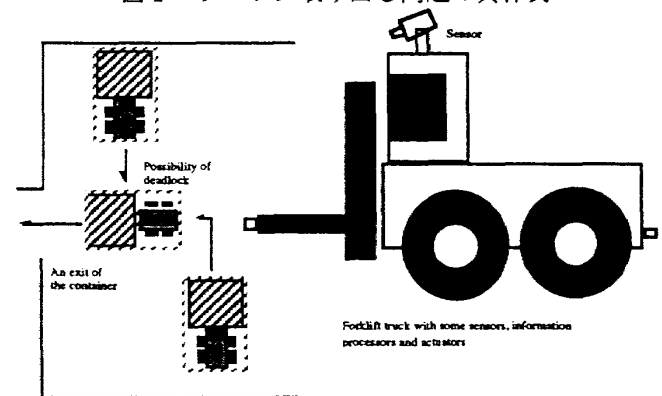


図2 フォークリフトとその動き

ここで、以下の目的を設定する。

- (1) 取り出すブロックの個数を最大化する。
- (2) ブロックを取り出す時間を最小化する。
- (3) 干渉を回避する移動経路の決定。

ブロック取り出し問題を解くために、移動プランは上記目的を満足していなければならない。移動ロボットの分野におけるピアノ問題のようにブロックは他のブロックやコンテナの壁によってできる障害物の間の干渉のない経路を決定しなければならない。積み木問題における「どのブロックを動かすか」、「どの位置へもっていくか」のような動かすブロックの順序の決定は重要である。カッティングとパッキングの分野では多くの最適化の問題がみられ、箱詰め問題、パズル問題等がもつ複雑性をブロック取り出し問題も持つ。

3. ブロック取り出しへのアプローチ

人間によってブロックを取り出す場合、出口に近いブロックの優先順位が高い。このヒューリスティックは取り出し順序を決定する際に有効である。すなわち、このアプローチのための戦略はブロックと出口の距離の関数として表現することができる。

$$b^* = \underset{j \in IB}{\operatorname{argmin}} [BD_j], IB = \{1, 2, \dots, NB\} \quad (1)$$

ここで、 b^* は $\min_{j \in IB} [BD_j]$ を満足する出口に最も近いブロックである。 BD_j はj番目のブロックと出口間の距離である。ここで、距離関数 $d(E, G_j)$ を次のように設定する。

$$d(E, G_j) = (E_x - G_{xj})^2 + (E_y - G_{yj})^2 \quad (2)$$

取り出すべきブロックが決定され、そのブロックが取り出し不可能である場合、再パッキングが必要となる。

3.1 再パッキングのためのGA的アプローチ

再パッキングのためのそれぞれのブロックの移動プランをGAにより発生させる。これは複雑な最適化問題へのヒューリスティックなアプローチへ遺伝的アルゴリズムが適用可能であるからである。この再パッキング処理もまた最適化問題の一つである。また、最適化に影響を与えるブロックの移動軌跡も重要な要素である。再パッキングにおける移動と回転の情報はすべてビットストリングに構築され、すべてのブロックができるだけ出口から遠ざかるように振る舞うような移動プランを作り出す。再パッキングの移動プランを次式により表わす。

$$\operatorname{maximize} \sum_j d(E, G'_j) \quad (3)$$

ここで、 $G'_j = P(G_j)$ であり、 P はブロック移動のプランを示す。 G'_j はプラン適用後のj番目のブロックの重心である。このプランは再パッキング後も取り出すブロックがない状態まで繰り返される。

3.2 GAのインプリメント

移動プランを獲得するためのGAストリング S_i を次のように与える。

$$P = S_i = \{bs_k; k=1, \dots, NB\} \quad (4)$$

$$bs_k = (BlkNo_k, Rot_k, Xmov_k, Ymov_k) \quad (5)$$

ここで、 S_i はi番目のストリングであると同時にプランそのものである。そして、 S_i はブロック番号、回転フラグ、x,y方向の移動を表わすサブストリング bs_k の集合で表現され、一つのストリングはNB個の bs_k によって構成される。

4. 計算機実験

上述方法論に従ってシミュレーションシステムを作成し、再パッキング及び取り出し処理の計算機実験を行った。図3、図4はパッキング結果であるが、最適に近い移動プランが得られている。しかし、表現系で扱う動作種類が少ないためややむだな領域が残っている。グラフ中の上部は最大フィットネス値で下は平均フィットネス値を表わ

している。また、図5は出口付近に取り出しのための十分な領域が存在する場合のブロック群の移動軌跡を示している。ここでは、フィットネス関数を「出口から出たブロックの個数×重み」として与え、パッキングと同様GAを用いて移動プランを得ている。

5. おわりに

本報では、ブロック取り出し問題を空間的制約を伴う大域的最適化問題としてとらえ、パッキング処理、取り出し処理にGA的アプローチを試み、ブロック群の初期配置に影響されないより適した移動プランを獲得できる方法論を提案した。ブロック群の自律性に関しては、さらに考察が必要と思われる。

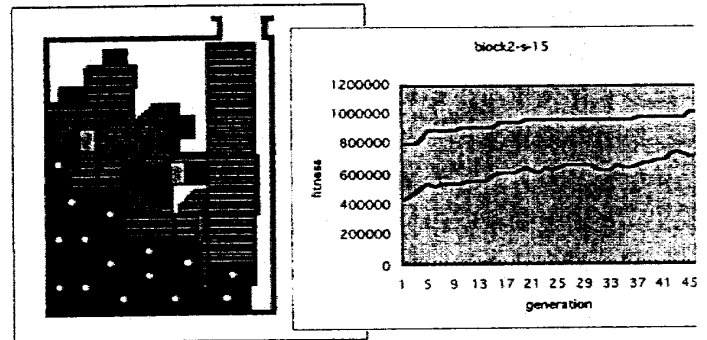


図3 再パッキング結果（同形状）

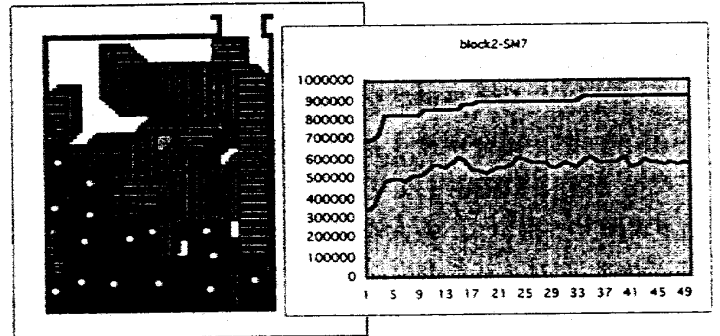


図4 再パッキング結果（異形状）

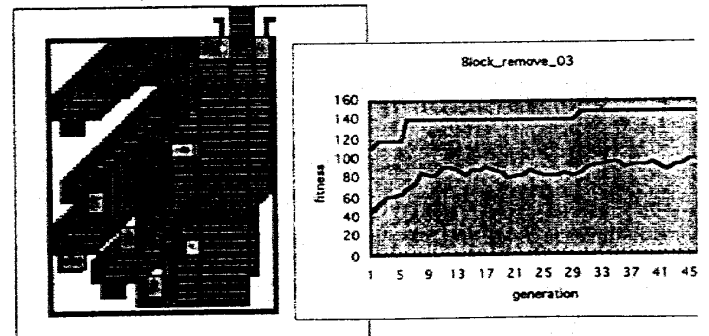


図5 取り出し処理の移動軌跡

参考文献

- 1) Li, Z., and Milenkovic.(1993), On the complexity of the compaction problem, Proceedings of the fifth Canadian Conference on Computational Geometry, Waterloo, Ont., pp.153-162
- 2) Hopcroft, J.E., et al.(1984), On the complexity of motion planning for multiple independent objects;PSPACE-hardness of the Warehouseman's problem, The International Journal of Robotics Research pp.76-88.