

多関節形マニピュレータの障害物回避の分散制御アルゴリズム

3P-5

森本 篤 桂 昌豪

Julio TANOMARU

徳島大学工学部 知能情報工学科

1. はじめに

一般にマニピュレータ[1]の障害物回避問題は多次元空間内の探索問題である。しかし全探索によって解空間を探索しようとすると、複雑な形状をしているロボットと周囲の物体(障害物等)との干渉計算が必要であり、この計算量は膨大なものとなるので実時間で最適解を求めるることは事実上できない[2]。この様な問題を解く方法として、仮説検定法、ペナルティ関数法、Free-Space計算法[3][4]などがあるが環境が動的に変化するような場合の解法は示されていない。

そこで本論ではこの問題を短時間で解くため、またこれにより動的な環境の変化に対応した障害物回避の分散的なアルゴリズムを提案する。

2. マニピュレータの障害物回避問題

一般的な多関節型マニピュレータの障害物回避問題を定義する。 n 個のジョイントからなるアームを障害物と接触しないように動かし、先端の効果器(Effector)を移動し目標点に導く問題を多関節形マニピュレータの障害物回避問題とする。

各リンクは同一平面上を移動するものとする。このとき、効果器のあるリンクから最も離れたジョイントの位置は固定されている。

各リンクの長さを R_k ($1 \leq k \leq n$) とするとアームの移動領域は最大で半径 $\sum R_k$ の円となり、目標点はこの円内に存在するものとする。

また、アームは障害物の領域を通過できないのでアームの形状、各ジョイントの移動方向は制限をうける。ここで障害物は環境中に m 個で位置は時間によってかわるものとし、また障害物は円で近似されているものとする。(Fig. 2-1)

- ・Joint(Link) n 個で各リンクの長さは不变
- ・Obstacle m 個で時間とともに移動する。
- ・Target 1 個

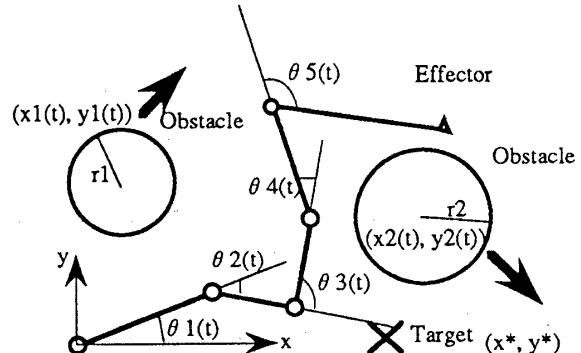


Fig. 2-1 一般的な障害物回避問題

3. 分散的な障害物回避アルゴリズム

3.1 分散制御マニピュレータの構成

ここでは分散的な制御のできるマニピュレータの構成を述べる。マニピュレータの形状は二次元平面上を可動範囲とする多関節形マニピュレータで、一つのリンクに対し 1 つの自由度を持つものである。

各リンクにはセンサーが装備されており、現在の状況を把握することができる。

マニピュレータの各リンクを制御するために各リンクに対応するプロセッサが存在し、隣合うリンクとの通信が可能となっている。効果器から一番遠いジョイントから順に PE0, PE1, ..., PEn が接続されているものとする。(Fig. 3-1)

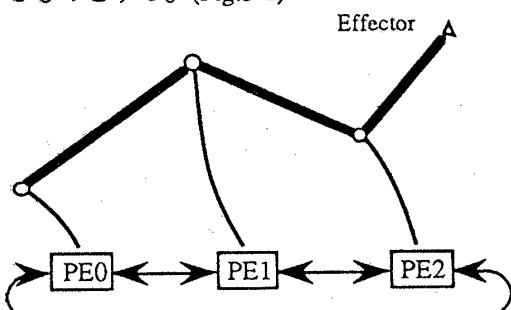


Fig. 3-1 プロセッサとリンクとの関係 (3-Link)

システムは、マルチコンピュータの構成をとっており、各プロセッサは障害物と自分の位置を知ることができ、それにより各々のリンクの角度をコントロールする事ができる。また、プロセッサ間の通信により隣のプロセッサとデータの受け渡しを行ない、他のプロセッサの協力を仰ぐことができる。

3.2 障害物回避アルゴリズム

3.1 で述べたような構成のマニピュレータのコンフィギュレーション中に障害物があるとき、各々のリンクが独立、または協力して障害物回避を行うアルゴリズムの一つを提案する。このようなアルゴリズムは多数考えられるが、ここでは3-Link マニピュレータを用いた、障害物が一つの簡単なアルゴリズムを示す。

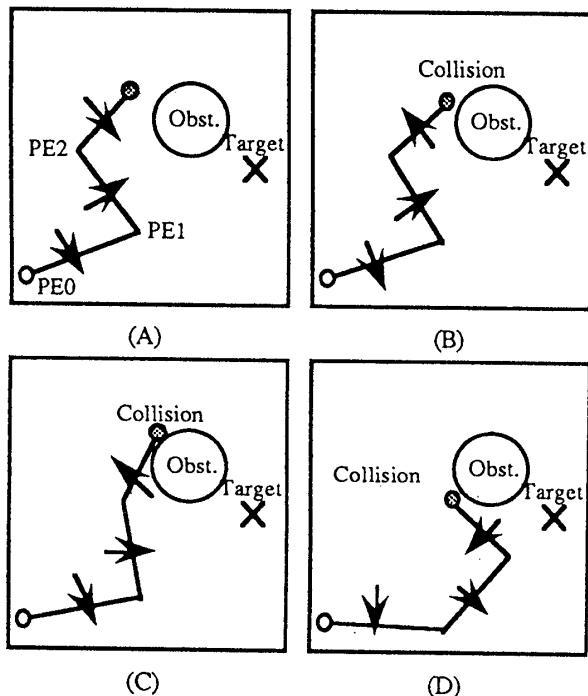


Fig. 3-2 3-Link マニピュレータの障害物回避の例

各プロセッサは担当するリンクの角度、目標点の座標、障害物の位置を得ることができるとする。

ALGORITHM (Fig. 3-2)

Processor 0, 1: 効果器のないリンク

- 1) 現在の状態をセンサーから、全リンクの角度を他のプロセッサから取得し、現在の状況を計算
- 2) 効果器の位置が目標点になつていれば処理終了
- 3) 他のプロセッサからの要請があれば、他のプロセッサの要請する方向にリンクの移動量を計算
- 4) 他のプロセッサからの要請がなければ、現在の効果器の位置が目標点に接近するようにリンクの移動量を計算
- 5) 障害物との接触があるなら、接触を回避する方向へリンクを移動するように計算
- 6) 1) に戻る

Processor 2: 効果器のあるリンク

- 1) 現在の状態をセンサーから、全リンクの角度を他のプロセッサから取得し、現在の状況を計算

- 2) 効果器の位置が目標点になつていれば処理終了
- 3) 現在の効果器の位置が目標点に接近するようにリンクの移動量を計算
- 4) 障害物との接触があるなら、接触を回避する方向へリンクを移動するように計算
- 5) 1) に戻る

目標点に効果器の位置を移動する分散的なアルゴリズムには角度制御モード、半径制御モード、及びこれらの混合したモードがあり[5]、これらを使用して目標点に効果器を移動させる。

4. 考察

今回提案したアルゴリズムの利点は、空間の探索よりも計算量が少ないので、障害物の移動等のコンフィギュレーションの変化があるときでも実時間で解を求めていくことができる点と、アルゴリズムが単純なのでマニピュレータのリンクが増加したときでもシステムの変更が少なくて済む点であると考える。また、分散処理を行っていることからリンクが損傷をうけてもある程度の処理の続行は可能である、すなわちフォールトトレラントの点で優れていると考えられる。

5. おわりに

本稿では分散制御による多関節形マニピュレータの障害物回避アルゴリズムを提案した。これらの方法の有効性・欠点はアルゴリズムをワークステーション上に実装し、シミュレーションにより検証する。また、分散制御実験装置上に実装し、マルチコンピュータ上で実験を行い妥当な解を得ることができるかを検証する。

さらに、障害物が移動するような環境の変化する場合について、また障害物が複数ある場合についてのアルゴリズムを考案し、検証している。

参考文献

- [1] 中野栄二, 小森谷清, 新井健生: ロボット工学応用技術, 養賢堂, 1985
- [2] Yong K. Hwang, Narendra Ahuja: Gross-Motion Planning, ACM Computing Surveys, Vol.24, No.3, Sep. 1992
- [3] 長谷川勉: 障害物回避, 人工知能学会誌, Vol.5 No.6(pp731-736), Nov. 1990
- [4] 花房秀郎, 長田正: ロボット・モーションIII, HBJ 出版局, Nov. 1985
- [5] 桂昌豪: マルチコンピュータによる多軸サーボ系の分散制御, 徳島大学大学院知能情報工学修士論文, Mar. 1993