

自立学習歩行ロボットのシミュレーション

1N-6

北添 徹郎

宮崎大学工学部情報工学科

1 はじめに

教師がいなくても自立的に学習する行為は動物界ではよく見られる。人間は生まれた時は歩けないが、筋肉が発達し神経組織が整備されてくるにつれて自立的に学習しながら、おそらく誰からも教えられなくても歩けるようになる。雀の雛は屋根裏の巣から地面に落ちる間に、飛び方を学習してしまう。(多分すでに予備的な訓練をしているのである)。また、興味深いことは、一つの動物の歩き方にもいくつかのモードがあることである。馬など四本足の動物では、並足、速走、駆足等、それぞれ足の運び方が違うモードがある。それぞれの動物には、ハード的な制約のもとで、運動を制御するソフトな機構が備わっており、一つ一つの運動を評価しながら最適な効率を得られる運動モードを選んでいるのであろう。これまで自立学習については、中野馨氏による先駆的研究 [1,2] があるが、ここでは二足歩行を例にあげながら次の点に主眼をおいて研究を行なう。

- (1) 安定性の問題。歩行は一見簡単な行為の繰り返しのように見えるが、実は自立学習の観点から見ればかなり難しい問題を含む。自立学習の場合、スタートの歩幅、各足のスピードについて自由に選びながら学習するわけであるが、そのためには一つの歩幅や足のスピードの組について周期的に安定した歩行が保証されている必要がある。
- (2) (1) が保証された場合において、歩幅や歩行スピードを変えながら、最も効率のよい歩き方を自立学習する方法。
- (3) 一つの歩行モードにおいて最適解が見つかった場合、他のモードにいかに移ることができるか。一つの最適運動モードは評価関数のローカルミニマムになっているであろう。したがって、一つのローカルミニマムから、他のローカルミニマムにいかにして自立的に移行するかという問題である。

以下においては最も単純な二足からなるロボットをコンピュータ上でシミュレーションしながら以上の諸点について論ずる。

2 二足ロボットの仕組み

図1, 2のように長さ a の足 AC, BC を持つ二足のロボットが直線上を動いていく問題を考える。一つの軸足 (BC) は B が直線 l を捕まえており、角度 α を変化させる。また、足 AC と BC のなす股開き角 2β も変化させる。 A が再び l に接する時、一つの歩行が終了し、 CA を軸足とした次の歩行が開始されるものとする。問題を簡単にするため、 C が l に接しても何ら変化は起こらないものとする。代表的な動きとして

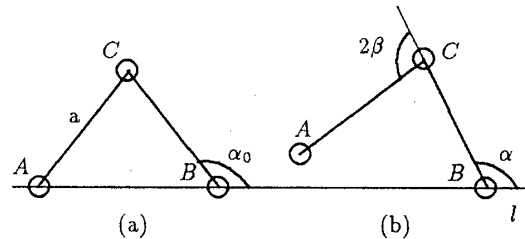


図1: 二足ロボット

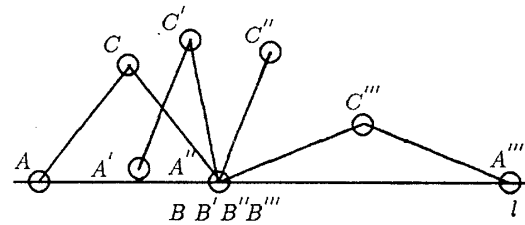


図2: 歩行モード

- (1) 回転。時間的に α と β を共に減少させていく場合。図1 (b) において、足 A が C のまわりを回転し B の右側に着地する。
- (2) 歩行。 α を減少、 β を増加させた場合。 A は一担 A'' で重なり、さらに運動して A''' で着地する場合。

歩行関数は回転と歩行で異なるが統一して書けば

$$walk(\alpha_0, e_0, e_1, e'_1; s_0, s_1, direction)$$

となる。ここに、 α_0 は軸足の初期角度を表す。 e_0 は軸足角 α の、 e_1, e'_1 は駆動足角 β の図2で A と B が接する前後の角速度 (一定) を表す。 e_0 および e_1, e'_1 は正の値をとり、 s_0, s_1 ($= \pm 1$) は、それらの符号因子を表すことにする。 $direction$ は右と左足のどちらを軸足とするかを決定する。このようにパラメータを二つの部分に分離したのは、後半部が運動の大局的モードを決定し、前半部が局所的な運動を決定するように役割を分担させるためである。例えば右方向に回転しながら動く場合は、 $s_0 = -1, s_1 = -1, direction = +1$ として、 $\alpha_0, e_0, e_1 (= e'_1)$ を調節しながら、最も効率の良い運動を探していくことになる。次に s_1 や $direction$ を変えると全く違った運動モードに移行することになる。

3 システムの安定性

以上のパラメータを変化させながら、ロボットは以下で述べる評価関数を検知しながら、それを最小にするような運動をいかに実現していくかということが我々の問題である。しかし、その前に重要なことは、システムが安定である

ことを保証しなければならない。動物や人間は、速度の速い遅いは別にして、何故、安定して歩くのかという問題は意外に簡単ではない。ここで安定という意味は、同じ周期的運動を定常的に存続することである。問題は walk で一歩あるいた後の形状は前のそれと変わってしまうことである。

この問題に対しては一つの良い解決法が見つかった。即ち walk によって $\alpha_0 \rightarrow \alpha'_0$ になったとしても、もしも次の一歩において、同じ $e_0, e_1(e_1')$ を用いるならば、再び同じ初期値にもどることが証明される。従って、二歩を単位として運動をさせるならば、周期的運動が保証されるのである。従ってもし最適値が見つかった場合、この条件のもとでその運動は安定したものとして実現される。

4 評価関数

上記のことから、評価関数は二歩について測ればよいことになるが、実は最初の一步と後の一步について評価関数が同じになることが示される。従って評価関数は一步について行なえば良い。

$$eval = -speed + c_1 loss + c_2 torque$$

として、speed は単位時間あたりの水平方向の重心移動とする。

$$speed = (x'_G - x_G)/T$$

$$T = (\alpha_0 - \alpha'_0)/e_0$$

と定義する。ここに α'_0 は一歩あるいた後の α の値を表す。loss は

$$loss = \sum_{i=1}^N |y_G(i+1) - y_G(i)|/N$$

と定義され一歩進む間の垂直方向の重心の変動率を表す。torque は歩き始める時または静止する時の軸足にかかる平均モーメントとして、

$$torque = \pi / (\alpha_0 - \alpha'_0)$$

とする。重み c_1, c_2 は正常な運動が実現されるように選ばれるべき係数である。

5 ネットワークモデル

我々のシステムは eval をもちいて、図3に示すような自立学習ネットワークによって決定される。まず、 $s_i, direction$ を決めると運動モードが決定され、つづいて α_0, e_i を変化させることによって、eval を検知しながら、最適値をもってゆく。その方法はいわゆる山登り法を用いて、eval を小さくする方向へ試行錯誤的にパラメータの組 x を変動させてゆく。即ち、ランダムに微小変動 Δx を選び

$$if \text{eval}(x + \Delta x) < \text{eval}(x) \text{ then } x = x + \Delta x$$

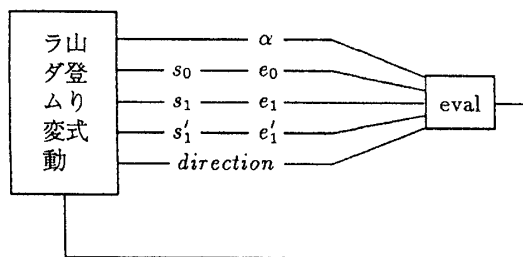


図3: ネットワークモデル

のようにする。こうしてローカルな極小値に到達すると、次に違った運動モードを起動するために、 $s_i, direction$ を変え、運動に巨大な変化を与える。その後、 α_0, e_i を変化させて、別の極小点に持って行く。例えば、最も良い回転モードに到達した後、次に歩行モードに移行し、最も効率の良い歩行が実現されるようにする。また、目標物が移動した場合には、direction を変えることによって、例えば右向きの運動を左向きの運動に変えられる。

6 シミュレーションの結果

例として $s_0 = +1, s_1 = -1, direction = +1$ として回転モードからスタートした場合のシミュレーション結果を示す。eval($\alpha_0, e_1/e_0$) は $+0.411(21/40 \pi, 1.0) \rightarrow +0.324(21/40 \pi, 0.9) \rightarrow +0.251(21/40 \pi, 0.8) \rightarrow \dots \rightarrow +0.041(26/40 \pi, 0.4) \rightarrow \dots \rightarrow +0.003(31/40 \pi, 0.3) \rightarrow \dots$ 等を経由しながら最終的には回転の極小値 $-0.007(35/40 \pi, 0.3)$ に到達する。次に、 $s_1 = -1 \rightarrow +1$ に変化させて、歩行モードに移った場合、eval($\alpha_0, e_1/e_0, e_1'/e_0$) を3自由度の関数とすると、回転と同様に eval の小さいの値方向に次第に移行する。この場合 eval の極小値は $-1.634(30/40 \pi, 1.05, 0.95)$ or $-1.634(29/40 \pi, 0.95, 1.05)$ 附近にある。但し、注目すべきことは、この付近では近傍に eval の鋭い山があり、歩行の安定性を確保するための特別の注意が必要となる事である。歩行の機構等については面白い問題があり、発表時に述べることにする。

結論として自立的な学習機構によって最大効率を得るために、第一に運動が安定(周期的)であること、第二に運動モードをグローバルに変える変数とロカルに変化させる変数に分離する必要があること、などが結論される。これに基づいて各変数の入力の評価関数によって検知しながら、新しい変数を生み出すネットワークを構成し、それらがシミュレーションによって最適な運動に向かって自立学習していく過程が実現された。

参考文献

[1] 中野 肇. ロボットの歩行. 日本臨床 45 巻 2 号 (s62) 109-112.
 [2] 中野 肇. 行動する機械. 生体の科学 37 巻 1 号 (1968) 41-48