

高速打撃動作におけるマニピュレータのオンライン軌道生成

Online Trajectory Generation of a Manipulator for High-speed Batting

学 妹尾 拓 (東大) 正 並木 明夫 (東大) 正 石川 正俊 (東大)

Taku SENOO, Akio NAMIKI and Masatoshi ISHIKAWA

University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

Key Words: *batting, visual feedback, online trajectory generation*

In this paper high-speed robotic batting is achieved using a high-speed arm and high-speed stereo vision. In this algorithm, in order to achieve both rapidity of swing and accuracy of hitting, a hybrid trajectory generator of both visual information and the time variable function is proposed. In addition, the generated trajectory is modified by visual feedback in realtime. As a result a manipulator can follow the ball while swinging a bat at high speed even if it is difficult to predict the trajectory of a ball. Experimental results are shown in which a high-speed manipulator hits a ball thrown by a human.

1. はじめに

未知な環境や動的に変化する環境でロボットを制御するには、外界の状況を把握する手段としてビジョンシステムが有効である。最近では、視覚情報をフィードバックループに組み込んでロボットの制御を行うビジュアルサーボの研究も盛んである。しかしロボット制御に必要な処理レートは1kHzと考えられており¹⁾、従来のCCDなどの視覚センサでは実時間での制御は難しかった。そのため外界の認識の大部分を内界センサや触覚・力覚センサに依存していたために、確実な予測や計画が不可能な環境ではタスクの遂行が困難であった。

これに対して予測を用いて遅れを補償するアプローチ^{2,3)}も行われているが、対象の運動が不確実な場合には対応できない。人間も外界センサ処理が遅いために行動制御において遠心性コピー(フィードフォワード)で大半を補っているが、機械システムでこの問題を解決するには視覚処理を1kHz以上に高速化することがシンプルかつ効果的であると思われる。

我々の研究室では、サンプリングレートが1kHz以上のビジョンシステムを開発している⁴⁾。これをロボット制御のサーボループに組み込み、グラスピング⁵⁾・キャッチング⁶⁾などの動的な遷移を伴うタスクを実現してきた。このような高速視覚を用いることで高い応答性能の行動制御が可能であるだけでなく、高次の運動計画やタスク設定を柔軟かつ適応的に行うことも可能となる⁷⁾。

以上のような観点から、本稿ではダイナミックマニピュレーションの実現を目指すタスクの一環としてバッティング動作を取り上げる。これまでスポーツをロボットに適応した例はいくつもあるが、そのほとんどは前述のような予測で視覚処理の遅れを補償したり、学習アルゴリズムを用いて対象の動きに対するロボットの運動を徐々に獲得する^{8,9)}といった研究である。しかしこれらの方法はフィードバックよりフィードフォワードが支配的となるので、対象の予期しない動きには対応するのが難しい。またタスクに特化したロボットを用いているのがほとんどで、汎用のマニピュレータでタスクを遂行している例は少ない。

そこで本実験では、高速な視覚フィードバックによるリアルタイムの軌道生成を実現した。この軌道は高速かつ正確なスウィングを実現するために、対象の特徴量と時間変数を陽に組

み込んだハイブリッドな関数で構成されている。4軸マニピュレータを用いて実験を行い、その有効性を示す。

2. システム構成

本章では、ロボットシステムの構成・能力について述べる。図2.1(a)にシステムの構成を示す。

処理系は合計16GFLOPSの高速な浮動小数点演算能力をもつ16個のDSP(TMS320C6701:Texas Instrument Inc.)からなり、センサフィードバックを1msのサイクルタイムで実現するための主要部分となる階層並列情報処理システムである。また、多数のI/Oを複数のDSPにできる限り分散して配置することでI/Oのボトルネックを小さく抑えている。

視覚系は列並列ビジョンシステム(CPV)を搭載した2自由度アクティブビジョンである。CPVはPD(photo detector)とPE(processing element)で構成されている。PDの解像度は128×128画素・8bit諸調であり、この出力を128列並列にPEに転送する。並列に視覚処理ができるため、領域分割・モーメント検出などの視覚処理を1kHz以上で実現する能力を持つ。アクチュエータ部は、チルト・パンの2自由度を持つDDモータからなり、高速に移動する物体に対しても十分な追従能力を持つ。2台のアクティブビジョンを用いて立体視を行うことで、3次元視覚情報が計算される。

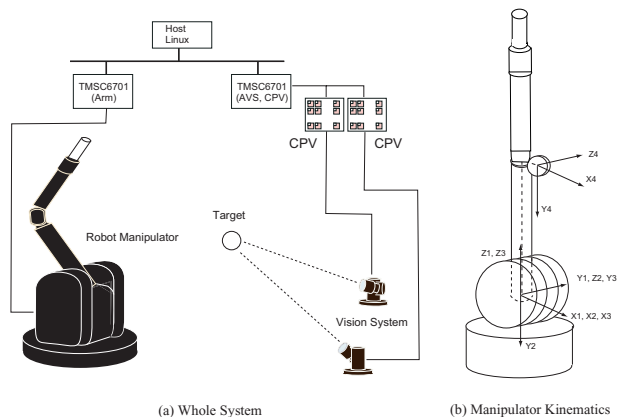


Fig. 2.1 System configuration

運動系はワイヤ駆動の4軸マニピュレータである (Barrett Technology Inc.)。図 2.1(b) に示すように、マニピュレータは肩・肘関節にそれぞれ旋回と屈曲の自由度を持っている。高出力・低減速なアクチュエータを用いるとともに、全アクチュエータを台座付近に配置して手先慣性を小さくすることで、最高速度 6 m/s・最高加速度 58 m/s² の高速な動作を実現している。人間の腕の伸展運動の速度は 6~10 m/s なので、人間と同様な高速で柔軟な動きができる。

3. 打撃動作

3-1 打撃戦略 バッティングではボールに速度を与える高速性とバットの芯で捉える正確性が重要である。トレードオフの関係にある両者を実現するために、マニピュレータの軌道生成に重点を置いて考える。

従来は視覚処理やアームの動きが遅かったため、対象の軌道を予測することでこれを補償する方法が用いられてきた³⁾。これはアームの目標軌道を

$$q = f(t) \quad (1)$$

のように時間 t の関数として与えることに相当する。しかしこの方法では対象が予測しない挙動をとったとき、タスクが実行されない可能性が大きい。

これに対して、最近では時間を陽に含まない形で視覚情報からダイレクトにアームの軌道を生成する方法も提案されている^{6,10)}。これはアームの目標軌道を

$$q = f(\xi) \quad (2)$$

のように対象の特徴量 ξ の関数として与えることに相当する。特徴量の例として対象の位置を考えると、アームの位置は対象の位置から一意に定まることになる。この方法では対象に合わせた動作が容易となるが、視覚処理やアームの動きに高速性が要求される。

バットの芯にボールを正確に当てただけならアームの軌道を式 (2) で与えても良いが、バッティングにおいてはボールの速度に関わらずバットを高速に振り切ったほうがボールに速度を与えられる。そこで、バットを高速にスウィングしながら正確に芯で捉えることを考慮して、アームの軌道を次の形で生成する。

$$q = f(\xi, t) \quad (3)$$

次節に、具体的なアルゴリズムを述べる。

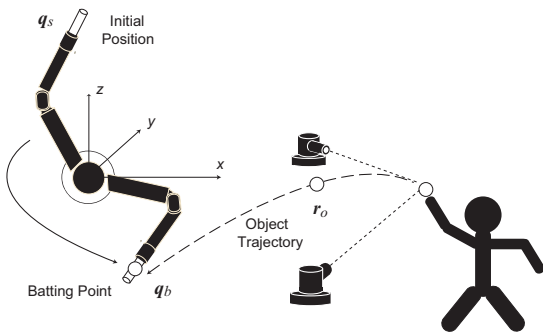


Fig. 3.1 Batting task

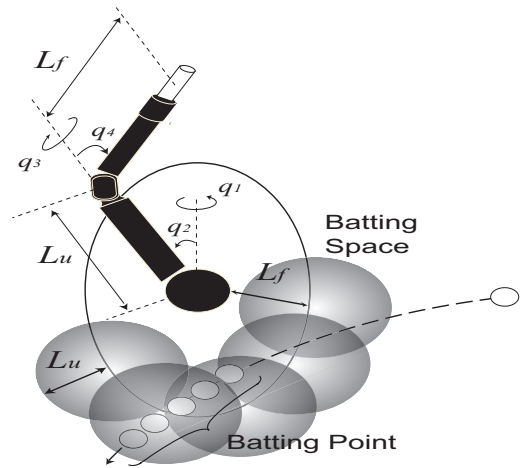


Fig. 3.2 Batting space and batting point

3-2 バッティングアルゴリズム 簡略化のために、アームにバットを固定して実験を行った。つまり、アームの先端でボールを打つことに相当する。

ビジョンによりボールの位置 $r_o \in \mathbb{R}^3$ が認識され、アームの目標関節角 $q_d \in \mathbb{R}^4$ が計算される。実際の関節角 $q \in \mathbb{R}^4$ は適切な制御器によって q_d に追従するように制御される。以後、 $q = q_d$ と仮定する。

上腕ベクトルと前腕ベクトルを $r_u \in \mathbb{R}^3, r_f \in \mathbb{R}^3$ で表す。ただし $L_u = \|r_u\|$ は上腕長、 $L_f = \|r_f\|$ は肘からバットの芯までの長さとする。バッティングとは、ある時刻 $t = t_b$ においてバットの位置とボールの位置がある打撃点 $r_b \in \mathbb{R}^3$ で一致することであり

$$r_u + r_f = r_b \quad (4)$$

と書ける。また打撃時の条件を、ボールが飛んでくる方向 (x 方向: 単位ベクトル $e_x \in \mathbb{R}^3$) と前腕ベクトルが直交することと設定して次式で表す。

$$r_f \cdot e_x = 0 \quad (5)$$

つまり、次式の平面上でバットとボールが衝突する。

$$x = r_b \cdot e_x \quad (6)$$

式 (5) より打撃時にバットの芯が取りうる空間は、平面 $x = 0$ 上において肩関節を中心とした半径 L_f の円 C を考え、この円上の各点を中心とした半径 L_u の球面が埋め尽くす空間 S となる。

$$C = \{r \mid \|r\| = L_f, x = 0\} \quad (7)$$

$$S = \{r \mid \|r - r'\| = L_u, r' \in C\} \quad (8)$$

以後、この空間 S を「打撃空間」と呼ぶことにする。ボールが打撃空間を横切るとき打撃空間内のどの位置でも打撃可能となるが、このうち打撃を行う1点「打撃点」 r_b を選ぶ必要がある。

ボールの速度はアームの速度に対して無視できない。そのためボールが打撃点に到達する前にアームを振り始めなくて

はならず、まずは仮の打撃点を単純な予測を用いて決定する。そして予測軌道と打撃空間が交わる曲線上のうち、

$$g(q, r_b) = 0 \quad (9)$$

を満たす点を打撃点 r_b として選ぶ。関数 g はボールの位置・アームの関節角などに関する何らかの条件を満たし、上記の曲線上のうち 1 点が定まる関数として定めればよい。

打撃点が決まれば、式 (4)(5) より打撃時の関節角 q_b が一意に求まる。またボールの予測軌道・打撃点から、ボールが打撃点に到達するまでの到達予測時間 $t = t_p$ が決定する。ここで $t_p \leq t_b$ ならバットを振り始め、 $t_p > t_b$ なら以後のサイクルは $t_p \leq t_b$ となるまで r_o, r_b, q_b の計算を繰り返す。つまり、ボールが現在位置から打撃点に到達するまでの予測時間はだんだん短くなっていくが、この予測時間が事前に設定した打撃時間と同じになった瞬間にスウィングを開始することに相当する。

バットの位置・速度・加速度を連続的に制御するため、打撃軌道を関節角において 5 次関数で与える。

$$q(t) = \sum_{i=0}^5 k_i t^i \quad (10)$$

スウィングを開始した時間を $t = 0$ とすると、境界条件は次式のようになる。

$$q(0) = q_s, \quad q(t_b) = q_b \quad (11)$$

ただし、 q_s は初期関節角である。

バットを振り始めてからボールに当たるまではボールの予測軌道・打撃点の補正を行い、予測誤差に対してもバッティングができるようにする。前腕は上腕に比べて慣性が小さく、高速にスウィングしている最中でも多少の補正に対して追従できるため、肘関節のみ関節角指令の補正を行う。補正した予測軌道と平面 (6) が交わる点を新たな打撃点 $r_{b'}$ とし、ここを通過するようにアームの軌道を修正する。ここで新たな打撃点 $r_{b'}$ に対応する関節角ベクトルを $q'_b \in \mathbb{R}^4$ とすると、 k_i は $q(t_b) = q'_b$ を満たすように補正される。

$$k_i = \begin{cases} k_i(r_{b'}) & \text{elbow-joint} \\ \text{constant} & \text{shoulder-joint} \end{cases} \quad (12)$$

つまり、関節角は

$$q = \begin{cases} f(r_{b'}, t) & \text{elbow-joint} \\ f(t) & \text{shoulder-joint} \end{cases} \quad (13)$$

と書ける。

式 (13) より、肩関節はボールの位置に依存せずバットを高速に振り切る動作、肘関節は視覚フィードバックによりボールをバットの芯付近に当てる動作に寄与することがわかる。

打撃後のスウィングも同様にして関節角において 5 次関数で与え、 $t = t_e$ において

$$q(t_e) = q_e \quad (14)$$

を満たすように滑らかな停止をした。ここで q_e はアームの終端姿勢である。図 3.3 に、バッティングアルゴリズムのフローチャートを示す。

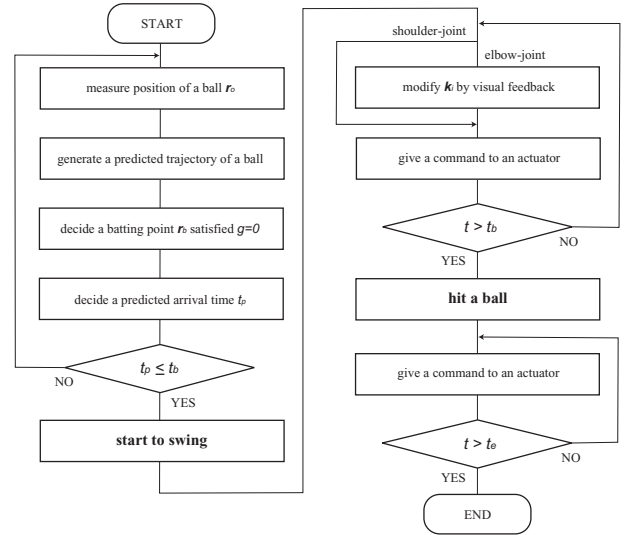


Fig. 3.3 Flow chart of a batting algorithm

4. 実験

4-1 実験設定 図 4.1 のように約 2.5 m 離れた位置から半径 4.5 cm の発泡スチロールのボールをアームに向かって人間が投げた。ボールを認識できる時間は 0.3 s 程度なので、あらかじめテイクバックした状態から振り切るという動作を行った。ボールの速度は 6~8 m/s であり、バットの速度は約 5 m/s である。

今回の実験では、逐次的にボールの位置をプロットし、最小自乗近似により関数のあてはめを行って予測軌道を求めた。 x, y 方向は 1 次関数、 z 方向は 2 次関数で近似した。また $g = q_2 - q_{s2}$ と設定した。これは 2 軸を固定してバッティングを行うことに相当する。センサが情報を取得してからアームのアクチュエータに指令を与えるまでのサイクルタイムは 1 ms とした。パラメータの設定として、初期姿勢を $q_s = (-\frac{\pi}{2}, \frac{\pi}{5}, -\frac{3\pi}{4}, \frac{\pi}{12})$ 、終端姿勢を $q_e = (\frac{\pi}{2}, \frac{\pi}{5}, \frac{\pi}{6}, \frac{\pi}{12})$ 、打撃時間を $t_b = 0.23$ s、 $t_e = 0.83$ s と設定した。

4-2 実験結果 図 4.2 に関節角の時間応答を示す。高速な視覚フィードバックにより、 r_b に基づく目標軌道が $r_{b'}$ に基づく目標軌道に補正され、かつ高速アームが補正された軌道に追

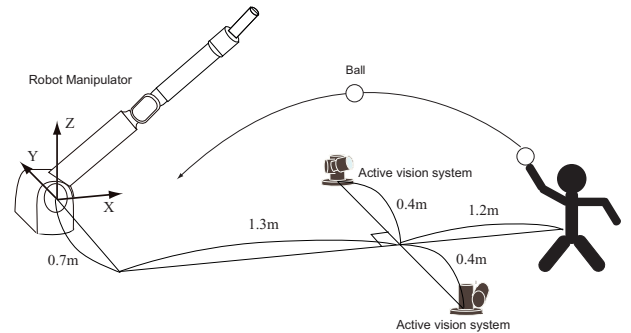


Fig. 4.1 Experimental configuration

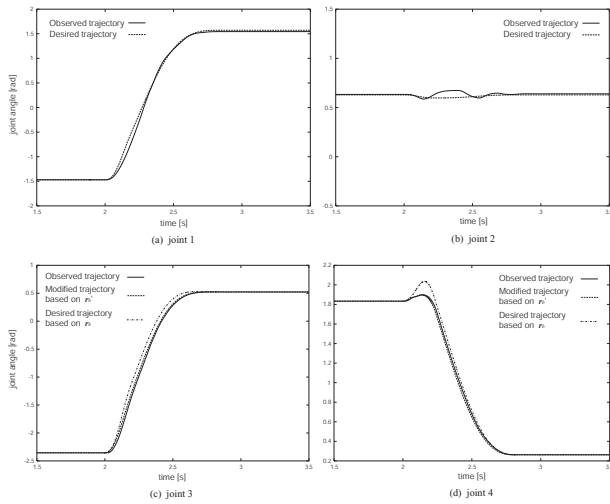


Fig. 4.2 Time response of joint angles

従していることがわかる。

図 4.3 にボールとアームの 3 次元的な動きを示す。視覚フィードバックによる打撃点の補正により、打撃点で正確にボールと衝突していることがわかる。また、図 4.4 にバッティングの様子を 30 ms 間隔の連続写真で示す。

アームの高速性によりスウィング開始後も軌道を補正でき、また打撃時間 t_b を短くできる。またビジョンの高速性により r_o が高速に計算できる。これはボール位置を多数計算できることに相当し、予測軌道の誤差を少なくすることができる。従って、ビジョンがボールを認識してから 0.1 s 以内にスウィングを開始するという条件のもと、高速な視覚フィードバックにより打撃点の補正がうまくいき、結果として高速にスウィングしながらバットの芯付近に当てることに成功している。

成功率は約 90 % であり、この結果の動画は本研究室のウェブサイト¹¹⁾ から見る事ができる。

5. まとめ

高速性を利用したバッティングアルゴリズムを提案し、実験によりその有効性を確認した。今後の課題としては、アームのダイナミクスを考慮した軌道計画・打撃直後のボール速度ベクトルの制御・ロボットハンドにバットを握らせての実験などを考えている。

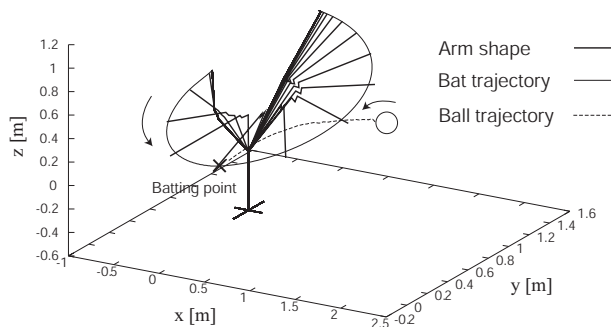


Fig. 4.3 Batting motion

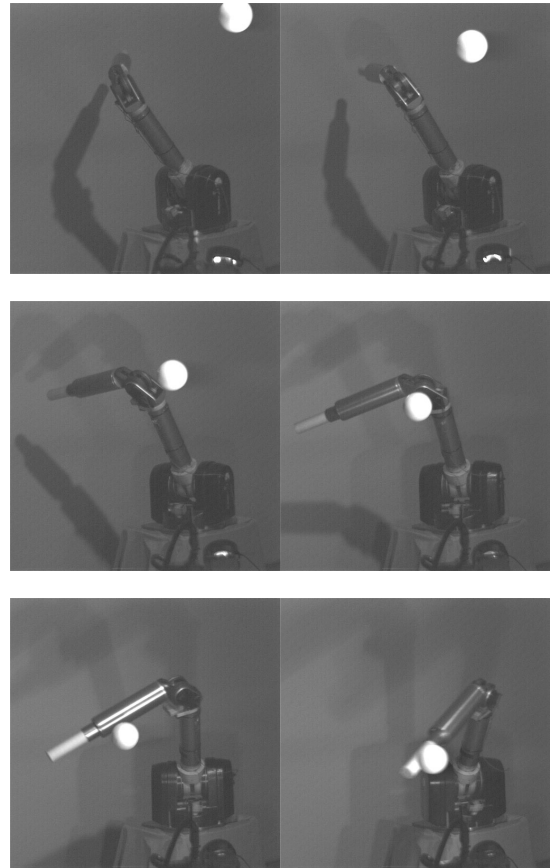


Fig. 4.4 Continuous sequence of pictures

参考文献

- 1) 垣本. ロボット工学ハンドブック, V.2.2 ソフトウェアサーボの実現, pp. 532-534. コロナ社, 1990.
- 2) F. Miyazaki, M. Takeuchi, M. Matsushima, T. Kusano, and T. Hashimoto. *Realization of the Table Tennis Task based on Virtual Targets*. Proc. IEEE Int. Conf. on Robotics and Automation, pp. 3844-3849, 2002.
- 3) W. Hong, and J. Slotine. *Experiments in hand-eye coordination using active vision*. Proc. 4th Int. Symp. on Experimental Robots, 1995.
- 4) Y. Nakabo, M. Ishikawa, H. Toyoda, and S. Mizuno. *1ms Column Parallel Vision System and It's Application of High Speed Target Tracking*. Proc. IEEE Int. Conf. on Robotics and Automation, pp. 650-655, 2000.
- 5) A. Namiki, Y. Nakabo, I. Ishii, and M. Ishikawa. *High Speed Grasping Using Visual and Force Feedback*. Proc. IEEE Int. Conf. on Robotics and Automation, pp. 3195-3200, 1999.
- 6) 並木, 石川. 視覚情報から運動指令へのダイレクトマッピングによる捕球動作の実現. 日本機械学会ロボティクスメカトロニクス講演会, 2A1-1F-C5, 2003.
- 7) 並木, 石井, 石川. 高速センサフィードバックに基づく把握行動. 日本ロボット学会誌, Vol.20, No.7, pp.47-56, 2002.
- 8) X.-Z. Zheng, W. Inamura, K. Shibata, and K. Ito. *A Learning and Dynamic Pattern Generating Architecture for Skillful Robotic Baseball Batting System*. Proc. IEEE Int. Conf. on Robotics and Automation, pp. 3227-3232, 2000.
- 9) A. Ming, and M. Kajitani. *A new golf swing robot to simulate human skill*. Mechatronics, 13, pp. 809-823, 2003.
- 10) M. Bühler, D. Koditschek, and P. Kindmann. *Planning and control of robotic juggling and catching tasks*. Int. J. of Robotics Research, 13(2), pp. 101-118, 1994.
- 11) <http://www.k2.t.u-tokyo.ac.jp/fusion/HighspeedBatting/>