

一般論文

バイラテラル制御システムの構築と制御

中 尾 哲 也^{*1}
田 中 秀 治^{*2}
松 本 雄 一 郎^{*2}
藤 木 亮 彦^{*2}
森 美 彦^{*3}

Design and Control of bilateral control system

Tetsuya NAKAO^{*1}
Shuji TANAKA^{*2}
Yu-ichiro MATSUMOTO^{*2}
Atsuhiko FUJIKI^{*2}
Yoshihiko MORI^{*3}

The remote operations of various fields have a benefit effect on a unified hand/arm master-slave system. We made master-slave system like scissors which is separated. We designed a position and force feedback control law of the master-slave bilateral control system. Bilateral control has four types, our control program adapted flexibly the bilateral four types on RTLinux. The experimental results showed that the angle is satisfactorily controlled by using pole assignment techniques, but the force reaction to the master equipment is not satisfied by proportional control. We propose a new control law of force reaction which will be more stability in future work.

Keywords: bilateral control, force feedback, poles assignment technique

1. はじめに

近年、原子力施設や深海・宇宙などの極限環境での作業が必要とされている。しかし、極限環境での生身の人間による作業にはおおきな危険が伴うため、そのような極限環境で作業するには遠隔操作ロボットが有効である。人が操作する遠隔作業は、突発事態への対応や微妙な力加減などが可能である。また、力覚は人が作業を行う上で重要な役割を果たす。したがって、反力などの力覚情報を操作者に返すことで、操作性や作業効率の向上が期待できる。

遠隔操作を行う場合、マスタ・スレーブシステムが一般に用いられる。マスタ・スレーブシステムは操作者がマスタ装置を操作すると、離れた場所に置かれたスレーブ装置がマスタ装置の動きに合わせて動き実際に作業するシステムである。このとき操作者はカメラの映像などを見て操作をする事になるが細かい部分が判り難く、特に力の加減が難しいと考えられる。バイラテラル制御は、

このマスタとスレーブの位置と力を同時に制御するものである。力覚の提示には、様々な手法が提示されている¹⁾²⁾が、本研究ではハサミ形状のマスタ・スレーブ装置を構築し、制御プログラムを変更することによって様々なバイラテラル制御を行えるようにした。本稿では、まずシステムの位置制御系を構築し、実験検討を行った。次にバイラテラル制御の中でも特に力帰還型について実験を行い、問題点と有用性を確認した。

2. バイラテラル制御システムの概要

2.1 バイラテラル制御の原理

バイラテラル制御系は、制御の対象をマスタの位置、力、スレーブの位置、力の4つとし、マスタとスレーブ間で位置および力どうしを一致させるように制御³⁾⁴⁾を働かせる。バイラテラル制御には主に4つのタイプがあり、それらに柔軟に対応できるように、RTLinux上のリアルタイムモジュール部分のプログラムを変更すれば対応できるようにした。試作したマスタ・スレーブ装置のパラメータを以下に示す。

マスタ側もスレーブ側も装置構成は同じであるので、

*1 久留米工業高等専門学校機械工学科

*2 久留米工業高等専門学校専攻科生

*3 久留米工業高等専門学校学生課

Copyright 2003 久留米工業高等専門学校

パラメータ値は同等とみなした。

- θ : 角度[rad]
- J : モータ軸換算イナーシャ[kgm²]
- D : モータ軸換算粘性摩擦[kgm²/s]
- τ : トルク[Nm]
- N : ギアドモータの減速比[-]
- R : モータ抵抗[]
- K_T : トルク定数[Nm/A]
- K_e : 逆起電力定数[]
- V : モータ印加電圧[V]
- K_p : 比例ゲイン
- K_I : 積分ゲイン
- K_D : 微分ゲイン
- F_m : 操作者がマスタアームに加える力
- F_s : スレーブアームが対象物に与える力

これ以降、各パラメータにある添字の m, s は、それぞれマスタ、スレーブを表すこととする。次にバイラテラル制御の4つの型を示す。

a) 対称型

図 2.1 より、マスタ側もスレーブ側も位置制御系となっており、力覚のフィードバックもスレーブの位置情報に依存するようになっている。この型では、力検出装置が不要なので簡単に構成できるが、慣性力や摩擦力の影響を受ける。マスタに変位が生じると、マスタとスレーブ間に相対変位が生じる。その相対変位を 0 にするようにマスタおよびスレーブに取り付けたモータの駆動電流を制御、両者に駆動トルク、拘束トルクを発生させるように制御するのが対称型バイラテラル制御である。対称型は、マスタおよびスレーブに力加わっている時には、必ず両者に相対変位が存在している。この方式は、力を検出して制御系の位置制御にフィードバックしないので制御が単純である。

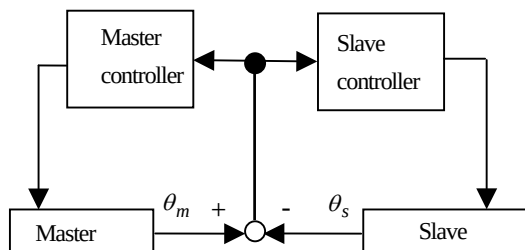


図 2.1 対称型ブロック線図

b) 力逆送型

図 2.2 より、スレーブは位置サーボ系を成し、マスタはスレーブの力センサで測定した反力が伝わる。スレーブの反力はよく伝えられるが、マスタのアーム操作は重くなってしまう。また、マスタのダイナ

ミクスが残ってしまう。しかし、接触したか否かをオペレータは確実に感じ取ることが出来る。対称型バイラテラル制御では、変位と力の間に比例と見なせる関係が成立している必要があり、荷重が与えられても変位しにくい機構には適用できない。例えば、スレーブ装置に高減速装置の付いたモータが取り付けられているような場合、外からモータを回転させる力を加えても、直ちにその力に応じた回転角度が得られないので適用できない。力逆送型は、このような不具合を解決するために、直接スレーブ側の荷重を計測して、その力をマスタの拘束力に反映させるものである。今回試作したシステムもギアドモータで減速比は 50 と比較的高いため、力逆送型では不具合が生じる可能性がある。

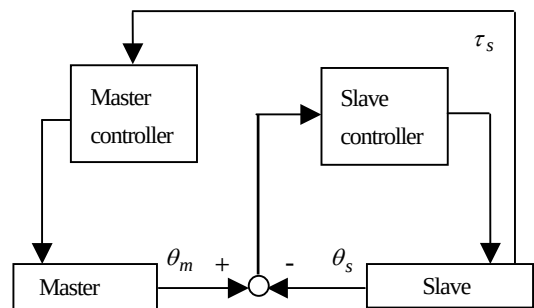


図 2.2 力逆送型

c) 力帰還型

図 2.3 より力逆送型と同等であるが、マスタに力のフィードバックを付加している。このことにより操作感が向上するが、安定性が損なわれやすいことが知られている。力帰還型は、マスタ、スレーブともに機構に力加わっても直ちに变位に反映されにくい場合に用いられる。たとえば、マスタ、スレーブ装置の双方に減速装置や、摩擦が大きい場合、力逆送型制御ではスレーブから帰還された力に対抗する力をマスタ装置に加えても直ちに变位に反映されないため、必要以上にマスタ位置が戻されてしまうことになる。これに対して、力帰還型では、マスタに加えられるべき力が再現されるように位置制御が

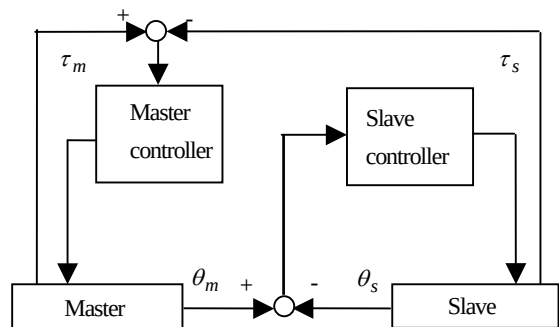


図 2.3 力帰還型

働くので、力逆送型の不具合を回避できる。

d) 並列型

図 2.4 よりマスタとスレーブを独立に安定性を考える事が出来、位相遅れがマンピュレータ 1 台分と小さくなる。このとき、マスタ、スレーブの駆動力は、位置指令生成部の位置指令に依存し、その位置指令は、マスタ、スレーブからの反力から計算する。マスタ側のトルクは、力帰還型に近い形が得られることが知られている⁴⁾。なお、位置指令生成部をどのような制御則にするかは考察する必要がある。

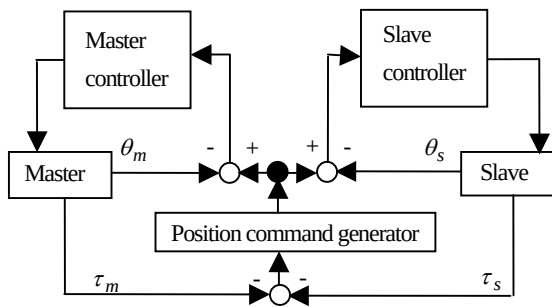


図 2.4 並列型ブロック線図

2.2 実験装置の概要

今回製作した実験装置の概要を図 2.5 に示す。図に示されるように、マスタ装置とスレーブ装置にそれぞれポテンショメータ、駆動用モータが接続されている。各モータには独立してモータドライバがありマスタ、スレーブはそれぞれ同等品である。マスタ、スレーブのトルクは、

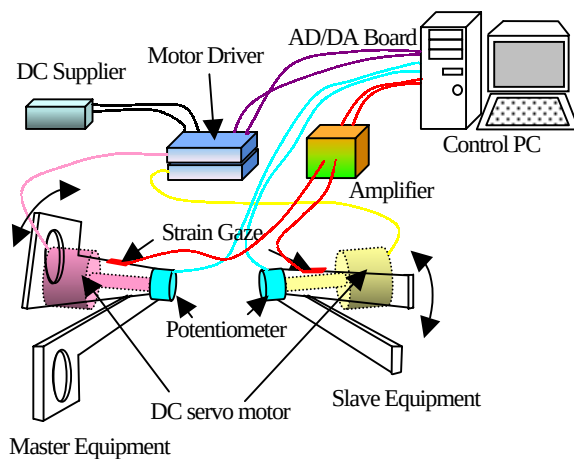


図 2.5 システム概要

歪ゲージによって測定する。歪ゲージからの出力はアンアンプで増幅し、さらに OP アンプで増幅した後、AD 変換して制御用 PC に送られる。また、マスタ部の把持部から回転中心までの距離は一般的なはさみと同じように、

90[mm]とした。次に実験装置に用いた機器の仕様を挙げる。

制御用 PC：自作 AT 互換機 (Pentium4 1.5GHz, OS RTLinux)

AD/DA 変換ボード：PCI-3523A, インターフェイス

歪測定用センサアンプ：F01-3167, 共和電業

DC サーボモータ：SS40E2-L1-50, 澤村電気工業

モータドライバ：MS100V05, 澤村電気工業

次に実際のシステム概観を示す。図 2.6 に示されるようにマスタ装置、スレーブ装置とも土台に固定されており、1 自由度しかない。制御発散時の対策のためリミットスイッチを設け非常停止するようにしている。また 90 度以上開口しないようにソフトウェア側でリミッタを設けた。

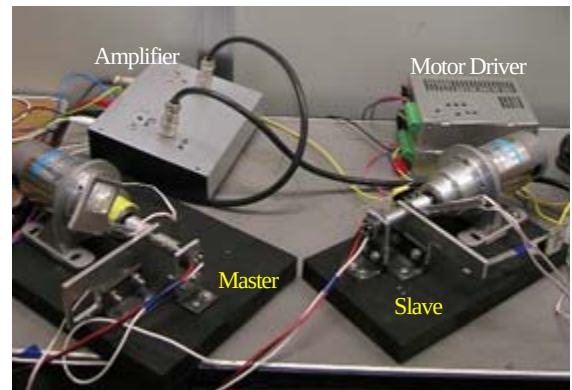


図 2.6 制御システム概観

2.3 位置制御系の設計

位置制御系には状態フィードバック制御を採用した。マスタ側は位置制御を行わないものとして、スレーブ側の運動方程式は、

$$J_s \ddot{\theta}_s + D_s \dot{\theta}_s = \tau_s \quad (2.1)$$

$$\tau_s = \frac{K_{Ts}}{N} V - \frac{K_{Ts} K_{Es}}{R_s} \dot{\theta}_s \quad (2.2)$$

となる。ただし、モータの電気的方程式は、DC サーボモータの電気的時定数が小さいため無視する。偏差をなくすために積分器を入れ制御則を以下のようにする。

$$V_s = K_{ps} (\theta_m - \theta_s) + K_{Is} \int (\theta_m - \theta_s) dt + K_{ds} (\dot{\theta}_m - \dot{\theta}_s) \quad (2.3)$$

従って伝達関数は以下のようになる。

$$\frac{\Theta_s}{\Theta_m} = \frac{\beta(K_{ps} + K_{Is}/s + sK_{ds})}{J_s s^2 + \alpha s + \beta(K_{ps} + K_{Is}/s + sK_{ds})} \quad (2.4)$$

ただし、

$$\alpha = \left(D_s + \frac{K_{Ts} K_{Es}}{R_s} \right), \beta = \frac{K_{Ts}}{NR_s} \quad (2.5)$$

である。各ゲインは極配置法⁵⁾を用いて求める。指定極を、

s_1, s_2, s_3 とすると、特性方程式は、

$$(s - s_1)(s - s_2)(s - s_3) = 0 \quad (2.6)$$

となり、解と係数の関係から、

$$K_{Ps} = J_s \frac{s_1 s_2 + s_2 s_3 + s_3 s_1}{\beta}$$

$$K_{Is} = -J_s \frac{s_1 s_2 s_3}{\beta} \quad (2.7)$$

$$K_{Ds} = -J_s \frac{(s_1 + s_2 + s_3 + \alpha / J)}{\beta}$$

となる。制御プログラムでは、極を指定すれば各ゲインが計算されるようにした。

2.4 GUI インターフェイスの作成

今回、マスタ・スレーブの制御は RTLinux で行った。RTLinux とは LinuxOS と共存可能なリアルタイム OS⁶⁾である。しかし、厳密な意味ではリアルタイム OS ではなく、RTLinux が提供するものはスケジューラとプロセス間通信のみである。RTLinux は、仮想マシンを Linux に提供し、Linux を優先順位の低い 1 つのリアルタイムプロセスとして実行することで、リアルタイム処理と従来の LinuxOS の処理の共存を実現している。このリアルタイム処理というのは、ある幅で処理開始タイミングと処理終了までの時間制限を保証した処理のことである。今回はフリーの GTK+ を使用して GUI を作成した⁷⁾。RTLinux でのリアルタイム制御は、モジュールと呼ばれるプログラムが、周期的に動作することにより確保している。リアルタイムモジュールは、非リアルタイムモジュールからは呼び出しが不可能なため、FIFO と呼ばれるところからデータのやり取りをしなければならない。今回までで、その部分を作成し、制御結果が逐次表示されることによって、制御の状況を把握できるようにした。制御系が発散した場合のリミッタも設置した。以下に今回作成したインターフェイス画面を示す。

図 2.7 に示すように、データは画面上に逐次表示される。画面左側にマスタ、スレーブの位置とトルクを AD 変換された電圧値と換算した値を表示している。画面右側にモータへの指令電圧、経過時間を表示している。その間のエントリに極を入力することによりフィードバックゲインを自動的に計算するようになっている。また、上部左右に分かれた部分にグラフィック出力をすることにより視覚的に制御の状態を知る事が出来るようにした。

グラフィックは図中向かって左側が、マスタ、スレーブの位置を表示しており、右側にマスタ、スレーブのトルク値を表示している。これにより、実機制御時にもリアルタイムで制御の状態量を知る事ができるようにした。GUI 画面下のボタンは、右から歪ゲージドリフトのリセット、制御開始、制御停止、制御結果の保存、プログラ

ムの終了の役割をもつ。今回、リアルタイムモジュールの周期は 5[msec]としたので、サンプリング間隔は 5[msec]となる。また、Linux 側は 10[msec]ごとに画面の書き換えを行うようにするが、データの遅延を保証するために、25[msec]毎にリアルタイムモジュールから Linux 側へ送るようにプログラミングした。

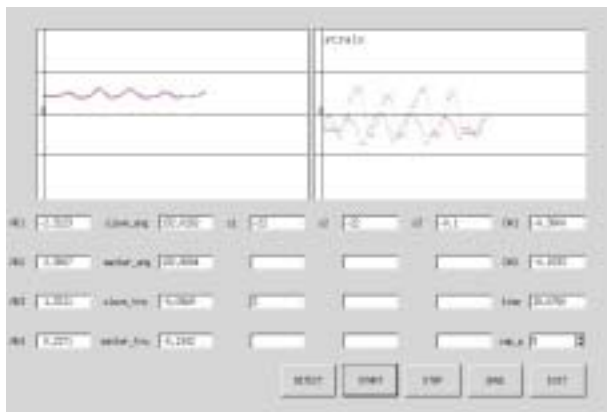


図 2.7 作成した GUI 画面

2.5 バイラテラル制御の座標系

図 2.8 にバイラテラル制御における座標系を示す。

マスタ、スレーブはいずれも下の部分が固定されているために相対的に開く動作の時を正とし、トルクについては、図に表記されているように考えた。制御則によって計算された電圧は、モータドライバへの指令値として変換され、対応するように調整した。

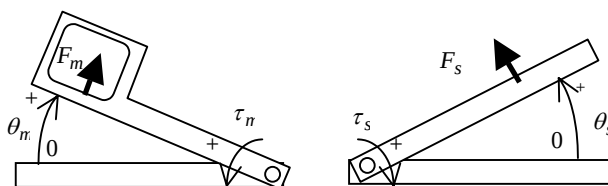


図 2.8 マスタ・スレーブの座標系

3. バイラテラル制御システムの性能評価

3.1 スレーブ位置制御について

スレーブの位置制御として、マスタ側の角度を 45 度を設定し、0.5[s]後にステップ入力を与えた。

-22, -12, -8 とは、それぞれ、 $s_1=s_2=-22, s_3=-0.1$, $s_1=s_2=-12, s_3=-0.1$, $s_1=s_2=-8, s_3=-0.1$ とした場合である。 s_3 のみをこのような極にしたのは、小さく値をとりすぎると発散するためである。結果より、極を小さくすることによって応答が早くなっていることが分かる。しかし、-22 と -12 の場合ではあまり差が見られない。定常特性は -22 にした場合、偏差が少ないのでこれを採用した。また、パラメ

ータの調整によっては制御が発散してしまうこともあった。これは、マスタ、スレーブ両方のポテンショメータのノイズが考えられる。また、モータイナーシャはカタログ値を用いているが経年変化して、パラメータ値が変化した可能性がある。

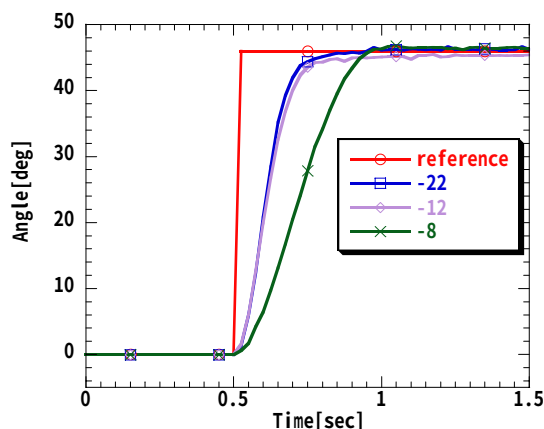


図 3.1 ステップ応答の結果

3.2 バイラテラル制御について

バイラテラル制御系では、力帰還型を採用した。最初にスレーブ側とマスタ側の歪ゲージからのトルク情報から、以下の比例制御則で制御を行った。

$$v_m = K_{pm}(\tau_m - \tau_s) \quad (3.1)$$

実験の結果無負荷時の操作感は劇的に向上した。図 3.2 に制御結果を示す。図に示すとおり、スレーブ側は無負荷なのでトルクは発生していない。

マスタ側は操作によってトルクが発生し、その分セトルク制御則から開閉する際のトルクをモータからアシストしてもらう形となり、操作感が向上した。

スレーブ側になにかものを挟んだ場合、スレーブのトルクがマスタ側に帰還するため、マスタをなおも閉じようとするとマスタ側のトルクからそれに相当するトルクが帰ってくる。これは制御式(3.1)から致し方の無いことで、これでは挟んだという感覚は戻ってくるが、挟んだものが柔らかいのか硬いのかの判別が出来ないことが判明した。また、接触した瞬間にスレーブ側に多大なトルクがかかるとそのままマスタ側に返され、マスタ側をきちんと操作していなければ、反動で位置がずれてしまい、制御系が発散する現象も見られた(図 3.3)。そこで、それらの問題を解消するために、トルクに閾値を設け、接触、把持、握り込みの段階で場合分けを行い、かつ比例ゲインを調整した。スレーブ側にスポンジを挟んだ場合と硬いものを挟んだ場合の結果をそれぞれ図 3.4、図 3.5 に示す。図 3.4 では、マスタをまず開いて、1.8 秒後に対象物を把持している。把握後、スレーブ側トルクが振動している

が、これは把握がうまくできていない為である。

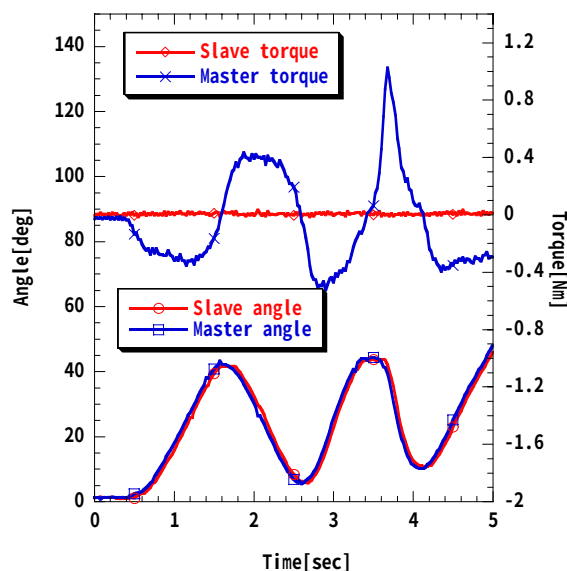


図 3.2 無負荷時のバイラテラル制御結果

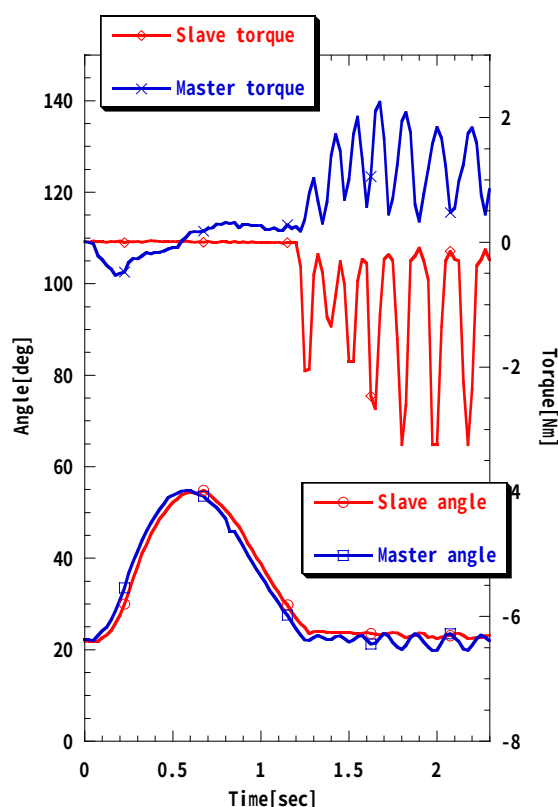


図 3.3 発散した例

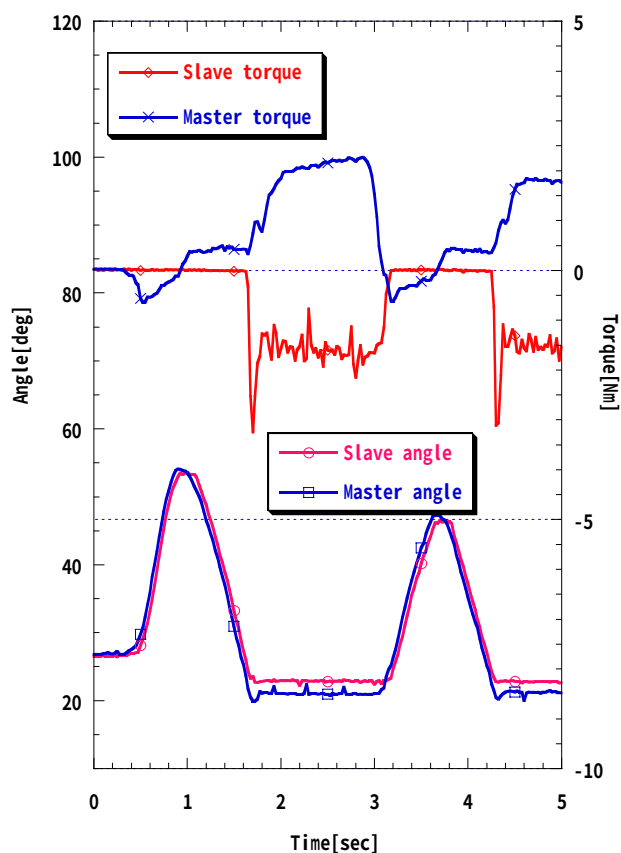


図 3.4 硬いものを把持した場合の制御結果

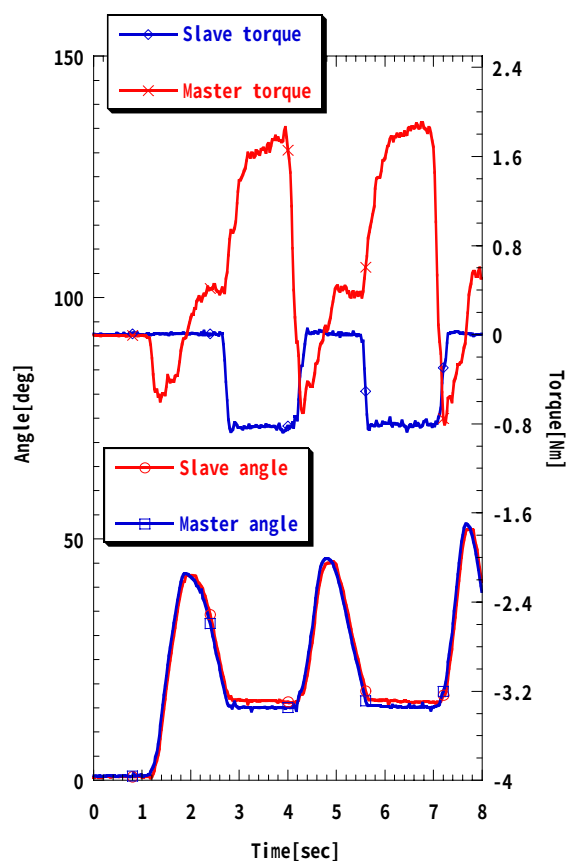


図 3.5 柔らかいものを把持した場合の制御結果

また、位置制御についても偏差があり、この偏差分がスレーブの指令値となってスレーブ側に反力を生じせしめられている。マスタを開こうとすると、スレーブトルクは 0 になり、制御がうまく行われていることがわかる。4 秒後にまた把握しているが最初の把握と同じように再現されている。図 3.5 では、まずマスタを開口し、対象物を入れる。それから 2.8 秒後に対象物を把握した。図 3.4 と比較してスレーブ側のトルク変動が小さくなっている。これは対象物に弾力があるためだと考えられる。スレーブトルクが変動しないのに、マスタトルクが変動しているのは、制御側の比例ゲインを変化させているからである。この実験結果から、柔らかいと認識するには、操作者の主観に委ねられる部分が多かった。また、対象物を離す際に、接着剤でくっついたような感触が残ってしまった。これも条件分岐のチューニングを行うことで改善の余地があると考えられる。

4. 結 言

今回の実験結果、以下のような結論が得られた。

バイラテラル制御系を構築し、位置制御、力制御の両方が出来る事を確認した。

位置制御については、システムのパラメータ同定に問題があると考えられるが、極を適宜調整することにより満足のできる応答特性を得た。

力制御については、力帰還型でバイラテラル制御を行うと、接触時に反力が大きくなり、安定性を失う事があった。スレーブが把持する対象物の弾性によって反力を正確にフィードバックするのは力帰還型では困難であることが示唆された。しかし、接触時には、マスタ側に反力が来る事は確かめられたので、している。把握後、スレーブ側トルクが振動しているが、これは把握がうまくできていない為である。また、位置チューニングによって改善されるものと思われる。また、今回力制御側は比例制御のみでマスタとスレーブに掛かっている荷重を常にモニタしゲインや条件分岐をする手法を試みた。その結果、限定的利用に際しては、本バイラテラル制御系は有効であると結論づけられる。今後の課題として、システムパラメータの正確な同定と、バイラテラルの対称型・力逆送型の実験評価と並列型制御方法の検討を挙げる。

参 考 文 献

- 1) マスタ-スレーブにおける弁別閾を考慮した力のスケーリング pp.402-407, 200 Vol.19 日本ロボット学会誌

- 2) 弾性要素を用いた力覚提示機構を有するマスタ・スレーブシステムの制御法 日本機学会学会ロボティクス・メカトロニクス講演会 講演論文集 2P1-103,2002
 - 3) 戸羽篤也, 桑野晃希; “バイラテラル方式による力覚制御技術” 北海道立工業試験場報告 Vol.297, 1998
 - 4) 宮崎友宏荻原史郎: “バイラテラル・マスタ・スレーブ・マニピュレータの並列型制御方式”, 日本ロボット学会誌 7 巻 5 号(1998)
 - 5) 増淵正美; “システム制御” pp.167-184, コロナ社, 1987
 - 6) RTLinux テキストブック ハードリアルタイム機能を使いこなす Matt Sherer/著 FSMLabs Technical Staff/著 西谷年代/訳 小山友里/訳吉元純子/監修 森友一朗/監修 FSMLabs Japan/総合監修 CQ 出版 2003 年 4 月
 - 7) GTK+ではじめるXプログラミング 森 友一朗 (著), 馬場秀忠 (著), 薬師 輝久 (著), 木村 新, 岡本 教佳 技術評論社
- (2004 年 3 月 5 日 受理)

(2004年3月5日 受理)

~