

力覚デバイスを用いたロボットアームの遠隔操縦システムの開発

Development of Remote Operation System for a Manipulator by a Haptic Device

○神原利彦*, 上野晃*

○Toshihiko Kanbara*, Akira Uwano*

*八戸工業大学

*Hachinohe Institute of Technology

キーワード: 遠隔操縦 (remote operation), 力覚デバイス (haptic device), 力覚センサ (haptic sensor), 仮想メス (virtual mes), 反力 (reaction force)

連絡先: 〒031-8501 青森県八戸市大字妙字大開 88-1 八戸工業大学 工学部 電気電子システム学科
神原利彦, Tel.: (0178)25-8133, Fax.: (0178)25-1430, E-mail: kanbara@hi-tech.ac.jp

1. 序論

近年、高齢化社会のため高齢な患者が増加しつつあり、相対的に医師不足の問題が顕著になっている。また、医師の多くが都市圏に在住する傾向にあり、地方の過疎地に常駐する医師は減少する一方である。そこで、本研究では都市部に居る医師が、ネットワークを経由して過疎地に設置されたロボットアームを遠隔操縦して、治療や手術を行うシステムの構築を目指す。

このような研究は「遠隔手術 (remote operation)」と呼ばれており、盛んに研究されている。一部実用化もされており、米 Intuitive Surgical 社が製造販売している「ダビンチ (Da Vinci)」や「ゼウス」がよく知られている。これらのシステムでは、医師が遠隔地から送られてくる患部の 3D 画像を立体視で見ながら、レバーやボタンを操作して遠隔地のロボットアームを操縦して手術を行っている。日本では、九州大学病

院で 2001 年から 2002 年にかけて、治験として、62 例の胸腹部の遠隔手術実験が行われたことが報告されている。このように、多くの遠隔手術の研究が視覚情報を重要と考えており、カメラで高解像度かつ鮮明な画像を、なるべく少ない遅延で高速に伝送することに注力している。あるいは、操作系や制御系を工夫してより精密な医療用の動きを実現することを目指している。だが、カメラですべての情報が伝送できるわけではない。例えば、力覚の情報をカメラで得ることはできない。

山内ら¹⁾は、操縦者側もロボット側も両方に力覚デバイスを用いて、マスタ・スレーブ型の触感伝達システムを開発した。この研究では粗さや摩擦・硬軟を含む多様なテクスチャ感覚を伝達できることが示されている。だが、このシステムでは触感しか感じるができない。そこで、本研究では力覚の情報を伝送する遠隔手術システムを構築する手法を提案する。具体的

には、ロボットアームに取り付けた医療用のメスが患者の肉や皮膚を切り裂く際にメスを持つ手にかかる反力を、遠隔地から仮想メスを操作する医師が感じることができるシステムの構築を目指す。本研究は、沼田²⁾らが提案した手法を改良し発展させたものである。

2. 手法

2.1 手法概要

本手法の概要を Fig.1 に示す。医師のいる操縦者側と、患者とロボットがいるロボット側の2箇所は、距離的に離れているが、操縦者側に設置されたマスター PC とロボット側に設置されたスレーブ PC とがインターネット経由で常時接続されている。まず、操縦者側で医師が力覚デバイス进行操作する。すると、その操作位置の情報をマスター PC が読み取り、インターネット経由でロボット側にあるスレーブ PC へと伝送する。スレーブ PC はロボットアームを駆動し、その先端を伝送されてきた位置情報の箇所へ正確に移動させる。ロボットアーム先端には力覚センサが取り付けられており、アーム先端にかかる反力やモーメントが計測され、それらの計測値がスレーブ PC に伝送される。スレーブ PC は、それらの計測値をインターネット経由でマスター PC に伝送し、マスター PC が、それらの反力情報を基に力を発生させて医師に呈示する。

Fig.??では、視覚情報の伝送を省略しているが、患部とアーム先端をステレオカメラで撮影した両眼立体視画像が、インターネット経由で操縦者側に送られているものとする。

2.2 情報共有の仕組み

前述のシステムを実現するためには、いくつかのプロセス間での情報共有が必要である。例えば、力覚デバイスから操作値（位置情報）を読

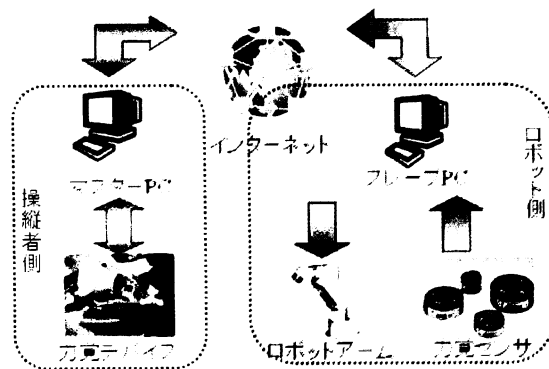


Fig. 1 システム概要. Overview of our system.

み取るプロセスとロボットアームを動作させるプロセスとの間では、アーム先端の位置情報を共有しなければならない。同様に、力覚センサから力やモーメントを読み取るプロセスと力覚デバイスで力を発生させるプロセスは力やモーメントの情報を共有しなければならない。

本手法では、情報共有の仕組みとしてプロセス間通信を使用する。概念図を Fig.2 に示す。これは、1つのサーバープロセスに対し、多くのクライアントプロセスが接続し、サーバープロセスが管理する共有メモリにクライアントプロセスが書き込んだり読み込んだりする手法である。共有メモリに書き込むクライアントプロセ

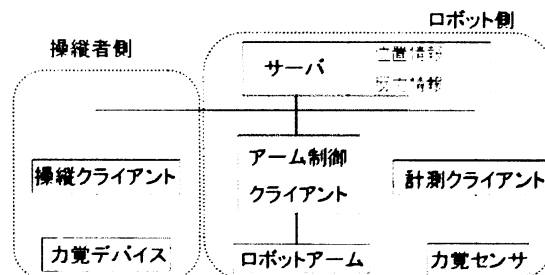


Fig. 2 プロセス間通信. Communication between processes.

スも、読み出すクライアントプロセスも、極短い処理時間サイクルで次々に新しい値を書き込む処理や新しい値を読み出す処理を繰り返す。そうすることで、書き込んだクライアントプロセスの持っていた情報が読み出した全てのクラ

クライアントプロセスに伝送され、ほぼ同時に複数のクライアントプロセスが同じ情報を共有できることになる。ただし、クライアントプロセスの書き込む処理の周期と読み出す処理の周期が一致しているとは限らないので、場合によっては「書き込んだが、誰にも読み出されることなく次の処理で新しい値に上書きされて消えた情報」や「何度読み出しても新しい値が共有メモリに書き込まれておらず、同じ値ばかりが読み出される状態」などが発生する。

共有メモリに書き込まれる情報は、操縦者が操作するロボットアーム先端の位置情報と、力覚センサが計測した反力情報の二種類である。それらの情報の次元数は、実際に使用する力覚デバイスや力覚センサの仕様に依存する。本手法では、アーム先端の位置情報については、位置3次元 (x, y, z) 、姿勢3次元 (α, β, γ) の合計6次元を共有メモリで記録する。力覚センサの計測情報も直交座標の3次元軸に沿った力3次元 (F_x, F_y, F_z) とその軸を中心に回転するモーメント3次元 (M_x, M_y, M_z) との合計6次元で記録する。

2.3 操縦クライアント

本手法では、力覚デバイス进行操作してロボットアームの先端を動かす。そのため、力覚デバイスを動かした場合のその操作量を計測し、その操作量をプロセス間通信で、サーバーの共有メモリに書き込まなければならない。また、共有メモリから反力情報を読み取って、力覚デバイスに力を発生させて、その力を操縦者に呈示しなければならない。

市販されている力覚デバイスの多くは、USB接続でPCに接続して使う方式で、操作量を計測したり、具体的な力を発生させるソフトウェアをSDKの形で提供している。そのSDKを基にソフトウェアを開発することで、操作量の計測機能と力の発生機能を実現する。

2.4 アーム制御クライアント

本手法では、サーバーの共有メモリからアーム先端の操作量情報 (x, y, z) を読み出して、その値通りにアーム先端を移動させるソフトウェアが必要である。また、最近のロボットアームの多くが、アームのリンク部分は長さを固定しておき多くの関節部分を回転させる回転機構中心の構造を持っている。そのため、共有メモリから読み出した情報 (x, y, z) へ正確にアーム先端を移動させるために適切な関節の回転角度を算出しなければならない。

関節の適切な回転角度を計算するには、逆運動学を用いる。逆運動学については、文献³⁾に詳しく記述されているが、解が無い場合(アーム先端が届かない位置を指定)や複数解が存在する場合などがあり、一意的な解が得られるとは限らないことが知られている。

本研究では、解が得られない場合には、向きを同じにしてアーム先端が届く最大位置で解を求めることとした。例えば、指定されたアーム先端が (x, y, z) で解がない場合は変数 k を用いて、 (kx, ky, kz) で解が求まるかどうかを調べる。 k を1から少しずつ小さくしていき、最も早く解が求まった場合を逆運動学の解とする。

複数解が存在する問題については、連続的に解を求めた履歴を記録しておいて、その履歴に最も近いものを解とする。これは、指定する位置情報がいきなり大きくは変わらないので、求められる解も履歴に沿ってゆるやかに変わるだろうという予測から、そのようにしている。例えば、 (x_i, y_i, z_i) という位置に、 n 個の回転関節を持つロボットのアーム先端を持っていく解 $(\theta_{i,1}, \theta_{i,2}, \dots, \theta_{i,n})$ が唯一解で求まり、次に指定された $(x_{i+1}, y_{i+1}, z_{i+1})$ という位置にアーム先端を持っていく解として $(\theta_{i+1,1}, \theta_{i+1,2}, \dots, \theta_{i+1,n})$ と $(\delta_{i+1,1}, \delta_{i+1,2}, \dots, \delta_{i+1,n})$ の2つの解が求まったとする。この場合は、履歴として保存しておいた $(\theta_{i,1}, \theta_{i,2}, \dots, \theta_{i,n})$ との差 E_1, E_2 を次式で

求めて、差の小さい方を選ぶ。

$$E_1 = \sqrt{\sum_{j=1}^n (\theta_{i+1,j} - \theta_{i,j})^2}$$

$$E_2 = \sqrt{\sum_{j=1}^n (\delta_{i+1,j} - \theta_{i,j})^2}$$

もし、 $E_1 < E_2$ であれば、 $(\theta_{i+1,1}, \theta_{i+1,2}, \dots, \theta_{i+1,n})$ が選ばれ、そうでなければ $(\delta_{i+1,1}, \delta_{i+1,2}, \dots, \delta_{i+1,n})$ が選ばれる。3つ以上の複数解がある場合は、差 E が最小となる解が選ばれる。

このようにして選ばれた解（関節角度）を関節にあるモーターの制御系に伝送し、指定された角度へと正確に回転させることで、アーム先端は指定された位置へと動かされる。

2.5 計測クライアント

Fig.2に示したように、力覚センサから出力される計測値はスレーブPCが読み取る。スレーブPCは計測値をネットワーク経由でサーバープロセスの共有メモリに書き込む。これらの反力情報は時々刻々と短い時間サイクルで繰り返し共有メモリに書き込まれる。力覚センサはアナログ出力のものが多く、その計測周期はスレーブPCに装着するA/D変換器の性能（時間分解能）に依存する。

2.6 ネットワーク

プロセス間通信は、WinSock2などのライブラリを使用して実装する。IPアドレスとポート番号を指定して接続する形式とする。後述する実験では、何の認証も行わずに接続しているが、ネットワークのセキュリティを考慮すると、将来的にはパスワードによる認証などを接続時に行うべきである。

操縦者側とロボット側の距離が大きく離れていれば、当然ながらネットワーク遅延が発生する。例えば、操縦者が力覚デバイスを操縦して、遠隔地のロボットアームが動いて、メスが肉を

切り裂いているのに、ネットワーク遅延で反力情報がなかなか届かず、遅延時間の後に操縦者が反力を感じるという可能性もある。この伝送遅延の問題は、力が時系列でかかり続ける場合には、遅延時間を考慮してカルマンフィルタなどで予測することで対応可能であるが、手術のような細かい作業では、そのような場合だけとは限らない。

3. 実験

3.1 実験装置

実験装置を以下に述べる。力覚デバイスとして、Novint製Falconを使用した。力覚センサとして、ニッタ製TFS12-25を使用した。ロボットアームとして、アールティ製RT0002を使用した。マスターPCとして、Aopen製A2661-S(Celeron2.4GHz/RAM2M)を、スレーブPCとして、IBM製ThinkPad X40(Pentium M 1.0GHz/RAM1M)を使用した。力覚センサからの信号を増幅する専用アンプとしてニッタ製AMP-15を使用した。さらに、アンプからのアナログ出力をPCに取り込むためにCONTEC製のアナログ入力PCカードADA16-8/2(CB)Lを使用した。ネットワーク回線は大学内の有線LANを使用した。開発用ソフトウェアには、Microsoft製Visual C/C++ 6.0を使用した。力覚デバイスに付属の開発用ライブラリには、Novint製SDKのHDALを使用した。

3.2 予備実験：力覚デバイスの位置情報を共有メモリに書き込む

Novint製SDKを使用して、プロセス間通信のサーバープロセスが管理する共有メモリに、力覚デバイスのグリップ部の位置情報を書き込むソフトウェアを作成した。実験に用いたFalconの外観をFig.3左に示す。作成したソフトウェアの実行画面をFig.3右に示す。Fig.3右の灰色



Fig. 3 Falcon と実験画像. Haptic Device Falcon and Experimental Image.

の正方形 CG が奥に位置し固定されている。手前にある水色の球体が、力覚デバイス Falcon の操作グリップと連動している。操縦者が、操作グリップを右手で握り、上下 (Y 方向) 左右 (X 方向) 手前奥 (Z 方向) へと動かすと、動かした量だけ画面上の水色の球体も移動する。手前に動かすと、球体が大きく表示され、奥に動かすと球体が小さく表示される。この球体の位置情報 (x, y, z) を共有メモリに書き込む実験を行い、スレーブ PC 側から、その位置情報 (x, y, z) を読み出すことができた。

Fig.3 左に示したように、被験者はグリップを握ってグリップの位置を動作させることができるが、グリップを傾けたりすることはできない。したがって、アーム先端の傾き角度 (α, β, γ) を操作することはできない。これは、Falcon の仕様限界であり、傾き角度まで操作するのであれば、別の力覚デバイス (例えば、Sensable 製 Phantom- Omni のような装置) を必要とする。

3.3 予備実験：逆運動学の実装

実験に用いたロボットアーム RT0002 の外観を Fig.4 に示す。5 個のサーボモーターを備えている 5 自由度のロボットアームであるが、そのうち 2 個はアーム先端の姿勢を変化させるものなので、先端位置を決める自由度は 3 である。これらの連続した 3 軸の交点は一点で交わる構造を持っているので、逆運動学で、解を求めることができる³⁾。つまり、アーム先端位置 (x, y, z) を入力すれば、その位置に先端が来るような回転角度 $\theta_1, \theta_2, \theta_3$ を計算できる。理論式は省略



Fig. 4 ロボットアーム RT0002. Manipulator RT0002.

するが、求めた式で 3 つの関節角度 $\theta_1, \theta_2, \theta_3$ を求め、その関節角度に回転させた場合に先端位置が正しい 3 次元位置 (x, y, z) に来るかどうかを CG で確認するシミュレーション実験を行った。CG の表示画面を Fig.5 に示す。

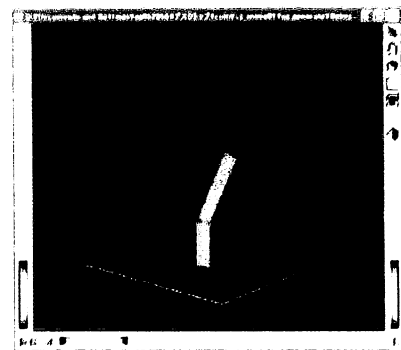


Fig. 5 逆運動学のシミュレーション. Simulation of Inverse Kinematics.

さらに、プロセス間通信のサーバプロセスが管理する共有メモリから (x, y, z) を読み出し、逆運動学で $\theta_1, \theta_2, \theta_3$ を求めて CG で表示するシミュレーションを行った。その結果、 (x, y, z) がアームが届く範囲内にある場合には正常に計算されるが、少しでも範囲外に出る場合には計算が破綻した。この問題に対しては、2.4 節で述べた手法の実装で解決を試みる。

3.4 予備実験：力覚センサによる力の知覚

ナイフの刃が物体を切り裂く時の反力を力覚センサで計測するための治具を設計した。Fig.6

に力覚センサと、それに取り付けるナイフの刃と、握るための柄のデザイン画を示す。中央の青い部分が力覚センサで、灰色で示した部分の左側が柄で、右側がナイフである。この構造により、人間の手やロボットアーム先端のハンドが柄を握って、ナイフを動かし、物体を切り裂く際に発生する反力を計測する。現在、治具を製作中である。

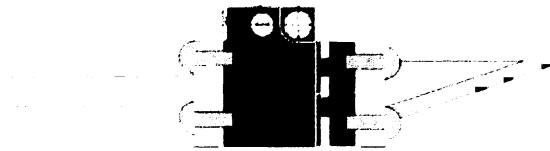


Fig. 6 力覚センサに取り付ける柄とナイフ。
Knife and Bar with Haptic Sensor.

センサの出力は3軸の力 (F_x, F_y, F_z) とそれらの軸まわりのモーメント (M_x, M_y, M_z) であるが、これらの出力値はアナログ電圧であり、専用アンプで増幅されてから、アナログ入力カードを経由してスレーブPCにデジタル信号として読み込まれる。

アナログ入力カードには、CONTEC製ADA16-8/2(CB)Lを使用したが、このカードのサポートソフトウェアAPI-AIO(WDM)を使用して、プロセス間通信のサーバプロセスが管理する共有メモリに力覚センサの計測値 (F_x, F_y, F_z) と (M_x, M_y, M_z) を書き込むソフトウェアを開発した。

3.5 予備実験：力覚デバイスでの力の発生

力覚デバイスFalconに付属のライブラリHaptic Device Abstraction Layer(HDAL)を使用して、共有メモリに書き込まれた力の値を読み取ってその力を被験者に感じさせるソフトウェアを開発中である。

このデバイスで使用している力学モデルは、 $F = mA$ というニュートンの法則と $F = kX + cV$ というバネ・ダンパーモデルの2種類であ

る。どちらのモデルが、本研究で発生させたい力（刃物で切り裂く際にかかる反力）の呈示に適しているかを、テスト用のソフトウェアを作成し実験中である。

4. 結論

4.1 まとめ

遠隔手術システムにおいて、操作者が手術道具にかかる反力を感じることができるシステムの構築手法を提案した。提案手法のうち、(1) 力覚デバイスから位置情報を共有させる部分、(2) 共有された位置情報の通りにアーム先端を動作させる部分を実装した。(3) アーム先端に取り付けた、力覚センサ付のナイフからの反力情報を共有させる部分と(4) 共有された反力情報を力覚デバイスで発生させる部分については、実装途中である。

4.2 今後の課題

実験において、未実装の部分の実装を完了させることを今後の課題とする。また、ネットワーク遅延の問題の解決法を考案することや、力覚センサFalconのグリップ部では、ナイフの姿勢 (α, β, γ) を操縦できない問題点を解決する方法を考案することも今後の課題とする。

参考文献

- 1) 山内敬大、昆陽雅司、岡本正吾、日高佑輔、前野隆司、田所諭: マスタ・スレーブ型触感伝達システムの開発、第14回日本バーチャルリアリティ学会論文集、(2009)
- 2) 沼田政紀、神原利彦: 力覚デバイスを用いたロボットアームの操縦、平成20年度八戸工業大学電子知能システム学科卒業論文、(2009)
- 3) John J. Craig(著) 三浦宏文(訳) 下山勲(訳): ロボティクス機構・力学・制御,91/124, 共立出版(1991)