

ORiN2 SDK を用いた小型多関節型ロボット VE026A の軌道追従制御

○永田 寅臣, 永富 智也, 林 将平, 大塚 章正 (山口東京理科大学 工学部 機械工学科)
渡辺 桂吾 (岡山大学大学院 自然科学研究科)

Trajectory Following Control of an Articulated Robot VE026A Incorporated with ORiN2 SDK

○ Fusaomi NAGATA, Tomoya NAGATOMI, Shohei HAYASHI and Akimasa OTSUKA,
Tokyo University of Science, Yamaguchi
Keigo WATANABE, Okayama University

Abstract: Industrial robots with an open architecture have been applied to several tasks such as machining, sanding, polishing and so on. The open architecture allows system engineers to realize desirable teachingless operation. However, transplantation, i.e., porting of an application software to other makers' industrial robots is not easy even though having an open architecture, because the specifications of interface between PC and robots differ respectively. That is the reason why the standardization of industrial robots' interfaces is now expected and desired from robotic users. Recently, ORiN (Open Robot/Resource interface for the Network) is gathering attention as a promising middleware for the standardization of robotic interface. In this paper, trajectory following control based on cutter location data (CL data) is developed on ORiN interface for an educational small robot VE026A with six-DOFs. The ORiN middleware is available on the robot. Trajectory following control of an end-effector attached to the flange of the robot is a basic but important strategy for robotic application without using any robot languages. The design with ORiN interface and its simulation experiment are shown, and its portability is discussed.

1. はじめに

ロボットアプリケーションの再利用性や互換性を高めるために産業オートメーションにおける標準化のための取り組みがなされている^{1,2,3}。例えば、オープンアーキテクチャ型の産業用ロボットが登場して以来、これまでに加工や研磨などの自動化工程に積極的に応用されてきた。サーボ系やキネマティクスなどのオープン化により、教示レス化が可能となったことが大きな理由といえる。しかしながら、オープンアーキテクチャ型のコントローラが搭載されていても、あるロボットシステムで開発されたアプリケーションを他メーカーのロボットシステムに移植することは容易ではない。これは、制御用 PC とロボットとのインタフェース仕様がメーカーごとに異なるためである。このような背景から産業用ロボットのインタフェースの標準化が期待されているところであるが、最近、ロボットインタフェースの標準化のための有望なミドルウェアとして ORiN (Open Robot/Resource interface for the Network) が注目されている^{4,5,6,7}。

筆者らは、オープンアーキテクチャ型産業用ロボットに適用できる「CL データに基づく軌道追従制御系」を提案しているが、これによりロボットメーカーが提供するロボット言語を用いることなく研磨や加工などのロボットアプリケーションの開発が可能となった。本稿では、ORiN

ミドルウェアが搭載された 6 自由度小型多関節型ロボット VE026A に対して、既に提案している多軸制御用 CL データに基づく軌道追従制御法⁸の適用を試み、WINCAP III を用いて移植性の検証実験を行ったので報告する。

2. 過去の開発システム

最初に、オープンアーキテクチャ型の産業用ロボットの API を用いて開発したロボットアプリケーションの事例を幾つか紹介する⁹。



Fig. 1 Polishing robot for PET bottle blow molds developed based on JS10 provided by Kawasaki Heavy Industries, LTD. in 1997.

まず、図 1 には川崎重工業の JS10 を用いて開発したペットボトルブロー成型用金型の磨きロボットを示す。 Visual C++ の環境でコールできる API が提供されており、アーム先端の位置と姿勢は次式で与えられる相対同次変換行列 ΔT を 4 ms 毎にサーボ系に送ることで制御できていた。

$$\Delta T = \begin{pmatrix} \Delta n_x & \Delta o_x & \Delta a_x & \Delta p_x \\ \Delta n_y & \Delta o_y & \Delta a_y & \Delta p_y \\ \Delta n_z & \Delta o_z & \Delta a_z & \Delta p_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1)$$

図 2 には自由曲面を研磨するために開発したロボットサンダーを示す¹⁰。このシステムは、川崎重工業の FS20 をベースに開発している。エアー駆動の研磨工具の姿勢は、CL データ内の法線ベクトルを参照することで曲面に対して法線方向となるように制御していた。FS20 のアーム先端の位置と姿勢は JS10 と同様に ΔT を用いて実時間制御が可能であった。



Fig. 2 Robot sander for wooden workpieces developed based on FS20 provided by Kawasaki Heavy Industries, LTD. in 2000.

さらに、図 3 には三菱電機製の小型産業用ロボット RV1A をベースに現在開発中の発泡スチロール型の加工ロボットを示す¹¹。RV1A の場合、アーム先端の位置と姿勢は、コマンドベクトル $[\mathbf{x}^T(k) \ \mathbf{o}^T(k)]^T$ を含む 192 バイトのパケットを UDP プロトコルを用いたイーサネット通信により送信することで制御できる。 $\mathbf{x}(k) = [x(k) \ y(k) \ z(k)]^T$ [mm] はロボット座標系における目標位置ベクトルであり、 $\mathbf{o}(k) = [\phi(k) \ \theta(k) \ \psi(k)]^T$ [rad] はロール角、ピッチ角、ヨー角で構成される目標姿勢ベクトルである。

これらのロボットシステムには、図 4 に示すような軌道追従制御法が適用されている¹⁰。 $\mathbf{v}_t(k)$, $\mathbf{v}_n(k)$, $\mathbf{v}_p(k)$ はそれぞれ、位置・姿勢制御、磨き力制御、ピックフィード制御のための操作量である。 $\mathbf{x}_d(k)$, $\mathbf{o}_d(k)$, $f_d(k)$ はそれぞれ、目標の位置ベクトル、姿勢ベクトル、磨き力である。また、 $\mathbf{x}(k)$ と $\mathbf{f}(k)$ はそれぞれ、センサで観測された位置・姿勢ベクトルと力ベクトルである。 \mathbf{S}_p はスイッチ行列であり、位置のフィードバック制御を行う方向を指定できる。なお、加工ロボットに必要とされる軌道追従制御系では、アーム先



Fig. 3 Machining robot developed based on a small industrial robot RV1A provided by Mitsubishi Electric Corp. in 2014.

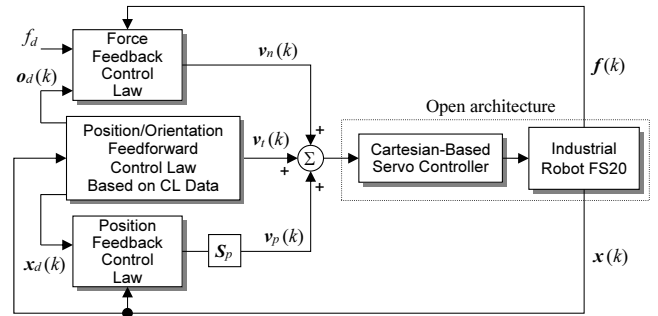


Fig. 4 Block diagram of CAD/CAM-based trajectory following and force controller used in the robot sander shown in Fig. 2.

端の位置と姿勢が同時に制御されるが、力制御は用いていない。

以上、二つのロボットメーカーが提供する産業用ロボットを用いて開発したシステムを紹介したが、一つのロボットアプリケーションを他メーカーのロボットに移植するためには、それぞれの API の仕様に合わせたソフトウェアの変更や修正が必要となり、ロボットアプリケーションの再利用性、移植性といった観点から大きな問題と言える。

3. ORiN インタフェイスを持つロボットへの軌道追従制御法の適用実験

本章では、前章で紹介した金型磨きロボット、曲面ロボットサンダー及び加工ロボットに実装されている CAD/CAM ベースの軌道追従制御法を、ORiN インタフェイスを利用できる小型多関節型ロボットに移植し、その再利用性、移植性を評価する。図 5 には今回の実験で用いた小型多関節ロボット VE026A を示す。デンソーから提供されているこのロボットは教育分野でよく利用されており、ORiN2 SDK を使ったアプリケーション開発が可能となっている^{12,13}。CAO (Controller Access Object) と呼ばれる標準プログラミングインタフェイスは、上位のロボットアプリケーション

とメーカーが提供する下位のロボットコントローラの間の共通インタフェースとして機能するものと期待される。

さて、今回の軌道追従制御法を適用するにあたり、CAOインタフェースの中で必要となる関数について調査した結果、主に二つの関数が必要となることがわかった。一つは、アーム先端の位置・姿勢情報 $[x^T(k) \ o^T(k)]^T$ をモニターするための `CaoGetPose()` であり、もう一つは、VE026Aのサーボ系に目標の位置・姿勢情報 $[x_d^T(k) \ o_d^T(k)]^T$ を与える `CaoMove()` である。つまり、Visual C++ や C# などの上位開発環境のマルチメディアタイマー割り込みの中で `CaoGetPose()` と `CaoMove()` を使うだけで位置と姿勢のフィードバック制御系を構成できる。

有効性及び妥当性を検証するために、デンソーが提供するシミュレーション環境「WINCAP III」を用いて動作実験を行った。その結果、目標軌道である CL データ内の正規化法線ベクトルから VE026A に与えることができるロール角、ピッチ角、ヨー角 $\alpha(k) = [\phi(k) \ \theta(k) \ \psi(k)]^T$ [deg] への姿勢表現の変換を確実に行えば、CAD/CAM で生成した CL データに沿ってアーム先端の位置と姿勢を比較的容易に軌道追従制御させることができた。



Fig. 5 Six-DOFs articulated robot arm VE026A incorporated with ORiN interface provided by Denso Corp.

4. おわりに

筆者らはこれまでに、オープンアーキテクチャ型産業用ロボットに適用できる「多軸制御用 CL データに基づく軌道追従制御法」を提案し、ロボットメーカーが提供するロボット言語を用いることなく、研磨や切削加工などのロボットアプリケーションの開発を可能にしてきた。最近、ロボットインタフェースの標準化のための有望なミドルウェアとして ORiN (Open Robot/Resource interface for the Network) が注目されている。本稿では、ORiN ミドルウェアが搭載された 6 自由度小型多関節型ロボット VE026A に対して、既に提案している軌道追従制御法の適用を試み、WINCAP III を用いて検証実験を行った。その結果、ロボットアプリケーションの高い移植性が確認された。今後は、ORiN インタフェースを有し、研磨ツールや切削加工ツールを搭載できる程度の可搬重量を持つ実際の産業用ロボットへの適

用とアプリケーションの評価実験を行いたい。

謝辞

本研究は平成 25 年度 JSPS 科学研究費助成事業（「基盤研究 (C)」）課題番号 25420232 の助成により実施している。

参考文献

- 1) S. Shin, “What are the standards in industrial automation sector?,” *Procs. of ICCAS-SICE 2009*, pp. 945–946, 2009.
- 2) T. Borangiu, F.D. Anton and S. Anton, “Open architecture for robot controllers,” *Procs. of 2010 IEEE 19th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD)*, pp. 181–186, 2010.
- 3) Z. Ying, W. Tianmiao, W. Hongxing, L. Miao, “Robot software architecture based on IPv6,” *Procs. of 6th IEEE Conference on Industrial Electronics and Applications*, pp. 1666–1671, 2011.
- 4) 水川 真, 尾崎安男, “産業用ロボットにおけるネットワークインタフェースの標準化活動”, 東芝レビュー, Vol. 56, No. 9, pp. 7–11, 2001.
- 5) M. Mizukawa, T. Koyama, T. Inukai, A. Noda, N. Kanamaru, Y. Noguchi, and N. Otera, “Proposal of open-network-interface for industrial robots (ORiN) and its experimental evaluation,” *Procs. of 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 689–694, 2001.
- 6) M. Mizukawa, H. Matsuka, T. Koyama, T. Inukai, A. Noda, H. Tezuka, Y. Noguchi, and N. Otera, “ORiN: open robot interface for the network— the standard and unified network interface for industrial robot applications,” *Procs. of the 41st SICE Annual Conference*, pp. 925–928, 2002.
- 7) M. Mizukawa, S. Sakakibara, and N. Otera, “Implementation and applications of open data network interface ORiN,” *Procs. of the 43rd SICE Annual Conference*, pp. 1340–1343, 2004.
- 8) 永田寅臣, 渡辺桂吾, 泉 清高, “多軸制御用 CL データに基づく位置補償器を用いた産業用ロボットの扱い制御”, 精密工学会誌, Vol. 66, No. 3, pp. 473–477, 2000.
- 9) F. Nagata and K. Watanabe, *Controller Design for Industrial Robots and Machine Tools: Applications to Manufacturing Processes*, Woodhead Publishing, UK, 2013.
- 10) F. Nagata, Y. Kusumoto, Y. Fujimoto, and K. Watanabe, “Robotic sanding system for new designed furniture with free-formed surface,” *Robotics and Computer-Integrated Manufacturing*, Vol. 23, No. 4, pp. 371–379, 2007.
- 11) F. Nagata, A. Otsuka, K. Watanabe and Maki K. Habib, “Fuzzy feed rate controller for a machining robot,” *Procs. of the 2014 IEEE International Conference on Mechatronics and Automation (IEEE ICMA 2014)*, pp. 198–203, 2014.
- 12) VE026A ORiN Option Users’ Guide, Version 1.0.0, DENSO WAVE Inc., December 14th, 2012.
- 13) ORiN2 Programming Guide, Version 1.0.12.0, DENSO WAVE Inc., September 7th, 2012.