環境タスク適応型電力制御を有するロボットシステムの構築に関する研究

小川 和哉*・水川 真*・安藤 吉伸*

Research on the Robot Power Management System Adapting to Task Constraints

Kazuya OGAWA*, Makoto MIZUKAWA* and Yoshinobu ANDO*

Efficient energy usage is crucial issue because energy consumption increases with the spread of RT (Robot Technology) application fields. Power consumption in a robot system changes according to a task and its operating environment. To adapt these conditions, we have proposed the system named the CEPC (Component for Electric Power Control) in a robot controller adopting distributed microprocessors. We defined the electric power profiles as power consumption characteristics of the tasks. The robot's power consumption is predicted and controlled by using this profile.

Key Words: robot, electric power control, power management, distributed control system

1. 研究背景

近年ロボットは製造業用ロボットから非製造業用となるオフィスや医療、そして一般家庭へとその用途を広げている。このようにロボット技術(RT: Robot Technology)は需要を拡大しつつあり、こうした背景を受け経済産業省ではRT分野の課題の整理と2025年までの方向性を示したロボット分野における技術戦略マップ¹⁾を策定した。この技術戦略マップによれば、今後ロボットを一般的に普及させるために、最初から単体で高機能の万能ロボットを開発するのではなく、単体では低機能だが複数のロボットが連携してサービスを実現できるロボットを開発していく方針を示している。

一方RTの普及によってRTによるエネルギー需要は増加するといえる。しかし、ロボット分野のエネルギーについては機能実現が優先されており、関心が薄い。このため、ロボットのための省電力システムが必要であると考えられる。また、このエネルギー問題に対するアプローチとして、エネルギー以外の技術動向を先読みし、それに合わせてシステムの開発を行なうためには、ロードマップを活用することが望ましい。経済産業省によるプロジェクト²⁾にはRTミドルウェアを適用して、RT要素を相互接続可能な部品として利用を普及促進するプロジェクトが複数ある。これらプロジェクトの方針として、ロボットの基盤技術にはモジュール化およびコンポーネント化がデザインポリシーとして含まれることが重

要であることがわかる。モジュール化およびコンポーネント化することが必ずしも省電力につながるとはいえないが、本研究では、省電力システムの構成法として、専用のロボットのための専用の省電力システムを開発するのではなく、個々の分散された機能の集合としてロボットを捉え、機能要素ごとの省電力化を行なうことで再利用性の高い省電力なロボットシステムを構築することを試みる。

以上より、エネルギー問題に対するアプローチとしては、ロボット内部の電力監視・制御・調停機能をモジュール化してロボットに組み込むことが有効となる。

2. 過去の研究と問題点

省電力に関する過去の研究としては、自動車や家電製品に代表とされる状態遷移によるシステムの省電力化^{3)~5)}や、省電力デバイスの開発および利用⁶⁾がある。また、効率的な経路計画生成のように、稼働時の消費エネルギーが状況により変化する状況の省電力化^{7),8)}がある。これら過去の研究をふまえ、ロボットに省電力技術を導入するための問題点を大きく3つ挙げることができる。

まず、状態遷移のためのエネルギーの見積もりとパターン化の問題が挙げられる。ロボットは自動車や家電製品のように定常パターンの組み合わせとなる状態が少ないので状態遷移による省電力化を行なうことが困難である。この問題を解決するため、前述のロードマップに従いロボットには人と同じような柔軟なサービスの実行は期待せず、「あれをとってきて」などの日常的に要求の多い代表的なサービスのみに着目することとする。そして、ロボットが実行可能なタスクと消費電力の関係を記述した「電力に関するプロファイル」を作成することで汎用性の向上が期待できる。

* 芝浦工業大学 東京都江東区豊洲 3-7-5

* Shibaura Institute of Technology, 3-7-5 Toyosu, Koto-ku, Tokyo

(Received March 10, 2008)

(Revised April 28, 2009)

2番目に、安全性の問題が挙げられる。ロボットはアームをはじめとする物理的運動をするアクチュエータを多くもつ。このため、省電力状態のために特定の機能を停止させることで、ロボット周囲が危険な状態になることがある。また、ロボットがサービスを実行中に、エネルギー不足により停止することで危険な状態になることがある。これを回避するために、サービス実行前に消費エネルギー量を予測できるシステムを開発し、実行の可否を判断可能にする。

最後に、ロボットの行動指針に関する問題が挙げられる。従来のロボット開発においては、機能実現を優先してきたためエネルギーを効果的に活用することは考えられていないものが多い。このため、現在のロボットの状態と消費エネルギー量を基準として、ユーザの許容する範囲内で省電力な行動にするタスク調停システムを構築する。

3. 目的

本研究の目的は、コントローラを機能ごとにマイクロプロセッサで構成し、これらを分散して全体を協調制御する分散制御ロボットにおいて、その分散機能の1つとして電力を基準としたタスク実行を調停するシステムを開発することである。分散制御系をもつロボットを用いることで、モジュール化およびコンポーネント化により汎用的なプラットフォームで開発することができる。また、ロボットは搭載するエネルギーが限定される二次電池を用いたロボットを選択する。

本研究において、ロボットコントローラ内部の電力監視・制御・調停モジュールをCEPC(The Component for Electric Power Control)と呼び、本研究では主にCEPCの機能および運用のためのデータ構造設計に着目していく。

目的達成時の効果として、電源容量の最適化による電源システムの小型化や、確実な電力管理によるロボットの命令遂行能力の向上が挙げられる。これにより長時間にわたりロボットが提供するサービス品質(QoS)の維持が可能となる。

4. アプローチ

4.1 サービスとタスクの関係

本研究の目的を達成するためのアプローチとして、ロボットサービスとタスクの関係について着目し、ロボットの利用方法から関係を明確にする。

まずロボットの形態による分類として、人が何らかの形で操作を行なう操縦型ロボットと、自ら状況判断して行動を決定する自律型ロボットがある。しかし、操縦型ロボットも、ロボットのひとつ一つの動作を逐次行なう方法がすべてではなく、ユーザが使いやすいように操作を簡略化すればするほど、ロボットにとって自律機能が増えることになり、その区分けは明確にできない部分が存在する。ここで、ロボット内部で命令遂行のためのコマンドに着目すると、最終的にアセンブリ的なコマンドがロボットに送信されることになり、操縦型も自律型もコマンドは同一であるといえる。ロボットに行動を与えるコマンドを「タスク」、ユーザがロボットに要

求する仕事内容を「サービス」とすると、サービスとは複数のタスクの組み合わせにより構成され、それが逐次実行されると考えることができる。本研究では、これをTask Sequence(以下TS)と呼ぶ。本研究ではタスクの有限な組み合わせで、限定されたサービスを実行するものとする。このときの「タスク」は、ロボットの独自機能に依存した情報はもたないものとする。この指令内容の例としては、「脚を前に動かす」では脚をもたないロボットには用いることができなくなるので「前進する」という内容のものである。

ロボットの機能に依存した指令はロボット内部で生成する。前述の「前進する」というタスクの例ならば、脚機構による前進するロボットは単純化して考えると「右足を前に動かす」「左足を前に動かす」というロボットの機能に依存した命令がロボット内部で生成されることになる。ここで、ロボット内部で生成されるコマンド群をOperation Script(以下OS)と呼ぶ。人が要求するサービスは、ロボットにTSという形で送られ、それを受信したロボットは内部でOSを生成し実行するということになる。

4.2 電力管理コンポーネント CEPC

前述のサービスとタスクの関係より、本研究では電力管理コンポーネントCEPC(The Component for Electric Power Control)を提案する。CEPCはTS実行のためにロボット内部で生成されるOSを、電力を基準に組み替えるものである。Fig.1にCEPCの概要を示す。

CEPCによる省電力化は、以下の3項目の手法で行なう。

(1) 省電力機能

これはOSの組み替えによる省電力動作切り替え機能である。具体的には、目的を達成するために生成されるタスクを複数ロボットに与えておき、その中でユーザが許した範囲における省電力となる代替となるタスクを選択し実行する。また、ロボット内の機能ごとの状態遷移の管理をすることで電力の使用状態を把握する。

(2) タスク保証機能

ロボットに与えたタスクが、バッテリーの残存容量と消費予定の電力を比較することで確実に実行できるかを判断する機能である。消費電力を予測するために、ロボットの内部状態に応じた消費電力特性をロボットに事前にもたせておく。この消費電力特性を本研究では電力プロファイルと呼ぶ。電力

Fig. 1 Outline of CEPC

プロファイルの詳細については後述する。

(3) バッテリ高効率利用

二次電池の多くは、消費電流に反比例して残存容量特性が悪化する。また、ロボットシステムでは、移動機構やマニピュレータなど大きな消費電流を発生させるデバイスが複数搭載されることが多い。したがって、ロボットへの二次電池利用は、使用条件として好ましくないことがわかる。これを考慮に入れ、CEPC はサービス実行時に負荷が集中しないようにタスクの調停を行なう。例として、大電流を消費するサブシステムを複数使用する場合、サービス達成のために必ずしも同時に稼働させなくてよいものであれば、稼働タイミングをシフトさせることで過大な電力が瞬間的に発生しないようにする。

4.3 CEPC が必要とする機能と情報

(1) ロボットに搭載されている電源容量の把握

ロボットが搭載しているエネルギーを、安全に使用できる範囲で最後まで効果的に利用するためには、ロボット自身が利用できる電源容量を把握する必要がある。これはリチウムイオン系の二次電池などで一般的に実現されている機能である。また、鉛蓄電池についても劣化を考慮に入れた正確な値を導出するための研究⁹⁾からも、この機能は既存技術により解決できていると考える。ここで取得する容量の精度については、高ければ高いほどよいといえる。本システムにおいては、この機能はシステムの機能としては必須であるが、主題ではないので、既存技術で対応できる範囲で検討することとした。

(2) ロボットが現在消費している電力の把握

ロボットは、そのロボット自身が現在消費している電力がどの程度か把握できないシステムの場合、残りの電源容量の推定や、電源に起因する瞬間電圧低下などの問題について対応できないことになる。省電力の機能面および電源管理の安全面の観点から、ロボット内部にエネルギー監視用のシステムは必須であり、構築する必要がある。

(3) ロボットが実行予定のタスク達成に必要な電力の予測

ロボットが確実にタスクを実行できるようにするには、タスクを実行した際にどの程度のエネルギーが必要かをロボットが把握している必要がある。電力の予測には、デバイスおよびタスクごとに消費電力量が記述された電力プロファイルを利用する。

4.4 電力プロファイルの提案

実行予定のタスク実行に必要な電力量を記述したものが電力プロファイルである。電力プロファイルには、分散制御ロボットのもつ各機能要素の全タスクに対応した消費電力が記述される。電力プロファイルは、TS に対応するサービスレイヤと OS に対応するデバイスレイヤの 2 つに分類される。サービスレイヤには、ロボットがもつサービス単位の最大電力量および平均電力量、そしてサービスの実行手順が記載される。デバイスレイヤには、搭載する機能要素名と状態別の最大電力量、平均電力量が記述される。時間変化により消費

Fig. 2 Example of electric-power profile

電力が変動する場合は、それを記述しておく。これらはテキストベースで他のシステムからも参照を可能とするために XML により記述する。電力プロファイルの XML における tree 構造例を Fig. 2 に示す。

電力プロファイルは、記述された情報によりサービス実行時の消費電力を導出できる構造である必要がある。そのため、ロボットに搭載されたデバイスの単動作を最小単位とし、サービスレベルまで組み立てることで、サービス単位の消費電力量が導出できるようにする。まず、ロボットが動作する最小単位コマンド (Fig. 2-4, 9 行目) の単位時間あたりの消費電力を記述する。消費電力は、使用するロボットに応じて実験により取得した最大値 (Fig. 2-7 行目) と平均値 (Fig. 2-8 行目) を記述する。これは、最大値によりタスクを確実に実行できるラインを定め、平均値により消費エネルギーの目安をユーザに提示するためである。さらに、コマンドの種類に応じて時間経過により変動する電力なのか、固定時間で一定の消費電力量を消費するのか区別できるようにする (Fig. 2-6, 10 行目、この場合どちらも時間で変動)。つぎに、デバイスごとのコマンドをどのように利用するとサービスを実現できるかを記述する (Fig. 2-17-26 行目)。これは、時間経過により変化するコマンドを、どのような手順で、どのくらい実行するのかを記述する。ユーザが指定位置までロボットを移動させたい場合などの距離情報は、プロファイルとして記述するのではなく、ロボットが必要に応じて導出する必要がある。サービスを実現する手順と、デバイスごとの最小動作コマンドの消費電力を組み合わせることで、サービス実行前に消費電力の予測が可能となり、サービスの確実性が向上する。

4.5 アプローチのまとめ

CEPC と電力プロファイルを提案し、TS の配置設計を述

べた。これにより QoS 保証につなげるためのフレームワークを提案した。

5. 開発環境

5.1 開発手順

CEPC を適用したロボットサービス運用システムの開発手順は、以下の 5 ステップからなる。

- 1. 各デバイスの消費電力特性の取得
- 2. 消費電力特性からタスク調停のパターンを作成
- 3. タスク組み替えによる省電力効果の検証
- 4. 実装実験により省電力効果の検証
- 5. 調停パターンの追加および最適化

本研究では主に項目 1~3 に着目し、タスクの配置設計についてまとめ、実験をもとにプロファイルを参照した TS 組み替えによる省電力効果を検証し、提案する CEPC の有効性を検証する。

まず、ロボットに省電力機能をもたせるにあたり、既存のロボットシステムに CEPC システムを組み込み、効果を確認する。このため、適用するシステムは汎用性が高く、今後のロードマップに反しないシステムが望ましい。本研究では、Physical Agent System を用いることとする。

5.2 Physical Agent System

われわれは遠隔地からの物理的アクションを伴うロボットサービスを実現する新たなメディアとして Physical Agent System(PAS)を提唱している。PAS の概要を Fig. 3 に示す。

PAS は ORiN¹⁰⁾ や RT ミドルウェア¹¹⁾ といったミドルウェアを利用した拡張性および汎用性が高いロボットシステムであり、これにより日常生活支援、災害現場協調支援、遠隔地間のコミュニケーションメディアとしての利用など、さまざまな分野へ利用可能なシステムである。システムはサーバ、ユーザ、ロボットの 3 種類のレイヤにより構成されている。また、PAS ではロボットの構成に依存しない汎用命令コマンドの RAC(Robot Action Command set)を利用可能¹²⁾ である。RAC は MTM 法により分割された動素を、ロボット向けに標準的なコマンドになるよう選定した命令コマ

ンドである。このため、RAC の集合によりロボットサービスを記述すれば、前述の TS のように構成できることになる。RAC のコマンドフォーマットを Fig. 4 に、RAC のトップレベルコマンド一覧を Table 1 に示す。

RAC は愛知万博において、ORiN により接続された仕様の異なるロボット間の連携を、統一化された命令コマンドによって容易に実現した実績をもつ。本研究では RAC の集合によりサービスが記述できると考え、システムの実装を行なう。

5.3 Physical Agent Robot : PAR04R

Physical Agent Robot : PAR04R は、PAS の実験用機体として開発されたロボットである。PAR04R は搭載機能ごとに CPU を配置した分散制御系をもつロボットシステムである。PAR04R の外観を Photo 1 に、搭載機能を Table 2 に示す。

電源は二次電池の鉛蓄電池を採用しており、搭載機能としては、クローラによる移動機構、アーム機構によるマニピュレーション、効果器としてレーザポインタとカメラを組み合わせたジンバル機構、無線 LAN を介して環境情報を送受信する PC、そして電力監視システムが搭載されている。これら分散配置された各要素をサブシステムと呼び、サブシステム間は CAN (Controller Area Network) を用いて接続されている。PAR04R は愛知万博において 11 日間の搬送

Request Commands

<Top-Level-Command>

: [<Region name>] : [<Region number>]

: [<Sub-Command>] : [<Parameter>]

Response Commands

<Processing-Result>

[, <Data-Type>, <Data-Sequence>]

Fig. 4 The format of robot action command set

Table 1 Top-level command for RAC

REACH	STAND
MOVE	TURN
GRASP	WALK
RELEASE	TWIST
LEG	EYE

Fig. 3 Outline of physical agent system

Photo 1 Appearance of PAR04R

Table 2 System configuration of PAR04R

Subsystem Name	Component
Crawler	DC Motor with Encoder (Maxson Japan Co. : 261042 x 2)
	Control CPU (Renesas Technology Co.: H8S/2636Q)
Arm	Five axes Manipulator (Sytronics Inc. : KATANA)
Camera (Gimbals mechanism)	RC Servo Motor (Futaba Co. Japan : S3101 x 2, S9001)
	Potentiometer (NIDEC COPAL ELECTRONICS Co. Japan : JC10 x
	CCD Camera (RF CO., LTD. Japan: TINY)
	CLASS-2 Laser pointer (IWASAKI METAL CO., LTD.: TLP-78)
	Control CPU (Renesas Technology Co.: H8S/2636Q)
Power monitoring	2.5(A) Max Current Sensor (NANALEM Co. Japan: LTS6-NP x 4)
Control	Control CPU (Intel Co.: Pentium M 725 1.6G Hz)
	Motherboard (PFU : Intel 855GME chipset)
	Memory (PC2700 1GB)
	HDD (HGST : 2.5inch 40GB)
Other	Size W440mm x D820mm x H575mm
	Weight 46kg
	Battery 12V-24Ah (JAPAN STORAGE BATTERY CO., LTD.: PE12V24A)

サービスを失敗なく実施した実績をもつ信頼性の高いロボットであり、ロボットの動作指令は RAC により行なわれる。本研究では、PAR04R を用いて CEPC 搭載に必要な情報の取得実験を行なう。

5.4 提案したフレームワークの PAS への実装

CEPC を用いたサービス運用モデルを PAS へ適用させたものを Fig. 5 に示す。ユーザは、環境サーバから空間内に配置されたロボットやセンサの情報を受け取り、実行できるサービスを選択し送信する。サービスを達成可能なロボットは、サービスを TS(RAC の集合)として受け取る。受け取った TS は、ロボット内で OS に変換されロボット内で実行される。このとき、CEPC は電力プロファイルを参照して省電力となるように OS の組み替えを行なう。この組み替えはユーザに通知され、ユーザの許可によって省電力な行動が実現する。

5.5 電力測定環境

ロボット内には電力監視用サブシステムが搭載されている。電力監視サブシステムでは、サブシステムごとに電圧値と電流値を測定しており、簡易的なログ機能と異常電圧・異常電流検知機能、サブシステム単位での電源遮断機能、ユーザへの電源情報報告機能を搭載している。

ロボットの消費電力の測定実験には、電力監視サブシステムに OMRON 製のデータロガー ZR-MDR 10 を用いて長時間の記録を可能とした。また、ロボットが受信する RAC の

Fig. 5 CEPC service operation model for PAS

Fig. 6 Electric-power log measurement system

ログをロボット搭載の PC を用いて行なうことにした。データロガーと RAC のログを用いることで、RAC コマンド 1 個単位での消費電力を得ることが可能になっている。実験用測定システムの構成を Fig. 6 に示す。

5.6 開発環境のまとめ

開発環境の実験系として、他のロボットシステムにおいても利用可能となる PAS および PAR04R を用いることを示した。これにより、提案したフレームワークを PAS へ実装させることでサービス運用モデルの明確化を行なった。また、PAR04R に電力計測系を実装し、省電力を行なうシステムに必要なハードウェアのフレームワーク例を提示した。

6. 実験による検証

6.1 実験手順

PAR04R で省エネルギーに対する実験を行なうにあたり、消費電力が大きいアームおよびクローラサブシステムに着目して消費エネルギーを測定する。これは消費電力が少ないサブシステムでは消費電力の変化幅が大きくなりすぎないため、省エネルギーの効果を得るために長時間の実験が必要となるためである。また、サービス内容は搬送サービスを選定した。これは PAR04R が行なう標準的なサービスであり、技術戦略

マップからも日常のよくあるサービス内容として挙げられているためである。

6.2 電力プロファイル用の消費電力特性の取得

サブシステムのもつ状態ごとの消費電力量をプロファイル化するために、電力プロファイル用の消費電力特性取得のための実験を行なう。ロボットがサービス実行時に消費する消費電力は、ロボットの動作環境に応じてばらつきが考えられるので、複数回のサービスを実行した際の平均消費電力および最大消費電力を求める。ここでは搬送サービス実行時のプロファイル作成のために、対象物を 2 m 離れた搬送先であるゴミ箱に捨てる動作を繰り返す。実験環境を Fig. 7 に示す。

このサービス実行時の実行したコマンドごとの消費電力量は、ロボットの操作コマンド (RAC) のログおよびデータロガーによる消費電力量のログから得ることができる。ここで、本来 RAC に記されるのは TS としてロボットに与えられたサービスを達成するための汎用的なコマンド列だが、今回の実験では PAR04R を手動操作モードで動作させることで、ロボットに送信される命令は逐次操作による OS と同等の位置付けで利用することができる。実験中のロボットの状態は、対象の把持、目標地点へのロボットの移動、対象の解放、初期地点へのロボットの移動という 4 つに分類することができる。この状態の分類を Stage として表記し、Stage 1~4 までのロボットの状態を達成させるためのロボットの命令手順を RAC のログから記述をする。さらに、ロボット内のサブシステムが実行する RAC 命令ごとの消費電力を得る。これによりプロファイルからサービス実行に必要な消費電力量を得ることができる。

実験により得られたログデータは、移動平均処理の後に

Fig. 7 Measurement environment for electric-power profiles

Fig. 8 Electric-power characteristics (Stage)

RAC の種類ごとに分割した消費電力量にする。まず、Stage ごとの各サブシステムの平均消費電力とサービス達成に必要な時間、Stage という状態に必要な消費電力量とすることができる。

サブシステム内の状態として、クローラでは 7 つの状態をもっている。実験より、それぞれの状態が実行された際の平均および最大消費電力量を得ることができた。Fig. 8 に取得した Stage 別の消費電力特性を、Table 3 に Stage 別の最大消費電力量を、Table 4 にクローラの状態別平均消費電力量を示す。

Fig. 8 より、Stage 1 と Stage 3 では、主に電力が変動しているのはアームサブシステムだけである。Stage 2 と Stage 4 では、移動するためにクローラサブシステムの電力変動が大きいのがわかる。サービス実行に最低限必要な消費電力量は、最大消費電力がサービス実行時間中に継続して消費した場合の消費電力量以下であると考えられる。よって最大消費電力と最大実行時間を電力プロファイルに用いる。また、平均消費電力を計算することで、ロボットがサービスを実行する消費電力量の目安とする。

Stage ごとのデータはサービスレイヤのプロファイルとして、RAC ごとのプロファイルはデバイスレイヤのプロファイルとして消費電力予測のために用いることができる。データより記述された電力プロファイルを Fig. 9 に示す。

Fig. 9 の 1 行目から 18 行目はデバイスレイヤの電力プロファイルを示している。ここでは、クローラサブシステム (Fig. 9-2 行目) は 7 つの状態 (Fig. 9-3 行目から 16 行目) があり、“Wait” 状態 (Fig. 9-3, 4 行目) では、時間に比例して

Table 3 Power consumption for electric-power profile (Service)

	Time[sec]	PC[W]	Crawler[W]	Arm[W]	Total[W]	Total[Wh]
Stage1	64.6	22.3	3.7	20.4	46.2	0.83
Stage2	26.9	20.5	22.1	17.7	59.4	0.44
Stage3	15.0	22.2	1.0	16.2	39.6	0.17
Stage4	21.1	20.7	24.7	18.0	62.2	0.36
Service(TS)	127.6					1.80

Table 4 Power consumption for electric-power profile (Crawler)

Command	State	Power Consumption / Service[Ws] (Average)	Execution Time / Service[sec] (Average)	RAC-Power Consumption [W](Average)
WAIT	Wait	51.4	65.79	0.78
WALK10	LowSpeed MoveF	100.6	7.47	13.47
WALK-10	LowSpeed MoveB	25.0	2.36	10.61
WALK120	HighSpeed MoveF	159.7	10.93	14.61
WALK-120	HighSpeed MoveB	4.4	0.27	16.56
TURN1	Turn1	255.7	6.99	36.59
TURN2	Turn2	310.4	8.48	36.62
Total[Ws]		907.3	102.3	

消費電力が増え(change="time"), 平均消費電力は 0.78 W である. また, Fig. 9 の 20 行目以降はサービスレイヤの電力プロファイルを示している. Delivery サービスは 4 つのステージで構成され(Fig. 9-23 行目から 27 行目), クローラサブシステムが Stage 1 で実行するコマンドごとの最大時間(MaxTime)と平均時間(AverageTime)が示されている. 実行コマンドに依存しない PC とアームは最大消費電力のみ(Fig. 9-50, 51 行目)記述される.

この実験によりタスク, RAC ごとの電力プロファイルの作成を行なうことができ, 運用システムに適用することが可能となった.

6.3 Operation Script の組み替えによる効果の検証

構築された電力プロファイルを利用して, OS の組み替えによる省電力効果を検証する. OS の組み替えを行なうためには, ロボットが要求されたサービスを達成するための手順をロボット内で複数定義しておく必要がある. ここでは, 組み替えによる効果の検証のために, 搬送サービスという目的

Fig. 9 Normal task and alternative task

```

1:<ElectricPowerProfile Layer="Device" Robot_ID="PAR04R">
2:  <Subsystem name="Crawler">
3:    <State change="time" MaxPower="0.96"
4:      AveragePower="0.78">Wait</State>
5:    <State change="time" MaxPower="37.80"
6:      AveragePower="13.47">LowSpeedMoveF</State>
7:    <State change="time" MaxPower="17.10"
8:      AveragePower="10.61">LowSpeedMoveB</State>
9:    <State change="time" MaxPower="21.79"
10:     AveragePower="14.61">Turn1</State>
11:    <State change="time" MaxPower="90.34"
12:     AveragePower="16.56">Turn2</State>
13:    <State change="time" MaxPower="69.58"
14:     AveragePower="36.59">HighSpeedMoveF</State>
15:    <State change="time" MaxPower="65.42"
16:     AveragePower="36.62">HighSpeedMoveB</State>
17:  </Subsystem>
18:</ElectricPowerProfile>
19:
20:<ElectricPowerProfile Layer="Service" Robot_ID="PAR04R">
21:  <Service name="Delivery"
22:    MaxPowerConsumption="">
23:    <Procedure stage="1">Search Target</Procedure>
24:    <Procedure stage="1">Grasp</Procedure>
25:    <Procedure stage="2">Delivery</Procedure>
26:    <Procedure stage="3">OpenHand</Procedure>
27:    <Procedure stage="4">InitPos</Procedure>
28:  </Service>
29:  <Stage name="Stage1" MaxTime="64.6">
30:    <Usesystem name="Crawler" MaxPower="3.7">
31:      <Commands
32:        MaxTime="62.00" AverageTime="49.94">Wait</Command>
33:      <Commands
34:        MaxTime="3.30" AverageTime="1.73">
35:        LowSpeedMoveF</Command>
36:      <Commands
37:        MaxTime="1.70" AverageTime="0.78">
38:        LowSpeedMoveB</Command>
39:      <Commands
40:        MaxTime="1.30" AverageTime="0.14">Turn1</Command>
41:      <Commands
42:        MaxTime="0.90" AverageTime="0.10">Turn2</Command>
43:      <Commands
44:        MaxTime="2.80" AverageTime="1.23">
45:        HighSpeedMoveF</Command>
46:      <Commands
47:        MaxTime="3.20"
48:        AverageTime="1.43">HighSpeedMoveB</Command>
49:    </Usesystem>
50:    <Usesystem name="PC" MaxPower="22.3">PC</Usesystem>
51:    <Usesystem name="Arm" MaxPower="20.4">ARM</Usesystem>
52:  </Stage>
.....

```

に対して状態の異なる 2 種類の行動を挙げて効果の検証を行なうものとした.

搬送サービスの流れとして, PAR04R はアーム機構により搬送物体を把持し, 把持姿勢を保持したまま搬送先へ移動する. ここでは把持姿勢を保つアームの姿勢による消費エネルギーの違いに着目した. アームは簡易的なモデルとして捕らえ, 3 軸のみを使用した 1 軸モデルとして動作させる. このとき, アームを水平に保ったまま搬送する状態を通常タスクとする. 一方, アームを垂直に伸ばす特異点を用いた搬送する状態を, OS の組み替え対象とする代替タスクとして用いることにする. 特異点を用いた搬送により, アーム本体の自重および搬送物体の重量を垂直に受けることができ, アーム内のモータに必要なエネルギーが減少すると考えたためである. 通常タスクと代替タスクの違いを Fig. 10 に示す.

この実験の搬送距離は 5 m とし, 350 g の物体を搬送した. 搬送のためのロボットの状態遷移は, 通常タスク・代替タスクともに「搬送開始のための姿勢変更(姿勢変更 1)」から「搬送」, そして「搬送終了のための姿勢変更(姿勢変更 2)」となる. このため, この 3 つの状態についてのアームサブシステムの消費エネルギーを測定した.

まず, 姿勢変更 1 および 2 に必要とする消費エネルギーおよび実行時間はそれぞれ固定値である. 姿勢変更に必要な消費エネルギーの合計を切片として, 移動距離に対する消費電力量の特性を示したものが Fig. 11 である. まず, 姿勢変更に必要な消費エネルギーは, 通常タスクよりも代替タスクのほうが大きい, 搬送距離のある点を境に代替タスクのほうが通常

Fig. 10 Distance-power consumption characteristics

Fig. 11 Electric-power profile

タスクよりも消費電力が少なくなることがわかる。また、通常タスクと代替タスクの搬送終了時における時間差は、姿勢変更に要する時間の差がそのまま反映されている。今回の実験において、5 mの搬送サービス終了時に発生した電力差は、代替タスクでの搬送で0.8 m分走行距離に置き換えることができるものとなった。またこの効果は、搬送距離に比例して上がることがわかる。

6.4 考察

実験により OS の組み替えによる省電力効果について確認することができた。代替タスクによる省電力効果は、アームの特異点姿勢による定常的消費エネルギーの減少効果と考えることができる。しかし、今回の実験のように搬送距離に応じて通常タスクと代替タスクの電力量が入れ替わる場合や、ユーザが1秒でも早くタスクを達成したい場合など、状況に応じた代替タスクの選定が必要になる。これらは CEPC のもつタスク調停機能として実装する必要があることがわかる。

また、この実験は提案する CPEC の有効性の検証の一例にすぎない。今回取得した電力プロファイルデータは、搬送サービスに限定して有効なものになったとも考えられる。また、搭載されたサブシステムのうち、消費電力が少ないサブシステムの省電力タスクの配置設計についても検討する必要がある。

6.5 実験のまとめ

実験により、タスクの組み替えによる省電力効果の検証および電力プロファイル作成のためのデータを得ることができた。これにより提案するフレームワークについての妥当性が確認できたといえる。

7. ま と め

本研究では、ロボットシステムを省電力にするために電力を基準とした調停システムを提案し、実装検証を行なった。このシステムの実現のため、ロボットのサービスとタスクの関係に着目し、現在のロボットのエネルギー環境に適したタスクの組み合わせを行なうシステムとして CEPC を提案した。

CEPC 運用のためのデータ構造設計としてデバイスごとの電力プロファイルを作成し、現在のロボットのエネルギー環境に適したタスクの組み合わせを行なうシステムフレームワークを構築した。また、これを実験により有効性を確認した。

参 考 文 献

- 1) 独立行政法人新エネルギー・産業技術総合開発機構：技術戦略マップ, <http://www.nedo.go.jp/roadmap/index.html>
- 2) 経済産業省：ロボット政策の全体像, 2007年5月22日付け公開資料, <http://www.meti.go.jp/policy/robotto/0705gaiyo.pdf>
- 3) 石塚仁司：稼働時電気損失削減最適制御技術開発の概要, 計測自動制御学会システムインテグレーション部門学術講演会 2002 講演論文集 II, 309/310 (2002)

- 4) 森田, 島村：ハイブリッド電気自動車の評価法に関する検討, JARI Research Journal, 25-11, 437/440 (2003)
- 5) 大林, 谷：Concept of Vehicle Electric Power Flow Management, デンソーテクニカルレビュー, 9-2, 142/148 (2004)
- 6) 石原, 富山：低消費電力化ソフトウェア技術, 組込みソフトウェアシンポジウム (ESS) 予稿集, 188/190 (2005)
- 7) 佐々木, 大道, 杉浦：知的運動制御によるロボットの高速度, 日本機械学会ロボティクス・メカトロニクス講演会 2003, 1P1-2F-F7 (2003)
- 8) 泉, 佐藤, 沖田：重力下で PTP 制御されるマニピュレータの消費エネルギーを最小にする最適動作時間, 日本ロボット学会誌, 15-7, 1082/1088 (1997)
- 9) 計盛, 水川, 小川：物理エージェントロボット搭載バッテリーのマネジメントに関する研究, 第8回計測自動制御学会システムインテグレーション部門講演会, 1L4-7 (2007)
- 10) 北垣, 末廣, 神徳, 平井, 谷江：RT ミドルウェア技術基盤の研究開発についてーロボット機能発現のために必要な要素技術開発ー, ロボティクスシンポジウム 2003 予稿集, 487/492 (2003)
- 11) 水川, 松家, 小山：産業用ロボットにおけるオープンネットワークインタフェース “ORiN” の概要と展開, 計測自動制御学会システムインテグレーション部門学術講演会 2002 講演論文集 II, 285/286 (2002)
- 12) 月川, 弘中, 水川, 安藤, 犬飼：ORiN 拡張汎用ロボット操作コマンド RAC の開発：PAS としてのペットロボット AIBO および PAS 自律移動サブシステムの遠隔制御への適用, 日本機械学会ロボティクス・メカトロニクス講演会 2004, 2P1-H-33 (2004)

[著 者 紹 介]

小 川 和 哉

2003年芝浦工業大学大学院工学研究科修士課程修了, 同年より2005年まで横浜創学館高等学校科学技術科教諭。2008年芝浦工業大学工学研究科博士(後期)課程修了。ロボット教材によるものづくり教育, ロボットシステムのためのエネルギー利用に関する研究に従事。

水 川 真 (正会員・フェロー)

1975年早稲田大学大学院修士課程修了, 同年電信電話公社(NTT)入社。2000年より芝浦工業大学工学部電気工学科教授。物理エージェントを用いた人間支援システム, ロボット作業システム, RT ミドルウェアの研究, 標準化に従事。工学博士。日本機械学会, バイオメカニクス学会, IEEE RAS 会員。

安 藤 吉 伸 (正会員)

1999年筑波大学大学院電子情報工学専攻博士課程修了, 博士(工学)。現在, 芝浦工業大学電気工学科准教授。自律移動ロボットのセンサシステム, ナビゲーションの研究に従事。日本機械学会, 日本ロボット学会, 電気学会, IEEE 会員。