岐阜大学機関リポジトリ
Gifu University Institutional Repository

Title	ロボット数式処理システム ROSAM II の開発(本文(Fulltext))
Author(s)	川崎, 晴久; 清水, 年美
Citation	[日本ロボット学会誌] vol.[17] no.[3] p.[408]-[415]
Issue Date	1999-04
Rights	The Robotics Society of Japan (日本ロボット学会)
Version	出版社版 (publisher version) postprint
URL	http://hdl.handle.net/20.500.12099/24262

※この資料の著作権は、各資料の著者・学協会・出版社等に帰属します。

学術・技術論文

ロボット数式処理システムROSAM IIの開発

川﨑晴久* 清水年美*

Development of Robot Symbolic Analysis System: ROSAM II

Haruhisa Kawasaki* *and* Toshimi Shimizu*

This paper presents a robot symbolic analysis system by Maple called ROSAM II for the symbolic modeling of robots. The ROSAM II permit to generate the symbolic models of forward kinematics, inverse kinematics, forward dynamics, inverse dynamics, base parameters and so on for serial robots, tree-structured robots and closed-loop robots. It has been developed under the computer algebra software Maple V and its GUI for settling robot parameters has been implemented using the C++ language under Windows95/NT. In this paper we represent a design concept of the ROSAM II and an overview of algorithms generating symbolic models. An example for the robot symbolic analysis is also shown.

Key Words: Robot, Computer Algebra, Symbolic Analysis, Base Parameters, Maple

1. はじめに

ロボットの数式モデルはロボットの機構設計や効率的な計算アルゴリズムを開発する上で重要である．一般にロボットの数式モデルは関節変数の三角関数を含む複雑な式であり，高い自由度のロボットの場合その導出を手計算で行うことは非常に困難となる．これまでに斉藤らによるRACER [1]，Netheryらによる Robotica [2] やKhalilらによるSYMORO+ [3] など，ロボットの数式モデルの自動生成に関する多くの研究・開発が行われてきた．しかし，これらのシステムでは一般的な機構を持つロボットのベースパラメータの解析を行うことはできない．ベースパラメータ [4] [5] はロボットの動力学モデルを記述するのに必要最小な動力学パラメータの組であり，モデルベースド適応制御やパラメータ同定に非常に重要なものである．これまでに著者らは数式処理ソフトウェアMaple V [6] を用いて一般的な n 自由度シリアルリンクロボットを対象として，順運動学，順動力学，逆動力学の各モデルの生成，ベースパラメータ解析および，制御シミュレーションが可能なROSAM (Robot Symbolic Analysis System by Maple) [7] を開発してきた．

本論文は，シリアルリンクロボットだけでなく木構造ロボット，閉ループロボットに対しても上述の解析ができるように拡張したロボット数式処理ライブラリROSAM IIの開発における基本的な考えと，数式モデルの導出の概要を示す．本システムは各種のロボット機構に対するベースパラメータの解析機能を持ち，逆運動学モデルの生成も可能であり，また，Windows95/NT環境ではロボットのパラメータを容易に設定するためのGUIがあることを特徴としている．ただし，現在のところ，閉ループロボットの解析では機構中に含むことのできる閉ループの数は一つに限定され，逆運動学では冗長駆動系でないシリアルリンクロボットに限定される．

原稿受付　1998年3月26日
*岐阜大学工学部
*Faculty of Engineering, Gifu University

2. ROSAM IIの基本方針

ROSAM IIはロボット工学の教育とロボットシステム開発に広く利用されることを目的としている．そのため，初心者にも使いやすく，かつ解析できるロボットの拡張性を考慮した設計とした．ROSAM IIは約50個の関数から構成される．**Table 1** にROSAM IIの主な関数を示す．表中の記号 m はブランチの数を示し，n_k はブランチ k の関節数を示す．これらの関数群により単に結果を得るだけでなく詳細な解析ができ，教育ではロボット工学の基本的な考えを理解するのに役立ち，ロボット研究・開発では関数の組合せにより新規のアルゴリズムやユーザーの必要とする関数を既存の関数の組み合わせにより容易に作成，検証できる．

ロボットのパラメータは **Table 2** に示すRobotData型と呼ぶデータ構造を用いて定義する．RobotData型はMapleのデータ構造であるテーブルを用いて作成されており，ロボットの機構の種類`RobotType`，ブランチの数`BranchNum`，各ブランチの情報を格納したテーブル`BranchInfo.k`，閉ループロボットの場合は一般化座標`General`，仮想切断点のあるブランチとリンク番号`ConCoor`および，そのリンク座標から仮想切断点の座標までのDHパラメータ`ConDH`の要素から構成される．各

Table 1 Main functions of ROSAM II

Function	Feature
Make_q	Make a joint variable vector $\mathbf{q}$.
Make_T	Evaluate homogenous matrices ${}^i\mathbf{T}^k_{i+1}$.
Make_R	Evaluate rotational matrices ${}^i\mathbf{R}^k_{i+1}$.
Make_p	Evaluate position vectors ${}^i\mathbf{p}^k_{i+1}$.
Make_T0	Evaluate homogenous matrices ${}^0\mathbf{T}^k_{i+1}$.
Make_R0	Evaluate rotational matrices ${}^0\mathbf{R}^k_{i+1}$.
Make_p0	Evaluate position vectors ${}^0\mathbf{p}^k_{i+1}$.
Make_B	Evaluate sub-homogenous matrices $\mathbf{B}^k$.
Make_TBase	Evaluate homogenous matrices ${}^B\mathbf{T}^k_{i+1}$.
Make_RBase	Evaluate rotational matrices ${}^B\mathbf{R}^k_{i+1}$.
Make_pBase	Evaluate position vectors ${}^B\mathbf{p}^k_{i+1}$.
EulerAn	Evaluate Euler angles from rotational matries ${}^B\mathbf{R}^k_{i+1}$.
EulerOr	Evaluate rotational matries ${}^B\mathbf{R}^k_{i+1}$ from Euler angles.
Make_Ja	Evaluate jacobian $\mathbf{J}$ (by Euler angles.)
Make_Jw	Evaluate jacobian $\mathbf{J}_\omega$ (by instantaneous axis of rotation.)
Make_r	Evaluate posiotion and orientation vector of hand $\mathbf{r}$.
Make_Dr	Evaluate velocity vector of hand $\partial\mathbf{r}/\partial t$.
Make_DDr	Evaluate acceleration vector of hand $\partial\mathbf{r}^2/\partial t^2$.
Make_InvKine	Evaluate inverse kinematics.
Make_Thand	Evaluate homogenous matrix of hand from posiotion and orientation of the hand.
Make_Serial	Evaluate geometric parameters of branch k viewed as a serial link robot connected to the base link.
Make_ODP	Make dynamic parameter list.
Make_MDP1	Evaluate base parameters using geometric relations of joints.
Make_MDP2	Evaluate base parameters using regressor (direct method and Grobner method).
Make_InvDyn	Make inverse dynamic model.
Make_DiffModel	Convert inverse dynamic model into differential expression.
Get_Mmatrix	Evaluate inertia matrix $\mathbf{M}$.
Get_Cvterm	Evaluate Coriolis and velocity term $\mathbf{h}$.
Get_Cmatrix	Evaluate dumping coefficient matrix $\mathbf{C}$.
Get_Gterm	Evaluate gravity term $\mathbf{g}$.
Make_LoopCnd	Evaluate constraint condition of closed-loop.
Make_JCnd	Evaluate constraint jacobian of closed-loop $\mathbf{J}_c$.

※ Evaluate ${}^i*^k_{i+1}$ for all $i = 1, \ldots, n_k$ and $k = 1, \ldots, m$.

ブランチの情報を格納したテーブルの各要素はROSAM [7] で用いられていたデータ構造と同じものを用いることでROSAMと互換性を保っている．このデータ構造によりすべてのロボットパラメータを一つの変数として扱い，ロボットの機構の違いを吸収し統一的な解析ができる．また，ROSAM II ではサポートしていないロボット機構を解析する場合，必要な情報を RobotData 型変数の要素に付け加えることで容易に拡張できる．

パラメータの設定のために Windows95/NT 上で動作する

Table 2 Data structure of RobotData

index	information of element
RobotType	type of robot
BranchNum	number of branches
BranchInfo.k	parameters of branch $k, (k = 1, \ldots, m)$
General	gengeralized coordinate vector (for closed-loop robots)
ConCoor	branch and link numbers where virtual cut joints are settled on (for closed-loop robots)
ConDH	DH parameters relate virtual cut joint coordinates to above link coordinates (for closed-loop robots)

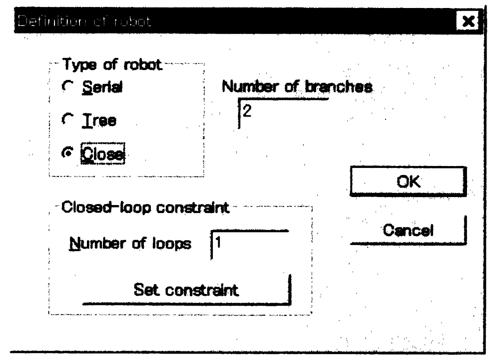

Fig. 1 Dialogbox for setting robot type

GUI がある．これは初めて ROSAM II を利用，あるいはロボット工学に精通していないユーザーでも容易にロボットの機構を定義できるようにすることを目的としている．パラメータの定義は以下のようにして行う．最初に **Fig. 1** に示すダイアログボックスを用いて，解析するロボットがシリアルであるか，木構造であるか，閉ループ機構であるかを選択する．木構造，閉ループ機構ならばブランチ数を入力する．次に DH パラメータや質量，慣性テンソル等の動力学パラメータなどを各リンクごとに **Fig. 2** に示すダイアログボックスを用いて入力する．このとき，各関節の回転型／直動型，駆動／非駆動の区別を右側のラジオボタンを用いて，また，リンクから分岐しているブランチがあれば右下のリストボックスに指定する．閉ループロボットの場合には **Fig. 3** に示すダイアログボックスを用いて仮想切断点の情報を入力する．GUI では各パラメータはデフォルトで与えられるためユーザーは必要なパラメータを変更するだけで設定できる．パラメータは記号と数値のいずれでも入力できる．

3. 運　動　学

3.1 順運動学

順運動学問題は与えられた関節角度に対する手先の位置姿勢を求める問題である．木構造ロボットではベースまたは連結リンクから分岐したリンクから，そのリンクに連なる先端リンクまでをまとめたブランチと呼ばれる複数のシリアルリンクロボットに分解して解析を行う．ブランチの番号はベースから適当な先端リンクまでをブランチ 1 とし，以降ベースから手先に

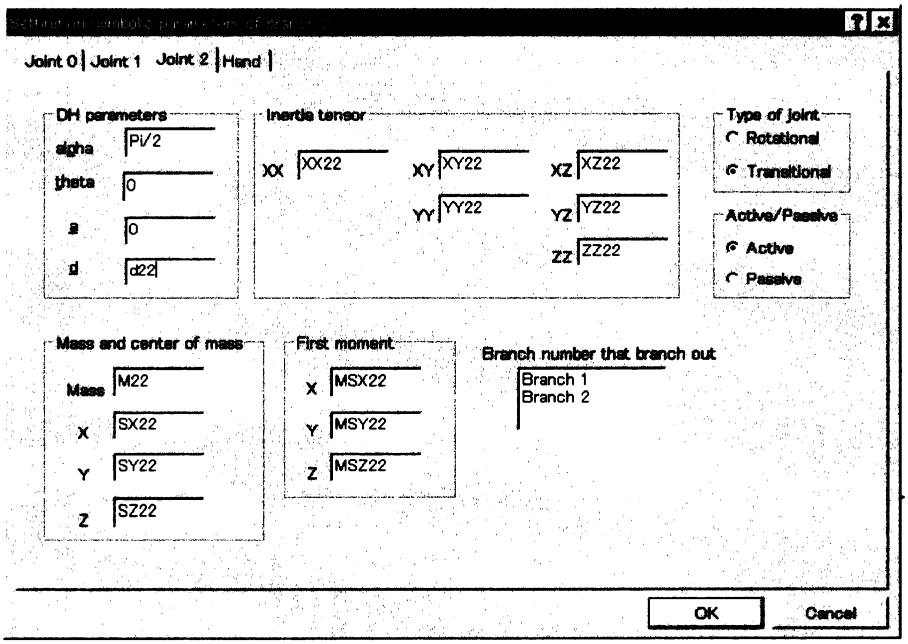

Fig. 2 Dialogbox for setting robot parameters

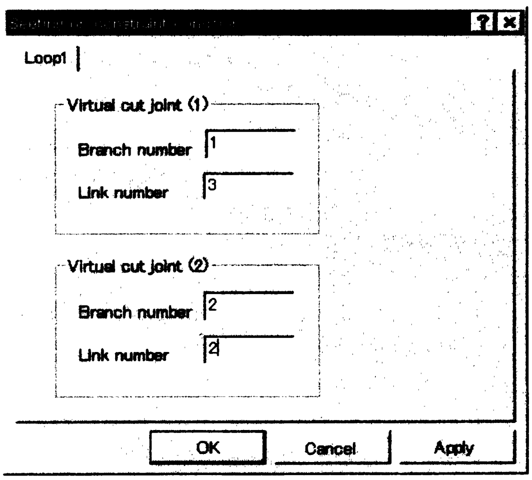

Fig. 3 Dialogbox for setting closed loop condition

向かう順に番号を付与する．ここで，連結リンクとは三つ以上のリンクが連結されているリンクを意味する．隣り合う二つのリンクの幾何学的関係を定義するために各リンクに座標系を設定し，二つの座標系間の関係を修正 DH パラメータ α_i^k，θ_i^k，a_i^k，d_i^k [8] で表す．ここで，右下添え字はリンク番号を示し，右上添え字はブランチ番号を表す．また，連結リンクから分岐する複数のブランチのリンク座標系の幾何学的関係を表すために **Fig. 4** に示す補助 DH パラメータ β^k，b^k を定義する．この座標変換のための同次変換行列を補助同次変換行列と呼び，補助 DH パラメータを用いて次式のように表される．

$$\mathbf{B}^k = \begin{bmatrix} \cos\beta^k & -\sin\beta^k & 0 & 0 \\ \sin\beta^k & \cos\beta^k & 0 & 0 \\ 0 & 0 & 1 & b^k \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

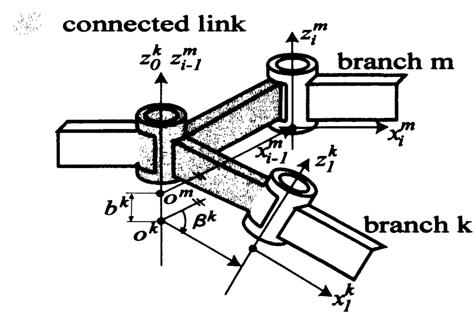

Fig. 4 Sub-DH parameters

このとき，ベース座標からみたブランチ k の手先座標の位置姿勢を表す同次変換行列 ${}^B\mathbf{T}_H^k$ は次式のように表される．

$${}^B\mathbf{T}_H^k = {}^B\mathbf{T}_1^1 \cdot {}^1\mathbf{T}_2^1 \cdots \mathbf{B}^k \cdot {}^0\mathbf{T}_1^k \cdots {}^{n_k}\mathbf{T}_H^k \tag{2}$$

ここで，${}^B\mathbf{T}_1^1$ はベース座標とブランチ 1 の第 1 リンク座標との同次変換行列であり，${}^{n_k}\mathbf{T}_H^k$ はブランチ k の先端リンク座標と手先座標との同次変換行列であり，${}^{i-1}\mathbf{T}_i^k$ はブランチ k の第 $i-1$ リンク座標と第 i リンク座標との同次変換行列である．本論文ではブランチ k の第 i 関節の関節変数を q_i^k と表す．q_i^k は回転関節の場合は θ_i^k であり，直動関節の場合は d_i^k である．

閉ループロボットは閉ループ中の任意の関節で仮想的に切断して仮想木構造ロボットを構成し，もとの閉ループロボットと仮想木構造ロボットが同じ運動を行うという拘束のもとで解析を行う．したがって，閉ループロボットの一般化座標と仮想木構造ロボットの関節変位の関係を示す拘束条件を考慮する必要がある．拘束条件については次節以降で述べる．

手先の姿勢をオイラー角で表したときの手先の位置姿勢ベクトルを $\mathbf{r}$ で表し，関節変数ベクトルを $\mathbf{q}$ で表すと，シリアル

リンクロボットの手先の速度および加速度は次式で与えられる．

$$\left.\begin{aligned}\dot{\mathbf{r}} &= \mathbf{J}\dot{\mathbf{q}} \\ \ddot{\mathbf{r}} &= \dot{\mathbf{J}}\dot{\mathbf{q}} + \mathbf{J}\ddot{\mathbf{q}}\end{aligned}\right\} \tag{3}$$

ここで $\mathbf{J}$ はヤコビ行列

$$\mathbf{J} = \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \tag{4}$$

である．

m 個のブランチを持つ木構造ロボットの場合にはベースからみたブランチ k の手先の位置姿勢ベクトルを $\mathbf{r}^k$ として木構造ロボットの手先の位置姿勢ベクトルを

$$\mathbf{r} = \left[\mathbf{r}^{1^T}, \mathbf{r}^{2^T}, \ldots, \mathbf{r}^{m^T}\right]^T \tag{5}$$

と定義し，またブランチ k の関節変数を $\mathbf{q}^k$ として木構造ロボットの関節変数ベクトルを

$$\mathbf{q} = \left[\mathbf{q}^{1^T}, \mathbf{q}^{2^T}, \ldots, \mathbf{q}^{m^T}\right]^T \tag{6}$$

と定義すると，木構造ロボットのヤコビ行列は下三角ブロック行列として

$$\mathbf{J} = \frac{\partial \mathbf{r}}{\partial \mathbf{q}} = \begin{bmatrix} \frac{\partial \mathbf{r}^1}{\partial \mathbf{q}^1} & 0 & \cdots & 0 \\ \frac{\partial \mathbf{r}^2}{\partial \mathbf{q}^1} & \frac{\partial \mathbf{r}^2}{\partial \mathbf{q}^2} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ \frac{\partial \mathbf{r}^m}{\partial \mathbf{q}^1} & \frac{\partial \mathbf{r}^m}{\partial \mathbf{q}^2} & \cdots & \frac{\partial \mathbf{r}^m}{\partial \mathbf{q}^m} \end{bmatrix} \tag{7}$$

のように表される．手先の速度，加速度ベクトルはシリアルリンクの場合と同様にして，式（3）のように与えられる．

3.2 逆運動学

逆運動学は手先の位置姿勢から関節変位を求める問題である．逆運動学の解法としては Pieper ら [9] の解法などが有名であるが，これらの方法は機構に対する幾何学的な洞察が必要であり，また解くことのできるロボットが限定され，一般的な解法ではない．Raghavan [10] らは運動学方程式を多変数多項式として表して変数消去を行い，1 変数多項式として数値解を求める方法を提案し，Manocha [11] らはこれを発展させて行列の固有値問題に帰着させた．ROSAM II ではグレブナ基底 [12] を用いて冗長性のないシリアルロボットに対する逆運動学の数式解を求められる [13]．グレブナ基底とは与えられた多変数多項式集合のイデアル基底であり，グレブナ基底を縮約規則として任意の多変数多項式を縮約すると唯一の正規形を得る．正規形とはそれ以上縮約できない多変数多項式である．正規形多変数多項式をゼロとしたときの方程式の解は，もとの多変数多項式のそれと一致し，一般に解を求めやすくなる．グレブナ基底は一変数多項式のユークリッドアルゴリズムを多変数多項式に拡張したブフバーガアルゴリズム [12] により求められる．ここでは，n 個の回転関節のみを持つロボットに対する関数 `Make_InvKine` での導出法について概要を簡単に述べる．

手先の位置姿勢を表す同次変換行列において，回転関節 q_i に関する三角関数に対して以下の変数変換を行う．

$$\sin(q_i) = Sq_i \quad \cos(q_i) = Cq_i \tag{8}$$

このとき，位置と姿勢に対する Sq_i，Cq_i を変数とする多変数多項式からなる 12 個の方程式

$$\left.\begin{aligned} f_1(Sq_1, Cq_1, \ldots Sq_n, Cq_n) &= 0 \\ f_2(Sq_1, Cq_1, \ldots Sq_n, Cq_n) &= 0 \\ &\vdots \\ f_{12}(Sq_1, Cq_1, \ldots Sq_n, Cq_n) &= 0 \end{aligned}\right\} \tag{9}$$

を得る．さらにこの式の集合に二つの変数 Sq_i, Cq_i を拘束する式

$$Sq_i^2 + Cq_i^2 - 1 = 0 \tag{10}$$

を加え，重複する式や自明な式を取り除いて得られる方程式の集合を F とする．そして集合 F に対して項間の順序規則として辞書編集順序を用いてブッフバーガーアルゴリズムによりグレブナ基底を求めると，運動学方程式は以下のような三角形式に変換される．

$$\left.\begin{aligned} g_1(x_1) &= 0 \\ g_2(x_1, x_2) &= 0 \\ &\vdots \\ g_{2n}(x_1, x_2, \ldots, x_{2n-1}, x_{2n}) &= 0 \end{aligned}\right\} \tag{11}$$

ここで，x_k，$(k = 1, \ldots, 2n)$ は Sq_i, Cq_i，$(i = 1, \ldots, n)$ のうちの一つを表す．この集合はもとの運動学方程式（9），（10）と同じ解を持ち，その解は変数 $x_1, x_2, \ldots$ について逐次求めることで，式（9）を直接解くよりも容易に解を得ることができる．ただし，ロボットの機構によっては変数の数が多くなり，グレブナ基底を求めるのに数メガバイトから百メガバイト以上の多大な計算機メモリと数分から数日以上の計算時間を必要とし，使用する計算機環境や受容する計算時間によってはグレブナ基底が得られないことがある．そこで，可能ならば集合 F をいくつかの部分集合に分解して，それぞれ個別にグレブナ基底を求めるよう工夫がなされている．手先側 3 関節の関節座標系の原点が一点で交わるような場合はこのような分解が可能となる．なお，q_i が直動関節のときは式（8）の変数変換をせず，三角関数の拘束条件（10）を加えずに直動関節変数 q_i を用いて集合 F を作成する．なお，文献 [13] には，6 自由度シリアルリンクロボットの逆運動学解析例が示されている．参考までに，その計算時間は Pentium100 MHz，メモリ 32 MB の環境で約 30 秒であった．

3.3 閉ループロボットの拘束条件

閉ループロボットでは，**Fig. 5** に示すように閉ループ中の適当な関節を仮想的に切断して得られる仮想木構造ロボットを用いて解析を行う．このとき，仮想木構造ロボットがもとの閉ループと同じ運動を行う拘束条件を加える．関数 `Make_LoopCnd` では仮想木構造ロボットを構成して得る二つのブランチに対して，閉ループの起点の座標系からみたこれら二つの仮想切断点の座

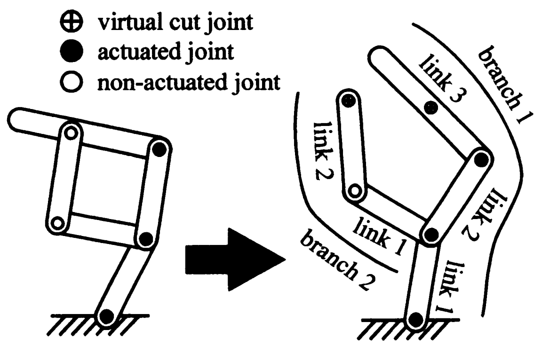

Fig. 5 Virtual tree structured robot

標系の位置姿勢が等しいという条件より，以下の拘束条件式を求める．

$$ {}^{C}\mathbf{T}^{1}_{CUT} = {}^{C}\mathbf{T}^{2}_{CUT} \tag{12} $$

ここで，左上添え字 C は閉ループの起点を表し，右下添え字 CUT は仮想切断点を表す．次に上式の各要素を比較することにより12個の式を求め，これらのうち $0=0$ のような自明な関係式や重複する式を取り除くことにより一般化座標に関して陰な関係式

$$ \mathbf{f}(\mathbf{q},\mathbf{q}_o) = 0 \tag{13} $$

で表される独立な拘束条件を求る．ここで $\mathbf{q}$，$\mathbf{q}_o$ はそれぞれ一般化座標，仮想木構造ロボットの関節変数である．Make_LoopCnd にオプション引数'grobner'が指定された場合は，逆運動学と同様にグレブナ基底を用いて陽な関係式

$$ \mathbf{q}_o = \mathbf{q}_o(\mathbf{q}) \tag{14} $$

を求める．

4. 動力学モデル

ロボットの運動方程式は一般に

$$ \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{F} \tag{15} $$

として与えられる．ここで，$\mathbf{M}(\mathbf{q})$ は慣性行列，$\mathbf{h}(\mathbf{q},\dot{\mathbf{q}})$ は遠心力とコリオリ力，$\mathbf{g}$ は重力，$\boldsymbol{\tau}$，$\mathbf{F}$ はそれぞれ関節トルクと手先に作用する外力を表す．関数 Make_InvDyn では動力学モデルを Newton-Euler 法を用いて作成している．木構造リンクロボットでは連結リンクから分岐するブランチの影響を考慮して導出する．閉ループロボットでは，閉ループ機構の一般化力を $\boldsymbol{\tau}$ とおき，閉ループ中の関節を仮想的に切断することにより得られる仮想木構造機構の関節トルクを $\boldsymbol{\tau}_o$ とおく．このとき，閉ループロボットの運動方程式は対応する仮想木構造ロボットの運動方程式

$$ \mathbf{M}_o(\mathbf{q}_o) + \mathbf{h}_o(\mathbf{q}_o,\dot{\mathbf{q}}_o) + \mathbf{g}(\mathbf{q}_o) = \boldsymbol{\tau}_o \tag{16} $$

と仮想木構造ロボットが閉ループロボットと同一の運動をするという拘束条件（13）から得られる拘束条件のヤコビ行列

$$ \mathbf{J}_c = -\frac{\partial \mathbf{f}}{\partial \mathbf{q}} \Big/ \frac{\partial \mathbf{f}}{\partial \mathbf{q}_o} \tag{17} $$

を用いて，

$$ \boldsymbol{\tau} = \mathbf{J}_c^T\,\boldsymbol{\tau}_o \tag{18} $$

の関係式から求められる．ただし，式（18）を一般化座標を用いて表すには拘束条件（13）から陽な関係式（14）が求められるときに限られる．式（15）の $\boldsymbol{\tau}$ が得られれば，$\mathbf{M}$ は加速度に関する成分を抽出することにより求める．$\mathbf{h}$ はロボットの関節加速度，重力加速度および手先に作用する外力を0とすることにより求め，$\mathbf{h}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}$ と表したときの粘性行列 $\mathbf{C}$ は $\mathbf{M}$，$\mathbf{C}$ の第 i,j 成分をそれぞれ M_{ij}，C_{ij} としたときに，

$$ C_{ij} = \frac{1}{2}\sum_{k=1}^{n}\left(\frac{\partial M_{ij}}{\partial q_k} + \frac{\partial M_{ik}}{\partial q_j} - \frac{\partial M_{jk}}{\partial q_i}\right)\dot{q}_k \tag{19} $$

として求める．$\mathbf{g}$ は重力加速度を含む項のみを抽出することにより求めている．これらはそれぞれ関数 Get_Mmatrix，Get_Cvterm，Get_Cmatrix，Get_Gterm を用いて求められる．

5. ベースパラメータ解析

ロボットの各リンクには質量，慣性テンソル，一次モーメントからなる10個の動力学パラメータが存在する．したがって，n 個のリンクを持つロボットでは $10n$ 個のパラメータ存在する．しかし，運動方程式にこれらすべてのパラメータが現れるわけではなく，運動方程式にまったく関与しないものや，他のパラメータとの線形結合としてのみ現れるものがある．ベースパラメータはロボットの運動方程式を記述するのに必要最小な動力学パラメータの組であり，ROSAM II ではこれらを数式的に解析できる．ベースパラメータを導出するアルゴリズムは，関節の種類と他の関節との幾何学的関係を用いる方法[14]と，ロボットの運動方程式から得られるリグレッサを用いる方法がある．前者は導出に要する計算機の負荷は小さいが，シリアルリンクロボットと木構造ロボットにしか適用できない．後者はロボットの運動方程式が求まればあらゆるロボットに適用できるが，計算機の負荷が大きくなる．後者は後述するように直接法[15]とグレブナ基底を用いる方法[16]とがある．関数 Make_MDP1 は関節の幾何学的条件によって導出し，関数 Make_MDP2 はリグレッサを用いて導出する．また，Make_MDP2 はオプション引数に拘束条件と変数の順序リストを加えることでグレブナ基底を用いて導出を行う．

5.1 直接法による解析

ロボットの運動方程式は，動力学パラメータベクトル $\boldsymbol{\sigma}$ とそのリグレッサ $\mathbf{W}$ を用いて，

$$ \boldsymbol{\tau} = \mathbf{W}\,\boldsymbol{\sigma} \tag{20} $$

として表される．ここで，$\mathbf{W}$ の列ベクトル $\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_s$ の一次独立性を調べ，一次従属となる列ベクトルに対応する動力学パラメータを独立な列ベクトルに対応するパラメータに組み込んでいくことによりベースパラメータを得る．ただし，列ベクトル $\mathbf{w}_i$ の要素は時間関数であるので，このようなベクトルに対する一次独立性の判別に基本関数[16]と呼ぶ概念を用いている．基本関数は，独立変数とその時間微分からなるスカラー関数であり，ある基本関数は他の基本関数の線形結合として表

されることはなく，また $\mathbf{W}$ に含まれるすべての要素は基本関数の線形結合として表すことができるような関数の集合である．

いま，基本関数で表されたリグレッサ $\mathbf{W}$ の列ベクトル $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_s$ のうち，$\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_i$ は一次独立であるものとして，$i+1$ 番目の列ベクトル $\mathbf{w}_{i+1}$ がこれらの線形結合として表されるかどうかを調べる．その結果 $\mathbf{w}_{i+1}$ が一次従属であったとすると，

$$\mathbf{w}_{i+1} = \sum_{j=1}^{i} k_j \mathbf{w}_j \tag{21}$$

として表される．また，パラメータベクトル $\boldsymbol{\sigma}$ を

$$\boldsymbol{\sigma} = [\,\sigma_1, \sigma_2, \ldots, \sigma_s\,]^T \tag{22}$$

とすると，式（20）は

$$\boldsymbol{\tau} = \sum_{j=1}^{s} \mathbf{w}_j \sigma_j \tag{23}$$

として表される．式（21）を式（23）に代入して

$$\boldsymbol{\tau} = \sum_{j=1}^{i} (\sigma_j + k_j \sigma_{i+1})\, \mathbf{w}_j + \sum_{j=i+2}^{s} \mathbf{w}_j \sigma_j \tag{24}$$

を得る．したがって，列ベクトル $\mathbf{w}_{i+1}$ が一次従属のとき，動力学パラメータとリグレッサを以下のように記述し直せる．

$$\begin{aligned} \sigma_j &= \begin{cases} \sigma_j + k_j \sigma_{i+1} & for \quad 1 \le j \le i \\ \sigma_{j+1} & for \quad i+1 \le j \le s \end{cases} \\ \mathbf{W} &= [\mathbf{w}_1, \cdots, \mathbf{w}_i, \mathbf{w}_{i+2}, \cdots, \mathbf{w}_s] \\ s &= s - 1 \end{aligned} \tag{25}$$

これを $i=1$ から $i=s$ まで繰り返すことにより列ベクトルは一次独立となり，ベースパラメータを得る．

5.2 グレブナ基底を用いた解析

閉ループロボットのベースパラメータはリグレッサを用いた解析法を用いる必要がある．閉ループロボットから作られる仮想木構造ロボットに課せられる拘束条件は一般に陽な関係式（14）として与えられるとは限らず，陰な関係式（13）として与えられることが多い．このような場合，拘束条件の1階と2階微分

$$\dot{\mathbf{f}}\,(\mathbf{q}_o, \mathbf{q}) = \mathbf{0} \tag{26}$$

$$\ddot{\mathbf{f}}\,(\mathbf{q}_o, \mathbf{q}) = \mathbf{0} \tag{27}$$

を拘束条件に加えてリグレッサの列ベクトルの一次独立性を解析する必要がある．しかし，これらの拘束条件を考慮しながらリグレッサの列ベクトルの一次独立性の解析を行うのは非常に困難である．そこで ROSAM II ではグレブナ基底を用いて一次独立性の解析を行う．そのために，回転関節 q_i に対して，前述した変数変換（8）を拘束条件とリグレッサの各要素に対して行う．このとき，リグレッサ $\mathbf{W}$ の各要素は Sq_i，Cq_i に関する多変数多項式として表される．また，二つの変数 Sq_i と Cq_i を関係付ける式（10）を拘束条件に加える．そして拘束条件のグレブナ基底を求めて $\mathbf{W}$ の各要素を縮約し，拘束条件のもとで最も簡潔な表現を求める．こうして得たリグレッサに対して，前述の直接法を適用することにより，複雑な閉ループロボットに対してもベースパラメータを求めることができる．しかし，先に述べたように，ロボット機構によってはグレブナ基底を求めるために多大な計算時間や計算機メモリを必要とし，計算機環境や受容される計算時間によってはグレブナ基底が求められない場合があることを示唆しておく．この場合にも逆運動学のところで述べた，計算機負荷を減少させるための工夫をしている．

6. 実　行　例

Fig. 6 に示す平面6関節3自由度閉ループロボットに対して ROSAM II を用いて解析を行った例を示す．**Fig. 7** に対応する仮想木構造ロボットを示す．閉ループロボットの一般化座標は $\mathbf{q} = [\ \theta 11 \quad d21 \quad \theta 31\ ]^T$ とし，仮想木構造ロボットの関節変数は $\mathbf{q}_o = [\ \theta 11 \quad d21 \quad \theta 31 \quad \theta 12 \quad d22\ ]^T$ である．はじめに GUI を用いて仮想木構造ロボットのパラメータファイルを作成する．このパラメータファイルは"`sample.txt`"に保存するものとする．次にこのパラメータファイルと ROSAM II ライブラリファイル"`ROSAMII.txt`"をロードする．これらは Maple V のユーザーインターフェイス上で以下のように行う．

```
> read 'ROSAMII.txt':
> read 'sample.txt':
```

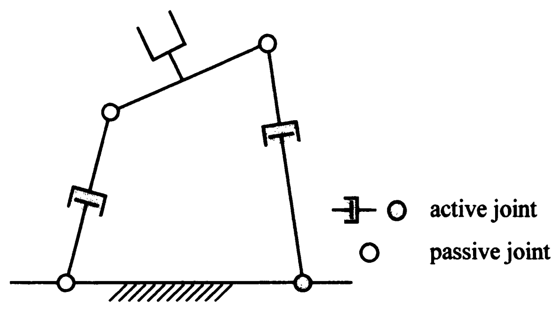

Fig. 6　3 D.O.F. with 6 joints closed-loop robot

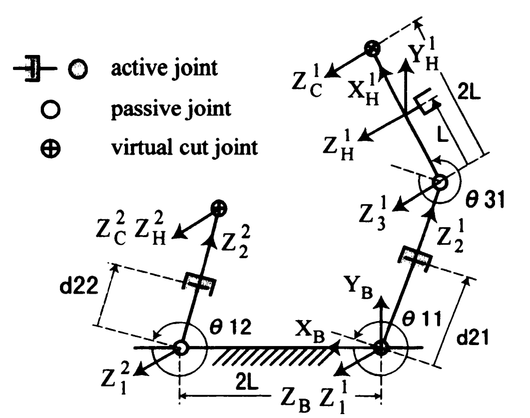

Fig. 7　Virtual tree-structured robot

以下，ロボットのパラメータは "RobotData" という変数名の RobotData 型変数に格納されているとする．ライブラリとロボットのパラメータファイルをロードした後は，ROSAM II のコマンドを用いて所望の解析ができる．紙面の都合上，解析例は閉ループ拘束条件，逆運動学，動力学および，ベースパラメータのみについて示す．

6.1 閉ループ拘束条件

閉ループロボットの解析では対応する仮想木構造ロボットの関節変数ともとの閉ループロボットの一般化座標の拘束条件を求め，他方のブランチの解を拘束条件より求める．これは関数 **Make_LoopCnd** を用いて以下のようにして求められる．

```
> tmp := Make_LoopCnd( RobotData ):
> cnd := map( combine, tmp, trig );
```

$$cnd := [\,2\cos(\theta 11+\theta 31)L+\sin(\theta 11)d21-\sin(\theta 12)d22,\\ -2\sin(\theta 11+\theta 31)L+\cos(\theta 11)d21-\cos(\theta 12)d22\,]$$

評価結果は Maple 関数 **combine** により三角関数の合成をして式の簡単化を行った．Pentium 333 MHz，メモリ 64 MB の計算機で計算したときの計算時間は約 1 秒であった．ここで **map** はリストの各要素に第 1 引数で与えられた関数を適用する Maple 関数である．この場合，拘束条件は陰な関係式として表されているが，**Make_LoopCnd** の引数に **'grobner'** を追加すると陽な関係式が求められる．ただし，一般に複雑な式となる．

6.2 逆運動学

閉ループロボットでは対応する仮想木構造ロボットのベースから手先に連なるブランチに対して逆運動学を求める．はじめに関数 **Make_Thand** を用いて所望する手先の位置姿勢に対する同次変換行列を作成する．引数はベース座標系で表した手先の位置と姿勢を指定し，姿勢表現にはオイラー角を用いる．例題では平面 3 自由度ロボットであるので X_B，Y_B 座標と Z_B 軸まわりの回転をそれぞれ px，py，ϕ と指定し，それ以外は 0 である．

```
> T_HAND := Make_Thand( px, py, 0, phi, 0, 0 );
```

$$T\_HAND := \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 & px \\ \sin(\phi) & \cos(\phi) & 0 & py \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

次に関数 **Make_InvKine** を用いて逆運動学を求めると以下の結果を得る．

```
> Make_InvKine( RobotData, T_HAND, cnd );
```

$$d21 = \left[\sqrt{-2Sp\,L\,py+py^2+px^2+L^2-2px\,L\,Cp},\right.\\ \left.-\sqrt{-2Sp\,L\,py+py^2+px^2+L^2-2px\,L\,Cp}\right]$$

$$\cos(\theta 11)=\frac{d21}{\%1}\left(2Cp\,px\,L^2Sp-2Cp\,L\,py\,px-2L^2Sp^2py\right.\\ \left.+3L\,Sp\,py^2+px^2L\,Sp+L^3Sp-py^3-py\,px^2-L^2py\right)$$

$$\sin(\theta 11)=-\frac{d21}{\%1}\left(Cp\,L^3-2Cp\,L^2Sp\,py-Cp\,px^2\,L-px^3\right.\\ \left.-2px\,L^2Sp^2+L^2px+2L\,Sp\,py\,px-py^2px+Cp\,L\,py^2\right)$$

$$d22=\left[\sqrt{-4L\,d21\sin(\theta 31)+d21^2+4L^2},\right.\\ \left.-\sqrt{-4L\,d21\sin(\theta 31)+d21^2+4L^2}\right]$$

$$\%1=4px^2L^2Sp^2-2L^2px^2+4L^2Sp^2py^2-4L\,Sp\,py^3\\ -4L\,Sp\,py\,px^2-4L^3Sp\,py+py^4+2py^2px^2+2L^2py^2\\ +px^4+L^4$$

ここで，$Cp=\cos(\phi)$，$Sp=\sin(\phi)$ である．四つの解があるが，$d21$ が負の解のときは $\theta 11$ が π だけずれた解となることで，$d22$ が負のときは $\theta 12$ が π だけずれた解となることで所望する手先の位置姿勢となる．ロボットの機構構成によってはこのような解も可能である．計算時間は約 5 秒であった．

6.3 動力学

閉ループロボットの運動方程式は関数 **Make_InvDyn** により求められる．評価結果は紙面の関係で第 1 関節の最初の第 5 項までとする．

```
> Fhh := vector( 6, 0 ):
> tau := Make_InvDyn( RobotData, Dq, DDq,
>   Fhh, 1, cnd );
```

$$\tau := [\,-msx31\cos(\theta 31)DDq_2+msy31\sin(\theta 31)DDq_2\\ -\ 2d21\cos(\theta 31)msy31DDq_1-msx21\,DDq_2\\ -\ 2msy31\cos(\theta 31)Dq_1Dq_2+\ \cdots]$$

関数 **Make_InvDyn** の第 5 引数に 1 を指定すると閉ループロボットの，2 を指定すると仮想木構造ロボットの運動方程式を得る．第 2，3 引数の変数 **Dq**，**DDq** はそれぞれ一般化速度，加速度ベクトルであり，第 4 引数には手先に作用する力／モーメントベクトル **Fhh** を指定する．この場合，手先に力／モーメントは作用しないとして，事前にすべて 0 とした．計算時間は約 20 秒であった．

6.4 ベースパラメータ

ベースパラメータは，閉ループロボットの運動方程式のリグレッサの列ベクトルの一次独立性を調べることにより求める．これは関数 **Make_MDP2** によって実行できる．

```
> ODP := Make_ODP( RobotData ):
> DiffModel := Make_DiffModel( tau, q ):
> MDPVZ := Make_MDP2( RobotData, DiffModel, ODP,
>   cnd, [theta11, d21, theta31] );
```

$$MDPVZ :=\\ [\,(0, m21+m31, 0, 0, m22)\,,(\,[\,0,0,0\,],\\ [\,msx21, 0, msz21\,],[\,msx31, msy31, 0\,],\\ [\,0,0,0\,],[\,msx22, 0, msz22\,]\,),\\ \left(\begin{bmatrix}0&0&0\\0&0&0\\0&0&Inzz11+Inyy21\end{bmatrix},\begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix},\right.\\ \begin{bmatrix}0&0&0\\0&0&0\\0&0&Inzz31\end{bmatrix}\begin{bmatrix}0&0&0\\0&0&0\\0&0&Inzz12+Inyy22\end{bmatrix}$$

$$\left.\left.\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\right)\right)\right]$$

この例では拘束条件が陰な関係式として表されているので，グレブナ基底法によりベースパラメータを求める必要がある．第4引数に拘束条件を，第5引数に変数の順序規則を指定することでグレブナ基底法により解析する．このロボットは五つのリンクを持つため，オリジナルの動力学パラメータは50個となる．これは関数 `Make_ODP` を用いて求められる．また，ベースパラメータ解析の前に，運動方程式を時間の微分記号で表しておく必要があるが，これは関数 `Make_DiffModel` で求められる．この関数の第2引数 $\mathbf{q}$ は一般化座標である．評価結果は初めの () でくくられた部分が質量を表し，次の () が一次モーメントを表す．最後の () にくくられた部分は慣性テンソルを表す．mik はブランチ k の第 i リンクの質量，$msxik$，$msyik$，$mszik$ はそれぞれブランチ k の第 i リンクの x，y，z 軸まわりの一次モーメント，$Inyyik$，$Inzzik$ はそれぞれブランチ k の第 i リンクの y，z 軸まわりの慣性モーメントである．解析の結果，この閉ループロボットのベースパラメータは11個であることが分かる．計算時間は約25分で，これはグレブナ基底の計算に多くの時間を要したのと，ベースパラメータの導出アルゴリズムが多くの計算量を要するためである．

7. まとめ

本論文では Maple V 上で動作するロボット数式処理ライブラリ ROSAM II を示した．ROSAM II を用いることによりシリアルリンク，木構造，および閉ループロボットに対して順運動学，逆運動学，順動力学，逆動力学および，ベースパラメータ解析，および数値シミュレーションができる．ロボットパラメータを設定しやすくするために Windows95/NT で動作する GUI を C++言語を用いて作成した．本論文では ROSAM II に用いられている計算アルゴリズムの概要と解析例を示した．今後は複数の閉ループをもつ場合でも解析をできるようにし，また，ユーザーがより使いやすい GUI を開発していく予定である．さらに，宇宙ロボット，柔軟ロボットなども解析できるよう拡張する予定である．本研究は，(財) 高度自動化技術振興財団の研究助成を一部受けており，ここに謝意を表す．

参考文献

[1] 斉藤制海，梅野孝治 他："REDUCE を用いたロボットアーム運動学および動力学方程式自動生成システム"，日本電気学会C編，vol.109, no.2, pp.89–96, 1989.

[2] J.F. Nethery and M.W. Spong: "Robotica: A Mathematica Package for Robot Analysis," IEEE R&A Magazine, vol.1, no.1, pp.13–23, 1994.

[3] W. Khalil and D. Creusot: "A system for the symbolic modelling of robots," Robotica, vol.15 pp.153–161, 1997.

[4] H. Mayeda, K. Yoshida and K. Osuka: "Base Parameters of Manipulator Dynamic Models," IEEE Trans. on Robotics and Automation, vol.6, no.3, pp.312–321, 1990.

[5] H. Kawasaki and K. Kanzaki: "Minimum Dynamics Parameters of Robot Models," Proc. of IFAC Robot Control, pp.33–38, 1991.

[6] B.W. Char, K.O. Geddes et. al: "Maple V Library Reference Manual," Springer-Verlag, 1991.

[7] 川﨑晴久："ロボット工学ソフトウェア利用の手引き (第1回) ROSAM: 数式処理によるロボット解析システム"，日本ロボット学会誌，vol.14, no.3, pp.370–376, 1996.

[8] W. Kalil and J.F. Kleinfinger: "A New Geometric Notation for Open and Closed-Loop Robots," Proc. IEEE Conf. on Robotics and Automation, pp.1174–1180, 1986.

[9] J.J. Craig: "Introduction to ROBOTICS," Addition Wesly, pp.129–136, 1995.

[10] M. Raghavan and B. Roth: "Kinematics Analysis of the 6R Manipulators of General Geometry," Int. Symp. Robotics Research, pp.314–320, 1989.

[11] D. Manocha and J.F. Canny: "Efficient Inverse Kinematics for General 6R Manipulators," IEEE Trans. on Robotics and Automation, vol.10, no.5, pp.648–657, 1994.

[12] B. Buchberger: "Applications of Grobner Bases in Non-linear Computational Geometry," Lecture Nots in Computer Science 296, pp.52–80, 1988.

[13] T. Shimizu and H. Kawasaki: "An Analysis of Inverse Kinematics of Robot Manipulators using Grobner Basis," Jour. of Robotics and Mechatronics, vol.9, no.5, pp.324–331, 1997.

[14] H. Kawasaki, A. Murata and K. Kanzaki: "An Efficient Algorithm for Generating Manipulator Inertia Matrix Using the Minimum Set of Dynamics Parameters," Jour. of Robotic Systems, pp.262–273, 1996.

[15] 川﨑晴久，村田敦，神崎一男："数式処理による閉リンクロボットの最小動力学パラメータの解析法"，日本ロボット学会誌，vol.13, no.4, pp.558–564, 1995.

[16] 川﨑晴久，清水年美："ロボット動的モデルにおけるベースパラメータのグレブナ基底に基づく解析"，計測自動制御学会論文集，vol.33, no.11, pp.1059–1065, 1997.

川﨑晴久 (Haruhisa Kawasaki)

1949年6月27日生．1974年名古屋大学大学院修士課程修了，同年日本電信電話公社（現NTT）入社，1990年金沢工業大学教授，1994年岐阜大学教授，1996年岐阜大学 VSL 施設長，1998年7月～1999年1月英国サリー大学客員教授，現在に至る．ロボット制御などの研究に従事．日本機械学会，計測自動制御学会，IEEE などの会員．工学博士．

（日本ロボット学会正会員）

清水年美 (Toshimi Shimizu)

1971年12月22日生．1992年岐阜工業高等専門学校卒業，1994年岐阜大学工学部機械工学科文部技官，1997年3月岐阜大学工学部機械工学科（夜間主コース）卒業．現在に至る．ロボット工学の研究に従事．計測自動制御学会の会員．

（日本ロボット学会正会員）