

# 修士論文

## 拘束を伴うマルチボディシステムのための 数値・数式ハイブリッドモデリング

東京都市大学大学院工学研究科  
機械システム工学専攻  
高橋 準  
指導教官 金宮 好和 教授  
佐藤 大祐 講師

平成24年2月29日



# 目 次

<b>第1章 序論</b>	<b>13</b>
1.1 研究背景 . . . . .	13
1.1.1 マルチボディダイナミクス . . . . .	13
1.1.2 ロボット開発のためのシミュレータ . . . . .	14
1.1.3 動力学計算法の歴史 . . . . .	19
1.1.4 数値・数式ハイブリッド手法 . . . . .	21
1.2 研究目的 . . . . .	21
1.3 本論文の構成 . . . . .	22
1.4 本論文で用いる記号 . . . . .	23
<b>第2章 順動力学計算法の比較 [4][41]</b>	<b>25</b>
2.1 運動方程式と関節の条件式 . . . . .	25
2.1.1 Articulated Body の運動方程式 . . . . .	25
2.1.2 関節の条件式 . . . . .	27
2.2 三つのアルゴリズムの概要 . . . . .	29
2.2.1 Assembly-Disassembly Algorithm ( ADA ) . . . . .	29
2.2.2 Articulated-Body Algorithm ( ABA ) . . . . .	32
2.2.3 Divide-and-Conquer Algorithm ( DCA ) . . . . .	34
2.3 アルゴリズムの比較 . . . . .	36
2.4 本章のまとめ . . . . .	37
<b>第3章 Assembly-Disassembly Algorithm ( ADA )</b>	<b>39</b>
3.1 リンク系のモデリング . . . . .	39
3.1.1 シリアルリンク機構 . . . . .	39
3.1.2 ブランチリンク機構 . . . . .	40
3.1.3 閉リンク機構 . . . . .	45
3.2 3節開リンク機構と4節閉リンク機構の比較 . . . . .	53
3.2.1 計算アルゴリズムの比較 . . . . .	53

3.2.2 開リンク機構と閉リンク機構の計算の違い . . . . .	54
3.3 4節閉リンク機構の特異姿勢 . . . . .	55
3.4 浮遊ベースのモデリング . . . . .	56
3.4.1 オイラー角による姿勢表現 . . . . .	56
3.4.2 単位クォータニオンによる姿勢表現 . . . . .	59
3.5 本章のまとめ . . . . .	65
<b>第4章 数値・数式ハイブリッドモデリング</b>	<b>67</b>
4.1 リンク系の運動方程式 . . . . .	67
4.2 Lagrange法とADA, ハイブリッドモデリングの比較 . . . . .	70
4.2.1 Lagrange法 . . . . .	70
4.2.2 ADA . . . . .	71
4.2.3 ハイブリッドモデリング . . . . .	72
4.3 ハイブリッドモデリングによる運動方程式の導出 . . . . .	73
4.3.1 根元の4リンクのABのパラメータの導出 . . . . .	73
4.3.2 手先の3リンクのABのパラメータの導出 . . . . .	75
4.3.3 関節の加速度の導出 . . . . .	76
4.4 順動力学計算の確認 . . . . .	77
4.4.1 固定ベースに接続された平面7リンクマニピュレータ . . . . .	77
4.4.2 浮遊ベースを含む三次元6リンクマニピュレータ . . . . .	80
4.5 有用性の検証 . . . . .	84
4.5.1 固定ベースに接続された二次元モデル . . . . .	84
4.5.2 浮遊ベースを含んだ三次元モデル . . . . .	86
4.6 本章のまとめ . . . . .	87
<b>第5章 マルチボディダイナミクスシミュレータ</b>	<b>89</b>
5.1 シミュレータのプログラムの流れ . . . . .	89
5.2 シミュレータのファイル構造 . . . . .	92
5.3 シミュレーション例: PA-10 . . . . .	94
5.4 本章のまとめ . . . . .	95
<b>第6章 ミュレータの実用例</b>	<b>99</b>
6.1 反動零空間法(Reaction Null-Space: RNS) . . . . .	99
6.2 人間の姿勢運動戦略 . . . . .	101
6.3 小型人型ロボットへの導入 . . . . .	105

6.3.1 Upper-body rotation strategy . . . . .	106
6.3.2 Combined rotation-extension/flexion strategy . . . . .	108
6.4 本章のまとめ . . . . .	112
<b>第7章 閉リンク機構を考慮した制御</b>	<b>113</b>
7.1 制御手法 . . . . .	113
7.2 制御例 . . . . .	115
7.2.1 制御入力なし . . . . .	115
7.2.2 4節平行リンク機構として制御 . . . . .	115
7.2.3 スライダークランク機構として制御 . . . . .	119
7.3 閉リンク機構の特異姿勢 . . . . .	124
7.3.1 4節平行リンク機構として制御 . . . . .	124
7.3.2 スライダークランク機構として制御 . . . . .	132
7.4 本章のまとめ . . . . .	132
<b>第8章 結論および今後の研究課題</b>	<b>137</b>
8.1 結論 . . . . .	137
8.2 今後の研究課題 . . . . .	139
<b>謝辞</b>	<b>141</b>
<b>参考文献</b>	<b>143</b>
<b>研究成果</b>	<b>149</b>
<b>付録A 空間ベクトルの基礎式</b>	<b>151</b>
A.1 モーションベクトルとフォースベクトル . . . . .	151
A.2 速度 . . . . .	152
A.3 力 . . . . .	153
A.4 変換行列 . . . . .	154
A.4.1 モーションベクトル(速度) . . . . .	154
A.4.2 フォースベクトル(力) . . . . .	156
A.4.3 一般的な変換 . . . . .	156
A.5 運動量 . . . . .	157
A.6 慣性テンソル . . . . .	158
A.7 慣性行列の逆行列 . . . . .	159

A.8 慣性行列の座標変換 . . . . .	160
A.8.1 慣性行列 . . . . .	160
A.8.2 慣性行列の逆行列 . . . . .	161
A.9 外積 . . . . .	162
A.10 本章のまとめ . . . . .	162

# 図 目 次

1.1	Multibody systems: (a) robot ©KAWADA, (b) industrial machine ©DMG, (c) construction machine ©HITACHI, (d) car ©TOYOTA, (e) train ©HITACHI and (f) satellite ©JAXA. . . . .	13
1.2	Structure of a simulator for robotics. . . . .	15
1.3	Robot simulators: (a) OpenHRP3 ©AIST and (b) Open Dynamics Engine ©Russell Smith. . . . .	16
1.4	Robot simulators: (a) Webots ©AAI japan and (b) V-REP ©HiBot.	17
1.5	Robot simulators: (a) Adams ©MSC software and (b) DADS ©LMS.	18
1.6	RecurDyn for Robot simulator ©FunctionBay K.K. . . . .	19
2.1	The relationship between the applied force, $\mathcal{F}$ , and the resulting ac- celeration, $\dot{\mathcal{V}}$ : (a) a rigid body and (b) an articulated body. . . . .	26
2.2	Representation of the Joint $j$ . . . . .	27
2.3	Illustration of the assembly and disassembly phases. . . . .	29
2.4	Assembly of articulated bodies for a serial link structure. $A$ , $B$ and $C$ have two handles, respectively. . . . .	30
2.5	Aspect of the articulated-body algorithm (ABA). . . . .	33
3.1	A planar 3R serial chain structure. . . . .	40
3.2	Snapshots from simulation of the planar 3R serial chain structure. . .	41
3.3	Simulation results of the planar 3R serial chain structure: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy. . . . .	42
3.4	Assembly of articulated bodies for a branch link structure. $D$ and $E$ have three handles, $F$ has four handles, respectively. . . . .	43
3.5	A planar 5R branch chain structure. . . . .	45
3.6	Snapshots from simulation of the planar 5R branch chain structure. .	46

3.7	Simulation results of the planar 5R branch chain structure: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy. . . . .	47
3.8	Assembly of articulated bodies for a closed chain structure. $G$ and $H$ have three handles, $I$ has two handles, respectively. The connection of joints are two handles at the same time. . . . .	48
3.9	A planar 4R closed chain structure. . . . .	50
3.10	Snapshots from simulation of the planar 4R closed chain structure. . .	51
3.11	Simulation results of the planar 4R closed chain structure: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy. . . . .	52
3.12	Compare 3R open chain to 4R closed chain. . . . .	53
3.13	Illustration of the assembly and disassembly phases of 3R open chain. .	54
3.14	Illustration of the assembly and disassembly phases of 4R closed chain. .	54
3.15	Singular configuration of four-bar linkage. . . . .	56
3.16	Snapshots from simulation of the planar 4R closed chain structure. .	57
3.17	Simulation results of the planar 4R closed chain structure: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) mechanical energy and (f) determinant $JJ^T$ . . . . .	58
3.18	Geometrical interpretation of the six-DOF joint position variables. .	59
3.19	The free base with a six-DOF joint. . . . .	61
3.20	Snapshots from simulation of the 3D free base. . . . .	62
3.21	Simulation results of the 3D free base: (a) base torque, (b) base force, (c) base angular velocity, (c) base angular acceleration, (d) base velocity and (e) base angle. . . . .	63
3.22	Simulation results of the 3D free base: (a) base position, (b) base velocity, (c) base acceleration, (d) CoM position, (e) CoM velocity and (f) mechanical energy. . . . .	64
4.1	Assembly of articulated bodies for hybrid modeling. . . . .	68
4.2	A planar 7R manipulator. . . . .	70
4.3	Illustration of the assembly and disassembly phases based on the ADA. .	71
4.4	Illustration of the assembly and disassembly phases based on the hybrid modeling. . . . .	72

4.5	Illustration of the hybrid modeling for the planar 7R manipulator. . . . .	73
4.6	Snapshots from simulation of the planar 7R manipulator with fixed base. .	78
4.7	Simulation results of the planar 7R manipulator with fixed base: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy. . . . .	79
4.8	A 3D model for the hybrid modeling with free base. . . . .	80
4.9	Snapshots from simulation of the 3D 5R manipulator with free base. .	81
4.10	Simulation results of the 3D 5R manipulator with free base: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy. . . . .	82
4.11	Simulation results of the 3D 5R manipulator with free base: (a) base position, (b) base velocity, (c) base acceleration, (d) CoM position and (e) CoM velocity. . . . .	83
4.12	Planar models for the hybrid modeling with fixed base. . . . .	84
4.13	Computation time of the planar $n$ R manipulator with fixed base. . . .	85
4.14	3D models for the hybrid modeling with free base. . . . .	86
4.15	Illustration of the hybrid modeling. We prepare the equation of motion of a three link manipulator except (a). (a) is a single rigid body. . . . .	86
4.16	Computation time of the 3D $n$ R manipulator with free base. . . . .	87
5.1	Program flowchart of the simulator: (a) overview of the simulator and (b) function of the Ode45 of MaTX. . . . .	90
5.2	program flowchart of the simulator. . . . .	92
5.3	The PA-10 7R manipulator: (a) kinematic structure and (b) initial angles. . . . .	94
5.4	Snapshots from simulation of the PA-10 7R manipulator. . . . .	96
5.5	Simulation results of the PA-10 7R manipulator: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy. . . . .	97
6.1	Reaction patterns in response to an impact force on the shoulder from behind (obtained via motion capture): (a) initial state, (b) upper-body rotation strategy (smaller impact) and (c) combined rotation-extension/flexion strategy (larger impact). . . . .	101
6.2	The joints used in the motion capture model. . . . .	102

6.3	Motion capture data from upper-body rotation strategy (impact on the front). . . . .	103
6.4	Motion capture data from combined rotation–extension/flexion strategy (impact from behind). . . . .	104
6.5	Simulation model: (a) coordinate frames ( $\theta_i = 0, i = 1, 2, \dots, 10$ ) and (b) initial position of the simulation of the upper-body rotation strategy.	105
6.6	Simulation results of upper-body rotation strategy. The robot approaches the door with its left hand. The door is suddenly opened, and an unexpected impact disturbance is applied to the left hand of the robot. . . . .	107
6.7	Spatial three-DOF model for the combined rotation–extension/flexion strategy. . . . .	109
6.8	Simulation results of the combined rotation–extension/flexion strategy (impact from behind). The robot is standing behind the door. The door is suddenly opened, and an unexpected impact disturbance is applied to the back at the right shoulder via the door knob. . . . .	111
7.1	Closed loop dynamic controller with virtual external force $\mathbf{F}^{ex}$ . . . . .	114
7.2	A planar four link manipulator (robot). . . . .	114
7.3	Snapshots from simulation of the planar four link manipulator. . . . .	116
7.4	Simulation results of the planar four link manipulator (not controlled): (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity. . . . .	117
7.5	A parallel four-bar linkage. . . . .	118
7.6	Calculation of the virtual external force $\mathbf{F}^{ex}$ for the parallel four-bar linkage. . . . .	118
7.7	Snapshots from simulation of the parallel four-bar linkage. . . . .	120
7.8	Simulation results of the parallel four-bar linkage: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity. . . . .	121
7.9	Simulation results of the parallel four-bar linkage: (a) virtual external torque $n_1^{ex}$ , (b) error of joint angle $\theta_1$ and (c) error of joint velocity $\dot{\theta}_1$ . .	122
7.10	A slider-crank mechanism. . . . .	123

7.11 Calculation of the virtual external force $\mathbf{F}^{ex}$ for the slider-crank mechanism. . . . .	123
7.12 Snapshots from simulation of the slider-crank mechanism. . . . .	125
7.13 Simulation results of the slider-crank mechanism: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity. . . . .	126
7.14 Simulation results of the slider-crank mechanism: (a) virtual external force $f_4^{ex}$ , (b) error of joint position $r_{4x}$ and (c) error of joint velocity $v_{4x}$ . . . . .	127
7.15 Singular configuration of the four-bar linkage. . . . .	128
7.16 Singular configuration of the slider-crank mechanism. . . . .	128
7.17 Snapshots from simulation of the parallel four-bar linkage around singularity. . . . .	129
7.18 Simulation results of the parallel four-bar linkage around singularity: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity. . . . .	130
7.19 Simulation results of the parallel four-bar linkage around singularity: (a) virtual external torque $\tau_1^{ex}$ , (b) error of joint angle $\theta_1$ and (c) error of joint velocity $\dot{\theta}_1$ . . . . .	131
7.20 Snapshots from simulation of the slider-crank mechanism around singularity. . . . .	133
7.21 Simulation results of the slider-crank mechanism around singularity: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity. . . . .	134
7.22 Simulation results of the slider-crank mechanism around singularity: (a) virtual external force $f_4^{ex}$ , (b) error of joint position $r_{4x}$ and (c) error of joint velocity $v_{4x}$ . . . . .	135
A.1 Vector spaces with their duals and the Jacobian map. . . . .	152
A.2 The velocity of a rigid body expressed in terms of $\boldsymbol{\omega}$ and $\mathbf{v}_O$ . . . . .	153
A.3 The force acting on a rigid body expressed in terms of $\mathbf{n}_O$ and $\mathbf{f}$ . . . . .	154
A.4 Coordinate frame: (a) $\{A\}$ and $\{B\}$ are two coordinate frames with a common origin at $O$ and (b) $O$ and $P$ are points in space. . . . .	155
A.5 $\{A\}$ and $\{B\}$ are two coordinate frames with origin at $O$ and $P$ . . . . .	157

A.6 Spatial momentum. . . . .	158
-------------------------------	-----

# 表 目 次

1.1	The summary of the written abbreviations. . . . .	23
1.2	The summary of the symbols. . . . .	23
1.3	The summary of the symbols. . . . .	24
2.1	Joint model formula. . . . .	28
2.2	Joint model formula. . . . .	29
2.3	Comparison of the four algorithms: selection of articulated-body inertia, joint constraint representation, applicability to closed chains (CC), applicability to parallel computation (PC), computation of joint constraint forces $\lambda$ , size of inverse matrix (Inverse) and complexity . . . . .	36
4.1	Initial state of each joints. . . . .	80
4.2	Computation time of the planar $nR$ manipulator with fixed base [s]. . . . .	85
4.3	Computation time of the 3D $nR$ manipulator with free base [s]. . . . .	87
5.1	Simulator structure. . . . .	93
5.2	Initial angles. . . . .	94
A.1	Spatial vector: motion vector and force vector. . . . .	151



# 第1章 序論

## 1.1 研究背景

### 1.1.1 マルチボディダイナミクス

ロボット、産業機械、建設機械、自動車、列車、宇宙構造物、衛星などの機械・構造物や人間・動物などの生体類は、多くの部品や部分が相互に関連した複雑な動きをなす機構系を構成している。マルチボディシステムとは、このようなシステムの総称であり、これらの動力学と制御を扱う学問をマルチボディダイナミクスと呼ぶ。マルチボディシステムに対して高速運動が要求されると、加速度や慣性力を無視した解析や設計は役に立た

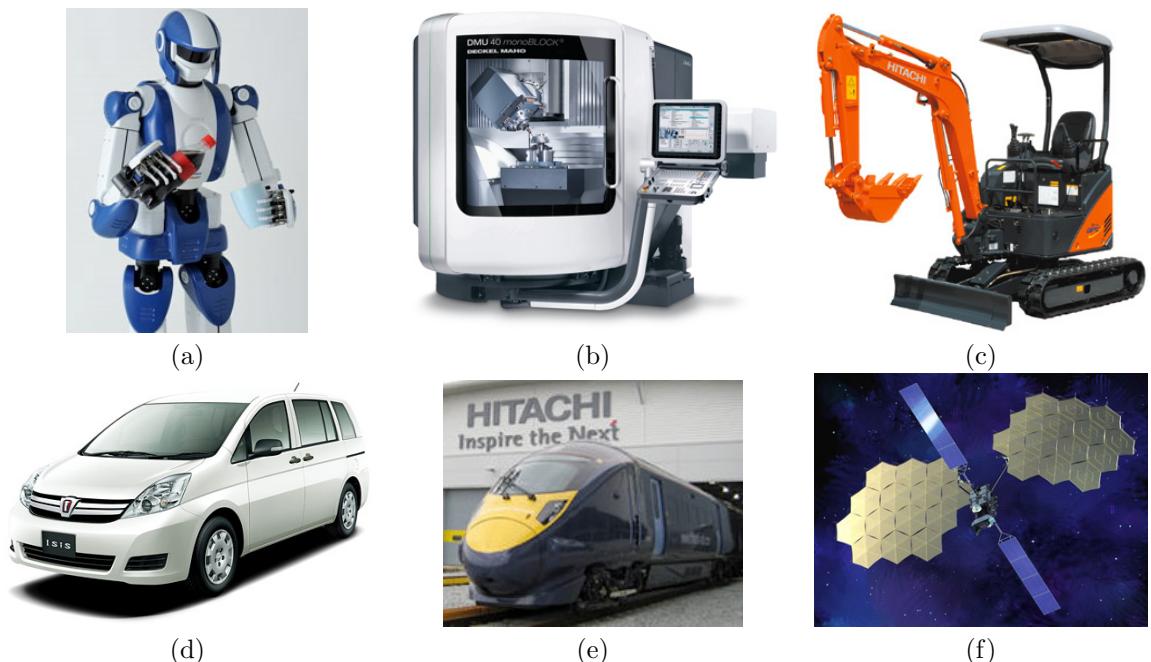


Fig. 1.1: Multibody systems: (a) robot ©KAWADA, (b) industrial machine ©DMG, (c) construction machine ©HITACHI, (d) car ©TOYOTA, (e) train ©HITACHI and (f) satellite ©JAXA.

なくなる。すなわち、マルチボディダイナミクスは、マルチボディシステムの解析や設計、評価に不可避の技術になってきた[1]–[5]。

本論文では、多くのマルチボディシステムの中でも、特にロボットに焦点を当てて述べていく。ロボットの中でも人型ロボットは大自由度系と呼ばれ、特に複雑なシステムを成している。人型ロボットは、歩行中の両足支持期だけでなく、物体把持や手すりを持つ際には閉リンク機構が現れ、また切れて開リンク機構に戻るなど、リンク系の構造変化を扱うことが必要になる。また、台座が床面に固定されている産業用ロボットと違い、人型ロボットの足裏は床面と接触しているだけで固定されていない。このため、一般的に、腰部をベース部とした浮遊リンク系として扱われる。さらに、胴体と手足など、一つのリンクに対して、三つ以上のリンクが繋がったプランチリンク機構を有している。このように、人型ロボットの研究では、閉リンク機構を含む様々なリンク系の扱いが必要になる[6]–[8]。また、このような技術は、自動車や宇宙構造物など、他のマルチボディシステムと共通する課題も多く、ロボティクス分野の発展が他のマルチボディシステムに対して大きな影響を与えることが期待できる。

### 1.1.2 ロボット開発のためのシミュレータ

ロボットの機構設計や運動解析、動作軌道の生成、制御系の設計などを効率的に行うために、動力学シミュレータは不可欠である。本論文では、特に運動解析や制御系の設計を行える動力学シミュレータについて述べていく。ロボットのためのシミュレータの構成はFig. 1.2に示すように、大きく四つに分類できる。一つ目は、ソルバである。運動学や動力学、衝突などの計算を行う部分であり、シミュレーションの物理環境を構築する部分である。二つ目は、コントローラであり、ロボットの運動などを制御する部分である。制御手法により必要な計算は異なるが、ソルバと同様に運動学や動力学を必要とする場合が多い。三つ目は、モデルである。ロボットの機構や物理パラメータ、搭載されているセンサなどをモデリングする部分である。最後は、ユーザインターフェースである。ユーザがシミュレーションの条件を入力したり、シミュレーション結果を確認する部分である。

この四つの中でも、特にソルバで用いる動力学計算法は、シミュレータの性能を大きく左右する。以下では、一般的な既存のシミュレータ(OpenHRP3やOpen Dynamics Engine(ODE), Adams, DADS, RecurDyn)に対して、特徴

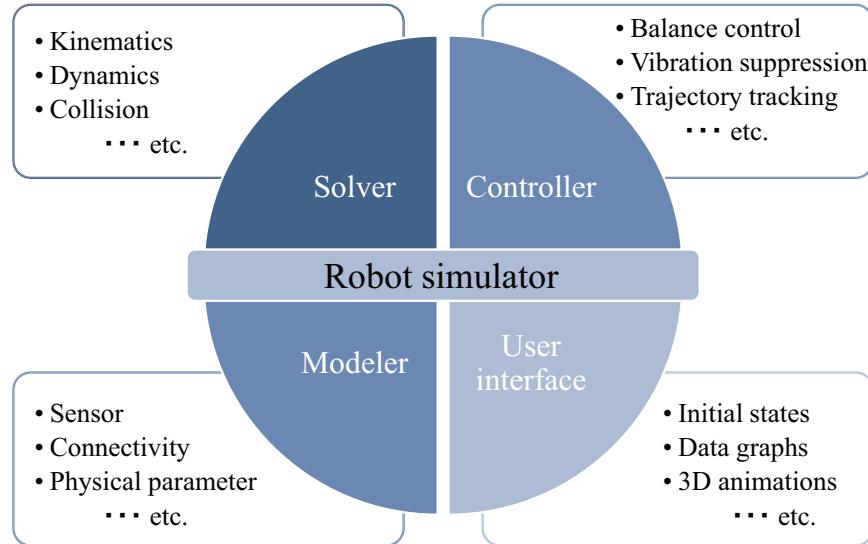


Fig. 1.2: Structure of a simulator for robotics.

と動力学計算法を示す。

### OpenHRP3[9]–[11]

OpenHRP3 はロボットのソフトウェア開発・シミュレーションのための統合ソフトウェアプラットフォームである。OpenHRP3 は分散オブジェクトシステムとして設計されており、各種機能を提供するサーバプログラム群と、サーバの管理を行うクライアントプログラムから構成される。この構成により、大規模なシミュレーションシステム開発における保守性・可搬性を向上させている。OpenHRP3 の力学計算エンジンは、東京大学の開発によるものと、産業技術総合研究所（産総研）の開発によるものがあり、それぞれ以下のような特徴がある。

#### 1. 東京大学版

順動力学計算法には計算量が関節数に比例し、閉リンク機構の計算が可能な独自の計算法 Assembly-Disassembly Algorithm (ADA) [12][13] を採用している。これは、同等の能力を持つアルゴリズムの中では世界的にも最速なもの一つであり、複雑なモデルにおいても高速なシミュレーションが可能である。接触力の計算には数値的に比較的安定とされる剛体接触モデルを用いている。具体的には、Lloyd の方法 [14] と同様に接触問題を線形相補性問題 (Linear Complementarity

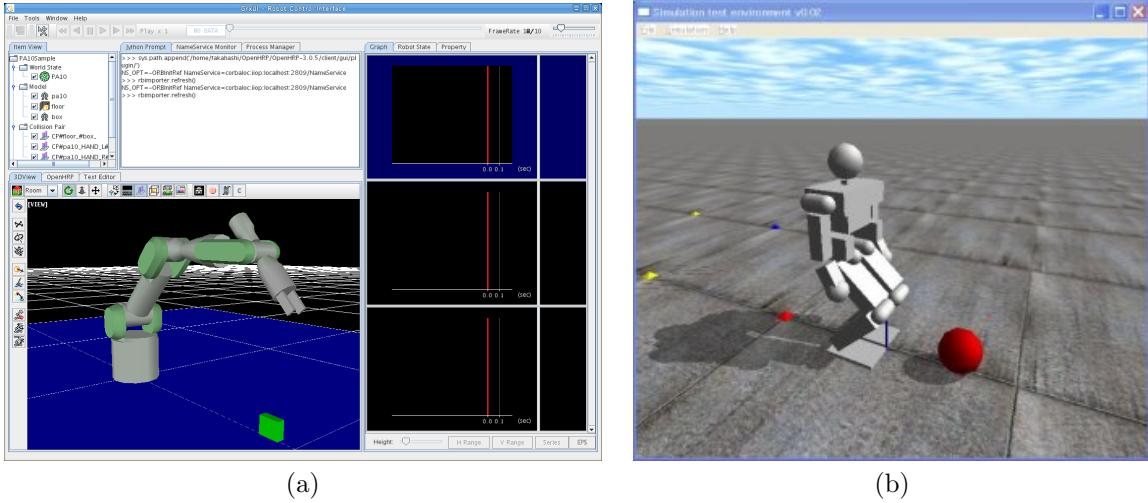


Fig. 1.3: Robot simulators: (a) OpenHRP3 ©AIST and (b) Open Dynamics Engine ©Russell Smith.

Problem: LCP) として定式化し , Lemke 法に基づく解法 [15] を適用して接触力を求めている . ただし , 剛体リンク系の接触問題では単純に Lemke 法を適用しても解が得られないことが多いため , 独自の工夫を加えて安定に解を得る方法を開発している [16] .

## 2. 産総研版

Featherstone のアルゴリズム [17] を採用し , 関節数に比例した計算量で順動力学計算を行う . 接触力の計算には , 拘束条件の解を収束計算で求めるアルゴリズム [18] を採用しており , 剛体接触を数値計算上安定に処理することができる . 力学計算プログラムは外部からライブラリとして使用することを意識した設計であり , コントローラやパターンジェネレータの開発に活用可能な使い勝手のよいライブラリである .

### Open Dynamics Engine ( ODE ) [19]–[21]

Open Dynamics Engine ( ODE ) は Russell Smith 氏が中心となって開発を行っている動力学シミュレーション用のライブラリである . ゲームなどに使用される動力学エンジンであり , 高速で安定した計算が行えることが特徴である . 拘束計算には解析法 [22] を用いている . 解析法では , 多剛体系の各剛体に対して成り立つ運動方程式と , 関節部や環境との接触による拘

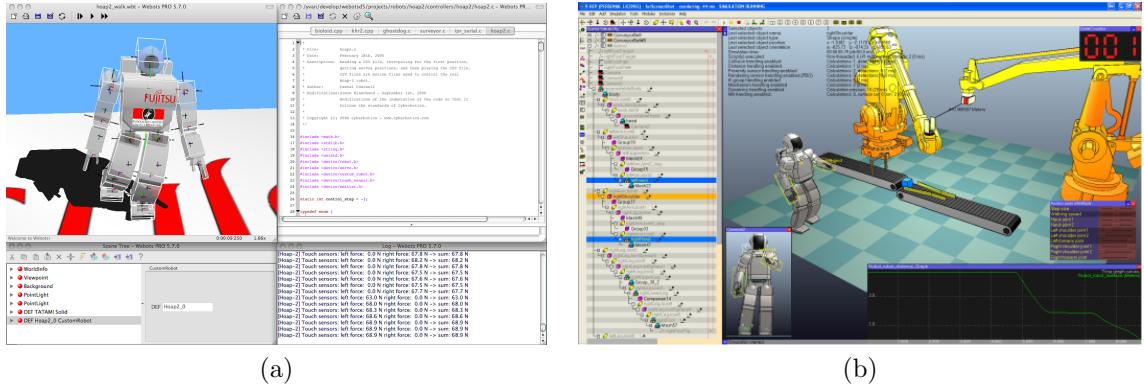


Fig. 1.4: Robot simulators: (a) Webots ©AAI japan and (b) V-REP ©HiBot.

束条件式を連立して解く。計算量は  $O(k^3)$  ( $k$ : 拘束される自由度数) となり低速であるが、環境からの拘束力を解析法で解くと、ばね・ダンパ接触モデルなどと比較して解の安定性がよく、積分刻みを大きくとることでシミュレーションの速度を上げることが可能である。しかし、外力の計算に一次のオイラー積分を使用しているなど、精度はあまり高くないため、精度を必要とするシミュレーションには不向きである。ODEはシミュレータの物理計算ライブラリとしてWebots 6やV-REPなどに用いられており、それぞれ以下のような特徴がある。

### 1. Webots 6[23]

株式会社 AAI ジャパンが開発した汎用ロボットシミュレータである。車輪移動型や歩行型、アーム型、飛行船などの各種移動ロボットを、ユーザが容易にモデル化しシミュレーションできる。また、多様なセンサやアクチュエータをモデル化するためのライブラリを持つ。衝突計算や動力学計算を含む物理計算ライブラリとしてODEを用いている。

### 2. V-REP[24]

株式会社ハイボットが提供するロボットシミュレータであり、ロボット全体、またはセンサなどの一つの機能のみを簡単に編集・シミュレーションできる。プランチ機構、閉リンク機構を含む様々なリンク機構に対応しており、それぞれのリンク機構の順運動学、逆運動学の計算が行える。衝突計算や動力学計算を含む物理計算ライブラリとしてBullet[25]とODEの二つのエンジンを採用している。

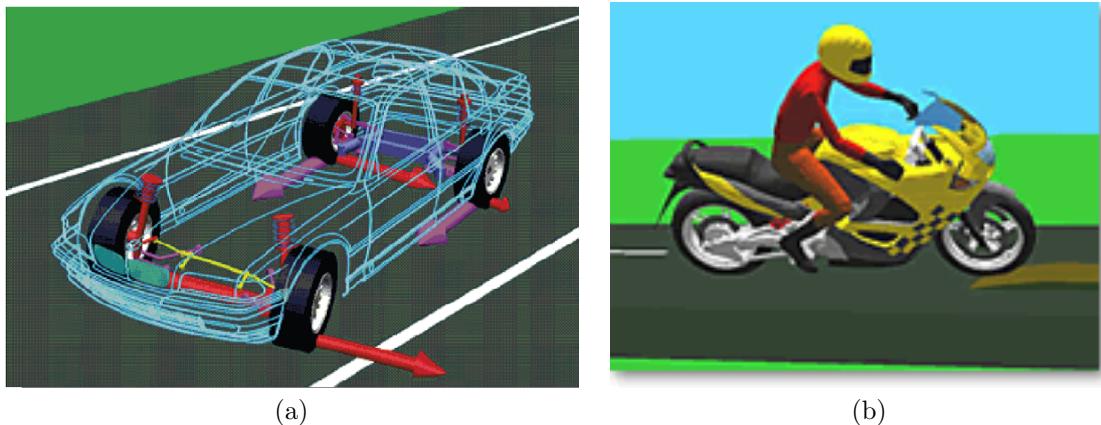


Fig. 1.5: Robot simulators: (a) Adams ©MSC software and (b) DADS ©LMS.

### Adams[26]

Adamsは、Mechanical Dynamics, Inc. が開発した、世界中で最も利用されている汎用機構解析ソフトウェアである。自動車をはじめとして建設機械や電機精密機器など、マルチボディシステムの複雑な挙動をシミュレーションすることができる。

Adamsは、特に動きが大きく複雑な機構システムを取り扱うのに適している。運動学解析、静的つり合い解析、準静的つり合い解析、動力学解析が行える。運動方程式の導出には、Euler-Lagrange法が用いられている。

### DADS[27][28]

DADS (Dynamic Analysis and Design System)は、マルチボディシステムの挙動や力などの運動学的あるいは動力学的な動作特性を予測・評価する汎用の機構解析ソフトウェアである。特に、部品が弾性体であるシステムや制御系・油圧系を組み込んだ機構、摩擦・衝突を含んだ非線形性の強いシステムに対して精度よく解析できる。

解析オプションとして、組立解析、静解析、運動学解析、動力学解析、逆動力学解析、予荷重解析がある。内部では、並進と回転の微小変化に対する仮想仕事の原理から変分形式のNewton-Euler運動方程式を導き、Lagrange乗数によって拘束方程式と結合させている[27]。

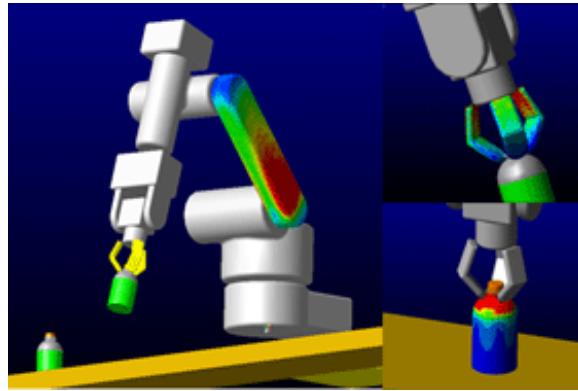


Fig. 1.6: RecurDyn for Robot simulator ©FunctionBay K.K.

### RecurDyn

RecurDyn は、FunctionBay K.K. が提供する機構解析ソフトウェアである。製品開発におけるマルチボディシステムの挙動を評価するための設計支援ツールとして用いられている。運動学解析、動力学解析などの標準的な機構解析機能から、大変形かつ接触を考慮した非線形弾性体まで扱える機構 + 構造連成解析や、複雑な制御系のシステムと完全融合した制御 + 機構連成解析などの解析機能がある。運動方程式は再帰的な方法 (Recursive Formulation) を用いており、関節数に比例する計算効率を実現している。

#### 1.1.3 動力学計算法の歴史

マルチボディシステムの解析には、運動学解析と動力学解析がある。運動学解析とは、機構がモータなどのアクチュエータにより駆動されたとき、どのような軌跡を通って運動するかを調べる解析である。つまり、各時刻における機構の位置、速度、加速度を求める問題である。この場合、純粋な幾何学の問題となる。一方、動力学解析には、順動力学解析と逆動力学解析がある。順動力学解析では、機構がモータなどのアクチュエータにより駆動されたとき、機構に作用する力、およびその結果として生じる運動（動的挙動）を調べることができる。逆動力学解析では、運動の加速度が与えられたときに、それを実現する駆動力を求めることができる。

動力学解析の計算法は開リンク機構に広く研究されてきた。特に Luh らの逆動力学計算は Newton-Euler 法 [29] と呼ばれている。この計算法は関節数を  $n$  とすると計算量が  $n$  に比例する（これを  $O(n)$  と書く）高効率なものであることが特徴である。Newton-Euler 法は、シリアルリンク機構やプラ

ンチリンク機構，浮遊ベースを扱うことができる。

順動力学計算は計算量の点で逆動力学よりも多くなる。順動力学計算の一般的な方法は，Newton-Euler法などの逆動力学計算を利用して運動方程式の各行列成分を求め，これを関節加速度について解き，関節加速度を積分して関節速度と関節角度を求める方法である。これは単位ベクトル法と呼ばれている[30]。この計算過程で慣性行列を求めるのに $O(n^2)$ の計算が，慣性行列の逆行列を求めるのに $O(n^3)$ の計算が必要になる。 $O(n^2)$ の計算に対する問題は，慣性行列を直接計算するアルゴリズム[1][30]を導入することで解決できる。しかし，慣性行列の逆行列を解く計算は，Gauss-Jordanの掃出し法やLU分解などで効率よく実現できるが， $O(n^3)$ の計算は避けられない。このため，自由度の大きな人型ロボットなどでは，この計算は負担が大きくなる。Featherstone の Articulated-Body Algorithm (ABA) [1][17] はこの計算を巧みに回避することで $O(n)$ の計算効率を初めて実現した。ただし，総計算量は逆動力学計算の $O(n)$ とは比べものにならないほど多い。ABA の他にも， $O(n)$ の効率をもつアルゴリズムが提案されている[31]–[35]。Jain はそれらのアルゴリズムは表記法が異なるものの，基本的には同一のアルゴリズムであることを示した[36]。

計算量では理論的に $O(n)$ が最も効率が高い。計算の高速性を追求するために順動力学の並列計算法が開発されている。最も高速なものは， $n$ 個のCPUを並列に用いて計算した際に $O(\log n)$ の計算時間となる。Fijany らの Constraint Force Algorithm (CFA) [37] はシリアルリンク機構に対して初めて $O(\log n)$ を実現したものである。Featherstone は ABA を拡張し， $O(\log n)$ の効率をもつ Divide-and-Conquer Algorithm (DCA) [38][39] を開発した。また，山根らは同様の効率をもつ Assembly-Disassembly Algorithm (ADA) [12][13] を開発した。DCA と ADA は閉リンク機構に対応したアルゴリズムである。Anderson らは DCA の閉リンク機構の計算を効率的に行う方法として Orthogonal Complement Based DCA (O-DCA) [40] を開発した。ここで，ADA と O-DCA では標記が異なるが，本質的には同じアルゴリズムである。

山根らは ABA と DCA, CFA, ADA の各計算法を統一的な方法で記述し，それぞれが用いている基礎方程式とそれに起因する計算量の違いをまとめた[41]。その結果，单一CPUで閉リンク機構のシミュレーションを行う場合は ABA が最も高速であるが，閉リンク機構のシミュレーションや並列計算を行う場合は，ADA が最も効率的であることを示した。

### 1.1.4 数値・数式ハイブリッド手法

数値計算は、近似計算であり計算結果の信憑性には保証を必要とするが、計算効率が高い。数式処理の計算は、整数演算をベースに多項式の操作（四則演算）を基本としており正確な計算を行う。ただし、数式処理の計算では、計算過程で現れる式は係数や項数が増大し（中間式拡張）計算効率が悪くなる。そこで、数値計算と数式処理を組み合わせることで数式処理の良いところを残しつつ計算効率を高めることを目指して、数値・数式ハイブリッド手法の研究が進められている[42]。

数値・数式ハイブリッド手法の最も自然な方法として、アルゴリズムの融合とまではいかないが、数式処理の手法と数式処理の手法を組み合わせて使うアプローチがある。特に、数値計算の「前処理」として数式処理を利用して数値計算の精度を向上させる研究である。連立代数方程式系を解く場合などに用いられる。

その他に、数式処理の欠点の一つである計算量の大きさへの対応として数値計算を導入して効率化をする研究[43][44]、誤差を含む入力へ対処する研究[45]、最適化における高品質計算への活用などが行われている[46][47]。

## 1.2 研究目的

本論文では、閉リンク機構を含むリンク系のシミュレーションを効率的に行うために、高速な順動力学エンジンを開発することを目的とする。そのために、まず、既存の順動力学計算法を比較し、最も効率的なアルゴリズムを選定する。そして、数値・数式ハイブリッド手法による順動力学計算の高速化を行い、その有用性を検証する。

この研究の応用例を下記に2点示す。一つ目は、開発した順動力学エンジンを搭載したシミュレータを作成することである。シミュレーション時間が従来よりも短縮され、ロボット開発の効率化に繋がると考えられる。

二つ目は、閉リンク機構を考慮した順動力学計算を制御に応用することである。人型ロボットの物体把持動作や壁に手をついた際など、ロボット自体は開リンク機構として扱えるモデルでも、環境や物体と接触した際に閉リンク機構が現れる。このような状態に対して、仮想的に閉リンク機構の計算を行うことで、ロボットと物体や環境との干渉を考えて制御できるようになると考えられる。

### 1.3 本論文の構成

本論文は以下の章から構成される。

#### 第1章「序論」

本研究の背景、目的などについて述べた。

#### 第2章「順動力学計算法の比較」

Articulated-Body Algorithm(ABA)とDivide-and-Conquer Algorithm(DCA), Assembly-Disassembly Algorithm(ADA)の比較を行い、我々のシミュレータに適用する順動力学計算法を選定する。

#### 第3章「Assembly-Disassembly Algorithm(ADA)」

ADAに基づくリンク系のモデリング手法と三次元浮遊ベースのモーデリング手法を示す。また、ADAに基づく3節開リンク機構と4節閉リンク機構のモーデリングを比較し、シリアルリンク機構から閉リンク機構へ容易に拡張できることを示す。

#### 第4章「数値・数式ハイブリッドモデリング」

ハイブリッドモデリングによるリンク系の運動方程式の導出を行う。また、ハイブリッドモデリングを行ったシミュレーション時間とADAのシミュレーション時間を比較し、ハイブリッドモデリングの有用性を示す。

#### 第5章「マルチボディダイナミクスシミュレータ」

シミュレータのプログラムの流れやファイル構成、シミュレーションの例などを示す。

#### 第6章「シミュレータの実用例」

開発したシミュレータの実用例として、人型ロボットHOAP-2のシミュレータを作成し、バランス制御の検証に用いた例を述べる。

#### 第7章「閉リンク機構を考慮した制御」

ADAの順動力学計算を用いて仮想的な閉リンク機構をモーデリングし、開リンク機構のロボットの制御を行う。

#### 第8章「結論および今後の研究課題」

結論および今後の研究課題について示す。

## 1.4 本論文で用いる記号

本論文で用いる省略記号を Table 1.1 に, 変数名を Tables 1.2, 1.3 に示す.

Table 1.1: The summary of the written abbreviations.

AB	Articulated Body	FD	Forward Dynamics
ABA	Articulated-Body Algorithm	IABI	Inverse Articulated-Body Inertia
ABI	Articulated-Body Inertia	ID	Inverse Dynamics
ADA	Assembly-Disassembly Algorithm	JOR	Joint-Oriented Representation
CC	Closed Chains	O-DCA	Orthogonal Complement Based DCA
CoM	Center of Mass	PC	Parallel Computation
COR	Constraint-Oriented Representation	RNS	Reaction Null-Space
DCA	Divide-and-Conquer Algorithm	ZMP	Zero-Moment Point
DOF	Degree of Freedom		

Table 1.2: The summary of the symbols.

$\mathcal{B}$	$\in \Re^6$	bias acceleration
$\mathcal{C}$	$\in \Re^6$	Coriolis and centrifugal force in spatial coordinates
$\mathcal{F}$	$\in \Re^6$	spatial force
$\mathcal{G}$	$\in \Re^6$	gravity force vector in spatial coordinates
$\mathcal{H}$	$\in \Re^6$	spatial momentum
$\mathcal{P}$	$\in \Re^6$	bias force
$\mathcal{V}$	$\in \Re^6$	spatial velocity
$\mathcal{X}$	$\in \Re^6$	spatial position
$(\circ)^A$		matrix of articulated body $A$
$(\circ)^{des}$		desired variable
$(\circ)^{ex}$		external variable
$(\circ)^{ref}$		reference variable
$i(\circ)$		coordinate frame $i$

Table 1.3: The summary of the symbols.

$\alpha$	$\in \Re^3$	angular acceleration
$\lambda_i$	$\in \Re^{n_J}$	constraint force variable of Joint $i$
$\tau$	$\in \Re^n$	generalized force
$\tau_i$	$\in \Re^{n_J}$	active force variable of Joint $i$
$\Phi$	$\in \Re^{6 \times 6}$	inverse inertia
$\omega$	$\in \Re^3$	angular velocity
$\Omega^\times$	$\in \Re^{3 \times 3}$	skew-symmetric matrix for angular velocity $\omega$
$a$	$\in \Re^3$	linear acceleration
$c$	$\in \Re^n$	Coriolis and centrifugal force
$d$	$\in \Re^3$	position
$D^\times$	$\in \Re^{3 \times 3}$	skew-symmetric matrix for position $d$
$f$	$\in \Re^3$	force
$F$		$F = [\mathcal{F}_1^T, \mathcal{F}_2^T, \dots, \mathcal{F}_i^T]^T$
$F^\times$	$\in \Re^{3 \times 3}$	skew-symmetric matrix for force $f$
$g$	$\in \Re^n$	gravity force vector
$i$		joint number
$I$	$\in \Re^{6 \times 6}$	spatial inertia tensor
$\bar{I}$	$\in \Re^{3 \times 3}$	inertia tensor
$J$		jacobian matrix
$K_p$		feedback gain for position
$K_d$		feedback gain for velocity
$l$	$\in \Re^3$	angular momentum
$M$	$\in \Re^{n \times n}$	inertia tensor
$n$		DOF of whole body
$n_J$		DOF of Joint $i$
$n$	$\in \Re^3$	moment
$\overrightarrow{OP}$	$\in \Re^3$	position vector from O to P
$p$	$\in \Re^3$	linear momentum
$q$	$\in \Re^n$	generalized coordinates
$q_i$	$\in \Re^{n_J}$	position variable of Joint $i$
$q_1, q_2, q_3, q_4$		parameters for unit quaternion
$R$	$\in \Re^{3 \times 3}$	rotation matrix ( ${}_B^A R$ : rotation from $B$ to $A$ )
$S$		$S = [S_1^T, S_2^T, \dots, S_i^T]^T$
$S_i$	$\in \Re^{6 \times n_J}$	motion subspace matrix of Joint $i$
$T_i$	$\in \Re^{6 \times (6-n_J)}$	constraint force subspace matrix of Joint $i$
$T_{ai}$	$\in \Re^{6 \times n_J}$	active force subspace matrix of Joint $i$
$U$		unit matrix
$v$	$\in \Re^3$	linear velocity

## 第2章 順動力学計算法の比較 [4][41]

効率的な順動力学計算法として，シリアル計算用のアルゴリズムや並列計算用のアルゴリズムが知られている。この方法による $n$ 個のリンクからなるロボットに対する計算量は，単一CPUによるシリアル計算で $O(n)$ ， $n$ 個のCPUによる並列計算で $O(\log n)$ である。しかし，論文によってアルゴリズムの記述が異なるため，計算量の比較が困難であった。本章では，Articulated-Body Algorithm (ABA)とDivide-and-Conquer Algorithm (DCA)，Assembly-Disassembly Algorithm (ADA)について，各計算法を統一的な方法で記述する。その際，それぞれのアルゴリズムが用いている基礎方程式と，それに起因する計算量の違いをまとめることにより，各アルゴリズムがどのような場合に有効であるかを考察する。また，その結果を基に，我々のシミュレータに適用する順動力学計算法を選定する。

### 2.1 運動方程式と関節の条件式

#### 2.1.1 Articulated Bodyの運動方程式

本論文では，空間ベクトル[1]を用いて数式を表現する。空間ベクトルでは，角速度と並進の速度やモーメントと並進の力など，回転成分と並進成分を一つに組み合わせた六次元のベクトルを用いて表現する。空間ベクトルの基礎式は付録Aを参照されたい。

Fig. 2.1 (a)に示す一つの剛体 $B$ について考える。加えられた力 $\mathcal{F} \in \Re^6$ と発生する加速度 $\dot{\mathcal{V}} \in \Re^6$ の関係は，剛体の運動方程式から次式で表される。

$$\mathcal{F} = \mathbf{I} \dot{\mathcal{V}} + \mathcal{P} \quad (2.1)$$

ここで，式中の $\mathbf{I} \in \Re^{6 \times 6}$ は $B$ の慣性行列， $\mathcal{P} \in \Re^6$ は $B$ のコリオリ力や外力などの加速度に依存しない力(バイアス力)である。次に，Fig. 2.1 (b)に示すリンク系について考える。二つ目の剛体 $B'$ が関節によって $B$ と繋がっ

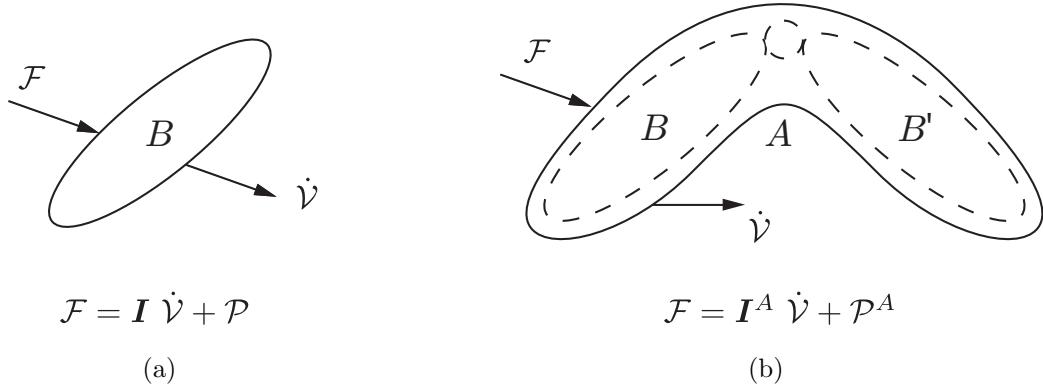


Fig. 2.1: The relationship between the applied force,  $\mathcal{F}$ , and the resulting acceleration,  $\dot{\mathcal{V}}$ : (a) a rigid body and (b) an articulated body.

ている。 $\mathcal{F}$ と $\dot{\gamma}$ の関係は Articulated Body (AB) の運動方程式から次のように表される。

$$\mathcal{F} = \mathbf{I}^A \dot{\mathcal{V}} + \mathcal{P}^A \quad (2.2)$$

ABの運動方程式を用いると、関節で接続された複数のリンクを一つの剛体として考えることができる。式中の  $I^A \in \mathbb{R}^{6 \times 6}$  を Articulated-Body Inertia (ABI),  $\mathcal{P}^A \in \mathbb{R}^6$  を ABのバイアス力と呼ぶ。

$I^A$  は正定値対称なので Eq. (2.2) は次式と等価である .

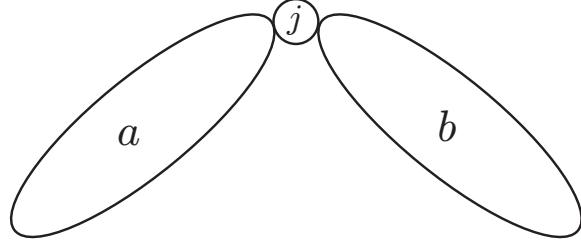
$$\dot{\mathcal{V}} = \Phi^A \mathcal{F} + \mathcal{B}^A \quad (2.3)$$

ただし、 $\Phi^A = (\mathbf{I}^A)^{-1}$ 、 $\mathcal{B}^A = -\Phi^A \mathcal{P}^A$ と置いた。式中の $\Phi^A \in \Re^{6 \times 6}$ をInverse Articulated-Body Inertia ( IABI )、 $\mathcal{B}^A \in \Re^6$ をABのバイアス加速度と呼ぶ。

力を加えられる点をハンドルと呼ぶ。Eq. (2.3) の表現を用いることで、一つの AB に複数のハンドルを設定できる。ハンドル  $i$  に加えられた力を  $F_i$ 、発生する加速度を  $\ddot{v}_i$  とすると、 $h$  個のハンドルの加速度は Eq. (2.3) から次のように表すことができる ( $i = 1 \sim h$ )。

$$\begin{bmatrix} \dot{\mathcal{V}}_1 \\ \dot{\mathcal{V}}_2 \\ \vdots \\ \dot{\mathcal{V}}_h \end{bmatrix} = \begin{bmatrix} \Phi_1^A & \Phi_{12}^A & \dots & \Phi_{1h}^A \\ \Phi_{21}^A & \Phi_2^A & \dots & \Phi_{2h}^A \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{h1}^A & \Phi_{h2}^A & \dots & \Phi_h^A \end{bmatrix} \begin{bmatrix} \mathcal{F}_1 \\ \mathcal{F}_2 \\ \vdots \\ \mathcal{F}_h \end{bmatrix} + \begin{bmatrix} \mathcal{B}_1^A \\ \mathcal{B}_2^A \\ \vdots \\ \mathcal{B}_h^A \end{bmatrix} \quad (2.4)$$

ここで、式中の  $\Phi_i^A$  と  $\beta_i^A$  はハンドル  $i$  の IABI とバイアス加速度、 $\Phi_{ij}^A$  はハンドル  $j$  に加えられた力によるハンドル  $i$  の加速度の関係を表す干渉項である。

Fig. 2.2: Representation of the Joint  $j$ .

### 2.1.2 関節の条件式

Fig. 2.2に示すような，関節 $j$ で接続された二つの剛体 $a, b$ を考える。関節を運動学的拘束として表現すると，速度レベルの拘束は次式で表される。

$$\mathcal{V}_j^b - \mathcal{V}_j^a = \mathbf{S}_j \dot{\mathbf{q}}_j \quad (2.5)$$

ここで， $\mathcal{V}_j^a$  と  $\mathcal{V}_j^b$  は剛体 $a$  と  $b$  の速度， $\mathbf{S}_j \in \Re^{6 \times n_J}$  は関節の可動方向を表す行列， $\dot{\mathbf{q}}_j \in \Re^{n_J}$  は関節の速度， $n_J$  は関節の自由度である。Eq. (2.5) を時間微分すると，加速度レベルの拘束は次式で表される。

$$\ddot{\mathcal{V}}_j^b - \ddot{\mathcal{V}}_j^a = \mathbf{S}_j \ddot{\mathbf{q}}_j + \dot{\mathbf{S}}_j \dot{\mathbf{q}}_j \quad (2.6)$$

ただし， $\dot{\mathbf{S}}_j \dot{\mathbf{q}}_j = \mathcal{V}_j^a \times \mathbf{S}_j = \mathcal{V}_j^b \times \mathbf{S}_j$  である。

関節 $j$ が剛体 $b$ に加える力 $\mathcal{F}_j^b$ は，関節の駆動力と関節の拘束力の和として考えると，次のように表される。

$$\mathcal{F}_j^b = \mathbf{T}_{aj} \boldsymbol{\tau}_j + \mathbf{T}_j \boldsymbol{\lambda}_j \quad (2.7)$$

ここで， $\mathbf{T}_{aj} \in \Re^{6 \times n_J}$  は関節の可動方向を表す行列， $\mathbf{T}_j \in \Re^{6 \times (6-n_J)}$  は関節の拘束方向を表す行列， $\boldsymbol{\tau}_j \in \Re^{n_J}$  は  $\dot{\mathbf{q}}_j$  に対応した関節の駆動力， $\boldsymbol{\lambda}_j \in \Re^{6-n_J}$  は関節の拘束力である。また， $\mathbf{T}_{aj}$  と  $\mathbf{T}_j$  は次の条件を満たす。

$$\begin{aligned} \mathbf{T}_{aj}^T \mathbf{S}_j &= \mathbf{U} \in \Re^{n_J \times n_J} \\ \mathbf{T}_j^T \mathbf{S}_j &= \mathbf{0} \in \Re^{(6-n_J) \times n_J} \end{aligned} \quad (2.8)$$

Eq. (2.6) の両辺に左から  $\mathbf{T}_j^T$  を掛け，Eq. (2.8) の関係を用いると次式を得る。

$$\mathbf{T}_j^T (\ddot{\mathcal{V}}_j^b - \ddot{\mathcal{V}}_j^a) = \mathbf{T}_j^T \dot{\mathbf{S}}_j \dot{\mathbf{q}}_j \quad (2.9)$$

また, Eq. (2.7) の両辺に左から  $S_j^T$  を掛け, Eq. (2.8) の関係を用いると次式を得る .

$$S_j^T \mathcal{F}_j^b = \tau_j \quad (2.10)$$

本論文では, Eqs. (2.7), (2.9) を用いた表現を Constraint-Oriented Representation (COR), Eqs. (2.6), (2.10) を用いた表現を Joint-Oriented Representation (JOR) と呼ぶ .

$S, T_a, T$  に関する各行列成分を Tables 2.1, 2.2 に示す . 関節の種類は回転関節 (R-joint) や直動関節 (P-joint), 球面関節 (S-joint), 固定関節 (fixed-joint), 浮遊関節 (6DOF-joint) である .

Table 2.1: Joint model formula.

Joint type	Motion subspace	Active force subspace	Constraint force subspace
	$S$	$T_a$	$T$
R-joint	$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$
U-joint	$\begin{bmatrix} -s_2 & 0 \\ 0 & 1 \\ c_2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -s_2 & 0 \\ 0 & 1 \\ c_2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$	$\begin{bmatrix} c_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ s_2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
S-joint	$\begin{bmatrix} -a_2 c_3 & -s_3 & 0 \\ s_2 s_3 & -c_3 & 0 \\ c_2 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\frac{1}{s_2} \begin{bmatrix} -c_3 & s_2 s_3 & s_2 s_3 \\ s_3 & s_2 c_3 & -c_2 s_3 \\ 0 & 0 & s_2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Table 2.2: Joint model formula.

Joint type	Motion subspace $S$	Active force subspace $T_a$	Constraint force subspace $T$
fixed-joint	—	—	$U \in \Re^{6 \times 6}$
6DOF-joint	$U \in \Re^{6 \times 6}$	$U \in \Re^{6 \times 6}$	—

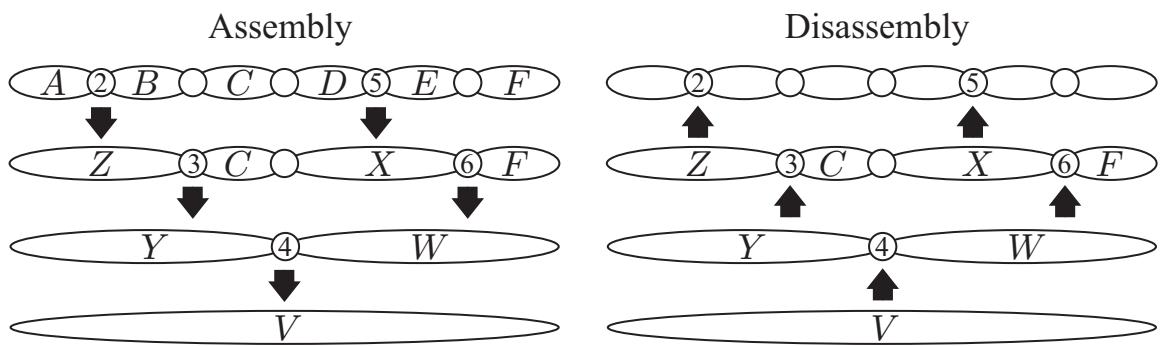


Fig. 2.3: Illustration of the assembly and disassembly phases.

## 2.2 三つのアルゴリズムの概要

### 2.2.1 Assembly-Disassembly Algorithm ( ADA )

ADA は assembly ステップと disassembly ステップの二つの計算ステップからなる。各計算ステップについて、Fig. 2.3 のシリアル 6 リンク機構を例に説明する。

#### Assembly ステップ

1. 各リンクごとの IABI とバイアス加速度を求める。
2. 隣り合う AB を接続し、新たな AB の IABI とバイアス加速度を求める。
3. 「2.」の計算を繰り返し、全体で一つの AB V を作成する。

#### Disassembly ステップ

1. 最後に接続した関節 4 に加わる拘束力を求める。
2. 拘束力と駆動力から関節 4 の両側のハンドルに加わる力を求める。

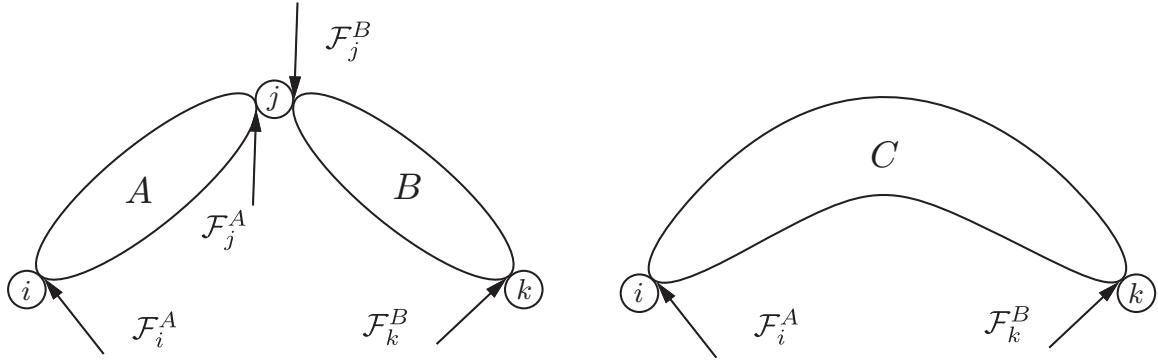


Fig. 2.4: Assembly of articulated bodies for a serial link structure.  $A$ ,  $B$  and  $C$  have two handles, respectively.

3. ハンドルに加わる力から関節4の加速度を求める .
4. ハンドルに加わる力から関節3, 6に加わる拘束力を求める .
5. 「2.」から「4.」の計算を assembly ステップと逆の順序で行う .

この計算法には以下の三つの特徴がある .

- 関節を追加していく順序は任意である . また , 順序を変更することによって数値安定性や並列性を変化させることができる .
- 関節加速度だけでなく , 全関節での拘束力が得られる . これは , 力センサのシミュレーションや接触のモデリングなどに利用可能である .
- 閉リンク機構を含む任意のリンク機構に適用可能である .

### 運動方程式の導出

Fig. 2.4 に示すように , それぞれ二つのハンドル  $A_i$  ,  $A_j$  ,  $B_j$  ,  $B_k$  を持った二つのAB  $A$  ,  $B$  を考える . ハンドル  $i$  ,  $j$  に加わる力  $F_i^A$  ,  $F_j^A$  を考えると , それとの加速度  $\dot{v}_i^A$  ,  $\dot{v}_j^A$  との関係は Eq. (2.4) より以下のように表される .

$$\begin{bmatrix} \dot{v}_i^A \\ \dot{v}_j^A \end{bmatrix} = \begin{bmatrix} \Phi_i^A & \Phi_{ij}^A \\ \Phi_{ji}^A & \Phi_j^A \end{bmatrix} \begin{bmatrix} F_i^A \\ F_j^A \end{bmatrix} + \begin{bmatrix} \mathcal{B}_i^A \\ \mathcal{B}_j^A \end{bmatrix} \quad (2.11)$$

ここで , 式中の  $\Phi^A$  は  $A$  の IABI ,  $\mathcal{B}^A$  は  $A$  のバイアス加速度とする . 同様に , AB  $B$  に対して以下の式が成り立つ .

$$\begin{bmatrix} \dot{v}_j^B \\ \dot{v}_k^B \end{bmatrix} = \begin{bmatrix} \Phi_j^B & \Phi_{jk}^B \\ \Phi_{kj}^B & \Phi_k^B \end{bmatrix} \begin{bmatrix} F_j^B \\ F_k^B \end{bmatrix} + \begin{bmatrix} \mathcal{B}_j^B \\ \mathcal{B}_k^B \end{bmatrix} \quad (2.12)$$

$A_j$  と  $B_j$  で接続した  $ABC$  を考える。 $A_i, B_k$  に加わる力  $\mathcal{F}_i^A, \mathcal{F}_k^B$  と、それとの加速度  $\dot{\mathcal{V}}_i^A, \dot{\mathcal{V}}_k^B$  との関係は以下のように表される。

$$\begin{bmatrix} \dot{\mathcal{V}}_i^A \\ \dot{\mathcal{V}}_k^B \end{bmatrix} = \begin{bmatrix} \Phi_i^C & \Phi_{ik}^C \\ \Phi_{ki}^C & \Phi_k^C \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^A \\ \mathcal{F}_k^B \end{bmatrix} + \begin{bmatrix} \mathcal{B}_i^C \\ \mathcal{B}_k^C \end{bmatrix} \quad (2.13)$$

関節の条件式は COR を用いる。Eqs. (2.9), (2.7) より以下の 2 式が求まる。

$$\mathbf{T}_j^T (\dot{\mathcal{V}}_j^B - \dot{\mathcal{V}}_j^A) = \mathbf{T}_j^T \dot{\mathbf{S}}_j \dot{\mathbf{q}}_j \quad (2.14)$$

$$\mathcal{F}_j^B = \mathbf{T}_{aj} \boldsymbol{\tau}_j + \mathbf{T}_j \boldsymbol{\lambda}_j \quad (2.15)$$

以降では、 $\Phi_i^A$  や  $\Phi_k^B$  などから  $\Phi_i^C$  や  $\Phi_k^C$  などを求める方法を示していく。

Eq. (2.14) に Eqs. (2.11), (2.12) の  $\dot{\mathcal{V}}_j^A, \dot{\mathcal{V}}_j^B$  を代入すると

$$\mathbf{T}_j^T (\Phi_j^B \mathcal{F}_j^B + \Phi_{jk}^B \mathcal{F}_k^B + \mathcal{B}_j^B - \Phi_{ji}^A \mathcal{F}_i^A - \Phi_j^A \mathcal{F}_j^A - \mathcal{B}_j^A) = \mathbf{T}_j^T \dot{\mathbf{S}}_j \dot{\mathbf{q}}_j \quad (2.16)$$

となる。さらに Eq. (2.15) を代入すると

$$\mathbf{T}_j^T (\Phi_j^B + \Phi_j^A) (\mathbf{T}_j \boldsymbol{\lambda}_j + \mathbf{T}_{aj} \boldsymbol{\tau}_j) = \mathbf{T}_j^T (\dot{\mathbf{S}}_j \dot{\mathbf{q}}_j + \Phi_{ji}^A \mathcal{F}_i^A - \Phi_{jk}^B \mathcal{F}_k^B + \mathcal{B}_j^A - \mathcal{B}_j^B) \quad (2.17)$$

となる。ここで、 $\mathcal{F}_j^B = -\mathcal{F}_j^A$  の関係を用いた。 $\Phi_j^B + \Phi_j^A$  は正定値対称であり、拘束力  $\boldsymbol{\lambda}_j$  は次のように求まる。

$$\boldsymbol{\lambda}_j = \mathbf{Y}_j \mathbf{T}_j^T (\Phi_{ji}^A \mathcal{F}_i^A - \Phi_{jk}^B \mathcal{F}_k^B + \boldsymbol{\beta}_j) \quad (2.18)$$

ここで、 $\mathbf{Y}_j = (\mathbf{T}_j^T (\Phi_j^A + \Phi_j^B) \mathbf{T}_j)^{-1}$ 、 $\boldsymbol{\beta}_j = \dot{\mathbf{S}}_j \dot{\mathbf{q}}_j + \mathcal{B}_j^A - \mathcal{B}_j^B - (\Phi_j^A + \Phi_j^B) \mathbf{T}_{aj} \boldsymbol{\tau}_j$  と置いた。最後に、Eqs. (2.15), (2.18) を Eqs. (2.11), (2.12) の  $\mathcal{F}_j^A, \mathcal{F}_j^B$  に代入し、 $\dot{\mathcal{V}}_i^A, \dot{\mathcal{V}}_k^B$  を求めると、次式を得る。

$$\dot{\mathcal{V}}_i^A = \Phi_i^A \mathcal{F}_i^A - \Phi_{ij}^A (\mathbf{T}_{aj} \boldsymbol{\tau}_j + \mathbf{T}_j \mathbf{Y}_j \mathbf{T}_j^T (\Phi_{ji}^A \mathcal{F}_i^A - \Phi_{jk}^B \mathcal{F}_k^B + \boldsymbol{\beta}_j)) + \mathcal{B}_i^A \quad (2.19)$$

$$\dot{\mathcal{V}}_k^B = \Phi_{kj}^B (\mathbf{T}_{aj} \boldsymbol{\tau}_j + \mathbf{T}_j \mathbf{Y}_j \mathbf{T}_j^T (\Phi_{ji}^A \mathcal{F}_i^A - \Phi_{jk}^B \mathcal{F}_k^B + \boldsymbol{\beta}_j)) + \Phi_k^B \mathcal{F}_k^B + \mathcal{B}_k^B \quad (2.20)$$

この 2 式は、 $\mathcal{F}_i^A$  と  $\mathcal{F}_k^B$  のみの関数で  $\dot{\mathcal{V}}_i^A, \dot{\mathcal{V}}_k^B$  を表現している。この式を Eq. (2.13) と比較すると、接続後の IABI について以下の関係があることが分かる。

$$\begin{aligned} \Phi_i^C &= \Phi_i^A - \Phi_{ij}^A \mathbf{W}_j \Phi_{ji}^A & \mathcal{B}_i^C &= \mathcal{B}_i^A - \Phi_{ij}^A \boldsymbol{\gamma}_j \\ \Phi_k^C &= \Phi_k^B - \Phi_{kj}^B \mathbf{W}_j \Phi_{jk}^B & \mathcal{B}_k^C &= \mathcal{B}_k^B + \Phi_{kj}^B \boldsymbol{\gamma}_j \\ \Phi_{ik}^C &= \Phi_{ij}^A \mathbf{W}_j \Phi_{jk}^B = (\Phi_{ki}^C)^T \end{aligned} \quad (2.21)$$

ただし， $\mathbf{W}_j = \mathbf{T}_j \mathbf{Y}_j \mathbf{T}_j^T$ ， $\gamma_j = \mathbf{W}_j \beta_j + \mathbf{T}_{aj} \tau_j$ と置いた。

以上から，Eq. (2.21)を用いて，ABを順に接続して新しいABを作成する計算を繰り返すことにより，それぞれのIABIが計算できる。IABIが計算できれば，Eqs. (2.15)，(2.18)を用いて関節に加わる力 $\mathcal{F}_j^B$ が得られる。これをさらにEqs. (2.11)，(2.12)へ代入して $\dot{\mathcal{V}}_j^A$ ， $\dot{\mathcal{V}}_j^B$ を得る。加速度レベルの関節の拘束式Eq. (2.6)から関節加速度を求める

$$\ddot{\mathbf{q}}_j = \mathbf{T}_{aj}^T \left( \dot{\mathcal{V}}_j^B - \dot{\mathcal{V}}_j^A - \dot{\mathbf{S}}_j \dot{\mathbf{q}}_j \right) \quad (2.22)$$

となる。これにより，求めたハンドルの相対加速度 $\dot{\mathcal{V}}_j^A$ ， $\dot{\mathcal{V}}_j^B$ から関節加速度が求まる。なお，Fig. 2.4においてCのIABIを計算する処理は明らかに独立である。したがって，別々のプロセスで並列に計算することが可能となる。

## 2.2.2 Articulated-Body Algorithm (ABA)

ABAは $n$ リンク系の動力学計算を $O(n)$ の計算量で行った最初のアルゴリズムであり，ABIの考え方もこのとき初めて提案された。そのままで閉リンク機構に適用できない。

### 運動方程式の導出

Fig. 2.5に示すように，リンク $i+1$ より先端側にあるすべてのリンクからなるAB $A_{i+1}$ にリンク $i$ が接続される場合を考える。リンク $i+1$ をハンドルとしたときの $A_{i+1}$ のABI $\mathbf{I}_{i+1}^A$ が既知であるとして，リンク $i$ に力 $\mathcal{F}_i$ を加えたときのリンク $i$ ， $i+1$ の加速度をそれぞれ $\dot{\mathcal{V}}_i$ ， $\dot{\mathcal{V}}_{i+1}$ とする。関節を通して $A_{i+1}$ に力 $\mathcal{F}_{i+1}$ が加わったとすると，ABIの性質より以下の運動方程式が成り立つ。

$$\mathcal{F}_i = \mathbf{I}_i^A \dot{\mathcal{V}}_i + \mathcal{P}_i^A \quad (2.23)$$

$$\mathcal{F}_{i+1} = \mathbf{I}_{i+1}^A \dot{\mathcal{V}}_{i+1} + \mathcal{P}_{i+1}^A \quad (2.24)$$

リンク $i$ 単体に働く力は $\mathcal{F}_i - \mathcal{F}_{i+1}$ であるから，リンク $i$ 単独の運動方程式は以下のようになる。

$$\mathcal{F}_i - \mathcal{F}_{i+1} = \mathbf{I}_i \dot{\mathcal{V}}_i + \mathcal{P}_i \quad (2.25)$$

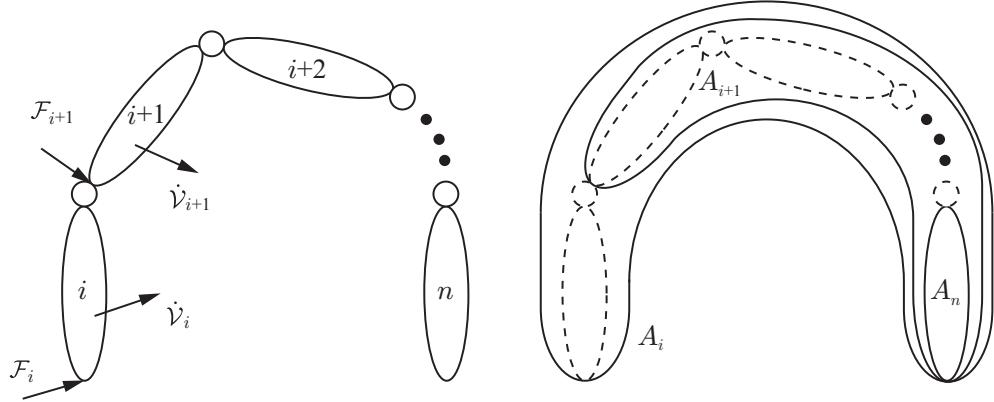


Fig. 2.5: Aspect of the articulated-body algorithm (ABA).

ここで,  $\mathbf{I}_i, \mathcal{P}_i$  はリンク  $i$  単体の質量, 慣性モーメント, 速度依存項から計算できる行列・ベクトルである。関節の表現として JOR を採用すると, Eq. (2.6), (2.10) より以下の式が成り立つ。

$$\dot{\mathcal{V}}_{i+1} - \dot{\mathcal{V}}_i = \mathbf{S}_{i+1} \ddot{\mathbf{q}}_{i+1} + \dot{\mathbf{S}}_{i+1} \dot{\mathbf{q}}_{i+1} \quad (2.26)$$

$$\mathbf{S}_{i+1}^T \mathcal{F}_{i+1} = \boldsymbol{\tau}_{i+1} \quad (2.27)$$

上式に Eq. (2.24) から得られる  $\mathcal{F}_{i+1}$  を代入すると

$$\mathbf{S}_{i+1}^T \left( \mathbf{I}_{i+1}^A \dot{\mathcal{V}}_{i+1} + \mathcal{P}_{i+1}^A \right) = \boldsymbol{\tau}_{i+1} \quad (2.28)$$

となる。さらに Eq. (2.26) から  $\dot{\mathcal{V}}_{i+1}$  の表現を求め, 上式に代入すると

$$\mathbf{S}_{i+1}^T \left( \mathbf{I}_{i+1}^A \left( \dot{\mathcal{V}}_i + \mathbf{S}_{i+1} \ddot{\mathbf{q}}_{i+1} + \dot{\mathbf{S}}_{i+1} \dot{\mathbf{q}}_{i+1} \right) + \mathcal{P}_{i+1}^A \right) = \boldsymbol{\tau}_{i+1} \quad (2.29)$$

となり, 式展開すると  $\ddot{\mathbf{q}}_{i+1}$  が次式のように得られる。

$$\ddot{\mathbf{q}}_{i+1} = \mathbf{W}_{i+1} \left( \boldsymbol{\tau}_{i+1} - \mathbf{S}_{i+1}^T \left( \mathbf{I}_{i+1}^A \left( \dot{\mathcal{V}}_i + \dot{\mathbf{S}}_{i+1} \dot{\mathbf{q}}_{i+1} \right) + \mathcal{P}_{i+1}^A \right) \right) \quad (2.30)$$

ただし  $\mathbf{W}_{i+1} = (\mathbf{S}_{i+1}^T \mathbf{I}_{i+1}^A \mathbf{S}_{i+1})^{-1}$  と置いた。Eq. (2.24) を Eq. (2.25) に代入するとなくなる。

$$\mathcal{F}_i = \left( \mathbf{I}_i^A \left( \dot{\mathcal{V}}_i + \mathbf{S}_{i+1} \ddot{\mathbf{q}}_{i+1} + \dot{\mathbf{S}}_{i+1} \dot{\mathbf{q}}_{i+1} \right) + \mathcal{P}_{i+1}^A \right) = \mathbf{I}_i \dot{\mathcal{V}}_i + \mathcal{P}_i \quad (2.31)$$

さらに (2.30) を代入すると

$$\begin{aligned} \mathcal{F}_i &= \left( \mathbf{I}_i + \mathbf{I}_{i+1}^A - \mathbf{I}_{i+1}^A \mathbf{S}_{i+1} \mathbf{W}_{i+1} \mathbf{S}_{i+1}^T \mathbf{I}_{i+1}^A \right) \dot{\mathcal{V}}_i + \mathcal{P}_i + \mathcal{P}_{i+1}^A + \mathbf{I}_{i+1}^A \dot{\mathbf{S}}_{i+1} \dot{\mathbf{q}}_{i+1} \\ &\quad + \mathbf{I}_{i+1}^A \mathbf{S}_{i+1} \mathbf{W}_{i+1} \left( \boldsymbol{\tau}_{i+1} - \mathbf{S}_{i+1}^T \left( \mathbf{I}_{i+1}^A \dot{\mathbf{S}}_{i+1} \dot{\mathbf{q}}_{i+1} + \mathcal{P}_{i+1}^A \right) \right) \end{aligned} \quad (2.32)$$

となる。この式を Eq. (2.23) と比較すると、リンク  $i$  を追加した後の ABI は以下のようになる。

$$\mathbf{I}_i^A = \mathbf{I}_i + \mathbf{I}_{i+1}^A - \mathbf{I}_{i+1}^A \mathbf{S}_{i+1} \mathbf{W}_{i+1} \mathbf{S}_{i+1}^T \mathbf{I}_{i+1}^A \quad (2.33)$$

$$\begin{aligned} \mathcal{P}_i^A &= \mathcal{P}_i + \mathcal{P}_{i+1}^A + \mathbf{I}_{i+1}^A \dot{\mathbf{S}}_{i+1} \dot{\mathbf{q}}_{i+1} \\ &\quad + \mathbf{I}_{i+1}^A \mathbf{S}_{i+1} \mathbf{W}_{i+1} \left( \boldsymbol{\tau}_{i+1} - \mathbf{S}_{i+1}^T \left( \mathbf{I}_{i+1}^A \dot{\mathbf{S}}_{i+1} \dot{\mathbf{q}}_{i+1} + \mathcal{P}_{i+1}^A \right) \right) \end{aligned} \quad (2.34)$$

以上から、開リンク機構のエンドリンクからベースリンクへ向かう計算より、ABI とバイアス力が順に計算される。

Eq. (2.30) が示すように、 $\mathbf{I}_{i+1}^A$  とリンク  $i$  の加速度  $\ddot{\mathbf{v}}_i$  がわかれば関節  $i+1$  の加速度がわかる。したがってベースリンク 0 が固定された機構であれば、 $\dot{\mathbf{v}}_0 = 0$  を初期条件としてエンドリンクへ向かう再帰計算を行うことにより全関節の加速度が計算できる。

### 2.2.3 Divide-and-Conquer Algorithm (DCA)

DCA は ABA を閉リンク機構や並列計算に拡張したものである。開リンク機構と簡単な閉リンク機構に対する計算量は、ADA と同様に、シリアル計算において  $O(n)$ 、 $n$  個の CPU による並列計算において  $O(\log n)$  である。

#### 運動方程式の導出

DCA は ADA と同様のステップで計算を行う。ADA と比較するために Fig. 2.4 と同じ状況を考え、Eqs. (2.11)–(2.13) を用いる。接続した関節の条件として JOR を採用すると、Eqs. (2.6), (2.10) より、以下の式が成り立つ。

$$\dot{\mathbf{v}}_j^B - \dot{\mathbf{v}}_j^A = \mathbf{S}_j \ddot{\mathbf{q}}_j + \dot{\mathbf{S}}_j \dot{\mathbf{q}}_j \quad (2.35)$$

$$\mathbf{S}_j^T \mathcal{F}_j^B = \boldsymbol{\tau}_j \quad (2.36)$$

Eqs. (2.11), (2.12) を Eq. (2.35) に代入すると次式が得られる。

$$\Phi_j^B \mathcal{F}_j^B + \Phi_{jk}^B \mathcal{F}_k^B + \mathcal{B}_j^B - \Phi_{ji}^A \mathcal{F}_i^A - \Phi_j^A \mathcal{F}_j^A - \mathcal{B}_j^A = \mathbf{S}_j \ddot{\mathbf{q}}_j + \dot{\mathbf{S}}_j \dot{\mathbf{q}}_j \quad (2.37)$$

$\mathcal{F}_j^A = -\mathcal{F}_j^B$  の関係を用いて上式を  $\mathcal{F}_j^B$  について解くと次のようになる。

$$\mathcal{F}_j^B = \mathbf{Y}_j (\Phi_{ji}^A \mathcal{F}_i^A - \Phi_{jk}^B \mathcal{F}_k^B + \beta_j + \mathbf{S}_j \ddot{\mathbf{q}}_j) \quad (2.38)$$

ここで  $\mathbf{Y}_j = (\Phi_j^B + \Phi_j^A)^{-1}$ ,  $\beta_j = \mathcal{B}_j^A - \mathcal{B}_j^B + \dot{\mathcal{S}}_j \dot{\mathbf{q}}_j$  とおいた。これを Eq. (2.36) に代入すると

$$\mathbf{S}_j^T \mathbf{Y}_j (\Phi_{ji}^A \mathcal{F}_i^A - \Phi_{jk}^B \mathcal{F}_k^B + \beta_j + \mathbf{S}_j \ddot{\mathbf{q}}_j) = \tau_j \quad (2.39)$$

となり、次のように関節加速度  $\ddot{\mathbf{q}}_j$  が得られる。

$$\ddot{\mathbf{q}}_j = (\mathbf{S}_j^T \mathbf{Y}_j \mathbf{S}_j)^{-1} (\tau_j - \mathbf{S}_j^T \mathbf{Y}_j (\Phi_{ji}^A \mathcal{F}_i^A - \Phi_{jk}^B \mathcal{F}_k^B + \beta_j)) \quad (2.40)$$

これをさらに Eq. (2.38) に代入すると

$$\mathcal{F}_j^B = \mathbf{W}_j (\Phi_{ji}^A \mathcal{F}_i^A - \Phi_{jk}^B \mathcal{F}_k^B) + \gamma_j \quad (2.41)$$

となる。ここで、

$$\mathbf{W}_j = \mathbf{Y}_j - \mathbf{Y}_j \mathbf{S}_j (\mathbf{S}_j^T \mathbf{Y}_j \mathbf{S}_j)^{-1} \mathbf{S}_j^T \mathbf{Y}_j \quad (2.42)$$

$$\gamma_j = \mathbf{W}_j \beta_j + \mathbf{Y}_j \mathbf{S}_j (\mathbf{S}_j^T \mathbf{Y}_j \mathbf{S}_j)^{-1} \tau_j \quad (2.43)$$

とおいた。Eq. (2.41) を Eqs. (2.11), (2.12) に代入して整理すると次式を得る。

$$\dot{\mathcal{V}}_j^A = (\Phi_j^A - \Phi_{ij}^A \mathbf{W}_j \Phi_{ji}^A) \mathcal{F}_i^A + (\Phi_{ij}^A \mathbf{W}_j \Phi_{jk}^B) \mathcal{F}_j^B + \mathcal{B}_j^A - \Phi_{ij}^A \gamma_j \quad (2.44)$$

$$\dot{\mathcal{V}}_j^B = (\Phi_{kj}^B \mathbf{W}_j \Phi_{ji}^A) \mathcal{F}_i^A + (\Phi_j^B - \Phi_{kj}^B \mathbf{W}_j \Phi_{jk}^B) \mathcal{F}_j^B + \mathcal{B}_j^B + \Phi_{kj}^B \gamma_j \quad (2.45)$$

この二式は、 $\mathcal{F}_i^A$  と  $\mathcal{F}_k^B$  のみの関数で  $\dot{\mathcal{V}}_i^A$ ,  $\dot{\mathcal{V}}_k^B$  を表現している。この式を Eq. (2.13) と比較すると、接続後の IABI について以下の関係があることが分かる。

$$\begin{aligned} \Phi_i^C &= \Phi_i^A - \Phi_{ij}^A \mathbf{W}_j \Phi_{ji}^A & \mathcal{B}_i^C &= \mathcal{B}_i^A - \Phi_{ij}^A \gamma_j \\ \Phi_k^C &= \Phi_k^B - \Phi_{kj}^B \mathbf{W}_j \Phi_{jk}^B & \mathcal{B}_k^C &= \mathcal{B}_k^B + \Phi_{kj}^B \gamma_j \\ \Phi_{ik}^C &= \Phi_{ij}^A \mathbf{W}_j \Phi_{jk}^B = (\Phi_{ki}^C)^T \end{aligned} \quad (2.46)$$

以上から、Eq. (2.46) を用いて、AB を順に接続して新しいABを作成する計算を繰り返すことにより、それぞれの IABI が計算できる。IABI が計算できれば、Eq. (2.40) を用いて関節加速度が計算できる。なお、Fig. 2.4において C の IABI を計算する処理は明らかに独立である。したがって、別々のプロセスで並列に計算することが可能となる。

Table 2.3: Comparison of the four algorithms: selection of articulated-body inertia, joint constraint representation, applicability to closed chains (CC), applicability to parallel computation (PC), computation of joint constraint forces  $\lambda$ , size of inverse matrix (Inverse) and complexity .

Algorithm	Inertia	Joint	CC	PC	$\lambda$	Inverse	Complexity
Lagrange	—	—	△	×	△	—	$O(n^3)$
ABA	ABI	JOR	×	branch only	×	$n_J \times n_J$	$O(n)$
DCA	IABI	COR	○	○	×	$6 \times 6, n_J \times n_J$	$O(n)$ or $O(\log n)$
ADA	IABI	COR	○	○	○	$(6 - n_J) \times (6 - n_J)$	$O(n)$ or $O(\log n)$

## 2.3 アルゴリズムの比較

ADA と ABA , DCA の三つのアルゴリズムが使っている慣性行列と関節の条件式 , およびそれぞれの適用範囲をまとめると Table 2.3 のようになる . また , 一般的な順動力学計算の例として Lagrange 法の特徴も同表に示す . 表中の Inertia は慣性行列の種類 , Joint は関節の条件式の種類 , Closed Chains (CC) は閉リンク機構の計算ができるか , Parallel Computation (PC) は並列計算ができるか , 入は関節の拘束力が計算できるかを示している . また , Inverse は関節の自由度を  $n_J$  としたときの逆行列の大きさ , Complexity はロボットの全自由度を  $n$  としたときの計算量を示している . このように , 使用している式の組が異なっており , それが適用可能な機構や計算量の違いとなって現れる .

Lagrange 法は Lagrange の未定乗数法を用いることで , 閉リンク機構の計算や , 関節の拘束力の算出が行える . しかし , 計算量が  $O(n^3)$  であるため , 自由度の大きなモデルには適さない . 以下では , 他の三つのアルゴリズムについて述べていく .

ABA は慣性行列として ABI を用いている . ABI は各 AB に対してハンドルを一つしか設定ができないため , 閉リンク機構を一つのプロセスで計算することしかできない . しかし ,  $n_J$  自由度の関節一つあたり  $n_J \times n_J$  行列の逆行列を求めればよいため , シリアル計算における計算量は三つの中で最も少ない . ADA と DCA は慣性行列として IABI を用いているため , 各 AB に対して任意の数のハンドルを設定でき , 閉リンク機構や並列計算にも対応できる最も汎用的な方法である . ただし , 複雑な閉リンク機構では計算複雑性が増大する可能性がある . また , 一般的に逆行列の計算

が計算の大部分を占めるため、逆行列の小さい方が計算量も少なくなる。ADAはDCAよりも逆行列が小さい。さらに、関節の自由度が大きくなるにつれて、逆行列の大きさが小さくなることや、追加の計算なしに全関節の拘束力が得られるという特徴がある。

本研究では、閉リンク機構に適用でき、計算量の最も少ないADAを用いて順動力学計算を行う。

## 2.4 本章のまとめ

ABAとDCA、ADAの三つの順動力学計算アルゴリズムを統一的な表記で導出し、比較した。その結果、単一CPUで開リンク機構のシミュレーションを行う場合はABAが最も高速であるが、閉リンク機構のシミュレーションや並列計算を行う場合はADAが最も効率的と考えられることを示した。本論文では、閉リンク機構のシミュレーションを効率的に行うために、ADAに基づく順動力学エンジンを開発し、シミュレータに適用していく。



# 第3章 Assembly-Disassembly Algorithm ( ADA )

ADA は閉リンク機構を含むリンク系のシミュレーションを効率的に行うことができる順動力学計算手法である。ロボットなどのリンク系の計算を行う際には、シリアルリンク機構とブランチリンク機構、閉リンク機構の三種類のリンク機構のモデリングが必要となる。また、宇宙ロボットや人型ロボットのように、ベース部が地面に固定されていないロボットをモデリングする際には、三次元の浮遊ベースのモデリングが必要となる。本章では、ADAに基づくリンク系のモデリング手法と三次元浮遊ベースのモデリング手法を示す。また、ADAに基づく3節開リンク機構と4節閉リンク機構のモデリングを比較し、シリアルリンク機構から閉リンク機構へ容易に拡張できることを示す。

## 3.1 リンク系のモデリング

### 3.1.1 シリアルリンク機構

シリアルリンク機構の解法は第2章のADAの運動方程式の導出で述べたように、Eqs. (2.11)–(2.22) のように求まる。

#### シミュレーション

シリアルリンク機構の解法例としてFig. 3.1に示す平面3リンクマニピュレータのシミュレーションを行った。ここで、関節はすべて $y$ 軸回りの回転関節とする。各リンクの長さは $l = 1.0\text{ m}$ 、質量は $m = 1.0\text{ kg}$ 、重心回りの慣性テンソルは $\bar{I}_C = \text{diag}(ml^2/12, ml^2/12, 0)\text{ kg}\cdot\text{m}$ とした。また、重心位置は各リンクの中心に設定した。 $\theta_1^{ini} = 135\text{ deg}$ ,  $\theta_2^{ini} = -45\text{ deg}$ ,  $\theta_3^{ini} = 225\text{ deg}$ の初期角度(Fig. 3.1)から初速および制御入力なしの条件で、自由落下す

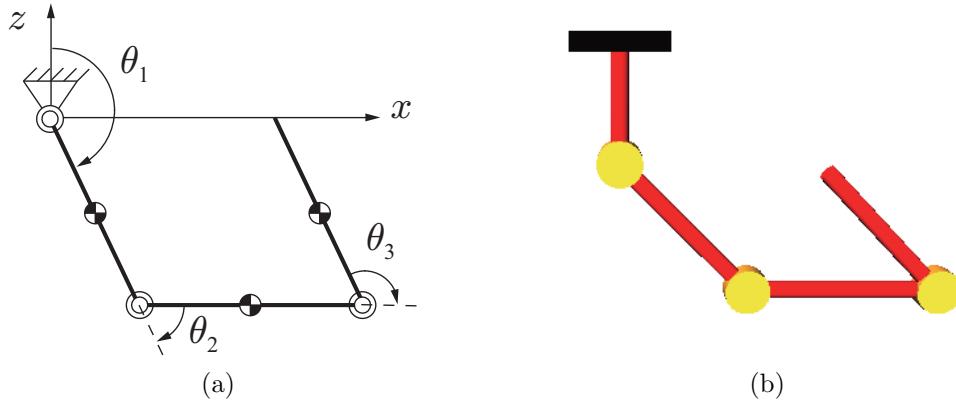


Fig. 3.1: A planar 3R serial chain structure.

るシミュレーションを行った。シミュレーション時間は3 s、積分の刻み幅を1 ms、重力加速度を $z$ 軸方向に $-9.81 \text{ m/s}^2$ とした。

運動の様子をFig. 3.2にスナップショットで示す。また、シミュレーション結果をFig. 3.3に示す。Fig. 3.3 (a)は関節角度、(b)は関節速度、(c)は関節加速度、(d)は関節トルク、(e)は力学的エネルギーである。Fig. 3.2, Figs. 3.3 (a)–(c)より、設定した初期状態から重力の影響を受け、自由落下する様子が確認できる。また、Figs. 3.3 (d), (e)より、制御入力を与えない状態で、位置エネルギー(P.E.)と運動エネルギー(K.E.)の総和が一定となり、力学的エネルギー保存則が確認できた。よって、正しく順動力学計算が行えていると判断できる。

### 3.1.2 ブランチリンク機構

プランチリンク機構とシリアルリンク機構は各 AB のハンドルの数が異なる。シリアルリンク機構は一つの AB に対して二つのハンドルを持つ。一方、プランチリンク機構は、一つの AB に対して三つ以上のハンドルを持つ。

Fig. 3.4 に示すように、それぞれ三つのハンドル  $A_i, A_j, A_k, B_k, B_l, B_m$  を持った二つのAB  $D, E$ を考える。ハンドル  $i, j, k$  に加わる力を  $\mathcal{F}_i^D, \mathcal{F}_j^D, \mathcal{F}_k^D$  を考えると、それぞれの加速度  $\dot{\nu}_i^D, \dot{\nu}_j^D, \dot{\nu}_k^D$  との関係は Eq. (2.4) より以

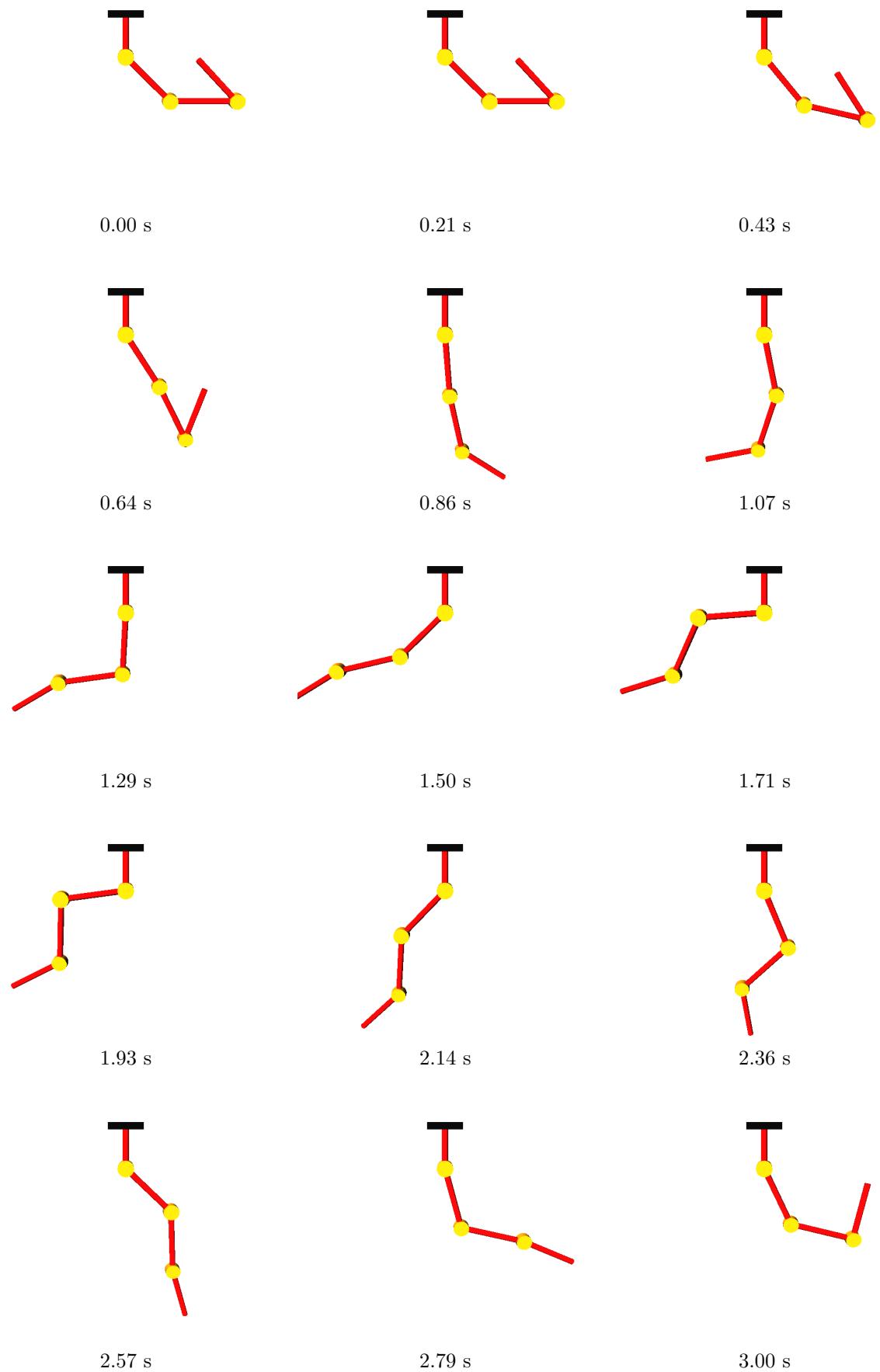


Fig. 3.2: Snapshots from simulation of the planar 3R serial chain structure.

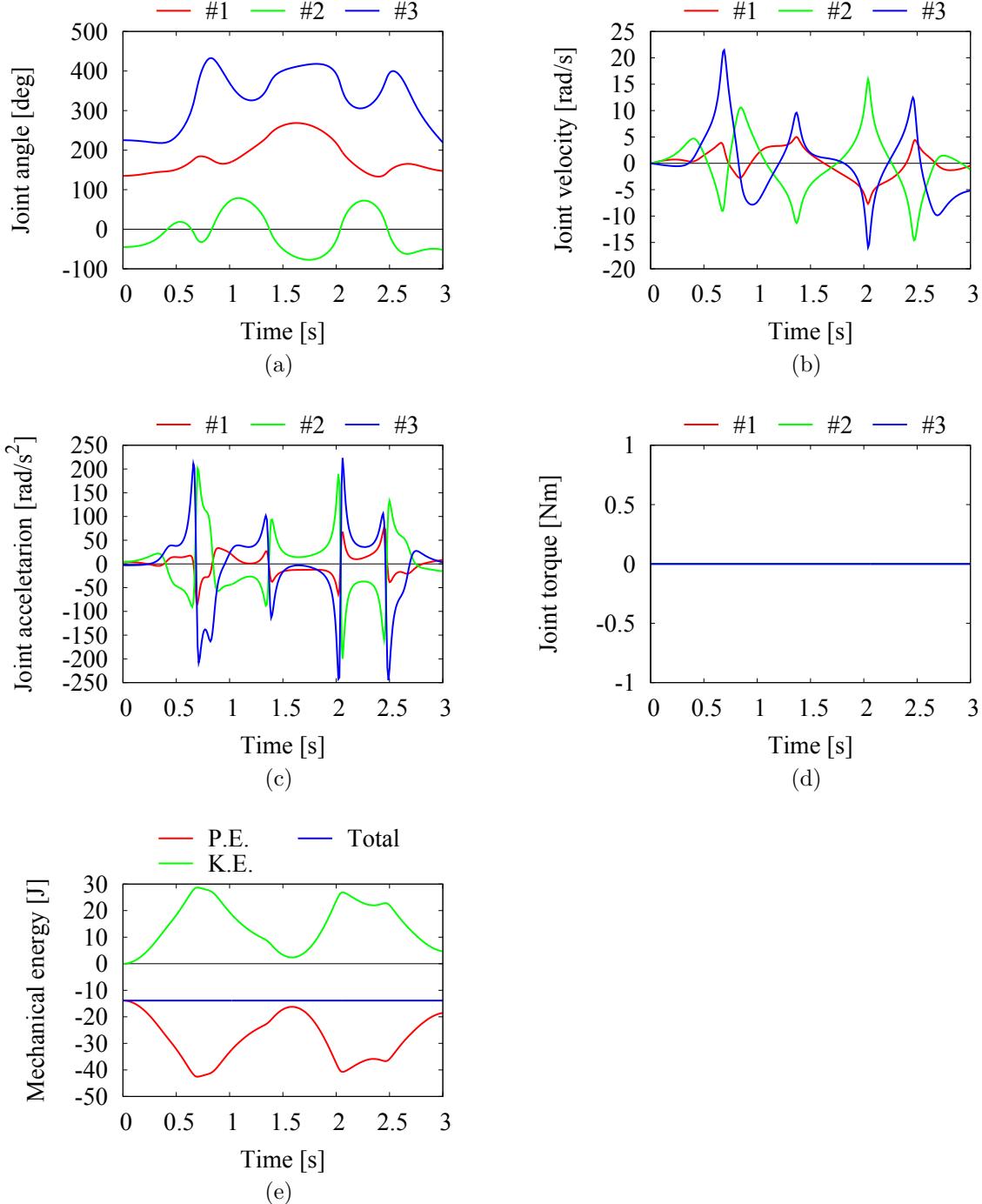


Fig. 3.3: Simulation results of the planar 3R serial chain structure: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy.

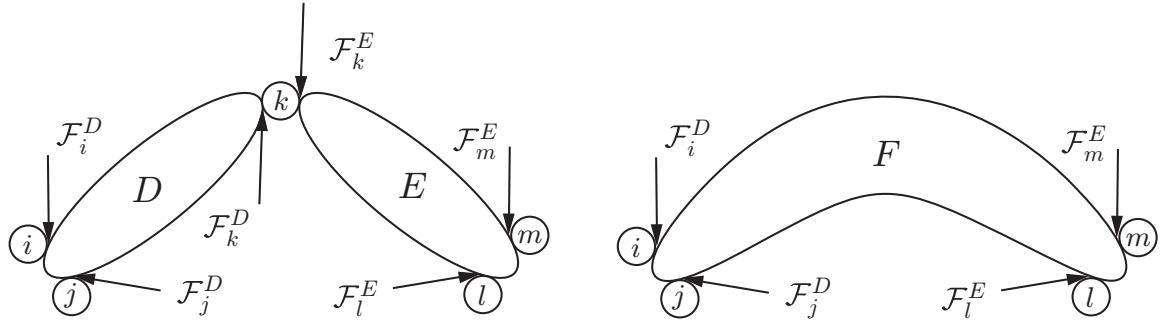


Fig. 3.4: Assembly of articulated bodies for a branch link structure.  $D$  and  $E$  have three handles,  $F$  has four handles, respectively.

下のように表される。

$$\begin{bmatrix} \dot{\mathcal{V}}_i^D \\ \dot{\mathcal{V}}_j^D \\ \dot{\mathcal{V}}_k^D \end{bmatrix} = \begin{bmatrix} \Phi_i^D & \Phi_{ij}^D & \Phi_{ik}^D \\ \Phi_{ji}^D & \Phi_j^D & \Phi_{jk}^D \\ \Phi_{ki}^D & \Phi_{kj}^D & \Phi_k^D \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^D \\ \mathcal{F}_j^D \\ \mathcal{F}_k^D \end{bmatrix} + \begin{bmatrix} \mathcal{B}_i^D \\ \mathcal{B}_j^D \\ \mathcal{B}_k^D \end{bmatrix} \quad (3.1)$$

同様に、AB  $E$  に対して以下の式が成り立つ。

$$\begin{bmatrix} \dot{\mathcal{V}}_k^E \\ \dot{\mathcal{V}}_l^E \\ \dot{\mathcal{V}}_m^E \end{bmatrix} = \begin{bmatrix} \Phi_k^E & \Phi_{kl}^E & \Phi_{km}^E \\ \Phi_{lk}^E & \Phi_l^E & \Phi_{lm}^E \\ \Phi_{mk}^E & \Phi_{ml}^E & \Phi_m^E \end{bmatrix} \begin{bmatrix} \mathcal{F}_k^E \\ \mathcal{F}_l^E \\ \mathcal{F}_m^E \end{bmatrix} + \begin{bmatrix} \mathcal{B}_k^E \\ \mathcal{B}_l^E \\ \mathcal{B}_m^E \end{bmatrix} \quad (3.2)$$

$D_k$  と  $E_k$  で接続した AB  $F$  を考える。 $D_i, D_j, D_l, E_m$  に加わる力  $\mathcal{F}_i^D, \mathcal{F}_j^D, \mathcal{F}_l^D, \mathcal{F}_m^E$  と、それぞれの加速度  $\dot{\mathcal{V}}_i^D, \dot{\mathcal{V}}_j^D, \dot{\mathcal{V}}_l^E, \dot{\mathcal{V}}_m^E$  との関係は以下のように表される。

$$\begin{bmatrix} \dot{\mathcal{V}}_i^D \\ \dot{\mathcal{V}}_j^D \\ \dot{\mathcal{V}}_l^E \\ \dot{\mathcal{V}}_m^E \end{bmatrix} = \begin{bmatrix} \Phi_i^F & \Phi_{ij}^F & \Phi_{il}^F & \Phi_{im}^F \\ \Phi_{ji}^F & \Phi_j^F & \Phi_{jl}^F & \Phi_{jm}^F \\ \Phi_{li}^F & \Phi_{lj}^F & \Phi_l^F & \Phi_{lm}^F \\ \Phi_{mi}^F & \Phi_{mj}^F & \Phi_{ml}^F & \Phi_m^F \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^D \\ \mathcal{F}_j^D \\ \mathcal{F}_l^E \\ \mathcal{F}_m^E \end{bmatrix} + \begin{bmatrix} \mathcal{B}_i^F \\ \mathcal{B}_j^F \\ \mathcal{B}_l^F \\ \mathcal{B}_m^F \end{bmatrix} \quad (3.3)$$

シリアルリンク機構と同様に、 $\Phi_i^D$  や  $\Phi_l^E$  などから  $\Phi_i^F$  や  $\Phi_l^F$  などを求める。

関節  $k$  をハンドル  $D_k$  と  $E_k$  の運動学的拘束として表現する。加速度レベルの拘束は次式で表される。

$$\dot{\mathcal{V}}_k^E - \dot{\mathcal{V}}_k^D = S_k \ddot{\mathbf{q}}_k + \dot{S}_k \dot{\mathbf{q}}_k \quad (3.4)$$

関節  $k$  がハンドル  $E_k$  に加える力  $\mathcal{F}_k^E$  は、関節の駆動力と関節の拘束力の和として考えると、次のように表される。

$$\mathcal{F}_k^E = T_{ak} \tau_k + T_k \lambda_k \quad (3.5)$$

拘束力  $\lambda_j$  は Eq. (2.18) と同様に次のように求まる .

$$\boldsymbol{\lambda}_k = \mathbf{Y}_k \mathbf{T}_k^T \left( \begin{bmatrix} \Phi_{ki}^D & \Phi_{kj}^D \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^D \\ \mathcal{F}_j^D \end{bmatrix} - \begin{bmatrix} \Phi_{kl}^E & \Phi_{km}^E \end{bmatrix} \begin{bmatrix} \mathcal{F}_l^E \\ \mathcal{F}_m^E \end{bmatrix} + \boldsymbol{\beta}_k \right) \quad (3.6)$$

ここで ,  $\mathbf{Y}_k = (\mathbf{T}_k^T (\Phi_k^D + \Phi_k^E) \mathbf{T}_k)^{-1}$  ,  $\boldsymbol{\beta}_k = \dot{\mathbf{S}}_k \dot{\mathbf{q}}_k + \mathcal{B}_k^D - \mathcal{B}_k^E - (\Phi_k^D + \Phi_k^E) \mathbf{T}_{ak} \boldsymbol{\tau}_k$  と置いた . Eqs. (3.5) , (3.6) を Eqs. (3.1) , (3.2) の  $\mathcal{F}_k^D$  ,  $\mathcal{F}_k^E$  に代入し ,  $\dot{\mathcal{V}}_i^D$  ,  $\dot{\mathcal{V}}_j^D$  ,  $\dot{\mathcal{V}}_l^E$  ,  $\dot{\mathcal{V}}_m^E$  を求め , Eq. (3.3) と比較すると , 接続後の IABI について以下の関係があることが分かる .

$$\begin{aligned} \begin{bmatrix} \Phi_i^F & \Phi_{ij}^F \\ \Phi_{ji}^F & \Phi_j^F \end{bmatrix} &= \begin{bmatrix} \Phi_i^D & \Phi_{ij}^D \\ \Phi_{ji}^D & \Phi_j^D \end{bmatrix} - \begin{bmatrix} \Phi_{ik}^D \\ \Phi_{jk}^D \end{bmatrix} \mathbf{W}_k \begin{bmatrix} \Phi_{ki}^D & \Phi_{kj}^D \end{bmatrix} \\ \begin{bmatrix} \Phi_l^F & \Phi_{lm}^F \\ \Phi_{ml}^F & \Phi_m^F \end{bmatrix} &= \begin{bmatrix} \Phi_l^E & \Phi_{lm}^E \\ \Phi_{ml}^E & \Phi_m^E \end{bmatrix} - \begin{bmatrix} \Phi_{lk}^E \\ \Phi_{mk}^E \end{bmatrix} \mathbf{W}_k \begin{bmatrix} \Phi_{kl}^E & \Phi_{km}^E \end{bmatrix} \\ \begin{bmatrix} \Phi_{il}^F & \Phi_{im}^F \\ \Phi_{jl}^F & \Phi_{jm}^F \end{bmatrix} &= \begin{bmatrix} \Phi_{ik}^D \\ \Phi_{jk}^D \end{bmatrix} \mathbf{W}_k \begin{bmatrix} \Phi_{kl}^E & \Phi_{km}^E \end{bmatrix} = \left( \begin{bmatrix} \Phi_{li}^F & \Phi_{lj}^F \\ \Phi_{mi}^F & \Phi_{mj}^F \end{bmatrix} \right)^T \\ \begin{bmatrix} \mathcal{B}_i^F \\ \mathcal{B}_j^F \end{bmatrix} &= \begin{bmatrix} \mathcal{B}_i^D \\ \mathcal{B}_j^D \end{bmatrix} - \begin{bmatrix} \Phi_{ik}^D \\ \Phi_{jk}^D \end{bmatrix} \boldsymbol{\gamma}_k \\ \begin{bmatrix} \mathcal{B}_l^F \\ \mathcal{B}_m^F \end{bmatrix} &= \begin{bmatrix} \mathcal{B}_l^E \\ \mathcal{B}_m^E \end{bmatrix} + \begin{bmatrix} \Phi_{lk}^E \\ \Phi_{mk}^E \end{bmatrix} \boldsymbol{\gamma}_k \end{aligned} \quad (3.7)$$

ただし ,  $\mathbf{W}_k = \mathbf{T}_k \mathbf{Y}_k \mathbf{T}_k^T$  ,  $\boldsymbol{\gamma}_k = \mathbf{W}_k \boldsymbol{\beta}_k + \mathbf{T}_{ak} \boldsymbol{\tau}_k$  と置いた .

Eq. (3.7) を用いて , AB を順に接続して新しいABを作成する計算を繰り返すことにより , それぞれの IABI が計算できる . IABI が計算できれば , Eqs. (3.5) , (3.6) を用いて関節に加わる力  $\mathcal{F}_j^E$  ,  $\mathcal{F}_k^E$  が得られる . これをさらに Eqs. (3.1) , (3.2) へ代入して  $\dot{\mathcal{V}}_k^D$  ,  $\dot{\mathcal{V}}_k^E$  を得る . 加速度レベルの関節の拘束式 Eq. (3.4) から関節加速度を求める

$$\ddot{\mathbf{q}}_k = \mathbf{T}_{ak}^T \left( \dot{\mathcal{V}}_k^E - \dot{\mathcal{V}}_k^D - \dot{\mathbf{S}}_k \dot{\mathbf{q}}_k \right) \quad (3.8)$$

となる . これにより , 求めたハンドルの相対加速度  $\dot{\mathcal{V}}_k^D$  ,  $\dot{\mathcal{V}}_k^E$  から関節加速度が求まる .

## シミュレーション

プランチリンク機構の解法例として Fig. 3.5 に示す平面 4 リンクマニピュレータのシミュレーションを行った . ここで , 関節はすべて  $y$  軸回りの回

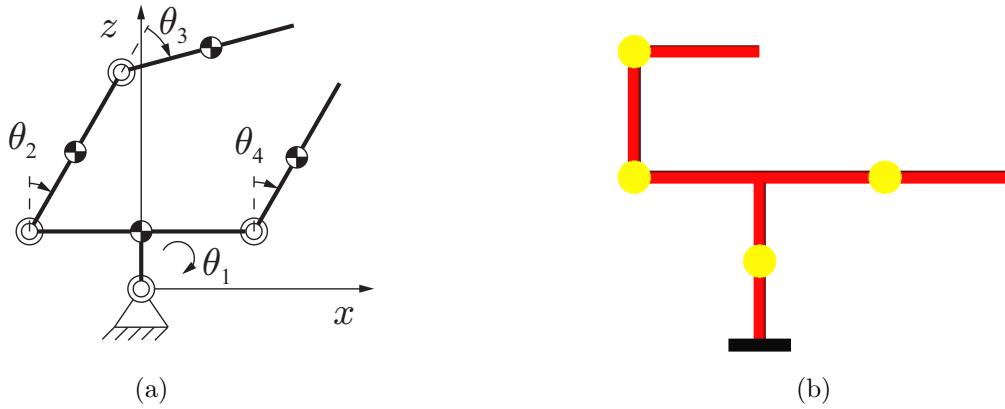


Fig. 3.5: A planar 5R branch chain structure.

転関節とする。各リンクの長さは  $l = 1.0 \text{ m}$ , 質量は  $m = 1.0 \text{ kg}$ , 重心回りの慣性テンソルは  $\bar{I}_C = \text{diag}(ml^2/12, ml^2/12, 0) \text{ kg} \cdot \text{m}$  とした。また、重心位置は各リンクの中心に設定した。 $\theta_1^{ini} = 0 \text{ deg}$ ,  $\theta_2^{ini} = 0 \text{ deg}$ ,  $\theta_3^{ini} = 90 \text{ deg}$ ,  $\theta_4^{ini} = 90 \text{ deg}$  の初期角度 (Fig. 3.5) から初速および制御入力なしの条件で、自由落下するシミュレーションを行った。シミュレーション時間を 3 s, 積分の刻み幅を 1 ms, 重力加速度を  $z$  軸方向に  $-9.81 \text{ m/s}^2$  とした。

運動の様子を Fig. 3.6 にスナップショットで示す。また、シミュレーション結果を Fig. 3.7 に示す。Fig. 3.7 (a) は関節角度, (b) は関節速度, (c) は関節加速度, (d) は関節トルク, (e) は力学的エネルギーである。Fig. 3.6, Figs. 3.7 (a)–(c) より、設定した初期状態から重力の影響を受け、自由落下する様子が確認できる。また、Figs. 3.7 (d), (e) より、制御入力を与えない状態で、位置エネルギー (P.E.) と運動エネルギー (K.E.) の総和が一定となり、力学的エネルギー保存則が確認できた。よって、正しく順動力学計算が行えていると判断できる。

### 3.1.3 閉リンク機構

閉リンク機構とシリアルリンク機構は各 AB のハンドルの数と、一度に接続する関節の数が異なる。シリアルリンク機構は一つの AB に対して二つのハンドルを持つ。一方、閉リンク機構は、ブランチリンク機構と同様に一つの AB に対して三つ以上のハンドルを持つ。また、シリアルリンク機構は一度に接続する関節の数は一つであるが、閉リンク機構は一度に接続する関節数は二つ以上である。

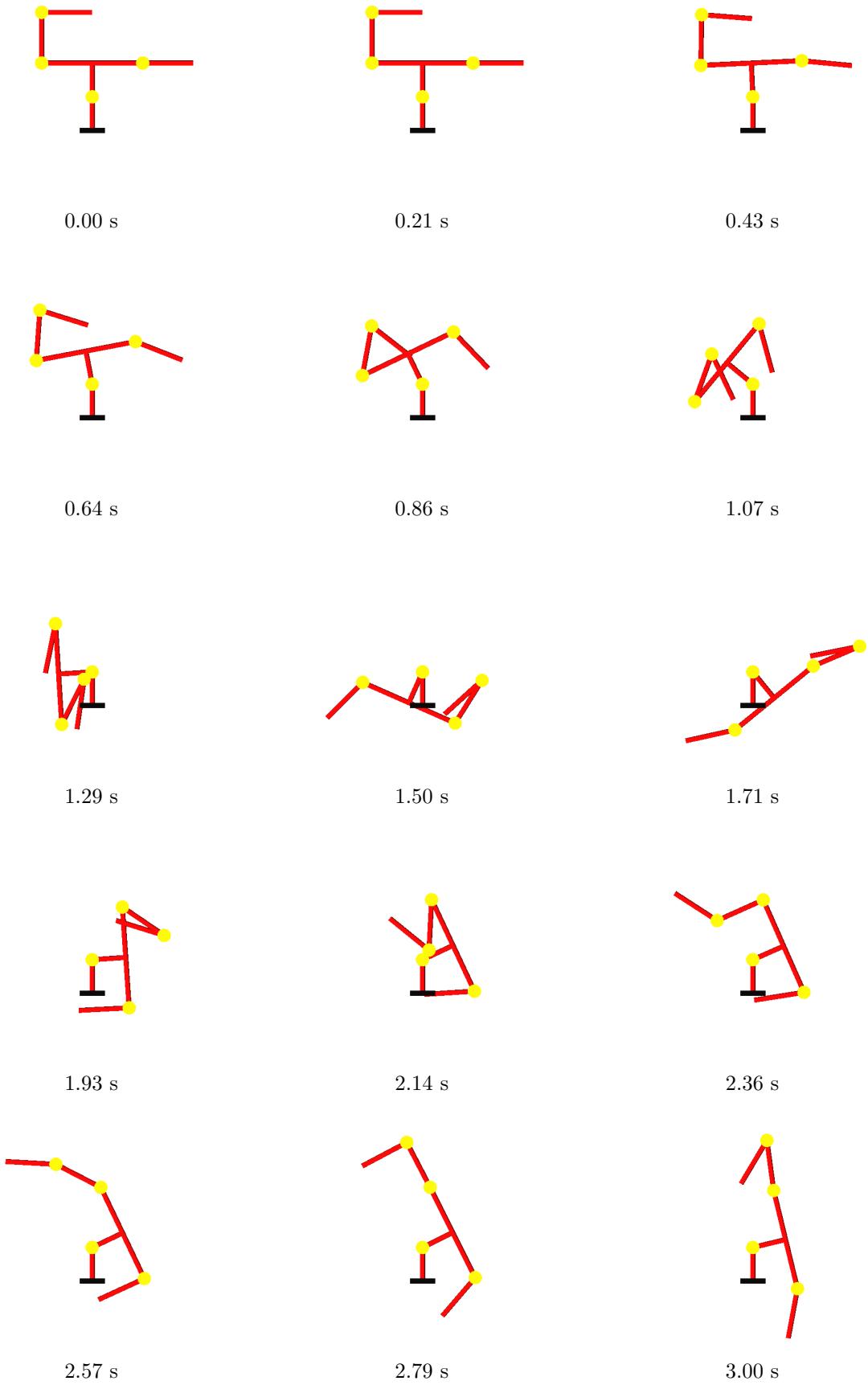


Fig. 3.6: Snapshots from simulation of the planar 5R branch chain structure.

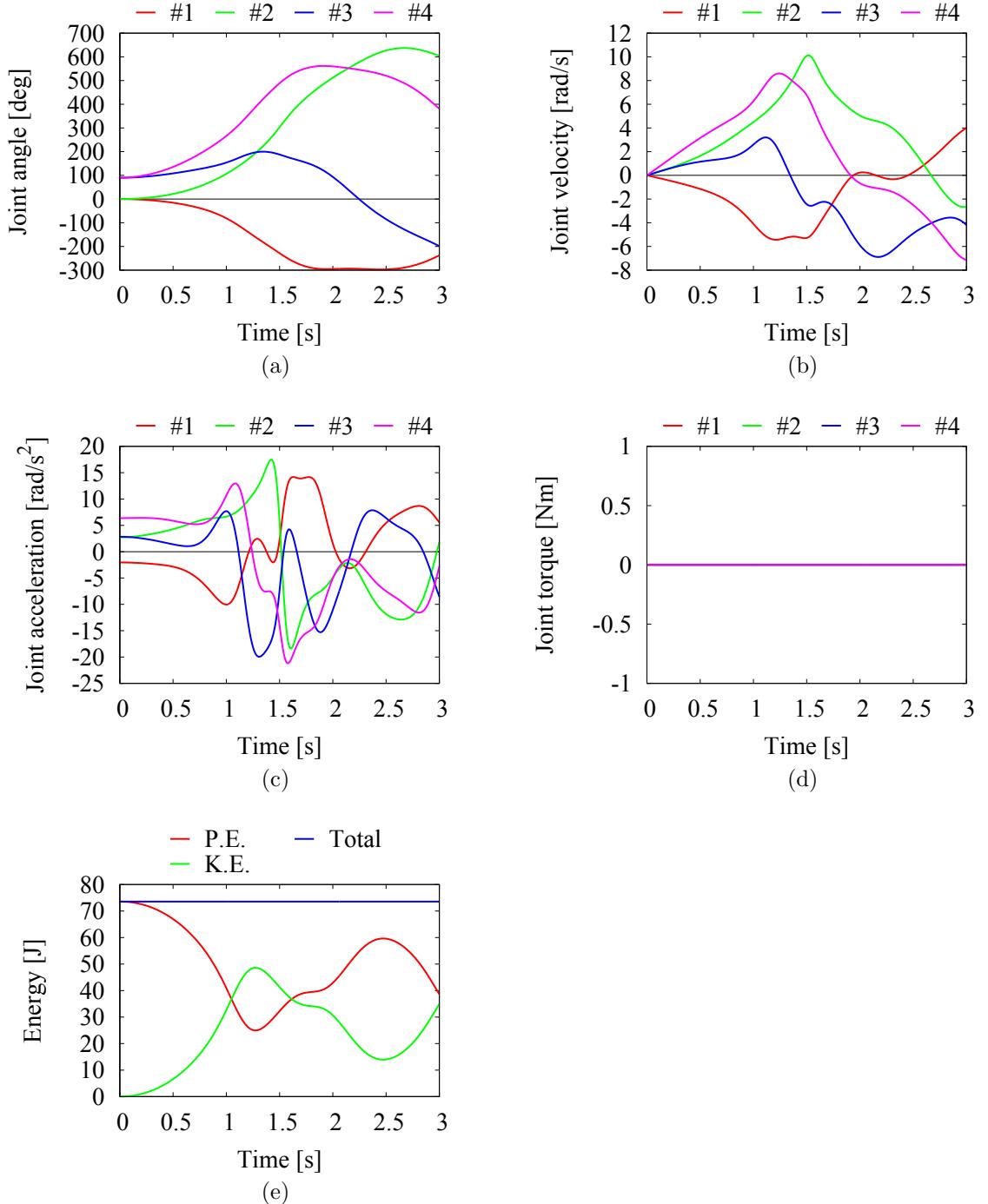


Fig. 3.7: Simulation results of the planar 5R branch chain structure: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy.

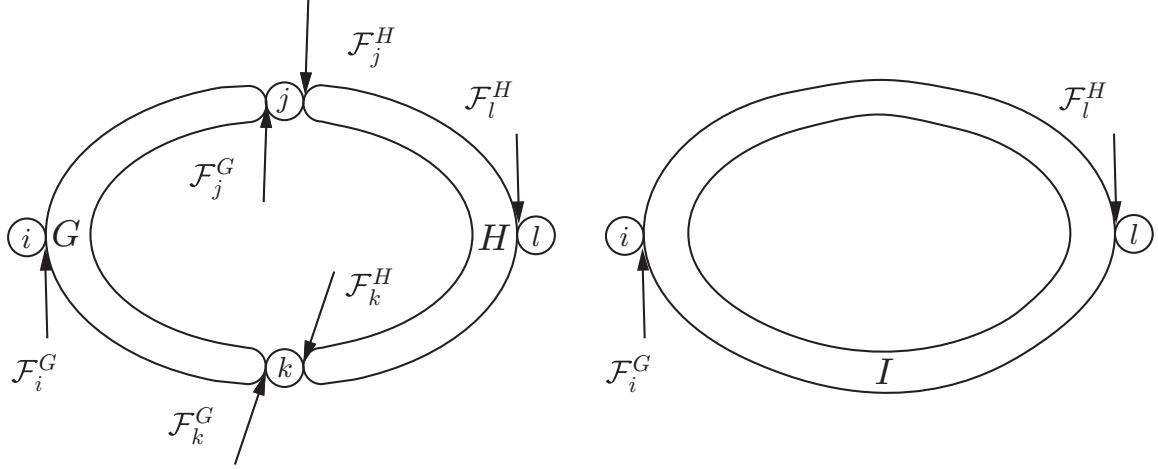


Fig. 3.8: Assembly of articulated bodies for a closed chain structure.  $G$  and  $H$  have three handles,  $I$  has two handles, respectively. The connection of joints are two handles at the same time.

Fig. 3.8 に示すように、それぞれ三つのハンドル  $A_i, A_j, A_k, B_j, B_k, B_l$  を持った二つのAB  $G, H$ を考える。ハンドル  $i, j, k$  に加わる力を  $\mathcal{F}_i^G, \mathcal{F}_j^G, \mathcal{F}_k^G$  を考えると、それぞれの加速度  $\dot{\mathcal{V}}_i^G, \dot{\mathcal{V}}_j^G, \dot{\mathcal{V}}_k^G$  との関係は Eq. (2.4) より以下のように表される。

$$\begin{bmatrix} \dot{\mathcal{V}}_i^G \\ \dot{\mathcal{V}}_j^G \\ \dot{\mathcal{V}}_k^G \end{bmatrix} = \begin{bmatrix} \Phi_i^G & \Phi_{ij}^G & \Phi_{ik}^G \\ \Phi_{ji}^G & \Phi_j^G & \Phi_{jk}^G \\ \Phi_{ki}^G & \Phi_{kj}^G & \Phi_k^G \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^G \\ \mathcal{F}_j^G \\ \mathcal{F}_k^G \end{bmatrix} + \begin{bmatrix} \mathcal{B}_i^G \\ \mathcal{B}_j^G \\ \mathcal{B}_k^G \end{bmatrix} \quad (3.9)$$

同様に、AB  $H$  に対して以下の式が成り立つ。

$$\begin{bmatrix} \dot{\mathcal{V}}_j^H \\ \dot{\mathcal{V}}_k^H \\ \dot{\mathcal{V}}_l^H \end{bmatrix} = \begin{bmatrix} \Phi_j^H & \Phi_{jk}^H & \Phi_{jl}^H \\ \Phi_{kj}^H & \Phi_k^H & \Phi_{kl}^H \\ \Phi_{lj}^H & \Phi_{lk}^H & \Phi_l^H \end{bmatrix} \begin{bmatrix} \mathcal{F}_j^H \\ \mathcal{F}_k^H \\ \mathcal{F}_l^H \end{bmatrix} + \begin{bmatrix} \mathcal{B}_j^H \\ \mathcal{B}_k^H \\ \mathcal{B}_l^H \end{bmatrix} \quad (3.10)$$

$G_j$  と  $H_j, G_k$  と  $H_k$  で接続したAB  $I$ を考える。 $G_i, H_l$  に加わる力  $\mathcal{F}_i^G, \mathcal{F}_l^H$  と、それぞれの加速度  $\dot{\mathcal{V}}_i^G, \dot{\mathcal{V}}_l^H$  との関係は以下のように表される。

$$\begin{bmatrix} \dot{\mathcal{V}}_i^G \\ \dot{\mathcal{V}}_l^H \end{bmatrix} = \begin{bmatrix} \Phi_i^I & \Phi_{il}^I \\ \Phi_{li}^I & \Phi_l^I \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^G \\ \mathcal{F}_l^H \end{bmatrix} + \begin{bmatrix} \mathcal{B}_i^I \\ \mathcal{B}_l^I \end{bmatrix} \quad (3.11)$$

シリアルリンク機構と同様に、 $\Phi_i^G$  や  $\Phi_l^H$  などから  $\Phi_i^I$  や  $\Phi_l^I$ などを求める。

関節  $j, k$  をハンドル  $G_j$  と  $H_j, G_k$  と  $H_k$  の運動学的拘束として表現する。加速度レベルの拘束は次式で表される。

$$\begin{bmatrix} \dot{\mathcal{V}}_j^H \\ \dot{\mathcal{V}}_k^H \end{bmatrix} - \begin{bmatrix} \dot{\mathcal{V}}_j^G \\ \dot{\mathcal{V}}_k^G \end{bmatrix} = \begin{bmatrix} S_j & 0 \\ 0 & S_k \end{bmatrix} \begin{bmatrix} \ddot{q}_j \\ \ddot{q}_k \end{bmatrix} + \begin{bmatrix} \dot{S}_j & 0 \\ 0 & \dot{S}_k \end{bmatrix} \begin{bmatrix} \dot{q}_j \\ \dot{q}_k \end{bmatrix} \quad (3.12)$$

関節  $j, k$  がハンドル  $H_j, H_k$  に加える力  $\mathcal{F}_j^H, \mathcal{F}_k^H$  は、関節の駆動力と関節の拘束力の和として考えると、次のように表される。

$$\begin{bmatrix} \mathcal{F}_j^H \\ \mathcal{F}_k^H \end{bmatrix} = \begin{bmatrix} T_{aj} & 0 \\ 0 & T_{ak} \end{bmatrix} \begin{bmatrix} \tau_j \\ \tau_k \end{bmatrix} + \begin{bmatrix} T_j & 0 \\ 0 & T_k \end{bmatrix} \begin{bmatrix} \lambda_j \\ \lambda_k \end{bmatrix} \quad (3.13)$$

拘束力  $\lambda_j, \lambda_k$  は Eq. (2.18) と同様に次のように求まる。

$$\begin{bmatrix} \lambda_j \\ \lambda_k \end{bmatrix} = Y_{jk} \begin{bmatrix} T_j^T & 0 \\ 0 & T_k^T \end{bmatrix} \left( \begin{bmatrix} \Phi_{ji}^G \\ \Phi_{ki}^G \end{bmatrix} \mathcal{F}_i^G - \begin{bmatrix} \Phi_{jl}^H \\ \Phi_{kl}^H \end{bmatrix} \mathcal{F}_l^H + \beta_{jk} \right) \quad (3.14)$$

ここで、 $Y_{jk}, \beta_{jk}$  は以下のように置いた。

$$Y_{jk} = \left( \begin{bmatrix} T_j^T & 0 \\ 0 & T_k^T \end{bmatrix} \left( \begin{bmatrix} \Phi_j^G & \Phi_{jk}^G \\ \Phi_{kj}^G & \Phi_k^G \end{bmatrix} + \begin{bmatrix} \Phi_j^H & \Phi_{jk}^H \\ \Phi_{kj}^H & \Phi_k^H \end{bmatrix} \right) \begin{bmatrix} T_j & 0 \\ 0 & T_k \end{bmatrix} \right)^{-1} \quad (3.15)$$

$$\beta_{jk} = \begin{bmatrix} \dot{S}_j & 0 \\ 0 & \dot{S}_k \end{bmatrix} \begin{bmatrix} \dot{q}_j \\ \dot{q}_k \end{bmatrix} + \begin{bmatrix} \mathcal{B}_j^G \\ \mathcal{B}_k^G \end{bmatrix} - \begin{bmatrix} \mathcal{B}_j^H \\ \mathcal{B}_k^H \end{bmatrix} - \left( \begin{bmatrix} \Phi_j^G & \Phi_{jk}^G \\ \Phi_{kj}^G & \Phi_k^G \end{bmatrix} + \begin{bmatrix} \Phi_j^H & \Phi_{jk}^H \\ \Phi_{kj}^H & \Phi_k^H \end{bmatrix} \right) \begin{bmatrix} T_{aj} & 0 \\ 0 & T_{ak} \end{bmatrix} \begin{bmatrix} \tau_j \\ \tau_k \end{bmatrix} \quad (3.16)$$

Eqs. (3.13), (3.14) を Eqs. (3.9), (3.10) の  $\mathcal{F}_j^G, \mathcal{F}_k^G, \mathcal{F}_j^H, \mathcal{F}_k^H$  に代入し、 $\dot{\mathcal{V}}_i^G, \dot{\mathcal{V}}_l^H$  を求め、Eq. (3.11) と比較すると、接続後の IABI について以下の関係があることが分かる。

$$\begin{aligned} \Phi_i^I &= \Phi_i^G - \begin{bmatrix} \Phi_{ij}^G & \Phi_{ik}^G \end{bmatrix} W_{jk} \begin{bmatrix} \Phi_{ji}^G \\ \Phi_{ki}^G \end{bmatrix} & \mathcal{B}_i^I &= \mathcal{B}_i^G - \begin{bmatrix} \mathcal{B}_{ij}^G & \mathcal{B}_{ik}^G \end{bmatrix} \gamma_{jk} \\ \Phi_j^I &= \Phi_l^H - \begin{bmatrix} \Phi_{lj}^H & \Phi_{lk}^H \end{bmatrix} W_{jk} \begin{bmatrix} \Phi_{jl}^H \\ \Phi_{kl}^H \end{bmatrix} & \mathcal{B}_j^I &= \mathcal{B}_l^H + \begin{bmatrix} \mathcal{B}_{lj}^H & \mathcal{B}_{lk}^H \end{bmatrix} \gamma_{jk} \\ \Phi_{ij}^I &= \begin{bmatrix} \Phi_{ij}^G & \Phi_{ik}^G \end{bmatrix} W_{jk} \begin{bmatrix} \Phi_{jl}^H \\ \Phi_{kl}^H \end{bmatrix} & &= (\Phi_{ji}^I)^T \end{aligned} \quad (3.17)$$

ここで、 $W_{jk}, \gamma_{jk}$  は以下のように置いた。

$$W_{jk} = \begin{bmatrix} T_j & 0 \\ 0 & T_k \end{bmatrix} Y_{jk} \begin{bmatrix} T_j^T & 0 \\ 0 & T_k^T \end{bmatrix} \quad (3.18)$$

$$\gamma_{jk} = W_{jk} \beta_{jk} + \begin{bmatrix} T_{aj} & 0 \\ 0 & T_{ak} \end{bmatrix} \begin{bmatrix} \tau_j \\ \tau_k \end{bmatrix} \quad (3.19)$$

Eq. (3.17) を用いて、AB を順に接続して新しいABを作成する計算を繰り返すことにより、それぞれの IABI が計算できる。IABI が計算できれば、

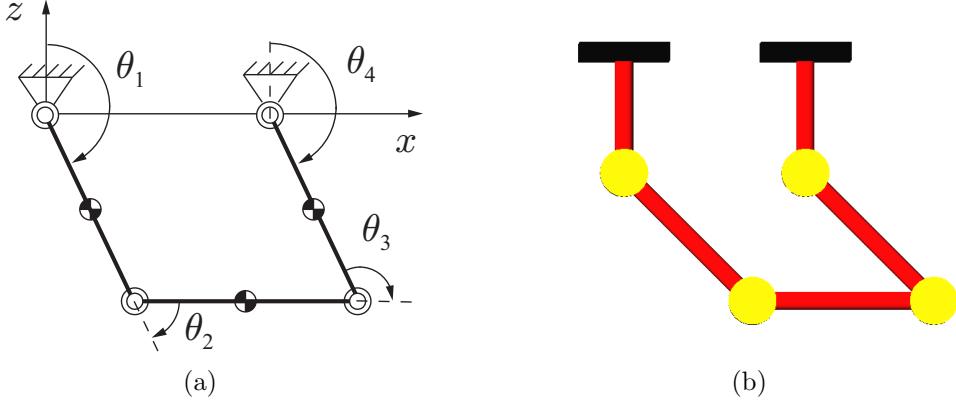


Fig. 3.9: A planar 4R closed chain structure.

Eqs. (3.13), (3.14) を用いて関節に加わる力  $\mathcal{F}_j^H, \mathcal{F}_k^H$  が得られる。これをさらに Eqs. (3.9), (3.10) へ代入して  $\dot{\mathcal{V}}_j^G, \dot{\mathcal{V}}_k^G, \dot{\mathcal{V}}_j^H, \dot{\mathcal{V}}_k^H$  を得る。加速度レベルの関節の拘束式 Eq. (3.12) から関節加速度を求める

$$\begin{bmatrix} \ddot{q}_j \\ \ddot{q}_k \end{bmatrix} = \begin{bmatrix} T_{aj}^T & \mathbf{0} \\ \mathbf{0} & T_{ak}^T \end{bmatrix} \left( \begin{bmatrix} \dot{\mathcal{V}}_j^H \\ \dot{\mathcal{V}}_k^H \end{bmatrix} - \begin{bmatrix} \dot{\mathcal{V}}_j^G \\ \dot{\mathcal{V}}_k^G \end{bmatrix} - \begin{bmatrix} \dot{S}_j & \mathbf{0} \\ \mathbf{0} & \dot{S}_k \end{bmatrix} \begin{bmatrix} \dot{q}_j \\ \dot{q}_k \end{bmatrix} \right) \quad (3.20)$$

となる。これにより、求めたハンドルの相対加速度  $\dot{\mathcal{V}}_j^G, \dot{\mathcal{V}}_k^G, \dot{\mathcal{V}}_j^H, \dot{\mathcal{V}}_k^H$  から関節加速度が求まる。

### シミュレーション

閉リンク機構の解法例として Fig. 3.9 に示す平面 3 リンクマニピュレータのシミュレーションを行った。ここで、関節はすべて  $y$  軸回りの回転関節とする。各リンクの長さは  $l = 1.0 \text{ m}$ , 質量は  $m = 1.0 \text{ kg}$ , 重心回りの慣性テンソルは  $\bar{I}_C = \text{diag}(ml^2/12, ml^2/12, 0) \text{ kg} \cdot \text{m}$  とした。また、重心位置は各リンクの中心に設定した。 $\theta_1^{ini} = 135 \text{ deg}, \theta_2^{ini} = -45 \text{ deg}, \theta_3^{ini} = 225 \text{ deg}, \theta_4^{ini} = 135 \text{ deg}$  の初期角度 (Fig. 3.9) から初速および制御入力なしの条件で、自由落下するシミュレーションを行った。シミュレーション時間を 3 s, 積分の刻み幅を 1 ms, 重力加速度を  $z$  軸方向に  $-9.81 \text{ m/s}^2$  とした。

運動の様子を Fig. 3.10 にスナップショットで示す。また、シミュレーション結果を Fig. 3.11 に示す。Fig. 3.11 (a) は関節角度, (b) は関節速度, (c) は関節加速度, (d) は関節トルク, (e) は力学的エネルギーである。Fig. 3.10, Figs. 3.11 (a)–(c) より、設定した初期状態から重力の影響を受け、自由落下する様子が確認できる。また、Figs. 3.11 (d), (e) より、制御入力を与えない

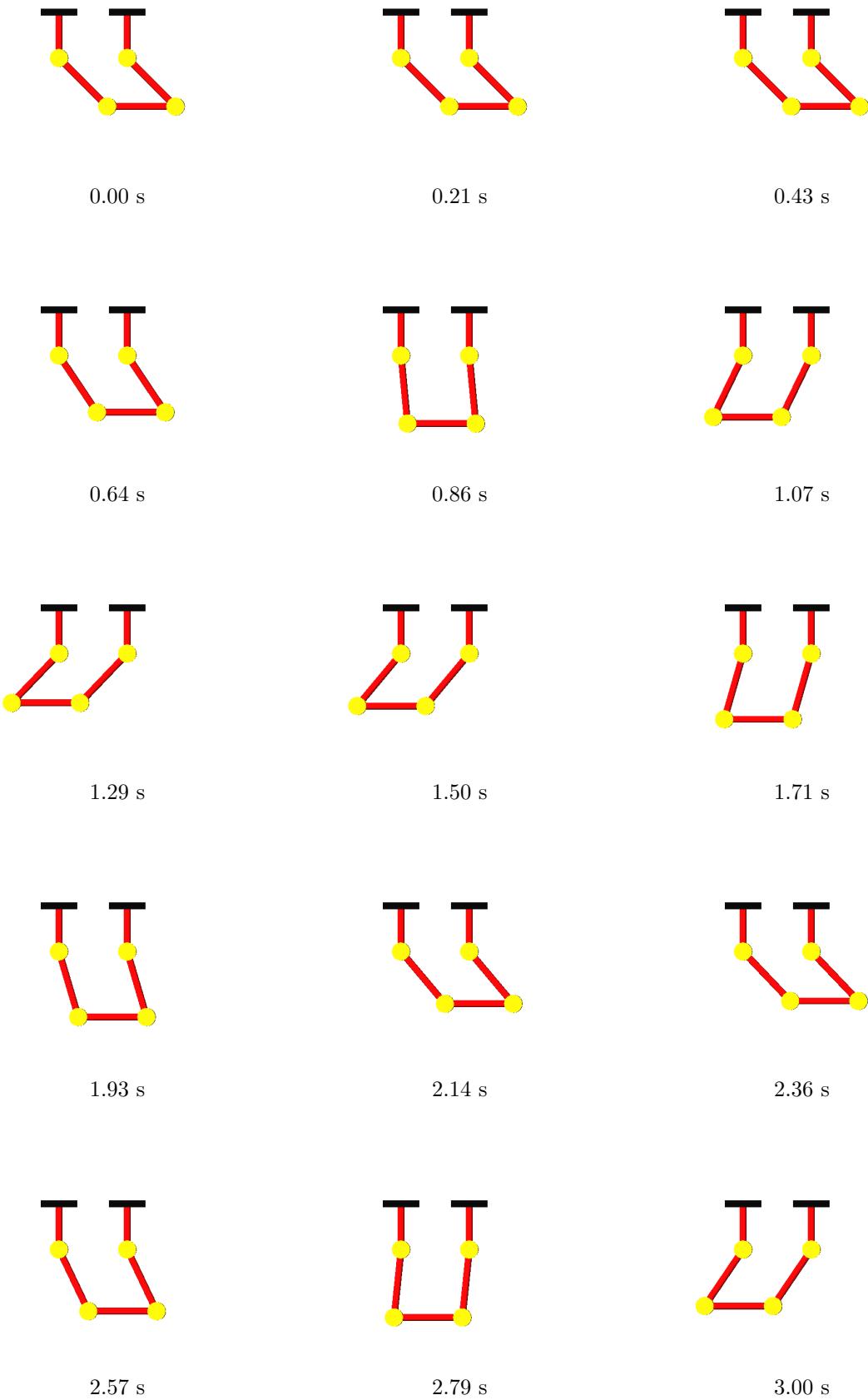


Fig. 3.10: Snapshots from simulation of the planar 4R closed chain structure.

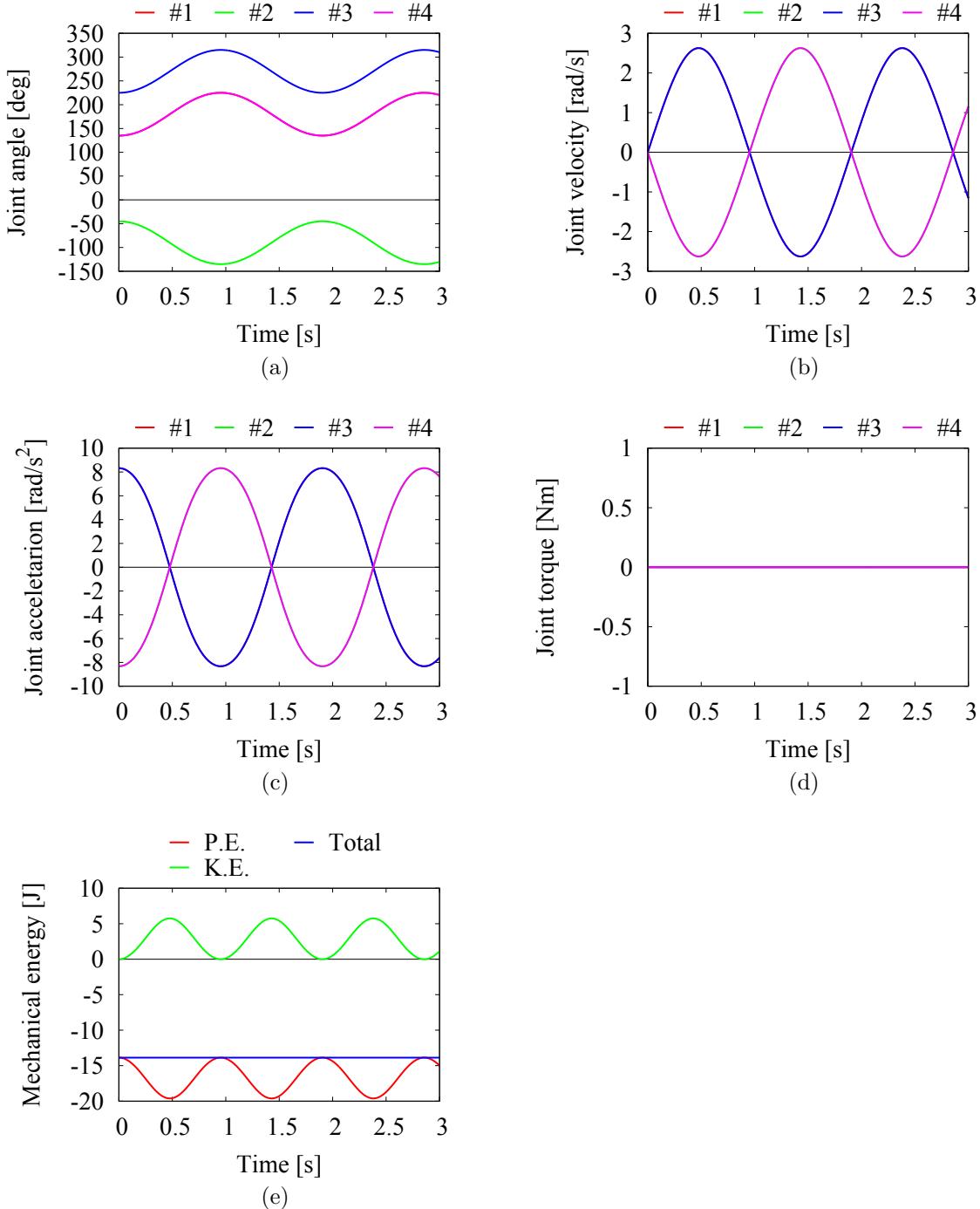


Fig. 3.11: Simulation results of the planar 4R closed chain structure: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy.

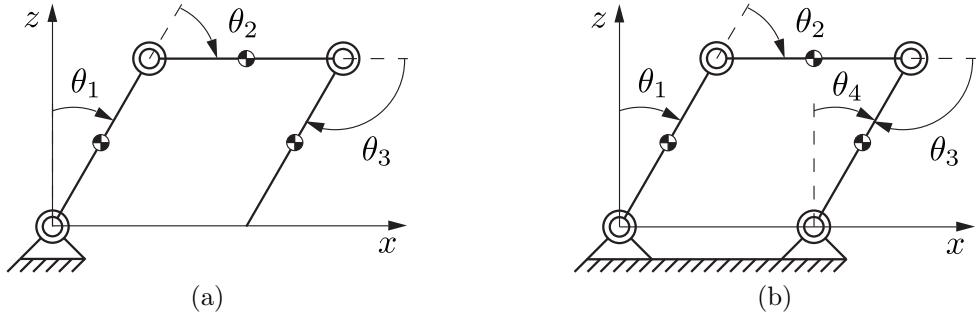


Fig. 3.12: Compare 3R open chain to 4R closed chain.

状態で、位置エネルギー(P.E.)と運動エネルギー(K.E.)の総和が一定となり、力学的エネルギー保存則が確認できた。よって、正しく順動力学計算が行えていると判断できる。

## 3.2 3節開リンク機構と4節閉リンク機構の比較

### 3.2.1 計算アルゴリズムの比較

3節開リンク機構と4節閉リンク機構の計算アルゴリズムを比較しながら、4節閉リンク機構の順動力学計算について述べる。シリアルリンク機構のシミュレーションで用いた3節開リンク機構のモデル図をFig. 3.12 (a)に、閉リンク機構のシミュレーションで用いた4節閉リンク機構の機構図をFig. 3.12 (b)に再度示す。

ADAに基づく3節開リンク機構の計算アルゴリズムをFig. 3.13に、4節閉リンク機構の計算アルゴリズムをFig. 3.14に示す。Fig. 3.13とFig. 3.14は共に同じ計算ステップを用いる。まず、(a)から(c)のassemblyステップでAとBを結合しZを、ZとCを結合しYを作成する。その後、(d)から(f)のdisassemblyステップでABを求めた逆の順に分解して、関節から伝わる力 $\mathcal{F}_i^{AB}$ と関節の加速度 $\ddot{\theta}_i$ を計算する。ただし、 $AB = A, B, C, Z, Y$ であり、Fig. 3.13の場合は*i* = 1 ~ 3、Fig. 3.14の場合は*i* = 1 ~ 4である。

開リンク機構と閉リンク機構の計算アルゴリズムの最大の違いは、ABを分解する際の関節の数である。開リンク機構の場合は、すべてのABに対して一つずつ関節を分解する。一方、閉リンク機構の場合は、Fig. 3.14 (d)のように少なくとも1回は二つ以上の関節を一度に分解する。

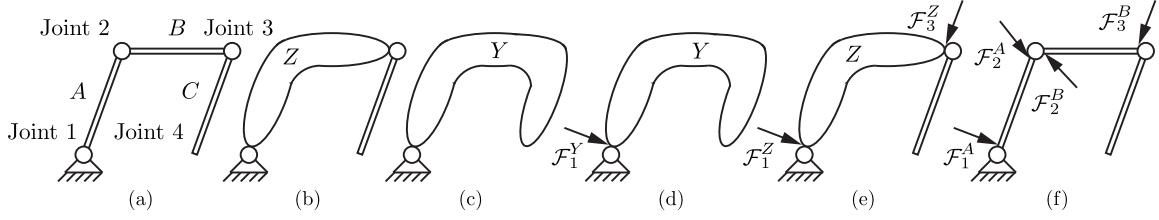


Fig. 3.13: Illustration of the assembly and disassembly phases of 3R open chain.

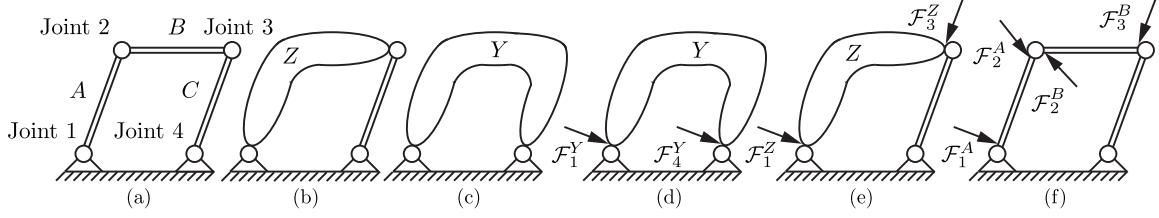


Fig. 3.14: Illustration of the assembly and disassembly phases of 4R closed chain.

### 3.2.2 開リンク機構と閉リンク機構の計算の違い

開リンク機構と閉リンク機構で計算方法が異なる Fig. 3.13 (d) と Fig. 3.14 (d) の部分の計算方法を以下に示す。まず、Fig. 3.13 (d) について述べる。関節 1 から Y に伝わる力  $\mathcal{F}_1^Y$  と、その結果発生する加速度  $\dot{\mathcal{V}}_1^Y$  の間には以下のような関係がある。

$$\dot{\mathcal{V}}_1^Y = \Phi_1^Y \mathcal{F}_1^Y + \mathcal{B}_1^Y \quad (3.21)$$

ただし、式中の  $\Phi_1^Y$  は Y の IABI,  $\mathcal{B}_1^Y$  は Y のバイアス加速度とする。 $\Phi_1^Y$  と  $\mathcal{B}_1^Y$  はすでに assembly ステップで求まっている。また、 $\mathcal{F}_1^Y$  は次式から求まる。

$$\mathcal{F}_1^Y = \mathbf{T}_1 (\mathbf{T}_1^T \Phi_1^Y \mathbf{T}_1)^{-1} \mathbf{T}_1^T \mathcal{B}_1^Y \quad (3.22)$$

関節 1 の角加速度  $\ddot{\theta}_1$  は次式から求まる。

$$\ddot{\theta}_1 = \mathbf{T}_{a1}^T \dot{\mathcal{V}}_1^Y \quad (3.23)$$

$y$  軸回りの回転関節の場合、 $\mathbf{T}_i$  と  $\mathbf{T}_{ai}$  は Table 2.1 より、次のようになる。

$$\mathbf{T}_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_{ai} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.24)$$

Eqs. (3.21), (3.22), (3.24) から求まる  $\dot{\nu}_1^Y$  を Eq. (3.23) に代入することで  $\ddot{\theta}_1$  が求まる。

次に, Fig. 3.14 (d) について述べる。関節 1, 4 から Y に伝わる力  $\mathcal{F}_1^Y, \mathcal{F}_4^Y$  と, その結果発生する加速度  $\dot{\nu}_1^Y, \dot{\nu}_4^Y$  には以下のような関係がある。

$$\begin{bmatrix} \dot{\nu}_1^Y \\ \dot{\nu}_4^Y \end{bmatrix} = \begin{bmatrix} \Phi_1^Y & \Phi_{14}^Y \\ \Phi_{41}^Y & \Phi_4^Y \end{bmatrix} \begin{bmatrix} \mathcal{F}_1^Y \\ \mathcal{F}_4^Y \end{bmatrix} + \begin{bmatrix} \mathcal{B}_1^Y \\ \mathcal{B}_4^Y \end{bmatrix} \quad (3.25)$$

また,  $\mathcal{F}_1^Y, \mathcal{F}_4^Y$  は次式から求まる。

$$\begin{bmatrix} \mathcal{F}_1^Y \\ \mathcal{F}_4^Y \end{bmatrix} = \begin{bmatrix} T_1 & \mathbf{0} \\ \mathbf{0} & T_4 \end{bmatrix} \mathbf{Y}^{-1} \begin{bmatrix} T_1 & \mathbf{0} \\ \mathbf{0} & T_4 \end{bmatrix}^T \begin{bmatrix} \mathcal{B}_1^Y \\ \mathcal{B}_4^Y \end{bmatrix} \quad (3.26)$$

ただし,

$$\mathbf{Y} = \begin{bmatrix} T_1 & \mathbf{0} \\ \mathbf{0} & T_4 \end{bmatrix}^T \begin{bmatrix} \Phi_1^Y & \Phi_{14}^Y \\ \Phi_{41}^Y & \Phi_4^Y \end{bmatrix} \begin{bmatrix} T_1 & \mathbf{0} \\ \mathbf{0} & T_4 \end{bmatrix} \quad (3.27)$$

とおいた。関節 1, 4 の角加速度  $\ddot{\theta}_1, \ddot{\theta}_4$  は次式から求まる。

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_4 \end{bmatrix} = \begin{bmatrix} T_{a1} & \mathbf{0} \\ \mathbf{0} & T_{a4} \end{bmatrix}^T \begin{bmatrix} \dot{\nu}_1^Y \\ \dot{\nu}_4^Y \end{bmatrix} \quad (3.28)$$

Eqs. (3.25), (3.26) を Eq. (3.28) に代入することで  $\ddot{\theta}_1, \ddot{\theta}_4$  が求まる。

### 3.3 4 節閉リンク機構の特異姿勢

閉リンク機構の順動力学計算では特異姿勢が存在することが知られている [40]。4 節閉リンク機構の特異姿勢を Figs. 3.15 に示す。 $\theta_1 = \pm 90 \text{ deg}$  が特異姿勢となる。3.1 節で用いた 4 節閉リンク機構のシミュレーションを再度行い, 特異姿勢がシミュレーションにどのような影響を与えるか確認する。シミュレーションのパラメータは, 基本的には 3.1 節と同様のものを用いる。ただし,  $\theta_1 = \pm 90 \text{ deg}$  の特異姿勢を通過するように, 初期角度は  $\theta_1^{ini} = 60 \text{ deg}, \theta_2^{ini} = 30 \text{ deg}, \theta_3^{ini} = 150 \text{ deg}, \theta_4^{ini} = 60 \text{ deg}$  とし, 初速および制御入力なしの条件で, 自由落下するシミュレーションを行った。シミュレーション時間は 3 s, 積分の刻み幅を 1 ms, 重力加速度を z 軸方向に  $-9.81 \text{ m/s}^2$  とした。

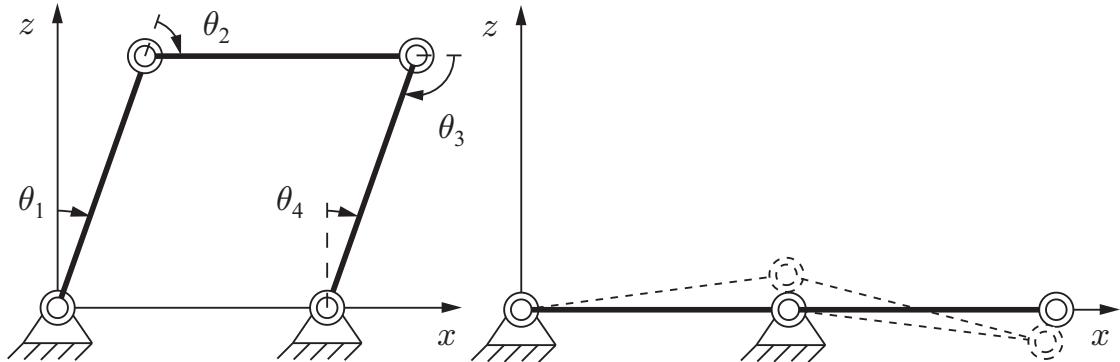


Fig. 3.15: Singular configuration of four-bar linkage.

運動の様子を Fig. 3.16 にスナップショットで示す。また、シミュレーション結果を Fig. 3.17 に示す。Fig. 3.17 (a) は関節角度、(b) は関節速度、(c) は関節加速度、(d) は関節トルク、(e) は力学的エネルギー、(f) はヤコビ行列  $J \in \mathbb{R}^{6 \times 4}$  の行列式  $\det(JJ^T)$  の値である。ここで、 $J$  は関節変位  $q$  と先端の速度  $v_e$  の関係を表す行列 ( $v_e = Jq$ ) であり、行列式がゼロの時が特異姿勢となる。Fig. 3.16, Figs. 3.17 (a)–(c) より、設定した初期状態から重力の影響を受け、自由落下する様子が確認できる。また Fig. 3.17 (f) から、5回、特異姿勢を通過していることが分かる。Figs. 3.17 (c), (b), (e) より、特異姿勢を通過する際に、関節速度と関節加速度のジャンプが発生し、位置エネルギー (P.E.) と運動エネルギー (K.E.) の総和が一定に保たれなくなっている。このことから、閉リンク機構のシミュレーションを行う際は、特異姿勢を考慮して行わなければならない。

## 3.4 浮遊ベースのモデリング

浮遊ベースのモデリングには 6DOF 関節を用いる。6DOF 関節をモデリングするには、三軸の姿勢表現が必要になる。この姿勢表現の手法として、オイラー角を用いる手法と単位クオータニオンを用いる手法がある。

### 3.4.1 オイラー角による姿勢表現

オイラー角は独立した三つの変数で姿勢を表現する手法である。オイラー角の関節軸のとり方は様々であるが、ここでは  $z, y, x$  軸の順でとった

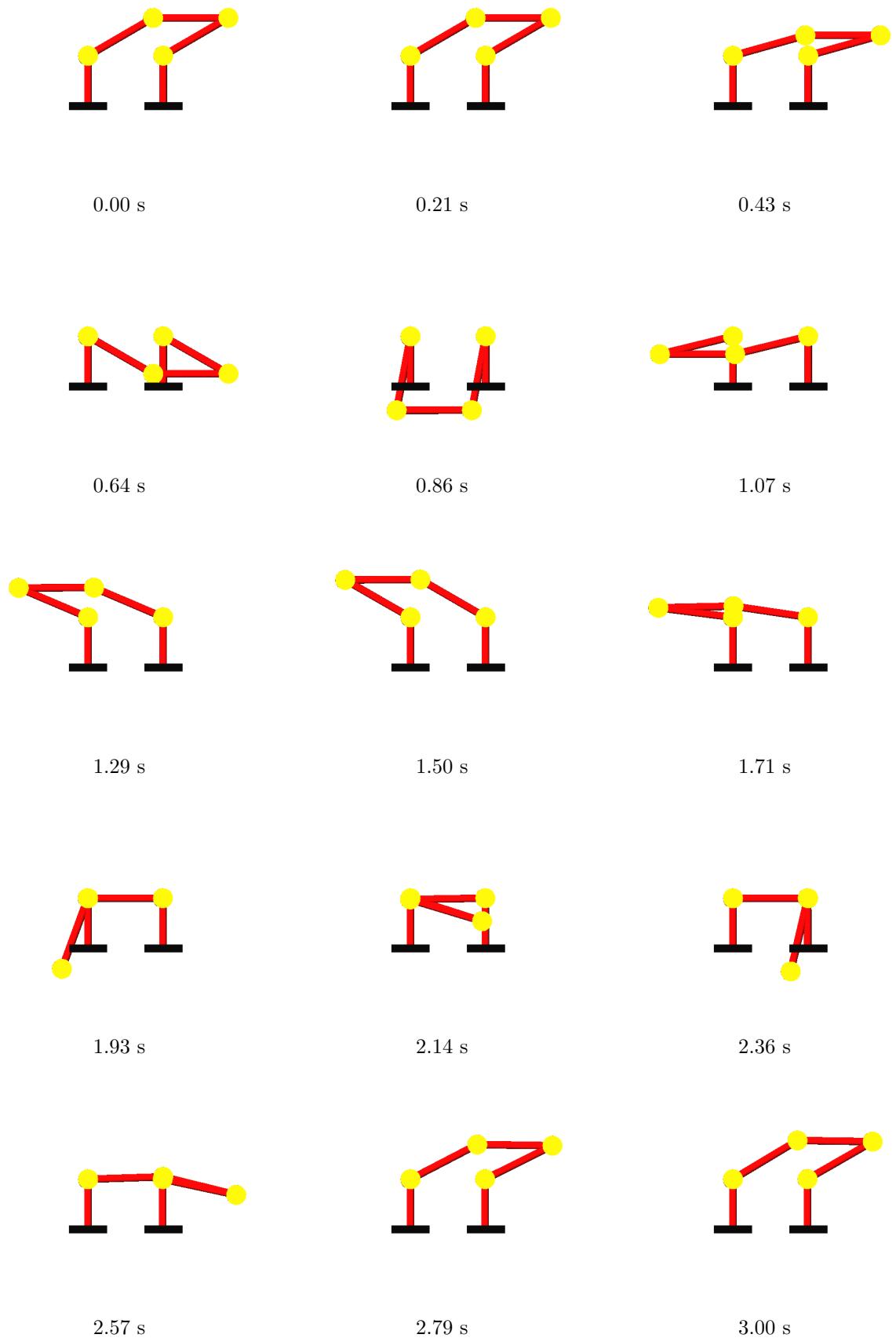


Fig. 3.16: Snapshots from simulation of the planar 4R closed chain structure.

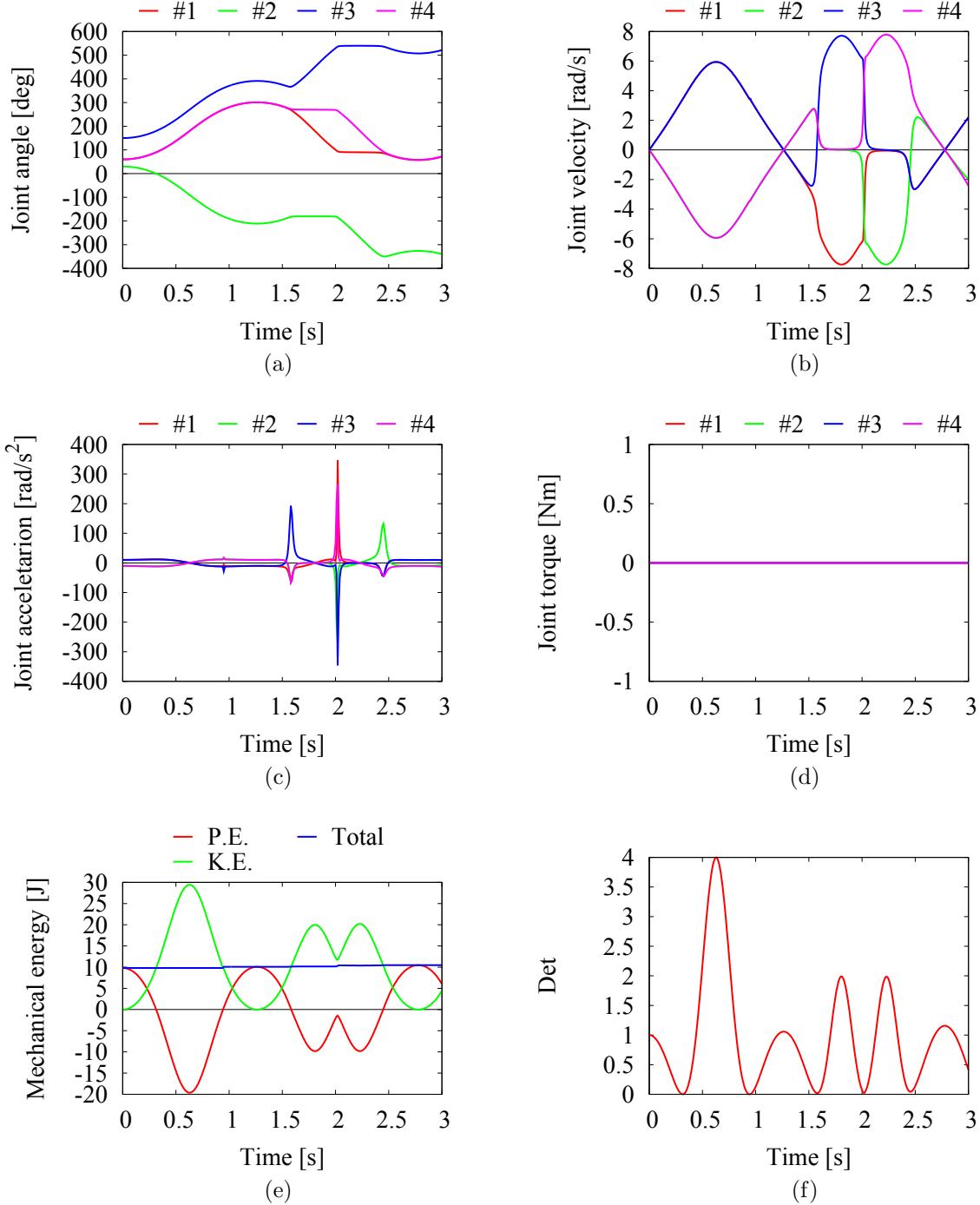


Fig. 3.17: Simulation results of the planar 4R closed chain structure: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) mechanical energy and (f) determinant  $JJ^T$ .

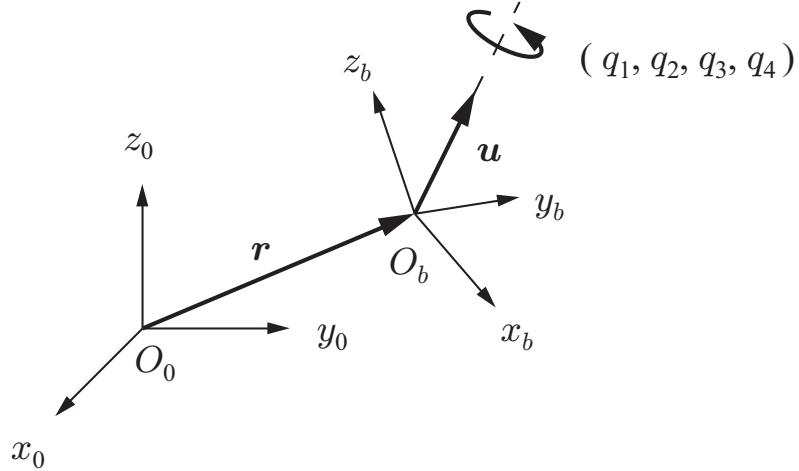


Fig. 3.18: Geometrical interpretation of the six-DOF joint position variables.

場合を示す。このとき、性座標系 $\{0\}$ からベース座標系 $\{b\}$ への変換を行う回転行列は各関節軸に対する変位を $\theta_1, \theta_2, \theta_3$ とする以下通りである。

$${}^b_0\mathbf{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_3 & s_3 \\ 0 & -s_3 & c_3 \end{bmatrix} \begin{bmatrix} c_2 & 0 & -s_2 \\ 0 & 1 & 0 \\ s_2 & 0 & c_3 \end{bmatrix} \begin{bmatrix} c_1 & s_1 & 0 \\ -s_1 & c_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.29)$$

$$= \begin{bmatrix} c_1c_2 & s_1c_2 & -s_2 \\ c_1s_2s_3 - s_1c_3 & s_1s_2s_3 + c_1c_3 & c_2s_3 \\ c_1s_2c_3 + s_1s_3 & s_1s_2c_3 - c_1s_3 & c_2c_3 \end{bmatrix} \quad (3.30)$$

ここで、 $s_\circ = \sin(\theta_\circ)$ ,  $c_\circ = \cos(\theta_\circ)$  である。オイラー角のメリットとして、各軸に対して独立した位置、速度、加速度が得られることが挙げられる。しかし、Eq. (3.30) に示す通り、 $\theta_2 = \pm 90 \text{ deg}$  のとき、他の二つの関節軸が重なるため表現上の特異点が発生する。

### 3.4.2 単位クオータニオンによる姿勢表現

単位クオータニオンは以下の式を満たす四つのパラメータで表現される。

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1 \quad (3.31)$$

$q_1, q_2, q_3, q_4$  は以下の通りである。ここで、 $\theta$  は回転角、 $\mathbf{u} = [u_{x_0}, u_{y_0}, u_{z_0}]^T$  は回転軸を示す。

$$\begin{aligned} q_1 &= \sin(\theta/2)u_{x_0} \\ q_2 &= \sin(\theta/2)u_{y_0} \\ q_3 &= \sin(\theta/2)u_{z_0} \\ q_4 &= \cos(\theta/2) \end{aligned} \quad (3.32)$$

単位クオータニオンを用いて6自由度関節をモデリングする場合には、慣性座標系  $\{0\}$  からベース座標系  $\{b\}$  への変換を行う回転行列  ${}^b R_0$  は次式となる。

$${}^b R_0 = 2 \begin{bmatrix} q_4^2 + q_1^2 - 1/2 & q_1 q_2 + q_4 q_3 & q_1 q_3 - q_4 q_2 \\ q_1 q_2 - q_4 q_3 & q_4^2 + q_2^2 - 1/2 & q_2 q_3 + q_4 q_1 \\ q_1 q_3 + q_4 q_2 & q_2 q_3 - q_4 q_1 & q_4^2 + q_3^2 - 1/2 \end{bmatrix} \quad (3.33)$$

単位クオータニオンのパラメータは非独立であるため、関節角速度  $\omega = [\omega_x, \omega_y, \omega_z]^T$  と単位クオータニオンの関係を微分で表すことができない。しかし、四つのパラメータで姿勢を表現するため、表現上の特異点が発生しないというメリットがある。

関節角速度  $\omega$  と単位クオータニオンの微分の関係は以下の式で表される。

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (3.34)$$

上式で計算されるクオータニオンの微分から姿勢を表現するためのパラメータを得ることができる。

### 浮遊ベースのための姿勢表現法の選択

このようにどちらの姿勢表現法にもメリット、デメリットがある。オイラー角を用いる方法はフィードバック制御など、各関節軸の変位を必要とするものには有用である。ただし、表現上の特異点が発生するため、特異点適合法[48]などを用いなければ、特異点近傍では姿勢の表現ができない。

単位クオータニオンを用いる方法は、上記のような制御には各関節の変位を直接求めることができないため不利である。しかし、表現上の特異点が発生しないため、制御を考慮しない場合には有用であるといえる。

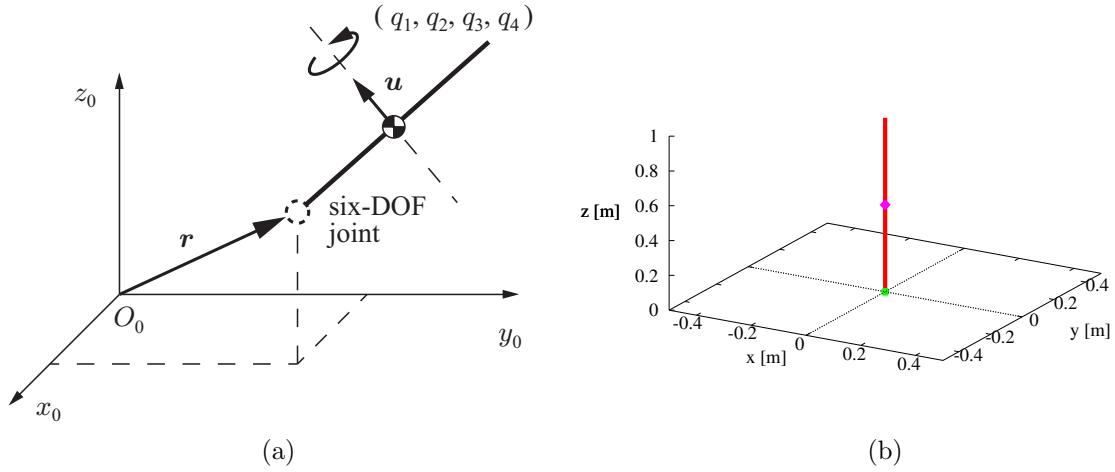


Fig. 3.19: The free base with a six-DOF joint.

今回はモデリングの対象である浮遊ベースに対して制御を行うことは考えないため、特異点の発生しない単位クオータニオンを採用する。

### シミュレーション

Fig. 3.19 に三次元の浮遊ベースのモデルを示す。このモデルは、根元に 6DOF 関節を設定した一つの剛体である。慣性座標系  $\{0\}$  からベース座標系  $\{b\}$  までの距離を  $r$ 、回転軸を  $u$ 、クオータニオンのパラメータを  $(q_1, q_2, q_3, q_4)$  とする。このときのベースの位置は以下の式で表される。ここで  ${}^b r$  と  ${}^0 r$  は、それぞれ、ベース座標系と慣性座標系から見た各軸方向への並進成分の変位である。

$${}^0 r = {}^0 R^{-1} {}^b r \quad (3.35)$$

リンクの長さは  $l = 1 \text{ m}$ 、質量は  $m = 1 \text{ kg}$ 、剛体の慣性テンソルは  $\bar{I}_C = \text{diag}(ml^2/12, ml^2/12, 0) \text{ kg} \cdot \text{m}$ とした。また、重心位置は剛体の中心に設定した。剛体が  $z_0$  軸に対して平行、 $x_0$  軸と  $y_0$  軸に対して垂直な状態を初期角度、 $r = 0$  を初期位置としシミュレーションを行った。また、6自由度関節の初期速度は、重心の位置と速度に影響を与えないように  $\omega_{0x}^{ini} = 0.72 \text{ rad/s}$ 、 $\omega_{0y}^{ini} = 0.48 \text{ rad/s}$ 、 $\omega_{0z}^{ini} = 0.24 \text{ rad/s}$ 、 $v_{0x}^{ini} = -0.24 \text{ m/s}$ 、 $v_{0y}^{ini} = 0.36 \text{ m/s}$ 、 $v_{0z}^{ini} = 0.00 \text{ m/s}$  と設定した。シミュレーション時間を  $10 \text{ s}$ 、積分の刻み幅を  $1 \text{ ms}$  とした。

運動の様子を Fig. 3.20 にスナップショットで示す。6自由度関節が三次元運動を行っている様子が確認できる。

Link — Joint • CoM \*

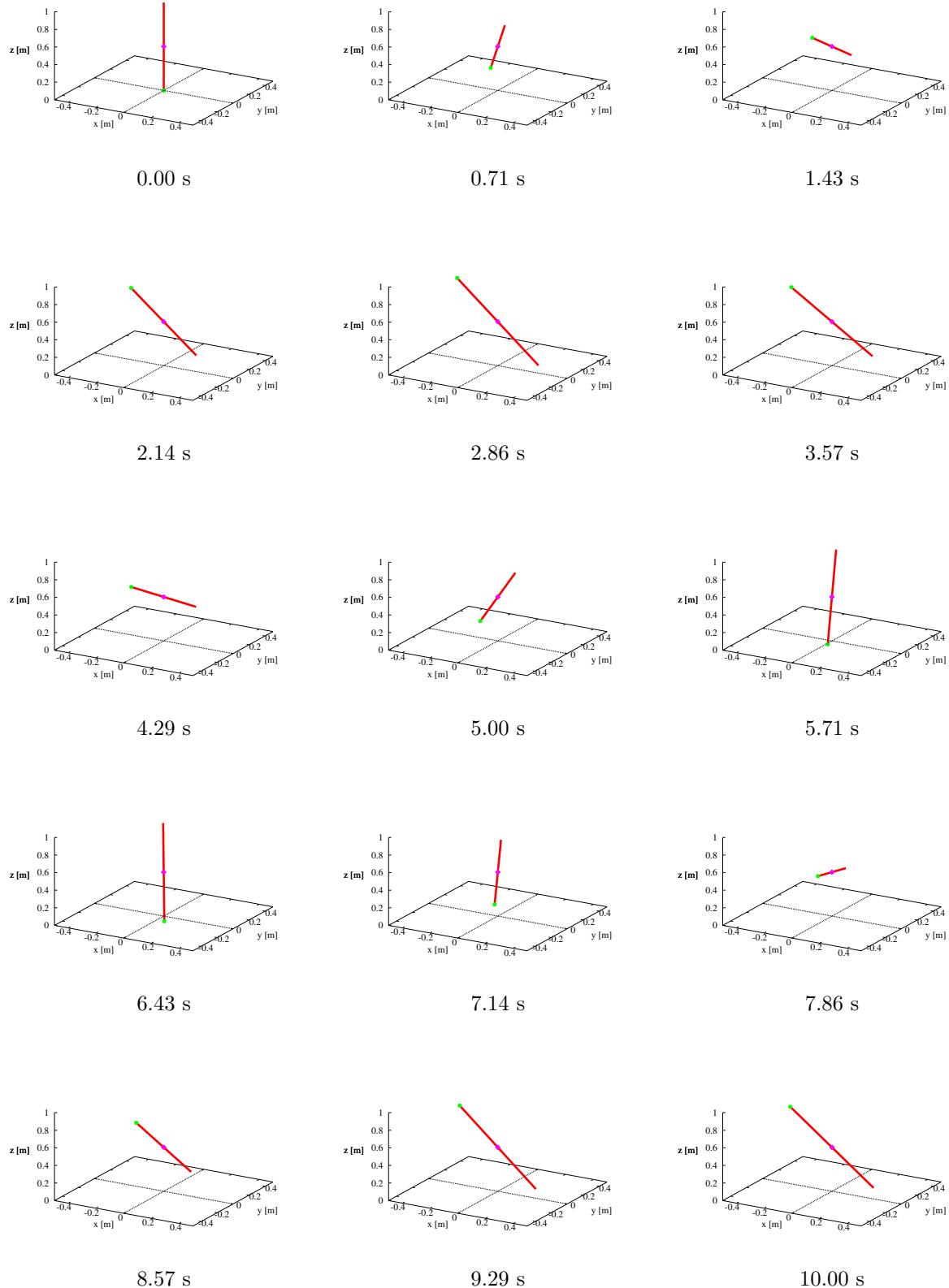


Fig. 3.20: Snapshots from simulation of the 3D free base.

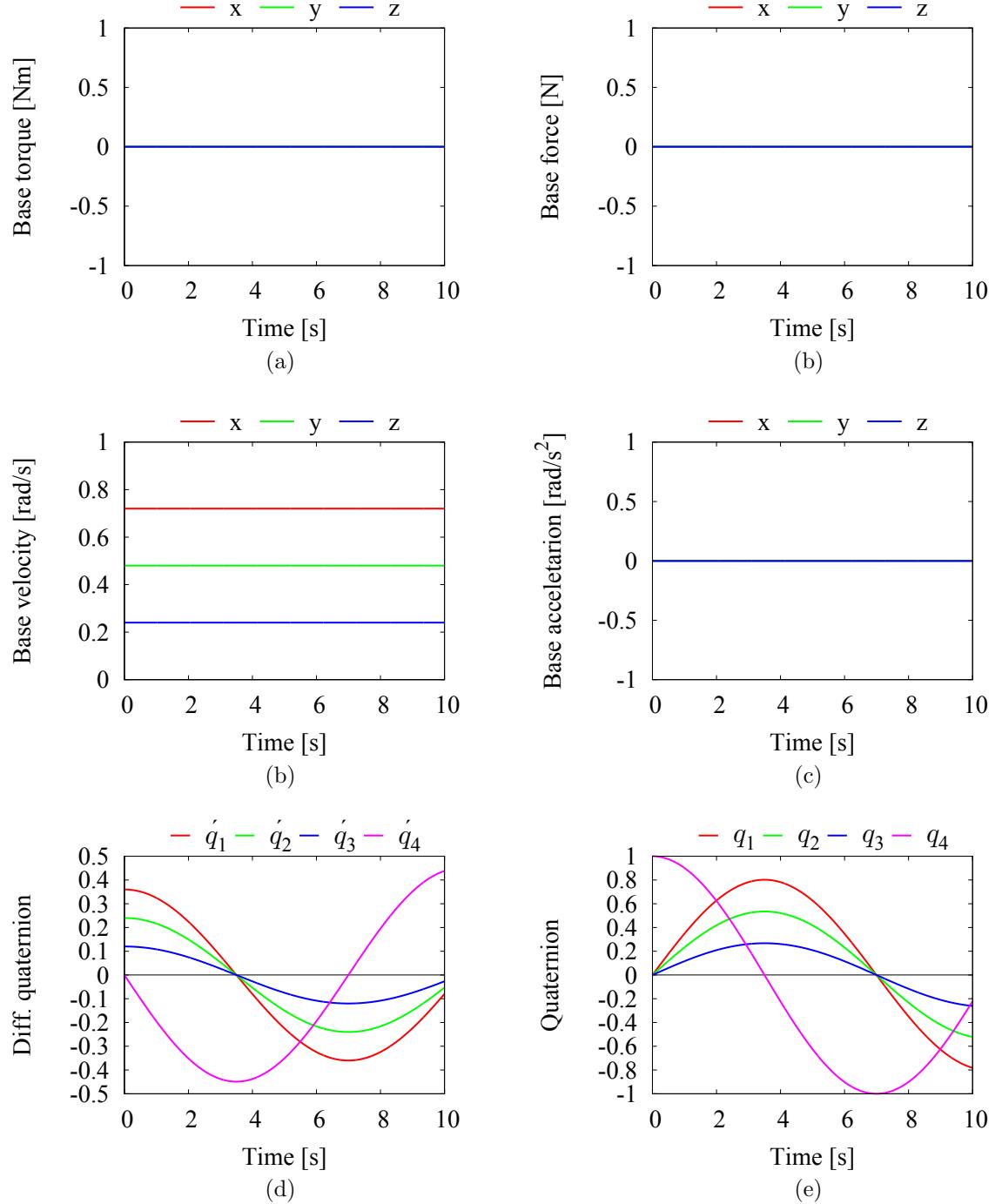


Fig. 3.21: Simulation results of the 3D free base: (a) base torque, (b) base force, (c) base angular velocity, (d) base velocity and (e) base angle.

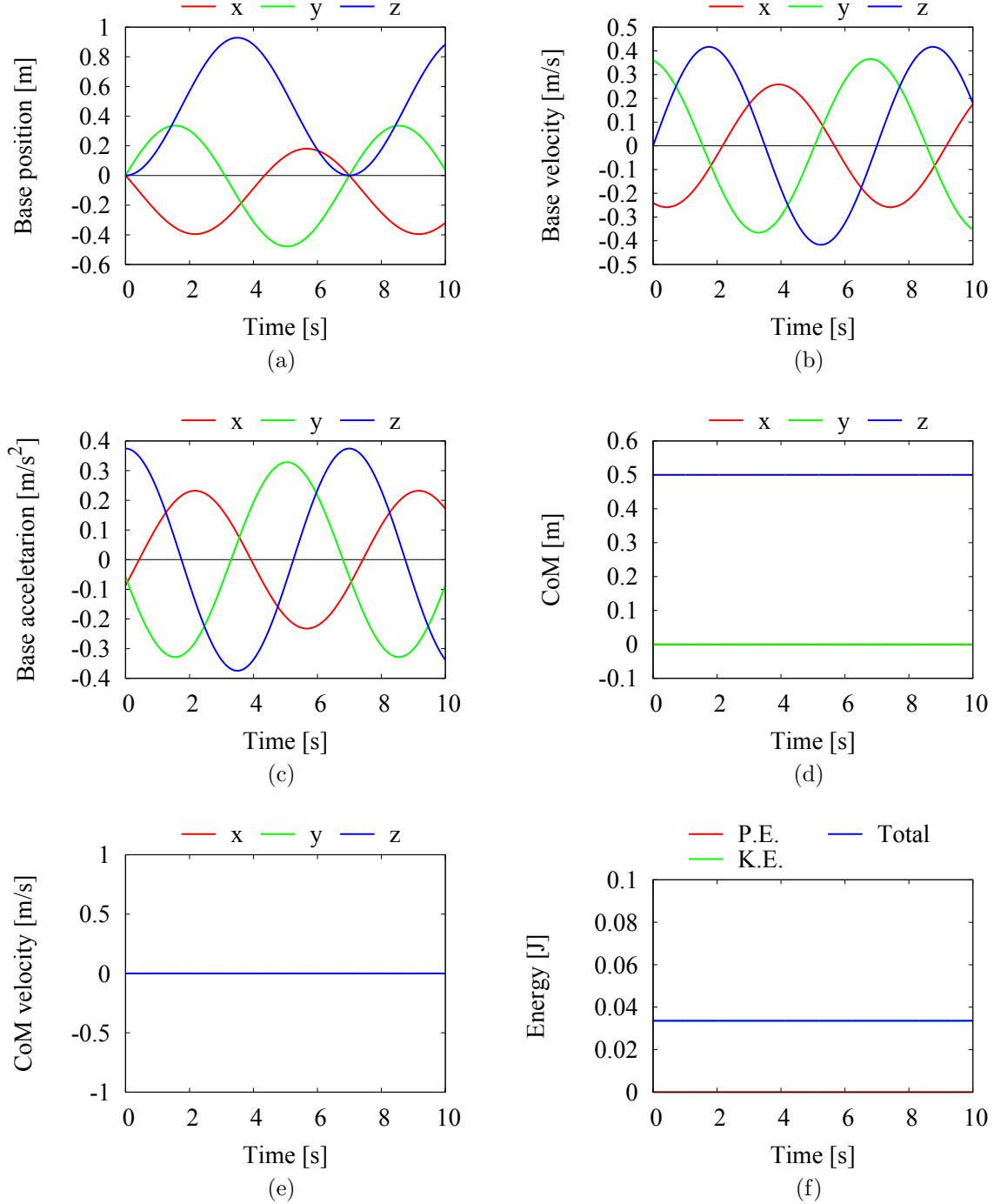


Fig. 3.22: Simulation results of the 3D free base: (a) base position, (b) base velocity, (c) base acceleration, (d) CoM position, (e) CoM velocity and (f) mechanical energy.

シミュレーション結果の回転成分を Fig. 3.21 に示す。Figs. 3.21 (a) と (b) は 6 自由度関節に加えたトルクと力、(c) は関節角速度、(d) は関節角加速度、(d) と (e) はクオータニオンのパラメータである。(a) と (b) より、関節にトルクと力を与えていないことが分かる。(c) と (d) より、与えられた初速から一定の角速度で運動していることが分かる。(d) はクオータニオンのパラメータの微分  $\dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4$  であり、(c) の角速度  $\omega$  と Eq. (3.34) の関係から、が求めた。これを積分して  $q_1, q_2, q_3, q_4$  が求まる。また、Eq. (3.33) の関係から、浮遊ベースの姿勢が表現できる。

また、シミュレーション結果の並進成分と力学的エネルギーを Fig. 3.22 に示す。Fig. 3.22 (a) は 6 自由度関節の位置  $r$ 、(b) は関節の並進速度、(c) は関節の並進加速度、(d) は重心位置、(e) は重心の並進速度、(f) は力学的エネルギーである。

Figs. 3.22 (a)–(c) より、6 自由度関節が三次元運動を行っている様子が確認できる。Figs. 3.22 (d), (e) より、重心位置が一定で重心速度がゼロとなり、運動量保存則が確認できた。また、Fig. 3.21 (a), Fig. 3.22 (f) より、制御入力を与えない状況で、位置エネルギー (P.E.) と運動エネルギー (K.E.) の総和が一定となり、力学的エネルギー保存則が確認できた。よって、正しく順動力学計算が行えていると判断できる。

### 3.5 本章のまとめ

ADAに基づくリンク系のモデリング手法として、シリアルリンク機構とブランチリンク機構、閉リンク機構の三つの機構の計算手法を示し、シリアルリンク機構から閉リンク機構へ容易に拡張できることを示した。ここで、閉リンク機構では、特異姿勢が存在することを確認した。また、三次元の浮遊ベースのモデリングに対して、単位クオータニオンによる計算手法を示した。さらに、それぞれのモデルでシミュレーションを行い、正しく順動力学計算が行われていることを示した。



## 第4章 数値・数式ハイブリッドモデリング

ロボットのシミュレーションの性能は順動力学エンジンや衝突エンジン、積分手法などに依存する。その中でも順動力学エンジンは重要な要素であり、この計算の高速化や高精度化が求められている。第2章で述べたように、我々は、閉リンク機構を含むリンク系の順動力学計算を効率的に行うことができる Assembly-Disassembly Algorithm (ADA) に基づく順動力学エンジンを開発している。この計算効率向上を図るアプローチとして、数値・数式ハイブリッド手法を用いていく。

数式処理により Lagrange 法を用いて求めた運動方程式を用意し、数値計算により ADA の計算と組み合わせる。これにより、数値計算の前処理として数式処理を用いることができる。本論文では、この計算をハイブリッドモデリングと呼ぶ。ハイブリッドモデリングには、数式処理と数値計算を組み合わせることと、Lagrange 法と ADA を組み合わせることの二つの意味を持つ。本章では、ハイブリッドモデリングによるリンク系の運動方程式の導出を行う。また、平面 7 リンクマニピュレータを例にして、ハイブリッドモデリングの解法例を示していく。さらに、平面 7 リンクマニピュレータと浮遊ベースを含む三次元 6 リンクマニピュレータのシミュレーションを行い、順動力学計算が正しく行われているか確認する。最後に、Lagrange 法と ADA、ハイブリッドモデリングのシミュレーション時間を比較し、ハイブリッドモデリングの有用性を検証する。

### 4.1 リンク系の運動方程式

Fig. 4.1 に示すように、三つの AB A, B, C を接続した AB D を考える。ハンドル  $i, l$  に加わる力を  $\mathcal{F}_i^A, \mathcal{F}_l^C$  と、それぞれの加速度  $\dot{\mathcal{V}}_i^A, \dot{\mathcal{V}}_l^C$  との関係は

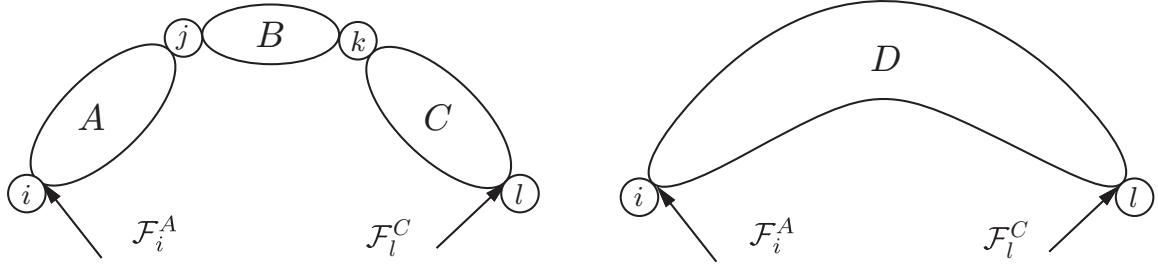


Fig. 4.1: Assembly of articulated bodies for hybrid modeling.

Eq. (2.4) より以下のように表される .

$$\begin{bmatrix} \dot{\mathcal{V}}_i^A \\ \dot{\mathcal{V}}_l^C \end{bmatrix} = \begin{bmatrix} \Phi_i^D & \Phi_{il}^D \\ \Phi_{li}^D & \Phi_l^D \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^A \\ \mathcal{F}_l^C \end{bmatrix} + \begin{bmatrix} \mathcal{B}_i^D \\ \mathcal{B}_l^D \end{bmatrix} \quad (4.1)$$

ここで , 式中の  $\Phi^D$  は  $D$  の IABI ,  $\mathcal{B}^D$  は  $D$  のバイアス加速度とする .

一方 , Fig. 4.1 を浮遊ベースを含んだ 3 リンクマニピュレータとして考え , 関節変位  $q_D = [q_j^T, q_k^T]^T \in \mathbb{R}^n$  を一般化座標とすると , このモデルの運動方程式は次式で表される . ただし ,  $q_j \in \mathbb{R}^{n_j}$  と  $q_k \in \mathbb{R}^{n_k}$  は関節  $j$  と関節  $k$  の変位とする .

$$\begin{bmatrix} \mathbf{M}_A & \mathbf{M}_{AD} \\ \mathbf{M}_{AD}^T & \mathbf{M}_D \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_i^A \\ \ddot{\mathbf{q}}_D \end{bmatrix} + \begin{bmatrix} \mathcal{C}_A \\ \mathbf{c}_D \end{bmatrix} + \begin{bmatrix} \mathcal{G}_A \\ \mathbf{g}_D \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau}_D \end{bmatrix} \quad (4.2)$$

式中の変数名は次の通りである .

$\mathbf{M}_A$	$\in \mathbb{R}^{6 \times 6}$	: $A$ におけるシステム全体の慣性行列
$\mathbf{M}_{AD}$	$\in \mathbb{R}^{6 \times n}$	: $A$ と $D$ の干渉慣性行列
$\mathbf{M}_D$	$\in \mathbb{R}^{n \times n}$	: $D$ の慣性行列
$\mathcal{C}_A$	$\in \mathbb{R}^6$	: $A$ のコリオリ遠心力項
$\mathbf{c}_D$	$\in \mathbb{R}^n$	: $D$ のコリオリ遠心力項
$\mathcal{G}_A$	$\in \mathbb{R}^6$	: $A$ の重力項
$\mathbf{g}_D$	$\in \mathbb{R}^n$	: $D$ の重力項
$\mathcal{V}_i^A$	$\in \mathbb{R}^6$	: 関節 $i$ の並進速度と角速度
$\mathbf{q}$	$\in \mathbb{R}^n$	: $D$ の関節変位
$\boldsymbol{\tau}_D$	$\in \mathbb{R}^n$	: $D$ の各関節を駆動する一般化力

また , 関節  $i$  と関節  $l$  の速度  $\mathcal{V}_i^A$  と  $\mathcal{V}_l^C$  は次式で表される .

$$\begin{bmatrix} \mathcal{V}_i^A \\ \mathcal{V}_l^C \end{bmatrix} = \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{J}_{li} & \mathbf{J}_{lD} \end{bmatrix} \begin{bmatrix} \mathcal{V}_i^A \\ \dot{\mathbf{q}}_D \end{bmatrix} \quad (4.3)$$

ここで,  $J_{li} \in \Re^{6 \times 6}$  は  $\mathcal{V}_i^A$  を  $\mathcal{V}_l^C$  に変換するヤコビ行列,  $J_{lD} \in \Re^{6 \times n}$  は  $q_D$  を  $\mathcal{V}_l^C$  に変換するヤコビ行列である。上式を時間微分すると次式を得る。

$$\begin{bmatrix} \dot{\mathcal{V}}_i^A \\ \dot{\mathcal{V}}_l^C \end{bmatrix} = \begin{bmatrix} U & \mathbf{0} \\ J_{li} & J_{lD} \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_i^A \\ \ddot{q}_D \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ J_{li} & J_{lD} \end{bmatrix} \begin{bmatrix} \mathcal{V}_i^A \\ \dot{q}_D \end{bmatrix} \quad (4.4)$$

また, 上式のヤコビ行列を用いて  $i$  と  $l$  に加わる力  $\mathcal{F}_i^A$  と  $\mathcal{F}_l^C$  を考えると, Eq. (4.2) は次式で表せる。

$$\begin{bmatrix} M_A & M_{AD} \\ M_{AD}^T & M_D \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_i^A \\ \ddot{q}_D \end{bmatrix} + \begin{bmatrix} \mathcal{C}_{\mathcal{A}} \\ c_D \end{bmatrix} + \begin{bmatrix} \mathcal{G}_{\mathcal{A}} \\ g_D \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \tau_D \end{bmatrix} + \begin{bmatrix} U & J_{li}^T \\ \mathbf{0} & J_{lD}^T \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^A \\ \mathcal{F}_l^C \end{bmatrix} \quad (4.5)$$

Eq. (4.5) から求めた  $[(\dot{\mathcal{V}}_i^A)^T, \ddot{q}_D^T]^T$  を Eq. (4.4) に代入すると次式が求まる。

$$\begin{bmatrix} \dot{\mathcal{V}}_i^A \\ \dot{\mathcal{V}}_l^C \end{bmatrix} = \begin{bmatrix} U & \mathbf{0} \\ J_{li} & J_{lD} \end{bmatrix} \begin{bmatrix} M_A & M_{AD} \\ M_{AD}^T & M_D \end{bmatrix}^{-1} \begin{bmatrix} U & J_{li}^T \\ \mathbf{0} & J_{lD}^T \end{bmatrix} \begin{bmatrix} \mathcal{F}_i^A \\ \mathcal{F}_l^C \end{bmatrix} \\ + \begin{bmatrix} U & \mathbf{0} \\ J_{li} & J_{lD} \end{bmatrix} \begin{bmatrix} M_A & M_{AD} \\ M_{AD}^T & M_D \end{bmatrix}^{-1} \left( \begin{bmatrix} \mathbf{0} \\ \tau_D \end{bmatrix} - \begin{bmatrix} \mathcal{C}_{\mathcal{A}} \\ c_D \end{bmatrix} - \begin{bmatrix} \mathcal{G}_{\mathcal{A}} \\ g_D \end{bmatrix} \right) + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ J_{li} & J_{lD} \end{bmatrix} \begin{bmatrix} \mathcal{V}_i^A \\ \dot{q}_D \end{bmatrix} \quad (4.6)$$

上式を Eq. (4.1) と比較すると,  $\Phi^D$  と  $\mathcal{B}^D$  は次のように求まる。

$$\begin{bmatrix} \Phi_i^D & \Phi_{il}^D \\ \Phi_{li}^D & \Phi_l^D \end{bmatrix} = \begin{bmatrix} U & \mathbf{0} \\ J_{li} & J_{lD} \end{bmatrix} \begin{bmatrix} M_A & M_{AD} \\ M_{AD}^T & M_D \end{bmatrix}^{-1} \begin{bmatrix} U & J_{li}^T \\ \mathbf{0} & J_{lD}^T \end{bmatrix} \quad (4.7)$$

$$\begin{bmatrix} \mathcal{B}_i^D \\ \mathcal{B}_l^D \end{bmatrix} = \begin{bmatrix} U & \mathbf{0} \\ J_{li} & J_{lD} \end{bmatrix} \begin{bmatrix} M_A & M_{AD} \\ M_{AD}^T & M_D \end{bmatrix}^{-1} \left( \begin{bmatrix} \mathbf{0} \\ \tau_D \end{bmatrix} - \begin{bmatrix} \mathcal{C}_{\mathcal{A}} \\ c_D \end{bmatrix} - \begin{bmatrix} \mathcal{G}_{\mathcal{A}} \\ g_D \end{bmatrix} \right) + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ J_{li} & J_{lD} \end{bmatrix} \begin{bmatrix} \mathcal{V}_i^A \\ \dot{q}_D \end{bmatrix} \quad (4.8)$$

上式より Eq. (4.5) の各行列成分とヤコビ行列から  $\Phi^D$ ,  $\mathcal{B}^D$  を求めることができる。つまり, 複数のリンクを接続した AB のパラメータを 1 回の assembly ステップで計算できるようになった。

次に disassembly ステップを考える。 $\mathcal{F}_l^C$  が既知であれば, Eq. (4.5) の二段目より,  $\ddot{q}_D$  が次式から求まる。

$$\ddot{q}_D = M_D^{-1} \left( \tau_D + J_{lD}^T \mathcal{F}_l^C - M_{AD}^T \dot{\mathcal{V}}_i^A - c_D - g_D \right) \quad (4.9)$$

disassembly ステップに関しても, 複数のリンクを接続した AB に対して 1 回の分解で計算できるようになった。

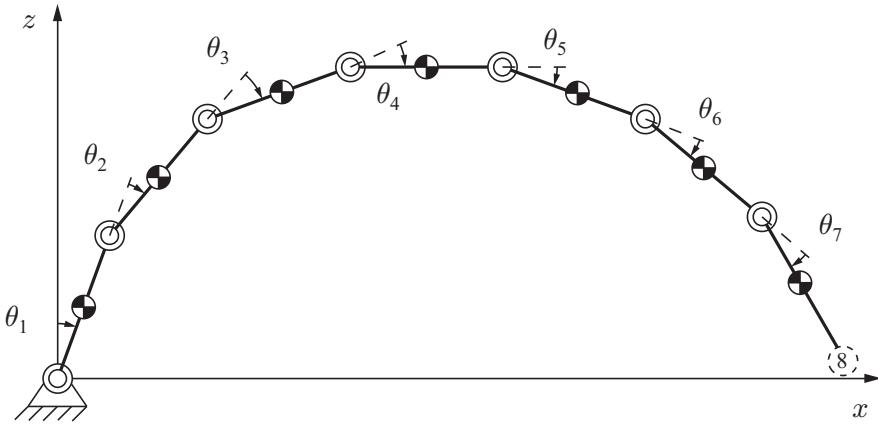


Fig. 4.2: A planar 7R manipulator.

## 4.2 Lagrange法とADA, ハイブリッドモデリングの比較

平面7リンクマニピュレータのモデル図をFig. 4.3に示す。このモデルに対して、Lagrange法とADA, ハイブリッドモデリングの三種類の方法に対する順動力学計算の解法を示しす。

### 4.2.1 Lagrange法

関節変位  $\mathbf{q} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7]^T$  を一般化座標とすると、このモデルの運動方程式は次式で表される。

$$\mathbf{M} \ddot{\mathbf{q}} + \mathbf{c} + \mathbf{g} = \boldsymbol{\tau} \quad (4.10)$$

ここで、式中の  $\mathbf{M} \in \Re^{7 \times 7}$  は慣性行列、 $\mathbf{c} \in \Re^7$  は非線形項、 $\mathbf{g} \in \Re^7$  は重力項、 $\boldsymbol{\tau} \in \Re^7$  は各関節を駆動する一般化力である。上式より、 $\ddot{\mathbf{q}}$  を求めると次のようになる。

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1} (\boldsymbol{\tau} - \mathbf{c} - \mathbf{g}) \quad (4.11)$$

この式を用いて順動力学計算を行う。ここで、 $\mathbf{M}^{-1}$  を求める際に  $O(n^3)$  の計算量となることに注意されたい。また、本論文では  $\mathbf{M}, \mathbf{c}, \mathbf{g}$  は事前に求めた数式を用いる。この数式を求めるために、ProPac[49] を用いる。ProPac は数式処理ソフト Mathematica[50] のパッケージであり、擬座標 (quasi-coordinate) を用いて Lagrange の運動方程式を導出する。

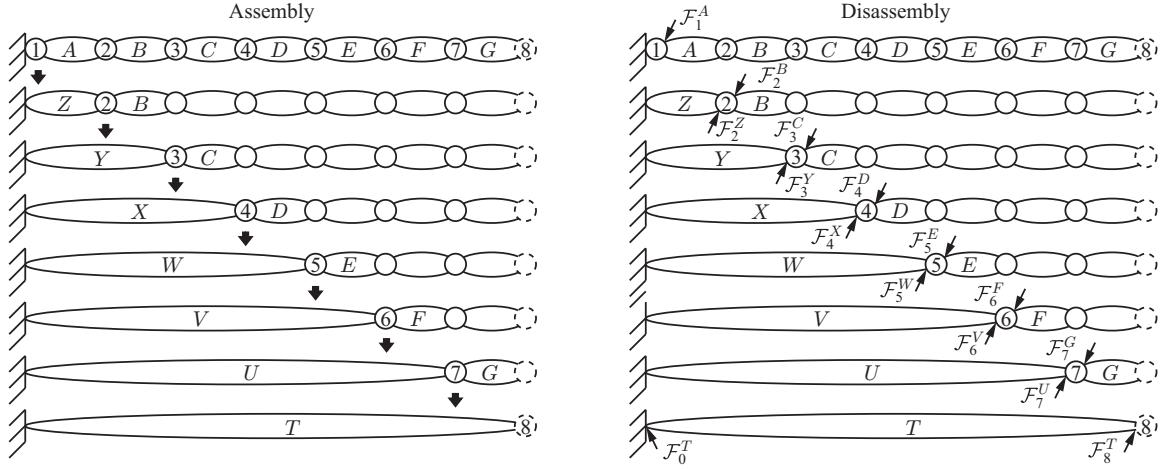


Fig. 4.3: Illustration of the assembly and disassembly phases based on the ADA.

#### 4.2.2 ADA

Fig. 4.3 に示した平面 7 リンクマニピュレータに対する ADA の計算ステップを Fig. 4.3 に示す。2.2 節で述べたように、ADA は二つのステップからなり、assembly ステップと disassembly ステップで隣り合う AB を一つずつ接続または分解していく。7 リンクの場合、接続と分解をそれぞれ 7 回ずつ行う。ADA の解法を具体的に示していく。

##### Assembly ステップ

1.  $A$  から  $G$  まで、各リンクの IABI  $\Phi^{AB}$  とバイアス加速度  $B^{AB}$  を求める ( $AB = A, B, \dots, G$ )。
2. Eq. (2.21) の関係を用いて、 $A$  から  $Z$  を求める。ここで、 $A$  と  $Z$  は固定ベースに接続されていることに注意されたい。
3. 同様に、 $Z$  と  $B$  から  $C$  を求める。
4. 同様の計算を繰り返し、最終的に  $T$  を求め、全体で一つの AB を作成する。

##### Disassembly ステップ

1.  $T$  の両側のハンドルに加わる力  $F_1^T, F_8^T$  を求める。ここで、固定ベースや先端の場合はハンドルから力が加わらないものとする。すなわち

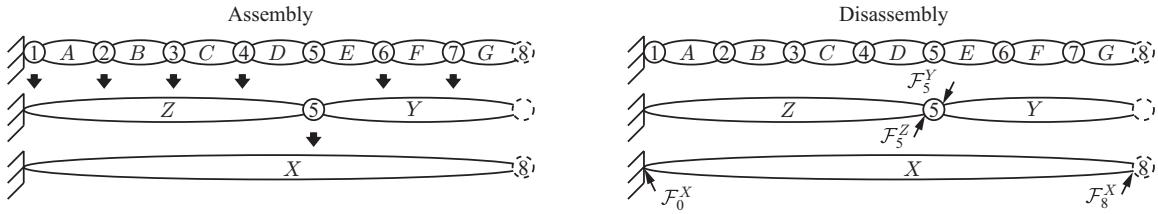


Fig. 4.4: Illustration of the assembly and disassembly phases based on the hybrid modeling.

ち ,  $\mathcal{F}_1^T = \mathcal{F}_8^T = 0$  となる .

2. Eq. (2.18) の関係から , 関節 7 の拘束力  $\lambda_7$  を求める .
3. Eq. (2.15) の関係から , 関節 7 から  $U, G$  に伝わる力  $\mathcal{F}_7^U, \mathcal{F}_7^G$  を求める .
4. Eqs. (2.11), (2.12) と Eq. (2.22) の関係から 関節 7 の加速度  $\ddot{\theta}_7$  を求める .  
ここで , 関節は 1 自由度の回転関節であり ,  $\ddot{q}_7 = \ddot{\theta}_7$  となる .
5. 「2.」から「4.」の計算を assembly ステップで接続した逆の順で行う .

### 4.2.3 ハイブリッドモデリング

Fig. 4.3 に示した平面 7 リンクマニピュレータに対するハイブリッドモデリングの計算ステップを Fig. 4.4 に示す . 複数の AB を接続した後の AB のパラメータを事前に数式で用意することで , ADA の計算ステップの最初と最後を省略することができる . 7 リンクの場合 , 接続と分解をそれぞれ 3 回ずつ行う . ハイブリッドモデリングの解法を具体的に示していく .

#### Assembly ステップ

1.  $A$  から  $G$  まで , 各リンクの IABI  $\Phi^{AB}$  とバイアス加速度  $B^{AB}$  を求める ( $AB = A, B, \dots, G$ ) .
2. Eqs. (4.7), (4.8) の関係を用いて ,  $Z$  と  $Y$  を求める .
3. Eq. (2.21) の関係を用いて ,  $Z$  と  $Y$  から  $X$  を求める .

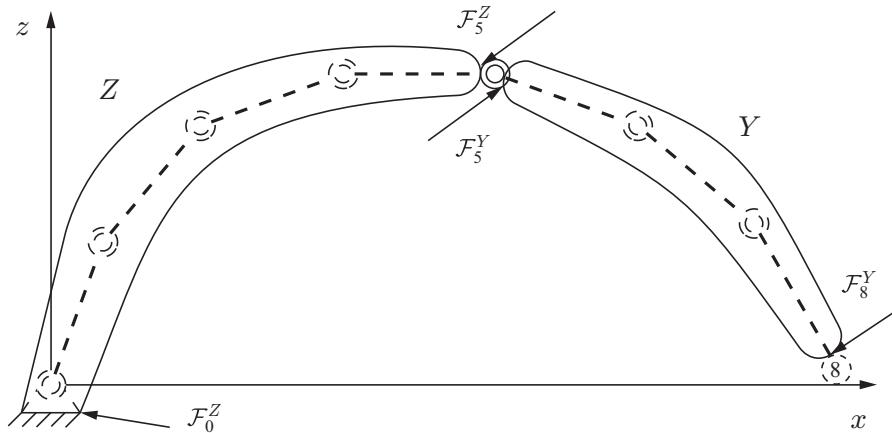


Fig. 4.5: Illustration of the hybrid modeling for the planar 7R manipulator.

### Disassembly ステップ

1.  $X$  の両側のハンドルに加わる力  $\mathcal{F}_1^X, \mathcal{F}_8^X$  を求める .
2. Eq. (2.18) の関係から , 関節 5 の拘束力  $\lambda_5$  を求める .
3. Eq. (2.15) の関係から , 関節 5 から  $Z, Y$  に伝わる力  $\mathcal{F}_5^Z, \mathcal{F}_5^Y$  を求める .
4. Eqs. (2.11), (2.12) と Eq. (2.22) の関係から 関節 5 の加速度  $\ddot{\theta}_5$  を求める .
5. Eq. (4.9) の関係から 残りの関節の加速度を求める .

ADA とハイブリッドモデリングの最大の違いは , assembly ステップの「2.」で  $Z$  と  $Y$  を求める部分と , disassembly ステップの「5.」で関節の加速度を求める部分である . 次節でこの部分の運動方程式の導出を行う .

## 4.3 ハイブリッドモデリングによる運動方程式の導出

Fig. 4.3 に示した平面 7 リンクマニピュレータに対するハイブリッドモデリングによる運動方程式の導出を行う . Fig. 4.5 に示すように根元の 4 リンクを接続した  $Z$  と , 手先の 3 リンクを接続した  $Y$  に別けて考える .

### 4.3.1 根元の 4 リンクの AB のパラメータの導出

Fig. 4.5 の根元の 4 リンクを平面 4 リンクマニピュレータとし , 固定ベースに接続された一つの AB  $Z$  として考える . 固定ベースから  $Z$  に伝わる力を

$\mathcal{F}_0^Z$ , 関節5からZに伝わる力を $\mathcal{F}_5^Z$ , 発生する加速度を $\dot{\mathcal{V}}_0^Z$ と $\dot{\mathcal{V}}_5^Z$ とすると, このモデルの運動方程式はEq. (4.1)より, 次式で表される.

$$\begin{bmatrix} \dot{\mathcal{V}}_0^Z \\ \dot{\mathcal{V}}_5^Z \end{bmatrix} = \begin{bmatrix} \Phi_0^Z & \Phi_{05}^Z \\ \Phi_{50}^Z & \Phi_5^Z \end{bmatrix} \begin{bmatrix} \mathcal{F}_0^Z \\ \mathcal{F}_5^Z \end{bmatrix} + \begin{bmatrix} \mathcal{B}_0^Z \\ \mathcal{B}_5^Z \end{bmatrix} \quad (4.12)$$

ここで, 式中の $\Phi^Z$ はZのIABI,  $\mathcal{B}^Z$ はZのバイアス加速度とする. 固定ベースの性質より $\Phi_0^Z = 0$ ,  $\Phi_{05}^Z = (\Phi_{50}^Z)^T = 0$ ,  $\mathcal{B}_0^Z = 0$ であるため, 上式は次のように表せる.

$$\dot{\mathcal{V}}_5^Z = \Phi_5^Z \mathcal{F}_5^Z + \mathcal{B}_5^Z \quad (4.13)$$

一方, 関節変位  $\mathbf{q}_Z = [\theta_1, \theta_2, \theta_3, \theta_4]^T$  を一般化座標とすると, このモデルの運動方程式は次式で表される.

$$M_Z \ddot{\mathbf{q}}_Z + \mathbf{c}_Z + \mathbf{g}_Z = \boldsymbol{\tau}_Z \quad (4.14)$$

ここで, 式中の  $M_Z \in \Re^{4 \times 4}$  は慣性行列,  $\mathbf{c}_Z \in \Re^4$  は非線形項,  $\mathbf{g}_Z \in \Re^4$  は重力項,  $\boldsymbol{\tau}_Z \in \Re^4$  は各関節を駆動する一般化力である. また, 関節5の速度  $\mathcal{V}_5^Z$  は次式で表される.

$$\mathcal{V}_5^Z = \mathbf{J}_{5Z} \dot{\mathbf{q}}_Z \quad (4.15)$$

ここで,  $\mathbf{J}_{5Z} \in \Re^{6 \times 4}$  は  $\dot{\mathbf{q}}_Z$  を  $\dot{\mathcal{V}}_5^Z$  に変換するヤコビ行列である. 上式を時間微分すると次式を得る.

$$\dot{\mathcal{V}}_5^Z = \mathbf{J}_{5Z} \ddot{\mathbf{q}}_Z + \dot{\mathbf{J}}_{5Z} \dot{\mathbf{q}}_Z \quad (4.16)$$

また, 上式のヤコビ行列を用いて5に加わる力  $\mathcal{F}_5^Z$  を考えると, Eq. (4.14) は次式で表せる.

$$M_Z \ddot{\mathbf{q}}_Z + \mathbf{c}_Z + \mathbf{g}_Z = \boldsymbol{\tau}_Z + \mathbf{J}_{5Z}^T \mathcal{F}_5^Z \quad (4.17)$$

上式から求めた  $\ddot{\mathbf{q}}_Z$  を Eq. (4.16) に代入すると次式が求まる.

$$\dot{\mathcal{V}}_5^Z = \mathbf{J}_{5Z} M_Z^{-1} (\boldsymbol{\tau}_Z + \mathbf{J}_{5Z}^T \mathcal{F}_5^Z - \mathbf{c}_Z - \mathbf{g}_Z) + \dot{\mathbf{J}}_{5Z} \dot{\mathbf{q}}_Z \quad (4.18)$$

上式を Eq. (4.13) と比較すると,  $\Phi_5^Z$  と  $\mathcal{B}_5^Z$  は次のように求まる.

$$\Phi_5^Z = \mathbf{J}_{5Z} M_Z^{-1} \mathbf{J}_{5Z}^T \quad (4.19)$$

$$\mathcal{B}_5^Z = \mathbf{J}_{5Z} M_Z^{-1} (\boldsymbol{\tau}_Z - \mathbf{c}_Z - \mathbf{g}_Z) + \dot{\mathbf{J}}_{5Z} \dot{\mathbf{q}}_Z \quad (4.20)$$

上式より  $M_Z$ ,  $\mathbf{c}_Z$ ,  $\mathbf{g}_Z$ ,  $\mathbf{J}_{5Z}$  から  $\Phi_5^Z$ ,  $\mathcal{B}_5^Z$  を求めることができる.

### 4.3.2 手先の3リンクのABのパラメータの導出

Fig. 4.5 の手先の3リンクを平面3リンクマニピュレータとし, 浮遊ベースを含んだ一つのAB Yとして考える。関節5と関節8からYに伝わる力を  $\mathcal{F}_5^Y$  と  $\mathcal{F}_8^Y$ , 発生する加速度を  $\dot{\mathcal{V}}_5^Y$  と  $\dot{\mathcal{V}}_8^Y$  とすると, このモデルの運動方程式は Eq. (4.1) より, 次式で表される。

$$\begin{bmatrix} \dot{\mathcal{V}}_5^Y \\ \dot{\mathcal{V}}_8^Y \end{bmatrix} = \begin{bmatrix} \Phi_5^Y & \Phi_{58}^Y \\ \Phi_{85}^Y & \Phi_8^Y \end{bmatrix} \begin{bmatrix} \mathcal{F}_5^Y \\ \mathcal{F}_8^Y \end{bmatrix} + \begin{bmatrix} \mathcal{B}_5^Y \\ \mathcal{B}_8^Y \end{bmatrix} \quad (4.21)$$

ここで, 式中の  $\Phi^Y$  は Y の IABI,  $\mathcal{B}^Y$  は Y のバイアス加速度である。一方, 関節変位  $q_Y = [\theta_6, \theta_7]^T$  を一般化座標とすると, このモデルの運動方程式は次式で表される。

$$\begin{bmatrix} M_5 & M_{5Y} \\ M_{5Y}^T & M_Y \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_5^Y \\ \ddot{q}_Y \end{bmatrix} + \begin{bmatrix} \mathcal{C}_5 \\ c_Y \end{bmatrix} + \begin{bmatrix} \mathcal{G}_5 \\ g_Y \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \tau_Y \end{bmatrix} \quad (4.22)$$

式中の変数名は次の通りである。

$M_5$	$\in \Re^{6 \times 6}$	: リンク5におけるシステム全体の慣性行列
$M_{5Y}$	$\in \Re^{6 \times 2}$	: リンク5とYの干渉慣性行列
$M_Y$	$\in \Re^{2 \times 2}$	: Yの慣性行列
$\mathcal{C}_5$	$\in \Re^6$	: リンク5のコリオリ遠心力項
$c_Y$	$\in \Re^2$	: Yのコリオリ遠心力項
$\mathcal{G}_5$	$\in \Re^6$	: リンク5の重力項
$g_Y$	$\in \Re^2$	: Yの重力項
$\mathcal{V}_5^Y$	$\in \Re^6$	: 関節5の並進速度と角速度
$q$	$\in \Re^2$	: Yの関節変位
$\tau_Y$	$\in \Re^2$	: 各関節を駆動する一般化力

また, 関節5と関節8の速度  $\mathcal{V}_5^Y$  と  $\mathcal{V}_8^Y$  は次式で表される。

$$\begin{bmatrix} \mathcal{V}_5^Y \\ \mathcal{V}_8^Y \end{bmatrix} = \begin{bmatrix} U & \mathbf{0} \\ J_{85} & J_{8Y} \end{bmatrix} \begin{bmatrix} \mathcal{V}_5^Y \\ q_Y \end{bmatrix} \quad (4.23)$$

ここで,  $J_{85} \in \Re^{6 \times 6}$  は  $\mathcal{V}_5^Y$  を  $\mathcal{V}_8^Y$  に変換するヤコビ行列,  $J_{8Y} \in \Re^{6 \times 2}$  は  $q_Y$  を  $\mathcal{V}_8^Y$  に変換するヤコビ行列である。上式を時間微分すると次式を得る。

$$\begin{bmatrix} \dot{\mathcal{V}}_5^Y \\ \dot{\mathcal{V}}_8^Y \end{bmatrix} = \begin{bmatrix} U & \mathbf{0} \\ J_{85} & J_{8Y} \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_5^Y \\ \ddot{q}_Y \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \dot{J}_{85} & \dot{J}_{8Y} \end{bmatrix} \begin{bmatrix} \mathcal{V}_5^Y \\ \dot{q}_Y \end{bmatrix} \quad (4.24)$$

また、上式のヤコビ行列を用いて<sub>5</sub>と<sub>8</sub>に加わる力 $\mathcal{F}_5^Y$ と $\mathcal{F}_8^Y$ を考えると、Eq. (4.22)は次式で表せる。

$$\begin{bmatrix} M_5 & M_{5Y} \\ M_{5Y}^T & M_Y \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_5^Y \\ \ddot{\mathbf{q}}_Y \end{bmatrix} + \begin{bmatrix} \mathcal{C}_5 \\ \mathbf{c}_Y \end{bmatrix} + \begin{bmatrix} \mathcal{G}_5 \\ \mathbf{g}_Y \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau}_Y \end{bmatrix} + \begin{bmatrix} \mathbf{U} & \mathbf{J}_{85}^T \\ \mathbf{0} & \mathbf{J}_{8Y}^T \end{bmatrix} \begin{bmatrix} \mathcal{F}_5^Y \\ \mathcal{F}_8^Y \end{bmatrix} \quad (4.25)$$

上式から求めた $[\dot{\mathcal{V}}_5^Y]^T, \ddot{\mathbf{q}}_Z^T]^T$ をEq. (4.24)に代入すると次式が求まる。

$$\begin{aligned} \begin{bmatrix} \dot{\mathcal{V}}_5^Y \\ \dot{\mathcal{V}}_8^Y \end{bmatrix} &= \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{J}_{85} & \mathbf{J}_{8Y} \end{bmatrix} \begin{bmatrix} M_5 & M_{5Y} \\ M_{5Y}^T & M_Y \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{U} & \mathbf{J}_{85}^T \\ \mathbf{0} & \mathbf{J}_{8Y}^T \end{bmatrix} \begin{bmatrix} \mathcal{F}_5^Y \\ \mathcal{F}_8^Y \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{J}_{85} & \mathbf{J}_{8Y} \end{bmatrix} \begin{bmatrix} M_5 & M_{5Y} \\ M_{5Y}^T & M_Y \end{bmatrix}^{-1} \left( \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau}_Y \end{bmatrix} - \begin{bmatrix} \mathcal{C}_5 \\ \mathbf{c}_Y \end{bmatrix} - \begin{bmatrix} \mathcal{G}_5 \\ \mathbf{g}_Y \end{bmatrix} \right) + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{j}_{85} & \mathbf{j}_{8Y} \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_5^Y \\ \dot{\mathbf{q}}_Y \end{bmatrix} \end{aligned} \quad (4.26)$$

上式をEq. (4.21)と比較すると、 $\Phi^Y$ と $\mathcal{B}^Y$ は次のように求まる。

$$\begin{bmatrix} \Phi_5^Y & \Phi_{58}^Y \\ \Phi_{85}^Y & \Phi_8^Y \end{bmatrix} = \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{J}_{85} & \mathbf{J}_{8Y} \end{bmatrix} \begin{bmatrix} M_5 & M_{5Y} \\ M_{5Y}^T & M_Y \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{U} & \mathbf{J}_{85}^T \\ \mathbf{0} & \mathbf{J}_{8Y}^T \end{bmatrix} \quad (4.27)$$

$$\begin{bmatrix} \mathcal{B}_5^Y \\ \mathcal{B}_8^Y \end{bmatrix} = \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{J}_{85} & \mathbf{J}_{8Y} \end{bmatrix} \begin{bmatrix} M_5 & M_{5Y} \\ M_{5Y}^T & M_Y \end{bmatrix}^{-1} \left( \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau}_Y \end{bmatrix} - \begin{bmatrix} \mathcal{C}_5 \\ \mathbf{c}_Y \end{bmatrix} - \begin{bmatrix} \mathcal{G}_5 \\ \mathbf{g}_Y \end{bmatrix} \right) + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{j}_{85} & \mathbf{j}_{8Y} \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_5^Y \\ \dot{\mathbf{q}}_Y \end{bmatrix} \quad (4.28)$$

上式よりEq. (4.25)の各行列成分とヤコビ行列から $\Phi^Y, \mathcal{B}^Y$ を求めることができる。

### 4.3.3 関節の加速度の導出

Eq. (2.18)より、関節<sub>5</sub>の拘束力 $\lambda \in \Re^5$ は次のように求まる。ただし、固定ベースや先端の場合はハンドルから力が加わらないため $\mathcal{F}_0^Z = \mathcal{F}_8^Y = \mathbf{0}$ となる。

$$\lambda_5 = (\mathbf{T}_5^T (\Phi_5^Z + \Phi_5^Y) \mathbf{T}_5)^{-1} \mathbf{T}_5^T (\dot{\mathbf{S}}_5 \dot{\theta}_5 + \mathcal{B}_5^Z - \mathcal{B}_5^Y - (\Phi_5^Z + \Phi_5^Y) \mathbf{T}_{a5} n_5) \quad (4.29)$$

ここで、式中の $\mathbf{T}_5 \in \Re^{6 \times 5}$ は関節<sub>5</sub>の拘束方向を表す行列、 $\mathbf{S}_5 \in \Re^6$ は関節<sub>5</sub>の可動方向を表す行列、 $\mathbf{T}_{a5} \in \Re^6$ は関節<sub>5</sub>の駆動方向を表す行列、 $n_5$ は関節<sub>5</sub>の駆動トルクである。関節<sub>5</sub>から<sub>Y</sub>に加わる力 $\mathcal{F}_5^Y$ はEq. (2.15)より次のように求まる。

$$\mathcal{F}_5^Y = \mathbf{T}_{a5} n_5 + \mathbf{T}_5 \lambda_5 \quad (4.30)$$

$\mathcal{F}_5^Y$  と  $\mathcal{F}_8^Y$  を Eqs. (4.12), (4.21) に代入することで,  $\dot{\mathcal{V}}_5^Y$  と  $\dot{\mathcal{V}}_5^Z$  が求まる . よって , 関節 5 の角加速度  $\ddot{\theta}_5$  は Eq. (2.22) より次式から求まる .

$$\ddot{\theta}_5 = \mathbf{T}_{a5}^T \left( \dot{\mathcal{V}}_5^Y - \dot{\mathcal{V}}_5^Z - \dot{\mathbf{S}}_5 \dot{\theta}_5 \right) \quad (4.31)$$

また , Eq. (4.17) より ,  $\ddot{q}_Z$  が次式から求まる .

$$\ddot{q}_Z = \mathbf{M}_Z^{-1} \left( \boldsymbol{\tau}_Z + \mathbf{J}_{5Z}^T \mathcal{F}_5^Y - \mathbf{c}_Z - \mathbf{g}_Z \right) \quad (4.32)$$

さらに , Eq. (4.25) の二段目より ,  $\ddot{q}_Y$  が次式から求まる .

$$\ddot{q}_Y = \mathbf{M}_Y^{-1} \left( \boldsymbol{\tau}_Y + \mathbf{J}_{8Y}^T \mathcal{F}_8^Y - \mathbf{M}_{5Y}^T \dot{\mathcal{V}}_5^Y - \mathbf{c}_Y - \mathbf{g}_Y \right) \quad (4.33)$$

## 4.4 順動力学計算の確認

### 4.4.1 固定ベースに接続された平面7リンクマニピュレータ

Fig. 4.3 に示した平面7リンクマニピュレータのシミュレーションを行った . 順動力学計算では , Fig. 4.4 に示したように , 根元の4リンクと手先の3リンクに別けたハイブリッドモデリングを行った . 各リンクの長さは  $l = 1.0 \text{ m}$  , 質量は  $m = 1.0 \text{ kg}$  , 重心回りの慣性テンソルは  $\bar{\mathbf{I}}_C = \text{diag}(ml^2/12, ml^2/12, 0) \text{ kg} \cdot \text{m}^2$  とした . また , 重心位置は各リンクの中心に設定した .  $\theta_1^{ini} = 90 \text{ deg}$  ,  $\theta_2^{ini} = \theta_3^{ini} = \theta_4^{ini} = \theta_5^{ini} = \theta_6^{ini} = \theta_7^{ini} = 0 \text{ deg}$  の初期角度から初速および制御入力なしの条件で , 自由落下するシミュレーションを行った . シミュレーション時間を 3 s , 積分の刻み幅を 1 ms , 重力加速度を  $z$  軸方向に  $-9.81 \text{ m/s}^2$  とした .

運動の様子を Fig. 4.6 にスナップショットで示す . また , シミュレーション結果を Fig. 4.7 に示す . Fig. 4.7 (a) は関節角度 , (b) は関節速度 , (c) は関節加速度 , (d) は関節トルク , (e) は力学的エネルギーである . Fig. 4.6 , Figs. 4.7 (a)–(c) より , 設定した初期状態から重力の影響を受け , 自由落下する様子が確認できる . また , Figs. 4.7 (d) , (e) より , 制御入力を与えない状態で , 位置エネルギー (P.E.) と運動エネルギー (K.E.) の総和が一定となり , 力学的エネルギー保存則が確認できた . よって , 正しく順動力学計算が行えていると判断できる .

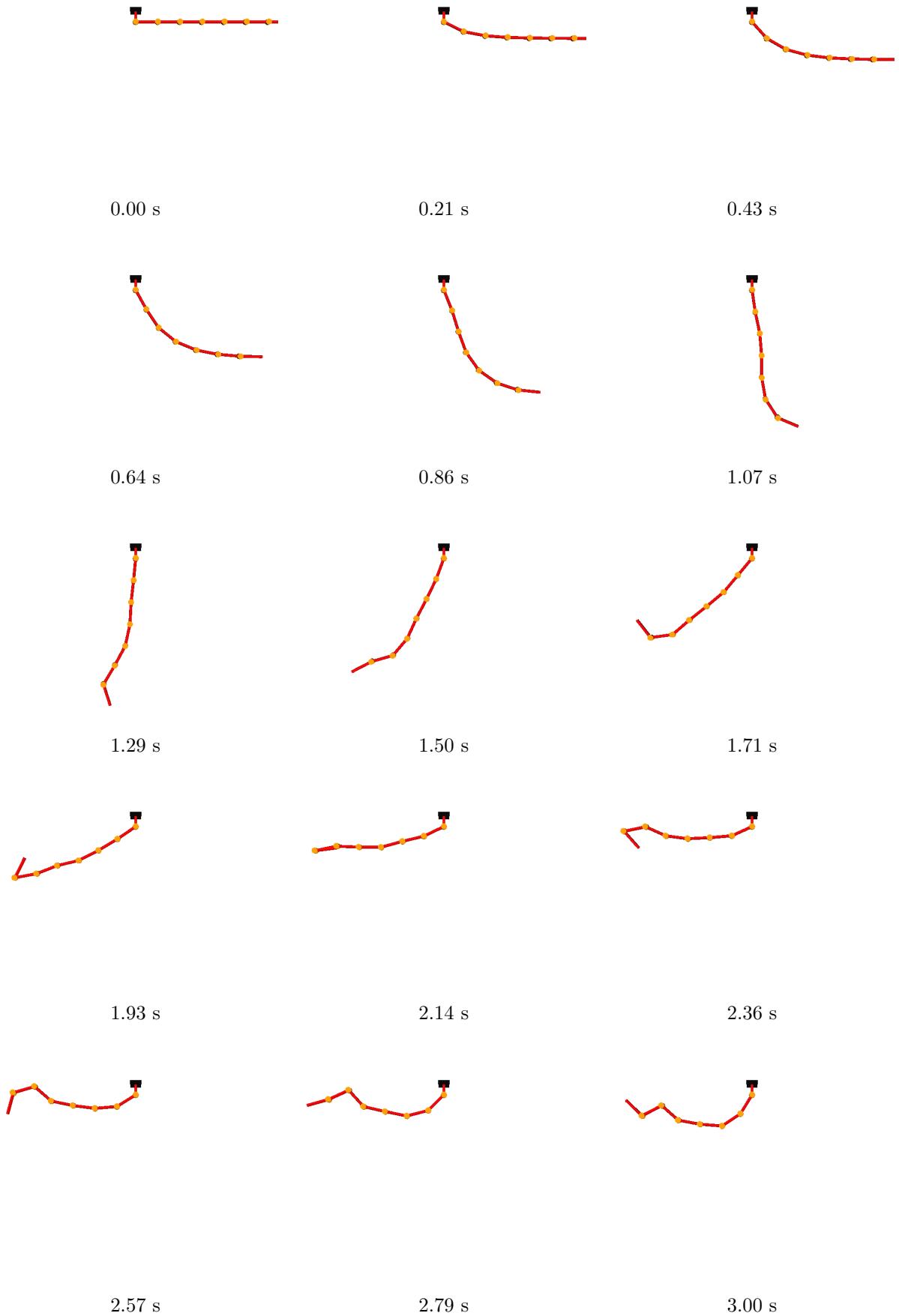


Fig. 4.6: Snapshots from simulation of the planar 7R manipulator with fixed base.

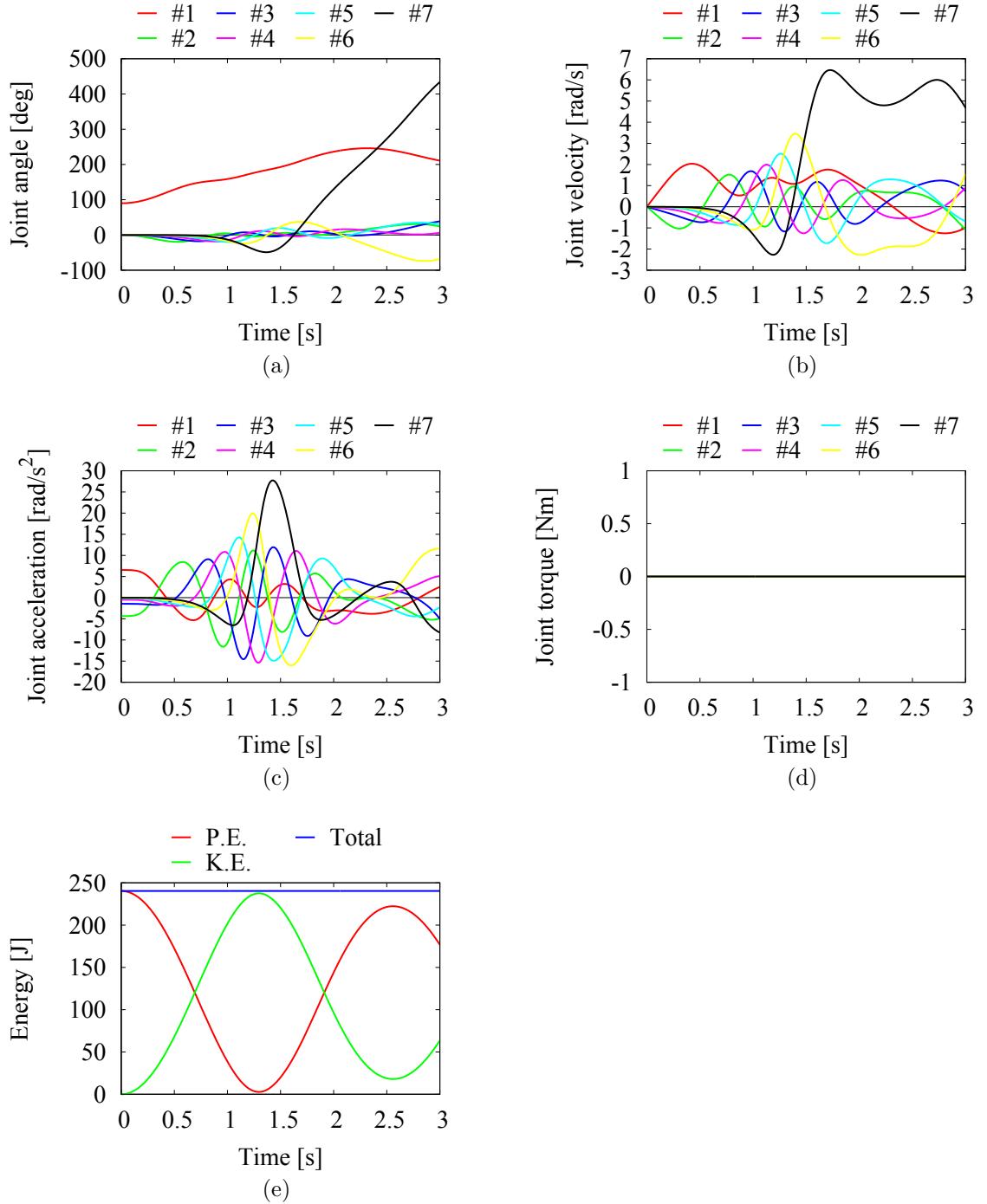


Fig. 4.7: Simulation results of the planar 7R manipulator with fixed base: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy.

Table 4.1: Initial state of each joints.

$i$	1	2	3	4	5
$\theta_i^{ini}$ [deg]	0	90	0	90	0
$\dot{\theta}_i^{ini}$ [rad/s]	-0.83	0.86	2.01	-0.44	-0.84

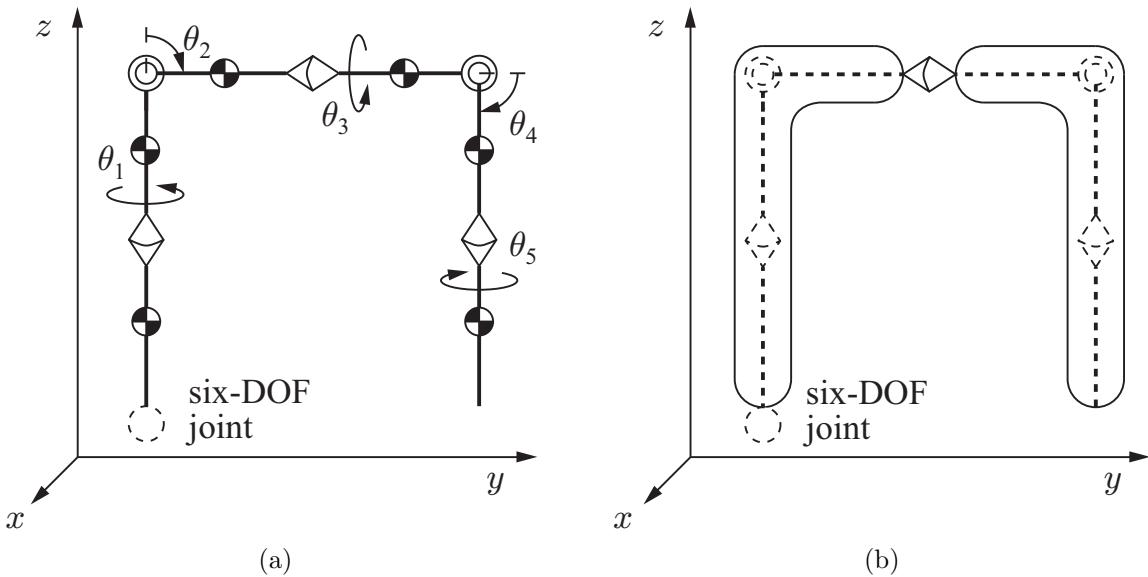


Fig. 4.8: A 3D model for the hybrid modeling with free base.

#### 4.4.2 浮遊ベースを含む三次元6リンクマニピュレータ

Fig. 4.8 に三次元の浮遊ベースのモデルを示す。このモデルは、根元に 6DOF 関節を設定した三次元の 6 リンクマニピュレータである。Fig. 4.8 (b) のように、根元の 3 リンクと手先の 3 リンクに別けてハイブリッドモデリングを行った。慣性座標系  $\{0\}$  からベース座標系  $\{b\}$  までの距離は第3章と同様に  $r$  とする。リンクの長さは  $l = 1 \text{ m}$ 、質量は  $m = 1 \text{ kg}$ 、剛体の慣性テンソルは  $\bar{I}_C = \text{diag}(ml^2/12, ml^2/12, 0) \text{ kg} \cdot \text{m}$  とした。また、重心位置は各リンクの中心に設定した。第1リンクが  $z_0$  軸に対して平行、 $x_0$  軸と  $y_0$  軸に対して垂直な状態を 6 自由度関節の初期角度、 $r = 0$  を初期位置としシミュレーションを行った。また、6 自由度関節の初期角速度は、重心の位置と速度に影響を与えないように  $\omega_{0x}^{ini} = -1.01 \text{ rad/s}$ ,  $\omega_{0y}^{ini} = -0.61 \text{ rad/s}$ ,  $\omega_{0z}^{ini} = 0.00 \text{ rad/s}$ ,  $v_{0x}^{ini} = 0.96 \text{ m/s}$ ,  $v_{0y}^{ini} = -1.18 \text{ m/s}$ ,  $v_{0z}^{ini} = 0.24 \text{ m/s}$  と設定した。回転関節の初期角度と初期角速度は Table 4.1 に示す通りである。シミュレーション時間は 10 s、積分の刻み幅を 1 ms とした。

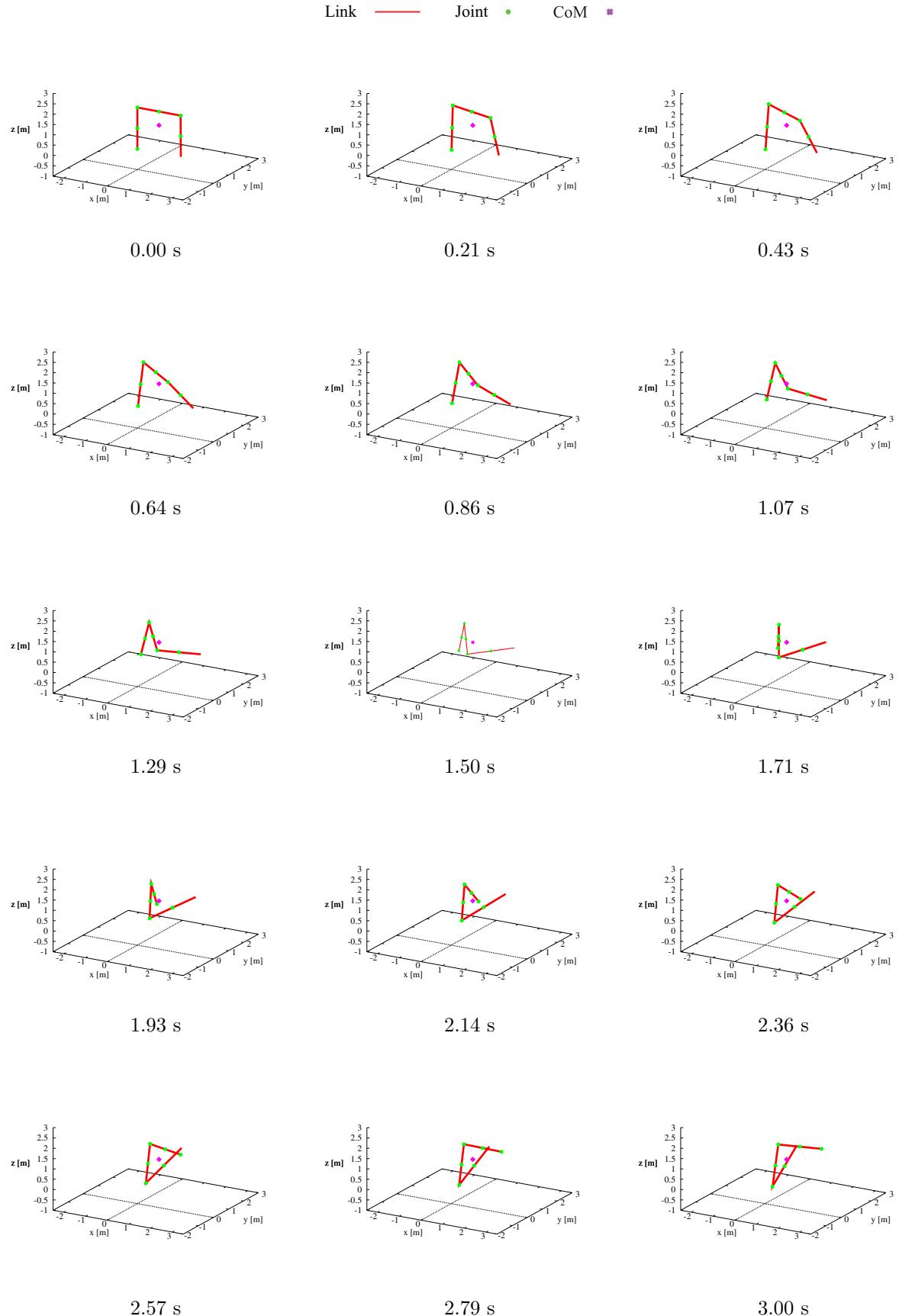


Fig. 4.9: Snapshots from simulation of the 3D 5R manipulator with free base.

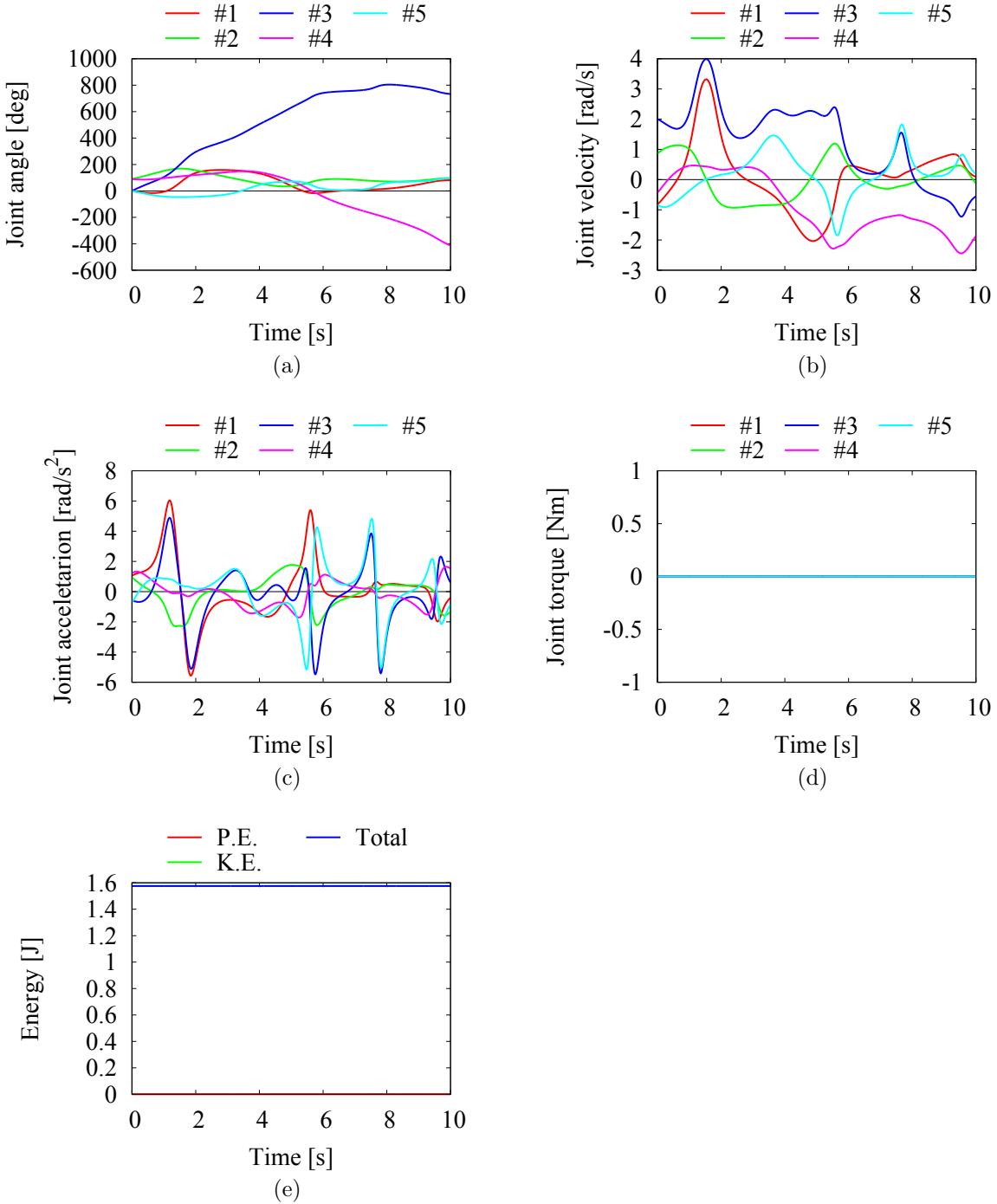


Fig. 4.10: Simulation results of the 3D 5R manipulator with free base: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy.

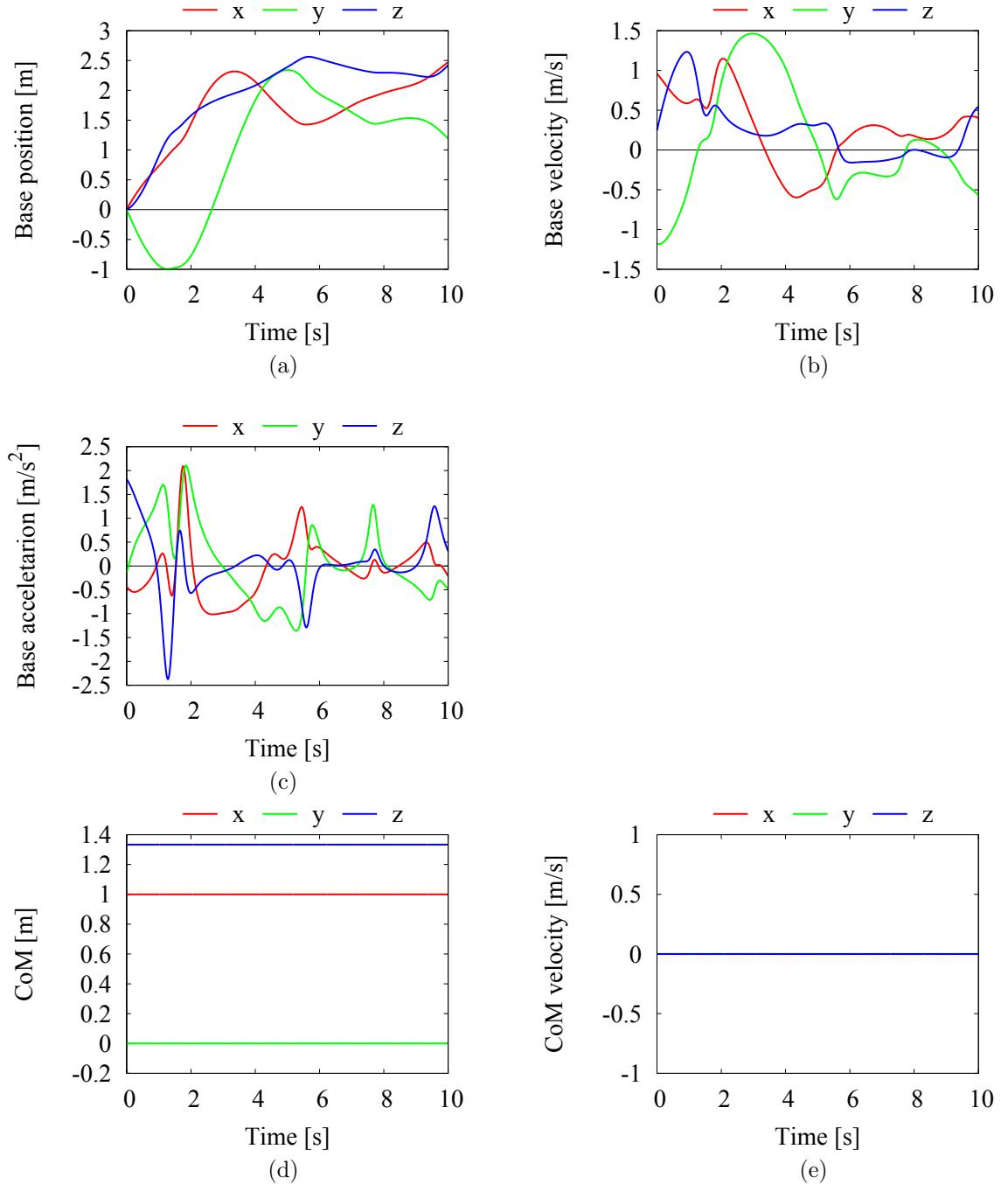


Fig. 4.11: Simulation results of the 3D 5R manipulator with free base: (a) base position, (b) base velocity, (c) base acceleration, (d) CoM position and (e) CoM velocity.

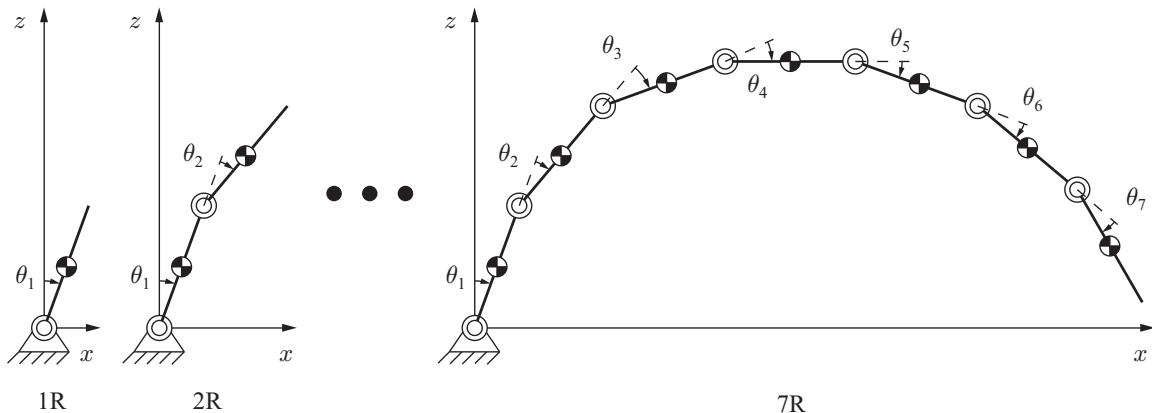


Fig. 4.12: Planar models for the hybrid modeling with fixed base.

運動の様子を Fig. 4.9 にスナップショットで示す。また、シミュレーション結果を Figs. 4.10, 4.11 に示す。Fig. 4.10 (a) は関節角度、(b) は関節角速度、(c) は関節加速度、(d) は関節トルク、(e) は力学的エネルギーである。Fig. 4.9, Figs. 4.10 (a)–(c) より、設定した初期状態からマニピュレータが三次元運動を行っている様子が確認できる。また、Figs. 4.10 (d), (e) より、制御を与えていない状態で、位置エネルギー (P.E.) と運動エネルギー (K.E.) の総和が一定となり、力学的エネルギー保存則が確認できた。

Fig. 4.11 (a) は 6 自由度関節で接続されたベース部の位置、(b) はベース部の速度、(c) はベース部の加速度、(d) と (e) は全体の重心位置と重心速度である。Fig. 4.9, Figs. 4.11 (a)–(c) より、ベース部が三次元運動を行っている様子が確認できる。Figs. 4.11 (d), (e) より、重心位置が一定で重心速度がゼロとなり、運動量保存則が確認できた。よって、正しく順動力学計算が行えていると判断できる。

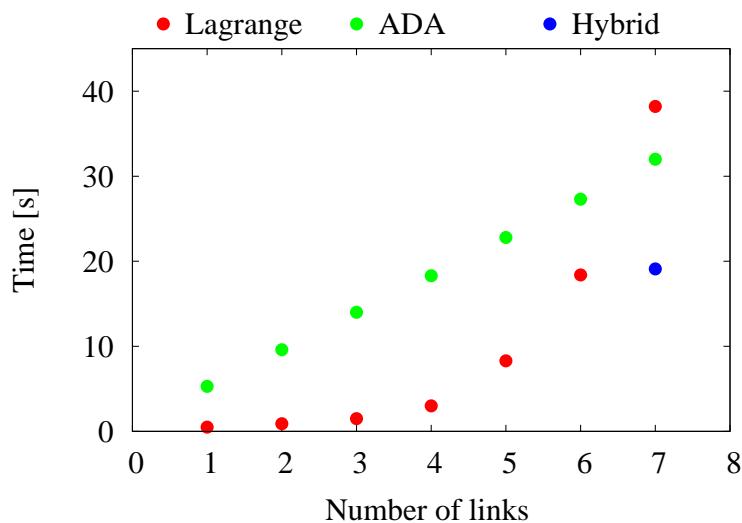
## 4.5 有用性の検証

### 4.5.1 固定ベースに接続された二次元モデル

ハイブリッドモデリングの有用性を検証するために、Lagrange 法と ADA、ハイブリッドモデリングの計算時間の比較を行う。第 4.2 節と第 4.3 節で述べた方法で Lagrange 法と ADA、ハイブリッドモデリングの計算を行う。比較するシミュレーションモデルを Fig. 4.12 に示す。このモデルは、固定ベースに接続された二次元モデルであり、1 リンクから 7 リンクのマニピュレー

Table 4.2: Computation time of the planar  $nR$  manipulator with fixed base [s].

	1 link (1 DOF)	2 link (2 DOF)	3 link (3 DOF)	4 link (4 DOF)	5 link (5 DOF)	6 link (6 DOF)	7 link (7 DOF)
Lagrange	0.5	0.9	1.5	3.0	8.3	18.4	38.2
ADA	5.3	9.6	14.0	18.3	22.8	27.3	32.0
Hybrid	—	—	—	—	—	—	19.1

Fig. 4.13: Computation time of the planar  $nR$  manipulator with fixed base.

タである。ハイブリッドモデリングではFig. 4.4のように、根元の4リンクと手先の3リンクの運動方程式をあらかじめ用意しておき、計算する。シミュレーション時間は10 s、刻み幅は1 msとし、3.16 GHz Intel CPU・メモリ4 GBの計算機を用いた。また、初期姿勢は第1関節を90 deg、他の関節を0 degとし、初速および制御入力なしの条件で、自由落下するシミュレーションを行った。シミュレーション時間をTable 4.3, Fig. 4.16に示す。

Lagrange法は計算量が $O(n^3)$ 、ADAは計算量が $O(n)$ となるアルゴリズムである。グラフから分かるように、二つの方法でこのような傾向が確認できた。また、6リンクまではLagrange法の方が速いが、7リンクでは、ADAの方が速くなる。一般的な人型ロボットの自由度は、HOAP-2が25自由度、HRP-4が34自由度、ASIMOが26自由度であり、このような大自由度系を扱う際には、ADAの方が効率的である。

ハイブリッドモデリングを行うことで、Lagrange法やADAよりも7リンクの計算時間が短くなることが確認できた。Lagrange法と比較して50 %、

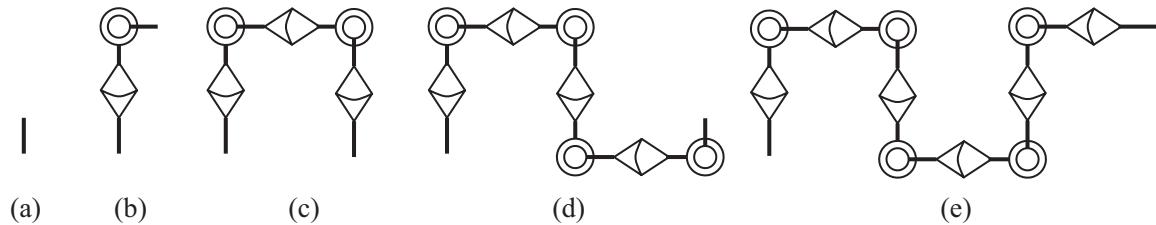


Fig. 4.14: 3D models for the hybrid modeling with free base.

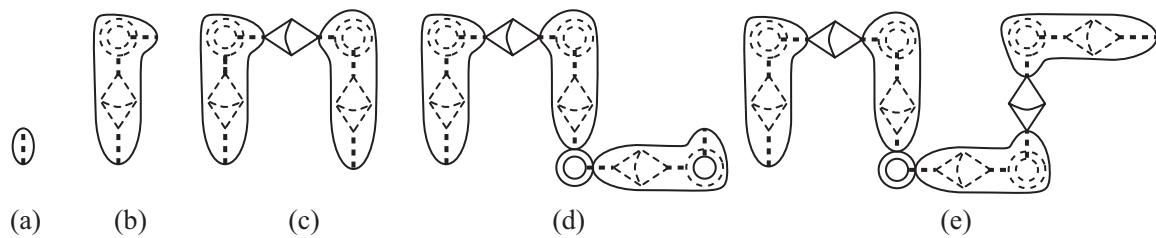


Fig. 4.15: Illustration of the hybrid modeling. We prepare the equation of motion of a three link manipulator except (a). (a) is a single rigid body.

ADA と比較して 60 % となった .

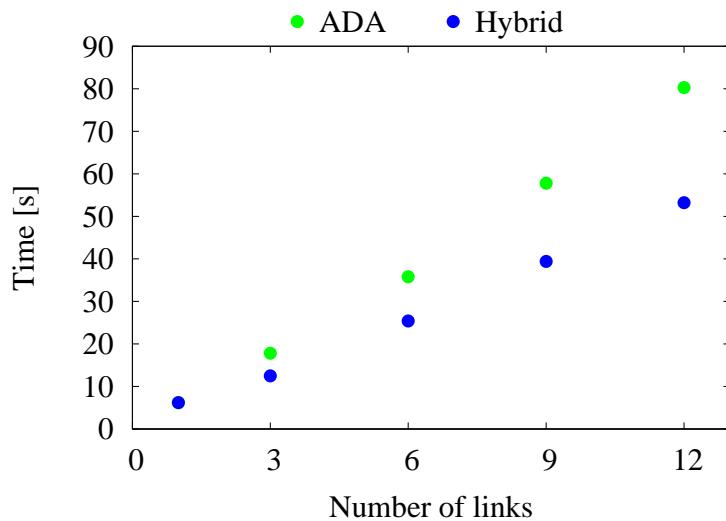
#### 4.5.2 浮遊ベースを含んだ三次元モデル

前節で扱ったモデルは固定ベースに接続された二次元のモデルである . 実用性を考えると , 浮遊ベースを含んだ三次元モデルでの比較を行う必要がある . また , ハイブリッドモデリングに関しては , 7リンクマニピュレータの計算時間のみを比較した . そのため , モデルの自由度の拡大に伴う , ハイブリッドモデリングの計算時間の変化も確認する必要がある . そこで , 浮遊ベースを含んだ三次元モデルに対して , ADA とハイブリッドモデリングの計算時間を比較する .

比較するシミュレーションモデルを Fig. 4.14 に示す . このモデルは , 浮遊ベースを含む三次元モデルであり , (a) から (e) はそれぞれ 1 リンク , 3 リンク , 6 リンク , 9 リンク , 12 リンクのマニピュレータである . ハイブリッドモデリングでは Fig. 4.15 のように , 3 リンクごとの数式をあらかじめ用意し , 計算する . シミュレーション時間は 10 s , 刻み幅は 1 ms とし , 3.16 GHz Intel CPU · メモリ 4 GB の計算機を用いた . また , 制御入力は与えず , ベース部  $x$  ,  $y$  ,  $z$  軸回りに初速を 1 rad/s 与え , シミュレーションを行った . シミュレーション時間を Table 4.3 , Fig. 4.16 に示す .

Table 4.3: Computation time of the 3D  $nR$  manipulator with free base [s].

	1 link (6 DOF)	3 link (8 DOF)	6 link (11 DOF)	9 link (14 DOF)	12 link (17 DOF)
Numeric (ADA)	6.2	17.8	35.8	57.8	80.3
Hybrid	6.2	12.5	25.4	39.4	53.2

Fig. 4.16: Computation time of the 3D  $nR$  manipulator with free base.

二次元のモデルと同様に，ADAは計算量が  $O(n)$  となることが確認できた。また，ハイブリッドモデリングも  $O(n)$  の傾向が確認できた。さらに，ハイブリッドモデリングを用いることで，ADAよりも計算時間が短くなった。浮遊ベースを含む三次元12リンクマニピュレータの計算時間を比較すると，ハイブリッドモデリングは，ADAの約66%となった。

## 4.6 本章のまとめ

ADAを用いた数値・数式ハイブリッドモデリングを行い，リンク系の運動方程式を導出した。また，固定ベースで接続された二次元モデルと浮遊ベースを含んだ三次元モデルのシミュレーションを行い，計算時間を比較した。その結果，ハイブリッドモデリングは，二次元モデルではADAの約60%，三次元モデルではADAの約66%となり，計算時間を短縮できた。

今後の課題は，用意する数式の自由度による計算時間の違いを検証し，あらかじめ用意しておく数式の自由度や機構を決定することである。



# 第5章 マルチボディダイナミクスシミュレータ

前章までに述べてきた，Assembly-Disassembly Algorithm (ADA) を用いた数値・数式ハイブリッドモデリングを行う順動力学エンジンを開発し，マルチボディダイナミクスシミュレータへ適用する．本章では，シミュレータのプログラムの流れやファイル構成，シミュレーションの例などを示していく．

## 5.1 シミュレータのプログラムの流れ

シミュレーションのプログラムの流れを Figs. 5.1, 5.2 に示す．Fig. 5.1 (a) に示すように，プログラムを実行するとまず，シミュレーションの初期化を行う．ここでは，シミュレーションの初期状態の読み込みやフラグの初期化，シミュレーション時間の読み込みなどを行う．次に，シミュレーションを行うかの確認画面が表示され，シミュレーションを行う場合は，MaTX[51]–[53] の Ode45 関数を用いて，常微分方程式を解く．シミュレーションを行わない場合は，一つ前のシミュレーション結果のデータが読み込まれる．その後，出力ファイルの追加計算を行い，グラフ表示やアニメーション表示などをを行う．

Ode45 関数は次のように用いられる．

$$\{\mathbf{T}, \mathbf{X}, \mathbf{U}\} = \text{Ode45}(t_0, t_1, \mathbf{x}_0, \text{diffEqs}, \text{linkEqs}, h) \quad (5.1)$$

ここで，引数の  $t_0$  は初期時刻， $t_1$  は終了時刻， $\mathbf{x}_0$  は初期状態， $h$  は積分の刻み幅である．diffEqs は時刻  $t$ ，状態  $\mathbf{x}$ ，制御入力  $u$  から状態の微分  $\dot{\mathbf{x}}$  を計算する関数であり，次のように表す．

$$\{\dot{\mathbf{x}}, t, \mathbf{x}, u\} = \text{diffEqs}(t, \mathbf{x}, u) \quad (5.2)$$

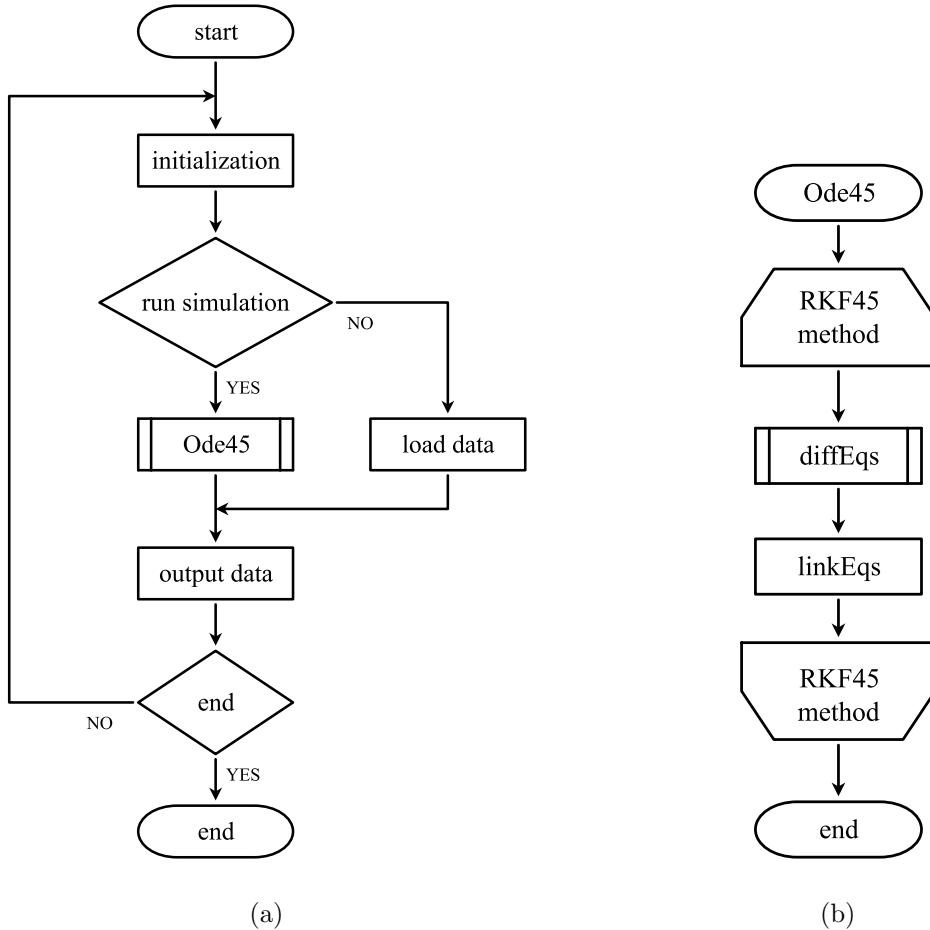


Fig. 5.1: Program flowchart of the simulator: (a) overview of the simulator and (b) function of the Ode45 of MaTX.

`linkEqs` は時刻  $t$  , 状態  $x$  から制御入力  $u$  を計算する関数であり , 次のように表す .

$$\{u, t, x\} = \text{linkEqs}(t, x) \quad (5.3)$$

Ode45 関数では , `linkEqs` の計算結果  $u$  を関数 `diffEqs` に渡しながら , 常微分方程式を RKF45 法を用いて刻み幅  $h$  で解き , 時間の系列  $T$  と状態の時系列  $X$  , 制御入力の時系列  $U$  を返す . そのフローチャートを Fig. 5.1 (b) に示す .

次に , Fig. 5.2 を用いて `diffEqs` での計算を説明する . 順動力学計算はこの部分で行われる . 各工程ごとの説明を行っていく .

`load current states ( $q, \dot{q}, \tau$ )`

状態  $x$  から関節変位  $q$  と関節速度ベクトル  $\dot{q}$  を , 制御入力  $u$  から , 関節駆動力  $\tau$  を求める .

calculate coordinate transform matrix of each joints( ${}_{\lambda(i)}^i X$ )

座標系 $\{\lambda(i)\}$ のモーションベクトルを座標系 $\{i\}$ へ変換する行列 ${}_{\lambda(i)}^i X$ を求める。ここで、 $\lambda(i)$ は関節*i*の親(根元側)の関節番号とする。

calculate joint velocity of each joints ( $\mathcal{V}_i$ )

関節*i*の速度 $\mathcal{V}_i = \mathcal{V}_{\lambda(i)} + S_i \dot{q}_i$ を計算する。

calculate joint acceleration of each joints ( $\dot{\mathcal{V}}_i$ )

関節*i*の加速度 $\dot{\mathcal{V}}_i = \mathcal{V}_i \times \dot{S}_i \dot{q}_i$ を計算する。ここで、 $\mathcal{V}_i \times$ は空間ベクトルのモーションベクトルの外積であり、 $\mathcal{V}_i = [\omega_{ix}, \omega_{iy}, \omega_{iz}, v_{ix}, v_{iy}, v_{iz}]^T$ とすると、次のように表される。

$$\mathcal{V}_i \times = \begin{bmatrix} 0 & -\omega_{iz} & \omega_{iy} & 0 & 0 & 0 \\ \omega_{iz} & 0 & -\omega_{ix} & 0 & 0 & 0 \\ -\omega_{iy} & \omega_{ix} & 0 & 0 & 0 & 0 \\ 0 & -v_{iz} & v_{iy} & 0 & -\omega_{iz} & \omega_{iy} \\ v_{iz} & 0 & -v_{ix} & \omega_{iz} & 0 & -\omega_{ix} \\ -v_{iy} & v_{ix} & 0 & -\omega_{iy} & \omega_{ix} & 0 \end{bmatrix} \quad (5.4)$$

load joint external force of each joints ( $\mathcal{F}_i^{ex}$ )

関節*i*に加わる外力 $\mathcal{F}_i^{ex}$ を読み込む。

calculate IABI and bias acceleration of each links ( $\Phi_i, \beta_i$ )

$\mathcal{V}_i$ と $\mathcal{F}_i^{ex}$ からリンク*i*のIABI $\Phi_i$ とバイアス加速度 $\beta_i$ を計算する。

hybrid modeling of assembly step ( $\Phi, \beta$ )

ハイブリッドモデリングを行い $\Phi, \beta$ を求める。

assembly step ( $\Phi, \beta$ )

残りのABをADAのassemblyステップで接続し、 $\Phi, \beta$ を求める。

disassembly step ( $\lambda_i, \ddot{q}_i$ )

Disassemblyで関節*i*の拘束力 $\lambda_i$ と関節の角加速度 $\ddot{q}_i$ を求める。

hybrid modeling of disassembly step ( $\ddot{q}_i$ )

ハイブリッドモデリングを行い、残りの関節加速度 $\ddot{q}_i$ を求める。

unit quaternion ( ${}_0^1 R, \dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4$ )

浮遊ベースを含む場合は、単位クオータニオンを用いて、慣性座標系 $\{0\}$ から第1関節の座標系 $\{1\}$ の変換行列 ${}_0^1 R$ を求める。

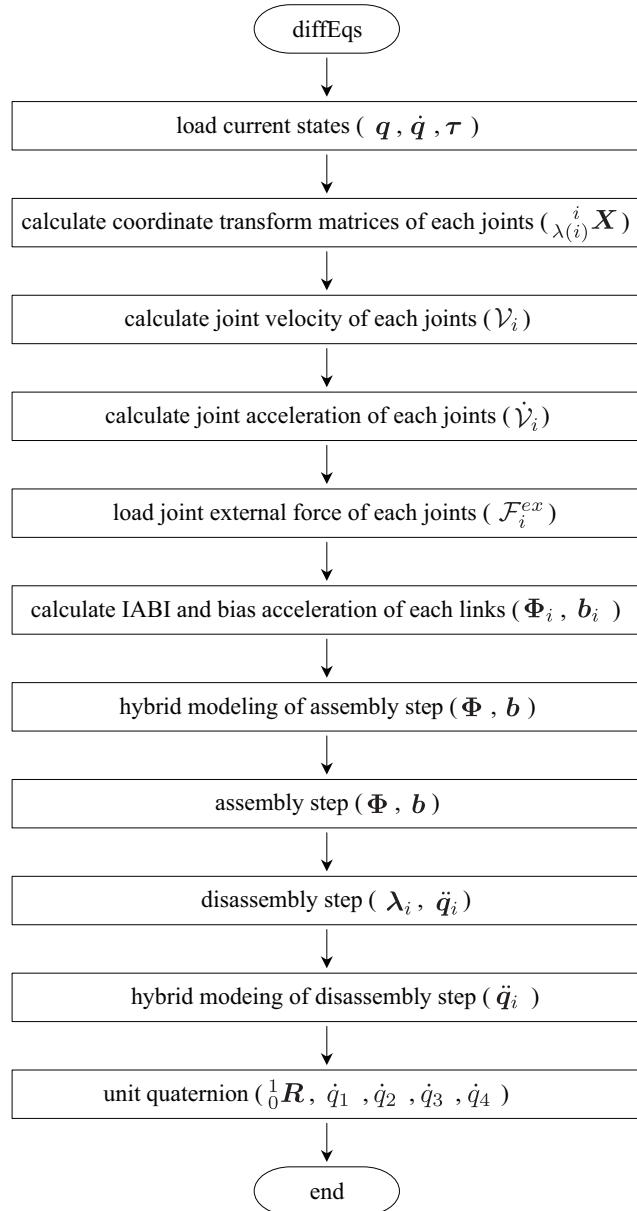


Fig. 5.2: program flowchart of the simulator.

このような流れで、ハイブリッドモデリングを用いた順動力学計算を行い、 $\dot{x} = [\dot{q}, \ddot{q}]^T$ を計算する。

## 5.2 シミュレータのファイル構造

シミュレータのファイル構成を Table 5.1 に示す。

Table 5.1: Simulator structure.

directory name	file name	detail
	Makefile main.mm init.mm	make file master routine initialization (initial states, times and flags)
include	define.h extern.h	define extern
spatialVector	spatialCrossFM.mm spatialInertia.mm spatialRotTransX.mm	cross product operator construct rigid-body inertia coordinate transform matrices
model	parameter.mm comVector.mm inertiaTensor.mm	setting of joints, links and connections CoM vector of each links inertia tensor of each links
kinematics	rotationMatrix.mm quaternion.mm positionJointMassNode.mm velocity.mm velocityD.mm	rotation matrix of each links unit quaternion position of each joints, masses and nodes spatial velocity of each joints spatial acceleration of each joints
dynamics	dynamics.mm iABIbiasA.mm externalForce.mm assemblyStep.mm disAssemblyStep.mm hybridModeling.mm	calculate forward dynamics IABI and bias acceleration of each links external force of each joints assembly step disassembly step hybrid modeling
controller	controller.mm desMotion.mm spline5.mm	calculate control torque desired motion fifth-order spline
output	output.mm animation.mm plotEqsPdf.mm	calculation of output files make gif-animation plot data, create EPS files and PDF file
analysis	zmpPosition.mm comPosVelMass.mm momentum.mm mechanicalEnergy.mm	position of the Zero-Moment Point (ZMP) CoM parameters of whole link linear/angular momentum of whole link mechanical energy
data		save datas (dat, eps, gif, gp and PDF)

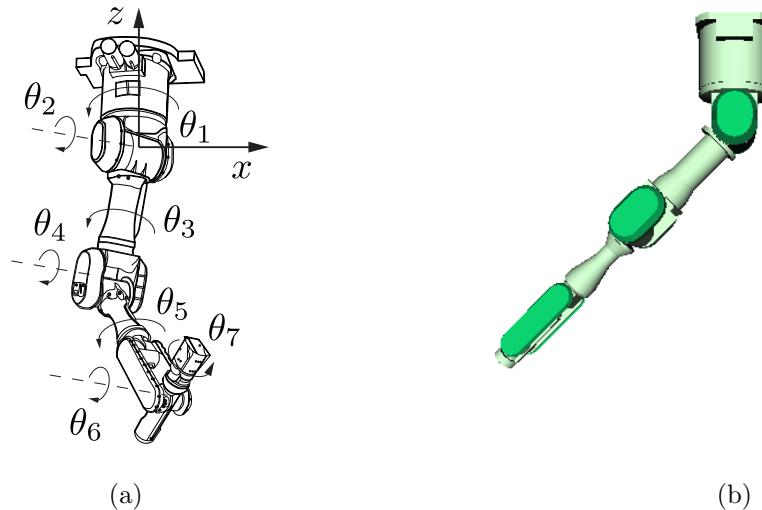


Fig. 5.3: The PA-10 7R manipulator: (a) kinematic structure and (b) initial angles.

Table 5.2: Initial angles.

$i$	1	2	3	4	5	6	7
$\theta_i^{ini}$ [deg]	0	45	10	0	0	0	0

### 5.3 シミュレーション例：PA-10

Fig. 5.3 (a) に示す三菱重工業社製の 7 自由度マニピュレータ PA-10 のシミュレータを作成した。このモデルは、七つの回転関節が  $z, y, z, y, z, y, z$  軸の順に配置された、シリアルリンク機構のマニピュレータである。各リンクの長さや質量、重心位置は、マニュアルに記載された値を設定した。Fig. 5.3 (b) と Table 5.2 に示す初期角度から、初速および制御入力なしの条件で、自由落下するシミュレーションを行った。シミュレーション時間を 10 s、積分の刻み幅を 1 ms、重力加速度を  $z$  軸方向に  $-9.81 \text{ m/s}^2$  とした。ただし、リンクの慣性モーメントはすべて  $\text{diag}(1, 1, 1) \text{ kg} \cdot \text{m}^2$  とし、関節の摩擦やギア比は考慮していない。

運動の様子をFig. 5.4にスナップショットで示す。また、シミュレーション結果をFig. 5.5に示す。Fig. 5.5 (a)は関節角度、(b)は関節速度、(c)は関節加速度、(d)は関節トルク、(e)は力学的エネルギーである。Fig. 5.4, Figs. 5.5 (a)–(c)より、設定した初期状態から重力の影響を受け、自由落下する様子が確認できる。また、Figs. 5.5 (d), (e)より、制御入力を与えない状態で、位置工

エネルギー(P.E.)と運動エネルギー(K.E.)の総和が一定となり、力学的エネルギー保存則が確認できた。よって、正しく順動力学計算が行えていると判断できる。

## 5.4 本章のまとめ

ADAを用いた数値・数式ハイブリッドモデリングを行う順動力学エンジンを開発し、マルチボディダイナミクスシミュレータへ適用した。本章では、このシミュレータのプログラムの流れやファイル構成を示した。また、例として、7自由度ロボットアームPA-10のシミュレーションを行い、正しく順動力学計算が行われていることを示した。

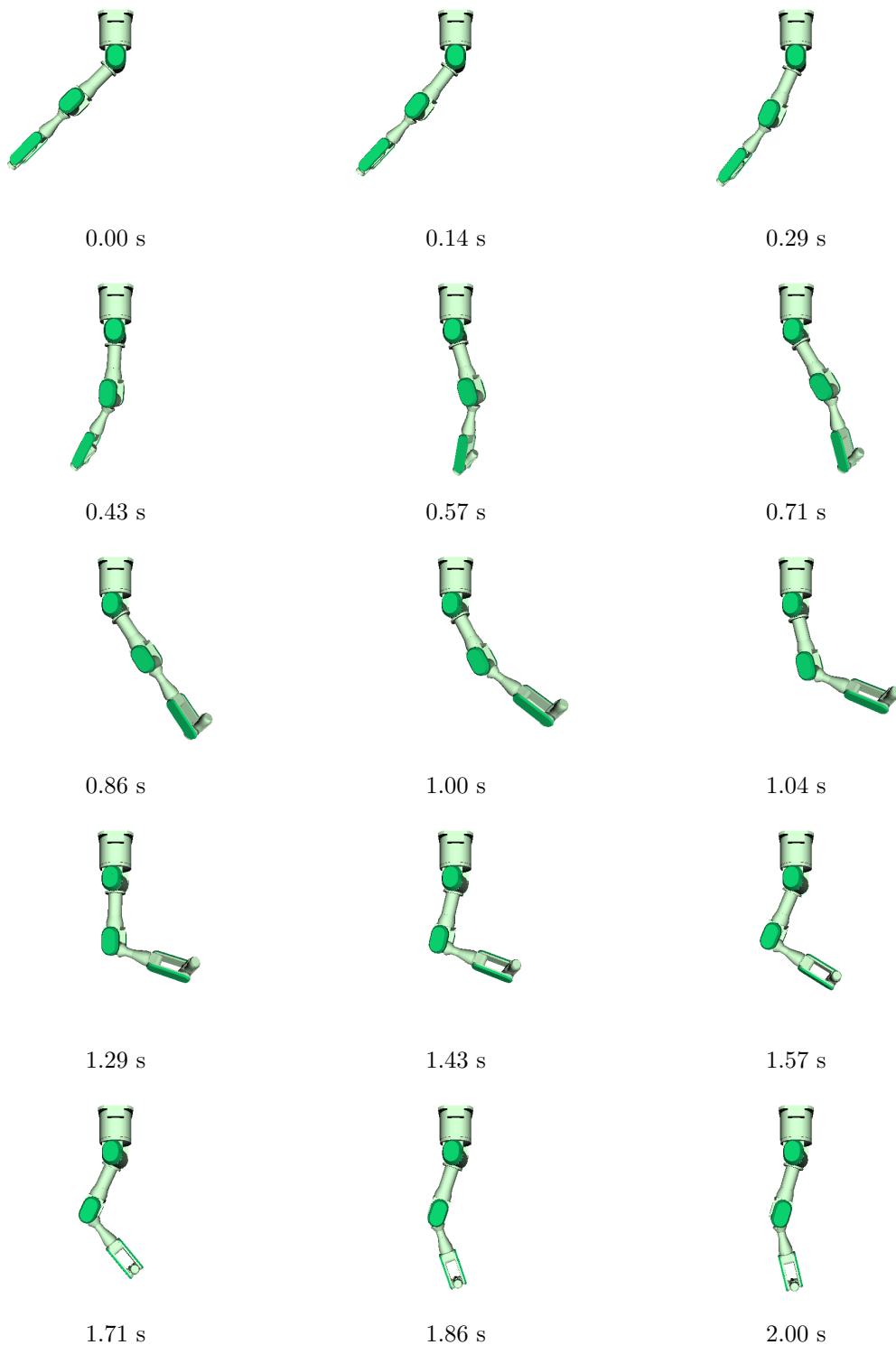


Fig. 5.4: Snapshots from simulation of the PA-10 7R manipulator.

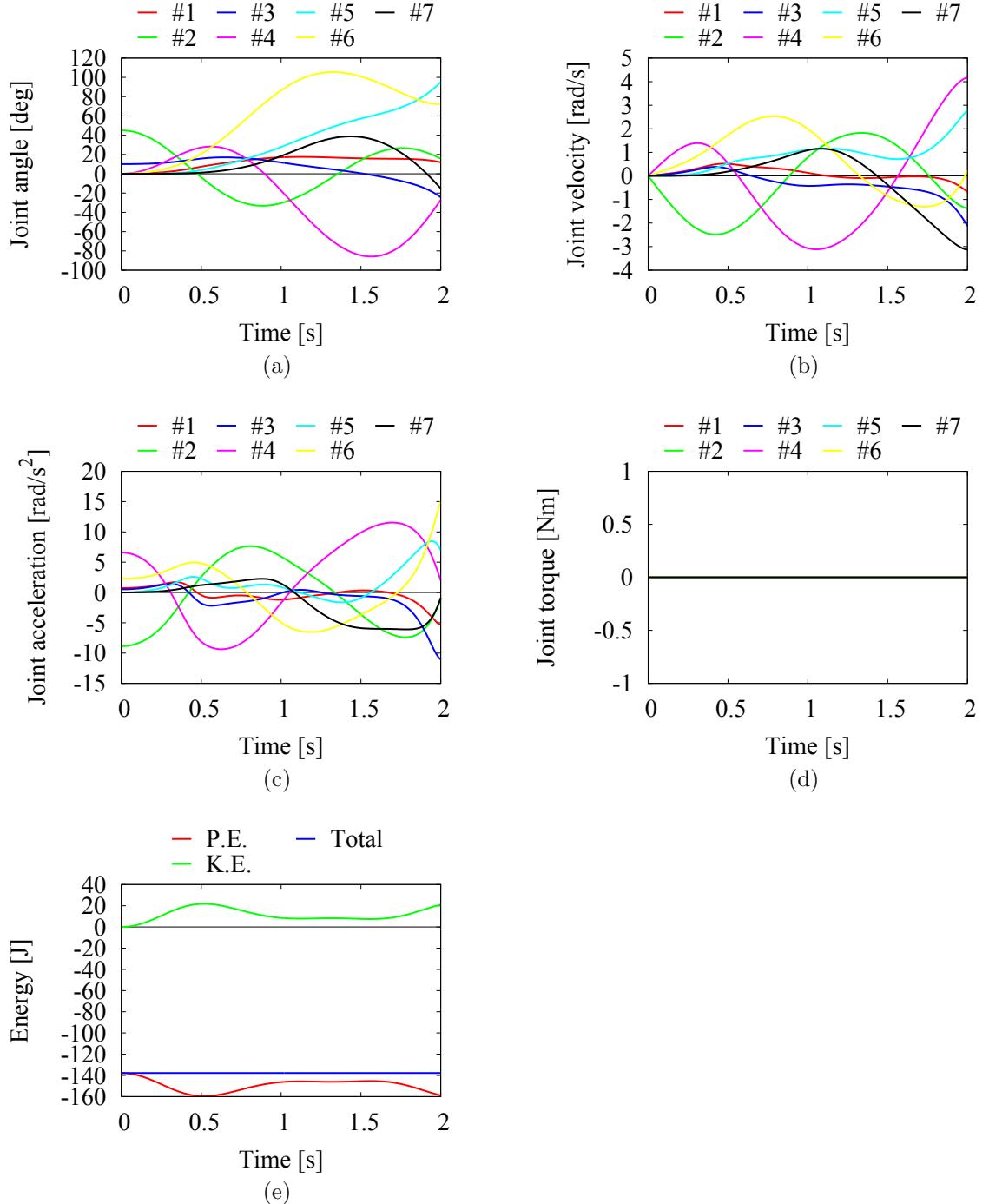


Fig. 5.5: Simulation results of the PA-10 7R manipulator: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque and (e) mechanical energy.



# 第6章 ミュレータの実用例

開発したミュレータの実用例として、人型ロボット HOAP-2 のミュレータを作成し、バランス制御の検証に用いた例を述べる。本章では、直立姿勢のロボットの肩や手先に突発的な衝撃力が加わった際の運動を扱う。このような運動に対する姿勢運動戦略は、ロボティクス分野はもちろんバイオメカニクス分野においても研究成果が存在しない。そこでまず、人間の運動解析を行い、姿勢運動戦略を提案する。さらに、その姿勢運動戦略を人型ロボットに導入し、バランス制御を行う。

## 6.1 反動零空間法 (Reaction Null-Space: RNS)

人型ロボットは、冗長自由度を持つ劣駆動システムとして扱うことができる。また、環境と接触したリンクに伝わる反力・反モーメントを制御することで、バランス制御を行う。与えられたロボットの動作に対して足裏が床面と接触を維持できるかを判定したり、あるいは、接触を維持できるロボットの動作を計画するために、Zero-Moment Point (ZMP) が幅広く用いられている。ZMP 規範の安定性の評価では、足部に働くすべての力を知ることができない。したがって、ZMP では、平らな床面との接触のみで用いられている。

一方、実環境での床面との接触は、ZMP で扱えるほど単純ではない。このような状況では、環境と接触したリンクのすべての状態を把握する必要がある。この問題を扱うために、我々は、宇宙ロボット工学の分野で扱われてきた反動零空間法 [54][55] の考え方を人型ロボットに適用してきた [56]。反動零空間法を用いることで、接触状態を保つすべてのリンクの動きを容易に得ることができる。人型ロボットの反動零空間法では、支持脚の足部と他のリンクの間の干渉慣性行列の零空間射影によって定義される。一つの足部でバランスを取るために、 $n$  自由度の人型ロボットの運動

方程式を次式に示す。

$$\begin{bmatrix} \mathbf{M}_f & \mathbf{M}_{fl} \\ \mathbf{M}_{fl}^T & \mathbf{M}_l \end{bmatrix} \begin{bmatrix} \dot{\mathcal{V}}_f \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} \mathcal{C}_f \\ \mathbf{c}_l \end{bmatrix} + \begin{bmatrix} \mathcal{G}_f \\ \mathbf{g}_l \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau} \end{bmatrix} + \begin{bmatrix} \mathbf{J}_{gf}^T \\ \mathbf{J}_{gl}^T \end{bmatrix} \mathcal{F}_g + \begin{bmatrix} \mathbf{J}_{ef}^T \\ \mathbf{J}_{el}^T \end{bmatrix} \mathcal{F}_e \quad (6.1)$$

式中の変数名は次の通りである。

$\mathbf{M}_f$	$\in \Re^{6 \times 6}$	: 足部におけるシステム全体の慣性行列
$\mathbf{M}_{fl}$	$\in \Re^{6 \times n}$	: 足部とリンク部の干渉慣性行列
$\mathbf{M}_l$	$\in \Re^{n \times n}$	: リンク部の慣性行列
$\mathcal{C}_f$	$\in \Re^6$	: 足部のコリオリ遠心力項
$\mathbf{c}_l$	$\in \Re^n$	: リンク部のコリオリ遠心力項
$\mathcal{G}_f$	$\in \Re^6$	: 足部の重力項
$\mathbf{g}_l$	$\in \Re^n$	: リンク部の重力項
$\boldsymbol{\theta}$	$\in \Re^n$	: リンク部の関節変位
$\mathcal{V}_f$	$\in \Re^6$	: 足部の並進速度と角速度
$\boldsymbol{\tau}$	$\in \Re^n$	: 関節トルク
$\mathcal{F}_g$	$\in \Re^6$	: 床反力
$\mathcal{F}_e$	$\in \Re^6$	: 任意のリンクに加わる力
$\mathbf{J}_{\{\circ\}}$		: 力変換行列

Eq. (6.1) から足部の運動方程式を抜き出すと次のようになる。

$$\mathbf{M}_f \dot{\mathcal{V}}_f + \mathbf{M}_{fl} \ddot{\boldsymbol{\theta}} + \mathcal{C}_f + \mathcal{G}_f = \mathbf{J}_{gf}^T \mathcal{F}_g + \mathbf{J}_{ef}^T \mathcal{F}_e \quad (6.2)$$

ここで、初期状態は静止しており、外力と床反力の総和が重力項とつりあう状態を仮定し、 $\mathcal{V}_f = \mathbf{0}$ 、 $\mathcal{C}_f \approx \dot{\mathbf{M}}_{fl} \dot{\boldsymbol{\theta}}$ 、 $\mathcal{G}_f = \mathbf{J}_{gf}^T \mathcal{F}_g + \mathbf{J}_{ef}^T \mathcal{F}_e$  とすると、次式が導ける。

$$\mathbf{M}_{fl} \ddot{\boldsymbol{\theta}} + \dot{\mathbf{M}}_{fl} \dot{\boldsymbol{\theta}} = \mathbf{0}. \quad (6.3)$$

上式の  $\ddot{\boldsymbol{\theta}}$  に関する一般解を求める。

$$\{\ddot{\boldsymbol{\theta}}_{RL}\} = -\mathbf{M}_{fl}^\# \dot{\mathbf{M}}_{fl} \dot{\boldsymbol{\theta}} + \{\mathcal{N}_{fl}\}, \quad (6.4)$$

ここで、式中の  $(\circ)^\#$  は一般化逆行列、 $\{\mathcal{N}_{fl}\}$  は干渉慣性行列  $\mathbf{M}_{fl}$  の零空間射影である。零空間射影は  $\{(U - \mathbf{M}_{fl}^\# \mathbf{M}_{fl})\zeta\}$  のようなベクトルとして求められる。ここで、 $U$  は単位行列、 $\zeta$  は任意のベクトルである。

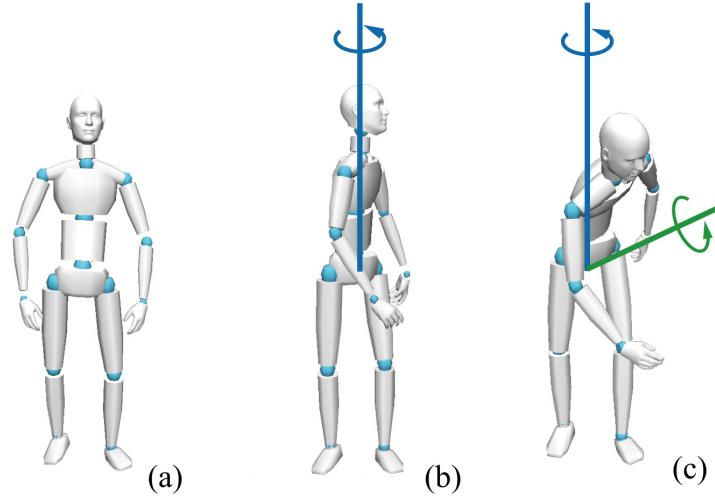


Fig. 6.1: Reaction patterns in response to an impact force on the shoulder from behind (obtained via motion capture): (a) initial state, (b) upper-body rotation strategy (smaller impact) and (c) combined rotation–extension/flexion strategy (larger impact).

## 6.2 人間の姿勢運動戦略

人間の姿勢運動戦略はバイオメカニクスの分野で多く研究されている。後方から外乱を受けた際の姿勢運動戦略である立位姿勢制御法[57]は、ankle strategy, hip strategy, step strategy の三段階の対応が存在し、加えられた外乱に応じて対応を変化させる。過去の研究では、加えられた衝撃力に対して、ankle strategy もしくは hip strategy による人型ロボットのバランス制御を行った[56]。また、側面から外乱を受けた際の姿勢運動戦略を提案し、人型ロボットのバランス制御を行った[58]。

一方、外乱が身体中央に作用しない、すなわち重心に対してオフセットを持ち、鉛直軸回りのモーメントが発生する条件下での人間の姿勢運動戦略は、バイオメカニクスの分野においても研究成果が存在しない。そこで、人間の動作解析を行い、新たな姿勢運動戦略を提案する。さらに、シミュレータ上で人型ロボットに適用し、バランス制御を行う。

人間の動作解析は、モーションキャプチャシステム OptiTrac[59] と床反力を計測するための Wii Board[60] を組み合わせて行う。外乱の大きさに応じて、Fig. 6.1 に示す二つの運動が確認された。比較的小さな衝撃力を肩に与えた際に、人間は上半身を垂直軸回り( $z$ 軸回り)に回転した。この運

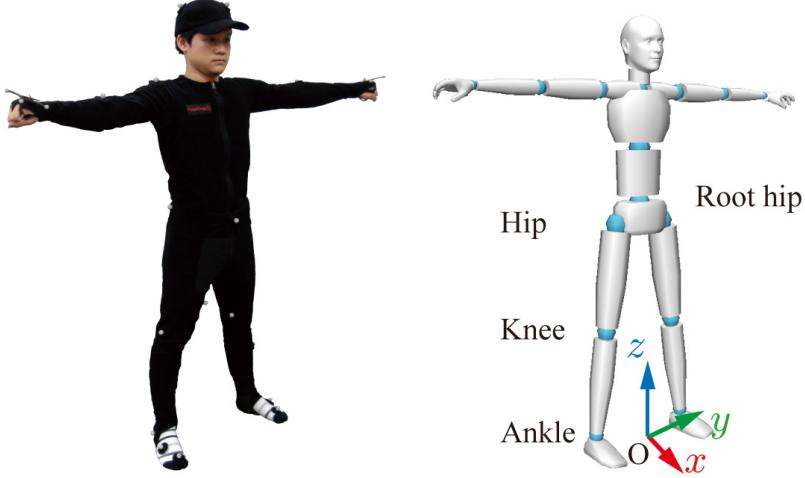


Fig. 6.2: The joints used in the motion capture model.

動は，腰関節と足首関節を使った下半身の運動によって実現された．我々は，この姿勢運動戦略を Upper-body rotation strategy と呼ぶ (Fig. 6.1 (b))．大きな衝撃力を与えた際に，人間は上半身の垂直軸回り ( $z$  軸回り) の回転と  $y$  軸回りの伸展・屈曲を組み合わせた運動により，バランスを保った．我々は，この姿勢運動戦略を Combined rotation-extension/flexion strategy と呼ぶ (Fig. 6.1 (c))．

様々な大きさの衝撃力を人間の肩の前方と後方から与え，人間の動作を確認した．代表的な実験データを示し，解析していく．

一つ目の実験結果を Fig. 6.3 に示す．比較的小さな衝撃力を左肩前方から与え，upper-body rotation strategy を行った．Fig. 6.3 (a) より， $x$  軸方向の力が最大約  $-15 \text{ N}$  であることが分かる．この力は次式より求めた [61]．

$$f_x = \frac{mg}{l} (r_{cx} - r_{px}), \quad (6.5)$$

ここで，式中の  $m$  は全質量， $g$  は重力加速度， $l$  は床面から力を加えた点までの長さ， $r_{cx}$  はモーションキャプチャシステムから得られた  $x$  軸方向の重心位置， $r_{px}$  は Wii ボードから得られた  $x$  軸方向の圧力中心点の位置である．

Fig. 6.3 (b) より， $y$  軸回りと  $z$  軸回りの角運動量が逆位相となっている．Fig. 6.3 (c) より，体幹部 (Root hip) は主に  $z$  軸方向に動いており， $y$  軸方向の動きはほとんどない．Figs. 6.3 (e), (f) より，下半身は  $y$  軸回りと  $z$  軸回りの左脚の腰関節と右脚の足首関節を主に動かしていることが分かる．このことから，右脚に体重をかけて左脚を動かしていることが分かる．また，同等の力を右肩後方から与えた場合も，同じような傾向が見られた．

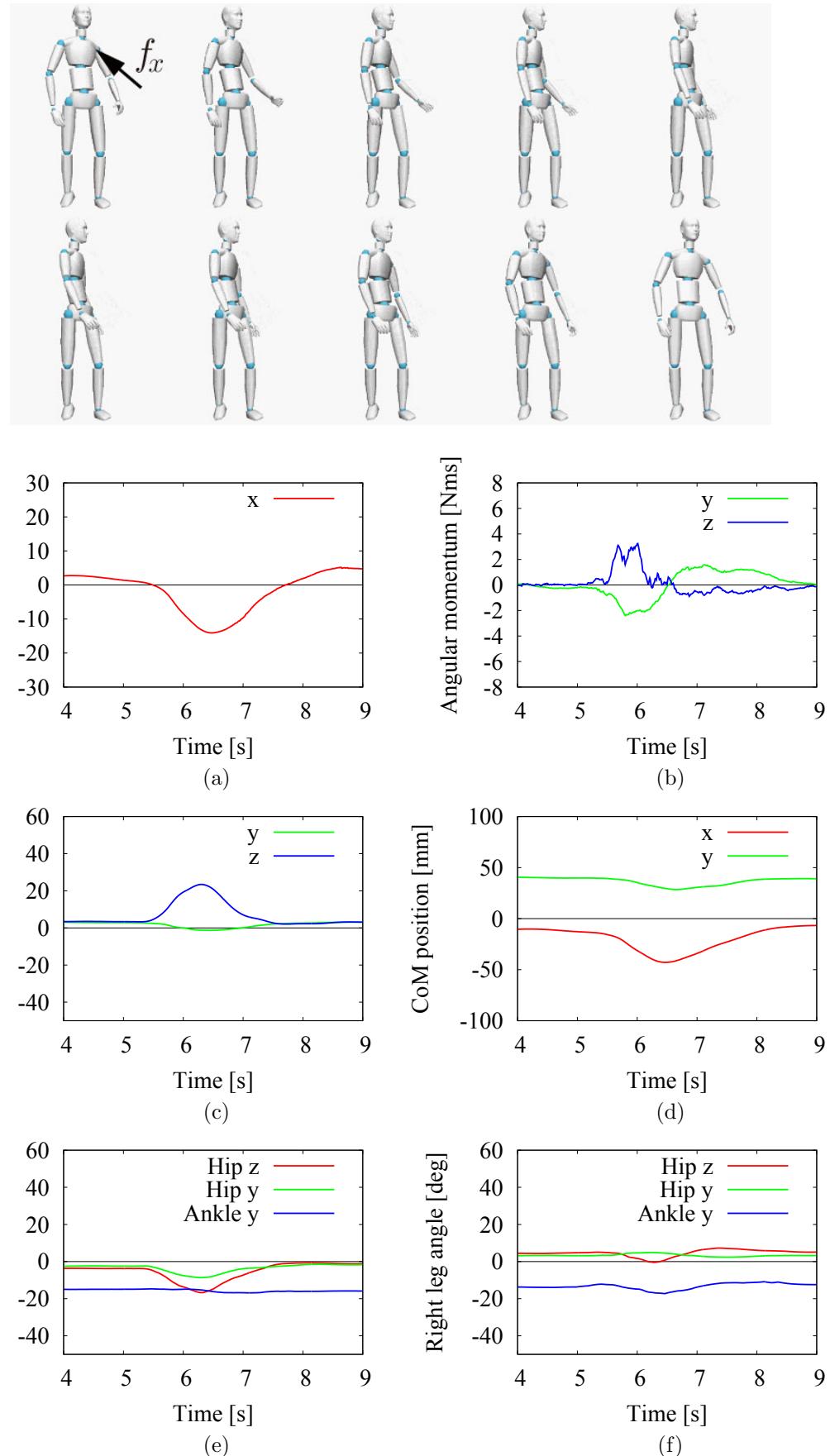


Fig. 6.3: Motion capture data from upper-body rotation strategy (impact on the front).

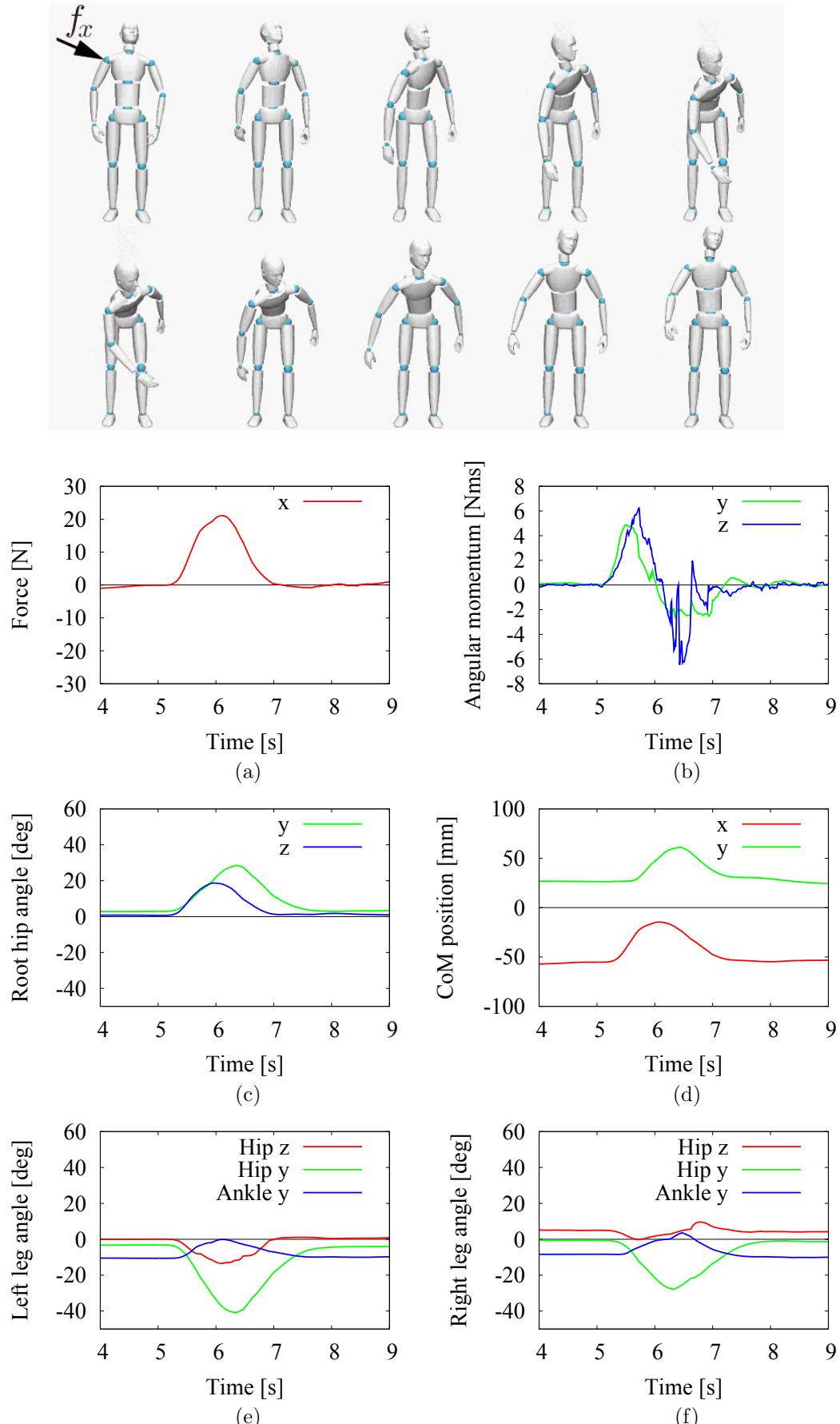


Fig. 6.4: Motion capture data from combined rotation–extension/flexion strategy (impact from behind).

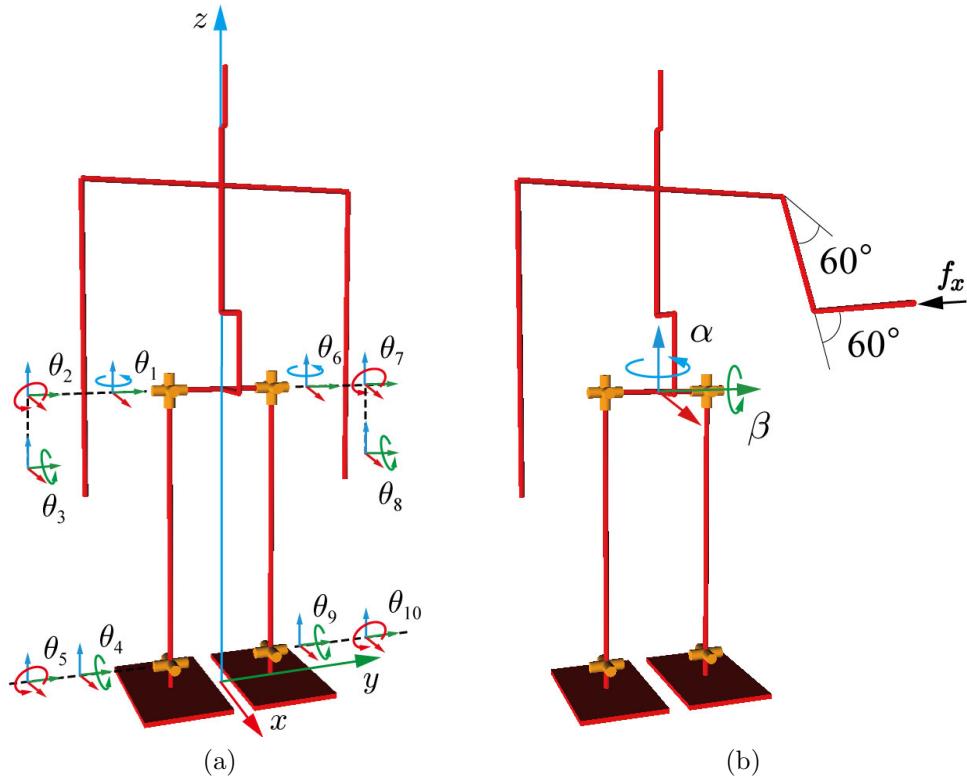


Fig. 6.5: Simulation model: (a) coordinate frames ( $\theta_i = 0, i = 1, 2, \dots, 10$ ) and (b) initial position of the simulation of the upper-body rotation strategy.

二つ目の実験結果を Fig. 6.4 に示す。combined rotation-extension/flexion strategy を行った。衝撃力は右肩後方から与えた。Fig. 6.4 (a) より、衝撃力の最大値は約 20 N である。さらに、Fig. 6.4 (b) より、二つの角運動量は同相である。Fig. 6.4 (c) から、体幹部の  $y$  軸回りと  $z$  軸回りの回転運動が比較的大きいことが分かる。Figs. 6.4 (e), 6.4 (f) より、左脚の腰関節 ( $y$ ) と左脚の足首関節 ( $y$ ) が逆方向に回転している。これは、過去の研究の hip strategy と同じ傾向である [57]。

### 6.3 小型人型ロボットへの導入

upper-body rotation strategy と combined rotation-extension/flexion strategy を小型人型ロボット HOAP-2 に導入する。このロボットは、人間のように体幹部の関節がない。そこで、他の関節を用いて上半身の垂直軸 ( $z$  軸) 回りの運動を実現する必要がある。Fig. 6.5 (a) に示す、三次元の 10 自由度モデルを用いる。腰関節に 3 自由度 (関節 1-3 と関節 6-8) と足首関節に 2 自由度 (関節 4,

5と関節9, 10)を持つ。関節角度 $\theta_i$ を一般化座標系とする( $i = 1, 2, \dots, 10$ )。同様に、 $\tau_i$ を一般化力とする。また、外乱に対応するために仮想ばね・ダンパを各関節に設定した。我々の過去の研究において、このアプローチは成功している。このばねとダンパの係数は経験則に基づいて一定値や変数を設定した[56]。モーションキャプチャのデータから、この二つの姿勢運動戦略を行う際に、人間の腕の動きがとっても小さいことを確認した。そこで、上半身は一つの剛体として扱う。物理パラメータはHOAP-2と同じものを用いる。二つの姿勢運動戦略の詳細について以下で述べていく。

### 6.3.1 Upper-body rotation strategy

反動動作とバランス回復の二つのフェーズの運動を、次に示す仮想ばね・ダンパシステムを用いて実現する。

$$\tau_i^{ref} = k_{p_i}(\theta_i^{ini} - \theta_i) - k_{d_i} \dot{\theta}_i, \quad (6.6)$$

ここで、式中の $k_{p_i}$ と $k_{d_i}$ はばねとダンパの値、上付きの $(\circ)^{ref}$ と $(\circ)^{ini}$ は指令値と初期値である。反動動作フェーズでは、 $k_{p_i}$ は定数とする。一方、 $k_{d_i}$ はバランス回復フェーズに円滑に遷移するために、非線形の変数 $k_{d_i} = k_{d_i}^{sp}(\alpha)$ を用いる。 $k_{d_i} = k_{d_i}^{sp}(\alpha)$ は上半身の回転角 $\alpha$ によって変化する五次のスプライン関数である。

### シミュレーション

すべての関節角度をゼロとした直立姿勢を初期姿勢とする。反動動作フェーズでの仮想ばね・ダンパは $k_{p_i} = 0.3 \text{ Nm/rad}$ ,  $k_{d_i} = 0.03 \text{ Nm}\cdot\text{s}/\text{rad}$  ( $i = 1, 2, \dots, 10$ )の定数を設定した。一方、バランス回復フェーズでは、 $0.0 \leq k_{d_i} \leq 0.5 \text{ Nm}\cdot\text{s}/\text{rad}$  ( $i = 1, 6$ ),  $0.0 \leq k_{d_i} \leq 3.5 \text{ Nm}\cdot\text{s}/\text{rad}$  ( $i = 4, 9$ )の非線形(スプライン)の変数を設定した。他の関節は、 $k_{p_i} = 1.5 \text{ Nm/rad}$ ,  $k_{d_i} = 1.5 \text{ Nm}\cdot\text{s}/\text{rad}$  ( $i = 2, 3, 5, 7, 8, 10$ )の定数を設定した。衝撃力はHOAP-2の肩に与えた。

我々は、現実に近い状況をシミュレーションする。例えば、ドアの近くにロボットが立っていて、左手でドアノブを握ろうとする。ドアノブを握る前に、ドアが突然開く状況を想定する。ロボットの初期角度をFig. 6.5 (b)に示す。左腕の肩と肘の角度をそれぞれ $60 \text{ deg}$ とする。ここで、左腕は一

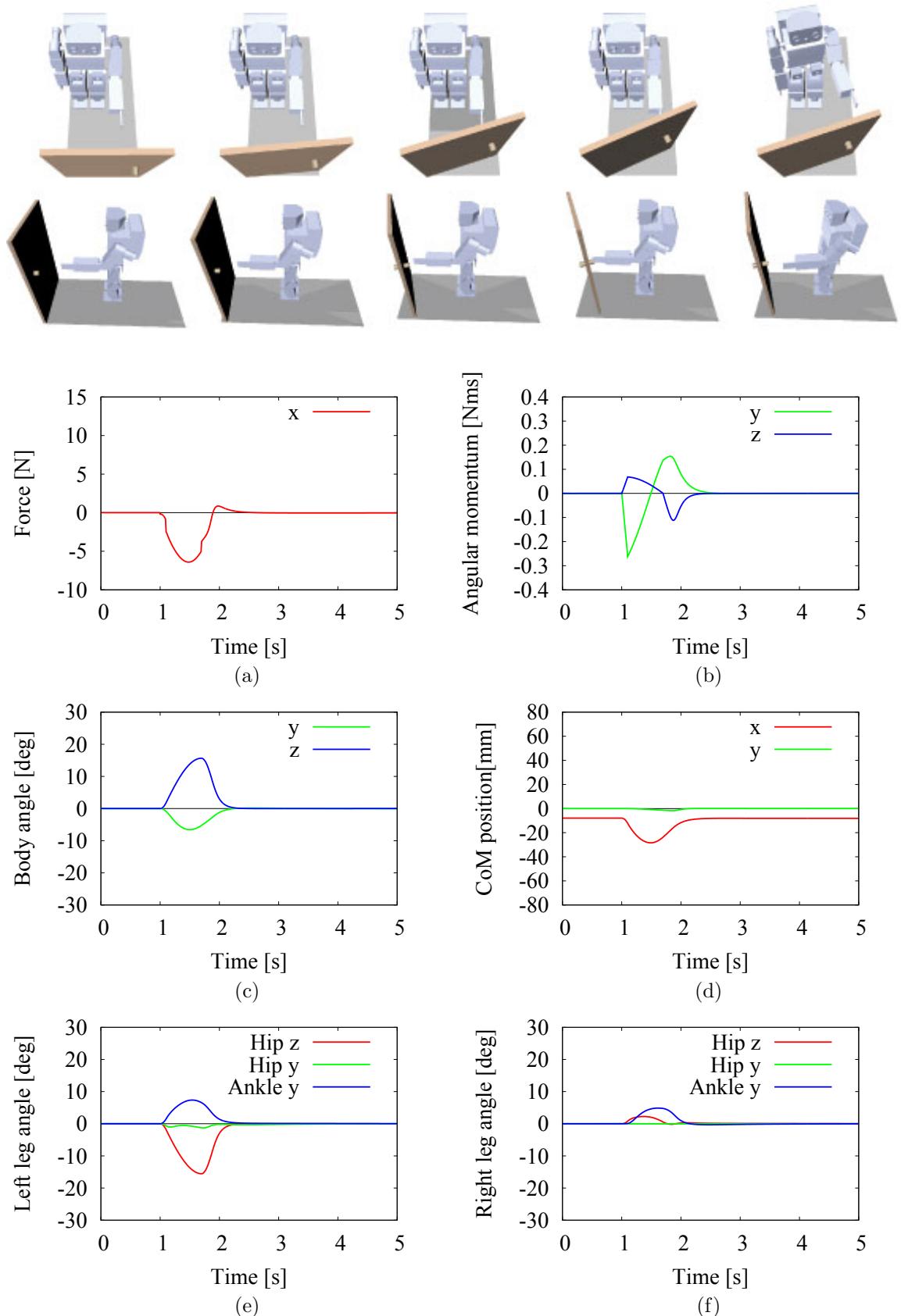


Fig. 6.6: Simulation results of upper-body rotation strategy. The robot approaches the door with its left hand. The door is suddenly opened, and an unexpected impact disturbance is applied to the left hand of the robot.

つの剛体と仮定する。 $f_x = -7\text{ N}$  の衝撃力を左腕の先端に前から与える。衝撃力を与えるタイミングは、 $1.0 \leq t \leq 1.1\text{ s}$  とする ( $t$  は時間)。Eq. (6.6) のみを用いた場合、ロボットは後方に倒れる。上半身を  $z$  軸回りに回転させるために、右脚の足首関節(関節4)の制御を行う。この関節は次式を用いて、左脚の足首関節(関節9)と逆方向に動かす。

$$\tau_4^{ref} = -(k_{p_4} \theta_9 + k_{d_4} \dot{\theta}_9), \quad (6.7)$$

ここで、式中の  $k_{p_4} = 5.0\text{ Nm/rad}$ ,  $k_{d_4} = 0.5\text{ Nm}\cdot\text{s}/\text{rad}$  とした。また、上半身の  $y$  軸回りの回転を小さくするために、 $y$  軸回りの腰関節の仮想ばね・ダンパ係数を  $k_{p_3} = k_{p_8} = 5.0\text{ Nm/rad}$ ,  $k_{d_3} = k_{d_8} = 0.5\text{ Nm}\cdot\text{s}/\text{rad}$  とし、他の関節より大きく設定した。これらの設定を行うことで、upper-body rotation strategy を確認した。Fig. 6.6 にスナップショットとシミュレーション結果を示す。Fig. 6.6 (a) より、最大約  $-7\text{ N}$  の力が加わったことが確認できる。Fig. 6.6 (b) より、二つの角運動量の成分が逆位相となっている。これは、上で述べた人間の拳動と同じ傾向である。Fig. 6.6 (c) より、上半身の運動は  $z$  軸回りの回転を主に用いており、 $y$  軸回りの回転は比較的小さい。

### 6.3.2 Combined rotation-extension/flexion strategy

衝撃力に対応する上半身の伸展動作は、重心の動きが重要である。ロボットは、重心の動きによって、簡単にバランスが保てなくなる。つまり、重心の動きの制御は、慎重に考えなければならない。6.1節に示したように、反動零空間法を適用することで容易に解決できる。矢状面の ankle-hip strategy[56] や前額面の lift-leg strategy[58] すでに示した。

Eq. (6.3) を時間積分して次式を得る。

$$M_{fl}(\theta) \dot{\theta} = \mathcal{L} \quad (6.8)$$

ここで、式中の積分定数  $\mathcal{L}$  は、角運動量であり、足部への反動を零にするには  $\mathcal{L} = \text{const}$ 、あるいは  $\mathcal{L} = 0$  でなければならない。一般性を失わないために、これより初期状態は  $\mathcal{L} = 0$  とする。したがって、次のように関節角度を得る。

$$\dot{\theta} = N_{fl}(\theta) b \quad (6.9)$$

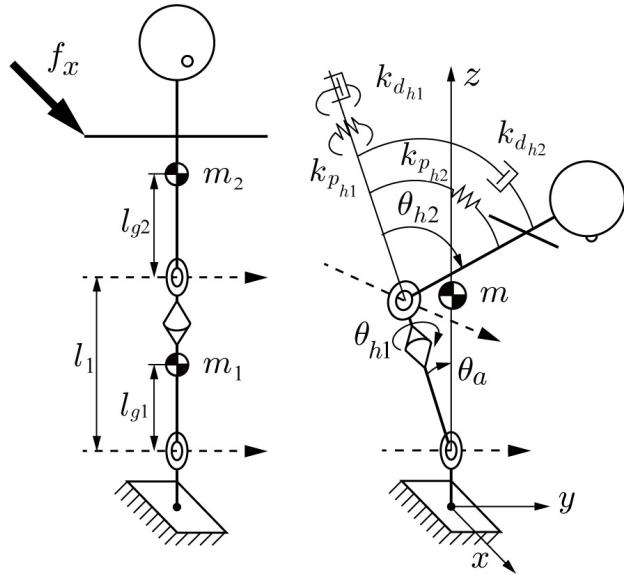


Fig. 6.7: Spatial three-DOF model for the combined rotation–extension/flexion strategy.

ここで、式中の  $N_{fl}(\theta) \in \Re^{n \times n}$  は干渉慣性行列  $M_{fl}(\theta)$  の零空間射影、 $b$  は任意の  $n$  ベクトルである。

Fig. 6.7 に示す 3 自由度の簡易モデルを用いて考えていく。三つの関節は、腰関節に 2 自由度（関節  $h_1, h_2$ ）、足首関節に 1 自由度（関節  $a$ ）である。干渉慣性行列の次元は  $M_{fl} \in \Re^{1 \times 3}$  である。関節速度は Eq. (6.9) より次のように得られる。

$$\dot{\theta} = b_1 \mathbf{n}_1 + b_2 \mathbf{n}_2, \quad (6.10)$$

ここで、 $b_1, b_2$  は任意のスカラー、 $\mathbf{n}_1, \mathbf{n}_2 \in \Re^3$  は  $M_{fl}$  の零空間射影から求まる二つの独立ベクトルである。

$$\begin{aligned} \mathbf{n}_1 &= [n_{11} \ 0 \ n_{31}]^T, \\ \mathbf{n}_2 &= [n_{12} \ n_{22} \ 0]^T. \end{aligned} \quad (6.11)$$

以下のシミュレーションでは、Eq. (6.10) の右辺の最初の成分のみ用いる。つまり、 $b_1 = 1, b_2 = 0$  とする。Eq. (6.11) の  $\mathbf{n}_1$  の式から、反動零空間の戦略は、関節  $\theta_a$  と  $\theta_{h2}$  の運動のみに関係する。 $\mathbf{n}_1$  の二つの成分は以下に示す通

りである。

$$\begin{aligned} n_{11} &= -m_2 l_{g2} (C_a C_{h1} C_{h2} - S_a S_{h2}), \\ n_{31} &= (m_1 l_{g1} + m_{h1} l_1) C_a + m_2 l_{g2} C_a C_{h2} - S_a S_{h1} S_{h2}, \end{aligned}$$

ここで、式中の各記号は  $C_i = \cos \theta_i$ ,  $S_i = \sin \theta_i$  ( $i = a, h1, h2$ ) である。それぞれの関節比の関係から、足首関節の指令値を求めることができる。

$$\dot{\theta}_a^{ref} = \frac{n_{11}}{n_{31}} \dot{\theta}_{h2}. \quad (6.12)$$

我々のミュレータでは関節トルクを入力する。そのため、次のように設定した。

$$\tau_a^{ref} = k_a (\dot{\theta}_a^{ref} - \dot{\theta}_a), \quad (6.13)$$

ここで、 $k_a$  はフィードバックゲインである。

### シミュレーション

Fig. 6.5 に示した 10 自由度モデルを用いてシミュレーションを行った。初期状態は前のシミュレーションと同様に直立姿勢とした。すなわち、すべての関節の初期角度はゼロである。Fig. 6.5 (a) のように、左の肩と肘の関節角度もゼロである。Fig. 6.5 のシミュレーションモデルと Fig. 6.7 の制御モデルの間の関係は  $\theta_a = (\theta_9 - \theta_4)/2$ ,  $\theta_{h1} = \theta_6$ ,  $\theta_{h2} = \theta_8$  である。

Eqs. (6.12), (6.13) から求めた反動零空間法に基づく解は、左の足首関節のみに対して制御入力として与える。一方、他の関節は、Eq. (6.6) の仮想ばね・ダンパにより、受動的に制御する。ばねとダンパ係数  $k_{p_i}$  と  $k_{d_i}$  は、リアクションとバランス回復の二つのフェーズを円滑に行うために、 $k_{p_i} = k_{p_i}^{sp}(\beta)$ ,  $k_{d_i} = k_{d_i}^{sp}(\beta)$  のような非線形の変数とする。ここで、 $k_{p_i}^{sp}(\beta)$  と  $k_{d_i}^{sp}(\beta)$  は上半身の伸展角度  $\beta$  に対する五次のスプライン関数である。初期の仮想ばね・ダンパ係数は経験的に  $k_{p_i} = 0.5 \text{ Nm/rad}$ ,  $k_{d_i} = 0.05 \text{ Nm} \cdot \text{s}/\text{rad}$  とした ( $i = 1, 2, \dots, 8, 10$ )。また、スプライン変数は  $0.5 \leq k_{p_9}^{sp} \leq 5.0 \text{ Nm/rad}$ ,  $0.05 \leq k_{d_9}^{sp} \leq 2.00 \text{ Nm} \cdot \text{s}/\text{rad}$  とした。さらに、 $k_a = 50 \text{ Nm} \cdot \text{s}/\text{rad}$  とした。

現実に近い状況をシミュレーションする。例えば、ドアの後ろに直立している状態で、ドアが突然開き、ドアノブによって右肩後方から突発的な衝撃力が加わった場合を想定する。Combined rotation-extension/flexion strategy

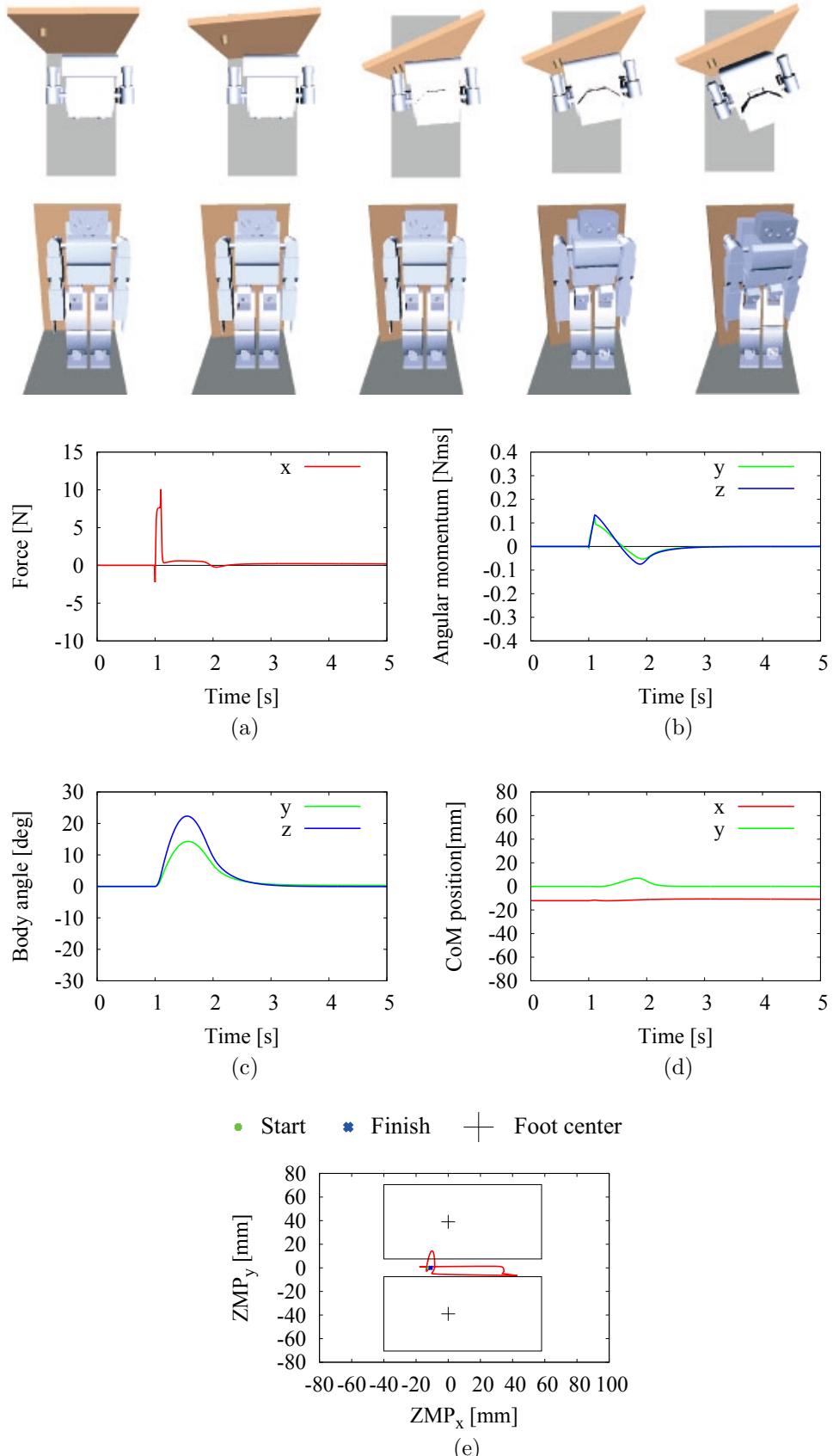


Fig. 6.8: Simulation results of the combined rotation-extension/flexion strategy (impact from behind). The robot is standing behind the door. The door is suddenly opened, and an unexpected impact disturbance is applied to the back at the right shoulder via the door knob.

を行う。シミュレーション結果を Fig. 6.8 に示す。Fig. 6.8 (a) より、衝撃力は最大で約 10 N であることが分かる。Fig. 6.8 (b) より、角運動量の  $y$  と  $z$  二つの成分が同相である。これは、モーションキャプチャーからのデータと同じ傾向である。Fig. 6.8 (c) を見て分かるように、上半身が大きく回転しているが Fig. 6.8 (d) の重心はほとんど動いていない。また、Fig. 6.8 (e) を見て分かるように、ZMP が常に足裏の支持多角形内に収まっている。このことから、ロボットはバランスを保っている。

## 6.4 本章のまとめ

突発的な衝撃力が肩に加わった際の人間の姿勢運動戦略を二つ提案した。一つ目は、比較的小さな衝撃力に対して、上半身を垂直軸周りに回転させてバランスを保つ Upper-body rotation strategy である。二つ目は、大きな衝撃力に対して、上半身の垂直軸周りの回転と水平軸周りの伸展・屈曲を組み合わせてバランスを保つ Combined rotation-extention/flexion strategy である。人間とロボットの間には、機構や自由度などの違いがあるが、この二つの姿勢運動戦略は、比較的容易に人型ロボットへ導入でき、バランス制御が行えることを示した。

ここでは、人型ロボットの両足裏は浮かないものと仮定し、足部が床面に拘束されたシミュレーションモデルを用い、仮想ばね・ダンパのみで制御した。今後は、足部が床面から離れるモデルを用意し、ロボットの両足と床面から成る閉リンク機構を考慮した制御を導入する必要がある。

# 第7章 閉リンク機構を考慮した制御

人型ロボットの両足支持期や物体把持、壁に手をついた際など、ロボット自体は閉リンク機構として扱えるモデルでも、環境や物体と接触した際に閉リンク機構が現れる。このような状態を制御する際に、仮想的に閉リンク機構の計算を行い、ロボットと環境や物体との干渉を考える必要がある。本章では、ADAの順動力学計算を用いて仮想的な閉リンク機構をモデリングし、閉リンク機構のロボットの制御を行う。

## 7.1 制御手法

閉リンク機構のロボットに対して、仮想的に閉リンク機構と考え、制御する方法を示していく。関節変位  $q \in \Re^n$  と関節速度  $\dot{q} \in \Re^n$ 、関節駆動力  $\tau \in \Re^n$  から関節加速度  $\ddot{q}$  を求める計算を順動力学計算と呼び、次のように表す。ただし、 $n$  はロボットの自由度とする。

$$\ddot{q} = \text{FD}(q, \dot{q}, \tau) \quad (7.1)$$

また、 $q, \dot{q}, \ddot{q}$  を実現するための  $\tau$  を求める計算を逆動力学計算と呼び、次のように表す。

$$\tau = \text{ID}(q, \dot{q}, \ddot{q}) \quad (7.2)$$

$\tau = 0$  として閉リンク機構の順動力学計算を解くと、それを実現する加速度を求めることができる。この加速度を指令値加速度  $\ddot{q}^{ref}$  とすると、

$$\ddot{q}^{ref} = \text{FD}(q, \dot{q}, 0) \quad (7.3)$$

となる。ここで、式中の上付き文字  $(\circ)^{ref}$  は指令値を表す。 $\ddot{q}^{ref}$  を用いて逆動力学計算を行うと、

$$\tau^{ref} = \text{ID}(q, \dot{q}, \ddot{q}^{ref}) \quad (7.4)$$

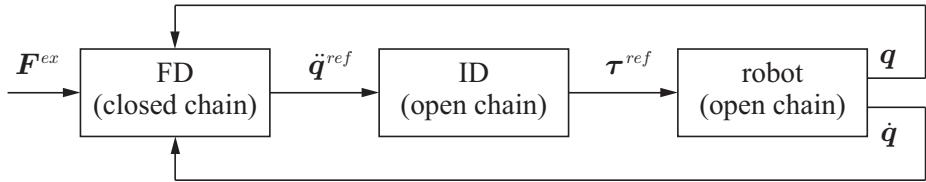
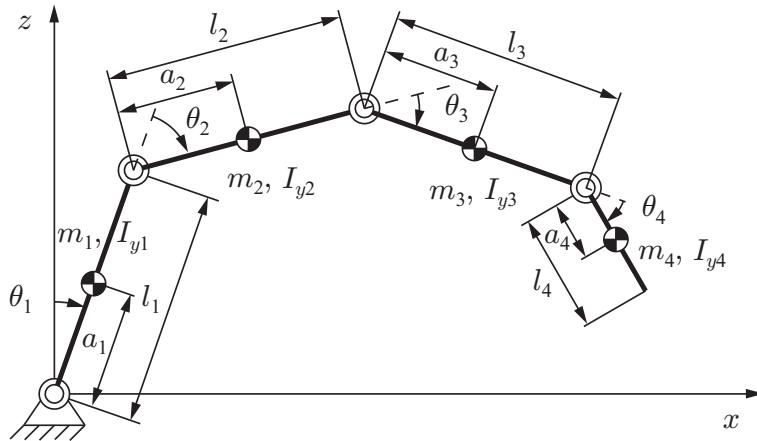
Fig. 7.1: Closed loop dynamic controller with virtual external force  $F^{ex}$ .

Fig. 7.2: A planar four link manipulator (robot).

となり、閉リンク機構の条件を満たすように制御するための関節駆動力  $\tau^{ref}$  を求めることができます。Eq. (7.3)を解く際に、 $\tau = 0$  の条件に加え、仮想的な外力  $F^{ex} \in \mathbb{R}^{6k}$  を加えると、閉リンク機構の条件を満たした、外力による運動を求めることができる。ただし、 $k$  はリンク数とする。

$$\ddot{q}^{ref} = \text{FD}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{0}, \mathbf{F}^{ex}) \quad (7.5)$$

ここで、 $\mathbf{F}^{ex}$  は各関節に加わる外力  $F_i$  をまとめた行列であり、ADA を解く際に用いられる。ただし、 $i = 1, 2, \dots, k$  である。 $\mathbf{F}^{ex}$  をフィードバックに用い、上式から求まる  $\ddot{q}^{ref}$  を Eq. (7.4) に代入して逆動力学計算を行うことで、閉リンク機構を考慮した制御が可能になる。つまり、閉リンク機構を考慮した順動力学計算から  $\ddot{q}^{ref}$  を求め、開リンク機構のモデルで逆動力学計算を行い  $\tau^{ref}$  を求め、開リンク機構のロボットの制御を行う。これをブロック線図で表すと Fig. 7.1 のようになる。

本章では、Fig. 7.2 に示す 4 節開リンク機構のロボットを対象として、制御を行っていく。このモデルの関節は、すべて  $y$  軸回りの回転関節であり、二次元モデルである。各リンクの長さは  $l_1 = l_2 = l_3 = 1.0 \text{ m}$ ,  $l_4 = 0.5 \text{ m}$ , 質量は  $m_i = 1.0 \text{ kg}$ , 各リンク重心の  $y$  軸回りの慣性テンソルは  $I_{yi} = m_i l_i^2 / 12 \text{ kg} \cdot \text{m}^2$

とした。また、重心位置は  $a_i = l_i/2$  とした。ただし、 $i = 1, 2, 3, 4$  である。このモデルの関節変位と外力は次のようになる。

$$\mathbf{q} = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 \end{bmatrix}^T \quad (7.6)$$

$$\mathbf{F}^{ex} = \left[ (\mathcal{F}_1^{ex})^T \quad (\mathcal{F}_2^{ex})^T \quad (\mathcal{F}_3^{ex})^T \quad (\mathcal{F}_4^{ex})^T \right]^T \quad (7.7)$$

## 7.2 制御例

### 7.2.1 制御入力なし

制御の確認を行う前に、制御入力を与えない場合に、このロボットがどのような挙動をするか確認するためのシミュレーションを行う。

#### シミュレーション

$\theta_1^{ini} = 60 \text{ deg}$ ,  $\theta_2^{ini} = 30 \text{ deg}$ ,  $\theta_3^{ini} = 150 \text{ deg}$ ,  $\theta_4^{ini} = 300 \text{ deg}$  の初期角度から、初速および制御入力なしの条件で自由落下するシミュレーションを行った。シミュレーション時間を 2 s, 積分の刻み幅を 1 ms, 重力加速度を  $z$  軸方向に  $-9.81 \text{ m/s}^2$  とした。

運動の様子を Fig. 7.3 にスナップショットで示す。また、シミュレーション結果を Fig. 7.4 に示す。Fig. 7.4 (a) は関節角度、(b) は関節速度、(c) は関節加速度、(d) は関節トルク、(e) は先端の位置、(f) は先端の速度である。Fig. 7.3, Figs. 7.4 (a)–(d) より、制御入力を与えない状態で、設定した初期状態から重力の影響を受け、自由落下する様子が確認できる。また、Figs. 7.4 (e), (f) より、先端の位置と速度が変化している。このモデルは、エネルギー保存則が成り立つことも確認しており、正しく順動力学計算が行えていると判断した。

### 7.2.2 4 節平行リンク機構として制御

開リンク機構のロボットに対して、仮想的に閉リンク機構と考えて制御していく。Fig. 7.1 に示したブロック線図の FD (closed loop) の計算を、Fig. 7.5 に示す 4 節平行リンク機構として行う。また、各関節に加える仮想的な外力  $\mathbf{F}^{ex} \in \mathbb{R}^{24}$  を次のように与える。

$$\mathbf{F}^{ex} = \mathbf{S} \boldsymbol{\tau}^{ex} \quad (7.8)$$

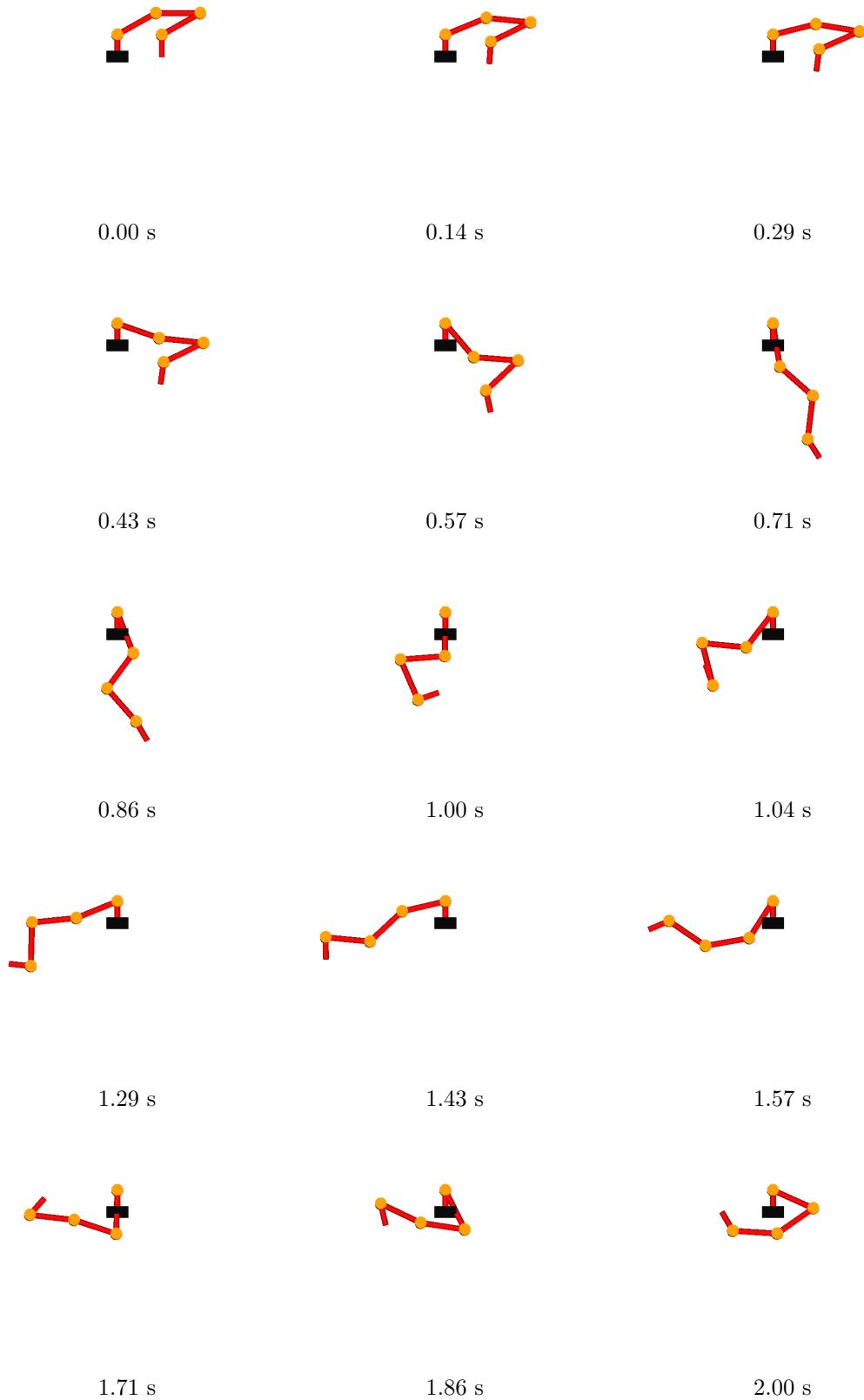


Fig. 7.3: Snapshots from simulation of the planar four link manipulator.

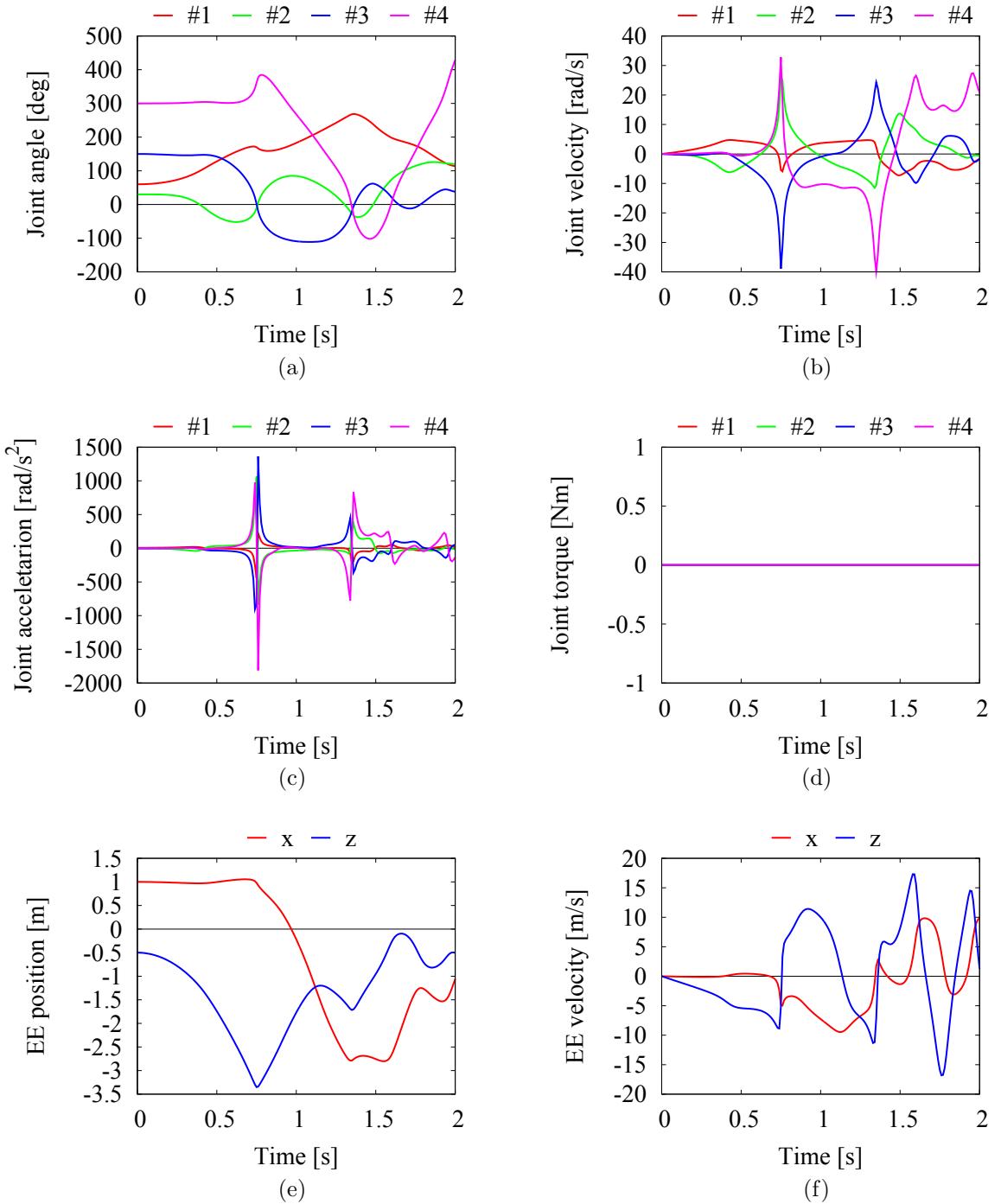


Fig. 7.4: Simulation results of the planar four link manipulator (not controlled): (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity.

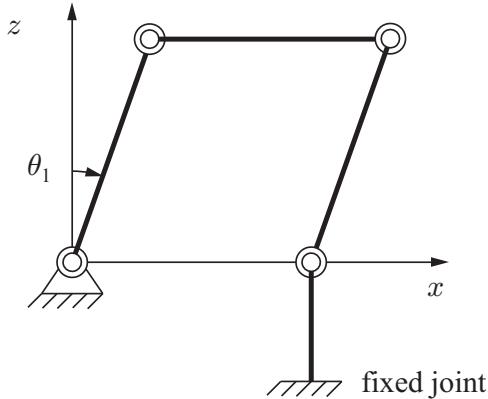
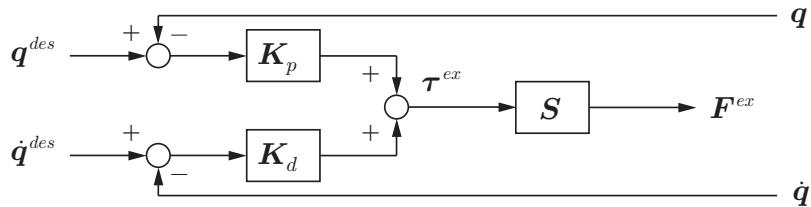


Fig. 7.5: A parallel four-bar linkage.

Fig. 7.6: Calculation of the virtual external force  $\mathbf{F}^{ex}$  for the parallel four-bar linkage.

ここで、

$$\tau^{ex} = \mathbf{K}_p (q^{des} - q) - \mathbf{K}_d (\dot{q}^{des} - \dot{q}) \quad (7.9)$$

$$\mathbf{S} = [S_1^T \ S_2^T \ S_3^T \ S_4^T]^T \quad (7.10)$$

と置いた。式中の  $\mathbf{K}_p \in \Re^{4 \times 4}$  と  $\mathbf{K}_d \in \Re^{4 \times 4}$  はフィードバックゲイン、 $S_i \in \Re^6$  は関節  $i$  の可動方向を表す行列、 $n_{J_i}$  は関節  $i$  の自由度であり、上付き文字  $(\circ)^{des}$  は目標値を表す。Eq. (7.9) を用いて、関節の目標角度  $q^{des}$  と目標角速度  $\dot{q}^{des}$  のフィードバックを行う。これをブロック線図で表すと Fig. 7.6 のようになる。このブロック線図は、Fig. 7.1 左側と繋がる。

### シミュレーション

制御なしの際と同様に、 $\theta_1^{ini} = 60 \text{ deg}$ ,  $\theta_2^{ini} = 30 \text{ deg}$ ,  $\theta_3^{ini} = 150 \text{ deg}$ ,  $\theta_4^{ini} = -60 \text{ deg}$  の初期角度から、初速なしの条件でシミュレーションを行った。Eq. (7.8) を Eq. (7.5) に代入して関節の指令加速度  $\ddot{q}^{ref}$  を求め、Eq. (7.4) に代入して関節の指令駆動力  $\tau^{ref}$  を求め、制御した。ここで、Eq. (7.9) のゲイ

ンを

$$\mathbf{K}_p = \text{diag}(5, 0, 0, 0) \text{ Nm} \quad (7.11)$$

$$\mathbf{K}_d = \text{diag}(5, 0, 0, 0) \text{ Nm} \cdot \text{s} \quad (7.12)$$

とし、第一関節の目標関節角度  $\theta_1^{des} = 0 \text{ deg}$ とした。シミュレーション時間 を 5 s, 積分の刻み幅を 1 ms, 重力加速度を  $z$  軸方向に  $-9.81 \text{ m/s}^2$  とした。

運動の様子を Fig. 7.7 にスナップショットで示す。また、シミュレーション結果を Figs. 7.8, 7.9 に示す。Fig. 7.8 (a) は関節角度、(b) は関節速度、(c) は関節加速度、(d) は関節トルク、(e) 先端の位置、(f) は先端の速度である。Fig. 7.7, Figs. 7.8 (a)–(c) より、 $\theta_1$  と  $\theta_3$ ,  $\theta_2$  と  $\theta_4$  は同じ方向に、 $\theta_1$ ,  $\theta_3$  と  $\theta_2$ ,  $\theta_4$  は逆に同じ大きさで動いており、4 節平行リンク機構の動きが確認できた。この動きは、関節 1 から関節 3 のトルクで実現していることが Fig. 7.8 (d) から分かる。また、Figs. 7.8 (e), (f) より、先端の位置が一定で、速度がゼロであり、先端位置が動かないように制御できていることが分かる。

Fig. 7.9 (a) は関節 1 に加える仮想的な力、(b) は第 1 関節の角度の誤差、(c) は第 1 関節の角速度の誤差である。Fig. 7.9 より、関節角度と関節角速度の誤差が共にゼロに収束しており、関節角度のフィードバックが行えていることを確認できた。

### 7.2.3 スライダークランク機構として制御

Fig. 7.1 に示したブロック線図の FD (closed loop) の計算を、Fig. 7.5 に示すスライダークランク機構として行う。また、 $\mathbf{F}^{ex}$  を次のように与える。

$$\mathcal{F}_4^{ex} = \mathcal{F}_e^{ex} \quad (7.13)$$

$$\mathcal{F}_1^{ex} = \mathcal{F}_2^{ex} = \mathcal{F}_3^{ex} = \mathbf{0} \quad (7.14)$$

ここで、

$$\mathcal{F}_e^{ex} = \mathbf{K}_p (\mathcal{X}_e^{des} - \mathcal{X}_e) - \mathbf{K}_d (\mathcal{V}_e^{des} - \mathcal{V}_e) \quad (7.15)$$

$$\mathcal{V}_e = \mathbf{J} \dot{\mathbf{q}} \quad (7.16)$$

$$(7.17)$$

と置いた。式中の、 $\mathbf{K}_p \in \mathbb{R}^{6 \times 6}$  と  $\mathbf{K}_d \in \mathbb{R}^{6 \times 6}$  はフィードバックゲイン、 $\mathcal{X}_e \in \mathbb{R}^6$  は先端の位置・姿勢、 $\mathcal{V}_e$  は先端の速度、 $\mathbf{J} \in \mathbb{R}^{6 \times 4}$  はヤコビ行列である。

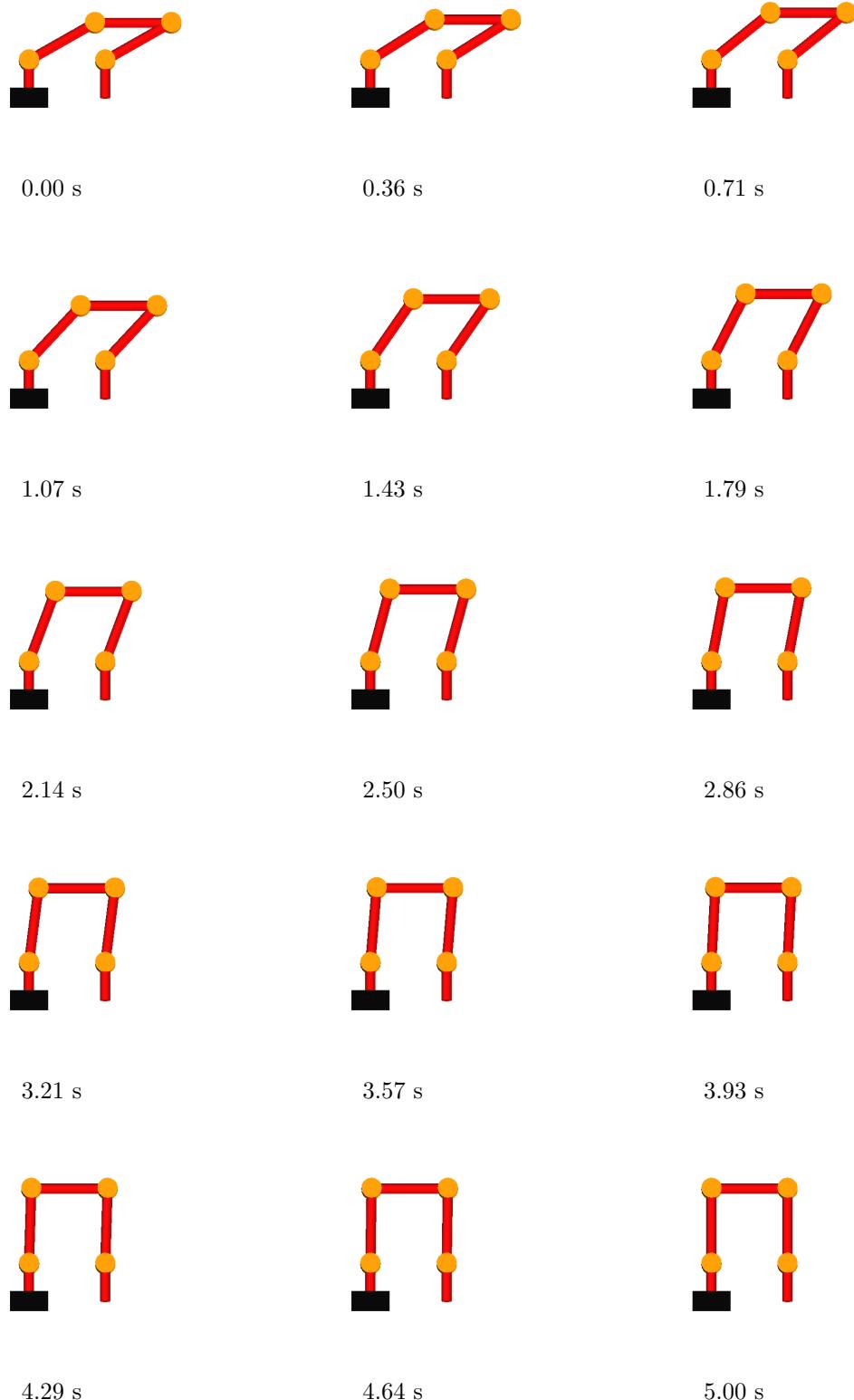


Fig. 7.7: Snapshots from simulation of the parallel four-bar linkage.

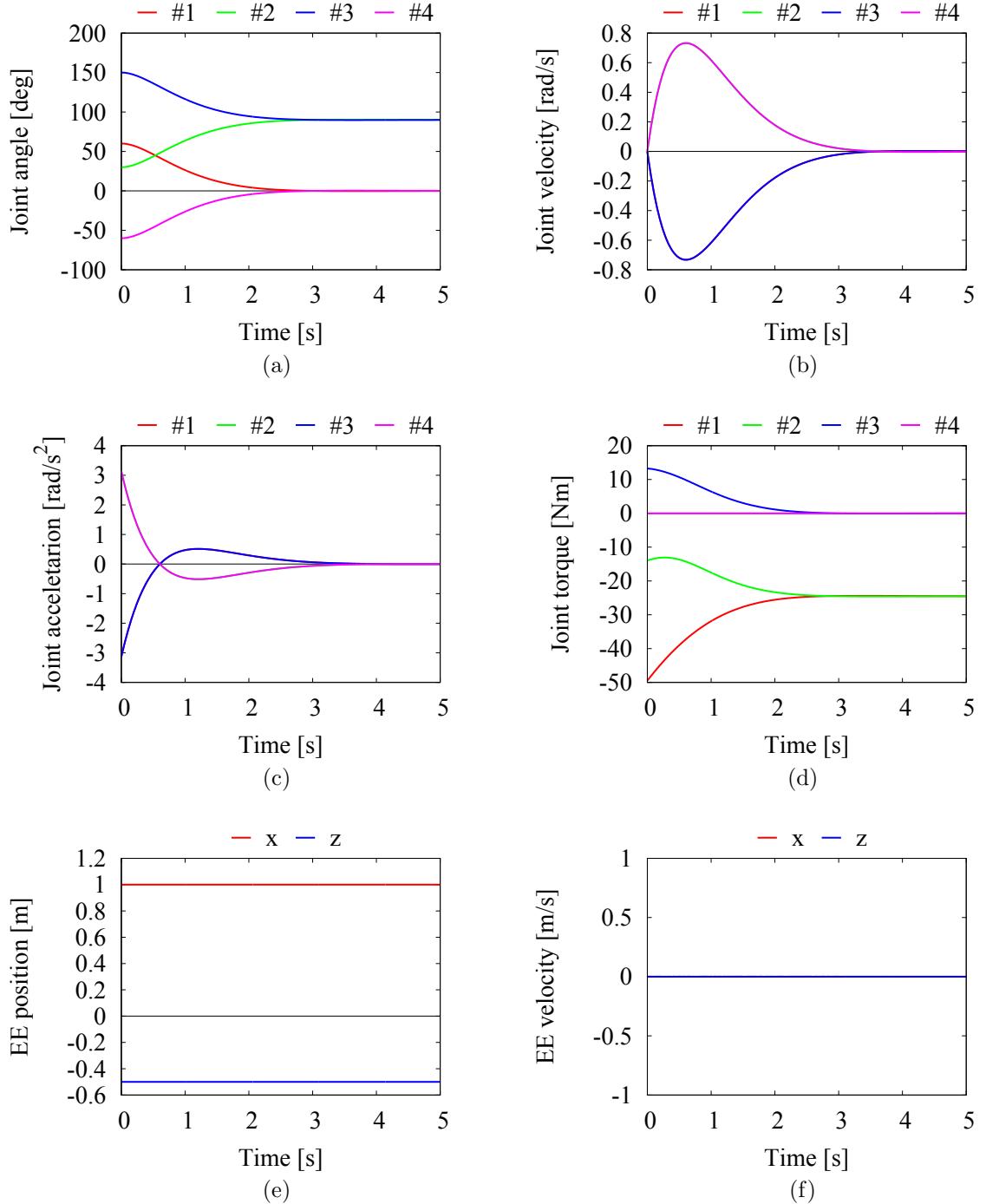


Fig. 7.8: Simulation results of the parallel four-bar linkage: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity.

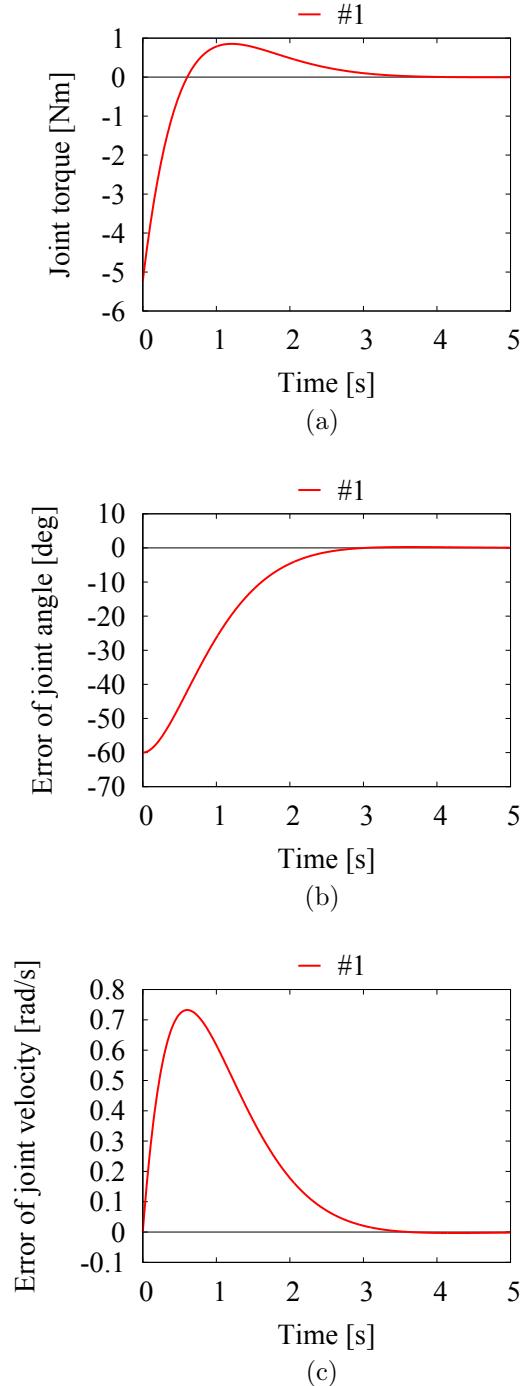


Fig. 7.9: Simulation results of the parallel four-bar linkage: (a) virtual external torque  $n_1^{ex}$ , (b) error of joint angle  $\theta_1$  and (c) error of joint velocity  $\dot{\theta}_1$ .

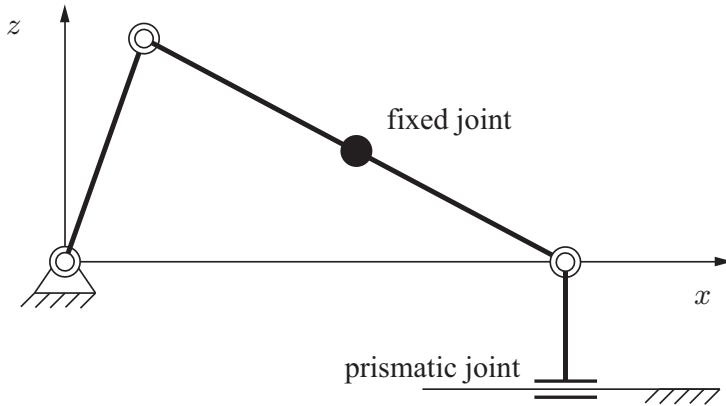
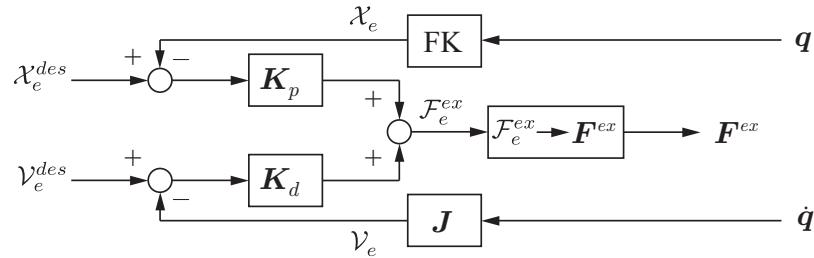


Fig. 7.10: A slider-crank mechanism.

Fig. 7.11: Calculation of the virtual external force  $\mathbf{F}^{ex}$  for the slider-crank mechanism.

Eq. (7.15) を用いて、先端の位置・姿勢と先端の速度のフィードバックを行う。これをブロック線図で表すと Fig. 7.11 のようになる。このブロック線図は、Fig. 7.1 左側と繋がる。

### シミュレーション

第1関節と第3関節の  $z$  軸方向の高さが等しくなる角度  $\theta_1^{ini} = 30 \text{ deg}$ ,  $\theta_2^{ini} = 85.7 \text{ deg}$ ,  $\theta_3^{ini} = 0 \text{ deg}$ ,  $\theta_4^{ini} = 64.3 \text{ deg}$  を初期角度とし、初速なしの条件でシミュレーションを行った。Eq. (7.13) を Eq. (7.5) に代入して関節の指令加速度  $\ddot{q}^{ref}$  を求め、Eq. (7.4) に代入して関節の指令駆動力  $\tau^{ref}$  を求め、制御した。ここで、Eq. (7.15) のゲインを

$$\mathbf{K}_p = \text{diag}(0, 0, 0, 10, 0, 0) \text{ kg/s}^2 \quad (7.18)$$

$$\mathbf{K}_d = \text{diag}(0, 0, 0, 10, 0, 0) \text{ kg/s} \quad (7.19)$$

とし、先端の  $x$  方向の目標位置を  $x_e^{des} = 1.49 \text{ m}$  とした。これは、先端の初期位置  $x_e^{ini}$  から  $-1 \text{ m}$  の位置である。シミュレーション時間を  $5 \text{ s}$ 、積分の刻み

幅を1 ms, 重力加速度を $z$ 軸方向に $-9.81 \text{ m/s}^2$ とした.

運動の様子をFig. 7.12にスナップショットで示す. また, シミュレーション結果をFigs. 7.13, 7.14に示す. Fig. 7.13 (a)は関節角度,(b)は関節速度,(c)は関節加速度,(d)は関節トルク,(e)先端の位置,(f)は先端の速度である. Fig. 7.12, Figs. 7.13 (a)–(c)より,  $\theta_1, \theta_2, \theta_4$ が動いているが,  $\theta_3$ は動いていないことが確認できた. また, Fig. 7.13 (d)より, 第1関節から第3関節を主に用いて制御していることが分かる. Figs. 7.13 (e), (f)より, 先端の位置は $z$ 軸方向には変位せず,  $x$ 軸方向のみに変位していることが確認できる. つまり, スライダークランク機構として制御できているといえる.

Fig. 7.14 (a)は関節4に加える仮想的な力の $x$ 成分,(b)は先端の $x$ 軸方向の位置の誤差,(c)は先端の $x$ 軸方向の速度の誤差である. Fig. 7.14より, 先端位置と先端速度の誤差が共にゼロに収束しており, 先端位置のフィードバックが行えていることを確認できた.

## 7.3 閉リンク機構の特異姿勢

閉リンク機構の順動力学計算では特異姿勢が存在することが知られている[40]. 4節平行リンク機構とスライダークランク機構の特異姿勢をFigs. 7.15, 7.16に示す. 4節平行リンク機構では $\theta_1 = \pm 90 \text{ deg}$ の姿勢, スライダークランク機構の場合は $\theta_2 = 0, \pm 180 \text{ deg}$ の姿勢が特異姿勢となる. 前章までに述べた制御を行う際に, このような特異姿勢がどのように影響するか, シミュレーションを行い検証していく. シミュレーションのパラメータは, 基本的には7.2節と同様のものを用いる.

### 7.3.1 4節平行リンク機構として制御

4節平行リンク機構の特異姿勢の近傍 $\theta_1^{ini} = 85 \text{ deg}, \theta_2^{ini} = 5 \text{ deg}, \theta_3^{ini} = 175 \text{ deg}, \theta_4^{ini} = -85 \text{ deg}$ を初期角度とし, 初速なしの条件でシミュレーションを行った. Eq. (7.8)をEq. (7.5)に代入して関節の指令加速度 $\ddot{q}^{ref}$ を求め, Eq. (7.4)に代入して関節の指令駆動力 $\tau^{ref}$ を求め, 制御した. ここで, 特異姿勢を通過するように, 第一関節の目標関節角度 $\theta_1^{des} = 95 \text{ deg}$ とした. シミュレーション時間を5 s, 積分の刻み幅を1 ms, 重力加速度を $z$ 軸方向に $-9.81 \text{ m/s}^2$ とした.

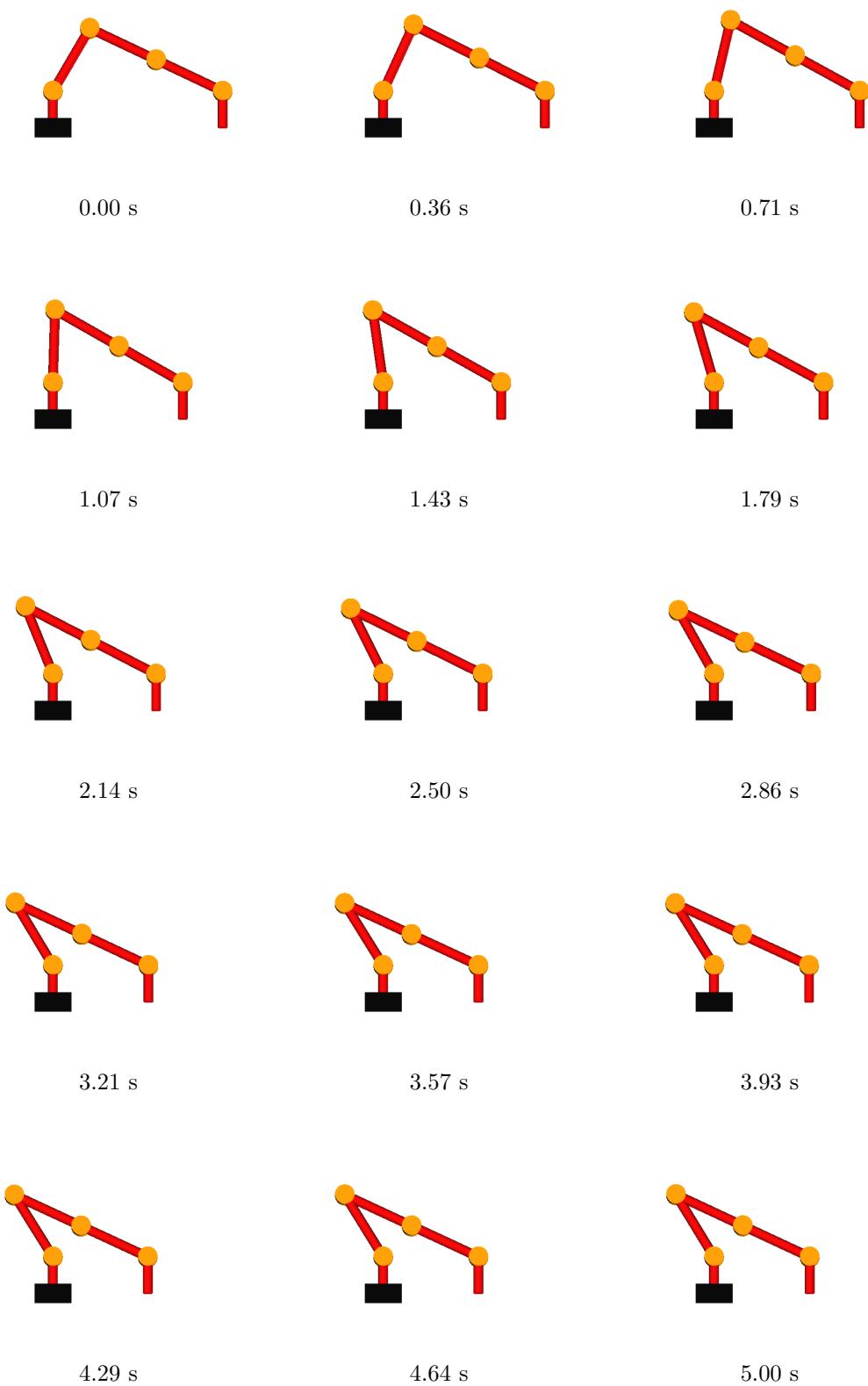


Fig. 7.12: Snapshots from simulation of the slider-crank mechanism.

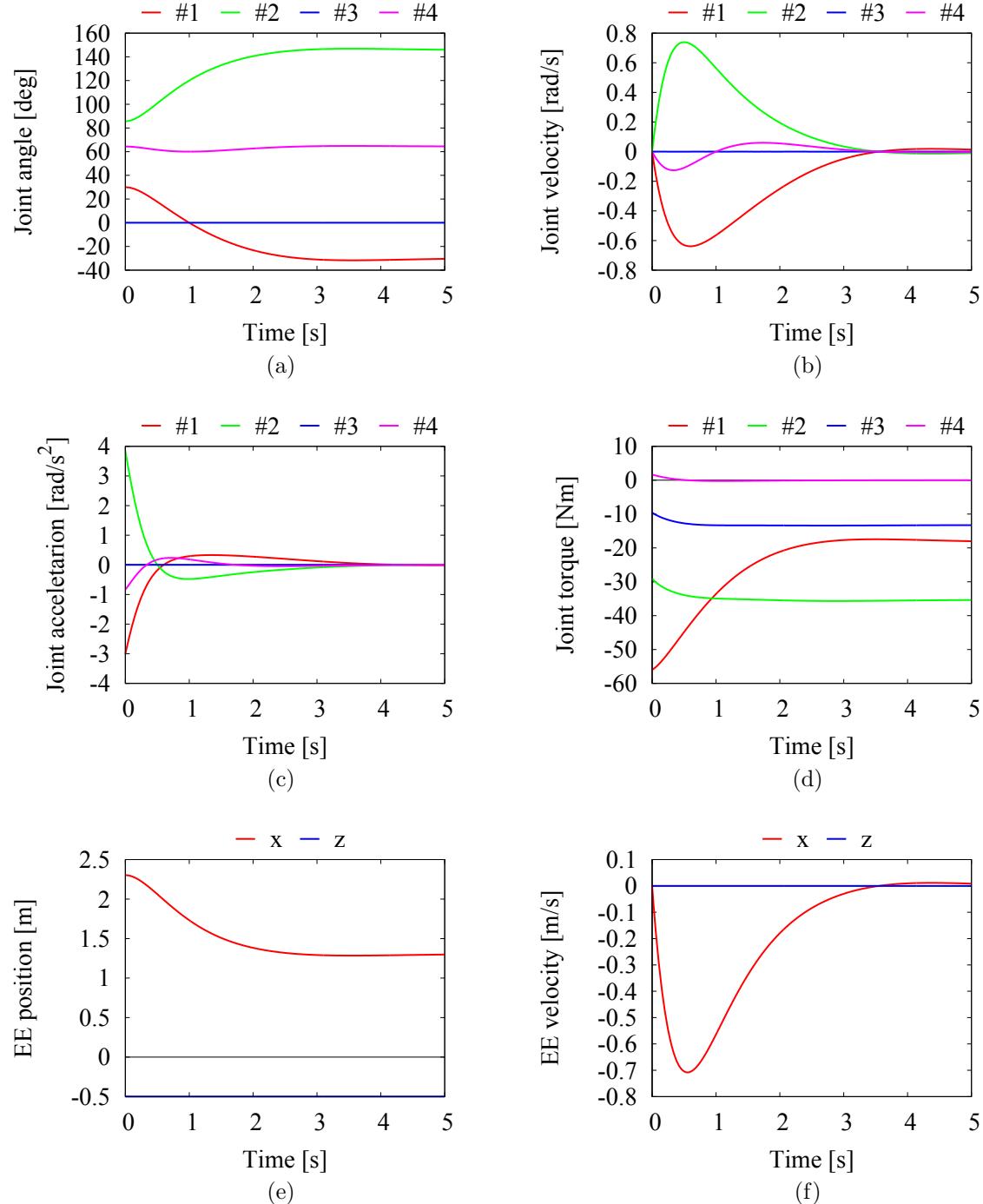


Fig. 7.13: Simulation results of the slider-crank mechanism: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity.

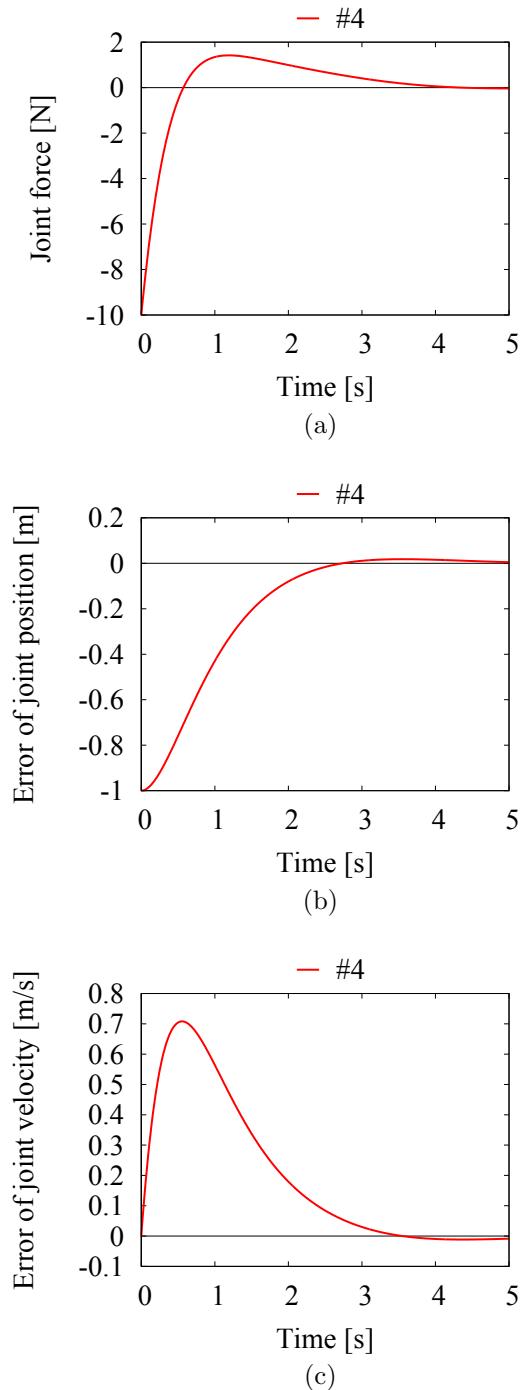


Fig. 7.14: Simulation results of the slider-crank mechanism: (a) virtual external force  $f_4^{ex}$ , (b) error of joint position  $r_{4x}$  and (c) error of joint velocity  $v_{4x}$ .

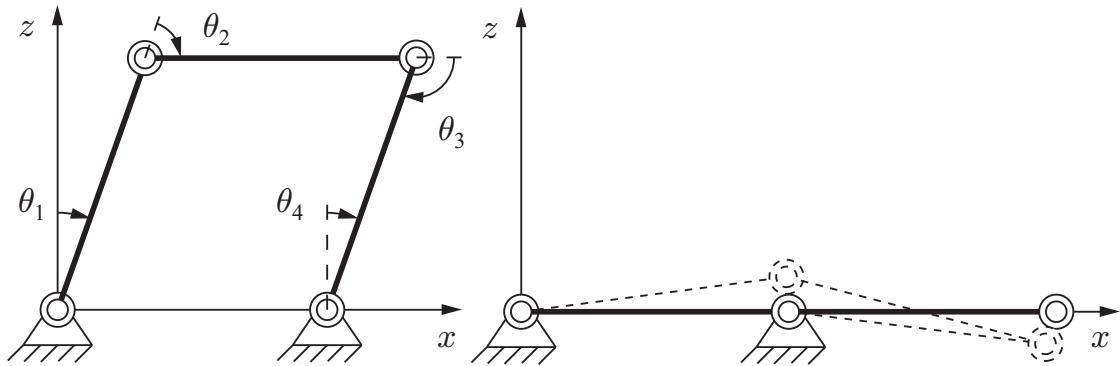


Fig. 7.15: Singular configuration of the four-bar linkage.

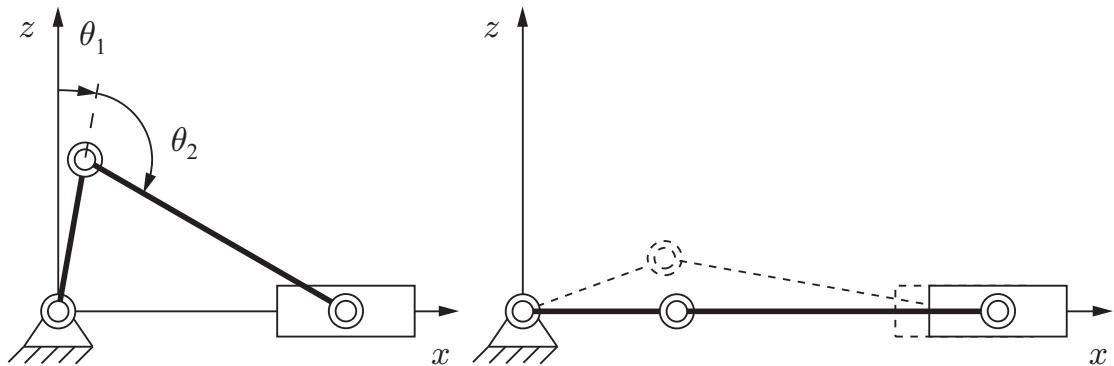


Fig. 7.16: Singular configuration of the slider-crank mechanism.

運動の様子を Fig. 7.17 にスナップショットで示す。また、シミュレーション結果を Figs. 7.18, 7.19 に示す。Fig. 7.18 (a) は関節角度、(b) は関節速度、(c) は関節加速度、(d) は関節トルク、(e) は先端の位置、(f) は先端の速度である。Fig. 7.17, Figs. 7.18 (a)–(c) より、特異姿勢である  $\theta_1 = 90$  deg を通過していることが分かる。この際、Fig. 7.18 (c) より関節加速度のジャンプを確認した。しかし、Fig. 7.18 (d) の関節トルクなど、他のグラフでは、波形のジャンプが発生しなかった。また、Figs. 7.18 (e), (f) より、先端の位置が一定で、速度がゼロであり、先端位置が動かないように制御できていることが分かる。

Fig. 7.19 (a) は関節 1 に加える仮想的な力、(b) は第 1 関節の角度の誤差、(c) は第 1 関節の角速度の誤差である。Fig. 7.19 より、関節角度と関節角速度の誤差が共にゼロに収束しており、関節角度のフィードバックが行えていることを確認できた。

このシミュレーションでは、関節加速度のジャンプによる影響はなかっ

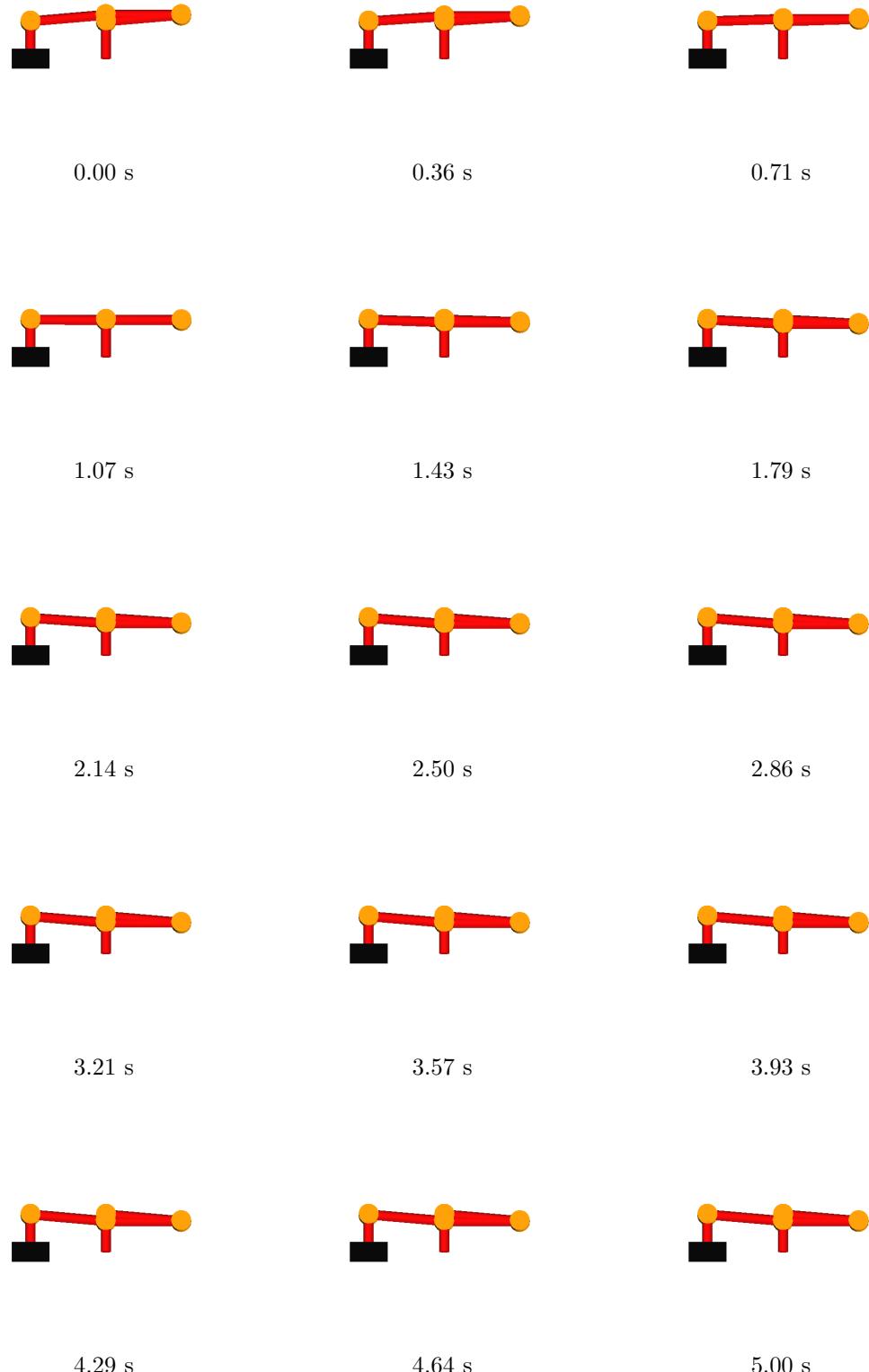


Fig. 7.17: Snapshots from simulation of the parallel four-bar linkage around singularity.

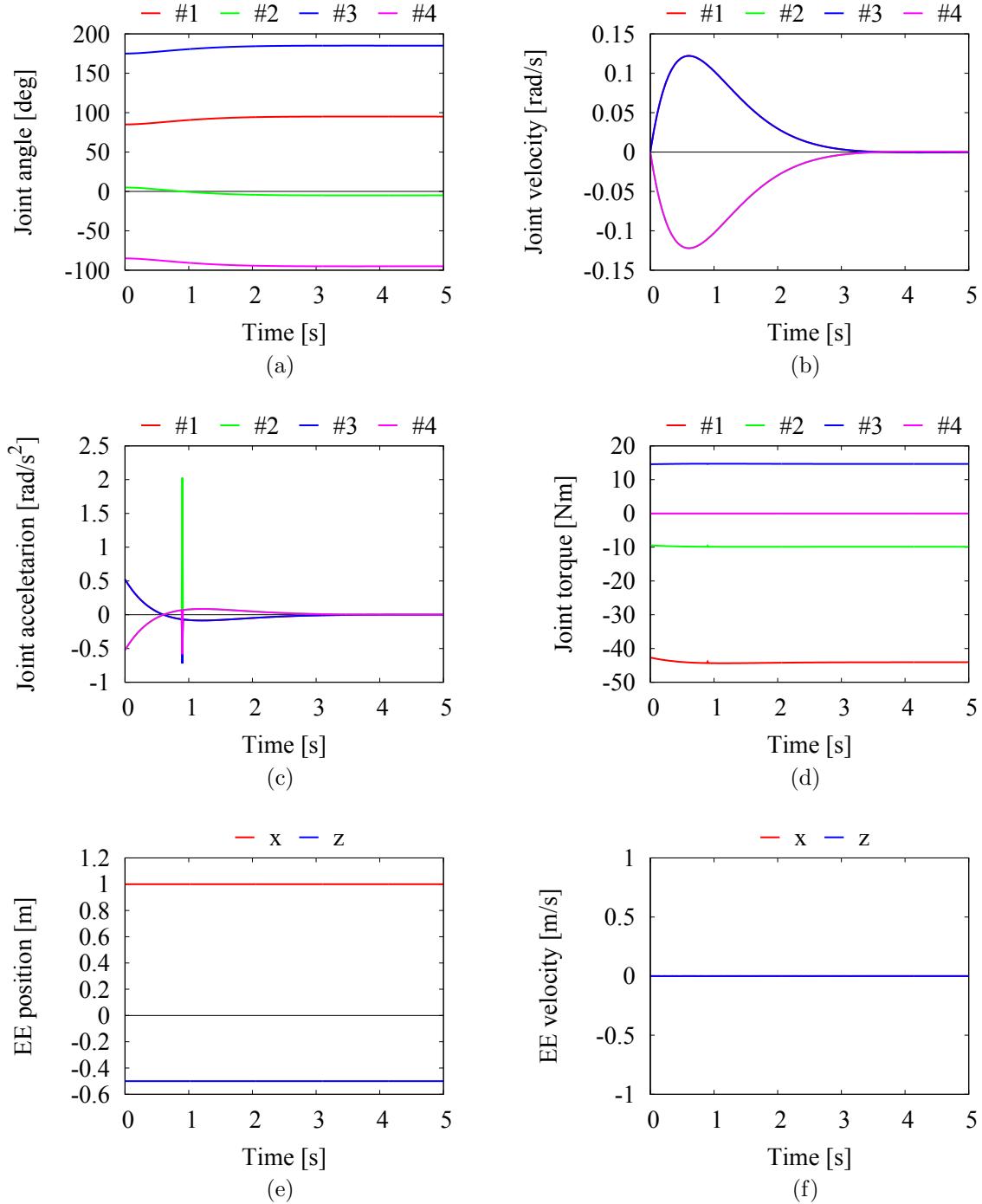


Fig. 7.18: Simulation results of the parallel four-bar linkage around singularity: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity.

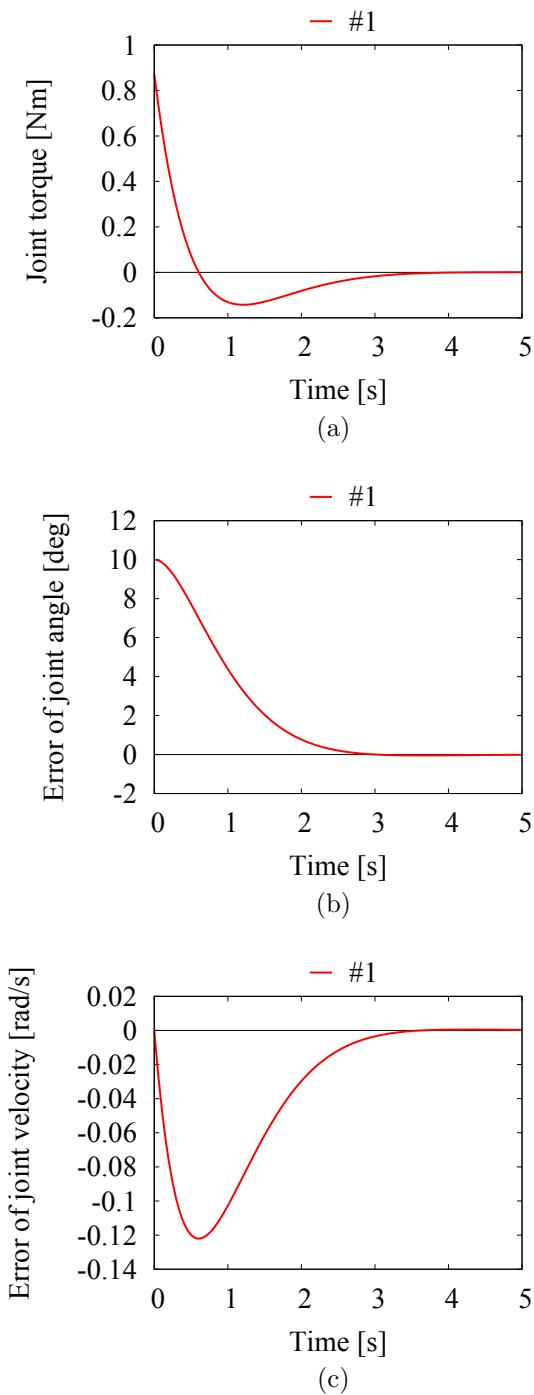


Fig. 7.19: Simulation results of the parallel four-bar linkage around singularity: (a) virtual external torque  $\tau_1^{ex}$ , (b) error of joint angle  $\theta_1$  and (c) error of joint velocity  $\dot{\theta}_1$ .

た。しかし、第3章で確認したように、加速度のジャンプがシミュレーション結果に影響した。同様に、制御へ影響を与える可能性がある。今後は、関節加速度のジャンプが発生する原因や制御へ影響を与える状況などを考えていく必要がある。

### 7.3.2 スライダークランク機構として制御

スライダークランク機構の特異姿勢の近傍  $\theta_1^{ini} = 80.0 \text{ deg}$ ,  $\theta_2^{ini} = 15.0 \text{ deg}$ ,  $\theta_3^{ini} = 0.0 \text{ deg}$ ,  $\theta_4^{ini} = 85.0 \text{ deg}$  を初期角度とし、初速なしの条件でシミュレーションを行った。Eq. (7.13) を Eq. (7.5) に代入して関節の指令加速度  $\dot{q}^{ref}$  を求め、Eq. (7.4) に代入して関節の指令駆動力  $\tau^{ref}$  を求め、制御した。ここで、特異姿勢で静止するように、先端の  $x$  方向の目標位置を  $x_e^{des} = 3.00 \text{ m}$  とした。シミュレーション時間を 20 s, 積分の刻み幅を 1 ms, 重力加速度を  $z$  軸方向に  $-9.81 \text{ m/s}^2$  とした。

運動の様子を Fig. 7.20 にスナップショットで示す。また、シミュレーション結果を Figs. 7.21, 7.22 に示す。Fig. 7.21 (a) は関節角度、(b) は関節速度、(c) は関節加速度、(d) は関節トルク、(e) 先端の位置、(f) は先端の速度である。Fig. 7.20, Figs. 7.21 (a)–(c) より、特異姿勢である  $\theta_2 = 0 \text{ deg}$  を通過していることが分かる。しかし、前のシミュレーションと同様に、過大な関節トルクが発生していないことが Fig. 7.21 (d) から分かる。Figs. 7.21 (e), (f) より、先端の位置は  $z$  軸方向には変位せず、 $x$  軸方向のみに変位していることが確認できる。つまり、スライダークランク機構として制御できている。

Fig. 7.22 (a) は関節 4 に加える仮想的な力の  $x$  成分、(b) は先端の  $x$  軸方向の位置の誤差、(c) は先端の  $x$  軸方向の速度の誤差である。Fig. 7.22 より、先端位置と先端速度の誤差が共にゼロに近づくが、特異姿勢を通過した後、スライダーが引き返し、特異姿勢で静止しなかった。これらのことから、特異姿勢を通過する際に、過大な関節トルクは発生させずに制御できるが、特異姿勢で静止させることは難しいことが分かった。

## 7.4 本章のまとめ

ADA に基づく閉リンク機構の順動力学計算を制御に応用する方法を示した。例として、4 節開リンク機構のロボットに対して、4 節平行リンク機構やスライダークランク機構のように、仮想的な閉リンク機構を設定し

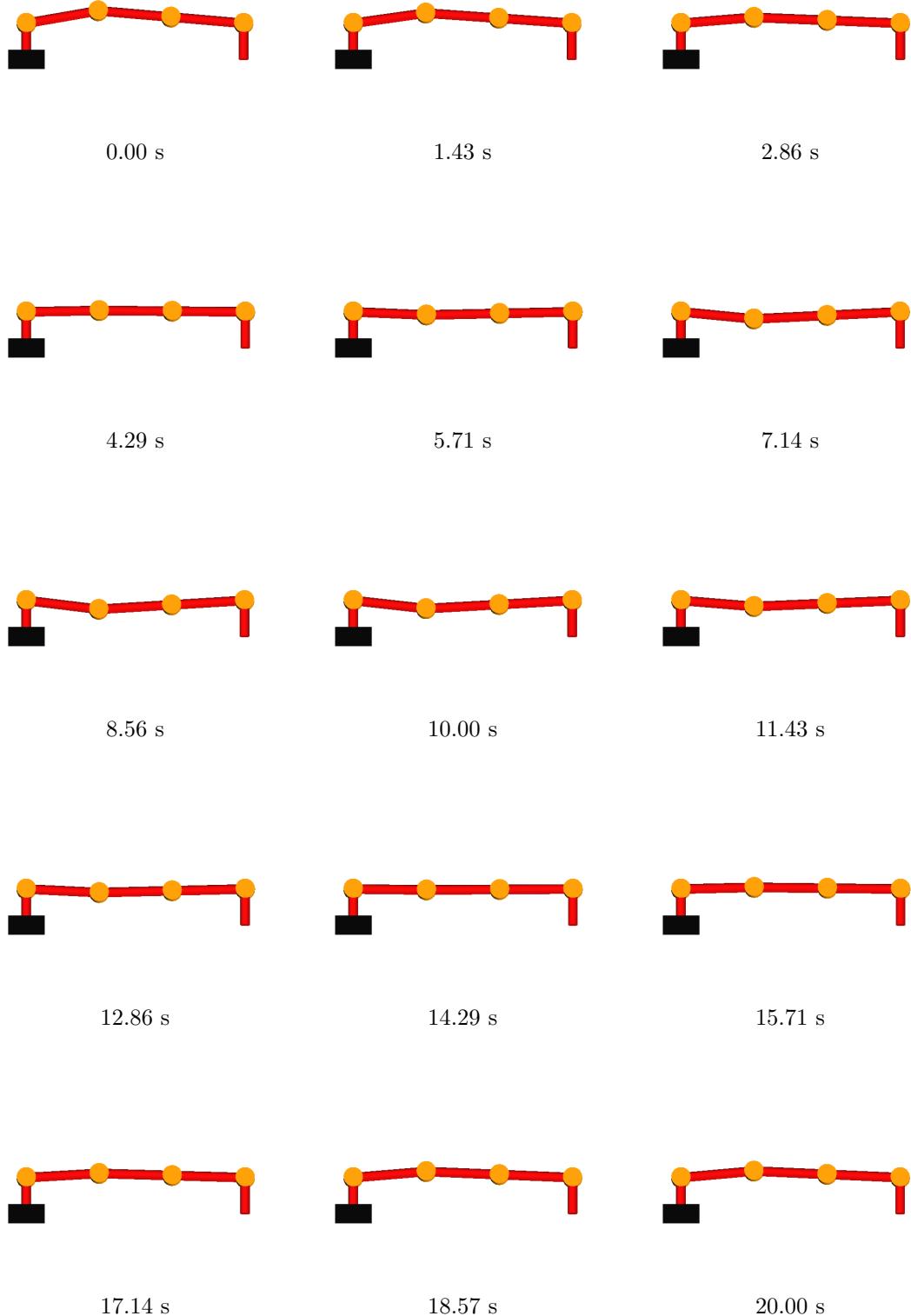


Fig. 7.20: Snapshots from simulation of the slider-crank mechanism around singularity.

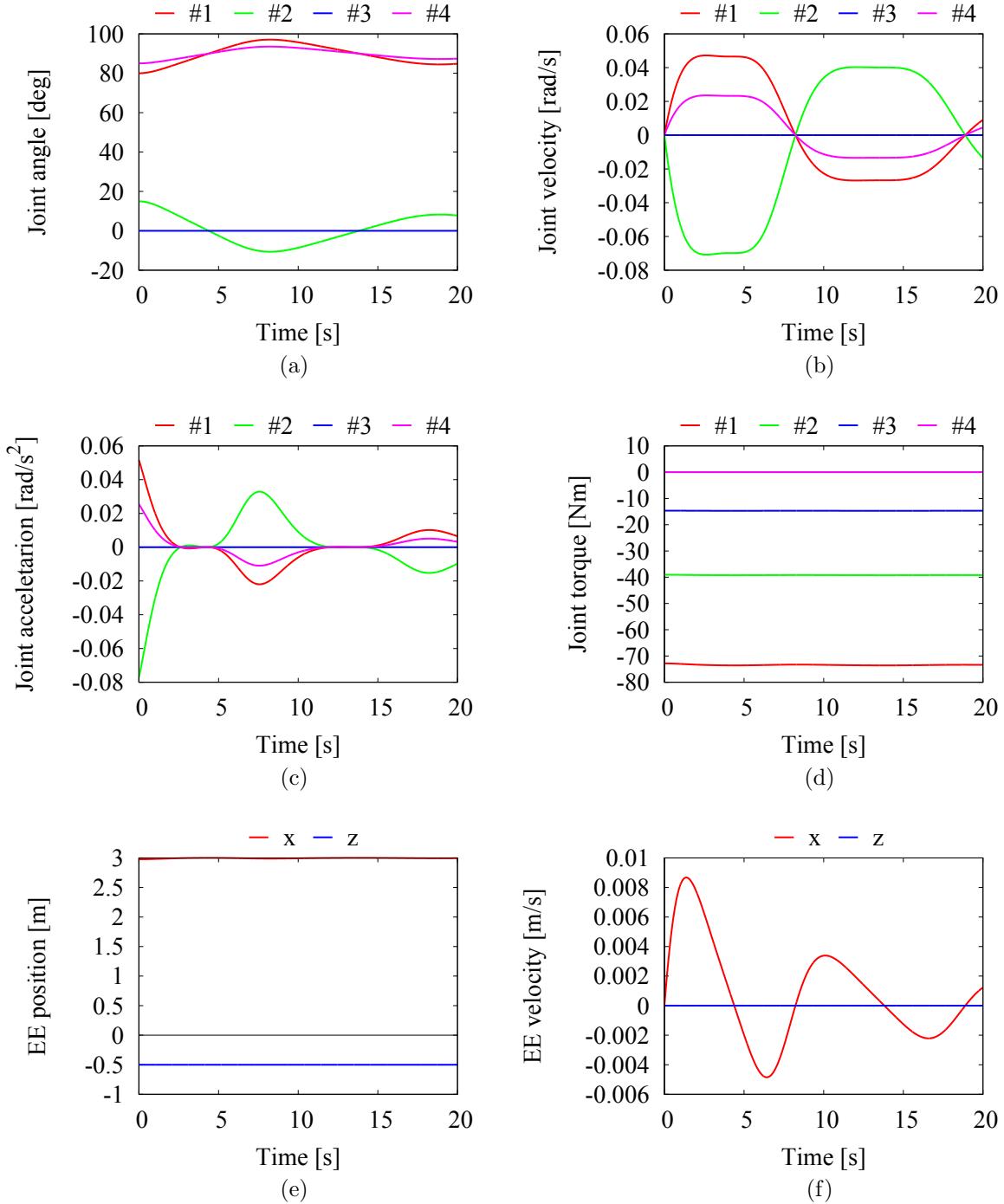


Fig. 7.21: Simulation results of the slider-crank mechanism around singularity: (a) joint angle, (b) joint velocity, (c) joint acceleration, (d) joint torque, (e) EE position and (f) EE velocity.

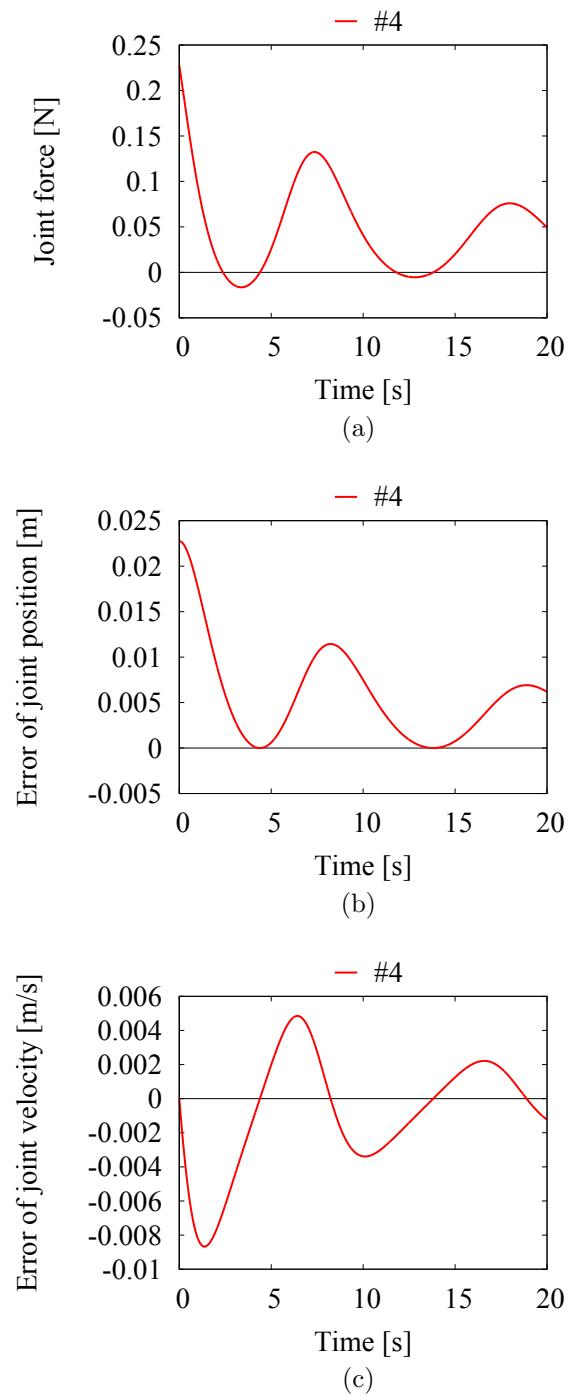


Fig. 7.22: Simulation results of the slider-crank mechanism around singularity: (a) virtual external force  $f_4^{ex}$ , (b) error of joint position  $r_{4x}$  and (c) error of joint velocity  $v_{4x}$ .

て制御した。この際、仮想的な外力を与えることで、関節位置や先端位置のフィードバック制御が可能であることを示した。

今後の課題は、浮遊ベースのモデルへの適用や三次元モデルへの拡張などである。第3章で述べたように、閉リンク機構の特異姿勢で発生する関節加速度のジャンプはシミュレーション結果に影響した。同様に、制御へ影響を与える可能性がある。この関節加速度のジャンプが発生する原因や制御への影響を与える状況などを考えていく必要がある。

# 第8章 結論および今後の研究課題

## 8.1 結論

本研究は、閉リンク機構を含むリンク系のシミュレーションを効率的に行うために、高速な順動力学エンジンを開発することを目的とした。そのアプローチとして、数値・数式ハイブリッド手法による順動力学計算(ADA)の高速化を行い、有用性を検証した。また、ロボットのシミュレータを作成し、人型ロボットのバランス制御の検証に用いた。さらに、ADAに基づく閉リンク機構の順動力学計算を制御に応用する方法を示した。以下に、各章ごとの要点をまとめる。

### 第2章 順動力学計算法の比較

ABAとDCA、ADAの三つの順動力学計算アルゴリズムを統一的な表記で導出し、比較した。その結果、単一CPUで閉リンク機構のシミュレーションを行う場合はABAが最も高速であるが、閉リンク機構のシミュレーションや並列計算を行う場合はADAが最も効率的と考えられることを示した。また、閉リンク機構のシミュレーションを効率的に行うために、ADAに基づく順動力学エンジンを開発することを決定した。

### 第3章 Assembly-Disassembly Algorithm (ADA)

ADAに基づくリンク系のモデリング手法として、シリアルリンク機構とプランチリンク機構、閉リンク機構の三つの機構の計算手法を示し、シリアルリンク機構から閉リンク機構へ容易に拡張できることを示した。ここで、閉リンク機構では、特異姿勢が存在することを確認した。また、三次元の浮遊ベースのモデリングに対して、単位クォータニオンによる計算手法を示した。さらに、それぞれのモデルでシミュレーションを行い、正しく順動力学計算が行われていることを示した。

#### 第4章 数値・数式ハイブリッドモデリング

ADAを用いた数値・数式ハイブリッドモデリングを行い、リンク系の運動方程式を導出した。また、固定ベースで接続された二次元モデルと浮遊ベースを含んだ三次元モデルのシミュレーションを行い、計算時間と比較した。その結果、ハイブリッドモデリングは、二次元モデルではADAの約60%，三次元モデルではADAの約66%となり、計算時間を短縮できた。

#### 第5章 マルチボディダイナミクスシミュレータ

ADAを用いた数値・数式ハイブリッドモデリングを行う順動力学エンジンを開発し、マルチボディダイナミクスシミュレータへ適用した。ここでは、このシミュレータのプログラムの流れやファイル構成を示した。また、例として、7自由度ロボットアームPA-10のシミュレーションを行い、正しく順動力学計算が行われていることを確認した。

#### 第6章 シミュレータの実用例

人型ロボットHOAP-2のシミュレータを作成し、バランス制御の検証に用いた例を示した。ここでは、突発的な衝撃力が肩に加わった際の人間の姿勢運動戦略を二つ提案した。一つ目は、比較的小さな衝撃力に対して、上半身を垂直軸周りに回転させてバランスを保つUpper-body rotation strategyである。二つ目は、大きな衝撃力に対して、上半身の垂直軸周りの回転と水平軸周りの伸展・屈曲を組み合わせてバランスを保つCombined rotation-extention/flexion strategyである。人間とロボットの間には、機構や自由度などの違いがあるが、この二つの姿勢運動戦略は、比較的容易に人型ロボットへ導入でき、バランス制御が行えることを示した。

#### 第7章 閉リンク機構を考慮した制御

ADAに基づく閉リンク機構の順動力学計算を制御に応用する方法を示した。例として、4節開リンク機構のロボットに対して、4節平行リンク機構やライダークランク機構のように、仮想的な閉リンク機構を設定して制御した。この際、仮想的な外力を与えることで、関節位置や先端位置のフィードバック制御が可能であることを示した。

以上をまとめると本研究では、主に以下のことを行った。

- 閉リンク機構を含むリンク系のシミュレーションを最も効率的に行うことができる順動力学計算法は、ADAであることを示した。
- 数値・数式ハイブリッドモデリングを行うことで、ADAの計算効率が向上することを示した。

- 開発した順動力学エンジンを搭載したシミュレータを作成し，ロボット開発に役立たせた．
- ADAに基づく閉リンク機構の順動力学計算を制御に応用することで，閉リンク機構のロボットを仮想的に閉リンク機構として制御できることを示した．

## 8.2 今後の研究課題

本研究の今後の課題を述べる．

### 1. ハイブリッドモデリングに対して，さらなる検証を行う．

ハイブリッドモデリングの計算時間と比較する際に，用意する数式の自由度による違いを検証し，あらかじめ用意しておく数式の自由度を決定する必要がある．また，既存のロボットの機構を把握し，それに合わせて用意する数式の機構を決め，一般的なロボットの大半にハイブリッドモデリングが対応できるようにしていく．

### 2. ハイブリッドモデリング以外のアプローチで計算効率の向上を図る．

この課題に対して二つのアプローチで解決を目指す．一つ目は，並列計算を行うことである．ADAは $n$ 個のCPUによる並列計算で $O(\log n)$ の計算量となるアルゴリズムであり，並列計算を行うことで計算効率の向上が見込める．二つ目は，球面関節を用いたモデリングを行うことである．三つの回転関節を一つの球面関節にまとめることで計算量を削減できると考えられる．また，ADAは関節の自由度が大きくなるにつれて，計算量が少なくなるという特徴を持つ．

### 3. 球面関節のモデリング手法の決定する．

球面関節の姿勢表現は，オイラー角を用いる手法と単位クオータニオンを用いる手法がある．オイラー角を用いる手法では，特異姿勢があるため，一般的には単位クオータニオンが用いられている．しかし，先に述べたように，三つの回転関節を一つの球面関節として表現しようとした場合，各関節の角度が求まらず，関節の可動範囲の考慮などができるない．そのため，オイラー角でのモデリングを検討していく．オイラー角の特異姿勢に関しては，特異点適合法[48]が適用できるか解析し，実用性や計算効率の評価・検証を行う必要がある．

4. 第6章で述べた人型ロボットのバランス制御を実機へ適用する .

この実機実験を通して，シミュレータと実機との違いの考察や，制御の実現性や有用性の検証を行う必要がある . また，第7章で述べた閉リンク機構を考慮した制御を適用し，バランス制御を改良する必要がある .

5. 閉リンク機構を考慮した制御を拡張する .

ADAに基づく閉リンク機構の順動力学計算を制御に応用する際に，固定ベースに接続されたロボットに対して適用してきた . 今後は，浮遊ベースのモデルへの適用や三次元モデルへの拡張などを行う必要がある . これにより，宇宙ロボットや人型ロボットのように，ベース部が固定されていないロボットに対してこの制御を適用できるか検討していく .

6. 閉リンク機構の特異姿勢の扱いを考える .

閉リンク機構の特異姿勢で発生する関節加速度のジャンプはシミュレーション結果に影響した . 同様に，制御へ影響を与える可能性がある . この関節加速度のジャンプが発生する原因や制御への影響を与える状況などを考えていく必要がある .

## 謝辞

本研究は，2010年4月より2012年1月の期間に東京都市大学大学院工学研究科機械システム工学専攻ロボティックライフサポート研究室において行なわれた。

本研究を行うにあたり，指導教官である金宮好和教授ならびに佐藤大祐講師には，的確なご指導をしていただきました。また，国内外での学会発表をなど，様々な体験をさせていただきました。この場を借りて，深い感謝の意を表します。

研究生活では，原直行先輩，同期の岩渕羽純，佐藤冬樹，西井達哉，羽鳥裕樹，半田陽一に，様々な面で支えてもらいました。先輩と同期のおかげで，公私ともに充実した研究生活を送ることができました。心から感謝いたします。また，後輩の三橋勝，梶原直人には，研究のサポートや資料作成の手伝いをしてもらいました。私のアバウトな要求に対しても，求めている事を感じ取り，迅速かつ適切な対応をしてくれて，とても助かりました。研究室の卒業生である原田先輩，宮坂先輩，そして多くの後輩達には，研究やその他に関しアドバイス・協力をしていただきました。ありがとうございました。

最後に，温かい励ましをいつも送り続けてくれた両親そして姉に，心から感謝いたします。

平成24年2月29日高橋 準



## 参考文献

- [1] R. Featherstone, *Robot Dynamics Algorithms*, Boston, Kluwer Academic Publishers, 1987.
- [2] 清水信行, 今西悦二郎: マルチボディダイナミクス(1) 基礎理論 . コロナ社 , 2006.
- [3] 田島洋: マルチボディダイナミクスの基礎 3次元運動方程式の立て方 . 東京電機大学出版局 , 2006.
- [4] R. Featherstone, *Rigid Body Dynamics Algorithms*, Canberra, AU: Springer, 2007.
- [5] 小林信之, 杉山博之: MATLABによる振動工学 基礎からマルチボディダイナミクスまで . 東京電機大学出版局 , 2008.
- [6] 遠山茂樹, 渡辺直樹, 田野博: “閉ループを持つ機構の効率的な順動力学解法 第1報 漸化式形式の順動力学解法”, 日本ロボット学会誌 , vol. 14, no. 2, pp. 279–286, 1996.
- [7] 中村仁彦, 山根克, 永嶋史朗: “構造変化を伴うリンク系の動力学計算法とヒューマンフィギュアの運動計算”, 日本ロボット学会誌 , vol. 16, no. 8, pp. 1152–1159, 1998.
- [8] 岩村誠人, 古賀智久, 永尾真史, 尾崎弘明: “マルチボディシステムの運動の最適化に関する研究:閉リンク系の場合”, Dynamics & Design Conf. , pp. 131-1–131-5, 2007.

- [9] F. Kanehiro, H. Hirukawa and S. Kajita, “OpenHRP: Open architecture humanoid robotics platform,” *The Int. J. of Robotics Research*, vol. 23, no. 2, pp. 155–165, Feb. 2004.
- [10] 中岡慎一郎, 山野辺夏樹, 比留川博久, 山根克, 川角祐一郎：“分散コンポーネント型ロボットシミュレータOpenHRP3”，日本ロボット学会誌，vol. 26, no. 5, pp. 399–406, 2008.
- [11] (2012, Jan.). OpenHRP3 Official Site. [Online]. Available: <http://www.openrtp.jp/openhrp3/jp/index.html>
- [12] K. Yamane and Y. Nakamura, “Efficient parallel dynamics computation of human figures,” in *Proc. of the 2002 IEEE Int. Conf. on Robotics and Automation*, Washington, 2002, pp. 530–537.
- [13] K. Yamane and Y. Nakamura, “Parallel O(log N) algorithm for dynamics simulation of humanoid robots,” in *Proc. of the 6th IEEE-RAS Int. Conf. on Humanoid Robots*, Genova, Italy, 2006, pp. 554–559.
- [14] J. E. Lloyd, “Fast Implementation of Lemke’s Algorithm for Rigid Body Contact Simulation,” in *Proc. of the 2005 IEEE Int. Conf. on Robotics and Automation*, Barcelona, Spain, 2005, pp. 4538–4543.
- [15] K. G. Murty, *Linear Complementarity: Linear and Non-linear Programming*, Heldermann Verlag, 1988.
- [16] 山根克, 中村仁彦：“ピボット法に基づくLCPの解法とその衝突・接触シミュレーションへの応用”, 第25回日本ロボット学会学術講演会予稿集, 2007.
- [17] R. Featherstone, “The calculation of robot dynamics using articulated-body inertias,” *The Int. J. of Robotics Research*, vol. 2, no. 1, pp. 13–30, Mar. 1983.
- [18] S. Nakaoka, S. Hattori, F. Kanehiro, S. Kajita and H. Hirukawa, “Constraint-based dynamics simulator for humanoid robots with shock absorbing mechanisms,” in *Proc. of the 2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Syst.*, San Diego, 2007, pp. 223–232.
- [19] (2012, Jan.). Open Dynamics Engine. [Online]. Available: <http://www.ode.org/>

- [20] (2012, Jan.). Open Dynamics Engine Community Wiki. [Online]. Available: <http://opende.sourceforge.net/wiki/>
- [21] 出村公成 , “ロボットシミュレーション Open Dynamics Engine によるロボットプログラミング” , コロナ社 , 2008.
- [22] D. Baraff, “Analytical methods for dynamic simulation of non-penetrating rigid bodies,” in *SIGGRAPH '89 Proc. of the 16th Ann. Conf. on Computer graphics and interactive techniques*, NY, USA, 1989, pp. 554–559.
- [23] (2012, Jan.). Webots 6 at AAI Japan, Co. Ltd. [Online]. Available: <http://www.aai.jp/>
- [24] (2012, Jan.). V-REP at HiBot. [Online]. Available: <http://www.hibot.co.jp/>
- [25] (2012, Jan.). Game Physics Simulation. [Online]. Available: <http://bulletphysics.org>
- [26] (2012, Jan.). Adams at MSC Software Simulating Reality, Delivering Certainty. [Online]. Available: <http://www.msccsoftware.co.jp/products/adams/>
- [27] E. J. Haug, *Computer-Aided Kinematics and Dynamics of Mechanical Systems*, Allyn and Bacon, 1996. ( 松井邦人(訳) , 横村幸辰(訳) , 井浦雄司(訳) : コンピュータを利用した機構解析の基本 . 大河出版社 , 1996. )
- [28] (2012, Jan.). LMS DADS at LMS Engineering Innovation. [Online]. Available: <http://www.lmsjapan.com/DADS>
- [29] J. Y. S. Luh, M. W. Walker and R. P. C. Paul, “Online computational scheme for mechanical manipulators,” *ASME J. Dyn. Syst., Meas., Contr.*, vol. 102, pp. 69–76, 1980.
- [30] M. W. Walker and D. E Orin, “Efficient dynamic computer simulation of robot mechanisms,” *ASME J. Dyn. Syst., Meas., Contr.*, vol. 104, pp. 205–211, 1982.
- [31] D. S. Bae and E. J. Haug, “A recursive formulation for constrained mechanical system dynamics: Part I. Open loop systems,” *Mechanics of Structures and Machines*, vol. 15, no. 3, pp. 359–382, 1987.

- [32] D. S. Bae and E. J. Haug, “A recursive formulation for constrained mechanical system dynamics: Part II. Closed loop systems,” *Mechanics of Structures and Machines*, vol. 15, no. 4, pp. 481–506, 1987–88.
- [33] D. E. Rosenthal, “An order n formulation for robotic systems,” *J. of the Astronautical Sciences*, vol. 38, no. 4, pp. 511–529, 1990.
- [34] D. Baraff, “Linear-Time dynamics using lagrange multipliers,” *Comput. Graph. Proc.*, pp. 137–146, 1996.
- [35] K. Yamane and Y. Nakamura, “O(n) forward dynamics computation of open kinematic chains based on the principle of virtual work,” *IEEE Trans. Robot. Autom.*, vol. 3, pp. 2824–2831, 2001.
- [36] A. Jain, “Unified formulation of dynamics for serial rigid multibody systems,” *J. of Guidance, Control and Dynamics.*, vol. 14, pp. 531–542, 1991.
- [37] A. Fijany, I. Sharf and G. M. T. D’Eleuterio, “Parallel O(log N) algorithms for computation of manipulator forward dynamics,” *IEEE Trans. Robot. Autom.*, vol. 11, no. 3, pp. 389–400, 1995.
- [38] R. Featherstone, “A divide-and-conquer articulated-body algorithm for parallel O(log((n))) calculation of rigid-body dynamics Part 1: Basic algorithm,” *The Int. J. of Robotics Research*, vol. 18, no. 9, pp. 867–875, Sept. 1999.
- [39] R. Featherstone, “A divide-and-conquer articulated-body algorithm for parallel O(log((n))) calculation of rigid-body dynamics Part 2: Trees, loops, and accuracy,” *The Int. J. of Robotics Research*, vol. 18, no. 9, pp. 876–892, Sept. 1999.
- [40] R. Mukherjee and K. Anderson, “Orthogonal complement based divide-and-conquer algorithm (O-DCA) or constrained multibody systems,” *Nonlinear Dynamics*, vol. 48, no. 1–2, pp. 199–215, Mar. 2006.
- [41] K. Yamane and Y. Nakamura, “Comparative study on serial and parallel forward dynamics algorithms for kinematic chains,” in *The int. J. of Robotics Research*, vol. 28, no. 5, pp. 622–629, May 2009.
- [42] 穴井宏和，菅野政明 “数値・数式ハイブリッド手法に基づく高品質計算とその展開,” *計測と制御*, vol. 5, 2010.

- [43] A. W. Strzebonski, “Cylindrical algebraic decomposition using validated numerics,” *The J. of Symbolic Computation*, vol. 41, no. 9, pp. 1021–1038, 2006.
- [44] H. Iwane, H. Yanami, H. Anai and K. Yokoyama, “An effective implementation of a symbolic-numeric cylindrical algebraic decomposition for quantifier elimination,” *Proceedings of the 2009 conference on Symbolic numeric computation*, 2009.
- [45] 関川浩 , “数式処理と数値計算の融合,” *情報処理*, vol. 50, 2009.
- [46] E. Hansen and G. E. Walster, “Global optimization using interval analysis,” *Marcel Dekker*, 2004.
- [47] S. Oishi and K. Tanabe, “Numerical inclusion of optimum point for linear programming,” *JSIAM Letter*, 2009.
- [48] D. N. Nenchev, Y. Tsumaki, M. Uchiyama, “Singularity-consistent parameterization of robot motion and control,” *The Int. J. of Robotics Research*, vol. 19, no. 2, pp. 159–182, Feb. 2000.
- [49] H. G. Kwatny and G. Blankenship, *Nonlinear Control and Analytical Mechanics: A Computational Approach*, Birkhauser, 2000.
- [50] (2012, Jan.). Wolfram Research:Mathematica ,技術・科学ソフトウェア [Online]. Available:<http://www.wolfram.com/index.ja.html>
- [51] 古賀雅伸：“制御・数値解析のためのMaTX” ,東京電機大学出版局 ,2000.
- [52] 古賀雅伸：“Linux・WindowsでできるMaTXによる数値計算” ,東京電機大学出版局 ,2000.
- [53] (2012, Jan.). The MaTX Home Page. [Online]. Available:<http://www.matx.org/>
- [54] D. N. Nenchev and K. Yoshida, “Impact analysis and post-impact motion control issues of a free-floating space robot subject to a force impulse,” *IEEE Trans. Robot. Autom.*, vol. 15, no. 3, pp. 548–57, June 1999.
- [55] D. N. Nenchev, K. Yoshida, P. Vichitkulsawat and M. Uchiyama, “Reaction null-space control of flexible structure mounted manipulator systems,” *IEEE Trans. on Robot. Autom.*, vol. 15, no. 6, pp. 1011–1023, Dec. 1999.

- [56] D. N. Nenchev and A. Nishio, “Ankle and hip strategies for balance recovery of a biped subjected to an impact,” *Robotica*, vol. 26, pp. 643–653, June 2008.
- [57] A. Shumway-Cook and F. B. Horak, “Vestibular rehabilitation: an exercise approach to managing symptoms of vestibular dysfunction,” *Seminars in Hearing*, vol. 10, no. 2, pp. 196–209, 1989.
- [58] Y. Yoshida, K. Takeuchi, D. Sato and D. Nenchev, “Postural balance strategies for humanoid robots in response to disturbances in the frontal plane,” in *IEEE Int. Conf. on Robotics and Biomimetics*, 2011, to be published.
- [59] (2011, Dec.). Optitrack - Optical Motion Capture Systems and Tracking Software. [Online]. Available: <http://www.naturalpoint.com/optitrack/>
- [60] (2011, Dec.). Wii Balance Board at Nintendo. [Online]. Available: <http://www.nintendo.com/wii/console/accessories/balanceboard>
- [61] Y. Hwang, A. Konno and M. Uchiyama, “Whole body cooperative tasks and static stability evaluations for a humanoid robot,” in *Proc. of the 2003 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Las Vegas, Nevada, 2003, pp. 1901–1906.

## 研究成果

- [1] 原田謙 , 高橋準 , 金宮好和 , 佐藤大祐：“扁平足を有した歩行モデルのリミットサイクル規範三次元歩行制御法の構築” , 第 22 回日本機械学会ロボティクス・メカトロニクス講演会 , 2010.
- [2] Y. Harada, J. Takahashi, D. N. Nenchev and D. Sato, “Limit cycle based walk of a powered 7DOF 3D biped with flat feet,” in *Proc. of the 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Syst.*, Taipei, Taiwan, 2010, pp. 3623–3628.
- [3] 佐藤冬樹 , 西井達哉 , 高橋準 , 吉田祐貴 , 三橋勝 , P. Kormushev , 金宮好和：“HOAP-2によるホワイトボードに書かれた文字を消す作業の実現” , 第 11 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 , 2010.
- [4] F. Sato, T. Nishii, J. Takahashi, Y. Yoshida, M. Mitsuhashi and D. N. Nenchev, “Experimental evaluation of a trajectory/force tracking controller for a humanoid robot cleaning a vertical surface,” in *Proc. of the 2011 IEEE/RSJ Int. Conf. on Intelligent Robots and Syst.*, San Francisco, 2011, pp. 3179–3184.



# 付 錄A 空間ベクトルの基礎式

角速度と並進の速度やモーメントと並進の力など，回転成分と並進成分を一つに組み合わせた六次元のベクトルを空間ベクトルと呼ぶ。本論文で述べた順動力学計算法は，すべて空間ベクトルを用いて表現されている。空間ベクトルを用いることで，式展開を速く行え，動力学のアルゴリズムを簡潔に表記することができる。

本章では，モーションベクトルとフォースベクトルの定義を行い，速度や力の関係式から変換行列の導出を行う。また，運動量と慣性テンソルの関係を明かにし，慣性テンソルを変換するための関係式を導く。さらに，空間ベクトルを用いた外積の計算方法について述べる。

## A.1 モーションベクトルとフォースベクトル

空間ベクトルではベクトルをモーションベクトルとフォースベクトルに別けて考える。角速度を  $\omega \in \mathbb{R}^3$ ，並進の速度を  $v \in \mathbb{R}^3$ ，角加速度を  $\alpha \in \mathbb{R}^3$ ，並進の加速度を  $a \in \mathbb{R}^3$ ，モーメントを  $n \in \mathbb{R}^3$ ，並進の力を  $f \in \mathbb{R}^3$  角運動量を  $l \in \mathbb{R}^3$ ，運動量を  $p \in \mathbb{R}^3$  としたとき，モーションベクトルとフォースベクトルは Table A.1 のよう分類できる。

並進速度  $v$  と角速度  $\omega$  を組み合わせた行列  $\nu \in \mathbb{R}^6$  を速度と呼ぶ。マニピュレータの先端の速度  $\nu_e$  と一般化速度  $\dot{q}$  の関係は，ヤコビ行列  $J$  を用い

Table A.1: Spatial vector: motion vector and force vector.

	motion vector			force vector	
	angular	linear		angular	linear
velocity	$\omega$ [rad/s]	$v$ [m/s]	force	$n$ [Nm]	$f$ [N]
acceleration	$\alpha$ [rad/s <sup>2</sup> ]	$a$ [m/s <sup>2</sup> ]	momentum	$l$ [Nms]	$p$ [Ns]

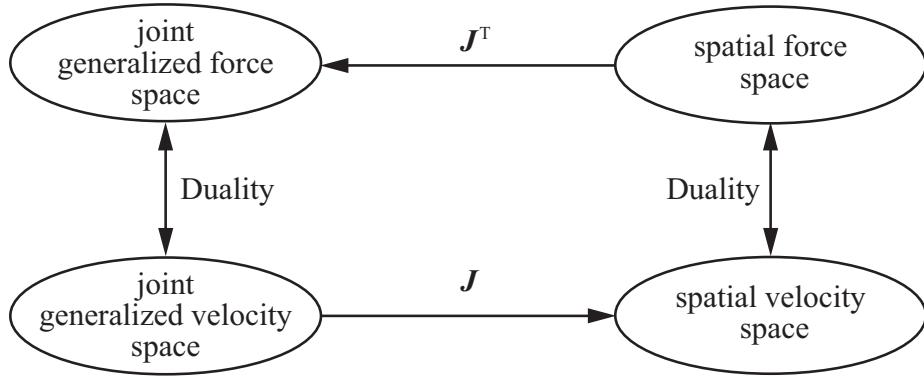


Fig. A.1: Vector spaces with their duals and the Jacobian map.

て次のように表せる。

$$\mathcal{V}_e = \begin{bmatrix} \omega \\ v_e \end{bmatrix} = J \dot{q} \quad (\text{A.1})$$

また、並進の力  $f$  とモーメント  $n$  を組み合わせた行列  $\mathcal{F} \in \mathbb{R}^6$  を力と呼ぶ。マニピュレータの先端に加わる力  $\mathcal{F}_e$  と一般化力  $\tau$  の関係はヤコビ行列の転置を用いて次のように書ける。

$$\tau = J^T \begin{bmatrix} n_e \\ f \end{bmatrix} = J^T \mathcal{F}_e \quad (\text{A.2})$$

一般的に、速度と力は二つの異なるベクトル空間に存在する。しかし、力のベクトル空間は速度のベクトル空間の双対空間である。このため  $(v \cdot f)$  や  $(\mathcal{V} \cdot \mathcal{F})$  のようなベクトル内積ができ、その結果は仕事となる。ベクトル空間の関係を Fig. A.1 に示す。

## A.2 速度

Fig. A.2 に示すような剛体  $B$  と、原点  $O$  をもつ座標系  $\{O\}$  について考える。回転軸が  $O$  を通る  $B$  の角速度を  $\omega = [\omega_x, \omega_y, \omega_z]^T$ ,  $\omega$  の歪対象行列を  $\Omega^\times$ ,  $O$  の並進速度を  $v_O$  とすると、剛体上の点  $P$  の並進速度  $v_P$  は次のように表せる。

$$v_P = v_O + \Omega^\times \overrightarrow{OP} \quad (\text{A.3})$$

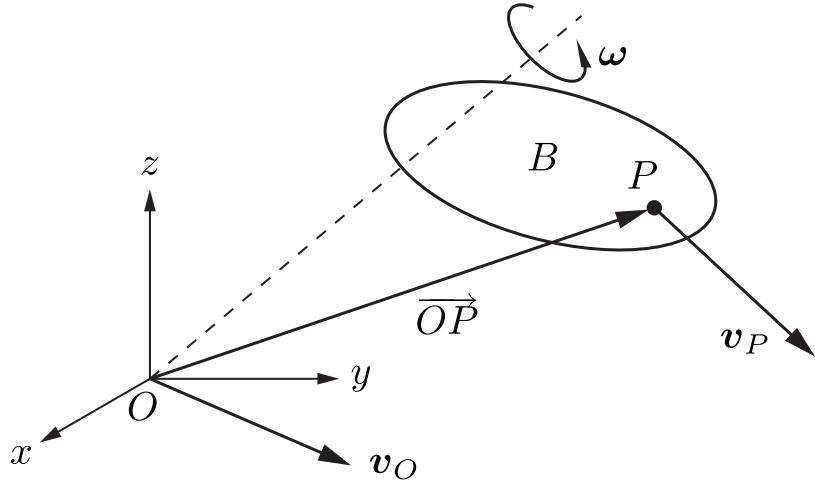


Fig. A.2: The velocity of a rigid body expressed in terms of  $\boldsymbol{\omega}$  and  $\mathbf{v}_O$ .

ここで  $\overrightarrow{OP} \in \mathbb{R}^3$  は  $O$  から  $P$  までの位置ベクトルである。また、 $\Omega^\times$  は次のようになる。

$$\Omega^\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (\text{A.4})$$

よって、 $P$  の速度  $\mathbf{v}_P$  は次のように表せる。

$$\mathbf{v}_P = \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{v}_P \end{bmatrix} = \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{v}_O + \Omega^\times \overrightarrow{OP} \end{bmatrix} \quad (\text{A.5})$$

### A.3 力

速度と同様に Fig. A.3 に示すような剛体  $B$  と、座標系  $\{P\}$  の力  $\mathcal{F}_P$  について考える。剛体には、力  $\mathbf{f} = [f_x, f_y, f_z]^T$  と  $\{O\}$  を通る回転軸をもつモーメント  $n_O$  が作用している。この剛体上の任意の点  $P$  のモーメント  $n_P$  は次式のように表せる。

$$\mathbf{n}_P = \mathbf{n}_O + \mathbf{F}^\times \overrightarrow{OP} \quad (\text{A.6})$$

ここで、 $\mathbf{F}^\times$  は  $\mathbf{f}$  の歪対象行列として次のように表す。

$$\mathbf{F}^\times = \begin{bmatrix} 0 & -f_z & f_y \\ f_z & 0 & -f_x \\ -f_y & f_x & 0 \end{bmatrix} \quad (\text{A.7})$$

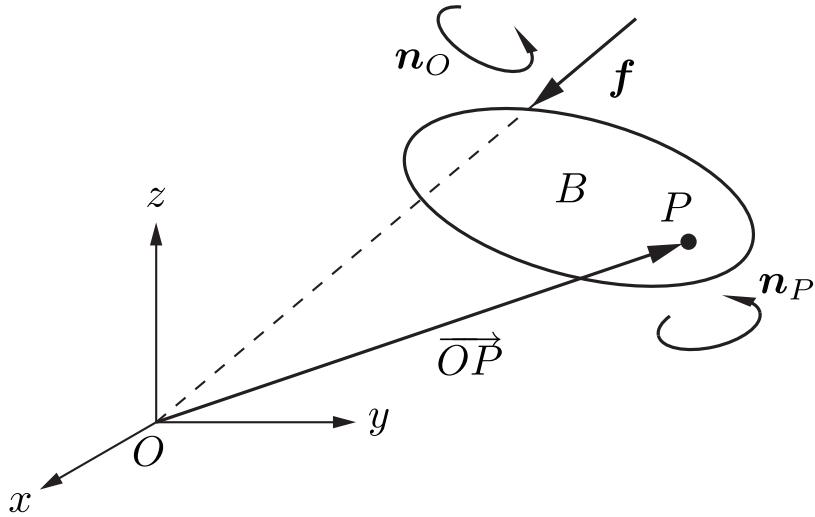


Fig. A.3: The force acting on a rigid body expressed in terms of  $\mathbf{n}_O$  and  $\mathbf{f}$ .

よって  $\mathcal{F}_P$  は次のように表せる .

$$\mathcal{F}_P = \begin{bmatrix} \mathbf{n}_P \\ \mathbf{f} \end{bmatrix} = \begin{bmatrix} \mathbf{n}_O + \mathbf{F}^\times \overrightarrow{OP} \\ \mathbf{f} \end{bmatrix} \quad (\text{A.8})$$

## A.4 変換行列

モーションベクトルとフォースベクトルでは変換行列が異なる . 座標系  $\{A\}$  から  $\{B\}$  へ変換を行うときの変換行列の記号は , モーションベクトルの場合  ${}^B_A X$  を , フォースベクトルの場合  ${}^B_A X^*$  を用いる . 以下に  $\{A\}$  から  $\{B\}$  への変換行列の導出を示す .

### A.4.1 モーションベクトル(速度)

回転

共通の原点  $O$  をもつ二つの座標系  $\{A\}$  ,  $\{B\}$  を Fig. A.4 (a) に示す .  $\{A\}$  ,  $\{B\}$  は  $O$ まわりの回転のみ行う座標系である .  $\{A\}$  からみた速度を  ${}^A v$  とする .  $\{A\}$  から  $\{B\}$  へ変換する回転行列を  ${}^B_A R \in \Re^{3 \times 3}$  とすると ,  $\{B\}$  からみた速度  ${}^B v$  は次のようになる .

$${}^B v = \begin{bmatrix} {}^B \omega \\ {}^B v_O \end{bmatrix} = \begin{bmatrix} {}^B R & \mathbf{0} \\ \mathbf{0} & {}^A R \end{bmatrix} \begin{bmatrix} {}^A \omega \\ {}^A v_O \end{bmatrix} = \begin{bmatrix} {}^B R & \mathbf{0} \\ \mathbf{0} & {}^A R \end{bmatrix} {}^A v \quad (\text{A.9})$$

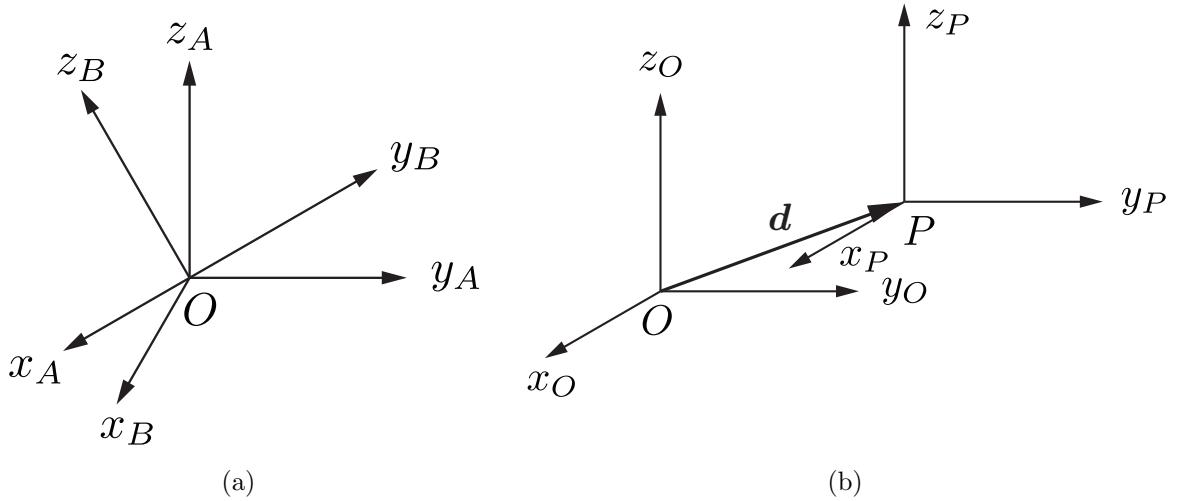


Fig. A.4: Coordinate frame: (a)  $\{A\}$  and  $\{B\}$  are two coordinate frames with a common origin at  $O$  and (b)  $O$  and  $P$  are points in space.

### 並進

別々の点  $O, P$  を原点とする座標系  $\{O\}, \{P\}$  を Fig. A.4 (b) に示す。ここでは  $\{O\}, \{P\}$  は平行であり、座標系の回転は考えない。 $\overrightarrow{OP} = \mathbf{d} = [d_x, d_y, d_z]^T$  とする。また、 $\{O\}$  の速度を  $v_O$ ,  $\{P\}$  の速度を  $v_P$  とする。Eq. (A.3) から  $v_O$  と  $v_P$  の関係は次のように表せる。

$$\begin{aligned} v_P &= \begin{bmatrix} \omega \\ v_p \end{bmatrix} = \begin{bmatrix} \omega \\ v_O + \boldsymbol{\Omega}^\times \mathbf{d} \end{bmatrix} = \begin{bmatrix} \omega \\ v_O - \boldsymbol{D}^\times \omega \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ -\boldsymbol{D}^\times & \mathbf{U} \end{bmatrix} v_O \end{aligned} \quad (\text{A.10})$$

ここで、 $\boldsymbol{D}^\times$  は  $d$  の歪対象行列として次のように表す。

$$\boldsymbol{D}^\times = \begin{bmatrix} 0 & -d_z & d_y \\ d_z & 0 & -d_x \\ -d_y & d_x & 0 \end{bmatrix} \quad (\text{A.11})$$

また、 $\mathbf{U} \in \Re^{3 \times 3}$  は単位行列である。

### A.4.2 フォースベクトル(力)

#### 回転

モーションベクトルのときと同様に Fig. A.4 (a) に示すような共通の原点  $O$  をもつ二つの座標系  $\{A\}, \{B\}$  について考える。 $\{A\}, \{B\}$  は  $O$  まわりの回転のみ行う座標系である。 $\{A\}$  からみた力を  ${}^A\mathcal{F}$  とする。 $\{A\}$  から  $\{B\}$  へ変換する回転行列を  ${}_A^B\mathbf{R} \in \mathbb{R}^{3 \times 3}$  とすると、 $\{B\}$  からみた力  ${}^B\mathcal{F}$  は次のようになる。

$${}^B\mathcal{F} = \begin{bmatrix} {}^B\mathbf{n}_O \\ {}^B\mathbf{f} \end{bmatrix} = \begin{bmatrix} {}^B\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}_A^B\mathbf{R} \end{bmatrix} \begin{bmatrix} {}^A\mathbf{n}_O \\ {}^A\mathbf{f} \end{bmatrix} = \begin{bmatrix} {}^B\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}_A^B\mathbf{R} \end{bmatrix} {}^A\mathcal{F} \quad (\text{A.12})$$

#### 並進

モーションベクトルのときと同様に Fig. A.4 (b) に示すような別々の点  $O, P$  を原点とする座標系  $\{O\}, \{P\}$  について考える。ここでは  $\{O\}, \{P\}$  は平行であり、座標系の回転は考えない。 $\{O\}$  の力を  $\mathcal{F}_O, \{P\}$  の力を  $\mathcal{F}_P$  とする。Eq. (A.6) から  $\mathcal{F}_O$  と  $\mathcal{F}_P$  の関係は次のように表せる。

$$\begin{aligned} \mathcal{F}_P &= \begin{bmatrix} \mathbf{n}_P \\ \mathbf{f} \end{bmatrix} = \begin{bmatrix} \mathbf{n}_O + \mathbf{F}^\times \mathbf{d} \\ \mathbf{f} \end{bmatrix} = \begin{bmatrix} \mathbf{n}_O - \mathbf{D}^\times \mathbf{f} \\ \mathbf{f} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{U} & -\mathbf{D}^\times \\ \mathbf{0} & \mathbf{U} \end{bmatrix} \mathcal{F}_O \end{aligned} \quad (\text{A.13})$$

### A.4.3 一般的な変換

Fig. A.5 に示すような、それぞれ原点  $O, P$  をもつ座標系  $\{A\}, \{B\}$  について考える。ここで、 ${}^A\mathbf{D}_{OP}^\times$  は  $\{A\}$  から  $\{B\}$  への位置ベクトルの歪対象行列、 ${}_A^B\mathbf{R} \in \mathbb{R}^{3 \times 3}$  は  $\{A\}$  から  $\{B\}$  へ変換する回転行列を示す。 $\{A\}$  から  $\{B\}$  への変換は、 ${}_A^B\mathbf{R}$  による回転と  $\mathbf{D}^\times$  による並進の積で計算できる。モーションベクトルの変換は

$${}^B\mathbf{X} = \begin{bmatrix} {}^B\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}_A^B\mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ -{}^A\mathbf{D}_{OP}^\times & \mathbf{U} \end{bmatrix} = \begin{bmatrix} {}^B\mathbf{R} & \mathbf{0} \\ -{}_A^B\mathbf{R} {}^A\mathbf{D}_{OP}^\times & {}_A^B\mathbf{R} \end{bmatrix} \quad (\text{A.14})$$

となり、フォースベクトルの変換は

$${}^B\mathbf{X}^* = \begin{bmatrix} {}^B\mathbf{R} & \mathbf{0} \\ \mathbf{0} & {}_A^B\mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{U} & -{}^A\mathbf{D}_{OP}^\times \\ \mathbf{0} & \mathbf{U} \end{bmatrix} = \begin{bmatrix} {}^B\mathbf{R} & -{}_A^B\mathbf{R} {}^A\mathbf{D}_{OP}^\times \\ \mathbf{0} & {}_A^B\mathbf{R} \end{bmatrix} \quad (\text{A.15})$$

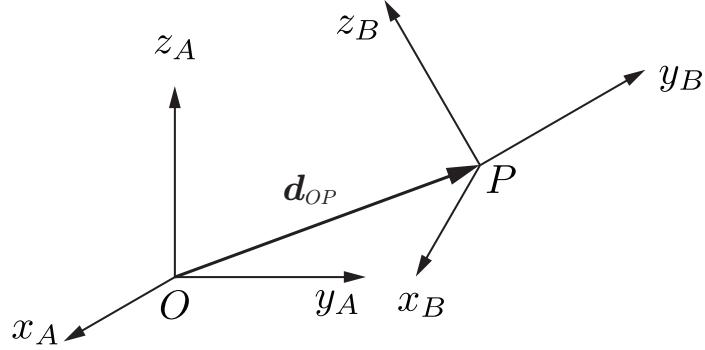


Fig. A.5: \$\{A\}\$ and \$\{B\}\$ are two coordinate frames with origin at \$O\$ and \$P\$.

となる。また \$\{B\}\$ から \$\{A\}\$ へ変換する回転行列を \$\overset{A}{B}\mathbf{R} \in \mathbb{R}^{3 \times 3}\$ とすると、\$\{B\}\$ から \$\{A\}\$ への変換は、モーションベクトルの場合、

$$\overset{A}{B}\mathbf{X} = \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \overset{A}{B}\mathbf{D}_{OP}^\times & \mathbf{U} \end{bmatrix} \begin{bmatrix} \overset{A}{B}\mathbf{R} & \mathbf{0} \\ \mathbf{0} & \overset{A}{B}\mathbf{R} \end{bmatrix} = \begin{bmatrix} \overset{A}{B}\mathbf{R} & \mathbf{0} \\ \overset{A}{B}\mathbf{D}_{OP}^\times \overset{A}{B}\mathbf{R} & \overset{A}{B}\mathbf{R} \end{bmatrix} \quad (\text{A.16})$$

となり、フォースベクトルの場合、

$$\overset{A}{B}\mathbf{X}^* = \begin{bmatrix} \mathbf{U} & \overset{A}{B}\mathbf{D}_{OP}^\times \\ \mathbf{0} & \mathbf{U} \end{bmatrix} \begin{bmatrix} \overset{A}{B}\mathbf{R} & \mathbf{0} \\ \mathbf{0} & \overset{A}{B}\mathbf{R} \end{bmatrix} = \begin{bmatrix} \overset{A}{B}\mathbf{R} & \overset{A}{B}\mathbf{D}_{OP}^\times \overset{A}{B}\mathbf{R} \\ \mathbf{0} & \overset{A}{B}\mathbf{R} \end{bmatrix} \quad (\text{A.17})$$

となる。さらに \$\overset{B}{A}\mathbf{X}\$, \$\overset{B}{A}\mathbf{X}^\*\$, \$\overset{A}{B}\mathbf{X}\$, \$\overset{A}{B}\mathbf{X}^\*\$ の間には次のような関係がある。

$$\overset{B}{A}\mathbf{X} = (\overset{A}{B}\mathbf{X}^*)^T = (\overset{B}{A}\mathbf{X}^*)^{-T} \quad (\text{A.18})$$

$$\overset{B}{A}\mathbf{X}^* = (\overset{A}{B}\mathbf{X})^T = (\overset{B}{A}\mathbf{X})^{-T} \quad (\text{A.19})$$

## A.5 運動量

力と同様に Fig. A.6 に示すような剛体 \$B\$ と、座標系 \$\{O\}\$ の運動量 \$\mathbf{h}\_O\$ について考える。剛体の重心の点を \$C\$, \$C\$ の重心まわりの慣性テンソルを \$\bar{\mathbf{I}}\_C\$, \$C\$ の角速度を \$\boldsymbol{\omega}\$, \$C\$ の並進速度を \$\mathbf{v}\_C\$ とすると、\$C\$ の角運動量 \$\mathbf{l}\_C\$ と並進の運動量 \$\mathbf{p}\$ はそれぞれ

$$\mathbf{l}_C = \bar{\mathbf{I}}_C \boldsymbol{\omega} \quad (\text{A.20})$$

$$\mathbf{p} = m \mathbf{v}_C \quad (\text{A.21})$$

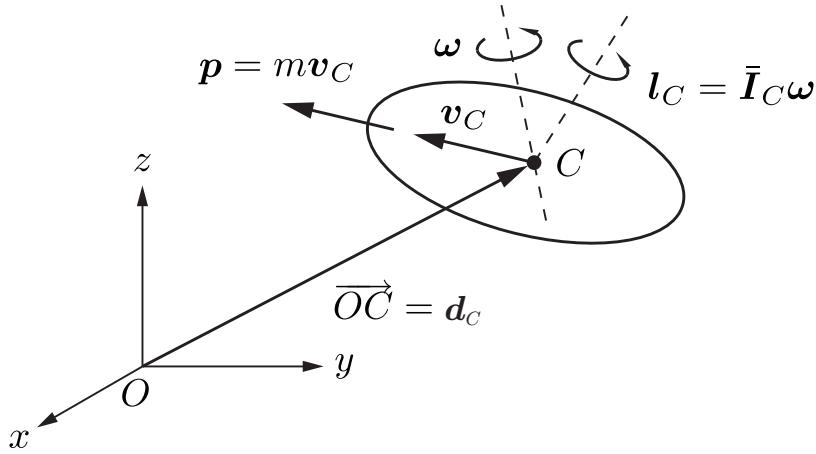


Fig. A.6: Spatial momentum.

となる。また、\$\{O\}\$ の角運動量 \$\mathbf{l}\_O\$ は、\$\{O\}\$ から \$C\$ までのベクトルの歪対象行列を \$\mathbf{D}\_C^\times\$ とすると Eq. (A.6) と同様に次のように表せる。

$$\mathbf{l}_O = \mathbf{l}_C + \mathbf{D}_C^\times \mathbf{p} \quad (\text{A.22})$$

Eq. (A.20)–(A.22) より \$C, O\$ の運動量 \$\mathcal{H}\_C, \mathcal{H}\_O \in \Re^6\$ は次のように表せる。

$$\begin{aligned} \mathcal{H}_C &= \begin{bmatrix} \mathbf{l}_C \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{I}}_C \boldsymbol{\omega} \\ m \mathbf{v}_C \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{I}}_C & \mathbf{0} \\ \mathbf{0} & m \mathbf{U} \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{v}_C \end{bmatrix} \\ &= \begin{bmatrix} \bar{\mathbf{I}}_C & \mathbf{0} \\ \mathbf{0} & m \mathbf{U} \end{bmatrix} \mathcal{V}_C \end{aligned} \quad (\text{A.23})$$

$$\begin{aligned} \mathcal{H}_O &= \begin{bmatrix} \mathbf{l}_O \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{l}_C + \mathbf{D}_C^\times \mathbf{p} \\ m \mathbf{v}_C \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{I}}_C \boldsymbol{\omega} + \mathbf{D}_C^\times m \mathbf{v}_C \\ m \mathbf{v}_C \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{U} & \mathbf{D}_C^\times \\ \mathbf{0} & \mathbf{U} \end{bmatrix} \mathcal{H}_C \end{aligned} \quad (\text{A.24})$$

## A.6 慣性テンソル

剛体の慣性テンソル \$\mathbf{I} \in \Re^{6 \times 6}\$ は速度 \$\mathcal{V}\$ と運動量 \$\mathcal{H}\$ から、次のように表すことができる。

$$\mathcal{H} = \mathbf{I} \mathcal{V} \quad (\text{A.25})$$

ここで \$\mathbf{I}\$ は速度 \$\mathcal{V}\$ を運動量 \$\mathcal{H}\$ に変換する行列として考えることができる。さらに \$\mathcal{V}\$ はモーションベクトル、\$\mathcal{H}\$ はフォースベクトルである。よって \$\mathbf{I}\$ は

モーションベクトルをフォースベクトルに変換する行列として考えることができる。

任意の点 $O$ と剛体の重心 $C$ を原点とする座標系 $\{O\}$ ,  $\{C\}$ について考える。剛体の重心の運動量 $\mathcal{H}_C$ は Eq. (A.23) より,

$$\mathcal{H}_C = \begin{bmatrix} \bar{\mathbf{I}}_C & \mathbf{0} \\ \mathbf{0} & m \mathbf{U} \end{bmatrix} \nu_C \quad (\text{A.26})$$

となる。よって重心まわりの慣性行列 $\bar{\mathbf{I}}_C$ は次のように表せる。

$$\bar{\mathbf{I}}_C = \begin{bmatrix} \bar{\mathbf{I}}_C & \mathbf{0} \\ \mathbf{0} & m \mathbf{U} \end{bmatrix} \quad (\text{A.27})$$

次に $\{O\}$ まわりの慣性行列 $\bar{\mathbf{I}}_O$ について考える。Eq. (A.23), (A.24) より

$$\begin{aligned} \mathcal{H}_O &= \begin{bmatrix} \mathbf{U} & \mathbf{D}_C^\times \\ \mathbf{0} & \mathbf{U} \end{bmatrix} \mathcal{H}_C \\ &= \begin{bmatrix} \mathbf{U} & \mathbf{D}_C^\times \\ \mathbf{0} & \mathbf{U} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{I}}_C & \mathbf{0} \\ \mathbf{0} & m \mathbf{U} \end{bmatrix} \nu_C \\ &= \begin{bmatrix} \mathbf{U} & \mathbf{D}_C^\times \\ \mathbf{0} & \mathbf{U} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{I}}_C & \mathbf{0} \\ \mathbf{0} & m \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ (\mathbf{D}_C^\times)^T & \mathbf{U} \end{bmatrix} \nu_O \\ &= \begin{bmatrix} \bar{\mathbf{I}}_C + m \mathbf{D}_C^\times (\mathbf{D}_C^\times)^T & m \mathbf{D}_C^\times \\ m (\mathbf{D}_C^\times)^T & m \mathbf{U} \end{bmatrix} \nu_O \end{aligned} \quad (\text{A.28})$$

となり $\bar{\mathbf{I}}_O$ は次のようになる。

$$\bar{\mathbf{I}}_O = \begin{bmatrix} \bar{\mathbf{I}}_C + m \mathbf{D}_C^\times (\mathbf{D}_C^\times)^T & m \mathbf{D}_C^\times \\ m (\mathbf{D}_C^\times)^T & m \mathbf{U} \end{bmatrix} \quad (\text{A.29})$$

## A.7 慣性行列の逆行列

Eq. (A.25) を $\nu$ を求める式に変形すると次のようになる。

$$\nu = \mathbf{I}^{-1} \mathcal{H} = \Phi \mathcal{H} \quad (\text{A.30})$$

ここで $\mathcal{H}$ はフォースベクトル,  $\nu$ はモーションベクトルである。よって $\Phi$ はフォースベクトルをモーションベクトルに変換する行列として考えることができる。

任意の点 $O$ と剛体の重心 $C$ を原点とする座標系 $\{O\}, \{C\}$ について考える。剛体の重心の速度 $v_C$ は Eq. (A.30) より、

$$v_C = \begin{bmatrix} \bar{I}_C^{-1} & \mathbf{0} \\ \mathbf{0} & \frac{1}{m} \mathbf{U} \end{bmatrix} \mathcal{H}_C \quad (\text{A.31})$$

となる。よって重心まわりの慣性行列の逆行列 $\Phi_C$ は次のように表せる。

$$\Phi_C = \begin{bmatrix} \bar{I}_C^{-1} & \mathbf{0} \\ \mathbf{0} & \frac{1}{m} \mathbf{U} \end{bmatrix} \quad (\text{A.32})$$

次に $\{O\}$ まわりの慣性行列の逆行列 $\Phi_O$ について考える。

$$\begin{aligned} v_O &= \begin{bmatrix} \boldsymbol{\omega} \\ v_O \end{bmatrix} = \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{D}_C^\times & \mathbf{U} \end{bmatrix} v_C \\ &= \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{D}_C^\times & \mathbf{U} \end{bmatrix} \begin{bmatrix} \bar{I}_C^{-1} & \mathbf{0} \\ \mathbf{0} & \frac{1}{m} \mathbf{U} \end{bmatrix} \mathcal{H}_C \\ &= \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{D}_C^\times & \mathbf{U} \end{bmatrix} \begin{bmatrix} \bar{I}_C^{-1} & \mathbf{0} \\ \mathbf{0} & \frac{1}{m} \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{U} & (\mathbf{D}_C^\times)^T \\ \mathbf{0} & \mathbf{U} \end{bmatrix} \mathcal{H}_O \\ &= \begin{bmatrix} \bar{I}_C^{-1} & \bar{I}_C^{-1} (\mathbf{D}_C^\times)^T \\ \mathbf{D}_C^\times \bar{I}_C^{-1} & \frac{1}{m} \mathbf{U} + \mathbf{D}_C^\times \bar{I}_C^{-1} (\mathbf{D}_C^\times)^T \end{bmatrix} \mathcal{H}_O \end{aligned} \quad (\text{A.33})$$

よって $\Phi_O$ は次のようになる。

$$\Phi_O = \begin{bmatrix} \bar{I}_C^{-1} & \bar{I}_C^{-1} (\mathbf{D}_C^\times)^T \\ \mathbf{D}_C^\times \bar{I}_C^{-1} & \frac{1}{m} \mathbf{U} + \mathbf{D}_C^\times \bar{I}_C^{-1} (\mathbf{D}_C^\times)^T \end{bmatrix} \quad (\text{A.34})$$

## A.8 慣性行列の座標変換

### A.8.1 慣性行列

先に述べたように、慣性行列 $I$ はモーションベクトルをフォースベクトルに変換する行列である。このような行列は、二つのフォースベクトル $f \in \mathbb{R}^6, g \in \mathbb{R}^6$ を用いると $fg^T$ のようにして表現することができる。右からモーションベクトル $m \in \mathbb{R}^6$ をかけると $f(g^T m)$ となる。モーションベクトルとフォースベクトルの内積は仕事となるので、括弧の中はスカラーとなり、全体はフォースベクトルとなる。よって $fg^T$ はモーションベクトルを

フォースベクトルに変換する行列であるといえる。さらに  $I$  は対称行列であることを考慮し次のように定義する。

$$I = g \ g^T \quad (\text{A.35})$$

座標系  $\{A\}$ ,  $\{B\}$  から見た慣性行列  ${}^A I$ ,  ${}^B I$  と  $g$  の座標変換を

$$\begin{aligned} {}^A I &= {}^A g ({}^A g)^T \\ {}^B I &= {}^B g ({}^B g)^T \\ {}^B g &= {}_A^B X^* {}^A g \end{aligned} \quad (\text{A.36})$$

とすると、 ${}^A I$  から  ${}^B I$  への座標変換は次のようになる。

$$\begin{aligned} {}^B I &= {}^B g ({}^B g)^T = {}_A^B X^* {}^A g ({}_A^B X^* {}^A g)^T = {}_A^B X^* {}^A g ({}^A g)^T ({}_A^B X^*)^T \\ &= {}_A^B X^* {}^A g ({}^A g)^T {}_B^A X \\ &= {}_A^B X^* {}^A I {}_B^A X \end{aligned} \quad (\text{A.37})$$

## A.8.2 慣性行列の逆行列

先に述べたように、慣性行列の逆行列  $\Phi$  はフォースベクトルをモーションベクトルに変換する行列である。 $I$  のときと同様に考え  $\Phi$  を次のように定義する。

$$\Phi = m \ m^T \quad (\text{A.38})$$

座標系  $\{A\}$ ,  $\{B\}$  から見た慣性行列の逆行列  ${}^A \Phi$ ,  ${}^B \Phi$  と  $m$  の座標変換を

$$\begin{aligned} {}^A \Phi &= {}^A m ({}^A m)^T \\ {}^B \Phi &= {}^B m ({}^B m)^T \\ {}^B m &= {}_A^B X {}^A m \end{aligned} \quad (\text{A.39})$$

とすると、 ${}^A \Phi$  から  ${}^B \Phi$  への座標変換は次のようになる。

$$\begin{aligned} {}^B \Phi &= {}^B m ({}^B m)^T = {}_A^B X {}^A m ({}_A^B X {}^A m)^T = {}_A^B X {}^A m ({}^A m)^T ({}_A^B X)^T \\ &= {}_A^B X {}^A m ({}^A m)^T {}_B^A X^* \\ &= {}_A^B X {}^A \Phi {}_B^A X^* \end{aligned} \quad (\text{A.40})$$

## A.9 外積

三次元の位置ベクトル  $d$  の時間微分  $\dot{d}$  を考える。角速度を  $\omega = [\omega_x, \omega_y, \omega_z]^T$  の歪対象行列を  $\Omega^\times$  とすると、 $\dot{d}$  は次のように求まる。

$$\dot{d} = \Omega^\times d \quad (\text{A.41})$$

これと同様にモーションベクトル  $m$  とフォースベクトル  $f$  の時間微分は速度  $\mathcal{V}$  を用いて次のように表せる。

$$\begin{aligned} \dot{m} &= \mathcal{V} \times m \\ \dot{f} &= \mathcal{V} \times^* f \end{aligned} \quad (\text{A.42})$$

ここで、演算記号  $\times$  は空間ベクトルの外積を示す。また、モーションベクトルとの外積には  $\times$ 、フォースベクトルとの外積には  $\times^*$  を用いて区別する。

ある点  $O$  の速度を  $\mathcal{V}_O = [\omega_{Ox}, \omega_{Oy}, \omega_{Oz}, v_{Ox}, v_{Oy}, v_{Oz}]^T$ 、とすると  $\mathcal{V} \times$  と  $\mathcal{V} \times^*$  はそれぞれ次のように表せる。

$$\mathcal{V}_O \times = \begin{bmatrix} 0 & -\omega_z & \omega_y & 0 & 0 & 0 \\ \omega_z & 0 & -\omega_x & 0 & 0 & 0 \\ -\omega_y & \omega_x & 0 & 0 & 0 & 0 \\ 0 & -v_{Oz} & v_{Oy} & 0 & -\omega_z & \omega_y \\ v_{Oz} & 0 & -v_{Ox} & \omega_z & 0 & -\omega_x \\ -v_{Oy} & v_{Ox} & 0 & -\omega_y & \omega_x & 0 \end{bmatrix} \quad (\text{A.43})$$

$$\mathcal{V}_O \times^* = \begin{bmatrix} 0 & -\omega_z & \omega_y & 0 & -v_{Oz} & v_{Oy} \\ \omega_z & 0 & -\omega_x & v_{Oz} & 0 & -v_{Ox} \\ -\omega_y & \omega_x & 0 & -v_{Oy} & v_{Ox} & 0 \\ 0 & 0 & 0 & 0 & -\omega_z & \omega_y \\ 0 & 0 & 0 & \omega_z & 0 & -\omega_x \\ 0 & 0 & 0 & -\omega_y & \omega_x & 0 \end{bmatrix} = -(\mathcal{V}_O \times)^T \quad (\text{A.44})$$

## A.10 本章のまとめ

本章では、空間ベクトルの基礎式をまとめた。モーションベクトルやフォースベクトルの定義、速度や力の定義、変換行列の導出、運動量と慣性テンソルの関係、慣性テンソルの変換、外積の定義について述べた。この基礎式は、本論文で述べた順動力学計算法を理解する上で欠かすことのできない式である。