

DYNAMIC STABILITY ANALYSIS OF MODULAR, SELF-RECONFIGURABLE
ROBOTIC SYSTEMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

TEVFİK ALİ BÖKE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MECHANICAL ENGINEERING

APRIL 2005

Approval of the Graduate School of Natural and Applied Sciences

Prof.Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of
Master of Science.

Prof.Dr. Kemal İDER
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully
adequate, in scope and quality, as a thesis for the degree of Master of Science.

Prof.Dr. Reşit SOYLU
Supervisor

Examining Committee Members

Prof.Dr. M.Kemal ÖZGÖREN (METU, ME) _____

Prof.Dr. Reşit SOYLU (METU, ME) _____

Asst.Prof. Dr. İlhan KONUKSEVEN (METU, ME) _____

Asst.Prof. Dr. Buğra KOKU (METU, ME) _____

Asst.Prof. Dr. Erol ŞAHİN (METU, CENG) _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Tevfik Ali BÖKE

Signature :

ABSTRACT

DYNAMIC STABILITY ANALYSIS OF MODULAR, SELF-RECONFIGURABLE ROBOTIC SYSTEMS

Böke, Tevfik Ali

M.S., Department of Mechanical Engineering

Supervisor: Prof.Dr. Reşit SOYLU

April 2005, 117 pages.

In this study, an efficient algorithm has been developed for the dynamic stability analysis of self-reconfigurable, modular robots. Such an algorithm is essential for the motion planning of self-reconfigurable robotic systems. The building block of the algorithm is the determination of the stability of a rigid body in contact with the ground when there exists Coulomb friction between the two bodies. This problem is linearized by approximating the friction cone with a pyramid and then solved, efficiently, using linear programming. The effects of changing the number of faces of the pyramid and the number of contact points are investigated. A novel definition of stability, called percentage stability, is introduced to counteract the adverse effects of the static indeterminacy problem between two contacting bodies.

The algorithm developed for the dynamic stability analysis, is illustrated via various case studies using the recently introduced self-reconfigurable robotic system, called I-Cubes.

Keywords: Modular Robots, Self-reconfigurable Robots, I-Cubes, Stability Analysis, Static Indeterminacy, Coulomb Friction.

ÖZ

MODÜLER, KENDİLİĞİNDEN ŞEKİL DEĞİŞTİREBİLEN ROBOTİK SİSTEMLERİN DİNAMİK DENGE ANALİZİ

Böke, Tevfik Ali

Yüksek Lisans, Makine Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Reşit SOYLU

Nisan 2005, 117 sayfa.

Bu çalışmada, kendiliğinden şekil değiştirebilen, modüler robotların dinamik denge analizi için kullanılan bir algoritma geliştirilmiştir. Bu tür bir algoritma, kendiliğinden şekil değiştirebilen robotik sistemlerin hareket planlaması için gereklidir. Algoritmanın temel taşı, aralarında Coulomb sürtünmesi bulunan rijit bir cisim ile zemin den oluşan sistemde, rijit cismin dengede olup olmadığının saptanmasıdır. Bu problem, sürtünme konisinin bir piramit olarak varsayılması yöntemiyle doğrusallaştırılmış ve doğrusal programlama kullanılarak çözülmüştür. Piramidin yüz sayısı ve zeminle temas ettiği nokta sayıları değiştirilmek suretiyle oluşabilecek etkiler incelenmiştir. Temas halindeki iki cisim arasındaki statik belirsizliğin yarattığı problemleri kısmen azaltabilmek amacıyla “yüzdellik denge” isimli yeni bir denge tanımı sunulmuştur.

Dinamik denge analizi için geliştirilen bu algoritma, I-Küpler adındaki, kendiliğinden şekil değiştirebilen bir robot sistemi üzerinde çeşitli örneklerle irdelenmiştir.

Anahtar Kelimeler: Modüler Robotlar, Kendiliğinden şekil değiştirebilen Robotlar, I-Küpler, Denge Analizi, Statik Belirsizlik, Coulomb Sürtünmesi.

ACKNOWLEDGMENTS

The author wishes to express his deepest gratitude to his supervisor Prof. Dr. Reşit SOYLU for his guidance, criticism, encouragements and insight throughout the research.

The author would like to thank Cem ÜNSAL, project scientist at Carnegie Mellon University, and Uğur KIRMIZIGÜL for their technical support and interest.

The author would also like to thank his parents, Asaf and Dilek BÖKE and his friends, especially to Sibel SİVRİKAYA, for their support and encouragements.

TABLE OF CONTENTS

ABSTRACT	iv
ÖZ	v
ACKNOWLEDGMENTS	vi
TABLE OF CONTENTS.....	vii
LIST OF TABLES	x
LIST OF FIGURES	xi
LIST OF SYMBOLS.....	xiv
CHAPTER 1	
INTRODUCTION	1
1.1 TYPES OF MODULAR ROBOTS.....	2
1.2 APPLICATIONS.....	3
1.3 STABILITY	4
CHAPTER 2	
RELATED WORK.....	8
2.1 ACM.....	8
2.2 TETROBOT	8
2.3 CEBOT	9
2.4 FRACTA	9
2.5 MOLECULE	10
2.6 METAMORPHIC ROBOTIC SYSTEM.....	10
2.7 PROTEO.....	11
2.8 CRYSTALLINE	11
2.9 FRACTAL ROBOT	12
2.10 FRACTUM	12
2.11 MINIATURIZED SELF-RECONFIGURABLE SYSTEM	13
2.12 CONRO	13
2.13 I-CUBES	14
2.14 POLYPOD AND POLYBOT	14
2.15 SEMI-CYLINDRICAL RECONFIGURABLE ROBOT	15
2.16 SELF-REPAIRING AND FAULT TOLERANT SYSTEMS	16
2.17 PARALLEL ROBOTS.....	16

2.18 INCHWORM	17
2.19 MILLIBOTS	17
2.20 FIELD ROBOTS.....	18
2.21 INDUSTRIAL TYPE MODULAR MANIPULATORS	18
2.22 VARIOUS TYPES OF RECONFIGURABLE ROBOTS.....	20
CHAPTER 3	
I-CUBES.....	21
3.1. GEOMETRIC DESIGN AND LINK ACTUATION	23
3.2. DESIGN OF THE LINKS AND THE CUBES	25
3.3. TWIST-AND-LOCK ATTACHMENT MECHANISM.....	27
3.4. 3-D RECONFIGURATION	28
CHAPTER 4	
STATIC STABILITY ANALYSIS.....	33
4.1. STATICAL EQUIVALENCE EQUATIONS.....	38
4.2. LINEARIZATION OF COULOMB'S LAW OF FRICTION.....	45
4.3. LINEAR PROGRAMMING	51
4.3. PERCENTAGE STABILITY.....	56
CHAPTER 5	
DYNAMIC STABILITY ANALYSIS OF I-CUBES	60
5.1. ACTIVE AND PASSIVE BODY SYSTEMS	60
5.1.0. The Notation	64
5.1.1. D-H Convention.....	66
5.1.2. Inertial Parameters of the Active and Passive Body Systems.....	68
5.2. RECURSIVE KINEMATIC RELATIONS FOR THE ACTIVE AND PASSIVE BODY SYSTEMS.....	75
5.2.1. Transformation Matrices.....	75
5.2.2. Angular Velocities	75
5.2.3. Angular Accelerations	76
5.2.4. Location of the Body System Origins.....	77
5.2.5. Locations of the Mass Centers	77
5.3. RECURSIVE NEWTON-EULER FORMULATION FOR THE INVERSE DYNAMIC ANALYSIS OF THE ACTIVE AND PASSIVE BODY SYSTEMS	80
5.3.1. Force Equations (Newton's Equations)	81
5.3.2. Moment Equations (Euler's Equations)	81
5.4. PROBABLE MOTION OF THE ACTIVE BODY SYSTEM	83

5.5. DISCRETIZATION OF THE MOTION	84
5.6. CONTACT POINT SELECTION.....	85
CHAPTER 6	
CASE STUDIES	86
6.3. CASE 1	88
6.5. CASE 2: 1C1L.....	92
6.5. CASE 3: 2C1L.....	98
6.5. CASE 4: 3C2L.....	101
6.5. CASE 5: 4C4L.....	105
6.6. CASE 6: TWO FACE CONTACT	107
CHAPTER 7	
CONCLUSIONS	111
REFERENCES	113

LIST OF TABLES

Table 2.1 Various Types of Reconfigurable Robots.....	20
Table 6.1 Stable and unstable points in VN	91
Table 6.2 Discretized resultant resistive loads.....	97

LIST OF FIGURES

Figure 1.1 A block on an incline [4].	5
Figure 1.2 A block resting on a grounded table with an external load [4].	6
Figure 2.1 (a) Active Cord Mechanism [7], (b) Tetrobot [8].	9
Figure 2.2 Fracta [10].	10
Figure 2.3 (a) Molecule [12], (b) Metamorphic Robotic System [13].	11
Figure 2.4 (a) Proteo [15], (b) Crystalline [16].	12
Figure 2.5 (a) Fractal Robot [17], (b) Fractum [11].	13
Figure 2.6 (a) Miniaturized Self-reconfigurable system [18], (b) CONRO [19].	14
Figure 2.7 I-Cubes [21].	14
Figure 2.8 (a) Polypod [22], (b) PolyBot [23].	15
Figure 2.9 Semi-Cylindrical Reconfigurable Robot [24].	16
Figure 2.10 Modular Parallel Robot [27].	17
Figure 2.11 (a) Inchworm [29], (b) Millibots [30].	17
Figure 2.12 Field Robots [31].	18
Figure 2.13 (a) RMMS [32], (b) Modular Robotic Workcell [36], and (c) A PUMA type robot configuration constructed by MoRSE modules.	19
Figure 3.1 Geometric requirements for the system.[37]	23
Figure 3.2.a Link moving from one cube face to another. [37]	24
Figure 3.2.b Link lifting a cube while attached to another. [37]	24
Figure 3.2.c Link moving from one cube to another. [37]	24
Figure 3.3 Physical view of a 3C2L I-Cube system.[41]	25
Figure 3.4 (a) The cube and (b) the link.[37]	26
Figure 3.5 CAD images: (a) The link is shown with two different connectors (servo holders expanded); (b) the cube is shown with five faceplates.[38]	26
Figure 3.6 ProE drawings of the link: (a) idle joint fully open and (b) at zero degrees.[37]	27
Figure 3.7 Cross-shaped connector with twist-and-lock mechanism [38].	28
Figure 3.8 Attachment mechanisms: (a) 4-pegs with latch at the center of the faceplate, (b) twist-and-lock mechanism with sliding latch.[37]	28
Figure 3.9 Link approaching the faceplate.[37]	29

Figure 3.10 A group of three links and three cubes in linear motion (image sequence left-to-right).[38]	30
Figure 3.11 A 4L4C group reconfiguring to move over an obstacle.[38]	31
Figure 3.12 A group of four links and four cubes capable of locomotion forming a tower.[38].....	32
Figure 4.1 A block resting on a ground with friction	34
Figure 4.2 Top view of the contact region between the cube and the ground.	35
Figure 4.3 Contact region between the cube and the ground with the first contact point be at the origin (O).	41
Figure 4.4 Nonlinear friction cone and the approximating rectangular pyramid.[4]..	46
Figure 4.5 Linear rectangular (a), hexagonal (b), octagonal (c), decagonal (d) and dodecagonal (e) pyramid approximations for nonlinear Coulomb friction.	47
Figure 4.6 Efficiencies of various linearizations of the friction law.	51
Figure 4.7 Distribution of contact points.	53
Figure 4.8 Comparison of the linearization methods in terms of CPU time for linear programming.	53
Figure 4.9 Comparison of the linearization methods in terms of Total CPU time. ...	54
Figure 4.10 Comparison of the linearization methods in terms of Session time.....	54
Figure 4.11 Comparison of the linearization methods in terms of μ_{\min}	55
Figure 4.12 Potentially stable and unstable portions of the set of valid normal forces.....	57
Figure 5.1 Active and passive body systems.....	61
Figure 5.2 Sub-links of a link.	62
Figure 5.3 The Denavit – Hartenberg Representation [43].	66
Figure 5.4 Inertial parameters of Active and Passive Body Systems.	68
Figure 5.5 Active cubes and sub-links.....	71
Figure 5.6 Locations of the mass centers.....	78
Figure 5.7 FBD of the k'th body system [43].....	80
Figure 5.8 Fifth order time - motion polynomial.	83
Figure 5.9 Discretization of the time - motion polynomial.	84
Figure 5.10 Given a 90° link motion with initial and final configurations (a), convex hull and the contact points (b) [37].	85
Figure 6.1 Flow chart of the stability analysis.	86
Figure 6.2 Case 1: Static Indeterminacy with Search Method.	88
Figure 6.3 Stable and unstable regions of VN	90

Figure 6.4 Case 3: 1C1L configuration with Search Method.....	92
Figure 6.5 The initial (a) and final (b) configurations of 1C1L.	94
Figure 6.6 ADAMS solutions of the initial (a), intermediate (b) and final (c) configurations of 1C1L.....	97
Figure 6.7 Case 4: 2C1L configuration with Search Method.....	98
Figure 6.8 The initial (a) and final (b) configurations of 2C1L.	99
Figure 6.9 ADAMS solutions of the initial (a) and unstable (b) configurations of 2C1L.....	100
Figure 6.10 Case 5: 3C2L configuration with Search Method.....	101
Figure 6.11 The initial (a) and final (b) configurations of 3C2L.	103
Figure 6.12 ADAMS solutions of the initial (a), intermediate (b) and final (c) configurations of 3C2L.....	104
Figure 6.13 Given a 90° link motion with initial and final configurations (a), convex hull and the contact points (b) [37].	105
Figure 6.14 Case 7: Two Face Contact with Potential Stability Method.....	107

LIST OF SYMBOLS

τ	: An external torque.
d	: The length of a cube edge.
J_1, J_2, J_3	: Three rotational degrees of freedom for the links.
\vec{F}_{ex}	: The resultant of all external forces acting on the rigid body (acting at O_1).
\vec{M}_{ex}	: The resultant of all external moments acting on the rigid body (acting at O_1).
\vec{F}_R	: The resultant force (acting at O) to be applied by the ground on the rigid body such that the body is in static equilibrium.
\vec{M}_O	: The resultant moment (acting at O) to be applied by the ground on the rigid body such that the body is in static equilibrium.
C_m	: The center of mass of the rigid body.
O	: Point of application of \vec{F}_R .
O_1	: Point of application of \vec{F}_{ex} .
m_b	: Mass of the rigid body.
\vec{g}	: Gravitational acceleration.
$Oxyz$: Body fixed coordinate system such that the z axis is directed from the ground towards the body.
F_{Rx}, F_{Ry}, F_{Rz}	: x , y and z components of \vec{F}_R in the $Oxyz$ system.
M_{Ox}, M_{Oy}, M_{Oz}	: x , y and z components of \vec{M}_O in the $Oxyz$ system.
P_i	: i 'th contact point. ($i = 1, 2, \dots, n_c$)
n_c	: Number of contact points.

\vec{F}_i	: Contact force (applied by the ground on the body) at the i 'th contact point.
F_{xi}, F_{yi}, F_{zi}	: x , y and z components of \vec{F}_i in the $Oxyz$ system.
$\vec{r}_i = (r_{xi}, r_{yi}, 0)$: Position vector of P_i with respect to the $Oxyz$ system.
μ	: The coefficient of static friction.
n_c	: The number of contact points.
\vec{M}_{Oi}	: The moment of the contact force \vec{F}_i about the origin (O).
$[A_{n_c}]$: The $(6 \times 3n_c)$ coefficient matrix.
\vec{x}_{n_c}	: The $(3n_c \times 1)$ vector of unknowns.
\vec{b}	: The known (6×1) right hand side vector.
$\vec{x}_{x1, x2, y1}$: Vector of unknown forces.
$\vec{f}_{x3, y2, y3}$: Vector of free forces.
\vec{f}	: Vector of independent force components.
c_i	: Constant coefficients. ($i = m, 1, 2, \dots, 16$).
η_ℓ	: Linearization efficiency.
n_f	: The number of faces of the approximating pyramids.
n_i	: The number of the linear inequalities which is given by $4n_c, 6n_c, 8n_c, 10n_c$ and $12n_c$ respectively if the friction cone is approximated by a pyramid with 4, 6, 8, 10 and 12 faces.
\vec{f}_z	: The vector of independent F_{zi} 's.
VN	: A "polygonal" region in the n_z dimensional \vec{f}_z -space.
S	: Potentially stable region.
U	: Unstable region.
h_s	: The distance for separating the gridlines of the VN from each other.

PS	: The index, percentage stability.
A_S	: "Area" of region S .
A_{VN}	: "Area" of region VN .
\vec{f}'_z	: A different set of independent F_{zi} 's related to \vec{f}_z .
$[B]$: $n_z \times n_z$ non singular coefficient matrix.
\vec{k}	: $n_z \times 1$ vector of constants.
G_1 or G_p	: Combined center of mass of the Passive Body System.
G_2 or G_a	: Combined center of mass of the Active Body System.
G_{pi}	: Center of mass of the i 'th passive body.
G_{ai}	: Center of mass of the i 'th active body.
n_a, n_p	: Number of active and passive bodies, respectively.
$F_0(O_0)$: Inertial frame fixed to the ground with origin O_0 and with unit vectors $\vec{u}_1^{(0)}, \vec{u}_2^{(0)}, \vec{u}_3^{(0)}$.
$\vec{u}_1^{(0)}, \vec{u}_2^{(0)}, \vec{u}_3^{(0)}$: The unit vectors of the inertial frame.
O_0	: The origin of the inertial frame.
$F_1(O_1)$: Body fixed reference frame (fixed to PBS) with origin O_1 and with unit vectors $\vec{u}_1^{(1)}, \vec{u}_2^{(1)}, \vec{u}_3^{(1)}$.
O_1	: The origin of the reference frame of the PBS.
$\vec{u}_1^{(1)}, \vec{u}_2^{(1)}, \vec{u}_3^{(1)}$: The orthonormal cartesian coordinate system of the Passive Body System (PBS).
$F_2(O_2)$: Body fixed reference frame (fixed to the body system ABS).
O_2	: The origin of the reference frame of the active body system.
$\vec{u}_1^{(2)}, \vec{u}_2^{(2)}, \vec{u}_3^{(2)}$: The orthonormal cartesian coordinate system of the Active Body System (ABS).
$\theta_2(t)$: Angular parameter which indicates the relative rotation of the ABS with respect to the PBS.

\vec{r}	: A vector.
\bar{r}	: A (3x1) column matrix.
\hat{R}	: A matrix.
r	: A scalar.
\vec{r}_{AK}	: Position vector of point A with respect to point K.
$F_a(A)$: Reference frame with unit vectors $\vec{u}_1^{(a)}, \vec{u}_2^{(a)}, \vec{u}_3^{(a)}$ and origin A .
$\left\{ \vec{r} \right\}^{(a)} = \bar{r}^{(a)}$: Column matrix representation of \vec{r} in $F_a(A)$.
$r_i^{(a)}$: i'th component of \vec{r} in $F_a(A)$.
$\hat{C}^{(a,b)}$: The (3x3) transformation matrix relating $F_b(B)$ and $F_a(A)$.
$\vec{u}_k^{(b)}$: The k'th unit vector of $F_b(B)$.
$\left\{ \vec{u}_k \right\}^{(a)} = \bar{u}_k^{(b/a)}$: Column matrix representation of $\vec{u}_k^{(b)}$ in $F_a(A)$.
$\bar{u}_1, \bar{u}_2, \bar{u}_3$: The elementary or basic columns.
$\sim r$: The cross product matrix of \bar{r} .
a_k	: The effective length of link _k .
s_k	: The translational distance of link _k with respect to link _{k-1} .
θ_k	: The angular position of body system _k with respect to body system _{k-1} .
α_k	:The offset angle.
\sphericalangle	: Angle between.
\rightarrow	: To.
@	: At.
$\vec{G}_1 = O_0\vec{G}_1$: Position vector of the center of mass of the PBS.
$\vec{G}_2 = O_2\vec{G}_2$: Position vector of the center of mass of the ABS.

- $\vec{G}_{pi} = O_0 \vec{G}_{pi}$: Position vector of the center of mass of the i 'th passive body.
 $(i = 1, 2, \dots, n_p)$
- $\vec{G}_{ai} = O_2 \vec{G}_{ai}$: Position vector of the center of mass of the i 'th active body.
 $(i = 1, 2, \dots, n_a)$
- $\vec{P}_{01} = O_0 O_1$: Location of the origin of the passive body system.
- $\overline{P}_{01}^{(0)}$: Coordinates of the origin of the PBS in $F_0(O_0)$.
- $\overline{G}_1^{(0)}$: Coordinates of the center of mass of the PBS in $F_0(O_0)$.
- $\overline{G}_{pi}^{(0)}$: Coordinates of the center of mass of the i 'th passive body in $F_0(O_0)$.
- m_{pi} : Mass of the i 'th passive body.
- m_p : Total mass of the passive body system.
- $\overline{G}_2^{(2)}$: Coordinates of the center of mass of the ABS in $F_2(O_2)$.
- $\overline{G}_{ai}^{(2)}$: Coordinates of the center of mass of the i 'th active body in $F_2(O_2)$.
- m_{ai} : Mass of the i 'th active body.
- m_a : Total mass of the active body system.
- d_{l1}, d_{l2}, d_{l3} : The sub-link parameters.
- m_c : Mass of each cube.
- $m_l = m_{l1} = m_{l2}$: Mass of each sub-link.
- $\hat{J}_{ci}^{(ai)}$: The Centroidal Inertia Dyadic of the i 'th active cube expressed in the body fixed frame $F_{ai}(G_{ai})$.
- $\hat{J}_{li}^{(ai)}$: The Centroidal Inertia Dyadic of the i 'th active sub-link expressed in the body fixed frame $F_{ai}(G_{ai})$.
- J_{l1}, J_{l2}, J_{l3} : Mass moments of inertia about 1st, 2nd and 3rd axis.

$\hat{I}_2^{(2)}$: The centroidal inertia dyadic of the ABS expressed in $F_2(G_2)$

$\hat{J}_i^{(2)}$: Centroidal Inertia Dyadic of the i 'th active body with respect to the active body system's body fixed frame $F_2(G_2)$, ($i = 1, 2, \dots, n_a$).

$\hat{\Gamma}^{(ai)} = \hat{C}^{(2,ai)}$: Transformation matrix of the i 'th active body with respect to the active body system's body fixed frame ($F_2(G_2)$).

$\hat{J}_i^{(ai)}$: Centroidal Inertia Dyadic of the i 'th active body (cube or sub-link) with respect to the active body's body fixed frame $F_{ai}(G_{ai})$.

$\overline{G}_{aai}^{(2)}$: The position vector of $G_a \overrightarrow{G_{ai}}$ in $F_2(G_2)$.

$\hat{\Phi}_k = \hat{C}^{(0,k)}$: Transformation matrix of the k 'th body system with respect to the base frame $F_0(O_0)$.

$\overrightarrow{\omega}_k = \overrightarrow{\omega}_{k/0}$: Angular velocity of the k 'th body system with respect to the base frame $F_0(O_0)$.

$\overrightarrow{\alpha}_k = \overrightarrow{\alpha}_{k/0} = D_0 \overrightarrow{\omega}_k$: Angular acceleration of the k 'th body system with respect to the base frame $F_0(O_0)$.

$\overrightarrow{P}_{Ok} = \overrightarrow{O_0 O_k}$: Location of the body system origins.

$\overrightarrow{P}_k = \overrightarrow{O_0 G_k}$: Location of the mass centers.

$\overrightarrow{V}_{OK} = D_0 \overrightarrow{P}_{OK}$: Velocity vector of the origin of the k 'th body system with respect to the inertial frame $F_0(O_0)$.

$\overrightarrow{V}_K = D_0 \overrightarrow{P}_K$: Velocity vector of the mass center of the k 'th body system with respect to the inertial frame $F_0(O_0)$.

$\vec{a}_{OK} = D_0^2 \vec{P}_{OK}$: Acceleration vector of the origin of the k'th body system with respect to the inertial frame $F_0(O_0)$.
$\vec{a}_K = D_0^2 \vec{P}_K$: Acceleration vector of the mass center of the k'th body system with respect to the inertial frame $F_0(O_0)$.
\vec{F}_{ij}	: Force applied by Body System _i on Body System _j .
\vec{M}_{ij}	: Moment applied by Body System _i on Body System _j .
$\vec{a}_k = a_{G_k}$: Acceleration of the mass center.
$\theta_2(t)$: The motion of the active body system.
$\vec{\rho}_k = G_k \vec{O}_k$: Moment arm of $\vec{F}_{(k+1)k}$.
$\vec{\rho}_k^* = G_k \vec{O}_{k-1}$: Moment arm of $\vec{F}_{(k-1)k}$.
\check{I}_k	: Centroidal Inertia Dyadic
$\vec{\alpha}_k$: Angular acceleration.
$\vec{\omega}_k$: Angular velocity.
$\hat{I}_1^{(1)}$: A dummy inertia tensor.
h	: Step Size.
t_f	: Final Time.
N_d	: Number of divisions.
q_0 and q_f	: The initial and final positions, respectively.

CHAPTER 1

INTRODUCTION

Robots with a fixed architecture constitute an overwhelming majority of robot designs. These robots are usually intended to perform a single task and the architecture is designed accordingly. Although the single task may be extremely complicated, such as space exploring, a robot designed for that task will have difficulty performing a different task. This is true for all single-purpose robots. They perform very well in the structured environment for which they were designed. However, they usually perform poorly in non-structured environments and in environments for which they were not designed. To be successful in unknown or unstructured environments, robots need to be able to change their architecture to suit the environment and the task. In this thesis, modular, self-reconfigurable robots, which are a class of robots that can change shape and functionality, and their stability problem have been studied.

Modular robots are in general the robots, which consist of a set of independent robotic modules that can be composed in different geometric configurations to create the optimal geometric shape for a given task. These modules cooperate to perform the tasks of the robot. The modules may be complete robots in themselves capable of performing some tasks without cooperation, or they may be units which are functional only when some minimum number of modules is present. Modular robots are versatile and extensible robots and have several advantages over the more traditional, fixed architecture robots. Some of these advantages are stated below.

- The module can be simple in design. Because it is only one part of a greater whole, each module needs only a small part of the overall capability of the entire system. This makes each module easier to design and build.
- Modular robots support multiple modalities of locomotion and manipulation. This can be achieved by requiring the robot to metamorphose from

one shape to another to best match the shape of the terrain, such as stair climbing, gap crossing, tower creating, etc.

- Modular robots are fault tolerant, i.e., if a module fails, some additional modules will replace it with the spare units.
- Modular robots are used in tasks that require self-assembly. Such as assembling a structure in space.

1.1 TYPES OF MODULAR ROBOTS

There are three types of reconfigurable robots, namely, chain, lattice and mobile kind.

The chain kind modular robots make themselves over by attaching and detaching chains of modules to and from themselves, with each chain always attached to the rest of the modules at one or more points. Nothing ever moves off on its own. The chains may be used as arms for manipulating objects, legs for locomotion, or short tentacles for both manipulation and locomotion.

The lattice kind modular robots change shape by moving into positions on a virtual grid, or lattice. They are like pawns moving on a chessboard, except this board has three dimensions.

The mobile kind reconfigurable robots change shape by having modules detach themselves from the main body and move independently. They then link up at new locations to form new configurations.

Modular robots may also be classified according to configuration competence, structure, working environment and control.

- **According to Configuration Competence:**
 - Self-reconfigurable modular robots, which can autonomously change their configurations, with no help from outside (mostly used in unknown or complex environments).

- Manually-reconfigurable modular robots, which can only change their shape by outside help.

- **According to Structure:**

- Homogenous, modular robots which have identical parts (modules), each of which contains all of the actuation, sensing, CPU and battery requirements. Indeed, such robots will be physically too large.

- Heterogeneous, modular robots which have different types of modules have different types of aims. One module may be responsible for actuating, while another may be power supply.

- **According to Working Environment:**

- Planar (2-D), modular robots which work in 2-D space.
- 3-D, modular robots which work in 3-D space.

- **According to Control:**

- Distributed, modular robots where each module thinks for itself within the group context.
- Centralized, modular robots where the whole system is considered.

1.2 APPLICATIONS

Self-reconfigurable robots have many applications in both the macro-robotics and the micro-robotics fields.

On the macro-robotics side, applications include designing versatile robots that can self-reconfigure in the best shape to fit the terrain, environment, and task. When a task or terrain varies, modular reconfigurable robots can change their shapes to get the job done. A modular, self-reconfigurable robot is most useful in an unknown, complex environment. For example, a damaged building by an

earthquake contains a variety of obstructions and may not be suitable for any standard robot. A reconfigurable robot, with the ability to locomate over a variety of terrains, through gaps and over obstacles can perform well in this situation. Another application is space/planetary exploration where unpredictable terrains on a planet have to be explored.

The self-reconfiguration algorithms for enabling several locomotion gates and manipulation modes have applications at the micro scale, in non-invasive medicine and in complex part assembly with MEMS devices. The geometric algorithms resulting from self-organizing robots could be mapped at the micro scale to create new gates for MEMS devices that can result in the self-propelling of parts (“walking chips”) and the self-assembly of complex parts. Microscopic self-reconfigurable robots could be used in numerous applications. One application is a NASA space probe that dumps nano-scale robots on an asteroid. These robots then mine materials to reproduce billions of identical robots which are used to transform the asteroid into a space station. Another application is minimally-invasive surgery. Future operations may take place by inserting a small tube into the patient through which many micro-scale self-reconfigurable robots could be inserted into the patient's body as a linear chain structure. This structure would then reconfigure into a manipulator, allowing the surgeon to perform procedures that would require a large incision using today's techniques. The benefit would be less tissue damage and therefore faster healing.

1.3 STABILITY

While planning the motion of a self-reconfigurable robotic system, one has to consider the dynamic stability of the system. To be more explicit, the bodies which are in contact with the ground should remain motionless during the motion from an initial configuration to a final configuration. The main problem here is static indeterminacy; that is, situations in which the stability of an assembly is indeterminate, because the distribution of normal forces is indeterminate. In discussing stability, one can either consider potential stability or guaranteed stability [1]. *An assembly is potentially stable if contact forces (i.e., normal and friction forces) could arise that cause the assembly to remain motionless.* On the other

hand, *an assembly has guaranteed stability if all normal and friction forces that can arise cause the assembly to remain motionless*. Determining the stability of contacting frictionless assemblies is relatively straightforward. Although the actual contact forces that arise at any given instant may be indeterminate, the overall acceleration of all the bodies is unique [2], [3]. Thus, there is no difference between potential and guaranteed stability, for frictionless systems; if there is any combination of legal contact forces which yields zero acceleration for each body, then all legal contact forces will yield zero acceleration for each body. The addition of friction, greatly complicate matters. Consider Figure 1.1, which shows an object in contact with an inclined plane at a number of points. If the object was frictionless, the distribution of weight among the contact points would be undetermined, but the acceleration would be unique. The object would slide down the plane. Suppose, however, that contact points 1 and 4 have friction, but there is no friction at contacts 2 and 3. Now the behavior is truly indeterminate. If all the weight rests on the interior two contacts, the object will slide down the plane. But if all the weight rests on the external two contact points, the object will remain motionless (assuming a large enough coefficient of friction). In fact, there are infinitely many behaviors. If the object's weight is distributed over both the exterior and interior contacts, the acceleration of the object down the plane will be inversely proportional to the weight resting on the exterior contacts. [4]

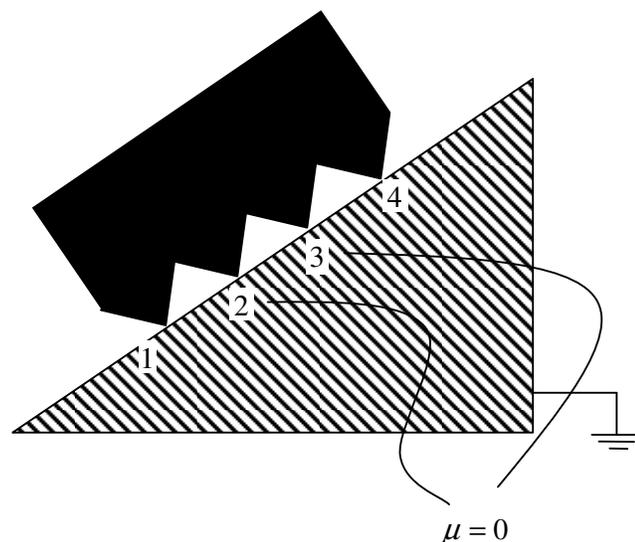


Figure 1.1 A block on an incline [4].

Consider, now, the problem, where a block is resting on a table with the gravity vector and an external torque (τ) acting on the block as shown in Figure 1.2. The contact surface between the block and the table is shown alongside. Let us consider unknown contact forces as occurring at the five points shown in Figure 1.2 (b). Normal forces arise at the contact points so as to balance the gravitational force. If these forces are distributed over the five contact points, then a nonzero torque can be generated over the surface. However, if the normal forces act at the center point, as shown in Figure 1.2 (c), then no torque can be generated. This is another example for the problem of static indeterminacy [4].

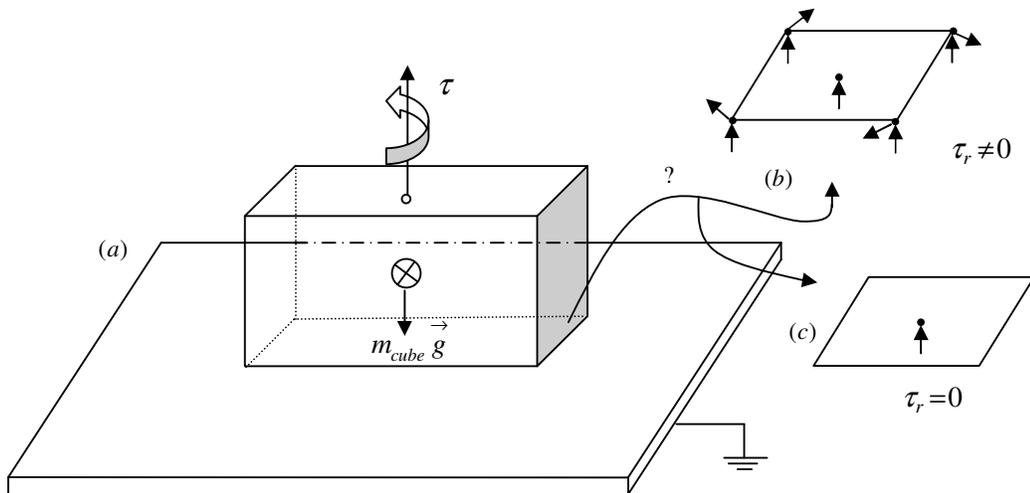


Figure 1.2 A block resting on a grounded table with an external load [4].

Figure 1.2 only gives two possible contact force distributions. Note that one of these distributions gives an unstable solution (Figure 1.2 (c)). The distribution in Figure 1.2 (b), however, may give a stable solution with a large enough coefficient of friction.

When the rigid-body assumption is relaxed to allow deformations, the indeterminacy in the force distribution is resolved. If actual deformations are allowed, indeterminacy can be either partly or completely eliminated [5].

Pang and Trinkle [6] are other researchers who have extensively dealt with the problem of static stability in detail. Actually, they have classified external loads into 3 categories, namely, weak stable, strongly stable and frictionless stable loads.

Stability of rigid bodies is a problem that is also studied regarding assemblies to be handled by robots [4] as well as part of fixture design [6] and computer simulations [5].

In this study, an algorithm has been developed for the dynamic stability analysis of self-reconfigurable, modular robots. This kind of an algorithm is necessary in order to realize motion planning for reconfigurable robots. To the author's knowledge, this is the first algorithm of its kind in the literature.

Throughout the thesis the algorithm is illustrated via I-Cubes which is a recently introduced self-reconfigurable, modular robot. The developed algorithm, however, is applicable to any reconfigurable robot with slight modifications.

The outline of the thesis is given next.

In Chapter 2, most popular modular and/or self-reconfigurable robots are discussed. Chapter 3 is specifically devoted to I-Cubes. Static stability analysis is detailed in Chapter 4. Dynamic stability analysis of I-Cubes is presented in Chapter 5. Chapter 6 is devoted to case studies and, finally, the conclusions are given in Chapter 7.

CHAPTER 2

RELATED WORK

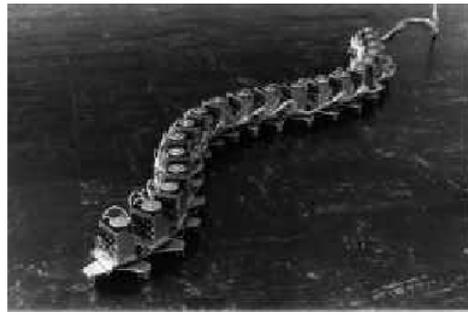
In this chapter, the earlier research efforts on modular robots are investigated and compared. The stability problem is an important challenge for all kind of modular robots and the work on this thesis could be adapted to any kind of modular robots.

2.1 ACM

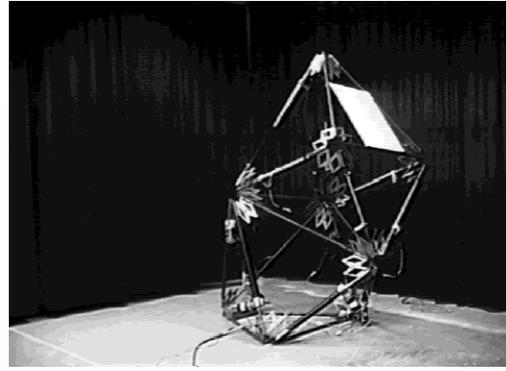
The active cord mechanism (ACM), a snake-like robotic mechanism, was an early development by Hirose [7]. The ACM is a homogeneous modular robot and it was used to mimic the snake movement. Both manipulation and locomotion have been implemented in the ACM. ACM-R1 is shown in Figure 2.1 (a). ACM may operate in 3D, but it does not have the ability to self-reconfigure.

2.2 TETROBOT

Hamlin and Sanderson [8] implemented a modular system, TETROBOT (Figure 2.1 (b)). Novel spherical joints were used to design a homogeneous truss structured robot. The joint design allows the structure to spherically move around a center of rotation. However, connecting parts are manually assembled. The authors presented possible configurations of the system such as Double Octahedral, Tetrahedral Manipulator and Six Legged Walker TETROBOT. Suggested applications were space/sea exploration and construction sites.



(a)



(b)

Figure 2.1 (a) Active Cord Mechanism [7], (b) Tetrobot [8].

2.3 CEBOT

A cellular robotic system (CEBOT) was developed by Fukuda and Kawauchi [9]. This is a homogeneous modular robot where each cell has limited sensing and computation. The problem of determining an optimal arrangement of cells for a particular task was studied. Experiments in automated reconfiguration were carried out but the robot did not self-reconfigure. A manipulator arm was required for self-reconfiguration.

2.4 FRACTA

Murata *et al* considered 3D [10] and 2D [11] categories of homogeneous distributed systems. In the 3D design, Fracta (as shown in Figure 2.2) has three symmetric axes with twelve degrees of freedom. A unit is composed of a 265 mm cube weighing 7 kg with connecting arms attached to each face. Self-reconfiguration is performed by means of rotating the arms and an automatic connection mechanism. Each unit has an on-board microprocessor and communication system. The drawback of this approach is that each module is quite big and heavy. The connection mechanism uses six sensors and encoders, further increasing system complexity. However, this is one of the few systems that can achieve 3D self-

reconfiguration. This system perfectly illustrates the fact that in a homogeneous design, the modules become big and cumbersome.

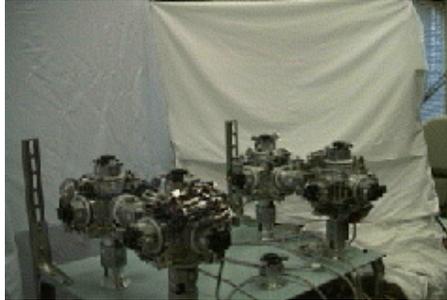


Figure 2.2 Fracta [10].

2.5 MOLECULE

A similar type of 3D homogeneous self-reconfigurable system is the Molecule [12] (See Figure 2.3 (a)). Each molecule consists of a pair of two-DOF atoms, connected by a link (called a bond). By suitably connecting a number of modules, one can form 3D shapes. Twelve movements of each atom can perform self-reconfiguration. Independent movement on a substrate of molecules including straight-line traversal and 90 degrees convex and concave transitions to adjacent surface can be performed.

2.6 METAMORPHIC ROBOTIC SYSTEM

The Metamorphic robotic system (Figure 2.3 (b)) was demonstrated by Chirikjian [13], [14]. Each module is a planar hexagonal shape with three DOFs that can combine with others with varying geometry. Each module has abilities to connect, disconnect and rotate around its neighbors. However, it is a limited, planar mechanism.

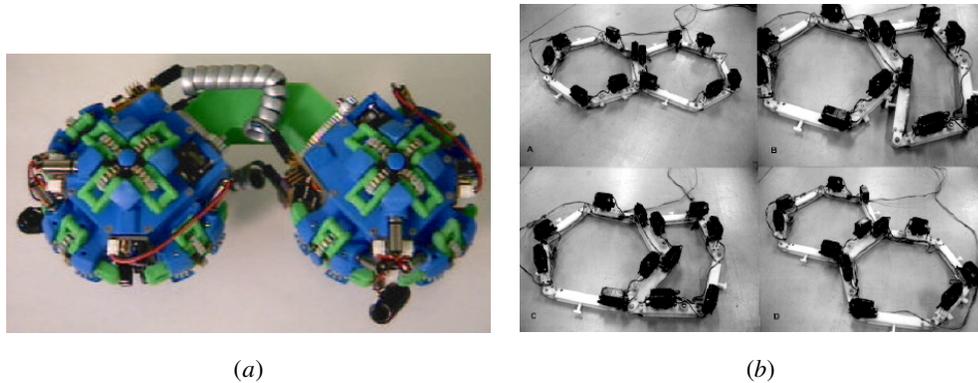


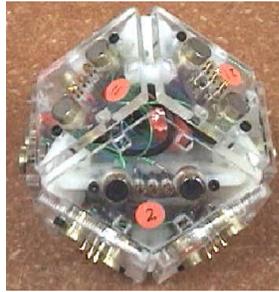
Figure 2.3 (a) Molecule [12], (b) Metamorphic Robotic System [13].

2.7 PROTEO

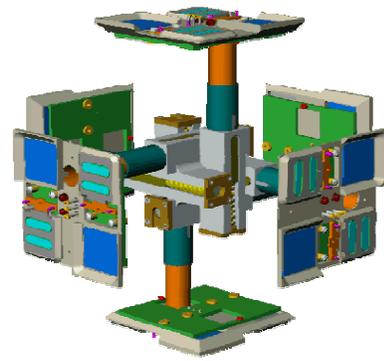
A metamorphic robot, Proteo, which is shown in Figure 2.4 (a), was proposed by Bojinov *et al* [15]. Each module is a rhombic dodecahedron with twelve identical connection faces which allow other modules to attach. Electromagnets are used for module connection. According to the simulations, the motion is simply composed of a number of rotations about the edges of the faces. This robot consists of compact homogeneous rhombus units. This is an interesting concept. However, the use of twelve connecting faces leads to high complexity and high cost.

2.8 CRYSTALLINE

The concept of a Crystalline module (see Figure 2.4 (b)) was described by Rus and Vona [16]. Each module has a square cross-section with a connection mechanism using channels and rotating keys to lock modules together. A distributed robotic system is actuated by expanding and contracting each module. Each module can expand its size by a factor of two from its original size. The module has an onboard CPU, IR communication and power supply. Note that although the Crystalline robot is planar, it could be extended to 3D. The connection mechanism has male and female parts which limits possible mating configurations.



(a)



(b)

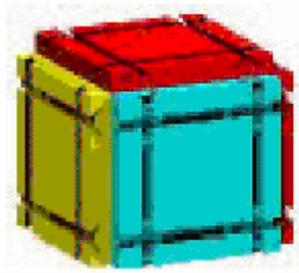
Figure 2.4 (a) Proteo [15], (b) Crystalline [16].

2.9 FRACTAL ROBOT

Figure 2.5 (a) shows a novel polymorphic robot called “Fractal robot” which was proposed by Michael [17]. The Fractal robot is composed of homogeneous cubes with screw and groove mechanisms at each cubic face to allow the robot to perform geometry changes and tasks. The structure formation is performed by sliding one or a group of cubes to another location along attached face(s). This mechanism seems difficult to implement and the results appear to be mainly in simulation. It is suggested that each module can attach special devices such as camera, gripper, etc.

2.10 FRACTUM

Fractum (see Figure 2.5 (b)) is a 2D homogeneous system developed by Tomita *et al* [11]. Each unit has six arms, three electromagnet male arms and three permanent magnet female arms. Based on simple magnetics, connection occurs when a neighbor (male) has the same polarity of a permanent magnet (female). On the other hand, reversing the polarity of the electromagnets causes disconnection. A unit has three ball wheels under a body, a processor and an optical communication. The Fractum robot has a simple mechanism. Therefore, it can only achieve planar motion.



(a)



(b)

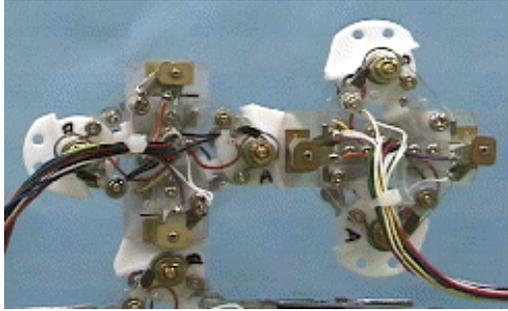
Figure 2.5 (a) Fractal Robot [17], (b) Fractum [11].

2.11 MINIATURIZED SELF-RECONFIGURABLE SYSTEM

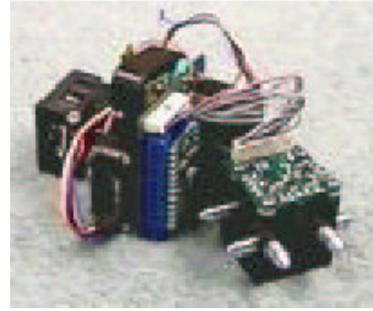
Figure 2.6 (a) shows the miniaturized self-reconfigurable robot which was presented by Yoshida *et al* [18]. A male and female connection mechanism is used, with locking pins holding the modules together. A shape memory alloy (SMA) spring is used to release the lock. The miniaturized robot is approximately 40 mm high, 50 mm long and it weighs 80 g. This is a planar design, but the researchers are considering a 3D mechanism. This system has been designed using novel SMA actuators which reduces the size of the system. However, limited torque and a short range of movement are the main disadvantages.

2.12 CONRO

CONRO [19] is a self-reconfigurable robot composed of two-DOF homogeneous modules (Figure 2.6 (b) shows a single module). Each module is 108 mm long and weighs 115 g. Docking connectors (active and passive) using a SMA locking mechanism allows modules to connect with pins and holes for alignment. Each module has two motors, two batteries, a micro-controller and an IR communication system. The design of homogeneous CONRO robot allows for self-reconfiguration. Its size is compact to reconfigure. However, the use of a bipartite active/passive connection mechanism limits reconfiguration.



(a)



(b)

Figure 2.6 (a) Miniaturized Self-reconfigurable system [18], (b) CONRO [19].

2.13 I-CUBES

Ünsal and Khosla [20] have designed a modular self-reconfigurable robotic system called I-Cubes (or ICES-Cubes). I-Cube is a bipartite system composed of a three-DOF link and a passive element as connector (see Figure 2.7). The link is 170 mm long and weighs 205 g. The passive element is a cube which has six faces for connecting. A novel mechanism provides inter-module attachment and detachment to perform various tasks such as moving over obstacles.

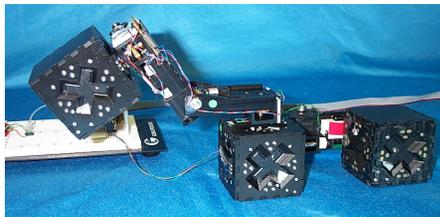
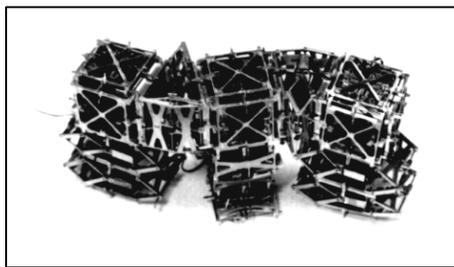


Figure 2.7 I-Cubes [21].

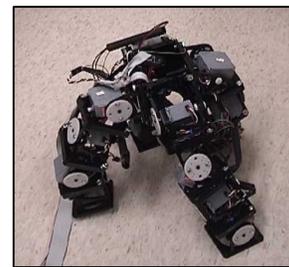
2.14 POLYPOD AND POLYBOT

Yim [22] proposed Polypod, a modular reconfigurable robot, with two types of modules: A two-DOF *segment* with two connection ports and a passive cubic *node*

with six connection ports. The modules are manually bolted in different ways in order to achieve versatility to many modes of locomotion gaits (Figure 2.8 (a)). Each module is approximately a 60 mm cube. PolyBot [23] (Figure 2.8 (b) shows PolyBot (G2)) is comprised of homogeneous one-DOF modules with hermaphroditic (genderless) connection plates. Each PolyBot module has a quite powerful on-board computer, but limited sensing. It is about 50 mm each side (with the motor protruding by about 50 mm) and weighs 416 g. Infrared is used to communicate between the modules. Polypod and PolyBot are homogeneous systems. They are simple and versatile. On the other hand, Polypod has to be manually reconfigured to form different structures.



(a)



(b)

Figure 2.8 (a) Polypod [22], (b) PolyBot [23].

2.15 SEMI-CYLINDRICAL RECONFIGURABLE ROBOT

Another 3D homogeneous self-reconfigurable structure was designed by Kurokawa [24] which is composed of two semi-cylindrical boxes (with a servo each) connected by a link mechanism (Figure 2.9). The semi-cylinders are 66 mm in size and they weigh 350 g. The connecting mechanism utilizes rare-earth magnets for attaching and SMA coil springs for detaching (on one side of the connection). A processor and a communication system are embedded in each module. The proposed mechanism allows the robot to globally move in 3D by moving each local module. The attachments and detachments are limited by the force of the magnets; therefore, a problem might occur if a module has to lift several other modules. The

use of magnets for connections severely limits the available connections (connection faces are either active or passive).

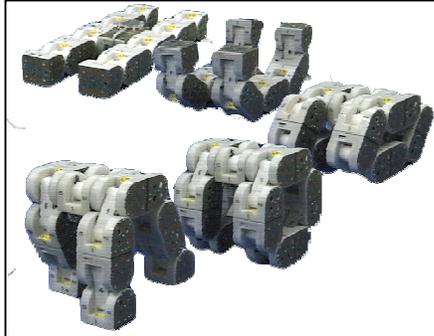


Figure 2.9 Semi-Cylindrical Reconfigurable Robot [24].

2.16 SELF-REPAIRING AND FAULT TOLERANT SYSTEMS

Murata proposed self-repairing systems [25] and Paredis and Khosla [26] proposed fault tolerant systems. A Modular system capable of “self-assembly” and “self-repair” is called as self-repairing system. Self-assembly means that a set of units can form a given shape of the system without outside help. Self-repair means that the system restores its original shape if an arbitrary part of the system is cut off. Fracta and Fractum are the examples of 3D and 2D self-repairing systems.

2.17 PARALLEL ROBOTS

Yang and Chen [27] and Zhiming [28] introduced parallel modular robots (see Figure 2.10), which consist of a set of standardized modules (such as actuators, passive joints, rigid links, mobile platforms, and end-effectors) that can be rapidly assembled into a complete robot with various configurations to overcome the complex kinematics of the closed-loop form.

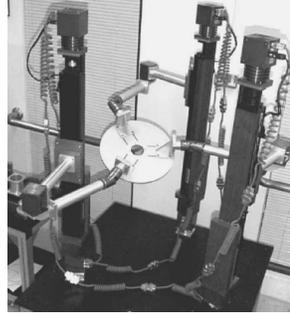


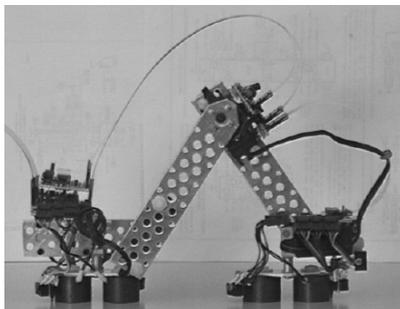
Figure 2.10 Modular Parallel Robot [27].

2.18 INCHWORM

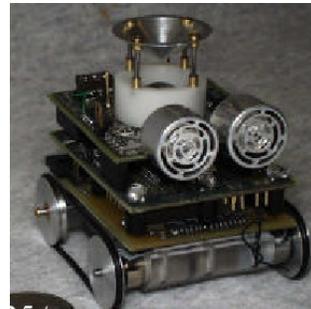
Inchworms [29] (also called loopers) move with a looping movement in which the anterior legs and posterior legs are alternately made fast and released. The Inchworm is a biologically-inspired robot, designed to imitate the movements of the inchworm caterpillar (see Figure 2.11 (a)).

2.19 MILLIBOTS

Figure 2.11 (b) shows Millibots, which was designed by Grabowsky [30]. Millibots are teams of heterogeneous robots that collaborate to map and explore unknown environments.



(a)



(b)

Figure 2.11 (a) Inchworm [29], (b) Millibots [30].

2.20 FIELD ROBOTS

A modular approach to field robots [31], which gives an inventory of prefabricated modules, is used to rapidly and cost-effectively produce a robotic system for a specific task. The inventory includes actuated joints, links, end-effectors, and power units. The same inventory can be assembled in different configurations to perform different tasks (see Figure 2.12).



Figure 2.12 Field Robots [31].

2.21 INDUSTRIAL TYPE MODULAR MANIPULATORS

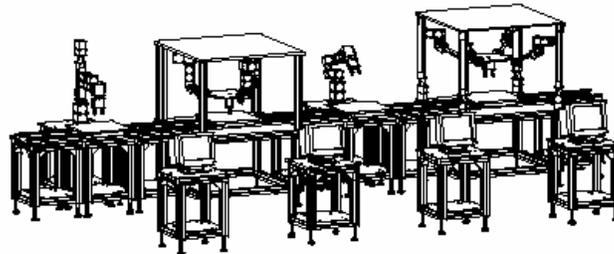
Reconfigurable Modular Manipulator System (RMMS) [32] at Carnegie Mellon University (see Figure 2.13 (a)) utilizes a stock of interchangeable joint (actuator) and link modules of different sizes and performance specifications. The modularity in mechanical, electrical and electronic design allows the user to design the optimal manipulator for the task at hand. The RMMS extends the concept of modularity to also include the control algorithms and task planning software. With this combination of capabilities, the RMMS can be configured to meet the task requirements as they arise at the application site. Some potential application areas of the RMMS are in construction, space, nuclear and manufacturing environments. Other such modular robotic systems include TOMMS at Toshiba Corp. [33], DRRS at the Science University of Tokyo [34] and others (MRS) [35].

Basically, these systems have serial-type geometries with large working envelopes. Using fixed and variable dimension modules, the modular robotic groups

at the Nanyang Technological University and GINTIC Institute of Manufacturing Technology have developed serial, parallel and hybrid type modular robotic workcells [36] (see Figure 2.13 (b)). In addition to the research type modular robotic systems, there are a few commercial modular systems such as RMD-1 by Engineering Services Inc. of Canada and MoRSE (see Figure 2.13 (c)) by AMTEC GmbH of Germany.



(a)



(b)



(c)

Figure 2.13 (a) RMMS [32], (b) Modular Robotic Workcell [36], and (c) A PUMA type robot configuration constructed by MoRSE modules.

2.22 VARIOUS TYPES OF RECONFIGURABLE ROBOTS

Table 2.1 shows a comparison of existing modular reconfigurable robots. Clearly, there is a wide range of possibilities. However, some general characteristics can be observed. Most reconfigurable robot research is based on a homogeneous design and aims to operate in 3D. Mostly, each proposed module can move over neighbors and reconfigure themselves over by attaching and detaching. Genderless connecting mechanisms are not very common. Generally, they are separated into male and female types. The number of internal degrees of freedom per module varies from zero to twelve depending on the desired mobility of each module. Two-DOF module is the most common design. Finally, according to configuration competence, self-reconfiguring robots are quite popular.

Table 2.1 Various Types of Reconfigurable Robots.

	Number of DOF's per unit	Module Composition (Homogeneous or Heterogeneous)	Dimension	Self – reconfiguring	Genderless connecting mechanism
ACM	1-3	Homo	3D	No	No
Tetrobot	3-5	Homo	3D	No	No
CEBOT	1-3	Homo	2D	No	No
Fracta	12	Homo	3D	Yes	Yes
Molecule	4	Homo	3D	Yes	No
Metamorphic	3	Homo	2D	Yes	Yes
Proteo	0	Homo	3D	Yes	Yes
Crystalline	2	Homo	3D	Yes	No
Fractal	6	Homo	3D	Yes	Yes
Fractum	0	Homo	2D	Yes	No
Miniaturized	2	Homo	2D	Yes	No
CONRO	2	Homo	3D	Yes	No
I-Cubes	3	Hetero	3D	Yes	No
Polypod	2	Hetero	3D	No	No
PolyBot	1	Homo	3D	No	No
SemiCylindrical	2	Homo	3D	Yes	No

CHAPTER 3

I-CUBES

The material covered in this Chapter has been taken and inspired from ([20], [21], [37], [38], [39], [40], [41] and [42]).

In this chapter the modular self-reconfigurable bipartite robotic system, which is called as I-Cubes, is introduced. I-Cubes have all the properties that a modular, reconfigurable 3-D robot has. They have the properties of the chain, lattice and mobile kind reconfigurable robots. Therefore, I-Cubes have been chosen for the case studies with the assumption that they represent all of the modular robots. Another reason to select I-Cubes for constituting the stability analysis is the sliding problem of I-Cubes. Ünsal and Khosla [37] made stability analysis which is a part of the decision-making algorithm for the link motions. However their current [37] analysis is limited to overturning, and does not consider sliding stability. In this chapter, the properties of I-Cubes are described to illustrate the stability problem by I-Cubes.

Drawing from the recent research on modular robots and self-reconfigurable structures as well as possible applications of small mobile robots with limited capabilities, Ünsal and Khosla [38] have suggested a modular self-reconfigurable group of robots that consists of two modules with different characteristics. A sufficient number of modules combined as a single entity is capable of self-reconfiguring themselves into defined shapes.

Ünsal and Khosla's design, called, I-Cubes (or ICES-Cubes), is a bipartite system composed of active elements, called links, (used for actuation) and passive elements, called cubes, (used as connectors). The active elements are 3-DOF manipulators that are capable of attaching or detaching to the passive elements which can be positioned and oriented using links. The self-reconfiguration capability enables the system to perform locomotion tasks over difficult terrains since the shape and size of the modular robot can be changed according to the task. A link

can move from one cube face to another, from one cube to another neighboring cube, or move a cube while attached to another (see Figures 3.1 and 3.2). All active and passive modules are capable of permitting power and information flow to the attached modules. When the links move, the structure and the shape of the 3-D system change. I-Cubes are actually both chain kind and lattice kind robots which possess the properties given below.

- Links can be independently controlled; only cubes attached to the moving end of a link are affected by the link motions. All elements are physically, mechanically, and computationally compatible (i.e., any link can connect to any cube, and the cubes have attachment points to receive the link connectors).
- Cube positions fit a cubic lattice to guarantee interlocking of neighboring elements (i.e., the distance from one cube to another is constant while in position to accept the link).
- All elements form a single connected (pseudo-) graph where all links are connected to at least one cube, and all cubes are connected to at least one link. Active elements have sufficient degrees of freedom to complete motions in three-dimensional space.

Since all of the actuation for self-reconfiguration, with the exception of the attachment mechanism is provided by the links, the cubes are reserved for computation, sensing and power resources. If the modules are designed to exchange power and information, the cubes can be equipped with batteries, microprocessors, and sensing modules to create a collective intelligence while the links becomes the muscles of the system. Furthermore, it is possible to remove some of the attachment points on the cubes to provide these modules with different and faster gaits, such as wheeled or treaded locomotion. Specifically, Ünsal and Khosla [38] envision small mobile robots that can reposition themselves to form a group that is capable of changing its gait in order to move over obstacles that a single element cannot overtake. Similar scenarios that require reconfiguration include climbing stairs and traversing pipes.

3.1. GEOMETRIC DESIGN AND LINK ACTUATION

Figure 3.1 and Figure 3.3 shows two links connecting three cubes. If the length of a cube edge is taken as d , the links should have four essential sections of length, $d/2$, d , d and $d/2$. The three rotational degrees of freedom for the links are provided by the joint ($J2$) at the middle, and the joints located at the ends ($J1, J3$). Joints $J1$ and $J3$ are both assumed to be capable of providing continuous 360-degree rotations, while $J2$ can only rotate 270 degrees. In order to keep the cubic lattice formed by the cubes intact, the distance between the cubes must be exactly d , while in position to accept the links. Therefore, the links have been designed [38] to provide this exact distance when the two middle sections are closed to touch (See the link on the right in Figure 3.1).

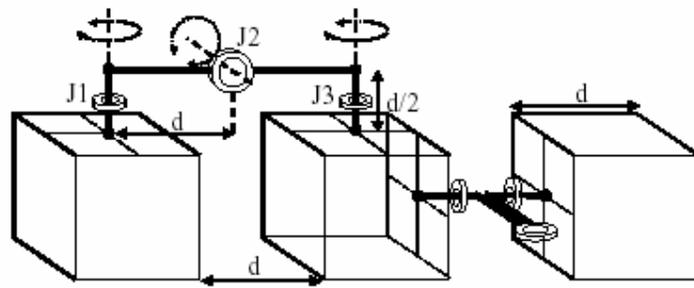


Figure 3.1 Geometric requirements for the system.[37]

Due to the design properties and the attachment capabilities of the I-Cubes, the links can

- Move from one face to another face of a cube. (See Figure 3.2.a)
- Move one cube while attached to another cube. (See Figure 3.2.b)
- Move from one cube to another. (See Figure 3.2.c)

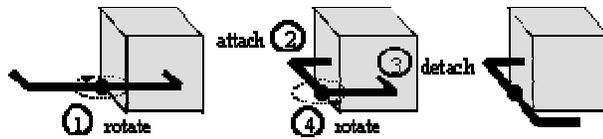


Figure 3.2.a Link moving from one cube face to another. [37]

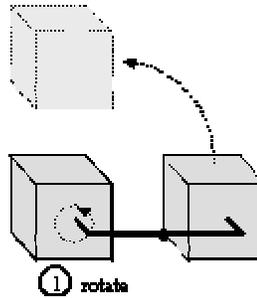


Figure 3.2.b Link lifting a cube while attached to another. [37]

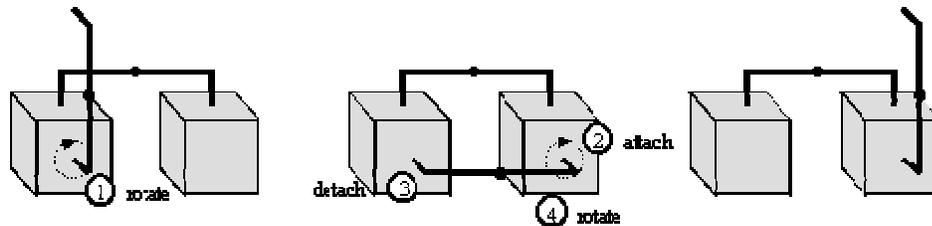


Figure 3.2.c Link moving from one cube to another. [37]

All of the aforementioned motions require the links to be capable of attaching to the cube faces, and performing middle and end joint rotations. Note that it is also possible for a link to move a cube by rotating its middle joint, although not shown here.

3.2. DESIGN OF THE LINKS AND THE CUBES

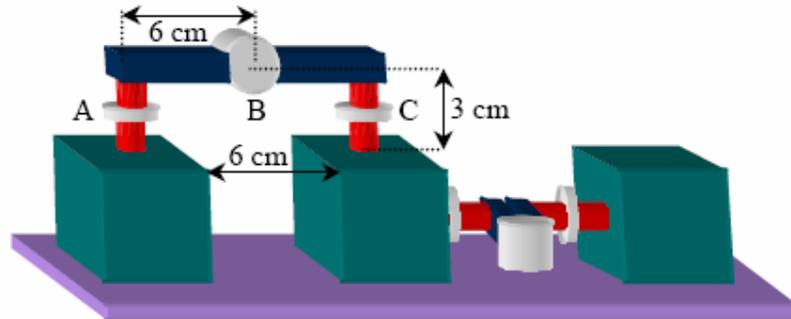


Figure 3.3 Physical view of a 3C2L I-Cube system.[41]

The links have three worm-wheel gear mechanisms driven by small servos to provide continuous rotation at the end joints, and 270-degree rotation at the middle joint. All servos are coupled to worm gears driving the wheels. At the end joints, the wheels are placed on a shaft that the connection pieces are attached. At the middle joint, rotating the servos will rotate one side of the link with respect to the other. Figures (3.4 (b), 3.5 (a) and 3.6) show the design of the link equipped with the servos, and gear structures. The distance between the joint shafts is again equal to d . Unsal and Khosla [38] state that there are two main advantages of using worm-wheel structures coupled with servos. Firstly, the rotational speed of the servo is reduced by the ratio of the gear mechanism. Similarly, the torque provided by the servo is increased by the same ratio. Secondly, the worm-wheel system is an energy efficient solution for actuation. Since the wheel cannot drive the worm, the servos do not have to be powered continuously to hold the links in a specific position.

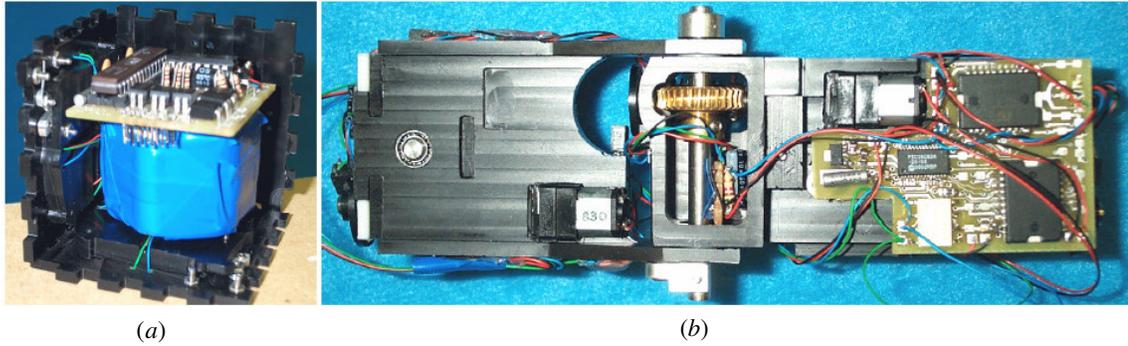


Figure 3.4 (a) The cube and (b) the link.[37]

The cubes are passive elements that consist of at most six attachment points for the link connectors. They are not capable of moving by themselves. A cube attached to a link can either be rotated, translated in a plane, or act as a pivot point for a moving link. The role of the cube depends on the position and motion of the link as well as the connections of the modules. Cubes do not contribute to the self-reconfiguring motions with the exception of the motion to lock the link connector in place. However, available space inside cubes (see Figure 3.4 (a)) can be used for batteries to power all actuators, sensing and control modules. Figure 3.5 (b) shows CAD image of a cube retrofitted with five attachment points for cross-shaped link connectors. The system is based on twist and-lock mechanical behavior, and the details are given in the next section.

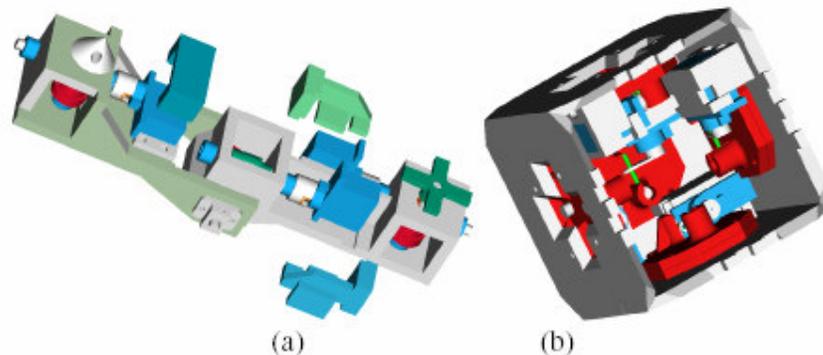


Figure 3.5 CAD images: (a) The link is shown with two different connectors (servo holders expanded); (b) the cube is shown with five faceplates.[38]

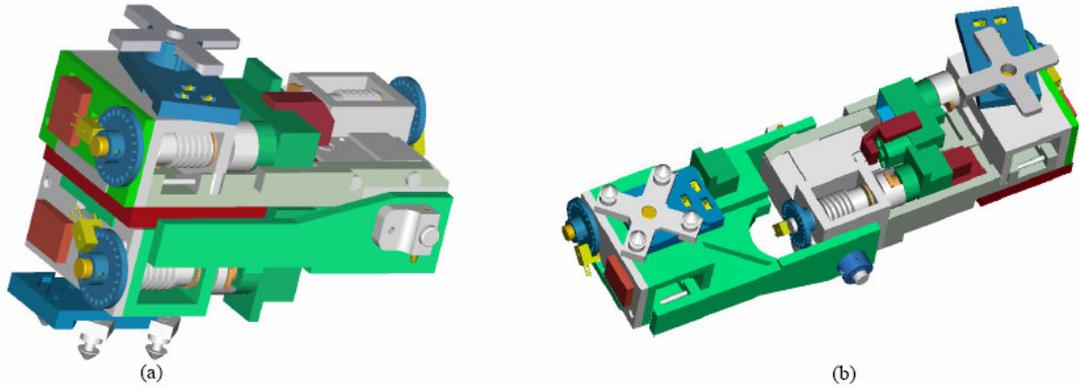


Figure 3.6 ProE drawings of the link: (a) idle joint fully open and (b) at zero degrees.[37]

3.3. TWIST-AND-LOCK ATTACHMENT MECHANISM

A self-reconfiguring modular robotic system must consist of elements that are capable of attaching/detaching themselves to/from neighboring elements. For that purpose, Ünsal and Khosla [38], [37] have designed a cross-shaped link connector for the links that enters and twists inside an opening on the cube face (see Figure 3.7 and Figure 3.8). After entering the similarly shaped opening, the link connector twists to lock in place. The twist-and-lock connection mechanism is designed [38], [37] for the connector to enter (see 1 in Figure 3.7) and rotate (see 2 in Figure 3.7) to its locked position inside an opening on the cube face plate. Once the connector is in locked position, a sliding latch rotates into a position to stop the connector from turning (see Figure 3.8). This limits the motion of the link away from the cube surface, while the sliding latch stops the free rotation of the link end with respect to the cube. The surface of the cross-shaped attachment piece is the connection point for power and communication lines.

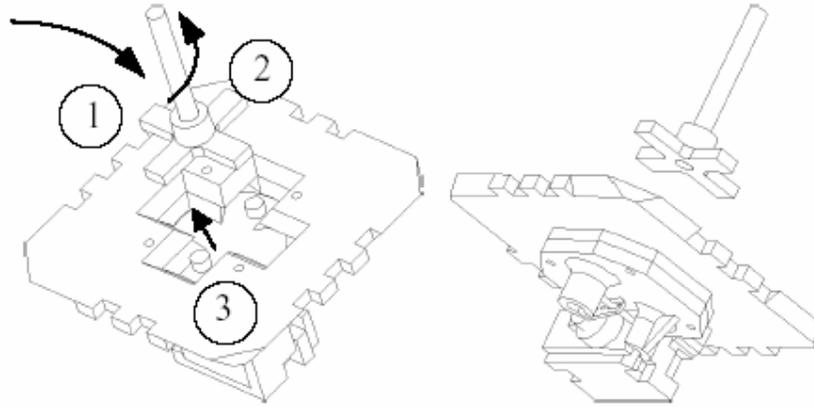


Figure 3.7 Cross-shaped connector with twist-and-lock mechanism [38].

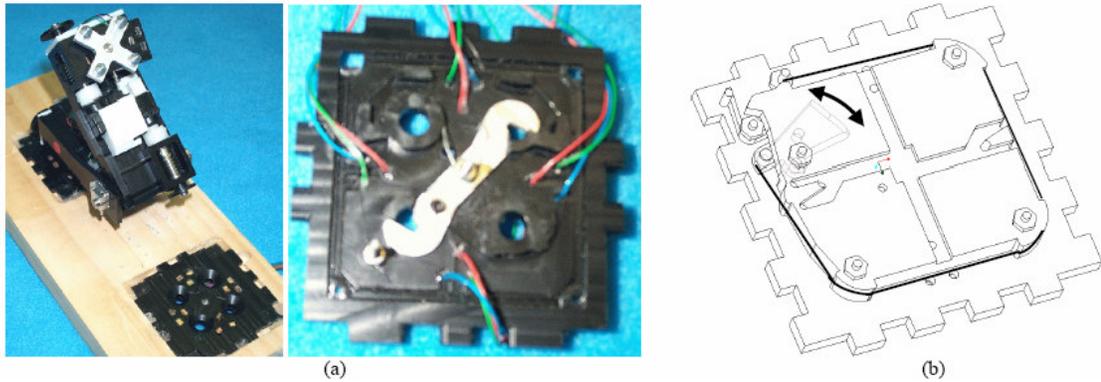


Figure 3.8 Attachment mechanisms: (a) 4-pegs with latch at the center of the faceplate, (b) twist-and-lock mechanism with sliding latch.[37]

3.4. 3-D RECONFIGURATION

Self-reconfiguring robots with lattice kind reconfiguration usually require a large number of modules to create stable gaits. A group of four links and four cubes is capable of moving in three dimensions for most situations (e.g., moving over obstacles, translating without tipping over) [38]. In situations that a statically stable configuration cannot be found, it may be possible to use a free link (with only one end attached) to support the structure.

If one link is capable of moving itself and an attached cube, simultaneous motions in 3-D are possible for the cubes. Combining several of these motions in sequence, it is possible for a group of links and cubes to change shape and / or move in certain direction.

In order for a group of cubes and links to move and self-reconfigure from one position / shape to another; suitable link actions such as detaching from and attaching to cubes (see Figure 3.9), and joint rotations should be combined into a sequence. Figure 3.10 shows a group of three links and three cubes (called as 3L3C) on the ground traveling from left to right. The sequence of states (i.e., configurations) from left to right show the changes in the cube and link positions. Gray color indicates active elements that are moved to reach the next state. Numbers between states give the total number of 90-degree rotations to be completed by the active link. Ünsal and Khosla state that the given sequence of actions may not be feasible for an actual implementation due to static and dynamic equilibrium constraints [38]. Indeed there are many alternative solutions such as combining individual link rotations in a different sequence, combining simultaneous link motions (which probably result in faster group movement) and solutions with link rotations other than 90-degree increments. The simulation program that is developed by Ünsal and Khosla, however, considers only 90-degree motions [38].

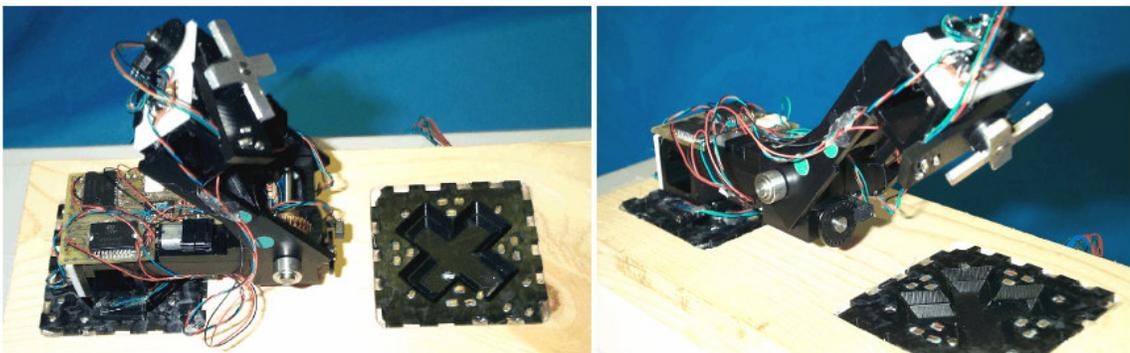


Figure 3.9 Link approaching the faceplate.[37]

As seen in Figure 3.10, the group returns to its initial configuration after thirty 90-degree rotations of links. This sequence is in fact a solution when the final

conditions of the cubes are defined as $4d$ units to the right of the initial configuration.

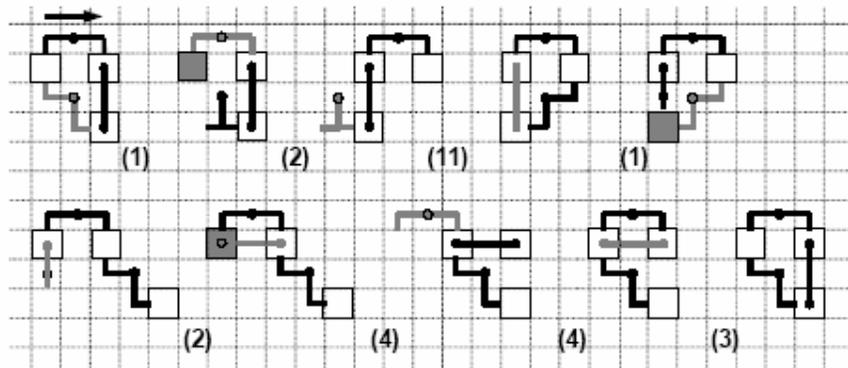


Figure 3.10 A group of three links and three cubes in linear motion (image sequence left-to-right).[38]

To illustrate more complex motion sequences, the snapshots of a possible scenario for a group of four links and four cubes (4L4C) are presented in Figure 3.11. The 4L4C group is capable of moving to a higher surface (e.g., stair climbing) by reconfiguring itself. Connections between the elements are kept such that the system forms a single connected graph at any time. This enables modules to exchange information and power during rearrangement. There are several time intervals, where multiple links move simultaneously. Furthermore, the links do not have to complete 90-degree motions to, for example, detach from a cube or move from one cube face to another. There are few positions where a link moves on a pivot cube held in the air by another cube.

Ünsal and Khosla speculate that, the middle joint of a link is strong enough to hold a cube and a nonmoving link. Also, note that the cube faces initially on the ground and several other faces (i.e. attachment points) are not used during the reconfiguration sequence.

As seen in the final image (Figure 3.11 (f)), the cubes are still oriented with the same faces looking up at the end of the action sequence. To guarantee this result, some of the cubes need to be re-oriented in the earlier phases of the solution

sequence, as seen in third and fifth images. This sequence for the problem has been generated manually by Ünsal and Khosla who believe that the combination of four cubes and four links is the minimal group that is feasible for 3-D motion and self-reconfiguration. Increasing the number of modules in the group will lead to more stable and capable system.

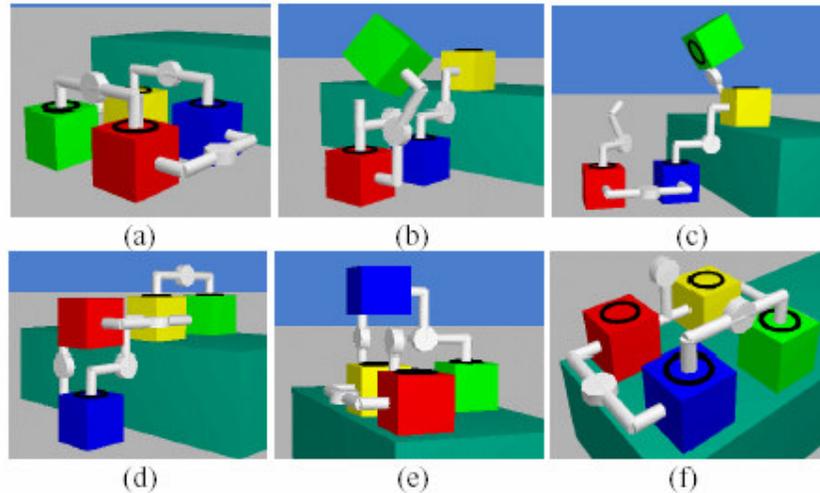


Figure 3.11 A 4LAC group reconfiguring to move over an obstacle.[38]

Another example that combines self-reconfiguration with faster locomotion capability is given in Figure 3.12. Since the cubes are passive elements that do not contribute to the reconfiguration motions with the exception of the locking mechanism, they can be equipped with capabilities that provide different gaits (such as wheel or treads) and task-oriented modules such as sensors. In Figure 3.12, the leftmost robot in the first image carries a camera directed at the wall. This robot is obviously not capable of seeing what is behind this obstacle. Assuming these wheeled robots are capable of carrying one or more links, they can move into a position to form a single entity. If initial actions forming the group are possible, then the newly formed group can self-reconfigure into a tower. For an initial configuration and a dependent sequence of actions, it is possible to move the robot with the camera on top of others. The required number of faces with attachment points for each robot is three or four for this specific scenario. These must include the face initially on top.

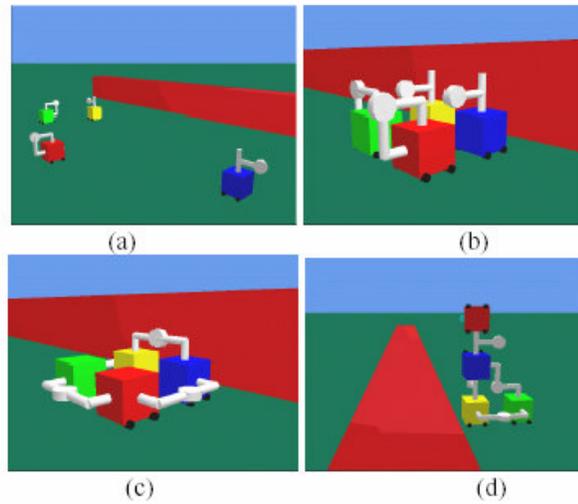


Figure 3.12 A group of four links and four cubes capable of locomotion forming a tower.[38]

This scenario illustrates an important characteristic of this robotic system. A heterogeneous group of small robots combines individual robot capabilities with self reconfiguration to complete a task that would not be possible with individual robots of relatively small size.

CHAPTER 4

STATIC STABILITY ANALYSIS

In this chapter, the necessary and sufficient conditions for a rigid body to be in static equilibrium are determined. Furthermore, a novel definition of stability, called percentage stability, is introduced. In order to define percentage stability, the friction cone is approximated by a pyramid and then linear programming is used. The effects of the number of faces of the pyramid and the number of contact points are also investigated.

Consider a rigid body resting on the ground and assume that there is Coulomb friction between the body and the ground (see Figure 4.1). Suppose that the resultant of all external forces and moments acting on the rigid body is reduced to a resultant force, \vec{F}_{ex} (acting at O_1), and an accompanying resultant moment, \vec{M}_{ex} . Let \vec{F}_R and \vec{M}_O be the resultant force (acting at O) and the accompanying resultant moment to be applied by the ground on the rigid body such that the body is in static equilibrium under the action of its weight, \vec{F}_{ex} , \vec{M}_{ex} and \vec{F}_R , \vec{M}_O .

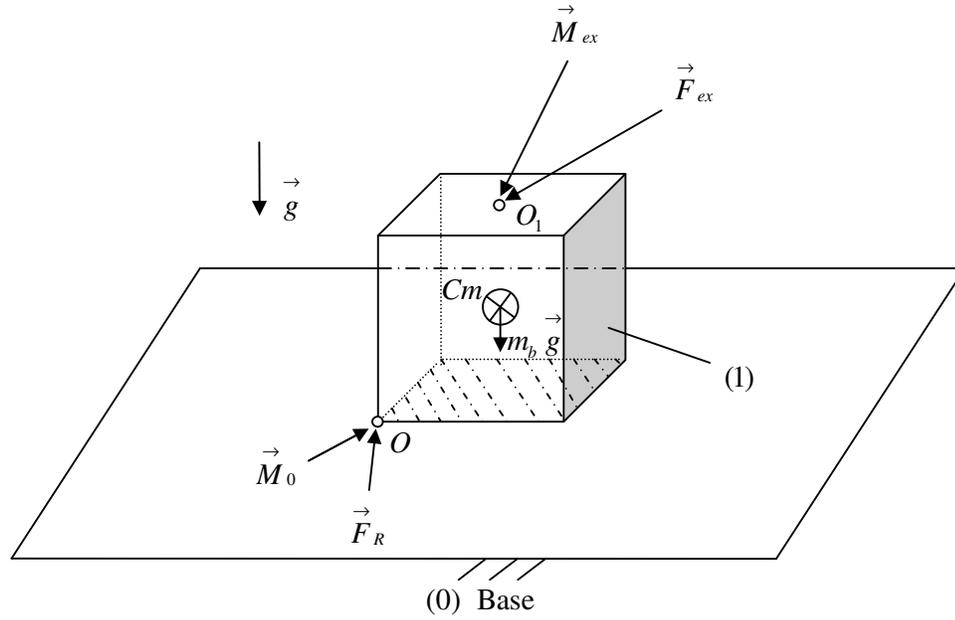


Figure 4.1 A block resting on a ground with friction

Clearly, \vec{F}_R and \vec{M}_O may be solved by using the static equilibrium equations

$$\sum \vec{F} = \vec{0}$$

and

$$\sum \vec{M} \Big|_{C_m} = \vec{0}$$

yielding

$$\vec{F}_R = -(\vec{F}_{ex} + m_b \vec{g})$$

(4.1)

$$\vec{M}_0 = -(C_m \vec{O} \times (-\vec{F}_{ex} - m_b \vec{g})) + C_m \vec{O}_1 \times \vec{F}_{ex} + \vec{M}_{ex}$$

where

C_m : The center of mass of the rigid body.

O : Point of application of \vec{F}_R .

O_1 : Point of application of \vec{F}_{ex} .

m_b : Mass of the rigid body.

\vec{g} : Gravitational acceleration.

Consider, now, the contact region between the rigid body and the ground (see Figure 4.2) and the following notation.

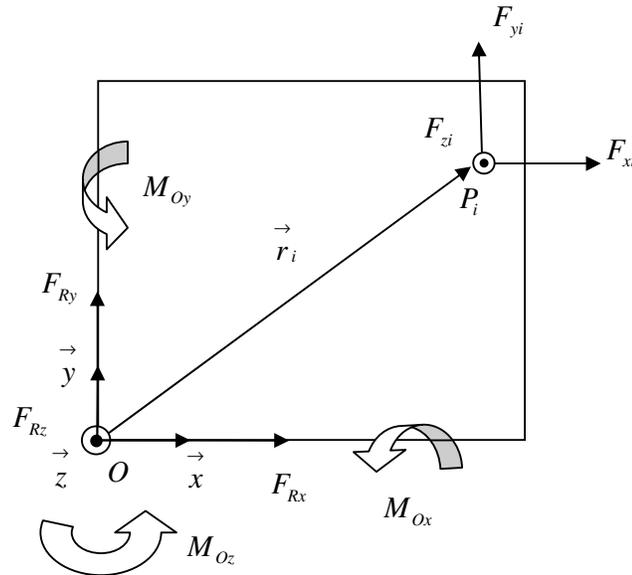


Figure 4.2 Top view of the contact region between the cube and the ground.

$Oxyz$: Body fixed coordinate system such that the z axis is directed from the ground towards the body.

F_{Rx}, F_{Ry}, F_{Rz} : x , y and z components of \vec{F}_R in the $Oxyz$ system.

M_{Ox}, M_{Oy}, M_{Oz} : x , y and z components of \vec{M}_O in the $Oxyz$ system.

- P_i : i 'th contact point. ($i = 1, 2, \dots, n_c$)
- n_c : Number of contact points.
- \vec{F}_i : Contact force (applied by the ground on the body) at the i 'th contact point.
- F_{xi}, F_{yi}, F_{zi} : x , y and z components of \vec{F}_i in the $Oxyz$ system.
- $\vec{r}_i = (r_{xi}, r_{yi}, 0)$: Position vector of P_i with respect to the $Oxyz$ system.

A necessary, but not sufficient, condition for the body shown in Figure 4.1 to be in static equilibrium is that the n_c contact forces, \vec{F}_i 's, are statically equivalent to the force system consisting of \vec{F}_R and \vec{M}_O , where

$$\vec{F}_i = F_{xi} \vec{x} + F_{yi} \vec{y} + F_{zi} \vec{z} \quad (4.2)$$

and the moment of the contact force \vec{F}_i about the origin (O) is

$$\vec{M}_{Oi} = \vec{r}_i \times \vec{F}_i \quad (4.3)$$

Therefore from equations (4.2) and (4.3), it follows that

$$\begin{aligned} F_{Rx} &= \sum_{i=1}^{n_c} F_{xi} \\ F_{Ry} &= \sum_{i=1}^{n_c} F_{yi} \\ F_{Rz} &= \sum_{i=1}^{n_c} F_{zi} \\ M_{Ox} &= \sum_{i=1}^{n_c} r_{yi} F_{zi} \\ M_{Oy} &= \sum_{i=1}^{n_c} -r_{xi} F_{zi} \\ M_{Oz} &= \sum_{i=1}^{n_c} (-r_{yi} F_{xi} + r_{xi} F_{yi}) \end{aligned} \quad (4.4)$$

F_{xi} and F_{yi} are the components of the friction force generated between the body and the ground. Hence, they should satisfy Coulomb's law of friction, i.e.

$$F_{xi}^2 + F_{yi}^2 - (\mu F_{zi})^2 \leq 0, \text{ for } i = 1, 2, \dots, n_c \quad (4.5)$$

where μ is the coefficient of static friction which is assumed to be the same for all contact points.

F_{zi} , on the other hand, is the normal force applied by the ground on the body. Therefore, one must have

$$F_{zi} \geq 0, \text{ for } i = 1, 2, \dots, n_c \quad (4.6)$$

Therefore, the necessary and sufficient conditions, for the body in Figure 4.1 to be in static equilibrium are given by equations (4.4), (4.5) and (4.6) where \vec{F}_R and \vec{M}_0 are obtained via equation (4.1).

Assuming that \vec{F}_R and \vec{M}_0 are obtained via equation (4.1), equation (4.4) constitutes a set of 6 linear equations in the $(3n_c)$ unknowns namely, $F_{x1}, F_{x2}, \dots, F_{xn_c}; F_{y1}, F_{y2}, \dots, F_{yn_c}; F_{z1}, F_{z2}, \dots, F_{zn_c}$. These equations can be decoupled into 3 equations involving n_c unknowns and another 3 equations involving another $2n_c$ unknowns as shown in the next section.

4.1. STATICAL EQUIVALENCE EQUATIONS

Equation (4.4) can be conveniently rearranged in the form

$$\left[A_{n_c} \right] \vec{x}_{n_c} = \vec{b} \quad (4.7)$$

$\left[A_{n_c} \right]$ is the $(6 \times 3n_c)$ coefficient matrix given by

$$\left[A_{n_c} \right] = \begin{bmatrix} 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ r_{y1} & r_{y2} & \cdots & r_{yn_c} & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ -r_{x1} & -r_{x2} & \cdots & -r_{xn_c} & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 \\ 0 & 0 & \cdots & 0 & -r_{y1} & -r_{y2} & \cdots & -r_{yn_c} & r_{x1} & r_{x2} & \cdots & r_{xn_c} \end{bmatrix}$$

where,

\vec{x}_{n_c} is the $(3n_c \times 1)$ vector of unknowns given by

$$\vec{x}_{n_c} = \begin{bmatrix} F_{z1} \\ F_{z2} \\ \vdots \\ F_{zn_c} \\ F_{x1} \\ F_{x2} \\ \vdots \\ F_{xn_c} \\ F_{y1} \\ F_{y2} \\ \vdots \\ F_{yn_c} \end{bmatrix}$$

and \vec{b} is the known (6×1) right hand side vector given by

$$\vec{b} = \begin{bmatrix} F_{Rz} \\ M_{0x} \\ M_{0y} \\ F_{Rx} \\ F_{Ry} \\ M_{0z} \end{bmatrix}$$

Although equation (4.7) seems to be a linear equation system involving $3n_c$ unknowns and 6 equations, actually the equations are fully decoupled into two sets of 3 linear equations involving the (n_c) unknowns $F_{z1}, F_{z2}, \dots, F_{zn_c}$ and the ($2n_c$) unknowns $F_{x1}, F_{x2}, \dots, F_{xn_c}; F_{y1}, F_{y2}, \dots, F_{yn_c}$ which may be represented as

$$[A_{n_c,11}] \vec{x}_{n_c,1} = \vec{b}_1 \quad (4.8)$$

$$[A_{n_c,22}] \vec{x}_{n_c,2} = \vec{b}_2 \quad (4.9)$$

where,

$$[A_{n_c,11}] = \begin{bmatrix} \vec{a}_{z1} & \vec{a}_{z2} & \dots & \vec{a}_{zn_c} \end{bmatrix}$$

$$\vec{a}_{zi} = \begin{bmatrix} 1 \\ r_{yi} \\ -r_{xi} \end{bmatrix}$$

$$\vec{x}_{n_c,1} = \begin{bmatrix} F_{z1} \\ F_{z2} \\ \vdots \\ F_{zn_c} \end{bmatrix}$$

$$\vec{b}_1 = \begin{bmatrix} F_{Rz} \\ M_{0x} \\ M_{0y} \end{bmatrix}$$

$$[A_{n_c,22}] = \begin{bmatrix} \vec{a}_{x1} & \vec{a}_{x2} & \cdots & \vec{a}_{xn_c} & \vec{a}_{y1} & \vec{a}_{y2} & \cdots & \vec{a}_{yn_c} \end{bmatrix}$$

$$\vec{a}_{xi} = \begin{bmatrix} 1 \\ 0 \\ -r_{yi} \end{bmatrix}$$

$$\vec{a}_{yi} = \begin{bmatrix} 0 \\ 1 \\ r_{xi} \end{bmatrix}$$

$$\vec{x}_{n_c,2} = \begin{bmatrix} F_{x1} \\ F_{x2} \\ \vdots \\ F_{xn_c} \\ F_{y1} \\ F_{y2} \\ \vdots \\ F_{yn_c} \end{bmatrix}$$

$$\vec{b}_2 = \begin{bmatrix} F_{Rx} \\ F_{Ry} \\ M_{0z} \end{bmatrix}$$

Now, let's have a look at the solutions of equations (4.8) and (4.9) for various values of n_c . Here, it is convenient to take the first contact point, P_1 , to be coincident with O as shown in Figure 4.3.

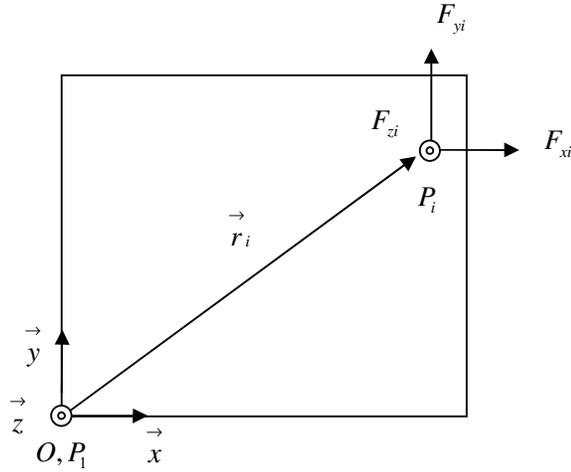


Figure 4.3 Contact region between the cube and the ground with the first contact point be at the origin (O).

1) $n_c = 2$:

Equation (4.8) leads to 3 equations in 2 unknowns (F_{z1}, F_{z2}). Therefore solution is not possible for a given general loading (F_{Rz}, M_{0x}, M_{0y}).

2) $n_c = 3$:

Equation (4.8) can be written as

$$[A_{z1,z2,z3}]x_{z1,z2,z3} = \vec{b}_1 \quad (4.10)$$

where,

$$[A_{z_1, z_2, z_3}] = [A_{3,11}] = \left[\begin{array}{c|c|c} \vec{a}_{z_1} & \vec{a}_{z_2} & \vec{a}_{z_3} \\ \hline \hline \hline \end{array} \right] = \begin{bmatrix} 1 & 1 & 1 \\ 0 & r_{y2} & r_{y3} \\ 0 & -r_{x2} & -r_{x3} \end{bmatrix}$$

$$x_{z_1, z_2, z_3}^{\rightarrow} = \begin{bmatrix} F_{z_1} \\ F_{z_2} \\ F_{z_3} \end{bmatrix}$$

Equation (4.10) has a unique solution given by

$$\begin{bmatrix} F_{z_1} \\ F_{z_2} \\ F_{z_3} \end{bmatrix} = \begin{bmatrix} \frac{M_{Ox}r_{x2} - M_{Ox}r_{x3} + M_{Oy}r_{y2} + F_{Rz}r_{x3}r_{y2} - M_{Oy}r_{y3} - F_{Rz}r_{x2}r_{y3}}{r_{x3}r_{y2} - r_{x2}r_{y3}} \\ \frac{M_{Ox}r_{x3} + M_{Oy}r_{y3}}{r_{x3}r_{y2} - r_{x2}r_{y3}} \\ \frac{M_{Ox}r_{x2} + M_{Oy}r_{y2}}{-r_{x3}r_{y2} + r_{x2}r_{y3}} \end{bmatrix}$$

provided that

$$\det[A_{z_1, z_2, z_3}] = r_{x2}r_{y3} - r_{y2}r_{x3} \neq 0$$

It should be noted that $\det[A_{z_1, z_2, z_3}] \neq 0$ implies that the 3 contact points are not on the same straight line.

Equation (4.9), on the other hand, can be written as

$$\left[\begin{array}{ccc|ccc} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & -r_{y2} & -r_{y3} & 0 & r_{x2} & r_{x3} \end{array} \right] \begin{bmatrix} F_{x1} \\ F_{x2} \\ F_{x3} \\ F_{y1} \\ F_{y2} \\ F_{y3} \end{bmatrix} = \begin{bmatrix} F_{Rx} \\ F_{Ry} \\ M_{Oz} \end{bmatrix} \quad (4.11)$$

which represents 3 linear equations in 6 unknowns. Using equation (4.11) any appropriate 3 unknowns can be solved in terms of the remaining 3. The unknown forces F_{x1}, F_{x2}, F_{x3} , for instance cannot be solved since the determinant of the coefficient matrix would be zero. Similarly, it is not possible to solve for F_{y1}, F_{y2} and F_{y3} from equation (4.11).

Firstly, assume that one would like to solve the unknown forces F_{x1}, F_{x2}, F_{y1} . Equation (4.11) can then be arranged as

$$\left[A_{x1,x2,y1} \right] x_{x1,x2,y1}^{\rightarrow} = \vec{b}_2 - \left[A_{x3,y2,y3} \right] f_{x3,y2,y3}^{\rightarrow} \quad (4.12)$$

where,

$$\left[A_{x1,x2,y1} \right] = \begin{bmatrix} \vec{a}_{x1} & \vec{a}_{x2} & \vec{a}_{y1} \end{bmatrix}$$

$$\left[A_{x3,y2,y3} \right] = \begin{bmatrix} \vec{a}_{x3} & \vec{a}_{y2} & \vec{a}_{y3} \end{bmatrix}$$

$$x_{x1,x2,y1}^{\rightarrow} = \begin{bmatrix} F_{x1} \\ F_{x2} \\ F_{y1} \end{bmatrix} = \text{Vector of unknown forces.}$$

$$f_{x3,y2,y3}^{\rightarrow} = \begin{bmatrix} F_{x3} \\ F_{y2} \\ F_{y3} \end{bmatrix} = \text{Vector of free forces.}$$

Equation (4.12) has the unique solution given by

$$x_{x1,x2,y1}^{\rightarrow} = \left[A_{x1,x2,y1} \right]^{-1} \vec{b}_2 - \left[A_{x1,x2,y1} \right]^{-1} \left[A_{x3,y2,y3} \right] f_{x3,y2,y3}^{\rightarrow} \quad (4.13)$$

provided that

$$\det \left[A_{x1,x2,y1} \right] = r_{y2} \neq 0$$

Using a similar notation, equation (4.9) can be arranged as

$$\left[A_{x_1, y_1, y_2} \right] x_{x_1, y_1, y_2}^{\rightarrow} = \vec{b}_2 - \left[A_{x_2, x_3, y_3} \right] f_{x_2, x_3, y_3}^{\rightarrow}$$

yielding the unique solution given by

$$x_{x_1, y_1, y_2}^{\rightarrow} = \left[A_{x_1, y_1, y_2} \right]^{-1} \vec{b}_2 - \left[A_{x_1, y_1, y_2} \right]^{-1} \left[A_{x_2, x_3, y_3} \right] f_{x_2, x_3, y_3}^{\rightarrow} \quad (4.14)$$

provided that

$$\det \left[A_{x_1, y_1, y_2} \right] = r_{x_2} \neq 0.$$

Note that one cannot have $r_{x_2} = r_{y_2} = 0$ since this would imply that P_2 is coincident with P_1 . Therefore, either one of the solutions given by (4.13) or (4.14) will always be valid.

3) $n_c \geq 4$:

In the case of 4 or more contact points, it is clear that one can always solve 6 appropriate unknown force components from the set $(F_{x_1}, F_{x_2}, \dots, F_{x_{n_c}}; F_{y_1}, F_{y_2}, \dots, F_{y_{n_c}}; F_{z_1}, F_{z_2}, \dots, F_{z_{n_c}})$ in terms of the remaining $(3n_c - 6)$ force components provided that there exists 3 contact points which do not lie on the same straight line. Therefore, the necessary and sufficient conditions for the body in Figure 4.1 to be in static equilibrium (namely, equations (4.4), (4.5) and (4.6)) reduce to $2n_c$ inequalities (given by (4.5) and (4.6)) involving $(3n_c - 6)$ independent force components designated by

$$\vec{f} = [f_1, f_2, \dots, f_{3n_c-6}]^T \quad (4.15)$$

Therefore, the body in Figure 4.1 will be potentially stable (for the given loading) if there exists a set of force components $f_1, f_2, \dots, f_{3n_c-6}$ (all of which are

real) which satisfy the $2n_c$ inequalities given by equations (4.5) and (4.6). This problem can be solved by using the FindInstance command of MATHEMATICA. Indeed, the execution time increases, nonlinearly, with increasing n_c . Actually, the execution time becomes infinite with a Centrino 1.6 MHz computer with 512 MB Ram (i.e., no solution can be obtained at all if n_c exceeds 13).

It should be noted that the problem (of determining potential stability) is nonlinear because of the Coulomb's law of friction given by the inequalities (4.5). Therefore, if these nonlinear inequalities are approximated by linear ones, the problem becomes linear and one can efficiently apply Linear Programming to determine the potential stability of a rigid body. Therefore, linearization of Coulomb's law will be discussed in the next section.

4.2. LINEARIZATION OF COULOMB'S LAW OF FRICTION

Coulomb's law of friction (i.e. inequality (4.5)) and inequality (4.6) written for the i 'th contact point imply that the feasible points (in the $F_{xi}F_{yi}F_{zi}$ -space) must lie inside the right circular friction cone shown in Figure 4.4. A conservative approximation to this cone would be the rectangular pyramid inside the interior of the cone shown in Figure 4.4.

In this case the (n_c) non-linear constraints given by inequality (4.5) are to be replaced by the ($4n_c$) linear inequalities given by

$$\begin{aligned}
 c_m F_{zi} + F_{xi} &\geq 0 \\
 c_m F_{zi} - F_{xi} &\geq 0 \\
 c_m F_{zi} + F_{yi} &\geq 0 \\
 c_m F_{zi} - F_{yi} &\geq 0
 \end{aligned}
 \quad \text{for } 1 \leq i \leq n_c
 \tag{4.16}$$

where,

$$c_m = \frac{\mu}{\sqrt{2}}$$

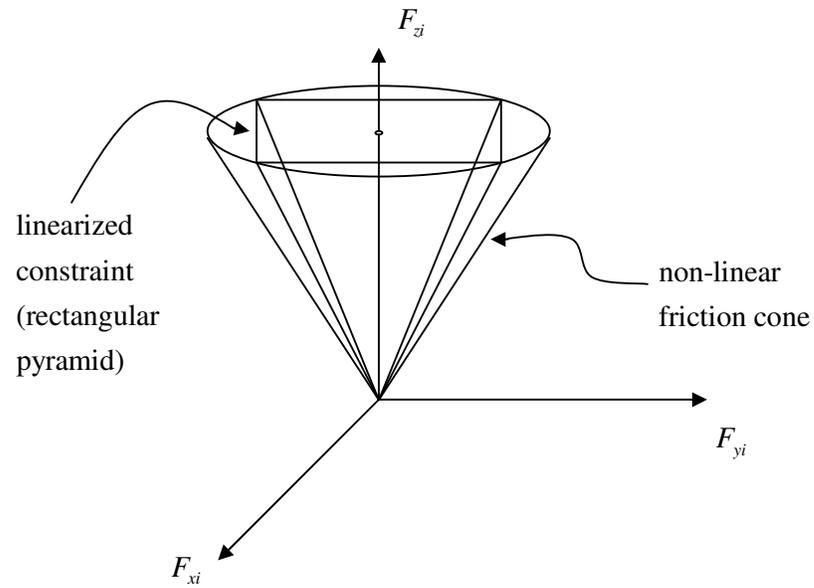


Figure 4.4 Nonlinear friction cone and the approximating rectangular pyramid.[4]

If one requires more accurate approximations to the friction cone, the hexagonal, octagonal, decagonal or dodecagonal pyramids (Figure 4.5 (b), (c), (d) and (e), respectively) could be used.

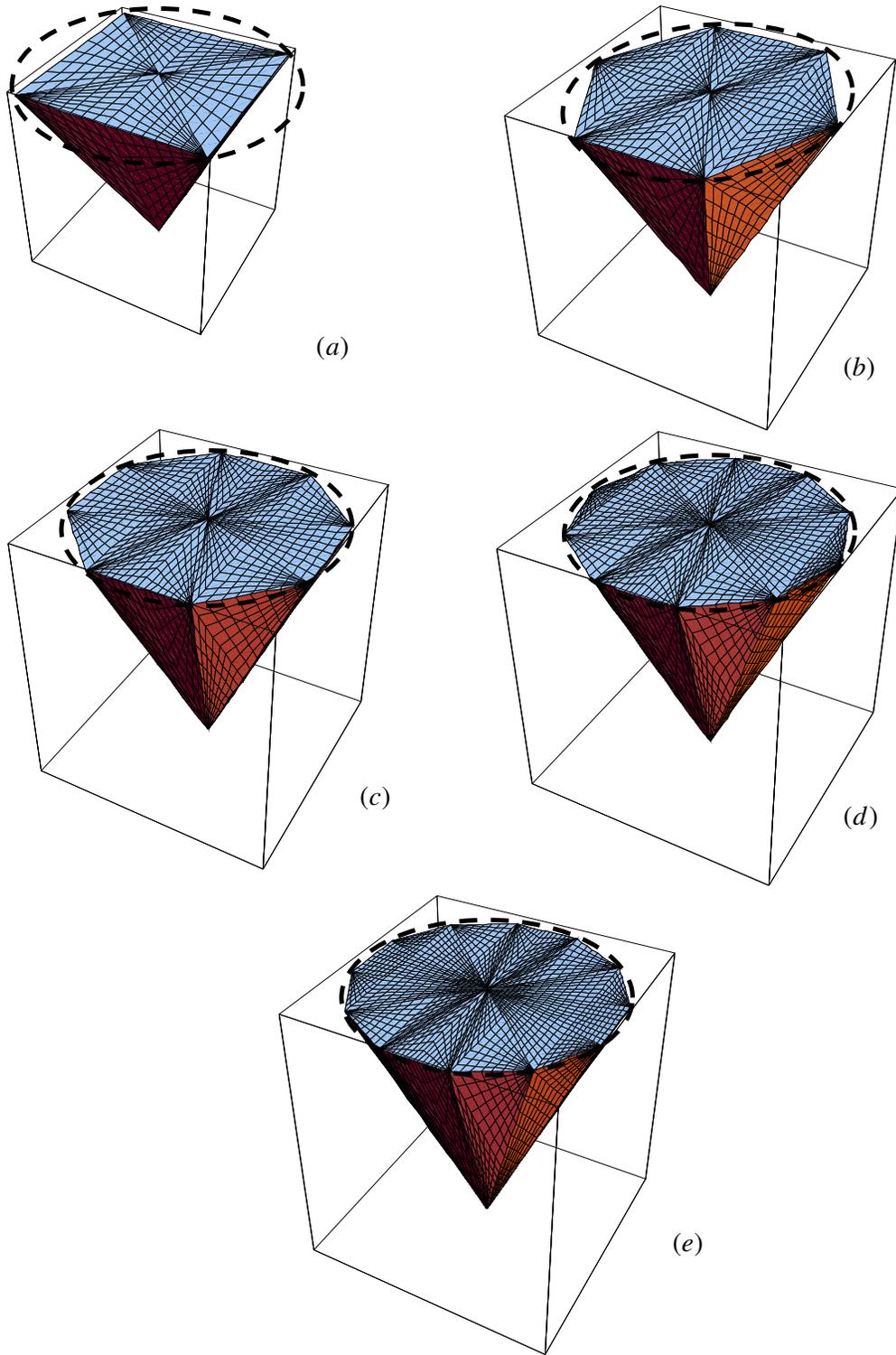


Figure 4.5 Linear rectangular (a), hexagonal (b), octagonal (c), decagonal (d) and dodecagonal (e) pyramid approximations for nonlinear Coulomb friction.

If a hexagonal pyramid is used to approximate the friction cone, the (n_c) non-linear constraints given by inequality (4.5) are to be replaced by the ($6n_c$) linear constraints given by

$$\begin{aligned}
-F_{yi} + c_1 F_{zi} &\geq 0 \\
F_{yi} + c_1 F_{zi} &\geq 0 \\
c_2 F_{xi} - F_{yi} + c_3 F_{zi} &\geq 0 \\
c_2 F_{xi} + F_{yi} + c_3 F_{zi} &\geq 0 \\
-c_2 F_{xi} + F_{yi} + c_3 F_{zi} &\geq 0 \\
-c_2 F_{xi} - F_{yi} + c_3 F_{zi} &\geq 0
\end{aligned}
\quad \text{for } 1 \leq i \leq n_c \quad (4.17)$$

where,

$$\begin{aligned}
c_1 &= 0.866025\mu \\
c_2 &= 1.73205 \\
c_3 &= 1.73205\mu
\end{aligned}$$

If an octagonal pyramid is used to approximate the friction cone, the (n_c) non-linear constraints given by inequality (4.5) are to be replaced by the ($8n_c$) linear constraints given by

$$\begin{aligned}
-F_{yi} + c_4 F_{zi} &\geq 0 \\
F_{yi} + c_4 F_{zi} &\geq 0 \\
F_{xi} + c_5 F_{zi} &\geq 0 \\
-F_{xi} + c_5 F_{zi} &\geq 0 \\
F_{xi} - F_{yi} + c_6 F_{zi} &\geq 0 \\
F_{xi} + F_{yi} + c_6 F_{zi} &\geq 0 \\
-F_{xi} + F_{yi} + c_6 F_{zi} &\geq 0 \\
-F_{xi} - F_{yi} + c_6 F_{zi} &\geq 0
\end{aligned}
\quad \text{for } 1 \leq i \leq n_c \quad (4.18)$$

where,

$$c_4 = 0.92388\mu$$

$$c_5 = \cos\left(\frac{\pi}{8}\right)\mu$$

$$c_6 = 1.30656\mu$$

If a decagonal pyramid is used to approximate the friction cone, the (n_c) non-linear constraints given by inequality (4.5) are to be replaced by the $(10n_c)$ linear constraints given by

$$\begin{aligned}
F_{yi} + c_7 F_{zi} &\geq 0 \\
-F_{yi} + c_7 F_{zi} &\geq 0 \\
c_8 F_{xi} + F_{yi} + c_9 F_{zi} &\geq 0 \\
c_8 F_{xi} - F_{yi} + c_9 F_{zi} &\geq 0 \\
-c_8 F_{xi} + F_{yi} + c_9 F_{zi} &\geq 0 \\
-c_8 F_{xi} - F_{yi} + c_9 F_{zi} &\geq 0 \\
c_{10} F_{xi} + F_{yi} + c_{11} F_{zi} &\geq 0 \\
c_{10} F_{xi} - F_{yi} + c_{11} F_{zi} &\geq 0 \\
-c_{10} F_{xi} + F_{yi} + c_{11} F_{zi} &\geq 0 \\
-c_{10} F_{xi} - F_{yi} + c_{11} F_{zi} &\geq 0
\end{aligned}
\quad \text{for } , 1 \leq i \leq n_c \tag{4.19}$$

where,

$$c_7 = 0.951057\mu$$

$$c_8 = 3.07768$$

$$c_9 = 3.07768\mu$$

$$c_{10} = 0.726543$$

$$c_{11} = 1.17557\mu$$

If a dodecagonal pyramid is used to approximate the friction cone, the (n_c) non-linear constraints given by inequality (4.5) are to be replaced by the $(12n_c)$ linear constraints given by

$$\begin{aligned}
F_{yi} + c_{12}F_{zi} &\geq 0 \\
-F_{xi} + c_{12}F_{zi} &\geq 0 \\
F_{xi} + c_{12}F_{zi} &\geq 0 \\
-F_{xi} + c_{12}F_{zi} &\geq 0 \\
c_{13}F_{xi} + F_{yi} + c_{14}F_{zi} &\geq 0 \\
c_{13}F_{xi} - F_{yi} + c_{14}F_{zi} &\geq 0 \\
-c_{13}F_{xi} + F_{yi} + c_{14}F_{zi} &\geq 0 \quad \text{for } 1 \leq i \leq n_c \\
-c_{13}F_{xi} - F_{yi} + c_{14}F_{zi} &\geq 0 \\
c_{15}F_{xi} + F_{yi} + c_{16}F_{zi} &\geq 0 \\
c_{15}F_{xi} - F_{yi} + c_{16}F_{zi} &\geq 0 \\
-c_{15}F_{xi} + F_{yi} + c_{16}F_{zi} &\geq 0 \\
-c_{15}F_{xi} - F_{yi} + c_{16}F_{zi} &\geq 0
\end{aligned} \tag{4.20}$$

where,

$$\begin{aligned}
c_{12} &= 0.965926\mu \\
c_{13} &= 0.57735 \\
c_{14} &= 1.11536\mu \\
c_{15} &= 1.73205 \\
c_{16} &= 1.93185\mu
\end{aligned}$$

Note that the constants (c_1) to (c_{16}) in inequalities (4.17) to (4.20) are not exact values but approximated (rounded) values. However, in the algorithm the exact values are computed and used.

The efficiency of the approximation of the nonlinear friction law may be quantified via the linearization efficiency, η_ℓ defined by

$$\eta_\ell = \frac{\text{Volume of the approximating pyramid}}{\text{Volume of the actual friction cone}}$$

Figure 4.6 shows the linearization efficiency for various linearizations where n_f is the number of faces of the approximating pyramid.

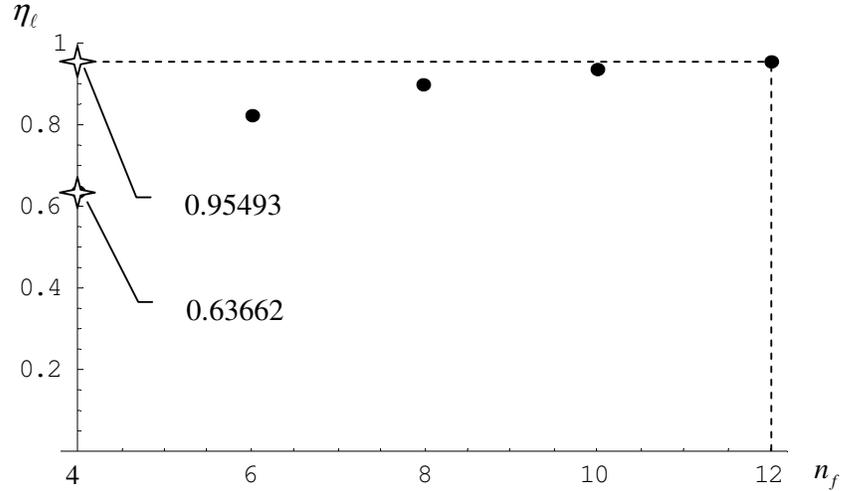


Figure 4.6 Efficiencies of various linearizations of the friction law.

4.3. LINEAR PROGRAMMING

In Section 4.1, it has been shown that the necessary and sufficient conditions for the body in Figure 4.1 to be in static equilibrium reduce to $2n_c$ inequalities (given by (4.5) and (4.6)) involving the $(3n_c - 6)$ independent force components obtained from the set $(F_{x1}, F_{x2}, \dots, F_{xn_c}; F_{y1}, F_{y2}, \dots, F_{yn_c}; F_{z1}, F_{z2}, \dots, F_{zn_c})$ and designated by the vector $\vec{f} = [f_1, f_2, \dots, f_{3n_c-6}]^T$. The n_c nonlinear equalities given by (4.5) may be replaced by n_i linear inequalities (see equations (4.16) to (4.20)). Here, n_i is given by $4n_c, 6n_c, 8n_c, 10n_c$ and $12n_c$ respectively if the friction cone is approximated by a pyramid with 4, 6, 8, 10 and 12 faces (see previous section). Therefore, one obtains $(n_i + n_c)$ linear inequalities to be satisfied by the $(3n_c - 6)$ components of \vec{f} . The solution of this problem may be conveniently formulated as a linear programming problem where a dummy linear objective function is to be minimized, with respect to \vec{f} , subject to the $(n_i + n_c)$ linear inequalities involving the components of \vec{f} . The body shown in Figure 4.1 will be potentially stable if and only if there exists a

solution to the aforementioned linear programming problem. In this study, the LinearProgramming command of MATHEMATICA has been used to solve such linear programming problems.

The execution time associated with the solution of the linear programming problem will indeed increase as the number of nodes (n_c) is increased. The number of faces of the approximating pyramid used to approximate the friction cone, n_f , is another factor that affects the aforementioned execution time. To illustrate the affects of these factors, consider the rigid body shown in Figure 4.1 and assume that there exists external loads \vec{F}_{ex} , \vec{M}_{ex} which lead to the ground reactions given by

$$\begin{aligned} F_{Rx} &= 0.2 \text{ N} \\ F_{Ry} &= 0.1 \text{ N} \\ F_{Rz} &= 5 \text{ N} \\ M_{Ox} &= 0.01 \text{ N.m} \\ M_{Oy} &= 0.01 \text{ N.m} \\ M_{Oz} &= 0.05 \text{ N.m} \end{aligned} \tag{4.21}$$

Let the coefficient of friction and the dimensions of the cube be

$$\begin{aligned} \mu &= 0.3 \\ d &= 0,06 \text{ m} \end{aligned}$$

Furthermore, let the distribution of the contact points be as shown in Figure 4.7. In this figure, it should be noted that the nodes other than the first four given by P_1, P_2, P_3, P_4 are at the middle of the squares formed by the equally spaced grid lines.

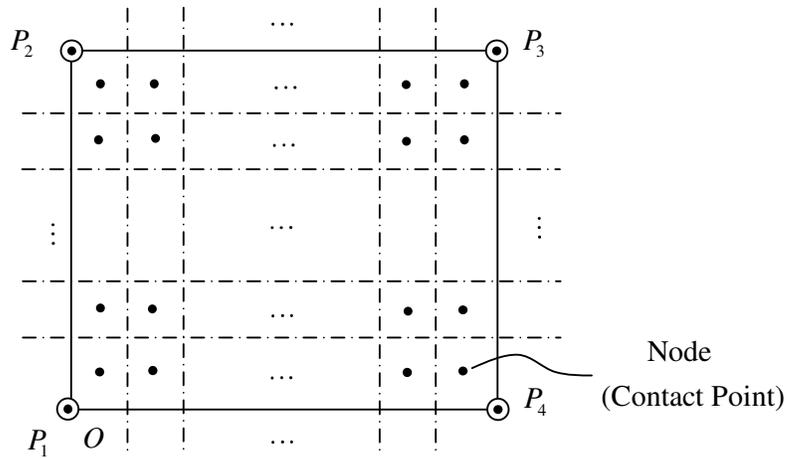


Figure 4.7 Distribution of contact points.

Given the number of nodes and the number of faces of the approximating pyramid, one can use the LinearProgramming command of MATHEMATICA to determine the potential stability of the body and also record the execution time. Execution times for various combinations of number of nodes and method of linearizations have been determined via the code developed in MATHEMATICA. The results are shown in Figures (4.8), (4.9) and (4.10) where the definitions of CPU time for LP, Total CPU time and Session time are given below.

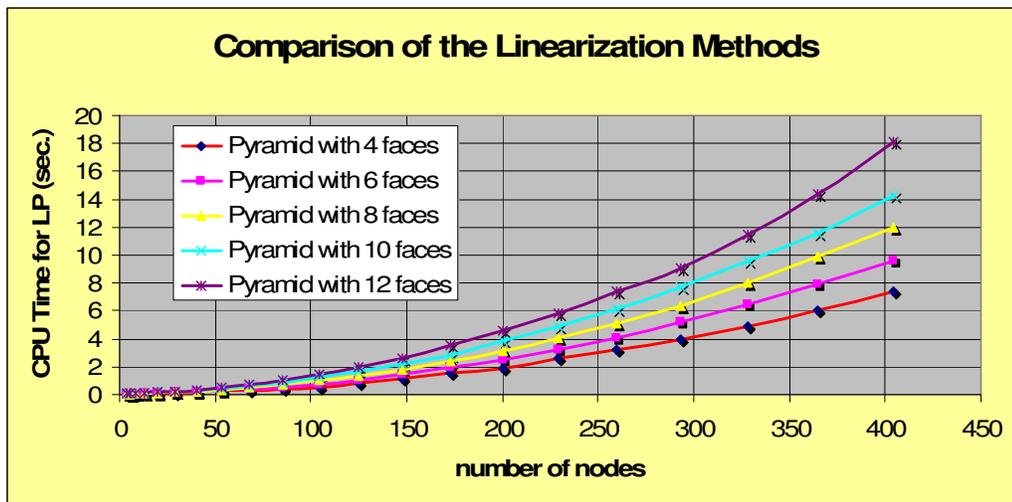


Figure 4.8 Comparison of the linearization methods in terms of CPU time for linear programming.

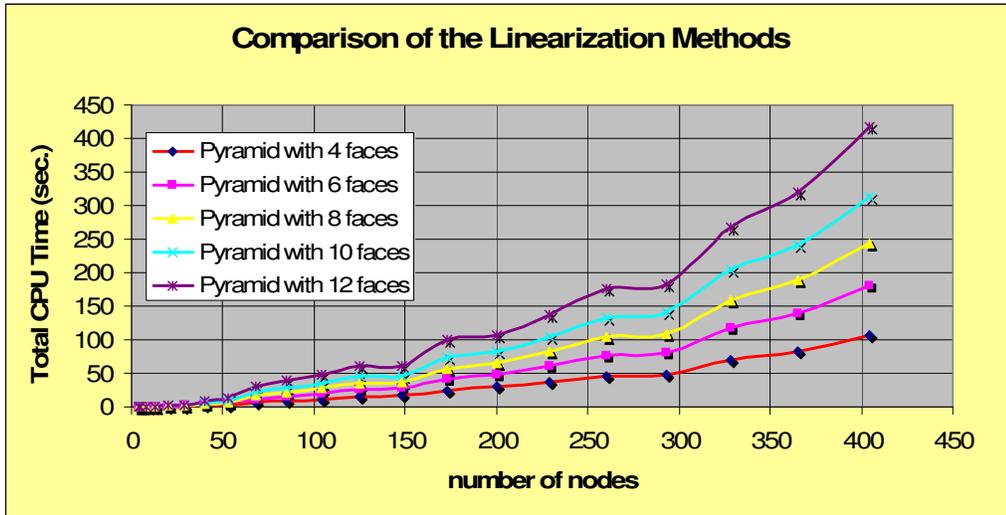


Figure 4.9 Comparison of the linearization methods in terms of Total CPU time.

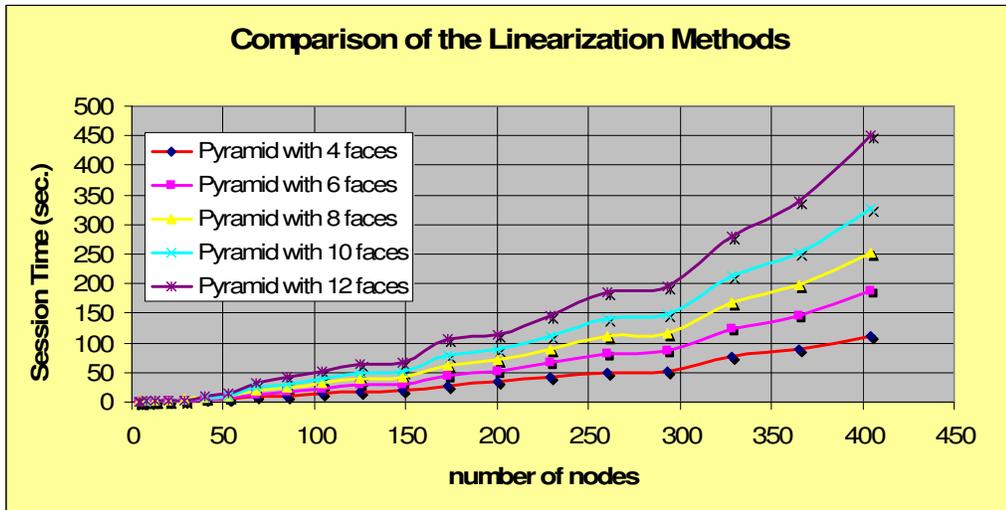


Figure 4.10 Comparison of the linearization methods in terms of Session time.

CPU Time for LP : The CPU time spent in *Mathematica* kernel in evaluation of the linear programming part (command) only.

Total CPU Time : The CPU time spent in *Mathematica* kernel in evaluation of the whole process.

Session Time : The total number of seconds of real time that have elapsed since the beginning of the *Mathematica* session.

As expected, the execution times increase as n_c and/or n_f is increased. A series of runs have also been performed to determine the “gains” associated with approximating pyramids with a large number of faces. To that purpose, the same cube (subject to the same loading leading to the ground reactions given by equation (4.21)) has been considered. The number of nodes has been fixed at 4, the nodes being given by P_1, P_2, P_3, P_4 as shown in Figure 4.7. Firstly, n_f is fixed at 4. Then, the value of μ , which is initially set to zero, is increased gradually until the body becomes potentially stable. That value of μ is denoted by μ_{\min} . Figure 4.11 shows the values of μ_{\min} for various linearization methods. As expected, μ_{\min} decreases as the n_f increases.

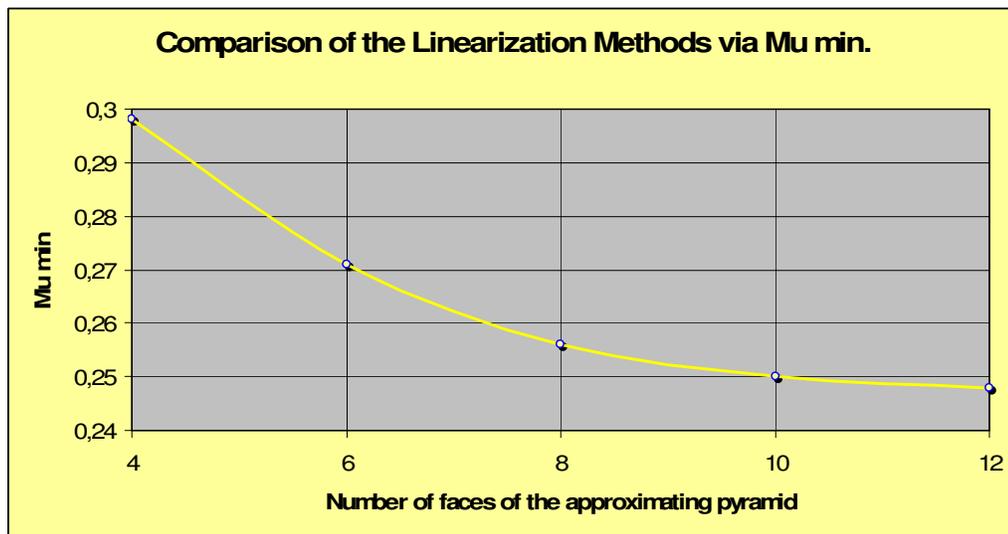


Figure 4.11 Comparison of the linearization methods in terms of μ_{\min} .

4.3. PERCENTAGE STABILITY

Consider the rigid body in Figure 4.1 and assume that $n_c \geq 4$. The set of F_{zi} forces which satisfy equations (4.8) and (4.6) is here defined to be the set of valid normal forces. To be more explicit, the set of valid normal forces satisfy the 3 equations given by

$$\begin{aligned} \sum_{i=1}^{n_c} F_{zi} &= F_{Rz} \\ \sum_{i=1}^{n_c} r_{yi} F_{zi} &= M_{Ox} \\ \sum_{i=1}^{n_c} -r_{xi} F_{zi} &= M_{Oy} \end{aligned} \quad (4.22)$$

and the n_c inequalities given by

$$F_{zi} \geq 0 \quad \text{for } i = 1, 2, \dots, n_c$$

Clearly, 3 of the F_{zi} 's can be eliminated using equation (4.22). Therefore, the number of independent F_{zi} 's will be given by

$$n_z \triangleq n_c - 3$$

Let, now, the vector of independent F_{zi} 's be designated by

$$\vec{f}_z = [f_1, f_2, \dots, f_{n_z}]^T$$

Therefore, the set of valid normal forces will be a "polygonal" region, VN , in the n_z dimensional \vec{f}_z -space (see Figure 4.12 where n_z is taken to be 2 for the ease of visualization).

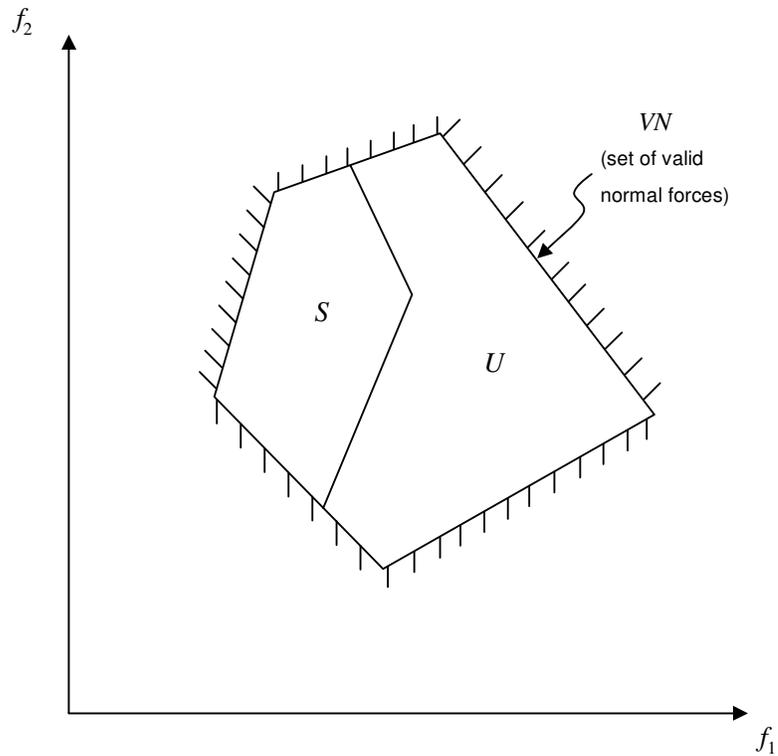


Figure 4.12 Potentially stable and unstable portions of the set of valid normal forces.

VN may be found by using the `InequalitySolve` command of `MATHEMATICA`. A point in the region VN will correspond to a potentially stable point if there exists a real $\vec{x}_{n_c,2}$ vector which satisfies equation (4.9) and the linearized friction inequalities corresponding to the selected point in VN . One may conveniently use Linear Programming to perform such a stability check for each point in VN leading to potentially stable and unstable regions in VN designated by S and U respectively such that

$$VN = S \cup U$$

Clearly, it is not possible to check every point in VN in practice. Therefore, gridlines, separated from each other by a distance of h_s , are used for each dimension of VN to generate a set of points to be tested for stability.

Now, percentage stability, PS , is defined as

$$PS \triangleq \frac{A_S}{A_{VN}} \times 100 \quad (4.23)$$

where,,

A_S : "Area" of region S .

A_{VN} : "Area" of region VN .

Clearly, 100 percent stability corresponds to guaranteed stability since the rigid body will be stable for all possible normal force distributions. As PS decreases, the "chances" of having a stable body decreases since some possible normal force distributions lead to instability of the body.

Percentage stability is not dependent upon the choice of the independent F_{zi} 's. This may be shown as follows. Let \vec{f}'_z be a different set of independent F_{zi} 's related to \vec{f}_z via the equation

$$\vec{f}'_z = [B] \vec{f}_z + k \quad (4.24)$$

where

$[B]$: $n_z \times n_z$ non singular coefficient matrix.

k : $n_z \times 1$ vector of constants.

Equation (4.24) is a linear equation since it is obtained based upon the linear equations given by equation (4.22). Under the linear transformation given by equation (4.24), the regions VN and S in the \vec{f}_z space will be mapped in a one-to-

one manner onto the regions VN' and S' in the \vec{f}'_z space respectively. Hence, PS will be defined, in the \vec{f}'_z space, to be

$$PS' \triangleq \frac{A'_S}{A'_{VN}} \times 100$$

Due to equation (4.24), one has

$$A'_S = |\det[B]| A_S$$

$$A'_{VN} = |\det[B]| A_{VN}$$

Hence, $PS = PS'$ indicating that percentage stability is not dependent upon the choice of independent F_{zi} 's.

CHAPTER 5

DYNAMIC STABILITY ANALYSIS OF I-CUBES

In this chapter, the motion of the I-Cubes is modeled by the fixed axis rotation of one body system (called the Active Body System) with respect to another body system which is considered to be fixed relative to the ground (called the Passive Body System). A probable motion to reach the final configuration, from the initial configuration, is assumed. By recursive kinematic relations and the Newton-Euler formulation, the resultant force and the accompanying resultant moment (to be applied by the ground on the I-Cubes so that the system is dynamically stable) is calculated.

5.1. ACTIVE AND PASSIVE BODY SYSTEMS

If it is assumed that one joint is activated at a time, any motion of the I-Cubes can be modeled by the fixed axis rotation of one body system (called the Active Body System) with respect to another body system which is considered to be fixed relative to the ground (called the Passive Body System). The definitions of the active and passive body systems (see Figure 5.1) are given next.

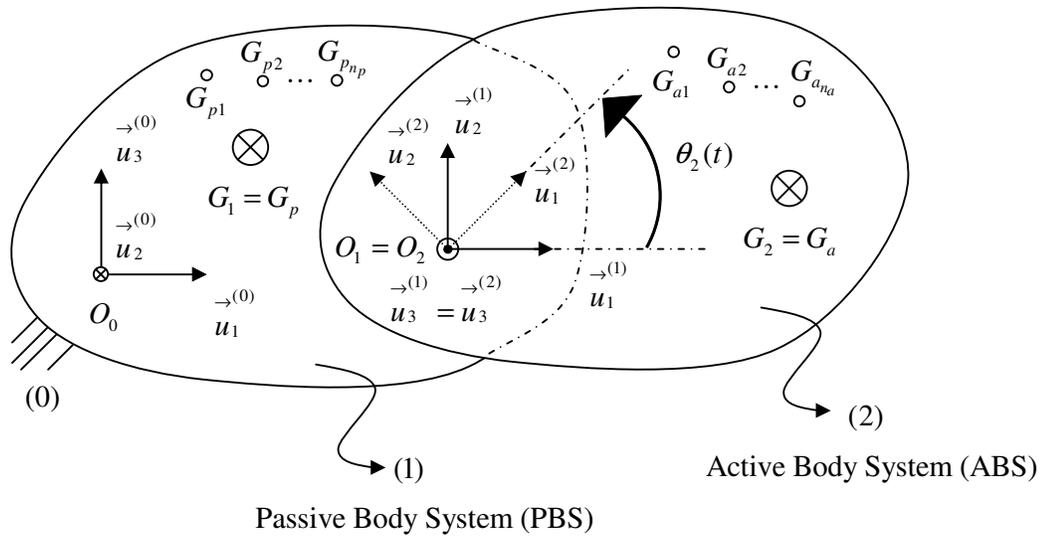


Figure 5.1 Active and passive body systems.

- Sub-links 1 and 2 : The two symmetrical parts of a link (see Figure 5.2)
- Body : Either a cube or a sub-link of the I-Cubes.
- Active Body System (ABS) : Rigid body system consisting of the moving bodies (which are considered to be rigidly attached to each other so that they form a single rigid body.)
- Passive Body System (PBS) : Rigid body system consisting of the non-moving bodies (which are considered to be rigidly attached to each other so that they form a single rigid body.)

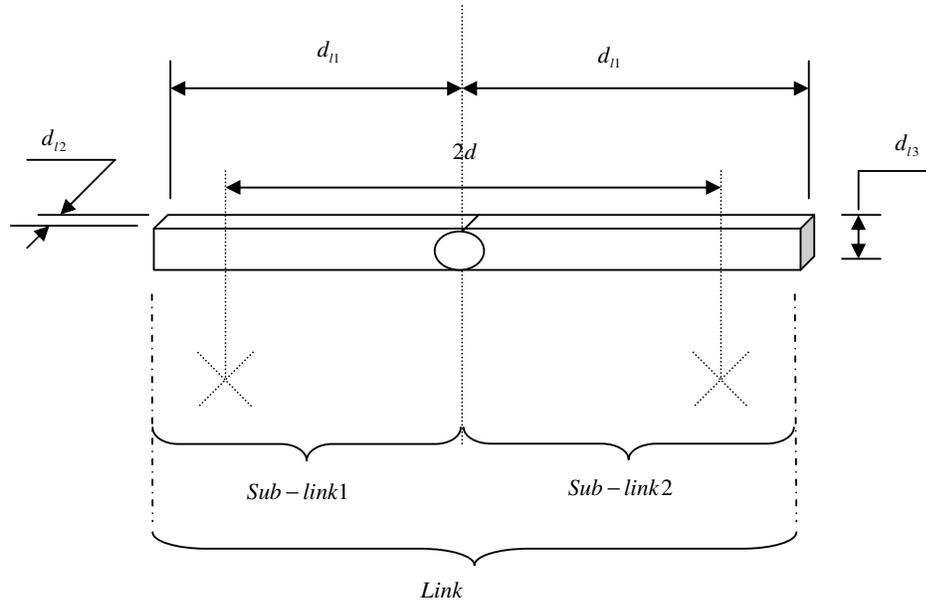


Figure 5.2 Sub-links of a link.

Nextly, the notation used in Figure 5.1 will be described.

- G_1 or G_p : Combined center of mass of the Passive Body System.
- G_2 or G_a : Combined center of mass of the Active Body System.
- n_a, n_p : Number of active and passive bodies, respectively.
- G_{pi} : Center of mass of the i 'th passive body where $i = 1, 2, \dots, n_p$.
- G_{ai} : Center of mass of the i 'th active body where $i = 1, 2, \dots, n_a$.
- $F\{O_0, u_1^{(0)}, u_2^{(0)}, u_3^{(0)}\}$ or $F_0(O_0)$: Inertial frame fixed to the ground with origin O_0 and with unit vectors $u_1^{(0)}, u_2^{(0)}, u_3^{(0)}$. O_0 is coincident with O in Chapter 4.
- $F\{O_1, u_1^{(1)}, u_2^{(1)}, u_3^{(1)}\}$ or $F_1(O_1)$: Body fixed reference frame (fixed to PBS) with origin O_1 and with unit vectors $u_1^{(1)}, u_2^{(1)}, u_3^{(1)}$.

$\vec{u}_3^{(1)}$: Unit vector coincident with the axes of rotation of the active body system.

$\vec{u}_1^{(1)}$: Unit vector perpendicular to $\vec{u}_3^{(1)}$ which may be conveniently selected to be parallel to $\vec{u}_1^{(0)}$.

$$\vec{u}_2^{(1)} = \vec{u}_3^{(1)} \times \vec{u}_1^{(1)}$$

$F\{\vec{O}_2, \vec{u}_1^{(2)}, \vec{u}_2^{(2)}, \vec{u}_3^{(2)}\}$ or $F_2(O_2)$: Body fixed reference frame (fixed to ABS) with origin O_2 (coincident with O_1) and with unit vectors $\vec{u}_1^{(2)}, \vec{u}_2^{(2)}, \vec{u}_3^{(2)}$.

where,

$\vec{u}_3^{(2)}$: Unit vector coincident with the axes of rotation of the active body system (coincident with $\vec{u}_3^{(1)}$).

$\theta_2(t)$: Angular parameter which indicates the relative rotation of the ABS with respect to the PBS. $\theta_2(t)$ is measured from $\vec{u}_1^{(1)}$ to $\vec{u}_1^{(2)}$ in a right hand sense around $\vec{u}_3^{(1)}$.

5.1.0. The Notation

This section is prepared to give information about the notation used (regarding vectors and matrices) in this Chapter (see [43], [44] for details of the notation).

- \vec{r} : A vector.
- \bar{r} : A (3x1) column matrix.
- \hat{R} : A matrix.
- r : A scalar.
- \vec{r}_{AK} : Position vector of point A with respect to point K.
- $F_a(A)$: Reference frame with unit vectors $\vec{u}_1^{(a)}, \vec{u}_2^{(a)}, \vec{u}_3^{(a)}$ and origin A .
- $\left\{ \vec{r} \right\}^{(a)} = \bar{r}^{-(a)}$: Column matrix representation of \vec{r} in $F_a(A)$.

where,

$$\vec{r} = \sum_{i=1}^3 r_i^{(a)} \vec{u}_i^{(a)}$$

$$\bar{r}^{-(a)} = \begin{bmatrix} r_1^{(a)} \\ r_2^{(a)} \\ r_3^{(a)} \end{bmatrix}$$

- $r_i^{(a)}$: i'th component of \vec{r} in $F_a(A)$.
- $\hat{C}^{(a,b)}$: The (3x3) transformation matrix relating $F_b(B)$ and $F_a(A)$

so that

$$\bar{r}^{-(a)} = \hat{C}^{(a,b)} \bar{r}^{-(b)}$$

where,

$$\widehat{C}^{(a,b)} = \left[\begin{array}{c|c|c} \overline{u_1}^{(b/a)} & \overline{u_2}^{(b/a)} & \overline{u_3}^{(b/a)} \end{array} \right]$$

$$\left\{ \overrightarrow{u_k}^{(b)} \right\}^{(a)} = \overline{u_k}^{(b/a)} \quad : \text{Column matrix representation of } \overrightarrow{u_k}^{(b)} \text{ in } F_a(A).$$

$\overline{u_1}, \overline{u_2}, \overline{u_3}$: The elementary or basic columns given by

$$\overline{u_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \overline{u_2} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \overline{u_3} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$D_a[\overrightarrow{r}] = \left(\frac{d \overrightarrow{r}}{dt} \right) \Big|_{\text{as if } F_a \text{ is fixed.}}$$

\widetilde{r} : The cross product matrix of \overline{r} given by

$$\widetilde{r} = \begin{bmatrix} 0 & -r_3 & r_2 \\ r_3 & 0 & -r_1 \\ -r_2 & r_1 & 0 \end{bmatrix}$$

where

$$\overline{r} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

5.1.1. D-H Convention

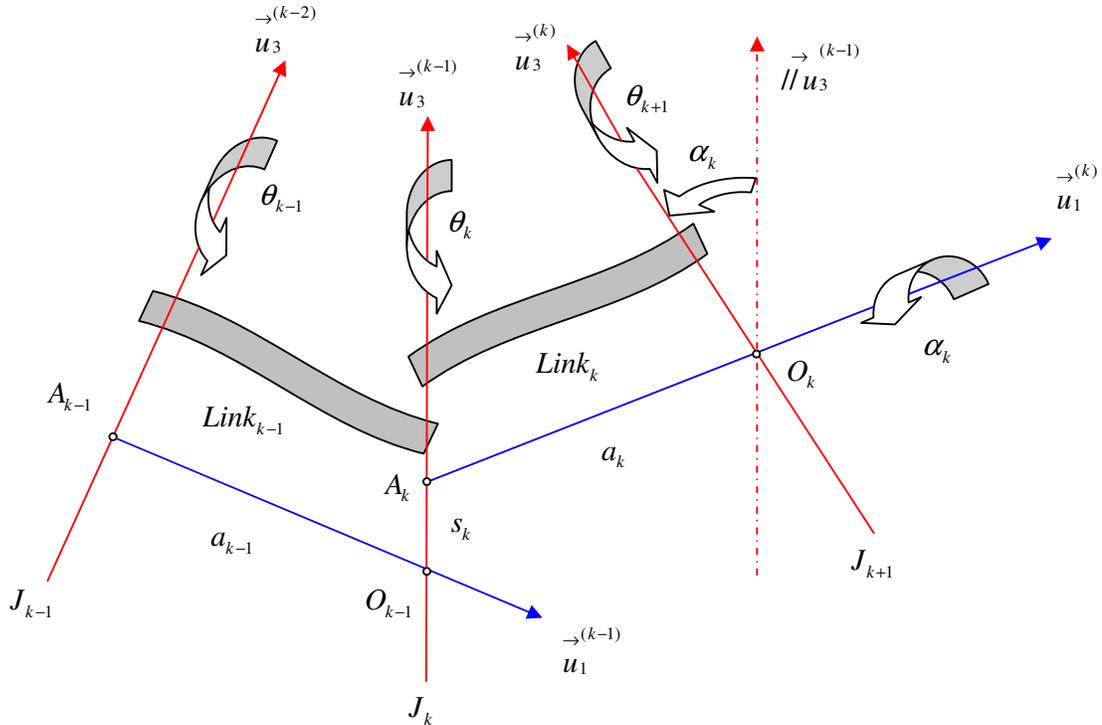


Figure 5.3 The Denavit – Hartenberg Representation [43].

To describe the translational and rotational relationships between adjacent links, Denavit and Hartenberg [45] proposed a method of systematically establishing a coordinate system (body-attached frame) to each link of an articulated chain (see Figure 5.3).

An orthonormal cartesian coordinate system $(u_1^{(k)}, u_2^{(k)}, u_3^{(k)})$ can be established for each link at its joint axis, where $k = 0, 1, 2, \dots, n$ and n is the number of links. Every coordinate frame is determined and established on the basis of the following rules [44] where larger index implies more distal link.

- $\vec{u}_3^{(k-1)}$ is a unit vector along the axes of rotation of the joint J_k between $Link_{k-1}$ and $Link_k$, its sense being arbitrary.
- $\vec{u}_1^{(k)}$ is a unit vector along the common normal between the axes of the joints J_k and J_{k+1} . If axes of J_k and J_{k+1} are not intersecting, then $\vec{u}_1^{(k)}$ is oriented from J_k to J_{k+1} . If axes of J_k and J_{k+1} are intersecting, then the orientation of $\vec{u}_1^{(k)}$ is arbitrary. However $\vec{u}_1^{(k)}$ must pass through link $(k+1)$.
- $\vec{u}_2^{(k)} = \vec{u}_3^{(k)} \times \vec{u}_1^{(k)}$
- $F_k(O_k)$ is fixed to $Link_k$ and its origin O_k is at the intersection of the axes $\vec{u}_1^{(k)}$ and $\vec{u}_3^{(k)}$.
- $a_k = A_k O_k$ along $\vec{u}_1^{(k)}$
= Effective length of link_k
- $s_k = O_{k-1} A_k$ along $\vec{u}_3^{(k-1)}$
= Translational distance of link_k wrt link_{k-1}
- θ_k is the joint angle measured from the $\vec{u}_1^{(k-1)}$ axis to the $\vec{u}_1^{(k)}$ axis about the $\vec{u}_3^{(k-1)}$ axis (using the right hand rule).
 $\theta_k = \sphericalangle[\vec{u}_1^{(k-1)} \rightarrow \vec{u}_1^{(k)} @ \vec{u}_3^{(k-1)}]$

- α_k is the offset angle measured from the $\vec{u}_3^{(k-1)}$ axis to the $\vec{u}_3^{(k)}$ axis about the $\vec{u}_1^{(k)}$ axis (using the right hand rule).

$$\alpha_k = \angle[\vec{u}_3^{(k-1)} \rightarrow \vec{u}_3^{(k)} @ \vec{u}_1^{(k)}]$$
- One of the parameters s_k or θ_k is constant and the other is the joint variable of J_k .

5.1.2. Inertial Parameters of the Active and Passive Body Systems

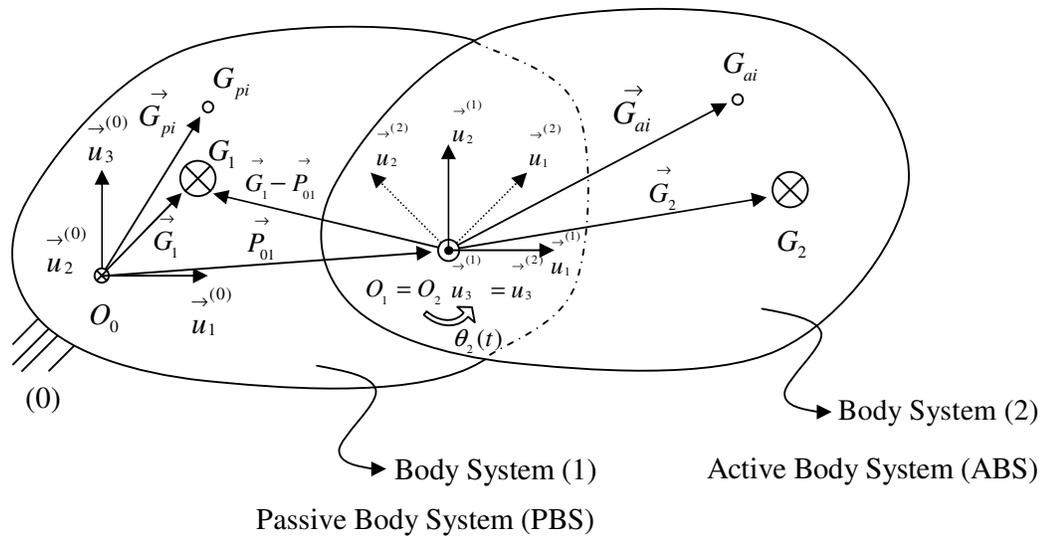


Figure 5.4 Inertial parameters of Active and Passive Body Systems.

Referring to Figure 5.4, consider the following notation.

$\vec{G}_1 = O_0 \vec{G}_1$: Position vector of the center of mass of the passive body system.

$\vec{G}_2 = O_2 \vec{G}_2$: Position vector of the center of mass of the active body system.

$\vec{G}_{pi} = O_0 \vec{G}_{pi}$: Position vector of the center of mass of the i 'th passive body.
 $(i = 1, 2, \dots, n_p)$

$\vec{G}_{ai} = O_2 \vec{G}_{ai}$: Position vector of the center of mass of the i 'th active body.
 $(i = 1, 2, \dots, n_a)$

$\vec{P}_{01} = O_0 \vec{O}_1$: Location of the origin of the passive body system.

$\vec{P}_{01}^{(0)}$: Coordinates of the origin of the passive body system in $F_0(0)$.

$$\theta_k = \angle [u_1^{(k-1)} \rightarrow u_1^{(k)} @ u_3^{(k-1)}]$$

$$\alpha_k = \angle [u_3^{(k-1)} \rightarrow u_3^{(k)} @ u_1^{(k)}]$$

$$k = 1, 2$$

Note that θ_2 is the joint variable and α_2 is zero.

The center of mass of the passive body system in $F_0(0_0)$ may be found from:

$$\vec{G}_1^{(0)} = \sum_{i=1}^{n_p} \left[\frac{(\vec{G}_{pi}^{(0)T} \cdot \vec{u}_1) m_{pi}}{m_p} \vec{u}_1 + \frac{(\vec{G}_{pi}^{(0)T} \cdot \vec{u}_2) m_{pi}}{m_p} \vec{u}_2 + \frac{(\vec{G}_{pi}^{(0)T} \cdot \vec{u}_3) m_{pi}}{m_p} \vec{u}_3 \right] \quad (5.1)$$

where,

$\overline{G}_1^{(0)}$: Coordinates of the center of mass of the passive body system in $F_0(O_0)$.

$\overline{G}_{pi}^{(0)}$: Coordinates of the center of mass of the i 'th passive body in $F_0(O_0)$.

$\overline{G}_{pi}^{(0)T} \cdot \overline{u}_1, \overline{G}_{pi}^{(0)T} \cdot \overline{u}_2, \overline{G}_{pi}^{(0)T} \cdot \overline{u}_3$: u_1, u_2, u_3 components of the center of mass of the i 'th passive body, respectively. ($i = 1, 2, \dots, n_p$)

m_{pi} : Mass of the i 'th passive body.

$m_p = \sum_{i=1}^{n_p} m_{pi}$: Total mass of the passive body system.

The center of mass of the active body system in $F_2(O_2)$ may be found from:

$$\overline{G}_2^{(2)} = \sum_{i=1}^{n_a} \left[\frac{(\overline{G}_{ai}^{(2)T} \cdot \overline{u}_1) m_{ai}}{m_a} \overline{u}_1 + \frac{(\overline{G}_{ai}^{(2)T} \cdot \overline{u}_2) m_{ai}}{m_a} \overline{u}_2 + \frac{(\overline{G}_{ai}^{(2)T} \cdot \overline{u}_3) m_{ai}}{m_a} \overline{u}_3 \right] \quad (5.2)$$

where,

$\overline{G}_2^{(2)}$: Coordinates of the center of mass of the active body system in $F_2(O_2)$.

$\overline{G}_{ai}^{(2)}$: Coordinates of the center of mass of the i 'th active body in $F_2(O_2)$.

$\overline{G}_{ai}^{(2)T} \cdot \overline{u}_1, \overline{G}_{ai}^{(2)T} \cdot \overline{u}_2, \overline{G}_{ai}^{(2)T} \cdot \overline{u}_3$: u_1, u_2, u_3 components of the center of mass of the i 'th active body, respectively. ($i = 1, 2, \dots, n_a$)

m_{ai} : Mass of the i 'th active body.

$m_a = \sum_{i=1}^{n_a} m_{ai}$: Total mass of the active body system.

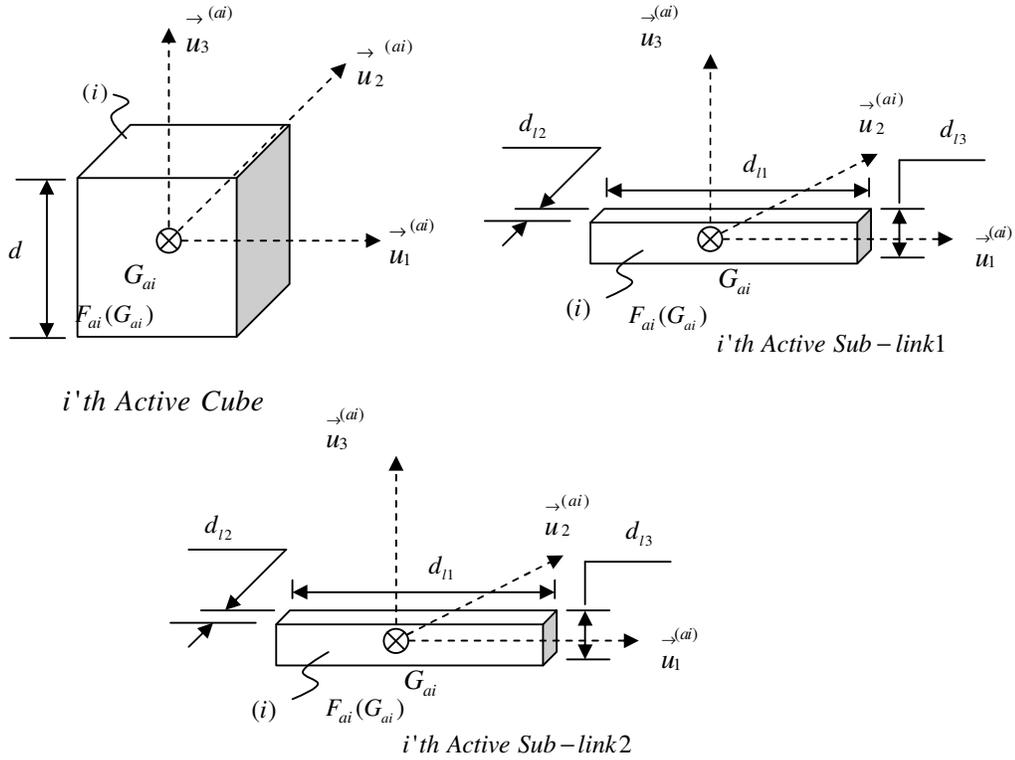


Figure 5.5 Active cubes and sub-links.

Cubes and sub-links are assumed to be homogeneous and their centers of masses are at their geometric centers (see Figure 5.5). The body fixed frames associated with the cubes and sub-links i.e., $F_{ai}(G_{ai})$ are also shown in Figure 5.5.

The dimensions of the cubes and the dimensions of the sub-links are [37]

$$d = 6 \text{ cm}$$

and

$$d_{11} = 8,5 \text{ cm}$$

$$d_{12} = 3,7 \text{ cm}$$

$$d_{13} = 1,8 \text{ cm}$$

The masses of the cubes and sub-links, on the other hand, are [37]

$$m_c = 205 \text{ gr}$$

$$m_l = m_{l1} = m_{l2} = 97,5 \text{ gr}$$

where m_c and m_l designate the mass of a cube and a sub-link respectively.

The Centroidal Inertia Dyadic of the i 'th active cube expressed in the body fixed frame $F_{ai}(G_{ai})$ is given by

$$\hat{J}_{ci}^{(ai)} = \begin{bmatrix} J_c & 0 & 0 \\ 0 & J_c & 0 \\ 0 & 0 & J_c \end{bmatrix} \quad (5.3)$$

where

$$J_c = \frac{1}{2} m_c (2d^2)$$

The Centroidal Inertia Dyadic of the i 'th active sub-link expressed in the body fixed frame $F_{ai}(G_{ai})$, on the other hand, is given by

$$\hat{J}_{li}^{(ai)} = \begin{bmatrix} J_{l1} & 0 & 0 \\ 0 & J_{l2} & 0 \\ 0 & 0 & J_{l3} \end{bmatrix} \quad (5.4)$$

where

$$J_{l1} = \frac{1}{2} m_l (d_{l2}^2 + d_{l3}^2)$$

$$J_{l2} = \frac{1}{2} m_l (d_{l1}^2 + d_{l3}^2)$$

$$J_{l3} = \frac{1}{2} m_l (d_{l1}^2 + d_{l2}^2)$$

Using the parallel axis theorem, one can determine the centroidal inertia dyadic of the ABS expressed in $F_2(G_2)$ to get

$$\hat{I}_2^{(2)} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{zx} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{yz} & I_{zz} \end{bmatrix} \quad (5.5)$$

where,

$$\begin{aligned} I_{xx} &= \sum_{i=1}^{n_a} [(J_{xxi}) + m_{ai} ((\bar{G}_{aai}^{(2)T} \bar{\mu}_2)^2 + (\bar{G}_{aai}^{(2)T} \bar{\mu}_3)^2)] \\ I_{yy} &= \sum_{i=1}^{n_a} [(J_{yyi}) + m_{ai} ((\bar{G}_{aai}^{(2)T} \bar{\mu}_1)^2 + (\bar{G}_{aai}^{(2)T} \bar{\mu}_3)^2)] \\ I_{zz} &= \sum_{i=1}^{n_a} [(J_{zzi}) + m_{ai} ((\bar{G}_{aai}^{(2)T} \bar{\mu}_1)^2 + (\bar{G}_{aai}^{(2)T} \bar{\mu}_2)^2)] \end{aligned} \quad (5.6)$$

$$\begin{aligned} I_{xy} &= \sum_{i=1}^{n_a} [m_{ai} ((\bar{G}_{aai}^{(2)T} \bar{\mu}_1)^2 + (\bar{G}_{aai}^{(2)T} \bar{\mu}_2)^2)] \\ I_{yz} &= \sum_{i=1}^{n_a} [m_{ai} ((\bar{G}_{aai}^{(2)T} \bar{\mu}_2)^2 + (\bar{G}_{aai}^{(2)T} \bar{\mu}_3)^2)] \\ I_{zx} &= \sum_{i=1}^{n_a} [m_{ai} ((\bar{G}_{aai}^{(2)T} \bar{\mu}_3)^2 + (\bar{G}_{aai}^{(2)T} \bar{\mu}_1)^2)] \end{aligned} \quad (5.7)$$

with

$$\hat{J}_i^{(2)} = \hat{\Gamma}^{(ai)} \hat{J}_i^{(ai)} \hat{\Gamma}^{(ai)T} = \begin{bmatrix} J_{xxi} & 0 & 0 \\ 0 & J_{yyi} & 0 \\ 0 & 0 & J_{zzi} \end{bmatrix} \quad (5.8)$$

where

$\hat{J}_i^{(2)}$: Centroidal Inertia Dyadic of the i'th active body with respect to the active body system's body fixed frame $F_2(G_2)$, ($i = 1, 2, \dots, n_a$). Note that $\hat{J}_i^{(2)}$ is always diagonal because of the assumption that I-Cubes can only make 90° motions.

$\hat{\Gamma}^{(ai)} = \hat{C}^{(2,ai)}$: Transformation matrix of the i'th active body with respect to the active body system's body fixed frame ($F_2(G_2)$).

$\hat{J}_i^{(ai)}$: Centroidal Inertia Dyadic of the i'th active body (cube or sub-link) with respect to the active body's body fixed frame $F_{ai}(G_{ai})$.

and

$$\overline{G}_{aai}^{(2)} = \overline{G}_{ai}^{(2)} - \overline{G}_2^{(2)} \tag{5.9}$$

where

$\overline{G}_{aai}^{(2)}$: The position vector of $G_a \vec{G}_{ai}$ in $F_2(G_2)$.

5.2. RECURSIVE KINEMATIC RELATIONS FOR THE ACTIVE AND PASSIVE BODY SYSTEMS

Note that this section is inspired from [43].

5.2.1. Transformation Matrices

$$\hat{\Phi}_{k+1} = \hat{\Phi}_k \cdot \hat{C}^{(k,k+1)} \quad \text{for } k = 0, 1. \quad (5.10)$$

where,

$\hat{\Phi}_k = \hat{C}^{(0,k)}$: Transformation matrix of the k' th body with respect to the base frame $F_0(O_0)$.

$$\hat{C}^{(k,k+1)} = \begin{bmatrix} \cos(\theta_{k+1}) & -\sin(\theta_{k+1}) & 0 \\ \sin(\theta_{k+1}) & \cos(\theta_{k+1}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha_{k+1}) & -\sin(\alpha_{k+1}) \\ 0 & \sin(\alpha_{k+1}) & \cos(\alpha_{k+1}) \end{bmatrix}$$

and

$$\hat{\Phi}_0 = \hat{I} = (3 \times 3) \text{ Identity Matrix.}$$

5.2.2. Angular Velocities

$\vec{\omega}_k = \vec{\omega}_{k/0}$: Angular velocity of the k'th body with respect to the base frame $F_0(O_0)$.

5.2.2.1. Vector Expression

$$\vec{\omega}_k = \vec{\omega}_{k-1} + \dot{\theta}_k \vec{u}_3^{(k-1)} \quad \text{for } k = 1, 2. \quad (5.11)$$

where

$$\vec{\omega}_0 = \vec{0}$$

5.2.2.2. Matrix Expression in $F_0(O_0)$

$$\bar{\omega}_k^{(0)} = \bar{\omega}_{k-1}^{(0)} + \dot{\theta}_k \hat{\Phi}_{k-1} \cdot \bar{u}_3 \quad \text{for } k = 1, 2. \quad (5.12)$$

5.2.2.3. Matrix Expression in the k' th Body System Frame

$$\bar{\omega}_k^{(k)} = \hat{\Phi}_k^T \cdot \bar{\omega}_k^{(0)} \quad \text{for } k = 1, 2. \quad (5.13)$$

where

$$\bar{\omega}_0^{(0)} = \vec{0} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

5.2.3. Angular Accelerations

$\vec{\alpha}_k = \vec{\alpha}_{k/0} = D_0 \vec{\omega}_k$: Angular acceleration of the k'th body system with respect to the base frame $F_0(O_0)$.

5.2.3.1. Matrix Expression in $F_0(O_0)$

$$\bar{\alpha}_k^{(0)} = D_0 [\bar{\omega}_k^{(0)}] \quad \text{for } k = 1, 2. \quad (5.14)$$

where

$$\bar{\alpha}_0^{(0)} = \vec{0}$$

5.2.3.2. Matrix Expression in the k' th Body System Frame

$$\bar{\alpha}_k^{(k)} = \hat{\Phi}_k^T \bar{\alpha}_k^{(0)} \quad \text{for } k = 1, 2. \quad (5.15)$$

5.2.4. Location of the Body System Origins

$$\vec{P}_{Ok} = \vec{O}_0 O_k.$$

5.2.4.1. Vector Expression

$$\vec{P}_{Ok} = \vec{P}_{O1} \quad \text{for } k = 1, 2. \quad (5.16)$$

5.2.4.2. Matrix Expression in $F_0(O_0)$

$$\bar{P}_{Ok}^{(0)} = \bar{P}_{O1}^{(0)} \quad \text{for } k = 1, 2. \quad (5.17)$$

5.2.5. Locations of the Mass Centers

Referring to Figure 5.6, define

$$\vec{P}_k = \vec{O}_0 G_k$$

5.2.5.1. Vector Expression

$$\vec{P}_k = \vec{P}_{O(k-1)} + \vec{G}_k \quad \text{for } k = 1, 2. \quad (5.18)$$

where

$$\vec{P}_{00} = \vec{0}$$

5.2.5.2. Matrix Expression in $F_0(O_0)$

$$\begin{aligned}\bar{P}_1^{(0)} &= \bar{G}_1^{(0)} \\ \bar{P}_2^{(0)} &= \bar{P}_{O1}^{(0)} + \hat{\Phi}_2 \cdot \bar{G}_2^{(2)}\end{aligned}\tag{5.19}$$

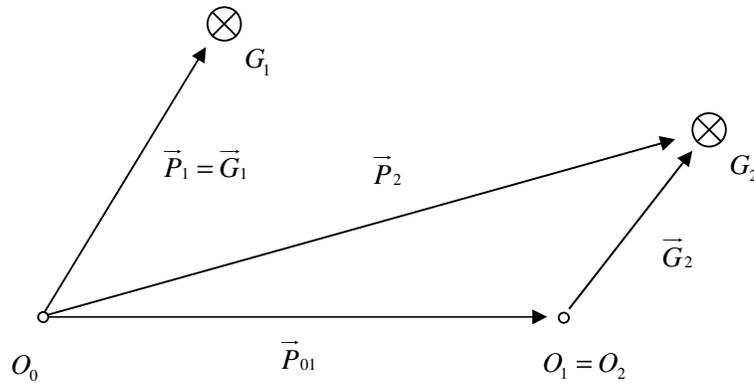


Figure 5.6 Locations of the mass centers.

5.2.6.1. Body System Origin Velocities in $F_0(O_0)$

$\vec{V}_{OK} = D_0 \vec{P}_{OK}$: Velocity vector of the origin of the k'th body system with respect to the inertial frame $F_0(O_0)$.

$$\bar{V}_{Ok}^{(0)} = D[\bar{P}_{Ok}^{(0)}] \quad \text{for } k = 1, 2.\tag{5.20}$$

5.2.6.2. Mass Center Velocities in $F_0(O_0)$

$\vec{V}_K = D_0 \vec{P}_K$: Velocity vector of the mass center of the k'th body system with respect to the inertial frame $F_0(O_0)$.

$$\bar{V}_k^{(0)} = D[\bar{P}_k^{(0)}] \quad \text{for } k = 1, 2. \quad (5.21)$$

5.2.7.1. Body System Origin Accelerations in $F_0(O_0)$

$\vec{a}_{OK} = D_0^2 \vec{P}_{OK}$: Acceleration vector of the origin of the k'th body system with respect to the inertial frame $F_0(O_0)$.

$$\bar{a}_{ok}^{(0)} = D[\bar{V}_{ok}^{(0)}] \quad \text{for } k = 1, 2. \quad (5.22)$$

5.2.7.2. Mass Center Accelerations in $F_0(O_0)$

$\vec{a}_K = D_0^2 \vec{P}_K$: Acceleration vector of the mass center of the k'th body system with respect to the inertial frame $F_0(O_0)$.

$$\bar{a}_k^{(0)} = D[\bar{V}_k^{(0)}] \quad \text{for } k = 1, 2. \quad (5.23)$$

5.3. RECURSIVE NEWTON-EULER FORMULATION FOR THE INVERSE DYNAMIC ANALYSIS OF THE ACTIVE AND PASSIVE BODY SYSTEMS

Suppose that the motion of the active body system is specified as $\theta_2(t)$ and it is desired to determine the forces, and moments, applied by the ground (body (0)) on the passive body system (body (1)). Consider the free body diagram of the k'th body system shown in Figure 5.7 and the following notation [43].

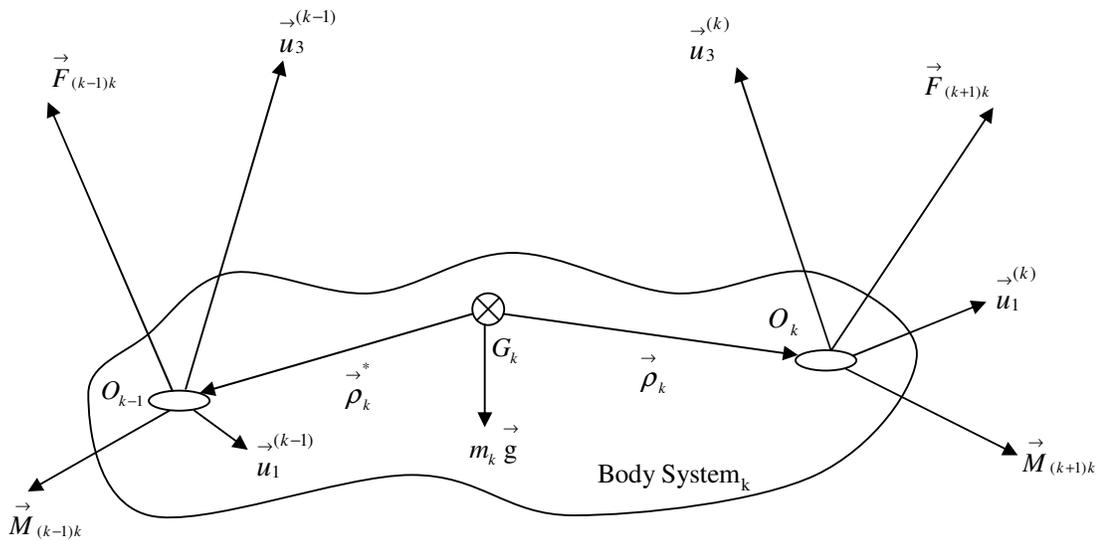


Figure 5.7 FBD of the k'th body system [43].

\vec{F}_{ij} : Force applied by Body System_i on Body System_j.

\vec{M}_{ij} : Moment applied by Body System_i on Body System_j.

$\vec{a}_k = \vec{a}_{G_k}$: Acceleration of the mass center.

$\vec{\rho}_k = \vec{G}_k \vec{O}_k$: Moment arm of $\vec{F}_{(k+1)k}$.

$\vec{\rho}_k^* = \vec{G}_k \vec{O}_{k-1}$: Moment arm of $\vec{F}_{(k-1)k}$.

5.3.1. Force Equations (Newton's Equations)

$$\sum \vec{F} = m_k \vec{a}_k \quad (5.24)$$

5.3.1.1. Vector Expression

$$\vec{F}_{(k-1)k} = \vec{F}_{k(k+1)} + m_k (\vec{a}_k - \vec{g}) \quad (5.25)$$

5.3.1.2. Matrix Expression in k' th Body System Frame

$$\vec{F}_{(k-1)k} = \overset{(k)}{C} \overset{(k,k+1)}{\wedge} \vec{F}_{k(k+1)} + m_k (\vec{a}_k - \vec{g}) \quad \text{for } k = 1, 2 \quad (5.26)$$

where

$$\vec{g} = -\overset{(k)}{\Phi}_k^T \vec{g} u_3 \quad (5.27)$$

g = Being the magnitude of the gravitational acceleration.

5.3.2. Moment Equations (Euler's Equations)

$$(\sum \vec{M})_{G_k} = \overset{\vee}{I}_k \cdot \vec{\alpha}_k + \vec{\omega}_k \times \overset{\vee}{I}_k \cdot \vec{\omega}_k \quad (5.28)$$

where

$\overset{\vee}{I}_k$: Centroidal Inertia Dyadic.

$\vec{\alpha}_k$: Angular acceleration.

$\vec{\omega}_k$: Angular velocity.

5.3.2.1. Vector Expression

$$\begin{aligned} \vec{M}_{(k-1)k} &= \vec{M}_{k(k+1)} - \vec{\rho}_k^* \times \vec{F}_{(k-1)k} + \vec{\rho}_k \times \vec{F}_{k(k+1)} \\ &+ \hat{I}_k \cdot \vec{\alpha}_k + \vec{\omega}_k \times \hat{I}_k \cdot \vec{\omega}_k \end{aligned} \quad (5.29)$$

5.3.2.2. Matrix Expression in k' th Body System Frame

$$\begin{aligned} \bar{M}_{(k-1)k} &= \hat{C}^{(k,k+1)} \bar{M}_{k(k+1)} - \bar{\rho}_k^* \bar{F}_{(k-1)k} \quad \text{for } k=1, 2. \\ &+ \bar{\rho}_k \hat{C}^{(k,k+1)} \bar{F}_{k(k+1)} + \hat{I}_k \bar{\alpha}_k + \bar{\omega}_k \hat{I}_k \bar{\omega}_k \end{aligned} \quad (5.30)$$

where,

$$\begin{aligned} \bar{\rho}_1^{(1)} &= -\hat{\Phi}_1^t \cdot [\bar{G}_1^{(0)} - \bar{P}_{01}^{(0)}] \\ \bar{\rho}_2^{(2)} &= \bar{0} \\ \bar{\rho}_1^{*(1)} &= -\hat{\Phi}_1^t \cdot [\bar{G}_1^{(0)}] \\ \bar{\rho}_2^{*(2)} &= -[\bar{G}_2^{(2)}] \end{aligned} \quad (5.31)$$

Note that $\hat{I}_1^{(1)}$ is a dummy inertia tensor and its components could be anything, because Body System₁ is a non-moving body system.

5.4. PROBABLE MOTION OF THE ACTIVE BODY SYSTEM

The motion of the active body system is specified via $\theta_2(t)$. A probable candidate for $\theta_2(t)$ is given by the fifth order polynomial

$$\theta_2(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \quad (5.32)$$

subjected to the boundary conditions

$$\theta_2(0) = q_0$$

$$\theta_2(t_f) = q_f$$

$$\dot{\theta}_2(0) = 0$$

$$\ddot{\theta}_2(0) = 0$$

$$\dot{\theta}_2(t_f) = 0$$

$$\ddot{\theta}_2(t_f) = 0$$

which is shown in Figure 5.8.

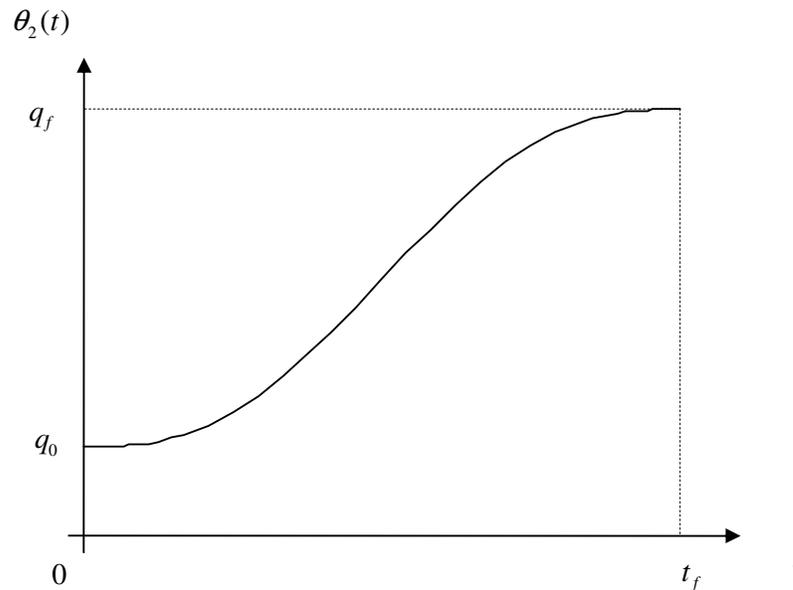


Figure 5.8 Fifth order time - motion polynomial.

Clearly, the 6 coefficients a_0 to a_5 may be determined using the 6 boundary conditions provided that the final time, t_f , and the initial and final positions, q_0 and q_f , are given.

5.5. DISCRETIZATION OF THE MOTION

In order to check the stability of the system throughout the motion given by $\theta_2(t)$, the elapsed time may be divided into N_d intervals (see Figure 5.9) given by

$$N_d = \frac{t_f}{h} \quad (5.33)$$

where h is the step size. Thus the stability of the system may be checked at the discrete times $t = 0, h, 2h, \dots, t_f$.

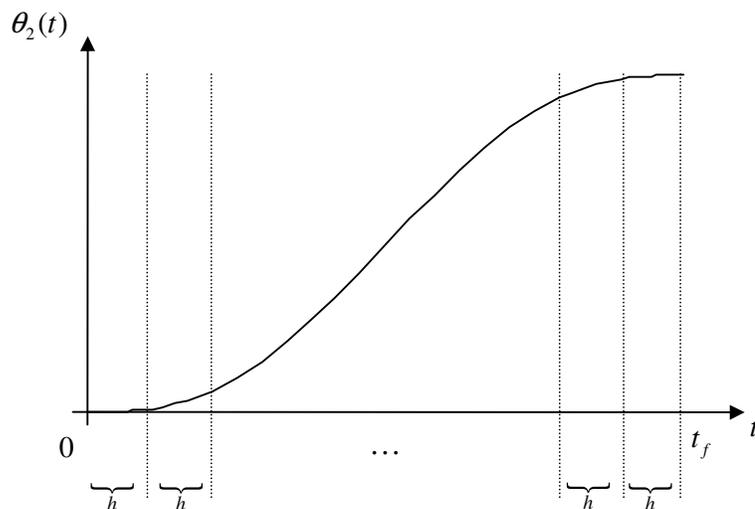


Figure 5.9 Discretization of the time - motion polynomial.

5.6. CONTACT POINT SELECTION

To investigate the stability of an assembly, one should decide upon the locations of the contact points. It appears that a good choice is the vertices of the convex hull. The convex hull of a set of points S is the smallest convex polygon that contains every point in S , where a simple polygon P is a convex polygon if, for any points p and q inside P , the line segment pq lies entirely inside P . One can roughly state that, the convex hull is defined to be the boundary formed by the outermost points. Those points are simply the corners of the cubes of the passive body system that lie on the ground. However, one may select any point on the contact region as a contact point rather than the vertices of the convex hull.

Figure 5.10 (b) shows the “convex hull” of the base points that form the outer boundary of the cubes that do not move, the assumed contact points (P_1 to P_5) and O_0 (which is arbitrarily selected on one of the edges of the convex hull) where, Figure 5.10 (a) shows the initial and final configurations of the system.

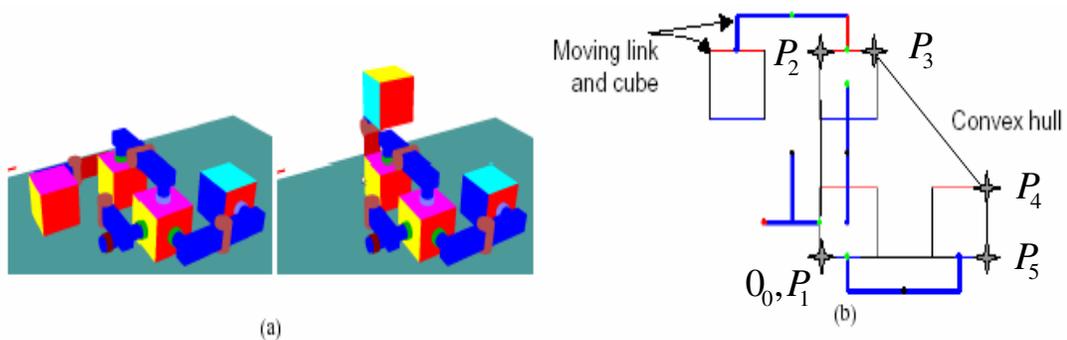


Figure 5.10 Given a 90° link motion with initial and final configurations (a), convex hull and the contact points (b) [37].

CHAPTER 6

CASE STUDIES

In this chapter, the developed algorithm is illustrated using the modular robotic system, I-Cubes, via various case studies. Some of the cases are also solved by the simulation program, ADAMS. The solutions obtained via ADAMS and the developed stability algorithm are consistent.

The algorithm used for the stability analysis of a given motion of the I-Cubes is shown via the flowchart in Figure 6.1. The inputs of the flowchart are designated by $I-1$ through $I-5$ which are described below.

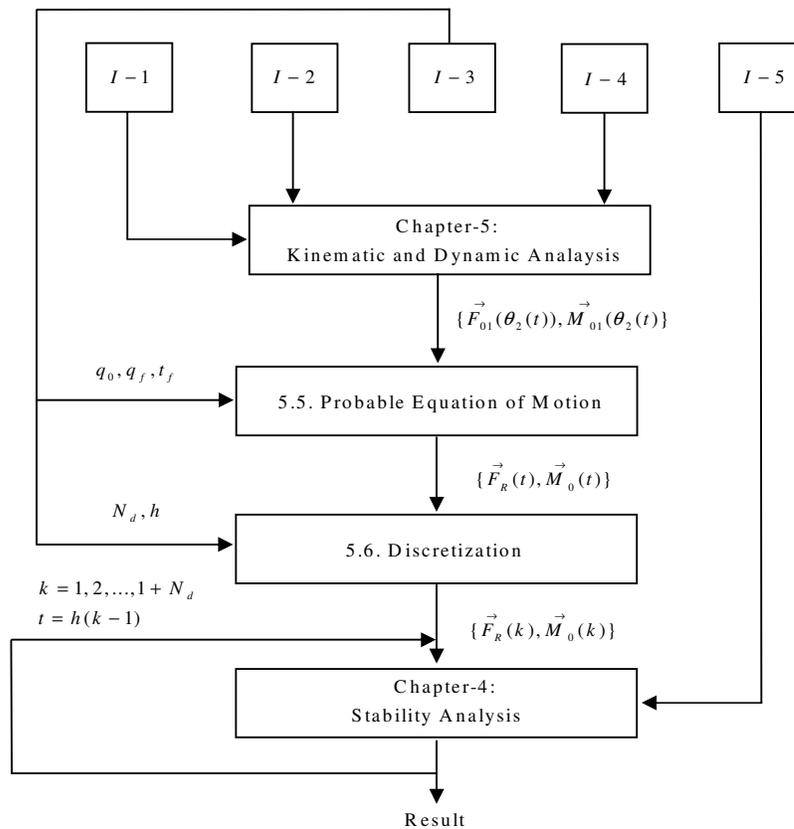


Figure 6.1 Flow chart of the stability analysis.

I – 1 (D-H Parameters) :

- α_k
- θ_k

where

$k = 1, 2.$

I – 2 :

- d
- n_a, n_p
- m_c, m_l
- d_{l1}, d_{l2}, d_{l3}
- μ

I – 3 (Discretization Inputs for $\theta_2(t)$) :

- h
- N_d
- t_f
- q_0, q_f

I – 4 :

- $\overline{G}_{pi}^{(0)}, i = 1, \dots, n_p$
- $\overline{G}_{ai}^{(2)}, i = 1, \dots, n_a$
- $\overline{P}_{01}^{(0)}$
- $\hat{\Gamma}^{(ai)}, i = 1, \dots, n_a$
- $m_{pi}, i = 1, \dots, n_p$
- $m_{ai}, i = 1, \dots, n_a$

I – 5 (Stability Analysis Inputs) :

- n_c
- h_s
- $r_i, i = 1, \dots, n_c$

6.3. CASE 1

Consider the single cube shown in Figure 6.2 where the external load is given by

$$\vec{F}_{ex} = \vec{0}$$

$$\vec{M}_{ex} = \vec{\tau} = 0.01 \vec{z}_0 \text{ N.m}$$

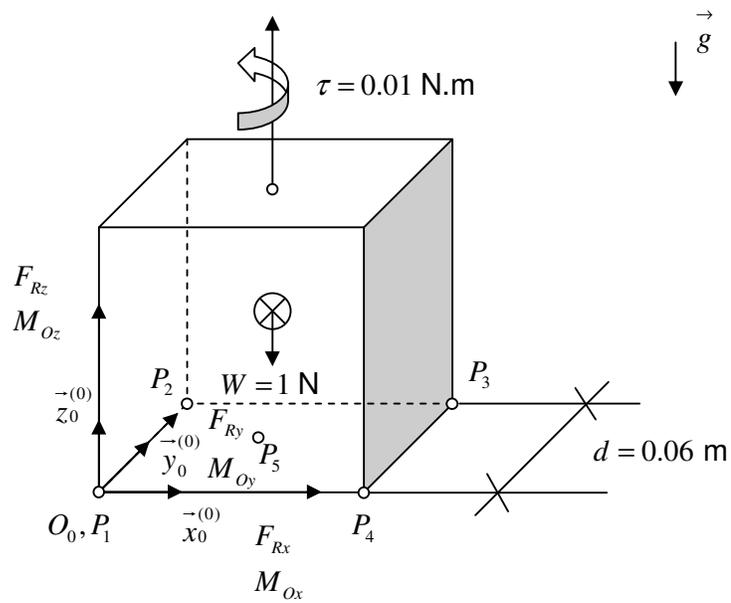


Figure 6.2 Case 1: Static Indeterminacy with Search Method.

The required inputs are given below

$$\mu = 0.3$$

$$d = 0.06 \text{ m}$$

$$h_s = 0.1 \text{ m}$$

$$n_c = 5$$

$$W = m_c g = 1 \text{ N}$$

$$\bar{r}_1^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{r}_2^{(0)} = \begin{bmatrix} 0 \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_3^{(0)} = \begin{bmatrix} d \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_4^{(0)} = \begin{bmatrix} d \\ 0 \\ 0 \end{bmatrix}, \quad \bar{r}_5^{(0)} = \begin{bmatrix} d/2 \\ d/2 \\ 0 \end{bmatrix}$$

The resultant loads to be applied by the ground on the cube are found from equation (4.1) to be

$$F_{Rx} = 0 \text{ N}$$

$$F_{Ry} = 0 \text{ N}$$

$$F_{Rz} = 1 \text{ N}$$

$$M_{Ox} = 0.03 \text{ N.m}$$

$$M_{Oy} = -0.03 \text{ N.m}$$

$$M_{Oz} = -0.01 \text{ N.m}$$

Equation (4.4) lead to the following equations

$$F_{Rx} = F_{x1} + F_{x2} + F_{x3} + F_{x4} + F_{x5}$$

$$F_{Ry} = F_{y1} + F_{y2} + F_{y3} + F_{y4} + F_{y5}$$

$$F_{Rz} = F_{z1} + F_{z2} + F_{z3} + F_{z4} + F_{z5}$$

$$M_{Ox} = d.F_{z2} + d.F_{z3} + d/2.F_{z5}$$

$$M_{Oy} = -d.F_{z3} - d.F_{z4} - d/2.F_{z5}$$

$$M_{Oz} = -d.F_{x2} - d.F_{x3} - d/2.F_{x5} + d.F_{y3} + d.F_{y4} + d/2.F_{y5}$$

Solving F_{z1} , F_{z2} and F_{z3} (in terms of F_{z4} , F_{z5}) from equation (4.22), one obtains

$$F_{z1} = 0.5 - F_{z4} - F_{z5} / 2$$

$$F_{z2} = F_{z4}$$

$$F_{z3} = 0.5 - F_{z4} - F_{z5} / 2$$

Therefore the inequalities given by (4.6) reduce to

$$0 \leq F_{z4} \leq 0.5$$

$$0 \leq F_{z5} \leq 1 - 2F_{z4}$$

which define the region VN in the $F_{z4} - F_{z5}$ space. Solving F_{x1} , F_{y1} and F_{x2} from equation (4.13), on the other hand, one obtains

$$F_{x1} = -0.166667 - F_{x4} - 0.5F_{x5} - F_{y3} - F_{y4} - 0.5F_{y5}$$

$$F_{y1} = -F_{y2} - F_{y3} - F_{y4} - F_{y5}$$

$$F_{x2} = 0.166667 - F_{x3} - 0.5F_{x5} + F_{y3} + F_{y4} + 0.5F_{y5}$$

The friction cone is approximated by a pyramid with 4 faces (see equation (4.16)). Furthermore, the region VN is discretized using $h_s = 0.1$ m leading to 36 points in VN . Using linear programming, the stability of each of the 36 points has been determined. The results given in Figure 6.3 and Table 6.1 indicate that 16 of the 36 points lead to stable configurations. Therefore the percentage stability may be approximated as

$$PS \cong \frac{16}{36} \times 100 = 44.4\%$$

which implies that the given configuration is potentially stable. Recall that a nonzero PS value implies a potentially stable configuration.

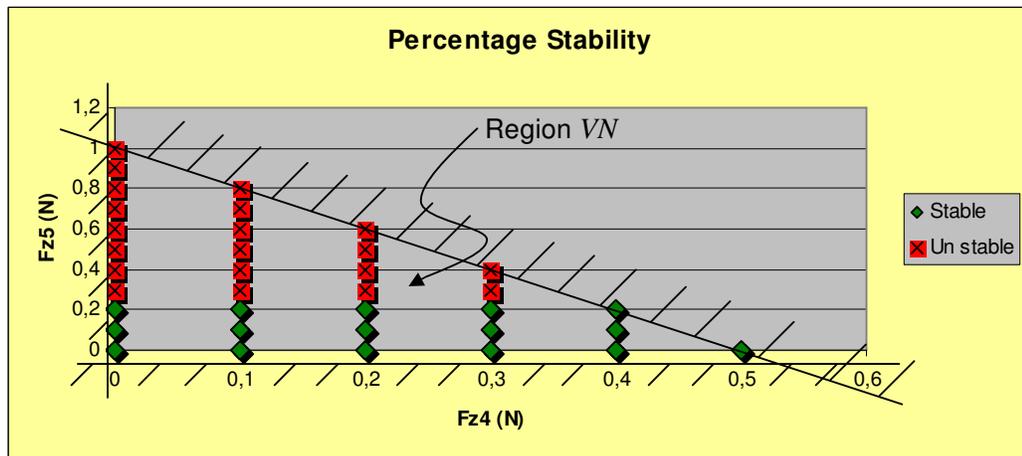


Figure 6.3 Stable and unstable regions of VN .

Table 6.1 Stable and unstable points in VN .

		SOLUTION with LP						
Fz4	Fz5	Fx3	Fx4	Fx5	Fy2	Fy3	Fy4	Fy5
0	0	0,060601	0	0	0	-0,10607	0	0
0	0,1	0,060601	0	0,212132	0	-0,09546	0	0
0	0,2	0,078775	0	0,0060779	0	-0,08485	0	0
0	0,3	-	-	-	-	-	-	-
0	0,4	-	-	-	-	-	-	-
0	0,5	-	-	-	-	-	-	-
0	0,6	-	-	-	-	-	-	-
0	0,7	-	-	-	-	-	-	-
0	0,8	-	-	-	-	-	-	-
0	0,9	-	-	-	-	-	-	-
0	1	-	-	-	-	-	-	-
0,1	0	0,039387	-0,02121	0	0,021213	-0,08485	-0,21213	0
0,1	0,1	0,039387	-0,02121	0,0212132	0,021213	-0,07425	-0,21213	0
0,1	0,2	0,057562	-0,02121	0,0060779	0,021213	-0,06364	-0,21213	0
0,1	0,3	-	-	-	-	-	-	-
0,1	0,4	-	-	-	-	-	-	-
0,1	0,5	-	-	-	-	-	-	-
0,1	0,6	-	-	-	-	-	-	-
0,1	0,7	-	-	-	-	-	-	-
0,1	0,8	-	-	-	-	-	-	-
0,2	0	0,018174	-0,04243	0	0,042426	-0,06364	-0,04243	0
0,2	0,1	0,018174	-0,04243	0,0212132	0,042426	-0,05303	-0,04243	0
0,2	0,2	0,036349	-0,04243	0,0060779	0,042426	-0,04243	-0,04243	0
0,2	0,3	-	-	-	-	-	-	-
0,2	0,4	-	-	-	-	-	-	-
0,2	0,5	-	-	-	-	-	-	-
0,2	0,6	-	-	-	-	-	-	-
0,3	0	-0,00304	-0,06364	0	0,06364	-0,04243	-0,06364	0
0,3	0,1	-0,00304	-0,06364	0,0212132	0,06364	-0,03182	-0,06364	0
0,3	0,2	0,015135	-0,06364	0,0060779	0,06364	-0,02121	-0,06364	0
0,3	0,3	-	-	-	-	-	-	-
0,3	0,4	-	-	-	-	-	-	-
0,4	0	-0,02121	-0,08485	0	0,081818	-0,01817	-0,08485	0
0,4	0,1	-0,01061	-0,08485	0,0151353	0,084853	0,010607	-0,08485	-0,02121
0,4	0,2	0	-0,08485	0	0,084853	0	-0,07877	-0,00608
0,5	0	0	-0,10607	0	0,060601	0	-0,0606	0

6.5. CASE 2: 1C1L

Consider the one cube one link configuration in Figure 6.4. The desired motion is shown in Figure 6.5 and the required inputs are given below.

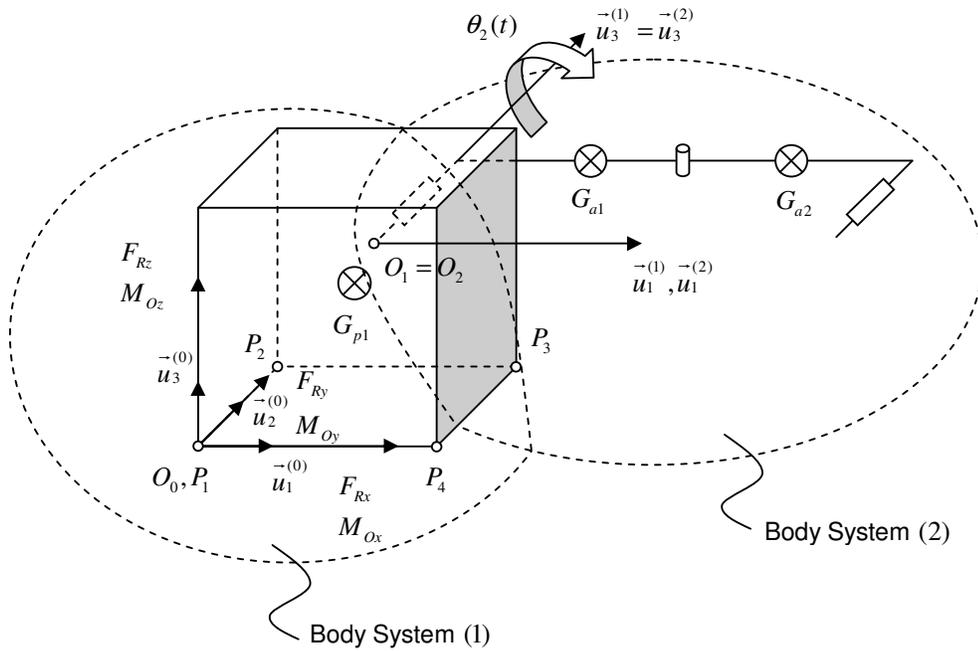


Figure 6.4 Case 3: 1C1L configuration with Search Method.

$$n_p = 1$$

$$n_a = 2$$

$$\bar{G}_{p1}^{(0)} = \begin{bmatrix} d/2 \\ d/2 \\ d/2 \end{bmatrix}, \quad \bar{G}_{a1}^{(2)} = \begin{bmatrix} d/2 \\ 0 \\ d/2 \end{bmatrix}, \quad \bar{G}_{a2}^{(2)} = \begin{bmatrix} 3d/2 \\ 0 \\ d/2 \end{bmatrix}, \quad \bar{P}_{01}^{(0)} = \begin{bmatrix} d/2 \\ d \\ d/2 \end{bmatrix}$$

$$\theta_1 = 0$$

$$\alpha_1 = -\pi/2$$

$$m_{p1} = m_c$$

$$m_{a1} = m_{a2} = m_l$$

$$\hat{\Gamma}^{(a1)} = \hat{\Gamma}^{(a2)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$J_{l1} = \frac{1}{2} m_l (d_{l2}^2 + d_{l3}^2)$$

$$J_{l2} = \frac{1}{2} m_l (d_{l1}^2 + d_{l3}^2)$$

$$J_{l3} = \frac{1}{2} m_l (d_{l1}^2 + d_{l2}^2)$$

$$n_c = 4$$

$$\bar{r}_1^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{r}_2^{(0)} = \begin{bmatrix} 0 \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_3^{(0)} = \begin{bmatrix} d \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_4^{(0)} = \begin{bmatrix} d \\ 0 \\ 0 \end{bmatrix}$$

$$\mu = 0.3$$

$$d = 0.06 \text{ m}$$

$$h = 2 \text{ sec}$$

$$t_f = 10 \text{ sec}$$

$$q_0 = 0 \text{ rad}$$

$$q_f = -\pi/2 \text{ rad}$$

$$h_s = 0.01 \text{ m}$$

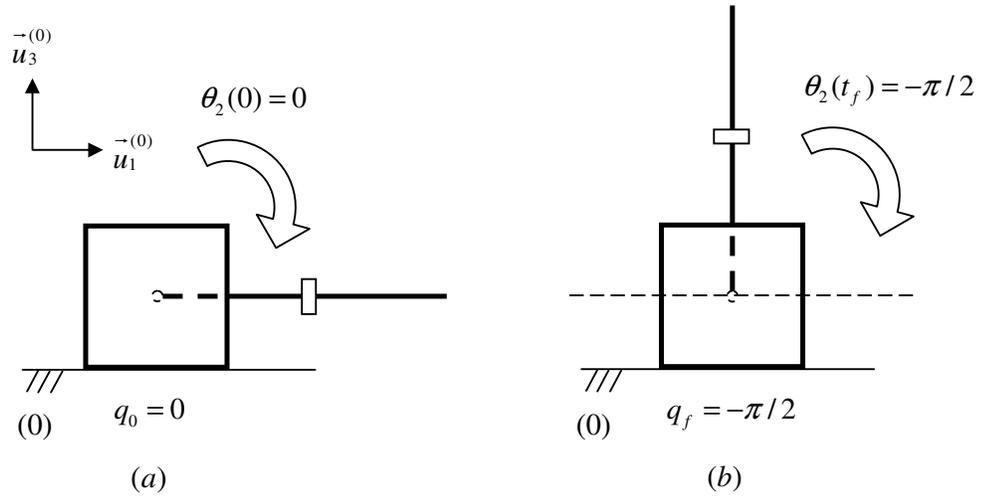


Figure 6.5 The initial (a) and final (b) configurations of 1C1L.

From equation (5.10), the orientation matrices are found as:

$$\hat{C}^{(0,1)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$

and

$$\hat{C}^{(0,2)} = \begin{bmatrix} \cos\theta_2(t) & -\sin\theta_2(t) & 0 \\ 0 & 0 & 1 \\ -\sin\theta_2(t) & -\cos\theta_2(t) & 0 \end{bmatrix}$$

From equations (5.12), (5.13), (5.14), and (5.15) the angular velocities and accelerations are found as:

$$\bar{\omega}_1^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{\omega}_2^{(0)} = \begin{bmatrix} 0 \\ \dot{\theta}_2(t) \\ 0 \end{bmatrix}, \quad \bar{\alpha}_1^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{\alpha}_2^{(0)} = \begin{bmatrix} 0 \\ \ddot{\theta}_2(t) \\ 0 \end{bmatrix}$$

$$\bar{\omega}_1^{(1)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{\omega}_2^{(2)} = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_2(t) \end{bmatrix}, \quad \bar{\alpha}_1^{(1)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{\alpha}_2^{(2)} = \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_2(t) \end{bmatrix}$$

From equations (5.1) and (5.2) the center of masses of the active and passive bodies are found to be:

$$\bar{G}_1^{(0)} = \begin{bmatrix} d/2 \\ d/2 \\ d/2 \end{bmatrix}$$

$$\bar{G}_2^{(2)} = \begin{bmatrix} d \\ 0 \\ d/2 \end{bmatrix}$$

Equation (5.19) yields the locations of the mass centers as:

$$\bar{P}_1^{(0)} = \begin{bmatrix} d/2 \\ d/2 \\ d/2 \end{bmatrix}, \quad \bar{P}_2^{(0)} = \begin{bmatrix} d/2 + d\cos\theta_2(t) \\ 3d/2 \\ d/2 - d\sin\theta_2(t) \end{bmatrix}$$

Equations (5.20) and (5.22) give the body system origin velocities and accelerations, respectively. However they are identically zero, because the body system frame origins are not moving.

The mass center velocities and accelerations, on the other hand, are found from equations (5.21) and (5.23), respectively as:

$$\bar{V}_1^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{V}_2^{(0)} = \begin{bmatrix} -d\text{Sin}\theta_2(t)\dot{\theta}_2(t) \\ 0 \\ -d\text{Cos}\theta_2(t)\dot{\theta}_2(t) \end{bmatrix}$$

$$\bar{a}_1^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{a}_2^{(0)} = \begin{bmatrix} -d(\text{Cos}\theta_2(t)\dot{\theta}_2(t)^2 + \text{Sin}\theta_2(t)\ddot{\theta}_2(t)) \\ 0 \\ d(\text{Sin}\theta_2(t)\dot{\theta}_2(t)^2 - \text{Cos}\theta_2(t)\ddot{\theta}_2(t)) \end{bmatrix}$$

$$\bar{a}_1^{(1)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{a}_2^{(2)} = \begin{bmatrix} -d\dot{\theta}_2(t)^2 \\ d\ddot{\theta}_2(t) \\ 0 \end{bmatrix}$$

The resultant resistive loads are obtained from equations (5.26) and (5.30) as

$$F_{Rx} = -0.0117(\text{Cos}\theta_2(t)\dot{\theta}_2(t)^2 + \text{Sin}\theta_2(t)\ddot{\theta}_2(t))$$

$$F_{Ry} = 0$$

$$F_{Rz} = 3.924 + 0.0117(\text{Sin}\theta_2(t)\dot{\theta}_2(t)^2 - \text{Cos}\theta_2(t)\ddot{\theta}_2(t))$$

$$M_{Ox} = 0.232497 + 0.001053(\text{Sin}\theta_2(t)\dot{\theta}_2(t)^2 - \text{Cos}\theta_2(t)\ddot{\theta}_2(t))$$

$$M_{Oy} = -0.0603315 - 0.114777\text{Cos}\theta_2(t) - 0.0573885\text{Cos}\theta_2(t)^2 -$$

$$0.0573885\text{Sin}\theta_2(t)^2 - 0.000351\dot{\theta}_2(t)^2(\text{Cos}\theta_2(t) + \text{Sin}\theta_2(t)) +$$

$$\ddot{\theta}_2(t)(0.00101715 + 0.000351(\text{Cos}\theta_2(t) - \text{Sin}\theta_2(t)))$$

$$M_{Oz} = 0.001053(\text{Cos}\theta_2(t)\dot{\theta}_2(t)^2 + \text{Sin}\theta_2(t)\ddot{\theta}_2(t))$$

Finally, the desired motion of the active body system is defined from equation (5.32) as

$$\theta_2(t) = -\frac{\pi t^3}{200} + \frac{3\pi t^4}{4000} - \frac{3\pi t^5}{100000}$$

After finding the resultant resistive loads in terms of t , they are discretized using a step size of $h = 2$ sec. The corresponding discretized resultant resistive loads are shown in Table 6.2. The percentage stability, corresponding to each resultant resistive load (i.e., $k = 1, 2, \dots, 6$), is calculated as in Case 1 and determined to be 100%. This implies that the desired motion of the active body system is guaranteed to be stable.

Table 6.2 Discretized resultant resistive loads.

k	t	FRx (k)	FRy (k)	FRz (k)	MOx (k)	MOy (k)	MOz (k)
1	0	0	0	3.924	0.232497	-0.232497	0
2	2	-0.000265748	0	3.92504	0.23259	-0.232153	0.0000239173
3	4	-0.00101017	0	3.92405	0.232502	-0.218599	0.0000909156
4	6	0.0000526105	0	3.92299	0.232406	-0.172531	-4.73495×10^{-6}
5	8	0.00103874	0	3.92373	0.232473	-0.128017	-0.0000934868
6	10	0	0	3.924	0.232497	-0.11772	0

Figure 6.6 shows the snapshots of the motion of the 1C1L configuration considered in this case study which are obtained by the simulation program, ADAMS. The solutions that are obtained via ADAMS indicate that the desired motion is stable. On the other hand, the experiments, which are made by Ünsal and Khosla, on this configuration, also reveal that the desired motion is stable [21]. Therefore, one may conclude that the developed algorithm yields reliable results.

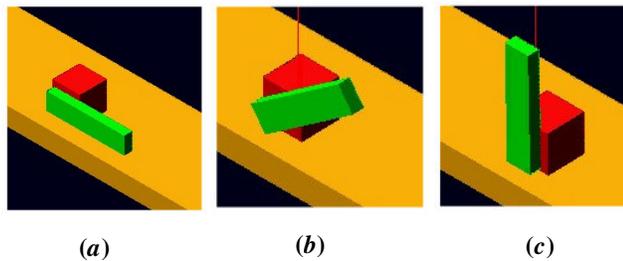


Figure 6.6 ADAMS solutions of the initial (a), intermediate (b) and final (c) configurations of 1C1L.

6.5. CASE 3: 2C1L

Consider the two cube one link configuration in Figure 6.7. The desired motion is shown in Figure 6.8 and the required inputs are given below.

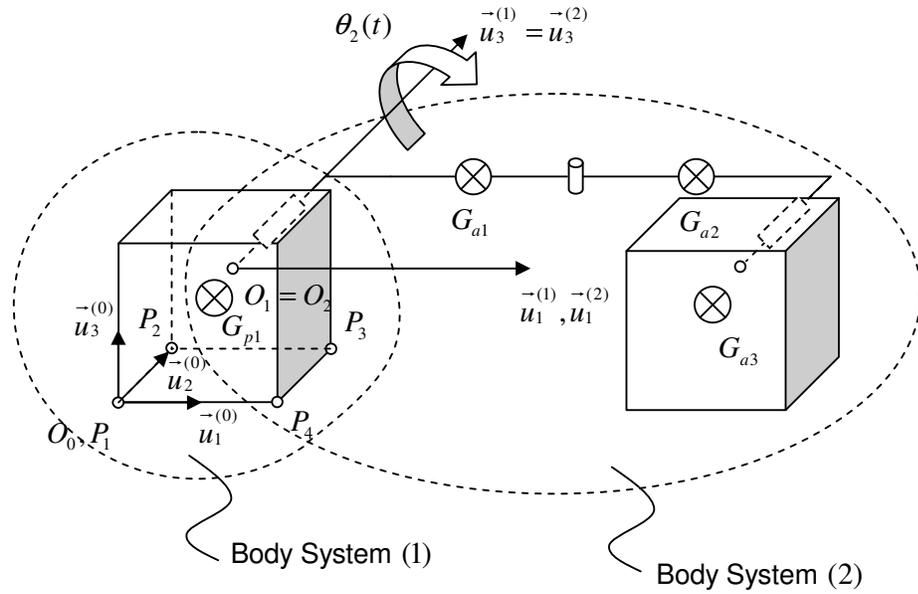


Figure 6.7 Case 4: 2C1L configuration with Search Method.

$$n_p = 1$$

$$n_a = 3$$

$$\bar{G}_{p1}^{(0)} = \begin{bmatrix} d/2 \\ d/2 \\ d/2 \end{bmatrix}, \quad \bar{G}_{a1}^{(2)} = \begin{bmatrix} d/2 \\ 0 \\ d/2 \end{bmatrix}, \quad \bar{G}_{a2}^{(2)} = \begin{bmatrix} 3d/2 \\ 0 \\ d/2 \end{bmatrix}, \quad \bar{G}_{a3}^{(2)} = \begin{bmatrix} 2d \\ 0 \\ -d/2 \end{bmatrix}, \quad \bar{P}_{01}^{(0)} = \begin{bmatrix} d/2 \\ d \\ d/2 \end{bmatrix}$$

$$\theta_1 = 0$$

$$\alpha_1 = -\pi/2$$

$$m_{p1} = m_{a3} = m_c$$

$$m_{a1} = m_{a2} = m_l$$

$$\hat{\Gamma}^{(a1)} = \hat{\Gamma}^{(a2)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \hat{\Gamma}^{(a3)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$n_c = 4$$

$$\bar{r}_1^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{r}_2^{(0)} = \begin{bmatrix} 0 \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_3^{(0)} = \begin{bmatrix} d \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_4^{(0)} = \begin{bmatrix} d \\ 0 \\ 0 \end{bmatrix}$$

$$\mu = 0.3$$

$$d = 0.06 \text{ m}$$

$$h = 2 \text{ sec}$$

$$t_f = 10 \text{ sec}$$

$$q_0 = 0 \text{ rad}$$

$$q_f = -\pi/2 \text{ rad}$$

$$h_s = 0.01 \text{ m}$$

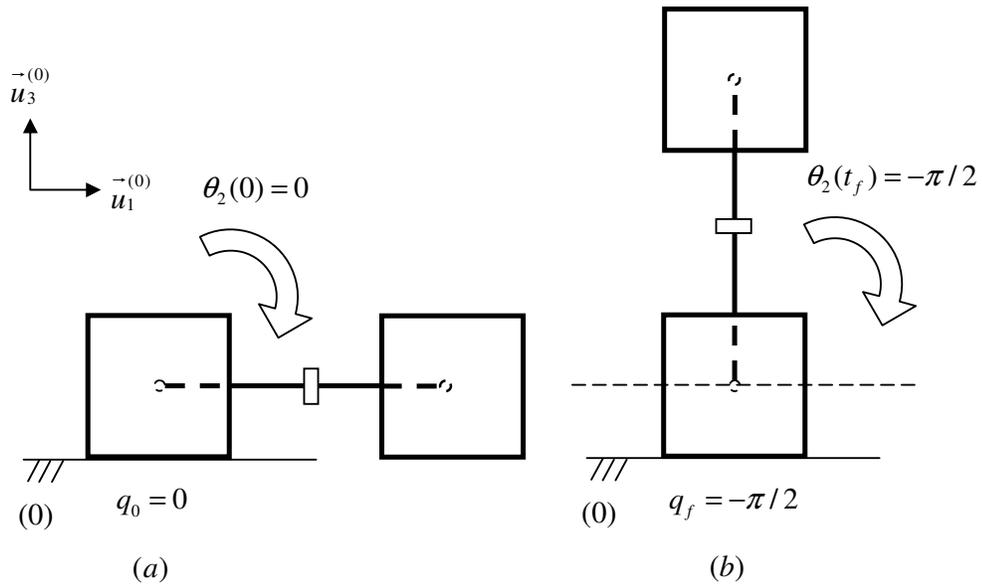


Figure 6.8 The initial (a) and final (b) configurations of 2C1L.

In this case, there is one more cube (as compared to Case 2) attached to the moving link. This extra cube effects the stability of the system adversely and the system becomes always unstable during the interval $0 < t < 6$ sec. Therefore, the active body system cannot execute the specified motion although the system is 100% stable during the rest of the motion.

Figure 6.9 shows the snapshots of the motion of the 2C1L configuration considered in this case study which are obtained by the simulation program, ADAMS. Figure 6.9 (b) shows one of the instant of the unstable configuration. The solutions that are obtained via ADAMS indicate that the desired motion is unstable. On the other hand, the experiments, which are made by Ünsal and Khosla, on this configuration, also reveal that the desired motion is unstable [21]. Therefore, one may conclude that the developed algorithm yields reliable results.

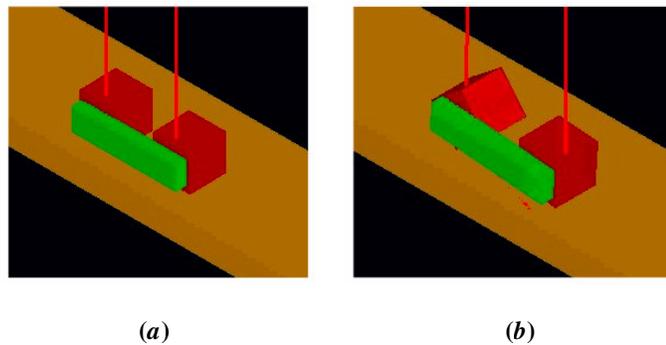


Figure 6.9 ADAMS solutions of the initial (a) and unstable (b) configurations of 2C1L.

6.5. CASE 4: 3C2L

Consider the three cube two link configuration in Figure 6.10. The desired motion is shown in Figure 6.11 and the required inputs are given below.

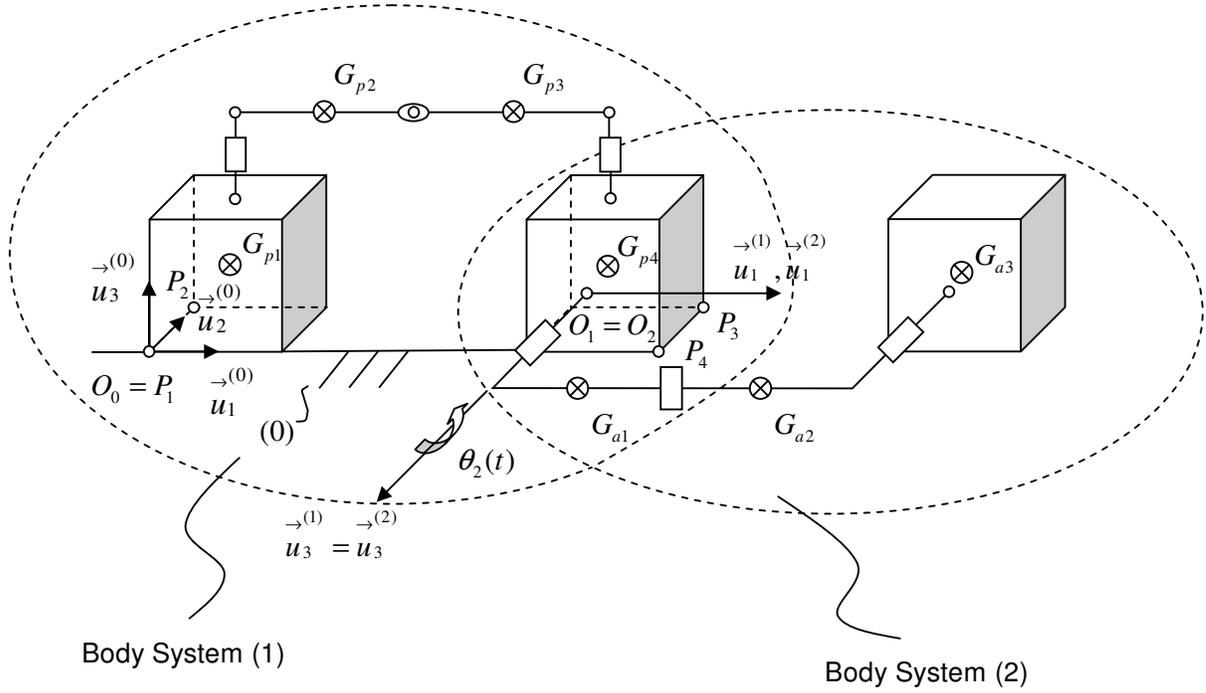


Figure 6.10 Case 5: 3C2L configuration with Search Method.

$$n_p = 4$$

$$n_a = 3$$

$$\begin{aligned} \bar{G}_{p1}^{(0)} &= \begin{bmatrix} d/2 \\ d/2 \\ d/2 \end{bmatrix}, & \bar{G}_{p2}^{(0)} &= \begin{bmatrix} d \\ d/2 \\ 3d/2 \end{bmatrix}, & \bar{G}_{p3}^{(0)} &= \begin{bmatrix} 2d \\ d/2 \\ 3d/2 \end{bmatrix}, & \bar{G}_{p4}^{(0)} &= \begin{bmatrix} 5d/2 \\ d/2 \\ d/2 \end{bmatrix} \\ \bar{G}_{a1}^{(2)} &= \begin{bmatrix} d/2 \\ 0 \\ d/2 \end{bmatrix}, & \bar{G}_{a2}^{(2)} &= \begin{bmatrix} 3d/2 \\ 0 \\ d/2 \end{bmatrix}, & \bar{G}_{a3}^{(2)} &= \begin{bmatrix} 2d \\ 0 \\ -d/2 \end{bmatrix}, & \bar{P}_{01}^{(0)} &= \begin{bmatrix} 5d/2 \\ 0 \\ d/2 \end{bmatrix} \end{aligned}$$

$$\theta_1 = 0$$

$$\alpha_1 = \pi/2$$

$$m_{p1} = m_{p4} = m_{a3} = m_c$$

$$m_{p2} = m_{p3} = m_{a1} = m_{a2} = m_l$$

$$\hat{\Gamma}^{(a1)} = \hat{\Gamma}^{(a2)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \hat{\Gamma}^{(a3)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$n_c = 4$$

$$\bar{r}_1^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{r}_2^{(0)} = \begin{bmatrix} 0 \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_3^{(0)} = \begin{bmatrix} 3d \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_4^{(0)} = \begin{bmatrix} 3d \\ 0 \\ 0 \end{bmatrix}$$

$$\mu = 0.3$$

$$d = 0.06 \text{ m}$$

$$h = 2 \text{ sec}$$

$$t_f = 10 \text{ sec}$$

$$q_0 = 0 \text{ rad}$$

$$q_f = \pi/2 \text{ rad}$$

$$h_s = 0.01 \text{ m}$$

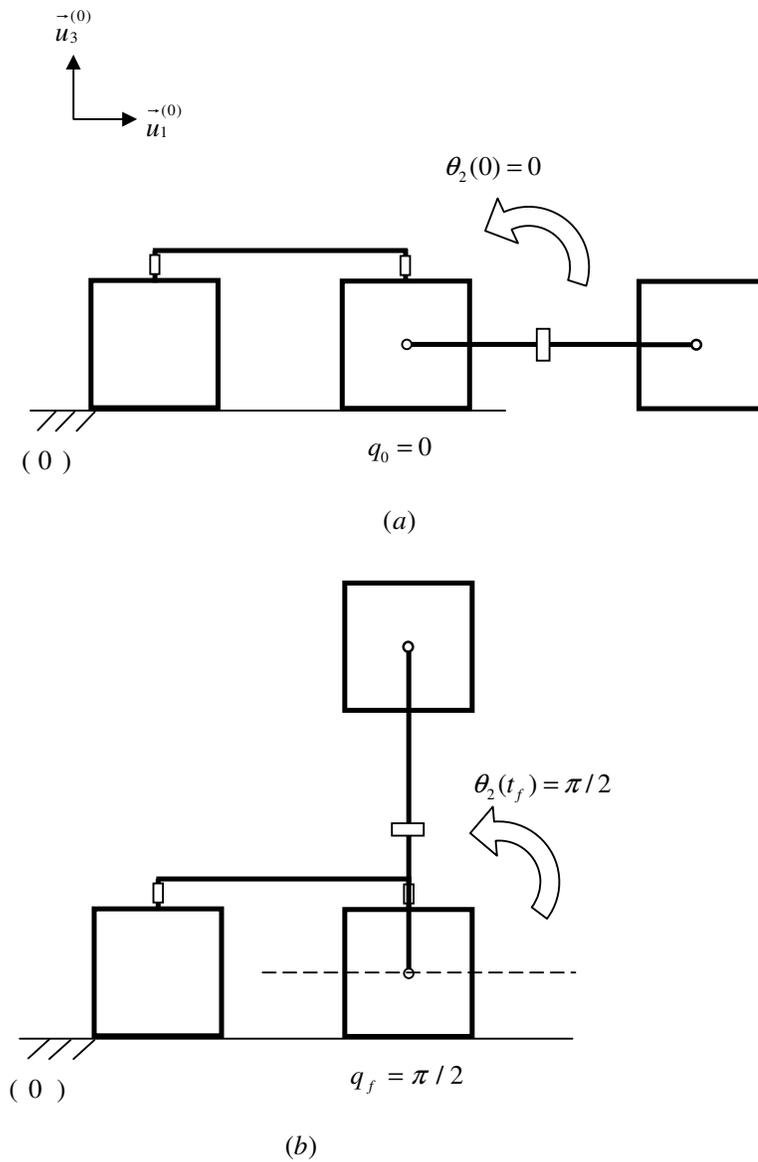


Figure 6.11 The initial (a) and final (b) configurations of 3C2L.

In this case, there is one more link and one more cube added to the system shown in Figure 6.7. It is expected that, the additional cube and link will affect the stability of the system positively, if the desired motion is the same.

The percentage stability is determined to be 100% throughout the motion. Therefore, the desired motion can be executed safely.

Figure 6.12 shows the snapshots of the motion of the 3C2L configuration considered in this case study which are obtained by the simulation program, ADAMS. The solutions that are obtained via ADAMS indicate that the desired motion is stable. On the other hand, the experiments, which are made by Ünsal and Khosla, on this configuration, also reveal that the desired motion is stable [21]. Therefore, one may conclude that the developed algorithm yields reliable results.

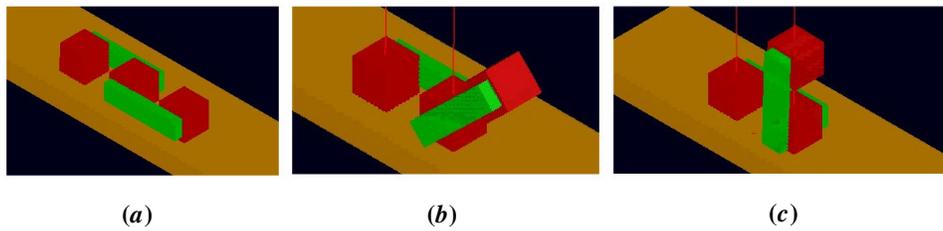


Figure 6.12 ADAMS solutions of the initial (a), intermediate (b) and final (c) configurations of 3C2L.

6.5. CASE 5: 4C4L

Consider the four cube four link configuration in Figure 6.13. The required inputs are given below.

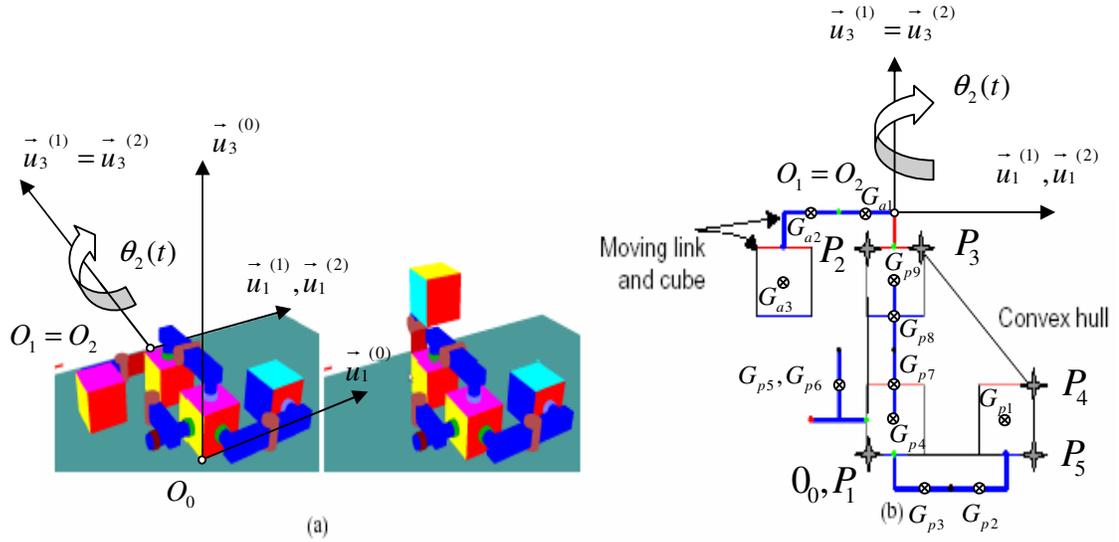


Figure 6.13 Given a 90° link motion with initial and final configurations (a), convex hull and the contact points (b) [37].

$$n_p = 9$$

$$n_a = 3$$

$$\begin{aligned} \bar{G}_{p1}^{(0)} &= \begin{bmatrix} 5d/2 \\ d/2 \\ d/2 \end{bmatrix}, & \bar{G}_{p2}^{(0)} &= \begin{bmatrix} 2d \\ -d/2 \\ d/2 \end{bmatrix}, & \bar{G}_{p3}^{(0)} &= \begin{bmatrix} d \\ -d/2 \\ d/2 \end{bmatrix}, & \bar{G}_{p4}^{(0)} &= \begin{bmatrix} d/2 \\ d/2 \\ d/2 \end{bmatrix} \\ \bar{G}_{p5}^{(0)} &= \begin{bmatrix} -d/2 \\ d \\ d/2 \end{bmatrix}, & \bar{G}_{p6}^{(0)} &= \begin{bmatrix} -d/2 \\ d \\ d/2 \end{bmatrix}, & \bar{G}_{p7}^{(0)} &= \begin{bmatrix} d/2 \\ d \\ 3d/2 \end{bmatrix}, & \bar{G}_{p8}^{(0)} &= \begin{bmatrix} d/2 \\ 2d \\ 3d/2 \end{bmatrix}, & \bar{G}_{p9}^{(0)} &= \begin{bmatrix} d/2 \\ 5d/2 \\ d/2 \end{bmatrix} \\ \bar{G}_{a1}^{(2)} &= \begin{bmatrix} -d/2 \\ 0 \\ d/2 \end{bmatrix}, & \bar{G}_{a2}^{(2)} &= \begin{bmatrix} -3d/2 \\ 0 \\ d/2 \end{bmatrix}, & \bar{G}_{a3}^{(2)} &= \begin{bmatrix} -2d \\ 0 \\ -d/2 \end{bmatrix}, & \bar{P}_{01}^{(0)} &= \begin{bmatrix} d/2 \\ 3d \\ d/2 \end{bmatrix} \end{aligned}$$

$$\theta_1 = 0$$

$$\alpha_1 = \pi/2$$

$$m_{p1} = m_{p4} = m_{p9} = m_{a3} = m_c$$

$$m_{p2} = m_{p3} = m_{p5} = m_{p6} = m_{p7} = m_{p8} = m_{a1} = m_{a2} = m_l$$

$$\hat{\Gamma}^{(a1)} = \hat{\Gamma}^{(a2)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \hat{\Gamma}^{(a3)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$n_c = 5$$

$$\bar{r}_1^{-(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \bar{r}_2^{-(0)} = \begin{bmatrix} 0 \\ 3d \\ 0 \end{bmatrix}, \quad \bar{r}_3^{-(0)} = \begin{bmatrix} d \\ 3d \\ 0 \end{bmatrix}, \quad \bar{r}_4^{-(0)} = \begin{bmatrix} 3d \\ d \\ 0 \end{bmatrix}, \quad \bar{r}_5^{-(0)} = \begin{bmatrix} 3d \\ 0 \\ 0 \end{bmatrix}$$

$$\mu = 0.3$$

$$d = 0.06 \text{ m}$$

$$h = 2 \text{ sec}$$

$$t_f = 10 \text{ sec}$$

$$q_0 = 0 \text{ rad}$$

$$q_f = \pi/2 \text{ rad}$$

$$h_s = 0.01 \text{ m}$$

The procedure is same as the aforementioned cases. The percentage stability is determined to be 100% throughout the motion. Therefore, the desired motion can be executed safely.

6.6. CASE 6: TWO FACE CONTACT

Consider the single cube which has two contacting faces with the ground as shown in Figure 6.14 where the external load is given by

$$\vec{F}_{ex} = (0.01\vec{u}_1 + 5\vec{u}_2 - 5\vec{u}_3)$$

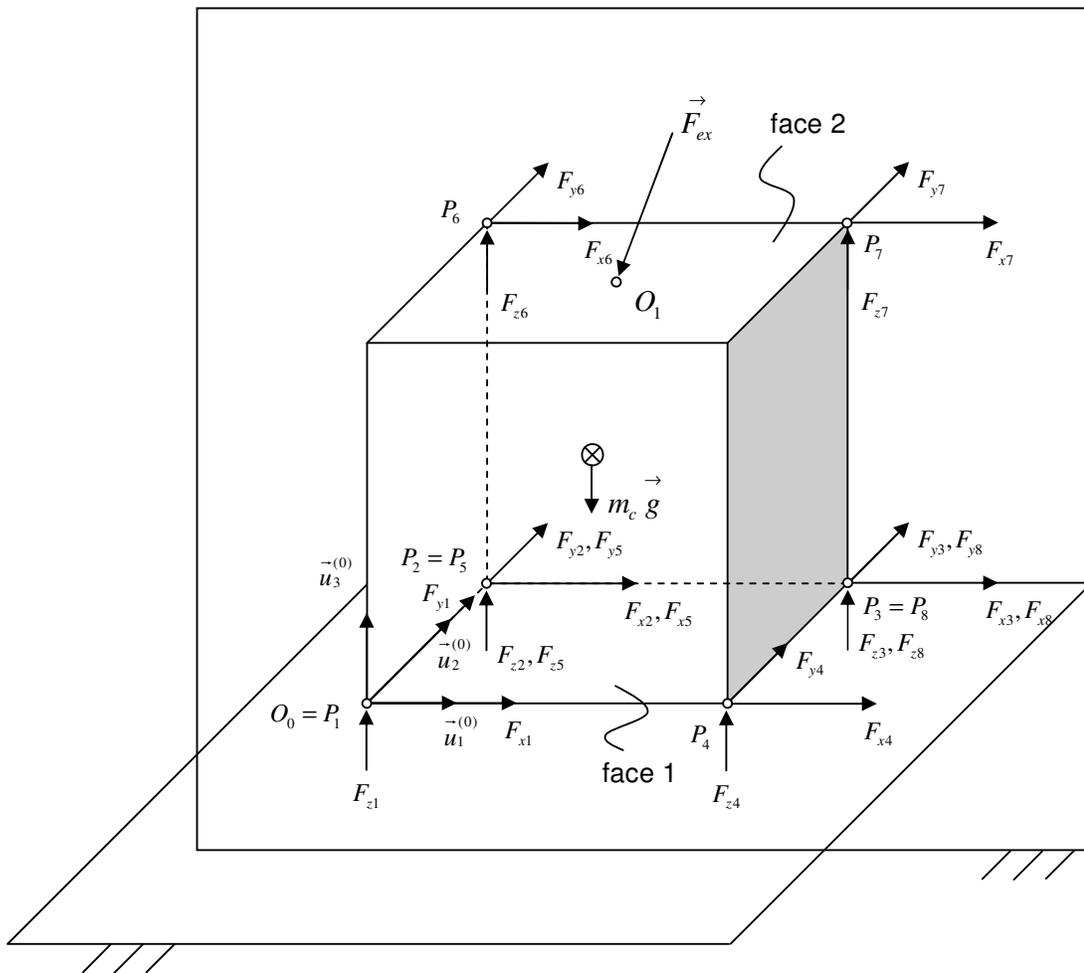


Figure 6.14 Case 7: Two Face Contact with Potential Stability Method.

The required inputs are given below.

$$\mu = 0.3$$

$$d = 0.06 \text{ m}$$

$$W = m_c g = 1 \text{ N}$$

$$n_c = 8 \text{ (4 contact points in each faces)}$$

$$\begin{aligned} \vec{r}_1^{(0)} &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, & \vec{r}_2^{(0)} = \vec{r}_5^{(0)} &= \begin{bmatrix} 0 \\ d \\ 0 \end{bmatrix}, & \vec{r}_3^{(0)} = \vec{r}_8^{(0)} &= \begin{bmatrix} d \\ d \\ 0 \end{bmatrix}, & \vec{r}_4^{(0)} &= \begin{bmatrix} d \\ 0 \\ 0 \end{bmatrix} \\ \vec{r}_6^{(0)} &= \begin{bmatrix} 0 \\ d \\ d \end{bmatrix}, & \vec{r}_7^{(0)} &= \begin{bmatrix} d \\ d \\ d \end{bmatrix} \end{aligned}$$

The resultant loads to be applied by the ground on the cube are found from equation (4.1) to be

$$F_{Rx} = -0.01 \text{ N}$$

$$F_{Ry} = -5 \text{ N}$$

$$F_{Rz} = 6 \text{ N}$$

$$M_{Ox} = 0.48 \text{ N.m}$$

$$M_{Oy} = -0.1806 \text{ N.m}$$

$$M_{Oz} = -0.1497 \text{ N.m}$$

Equations (4.2) and (4.3) lead to the following equations

$$\begin{aligned}
F_{Rx} &= F_{x1} + F_{x2} + F_{x3} + F_{x4} + F_{x5} + F_{x6} + F_{x7} + F_{x8} \\
F_{Ry} &= F_{y1} + F_{y2} + F_{y3} + F_{y4} + F_{y5} + F_{y6} + F_{y7} + F_{y8} \\
F_{Rz} &= F_{z1} + F_{z2} + F_{z3} + F_{z4} + F_{z5} + F_{z6} + F_{z7} + F_{z8} \\
M_{Ox} &= -d.F_{y6} - d.F_{y7} + d.F_{z2} + d.F_{z3} + d.F_{z5} + \\
&\quad d.F_{z6} + d.F_{z7} + d.F_{z8} \\
M_{Oy} &= d.F_{x6} + d.F_{x7} - d.F_{z3} - d.F_{z4} - d.F_{z7} - d.F_{z8} \\
M_{Oz} &= -d.F_{x2} - d.F_{x3} - d.F_{x5} - d.F_{x6} - d.F_{x7} - d.F_{x8} + \\
&\quad d.F_{y3} + d.F_{y4} + d.F_{y7} + d.F_{y8}
\end{aligned} \tag{6.1}$$

Solving $F_{z1}, F_{z2}, F_{z3}, F_{x1}, F_{x2}, F_{y1}$ (in terms of $F_{z4}, F_{z5}, F_{z6}, F_{z7}, F_{z8}, F_{x3}, F_{x4}, F_{x5}, F_{x6}, F_{x7}, F_{x8}, F_{y2}, F_{y3}, F_{y4}, F_{y5}, F_{y6}, F_{y7}, F_{y8}$) from the equalities (6.1), one obtains

$$\begin{aligned}
F_{z1} &= 6 - \frac{0.48}{d} - F_{y6} - F_{y7} - F_{z4} \\
F_{x1} &= -0.01 - \frac{0.1497}{d} - F_{x4} - F_{y3} - F_{y4} - F_{y7} - F_{y8} \\
F_{y1} &= -5 - F_{y2} - F_{y3} - F_{y4} - F_{y5} - F_{y6} - F_{y7} - F_{y8} \\
F_{z2} &= \frac{0.2994}{d} - F_{x6} - F_{x7} + F_{y6} + F_{y7} + F_{z4} - F_{z5} - F_{z6} \\
F_{x2} &= \frac{0.1497}{d} - F_{x3} - F_{x5} - F_{x6} - F_{x7} - F_{x8} + F_{y3} + F_{y4} + F_{y7} + F_{y8} \\
F_{z3} &= \frac{0.1806}{d} + F_{x6} + F_{x7} - F_{z4} - F_{z7} - F_{z8}
\end{aligned} \tag{6.2}$$

For each face, the following normal and friction contact force inequalities must be satisfied.

$$\sum_{i=1}^4 F_{zi} \geq 0 \tag{4.6}$$

$$\sum_{i=5}^8 -F_{yi} \geq 0 \quad (6.3)$$

$$\begin{aligned} \frac{\mu}{\sqrt{2}} F_{zi} + F_{xi} &\geq 0 \\ \frac{\mu}{\sqrt{2}} F_{zi} - F_{xi} &\geq 0 \\ \frac{\mu}{\sqrt{2}} F_{zi} + F_{yi} &\geq 0 \\ \frac{\mu}{\sqrt{2}} F_{zi} - F_{yi} &\geq 0 \end{aligned} \quad (4.16)$$

$$i = 1, \dots, 4$$

$$\begin{aligned} -\frac{\mu}{\sqrt{2}} F_{yi} + F_{xi} &\geq 0 \\ -\frac{\mu}{\sqrt{2}} y_{zi} - F_{xi} &\geq 0 \\ -\frac{\mu}{\sqrt{2}} F_{yi} + F_{yi} &\geq 0 \\ -\frac{\mu}{\sqrt{2}} F_{yi} - F_{yi} &\geq 0 \end{aligned} \quad (6.4)$$

$$i = 5, \dots, 8$$

If we substitute equalities (6.2) into the inequalities (4.6), (6.3), (4.16), (6.4), with Linear Programming, the free variables could be found as

$$\begin{aligned} \{F_{z4}, F_{z5}, F_{z6}, F_{z7}, F_{z8}, F_{x3}, F_{x4}, F_{x5}, F_{x6}, F_{x7}, F_{x8}, F_{y2}, F_{y3}, F_{y4}, F_{y5}, F_{y6}, F_{y7}, F_{y8}\} = \\ \{1.10897 \times 10^{-15}, -0.768973, 0., -0.624388, 0., 0.638517, -1.20692 \times 10^{-17}, -0.553733, \\ 0., -0.624388, 0., 0.729727, 0.638517, 0., -3.62498, -7.61448 \times 10^{-16}, -2.94339, 0.\} \end{aligned}$$

which implies that the given configuration is potentially stable.

CHAPTER 7

CONCLUSIONS

In this study, an efficient algorithm has been developed for the dynamic stability analysis of self-reconfigurable, modular robots. Although the algorithm is illustrated specifically using the modular robotic system, I-Cubes, it can be easily extended to other reconfigurable systems. To the author's knowledge, this algorithm is first of its kind in the literature.

While planning the motion of a self-reconfigurable robotic system, one has to consider the dynamic stability of the system. In order to achieve this goal, the algorithm reduces the whole I-Cubes structure into a one-DOF system and models the motion by the fixed axis rotation of one body system (called the Active Body System) with respect to another body system which is considered to be fixed relative to the ground (called the Passive Body System). The algorithm also assumes a probable motion to reach the final configuration from the initial configuration. By recursive kinematic relations and the Newton-Euler formulation, the algorithm calculates the resultant force and the accompanying resultant moment to be applied by the ground on the system so that the system is dynamically stable. Finally, by a novel definition of stability, called percentage stability, the algorithm determines whether the assumed motion is allowable or not. In the algorithm, the friction cone is approximated by a pyramid and then linear programming is used to solve the resulting linearized friction inequalities. The effects of changing the number of faces of the pyramid and the number of contact points are also investigated.

Percentage stability is introduced to counteract the adverse effects of the static indeterminacy problem between two contacting bodies. In the literature, potential stability seems to be the only definition for the stability analysis. Potential stability investigates if there exists a valid set of contact forces that makes the system motionless. The system which is not potentially stable is definitely unstable; however, the opposite is not always true. A system which is potentially stable could be unstable because of the static indeterminacy. On the other hand, percentage

stability investigates all possible sets of contact forces that could arise to make the system motionless. This feature renders percentage stability to be more reliable than potential stability. A system which is zero percent stable is definitely unstable whereas hundred percent stability corresponds to guaranteed stability since the system will be stable for all possible normal force distributions. However, as percentage stability decreases, the “chances” of having a stable system decreases since some possible normal force distributions lead to instability of the system. One can roughly state that, percentage stability is the possibility of the system to be motionless. However, in potential stability, there is a possibility that the system is motionless.

The computation costs associated with percentage stability may increase extensively as the number of nodes is increased. Furthermore, this novel concept is not applicable if the normal forces (between the ground and the PBS) are acting in two or more different directions.

The developed algorithm has been applied to different configurations of I-Cubes to investigate the feasibility of various assumed motions. The results reveal that this kind of an algorithm is necessary in order to realize motion planning for reconfigurable robots.

The studies performed in this thesis are restricted to the stability analysis of I-Cubes. However, the algorithm could also be used for other types of modular robots by slight modifications. In the future, optimal motion planning of I-Cubes can be investigated. Also a graphical user interface could be designed to simulate the motion of I-Cubes.

REFERENCES

- [1] R.S. Palmer. "Computational Complexity of Motion and Stability of Polygons", *Ph.D. dissertation, Dep. Comput. Sci., Cornell Univ., Ithaca, NY*, 1989.
- [2] R. Cottle. "Computational complexity of motion and stability of polygons", Ph. D dissertation, Dept. Computer. Sci., Cornell Univ., Ithaca, NY, May 1989.
- [3] W. Ingleton. "A problem in linear inequalities", in *Proc. London Mathematical Society*, 1966, vol. 16, no. 3, pp. 519-536.
- [4] R. Mattikalli, D. Baraff, and P. Khosla. "Finding All Stable Orientations of Assemblies with Friction", *IEEE Transactions on Robotics and Automation* 12(2), pages 290-301, 1996.
- [5] D. Baraff. "Issues in Computing Contact Forces for Non-Penetrating Rigid Bodies", *Algorithmica* 10, pages 292-352, 1993.
- [6] J.-S. Pang, and J. Trinkle. "Stability Characterizations of Rigid Body Contact Problems", *ZAMM-Z. Angew. Math. Mech.* 80, pages 643-663, 2000.
- [7] S. Hirose. "Biologically Inspired Robots: Snake-Like Locomotors and Manipulators", *Oxford University Press, New York*, 1993.
- [8] G.J. Hamlin, and A.C. Sanderson. "Tetrobot, modular robotics: Prototype and experiments", *Proceedings of the 1996 IEEE International Conference on Intelligent Robots and Systems*, pages 390-395, 1996.
- [9] T. Fukuda, and Y. Kawauchi. "Cellular robotic system (CEBOT) as one of the realization of self-organizing intelligent universal manipulator", *Proceedings of*

the 1990 IEEE International Conference on Robotics and Automation, pages 662–667, 1990.

[10] S. Murata, H. Kurokawa, E. Yoshida, K. Tomita, and S. Kokaji. “A 3-D self-reconfigurable structure”, *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, pages 432–439, 1998.

[11] K. Tomita, S. Murata, H. Kurokawa, E. Yoshida, and S. Kokaji. “Self-assembly and self-repair method for a distributed mechanical system”, *IEEE Transactions on Robotics and Automation*, pages 1035–1045, 1999.

[12] D. Rus, and C. McGray. “Self-reconfigurable modular as 3-D metamorphic robots”, *Proceedings of the 1998 IEEE International Conference on Intelligent Robots and Systems*, pages 837–842, 1998.

[13] G.S. Chirikjian. “Kinematics of a Metamorphic System”, *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 449–455, 1994.

[14] A. Pamecha, and G.S. Chirikjian. “A useful metric for modular robot motion planning”, *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, pages 442–447, 1996.

[15] H. Bojinov, A. Casal, and T. Hogg. “Multiagent control of self-reconfigurable robots”, *Proceedings of International Conference of Multiagent Systems*, pages 143–150, 2000.

[16] D. Rus, and M. Vona. “A physical implementation of the self-reconfigurable crystalline robot”, *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, pages 1726–1733, 2000.

[17] J. Michael. “Fractal robots”, <http://www.stellar.demon.co.uk>, 1994.

[18] E. Yoshida, S. Kokaji, S. Murata, H. Kurokawa, and K. Tomita. “Miniaturized self-reconfigurable system using shape memory alloy”, *Proceedings of the 1999*

IEEE International Conference on Intelligent Robots and Systems, pages 1579–1585, 1999.

[19] W.-M. Shen, B. Salemi, and P. Will, “Hormones for self-reconfigurable robotics”, *Intelligent Autonomous System 6*, pages 918–925, 2000.

[20] C. Ünsal, and P.K. Khosla, “Mechatronic design of a modular self-reconfigurable robotics system”. *Proceedings of the 2000 IEEE International Conference on Intelligent Robots and Systems*, pages 1742–1747, 2000.

[21] C. Ünsal, I-Cubes: A Modular Self-Reconfiguring Bipartite Robotic System <http://www-2.cs.cmu.edu/~unsal/research/ices/cubes/index.htm>

[22] M. Yim. “New locomotion gaits”, *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2508–2514, 1994.

[23] M. Yim, D. G. Duff, and K.D. Roufas. “Polybot: A Modular Reconfigurable Robot”, in *Proceedings of IEEE International Conference on Robotics and Automation*, pages 514-520, 2000.

[24] S. Murata, E. Yoshida, K. Tomita, H. Kurokawa, A. Kamimura, and S. Kokaji. “Hardware design of modular robotic system”, *Proceedings of the 2000 IEEE International Conference on Intelligent Robots and Systems*, pages 2210–2217, 2000.

[25] S. Murata, E. Yoshida, H. Kurokawa, K. Tomita, and S. Kokaji. “Self-repairing mechanical systems”, *Autonomous Robots 10*, pages 7-21, 2001.

[26] C.J.J. Paredis, and P.K. Khosla. “Design of modular fault tolerant manipulators”, in K. Goldberg, D.Halperin, J.-C. Latombe, and R. Wilson, editors, *Algorithmic Foundations of Robotics*, pages 371-383, 1995.

[27] G. Yang, I.M. Chen, W.K. Lim, and S.H. Yeo. “Design and Kinematic Analysis of Modular Reconfigurable Parallel Robots”, *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, 1999.

- [28] J. Zhiming, and L. Zhenqun. "Identification of Placement Parameters for Modular Platform Manipulators", *Journal of Robotic Systems* 16(4), pages 227-236, 1999.
- [29] K. Kotay, and D. Rus. "The Inchworm Robot: A Multi-Functional System", *Autonomous Robots* 8, pages 53-69, 2000.
- [30] R. Grabowsky, and L.E. Navarro-Serment. "Heterogeneous Teams of Modular Robots for Mapping and Exploration", *Autonomous Robots* 8, pages 293-308, 2000.
- [31] S. Farritor, and S. Dubowsky. "On Modular Design of Field Robotic Systems", *Autonomous Robots* 10, pages 57-65, 2001.
- [32] C.J.J. Paredis, H.B. Brown, R.W. Casciola, J.E. Moody, and P.K. Khosla. "A rapidly deployable manipulator system", in *International Workshop on Some Critical Issues in Robotics*, pages 175-185, Singapore, 1995.
- [33] T. Matsumaru. "Design and Control of the Modular Robot System: TOMMS", in *Proceedings of IEEE Conference on Robotics and Automation*, pages 2125-2131, Nagoya, Japan, 1995.
- [34] T. Fukuda, and S. Nakagawa. "Dynamically Reconfigurable Robotic System", in *Proceedings of IEEE Conference on Robotics and Automation*, pages 1581-1586, Philadelphia, PA, 1988.
- [35] D. Tesar, and M.S. Butler. "A Generalized Modular Architecture for Robot Structures", *ASME Journal of Manufacturing Review* 2(2), pages 91-117, 1989.
- [36] I.M. Chen. "Rapid Response Manufacturing Through a Rapidly Reconfigurable Robotic Workcell", *Robotics and Computer Integrated Manufacturing* 17, pages 199-213, 2001.
- [37] C. Ünsal, and P.K. Khosla. "Solutions for 3-D self-reconfiguration in a modular robotic system: Implementation and motion planning", *Proceedings of*

SPIE, Sensor Fusion and Decentralized Control in Robotic Systems III, vol. 4196, pages 388-401, 2000.

[38] C. Ünsal, H. Kiliççöte, and P.K. Khosla. "A Modular Self-Reconfigurable Bipartite Robotic System: Implementation and Motion Planning", in *Autonomous Robots 10*, pages 23-40, 2001.

[39] C. Ünsal, H. Kiliççöte, and P.K. Khosla. "I (CES)-Cubes: A Modular Self-Reconfigurable Bipartite Robotic System", *Proceedings of SPIE, Sensor Fusion and Decentralized Control in Robotic Systems II*, vol. 3839, pages 258-269, 1999.

[40] C. Ünsal, H. Kiliççöte, M. Patton, and P.K. Khosla. "Motion Planning for a Modular Self-Reconfiguring Robotic System", *5th International Symposium on Distributed Autonomous Robotic Systems*, 2000.

[41] K.C. Prevas, C. Ünsal, M.Ö. Efe, and P.K. Khosla. "A Hierarchical Motion Planning Strategy for a Uniform Self-Reconfigurable Modular Robotic System", *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 787-792, May 2002.

[42] C. Ünsal, and P.K. Khosla. "A Multi-Layered Planner for Self-Reconfiguration of a Uniform Group of I-Cube Modules", *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 598-605, 2001.

[43] M.K. Özgören. "Lecturer notes for ME 522", METU.

[44] M.K. Özgören. "Topological analysis of 6-joint serial manipulators and their inverse kinematic solutions", *Mechanism and Machine Theory*, 37, pages 511-547, 2002.

[45] J. Denavit, and R.S. Hartenberg. "Kinematic notation for lower-pair mechanisms based on matrices", *ASME Journal of Applied Mechanics*, 2:215-231, 1995.