

Simulation of Contact Forces Between Cylindrical Slack and
Flexible Tethers for Situations of Self-Collision

by

André R. Roy

BScE, University of New Brunswick, 2005

A Thesis Submitted in Partial Fulfillment of
the Requirements for the Degree of

MASTER OF SCIENCE IN ENGINEERING

in the Graduate Academic Unit of Mechanical Engineering.

Supervisor: Dr. Juan Carretero, Dept. of Mechanical Engineering

Examining Board: Dr. Bob Rogers, Department of Mechanical Engineering
Dr. Simon Li, Department of Mechanical Engineering
Dr. Joseph Horton, Department of Computer Science

This thesis is accepted by the
Dean of Graduate Studies

UNIVERSITY OF NEW BRUNSWICK

September, 2007

© ANDRÉ R. ROY, 2007



Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-69279-0
Our file *Notre référence*
ISBN: 978-0-494-69279-0

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

To two years well spent.

Acknowledgements

To Dr. Juan A. Carretero, I've appreciated the time and patience you've afforded me. Our discussions were always stimulating, and I'm pretty sure we always both came out of them having gained new knowledge. I would most of all like to thank you for the opportunity that you gave me, these two years were very enlightening. I can't think of any other way I would rather have spent them.

To my parents, thank you for all of your love and support. You've encouraged me and gave me the opportunities I needed to prove myself. For that, I thank you.

To Jawn, Letchmann and Brad, if it wasn't for you guys as role-models, I would never have made it this far. I will always look back at these years as among the best in my life.

To Jenn, all the kitties in the world wouldn't be enough to show you what you mean to me. Thank you for being there.

To Life, Liberty and the pursuit of Happiness.

Abstract

Currently, the only viable means of providing power and maintaining human control during underwater Remotely Operated Vehicle (ROV) operation is through the vehicle's tether. Due to the high cost of ROVs and their tethers, as well as potential risks to equipment and personnel, a realistic simulator is needed to train their pilots. To accurately simulate the tether, it is important to detect collisions of the tether with the environment and with itself, as well as to calculate the forces involved during such contact. The aim of this work is to present a computationally efficient and accurate method of detecting tether self-contact and determine the contact forces. To this end, a combinatorial global optimisation method is first used to determine the approximate separation distances. Then, a local optimisation scheme is used to find the exact separation distance and the location of the closest points. This information can then be used to determine whether or not a collision has occurred. If a collision is detected, a force needs to be calculated and applied at the collision site to maintain separation.

In this work, a contact model for cylindrical tethers based on interference volume is described. The model is theoretically analogous to a Winkler foundation model with damping. The model is successfully compared to Hertzian theory for general contact to ensure fidelity. This model, coupled with a frictional model that exhibits stick/slip transitions, Stribeck effect, dwell-time dynamics of sticking and viscous friction is used in a tether simulation. The model is coupled to a lumped-mass model of slack flexible tethers and has shown to be effective at providing forces and maintaining tether separation during tether self-contact simulations.

Table of Contents

Dedication	ii
Abstract	iv
Table of Contents	v
List of Figures	vii
1 Introduction	1
1.1 Motivation	1
1.2 Literature Review	2
1.2.1 Collision Detection	3
1.2.2 The Minimum Separation Distance	7
1.2.3 Contact Dynamics	8
1.2.4 Other Simulations	12
1.3 The Physical Simulation and the Tether Model	14
1.4 Thesis Objectives	16
1.5 Thesis Overview	16
2 Self-Contact Collision Detection of Strand-Like Objects	18
2.1 Finding the Minimum Separation Distance	19
2.1.1 Step 1: Identifying the Closest Regions	20
2.1.2 Step 2: Determining the Exact Minimum Separation Distance	24
2.2 Defining Collisions	27
2.3 Numerical Examples	32
3 Contact Dynamics	38
3.1 Interference Geometry Between Two straight Skew Cylinders	38
3.2 Contact Surface Area Between Two Interfering Cylinders	42
3.3 Determining the Normal Contact Force	46
3.4 Determining the Tangential Contact Forces	50

3.4.1	Measuring $\sigma_0, \sigma_1, \sigma_2, \nu_s, \mu_s, \mu_c,$ and τ_{dw}	54
3.5	Numerical Example	56
4	Conclusions	59
4.1	Thesis Contributions	60
4.2	Future Work	61
	References	63
A	Algorithm Flow Diagrams	68
B	The Minimum Separation Distance Gradient and Hessian Between Two Cubic Splines	72
C	Obtaining the Normal Contact Force through Hertz' Theory of General Contact	76
	Curriculum Vita	

List of Figures

1.1	Examples of contacting tethers (contacting portions highlighted): a) tether-tether contact, b) tether-environment contact.	2
1.2	A simulation environment with 8 objects where the space is a) unpartitioned b) partitioned into two groups and c) partitioned into four groups.	4
1.3	Examples of different space partitioning methods: a) uniform grids b) k-d tree c) BSP tree d) bounding volume hierarchy	5
1.4	A similar bounding volume hierarchy example as Figure 1.3d and its tree representation.	6
1.5	Friction force versus displacement showing the change from sticking to sliding friction [1].	13
2.1	Example of MLSDist with three different starting node pairs.	21
2.2	The 3D surface of the distance between a point on the tether at s_1 and a point on the tether at s_2	23
2.3	The tether with a) minimum distances found. b) exact minimum found on the cubic spline segment.	25
2.4	a) Nine minimum separation distance pairs. b) Initial pairs prior to being minimised.	26
2.5	A collision that is detected by verifying if the separation distance is less than $r_1 + r_2$	27
2.6	Two cylinder segments crossing each other between time steps.	28
2.7	a) Defining β_{min} to determine if a collision has occurred between time steps i and $i + 1$. b) Defining β and P_{start} to determine if a collision has occurred between time steps i and $i + 1$	29
2.8	Case where the time step was too large and the tether travelled too far to be considered accurate. The angle β can be used to judge the accuracy of the collision.	30
2.9	The minimum separation distance between two skewed lines and the line normal to both.	31
2.10	Continuous collision detection flow diagram	33

2.11	A spiral tether showing all minima found.	34
2.12	A snake-shaped tether showing all minima found.	34
2.13	A tether forming a knot showing 4 minima.	35
2.14	Close-up of the shortest minimum (right) prior to the collision.	35
2.15	The shortest minimum (right) during the collision. The two added dark lines represent the forces being applied at the contact points.	36
2.16	The shortest minimum (right) after the collision.	36
2.17	The minimum separation distances of the three shortest minima. The shortest of which showing a collision at time ≈ 0.21 seconds. Distances have been adjusted to account for the tether radius.	37
3.1	Two intersecting cylinders where a) shows α , the angle between them and b) shows the minimum separation distance d^* as well as ϕ_{\max} and x_{\max}	39
3.2	Cylinder cut at an angle where a) Section A-A shows an elliptical profile and b) geometrical information about an ellipse.	41
3.3	Some important geometrical information in contact region (mainly c_2).	41
3.4	Section C-C at a distance x from the common perpendicular.	42
3.5	Finding P_1 in terms of $(x_{\text{ellipse}}, y_{\text{ellipse}})$ using x_{\max} and α	43
3.6	Finding the magnitude of the major axis a using $z_{\max}(x)$ and x	44
3.7	A comparison of two intersecting: a) undeformed cylinders (dashed lines) b) deformed by contact pressure (solid lines).	45
3.8	Two intersecting cylinders showing a) their undeformed state and the plane A where the surface would meet under deformation and b) the deformed cylinders after deformation.	46
3.9	a) The volume of intersection showing the differential volumes. b) One of the differential volumes showing its spring dashpot representation.	48
3.10	Percentage difference between contact patch area of the Hertzian model and the contact patch area from the interference geometry.	49
3.11	Friction force versus displacement showing the change from sticking to sliding friction.	54
3.12	Friction force versus velocity showing the Stribeck effect.	55
3.13	The break-away friction force at the initiation of sticking as a function of time.	56
3.14	A tether self-collision showing the forces involved, tether penetration and their relative velocity. The thin solid lines represent the simulation with friction, the thick dashed lines represent the simulation with no friction, and the thin dashed lines are the % difference between the two.	58
A.1	Flow Diagram of the general simulation.	69
A.2	Flow diagram of the Contact Dynamics Package.	70

A.3 Flow diagram of how minima are found.	71
---	----

Chapter 1

Introduction

1.1 Motivation

Tethered systems are used in many applications, most notably in underwater exploration. In the operation of such systems, especially for underwater Remotely Operated Vehicle (ROV) operations, it is critical that human operators remain in the control loop. Currently, the only viable means of maintaining human control during underwater ROV operation is through the vehicle's tether, which also provides power and communications to and from the system. Due to the high cost of ROVs and their tethers, as well as the risks involved in their operation in the ocean, computer simulations are required for the training of pilots. Since the tethers have a dominant effect on the dynamics of an ROV system [2, 3], it is imperative that the forces applied to the tether are taken into account within the simulations. These include contact forces due to contact of the tether with the environment or with itself (*i.e.*, self-collision, see Figure 1.1). The tethered-system simulator being developed at the University of Victoria and the University of New Brunswick currently lacks this capability.

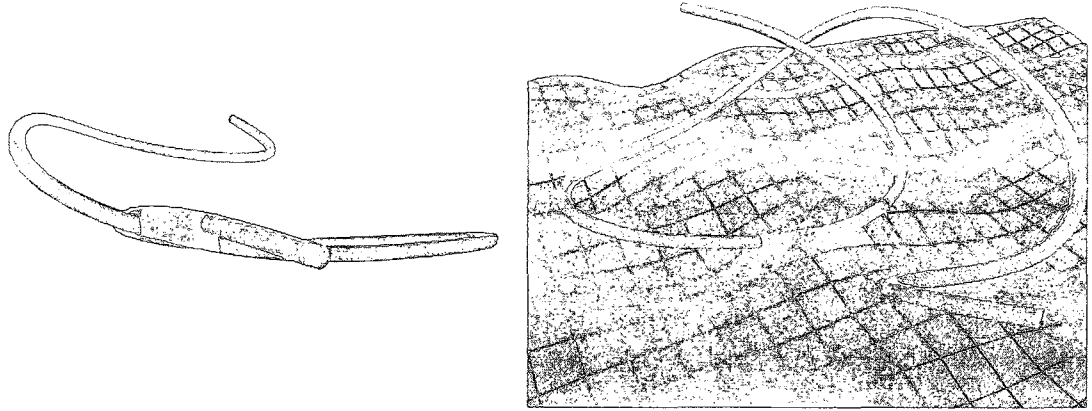


Figure 1.1. Examples of contacting tethers (contacting portions highlighted): a) tether-tether contact, b) tether-environment contact.

This is the chief motivation for extending the tether model to include a collision detection method as well as a contact dynamics model.

1.2 Literature Review

In the computer simulations of multi-body systems, it is often important to simulate the interactions of two or more objects that are in contact with each other. Contact dynamics simulations are important to a variety of fields such as haptics [4, 5], robotics simulations [6], Computer Graphics [7, 8], as well as vehicle crash simulations [9]. In such computer simulations, two problems arise which must be addressed: a) detecting collisions, and b) determining the forces involved so as to evaluate the objects' dynamics. Because these two problems are so fundamentally different from each other, they are generally regarded as two different problems that are co-dependent in solving the greater problem of simulating contacting objects. To calculate the reactions of

two objects in collision, these collisions must first be detected.

In reality, forces are generated between objects at the collision interface by their deformations, which keeps them separated. In computer simulations, a contact dynamics model describes how objects interact with each other during a collision.

1.2.1 Collision Detection

In general terms, in computer simulations, collisions are said to occur when any part of an object simultaneously shares the same space as part of another object. Many different ways of detecting collisions have been used and proposed. In situations where there are N_o objects in a scene that must be checked for collisions, verifying whether each object is in contact with every other one would require on the order of N_o^2 or more accurately $\binom{N_o}{2}$ collision checks or queries. This means that increasing the number of objects in the scene quadratically increases the number of collision checks required. This, in turn, quadratically increases the computational time required to process the scene. Therefore, as the complexity of a scene is increased, the simulation becomes bogged down quite rapidly.

In order to achieve fluid motion in a real-time computer simulation, such as a video-game, a frame must be rendered in less than $\frac{1}{25}$ s in order to achieve 25 frames per second (FPS) where 60 FPS is considered to be ideal for maximum perceivable fluidity of motion [10]. In haptic simulations, the emphasis on speed is even stronger as update rates of about 1 kHz can be required [4, 11].

To lighten some of the computational burden imposed by the collision detection systems, these algorithms generally employ spatial approximations (*e.g.*, Bounding Volumes), partitioning schemes as well as temporal strategies, where intersection

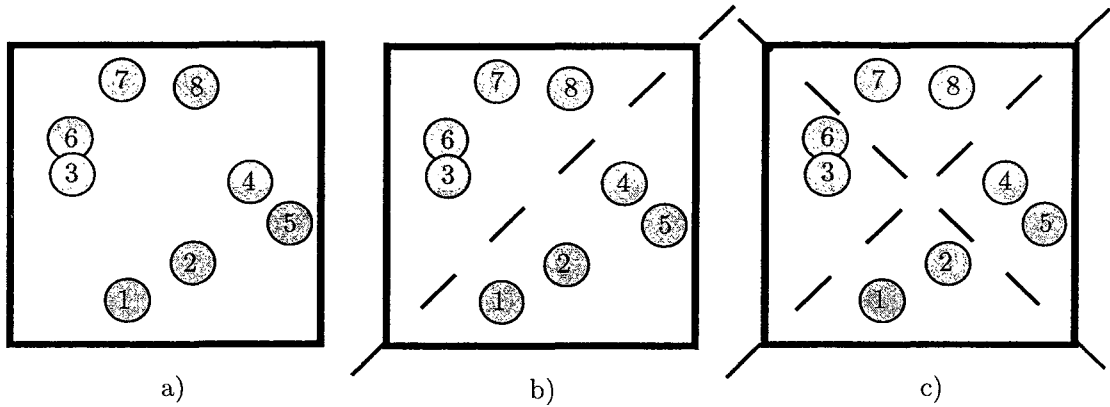


Figure 1.2. A simulation environment with 8 objects where the space is a) unpartitioned b) partitioned into two groups and c) partitioned into four groups.

checks are only performed when a collision is imminent [11, 12]. These methods, often referred to as pruning strategies, can lighten the computational load enough to allow for their use in some real-time applications [10].

To reduce the number of object pairs to be checked for collision, people have resorted to spatial partitioning techniques to determine which groups of objects are definitely not in contact with which other groups of objects. For instance, dividing the objects into two groups of $N_o/2$ objects each, where it is known that none of the objects from one group are in contact with the objects from the other group, will reduce the number of collision checks at least by half (Figure 1.2).

There exists a multitude of spatial partitioning schemes [10, 11, 12] such as dividing the space into grids, bounding volume hierarchies (BVH) and trees (octrees, k-d trees, BSP-trees), etc. Each of these methods attempts to reduce the number of

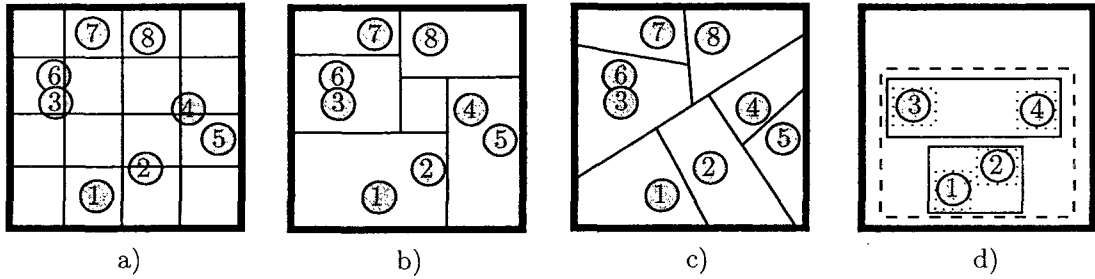


Figure 1.3. Examples of different space partitioning methods: a) uniform grids b) k-d tree c) binary space partitioning tree d) bounding volume hierarchy

collision checks by separating the objects in spatial compartments (Figure 1.3).

Bounding Volumes can be used to either approximate more complex objects or decompose them into simpler pieces [11]. This reduces the complexity of the collision checks but does not reduce the number of collision queries that must be performed. To do this, the bounding volumes are placed in a bounding volume hierarchy where collision checks between hierarchies generally need only run through the first few layers of the hierarchy if there is not collision. For instance, using the example in Figure 1.4, the first check is to see if the bounding boxes represented by solid lines are interfering. Since they are not, only the dotted lined bounding boxes of object A are checked for interference and similarly object B's boxes but no check is made of the objects within A to those within B. This required a total of 3 collision checks instead of $\binom{4}{2}$ or 6 checks if all four objects had to be checked. The savings can be even more dramatic for larger collections of objects.

There are a few different bounding volumes available and each have their own advantages and disadvantages but all have the same objectives: “provide inexpensive intersection tests, fit tightly around the objects they are bounding, can be efficiently

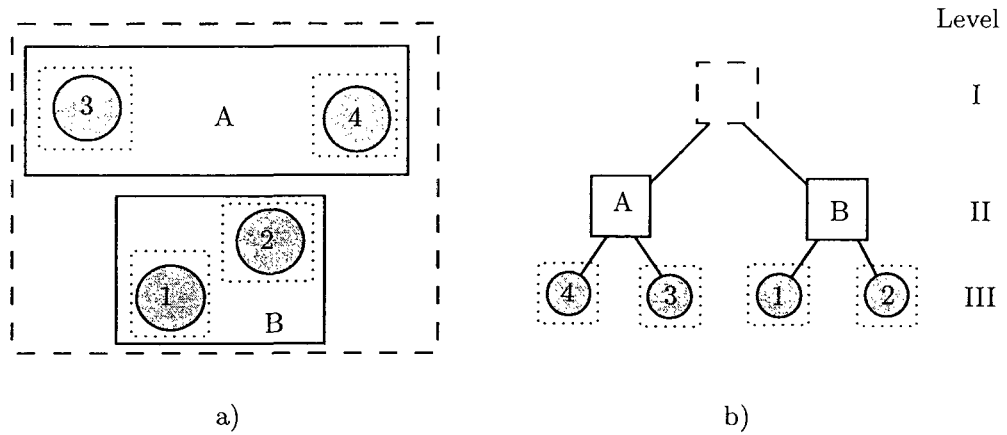


Figure 1.4. A similar bounding volume hierarchy example as Figure 1.3d and its tree representation.

created, easily transformed, and exhibit a small memory footprint” [10]. Some of the bounding volume types available include: spheres, axis-aligned bounding boxes (AABB), object aligned bounding boxes (OBB), Convex Hulls, and 8-DOPs (discrete oriented polytope) [10]. Convex objects, which are also useful bounding boxes, are particularly well suited for collision detection as there always exists a separating plane between two non-intersecting convex objects. As a result, concave objects tend to be broken down into smaller convex pieces. This allows for easy partitioning and efficient collision checks.

These bounding volume hierarchies can also be extended to deal with deformable object collisions and self-collisions [7]. However, their efficiency should be carefully investigated as hierarchy updates would be frequently required [7].

Recently, the graphical processing unit (GPU) has been used to detect collisions [13] and has been shown to even handle deformable and concave models [14]. Because such image-based methods are tested to the resolution of the buffers to which

the objects are rendered, such methods are generally approximate [10].

As computer simulations are discrete in nature, detecting collisions that occur in between time steps, and that are not in collision during a time step is not covered by the previously-mentioned methods. This effect is often called tunneling [10]. To solve the logistics of this, methods such as the use of swept volumes [10, 11, 15], have been introduced to handle continuous collision detection.

1.2.2 The Minimum Separation Distance

It is sometimes important in computer simulations to know how far an object is from other objects in the scene. There are a few reasons why the minimum separation distance could be important: a) in path planning it is necessary to determine how close the objects are to the environment and b) for collision detection or collision prediction algorithms [11].

Separation distance algorithms can be separated into two categories: optimisation-based and geometrically-based [11]. Both can also be separated again into two other categories: convex and concave methods.

The use of optimisation-based methods to find the minimum separation distance was first introduced in [16] and extended to find the interference distance in [17]. These methods treat the minimum separation distance as a function to be minimised, subject to constraints representing the bounds of the objects. However, these methods only handle convex polyhedra or quadratically-bound convex objects [18].

Geometrical methods, on the other hand, try to solve the minimum separation distance using geometrical properties of the objects. The GJK algorithm for instance, first proposed in [19], uses the Minkowski difference of two convex polytopes

to compute the minimum separation distance. Another such algorithm is the V-Clip algorithm [20], which makes use of Voronoi regions to determine the closest features between two convex objects from which the minimum separation distance can easily be found. Both of these are confined to convex objects or a concave object represented as the union of two or more convex objects.

Most distance determination algorithms deal with purely convex objects, so, in order to determine the separation distance of concave objects, they must first be decomposed into convex sub-objects [11]. In addition to generating additional fictitious object features, this uses more memory and requires pre-processing. The more convex subobjects that are generated to represent the concave object, the more separation distance checks must be processed [11]. That being said, obtaining the minimum separation distance between convex objects is a simple problem as there is always a single solution that can easily be found using the methods mentioned earlier.

Some optimisation methods have been developed to handle non-partitioned concave objects. For example, the algorithms proposed in [21] and [22] used genetic or simulated annealing algorithms to find the approximate minimum separation distance. The algorithm in [22] was later extended to find the exact minimum separation distance in [23].

1.2.3 Contact Dynamics

After a collision is detected, a model to calculate the contact forces is to be used to determine the forces involved. For increased fidelity of the contact model, forces that are both normal and tangent (*i.e.*, friction) to the contact plane should be taken into consideration.

Normal Force Models

Normal force contact dynamics models can be categorised as either discrete or continuous models [17, 24]. Discrete methods assume that contact is instantaneous. They are an attempt at modelling the result of a collision rather than the collision itself. The most well-known discrete contact models are arguably the Newtonian impulse-momentum models or coefficient of restitution models [24]. Discrete models cannot be easily extended to multi-body contacts, moreover, the inclusion of friction for these models might violate energy conservation principles [17].

On the other hand, continuous/compliant models integrate contact forces over time. This allows contacting objects to “deform” under pressure as is the case in real contact situations. The three main continuous contact models are: Hertzian Contact, spring-dashpot methods and finite element methods [17]. The Hertzian contact model [25], based on elasto-static theory, is limited to elastic deformation of smooth objects where there would be a common surface normal vector between any possible contact interfaces [17]. For cases where the simulated objects are represented by polygonal objects, the contacting surfaces are rarely smooth (unless the polygonal surfaces are highly refined). As a result, the Hertzian model is ill-suited for these cases. In some computer simulations, modelling pressure deformations may not be worthwhile due to the high computational cost associated in computing the deformation within the contact region which can be done using a finite element method. Fortunately, it can generally be assumed that the interfering geometry is a good approximation for the deformations involved [4, 6, 17, 24, 26]. It is important to note that the volume of intersection is an extension of the spring-dashpot model as will be shown in this work.

The weaknesses of the spring-dashpot model, and hence the volume of interference model, are: a) the discontinuous contact forces at impact, b) the possible exhibition of tensile forces when contacting objects move away from each other and c) the lack of a coefficient of restitution that is independent of velocity [17].

These misgivings can most likely be solved by: a) using fine time steps during collisions (or more efficient integrators) b) setting the contact force to be 0 when it is in tension and c) using a coefficient of restitution model that is not solely based on velocity.

Calculating the volume of intersection, being the most important part of this model, can be computationally intensive. The model's useability is affected by how fast the volume can be calculated or approximated. In [26], the approximated interference volume is calculated as the contact patch area times the interpenetration distance. In [6], a geometrical shape factor was included in the formulation to more accurately approximate the interference volume. Finally, [4] and [17] use duality transformations and convex hull approximations to calculate the interference volume. Unfortunately, these methods are limited to intersecting convex polyhedral objects.

In regards to the field of computer graphics, where engineering accuracy is not necessarily required, a different breed of geometrically-based contact models have been used. In [27], penetration distances of vertices of one object to faces of the other are used to determine the contact forces [28, 29].

Friction Models

Friction models can be broken down into two general groups: static models and dynamic models [30]. Notable static friction models include the classical models like Karnopp's model and Armstrong's model.

The classical models are a group of models that each take into consideration a specific behaviour of friction. For example, the first is the Coulomb friction model which states that the friction force is equal to the normal force times a friction factor. The viscous friction model, extends the Coulomb model by saying that, depending on the state of lubrication at the contact interface, the friction force is linearly related to the velocity. Many such models have attempted to also address stiction as well as the so-called Stribeck effect¹ [30].

The Karnopp model was introduced to deal with issues of zero velocities detection and stick/slip transition by using a dead zone for zero velocities [30]. That is, if the velocity is below a certain tolerance level, the object is regarded to be in a zero-velocity state.

Armstrong [31] extended the classical method to account for the time dependencies of stiction and the Stribeck effect. However, this method neglects handling pre-sliding displacements [30].

Static models only consider the current state of friction therefore neglecting some aspects of friction that are time dependant. Dynamic models have a memory of the state of friction from the previous time step. The first such model is the Dahl model [30] which borrows from the stress-strain curve from solid mechanics. Unfortunately, it neglects the Stribeck effect as well as stiction [30]. The Stribeck effect was taken into account in the Bliman/Sorine model though it was not without its

¹The Stribeck velocity is the velocity at which the friction is in 37% transition from sticking to sliding. It is the characteristic drop in friction at low increasing velocities that occurs during the transition from boundary lubrication to full fluid lubrication. It was first observed by R. Stribeck in 1902 [30]. The Stribeck curve is defined as friction force as a function of velocity for constant velocities.

flaws [30, 32].

The bristle model [33], which tries to model the deflection and snapping of asperities at the contact interface, captures some interesting phenomena of friction but it is not very efficient with a complexity of $O(N_{bristles})$. The authors of [33] also presented the reset integrator model which is a computationally efficient bristle model.

Finally, [32] introduced the so-called LuGre model which models lubricated surfaces, the Stribeck effect, time-dependent break-away force, frictional lag as well as a constant time complexity.

The LuGre model was extended by [26] to consider 3 dimensional vectorial friction forces, and later in [34, 35, 36] accounted for both rolling and spinning friction.

The effects of hydrodynamic and hydrostatic lubrication are usually negligible in low viscosity fluids such as air but could become rather important for higher viscosity fluids such as water and oils [37]. That is, when a fluid is squeezed in between two surfaces, the pressures can become high enough to maintain a separation between the surfaces and hence reduce the friction caused by asperity contact.

The LuGre model takes hydrodynamic lubrication (see Figure 1.5) into account but hydrostatic squeeze films are not modelled.

1.2.4 Other Simulations

In [38], a knot-tying simulation was proposed using a bounding volume hierarchy which is used to achieve real-time collision detection. Unfortunately, the simplistic tether and contact models within [38] limit the method to simplistic dynamics simulations. Another real-time tether dynamics model was developed by [39] for an ROV simulator, though because of its corporate potential, no in-depth details about the

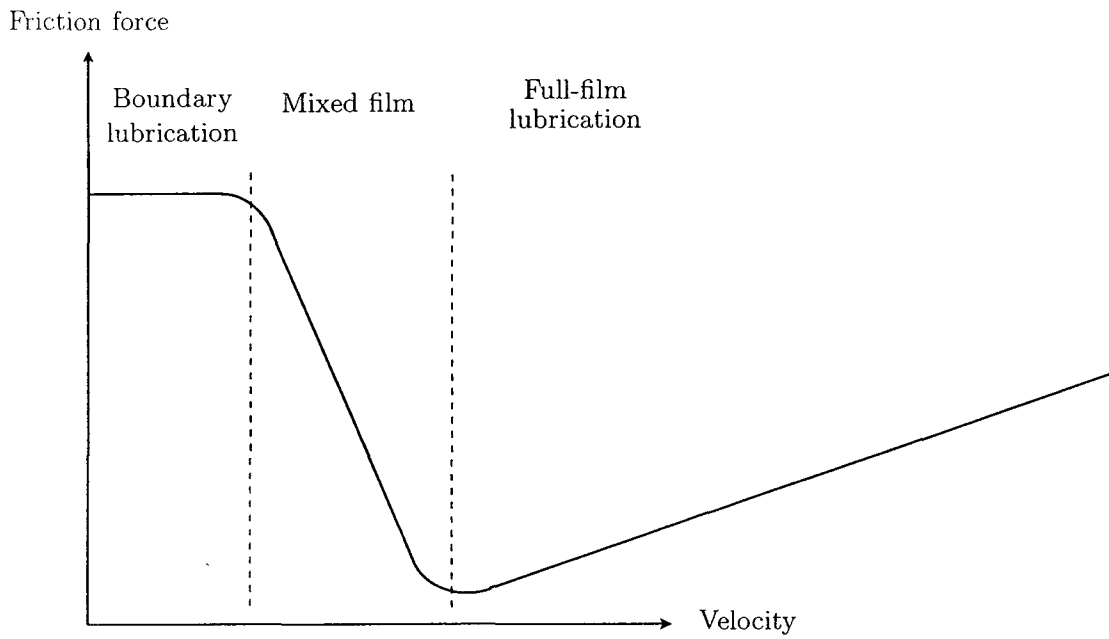


Figure 1.5. Friction force versus displacement showing the change from sticking to sliding friction [1].

methods are publicly available to the author’s knowledge.

The motivation for this work was first introduced in [40]. It covered detecting collisions between the tether and the environment but not tether self-contact.

The tether can also be seen as a kinematic chain or an articulated model. For these, [41] presents a method of detecting self-collisions that uses bounding volumes and [15] follows a similar approach but extends the use of bounding volumes to the use of swept volumes which handles continuous collision detection.

In [7], the authors discuss the use of stochastic methods, a family to which some optimisation algorithms belong, and mentions the use of temporal coherence to keep track of colliding regions in [42]. Also, in [43], the use of a stochastic method for finding self-collisions in strand-like structures is proposed.

In [27], wire-like objects are simulated as an array of Cosserat rods². They are approximated by axis-aligned bounding box hierarchies for collision detection which guarantees them detection of all collisions but can get computationally intensive in multi-collision situations. The forces were calculated on a penalty method similar to volume of intersection and penetration depth methods adapted from [28, 29].

1.3 The Physical Simulation and the Tether Model

The tether model, described in [44] and in more detail in [45], is used in this work. The tether is modelled as an assembly of finite elements that are based on the twisted cubic spline geometry described in [46]. The finite element tether model uses a lumped mass approximation at each tether node. The mass and stiffness matrices of the

²Cosserat rods are based on elastica theory, developed by Euler, which allows for very large scale elastic deflections of structures.

model are derived from the continuum equations using a weighted residuals approach. The model also includes a complete accounting of the axial, bending and torsional mechanics.

As discussed in [45], the lumped-mass approximation is synergistic with the use of twisted spline elements. This allows the use of a full cubic geometry description of the tether given only the node positions. However, a consequence of the approach is that external effects, including contact forces, must be discretisable into point loads applied at the node points.

To resolve this issue, if a collision is detected in between existing nodes, a new node can be added at the contact point to impose forces accurately at the contact region. The model described in [45] includes a calculation of residual forces that can be used to gauge the necessity of adding new node points during a simulation.

At the core of most realistic physical computer simulation is an integrator. Physical simulations normally require numerical methods to solve sets of differential equations defining the equations of motion derived for the objects being simulated [47]. Arguably the most well known such method could be Euler's method which is simply an application of Taylor's theorem with the order 2 [47]. Since Euler's method only considers the first and second derivatives of position, *i.e.*, velocity and acceleration, a large amount of truncation error can occur. It is the lack of stability and accuracy that makes it a poor choice for most practical applications. To reduce their inherent error, the simulation time step is generally reduced to achieve a desired accuracy. However, this can greatly affect the computational time.

As an alternative, most engineering applications resort to higher-order methods such as the well known Runge-Kutta methods. The integration method used in [44] is the Runge-Kutta 4/5. This method is simply the RK4 method [48] with a fifth

order term calculated which is considered to be the inherent error of RK4. This error is used to determine an appropriate step size. Such a method is an attempt to keep integration error low while minimising computational time [48].

1.4 Thesis Objectives

The objective of this work, as described in section 1.1, is to develop efficient methods for detecting and resolving tether self-collisions in ROV tether simulations. It is also important that the methods in question are as accurate as possible. The objective then, is to develop methods to detect and resolve tether-self collisions while finding a proper balance in accuracy and efficiency.

1.5 Thesis Overview

In Chapter 1, this work is introduced and a review of the literature is provided. Chapter 2 then presents a continuous collision detection method. The use of a two-stage optimisation method (discrete and continuous stages), for detecting tether self-collisions, is discussed. To ensure detection of all collisions, an efficient method for detecting collisions missed between time steps is introduced using some simple vector manipulations.

In Chapter 3, methods for determining the interference volume between two skew cylinders, as well as the contact surface area are presented. These are, in turn, used to calculate the normal and tangent contact forces. The volume of intersection method is shown to be a derivative of the Winkler elastic foundation model and compared against Hertzian theory of general contact. Friction is modelled to include effects

such as dwell-time stiction dynamics, stiction/sliction transitions, Stribeck effect, frictional lag and fluid or dry lubrication. Some numerical examples are presented to demonstrate the validity of the models and methods.

In Chapter 4, conclusions are drawn as to the effectiveness and accuracy of this work as well as discuss some of the topics that should be researched further.

Readers may refer to Appendix A for a simplified flow diagram of the contact dynamics and how its sub-components, described in this work, fit together.

Chapter 2

Self-Contact Collision Detection of Strand-Like Objects

Determining self-collisions is not necessary when dealing with rigid bodies as they cannot change their shapes in order to make self-contact. On the other hand, when dealing with flexible objects, such as ROV tethers, contacts with itself must be determined and resolved.

Fortunately, the nature of the tether model used here greatly simplifies the process of detecting collisions. Since the tether is considered to be circular in profile with constant cross-sectional area over its entire length, it can be substituted by a spline with an associated cross-sectional radius. Hence, the minimum separation distance queries can be done on the spline rather than the tether surface which is then compensated for its inherent radius.

2.1 Finding the Minimum Separation Distance

The minimum separation distance between two straight lines can be obtained by the length of the line segment perpendicular to both and intersecting both. This can be extended to the curved tether segments by saying that the line segments locally representing the minimum separation distances are always perpendicular to the gradients of the tether at each closest point on the tether ¹. Therefore, an approximate and simplistic way of detecting collisions in long flexible objects of circular cross-section would be to discretise the tether into a finite number of segments and determine the distances between all possible pairs of tether segments represented by their centre lines. Unfortunately, being an exhaustive method this would result in N^2 checks, where N is the number of segments, because of the fact that all possible segment pairs represent $N \times N$ distance queries (assuming a tether with $N + 1$ nodes). Using this method, contacts between two tether segments can be identified when the length of the minimum separation distance segments are less than the sum of the tether segment radii.

As the number of sample points is increased to achieve higher simulation accuracy, the number of collision checks increases quadratically which can quickly become too large of a computational burden. To avoid this “ N -body” effect, the method proposed here tracks pairs of tether segments which are closest to each other (therefore in contact or near contact) and only performs contact checks on those. Only considering pairs for which a collision is imminent will save numerous collision checks therefore completely ignoring the vast majority of possible contact pairs [49, 50]. This is known

¹At a point s along the tether, the gradient of the tether can be represented by a line tangent to the tether at point s .

as a spatial pruning strategy. If the velocities and accelerations of those minimum separation distance pairs is taken into consideration, it is possible to predict whether or not those pairs may be in contact in the next few time steps. This is known as a temporal pruning strategy [50].

To perform such pruning, the minimum separation distances between the tether and itself can be found by treating the problem as an optimisation problem. Two free moving points constrained to move along the tether are allowed to move in order to minimise the distance between them. This concept is similar to the local optimisation algorithm proposed in [22], which is based on the distance determination algorithm described in [18]. As opposed to having an infinite search space and a large number of constraints as in [18], the method in [22] uses a discrete search space and the optimisation problem is unconstrained.

2.1.1 Step 1: Identifying the Closest Regions

The algorithm proposed in [22], a multi-point search algorithm that has been later named MLSDist (multi local search for minimum separation distance) is used to roughly identify all of the minima. In MLSDist, a number of node pairs² are randomly selected from the tether nodes. Each random pair is locally optimised using a simple greedy hill-climbing method that moves each of the nodes in the pair to its neighbouring nodes with the goal of minimising the Euclidean distance between them [22].

²The tether is made up of nodes and spline segments. A node pair is a pair of nodes. An optimised node pair is the case where moving any one of the nodes in the pair to either of its neighbours does not reduce the distance between them.

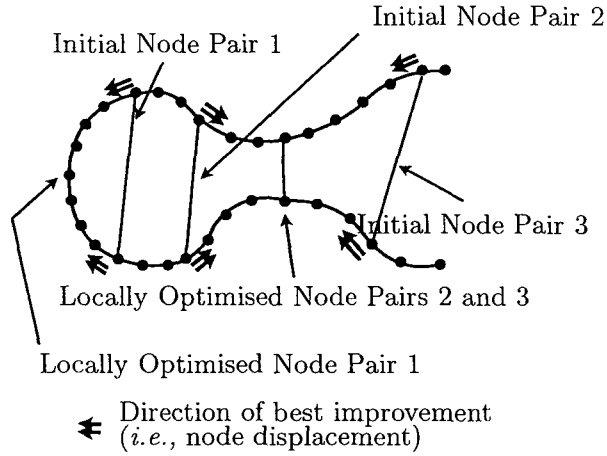


Figure 2.1. Example of MLSDist with three different starting node pairs.

After this first minimisation step, only a group of locally optimised node pairs is left (see Figure 2.1). The pairs with the separation distance under a certain threshold are said to be the pairs which are most likely to be close to a collision. This saves the simulation from performing many redundant calculations.

The minimum separation distance between two portions of the tether is only useful when the pair of locally optimised nodes is not the same node as is the case for the optimised node pair 1 of Figure 2.1.

The objective function that is being minimised here is the square of the Euclidean distance between two points $\mathbf{P}(s_1)$ and $\mathbf{P}(s_2)$, where s_i is the distance along the tether where point i is located. That is, the optimisation problem is

$$\min_{s_1, s_2} \quad d = (\mathbf{P}(s_1) - \mathbf{P}(s_2))^T (\mathbf{P}(s_1) - \mathbf{P}(s_2)) \quad (2.1)$$

$$s.t. \quad s_1 \neq s_2 \quad (2.2)$$

Note that d is the square of the Euclidean distance between points $\mathbf{P}(s_1)$ and $\mathbf{P}(s_2)$ as the points that minimise d also minimise \sqrt{d} as the separation distance is always

positive. To calculate d , each point $\mathbf{P}(s_i)$ can be broken down into its independent Cartesian components as $\mathbf{P}(s_i) = [X(s_i), Y(s_i), Z(s_i)]^T$, for $i = 1, 2$. Equations to determine $X(s_i), Y(s_i)$ and $Z(s_i)$ are the cubic spline equations described in [45], which are a function of the node location as well as the tether’s curvature at each of those nodes.

Fortunately, the solution space for this objective function (equation (2.1)) is two dimensional, that is, the minimum distance is dependent only on s_1 and s_2 . Figure 2.2 shows a case where the separation distance between a tether and itself is a function of s_1 and s_2 which results in a relatively simple 3D surface with several peaks and valleys. Note that, as expected, the surface shown in Figure 2.2 is symmetrical in $s_1 = s_2$ as exchanging the values of s_1 and s_2 in equation (2.1) results in the same value of d . Also note that $d = 0$ when $s_1 = s_2$ which is also expected as $P(s_1) = P(s_2)$ if $s_1 = s_2$.

The intended use of the distance and collision detection algorithm is for dynamic environments (*i.e.*, ROV simulations) where the distance queries are performed at every time-step. For this reason and to achieve better computational efficiency in any future trials of MLSDist (*e.g.*, the new time step), current optimal node pairs are tracked and used as some of the initial trial point pairs for MLSDist in the following time-steps. That is, those closest node pairs in time-step i are likely to remain near or at a minimum in time step $i + 1$ if the objects’ geometry remains fairly similar. This is generally the case for high inertia or slow motion systems such as most underwater applications as well as for systems whose simulations allow for a high update rate.

Once all closest node pairs are located, the algorithm is left with N_{min} locally-optimal node pairs. However, since the tether is discretised, MLSDist is only able to minimise the distance by picking between the nodes themselves. As a result, the

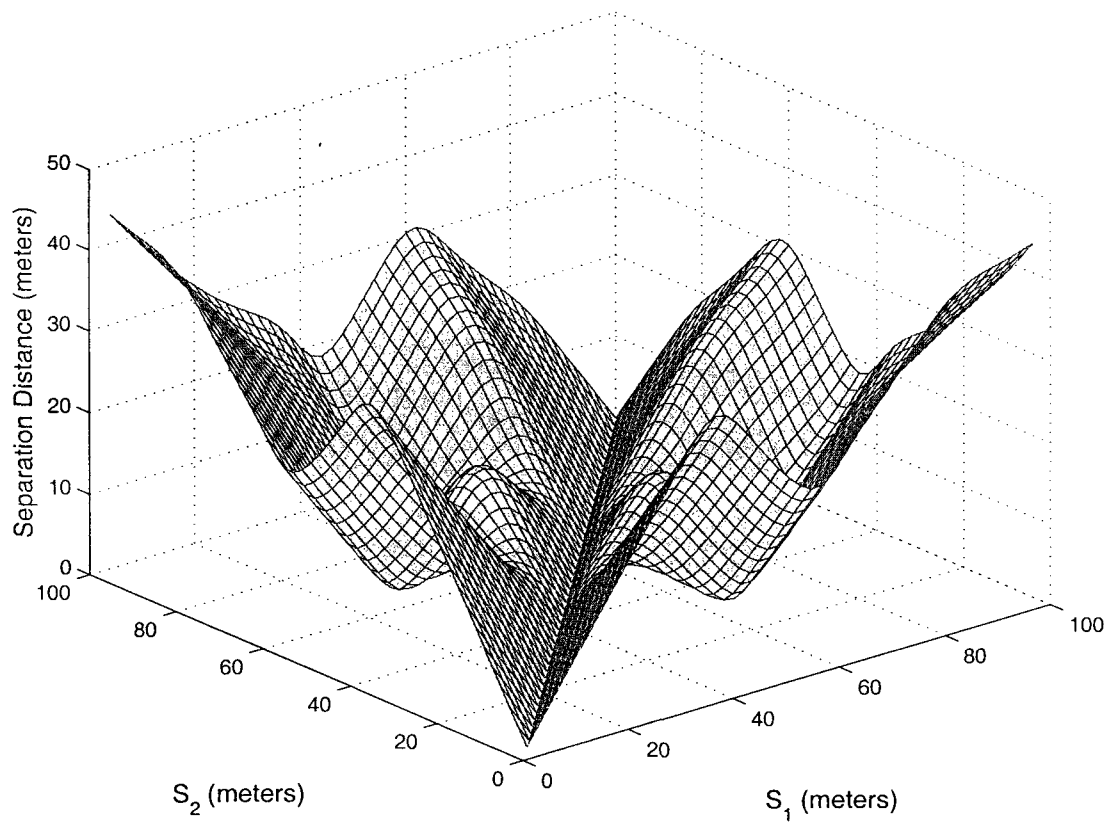


Figure 2.2. The 3D surface of the distance between a point on the tether at s_1 and a point on the tether at s_2 .

solution is only as accurate as the size of the segment lengths.

2.1.2 Step 2: Determining the Exact Minimum Separation Distance

To achieve a more accurate solution, the minimised node pairs are used on a second optimisation stage using the two cubic spline segments that lie on each side of the selected node pairs (see Figure 2.3). Taking advantage of the equations of the twisted cubic spline, analytical expressions that provide the gradient and Hessian information can be obtained as a function of 2 point locations along the tether: s_1 and s_2 . The equations which dictate the position and orientation of the continuous tether between two nodes are:

$$\mathbf{P}(s) = \mathbf{P}^{m-1}\phi_1 + \mathbf{C}^{m-1}\phi_2 + \mathbf{P}^m\phi_3 + \mathbf{C}^m\phi_4 \quad (2.3)$$

where \mathbf{P}^m is the position vector in Cartesian coordinates of the adjacent node upstream of s , \mathbf{P}^{m-1} is the position vector of the adjacent node downstream of s , \mathbf{C}^{m-1} is the vector of curvature of the cubic spline of the adjacent node downstream of s , ϕ is a shape function used to interpolate between the two nodes and L_u is the undeformed tether segment length between nodes m and $m - 1$.

$$\phi_1 = \frac{s_m - s}{L_u} \quad (2.4)$$

$$\phi_2 = \frac{1}{6}(\phi_1^3 - \phi_1)L_u^2 \quad (2.5)$$

$$\phi_3 = \frac{s - s_{m-1}}{L_u} \quad (2.6)$$

$$\phi_4 = \frac{1}{6}(\phi_3^3 - \phi_3)L_u^2 \quad (2.7)$$

Using these equations and equation 2.1, the first and second derivatives with respect to the positions s_1 and s_2 at s_1 and s_2 can be obtained in closed-form. These

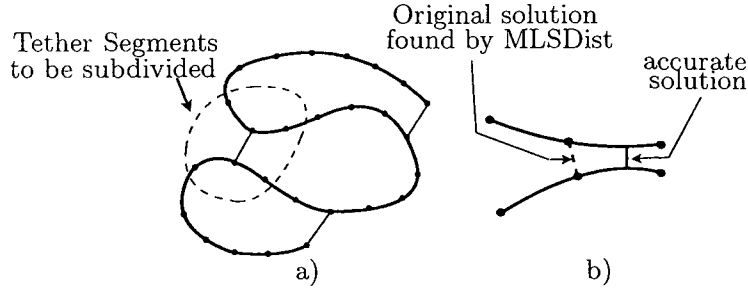


Figure 2.3. The tether with a) minimum distances found. b) exact minimum found on the cubic spline segment.

are:

$$\nabla d = \begin{bmatrix} \frac{\partial d}{\partial s_1} \\ \frac{\partial d}{\partial s_2} \end{bmatrix} \quad (2.8)$$

$$\nabla^2 d = \begin{bmatrix} \frac{\partial^2 d}{\partial s_1^2} & \frac{\partial^2 d}{\partial s_1 \partial s_2} \\ \frac{\partial^2 d}{\partial s_2 \partial s_1} & \frac{\partial^2 d}{\partial s_2^2} \end{bmatrix} \quad (2.9)$$

where ∇d and $\nabla^2 d$ represent the gradient vector and Hessian matrix, respectively, of the square of the Euclidean distance between the point pair (see equation (2.1) and refer to Appendix B for their derivations). Having this information in closed-form allows the use of some fast-converging continuous optimisation methods such as Newton's method [51]. The objective function is not necessarily positive definite throughout the search space as is required by Newton's method which uses a quadratic approximation of the non-linear solution space. However, with a few modifications to Newton's method, these problems can be overcome³ [51].

Due to the cubic spline representation of the tether and by maintaining segment lengths less than 2π times the minimum possible radius of curvature of a simulated

³Please see [51] for more explanations of modifications and algorithms.

tether, which is a tether's standard specification, an assumption can be made that there is a maximum of nine separation distance local minima between the tether segments neighbouring a node pair (Figure 2.4a). Finding all 9 possible local minima can be guaranteed by using the nine initial trial pairs in figure 2.4b.

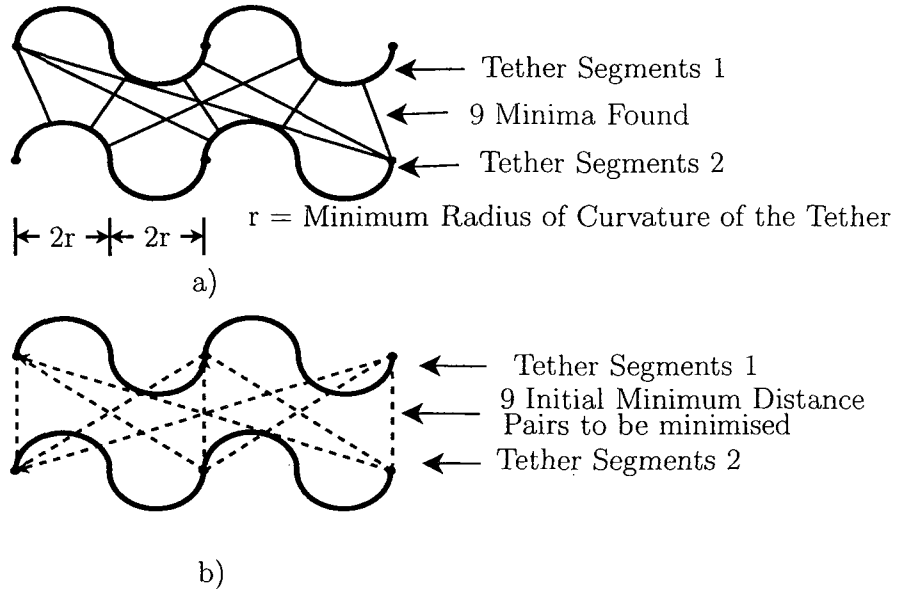


Figure 2.4. a) Nine minimum separation distance pairs. b) Initial pairs prior to being minimised.

Running this local optimisation the optional 9 neighbouring-node pairs of all pairs returned by MLSdist results in a list of point pairs that are closest to each other. Once found, they can be used to determine whether or not a collision is occurring or has occurred.

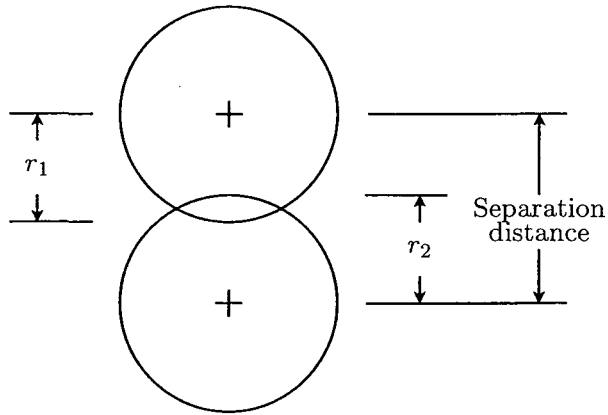


Figure 2.5. A collision that is detected by verifying if the separation distance is less than $r_1 + r_2$.

2.2 Defining Collisions

The minimum separation distance algorithms described in the previous section provides the ROV operators with an idea of possible tether entanglements and most importantly, the methods serve as a precursor to detecting collisions. In a static sense, the existence of collisions can be determined simply by continuously determining the separation distance between the tether centre-lines. That is, if the minimum separation distance between tether centre-lines is less than the sum of the radii of the tether segments, then, the tether is said to be in a self-collision situation (Figure 2.5).

However, during a simulated ROV maneuver the tether segments could move at a velocity such that the tether segments will completely pass through one another during a time step interval, see Figure 2.6. Therefore, the simple static algorithm is not sufficient for collision detection in dynamic environments.

To detect collisions in dynamic environments, it is necessary to take into account

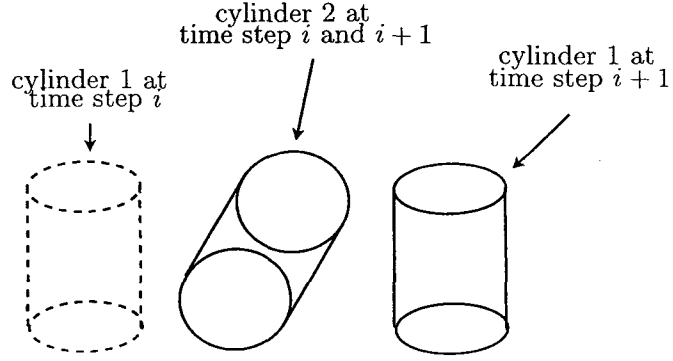


Figure 2.6. Two cylinder segments crossing each other between time steps.

where the segments were in the previous time step and determine whether a collision occurred while transiting between consecutive time steps. To overcome this issue, the closest node pairs are tracked from frame to frame and checked for such inter-frame collisions.

Using the position of the first point in the point pair (*i.e.*, $\mathbf{P}(s_1)$) in time steps i and $i + 1$ relative to the second node on the pair (*i.e.*, $\mathbf{P}(s_2)$), a difference vector is found between the current and last frames (see Figure 2.7a). To avoid collision between time steps, a minimum angle β_{min} is needed between the relative position vector $\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i$ and the relative displacement vector $(\mathbf{P}(s_1)_{i+1} - \mathbf{P}(s_2)_{i+1}) - (\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i)$. The minimum angle β_{min} is calculated as:

$$\beta_{min} = \sin^{-1} \left(\frac{r_1 + r_2}{|\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i|} \right) \quad (2.10)$$

where r_j is the radius of the tether at s_j . Since in self-collision situations the tether is considered to have constant cross-section, $r_1 = r_2 = r$, and thus, equation 2.11 becomes:

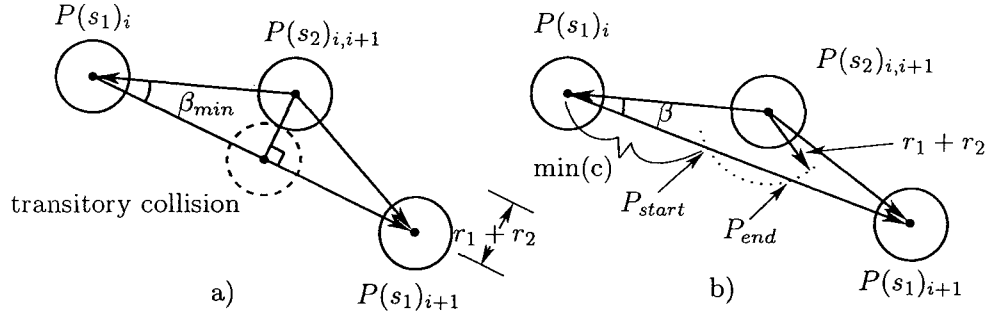


Figure 2.7. a) Defining β_{min} to determine if a collision has occurred between time steps i and $i + 1$. b) Defining β and P_{start} to determine if a collision has occurred between time steps i and $i + 1$.

$$\beta_{min} = \sin^{-1} \left(\frac{2r}{|\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i|} \right) \quad (2.11)$$

To determine possible contact, the actual angle β between the relative displacement of point $\mathbf{P}(s_1)$ between frames i and $i + 1$ and the shortest distance vector needs to be calculated. That is,

$$\beta = \cos^{-1} \left(\frac{(\mathbf{P}(s_1)_i - \mathbf{P}(s_1)_{i+1})^T (\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i)}{|\mathbf{P}(s_1)_i - \mathbf{P}(s_1)_{i+1}| |\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i|} \right) \quad (2.12)$$

where $|\ast|$ denotes the Euclidean norm of \ast .

If the actual cone half-angle β (Figure 2.7b) is greater than β_{min} , then no collision is said to have occurred between frames i and $i + 1$. This would mean that $\mathbf{P}(s_1)$ is moving relative to $\mathbf{P}(s_2)$ at an angle greater than the minimum angle required to initiate a collision (given a large enough travel distance).

If, however, β is less than β_{min} , then a transitory collision has possibly happened. Solving the following vector equations for c will determine this:

$$(\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i) + c \frac{(\mathbf{P}(s_1)_{i+1} - \mathbf{P}(s_2)_{i+1}) - (\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i)}{|\mathbf{P}(s_1)_{i+1} - \mathbf{P}(s_2)_{i+1}) - (\mathbf{P}(s_1)_i - \mathbf{P}(s_2)_i|} = \mathbf{P}_{start} \quad (2.13)$$

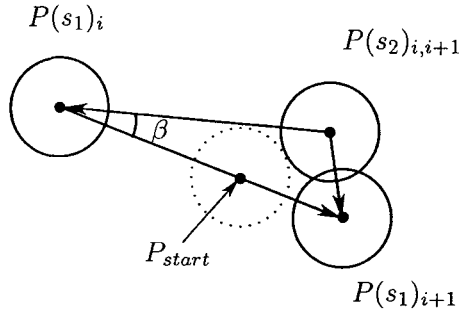


Figure 2.8. Case where the time step was too large and the tether travelled too far to be considered accurate. The angle β can be used to judge the accuracy of the collision.

and

$$|\mathbf{P}_{start} - \mathbf{P}(s_2)_i| = r_1 + r_2 \quad (2.14)$$

where \mathbf{P}_{start} is the position of $\mathbf{P}(s_1)$ at the start of a collision on a travel path of angle β . Solving the system of equations shown in equation (2.13) for c provides two possible solutions, one corresponding to \mathbf{P}_{start} and the other to \mathbf{P}_{end} (see Figure 2.7b). A collision is said to have occurred if $|\mathbf{P}(s_1)_{i+1} - \mathbf{P}(s_1)_i| > \min(c)$ considering only positive values of c . This method is simple and effective at detecting collisions in between frames for cylindrical objects.

There is a problem that shows up when there is no collision in frame i but one in frame $i + 1$, and the time step or relative displacements are so large that the tether has penetrated too far to provide a force in a proper direction as seen in Figure 2.8. This is easily mitigated by ensuring that the following equation is satisfied:

$$|\mathbf{P}(s_1)_{i+1} - \mathbf{P}(s_1)_i| < \left| \mathbf{P}_{start} + \frac{\mathbf{P}_{end} - \mathbf{P}_{start}}{P_{fineness}} \right|. \quad (2.15)$$

where $P_{fineness}$ is a user-defined scalar used to inhibit large interpenetration changes. It is recommended that $P_{fineness}$ be chosen to be larger than 5-6.

It is possible that in between time steps, the tether centre axes will cross each other

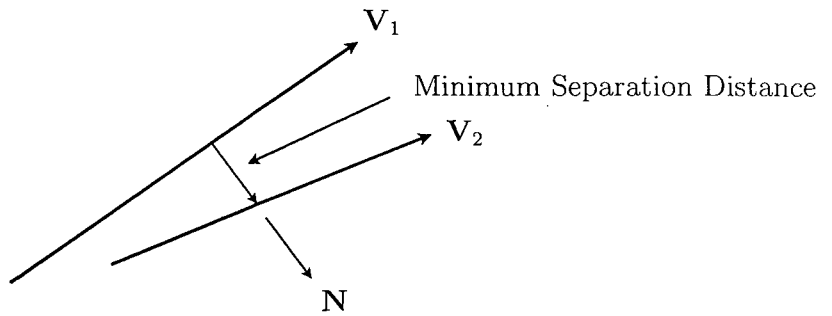


Figure 2.9. The minimum separation distance between two skewed lines and the line normal to both.

instantly flipping the contact force's direction. Using some simple vector manipulation, one can determine whether or not the two centre lines have crossed. Representing the first segment as vector \mathbf{V}_1 and the second segment as vector \mathbf{V}_2 , one can find a vector \mathbf{N} perpendicular to both by using the vector cross-product (see Figure 2.9):

$$\mathbf{N} = \mathbf{V}_1 \times \mathbf{V}_2.$$

This vector and the line that represents the minimum separation distance will always be parallel to each other as they are always perpendicular to the vectors \mathbf{V}_1 and \mathbf{V}_2 . Because the direction of the minimum separation distance is dependant on the relative positions of \mathbf{V}_1 and \mathbf{V}_2 and \mathbf{N} is not, the vectors \mathbf{V}_1 and \mathbf{V}_2 will have crossed if the dot product of \mathbf{N} and the minimum separation distance vector changes from positive to negative or vice-versa.

A problem arises if the angle between \mathbf{V}_1 and \mathbf{V}_2 changes from positive to negative or vice-versa when the cross-product also changes polarity. To overcome this issue one must insure that \mathbf{N} did not inverse. If the dot product of \mathbf{N} from the time step i against that of time step $i - 1$ returns a negative value, \mathbf{N} has changed direction and this must be taken into account while determining if the minimum separation

distance has also changed.

If it is found that a collision has occurred in between time steps rather than at a current time step or if their centre lines have crossed, the simulation is also to ignore the new tether position, reduce the time step and retake the current iteration again (see Figure 2.10 for a flow diagram describing this process visually).

In hopes of achieving a higher simulation stability, it is recommended that the time step be set proportional to the tether interpenetration during collisions in hopes of achieving a stability in the contact/environment/internal force balance and prevent centre-crossing or large impulses.

2.3 Numerical Examples

Here, three numerical examples are given where the first two are static tests and the third is a dynamic one. First, to test the algorithm's capabilities at finding multiple local minima, a special case is tested. When the tether is parallel with itself, as is the case with a perfectly spiralled tether, there exists an infinite number of minima. Figure 2.11 shows a perfectly spiral tether with all minima found. There is only a discrete number of minima found because only N minimum pairs was generated and $N_{min} - q$ pairs were returned where q is the number of invalid pairs formed (*i.e.*, the pair of nodes are identical). In most cases, however, there exists a finite number of minima which are all effectively found as can be seen in the snake shaped tether shown in Figure 2.12.

The results of a dynamic example are shown in Figures 2.13 to 2.17 where a tether is being tied into a knot while a weight is attached to the free end of the tether to tighten the knot. At the instant shown in Figure 2.13, there are four current

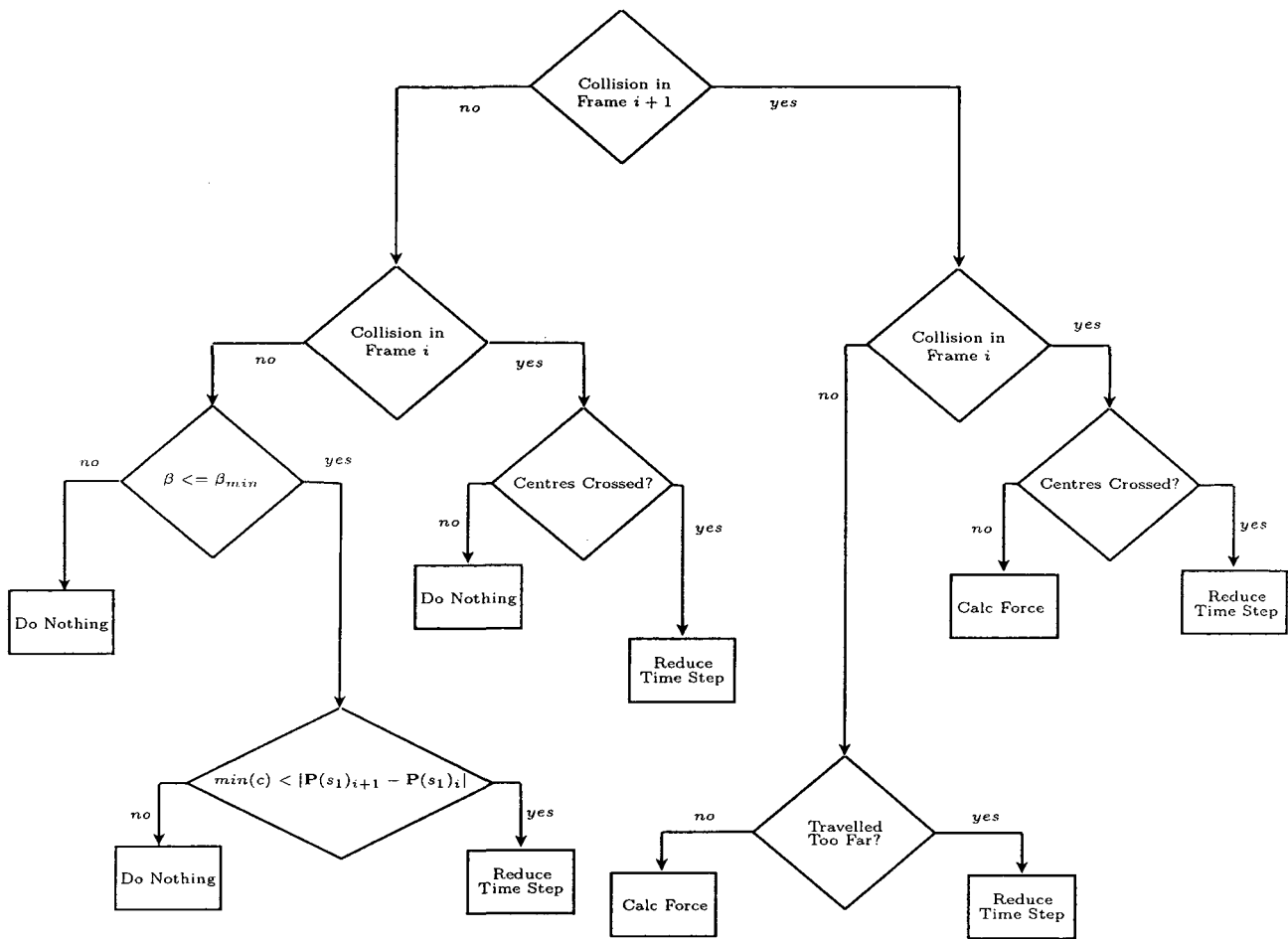


Figure 2.10. Continuous collision detection flow diagram

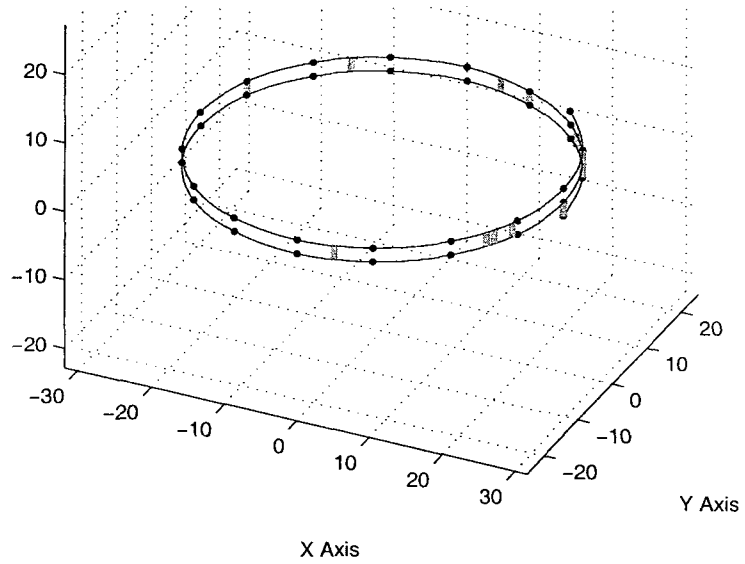


Figure 2.11. A spiral tether showing all minima found.

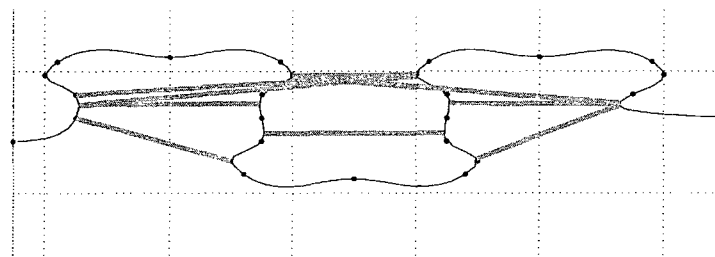


Figure 2.12. A snake-shaped tether showing all minima found.

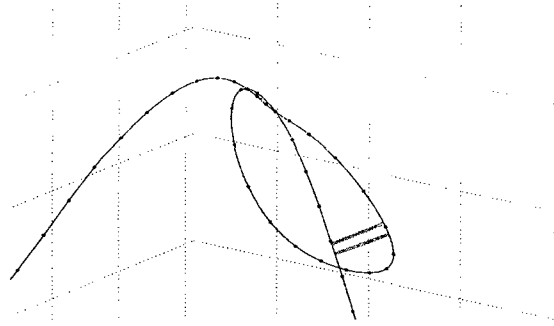


Figure 2.13. A tether forming a knot showing 4 minima.

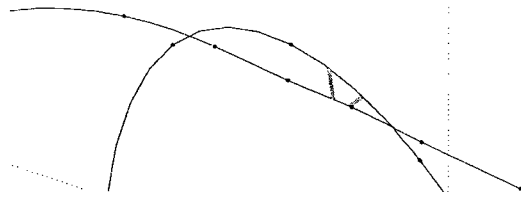


Figure 2.14. Close-up of the shortest minimum (right) prior to the collision.

separation distance minima, two of which, located at the top centre of the scene, are hardly visible due to scale. An imminent collision is evident at such location as can be seen in a close-up of the two shortest minima (Figure 2.14).

When the minimum separation distance becomes equal to $r_1 + r_2$ (or zero if the radii are accounted for), a collision occurs and a simulated force proportional to how much the cylinders have interpenetrated and normal to both tether axes is provided to the tether at the collision point and shown as the dark lines in Figure 2.15.

As seen in Figure 2.16, the tether successfully reacts to the force and remains separated. This is better seen in Figure 2.17 where the separation distance reaches

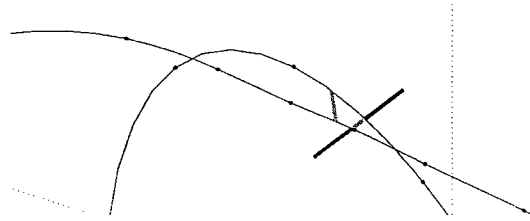


Figure 2.15. The shortest minimum (right) during the collision. The two added dark lines represent the forces being applied at the contact points.

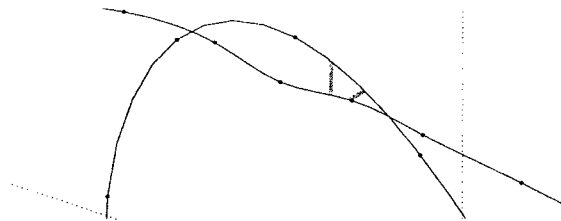


Figure 2.16. The shortest minimum (right) after the collision.

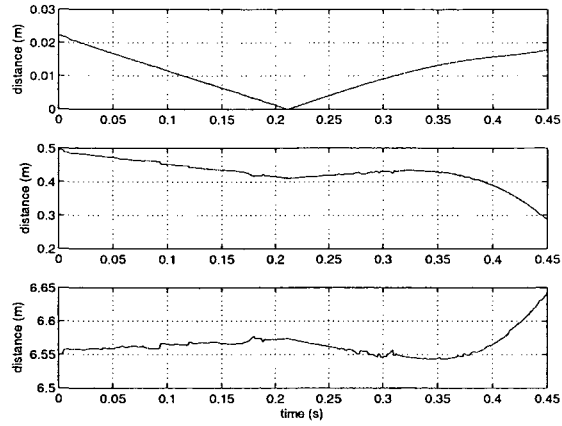


Figure 2.17. The minimum separation distances of the three shortest minima. The shortest of which showing a collision at time ≈ 0.21 seconds. Distances have been adjusted to account for the tether radius.

0 m (actually a little less than 0 as intersection occurs). During the collision period, forces are applied on each of the tether segments and the tether “bounces” in reaction bringing the tether back to an equilibrium condition.

Chapter 3

Contact Dynamics

Once a collision has been detected, it is imperative the collision forces be resolved. If a momentum conservation model was used, only a change of their instantaneous velocity vectors would be necessary. This also holds for all other contact models except the velocity vectors are generally changed in a more indirect manner, like using a time-varying force to change the velocity. This Chapter describes the use of the volume of intersection to determine the normal contact force. Once the normal force is obtained, it is used in the LuGre friction model [32] to find the tangential friction forces involved with the collision.

3.1 Interference Geometry Between Two straight Skew Cylinders

Finding the volume of intersection between two cylinders can be solved analytically. There are four independent variables which influence the volume of intersection of two intersecting cylinders (Figure 3.1): the radii of both cylinders (r_1 and r_2), the angle α

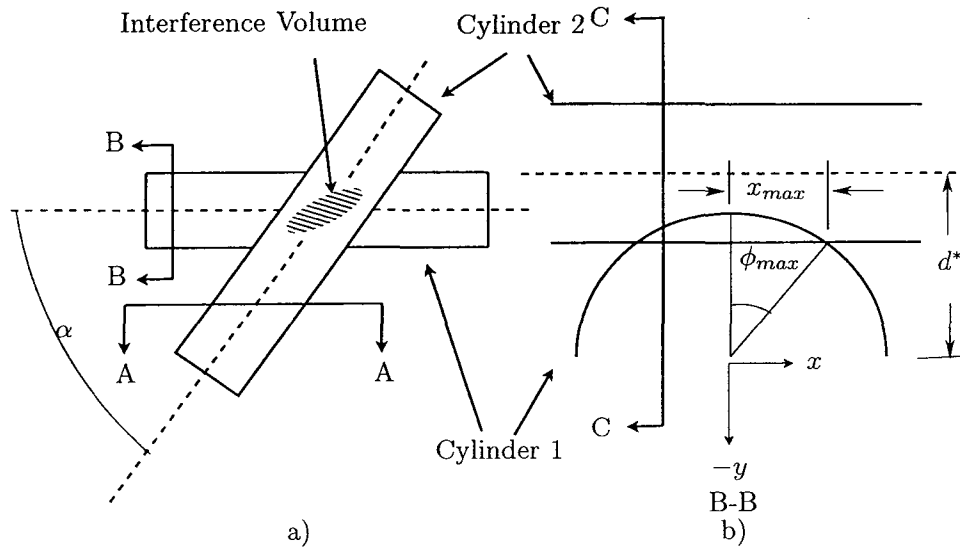


Figure 3.1. Two intersecting cylinders where a) shows α , the angle between them and b) shows the minimum separation distance d^* as well as ϕ_{max} and x_{max} .

between their centre lines which provides their relative orientation (see Figure. 3.1a), as well as their separation distance d^* (minimum separation distance between their centre lines).

Using this information, the problem can be formulated as a volume integral. Looking along the axis of cylinder 1 as in Figure 3.1b, certain geometrical information about the volume of intersection can be determined. The angle ϕ_{max} is the angle between the common perpendicular to the two centre lines and the outer edge of con-

tact. The distance from the common perpendicular to the edge of contact is denoted by x_{\max} and can be defined as $x_{\max} = r_1 \sin(\phi_{\max})$. The distance x_{\max} is used as an integration bound for cylinder 1 (see Figure 3.1b).

Due to the symmetry of the contact region, the volume integral can be formulated as:

$$V = 2 \int_0^{x_{\max}} A(x) dx, \quad (3.1)$$

where V is the interference volume, $A(x)$ is the area of the slice of the volume of intersection at x and dx is the x differential. The area $A(x)$ of a slice of the volume taken at a distance x from the common perpendicular is represented in Figure 3.2. Before analysing the area $A(x)$, the following dimensions should be known (see Figure 3.3):

$$c_1 = d^* - r_1, \quad (3.2)$$

$$c'_1 = r_2 - c_1, \quad (3.3)$$

$$d_1 = r_1 - c'_1 = r_1 - r_2 + c_1 = r_1 - r_2 + d^* - r_1 = d^* - r_2, \quad (3.4)$$

$$\phi = \tan^{-1}(x/d_1), \quad (3.5)$$

$$h = \sqrt{x^2 + d_1^2}, \quad (3.6)$$

$$c_2 = r_1 - h, \quad (3.7)$$

$$c'_2 = \sqrt{r_1^2 - (h \sin \phi)^2} - d_1, \quad (3.8)$$

where equation (3.8) is a rearranged version of pythagoras' theorem where the right angled triangle is defined as: the hypotenuse (r_1), the opposite side ($h \sin \phi$) and the adjacent side is ($c'_2 + d_1$). With c'_2 in hand, the area $A(x)$ can be defined as:

$$A(x) = 2 \int_0^{z_{\max}} (r_2 \cos \theta - d_2) dz, \quad (3.9)$$

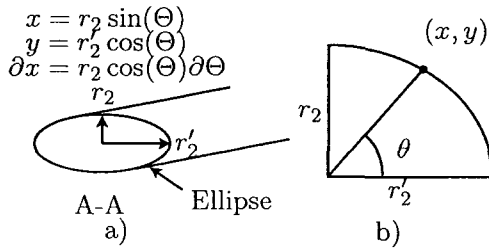


Figure 3.2. Cylinder cut at an angle where a) Section A-A shows an elliptical profile and b) geometrical information about an ellipse.

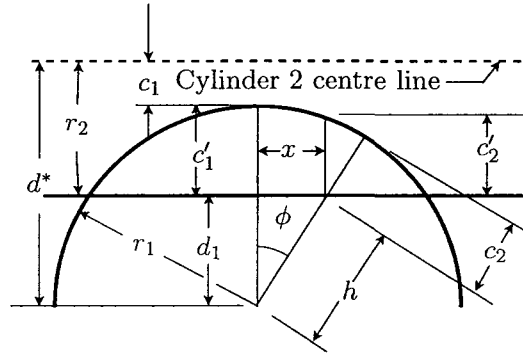


Figure 3.3. Some important geometrical information in contact region (mainly c'_2).

where $z_{\max} = r'_2 \sin(\theta_{\max})$, $\theta_{\max} = \cos^{-1}(d_2/r_2)$, $d_2 = r_2 - c'_2$, and r_2 and r'_2 are the minor and major semi-axes of an ellipse (see Figure 3.4). When angle α is increased, the eccentricity of the profile of B-B of cylinder 2 becomes more pronounced.

Because the integration of the area is about z , the angle θ of the ellipse can be found for a specific z position as:

$$\theta = \sin^{-1}(z/r'_2). \quad (3.10)$$

Any integration method could be used to solve for $A(x)$ in equation (3.9) and V in

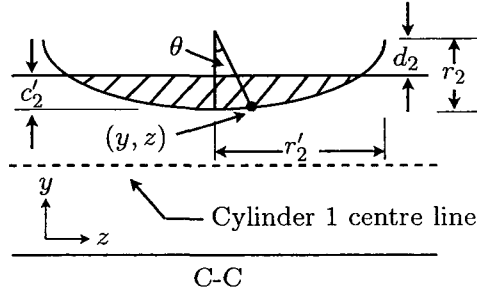


Figure 3.4. Section C-C at a distance x from the common perpendicular.

equation (3.1), however, for computational efficiency, a 5-point Gaussian quadrature method is recommended as there is minimal loss of accuracy for the large performance gain over other numerical integration methods.

3.2 Contact Surface Area Between Two Interfering Cylinders

The projection of the intersection volume between two cylinders onto a plane whose normal is perpendicular to both is always elliptical with only two exceptions: when the angle α between them is 90° or 0° , where their projections are circular and rectangular, respectively.

The major axis of the ellipse formed by the projection of the intersection volume is always located half-way between the acute angle α formed by the two intersecting cylinders (see Figure 3.8). The general equation for such an ellipse is as follows:

$$\frac{x_{ellipse}^2}{a^2} + \frac{y_{ellipse}^2}{b^2} = 1, \quad (3.11)$$

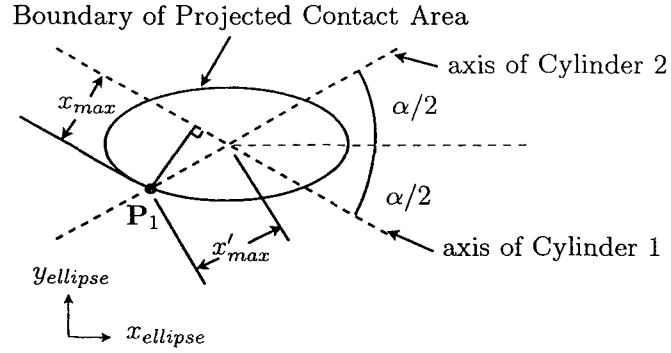


Figure 3.5. Finding P_1 in terms of $(x_{ellipse}, y_{ellipse})$ using x_{max} and α

where a and b are the major and minor semi-axes, respectively, and $x_{ellipse}$ and $y_{ellipse}$ are the x and y coordinates of a point on the ellipse in the ellipse's frame (where the x axis is aligned with the major axis). Knowing any 3 amongst $a, b, x_{ellipse}$ or $y_{ellipse}$ allows to solve for the fourth parameter using equation (3.11). Unfortunately, analytically, only two of these can be determined. That is, the x and y coordinates of a point on the ellipse can be determined from the geometrical properties shown in Section 3.1. This point is defined using one of the boundaries of the intersection volume.

As seen in Figure 3.5, x_{max} represents the distance from the centre of intersection to the outer edge of integration located in-line with the axis of cylinder two. In order to define the $x_{ellipse}$ and $y_{ellipse}$ coordinates, x'_{max} is projected onto a line perpendicular to cylinder 1, x_{max} (Figure 3.5):

$$x'_{max} = \frac{x_{max}}{\sin \alpha}. \quad (3.12)$$

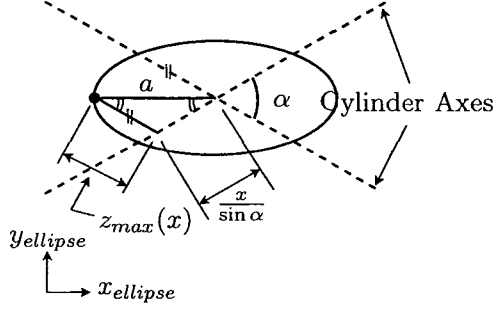


Figure 3.6. Finding the magnitude of the major axis a using $z_{\max}(x)$ and x .

With x'_{\max} in hand, the coordinates of \mathbf{P}_1 on the ellipse's frame can be defined by projecting x'_{\max} onto the major and minor axes as:

$$x_{\text{ellipse}}(\mathbf{P}_1) = x'_{\max} \cos \frac{\alpha}{2}, \quad (3.13)$$

and

$$y_{\text{ellipse}}(\mathbf{P}_1) = x'_{\max} \sin \frac{\alpha}{2}, \quad (3.14)$$

where $x_{\text{ellipse}}(\mathbf{P}_1)$ and $y_{\text{ellipse}}(\mathbf{P}_1)$ are the x and y coordinates relative to the ellipse's frame of point P_1 as seen in Figure 3.5 assuming the centre of the ellipse is $(0, 0)$. Variable a , the length of the major semi-axis, though not found analytically can be found by numerical means. Take note that a different z_{\max} is obtained at each different slice of volume at x (Figure 3.3). For each x , z_{\max} is found and compared with $\frac{x}{\sin \alpha}$ for equality. That is, when z_{\max} for a certain x is equal to $\frac{x}{\sin \alpha}$, one can say that the third side of a triangle defined by $z_{\max}(x)$ and $\frac{x}{\sin \alpha}$ is the major axis of the triangle. Therefore, using the cosine law, one can easily determine the length of a (see Figure 3.6). This is true because the side of the triangle that represents z_{\max} is parallel to the axis of cylinder 1 which makes the triangle an isosceles triangle whose

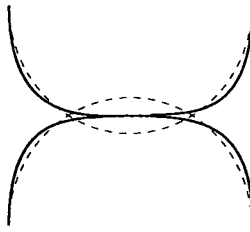


Figure 3.7. A comparison of two intersecting: a) undeformed cylinders (dashed lines) b) deformed by contact pressure (solid lines).

base angles are $\frac{\alpha}{2}$. With $x_{\text{ellipse}}(\mathbf{P}_1)$, $y_{\text{ellipse}}(\mathbf{P}_1)$ and a , the minor axis b is easily found using equation (3.11). The area of an ellipse, knowing a and b , is easily found as πab .

Though this method indeed provides the projected area of the volume of interference, it is only a crude approximation of the actual contact patch area. Figure 3.7 shows the differences between the actual contact patch and the intersection geometry. A better approximation of this contact patch area may be to determine it by calculating the area of a slice of the interference geometry at plane A located directly in the middle (see Figure 3.8a). That is, since for tether self-collision both contacting segments are made of the same material and have similar geometry, both objects will deform equally, and hence meet half way at A (Figure 3.8b). A fast way of estimating the area of A is by separating the cylinders by $\frac{c_1}{2}$ and finding the elliptical area of the projection of its interference geometry as described earlier in this section.

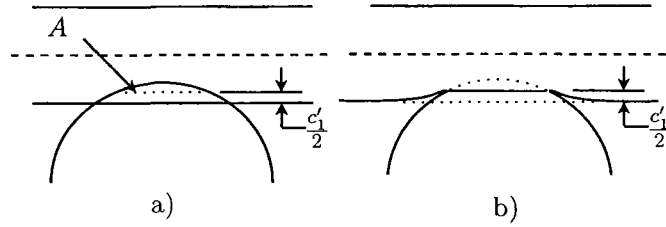


Figure 3.8. Two intersecting cylinders showing a) their undeformed state and the plane A where the surface would meet under deformation and b) the deformed cylinders after deformation.

3.3 Determining the Normal Contact Force

When two objects come into contact, a normal pressure field, as well as a tangential shear stress field, is formed between them. This pressure is caused by the elastic properties of the two bodies undergoing deformation during contact like in Figure 3.8b. Integrating this pressure over the area can provide the total normal force keeping them apart:

$$f_n = \int \sigma \partial a. \quad (3.15)$$

Unfortunately, during a computer simulation, the pressures are just as unknown as the actual contact forces. However, the pressures can be obtained through the approximations of the deformations. Thus, the normal force can be approximated based on the elastic properties of the material and the contact geometry as:

$$\sigma = E\epsilon, \quad (3.16)$$

$$\epsilon = \frac{\delta r}{r}, \quad (3.17)$$

$$\partial V = \delta r \partial a, \quad (3.18)$$

$$V' = \int \partial V = \int \delta r \partial a. \quad (3.19)$$

where σ is the normal stress/pressure, E is Young's modulus, ϵ is the strain, dr is the change in radius of a point on the contact patch surface, r is the undeformed radius, ∂a is the differential area and ∂V is the differential volume (see Figure 3.9a). The volume V' found using the method described in Section 3.1 approximates the actual deformation of both cylinders. That is, the volume V' represents the deformation of cylinder 1 caused by a force applied to cylinder 1 plus the deformation caused by a force applied to cylinder 2. Therefore, to find the force from deformation on one cylinder, half of the interference volume is used assuming that both cylinders have the same elastic properties. That is,

$$V = \frac{V'}{2}. \quad (3.20)$$

Therefore, the total normal force on one cylinder f_n can be expressed as,

$$f_n = \frac{EV}{r}. \quad (3.21)$$

This method for obtaining the normal contact forces has been previously presented in [4, 17, 26] and is technically the Winkler elastic foundation model. To model deformation rate effects, the rate of change of the volume is taken into account with an associated damping coefficient which then becomes a spring-dashpot model. Each differential volume is modelled as a spring and dashpot (Figure 3.9b) that is said to have been compressed down to the contact surface in the middle of the intersection geometry. In this case, the spring coefficient in equation (3.21) is the Young's Modulus E of the deforming object. Taking damping into consideration, equation (3.21) becomes:

$$f_n = \frac{EV}{r} + B \frac{dV}{dt}. \quad (3.22)$$

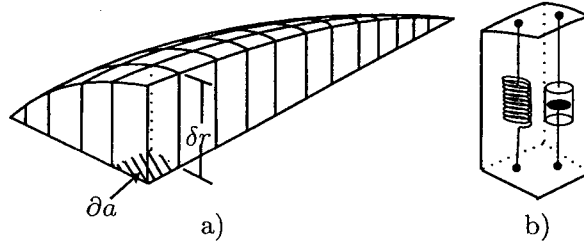


Figure 3.9. a) The volume of intersection showing the differential volumes. b) One of the differential volumes showing its spring dashpot representation.

where $\frac{dV}{dt}$ is the rate of change of the volume with respect to time, and B is the damping factor.

In order to verify the accuracy of this force, it was compared with the Hertzian model of general contact¹ [25]. It is compared to the Hertzian contact model due to its wide acceptance by the scientific community. Here, the force provided through the volume of interference is used in the Hertzian model (refer to Appendix C) which can also provide the major and minor axes of the elliptical contact patch. Through a comparison of the contact patch area from the volumetric model and the Hertzian model², the accuracy of the volumetric model is determined.

Figure 3.10 shows the percentage differences between the contact patch areas of both methods. The model described here adheres to the Hertzian contact model at

¹Hertzian contact theory [25, 1] requires the geometry of the contact patch as input in order to provide a force. The method of describing the contact patch area, described using Figure 3.6, has a greater complexity than calculating the volume of intersection.

²Please note that Hertzian contact theory was rejected as a potential contact model for the contact of tethers because Hertzian theory assumes quadratic surfaces, but most of the time, the tether's surface shape is cubic. In addition to this, Hertzian contact theory does not directly consider damping.

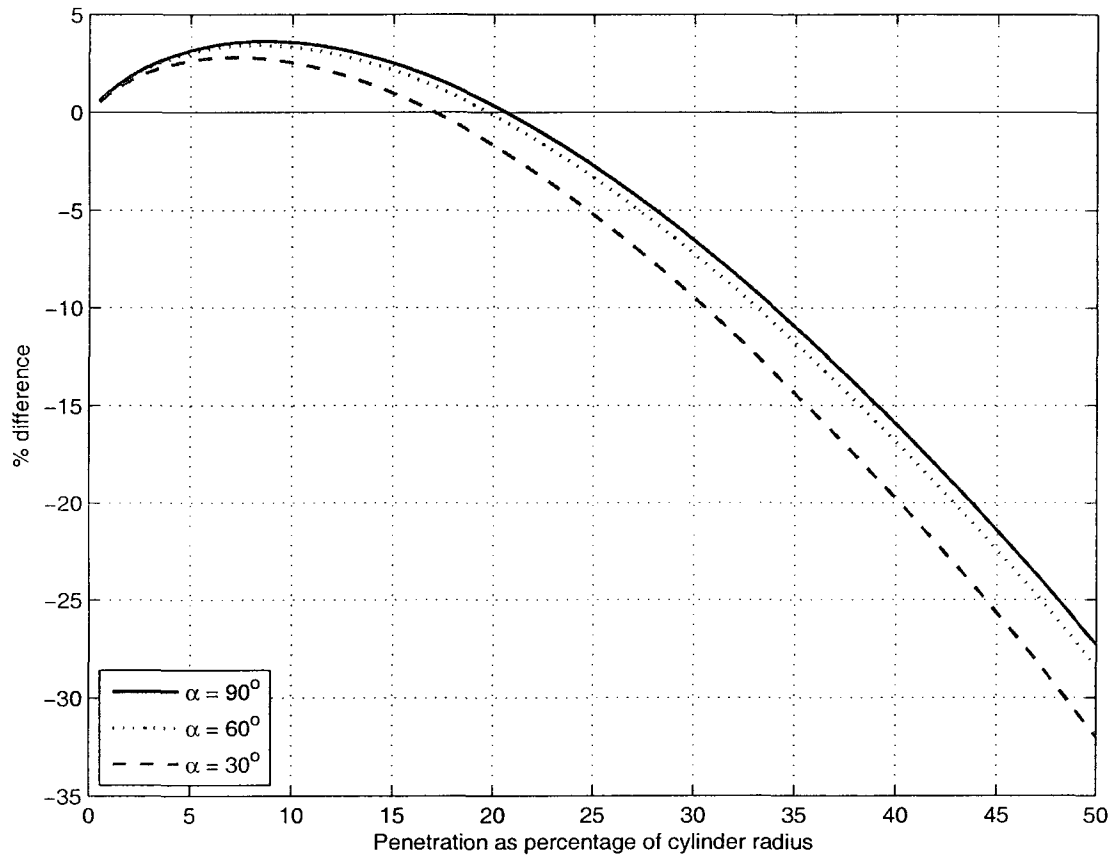


Figure 3.10. Percentage difference between contact patch area of the Hertzian model and the contact patch area from the interference geometry.

almost any angle to within $\pm 5\%$ as long as penetration depth is maintained under 25% of the tether's radius. However promising this seems, all that this means is that the volume of interference model adheres to the Hertzian model, it is therefore only as accurate as the Hertzian model. Since the Hertzian model assumes small deflections relative to the contacting surfaces, it is doubtful that penetrations as high as 25% of the radius would provide realistic results.

It should be noted that a maximum volume of interference exists when the minimum separation distance between the tether centres are zero. The algorithms here will only calculate the volume of intersection up to a separation distance of the greatest radius in the cylinder pair. At this point, the contact forces should tend to infinite as the centre line separation distance nears 0. An expression resembling $\frac{d_{cutoff}^*}{d^*}$ is proposed to be used to scale the force from the volume of intersection at the cut-off point where d_{cutoff}^* is the volume of intersection where the user decides the volume of intersection method is no longer accurate.

3.4 Determining the Tangential Contact Forces

Anytime there is an object in contact with another or with itself, friction could be said to occur. The modelling of frictional forces is not an easy task as there is a multitude of factors that influences the resulting forces [30]. The material properties, the normal forces pushing the two surfaces together, surface quality, the existence of debris, fluids at the contact interface, temperature, and relative velocities all have an influence on friction. Some of the phenomena attributed to contact cases with friction that a model must try to capture in order to maintain fidelity are: the Stribeck effect, static friction/break away force, frictional lag (dwell time) and viscous friction [30].

That being said, friction models can be categorised into two types: static models and dynamic models. Besides computer simulations, such friction models are used in controller compensation in order to predict the friction in moving parts and compensate to maintain fidelity. In this field, the most widely used friction models are static models such as the classical models like the Coulomb model and the Karnopp model [30].

Among the dynamic friction models, the most significant is arguably the bristle model [33] which captures stick/slip behaviour and the random nature of friction. However interesting this model is, it is not efficient enough to be used in real-time applications [30, 33] and ignores other important phenomena.

The LuGre model [32], on the other hand, is a relatively new model that covers most phenomena of friction. This computationally efficient model (also employed in [26]), calculates the friction force based on bristle deflections z^{br} and deflection rates \dot{z}^{br} . The original bristle model [33] loses efficiency in the fact that it takes account of N bristles as well as each of their deflections and stick/slip states. The LuGre model amasses the bristles into one and keeps track of the stick/slip states with an average bristle deflection.

In [26], the LuGre model was extended to transform the unidimensional bristle deflections z^{br} to a tridimensional vectorial form \mathbf{z}^{br} . As outlined in [26], the friction force caused by the deflection of the bristles is:

$$\mathbf{f}_{br} = \sigma_0 \mathbf{z}^{br} + \sigma_1 \dot{\mathbf{z}}^{br}, \quad (3.23)$$

where σ_0 and σ_1 are the bristle stiffness and damping coefficients, respectively. Every contact has its own bristle deflection state \mathbf{z}^{br} , where $|\mathbf{z}^{br}| = 0$ when a contact is initiated. The rate of change of the bristle deflection $\dot{\mathbf{z}}^{br}$ can be said to have two

different behaviours, $\dot{\mathbf{z}}_{st}^{br}$ for sticking and $\dot{\mathbf{z}}_{sl}^{br}$ for sliding. A sticking state function s_{st} (which cycles between 0 and 1) is introduced to determine the current state of friction:

$$\dot{\mathbf{z}}^{br} = s_{st}\dot{\mathbf{z}}_{st}^{br} + (1 - s_{st})\dot{\mathbf{z}}_{sl}^{br}, \quad (3.24)$$

with

$$s_{st} = e^{-\frac{(\mathbf{V}_t^T \mathbf{V}_t)}{(v_s^2)}}, \quad (3.25)$$

where v_s is the Stribeck velocity³ and \mathbf{V}_t is the tangential velocity vector. When s_{st} is 1, the friction surfaces are under a full sticking state. Conversely when s_{st} is 0, the surfaces are in a full sliding state. When undergoing full sticking, it is said that the bristles of one surface are attached to the bristles on the other surface and therefore their deflection rate is identical to their relative tangential velocity:

$$\dot{\mathbf{z}}_{st}^{br} = \mathbf{V}_t. \quad (3.26)$$

Coulomb friction, a friction model where the friction force between two sliding bodies is defined as directly proportional to the magnitude of the normal force f_n as $\mu_c f_n$ is defined here in vector form as:

$$\mathbf{f}_c = \mu_c f_n \text{dir}_\epsilon(\mathbf{V}_t, v_\epsilon), \quad (3.27)$$

where v_ϵ is a numerical tolerance for the velocity and

$$\text{dir}_\epsilon(\mathbf{V}_t, v_\epsilon) = \begin{cases} \frac{\mathbf{V}_t}{|\mathbf{V}_t|}; & |\mathbf{V}_t| \geq v_\epsilon \\ \frac{\mathbf{V}_t}{v_\epsilon} \left(\frac{3}{2} \frac{|\mathbf{V}_t|}{v_\epsilon} - \frac{1}{2} \left(\frac{|\mathbf{V}_t|}{v_\epsilon} \right)^3 \right); & |\mathbf{V}_t| < v_\epsilon \end{cases} \quad (3.28)$$

³The Stribeck velocity, is the velocity at which only 37% of the bristles will be in a sticking condition.

where as a guideline v_ϵ is generally no more than one tenth of v_s [26].

Assuming that under sliding conditions the friction force, *i.e.*, the force of the bristles, is the Coulomb friction force, equation (3.23) is set equal to equation (3.27).

Solving for $\dot{\mathbf{z}}$ where $\dot{\mathbf{z}}^{br} \simeq \dot{\mathbf{z}}_{sl}^{br}$, it is possible to obtain $\dot{\mathbf{z}}_{sl}^{br}$ as:

$$\dot{\mathbf{z}}_{sl}^{br} = \frac{1}{\sigma_1} \mathbf{f}_c - \frac{\sigma_0}{\sigma_1} \mathbf{z}^{br}. \quad (3.29)$$

Combining equations (3.24), (3.26) and (3.29) gives the following equation for $\dot{\mathbf{z}}$:

$$\dot{\mathbf{z}}^{br} = s_{st} \mathbf{V}_t + (1 - s_{st}) \left(\frac{1}{\sigma_1} \mathbf{f}_c - \frac{\sigma_0}{\sigma_1} \mathbf{z}^{br} \right). \quad (3.30)$$

The maximum friction force for a given f_n is generally dictated by the friction coefficient of sticking μ_s where the force of friction is $\mu_s f_n$. However, when sticking is first initiated, the force required to dislodge the objects is generally not immediately $\mu_s f_n$ [30]. This effect is known as dwell-time dependency [26]. To model this dwell time effect, another state variable, s_{dw} , is introduced which again cycles between 0 and 1 and is set to 0 at the onset of contact. The rate of change of s_{dw} , \dot{s}_{dw} is defined as:

$$\dot{s}_{dw} = \begin{cases} \frac{1}{\tau_{dw}} (s_{st} - s_{dw}); & s_{st} - s_{dw} \geq 0, \\ \frac{1}{\tau_{br}} (s_{st} - s_{dw}); & s_{st} - s_{dw} < 0, \end{cases} \quad (3.31)$$

where τ_{dw} is the dwell-time time constant and τ_{br} is the bristle dynamics time constant (where $\tau_{br} = \frac{\sigma_1}{\sigma_0}$). The magnitude of the maximum stiction force is defined as:

$$f_{max} = f_n (\mu_c + (\mu_s - \mu_c) s_{dw}). \quad (3.32)$$

The tangential friction force \mathbf{f}_t is then defined as follows:

$$\mathbf{f}_t = -(\text{sat}(\mathbf{f}_{br}, f_{max}) + \sigma_2 \mathbf{V}_t), \quad (3.33)$$

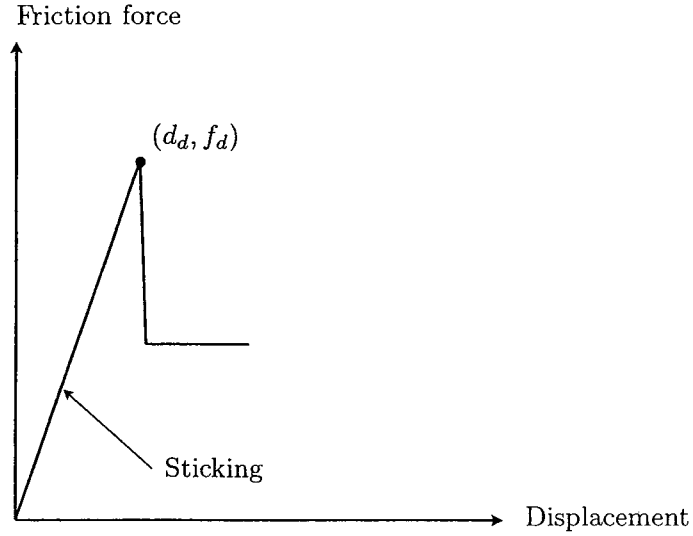


Figure 3.11. Friction force versus displacement showing the change from sticking to sliding friction.

where σ_2 is the viscous damping friction coefficient to model lubrication and the saturation function $sat(\mathbf{f}_{br}, f_{max})$ is defined as:

$$sat(\mathbf{f}_{br}, f_{max}) = \begin{cases} \mathbf{f}_{br}; & |\mathbf{f}_{br}| \leq f_{max}, \\ f_{max} \frac{\mathbf{f}_{br}}{|\mathbf{f}_{br}|}; & |\mathbf{f}_{br}| > f_{max}. \end{cases} \quad (3.34)$$

which completes the LuGre model as described be [26].

3.4.1 Measuring σ_0 , σ_1 , σ_2 , ν_s , μ_s , μ_c , and τ_{dw}

In order to measure the variables which affect the friction force of the LuGre model; σ_0 , σ_1 , σ_2 , ν_s , μ_s , μ_c , and τ_{dw} , one simply needs to generate some force vs. displacement and force vs. velocity data (see Figures 3.11 and 3.12). During sticking, the friction force is equal to force from the bristle deflection. Since the bristles are sticking to each other, the bristle deflection is equal to half the relative displacement of the contacting

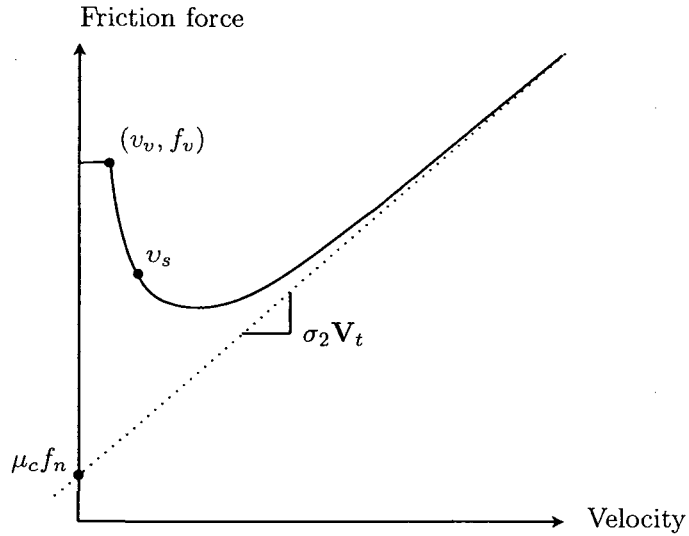


Figure 3.12. Friction force versus velocity showing the Stribeck effect.

surfaces, hence $\sigma_0 = \frac{f_d}{d_d}$, see Figure 3.13. Also during sticking, the bristle deflection rate, \dot{z}^{br} , is equal to the relative tangential velocities of the two surfaces, $\sigma_1 = \frac{f_v - z^{br} \sigma_0}{v_v}$, see Figure 3.12.

The sticking friction coefficient, μ_s , is the relationship between the normal force and the friction force, *i.e.*, $\mu_s = f_d/f_n$. During pure sliding friction, either asperities which are breaking off or a fluid at the interface lubricates the contact region. The faster the relative tangential motion, the higher the drag forces become. The viscous damping coefficient, σ_2 , is simply the slope of the force-velocity curve during full sliding (Figure 3.12).

Because the viscous damping force component is 0 when the velocity is 0, the entire sliding friction force is caused by $\mu_c f_n$. Therefore, $\mu_c f_n$ is the force at which a line asymptotic to the force velocity curve at high velocities, whose slope is σ_2 , intercepts the force axis, see Figure 3.12. The dwell time constant τ_{dw} is simply the

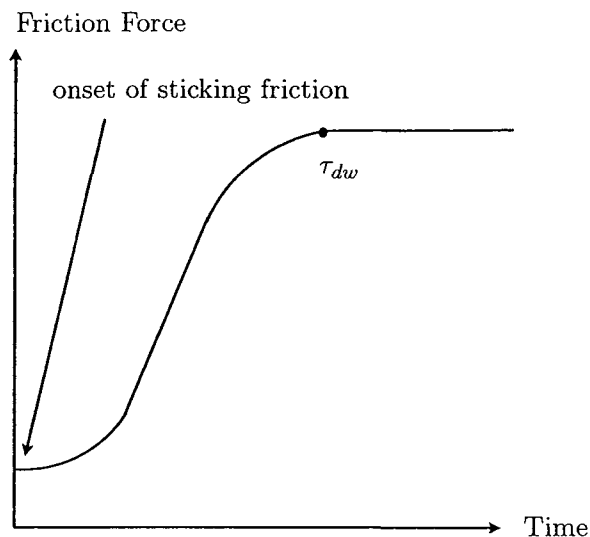


Figure 3.13. The break-away friction force at the initiation of sticking as a function of time.

time it takes for the sticking friction to reach its full sticking force, measured from force-time data of a sticking friction situation (Figure 3.13).

3.5 Numerical Example

For a detailed numerical example of the LuGre model undergoing stick/slip transitions please see [26, 30], or [32]. Here, the numerical example of a collision that was first presented in Chapter 2 is revisited using the contact models described here. This contact model is incorporated into the tether model described in [44] and is provided the same initial conditions as in Chapter 2. It should be noted that in Chapter 2, the collision is almost instantaneous due to the large modulus of elasticity used in that example. Here, a modulus of elasticity E of 0.02 GPa, which is closer to low density polyethelene, was used along with a damping factor B of $0.01 \frac{Ns}{m^3}$. The friction model

used the following values: $\mu_s = 0.15$, $\mu_c = 0.1$, $\sigma_0 = 1 \times 10^5(\frac{N}{m})$, $\sigma_1 = \sqrt{1^5}(\frac{Ns}{m})$, $\sigma_2 = 0.1(\frac{Ns}{m})$, $v_s = 0.001(\frac{m}{s})$, $\tau_{dw} = 2(s^{-1})$ and $v_\epsilon = \frac{v_s}{100}(\frac{m}{s})$.

Note that no physical tests were undertaken to obtain any of these contact model settings. The frictional model settings are the same as those used in the numerical example presented in [26]. Physical tests should be undertaken on real tethers to determine what all these model settings should be set to in order to provide an accurate model. Figure 3.14 shows the contact beginning close to 0.21 seconds and remaining in contact for much longer than the example in Chapter 2. Since the modulus of elasticity is lower, the tether allows more deformation. This provides for a sustained contact where friction is introduced. Because the relative velocities remain rather high, the friction force undergoes a mixture of Coulomb and viscous friction. This explains the almost linear relationship between the friction force and the normal force evident by comparing the first and third subfigures in Figure 3.14. The second plot in Figure 3.14 also shows a small reduction in relative tangential velocity which is caused by friction.

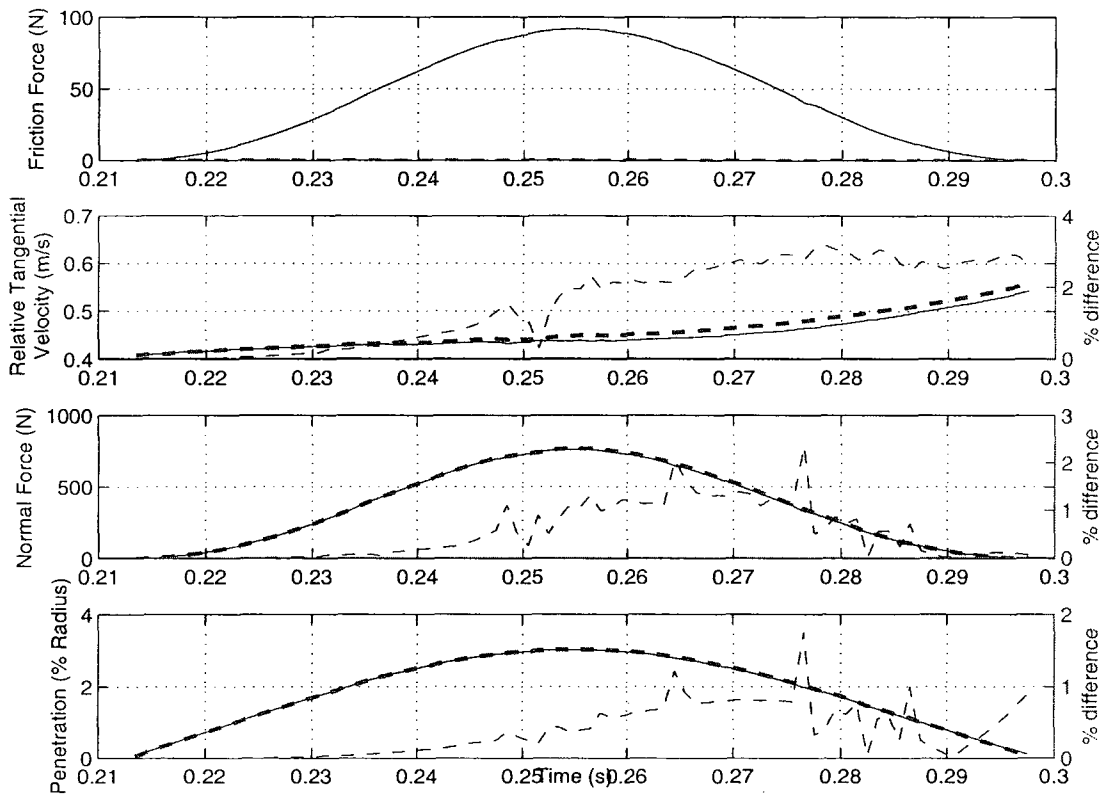


Figure 3.14. A tether self-collision showing the forces involved, tether penetration and their relative velocity. The thin solid lines represent the simulation with friction, the thick dashed lines represent the simulation with no friction, and the thin dashed lines are the % difference between the two.

Chapter 4

Conclusions

The problem of finding the minimum separation distances between different segments of a single tether was discussed. It was shown that detecting self-collision situations of tethers is well suited to local optimisation techniques due the separation distance function's near quadratic continuous features of its simple 2D solution space.

A continuous collision detection method for slack, flexible cylindrical objects was presented to deal with tunneling and also to determine how accurate a reaction the collision force is going to provide.

A method for determining the volume of interference between two skew cylinders was demonstrated. This volume is then used to model the forces involved in the contact region. To compare this method against the Hertzian model of general contact, the corrected area of projection of the contact volume was calculated. It was then compared against the contact patch area of the Hertzian model given the same force provided by the volume of interference. It was shown that the volume of interference model corresponds fairly well with the Hertzian model provided the penetration depth remains within about 25% of the tether's radius. It should be noted that this con-

clusion only means that the volume of intersection is only as accurate as the Herzian contact model for that penetration amount.

The LuGre friction model [32] was employed as it takes into account many of the known phenomena of friction including viscous friction and retains computational efficiency. With the interference-volume-based spring-dashpot contact model and the LuGre friction model implemented in a lumped-mass tether model, a numerical example demonstrated the tether behaviour during a contact situation. The model is efficient, could be used in real-time applications, and has the potential to be quite accurate provided the appropriate material properties.

4.1 Thesis Contributions

The main contributions of this work are:

- Developed an efficient two-stage separation distance optimisation-based continuous collision detection method for use with continuous cubic spline based tethers.
- Developed a method for determining the volume of interference between two straight skew cylinders.
- Developed a method for determining the approximate contact patch area between two straight skew cylinders.
- Showed that the volume of intersection method is comparable to the Hertzian theory of general contact.

- Showed a coherence between the volume of interference contact model and Hertzian contact model.
- Demonstrated the volume of interference to be the Winkler elastic foundation model.
- Discussed the LuGre model's inability to handle rotational friction.

4.2 Future Work

The following is a list of avenues that remain to be investigated which this work has spawned:

- Compare the use of both two-stage optimisation and purely continuous optimisation methods.
- Extend volume of interference method to include tether curvature.
- Further develop the material tests and methods for obtaining material properties for use in the contact models.
- Compare d^* from Hertz' model of general contact against the volume on interference model's.
- Validate contact dynamics model.
- Implement some node management code: in the addition of nodes, keeping track of the ones generated for contact and deleting them during none CPU-intensive periods. Reuse added contact nodes by moving them instead of simply adding a extra one (twice the garbage).

- Investigate dynamically changing the number of minima tracked and tracking minima based on proximity rather than N_{min} smallest minima.
- Investigate other Wrinkler elastic foundation model variations such as investigating Mooney-Rivlin solid, hyper-elastic, and neo-Hookean models as rubber is considered not to obey Hooke's law.
- Extend methods to general polygonal objects: The methods of finding the minimum separation distance can easily be extended to polygonal objects. Using MLSDist and a continuous optimisation (gradient based) and the UV coordinates of polygons.
- Develop method for calculating the volume of interference between general polygonal concave objects. There currently exists methods for determining the volume of intersection for convex objects only. One might need to use convex subdivision of concave objects.
- Investigate the use of texture map information to map accurate surface details of very fine polygonal meshes onto rough polygonal meshes. For minimum separation distance code, the surface normal could easily be mapped.

References

- [1] R. L. Norton, *Machine Design, An Integrated Approach*, Prentice Hall, 2000.
- [2] P. Williams, “Towing and winch control strategy for underwater vehicles in sheared currents”, *International Journal of Offshore and Polar Engineering*, vol. 16(3), pp. 218–227, 2006.
- [3] M. Grosenbaugh, “Transient behavior of towed cable systems”, *Ocean engineering, article in-press*, 2007.
- [4] S. Hasegawa and M. Sato, “Real-time rigid body simulation for haptic interactions based on contact volume of polygonal objects”, *Eurographics*, vol. 23, no. 3, 2004.
- [5] C. Duriez, F. Dubois, A. Kheddar, and C. Andriot, “Realistic haptic rendering of interacting deformable objects in virtual environments”, in *IEEE Transactions on Visualization and Computer Graphics*, 2006, vol. 12, pp. 36–47.
- [6] O. Ma, “Contact dynamics modelling for the simulation of the space station manipulators handling payloads”, in *IEEE International Conference on Robotics and Automation*, May 1995, pp. 1252–1258.
- [7] M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrmann, M.-P. Cani, F. Faure, N. Magnenat-Thalmann, W. Strasser, and P. Volino, “Collision detection for deformable objects”, *Computer Graphics Forum*, vol. 24, no. 1, pp. 61–81, 2005.
- [8] S.-E. Yoon and D. Manocha, “Cache-efficient layouts of bounding volume hierarchies”, in *EuroGraphics*, E. Gröller and L. Szirmay-Kalos, Eds., 2006, vol. 25.
- [9] G. Hippmann, “An algorithm for compliant contact between complexly shaped surfaces in multibody dynamics”, in *IDMEC/IST*, J. A. C. Ambrósio, Ed., 2003.

- [10] E. Ericson, *Real-Time Collision Detection*, The Morgan Kaufmann Series in Interactive 3D Technology. Morgan Kaufmann Publishers, 2005.
- [11] P. Jiménez, F. Thomas, and C. Torras, “3D Collision Detection: A Survey”, *Computers and Graphics*, vol. 25, no. 2, pp. 269–285, April 2001.
- [12] M. Lin and D. Manocha, “Collision and proximity queries”, *In Handbook of Discrete and Computational Geometry*, Chapman and Hall/CRC, 2003.
- [13] N. K. Govindaraju, M. C. Lin, and D. Manocha, “Fast and reliable collision culling using graphics hardware”, in *VRST '04: Proceedings of the ACM symposium on Virtual reality software and technology*, New York, NY, USA, 2004, pp. 2–9, ACM Press.
- [14] N. K. Govindaraju, I. Kabul, Lin M. C., and D. Manocha, “Fast continuous collision detection among deformable models using graphics processors”, in *Computer & Graphics*, 2007, vol. 31, pp. 5–14.
- [15] S. Redon, Y. J. Kim, M. C. Lin, and D. Manocha, “Fast continuous collision detection for articulated models (tr03-038)”, Tech. Rep., University of North Carolina at Chapel Hill, 2003.
- [16] J. E. Bobrow, “A direct minimization approach for obtaining the distance between convex polyhedra”, *International Journal of Robotics Research*, vol. 8, no. 3, pp. 65–76, June 1989.
- [17] L. Luo and M. Nahon, “A compliant contact model including interference geometry for polyhedral objects”, *ASME Journal of Computational and Nonlinear Dynamics*, vol. 1, no. 2, pp. 150–159, 2006.
- [18] O. Ma and M. Nahon, “A general method for computing the distance between two moving objects using optimization techniques”, in *ASME Advances in Design Automation*, New York, 1992, vol. 1, pp. 109–117, ASME.
- [19] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi, “A fast procedure for computing the distance between complex objects in three dimensional space”, in *IEEE Journal of Robotics and Automation*, 1988, vol. 4, pp. 193–203.
- [20] B. Mirtich, “V-clip: fast and robust polyhedral collision detection.”, in *ACM Transactions on Graphics*, 1998, vol. 17, pp. 177–208.
- [21] J. A. Carretero, *Distance determination algorithms for convex and concave objects*, PhD Thesis, Department of Mechanical Engineering, University of Victoria, Victoria, British Columbia, Canada, 2001.

- [22] J. A. Carretero and M. A. Nahon, “Solving minimum distance problems with convex or concave bodies using combinatorial global optimization”, *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 35, no. 6, pp. 1144–1155, December 2005.
- [23] J. A. Carretero and R. Uppuluri, “A novel optimization-based pruning strategy for concave minimum distance problems”, in *Proceedings of the 2004 CSME Forum, London, Ontario, Canada*, June 1-4 2004, pp. 581–591.
- [24] G. Gilardi and I. Sharf, “Literature survey of contact dynamics modelling”, *Mechanism and Machine Theory*, vol. 37, no. 10, pp. 1213–1239, October 2002.
- [25] K. L. Johnson, *Contact Mechanics*, Cambridge University Press, 1987.
- [26] Y. Gonthier, J. McPhee, C. Lange, and J.-C. Piedboeuf, “A regularized contact model with asymmetric damping and dwell-time dependent friction”, *Multibody System Dynamics*, vol. 11, pp. 209–233, 2004.
- [27] J. Spillman and M. Teschner, “Corde: Cosserate rod elements for the dynamic simulation of one-dimensional elastic objects”, *SIGGRAPH Symposium on Computer Animation*, pp. 1–10, 2007.
- [28] J. Spillman, M. Becker, and M. Teschner, “Non-iterative computation of contact forces for deformable objects”, *Journal of WSCG*, vol. 15, no. 1-3, pp. 33–40, August 2007.
- [29] B. Heidelberger, M. Teschner, R. Keiser, Mueller M., and Gross M., “Consistent penetration depth estimation for deformable collision response”, *Proceedings of Vision, Modeling, Visualization*, pp. 339–346, November 2004.
- [30] H. Olsson, K. J. Åström, C. C. De Wit, M. Gafvert, and P. Lischinsky, “Friction models and friction compensation”, *European Journal of Control*, vol. 4, no. 3, pp. 176–195, 1998.
- [31] B. Armstrong-Hélouvry, P. Dupont, and C. Canudas de Wit, “A survey of models, analysis tools and compensation methods for the control of machines with friction”, *Automatica (Journal of IFAC)*, vol. 30, no. 7, pp. 1083–1138, July 1994.
- [32] C. C. de Wit, H. Olsson, K. J. Åström, and P. Lischinsky, “A new model for control of systems with friction”, *IEEE Transactions on Automatic Control*, vol. 40, no. 3, pp. 419–425, 1995.

- [33] D. A. Jr. Haesig and B. Friedland, "On the modeling and simulation of friction", *Journal of Dynamic Systems, Measurement, and Control*, vol. 113, no. 3, pp. 354–362, 1991.
- [34] Y. Gonthier, J. Mcphee, C. Lange, and J-C Piedbœuf, "A contact modeling method based on volumetric properties", *Proceedings of IDETC'05 2005 ASME Design Engineering Technical Conferences and 5th International Conference on Multibody Systems, Nonlinear Dynamics and Control*, September 24-28 2005.
- [35] Y. Gonthier, J. Mcphee, and C. Lange, "On the implementation of coulomb friction in a volumetric-based model for contact dynamics", *Proceedings of IDETC'07 2007 ASME Design Engineering Technical Conferences and 6th International Conference on Multibody Systems, Nonlinear Dynamics and Control*, September 4-7 2007.
- [36] Y. Gonthier, J. Mcphee, and C. Lange, "On implementing a bristle friction model in a contact model based on volumetric properties", *Multibody Dynamics 2007, ECCOMAS Thematic Conference*, June 25-28 2007.
- [37] D. Fuller, *Theory and Practice of Lubrication for Engineers*, John Wiley & Sons, second edition, 1984.
- [38] J. Brown, J.-C. Latombe, and K. Montgomery, "Real-time knot-tying simulation", *The Visual Computer: International Journal of Computer Graphics*, vol. 20, no. 2, pp. 165–179, May 2004.
- [39] GRL, "Grl simulator homepage", Website, last accessed September 2006, Provides information about the GRL ROV simulator as well as a download link.
- [40] J. A. Carretero and B. J. Buckham, "Simulation of submerged slack tethers and their interaction with the environment", in *Proceedings of the 23rd ASME International Conference on Offshore Mechanics and Arctic Engineering (OMAE)*, Vancouver, British Columbia, Canada, June 20-25 2004.
- [41] I. Lotan, F. Schwarzer, J. C. Latombe, and D. Halperin, "Efficient maintenance and self-collision testing for kinematic chains", in *Symposium on Computational Geometry*, Barcelona, Spain, June 2002, pp. 43–52.
- [42] M. C. Lin and J. F. Canny, "Efficient collision detection for animation", in *3rd Eurographics Workshop on Animation and Simulation*, September 1992.
- [43] L. Raghupathi, L. Grisoni, L. Faure, D. Marchal, M.-P. Cani, and C. Chaillou, "An intestine surgery simulator: Real-time collision processing and visualization", in *IEEE Transactions on Visualization and Computer Graphics*, November/December 2004, vol. 10.

- [44] B. J. Buckham, F. R. Driscoll, and M. A. Nahon, "Development of a finite element cable model for use in low-tension dynamics simulations", *Journal of Applied Mechanics*, vol. 71(4), pp. 476–485, 2004.
- [45] B. J. Buckham, *Dynamics Modelling of Low-Tension Tethers for Submerged Remotely Operated Vehicles*, PhD Thesis, Department of Mechanical Engineering, University of Victoria, Victoria, British Columbia, Canada, 2003.
- [46] B.E. Howard and J.M. Syck, "Calculation of the shape of a towed underwater acoustic array", *Journal of Oceanic Engineering*, vol. 17(2), pp. 193–201, 1992.
- [47] D. H. Eberly, *Game Physics*, Morgan Kaufmann Publishers, 2004.
- [48] S. C. Chapra and R. P. Canale, *Numerical Methods for Engineers*, McGraw-Hill College, 5th edition, July 2001.
- [49] J. A. Carretero and R. L. Uppuluri, "A novel optimization-based pruning strategy for concave minimum distance problems", in *Proceedings of the 2004 CSME Forum*, London, Ontario, Canada, June 2004, pp. 1–4.
- [50] R. L. Uppuluri, "Distance determination algorithms with novel pruning strategies for complex dynamics simulations", MScE thesis, Department of Mechanical Engineering, University of New Brunswick, Fredericton, New Brunswick, Canada, 2005.
- [51] Belegundu A. D. and T. R. Chandrupatla, *Optimization Concepts and Applications in Engineering*, Pearson Education, 2002.

Appendix A

Algorithm Flow Diagrams

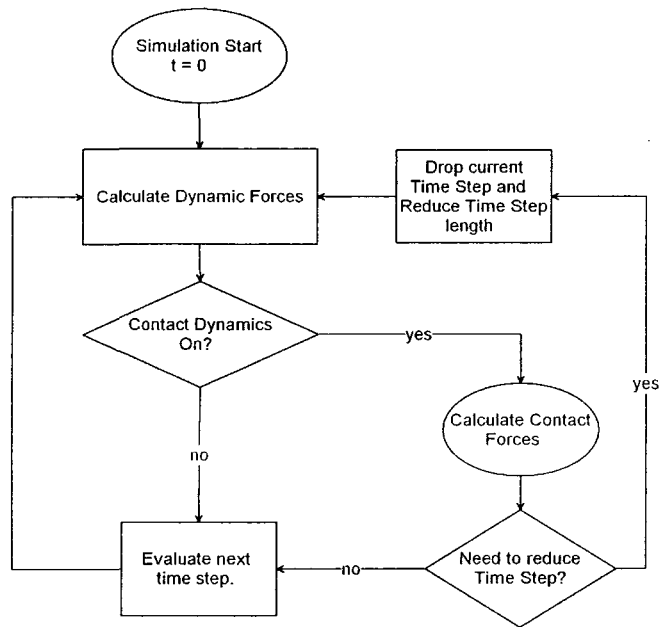


Figure A.1. Flow Diagram of the general simulation.

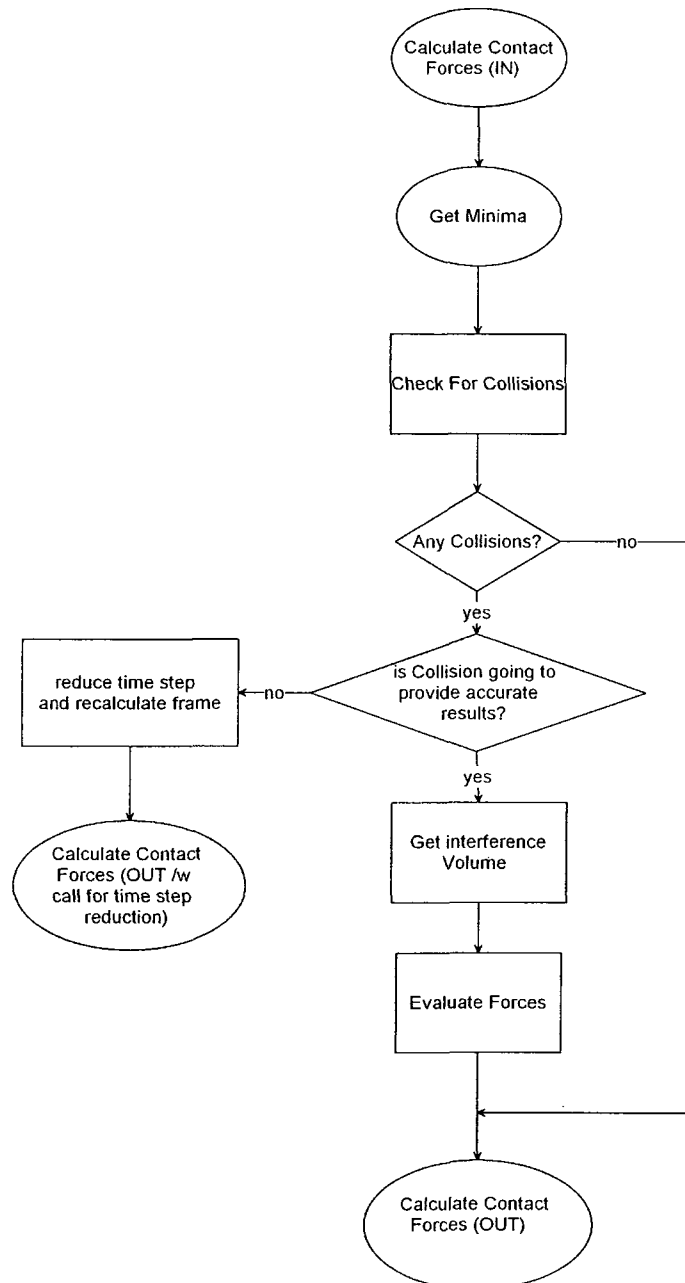


Figure A.2. Flow diagram of the Contact Dynamics Package.

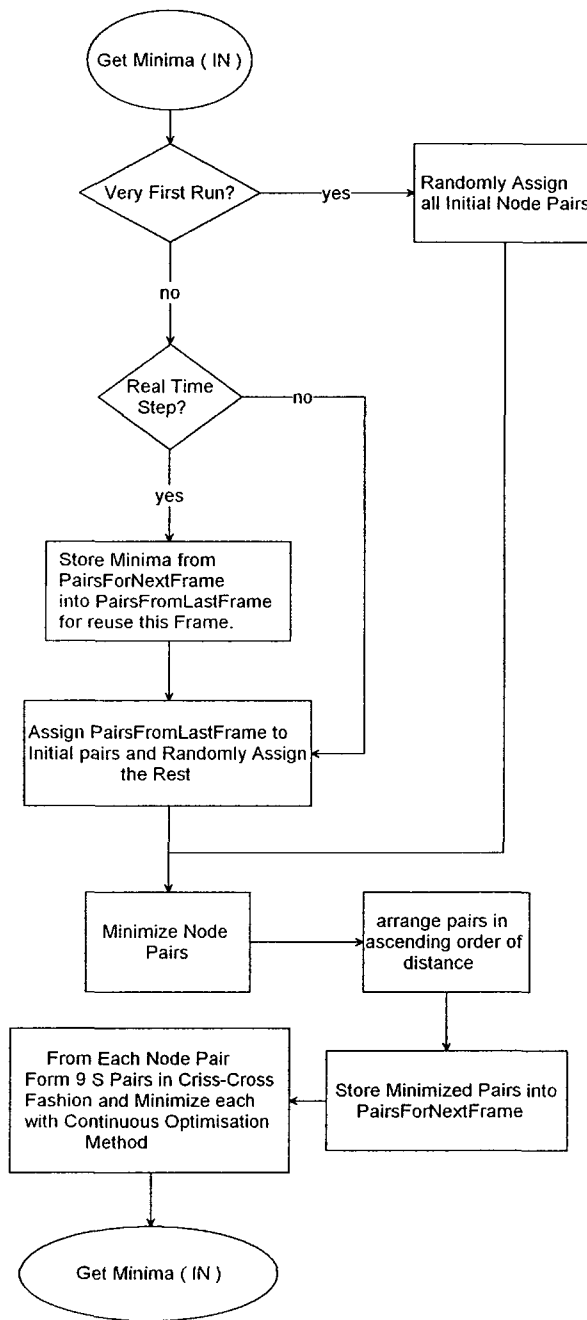


Figure A.3. Flow diagram of how minima are found.

Appendix B

The Minimum Separation Distance Gradient and Hessian Between Two Cubic Splines

$$\phi_1(s) = \frac{s_m - s}{Lu} \quad (\text{B.1})$$

$$\phi_2(s) = \frac{1}{6}(\phi_1^3 - \phi_1)(Lu)^2 \quad (\text{B.2})$$

$$\phi_3(s) = \frac{s - s_{m-1}}{Lu} \quad (\text{B.3})$$

$$\phi_4(s) = \frac{1}{6}(\phi_3^3 - \phi_3)(Lu)^2 \quad (\text{B.4})$$

$$\mathbf{X}(s) = \mathbf{X}^{m-1}\phi_1 + \mathbf{C}_x^{m-1}\phi_2 + \mathbf{X}^m\phi_3 + \mathbf{C}_x^m\phi_4 \quad (\text{B.5})$$

$$\mathbf{Y}(s) = \mathbf{Y}^{m-1}\phi_1 + \mathbf{C}_y^{m-1}\phi_2 + \mathbf{Y}^m\phi_3 + \mathbf{C}_y^m\phi_4 \quad (\text{B.6})$$

$$\mathbf{Z}(s) = \mathbf{Z}^{m-1}\phi_1 + \mathbf{C}_z^{m-1}\phi_2 + \mathbf{Z}^m\phi_3 + \mathbf{C}_z^m\phi_4 \quad (\text{B.7})$$

$$\delta X = X(s_1) - X(s_2) \quad (\text{B.8})$$

$$\delta Y = Y(s_1) - Y(s_2) \quad (\text{B.9})$$

$$\delta Z = Z(s_1) - Z(s_2) \quad (\text{B.10})$$

$$d = \sqrt{(\delta X)^2 + (\delta Y)^2 + (\delta Z)^2} \quad (\text{B.11})$$

$$\nabla d = \begin{bmatrix} \frac{\partial d}{\partial s_1} \\ \frac{\partial d}{\partial s_2} \end{bmatrix} \quad (\text{B.12})$$

$$\nabla^2 d = \begin{bmatrix} \frac{\partial^2 d}{\partial s_1^2} & \frac{\partial^2 d}{\partial s_1 \partial s_2} \\ \frac{\partial^2 d}{\partial s_2 \partial s_1} & \frac{\partial^2 d}{\partial s_2^2} \end{bmatrix} \quad (\text{B.13})$$

$$\frac{\partial d}{\partial s_1} = \frac{1}{d} \left(\delta X \frac{\partial X(s_1)}{\partial s_1} + \delta Y \frac{\partial Y(s_1)}{\partial s_1} + \delta Z \frac{\partial Z(s_1)}{\partial s_1} \right) \quad (\text{B.14})$$

$$\frac{\partial d}{\partial s_2} = \frac{1}{d} \left(-\delta X \frac{\partial X(s_2)}{\partial s_2} - \delta Y \frac{\partial Y(s_2)}{\partial s_2} - \delta Z \frac{\partial Z(s_2)}{\partial s_2} \right) \quad (\text{B.15})$$

$$\frac{\partial X(s_1)}{\partial s_1} = X^{m-1} \frac{\partial \phi_1(s_1)}{\partial s_1} + C_x^{m-1} \frac{\partial \phi_2(s_1)}{\partial s_1} + X^m \frac{\partial \phi_3(s_1)}{\partial s_1} + C_x^m \frac{\partial \phi_4(s_1)}{\partial s_1} \quad (\text{B.16})$$

$$\frac{\partial Y(s_1)}{\partial s_1} = Y^{m-1} \frac{\partial \phi_1(s_1)}{\partial s_1} + C_x^{m-1} \frac{\partial \phi_2(s_1)}{\partial s_1} + Y^m \frac{\partial \phi_3(s_1)}{\partial s_1} + C_x^m \frac{\partial \phi_4(s_1)}{\partial s_1} \quad (\text{B.17})$$

$$\frac{\partial Z(s_1)}{\partial s_1} = Z^{m-1} \frac{\partial \phi_1(s_1)}{\partial s_1} + C_x^{m-1} \frac{\partial \phi_2(s_1)}{\partial s_1} + Z^m \frac{\partial \phi_3(s_1)}{\partial s_1} + C_x^m \frac{\partial \phi_4(s_1)}{\partial s_1} \quad (\text{B.18})$$

$$\frac{\partial X(s_2)}{\partial s_2} = X^{m-1} \frac{\partial \phi_1(s_2)}{\partial s_2} + C_x^{m-1} \frac{\partial \phi_2(s_2)}{\partial s_2} + X^m \frac{\partial \phi_3(s_2)}{\partial s_2} + C_x^m \frac{\partial \phi_4(s_2)}{\partial s_2} \quad (\text{B.19})$$

$$\frac{\partial Y(s_2)}{\partial s_2} = Y^{m-1} \frac{\partial \phi_1(s_2)}{\partial s_2} + C_x^{m-1} \frac{\partial \phi_2(s_2)}{\partial s_2} + Y^m \frac{\partial \phi_3(s_2)}{\partial s_2} + C_x^m \frac{\partial \phi_4(s_2)}{\partial s_2} \quad (\text{B.20})$$

$$\frac{\partial Z(s_2)}{\partial s_2} = Z^{m-1} \frac{\partial \phi_1(s_2)}{\partial s_2} + C_x^{m-1} \frac{\partial \phi_2(s_2)}{\partial s_2} + Z^m \frac{\partial \phi_3(s_2)}{\partial s_2} + C_x^m \frac{\partial \phi_4(s_2)}{\partial s_2} \quad (\text{B.21})$$

$$\frac{\partial^2 d}{\partial s_1 \partial s_2} = \frac{\partial^2 d}{\partial s_2 \partial s_1} \quad (\text{B.33})$$

$$\frac{\partial^2 \phi_1(s_1)}{\partial s_1^2} = 0 \quad (\text{B.34})$$

$$\frac{\partial^2 \phi_2(s_1)}{\partial s_1^2} = \left(\phi_1(s_1) \left(\frac{\partial \phi_1(s_1)}{\partial s_1} \right)^2 \right) L_u^2 \quad (\text{B.35})$$

$$\frac{\partial^2 \phi_3(s_1)}{\partial s_1^2} = 0 \quad (\text{B.36})$$

$$\frac{\partial^2 \phi_4(s_1)}{\partial s_1^2} = \left(\phi_3(s_1) \left(\frac{\partial \phi_3(s_1)}{\partial s_1} \right)^2 \right) L_u^2 \quad (\text{B.37})$$

$$\frac{\partial^2 \phi_1(s_2)}{\partial s_2^2} = 0 \quad (\text{B.38})$$

$$\frac{\partial^2 \phi_2(s_2)}{\partial s_2^2} = \left(\phi_1(s_2) \left(\frac{\partial \phi_1(s_2)}{\partial s_2} \right)^2 \right) L_u^2 \quad (\text{B.39})$$

$$\frac{\partial^2 \phi_3(s_2)}{\partial s_2^2} = 0 \quad (\text{B.40})$$

$$\frac{\partial^2 \phi_4(s_2)}{\partial s_2^2} = \left(\phi_3(s_2) \left(\frac{\partial \phi_3(s_2)}{\partial s_2} \right)^2 \right) L_u^2 \quad (\text{B.41})$$

Appendix C

Obtaining the Normal Contact Force through Hertz' Theory of General Contact

When two bodies with general curvature¹ come into contact, Hertz' theory of general contact says that the contact patch is elliptical in shape and the pressure is distributed as a semi-ellipsoid. The equations that follow were taken from [1].

The total applied contact force can be formulated as a relationship to the maximum pressure, p_{max} in the distribution, which is located at the centre of the contact patch:

$$f_n = \frac{2}{3}\pi ab p_{max} \quad (\text{C.1})$$

¹Hertz' Theory of General Contact assumes that all curvatures have an associated radius of curvature which defines the quadratic surface.

If however, the normal force is known along with the shape and material of the contacting surfaces, one can estimate the dimensions of the contact patch:

$$a = k_a \sqrt[3]{\frac{3F(m_1 + m_2)}{4A}} \quad (\text{C.2})$$

$$b = k_b \sqrt[3]{\frac{3F(m_1 + m_2)}{4A}} \quad (\text{C.3})$$

$$m_1 = \frac{1 - \nu_1^2}{E_1} \quad (\text{C.4})$$

$$m_2 = \frac{1 - \nu_2^2}{E_2} \quad (\text{C.5})$$

where a is the half-width of the major axis of the elliptical contact patch and b is its minor axis counterpart, k_a and k_b can be found in Table (C.1), and ν_i and E_i are the Poisson's ratio and Young's modulus of the material of object i . A and B can be obtained using the following equations:

$$A = \frac{1}{2} \left(\frac{1}{R_1} + \frac{1}{R'_1} + \frac{1}{R_2} + \frac{1}{R'_2} \right) \quad (\text{C.6})$$

$$B = \frac{1}{2} \left[(\mathfrak{R}_1)^2 + (\mathfrak{R}_2)^2 + 2(\mathfrak{R}_1)(\mathfrak{R}_2) \cos(2\alpha) \right]^{\frac{1}{2}} \quad (\text{C.7})$$

$$\mathfrak{R}_1 = \frac{1}{R_1} - \frac{1}{R'_1} \quad (\text{C.8})$$

$$\mathfrak{R}_2 = \frac{1}{R_2} - \frac{1}{R'_2} \quad (\text{C.9})$$

$$\phi_{Hz} = \cos^{-1} \left(\frac{B}{A} \right) \quad (\text{C.10})$$

where R_1 and R'_1 represent the maximum and minimum surface radii of curvatures for object 1, and α is the angle for orientation difference between R_1 and R_2 .

By rearranging either Equation (C.2) or (C.3), the load required to deform the tether as to form a contact patch of dimensions a and b can be obtained.

Table C.1. Factors for use in equations C.2 and C.3.

ϕ_{Hz}	0	10	20	30	40	50	60	70	80	90
k_a	∞	6.612	3.778	2.731	2.136	1.754	1.486	1.284	1.128	1
k_b	0	0.319	0.408	0.493	0.567	0.641	0.717	0.802	0.893	1

Curriculum Vita

Surname: Roy

Given Names: André

Date of birth: April 1, 1981

Place of birth: Canada

Educational Institutions Attended:

University of New Brunswick

1999 to 2008

Degrees Awarded

BSc.E (Mechanical) University of New Brunswick, Fredericton

2005

Honours and Awards:

MITACS Internship (University of Victoria & DSA Ltd.) Feb. 2008 to May 2008

Research Assistantship (University of New Brunswick) Sept. 2005 to Apr 2008

Teaching Assistantship (University of New Brunswick) Sept. 2005 to Apr 2008

Publications:

A. R. Roy, J. A. Carretero, B. J. Buckham, Detecting Tether Self-Collisions in Tethered ROV Simulations, Proceedings the 26th conference on Offshore Mechanics and Arctic Engineering, June 10-15th, 2007, San Diego, California ,USA

A. R. Roy, J. A. Carretero, Simulation of Contact Forces Between Cylindrical Slack Flexible Tethers in Self-Collisions, Proceeding of the 2007 CCToMM Symposium on Mechanisms, Machines, and Mechatronics, May 31 - June 1, 2007, Montreal, Canada

A. R. Roy, J. A. Carretero, B. J. Buckham, Detecting Tether Self-Collisions Continuously in Tethered ROV Simulations, submitted to Journal of Offshore Mechanics & Arctic Engineering.

A. R. Roy, J. A. Carretero, A Compliant Contact Dynamics Model for Volumetric Cubic-Spline Cables Based on Interference Geometry, in Preparation

A. R. Roy, J. A. Carretero, Validation of an Interference Volume Based Contact Model for Long Strand-like Objects, in Preparation.