Vít Vondrák; Zdeněk Dostál; Michael Damsgaard; John Rasmussen
Solution of muscle-bone contact problem in biomechanics

In: Jan Chleboun and Petr Přikryl and Karel Segeth (eds.): Programs and Algorithms of Numerical Mathematics, Proceedings of Seminar. Dolní Maxov, June 6-11, 2004. Institute of Mathematics AS CR, Prague, 2004. pp. 284–289.

Persistent URL: `http://dml.cz/dmlcz/702806`

Terms of use:

© Institute of Mathematics AS CR, 2004

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.

This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library* `http://dml.cz`

SOLUTION OF MUSCLE-BONE CONTACT PROBLEM IN BIOMECHANICS *

Vít Vondrák, Zdeněk Dostál, Michael Damsgaard, John Rasmussen

1. Introduction

Mechanical analysis of the musculoskeletal system of the human is a subject of importance to fundamental research as well as to practical engineering design and medical applications. The AnyBody Modeling System, or in short just AnyBody, is a software that is aimed at analysis of this particular type of systems. For instance, we may want to investigate the basic function of the body, or we may want to design an rehabilitation exercise, a man-driven or -operated machine or tool. In either of these cases and many others it will be a great asset to know motion and forces accurately during a given exercise. Computational models can assist us by quantifying certain parameters and estimating performance measures that cannot possibly be measured.

The musculoskeletal systems of humans or animals are mechanically very complex due complex shapes and material properties of the structures. In general such a system must be highly simplified in order to achieve efficient computational models. In AnyBody the soft tissue is assumed to be fixed rigidly onto the bony structures providing a model with a rigid-body structure and well-defined movement patterns governed by kinematical joints. The force-carrying function of the soft tissue, i.e., ligaments and muscles, are modeled by attaching elements with particular properties to the rigid bodies. Figure 1 shows an example of such a model.

The actuators of the system are the muscles and they are activated by the Central Nervous System (CNS) by mechanisms that are not understood well enough for detailed modeling. A fundamental problem is that there are more muscles than necessary to drive the degrees of freedom of the system. This implies that there are infinitely many muscle activation patterns that provide dynamical equilibrium. Therefore, the model of the muscle recruitment mechanisms must be based on assumptions, typically some kind of optimality condition. This problem is often referred to as the redundancy problem of the muscle recruitment.

AnyBody currently allows for inverse dynamic analysis using an efficient min/max criterion [1] for solving this redundancy. This criterion assumes that the CNS minimizes the maximal stress on the muscles. The efficiency of this approach allows

*This work was supported by grant No. 101/02/0072 of the Grant Agency of the Czech Republic, No. S3086102 and No. ET400300415 of the Grant Agency of the Academy of Sciences of the Czech Republic.

Fig. 1: *A bicycling full-body model made in AnyBody by the AnyBody Research Group [2]. It comprises more than 400 muscles and all the major segments of the upper and lower extremities of the body as well as the spine. More than 150 of these use the string-contact model described in this paper. The bicycle is modeled by a single segment, the crank, whereas the rest is pure visualization.*

for making and analyzing models of the size depicted in Figure 1 on an ordinary PC. In an inverse dynamics approach the motion must be completely prescribed as input, contrarily to ordinary forward dynamics, which however is much more computational demanding. In practical applications the motion is however often known from measurements.

In both forward and inverse dynamic model we need to generate the equations of motion, i.e., the Newton-Euler equations for each rigid segment. In order to find the contribution for each muscle we must have a mathematical model for the line-of-action of the muscles. The complex shapes of muscles are typically idealized to simple line models that directly gives the line-of-action. Particular complex shapes are divided into several parts represented by the simple line approach. For many cases a line between the muscle origin and insertion or a piecewise line with a number of body-fixed via-points provides an adequate representation of muscle geometry. However, there are a significant amount of the muscles in the body that wrap and slide over obstacles.

In this paper, we shall demonstrate a much more general method. We still stay with the shape of a string, but we find the path of the string using a general numerical method. We model the string by the finite element method and solve this in a contact algorithm. This approach does not put any limits to the shape and number of obstacles we can deal with. Moreover notice that this formulation is not restricted to deal with a string shaped muscle. More complex geometric shapes can perhaps be introduced at the cost of a more complex finite element model, but for the sake of efficiency we must currently stay with the string shape.

To indicate the need for this type of geometric modeling of the muscles, it can be noticed that more than one third of the muscles in the model in Figure 1 is using this type of muscle wrapping model. This also explains the need for efficiency of muscle wrapping model, since it must be solved for each muscle in each time-step of the dynamic analysis.

2. Muscle wrapping

We shall identify a homogenized muscle as an elastic string between two points in 3-D. Let the string be parameterized by a parameter $p \in R$ and each point of the string be represented by a vector $\Phi(p) = [x(p), y(p), z(p)]$ where the parameter $p \in \langle p_0, p_n \rangle$. The position of the string can be found by minimizing potential energy of the string that has the form

$$J(x, y, z) = \frac{1}{2} \int_{p_0}^{p_n} \left[{x'}^2(p) + {y'}^2(p) + {z'}^2(p) \right] dp. \tag{1}$$

While the solution of such a problem is trivial, the problem becomes more difficult when we introduce a rigid obstacle between string endpoints as depicted in Figure 2. In this case, the final position of the string is governed by the shortest path between the given endpoints wrapping around the rigid obstacle. For human body simulations, we identify a rigid obstacle as a bone, and muscle wrapping around a bone could be solved as the minimization of the potential energy with conditions of non-penetration of the rigid obstacle.

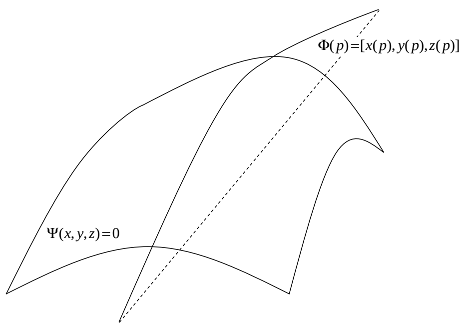


Fig. 2: *Contact of a string and surface.*

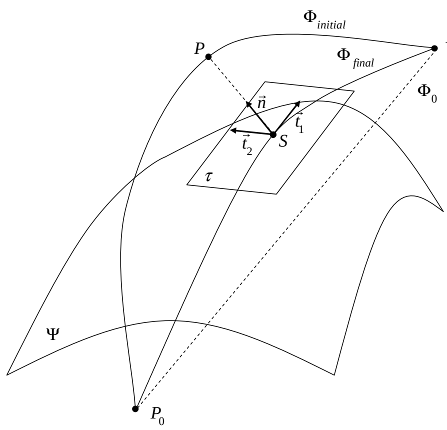


Fig. 3: *Linearized conditions of non-penetration.*

The linearized conditions of non-penetration, proposed for example by Kikuchi and Oden in [5] for contact problems in elasticity, seem to be suitable for efficient solution of described contact problems as being proved for example by the authors in [6] or [7] in such difficult problems as problems of contact shape optimization. Let us assume the situation shown in Figure 3. Do not let the string in an initial position

$\Phi_{initial}$ penetrate the surface Ψ, and let the position of the surface Ψ be between the initial string position $\Phi_{initial}$ and a straight line Φ_0 as depicted in Figure 3. We can interpret the conditions of non-penetration in such a way that the position of any point $P = \Phi(p)$ of the string Φ must be above the surface. The surface Ψ can be locally linearized at a point S by the tangential plane $\tau(S)$ that is given by two tangential vectors $\vec{t_1}(S)$, $\vec{t_2}(S)$ or by the outer unit normal vector $\vec{n}(S)$ to the surface Ψ at the point S. For simplification of further expressions we shall drop (S) from the notation of tangential and normal vectors. Hence the linearized condition of non-penetration can be defined as non-penetration of the tangential plane τ by the string point P, the position of which is in the normal direction $\vec{n}$ to the surface Ψ at point S as it is drawn in Figure 3 by the dashed line. This uniquely identifies the pair of possible contact points P and S. Then the condition of non-penetration can be represented by the following simple inequality

$$\vec{n}^T(P - S) \geq 0. \tag{2}$$

Prescribing this condition of non-penetration for all points on the string we obtain the linearized conditions of the non-penetration of a surface by a string.

Now, let us discretize the string for example by the finite element method. The potential energy of the discrete string has the form

$$J(u) = \frac{1}{2}u^T K u - u^T f \tag{3}$$

where $u = [x_1, \ldots, x_n, y_1, \ldots, y_n, z_1, \ldots, z_n]^T$ is the vector of unknown positions of n string points $\Phi_i = [x_i, y_i, z_i]$, $i = 1, \ldots, n$, K is a tridiagonal $3n \times 3n$ stiffness matrix and f is a $3n$ vector applying external load and fixing positions of the end points. Using a discretized string, we prescribe conditions of non-penetration only for a finite number of string points $\Phi_1, \ldots, \Phi_n$. That means that for each string point Φ_i we have to find corresponding contact pair point S_i on the surface Ψ and the normal vector $\vec{n_i}$. Then we prescribe n conditions of non-penetration in the form

$$\vec{n}_i^T(\Phi_i - S_i) \geq 0, \quad i = 1, \ldots, n. \tag{4}$$

Defining $3n$ dimensional vectors $c_i = [0, \ldots, 0, -n_{i1}, 0, \ldots, 0, -n_{i2}, 0, \ldots, 0, -n_{i3}, 0, \ldots, 0]^T$ with the normal vector $\vec{n}_i$ entries at positions n, $2n$ and $3n$, and denoting $d_i = -n_{i1}S_{i1} - n_{i2}S_{i2} - n_{i3}S_{i3}$, we can rewrite conditions (4) into the vector form $c_i^T u \leq d_i$, $i = 1, \ldots, n$. Finally, assembling matrix C and vector d from row vectors c_i^T and values d_i we can introduce the matrix form of conditions of non-penetration

$$Cu \leq d. \tag{5}$$

Now, we can define the contact problem of a string and a rigid obstacle as the following minimization problem

$$\min J(u) \quad \text{subject to} \quad Cu \leq d. \tag{6}$$

We should notice that this appears to be a quadratic programming problem with linear inequality constraints that could be solved very efficiently using its dual form. The dual form then has the form of the quadratic programming problem with simple bounds. All the advantages of dual formulations are described in [8], [9] in more details.

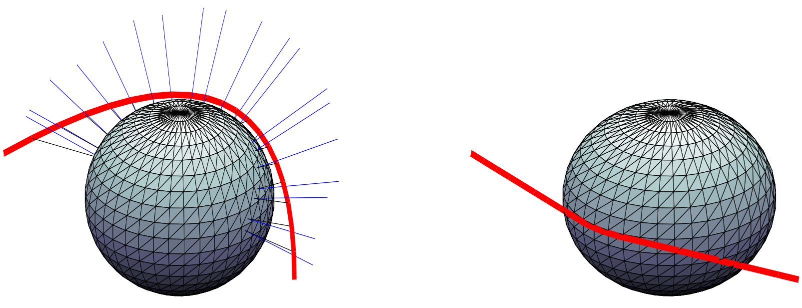
Unfortunately, solving contact problem (6), we do not usually obtain expected solution because some string points are penetrating surface and some of them are too high over the surface even though their contact with surface is expected. This is caused by the inaccuracy of the linearized contact conditions, which prescribe non-penetration of the surface only in the normal direction at the contact pair point on the surface. Therefore, we have to construct new conditions of non-penetration for the new string position and solve again the contact problem with updated contact conditions. We repeat this loop of contact condition improvements until we do not obtain a satisfactory solution. As the suitable stopping criterion for this loop a sufficiently small relative change of the string position could be used, i.e. $||u^{k+1} - u^k||/||u^k|| < \epsilon$. Now we can summarize the iterative algorithm for solution of the contact problem of a string and a surface:

1. Initial step. Assembling of stiffness matrix K and vector f.
   Initial position of the string u^0, $k = 0$.

2. Building contact conditions for string position u^k, i.e. C^k, d^k.

3. Find solution u^{k+1} of the contact problem
$$\min \frac{1}{2} u^T K u - u^T f \text{ subject to } C^k u \leq d^k.$$

4. If stopping criterion is fulfilled, then u^{k+1} is solution. Otherwise, set $k = k + 1$ and return to step 2.



a) Initial string position b) Final string position

Fig. 4: *Contact problem of a string and STL-sphere.*

The algorithm described above was successfully tested on practical examples. We shall present here at least the contact of a string and a discrete sphere. The sphere is described as the general STL-surface used by almost all CAD systems. The sphere was discretized by 686 points and these were used as vertices of 1368 triangles. The finite element string mesh was defined by 20 string nodes. In Figure 4 you can see the initial and final string position that was found in 9 iteration steps of contact condition searching algorithm. The string length was reduced from initial 4.888394 to 4.132996. The perpendicular lines to the sphere in the initial position represent the outer unit normal vectors at potential surface contact nodes.

References

[1] Damsgaard M., Rasmussen J., Christensen S.T.: *Inverse dynamics of musculo-skeletal systems using an efficient min/max muscle recruitment model.* Proceedings of IDETC: 18-th Biennial Conference on Mechanical Vibration and Noise, Pittsburgh, September 9-13, 2001.

[2] The AnyBody Research Project and the AnyBody Repository, `http://www.anybody.aau.dk`.

[3] AnyBody Tech. A/S, `http://www.anybodytech.com`.

[4] Rasmussen J., Damsgaard M., Voigt M.: *Muscle recruitment by the min/max criterion – a comparative study.* Journal of Biomechanics, **34**, 3, 2001, 409–415.

[5] Kikuchi N., Oden J.T.: *Contact problems in elasticity: A study of variational inequalities and finite element methods.* SIAM, Philadelphia, 1988.

[6] Vondrák V., Dostál Z., Rasmussen J.: *Duality based contact shape optimization.* ZAMM, Z. Angew. Math. Mech., **3**, 81, 2001, 703–704.

[7] Vondrák V.: *Methods for solution of contact shape optimization problems.* Ph.D. thesis, Palacky University in Olomouc, Dept. of Mathematical Analysis and Mathematical Applications, September 2002.

[8] Dostál Z., Gomes F.A.M., Santos S.A.: *Solution of contact problems by FETI domain decomposition.* Computer Meth. Appl. Mech. Engrg., **190**, 2000, 1611–1627.

[9] Dostál Z., Horák D.: *Scalability and FETI based algorithm for large discretized variational inequalities.* Mathematics and Computers in Simulation, **61**, 3-6, 2003, 347–357.