Interpolating Muscle Forces in an Inverse Dynamics Approach

Charles Pontonnier, Georges Dumont

HAL Id: inria-00535796

https://inria.hal.science/inria-00535796

Submitted on 5 Apr 2013

Interpolating Muscle Forces in an Inverse Dynamics Approach

C. Pontonnier[1], G. Dumont[2]

1. ABSTRACT

The goal of our work is obtain a method usable to estimate in real-time muscle forces involved in the motion of the human at work. We want to use this method in order to test the ergonomics of the workstations during the conception process. This test will take place in a virtual scene in which we immerse the operator. For this reason, the method has to meet the trade-off between accuracy and performance. The application case of this method is based on the upper extremity. Our idea is that interpolation is more efficient in terms of computation time than optimization. In this article, we first present how we have built a database with the results of a muscular estimation based on an inverse dynamics approach and an optimization step. In a second time we present our interpolation algorithm, which is based on Delaunay tessellation of each joint muscle situation. For a given frame of motion capture, we search in the Delaunay tessellation that classifies the database the nearest neighbors in terms of joint position, speed and acceleration. From these nearest neighbors we perform a weighted interpolation of the muscle forces involved in the joint motion. Some results for the elbow flexion/extension joint are presented in order to discuss these results and compare with the classical approach. Further, we compare computation time since it is a fundamental point for immersion in virtual reality. At last we conclude on the perspectives opened by this new algorithm.

2. INTRODUCTION AND CONTEXT

Mainly inverse dynamics methods for the estimation of muscle forces are based on an optimization step, allowing the resolution of the redundant problem defined as:

$$c - \sum_m F_i R_i = 0 \qquad (1)$$

In which C is the joint torque associated with a joint, and $\sum_m F_i R_i$ is the sum of the contribution of each muscle associated with the joint. So there is more unknown than equations. If this method has proven its accuracy [1][2], it implies a very long computation time (10 minutes for 3 seconds of simulation for OpenSim for example).
We want to use the estimation of the muscle forces in order to test the ergonomics of the workstations during the conception process, because muscle forces are a good indicator in order to define the quality of a motion or a posture [3][4]. This test will take place in a virtual scene in which we immerse the operator. For this reason, the method has to meet the trade-off between accuracy and performance.

[1]Ph.D Student, Bunraku Project team, IRISA, Campus de Beaulieu, 35042 Rennes Cédex, France

[2] Assistant Professor, Head of Bunraku Project team, IRISA, Campus de Beaulieu, 35042 Rennes Cédex, France. ENS Cachan Antenne de Bretagne, Campus de Ker Lann, 35170 Bruz, France

In order to obtain accurate information on the muscle forces in real time, we want to develop an alternative method that uses interpolation instead of optimisation. The presentation of this method is the purpose of this article.

The article is structured in four parts. First we present the conception of the database for a sample subject. Then we present the interpolation method. Results are provided for a sample motion and discussed. At last we conclude and present the perspectives of this work.

3. DATABASE CONCEPTION

For a sample subject, we define some sample motions to perform and capture these motions with a motion capture system -VICON© system. The subject has to perform flexion/extension of the elbow at different speed, without any load carrying. Table 1 presents the subject's anthropometric characteristics:

Subject	Gender	Height (mm)	Weight (kg)
Subject1	Male	1840	72

Table 1: subject's anthropometric characteristics

The motion database represents the range of motion in which we can interpolate muscle forces. On the top right, the database is represented in the $(q, \dot{q}, \ddot{q})$ joint space. On the left, from top to bottom:
- Database in the $(\dot{q}, \ddot{q})$ plane
- Database in the $(q, \dot{q})$ plane
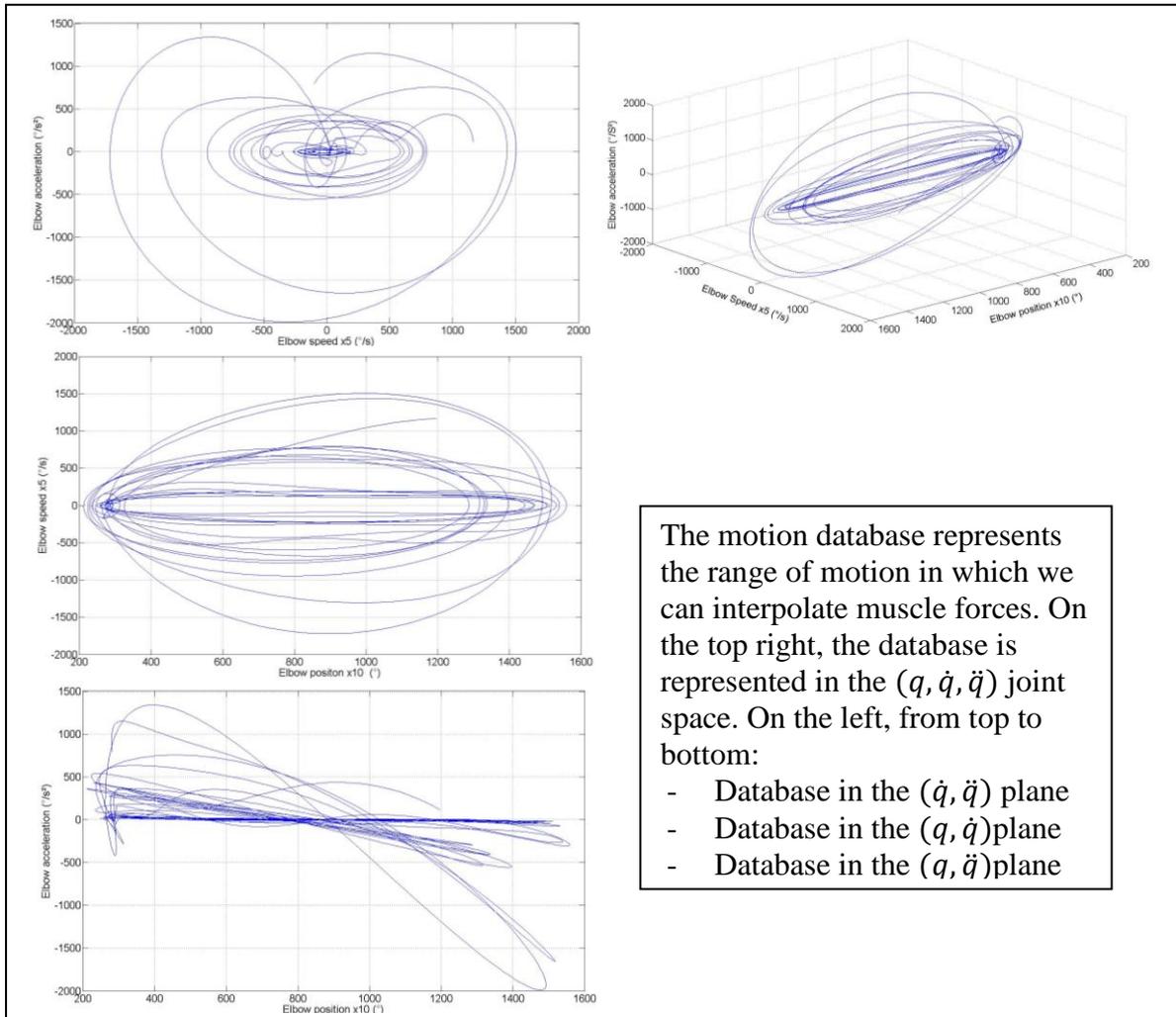- Database in the $(q, \ddot{q})$ plane

Fig. 1: Motion database represented in the $(q, \dot{q}, \ddot{q})$ joint space.

Once the subject has done these motions, we use the method that is fully described in

[5] in order to estimate the muscle forces associated to the motions for the elbow flexion/extension. This method includes an inverse kinematics step (computing joint coordinates), an inverse dynamics step (computing joint torques) and an optimisation step (computing muscle forces).

Fig.1 presents the joint motion database for the sample subject. As you can see, the database presents several lacks of information in certain parts of the $(q, \dot{q}, \ddot{q})$ space, especially in the $(q, \ddot{q})$ plane. We will see that this lack of information has an importance in the results. The database contains for this subject ~10 000 computed points.

For each frame :

$$\min f(F) = \sum_m \left(\frac{F_i}{(F_{max})_i} \right)^2$$

$$F = (F_1, F_2, F_3, F_4) \tag{2}$$

Under constraints :

$$h_1(F) = c - \sum_m F_i R_i = 0$$

$$g_i(F_i) = F_i - (F_{max})_i \le 0$$

Equation 2 resumes the optimisation problem computed. Estimated muscle forces are: Biceps (F1), Brachialis (F3), Brachioradialis (F4) and Triceps (F2).

4. INTERPOLATION METHOD

The interpolation is based on a Delaunay tessellation of the $(q, \dot{q}, \ddot{q})$ joint space and a search of the simplex including the interpolated point. We are using the joint space as a unique parameter of optimization, because the subject does not carry any load (that could be an additional parameter in future works, such as subject height or weight if we want to obtain muscles forces for a new subject from a database filled with other subjects). The three coordinates $(q, \dot{q}, \ddot{q})$ are completely defining the muscle estimation for a given load, so we believe that it is sufficient to interpolate on these three variables in order to obtain close results. Delaunay tessellation is based on the Quickhull algorithm [6] and is performed with MATLAB© 'delaunay3' method.
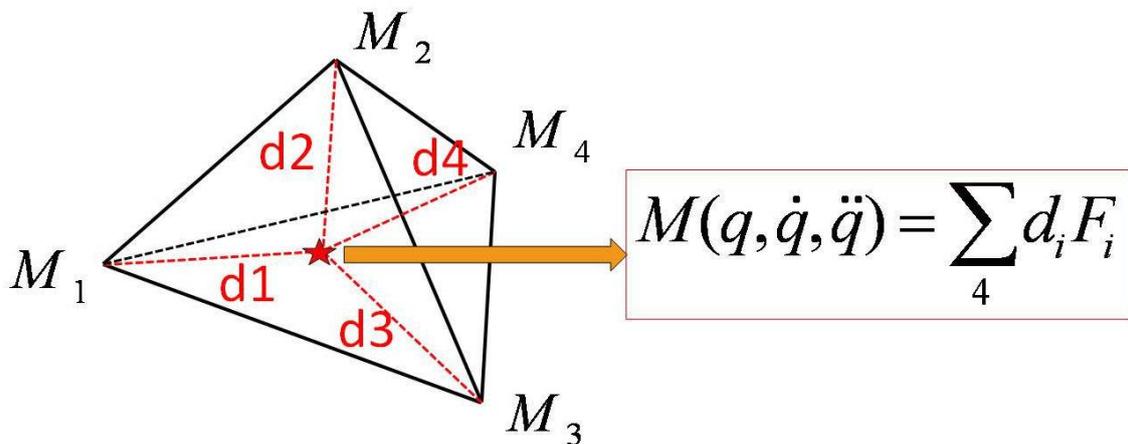

Fig.2: Linear interpolation of the muscle force F for a sample point $(q, \dot{q}, \ddot{q})$

Once the Delaunay tessellation is computed, the database is ready for interpolation. For each frame, we compute a search of the simplex enclosing the point to interpolate, using 'tsearch' method. At last, a linear interpolation is done using barycentric coordinates of the points of the enclosing simplex, as defined in Fig. 2.

5. RESULTS AND DISCUSSION

This part presents some comparisons between interpolated data and computed data (using optimisation method) for the sample subject. It also presents some comparisons of the computation frequencies of both methods.

Fig. 3 presents the interpolated forces in regard of the computed forces with the optimisation method. First of all, computed forces with the optimization method are not realistic, because we are using a very simple adaptation of the global method, in order to generate a lot of data. Particularly, Triceps force is quite constant during the whole motion. In fact, our simplified optimisation does not take into account the co-contraction of the antagonistic muscles. Also, brachioradialis has here an important activity that is not relevant. This is due to the maximum forces reachable for each muscle forces that are not here well defined. In future works about muscle forces interpolation, we will use a more complete optimisation step, already presented in these articles [7][8].

Despite of these limitations, we can see that interpolated data is very close from the computed data. When the point to interpolate is quite close from the edges of the simplex the interpolation is very accurate. On the curves, we can see that interpolated data is close on the increasing and decreasing forces, but do not have good results on the peak forces. This is due to a lack of data on these peaks, which are rarely reached during sample motions. This is a point that we can improve, by defining sample motions with more peak situations. In the same idea, some situations do not find any results, because the point to interpolate is out of the triangulation boundaries. It can be solved by increasing the number of points in the database. Otherwise, it is quite difficult to obtain a wide range of accelerations, especially for extreme positions. Natural motions do not imply this part of the joint coordinates space, so we do not need an important amount of points in this part of the space.

Fig. 4 illustrates the comparison between computation frequencies of both methods (optimization and interpolation) for two sample motions. The two methods have been computed on the same Laptop (Pentium 4 dual core 2,3GHz processor), in the same conditions. We can see that the global computation time is divided by more than 6. The optimisation method runs at approximately 20 Hz and interpolation method runs at more than 130 Hz. This is a very important result, because our goal is to estimate muscle forces in real time, in order to use this information in virtual reality scenes. With the interpolation method, we can see that it is possible, because the gap for real time applications is around 100 Hz. The interpolation method presented here is a prototypal method, and we think that code and implementation can be improved in order to decrease computation time.
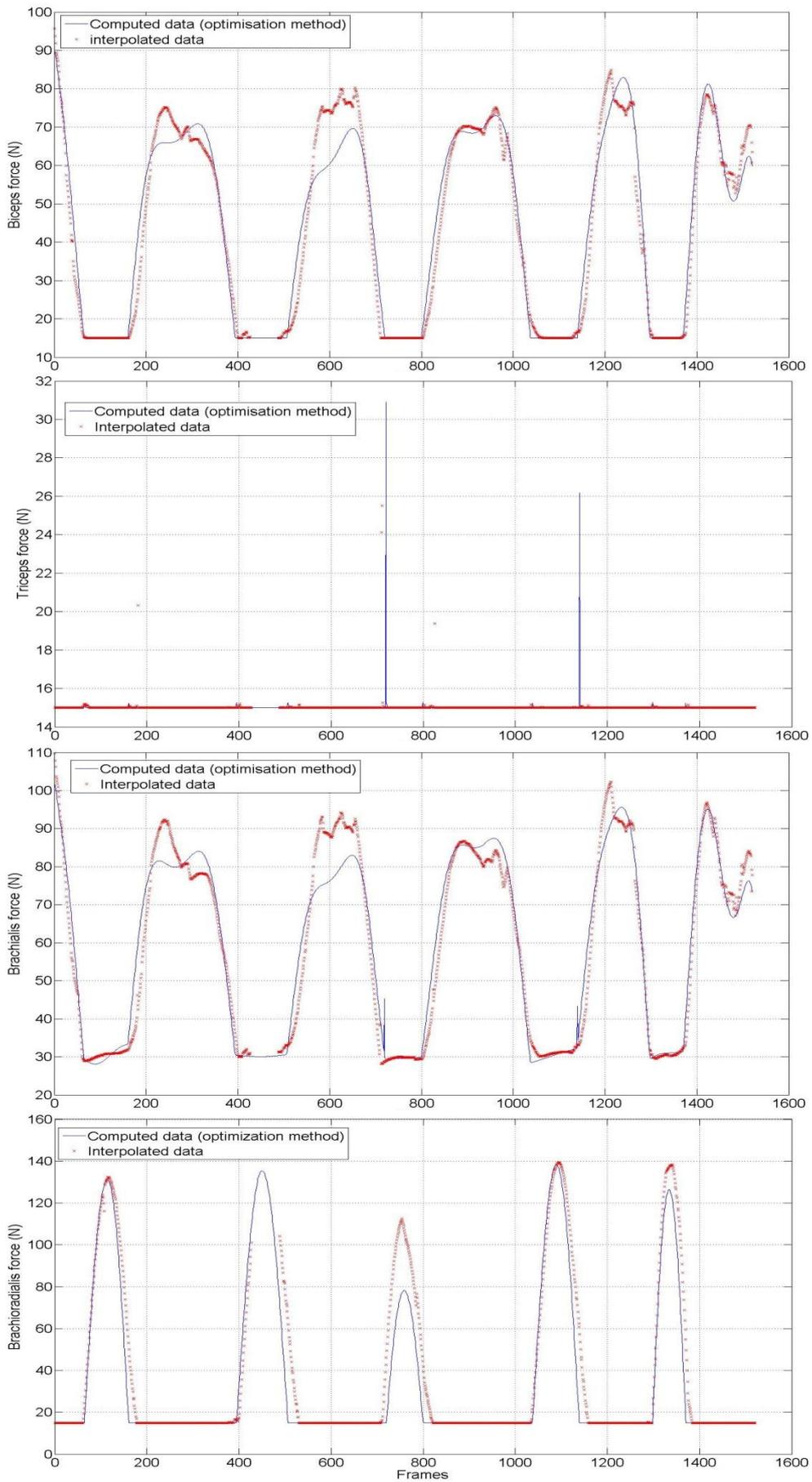
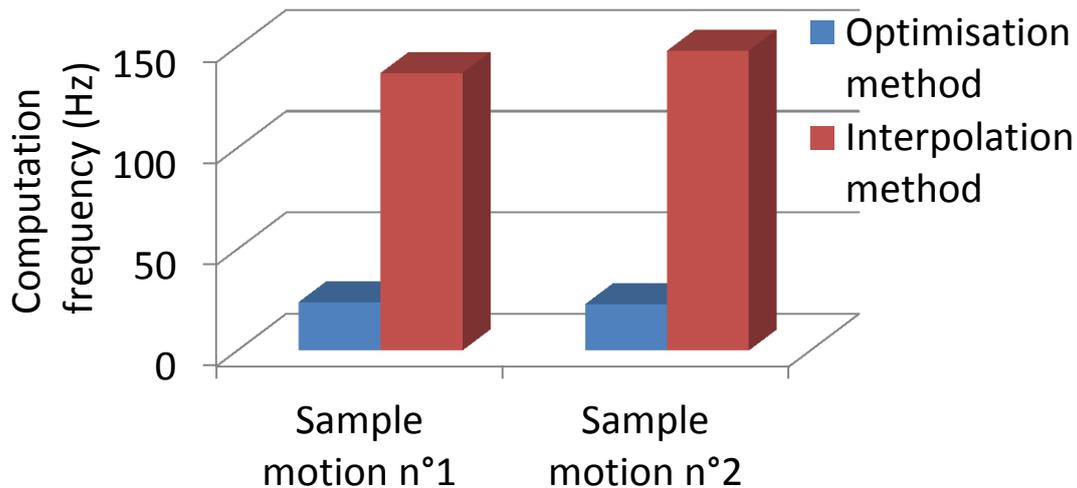**Fig. 3: Comparison between computed forces and interpolated forces**

Fig. 4: Comparison of the computation frequencies for the two methods

6. CONCLUSION AND PERSPECTIVES

In this article we presented an interpolation method usable to estimate muscle forces involved in the flexion extension for a sample subject. As we want to use the muscle forces estimation as information for ergonomics studies on virtual workstations, we want to obtain it in real time, or in a very quick time.

The method presents two phases: first, we fill in a database with captured motions and forces computed with an optimisation method. Then we are building a Delaunay tessellation of the joint coordinates space and we are interpolating the muscle forces from the edges of the simplex enclosing each point of a sample motion data that we want to analyse. The results presented for a sample motion are quite close from computed muscle forces, but limitations have been detected. For many situations, the database does not provide sufficient information in order to obtain a good interpolation, or does not have any data to interpolate the forces. These are some points we are actually exploring.

This method is a first step in a global approach. In fact we want to use the database with adimensional data, because we think that we can possibly interpolate muscle forces for a subject from other ones. This idea is very important because if it is done, the real time constraint for muscle forces estimation will be reached. So, we are defining some new spaces of interpolation in order to obtain the more complete and efficient data associated to muscle forces.

7. REFERENCES

1. Delp, S. L., Anderson, F. C., Arnold, A. S., Loan, P., Habib, A., John, C. T., Guendelman, E., Thelen, D. G.: OpenSim: Open-Source software to create and analyze dynamic simulations of movement biomedical engineering. IEEE Transactions,2007, Vol 54,1940-1950
2. Damsgaard, M., Rasmussen, J., Christensen, S.T., Surma, E., de Zee, M.: Analysis of musculoskeletal systems in the AnyBody Modeling System , Simulation Modelling Practice and Theory, 2006,Vol 14, 1100 - 1111

3. Bidault, F., Chablat, D., Chedmail, P., Pino, L. : Distributed approach for access and visibility task under ergonomic constraints with a manikin in virtual reality environment, Proceedings of the 10th IEEE International Workshop on Robot and Human Communication, 2001, 32-37

4. Ma, L., Chablat, D., Bennis, F., Zhang, W.: A new simple dynamic muscle fatigue model and its validation, International Journal of Industrial Ergonomics, 2009, 39, 211 - 220

5. Pontonnier, C. , Dumont, G.: From motion capture to muscle forces in the human elbow aimed at improving the ergonomics of workstations, Virtual and Physical Prototyping, 2010, 5: 3, 113-122

6. Barber, C. B., D.P. Dobkin, and H.T. Huhdanpaa: The Quickhull Algorithm for Convex Hulls, ACM Transactions on Mathematical Software, 1996, Vol 22, No. 4, 469-483

7. Pontonnier, C., Dumont, G.: Inverse Dynamics Method using Optimisation Techniques for the Estimation of Muscle Forces Involved in the Elbow Motion, International Journal on Interactive Design and Manufacturing (IJIDeM), 2009, 3, 227-235

8. Pontonnier, C., Dumont, G.: Motion Analysis of the Arm using Functional Anatomy, 3D Physiological Human 2009, Lecture Notes in Computer Sciences, 2009, Vol 5903