

Dealing with Modularity of Multibody Models

Antoine Muller, Charles Pontonnier, C Germain, Georges Dumont

► **To cite this version:**

Antoine Muller, Charles Pontonnier, C Germain, Georges Dumont. Dealing with Modularity of Multibody Models. Computer Methods in Biomechanics and Biomedical Engineering, Taylor & Francis, 2015, pp.2. 10.1080/10255842.2015.1069599 . hal-01174136

HAL Id: hal-01174136

<https://hal.inria.fr/hal-01174136>

Submitted on 8 Jul 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Dealing with Modularity of Multibody Models

A. Muller^{a,b,*}, C. Pontonnier^{a,b,c}, C. Germain^{a,b} and G. Dumont^{a,b}

^aIRISA/INRIA MimeTIC, Rennes, France; ^bENS Rennes, Bruz, France; ^cEcoles de Saint-Cyr Coëtquidan, Guer, France

Keywords: Complexity; Kinematical Loop; Scaling; Generic Model

1. Introduction

Musculoskeletal modeling is a very active field, and numerous musculoskeletal models can be found in the literature. These models are often parts of a body and even if common descriptive rules have been adopted by the research community (Wu and Cavanagh 1995), make them cohabit in wider simulations is still challenging. Indeed, a whole body model is an assembly of body parts that need to be adapted together. It results in non-adaptable models, difficult to use and modify as they were thought as body parts before being a whole body. Moreover, if a simulation of a simple model can give you insights about a complex phenomenon taking place in a specific limb or joint, an interesting feature would be to be able to refine or replace by a more accurate model the zone of interest. This is also something quite difficult to achieve with the current tools available in the literature. In the current paper, we propose to adapt a descriptive method used in robotics to describe and simulate a skeletal model and make it modular for the reasons evoked above. We also propose an illustration of the descriptive method by analyzing the elbow kinematics of an overhead throwing with two different forearm descriptions.

2. Methods

The descriptive method used is based on a systematic structural representation (Figure 1(a)). Except for the root, each segment integrates a joint and its adjoined body. Thus, the segment representation does not depend on the other segments connected to it.

This structural representation is described according to a hierarchical tree (Figure 1(b)). From the root, each solid owns one child and one sister. A library containing several body part models issued from the literature, which can correspond to a segment or a set of segments, is used to build the model. According to the user's choices (Spine A, Arm B,...), the corresponding hierarchical tree, which constitute the *model* is generated and adjoined functions needed for motion analysis are automatically created. This description allows the use of recursive functions.

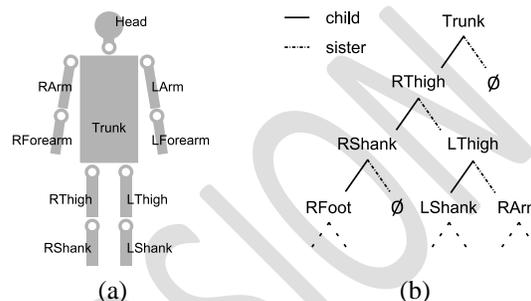


Figure 1 Hierarchical description of musculoskeletal model. (a) – structural representation; (b) – Hierarchical tree.

For example, forward kinematics of the model can be easily obtained thanks to a recursive function (Kajita et al. 2009). This function allows the model to be enriched with the transform matrix of each segment according to the joint coordinates :

```
function model = FKinematics(model,solid j)
```

```
Input: model and solid j
```

```
Output: model with the transform matrix of the solid j
```

```
if solid j =  $\emptyset$  then  
    Quit the function
```

```
else if  
    if solid j  $\neq$  root then  
        i <- get(model): mother of solid j  
        Calculate the matrix transformation of the solid j according to the matrix transformation of the solid i  
    end if  
end if
```

```
model = FKinematics(model,sister of j)  
model = FKinematics(model,child of j)
```

Each function call aims at computing the position and the orientation of the solid *j*. The forward kinematics is calculated hierarchically from the root to the extremities. Further, the method allows the implementation of kinematical closed loops. Such a feature is of interest for biomechanical models exhibiting complex anatomical relationships. Thus, the hierarchical tree can be considered as an equivalent open loop structure by virtually cutting each loop at an arbitrary joint.

A serial chain was obtained with k as the origin of the loop and j the extremity. The algorithm to obtain such closed loop relationship can be described as follows:

```
function Out = ClosedLoop(model)
Input: model
Output: closed loop relationship
for each terminal solid of the loop, j
    k <- get(model) : origin of the loop
    model = FKinematics(model,k)
     ${}^kT_j$  <- get(model) : transform matrix from
k to j
    Out =  ${}^kT_j - \mathbf{I}$  (Out = 0 : closed loop
relationships)
end for
```

This kind of function allows the kinematical closed loop relationships to be generated automatically from the hierarchical description. These relations can be used, for example, as hard or soft constraints in an optimization scheme.

3. Illustration

The aim of this part is to propose an illustration of the descriptive method by analysing the kinematics of an overhead throwing motion with two different models. The motion was recorded at 100 Hz using a Vicon motion capture system. The markers trajectories were filtered using a 4-th order Butterworth low pass filter with a cut-off frequency of 5 Hz and no phase shift.

Two whole-body models and their respective kinematics were created thanks to the method described above: one exhibiting a simple open-loop model of the arm (Holzbaur et al. 2005) and one exhibiting a more complex closed-loop model (Pennestri et al. 2007). Each model was selected among the models library. Inverse kinematics with kinematical calibration were automatically computed (Muller et al. 2015).

Figure 2 shows the elbow flexion angle which is invariant in both the open and closed loop models. The two angles were very similar (the correlation coefficient was higher than 0.99) and this illustrates the fact that the closed-loop relationship was handled properly by the method without additional development or processing.

4. Conclusions

The current abstract was a first contribution to the idea of making musculoskeletal models more modular and inter-compatible. Further investigations remain necessary to extend this study to inverse dynamics. The different masses distribution will probably generate discrepancies between joint torques and thus highlight the benefit of refining model in the zone of interest.

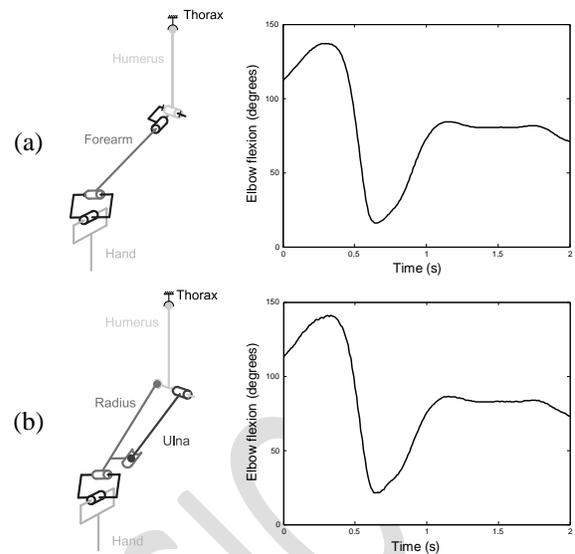


Figure 2 Models on the right arm and elbow flexion during a throwing motion. (a) – (Holzbaur et al. 2005) model; (b) – (Pennestri et al. 2007) model.

At last, we are currently extending the descriptive method to the muscular topology. Such a work is challenging because muscles paths belong to multiple body parts and muscles can act on multiple joints.

Acknowledgements

The authors wish to thank Anthony Sorel for his precious work on motion data. This study was partially funded by the ANR project ENTRACTE (Grant agreement: ANR 13-CORD-002-01).

References

- Holzbaur KR, Murray WM, Delp SL. 2005. A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control. *Annals of biomedical engineering*. 33(6), 829-840.
- Kajita S, Sakka S, Hirukawa H, Harada K, Yokoi K. 2009. *Introduction à la commande des robots humanoïdes: De la modélisation à la génération du mouvement*. Springer Science & Business Media.
- Pennestri E, Stefanelli R, Valentini PP, Vita L. 2007. Virtual musculo-skeletal model for the biomechanical analysis of the upper limb. *Journal of biomechanics*. 40(6), 1350-1361.
- Muller A, Germain C, Pontonnier C, Dumont G. 2015. A simple method to calibrate kinematical invariants: application to overhead throwing. *ISBS2015, Poitiers, France*.
- Wu G, Cavanagh PR. 1995. ISB recommendations for standardization in the reporting of kinematic data. *Journal of biomechanics*. 28(10), 1257-1261