Proc. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin S. and Adriaansen T. (Eds.), 10-12 Dec. 2003, Sydney

The Parameterization of Joint Rotation with the Unit Quaternion

Qiang Liu and Edmond C. Prakash

School of Computer Engineering, Nanyang Technological University, Singapore 639798
{pg04825433, asprakash}@ntu.edu.sg

Abstract. Unit quaternion is an ideal parameterization for joint rotations. However, due to the complexity of the geometry of S^3 group, it's hard to specify meaningful joint constraints with unit quaternion. In this paper, we have proposed an effective and accurate method to specify the rotation limits for joints parameterized with the unit quaternion. Joint constrains constructed with our method are adequate for most applications.

1 Introduction

In computer graphics and animation, articulated characters are among the most commonly used objects, and are a convenient model to synthesize moving humans and animals. An articulated character is a hierarchical structure consisting of a set of segments, connected by joints. To manipulate a character, we need to specify the position of the joints as well as their orientation. For the rotation motion, a joint has up to 3 degrees of freedom(DOF). It's easy to deal with a joint with only one DOF, such as interphalangeal joints, or two independent DOF, such as the knee joint. We can use one parameter to specify the angle of rotation for each DOF. For joints with 3 DOF, such as shoulder and hip, the rotation can be decoupled into a spherical motion, which has 2 DOF, and a twist motion, which is independent of the spherical motion. Therefore, what is most important and difficult for the parameterization of the joints is to parameterize the spherical rotation.

1.1 Background and Related Works

There are quite some well know methods of parameterization for spherical rotation. The most widely used one is Euler angles, which represent a general rotation as successive rotations about the three principal axes. Other parameterizations include rotation matrices [7], unit quaternion [11, 12], axis-angle (or exponential map) [3]. Good comparison and investigation of these parameterization for the purposes of animation of articulated bodies can be found in [1–3]. As noted by Grassia, each one possesses its advantages and drawbacks, with respect to the intended application[3]. In addition to these parameterizations, Huang and Prakash advocated a new sinus cone parameterization [5].

Among all these parameterizations, unit quaternion possesses some important advantages over other ones for the parameterization of spherical joint motion. First of all, it's free of singularity. Also, unit quaternion is a natural way to specify rotations. With methods developed by Shoemaker [11] and Kim et al. [6], it's easy to produce curves with high order continuity, such as splines and Bézier curves, in SO(3), and hence produce smooth and natural motions between key orientations. In addition, the result of two consecutive rotation can be calculated directly as the product of the two unit quaternion, which is not possible for other parameterization such as Euler angles.

However, because the geometry of the rotation space SO(3) is much more complicated than Euclidian space $\mathbb{R}^3$, it's very difficult to impose rotation limits directly in unit quaternion space. Lee showed how to specify conic, axial and spherical limits [8]. Yet, to model spherical joints exactly, more meaningful and complex limits are need.

In this paper, we will discuss how to parameterize spherical joints with unit quaternion so that complex joint limits can be imposed. The primary value of our work is that we present a method to specify the rotation limits of a joint parameterized with unit quaternion. Joint constrains specified by our method are complex and accurate enough for a wide range of applications.

2 Preliminary

Quaternions, discovered by Sir William Rowan Hamilton [4], provide a solid base to represent 3D rotation. In this section, we give a brief introduction to quaternions and unit quaternion.

2.1 Quaternion Basics

The four-dimensional space of quaternions is spanned by a real axis and three orthogonal imaginary axes, denoted by $\hat{\imath}, \hat{\jmath}$, and $\hat{k}$. A quaternion $\mathbf{q} = w + x\hat{\imath} + y\hat{\jmath} + z\hat{k}$ can be denoted as an ordered pair of real number and a vector: $\mathbf{q} = (w, \boldsymbol{v}) \in \mathbb{R} \times \mathbb{R}^3$, where $\boldsymbol{v} = (x\ y\ z)$. The product of two quaternions $\mathbf{q}_1$ and $\mathbf{q}_2$ can be written as:

$$\begin{aligned}\mathbf{q}_1\mathbf{q}_2 &= (w_1, \boldsymbol{v}_1)(w_2, \boldsymbol{v}_2)\\ &= (w_1w_2 - \boldsymbol{v}_1 \cdot \boldsymbol{v}_2, w_1\boldsymbol{v}_2 + w_2\boldsymbol{v}_1 + \boldsymbol{v}_1 \times \boldsymbol{v}_2).\end{aligned} \tag{1}$$

Quaternions form a non-commutative group under multiplication. A quaternion of unit length is called a unit quaternion, which can be considered as a point on the unit hyper-sphere $\mathbf{S}^3$. The inverse of a quaternion $\mathbf{q}$ is $\mathbf{q}^{-1} = (w, -x\ -y\ -z)/(w^2 + x^2 + y^2 + z^2)$.

Euler proved that, any orientation of a rigid body can be represented as a rotation about a fixed axis $\boldsymbol{v}$ by an angle θ from a reference orientation, where $\boldsymbol{v}$ is a 3-dimensional vector of unit length. With a unit quaternion $\mathbf{q} = (\cos\frac{\theta}{2}, \boldsymbol{v}\sin\frac{\theta}{2}) \in \mathbf{S}^3$, we can describe a rotation map

$$R_q(\boldsymbol{a}) = \mathbf{q}\mathbf{p}\mathbf{q}^{-1}, \qquad \text{for } p \in \mathbf{R}^3. \tag{2}$$

Here $\boldsymbol{a} = (x\ y\ z)$ is a unit vector in $\mathbb{R}^3$, and $\mathbf{q}$ is a unit quaternion whose real part is zero: $(0, x\ y\ z)$. Note that

$$R_{q_1}(R_{q_2}(\boldsymbol{a})) = R_{q_1 q_2}(\boldsymbol{a}). \tag{3}$$

which means that the effect of two consecutive rotations $\mathbf{q}_1$ and $\mathbf{q}_2$ can be calculated directly as the product of the two unit quaternions $\mathbf{q}_1\mathbf{q}_2$.

2.2 Exponential and Logarithmic Maps

One of the main connections between vectors and unit quaternions is the exponential mapping. Quaternion exponentiation is defined in the standard way as:

$$\exp(\mathbf{q}) = 1 + \frac{\mathbf{q}}{1} + \frac{\mathbf{q}^2}{2!} + \cdots + \frac{\mathbf{q}^n}{n!} + \cdots \tag{4}$$

If the real part of $\mathbf{q}$ is zero, then exponential mapping gives a unit quaternion which can be expressed in a closed-form:

$$\exp(\mathbf{q}) = \exp(0, \boldsymbol{v}) = \left(\cos\|\boldsymbol{v}\|, \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}\sin\|\boldsymbol{v}\|\right). \tag{5}$$

This map is not one-to-one. To define its inverse function, we limit the domain such that $\|\mathbf{v}\| < \pi$. Then, the exponential map becomes one-to-one and thus its inverse map $\mathbf{S}^3 \backslash (-1, 0\ 0\ 0) \rightarrow \mathbb{R}^3$ is defined as:

$$\log(\mathbf{q}) = log(w, \boldsymbol{v}) = \begin{cases} \frac{\pi}{2}\boldsymbol{v}, & \text{if } w = 0, \\ \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}\tan^{-1}\frac{\|\boldsymbol{v}\|}{w}, & \text{if } 0 < |w| < 1, \\ 0, & \text{if } w = 1. \end{cases} \tag{6}$$

3 Method

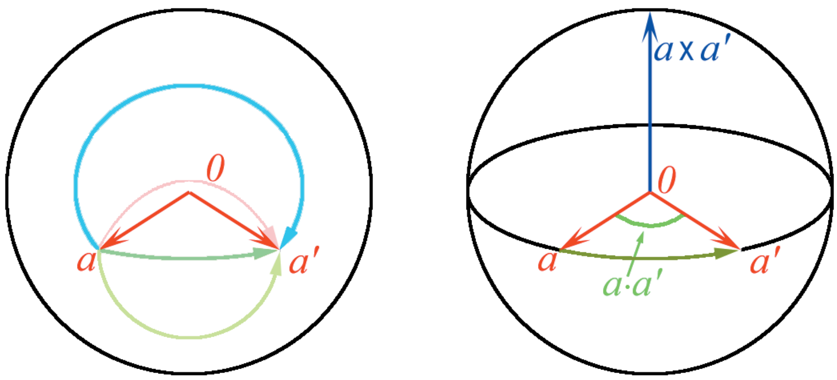
3.1 Standard Unit Quaternion

For an orientation donated by vector $\boldsymbol{a}$, giving a unit quaternion $\mathbf{q}$, we can get a unique resultant orientation $\boldsymbol{a'}$. However, the reverse is not always true. A unit quaternion $\mathbf{q} = (\cos\frac{\theta}{2}, \boldsymbol{v}\sin\frac{\theta}{2})$ represents a rotation around the vector $\boldsymbol{v}$ with the amount θ. For two orientations $\boldsymbol{a}$ and $\boldsymbol{a'}$, there are numerous ways of rotation from $\boldsymbol{a}$ to $\boldsymbol{a'}$, as illustrated in Fig. 1(a). To find a one-to-one relation between resultant orientation and unit quaternion, we need a standard form for all unit quaternions that transform a vector into the same orientation.

For any two vectors $\boldsymbol{a}$ and $\boldsymbol{a'}$, unless $\boldsymbol{a} = -\boldsymbol{a'}$, the transform from $\boldsymbol{a}$ into $\boldsymbol{a'}$ can be achieved by a single direct rotation. Direct rotation means that the angle of rotation is minimum among all the rotations from $\boldsymbol{a}$ to $\boldsymbol{a'}$, which is $\cos^{-1}(\boldsymbol{a} \cdot \boldsymbol{a'})$, and the axis of rotation is $\boldsymbol{a} \times \boldsymbol{a'}$, as illustrated in Fig. 1(b).

Thus, for a reference orientation $\boldsymbol{a}$, any unit quaternion $\mathbf{q} \neq (-1\ 0\ 0\ 0)$ can be standardized as:

$$\mathbf{q}_s = \left(\cos\frac{\psi}{2}, \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}\sin\frac{\psi}{2}\right) \tag{7}$$



(a) (b)

Fig. 1. (a)Different ways to map rotation from $\boldsymbol{a}$ to $\boldsymbol{a}'$. (b) Direct rotation from $\boldsymbol{a}$ to $\boldsymbol{a}'$.

in which

$$\psi = \cos^{-1}(\boldsymbol{a} \cdot \boldsymbol{a}'), \tag{8}$$

$$\boldsymbol{v} = \boldsymbol{a} \times \boldsymbol{a}', \tag{9}$$

and

$$(0, \mathbf{a}') = \mathbf{q}(0, \mathbf{a})\mathbf{q}^{-1}. \tag{10}$$

Let $k = \boldsymbol{a} \cdot \boldsymbol{a}'$, and $\boldsymbol{u} = (\boldsymbol{a} \times \boldsymbol{a}')/(\|\boldsymbol{a} \times \boldsymbol{a}'\|)$, (7), (8), and (9) can be simplified as

$$\mathbf{q}_s = \left(\sqrt{\frac{1+k}{2}}, \boldsymbol{u}\sqrt{\frac{1-k}{2}} \right). \tag{11}$$

Note that unit vector $\boldsymbol{u}$ always lies in the plane orthogonal to the reference vector $\boldsymbol{a}$.

3.2 Imposing Joint Constraints

A standard unit quaternion $\mathbf{q}_s = \left(\sqrt{\frac{1+k}{2}}, \boldsymbol{u}\sqrt{\frac{1-k}{2}} \right)$ represents the rotation of the reference vector around unit vector $\boldsymbol{u}$ with the amount of $\cos^{-1} k$. If we impose limits on the amount of rotation for each vector $\boldsymbol{u}$, the rotation limits of the spherical rotation are imposed. Therefore the joint constrain is denoted by

$$L(\boldsymbol{a}) = \left\{ \left(\sqrt{\frac{1+k}{2}}, \boldsymbol{u}\sqrt{\frac{1-k}{2}} \right) \;\middle|\; 0 \le \cos^{-1} k \le f(\boldsymbol{u}), \|\boldsymbol{u}\| = 1, \boldsymbol{u} \cdot \boldsymbol{a} = 0 \right\}, \tag{12}$$

or

$$L(\boldsymbol{a}) = \left\{ \left(\sqrt{\frac{1+k}{2}}, \boldsymbol{u}\sqrt{\frac{1-k}{2}} \right) \;\middle|\; g(\boldsymbol{u}) \leq k \leq 1, \|\boldsymbol{u}\| = 1, \boldsymbol{u} \cdot \boldsymbol{a} = 0 \right\}, \quad (13)$$

in which f and g are functions of unit vector, and $g(\boldsymbol{u}) = \cos(f(\boldsymbol{u}))$.

In the form of exponential mapping, the rotation limits can be represented as

$$L(\mathbf{a}) = \left\{ e^{\frac{\psi}{2}\boldsymbol{u}} \;\middle|\; 0 \leq \psi \leq f(\boldsymbol{u}), \|\boldsymbol{u}\| = 1, \boldsymbol{u} \cdot \boldsymbol{a} = 0 \right\}, \quad (14)$$

For all standard unit quaternions $\mathbf{q}_s = \left(\sqrt{\frac{1+k}{2}}, \boldsymbol{u}\sqrt{\frac{1-k}{2}} \right)$, vector $\boldsymbol{u}$ must lie in the same plane which is orthogonal to the reference vector. We can utilize this feature and further parameterize vector $\boldsymbol{u}$.

For the simplicity of calculation, without losing generality, we can arrange the local coordinate frame in such a way: the origin of the frame is located at the center of the joint, and the positive orientation of z-axis aligns with reference orientation $\boldsymbol{a}$. We choose the reference orientation in a such way that the only singularity orientation $-\boldsymbol{a}$ is out of reach in nature. Thus vector $\boldsymbol{u}$ must lie in x-y plane. Let ϕ be the angle of rotation from positive orientation of x-axis to vector $\boldsymbol{u}$, in a counter clockwise sense, then ϕ can uniquely represent the vector $\boldsymbol{u}$, as shown in Fig. 2. Function f and g thus become functions with only one variable ϕ. If function f and g are continuous with respect to ϕ, the rotation limits $L(\boldsymbol{a})$ is a closed region in S^3.

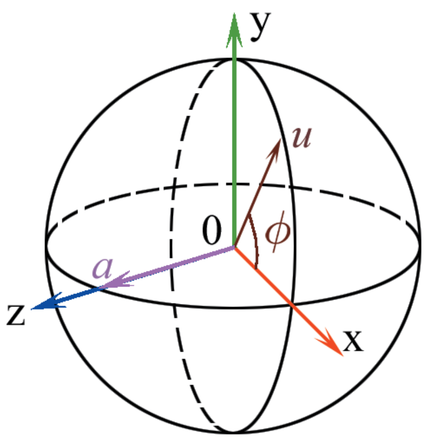


Fig. 2. Arrangement of local coordinate frame and the parameterization of vector $\boldsymbol{u}$.

The rotation boundary, or the formulation of function f and g, can be obtained using motion capture techniques, or using spherical polygons, which are used in our experiment. A spherical polygon can be set up by measuring a set of key orientations on rotation boundary $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, ..., $\boldsymbol{v}_n$, and connecting these points on a unit sphere with great arcs to form a polygon as illustrated in Fig. 3. Any orientation out of the spherical polygon is invalid.

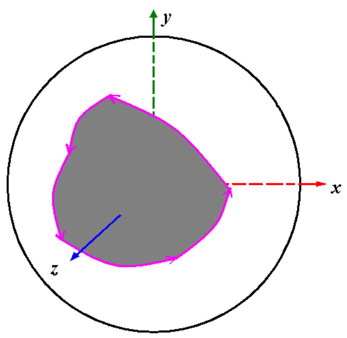


Fig. 3. A spherical polygon with five key orientations.

3.3 Constraint Checking

Given a unit quaternion $\mathbf{q}$, we need to check if this quaternion satisfies the joint constraint $L(\boldsymbol{a})$. If it is out of the rotation limit, we need to clamp it back to the boundary of the constraint. The procedure of constraint checking is described as below.

1. Standardize $\mathbf{q}$ to get the corresponding standard unit quaternion $\mathbf{q}_s = \left(\sqrt{\frac{1+k}{2}}, \boldsymbol{u}\sqrt{\frac{1-k}{2}}\right)$.
2. Get the angle ϕ from positive x-axis to vector $\boldsymbol{u}$, and calculate the corresponding function value $g(\phi)$.
3. Check if k satisfies $g(\phi) \leq k \leq 1$. If it does, stop checking; if it does not, go on to the next step.
4. Set $k = g(\phi)$. Recompute $\mathbf{q}_s = \left(\sqrt{\frac{1+g(\phi)}{2}}, \boldsymbol{u}\sqrt{\frac{1-g(\phi)}{2}}\right)$.

To speed up the calculation of $g(\boldsymbol{u})$, we can discretize the x-y plane with respect to ϕ. For each discretized angle ϕ, we pre-compute the corresponding limit $g(\phi)$ and construct a look-up table. Thus when checking the validity of a unit quaternion, after get the corresponding angle ϕ, we only need to check the look-up table to see if this unit quaternion is valid. An example of the look-up table is shown in Fig. 4.

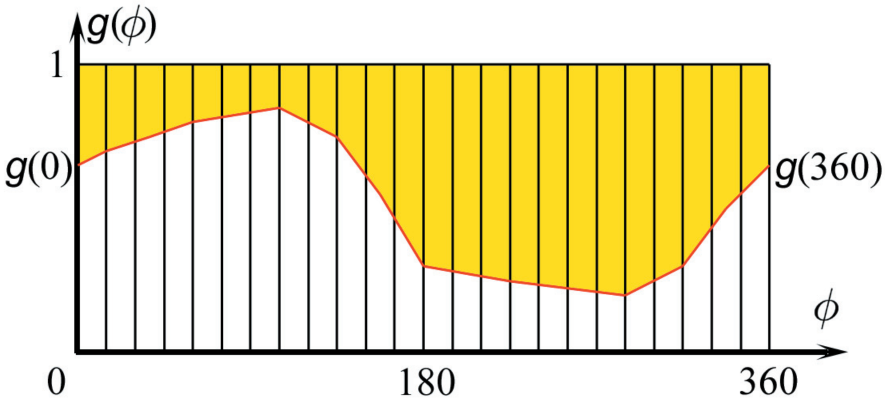


Fig. 4. A look-up table. The curve line from $g(0)$ to $g(360)$ is the value of function $g(\phi)$, and the area above the line is the valid area. Note that $g(0) = g(360)$.

4 Result

In our experiment, we constructed the constraint of a virtual human shoulder joint using a spherical polygon with eight key boundary orientations. After each time interval, a random unit quaternion was generated. We interpolated between unit quaternions using spherical linear interpolation, or slerp. For each in-between unit quaternions, we checked it's validity and clamped it to the limit boundary if it was invalid, as shown in Fig. 5.

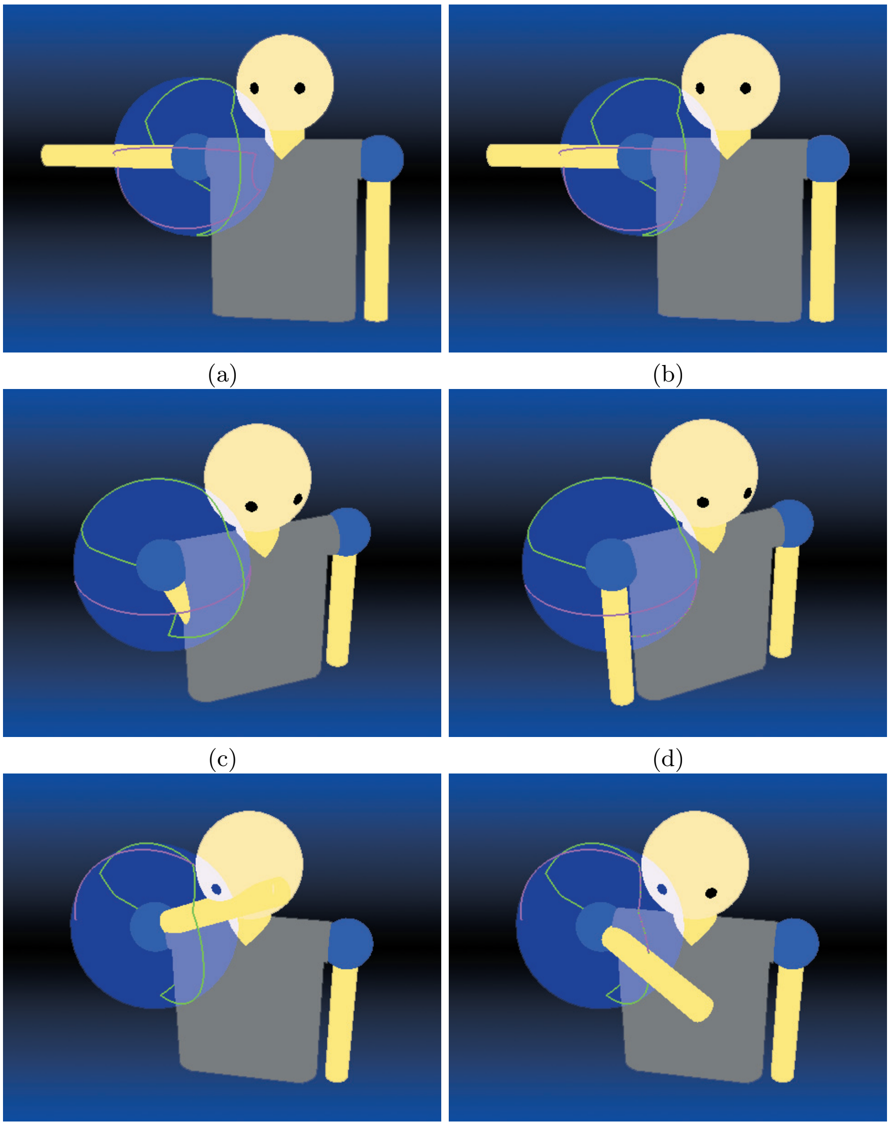
The results showed that parameterizing rotation with unit quaternion can produce smooth and natural motion. Our method of *rotation limit* is effective and efficient to constrain the motion inside the limit.

5 Conclusion

This paper describes a method to specify the constraint of a joint which is parameterized with a unit quaternion. By using this method, joint constraints can be easily specified and the validity of an orientation can be effectively checked. Our method can be used to construct complex and meaningful joint constrains accurately, and hence can be used in all kinds of articulated characters.

References

1. Bearlocher, P.: Inverse Kinematics Techniques for the Interactive Posture Control of Articulated Figures. Ph.D. Thesis, EPFL (2001)
2. Baerlocher, P., Boulic, R.: Parametrization and range of motion of the ball-and-socket joint. Proceedings of AVATARS'2000 Conference, Lausanne (2000) 180–190



(a) (b) (c) (d) (e) (f)

Fig. 5. The results of joint constraint. The green line is the boundary of joint constraint, and the purple line is the track of the joint rotation. Figure (a), (c), and (e) show the joint rotation without constraint checking. In (c), the arm intersects the body. In (e), the arm cuts the head. Figure (b), (d), and (f) show the joint rotation with constraint checking.

3. Grassia, F.S.: Practical parameterization of rotations using the exponential map. Journal of Graphics Tools, Vol. 3 (1998) 29–48
4. Hamilton, W. R.: Elements of Quaternions I & II, Chelsea Publishing Company (1969)
5. Huang, J., Prakash, E. C.: Sinus cone–a theta-phi algorithm for human arm animation. Proceedings for 2000 IEEE Comference on Information Visualisation (2000) 318–322
6. Kim, M. J., Kim, M. S., Shin, S. Y.: A general construction scheme for unit quaternion curves with simple high order derivatives. Computer Graphics (Proceedings of SIGGRAPH 95) (1995) 369–376
7. Korein, J. U.: A Geometric Investigation of Reach. ACM distinguished disserations series, MIT Press (1985)
8. Lee, J.: A hierarchical approach to motion analysis and synthesis for articulated figures. Ph.D. Thesis, Department of Computer Science, KAIST (2000)
9. Maurel,W.: 3D modeling of the human upper limb including the biomechanics of joints, muscles and soft tissues. Ph.D. Thesis, Laboratoire d'Infographie, EPFL (1999)
10. Maurel, W., Thalmann, D.: Human shoulder modeling including scapulo-thoracic constraint and joint sinus cones. Computer & Graphics, Vol. 24, No. 2 (2000) 203–218
11. Shoemake, K.: Animating rotation with quaternion curves. Computer Graphics (Proceedings of SIGGRAPH 85), Vol. 19 (1985) 245–254
12. Sinav, A.: Analysis and Modeling of the Virtual Human Interface for the MARG Body-Tracking System Using Quaternions, Math and CS Master's Thesis, Naval Postgraduate School (2002)