Subject: moment arms and moving muscle points
From: "Antonie J. van den Bogert" <a.vandenbogert@csuohio.edu>
Date: 11/15/2013, 2:27 PM
To: Ajay Seth <aseth@stanford.edu>

Hi Ajay,

I saw your posting on the Opensim forum regarding knee moment arms being different in different versions of the leg model.

I thought it would be good to have some private discussion so we do not generate more confusion. Please share this with Sherm, his input may be very useful here. And I apologize for the length of this and for saying things several times.

I read the DETC paper and I disagree on two points:

1. In my opinion, moving muscle points <u>should</u> be allowed
2. In my opinion, moment arm should always be defined with virtual work, i.e. dL/dtheta

The good news is that I also believe that dL/dtheta will give the same result as "effective torque" for moving points if you apply the latter method with care.

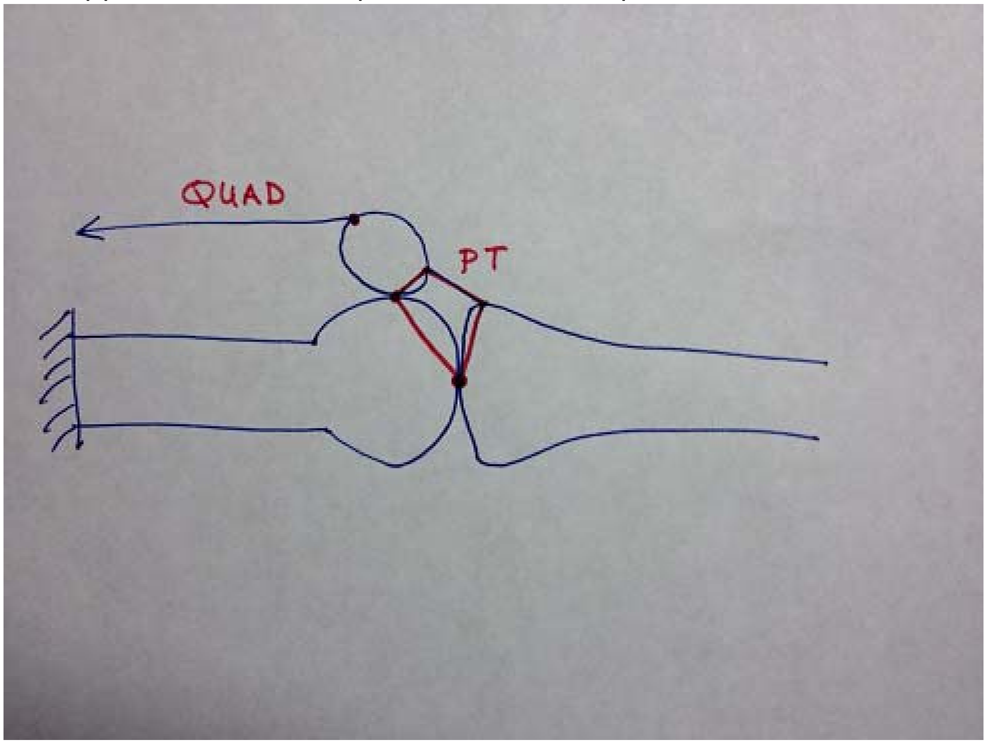
Scott's original leg model had a patella that translated and rotated as a function of knee angle. This effectively creates a moving muscle point on which the quadriceps inserts. Quadriceps moment arm was calculated with virtual work and I am 100% certain that this was correct (assuming that the model for patella kinematics was good enough).

I suspect that you are computing effective torque based on the force vector and the point of application. And that is wrong for moving muscle points. If my suspicion is incorrect, you can pretty much ignore the rest of this message, but I will continue just in case.

I use Autolev for dynamics, and it always reminds me that it can only include a force in the equations of motion if I know the velocity of its point of application, and how that velocity depends on the generalized velocities (joint angular velocities) in the system. Knowing its position on the bone is not sufficient. An illustrative example is, create a point on the tibia with a relative velocity that is exactly opposite to the velocity of a point that would be rigidly attached. Now you have a point that has zero velocity relative to ground, while still attached to the tibia. No matter how hard you pull on this point, the tibia won't move.

If a muscle pulls on a point that moves relative to a link, and that movement is a function of the generalized coordinates, you effectively have a gearbox between the muscle and the link. The "effective torque" that you calculate from cross product of position and force is the torque at the input of the gearbox. The torque that moves the tibia should be defined and measured at the output of the gearbox.

This is not just a mathematical issue. The gearbox exists physically. Look at my simple knee model sketched below. I indicated the instant centers of rotation of the patellofemoral and tibiofemoral joints (assuming no slip, but that can be generalized as long as each of those joints still has 1 DOF). This shows that the quadriceps force is transmitted through a (instantaneous) 4-bar linkage. This type of transmission is also common in hydraulic systems, see http://www.cmbol.com/upload/12040614254585_0.jpg for actuation of the scoop on an excavator. It gives extra range of motion in that application, at the expense of lower torque.



In my undergraduate kinematics class, I teach how to calculate the mechanical advantage, and this goes via virtual work. You calculate the velocity ratio between angular velocity of the scoop and velocity of the actuator length change. You can calculate this based on the kinematic equations of the 4-bar linkage. The mechanical advantage is the reciprocal of that ratio, i.e. dL/dtheta. You get exactly the same answer if you replace the linkage by a kinematic black box (possibly with gears inside) that just moves the insertion point as a function of rotation angle in exactly the same way. Which is a moving muscle point. This allows you to have a model without a patella and still have correct function of the quadriceps. Scott's original leg model effectively did this. And you should get the same moment arm in the gait23 models where the patella has been replaced by a moving muscle point that moves the same way as in Scott's model. If you treat the moving muscle point as the gearbox that it is, you can use "effective torque" and multiply by the gear ratio to get the output torque, and still get the correct answer, but why bother. Virtual work is so much easier and SIMM has done it since 25 years ago...

The gear ratio is simply the reciprocal of the velocity ratio. The velocity ratio can come from analysis of a kinematic model of the physical linkage, or from the derivative of the mathematical function you use to create the moving muscle point.

Fundamentally I think we should always use virtual work (leading to moment arm = dL/dtheta) because that is consistent with how joint moments are used as generalized forces in equations of motion (e.g. Kane's equations).

If you use a different method and you get an answer that is not consistent with virtual work, you need to explain where the energy is going. Force x length change must be equal to torque x angle change if there is no energy loss.

This is an important issue, because people report moment arms calculated with Opensim. Forward dynamic simulations may be using muscle moments computed with those same (incorrect) moment arms.

I hope we can clear this up and present a clear solution to the community. This confusion is not healthy and does not instill confidence in our methods for musculoskeletal modeling.

We can extend this discussion to models where several joint rotations are functions of one generalized coordinate. It is a similar problem with a similar solution.

Ton

--
Antonie J. (Ton) van den Bogert
Parker-Hannifin Endowed Chair in Human Motion and Control
Department of Mechanical Engineering
Cleveland State University
a.vandenbogert@csuohio.edu