Proyecto Fin de Carrera
Ingeniería de las Tecnologías Industriales

Modelado y estudio cinemático de un exoesqueleto pasivo de tren superior

Autor: Francisco Rus García de Leaniz
Tutor: Joaquín Ojeda Granja

Dpto. Ingeniería Mecánica y Fabricación
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2020

Proyecto Fin de Carrera
Ingeniería de las Tecnologías Industriales

Modelado y estudio cinemático de un exoesqueleto pasivo de tren superior

Autor:
Francisco Rus García de Leaniz

Tutor:
Joaquín Ojeda Granja
Profesor Contratado Doctor

Dpto. de Ingeniería Mecánica y Fabricación
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2020

Proyecto Fin de Carrera: Modelado y estudio cinemático de un exoesqueleto pasivo de tren superior

Autor: Francisco Rus García de Leaniz

Tutor: Joaquín Ojeda Granja

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2020

El Secretario del Tribunal

A mi familia

A mis amigos

A mis maestros

Agradecimientos

Este apartado va dedicado a cada una de las personas que, en mayor o menor medida, me han acompañado a lo largo de estos años de la carrera.

En primer lugar y cómo no, agradecer a mis padres el haberme dado la oportunidad de haber estudiado una carrera, sin ellos esto hubiera sido muy difícil.

A mi hermana, por haberme aconsejado cuando se lo he pedido, teniendo la suerte de que ella hubiese estudiado antes mi misma carrera.

A mi abuela, por estar siempre pendiente y preocuparse por mí.

A toda mi familia y amigos, que al fin y al cabo son lo mismo, por todos los momentos de risas y desconexión.

A todos los profesores desde el colegio hasta la universidad, porque cada uno de ellos me ha educado al igual que mis padres y han hecho que sea quien soy hoy.

Gracias a todos y cada uno de vosotros, a los que están y a los que se han ido, sin vosotros nunca habría llegado hasta aquí.

Francisco Rus García de Leaniz

Grado en la Ingeniería de las Tecnologías Industriales

Sevilla, 2020

Resumen

La investigación en el campo de los exoesqueletos está siendo ampliamente desarrollada en la actualidad con una gran perspectiva de futuro. En el Dpto. de Ingeniería Mecánica y Fabricación de la escuela se está investigando el uso de exoesqueletos pasivos de tren superior en sujetos que tengan que estar expuestos a una alta carga de trabajo pesado durante un periodo de tiempo prolongado.

En este trabajo se va a estudiar la cinemática de realizar un movimiento de flexión del codo con un exoesqueleto pasivo incorporado en el brazo de un sujeto. Se compararán los resultados obtenidos tanto con exoesqueleto y en la ausencia de este. Para la obtención de los resultados cinemáticos se usarán los programas Opensim® y Matlab®.

El objetivo será comparar los resultados obtenidos entre los diferentes programas para analizarlos y comprobar la validez de Opensim®. También se verá el efecto que produce el exoesqueleto incorporado al brazo del sujeto, en el movimiento de flexión en distintas configuraciones y se determinará cuál es la mejor.

En la memoria se incluirán imágenes del modelo del exoesqueleto visualizado en Opensim®, los códigos utilizados para su creación, los códigos creados en Matlab® y sus respectivas gráficas con los resultados cinemáticos a comparar.

Abstract

Research in the field of exoskeletons is being widely developed today, having a great prospect of the future. The school's Department of Mechanical Engineering and Manufacturing is investigating the use of passive upper-train exoskeletons in subjects who have to be exposed to a high heavy workload for an extended period of time.

In this work we will study the kinematics of performing a bending movement of the elbow with a passive exoskeleton incorporated in the arm of a subject. The results obtained both with and in the absence of exoskeleton will be compared. Opensim® and Matlab® programs will be used to obtain the kinematic results.

The objective will be to compare the results obtained between the different programs to analyze them and check the validity of Opensim®. You will also see the effect on bending motion of the exoskeleton incorporated into the arm in different configurations.

The memory will include images of the exoskeleton model displayed in Opensim®, the codes used for its creation, the codes created in Matlab® and their respective graphs with the kinematic results to compare.

Índice

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

1 INTRODUCCIÓN

La bioingeniería es uno de los campos de la ingeniería que más interés está levantando en la actualidad debido a sus avances tecnológicos en los diversos campos de aplicación que presenta. Los exoesqueletos artificiales son un claro ejemplo, algo que parecía cosa de películas de ciencia ficción hace unos años, se está llevando a cabo actualmente, con una gran perspectiva de desarrollo y mejora en un futuro próximo.

Un exoesqueleto artificial consiste en un dispositivo mecánico que, incorporado externamente al cuerpo humano, es capaz de aportar una mejor movilidad o algún tipo de capacidad física extra a la persona que lo porta, como puede ser una mayor fuerza, resistencia, etc. Esto hace de ellos un gran instrumento de ayuda para distintas aplicaciones como son la médica, la industrial y la militar.

Para contextualizar cuál es el estado actual de los exoesqueletos artificiales, se va a exponer inicialmente una breve narración de cómo ha sido su evolución a lo largo de la historia [1], [2], [3].

1.1 Historia de los exoesqueletos

1.1.1 Comienzos

Para introducir la historia del inicio, evolución y actualidad de los exoesqueletos es preciso comenzar hablando de sus orígenes más profundos, esto se refiere a los cimientos sobre los que se basa una gran parte de su funcionamiento, centrado en una de las ramas de la física, la mecánica. La mecánica estudia el movimiento y equilibrio de los cuerpos, así como de las fuerzas que los producen. No se podría estar hablando hoy día de nada parecido a un exoesqueleto sin la evolución y comprensión de esta disciplina por el ser humano. A falta de documentos u otras pruebas que indiquen lo contrario, la historia de la mecánica comienza en la antigua Grecia con Aristóteles (384-322 DC). No es hasta finales de la Edad Media y comienzos de la Edad Moderna, ya en el siglo XVI, cuando el campo de la dinámica, fundamental a la hora del estudio de sistemas vivos, comienza a desarrollarse gracias al trabajo de figuras como Kepler, Galileo, Descartes, Huygens y Newton. Un poco más adelante otras figuras como son Euler, LaGrange y LaPlace también realizarían una gran aportanción al mundo de la mecánica con sus estudios.

La biomecánica nace de usar los conceptos de la mecánica clásica que se han ido desarrollando a lo largo de los años y de aplicarlos en estudiar el movimiento de los seres vivos. Los pioneros en esta ciencia son los antes mencionados Aristóteles y Galileo con obras como "*De Motu Animalium*" y "*Discorsi e dimonstrazioni matematiche, intorno a due nuove scienze attentanti alla meccanica ed a muovementi locali*", respectivamente, y el polímata Leonardo da Vinci con sus observaciones sobre el movimiento de los pájaros.

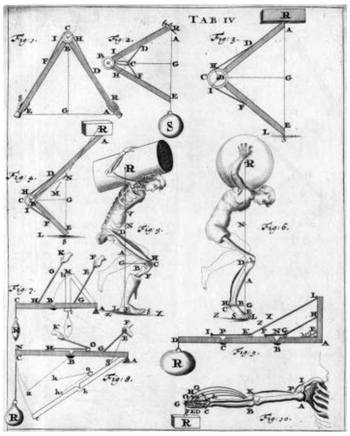


Figura 1-1. Biomecánica. [2]

La biomecánica, dentro de la bioingeniería, es una ciencia que tiene muchas aplicaciones, como la de poder llegar a explicar el proceso de remodelación ósea, el cómo los vasos sanguíneos saben el diámetro que deben de tener, etc. Los exoesqueletos, que es lo que concierne a este trabajo, son un ejemplo de aplicación del análisis estático y cineto-dinámico de la biomecánica, y por tanto gracias a esta ciencia derivada de la mecánica es por la que hoy día se puede hablar de ellos y de su desarollo.

1.1.2 Evolución

El primer dispositivo más parecido a un exoesqueleto del que se tiene constancia fue un aparato diseñado para ayudar a caminar. La versión final fue patentada en 1890 por su creador ruso Nicholas Yagn. Este aparato usaba energía almacenada en bolsas de gas comprimido para asistir en el movimiento, aunque era pasivo y requería de energía humana, es decir, su accionamiento era totalmente activado por elementos mecánicos que no usaban ningún tipo de fuente de alimentación.

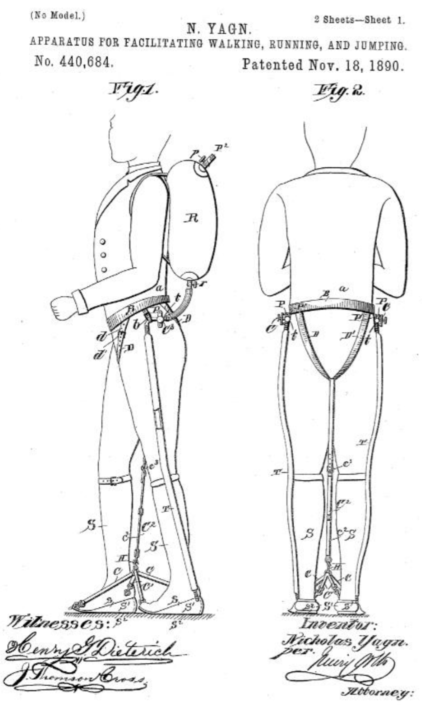


Figura 1-2. Patente de Nicholas Yagn (1890). [4]

En 1917, el inventor estadounidense Leslie C. Kelley desarrolló lo que él llamó un podómetro, que funcionaba con energía de vapor con ligamentos artificiales que actuaban en paralelo a los movimientos del usuario. Este sistema pudo complementar la energía humana con energía externa.

Ya en la década de 1960, comenzaron a aparecer las primeras y verdaderas "máquinas móviles". Estos exoesqueletos ya estaban compuestos por una tecnología más sofisticada y se buscaba que su aportación energética fuera totalmente activa que, al contrario de la pasiva, sí utilizaba actuadores que requerían de algún tipo de fuente de alimentación. El más famoso de esa época fue el traje robótico llamado Hardiman, diseñado y desarrollado por la empresa General Electric y las Fuerzas Armadas de Estados Unidos. El traje funcionaba con sistemas hidráulicos y eléctricos, consiguiendo amplificar la fuerza del usuario en un factor de 25, de tal forma que levantar 100 kilogramos se sentiría como si se levantara 4 kilogramos. También usaba una especie de retroalimentación para que el usuario sintiera las fuerzas y los objetos que manipulaba. La limitación de este traje en su peso, de 680 kilogramos, y su lenta velocidad, 0,76 metros por segundo, hicieron de él que fuera poco práctico, y el proyecto no tuvo éxito.

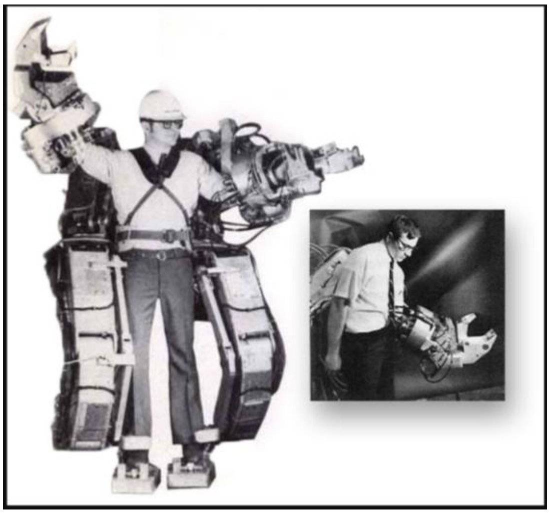


Figura 1-3. Traje del Hardiman. [5]

Aproximadamente al mismo tiempo que el Hardiman, se estaban desarrollando otros exoesqueletos activos en el Instituto Mihajlo Pupin, situado en Yugoslavia. El equipo dirigido por el profesor Miomir Vukobratović se centró primero en exoesqueletos de tren inferior, con la intención de ayudar en la rehabilitación de parapléjicos. Fue en 1972 cuando se probó un exoesqueleto de este tipo que se accionaba neumáticamente y se programaba electrónicamente.

En 1986 Monty Reed, un guardabosques del ejército estadounidense cuya espalda se rompió en un accidente de paracaídas, diseñó un exoesqueleto llamado Lifesuit que pudiera ser utilizado como herramienta de rehabilitación. En 2005 usó el duodécimo prototipo en la carrera a pie de Saint Patrick's Day Dash en Seattle, Washington.

Ya a finales del siglo XX y comienzos del XXI empiezan a aparecer empresas creando nuevos exoesqueletos que actualmente siguen desarrollándose para distintas aplicaciones (incluido el Lifesuit de Monty Reed). Cabe destacar el HAL (hybrid assistive limb) creado por la empresa Cyberdyne, el primer exoesqueleto robótico en recibir un certificado de seguridad global.

1.1.3 Actualidad

Los exoesqueletos es uno de los campos de aplicación de la biomecánica con más potencial de desarrollo en los próximos años; ya a día de hoy su uso en las industrias y su uso con fines médicos es una realidad. Empresas como Ekso Bionics, Sarcos, SuitX y Cyberdyne son unas de las pioneras en este campo.

Por ejemplo, ya en la fábrica de Almussafes (Valencia) de Ford, unos de los exoesqueletos pasivos más utilizados por los operarios son unos prototipos creados para la espalda y para los brazos.

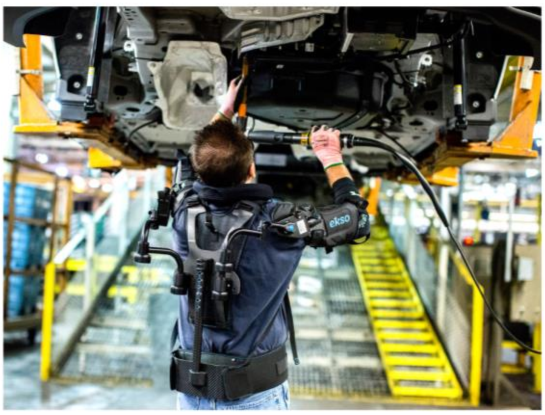


Figura 1-4. Trabajador de Ford con exoesqueleto pasivo EVO, fabricado por Ekso. [6]

Cyberdyne, la gran empresa japonesa creadora del conocido HAL, posee actualmente un considerable número de exoesqueletos activos, enfocados tanto a la asistencia médica para personas con algún tipo de déficit en su movilidad, como para prevenir algún tipo de lesión futura.

Figura 1-5. Exoesqueletos activos creados por Cyberdyne. [7]

En el mundo militar ya hay varios prototipos muy desarrollados aunque todavía no se comercializan. La Agencia de Proyectos de Investigación Avanzada de Defensa de los Estados Unidos (DARPA) inició en 2001 una inversión de 50 millones de dólares de cara al desarrollo de exoesqueletos en el marco del programa “Exoesqueletos para el aumento del rendimiento humano”.

Uno de los programas en activo es el prototipo XOS 2, desarrollado por Raytheon Company, un traje robótico más ligero y más eficiente que su versión anterior (XOS 1), que se espera que pueda trabajar sin ataduras para el año 2035, dado que le falta mejorar varios aspectos, resaltando el de su autonomía energética, ver [8].

Figura 1-6. Soldado realizando flexiones con el prototipo XOS 2. [8]

Los exoesqueletos se pueden clasificar actualmente en dos grandes grupos ya mencionados anteriormente en alguna ocasión: activos o pasivos; y dentro de ellos en exoesqueletos de tren superior o de tren inferior. Esta clasificación, referida al tipo de accionamiento y a la localización del exoesqueleto, será explicada con más detalle a continuación [9]. También se hablará de las distintas aplicaciones [10], [9] a las que se ha hecho referencia al inicio de este capítulo, médica, industrial y militar, siendo las más destacables dentro del amplio número de aplicaciones existentes en la actualidad.

1.2 Clasificación de los exoesqueletos

1.2.1 Según su accionamiento

1.2.1.1 Exoesqueletos activos

Los exoesqueletos activos disponen, junto a elementos mecánicos, de otros elementos de accionamiento que requieren de una fuente de alimentación. Estos exoesqueletos suelen usar motores eléctricos o algún tipo de sistema hidráulico para producir el movimiento, guiándose por sensores que normalmente reciben las tensiones musculares, informando así a estos actuadores. Otra forma alternativa es realizar el control del exoesqueleto desde un panel de mando.

1.2.1.2 Exoesqueletos pasivos

Estos exoesqueletos, a diferencia de los activos, apoyan al cuerpo portante solamente mediante componentes mecánicos tales como muelles, férulas y pesos; no disponiendo así de ningún tipo de accionamiento activo. Las cargas que se generan son recibidas por la estructura de apoyo y transferidas en energía.

1.2.2 Según su localización

1.2.2.1 Exoesqueletos de tren superior

Como su propio nombre indica son los exoesqueletos que se ocupan de ayudar a las partes situadas en lo que se conoce como tren superior del cuerpo humano, comprendido entre la cadera y la cabeza. Estos exoesqueletos suelen centrarse en beneficiar o proteger zonas como la espalda, hombros y brazos del usuario.

Figura 1-7. Exoesqueleto de tren superior, modelo shoulderX V3. [11]

1.2.2.2 Exoesqueletos de tren inferior

Estos exoesqueletos engloban a todos aquellos que se ocupan del tren inferior de las personas, comprendido entre la cadera y los pies. Según su campo de aplicación pueden servir para una gran variedad de actividades. La articulación de la rodilla tiene una gran importancia en este tipo de exoesqueletos, ya que cuanto mejor imiten el movimiento de una rodilla real, más eficaces serán en la función que desempeñen. Hay varios estudios dedicados a la compatibilidad de la articulación de la rodilla con exoesqueletos, por ejemplo, ver [12].

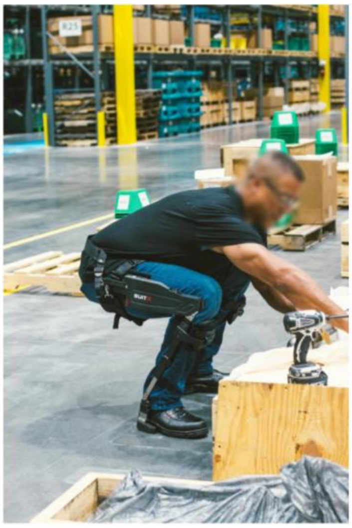


Figura 1-8. Exoesqueleto de tren inferior, modelo legX. [13]

1.3 Aplicaciones de los exoesqueletos

1.3.1 Médica

La introducción de esta tecnología en el ámbito sanitario ha sido muy positiva, ayudando a la rehabilitación de lesiones, ya que permite automatizar los procesos de terapia clínica. No es sólo capaz de ayudar a la rehabilitación de lesiones, también la incorporación de exoesqueletos en personas es capaz de prevenirlas, ya que consiguen una mejor repartición del peso corporal, pudiendo prevenir lesiones como son las lumbalgias.

No obstante, es en personas que han perdido algún tipo de movilidad o directamente la capacidad de andar donde se está produciendo un gran avance y uso de los exoesqueletos. Personas con lesiones medulares, atrofia muscular, o cualquier tipo de parálisis cerebral cuyas vidas estarían destinadas a una silla de ruedas, tienen la posibilidad de volver a andar. El HAL es uno de los exoesqueletos que más está ayudando en el tratamiento de estas lesiones actualmente.

1.3.2 Industrial

Son muchos los operarios sometidos a exigentes esfuerzos físicos durante un período de tiempo prolongado cuando elevan, transportan, y manipulan cargas, ya sea en la cadena de montaje, en el almacén, o en el suministro de mercancías. El uso de exoesqueletos tanto activos como pasivos ayudan a evitar lesiones incluso antes de que estas se produzcan, además de crear un efecto positivo tanto anímico como en eficiencia para el operario que lo lleva, ya que reduce su fatiga. Todo esto demora en un menor porcentaje de absentismo y un menor número de bajas laborales, consiguiendo ahorrar dinero para las empresas.

1.3.3 Militar

Al contrario que en la aplicación médica, que busca principalmente recuperar la capacidad física que un determinado paciente ha perdido, la miltar busca añadir o mejorar algún tipo de capacidad del usuario que porte el exoesqueleto. Fue la primera aplicación que despertó verdadero interés en el mundo de los exoesqueletos. Su intención es conseguir que los soldados se conviertan en una especie de superhombres, capaces de correr largas distancias sin apenas fatiga, saltar distancias y coger grandes pesos que no estén al alcance de un humano normal y corriente.

Otros campos de aplicación fuera de estos tres principales pueden ser los de incorporar exoesqueletos a otros trabajadores como son los bomberos, los del área de cuidado de personas, la construcción y en general otras profesiones que supongan esfuerzos físicos.

1.4 Objetivos

En general, los exoesqueletos cada vez están introduciéndose más en sus distintos campos de aplicación, proporcionando grandes ventajas, como son la reducción del riesgo en la salud física de los trabajadores, rehabilitaciones más rápidas en pacientes, o mayores capacidades físicas en sus usuarios. Pero todavía la tecnología de los exoesqueletos es un campo en desarrollo, y presenta desventajas en algunos casos como puede ser el peso (aunque la introducción de materiales como la fibra de carbono está solucionando este problema), o la reducción de la velocidad y maniobrabilidad del usuario en determinados movimientos.

En lo referente a este trabajo, y centrándose en la última de las desventajas anteriormente nombradas, la maniobrabilidad, se ha realizado un estudio cinemático, específicamente, del movimiento de flexión del brazo derecho de un sujeto que porta un exoesqueleto pasivo de tren superior. El exoesqueleto está enfocado en una aplicación industrial, cuya función es proteger a los operarios que a lo largo de su jornada laboral ejercen en repetidas ocasiones algún tipo de movimiento o levantamiento pesado con sus brazos. Este esfuerzo constante a largo plazo puede conllevar algún tipo de lesión o secuela física en estos operarios.

El empresario, incluidas las administraciones públicas, tienen el deber de proteger a sus trabajadores, garantizando su salud y seguridad en todos los aspectos relacionados con su trabajo, mediante la integración de la actividad preventiva en la empresa y la adopción de cuantas medidas sean necesarias, de acuerdo con la ley de prevención de riesgos laborales [14]. Con la incorporación, en los operarios, del exoesqueleto que se ha estudiado en este trabajo, se pretende solucionar parte de este problema.

Este estudio tiene como fin, a partir de un modelo biomecánico del sujeto y de los datos experimentales recogidos de su posición durante el movimiento de flexión del brazo derecho, obtener las coordenadas generalizadas que lo gobiernan, verificar su validez y buscar la configuración más adecuada del exoesqueleto para este movimiento del sujeto en cuestión, permitiéndole así una buena maniobrabilidad.

Para la obtención de resultados se ha trabajado con dos programas. El programa con el que se ha trabajado principalmente es Opensim®, el cual consiste en un software libre poseedor de una serie de herramientas que facilitan la resolución de problemas cinemáticos, entre otros. El otro programa utilizado ha sido Matlab®, que ha servido como herramienta para validar en un principio los resultados obtenidos por Opensim®.

2 FORMULACIÓN GENERAL

Este capítulo se va a centrar en exponer las ideas generales empleadas para la resolución del problema principal que se plantea en este estudio. El problema que se plantea es un problema cinemático inverso, en el cual se dispone de unos datos de entrada, que son el modelo biomecánico y las posiciones recogidas experimentalmente de diferentes puntos del sujeto mientras realiza el movimiento, y del que se quiere obtener, como datos de salida, las coordenadas generalizadas que gobiernan ese movimiento y consiguen que el sujeto adopte esas posiciones a lo largo del tiempo.

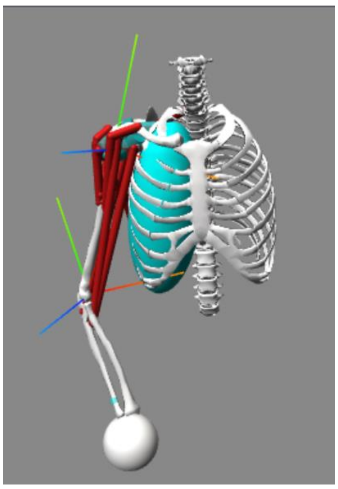
Por tanto, se explicará con más detalle a continuación, en qué consisten estos datos de entrada, cómo se recogen, y qué herramientas teóricas y matemáticas son utilizadas para obtener los resultados cinemáticos, o datos de salida, que más tarde se mostrarán y se analizarán.

2.1 Implementación del modelo biomecánico

La correcta creación del modelo del sujeto es fundamental para una buena obtención de resultados. Debe definirse una geometría y un escalado que refleje lo mejor posible al sujeto real. Es importante definir, aparte de lo antes mencionado, unos sistemas de coordenadas locales para cada sólido a los que atenerse con rigurosidad, y unas coordenadas generalizadas con un sentido definido. Todo esto debe hacerse siguiendo los conocimientos teóricos implantados por la cinemática de sólidos rígidos tridimensional.

Para este estudio, se ha partido de un modelo biomecánico en el cual se han añadido diferentes sólidos (o segmentos) correspondientes a cada uno de los huesos que forman el cuerpo real del sujeto. Las articulaciones han sido diseñadas acorde con unos grados de libertad permitidos; estos grados de libertad corresponden con las coordenadas generalizadas que permiten el movimiento relativo entre los diferentes sólidos del modelo.

El estudio se ha centrado en las tres coordenadas generalizadas relativas de elevación frontal y lateral del hombro, y la de flexión del codo. Los sistemas de coordenadas locales utilizados de mayor interés fueron los definidos para el húmero, con origen en su parte proximal (zona pegada al hombro), y para el cúbito, con origen de igual manera en su parte proximal (zona del codo). A continuación, se puede ver una imagen del modelo biomecánico del que se ha partido en este estudio, con los sistemas de coordenadas locales definidos, proporcionado por el Dpto. de Ingeniería Mecánica y Fabricación de la Escuela Técnica Superior de Ingeniería (ETSI) de Sevilla:



Eje rojo = Eje X

Eje verde = Eje Y

Eje azul = Eje Z

Figura 2-1. Modelo biomecánico. [15]

2.2 Captura de movimiento

Para obtener lo que servirá como segundo dato de entrada en el problema cinemático inverso, hay que realizar una recopilación de datos experimentales de la trayectoria que las partes del cuerpo del sujeto experimentan al realizar el movimiento de flexión del codo del brazo derecho. Estas trayectorias serán comparadas con las del modelo creado en los dos programas con los que se trabaja. Los datos experimentales se obtienen mediante un sistema de captura de movimiento basado en cámaras infrarrojas.

Este sistema de captura de movimiento requiere del uso de marcadores reflectantes. Estos marcadores consiguen recopilar la posición y orientación en cada instante de tiempo de los diferentes sólidos de interés en los que se puede dividir el sujeto a estudiar, como son para este estudio el tórax, escápula, brazo, antebrazo, etc. Para obtener estos datos experimentales correctamente es necesario colocar al menos tres marcadores no colineales en cada segmento de sólido rígido del que se quiere saber su posición y orientación, también es conveniente colocarlos en zonas donde el movimiento relativo entre la piel y el músculo del sujeto sea mínimo. Esto minimizará el error.

Figura 2-2. Captación de movimiento por cámaras infrarrojas y marcadores reflectantes. [16]

En este estudio se han usado los datos recogidos por un total de trece marcadores repartidos por toda la zona del tren superior del sujeto, centrándose en mayor medida en la zona del brazo derecho, la cual es la parte de interés a estudiar. En la siguiente figura se puede observar una imagen con el nombre de cada marcador y en la zona del cuerpo en la que se situaron:

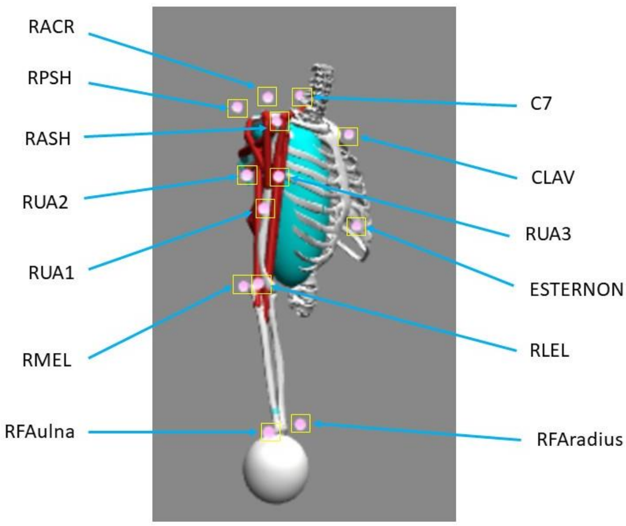


Figura 2-3. Marcadores en el modelo biomecánico. [15]

Los datos recopilados de la trayectoria en 3D fueron almacenados en un archivo ".trc", donde se recogen las coordenadas globales X, Y, Z para cada marcador y en cada instante de tiempo, que servirán como dato de entrada para resolver el problema cinemático. La duración del movimiento de flexión y, por tanto, la de recopilación de datos es de 7,81 segundos, con un paso de 0,01 segundos. Esto quiere decir que cada 0,01 segundos se recogen unas nuevas coordenadas para cada marcador.

2.3 Problema cinemático inverso

Para la resolución del problema que se plantea, en el cual se quiere obtener las coordenadas generalizadas que modelan el movimiento del sujeto, y más en concreto las de elevación frontal y lateral del hombro, y la de flexión del codo, que han sido las de más interés a la hora de comparar los resultados, es necesario aplicar cinemática inversa.

Se conoce las posiciones que el sujeto debe adoptar en cada instante de tiempo gracias a los marcadores, por lo que junto al modelo creado, se puede llegar a obtener las coordenadas generalizadas que consiguen que el sujeto alcance esas posiciones.

2.3.1 Mínimos cuadrados ponderados

Para minimizar el error entre el movimiento que realiza el sujeto real y el del modelo, la distancia entre la posición experimental de los diferentes marcadores reales, recogida mediante el sistema de captura de movimiento, y la de los marcadores del modelo, que van a depender de las coordenadas generalizadas, debe ser la menor posible. Se busca por tanto las coordenadas generalizadas que minimicen al máximo este error.

Para esto se usa la ecuación de los mínimos cuadrados ponderados que se muestra a continuación:

$$\min_{q} \sum_{i \in marcadores} w_i \| x_i^{exp} - x_i(q) \|^2 \quad (2.1)$$

Donde w_i se refiere a los pesos que se le puede asignar a cada marcador. Sirven para dar más importancia a unos marcadores que a otros a la hora de minimizar su error según el interés. En el caso de este estudio todos tendrán el mismo peso, uno. El parámetro x_i^{exp} se refiere a la posición experimental de los marcadores reales, y $x_i(q)$, a la posición de los marcadores del modelo que dependen del vector de coordenadas generalizadas q, el cual es incógnita en un principio.

2.3.2 Datos de salida

Los datos de salida, que se obtienen de aplicar todo lo anteriormente dicho, son las coordenadas generalizadas contenidas en el vector q. Se obtendrán un conjunto de coordenadas generalizadas que minimicen el error para cada instante de tiempo. Es importante no olvidar esto y recordar que en cada instante se recoge la posición de cada marcador y, por lo tanto, habrá un vector de coordenadas generalizadas diferente.

Obtener las coordenadas generalizadas de un modelo es de gran interés, ya que una vez obtenidas estas, se es capaz de calcular cualquier otro dato cinemático que se quiera. Esto servirá más adelante para calcular la posición y la velocidad del centro de masas de cada sólido (o segmento), que servirá también para comparar resultados.

3 MODELO SISTEMA BRAZO-EXOESQUELETO

Dada por concluida la explicación de la formulación general para la resolución del problema que ocupa este estudio, el siguiente paso es el de entrar en detalle en cómo se ha trabajado con cada programa, y cómo se ha hecho el modelado, incluyendo ya, en el caso de Opensim®, al propio exoesqueleto en el modelo.

3.1 Modelo en Opensim®

Para este estudio en Opensim®, se ha partido del modelo biomecánico, que pasará a llamarse a partir de ahora modelo original del sujeto. Este modelo, guardado en un archivo ".osim", consiste en una representación visual de una serie de códigos creados con el objetivo de reflejar con la mayor aproximación posible, el tren superior del sujeto humano del cual se tomaron los datos experimentales, es decir, del sujeto en el que se colocaron los marcadores para la captura de movimiento. Este modelo ha servido como guía para la creación del modelo de Matlab®, que se verá en la siguiente sección.

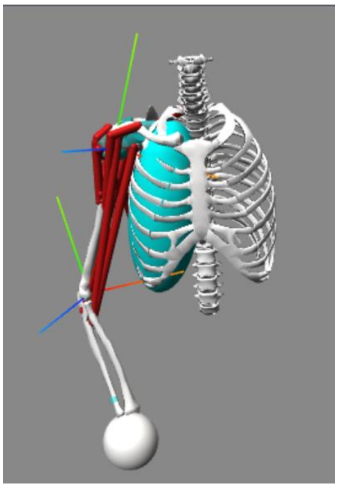


Figura 3-1. Modelo original del sujeto. [15]

La siguiente tabla va a recoger, para conocer más en detalle al modelo biomecánico, a todos los sólidos y uniones existentes entre ellos, y el nombre de todas las coordenadas generalizadas que permiten, según el caso, el movimiento relativo en dichas uniones entre los diferentes sólidos que forman el modelo original del sujeto:

Tabla 3-1. Sólidos, uniones y coordenadas generalizadas del modelo original

Sólidos	**Uniones**	**Coordenadas generalizadas**
Tórax	Suelo-Tórax	ground_thorax_rot_x, ground_thorax_rot_y, ground_thorax_rot_z, ground_thorax_tx, ground_thorax_ty, ground_thorax_tz
Clavícula	Esternoclavicular	clav_prot, clav_elev
Escápula	Escapulotorácica	scapula_abduction, scapula_elevation, scapula_upward_rot, scapula_winging
Húmero	Glenohumeral	plane_elv, shoulder_elv, axial_rot

Cúbito	Codo	elbow_flexion
Radio	Radio-Cúbito	pro_sup
Mano	Radio-Mano	---
Masa puntual	Radio-Masa puntual	---

Para el estudio cinemático, al modelo del sujeto se le ha añadido lo que sería una simplificación del exoesqueleto pasivo de tren superior a usar en la realidad, ya que sería más complejo y estaría formado por un mayor número de barras y muelles. El de este estudio, sin embargo, está compuesto por dos barras y un único muelle. Al añadir el exoesqueleto al sujeto, se pretende conocer qué configuración es la más apropiada para que el sujeto consiga realizar el movimiento de flexión del brazo derecho sin ningún impedimento, atendiendo a las longitudes de las barras y a los grados de libertad de los pares cinemáticos que las unen con el sujeto. Esto se comprobará realizando varias simulaciones con diferentes configuraciones.

A continuación se va a describir cómo ha sido el desarrollo para la adición del exoesqueleto al modelo del sujeto, en el que se acabará mostrando una imagen del modelo final (Figura 3-7). Se expondrán algunos de los fragmentos del código del archivo del modelo ".osim" más relevantes. La modificación del archivo, programado en el lenguaje de programación XML, se ha realizado con el editor de textos Notepad++, en el que se añadirán las dos barras, el muelle, y las uniones que componen el exoesqueleto. Dada la longitud del archivo, el total de su contenido estará recogido en el Anexo A.

3.1.1 Adición de sólidos

En primer lugar hay que añadir las dos barras. Para esto se comienza abriendo el archivo ".osim" con Notepad++, y en la sección <BodySet name="bodyset">, crear una nueva línea donde escribir. Un problema que presenta el programa Opensim® es que sólo funciona con mecanismos de cadena abierta. Al añadir el exoesqueleto y uniéndolo al brazo completo del sujeto, el modelo se estaría convirtiendo en un mecanismo de cadena cerrada. Para solucionar esto hay que crear un tipo de restricción llamada "Constraint". Esta restricción conseguirá que las dos partes del antebrazo del exoesqueleto, que habrá que crear, no presenten ningún movimiento relativo entre ellas. Esto significa que finalmente para crear las dos barras, se añadirán tres sólidos, uno correspondiente a la barra completa que simula el brazo del exoesqueleto, y los otros dos correspondientes a cada una de las dos partes en las que hay que dividir la segunda barra, que simula el antebrazo del exoesqueleto. A continuación, se puede ver el ejemplo del código creado para el sólido que representa el brazo del exoesqueleto a través de Notepad ++. Los otros dos sólidos que formarán el antebrazo se añaden de forma análoga (ver Anexo A):

```
<Body name="brazo_exoesqueleto">
    <!--The geometry used to display the axes of this Frame.-->
    <FrameGeometry name="frame_geometry">
        <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
        <socket_frame>..</socket_frame>
        <!--Scale factors in X, Y, Z directions respectively.-->
        <scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
    </FrameGeometry>
    <!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the fran
    <attached_geometry>
        <Mesh name="brazo_exoesqueleto_geom_1">
            <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
            <socket_frame>..</socket_frame>
            <!--Scale factors in X, Y, Z directions respectively.-->
            <scale_factors>0.3 3.25 0.15</scale_factors>
            <!--Default appearance attributes for this Geometry-->
            <Appearance>
                <!--Flag indicating whether the associated Geometry is visible or hidden.-->
                <visible>true</visible>
                <!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
                <opacity>1</opacity>
                <!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
                <color>1 1 0</color>
            </Appearance>
            <!--Name of geometry file.-->
            <mesh_file>block.vtp</mesh_file>
        </Mesh>
    </attached_geometry>
    <!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used t
    <WrapObjectSet name="wrapobjectset">
        <objects />
        <groups />
    </WrapObjectSet>
    <!--The mass of the body (kg)-->
    <mass>0.29999999999999999</mass>
    <!--The location (Vec3) of the mass center in the body frame.-->
    <mass_center>0 0 0</mass_center>
    <!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the ma
    <inertia>0.01 0.01 0.01 0 0 0</inertia>
</Body>
```

Figura 3-2. Código para la adición del brazo del exoesqueleto en Opensim®

En el código visto en la Figura 3-2 se puede destacar principalmente tres aspectos. El primero es la longitud del sólido, la cual no se ingresa por números, si no por factores de escala en las direcciones locales X, Y, Z. Se puede ver que la dirección Y (la longitudinal de la barra), es la multiplicada por un mayor factor, 3,25, ya que es la dirección en la que la barra es más larga. Otro aspecto importante, y relacionado en su medida con lo dicho anteriormente, es el archivo de geometría. El archivo de geometría posee el nombre de “block.vtp”. Se puede decir que tiene relación con lo anteriormente dicho puesto que el sólido no se crea como una barra originalmente, si no que es un cubo al que se le da unas dimensiones, convirtiéndolo así en una barra. El último aspecto que destacar es la masa que se le da al sólido, la cual no tiene ninguna importancia, ya que la masa del exoesqueleto no influirá nada en la cinemática, que es lo que interesa en este estudio.

3.1.2 Adición de pares cinemáticos

Una vez definidos los sólidos, con su geometría, longitud, y masa, hay que decirles dónde se tienen que situar, es aquí donde la definición de las uniones juega un papel vital. Estas uniones no serán más que pares cinemáticos a los que se les definirá una serie de grados de libertad.

El exoesqueleto, en lo que la cinemática interesa, se pretende que se mueva lo más parecido posible al brazo real del sujeto que lo porta, por lo tanto, es importante darle una longitud parecida a las barras que lo componen, como se hizo a la hora de definir los sólidos, y una libertad de movimiento que corresponda con la de un brazo humano. Esto último es lo que se realiza en la sección <JointSet name="jointset"> del archivo del modelo abierto en Notepad ++. En la siguiente figura, se puede ver el código utilizado para crear la unión entre las dos barras del exoesqueleto, que simula el codo, y representa un par de rotación. La unión del brazo del exoesqueleto con el hombro, y la del antebrazo con la muñeca se realiza análogamente (ver Anexo A):

```xml
<CustomJoint name="par_rot_brazoantebrazo">
    <!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: Th
    <socket_parent_frame>brazo_exoesqueleto_offset</socket_parent_frame>
    <!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The
    <socket_child_frame>antebrazo_exoesqueleto1_offset</socket_child_frame>
    <!--List containing the generalized coordinates (q's) that parameterize this joint.-->
    <coordinates>
        <Coordinate name="elbow2_flexion">
            <!--The value of this coordinate before any value has been set. Rotational coordinate value is
            <default_value>0.5939961667846837</default_value>
            <!--The speed value of this coordinate before any value has been set. Rotational coordinate va
            <default_speed_value>0</default_speed_value>
            <!--The minimum and maximum values that the coordinate can range between. Rotational coordinate
            <range>0 2.2689280300000001</range>
            <!--Flag indicating whether or not the values of the coordinates should be limited to the range
            <clamped>false</clamped>
            <!--Flag indicating whether or not the values of the coordinates should be constrained to the c
            <locked>false</locked>
            <!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim
            <prescribed_function />
            <!--Flag indicating whether or not the values of the coordinates should be prescribed according
            <prescribed>false</prescribed>
        </Coordinate>
    </coordinates>
    <!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's par
    <frames>
        <PhysicalOffsetFrame name="brazo_exoesqueleto_offset">
            <!--The geometry used to display the axes of this Frame.-->
            <FrameGeometry name="frame_geometry">
                <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
                <socket_frame>..</socket_frame>
                <!--Scale factors in X, Y, Z directions respectively.-->
                <scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
            </FrameGeometry>
            <!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent f

            <socket_parent>/bodyset/brazo_exoesqueleto</socket_parent>
            <!--Translational offset (in meters) of this frame's origin from the parent frame's origin, exp
            <translation>0 -0.16 0</translation>
            <!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fix
            <orientation>0 0 0</orientation>
        </PhysicalOffsetFrame>
        <PhysicalOffsetFrame name="antebrazo_exoesqueleto1_offset">
            <!--The geometry used to display the axes of this Frame.-->
            <FrameGeometry name="frame_geometry">
                <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
                <socket_frame>..</socket_frame>
                <!--Scale factors in X, Y, Z directions respectively.-->
                <scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
            </FrameGeometry>
            <!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent f
            <socket_parent>/bodyset/antebrazo_exoesqueleto1</socket_parent>
            <!--Translational offset (in meters) of this frame's origin from the parent frame's origin, exp
            <translation>0 0.059999999999999998 0</translation>
            <!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fix
            <orientation>0 0 0</orientation>
        </PhysicalOffsetFrame>
    </frames>
    <!--Defines how the child body moves with respect to the parent as a function of the generalized coord
    <SpatialTransform>
        <!--3 Axes for rotations are listed first.-->
        <TransformAxis name="rotation1">
            <!--Names of the coordinates that serve as the independent variables         of the transform f
            <coordinates></coordinates>
            <!--Rotation or translation axis for the transform.-->
            <axis>1 0 0</axis>
            <!--Transform function of the generalized coordinates used to        represent the amount of di
            <Constant name="function">
                <value>0</value>
            </Constant>
        </TransformAxis>
```

```
        <TransformAxis name="rotation2">
            <!--Names of the coordinates that serve as the independent variables
            <coordinates></coordinates>
            <!--Rotation or translation axis for the transform.-->
            <axis>0 1 0</axis>
            <!--Transform function of the generalized coordinates used to        repres
            <Constant name="function">
                <value>0</value>
            </Constant>
        </TransformAxis>
        <TransformAxis name="rotation3">
            <!--Names of the coordinates that serve as the independent variables
            <coordinates>elbow2_flexion</coordinates>
            <!--Rotation or translation axis for the transform.-->
            <axis>0.049400010000000001 0.036600010000000002 0.99810825000000003</axis>
            <!--Transform function of the generalized coordinates used to        repres
            <LinearFunction name="function">
                <coefficients> 1 0</coefficients>
            </LinearFunction>
        </TransformAxis>
        <!--3 Axes for translations are listed next.-->
        <TransformAxis name="translation1">
            <!--Names of the coordinates that serve as the independent variables
            <coordinates></coordinates>
            <!--Rotation or translation axis for the transform.-->
            <axis>1 0 0</axis>
            <!--Transform function of the generalized coordinates used to        repres
            <MultiplierFunction name="function">
                <function>
                    <Constant>
                        <value>0</value>
                    </Constant>
                </function>
                <scale>1.0000002849560037</scale>
            </MultiplierFunction>

        </TransformAxis>
        <TransformAxis name="translation2">
            <!--Names of the coordinates that serve as the independent variables        of
            <coordinates></coordinates>
            <!--Rotation or translation axis for the transform.-->
            <axis>0 1 0</axis>
            <!--Transform function of the generalized coordinates used to        represent
            <MultiplierFunction name="function">
                <function>
                    <Constant>
                        <value>0</value>
                    </Constant>
                </function>
                <scale>1.0000053431078386</scale>
            </MultiplierFunction>
        </TransformAxis>
        <TransformAxis name="translation3">
            <!--Names of the coordinates that serve as the independent variables        of
            <coordinates></coordinates>
            <!--Rotation or translation axis for the transform.-->
            <axis>0 0 1</axis>
            <!--Transform function of the generalized coordinates used to        represent
            <MultiplierFunction name="function">
                <function>
                    <Constant>
                        <value>0</value>
                    </Constant>
                </function>
                <scale>0.99999410549189738</scale>
            </MultiplierFunction>
        </TransformAxis>
    </SpatialTransform>
</CustomJoint>
```

Figura 3-3. Código para la adición del par de rotación del exoesqueleto en Opensim®

El primer paso para definir una unión es indicar qué tipo de unión es. En este caso es una unión costumizada (CustomJoint), que se puede diseñar como se quiera. Lo próximo es indicar qué sólido será el padre (parent), y que sólido será el hijo (child). Hay que tener en cuenta que Opensim® no admite que un mismo sólido tenga dos padres. El siguiente paso es definir las coordenadas o grados de libertad que se le va a querer dar a esa unión, darle un nombre, e imponer una serie de valores preestablecidos como puede ser su valor inicial, su rango de movimiento, etc. En el caso del par de rotación para permitir la flexión relativa entre las dos barras, se le ha puesto a esa coordenada generalizada el nombre de elbow2_flexion (Figura 3-3). A continuación, se define a qué distancia del origen del sistema de referencia local de cada sólido (padre e hijo) se encuentra el origen del par cinemático, y su orientación. Para esto último es importante tener claro que, para cada sólido, se define la distancia y la orientación en su sistema de coordenadas locales, expresando la distancia en metros y la orientación en radianes. Por último, queda únicamente indicar que tipo de movimiento se quiere que realice la coordenada creada, de rotación o de traslación, y alrededor de qué eje o en qué dirección, respectivamente. También se le indica si se quiere que el movimiento siga una función lineal, cuadrática, etc.

El tipo de restricción Constraint mencionado al principio de la subsección 3.1.1, aunque no se considera una unión como tal, cumple una función similar. Puesto que ha sido una parte importante del código a la hora de añadir el exoesqueleto al modelo, se va a mostrar a continuación cómo se hizo. Esta parte del código se escribe en la sección <ConstraintSet name="constraintset">, a través de Notepad ++:

```
<WeldConstraint name="antebrazo_exoesqueleto">
    <!--List of components that this component owns and serializes.-->
    <components />
    <!--Flag indicating whether the constraint is enforced or not.Enforced means that the constraint is act
    <isEnforced>true</isEnforced>
    <!--Path to a Component that satisfies the Socket 'frame1' of type F (description: The first frame part
    <socket_frame1>antebrazo_exoesqueleto1_offset</socket_frame1>
    <!--Path to a Component that satisfies the Socket 'frame2' of type F (description: The second frame par
    <socket_frame2>antebrazo_exoesqueleto2_offset</socket_frame2>
    <!--Frames created/added to satisfy this component's connections.-->
    <frames>
        <PhysicalOffsetFrame name="antebrazo_exoesqueleto1_offset">
            <!--The geometry used to display the axes of this Frame.-->
            <FrameGeometry name="frame_geometry">
                <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
                <socket_frame>..</socket_frame>
                <!--Scale factors in X, Y, Z directions respectively.-->
                <scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
            </FrameGeometry>
            <!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent f
            <socket_parent>/bodyset/antebrazo_exoesqueleto1</socket_parent>
            <!--Translational offset (in meters) of this frame's origin from the parent frame's origin, exp
            <translation>0 -0.059999999999999998 0</translation>
            <!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fix
            <orientation>0 0 0</orientation>
        </PhysicalOffsetFrame>
        <PhysicalOffsetFrame name="antebrazo_exoesqueleto2_offset">
            <!--The geometry used to display the axes of this Frame.-->
            <FrameGeometry name="frame_geometry">
                <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
                <socket_frame>..</socket_frame>
                <!--Scale factors in X, Y, Z directions respectively.-->
                <scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
            </FrameGeometry>
            <!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent f
```

```
                <socket_parent>/bodyset/antebrazo_exoesqueleto1</socket_parent>
                <!--Translational offset (in meters) of this frame's origin from the parent frame's origin, exp
                <translation>0 -0.059999999999999998 0</translation>
                <!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fix
                <orientation>0 0 0</orientation>
            </PhysicalOffsetFrame>
            <PhysicalOffsetFrame name="antebrazo_exoesqueleto2_offset">
                <!--The geometry used to display the axes of this Frame.-->
                <FrameGeometry name="frame_geometry">
                    <!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
                    <socket_frame>..</socket_frame>
                    <!--Scale factors in X, Y, Z directions respectively.-->
                    <scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
                </FrameGeometry>
                <!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent f
                <socket_parent>/bodyset/antebrazo_exoesqueleto2</socket_parent>
                <!--Translational offset (in meters) of this frame's origin from the parent frame's origin, exp
                <translation>0 0.059999999999999998 0</translation>
                <!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fix
                <orientation>0 0 0</orientation>
            </PhysicalOffsetFrame>
        </frames>
</WeldConstraint>
```

Figura 3-4. Código para la adición de la restricción "Constraint" en Opensim®

En este caso el tipo de restricción es imitando a una soldadura (WeldConstraint). Esto evita el movimiento relativo entre los dos sólidos indicados, el antebrazo_exoesqueleto1 y antebrazo_exoesqueleto2. De forma parecida a los pares cinemáticos hay que indicar que dos sólidos intervienen, sin haber en este caso un padre y un hijo. Se le da de igual manera unas coordenadas locales para cada sólido que indiquen la posición donde se quiere colocar exactamente esa restricción, y una orientación.

3.1.3 Adición de accionamiento mecánico

El último elemento creado para dar por finalizado el exoesqueleto es el muelle, que realizará el accionamiento pasivo, ayudando a flexionar el brazo del sujeto para que este realice una menor fuerza. El código utilizado se recoge a continuación, escrito en la sección <ForceSet name="forceset">:

```
<PathSpring name="muelle">
    <!--Flag indicating whether the force is applied or not. If true the forceis applied to the
    <appliesForce>true</appliesForce>
    <!--The GeometryPath defines the set of points and wrapping surfaceinteractions that form th
    <GeometryPath name="geometrypath">
        <!--The set of points defining the path-->
        <PathPointSet>
            <objects>
                <PathPoint name="P1">
                    <!--Path to a Component that satisfies the Socket 'parent_frame' of type Phy
                    <socket_parent_frame>/bodyset/brazo_exoesqueleto</socket_parent_frame>
                    <!--The fixed location of the path point expressed in its parent frame.-->
                    <location>0 -0.02 0</location>
                </PathPoint>
                <PathPoint name="P2">
                    <!--Path to a Component that satisfies the Socket 'parent_frame' of type Phy
                    <socket_parent_frame>/bodyset/antebrazo_exoesqueleto1</socket_parent_frame>
                    <!--The fixed location of the path point expressed in its parent frame.-->
                    <location>0 0 0</location>
                </PathPoint>
            </objects>
            <groups />
        </PathPointSet>
        <!--The wrap objects that are associated with this path-->
        <PathWrapSet>
            <objects />
            <groups />
        </PathWrapSet>
        <!--Default appearance attributes for this GeometryPath-->
        <Appearance>
            <!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
            <color>0 1 0</color>
        </Appearance>
    </GeometryPath>
    <!--The resting length (m) of the PathSpring-->
```

```
    <resting_length>0.059999999999999998</resting_length>
    <!--The linear stiffness (N/m) of the PathSpring-->
    <stiffness>10000</stiffness>
    <!--The dissipation factor (s/m) of the PathSpring-->
    <dissipation>10</dissipation>
</PathSpring>
```

Figura 3-5. Código para la adición del accionamiento mecánico en Opensim®

Lo más importante a destacar de este código es la definición de los dos puntos, P1 y P2, donde se quiere anclar los extremos del muelle. Hay que definir el sólido en el que se encuentran esos puntos, y sus posiciones según las coordenadas locales de cada uno de ellos respecto a su origen. Lo último es dar una longitud natural al muelle, una rigidez, y un factor de disipación.

Dado por explicado lo más importante a la hora de incorporar el exoesqueleto al modelo original del sujeto con la ayuda del editor de textos Notepad ++, se va a recoger en la siguiente tabla el conjunto de sólidos, uniones, y coordenadas generalizadas finalmente existentes en el modelo ya con el exoesqueleto incorporado:

Tabla 3-2. Sólidos, uniones y coordenadas generalizadas finales del modelo con exoesqueleto

Sólidos	**Uniones**	**Coordenadas generalizadas**
Tórax	Suelo-Tórax (ground_thorax)	ground_thorax_rot_x, ground_thorax_rot_y, ground_thorax_rot_z, ground_thorax_tx, ground_thorax_ty, ground_thorax_tz
Clavícula	Esternoclavicular (sternoclavicular)	clav_prot, clav_elev
Escápula	Escapulotorácica (scapulothoracic)	scapula_abduction, scapula_elevation, scapula_upward_rot, scapula_winging
Húmero	Glenohumeral [*][1]	plane_elv, shoulder_elv, axial_rot
Cúbito	Codo (elbow) [**][2]	elbow_flexion
Radio	Radio-Cúbito (radioulnar)	pro_sup
Mano	Radio-Mano (rc)	---
Masa puntual	Radio-Masa puntual (rmasapuntual)	---
Brazo exoesqueleto	Escápula-Brazo exoesqueleto (Glenobrazo) [*]	plane2_elv, shoulder2_elv, axial2_rot
Antebrazo exoesqueleto1	Brazo exoesqueleto- Antebrazo exoesqueleto1 (par_rot_brazoantebrazo) [**]	elbow2_flexion
Antebrazo exoesqueleto2	Antebrazo exoesqueleto2-Cúbito (antexo2_hand)	rot_x, rot_y, rot_z

[1] El símbolo [*] hace referencia a que esas dos uniones son iguales, con la única diferencia de que la primera representa la articulación del hombro real del sujeto y la segunda representa el par esférico del exoesqueleto que simula al hombro.
[2] El símbolo [**] hace referencia a que esas dos uniones son iguales, con la única diferencia de que la primera representa la articulación del codo real del sujeto y la segunda representa el par de rotación del exoesqueleto que simula al codo.

En la siguiente figura se puede ver el esquema topológico de cómo se relacionan estos sólidos y uniones mencionados en la anterior tabla:

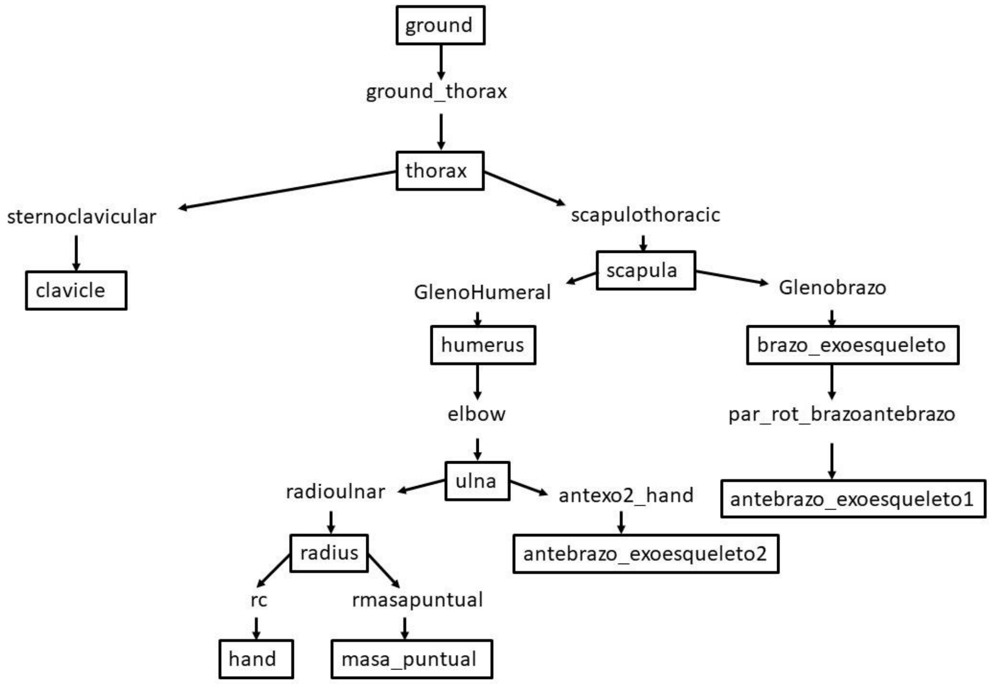


Figura 3-6. Esquema topológico del modelo con exoesqueleto

Finalmente, el modelo creado queda de la siguiente manera:

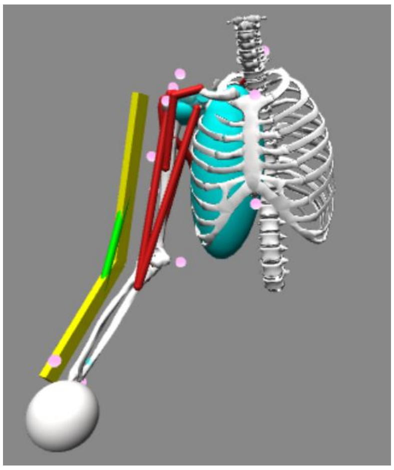


Figura 3-7. Modelo con exoesqueleto

3.2 Modelo en Matlab®

La única función del modelo en Matlab® es la de verificar que los resultados cinemáticos obtenidos por Opensim® sean válidos. Por este motivo la simulación de Matlab® se basa simplemente en simular la flexión de un brazo, sin tener en cuenta la adición de ningún exoesqueleto. En consecuencia, los resultados que se obtengan serán comparados con los de Opensim® sólo para el caso del sujeto sin exoesqueleto.

Una vez entendido esto, se va a proceder a explicar el modelo para la simulación en Matlab®. El brazo se ha modelado como un mecanismo de dos barras, muy similar al exoesqueleto visto en la Figura 3-7, simulando el brazo y el antebrazo. La barra que simula el brazo posee un par cinemático con dos grados de libertad en su extremo superior ($\psi\ y\ \theta_1$). Estos grados de libertad intentan simular la articulación del hombro de manera simplificada; más en concreto, intentan simular las coordenadas generalizadas de elevación frontal y lateral del hombro, respectivamente, definidas en Opensim®. Por tanto, serán comparadas con estas. De igual manera, estas dos barras se unen mediante un par de rotación, cuyo giro es medido por una coordenada relativa (θ_2) que imita a la coordenada generalizada de flexión de codo que aparece en Opensim®.

En resumen, el modelo se ha basado en un mecanismo formado por dos barras, cuyo movimiento se rige por tres coordenadas relativas, que se identifican como $\psi, \theta_1\ y\ \theta_2$. El sistema de coordenadas locales para cada barra se define aproximadamente con el mismo sentido y orientación que los que se definieron para el húmero y el cúbito del modelo original del sujeto (ver Figura 3-1), para así poder tomar las coordenadas locales en Matlab® iguales a las de Opensim®. Aquí se puede ver una ilustración de lo que sería el modelo a implemetar:

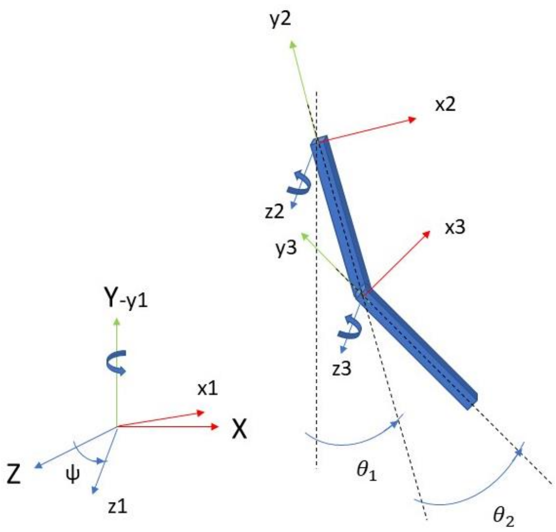


Figura 3-8. Modelo brazo de Matlab®

4 RESULTADOS Y COMPARACIONES

Este capítulo se va a dividir en dos secciones. La primera comparará y explicará cómo se han obtenido los resultados cinemáticos tanto en el programa de Opensim® como en el de Matlab®. Aquí se verá el modelo sin exoesqueleto, con el fin de verificar si los resultados del programa Opensim® son válidos.

La segunda sección pasará a ver los resultados cinemáticos únicamente de Opensim®, obtenidos para el caso del modelo con exoesqueleto, al que se le irá cambiando la configuración alargando o acortando sus barras y cambiando la movilidad de sus pares cinemáticos. Con esto se intentará ver que configuración es mejor o peor para el tamaño del sujeto en estudio.

4.1 Sujeto sin exoesqueleto

4.1.1 Coordenadas generalizadas

4.1.1.1 Posición

En primer lugar y puesto que son los resultados más importantes, se va a comparar las coordenadas generalizadas obtenidas de ejecutar la herramienta de cinemática inversa de Opensim®, y las obtenidas por Matlab®. Sabiendo estas coordenadas generalizadas se puede saber el resto de resultados cinemáticos. Es por esto de su importancia. Se va a centrar toda la atención en las coordenadas generalizadas de elevación frontal y lateral del hombro y en la flexión del codo.

En Opensim®, los resultados se obtienen de ejecutar, como se ha dicho, la herramienta de cinemática inversa, mientras se tiene abierto el modelo en el programa. Para ejecutar esta herramienta simplemente hay que meter los datos de entrada de las posiciones de los marcadores. Al ejecutarla se crea un archivo ".mot", en el que se recoge la evolución temporal de las coordenadas generalizadas del modelo.

Para Matlab® hay que explicar cómo se ha implementado su modelo para obtener los resultados cinemáticos, ya que a diferencia de Opensim®, no implementa los datos de entrada por si sólo, sino que hay que programar lo que se quiere obtener.

Para obtener las tres coordenadas generalizadas $\psi, \theta_1\ y\ \theta_2$, habrá que escribir en primer lugar las posiciones de todos los marcadores en función de estas coordenadas. En el Anexo B se puede ver cómo se ha hecho esto. Una vez reflejadas las posiciones en función de las coordenadas generalizadas, hay que aplicar la ecuación de los mínimos cuadrados ponderados (2.1), en la que se sacará el error que existe entre la posición de los marcadores experimentales y los del modelo (Anexo C). Para tener los mejores resultados posibles, hay que conseguir que este error sea mínimo. Esto se consigue con la ayuda de la función fmincon disponible en el programa de Matlab®, que consigue devolver los valores de las variables que hacen que la función objetivo (el error en este caso) tenga el mínimo valor posible (Anexo D).

Ejecutando todos estos códigos recogidos en los anteriores anexos, teniendo en cuenta que se tiene tres coordenadas generalizadas, y que este problema se resolverá para cada instante de tiempo, siendo el total 7,81 segundos con un paso de 0,01 segundos, se obtendrá como resultado una matriz de 3x782, correspondiendo cada fila a las coordenadas $\psi, \theta_1\ y\ \theta_2$, y cada columna a un instante de tiempo.

Finalmente, realizando todo lo anterior se obtienen las coordenadas generalizadas en cada instante de tiempo tanto en un programa como en otro. En las siguientes gráficas se mostrará la evolución temporal de estas coordenadas y se hará un breve comentario sobre sus diferencias o similitudes:

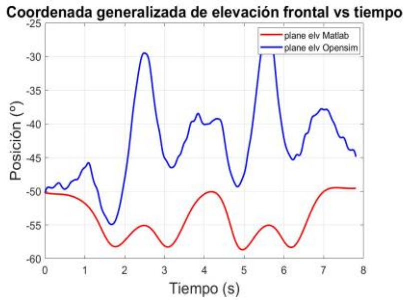


Figura 4-1. Posición elevación frontal vs tiempo

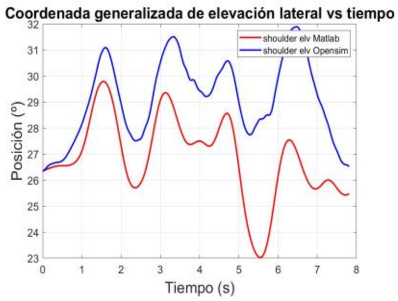


Figura 4-2. Posición elevación lateral vs tiempo

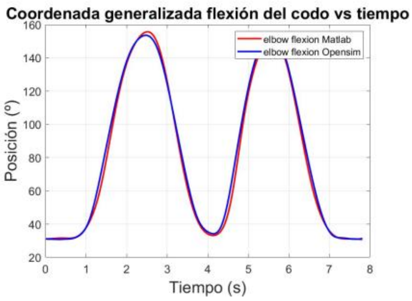


Figura 4-3. Posición flexión codo vs tiempo

En las gráficas de arriba se puede destacar los siguientes aspectos importantes. En la Figura 4-1 se puede ver como las subidas y las bajadas de la gráfica coinciden en el tiempo tanto en una curva como en otra, pero los valores se ven mucho más diferentes. Esto se puede explicar debido a que la articulación del hombro en el modelo de Opensim® se definió con tres coordenadas generalizadas (plane_elv, shoulder_elv y axial_rot), mientras que en Matlab® se definió con dos (ψ y θ_1) para simplificarlo, por lo tanto, estas coordenadas no coinciden exactamente entre ellas, y en el caso de la coordenada de elevación frontal puede verse una diferencia aún mayor. Otro aspecto que destacar es ver que en la Figura 4-3, correspondiente a la flexión del codo, sí se corresponde casi a la perfección el resultado obtenido tanto con un programa como con el otro. En la Figura 4-2, aunque no tanto como en la Figura 4-3, también se puede apreciar la similitud de los resultados.

4.1.1.2 Velocidad

A partir de la posición de las coordenadas generalizadas obtenidas en el apartado anterior, se pueden calcular las velocidades de estas mismas. Esto será necesario para el programa de Matlab®, por el que a través de derivación numérica, y más en concreto por el método de la diferencia central, se obtendrá programando las velocidades (Anexo E). En Opensim®, tan sólo será necesario ejecutar una herramienta para obtener los resultados, que en este caso será la de Análisis, en la que se le indica el tipo de análisis a realizar, siendo en este caso el análisis “Kinematics” y “BodyKinematics”. Esta herramienta además de las posiciones de las coordenadas generalizadas (igual que hacía la de cinemática inversa), devolverá sus velocidades, guardadas en un archivo “.sto”, de interés en este apartado (análisis “Kinematics”). También devolverá la posición y velocidad de los centros de masas de cada sólido que se creó en el modelo, resultados que se utilizarán en la siguiente subsección y que salen gracias a la indicación del análisis “BodyKinematics”.

A continuación, se muestran las velocidades de las coordenadas generalizadas a lo largo del tiempo obtenidas para los diferentes programas:

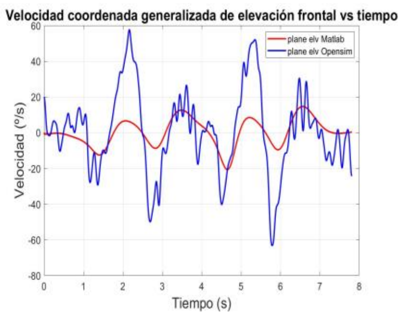


Figura 4-4. Velocidad elevación frontal vs tiempo

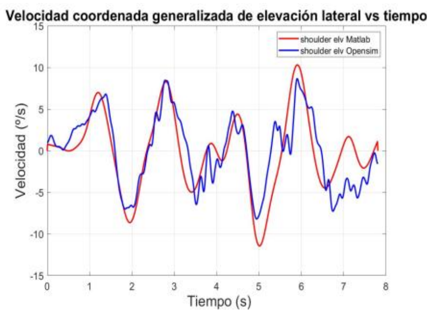


Figura 4-5. Velocidad elevación lateral vs tiempo

Figura 4-6. Velocidad flexión codo vs tiempo

Se obtiene algo similar a lo obtenido con las gráficas de posición. La velocidad de la coordenada generalizada de elevación frontal no coincide igual de bien que las otras por el mismo motivo que se dijo para la posición. En cambio, tanto las velocidades de las coordenadas de elevación lateral como la de flexión del codo salen casi idénticas para los dos programas.

4.1.2 Centro de masas del húmero y del cúbito

Para comparar los resultados de la posición y velocidad del centro de masas de los sólidos del modelo de Opensim® y de Matlab®, se centrará toda la atención en los sólidos del húmero y el cúbito, pues son los que más se identifican con las dos barras del modelo de Matlab®. Estas barras son la del brazo y el antebrazo, respectivamente.

4.1.2.1 Posición

Las posiciones de los centros de masas en cada instante de tiempo sacados de Opensim® se obtienen, como se adelantó en el apartado anterior, de la herramienta Análisis “BodyKinematics”, y quedan registradas en un archivo “.sto”.

Para obtener las posiciones de los centros de masas en Matlab® hay que ayudarse de la posición de las coordenadas generalizadas en cada instante obtenidas anteriormente por este programa; de la posición aproximada del origen del sistema de coordenadas locales del húmero y el cúbito del modelo de Opensim®; y de las posiciones en coordenadas locales de los centros de masas de las barras, que se cogerán iguales a las de los sólidos del húmero y el cúbito, definidos en el modelo original de Opensim®. El código programado para calcular estas posiciones queda recogido en el Anexo F, en el que también vendrá incluido el cálculo de la velocidad, de la que se hablará en el siguiente apartado.

A continuación, se puede ver los resultados de las posiciones de los centros de masas, para cada coordenada X, Y, Z, en el sistema de referencia global, calculadas por ambos programas y para cada sólido. Primero se mostrarán las del húmero:

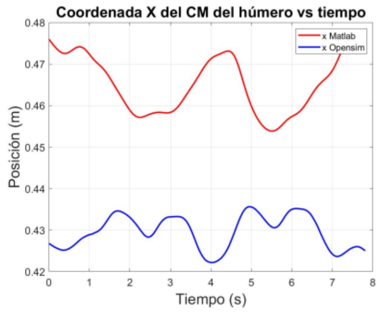


Figura 4-7. Coordenada X del CM del húmero vs tiempo

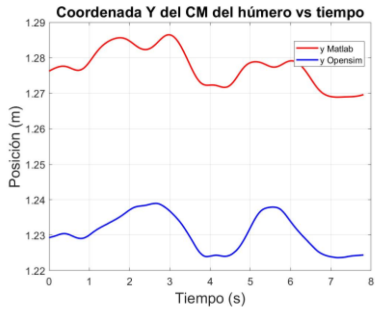


Figura 4-8. Coordenada Y del CM del húmero vs tiempo

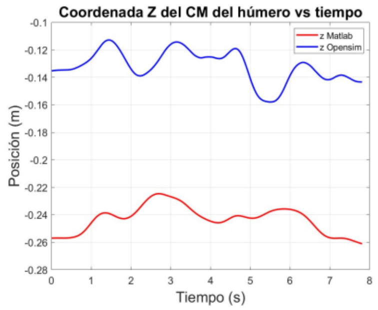


Figura 4-9. Coordenada Z del CM del húmero vs tiempo

A simple vista los resultados pueden verse bastante desiguales, pero hay que tener en cuenta la acumulación de error debida, por ejemplo, a la coordenada generalizada de elevación frontal, la cual no coincidía bien desde el primer instante de su cálculo con Matlab®. Otro motivo del error es el origen aproximado que se ha tomado del sistema de coordenadas locales del húmero y cúbito, de los cuales no se disponía de su posición numérica exacta. Se ha cogido la posición de dos marcadores, situado uno en el hombro y otro en el codo, que más se asemejaban visualmente a través de la interfaz gráfica de Opensim® a la posición real de estos orígenes. No por esto, se puede decir que todos los resultados sean negativos. Fijándose en la Figura 4-8, correspondiente a la coordenada Y, la distancia en todo momento de una curva a otra no dista más de algo menos de 5 centímetros, y describen una trayectoria bastante parecida.

En segundo lugar, se mostrarán los resultados del cúbito, mucho más interesantes ya que se sitúa en la parte del cuerpo (antebrazo) que más recorrido realiza en la acción de flexionar el brazo:

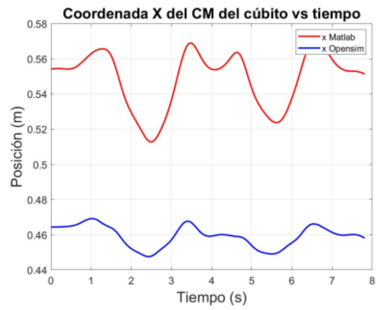


Figura 4-10. Coordenada X del CM del cúbito vs tiempo

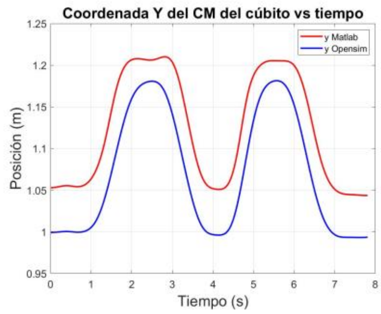


Figura 4-11. Coordenada Y del CM del cúbito vs tiempo

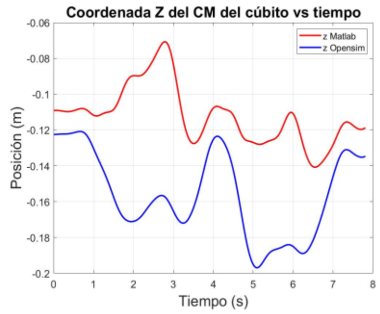


Figura 4-12. Coordenada Z del CM del cúbito vs tiempo

Fijándose principalmente en las coordenadas X e Y, sobre todo en esta última, se puede ver una gran similitud entre la forma de las curvas. Para la coordenada X, las curvas están muy separadas. Esto se puede deber a lo que se comentó anteriormente del error al coger el origen del sistema de coordenadas locales para cada sólido. Sin embargo, para la coordenada Y, sí se ve una buena aproximación, teniendo en cuenta además que esta coordenada representa el movimiento vertical, el cual es el más representativo, ya que el antebrazo al flexionarse no se mueve apenas a izquierda o derecha, pero sí se mueve más hacia arriba o hacia abajo.

4.1.2.2 Velocidad

Igual que se hizo en el caso del cálculo de la velocidad de las coordenadas generalizadas, hay que aplicar derivación numérica, para el caso de Matlab®, partiendo de las posiciones de los centros de masas de cada sólido obtenidas en el apartado anterior. Al final del Anexo F se ve recogido el programa, en el que se ha implantado el método de la diferencia central.

Las velocidades calculadas por Opensim® se obtienen, al igual que las posiciones, de la herramienta Análisis “BodyKinematics”, y quedan registradas en un archivo diferente pero también del tipo “.sto”.

Los resultados obtenidos para cada coordenada y para cada sólido son los siguientes, empezando primero por el húmero:

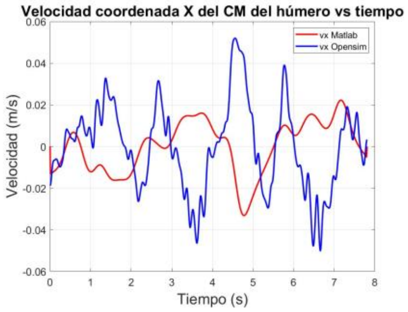


Figura 4-13. Vx del CM del húmero vs tiempo

Figura 4-14. Vy del CM del húmero vs tiempo

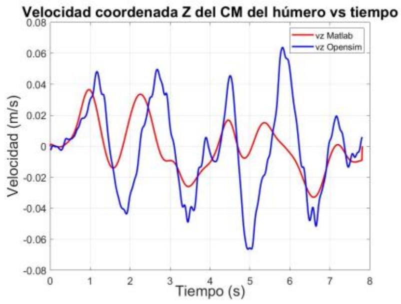


Figura 4-15. Vz del CM del húmero vs tiempo

En estos resultados se podría encontrar algo de similitud entre las curvas de la Figura 4-14 y Figura 4-15. Se puede ver como las subidas y las bajadas coinciden aproximadamente en el tiempo, mientras que los valores por el contrario son algo más diferentes. En la Figura 4-13, correspondiente a la velocidad de la coordenada X, no se ve ninguna similitud aparente, sabiendo que puede ser debido al error acumulado por la elección del origen de las coordenadas locales.

A continuación, se muestran las gráficas obtenidas para la velocidad del centro de masas del cúbito:

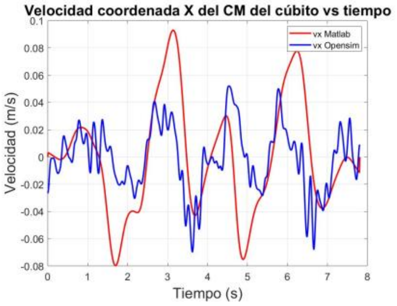


Figura 4-16. Vx del CM del cúbito vs tiempo

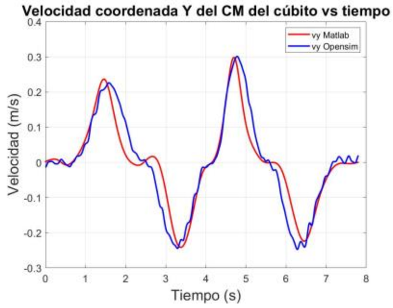


Figura 4-17. Vy del CM del cúbito vs tiempo

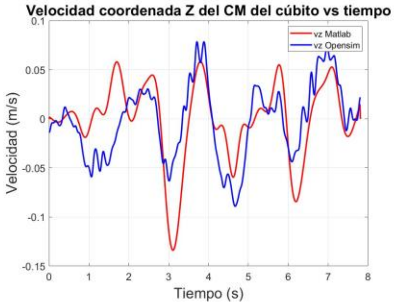


Figura 4-18. Vz del CM del cúbito vs tiempo

Al igual que pasaba para la posición, la velocidad del centro de masas del cúbito es mucho más interesante de analizar que la del húmero. En las gráficas anteriores se puede ver una gran coincidencia entre las curvas de cada una, destacando sobre todo la Figura 4-17, correspondiente a la coordenada Y, que representa la velocidad en la dirección vertical, siendo prácticamente iguales las curvas calculadas por Matlab® y por Opensim®.

4.2 Sujeto con exoesqueleto

4.2.1 Cambio en restricciones de los pares cinemáticos del exoesqueleto

En esta subsección se mostrarán los resultados cinemáticos obtenidos del modelo con exoesqueleto habiendo sido expuesto a distintas simulaciones en Opensim®. En estas simulaciones se irán cambiando las restricciones de las coordenadas que definen los pares cinemáticos entre el exoesqueleto y el sujeto. Todo esto se hará con el fin de ver qué configuración adopta un mejor comportamiento, que es el de proporcionar una buena maniobrabilidad al brazo del sujeto, y que el exoesqueleto se mueva de forma parecida a este.

Para llegar a elegir la mejor configuración se va a prestar únicamente atención a las coordenadas generalizadas de elevación frontal y lateral del hombro, y a la de flexión del codo, del exoesqueleto y del brazo del sujeto cuando lo porta, obtenidas mediante la herramienta Análisis "Kinematics". Estas coordenadas serán mejores cuanto más se asemejen a las coordenadas generalizadas de elevación frontal y lateral del hombro, y a la de flexión del codo del brazo del sujeto cuando no llevaba exoesqueleto.

Tras realizar varias simulaciones con un gran número de combinaciones posibles, se mostrarán los resultados obtenidos con seis de ellas, las consideradas más interesantes y relevantes. Estas seis simulaciones corresponden a los siguientes seis casos mostrados en la tabla:

Tabla 4-1. Simulaciones para el cambio en los pares cinemáticos

	Par cinemático al que afecta	Coordenada bloqueada
Simulación 1	Ninguno	Ninguna
Simulación 2	Antebrazo exoesqueleto2-Cúbito	Rotación de X
Simulación 3	Antebrazo exoesqueleto2-Cúbito	Rotación de X y Z
Simulación 4	Antebrazo exoesqueleto2-Cúbito	Rotación de X, Y y Z
Simulación 5	Escápula-Brazo exoesqueleto	Elevación lateral

Simulación 6	Escápula-Brazo exoesqueleto y Antebrazo exoesqueleto2-Cúbito	Elevación lateral y Rotación de X

Para empezar a descartar posibilidades, lo primero que se hará es comparar los resultados de cada coordenada generalizada, y para cada simulación del brazo del sujeto cuando porta al exoesqueleto, con los resultados que se obtuvieron de las coordenadas generalizadas del brazo del sujeto al hacer la flexión sin exoesqueleto. Cuanto más se parezcan mejor será, ya que significará que el exoesqueleto no afecta apenas al movimiento del brazo.

En las siguientes tres gráficas se verán los resultados por separado de las tres coordenadas generalizadas para lo explicado justo anteriormente:

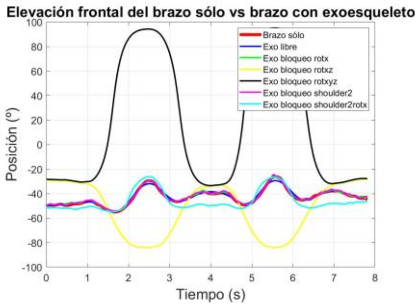


Figura 4-19. Elev. frontal del brazo sólo vs con exoesq.

Figura 4-20. Elev. lateral del brazo sólo vs con exoesq.

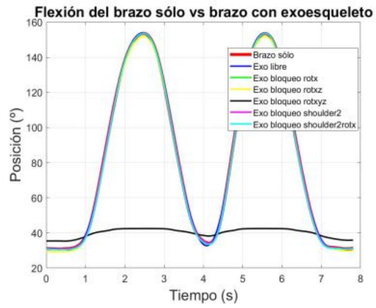


Figura 4-21. Flexión del codo del brazo sólo vs con exoesq.

Observando las anteriores gráficas ya se pueden descartar tres simulaciones como válidas. Puesto que la curva en color rojo corresponde a la del movimiento del brazo sin exoesqueleto, es la curva a la que se quiere que se parezcan las demás. Dicho esto, las tres simulaciones que se descartan son las simulaciones 3, 4 y 6 (Tabla 4-1), ya que se puede ver que sus curvas en color amarillo, negro y cian, respectivamente, son las curvas que no coinciden para nada de forma general con la roja, a excepción de alguna gráfica, mientras que las demás simulaciones si coinciden casi perfectamente.

Quedando tres simulaciones como candidatas a la mejor configuración, ahora se van a comparar los mismos resultados, pero obtenidos para el movimiento del exoesqueleto frente a los del brazo del sujeto sin él, para ver cuál es el que más se asimila:

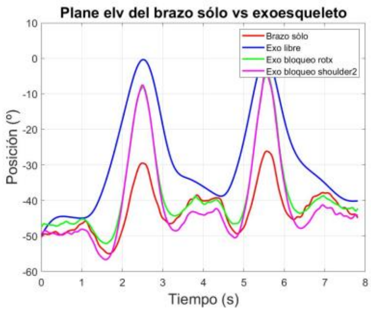


Figura 4-22. Elev. frontal del brazo sólo vs exoesq.

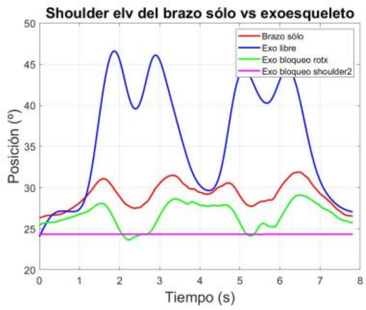


Figura 4-23. Elev. lateral del brazo sólo vs exoesq.

Figura 4-24. Flexión del codo del brazo sólo vs exoesq.

En las anteriores gráficas hay que fijarse de nuevo en qué curvas se parecen más a la de color rojo, cuanto más coincidan significará que el exoesqueleto se mueve lo más parecido posible a como se mueve el brazo del sujeto que lo porta, que es lo que más conviene conseguir, evitando así cualquier tipo de interferencia entre brazo del sujeto y exoesqueleto que pudiera comprometer la integridad del sujeto.

En la Figura 4-24 se puede ver cómo las tres simulaciones adoptan el mismo comportamiento, recorriendo aproximadamente las mismas trayectorias, por lo tanto habrá que fijarse en las otras dos gráficas para ver las diferencias y tomar la decisión de cuál es mejor.

La primera y más clara en ser descartada es la simulación 1 (Tabla 4-1), correspondiente a la curva azul. Es el caso en el que el exoesqueleto está libre. Puede apreciarse con claridad como en la Figura 4-22 y Figura 4-23 se aleja con mucha diferencia de la trayectoria que realiza el brazo del sujeto si, además, se compara con la de la curva verde y majenta, las cuales son muy parecidas entre ellas en la Figura 4-22.

La segunda y última simulación en ser descartada es la simulación 5 (Tabla 4-1), correspondiente a la curva majenta. Esta decisión se toma al observar la Figura 4-23, en la que se aprecia como la trayectoria de la simulación 5 se mantiene plana mientras que la de la simulación 2 (curva verde), aunque no igual, sí toma una forma similar a la curva roja, que es la curva a seguir.

Por tanto la configuración que mejor se acoge a los criterios que se quieren es la de la simulación 2, en la que la única coordenada bloqueada a la se le impide su movimiento es la de rotación alrededor del eje X, perteneciente al par cinemático que une el sólido del exoesqueleto llamado antebrazo_exoesqueleto2 con el cúbito del sujeto en el modelo de Opensim®, situado en la zona de la muñeca.

4.2.2 Cambio en las longitudes de las barras del exoesqueleto

Dado que todos los resultados anteriormente expuestos han sido sacados para unas longitudes de las barras del exoesqueleto aparentemente correctas para las dimensiones del sujeto, se va a proceder en esta subsección a modificar esas longitudes, llamadas a partir de ahora longitudes nominales. Esto servirá para mostrar qué efectos puede producir en el exoesqueleto y en el brazo del sujeto un mal diseño, queriendo así resaltar que no cualquier sujeto puede portar cualquier exoesqueleto, teniendo que estar adaptado este último, de una manera u otra, a las características fisiológicas del usuario para que cumpla un correcto funcionamiento y no ponga en peligro la integridad de este.

Para este estudio se ha hecho un total de tres simulaciones diferentes en Opensim®, en el que se han probado los siguientes casos alargando y acortando las barras que componen el exoesqueleto:

Tabla 4-2. Simulaciones para el cambio de longitudes en el exoesqueleto

	Barra alargada (% sobre el nominal)	Barra acortada (% sobre el nominal)
Simulación 1	Brazo (20%) y Antebrazo (36%)	---
Simulación 2	Brazo (20%)	Antebrazo (36%)
Simulación 3	Antebrazo (36%)	Brazo (20%)

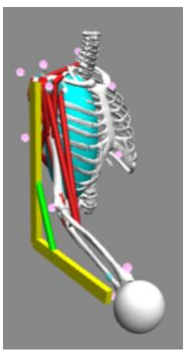


Figura 4-25. Simulación 1

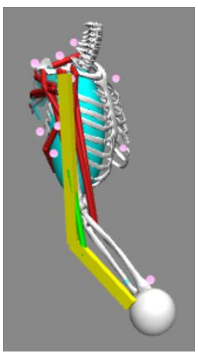


Figura 4-26. Simulación 2

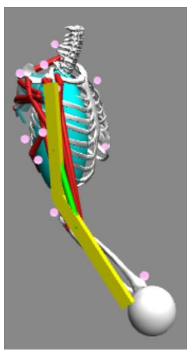


Figura 4-27. Simulación 3

Estas simulaciones han sido realizadas estando el modelo en la configuración que se eligió como mejor en la subsección anterior, es decir, para el caso en el que la única coordenada bloqueada del exoesqueleto era la de rotación alrededor del eje X en el par cinemático situado en la zona de la muñeca. Con esto y aplicando la herramienta Análisis "Kinematics" se logra obtener los siguientes resultados, que serán comparados a continuación con los obtenidos para el caso en el que las longitudes del exoesqueleto eran las nominales. Se verá el efecto en la trayectoria de las coordenadas generalizadas de siempre, tanto para el brazo del sujeto, como para el exoesqueleto:

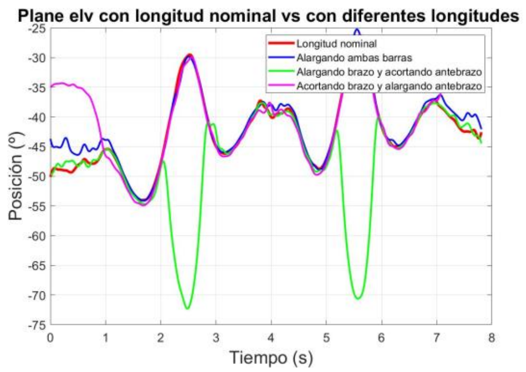


Figura 4-28. Elev. frontal del brazo con long. nominal vs diferentes longitudes

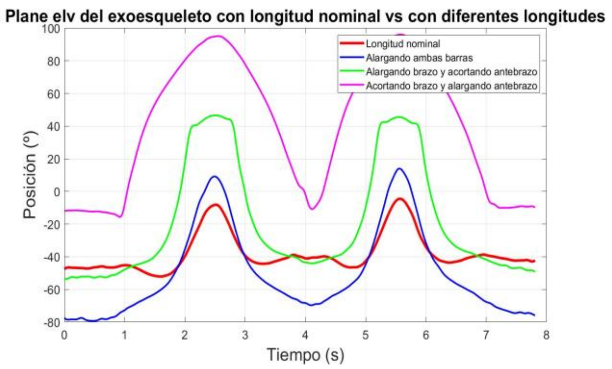


Figura 4-29. Elev. frontal del exoesq. con long. nominal vs diferentes longitudes

Para la elevación frontal del brazo (Figura 4-28) puede verse como la única simulación que desentona notablemente es la simulación 2 (Tabla 4-2), siendo la simulación 1 (Tabla 4-2) la que más se ajusta a la trayectoria de longitud nominal. Por otro lado, en la elevación frontal del exoesqueleto (Figura 4-29), partiendo de que ninguna de las simulaciones termina de ser correcta, la simulación 1 vuelve a ser la más parecida.

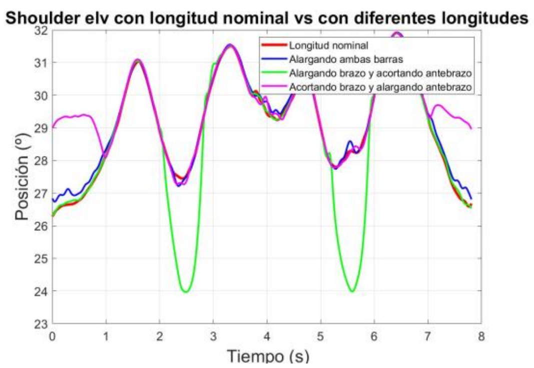


Figura 4-30. Elev. lateral del brazo con long. nominal vs diferentes longitudes

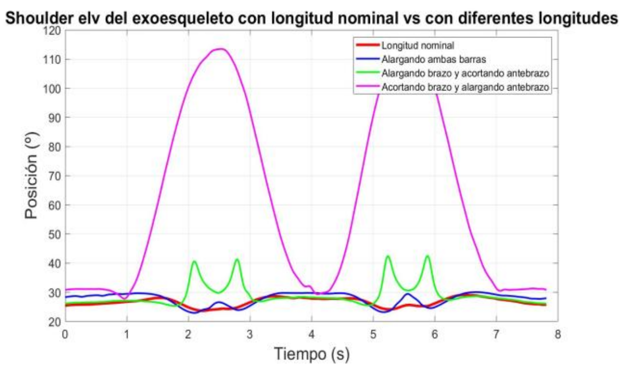
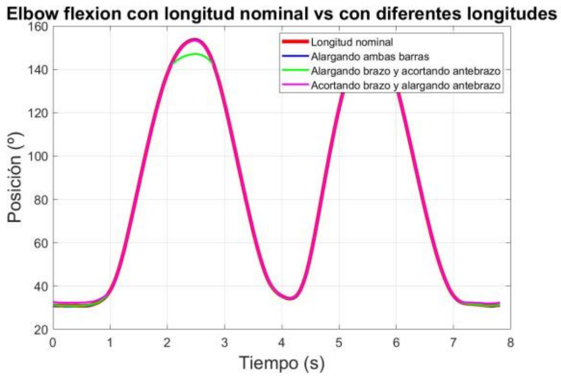


Figura 4-31. Elev. lateral del exoesq. con long. nominal vs diferentes longitudes

En las dos gráficas anteriores, fijándose en la coordenada generalizada de elevación lateral tanto para el brazo como para el exoesqueleto, se vuelve a apreciar algo parecido a lo comentado para las primeras dos gráficas de elevación frontal. La simulación 1 vuelve a dar los resultados más parecidos a los nominales, mientras que las otras dos simulaciones reflejan unos resultados que en la práctica no serían deseables desde el punto de vista cinemático.



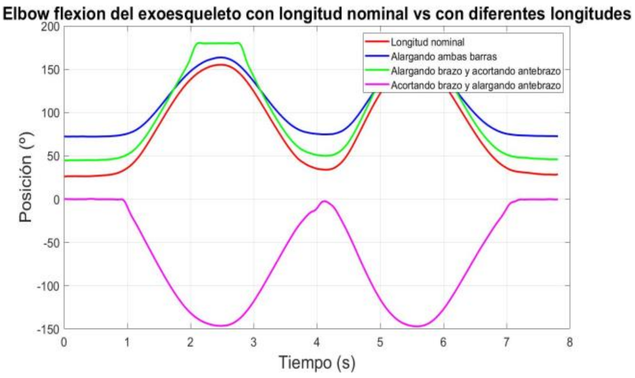


Figura 4-32. Flexión de codo del brazo con long. nominal vs diferentes longitudes Figura 4-33. Flexión de codo del exoesq. con long. nominal vs diferentes longitudes

En la flexión del codo, para el caso del brazo (Figura 4-32) puede verse como las tres simulaciones presentan un resultado que en principio podría ser válido, pero a la hora de mirar el movimiento del exoesqueleto, puede verse como la simulación 3 (Tabla 4-2) realiza un movimiento totalmente contrario al que se espera que haga, algo que es totalmente indeseable.

5 CONCLUSIONES Y ESTUDIOS FUTUROS

El objetivo de este estudio era el de validar en primer lugar los resultados cinemáticos proporcionados por el programa Opensim®. Esto se ha realizado al comparar estos resultados con los obtenidos por Matlab® en la sección 4.1. En esta sección se han expuesto las coordenadas generalizadas que gobiernan el movimiento de flexión del brazo del sujeto, su velocidad, y las posiciones y velocidades de los centros de masas del sólido del húmero y del cúbito. Finalmente, y de forma general, teniendo en cuenta el error dado por la forma de definir el modelo en Matlab®, más simplificada que en Opensim®, se puede concluir que los resultados entre ambos programas son bastante similares, concluyendo con esto que el programa Opensim® da unos resultados cinemáticos válidos con los que se puede trabajar de cara al estudio realizado en este trabajo.

En segundo lugar y una vez validado lo primero, lo que se quería de este estudio era ver qué configuración del exoesqueleto era la más conveniente para una buena maniobrabilidad del sujeto con él puesto, significando esto que el exoesqueleto no impidiera el movimiento natural del brazo del sujeto, ni creara ningún tipo de interferencia con él. Este estudio se ha realizado cambiando las restricciones de los pares cinemáticos, mediante distintas simulaciones de las que se han recogido los resultados, para unas longitudes de las barras que componen al exoesqueleto (longitudes nominales), similares al brazo y antebrazo del sujeto. La mejor configuración, demostrada en la subsección 4.2.1, es en la que la única coordenada generalizada a la que se le impide su movimiento es la de rotación alrededor del eje X para el par cinemático situado en la zona de la muñeca. Esta configuración permite un libre movimiento del brazo y consigue que el exoesqueleto se mueva de forma similar a este.

Por último, se probó a cambiar las longitudes de las barras que componían al exoesqueleto, alargándolas o acortándolas para ver cómo podía afectar esto a la cinemática cuando las dimensiones del exoesqueleto no correspondieran bien con las del usuario. Como conclusión, se ha podido ver que la única simulación de las tres que al menos podría llevarse, sin poner en riesgo la integridad del usuario, sería la del caso de alargar ambas barras. En definitiva, tanto por ahorro de material con respecto al modelo en que se alargan ambas barras, como por optimización del espacio y de funcionabilidad, el modelo con la longitud nominal de las barras es la mejor opción.

De cara al futuro del diseño de este exoesqueleto, falta por concretar y estudiar la dinámica. Esto conseguirá responder a preguntas del tipo dónde deben anclarse los extremos de los muelles para optimizar la transmisión de fuerza aportada al brazo del sujeto, qué rigidez y longitud natural debe llevar cada muelle, etc. Esto conseguirá realizar una optimización general del exoesqueleto, consiguiendo el mejor rendimiento posible. Otro aspecto interesante es la posible mejora de la articulación del hombro definida para Matlab®, ya que para este estudio se hizo una simplificación, que como se vio, fue válida. Según la finalidad del exoesqueleto, si esta cambiara, podría ser conveniente modelar esta articulación de una mejor manera, para obtener resultados más precisos en esta zona.

ANEXOS

Anexo A. Modelo con exoesqueleto en Opensim®


```xml
<?xml version="1.0" encoding="UTF-8" ?>
<OpenSimDocument Version="40000">
<Model name="Exoesqueleto_solobrazo">
<!--The model's ground reference frame.-->
<Ground name="ground">
<!--The geometry used to display the axes of this Frame.-->
<FrameGeometry name="frame_geometry">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
</FrameGeometry>
<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->
<attached_geometry />
<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->
<WrapObjectSet name="wrapobjectset">
<objects />
<groups />
</WrapObjectSet>
</Ground>
<!--Acceleration due to gravity, expressed in ground.-->
<gravity>0 -9.8066499999999994 0</gravity>
<!--Credits (e.g., model author names) associated with the model.-->
<credits>Ajay Seth, Meilin Dong, Ricardo Matias, Scott Delp. Parameters from van der Helm and Klein-Breteler</credits>
<!--Publications and references associated with the model.-->
<publications>Frontiers in Neurorobotics: in In Press DSEM: van der Helm 1994 Klein-Breteler et al. 1996.</publications>
<!--Units for all lengths.-->
<length_units>meters</length_units>
<!--Units for all forces.-->
```

<force_units>N</force_units>

<!--List of bodies that make up this model.-->

<BodySet name="bodyset">

<objects>

<Body name="thorax">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->

<attached_geometry>

<Mesh name="thorax_geom_1">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>1 1 1</scale_factors>

<!--Default appearance attributes for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>1 1 1</color>

</Appearance>

<!--Name of geometry file.-->

<mesh_file>thorax.vtp</mesh_file>

</Mesh>

<Mesh name="thorax_geom_2">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>1 1 1</scale_factors>

<!--Default appearance attributes for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>1 1 1</color>

</Appearance>

<!--Name of geometry file.-->

<mesh_file>groundspine.vtp</mesh_file>

</Mesh>

</attached_geometry>

<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->

<WrapObjectSet name="wrapobjectset">

<objects>

<WrapEllipsoid name="Thorax">

<!--Whether or not the WrapObject is considered active in computing paths-->

<active>true</active>

<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->

<xyz_body_rotation>-0.17000000000000001 -0.038397199999999999 0.212232</xyz_body_rotation>

<!--Translation of the WrapObject.-->

<translation>-0.021999879226828334 -0.14670090464761937 0.069998363349739287</translation>

<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->

<quadrant>all</quadrant>

<!--Default appearance for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

<representation>3</representation>

</SurfaceProperties>

</Appearance>

<dimensions> 0.0740668 0.169952 0.0704834</dimensions>

</WrapEllipsoid>

<WrapEllipsoid name="EllipsoidSurface">

<!--Whether or not the WrapObject is considered active in computing paths-->

<active>true</active>

<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->

<xyz_body_rotation>0 0 0</xyz_body_rotation>

<!--Translation of the WrapObject.-->

<translation>-0.019999932940622696 -0.017300088530082611 0.079672585302318549</translation>

<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->

<quadrant>all</quadrant>

<!--Default appearance for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>false</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>0.5</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

<representation>3</representation>

</SurfaceProperties>

</Appearance>

<dimensions> 0.0994228 0.200001 0.0993853</dimensions>

</WrapEllipsoid>

</objects>

<groups />

</WrapObjectSet>

<!--The mass of the body (kg)-->

<mass>0</mass>

<!--The location (Vec3) of the mass center in the body frame.-->

<mass_center>0 0 0</mass_center>

<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->

<inertia>0 0 0 0 0 0</inertia>

</Body>

<Body name="clavicle">

<!--List of components that this component owns and serializes.-->

<components>

<PhysicalOffsetFrame name="cp_in_clavicle">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>..</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>-0.029239999999999999 0.020240000000000001 0.12005</translation>

</PhysicalOffsetFrame>

</components>

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->

<attached_geometry>

<Mesh name="clavicle_geom_1">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>1 1 1</scale_factors>

<!--Default appearance attributes for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>1 1 1</color>

</Appearance>

<!--Name of geometry file.-->

<mesh_file>clavicle.vtp</mesh_file>

</Mesh>

</attached_geometry>

<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->

<WrapObjectSet name="wrapobjectset">

<objects />

<groups />

</WrapObjectSet>

<!--The mass of the body (kg)-->

<mass>0</mass>

<!--The location (Vec3) of the mass center in the body frame.-->

<mass_center>0 0 0</mass_center>

<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->

<inertia>0 0 0 0 0 0</inertia>

</Body>

<Body name="scapula">

<!--List of components that this component owns and serializes.-->

<components>

<PhysicalOffsetFrame name="cp_in_scapula">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>..</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the

parent frame.-->

<translation>-0.01357 0.00011 -0.01523</translation>

</PhysicalOffsetFrame>

</components>

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->

<attached_geometry>

<Mesh name="scapula_geom_1">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>1 1 1</scale_factors>

<!--Default appearance attributes for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>1 1 1</color>

</Appearance>

<!--Name of geometry file.-->

<mesh_file>scapula.vtp</mesh_file>

</Mesh>

</attached_geometry>

<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->

<WrapObjectSet name="wrapobjectset">

<objects>

<WrapCylinder name="Subscapuralis">

<!--Whether or not the WrapObject is considered active in computing paths-->

<active>true</active>

<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->

<xyz_body_rotation>1 -1 1</xyz_body_rotation>

<!--Translation of the WrapObject.-->

<translation>-0.025100932507589654 -0.038999500038472509 -0.028389876540875353</translation>

<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->

<quadrant>-y</quadrant>

<!--Default appearance for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>0.5</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

<representation>1</representation>

</SurfaceProperties>

</Appearance>

<!--The radius of the cylinder.-->

<radius>0.012555905670436653</radius>

<!--The length of the cylinder.-->

<length>0.085675855438011261</length>

</WrapCylinder>

<WrapCylinder name="Tmajscapula">

<!--Whether or not the WrapObject is considered active in computing paths-->

<active>true</active>

<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->

<xyz_body_rotation>-1.4854000000000001 -0.33539999999999998 -
0.87539999999999996</xyz_body_rotation>

<!--Translation of the WrapObject.-->

<translation>-0.089299517505347703 -0.1062046384893849 -0.050082241326954553</translation>

<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->

<quadrant>-y</quadrant>

<!--Default appearance for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>0.5</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

<representation>3</representation>

</SurfaceProperties>

</Appearance>

<!--The radius of the cylinder.-->

<radius>0.0075920502390882964</radius>

<!--The length of the cylinder.-->

<length>0.050291503590523694</length>

</WrapCylinder>

<WrapEllipsoid name="deltsca">

<!--Whether or not the WrapObject is considered active in computing paths-->

<active>true</active>

<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->

<xyz_body_rotation>1.4823 0.16059999999999999 -0.44719999999999999</xyz_body_rotation>

<!--Translation of the WrapObject.-->

<translation>-0.034299774247472652 -0.033401371804744902 -0.044284308793958879</translation>

<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->

<quadrant>all</quadrant>

<!--Default appearance for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>0.5</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

```
<representation>3</representation>
</SurfaceProperties>
</Appearance>
<dimensions> 0.0403287 0.0906472 0.0300459</dimensions>
</WrapEllipsoid>
<WrapCylinder name="Infra">
<!--Whether or not the WrapObject is considered active in computing paths-->
<active>true</active>
<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->
<xyz_body_rotation>1.8700000000000001 -0.54300000000000004 -2.355</xyz_body_rotation>
<!--Translation of the WrapObject.-->
<translation>-0.058119659163738707 -0.045901911551691904 -0.045633927537544652</translation>
<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->
<quadrant>-y</quadrant>
<!--Default appearance for this Geometry-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
<opacity>0.5</opacity>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0 1 1</color>
<!--Visuals applied to surfaces associated with this Appearance.-->
<SurfaceProperties>
<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->
<representation>3</representation>
</SurfaceProperties>
</Appearance>
<!--The radius of the cylinder.-->
<radius>0.019735315920344755</radius>
<!--The length of the cylinder.-->
<length>0.10116138098774374</length>
</WrapCylinder>
</objects>
<groups />
</WrapObjectSet>
<!--The mass of the body (kg)-->
```

<mass>0</mass>

<!--The location (Vec3) of the mass center in the body frame.-->

<mass_center>0 0 0</mass_center>

<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->

<inertia>0 0 0 0 0 0</inertia>

</Body>

<Body name="humerus">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->

<attached_geometry>

<Mesh name="humerus_geom_1">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>1 1 1</scale_factors>

<!--Default appearance attributes for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>1 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

<representation>3</representation>

</SurfaceProperties>

</Appearance>

<!--Name of geometry file.-->

<mesh_file>humerus.vtp</mesh_file>

</Mesh>

</attached_geometry>

<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->

<WrapObjectSet name="wrapobjectset">

<objects>

<WrapSphere name="hhead">

<!--Whether or not the WrapObject is considered active in computing paths-->

<active>true</active>

<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->

<xyz_body_rotation>0 0 0</xyz_body_rotation>

<!--Translation of the WrapObject.-->

<translation>-0.0032000013682117641 0.0049000028324595118 -0.0019999875686123655</translation>

<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->

<quadrant>all</quadrant>

<!--Default appearance for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

<representation>3</representation>

</SurfaceProperties>

</Appearance>

<radius>0.024539745611113304</radius>

</WrapSphere>

<WrapSphere name="hheadsupraTminor">

<!--Whether or not the WrapObject is considered active in computing paths-->

<active>true</active>

<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->

<xyz_body_rotation>0 0 0</xyz_body_rotation>

<!--Translation of the WrapObject.-->

<translation>-0.00080000034205294102 0.0048000244811105324 0.0022999872039023961</translation>

<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->

<quadrant>all</quadrant>

<!--Default appearance for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

<representation>3</representation>

</SurfaceProperties>

</Appearance>

<radius>0.023472800149760553</radius>

</WrapSphere>

<WrapEllipsoid name="hheadInfra">

<!--Whether or not the WrapObject is considered active in computing paths-->

<active>true</active>

<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->

<xyz_body_rotation>0.1419 0.13474 0.28116999999999998</xyz_body_rotation>

<!--Translation of the WrapObject.-->

<translation>9.9999417760256634e-05 0.0010000054335775059 -0.0047999741646648133</translation>

<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->

<quadrant>all</quadrant>

<!--Default appearance for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0 1 1</color>

<!--Visuals applied to surfaces associated with this Appearance.-->

<SurfaceProperties>

<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->

<representation>3</representation>

</SurfaceProperties>

</Appearance>

<dimensions> 0.0298252 0.0233659 0.0302255</dimensions>

</WrapEllipsoid>

</objects>

<groups />

</WrapObjectSet>

<!--The mass of the body (kg)-->

<mass>2.2123566462885935</mass>

<!--The location (Vec3) of the mass center in the body frame.-->

<mass_center>0.018059994846920106 -0.14010038452973017 -0.012749884499947897</mass_center>

<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->

<inertia>0.013588299999999999 0.00282526 0.013588299999999999 -0.00038473799999999997 - 0.00025749600000000002 0.00013610799999999999</inertia>

</Body>

<Body name="ulna">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->

<attached_geometry>

<Mesh name="ulna_geom_1">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>1 1 1</scale_factors>

<!--Default appearance attributes for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

```
<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
<opacity>1</opacity>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>1 1 1</color>
</Appearance>
<!--Name of geometry file.-->
<mesh_file>ulna.vtp</mesh_file>
</Mesh>
</attached_geometry>
<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->
<WrapObjectSet name="wrapobjectset">
<objects>
<WrapCylinder name="PQ2">
<!--Whether or not the WrapObject is considered active in computing paths-->
<active>true</active>
<!--Body-fixed Euler angle sequence for the orientation of the WrapObject-->
<xyz_body_rotation>1.3913800000000001 0.068940500000000002
1.5224500000000001</xyz_body_rotation>
<!--Translation of the WrapObject.-->
<translation>-0.0011999999999999999 -0.2092 0.042799999999999998</translation>
<!--The name of quadrant over which the wrap object is active. For example, '+x' or '-y' to set the sidedness of the wrapping.-->
<quadrant>all</quadrant>
<!--Default appearance for this Geometry-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
<opacity>0.5</opacity>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0 1 1</color>
<!--Visuals applied to surfaces associated with this Appearance.-->
<SurfaceProperties>
<!--The representation (1:Points, 2:Wire, 3:Shaded) used to display the object.-->
<representation>3</representation>
</SurfaceProperties>
</Appearance>
<!--The radius of the cylinder.-->
```

```
<radius>0.0069999999999999923</radius>
<!--The length of the cylinder.-->
<length>0.0099999999999999846</length>
</WrapCylinder>
</objects>
<groups />
</WrapObjectSet>
<!--The mass of the body (kg)-->
<mass>1.1529186312337683</mass>
<!--The location (Vec3) of the mass center in the body frame.-->
<mass_center>0.0097179999999999992 -0.095949999999999994 0.024289999999999999</mass_center>
<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and
not the body origin.-->
<inertia>0.0056461999999999997 0.0012026700000000001 0.0051559500000000003
0.00033055299999999998 -7.9430699999999998e-05 0.00113905</inertia>
</Body>
<Body name="radius">
<!--The geometry used to display the axes of this Frame.-->
<FrameGeometry name="frame_geometry">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
</FrameGeometry>
<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they
share the transform of the frame when visualized-->
<attached_geometry>
<Mesh name="radius_geom_1">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>1 1 1</scale_factors>
<!--Default appearance attributes for this Geometry-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
<opacity>1</opacity>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
```

<color>1 1 1</color>

</Appearance>

<!--Name of geometry file.-->

<mesh_file>radius.vtp</mesh_file>

</Mesh>

</attached_geometry>

<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->

<WrapObjectSet name="wrapobjectset">

<objects />

<groups />

</WrapObjectSet>

<!--The mass of the body (kg)-->

<mass>0.24366397562309627</mass>

<!--The location (Vec3) of the mass center in the body frame.-->

<mass_center>0.03363 -0.18156 0.015599999999999999</mass_center>

<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->

<inertia>0.000457496 9.2406599999999997e-05 4.1994499999999999e-05 3.1438500000000001e-05 -4.4226699999999996e-06 6.6945000000000004e-05</inertia>

</Body>

<Body name="hand">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->

<attached_geometry />

<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->

<WrapObjectSet name="wrapobjectset">

<objects />

<groups />

</WrapObjectSet>

<!--The mass of the body (kg)-->

<mass>0.54761809589951005</mass>

<!--The location (Vec3) of the mass center in the body frame.-->

<mass_center>0.00059999999999999995 -0.090499999999999997 -0.036499999999999998</mass_center>

<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->

<inertia>0.00066621699999999998 0.000198603 0.00066621699999999998 0 0 0</inertia>

</Body>

<Body name="masa_puntual">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->

<attached_geometry>

<Mesh name="masa_puntual_geom_1">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>1 1 1</scale_factors>

<!--Default appearance attributes for this Geometry-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->

<opacity>1</opacity>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>1 1 1</color>

</Appearance>

<!--Name of geometry file.-->

<mesh_file>masa_puntual.stl</mesh_file>

</Mesh>

</attached_geometry>

<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->

<WrapObjectSet name="wrapobjectset">

```xml
<objects />
<groups />
</WrapObjectSet>
<!--The mass of the body (kg)-->
<mass>4.9500000000000002</mass>
<!--The location (Vec3) of the mass center in the body frame.-->
<mass_center>0.00059999999999999995 -0.090499999999999997 -
0.036499999999999998</mass_center>
<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->
<inertia>0.01 0.01 0.01 0 0 0</inertia>
</Body>
<Body name="brazo_exoesqueleto">
<!--The geometry used to display the axes of this Frame.-->
<FrameGeometry name="frame_geometry">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
</FrameGeometry>
<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->
<attached_geometry>
<Mesh name="brazo_exoesqueleto_geom_1">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>0.3 3.25 0.15</scale_factors>
<!--Default appearance attributes for this Geometry-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
<opacity>1</opacity>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>1 1 0</color>
</Appearance>
<!--Name of geometry file.-->
<mesh_file>block.vtp</mesh_file>
```

```
</Mesh>
</attached_geometry>
<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->
<WrapObjectSet name="wrapobjectset">
<objects />
<groups />
</WrapObjectSet>
<!--The mass of the body (kg)-->
<mass>0.29999999999999999</mass>
<!--The location (Vec3) of the mass center in the body frame.-->
<mass_center>0 0 0</mass_center>
<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->
<inertia>0.01 0.01 0.01 0 0 0</inertia>
</Body>
<Body name="antebrazo_exoesqueleto1">
<!--The geometry used to display the axes of this Frame.-->
<FrameGeometry name="frame_geometry">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
</FrameGeometry>
<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->
<attached_geometry>
<Mesh name="antebrazo_exoesqueleto1_geom_1">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>0.3 1.25 0.15</scale_factors>
<!--Default appearance attributes for this Geometry-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
<opacity>1</opacity>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
```

```xml
<color>1 1 0</color>
</Appearance>
<!--Name of geometry file.-->
<mesh_file>block.vtp</mesh_file>
</Mesh>
</attached_geometry>
<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->
<WrapObjectSet name="wrapobjectset">
<objects />
<groups />
</WrapObjectSet>
<!--The mass of the body (kg)-->
<mass>0.29999999999999999</mass>
<!--The location (Vec3) of the mass center in the body frame.-->
<mass_center>0 0 0</mass_center>
<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->
<inertia>0.01 0.01 0.01 0 0 0</inertia>
</Body>
<Body name="antebrazo_exoesqueleto2">
<!--The geometry used to display the axes of this Frame.-->
<FrameGeometry name="frame_geometry">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>
</FrameGeometry>
<!--List of geometry attached to this Frame. Note, the geometry are treated as fixed to the frame and they share the transform of the frame when visualized-->
<attached_geometry>
<Mesh name="antebrazo_exoesqueleto2_geom_1">
<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->
<socket_frame>..</socket_frame>
<!--Scale factors in X, Y, Z directions respectively.-->
<scale_factors>0.3 1.25 0.15</scale_factors>
<!--Default appearance attributes for this Geometry-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
```

```
<visible>true</visible>
<!--The opacity used to display the geometry between 0:transparent, 1:opaque.-->
<opacity>1</opacity>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>1 1 0</color>
</Appearance>
<!--Name of geometry file.-->
<mesh_file>block.vtp</mesh_file>
</Mesh>
</attached_geometry>
<!--Set of wrap objects fixed to this body that GeometryPaths can wrap over.This property used to be a member of Body but was moved up with the introduction of Frames.-->
<WrapObjectSet name="wrapobjectset">
<objects />
<groups />
</WrapObjectSet>
<!--The mass of the body (kg)-->
<mass>0.29999999999999999</mass>
<!--The location (Vec3) of the mass center in the body frame.-->
<mass_center>0 0 0</mass_center>
<!--The elements of the inertia tensor (Vec6) as [Ixx Iyy Izz Ixy Ixz Iyz] measured about the mass_center and not the body origin.-->
<inertia>0.01 0.01 0.01 0 0 0</inertia>
</Body>
</objects>
<groups />
</BodySet>
<!--List of joints that connect the bodies.-->
<JointSet name="jointset">
<objects>
<CustomJoint name="ground_thorax">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->
<socket_parent_frame>ground_offset</socket_parent_frame>
<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->
<socket_child_frame>thorax_offset</socket_child_frame>
<!--List containing the generalized coordinates (q's) that parameterize this joint.-->
<coordinates>
```

<Coordinate name="ground_thorax_rot_x">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>0.0697165594726359</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="ground_thorax_rot_y">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.063169559111135171</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="ground_thorax_rot_z">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.18216708121948563</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="ground_thorax_tx">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.0042313069906338074</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>true</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="ground_thorax_ty">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>0.0040303597703426784</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>true</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="ground_thorax_tz">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.00078351846075070297</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>true</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

</coordinates>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="ground_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/ground</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="thorax_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/thorax</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>-0.20000000000000001 -1.3999999999999999 0.20000000000000001</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

<!--Defines how the child body moves with respect to the parent as a function of the generalized coordinates.-->

<SpatialTransform>

<!--3 Axes for rotations are listed first.-->

<TransformAxis name="rotation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>ground_thorax_rot_x</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>1 0 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<TransformAxis name="rotation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>ground_thorax_rot_y</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 1 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<TransformAxis name="rotation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>ground_thorax_rot_z</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 0 1</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<!--3 Axes for translations are listed next.-->

<TransformAxis name="translation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>ground_thorax_tx</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>1 0 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<TransformAxis name="translation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>ground_thorax_ty</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 1 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<TransformAxis name="translation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>ground_thorax_tz</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 0 1</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

</SpatialTransform>

</CustomJoint>

<CustomJoint name="sternoclavicular">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->

<socket_parent_frame>thorax_offset</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->

<socket_child_frame>clavicle_offset</socket_child_frame>

<!--List containing the generalized coordinates (q's) that parameterize this joint.-->

<coordinates>

<Coordinate name="clav_prot">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.21559362239068142</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="clav_elev">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.10462137377195073</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s

and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

</coordinates>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="thorax_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/thorax</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0.006324976528009351 0.0069300434097705312 0.025464431687451307</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="clavicle_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/clavicle</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

<!--Defines how the child body moves with respect to the parent as a function of the generalized coordinates.-->

<SpatialTransform>

<!--3 Axes for rotations are listed first.-->

<TransformAxis name="rotation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>clav_prot</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0.015299999999999999 0.98929869999999998 -0.14509996</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<TransformAxis name="rotation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>clav_elev</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>-0.99447253999999996 0 -0.10499695000000001</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along

a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<TransformAxis name="rotation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 1 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<Constant name="function">

<value>0</value>

</Constant>

</TransformAxis>

<!--3 Axes for translations are listed next.-->

<TransformAxis name="translation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>1 0 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

<value>0</value>

</Constant>

</function>

<scale>0.99999644921000319</scale>

</MultiplierFunction>

</TransformAxis>

<TransformAxis name="translation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 1 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

<value>0</value>

</Constant>

</function>

<scale>1.000006693199808</scale>

</MultiplierFunction>

</TransformAxis>

<TransformAxis name="translation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 0 1</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

<value>0</value>

</Constant>

</function>

<scale>0.99997632556685589</scale>

</MultiplierFunction>

</TransformAxis>

</SpatialTransform>

</CustomJoint>

<ScapulothoracicJoint name="scapulothoracic">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->

<socket_parent_frame>thorax_offset</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->

<socket_child_frame>scapula_offset</socket_child_frame>

<!--List containing the generalized coordinates (q's) that parameterize this joint.-->

<coordinates>

<Coordinate name="scapula_abduction">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.18991246216762417</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="scapula_elevation">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.055819649390285428</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="scapula_upward_rot">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>0.13672537563074741</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415926500000002 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="scapula_winging">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>0.27710651929726871</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-1.5707963300000001 1.5707963300000001</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

</coordinates>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="thorax_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/thorax</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>-0.029422884944949112 -0.017300088530082611 0.070285568667052628</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 -0.87 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="scapula_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/scapula</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>-0.059819622317908099 -0.039041599498770452 -0.055980238178321401</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>-0.5181 -1.1415999999999999 -0.28539999999999999</orientation>

</PhysicalOffsetFrame>

</frames>

<!--Radii of the thoracic surface ellipsoid a Vec3(rX, rY, rZ).-->

<thoracic_ellipsoid_radii_x_y_z>0.082997500000000002 0.199991 0.083001000000000005</thoracic_ellipsoid_radii_x_y_z>

<!--Winging axis origin (x,y coordinates) in the scapula plane (tangent to the thoracic surface).-->

<scapula_winging_axis_origin>0 0</scapula_winging_axis_origin>

<!--Winging axis orientation (in radians) in the scapula plane.-->

<scapula_winging_axis_direction>0</scapula_winging_axis_direction>

</ScapulothoracicJoint>

<CustomJoint name="GlenoHumeral">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->

<socket_parent_frame>scapula_offset</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->

<socket_child_frame>humerus_offset</socket_child_frame>

<!--List containing the generalized coordinates (q's) that parameterize this joint.-->

<coordinates>

<Coordinate name="plane_elv">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>0.03392367703351757</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>1</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-1.5 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with

valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="shoulder_elv">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>0.60413286502026475</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-0.78000000000000003 2.5299999999999998</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="axial_rot">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.35844586211312002</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>1</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-1.5707950000000199 1.5707950000000199</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g.

default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

</coordinates>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="scapula_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/scapula</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>-0.0095499447834722345 -0.034001364113029092 0.0089968218508511211</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="humerus_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

```
<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->
<socket_parent>/bodyset/humerus</socket_parent>
<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->
<translation>0 0 0</translation>
<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->
<orientation>0 0 0</orientation>
</PhysicalOffsetFrame>
</frames>
<!--Defines how the child body moves with respect to the parent as a function of the generalized coordinates.--
>
<SpatialTransform>
<!--3 Axes for rotations are listed first.-->
<TransformAxis name="rotation1">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates>plane_elv</coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 1 0</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<LinearFunction name="function">
<coefficients> 1 0</coefficients>
</LinearFunction>
</TransformAxis>
<TransformAxis name="rotation2">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates>shoulder_elv</coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>-1 0 0</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<LinearFunction name="function">
<coefficients> 1 0</coefficients>
</LinearFunction>
</TransformAxis>
<TransformAxis name="rotation3">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates>axial_rot</coordinates>
```

```
<!--Rotation or translation axis for the transform.-->
<axis>-0.084599999999999995 0.99470000000000003 -0.058400000000000001</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<LinearFunction name="function">
<coefficients> 1 0</coefficients>
</LinearFunction>
</TransformAxis>
<!--3 Axes for translations are listed next.-->
<TransformAxis name="translation1">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>1 0 0</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>0.99999397216424735</scale>
</MultiplierFunction>
</TransformAxis>
<TransformAxis name="translation2">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 1 0</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>1.0000402623835893</scale>
```

```
</MultiplierFunction>
</TransformAxis>
<TransformAxis name="translation3">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 0 1</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>0.99964647686094243</scale>
</MultiplierFunction>
</TransformAxis>
</SpatialTransform>
</CustomJoint>
<CustomJoint name="elbow">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->
<socket_parent_frame>humerus_offset</socket_parent_frame>
<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->
<socket_child_frame>ulna_offset</socket_child_frame>
<!--List containing the generalized coordinates (q's) that parameterize this joint.-->
<coordinates>
<Coordinate name="elbow_flexion">
<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->
<default_value>0.5939476268819579</default_value>
<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->
<default_speed_value>0</default_speed_value>
<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->
<range>0 2.2689280300000001</range>
<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->
```

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

</coordinates>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="humerus_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/humerus</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0.0061000044831427585 -0.29040116109488806 -0.01229988504701285</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="ulna_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/ulna</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

<!--Defines how the child body moves with respect to the parent as a function of the generalized coordinates.-->

<SpatialTransform>

<!--3 Axes for rotations are listed first.-->

<TransformAxis name="rotation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>1 0 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<Constant name="function">

<value>0</value>

</Constant>

</TransformAxis>

<TransformAxis name="rotation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 1 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<Constant name="function">

<value>0</value>

</Constant>

</TransformAxis>

<TransformAxis name="rotation3">

```
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates>elbow_flexion</coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0.049400010000000001 0.036600010000000002 0.99810825000000003</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<LinearFunction name="function">
<coefficients> 1 0</coefficients>
</LinearFunction>
</TransformAxis>
<!--3 Axes for translations are listed next.-->
<TransformAxis name="translation1">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>1 0 0</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>1.0000002849560037</scale>
</MultiplierFunction>
</TransformAxis>
<TransformAxis name="translation2">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 1 0</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
```

</function>

<scale>1.0000053431078386</scale>

</MultiplierFunction>

</TransformAxis>

<TransformAxis name="translation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 0 1</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

<value>0</value>

</Constant>

</function>

<scale>0.99999410549189738</scale>

</MultiplierFunction>

</TransformAxis>

</SpatialTransform>

</CustomJoint>

<CustomJoint name="radioulnar">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->

<socket_parent_frame>ulna_offset</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->

<socket_child_frame>radius_offset</socket_child_frame>

<!--List containing the generalized coordinates (q's) that parameterize this joint.-->

<coordinates>

<Coordinate name="pro_sup">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.011244544653113928</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-1.5707963300000001 1.5707963300000001</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

</coordinates>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="ulna_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/ulna</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0.00040000000000000002 -0.011502999999999999 0.019998999999999999</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="radius_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/radius</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

<!--Defines how the child body moves with respect to the parent as a function of the generalized coordinates.-->

<SpatialTransform>

<!--3 Axes for rotations are listed first.-->

<TransformAxis name="rotation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>1 0 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<Constant name="function">

<value>0</value>

</Constant>

</TransformAxis>

<TransformAxis name="rotation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>pro_sup</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>-0.017160990000000001 0.99266564000000002 -0.11966796</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

```
</TransformAxis>
<TransformAxis name="rotation3">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 0 1</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<Constant name="function">
<value>0</value>
</Constant>
</TransformAxis>
<!--3 Axes for translations are listed next.-->
<TransformAxis name="translation1">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>1 0 0</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>1</scale>
</MultiplierFunction>
</TransformAxis>
<TransformAxis name="translation2">
<!--Names of the coordinates that serve as the independent variables        of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 1 0</axis>
<!--Transform function of the generalized coordinates used to        represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
```

<value>0</value>

</Constant>

</function>

<scale>1</scale>

</MultiplierFunction>

</TransformAxis>

<TransformAxis name="translation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 0 1</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

<value>0</value>

</Constant>

</function>

<scale>1</scale>

</MultiplierFunction>

</TransformAxis>

</SpatialTransform>

</CustomJoint>

<WeldJoint name="rc">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->

<socket_parent_frame>radius_offset</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->

<socket_child_frame>hand_offset</socket_child_frame>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="radius_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/radius</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0.017999999999999999 -0.24199999999999999 0.025000000000000001</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="hand_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/hand</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

</WeldJoint>

<WeldJoint name="rmasapuntual">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->

<socket_parent_frame>radius_offset</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->

<socket_child_frame>masa_puntual_offset</socket_child_frame>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="radius_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/radius</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0.017999999999999999 -0.28999999999999998 0.025000000000000001</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="masa_puntual_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/masa_puntual</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

</WeldJoint>

<CustomJoint name="Glenobrazo">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->

<socket_parent_frame>scapula_offset</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->

<socket_child_frame>brazo_exoesqueleto_offset</socket_child_frame>

<!--List containing the generalized coordinates (q's) that parameterize this joint.-->

<coordinates>

<Coordinate name="plane2_elv">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.12777434122024922</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>1</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-1.5 3.1415926500000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="shoulder2_elv">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>0.55061086988285379</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-0.78000000000000003 2.5299999999999998</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="axial2_rot">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.47957994684189187</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>1</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-1.5707950000000199 1.5707950000000199</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

</coordinates>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="scapula_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/scapula</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0.029999999999999999 0.059999999999999998</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="brazo_exoesqueleto_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/brazo_exoesqueleto</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0.17999999999999999 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

<!--Defines how the child body moves with respect to the parent as a function of the generalized coordinates.-->

<SpatialTransform>

<!--3 Axes for rotations are listed first.-->

<TransformAxis name="rotation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>plane2_elv</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 1 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<TransformAxis name="rotation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>shoulder2_elv</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>-1 0 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<TransformAxis name="rotation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>axial2_rot</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>-0.084599999999999995 0.99470000000000003 -0.058400000000000001</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<!--3 Axes for translations are listed next.-->

<TransformAxis name="translation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>1 0 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

<value>0</value>

</Constant>

</function>

<scale>0.99999397216424735</scale>

</MultiplierFunction>

</TransformAxis>

<TransformAxis name="translation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 1 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

<value>0</value>

</Constant>

</function>

<scale>1.0000402623835893</scale>

</MultiplierFunction>

</TransformAxis>

<TransformAxis name="translation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 0 1</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

```xml
<value>0</value>
</Constant>
</function>
<scale>0.99964647686094243</scale>
</MultiplierFunction>
</TransformAxis>
</SpatialTransform>
</CustomJoint>
<CustomJoint name="par_rot_brazoantebrazo">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->
<socket_parent_frame>brazo_exoesqueleto_offset</socket_parent_frame>
<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->
<socket_child_frame>antebrazo_exoesqueleto1_offset</socket_child_frame>
<!--List containing the generalized coordinates (q's) that parameterize this joint.-->
<coordinates>
<Coordinate name="elbow2_flexion">
<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->
<default_value>0.5939961667846837</default_value>
<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->
<default_speed_value>0</default_speed_value>
<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->
<range>0 2.2689280300000001</range>
<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->
<clamped>false</clamped>
<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->
<locked>false</locked>
<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->
<prescribed_function />
<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->
<prescribed>false</prescribed>
</Coordinate>
</coordinates>
```

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="brazo_exoesqueleto_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/brazo_exoesqueleto</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 -0.16 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="antebrazo_exoesqueleto1_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/antebrazo_exoesqueleto1</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0.059999999999999998 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

<!--Defines how the child body moves with respect to the parent as a function of the generalized coordinates.--
>

<SpatialTransform>

<!--3 Axes for rotations are listed first.-->

<TransformAxis name="rotation1">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>1 0 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<Constant name="function">

<value>0</value>

</Constant>

</TransformAxis>

<TransformAxis name="rotation2">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 1 0</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<Constant name="function">

<value>0</value>

</Constant>

</TransformAxis>

<TransformAxis name="rotation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates>elbow2_flexion</coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0.049400010000000001 0.036600010000000002 0.99810825000000003</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<LinearFunction name="function">

<coefficients> 1 0</coefficients>

</LinearFunction>

</TransformAxis>

<!--3 Axes for translations are listed next.-->

```
<TransformAxis name="translation1">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>1 0 0</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>1.0000002849560037</scale>
</MultiplierFunction>
</TransformAxis>
<TransformAxis name="translation2">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 1 0</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>1.0000053431078386</scale>
</MultiplierFunction>
</TransformAxis>
<TransformAxis name="translation3">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 0 1</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
```

```
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>0.99999410549189738</scale>
</MultiplierFunction>
</TransformAxis>
</SpatialTransform>
</CustomJoint>
<CustomJoint name="antexo2_hand">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The parent frame for the joint.).-->
<socket_parent_frame>ulna_offset</socket_parent_frame>
<!--Path to a Component that satisfies the Socket 'child_frame' of type PhysicalFrame (description: The child frame for the joint.).-->
<socket_child_frame>antebrazo_exoesqueleto2_offset</socket_child_frame>
<!--List containing the generalized coordinates (q's) that parameterize this joint.-->
<coordinates>
<Coordinate name="rot_x">
<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->
<default_value>-0.13880854938269332</default_value>
<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->
<default_speed_value>1</default_speed_value>
<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->
<range>-3.1415000000000002 3.1415000000000002</range>
<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->
<clamped>true</clamped>
<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->
<locked>false</locked>
<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->
<prescribed_function />
<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->
```

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="rot_y">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.21279812613915966</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>0</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415000000000002 3.1415000000000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>false</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

<Coordinate name="rot_z">

<!--The value of this coordinate before any value has been set. Rotational coordinate value is in radians and Translational in meters.-->

<default_value>-0.09965793130241761</default_value>

<!--The speed value of this coordinate before any value has been set. Rotational coordinate value is in rad/s and Translational in m/s.-->

<default_speed_value>1</default_speed_value>

<!--The minimum and maximum values that the coordinate can range between. Rotational coordinate range in radians and Translational in meters.-->

<range>-3.1415000000000002 3.1415000000000002</range>

<!--Flag indicating whether or not the values of the coordinates should be limited to the range, above.-->

<clamped>true</clamped>

<!--Flag indicating whether or not the values of the coordinates should be constrained to the current (e.g. default) value, above.-->

<locked>false</locked>

<!--If specified, the coordinate can be prescribed by a function of time. It can be any OpenSim Function with valid second order derivatives.-->

<prescribed_function />

<!--Flag indicating whether or not the values of the coordinates should be prescribed according to the function above. It is ignored if the no prescribed function is specified.-->

<prescribed>false</prescribed>

</Coordinate>

</coordinates>

<!--Physical offset frames owned by the Joint that are typically used to satisfy the owning Joint's parent and child frame connections (sockets). PhysicalOffsetFrames are often used to describe the fixed transformation from a Body's origin to another location of interest on the Body (e.g., the joint center). When the joint is deleted, so are the PhysicalOffsetFrame components in this list.-->

<frames>

<PhysicalOffsetFrame name="ulna_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/ulna</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 -0.25 0.10000000000000001</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="antebrazo_exoesqueleto2_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/antebrazo_exoesqueleto2</socket_parent>

```
<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->
<translation>0 -0.059999999999999998 0</translation>
<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->
<orientation>0 0 0</orientation>
</PhysicalOffsetFrame>
</frames>
<!--Defines how the child body moves with respect to the parent as a function of the generalized coordinates.-->
<SpatialTransform>
<!--3 Axes for rotations are listed first.-->
<TransformAxis name="rotation1">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates>rot_x</coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>1 0 0</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<LinearFunction name="function">
<coefficients> 1 0</coefficients>
</LinearFunction>
</TransformAxis>
<TransformAxis name="rotation2">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates>rot_y</coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 1 0</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<LinearFunction name="function">
<coefficients> 1 0</coefficients>
</LinearFunction>
</TransformAxis>
<TransformAxis name="rotation3">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates>rot_z</coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 0 1</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along
```

```
a specified axis.-->
<LinearFunction name="function">
<coefficients> 1 0</coefficients>
</LinearFunction>
</TransformAxis>
<!--3 Axes for translations are listed next.-->
<TransformAxis name="translation1">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>1 0 0</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>0.99999397216424735</scale>
</MultiplierFunction>
</TransformAxis>
<TransformAxis name="translation2">
<!--Names of the coordinates that serve as the independent variables of the transform function.-->
<coordinates></coordinates>
<!--Rotation or translation axis for the transform.-->
<axis>0 1 0</axis>
<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->
<MultiplierFunction name="function">
<function>
<Constant>
<value>0</value>
</Constant>
</function>
<scale>1.0000402623835893</scale>
</MultiplierFunction>
</TransformAxis>
```

<TransformAxis name="translation3">

<!--Names of the coordinates that serve as the independent variables of the transform function.-->

<coordinates></coordinates>

<!--Rotation or translation axis for the transform.-->

<axis>0 0 1</axis>

<!--Transform function of the generalized coordinates used to represent the amount of displacement along a specified axis.-->

<MultiplierFunction name="function">

<function>

<Constant>

<value>0</value>

</Constant>

</function>

<scale>0.99964647686094243</scale>

</MultiplierFunction>

</TransformAxis>

</SpatialTransform>

</CustomJoint>

</objects>

<groups />

</JointSet>

<!--Controllers that provide the control inputs for Actuators.-->

<ControllerSet name="controllerset">

<objects />

<groups />

</ControllerSet>

<!--Constraints in the model.-->

<ConstraintSet name="constraintset">

<objects>

<PointConstraint name="AC">

<!--Flag indicating whether the constraint is enforced or not.Enforced means that the constraint is active in subsequent dynamics realizations. NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant the constraint was not being enforced. Thus, if 'isDisabled' is'true', then 'isEnforced' is false.-->

<isEnforced>true</isEnforced>

<!--Path to a Component that satisfies the Socket 'body_1' of type PhysicalFrame (description: A frame fixed to the first body participating in the constraint.).-->

<socket_body_1>/bodyset/clavicle</socket_body_1>

<!--Path to a Component that satisfies the Socket 'body_2' of type PhysicalFrame (description: A frame fixed to the second body participating in the constraint.).-->

<socket_body_2>/bodyset/scapula</socket_body_2>

<!--Location of the point in first body specified in body_1 reference frame.-->

<location_body_1>-0.029239999999999999 0.020240000000000001 0.12005</location_body_1>

<!--Location of the point in second body specified in body_2 reference frame.-->

<location_body_2>-0.01357 0.00011 -0.01523</location_body_2>

</PointConstraint>

<WeldConstraint name="antebrazo_exoesqueleto">

<!--List of components that this component owns and serializes.-->

<components />

<!--Flag indicating whether the constraint is enforced or not.Enforced means that the constraint is active in subsequent dynamics realizations. NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant the constraint was not being enforced. Thus, if 'isDisabled' is'true', then 'isEnforced' is false.-->

<isEnforced>true</isEnforced>

<!--Path to a Component that satisfies the Socket 'frame1' of type F (description: The first frame participating in this linker.).-->

<socket_frame1>antebrazo_exoesqueleto1_offset</socket_frame1>

<!--Path to a Component that satisfies the Socket 'frame2' of type F (description: The second frame participating in this linker.).-->

<socket_frame2>antebrazo_exoesqueleto2_offset</socket_frame2>

<!--Frames created/added to satisfy this component's connections.-->

<frames>

<PhysicalOffsetFrame name="antebrazo_exoesqueleto1_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/antebrazo_exoesqueleto1</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 -0.059999999999999998 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

<PhysicalOffsetFrame name="antebrazo_exoesqueleto2_offset">

<!--The geometry used to display the axes of this Frame.-->

<FrameGeometry name="frame_geometry">

<!--Path to a Component that satisfies the Socket 'frame' of type Frame.-->

<socket_frame>..</socket_frame>

<!--Scale factors in X, Y, Z directions respectively.-->

<scale_factors>0.20000000000000001 0.20000000000000001 0.20000000000000001</scale_factors>

</FrameGeometry>

<!--Path to a Component that satisfies the Socket 'parent' of type C (description: The parent frame to this frame.).-->

<socket_parent>/bodyset/antebrazo_exoesqueleto2</socket_parent>

<!--Translational offset (in meters) of this frame's origin from the parent frame's origin, expressed in the parent frame.-->

<translation>0 0.059999999999999998 0</translation>

<!--Orientation offset (in radians) of this frame in its parent frame, expressed as a frame-fixed x-y-z rotation sequence.-->

<orientation>0 0 0</orientation>

</PhysicalOffsetFrame>

</frames>

</WeldConstraint>

</objects>

<groups />

</ConstraintSet>

<!--Forces in the model (includes Actuators).-->

<ForceSet name="forceset">

<objects>

<Millard2012EquilibriumMuscle name="TrapeziusScapula_M">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="TrapeziusScapula5-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

```xml
<socket_parent_frame>/bodyset/thorax</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.08448942449136522 -0.00037054196581871814 -0.0017665958857776051</location>
</PathPoint>
<PathPoint name="TrapeziusScapula5-P2">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/scapula</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.058246463874573863 -0.001956694915776408 -0.0363195489232871</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
```

```
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>470.39999999999998</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.083199999999999996</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.032000000000000001</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="TrapeziusScapula_M_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="TrapeziusScapula_M_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
```

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="TrapeziusScapula_M_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="TrapeziusScapula_M_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="TrapeziusScapula_S">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="TrapeziusScapula2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.07484987069505504 0.04504422710665909 -0.0036263178646243565</location>

</PathPoint>

<PathPoint name="TrapeziusScapula2-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.051749722518575597 0.0068902816687202518 -0.024441384521551281</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

```
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>1043</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.11269999999999999</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.027</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
```

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="TrapeziusScapula_S_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="TrapeziusScapula_S_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="TrapeziusScapula_S_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="TrapeziusScapula_S_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="TrapeziusScapula_I">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

```
<!--The set of points defining the path of the actuator.-->
<GeometryPath name="geometrypath">
<!--The set of points defining the path-->
<PathPointSet>
<objects>
<PathPoint name="TrapeziusScapula10-P1">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/thorax</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.10161967550752138 -0.13457125106857348 0.0011499717550883652</location>
</PathPoint>
<PathPoint name="TrapeziusScapula10-P2">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/scapula</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.075719513003881009 -0.0094503788490662675 -0.073863861111992529</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
```

```
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>414.39999999999998</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.12640000000000001</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.035000000000000003</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="TrapeziusScapula_I_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="TrapeziusScapula_I_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
```

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="TrapeziusScapula_I_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="TrapeziusScapula_I_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="TrapeziusClavicle_S">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="TrapeziusClavicle1-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.062265239358868185 0.12030948260532234 0.0024457573928283226</location>

</PathPoint>

<PathPoint name="TrapeziusClavicle1-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/clavicle</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

```xml
<location>-0.030106449153495463 0.024900173680646273 0.090973072017417431</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>201.59999999999999</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.11269999999999999</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.027</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
```

<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="TrapeziusClavicle_S_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="TrapeziusClavicle_S_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
</ForceVelocityCurve>
<!--Passive-force-length curve.-->
<FiberForceLengthCurve name="TrapeziusClavicle_S_FiberForceLengthCurve">
<!--All properties of this object have their default values.-->
</FiberForceLengthCurve>
<!--Tendon-force-length curve.-->
<TendonForceLengthCurve name="TrapeziusClavicle_S_TendonForceLengthCurve">
<!--All properties of this object have their default values.-->
</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="SerratusAnterior_I">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="SerratusAnterior4-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.11354979080726471 -0.11220745121130185 -0.085518250017048519</location>

</PathPoint>

<PathPoint name="SerratusAnterior4-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.00053650240154909108 -0.17049632001723969 0.14824508598636632</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Thorax</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>midpoint</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>430</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.15870000000000001</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="SerratusAnterior_I_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->

<ForceVelocityCurve name="SerratusAnterior_I_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="SerratusAnterior_I_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="SerratusAnterior_I_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="SerratusAnterior_M">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="SerratusAnterior7-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

```
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.10311842013386459 -0.083083588833203964 -0.098596649512748102</location>
</PathPoint>
<PathPoint name="SerratusAnterior7-P2">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/thorax</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>0.01669205437318563 -0.076276553111987985 0.11473277572966468</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
```

<max_isometric_force>508</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.15379999999999999</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.012</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="SerratusAnterior_M_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="SerratusAnterior_M_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="SerratusAnterior_M_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="SerratusAnterior_M_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="SerratusAnterior_S">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="SerratusAnterior12-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.048509702150793158 0.0090003646181095135 -0.084640018475043113</location>

</PathPoint>

<PathPoint name="SerratusAnterior12-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.028461266034923102 -0.014528397201316316 0.099347168830618449</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

```xml
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>387.80000000000001</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.094500000000000001</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
```

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="SerratusAnterior_S_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="SerratusAnterior_S_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

*<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>*

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="SerratusAnterior_S_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="SerratusAnterior_S_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Rhomboideus_S">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>false</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

```xml
<!--The set of points defining the path-->
<PathPointSet>
<objects>
<PathPoint name="Rhomboideus1-P1">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/thorax</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.073299729881960707 0.040180265023024381 -0.0020399444179517892</location>
</PathPoint>
<PathPoint name="Rhomboideus1-P2">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/scapula</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.082669471197028249 -0.022910906289267983 -0.10173372308327058</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
```

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>200.19999999999999</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.098599999999999993</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.014999999999999999</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="Rhomboideus_S_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="Rhomboideus_S_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="Rhomboideus_S_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="Rhomboideus_S_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Rhomboideus_I">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>false</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Rhomboideus4-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.090659686945963935 -0.0092700594962635344 -0.00050998760442502039</location>

</PathPoint>

<PathPoint name="Rhomboideus4-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.10625594744208829 -0.10671263197706551 -0.092724435677473432</location>

```xml
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>407.39999999999998</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.1152</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.028000000000000001</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
```

<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->

<ignore_tendon_compliance>false</ignore_tendon_compliance>

<!--Activation time constant (in seconds).-->

<activation_time_constant>0.01</activation_time_constant>

<!--Deactivation time constant (in seconds).-->

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="Rhomboideus_I_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="Rhomboideus_I_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="Rhomboideus_I_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="Rhomboideus_I_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="LevatorScapulae">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="LevatorScapulae1-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.034809816495013239 0.12106052307344532 0.031079296126327739</location>

</PathPoint>

<PathPoint name="LevatorScapulae1-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.078009598081266673 -0.0080803264127147275 -0.102883313154468</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Thorax</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

```xml
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>280</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.1578</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.019</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="LevatorScapulae_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="LevatorScapulae_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
```

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="LevatorScapulae_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="LevatorScapulae_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Coracobrachialis">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Coracobrachialis2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.0082682171668854807 -0.042434755999553653 -0.028091382979780092</location>

</PathPoint>

```xml
<PathPoint name="Coracobrachialis2-P2">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/humerus</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.0023100065501454803 -0.14500057286047488 -0.0093299486575685996</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects />
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>648.20000000000005</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.0683</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.104</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
```

<activation_time_constant>0.01</activation_time_constant>

<!--Deactivation time constant (in seconds).-->

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="Coracobrachialis_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="Coracobrachialis_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

*<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>*

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="Coracobrachialis_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="Coracobrachialis_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="DeltoideusClavicle_A">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then

'appliesForce` is false.-->

<appliesForce>false</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="DeltoideusClavicle2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/clavicle</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.022762490803495135 0.019491266287305724 0.084093007753211754</location>

</PathPoint>

<PathPoint name="DeltoideusClavicle2-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.019562226742916928 -0.0065870855556006396 0.01039769231874158</location>

</PathPoint>

<PathPoint name="DeltoideusClavicle2-P3">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.0066436613177381703 -0.10980522018450981 0.0011474186050816253</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>deltsca</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

```
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>707.70000000000005</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.093999999999999986</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.087999999999999981</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0.087266460000000004</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="DeltoideusClavicle_A_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
```

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="DeltoideusClavicle_A_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="DeltoideusClavicle_A_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="DeltoideusClavicle_A_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="DeltoideusScapula_P">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>false</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="DeltoideusScapulaPost2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The

frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.057737544968075358 -0.0069115913955361454 -0.027229490319444081</location>

</PathPoint>

<PathPoint name="DeltoideusScapulaPost2-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.055002743369330122 -0.032365885079364429 0.0073526981238954681</location>

</PathPoint>

<PathPoint name="DeltoideusScapulaPost2-P3">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.0047659122508031749 -0.086162511515571069 0.0062390724151510932</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>deltsca</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

```
<visible>true</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>1324.4000000000001</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.094900000000000026</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.076000000000000012</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0.087266460000000004</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="DeltoideusScapula_P_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="DeltoideusScapula_P_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
```

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="DeltoideusScapula_P_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="DeltoideusScapula_P_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="DeltoideusScapula_M">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>false</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="DeltoideusScapulaLat10-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.016020195155322037 0.0020340339243142404 0.0077978793780755194</location>

</PathPoint>

<PathPoint name="DeltoideusScapulaLat10-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

```
<location>-0.005137830872043541 -0.0067412235795988491 0.031028935486928554</location>
</PathPoint>
<PathPoint name="DeltoideusScapulaLat10-P3">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/humerus</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.0043127648892225725 -0.10045486081870107 0.0038455353249774223</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>deltsca</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>2597.8000000000002</max_isometric_force>
```

<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.074800000000000005</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.064000000000000001</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0.087266460000000004</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="DeltoideusScapula_M_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="DeltoideusScapula_M_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier>
</ForceVelocityCurve>
<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="DeltoideusScapula_M_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="DeltoideusScapula_M_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="LatissimusDorsi_S">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="LatissimusDorsi1-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.1008796560238172 -0.11663128441390257 0.0017099582619030595</location>

</PathPoint>

<PathPoint name="LatissimusDorsi1-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.11550004095862107 -0.11639127148134662 0.028799291779640745</location>

</PathPoint>

<MovingPathPoint name="LatissimusDorsi1-P3">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'x_coordinate' of type Coordinate (description: The x_location function is a function of this coordinate's value.).-->

<socket_x_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_x_coordinate>

<!--Path to a Component that satisfies the Socket 'y_coordinate' of type Coordinate (description: The y_location function is a function of this coordinate's value.).-->

<socket_y_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_y_coordinate>

<!--Path to a Component that satisfies the Socket 'z_coordinate' of type Coordinate (description: The z_location function is a function of this coordinate's value.).-->

<socket_z_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_z_coordinate>

<!--Function defining the x component of the point's location expressed in the Frame of the Point.-->

<x_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.122564 -0.123775</y>

</SimmSpline>

</function>

<scale>0.99999644921000319</scale>

</MultiplierFunction>

</x_location>

<!--Function defining the y component of the point's location expressed in the Frame of the Point.-->

<y_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.098615 -0.106769</y>

</SimmSpline>

</function>

<scale>1.000006693199808</scale>

</MultiplierFunction>

</y_location>

<!--Function defining the z component of the point's location expressed in the Frame of the Point.-->

<z_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> 0.101016 0.114972</y>

</SimmSpline>

</function>

<scale>0.99997632556685589</scale>

</MultiplierFunction>

</z_location>

</MovingPathPoint>

<MovingPathPoint name="LatissimusDorsi1-P4">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'x_coordinate' of type Coordinate (description: The x_location function is a function of this coordinate's value.).-->

<socket_x_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_x_coordinate>

<!--Path to a Component that satisfies the Socket 'y_coordinate' of type Coordinate (description: The y_location function is a function of this coordinate's value.).-->

<socket_y_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_y_coordinate>

<!--Path to a Component that satisfies the Socket 'z_coordinate' of type Coordinate (description: The z_location function is a function of this coordinate's value.).-->

<socket_z_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_z_coordinate>

<!--Function defining the x component of the point's location expressed in the Frame of the Point.-->

<x_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.09279 -0.094357</y>

</SimmSpline>

</function>

<scale>0.99999644921000319</scale>

</MultiplierFunction>

</x_location>

<!--Function defining the y component of the point's location expressed in the Frame of the Point.-->

<y_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.08329 -0.084707</y>

</SimmSpline>

</function>

<scale>1.000006693199808</scale>

</MultiplierFunction>
</y_location>
<!--Function defining the z component of the point's location expressed in the Frame of the Point.-->
<z_location>
<MultiplierFunction>
<function>
<SimmSpline>
<x> 0 3.14159</x>
<y> 0.13581 0.144557</y>
</SimmSpline>
</function>
<scale>0.99997632556685589</scale>
</MultiplierFunction>
</z_location>
</MovingPathPoint>
<PathPoint name="LatissimusDorsi1-P5">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/humerus</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>0.0058799957641284184 -0.019040131776404189 -0.0034499858058475164</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />

```
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>201.59999999999999</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.2109</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.081000000000000003</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="LatissimusDorsi_S_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="LatissimusDorsi_S_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
```

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

*<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>*

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="LatissimusDorsi_S_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="LatissimusDorsi_S_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="LatissimusDorsi_M">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="LatissimusDorsi3-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.090819691158656735 -0.21745171689568252 -0.0015599645546081824</location>

</PathPoint>

<PathPoint name="LatissimusDorsi3-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.10767983506326086 -0.20544206978291463 0.026229399594655183</location>

</PathPoint>

<MovingPathPoint name="LatissimusDorsi3-P3">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'x_coordinate' of type Coordinate (description: The x_location function is a function of this coordinate's value.).-->

<socket_x_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_x_coordinate>

<!--Path to a Component that satisfies the Socket 'y_coordinate' of type Coordinate (description: The y_location function is a function of this coordinate's value.).-->

<socket_y_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_y_coordinate>

<!--Path to a Component that satisfies the Socket 'z_coordinate' of type Coordinate (description: The z_location function is a function of this coordinate's value.).-->

<socket_z_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_z_coordinate>

<!--Function defining the x component of the point's location expressed in the Frame of the Point.-->

<x_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.103889 -0.101761</y>

</SimmSpline>

</function>

<scale>0.99999644921000319</scale>

</MultiplierFunction>

</x_location>

<!--Function defining the y component of the point's location expressed in the Frame of the Point.-->

<y_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.147585 -0.149604</y>

</SimmSpline>

</function>

<scale>1.000006693199808</scale>

</MultiplierFunction>

</y_location>

<!--Function defining the z component of the point's location expressed in the Frame of the Point.-->

<z_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> 0.113647 0.122566</y>

</SimmSpline>

</function>

<scale>0.99997632556685589</scale>

</MultiplierFunction>

</z_location>

</MovingPathPoint>

<MovingPathPoint name="LatissimusDorsi3-P4">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'x_coordinate' of type Coordinate (description: The x_location function is a function of this coordinate's value.).-->

<socket_x_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_x_coordinate>

<!--Path to a Component that satisfies the Socket 'y_coordinate' of type Coordinate (description: The y_location function is a function of this coordinate's value.).-->

<socket_y_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_y_coordinate>

<!--Path to a Component that satisfies the Socket 'z_coordinate' of type Coordinate (description: The z_location function is a function of this coordinate's value.).-->

<socket_z_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_z_coordinate>

<!--Function defining the x component of the point's location expressed in the Frame of the Point.-->

<x_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.076222 -0.077789</y>

</SimmSpline>

</function>

<scale>0.99999644921000319</scale>
</MultiplierFunction>
</x_location>
<!--Function defining the y component of the point's location expressed in the Frame of the Point.-->
<y_location>
<MultiplierFunction>
<function>
<SimmSpline>
<x> 0 3.14159</x>
<y> -0.110351 -0.111768</y>
</SimmSpline>
</function>
<scale>1.000006693199808</scale>
</MultiplierFunction>
</y_location>
<!--Function defining the z component of the point's location expressed in the Frame of the Point.-->
<z_location>
<MultiplierFunction>
<function>
<SimmSpline>
<x> 0 3.14159</x>
<y> 0.140024 0.148771</y>
</SimmSpline>
</function>
<scale>0.99997632556685589</scale>
</MultiplierFunction>
</z_location>
</MovingPathPoint>
<PathPoint name="LatissimusDorsi3-P5">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/humerus</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>0.0057799963463681624 -0.04100027577871479 -0.0009999977843013195</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Thorax</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>false</visible>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>

</Appearance>

</GeometryPath>

<!--The maximum force this actuator can produce.-->

<optimal_force>1</optimal_force>

<!--Maximum isometric force that the fibers can generate-->

<max_isometric_force>315</max_isometric_force>

<!--Optimal length of the muscle fibers-->

<optimal_fiber_length>0.2656</optimal_fiber_length>

<!--Resting length of the tendon-->

<tendon_slack_length>0.094500000000000001</tendon_slack_length>

<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->

<pennation_angle_at_optimal>0</pennation_angle_at_optimal>

<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->

<max_contraction_velocity>10</max_contraction_velocity>

<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->

<ignore_tendon_compliance>true</ignore_tendon_compliance>

<!--Activation time constant (in seconds).-->

<activation_time_constant>0.01</activation_time_constant>

```
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="LatissimusDorsi_M_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="LatissimusDorsi_M_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
</ForceVelocityCurve>
<!--Passive-force-length curve.-->
<FiberForceLengthCurve name="LatissimusDorsi_M_FiberForceLengthCurve">
<!--All properties of this object have their default values.-->
</FiberForceLengthCurve>
<!--Tendon-force-length curve.-->
<TendonForceLengthCurve name="LatissimusDorsi_M_TendonForceLengthCurve">
<!--All properties of this object have their default values.-->
</TendonForceLengthCurve>
</Millard2012EquilibriumMuscle>
<Millard2012EquilibriumMuscle name="LatissimusDorsi_I">
<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem
otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the
'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then
'appliesForce` is false.-->
```

<appliesForce>false</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="LatissimusDorsi6-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.062738100000000005 -0.25480799999999998 0.090611700000000003</location>

</PathPoint>

<PathPoint name="LatissimusDorsi6-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.067190261176049607 -0.1861642427323813 0.14558713504239773</location>

</PathPoint>

<MovingPathPoint name="LatissimusDorsi6-P3">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--Path to a Component that satisfies the Socket 'x_coordinate' of type Coordinate (description: The x_location function is a function of this coordinate's value.).-->

<socket_x_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_x_coordinate>

<!--Path to a Component that satisfies the Socket 'y_coordinate' of type Coordinate (description: The y_location function is a function of this coordinate's value.).-->

<socket_y_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_y_coordinate>

<!--Path to a Component that satisfies the Socket 'z_coordinate' of type Coordinate (description: The z_location function is a function of this coordinate's value.).-->

<socket_z_coordinate>/jointset/GlenoHumeral/shoulder_elv</socket_z_coordinate>

<!--Function defining the x component of the point's location expressed in the Frame of the Point.-->

<x_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.040258 -0.031577</y>

</SimmSpline>

</function>

<scale>0.99997012066675772</scale>

</MultiplierFunction>

</x_location>

<!--Function defining the y component of the point's location expressed in the Frame of the Point.-->

<y_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> -0.116638 -0.109479</y>

</SimmSpline>

</function>

<scale>0.9999528100929127</scale>

</MultiplierFunction>

</y_location>

<!--Function defining the z component of the point's location expressed in the Frame of the Point.-->

<z_location>

<MultiplierFunction>

<function>

<SimmSpline>

<x> 0 3.14159</x>

<y> 0.146215 0.149666</y>

</SimmSpline>

</function>

<scale>1.0000182777011157</scale>

</MultiplierFunction>

</z_location>

</MovingPathPoint>

<PathPoint name="LatissimusDorsi6-P4">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.0042200299999999996 -0.049998399999999998 -0.000179999</location>

</PathPoint>

</objects>

```xml
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>270.19999999999999</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.30620000000000003</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.062100000000000002</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
```

```
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="LatissimusDorsi_I_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="LatissimusDorsi_I_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
</ForceVelocityCurve>
<!--Passive-force-length curve.-->
<FiberForceLengthCurve name="LatissimusDorsi_I_FiberForceLengthCurve">
<!--All properties of this object have their default values.-->
</FiberForceLengthCurve>
<!--Tendon-force-length curve.-->
<TendonForceLengthCurve name="LatissimusDorsi_I_TendonForceLengthCurve">
<!--All properties of this object have their default values.-->
</TendonForceLengthCurve>
</Millard2012EquilibriumMuscle>
<Millard2012EquilibriumMuscle name="PectoralisMajorClavicle_S">
<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem
```

otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="PectoralisMajorClavicle1-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/clavicle</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.0045991607738268781 0.0068185965088977942 0.04738913830637137</location>

</PathPoint>

<PathPoint name="PectoralisMajorClavicle1-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.010075141338181376 -0.042144919771239353 -0.0026007368382806521</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Thorax</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>408.80000000000001</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.1091</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.014</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="PectoralisMajorClavicle_S_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="PectoralisMajorClavicle_S_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="PectoralisMajorClavicle_S_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="PectoralisMajorClavicle_S_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="PectoralisMajorThorax_I">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="PectoralisMajorThorax2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.053209800941519628 -0.1156612321448223 0.024739462102192737</location>

</PathPoint>

<PathPoint name="PectoralisMajorThorax2-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.0099999417760256639 -0.030000153006940828 0</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Thorax</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>false</visible>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>

</Appearance>

</GeometryPath>

<!--The maximum force this actuator can produce.-->

<optimal_force>1</optimal_force>

<!--Maximum isometric force that the fibers can generate-->

<max_isometric_force>571.20000000000005</max_isometric_force>

<!--Optimal length of the muscle fibers-->

<optimal_fiber_length>0.183</optimal_fiber_length>

<!--Resting length of the tendon-->

```
<tendon_slack_length>0.042999999999999997</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="PectoralisMajorThorax_I_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="PectoralisMajorThorax_I_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
</ForceVelocityCurve>
<!--Passive-force-length curve.-->
<FiberForceLengthCurve name="PectoralisMajorThorax_I_FiberForceLengthCurve">
<!--All properties of this object have their default values.-->
</FiberForceLengthCurve>
```

```
<!--Tendon-force-length curve.-->
<TendonForceLengthCurve name="PectoralisMajorThorax_I_TendonForceLengthCurve">
<!--All properties of this object have their default values.-->
</TendonForceLengthCurve>
</Millard2012EquilibriumMuscle>
<Millard2012EquilibriumMuscle name="PectoralisMajorThorax_M">
<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->
<appliesForce>true</appliesForce>
<!--The set of points defining the path of the actuator.-->
<GeometryPath name="geometrypath">
<!--The set of points defining the path-->
<PathPointSet>
<objects>
<PathPoint name="PectoralisMajorThorax5-P1">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/thorax</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>0.019539914459309596 -0.038002047654690653 0.012999454632222693</location>
</PathPoint>
<PathPoint name="PectoralisMajorThorax5-P2">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/humerus</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>0.010185240697135417 -0.026204007108265964 -0.0031506861697064947</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
```

```
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>683.20000000000005</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.14999999999999999</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.025999999999999999</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="PectoralisMajorThorax_M_ActiveForceLengthCurve">
```

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="PectoralisMajorThorax_M_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="PectoralisMajorThorax_M_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="PectoralisMajorThorax_M_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="TeresMajor">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="TeresMajor1-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.10440887882290427 -0.11741549476967501 -0.072094492045888692</location>

</PathPoint>

<PathPoint name="TeresMajor1-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.0043200048470684151 -0.039000198909023079 -0.0016999979333101767</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Thorax</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

<PathWrap name="pathwrap_0">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Tmajscapula</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>

```xml
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>851.20000000000005</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.14099999999999999</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.0060000000000000001</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="TeresMajor_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="TeresMajor_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
```

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

*<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>*

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="TeresMajor_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="TeresMajor_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Infraspinatus_I">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>false</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Infraspinatus3-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.099069480462564738 -0.080853263486170973 -0.081261223309262887</location>

</PathPoint>

<PathPoint name="Infraspinatus3-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The

```
frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/humerus</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>-0.017162031747030893 -0.0057299955321240209 0.024807138325052238</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Infra</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
<PathWrap name="pathwrap_0">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>hheadInfra</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
```

<!--The maximum force this actuator can produce.-->

<optimal_force>1</optimal_force>

<!--Maximum isometric force that the fibers can generate-->

<max_isometric_force>1037.4000000000001</max_isometric_force>

<!--Optimal length of the muscle fibers-->

<optimal_fiber_length>0.067699999999999996</optimal_fiber_length>

<!--Resting length of the tendon-->

<tendon_slack_length>0.084000000000000005</tendon_slack_length>

<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->

<pennation_angle_at_optimal>0</pennation_angle_at_optimal>

<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->

<max_contraction_velocity>10</max_contraction_velocity>

<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->

<ignore_tendon_compliance>false</ignore_tendon_compliance>

<!--Activation time constant (in seconds).-->

<activation_time_constant>0.01</activation_time_constant>

<!--Deactivation time constant (in seconds).-->

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="Infraspinatus_I_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="Infraspinatus_I_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="Infraspinatus_I_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="Infraspinatus_I_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Infraspinatus_S">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>false</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Infraspinatus5-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.083429499432317811 -0.032271286292091625 -0.086879319969381641</location>

</PathPoint>

<PathPoint name="Infraspinatus5-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.014333616543501905 0.0027190845035771852 0.022203973006599511</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

```xml
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Infra</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
<PathWrap name="pathwrap_0">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>hheadInfra</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>true</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>967.39999999999998</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.069800000000000001</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.050000000000000003</tendon_slack_length>
```

```
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="Infraspinatus_S_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="Infraspinatus_S_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
</ForceVelocityCurve>
<!--Passive-force-length curve.-->
<FiberForceLengthCurve name="Infraspinatus_S_FiberForceLengthCurve">
<!--All properties of this object have their default values.-->
</FiberForceLengthCurve>
<!--Tendon-force-length curve.-->
```

<TendonForceLengthCurve name="Infraspinatus_S_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="PectoralisMinor">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="PectoralisMinor3-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.047729456651525642 -0.10400960432311702 0.091178074794605088</location>

</PathPoint>

<PathPoint name="PectoralisMinor3-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.01087460399498133 -0.035041350780721695 -0.022940819603399599</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Thorax</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

```
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>429.80000000000001</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.1183</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.032000000000000001</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>true</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="PectoralisMinor_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
```

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="PectoralisMinor_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="PectoralisMinor_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="PectoralisMinor_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="TeresMinor">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="TeresMinor2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.084369434351197634 -0.066052653225416508 -0.042285021713615541</location>

</PathPoint>

<PathPoint name="TeresMinor2-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.015870007597970491 -0.0093600497382424104 0.010849886809777699</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>hheadsupraTminor</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>

</Appearance>

</GeometryPath>

<!--The maximum force this actuator can produce.-->

<optimal_force>1</optimal_force>

<!--Maximum isometric force that the fibers can generate-->

<max_isometric_force>695.79999999999995</max_isometric_force>

<!--Optimal length of the muscle fibers-->

<optimal_fiber_length>0.055</optimal_fiber_length>

<!--Resting length of the tendon-->

<tendon_slack_length>0.050999999999999997</tendon_slack_length>

<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->

<pennation_angle_at_optimal>0</pennation_angle_at_optimal>

<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->

<max_contraction_velocity>10</max_contraction_velocity>

<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->

<ignore_tendon_compliance>false</ignore_tendon_compliance>

<!--Activation time constant (in seconds).-->

<activation_time_constant>0.01</activation_time_constant>

<!--Deactivation time constant (in seconds).-->

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="TeresMinor_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="TeresMinor_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="TeresMinor_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="TeresMinor_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Subscapularis_S">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Subscapularis3-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.078789527055875025 -0.031271199112899918 -0.090268111570563608</location>

</PathPoint>

<PathPoint name="Subscapularis3-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.018000095195681718 -0.00023000183974550295 -0.011999885411722819</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

```
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Subscapuralis</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
<PathWrap name="pathwrap_0">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>hhead</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>540.39999999999998</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.067599999999999993</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.058999999999999997</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
```

<pennation_angle_at_optimal>0.087266460000000004</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="Subscapularis_S_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="Subscapularis_S_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
</ForceVelocityCurve>
<!--Passive-force-length curve.-->
<FiberForceLengthCurve name="Subscapularis_S_FiberForceLengthCurve">
<!--All properties of this object have their default values.-->
</FiberForceLengthCurve>
<!--Tendon-force-length curve.-->
<TendonForceLengthCurve name="Subscapularis_S_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Subscapularis_M">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Subscapularis4-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.083569504633555378 -0.054912196042631707 -0.074743547427343598</location>

</PathPoint>

<PathPoint name="Subscapularis4-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.016391104563919189 0.00095780581170388059 -0.019435076372750954</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Subscapuralis</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

```
<range>-1 -1</range>
</PathWrap>
<PathWrap name="pathwrap_0">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>hhead</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>609</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.074399999999999994</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.055</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0.087266460000000004</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
```

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="Subscapularis_M_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="Subscapularis_M_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="Subscapularis_M_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="Subscapularis_M_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Subscapularis_I">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Subscapularis9-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.095069431859492709 -0.097653948106590768 -0.085369758259368406</location>

</PathPoint>

<PathPoint name="Subscapularis9-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.010983836047368828 -0.0034073409555911483 -0.019525776262486971</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>Subscapuralis</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

<PathWrap name="pathwrap_0">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>hhead</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>false</visible>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>

</Appearance>

</GeometryPath>

<!--The maximum force this actuator can produce.-->

<optimal_force>1</optimal_force>

<!--Maximum isometric force that the fibers can generate-->

<max_isometric_force>854</max_isometric_force>

<!--Optimal length of the muscle fibers-->

<optimal_fiber_length>0.072099999999999997</optimal_fiber_length>

<!--Resting length of the tendon-->

<tendon_slack_length>0.058999999999999997</tendon_slack_length>

<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->

<pennation_angle_at_optimal>0</pennation_angle_at_optimal>

<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->

<max_contraction_velocity>10</max_contraction_velocity>

<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->

<ignore_tendon_compliance>false</ignore_tendon_compliance>

<!--Activation time constant (in seconds).-->

<activation_time_constant>0.01</activation_time_constant>

<!--Deactivation time constant (in seconds).-->

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="Subscapularis_I_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="Subscapularis_I_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="Subscapularis_I_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="Subscapularis_I_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Supraspinatus_P">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Supraspinatus2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The

frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.060529648695612823 -0.0013800523081562756 -0.049452565861000421</location>

</PathPoint>

<PathPoint name="Supraspinatus2-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.010169740787382579 0.0095677577211598953 0.020050375624730808</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>hheadsupraTminor</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>false</visible>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>

</Appearance>

</GeometryPath>

<!--The maximum force this actuator can produce.-->

<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>326.19999999999999</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.0591</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.025000000000000001</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="Supraspinatus_P_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="Supraspinatus_P_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier

>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="Supraspinatus_P_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="Supraspinatus_P_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="Supraspinatus_A">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="Supraspinatus4-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.048993120109923384 0.0013557026203117424 -0.069247307314771744</location>

</PathPoint>

<PathPoint name="Supraspinatus4-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.017779996477191390 0.017640077967619992 0.0096499482685446329</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

```
<PathWrapSet>
<objects>
<PathWrap name="pathwrap">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>Thorax</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
<PathWrap name="pathwrap_0">
<!--A WrapObject that this PathWrap interacts with.-->
<wrap_object>hheadsupraTminor</wrap_object>
<!--The wrapping method used to solve the path around the wrap object.-->
<method>hybrid</method>
<!--The range of indices to use to compute the path over the wrap object.-->
<range>-1 -1</range>
</PathWrap>
</objects>
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--Flag indicating whether the associated Geometry is visible or hidden.-->
<visible>false</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>543.20000000000005</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.055399999999999998</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.031</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
```

<pennation_angle_at_optimal>0</pennation_angle_at_optimal>

<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->

<max_contraction_velocity>10</max_contraction_velocity>

<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->

<ignore_tendon_compliance>false</ignore_tendon_compliance>

<!--Activation time constant (in seconds).-->

<activation_time_constant>0.01</activation_time_constant>

<!--Deactivation time constant (in seconds).-->

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="Supraspinatus_A_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="Supraspinatus_A_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="Supraspinatus_A_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="Supraspinatus_A_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="TRIlong">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="TRIlong2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/ulna</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.019462 -0.029108700000000001 0.0086597700000000007</location>

</PathPoint>

<PathPoint name="TRIlong2-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.042099764019562176 -0.051904634598639157 -0.012920792614134868</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects />

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

```
<visible>true</visible>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>
</Appearance>
</GeometryPath>
<!--The maximum force this actuator can produce.-->
<optimal_force>1</optimal_force>
<!--Maximum isometric force that the fibers can generate-->
<max_isometric_force>1580.5999999999999</max_isometric_force>
<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.0969</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.2412</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0.17453299999999999</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="TRIlong_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="TRIlong_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
```

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier>

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="TRIlong_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="TRIlong_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="BIC_long">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="BIC_long-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.022587239123084576 -0.01632949066098063 -0.01813583299885644</location>

</PathPoint>

<PathPoint name="BIC_long-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.011830631116962682 0.02814188158731026 0.020038375639319206</location>

</PathPoint>

<PathPoint name="BIC_long-P3">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/radius</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>-0.00088464999999999995 -0.043994400000000003 -0.0094551600000000006</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects>

<PathWrap name="pathwrap">

<!--A WrapObject that this PathWrap interacts with.-->

<wrap_object>hhead</wrap_object>

<!--The wrapping method used to solve the path around the wrap object.-->

<method>hybrid</method>

<!--The range of indices to use to compute the path over the wrap object.-->

<range>-1 -1</range>

</PathWrap>

</objects>

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>

</Appearance>

</GeometryPath>

<!--The maximum force this actuator can produce.-->

<optimal_force>1</optimal_force>

<!--Maximum isometric force that the fibers can generate-->

<max_isometric_force>485.80000000000001</max_isometric_force>

<!--Optimal length of the muscle fibers-->
<optimal_fiber_length>0.1421</optimal_fiber_length>
<!--Resting length of the tendon-->
<tendon_slack_length>0.25679999999999997</tendon_slack_length>
<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->
<pennation_angle_at_optimal>0</pennation_angle_at_optimal>
<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->
<max_contraction_velocity>10</max_contraction_velocity>
<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->
<ignore_tendon_compliance>false</ignore_tendon_compliance>
<!--Activation time constant (in seconds).-->
<activation_time_constant>0.01</activation_time_constant>
<!--Deactivation time constant (in seconds).-->
<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>
<!--Activation lower bound.-->
<minimum_activation>0.01</minimum_activation>
<!--Active-force-length curve.-->
<ActiveForceLengthCurve name="BIC_long_ActiveForceLengthCurve">
<!--Minimum value of the active-force-length curve-->
<minimum_value>0</minimum_value>
</ActiveForceLengthCurve>
<!--Force-velocity curve.-->
<ForceVelocityCurve name="BIC_long_ForceVelocityCurve">
<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->
<concentric_slope_at_vmax>0</concentric_slope_at_vmax>
<!--Curve slope just before reaching concentric_slope_at_vmax-->
<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>
<!--Curve slope at isometric (normalized velocity of 0)-->
<isometric_slope>5</isometric_slope>
<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->
<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>
<!--Curve slope just before reaching eccentric_slope_at_vmax-->
<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>
<!--Curve value at the maximum normalized eccentric contraction velocity-->
<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>
</ForceVelocityCurve>
<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="BIC_long_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="BIC_long_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<Millard2012EquilibriumMuscle name="BIC_brevis">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The set of points defining the path of the actuator.-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="BIC_brevis2-P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.0095474146894784402 -0.038223609985142994 -0.024238756775958485</location>

</PathPoint>

<PathPoint name="BIC_brevis2-P2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/radius</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0.00020306999999999999 -0.044256299999999998 -0.00985953</location>

</PathPoint>

</objects>

<groups />

</PathPointSet>

<!--The wrap objects that are associated with this path-->

<PathWrapSet>

<objects />

<groups />

</PathWrapSet>

<!--Default appearance attributes for this GeometryPath-->

<Appearance>

<!--Flag indicating whether the associated Geometry is visible or hidden.-->

<visible>true</visible>

<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->

<color>0.80000000000000004 0.10000000000000001 0.10000000000000001</color>

</Appearance>

</GeometryPath>

<!--The maximum force this actuator can produce.-->

<optimal_force>1</optimal_force>

<!--Maximum isometric force that the fibers can generate-->

<max_isometric_force>693</max_isometric_force>

<!--Optimal length of the muscle fibers-->

<optimal_fiber_length>0.12640000000000001</optimal_fiber_length>

<!--Resting length of the tendon-->

<tendon_slack_length>0.21199999999999999</tendon_slack_length>

<!--Angle between tendon and fibers at optimal fiber length expressed in radians-->

<pennation_angle_at_optimal>0</pennation_angle_at_optimal>

<!--Maximum contraction velocity of the fibers, in optimal fiberlengths/second-->

<max_contraction_velocity>10</max_contraction_velocity>

<!--Compute muscle dynamics ignoring tendon compliance. Tendon is assumed to be rigid.-->

<ignore_tendon_compliance>false</ignore_tendon_compliance>

<!--Activation time constant (in seconds).-->

<activation_time_constant>0.01</activation_time_constant>

<!--Deactivation time constant (in seconds).-->

<deactivation_time_constant>0.040000000000000001</deactivation_time_constant>

<!--Activation lower bound.-->

<minimum_activation>0.01</minimum_activation>

<!--Active-force-length curve.-->

<ActiveForceLengthCurve name="BIC_brevis_ActiveForceLengthCurve">

<!--Minimum value of the active-force-length curve-->

<minimum_value>0</minimum_value>

</ActiveForceLengthCurve>

<!--Force-velocity curve.-->

<ForceVelocityCurve name="BIC_brevis_ForceVelocityCurve">

<!--Curve slope at the maximum normalized concentric (shortening) velocity (normalized velocity of -1)-->

<concentric_slope_at_vmax>0</concentric_slope_at_vmax>

<!--Curve slope just before reaching concentric_slope_at_vmax-->

<concentric_slope_near_vmax>0.25</concentric_slope_near_vmax>

<!--Curve slope at isometric (normalized velocity of 0)-->

<isometric_slope>5</isometric_slope>

<!--Curve slope at the maximum normalized eccentric (lengthening) velocity (normalized velocity of 1)-->

<eccentric_slope_at_vmax>0</eccentric_slope_at_vmax>

<!--Curve slope just before reaching eccentric_slope_at_vmax-->

<eccentric_slope_near_vmax>0.14999999999999999</eccentric_slope_near_vmax>

<!--Curve value at the maximum normalized eccentric contraction velocity-->

*<max_eccentric_velocity_force_multiplier>1.3999999999999999</max_eccentric_velocity_force_multiplier
>*

</ForceVelocityCurve>

<!--Passive-force-length curve.-->

<FiberForceLengthCurve name="BIC_brevis_FiberForceLengthCurve">

<!--All properties of this object have their default values.-->

</FiberForceLengthCurve>

<!--Tendon-force-length curve.-->

<TendonForceLengthCurve name="BIC_brevis_TendonForceLengthCurve">

<!--All properties of this object have their default values.-->

</TendonForceLengthCurve>

</Millard2012EquilibriumMuscle>

<PathSpring name="muelle">

<!--Flag indicating whether the force is applied or not. If true the forceis applied to the MultibodySystem otherwise the force is not applied.NOTE: Prior to OpenSim 4.0, this behavior was controlled by the 'isDisabled' property, where 'true' meant that force was not being applied. Thus, if 'isDisabled' is true, then 'appliesForce` is false.-->

<appliesForce>true</appliesForce>

<!--The GeometryPath defines the set of points and wrapping surfaceinteractions that form the path of action of the PathSpring-->

<GeometryPath name="geometrypath">

<!--The set of points defining the path-->

<PathPointSet>

<objects>

<PathPoint name="P1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->

<socket_parent_frame>/bodyset/brazo_exoesqueleto</socket_parent_frame>

<!--The fixed location of the path point expressed in its parent frame.-->

<location>0 -0.02 0</location>

</PathPoint>

```xml
<PathPoint name="P2">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame in which this path point is defined.).-->
<socket_parent_frame>/bodyset/antebrazo_exoesqueleto1</socket_parent_frame>
<!--The fixed location of the path point expressed in its parent frame.-->
<location>0 0 0</location>
</PathPoint>
</objects>
<groups />
</PathPointSet>
<!--The wrap objects that are associated with this path-->
<PathWrapSet>
<objects />
<groups />
</PathWrapSet>
<!--Default appearance attributes for this GeometryPath-->
<Appearance>
<!--The color, (red, green, blue), [0, 1], used to display the geometry. -->
<color>0 1 0</color>
</Appearance>
</GeometryPath>
<!--The resting length (m) of the PathSpring-->
<resting_length>0.059999999999999998</resting_length>
<!--The linear stiffness (N/m) of the PathSpring-->
<stiffness>10000</stiffness>
<!--The dissipation factor (s/m) of the PathSpring-->
<dissipation>10</dissipation>
</PathSpring>
</objects>
<groups />
</ForceSet>
<!--Markers in the model.-->
<MarkerSet name="markerset">
<objects>
<Marker name="RACR">
<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->
<socket_parent_frame>/bodyset/clavicle</socket_parent_frame>
```

<!--The fixed location of the station expressed in its parent frame.-->

<location>-0.030124751698960228 0.054237361396990424 0.12516058883262229</location>

<!--Flag (true or false) specifying whether the marker is fixed in its parent frame during the marker placement step of scaling. If false, the marker is free to move within its parent Frame to match its experimental counterpart.-->

<fixed>false</fixed>

</Marker>

<Marker name="C7">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>-0.095160771279519774 0.071160331534518395 -0.013437400733183169</location>

</Marker>

<Marker name="CLAV">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>0.034776600645245098 -0.0001928308874930007 -0.015659999347533149</location>

</Marker>

<Marker name="ESTERNON">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/thorax</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>0.084919673557353498 -0.17942376877219623 -0.0063977947329613427</location>

</Marker>

<Marker name="RASH">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>0.0094790856238866628 -0.011292718803957547 0.025373221623911812</location>

</Marker>

<Marker name="RPSH">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/scapula</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>-0.070832155659136087 0.023031587505544682 0.0043265068257682393</location>

</Marker>

<Marker name="RUA1">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>-0.034592010760636005 -0.15435002181834689 0.030437079717797744</location>

</Marker>

<Marker name="RUA2">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>-0.066882549645711775 -0.099183312026729586 0.020077981975332393</location>

</Marker>

<Marker name="RUA3">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>-0.010677079821585 -0.092637376843010433 0.039704554053952457</location>

</Marker>

<Marker name="RLEL">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>-0.020891606168357058 -0.28331451842607447 0.0059757030263689614</location>

</Marker>

<Marker name="RMEL">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/humerus</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>0.00025417165441965484 -0.28046576107626064 -0.064047131321694817</location>

</Marker>

<Marker name="RFAradius">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/radius</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>0.061755891041951783 -0.21885970961466911 0.051672473291321186</location>

</Marker>

<Marker name="RFAulna">

<!--Path to a Component that satisfies the Socket 'parent_frame' of type PhysicalFrame (description: The frame to which this station is fixed.).-->

<socket_parent_frame>/bodyset/ulna</socket_parent_frame>

<!--The fixed location of the station expressed in its parent frame.-->

<location>-0.021276903522185586 -0.24198493811111288 0.052560799981544393</location>

</Marker>

</objects>

<groups />

</MarkerSet>

<!--Geometry to be used in contact forces.-->

<ContactGeometrySet name="contactgeometryset">

<objects />

<groups />

</ContactGeometrySet>

<!--Probes in the model.-->

<ProbeSet name="probeset">

<objects />

<groups />

</ProbeSet>

<!--Additional components in the model.-->

<ComponentSet name="componentset">

<objects />

<groups />

</ComponentSet>

</Model>

</OpenSimDocument>

Anexo B. Programa posición de marcadores para Matlab®

```
function R = posicionmarcadores(q)
%q=(psi(t),thet1(t),thet2(t)) en rad
% Meter datos experimentales de los marcadores (mm)
%RACR(a),CLAV(b),ESTERNON(c),C7(d),RASH(e),RPSH(f),RUA1(g),RUA2(h),
%RUA3(i),RLEL(j),RMEL(k),RFAradius(l) y RFAulna(m)
Ra(1,:)=[423.0845
422.95389
```

```
422.82483…];

Ra(2,:)=[1442.7261
1442.7914
1442.8553…];

Ra(3,:)=[-151.83218
-151.77823
-151.72659…];
.
.
.
Y así con todos los marcadores

%Coordenadas generalizadas en el instante inicial en rad(sacadas de
Opensim)
q0= pi/180*[-50.2370146 26.33997303 31.10430955];

%Matrices de giro en instante inicial
Apsi_0(1,:)= [cos(q0(1))   0  -sin(q0(1))];
Apsi_0(2,:)= [         0   1            0];
Apsi_0(3,:)= [sin(q0(1))   0   cos(q0(1))];

Athet1_0(1,:)= [cos(q0(2)) -sin(q0(2)) 0];
Athet1_0(2,:)= [sin(q0(2))  cos(q0(2)) 0];
Athet1_0(3,:)= [        0            0  1];

Athet2_0(1,:)= [cos(q0(3)) -sin(q0(3)) 0];
Athet2_0(2,:)= [sin(q0(3))  cos(q0(3)) 0];
Athet2_0(3,:)= [        0             0 1];

A1_0= Apsi_0*Athet1_0;
A2_0= A1_0*Athet2_0;

%Sacar coordenadas locales de los marcadores
%RACR(a),CLAV(b),ESTERNON(c),C7(d),RASH(e),RPSH(f),RUA1(g),RUA2(h),RUA
3(i),RLEL(j)
%RMEL(k),RFAradius(l) y RFAulna(m)

RH= Re(:,1);    %RASH(hombro) %coordenadas globales del origen del
húmero (mm)
RU= Rj(:,1);    %RLEL(codo) %coordenadas globales del origen del
cúbito (mm)

%De la ecuación tipo "Rb= Ra+A*[vector de coord locales]" se calcula a
%continuación b= Rb-Ra para resolver el sistema por x=A\b
b1= Ra(:,1)-RH;
b2= Rb(:,1)-RH;
b3= Rc(:,1)-RH;
b4= Rd(:,1)-RH;
b5= Re(:,1)-RH;
b6= Rf(:,1)-RH;
b7= Rg(:,1)-RH;
b8= Rh(:,1)-RH;
b9= Ri(:,1)-RH;
b10= Rj(:,1)-RH;
```

```
b11= Rk(:,1)-RH;
b12= Rl(:,1)-RU;  %aquí se resta por RU para que se quede un sistema
del tipo A2*x=b
b13= Rm(:,1)-RU;

%Vectores de coordenadas locales para cada marcador
x1= A1_0\b1;
x2= A1_0\b2;
x3= A1_0\b3;
x4= A1_0\b4;
x5= A1_0\b5;
x6= A1_0\b6;
x7= A1_0\b7;
x8= A1_0\b8;
x9= A1_0\b9;
x10= A1_0\b10;
x11= A1_0\b11;
x12= A2_0\b12;
x13= A2_0\b13;


%Matrices de giro en función de las coordenadas generalizadas
Apsi(1,:)= [cos(q(1))    0   -sin(q(1))];
Apsi(2,:)= [         0    1            0];
Apsi(3,:)= [sin(q(1))    0    cos(q(1))];

Athet1(1,:)= [cos(q(2)) -sin(q(2)) 0];
Athet1(2,:)= [sin(q(2))  cos(q(2)) 0];
Athet1(3,:)= [        0          0  1];

Athet2(1,:)= [cos(q(3)) -sin(q(3)) 0];
Athet2(2,:)= [sin(q(3))  cos(q(3)) 0];
Athet2(3,:)= [        0           0 1];

A1= Apsi*Athet1;
A2= A1*Athet2;

      %Posición marcadores modelo en funcion de las coordenadas
generalizadas (mm) para

%RACR(a),CLAV(b),ESTERNON(c),C7(d),RASH(e),RPSH(f),RUA1(g),RUA2(h),RUA
3(i),RLEL(j)
      %RMEL(k),RFAradius(l) y RFAulna(m) en este orden respectivamente
      R= zeros(3,13);

      R(:,1)= RH+A1*x1;
      R(:,2)= RH+A1*x2;
      R(:,3)= RH+A1*x3;
      R(:,4)= RH+A1*x4;
      R(:,5)= RH+A1*x5;
      R(:,6)= RH+A1*x6;
      R(:,7)= RH+A1*x7;
      R(:,8)= RH+A1*x8;
      R(:,9)= RH+A1*x9;
      R(:,10)= RH+A1*x10;
      R(:,11)= RH+A1*x11;
      R(:,12)= RU+A2*x12;
```

```
R(:,13)= RU+A2*x13;
```

Anexo C. Programa mínimos cuadrados para Matlab®

```
function Q= errormarcadores(q)
global S
R= posicionmarcadores(q);

    %Distancia entre valor experimental y modelo

    F(1)= (norm(S(:,1)-R(:,1)))^2;
    F(2)= (norm(S(:,2)-R(:,2)))^2;
    F(3)= (norm(S(:,3)-R(:,3)))^2;
    F(4)= (norm(S(:,4)-R(:,4)))^2;
    F(5)= (norm(S(:,5)-R(:,5)))^2;
    F(6)= (norm(S(:,6)-R(:,6)))^2;
    F(7)= (norm(S(:,7)-R(:,7)))^2;
    F(8)= (norm(S(:,8)-R(:,8)))^2;
    F(9)= (norm(S(:,9)-R(:,9)))^2;
    F(10)= (norm(S(:,10)-R(:,10)))^2;
    F(11)= (norm(S(:,11)-R(:,11)))^2;
    F(12)= (norm(S(:,12)-R(:,12)))^2;
    F(13)= (norm(S(:,13)-R(:,13)))^2;

    Q= sum(F(:));   %sumatorio del error de todos los marcadores para
cada instante de tiempo (función escalar objetivo)
```

Anexo D. Programa solución coordenadas generalizadas para Matlab®

```
global S

%meter datos experimentales de los marcadores (mm)
Ra(1,:)=[423.0845
422.95389
422.82483…];

Ra(2,:)=[1442.7261
1442.7914
1442.8553…];

Ra(3,:)=[-151.83218
-151.77823
-151.72659…];
.
.
.
Y así con todos los marcadores
%Valor inicial para buscar el mínimo
x0= [0.5 0.5 0.5];
ts= 0:0.01:7.81;
x= zeros(3,length(ts));

for i=1:length(ts)
```

```
    S(:,1)= Ra(:,i);
    S(:,2)= Rb(:,i);
    S(:,3)= Rc(:,i);
    S(:,4)= Rd(:,i);
    S(:,5)= Re(:,i);
    S(:,6)= Rf(:,i);
    S(:,7)= Rg(:,i);
    S(:,8)= Rh(:,i);
    S(:,9)= Ri(:,i);
    S(:,10)= Rj(:,i);
    S(:,11)= Rk(:,i);
    S(:,12)= Rl(:,i);
    S(:,13)= Rm(:,i);

    x(:,i)= fmincon(@errormarcadores,x0,[],[]);
    incremento= x(2,1)-x(2,i);
    x(2,i)= x(2,i)+2*incremento; % ponemos el valor de q(2) como en
Opensim (si no, sale al contrario)
    x0=x(:,i);

end

q= 180/pi*x; %Coordenadas generalizadas en grados(°)
```

Anexo E. Programa velocidad coordenadas generalizadas para Matlab®

```
ts= 0:0.01:7.81;
dq=zeros(3,length(ts));
h= 0.01;

for i=2:(length(ts)-1) %Derivación numérica, diferencia central

%La velocidad en el instante inicial y final es cero
    dq(:,i)= (q(:,i+1)-q(:,i-1))/(2*h);  %velocidad

end
```

Anexo F. Programa posición y velocidad centro de masas para Matlab®

```
%Posición y velocidad del centro de masas de cada eslabón (brazo y
antebrazo)
%El brazo se corresponde con el húmero y el antebrazo con el cúbito

ts= 0:0.01:7.81;
CM_brazo= zeros(3,length(ts));
CM_antebrazo= zeros(3,length(ts));
dCM_brazo= zeros(3,length(ts));
dCM_antebrazo= zeros(3,length(ts));

for i=1:length(ts)

Apsi(1,:)= [cos(x(1,i))   0  -sin(x(1,i))];
```

```
Apsi(2,:)= [          0    1             0];
Apsi(3,:)= [sin(x(1,i))    0    cos(x(1,i))];

Athet1(1,:)= [cos(x(2,i)) -sin(x(2,i)) 0];
Athet1(2,:)= [sin(x(2,i))  cos(x(2,i)) 0];
Athet1(3,:)= [         0            0  1];

Athet2(1,:)= [cos(x(3,i)) -sin(x(3,i)) 0];
Athet2(2,:)= [sin(x(3,i))  cos(x(3,i)) 0];
Athet2(3,:)= [         0            0  1];

A1= Apsi*Athet1;
A2= A1*Athet2;

%Posición centro de masas (m)
CM_brazo(:,i)= Re(:,i)/1000+A1*[0.01806 -0.1401 -0.0127499]'; %vector
posición sacado de opensim
CM_antebrazo(:,i)= Rj(:,i)/1000+A2*[0.009718 -0.09595 0.02429]';

end

for j=2:(length(ts)-1) %Derivación numérica, diferencia central

    %La velocidad en el instante inicial y final es cero
    %Velocidad centro de masas (m/s)
  dCM_brazo(:,j)= (CM_brazo(:,j+1)-CM_brazo(:,j-1))/(2*h);
  dCM_antebrazo(:,j)= (CM_antebrazo(:,j+1)-CM_antebrazo(:,j-1))/(2*h);

end
```

REFERENCIAS

[1] J. L. Palladino y R. B. Davis, «Chapter 4 - Biomechanics,» de *Introduction to Biomedical Engineering (Third Edition)*, J. D. E. a. J. D. Bronzino, Ed., Boston, Academic Press, 2012, pp. 134-136. doi: 10.1016/B978-0-12-374979-6.00004-6

[2] J. Martínez Reina, *Tema 5.1. Introducción a la Biomecánica del Sistema Musculoesquelético (SME),* Sevilla, 2020.

[3] «Wikipedia,» [En línea]. Available: https://es.wikipedia.org/wiki/Exoesqueleto_mec%C3%A1nico#Historia. [Último acceso: 9 Noviembre 2020].

[4] «Espacenet,» [En línea]. Available: https://worldwide.espacenet.com/patent/search?q=pn%3DUS440684A. [Último acceso: 9 Noviembre 2020].

[5] «Guiadisc,» [En línea]. Available: https://www.guiadisc.com/exoesqueletos-evolucion-trajes-roboticos.html. [Último acceso: 9 Noviembre 2020].

[6] A. Sanchis, «Para qué (y cómo) se stán usando los exoesqueletos en las fábricas españolas,» *Xataka,* 20 Julio 2020 [En línea]. Available: https://www.xataka.com/robotica-e-ia/como-se-estan-usando-exoesqueletos-fabricas-espanolas [Último acceso: 9 Noviembre 2020].

[7] «Cyberdyne,» [En línea]. Available: https://www.cyberdyne.jp/english/. [Último acceso: 9 Noviembre 2020].

[8] «Raytheon XOS 2 Exoskeleton, Second-Generation Robotics Suit,» *Army Technology,* 29 Mayo 2020 [En línea]. Available: https://www.army-technology.com/projects/raytheon-xos-2-exoskeleton-us/. [Último acceso: 9 Noviembre 2020].

[9] «Harmonic Drive SE,» [En línea]. Available: https://harmonicdrive.de/es/glosario/el-exoesqueleto. [Último acceso: 9 Noviembre 2020].

[10] «Exoesqueleto.net,» [En línea]. Available: https://exoesqueleto.net/. [Último acceso: 9 Noviembre 2020].

[11] «SuitX,» [En línea]. Available: https://www.suitx.com/shoulderx. [Último acceso: 9 Noviembre 2020].

[12] M. Lyu, W. Chen, X. Ding, J. Wang, S. Bai y H. Ren, «Design of a biologically inspired lower limb exoskeleton for human gait rehabilitation,» *Review Of Scientific Instruments,* vol. 87, nº 10, p. 104301, 25 Octubre 2016. doi: 10.1063/1.4964136

[13] «SuitX,» [En línea]. Available: https://www.suitx.com/legx. [Último acceso: 9 Noviembre 2020].

[14] Jefatura del Estado español, «Agencia Estatal Boletín Oficial del Estado,» 8 Noviembre 1995. [En línea]. Available: https://www.boe.es/eli/es/l/1995/11/08/31. [Último acceso: 7 Noviembre 2020].

[15] Dpto. de Ingeniería Mecánica y Fabricación de la ETSI de Sevilla, *ThoracoScapularShoulderModel,* Sevilla, 2020.

[16] M. A. Crespo, «AMR Producciones,» 16 Enero 2012. [En línea]. Available: https://amrproducciones.blogspot.com/2011/07/captura-de-movimiento-3d.html. [Último acceso: 25 Noviembre 2020].