# Soft Body Simulation

BY MAYANK MEGHWANSHI AND SHIVAM H PRASAD

# What is Soft Body Locomotion?


Source: Jie Tan, Soft Body Locomotion

# Introduction

- Physics based system to simulate and control the locomotion of Soft Body Characters without skeletons

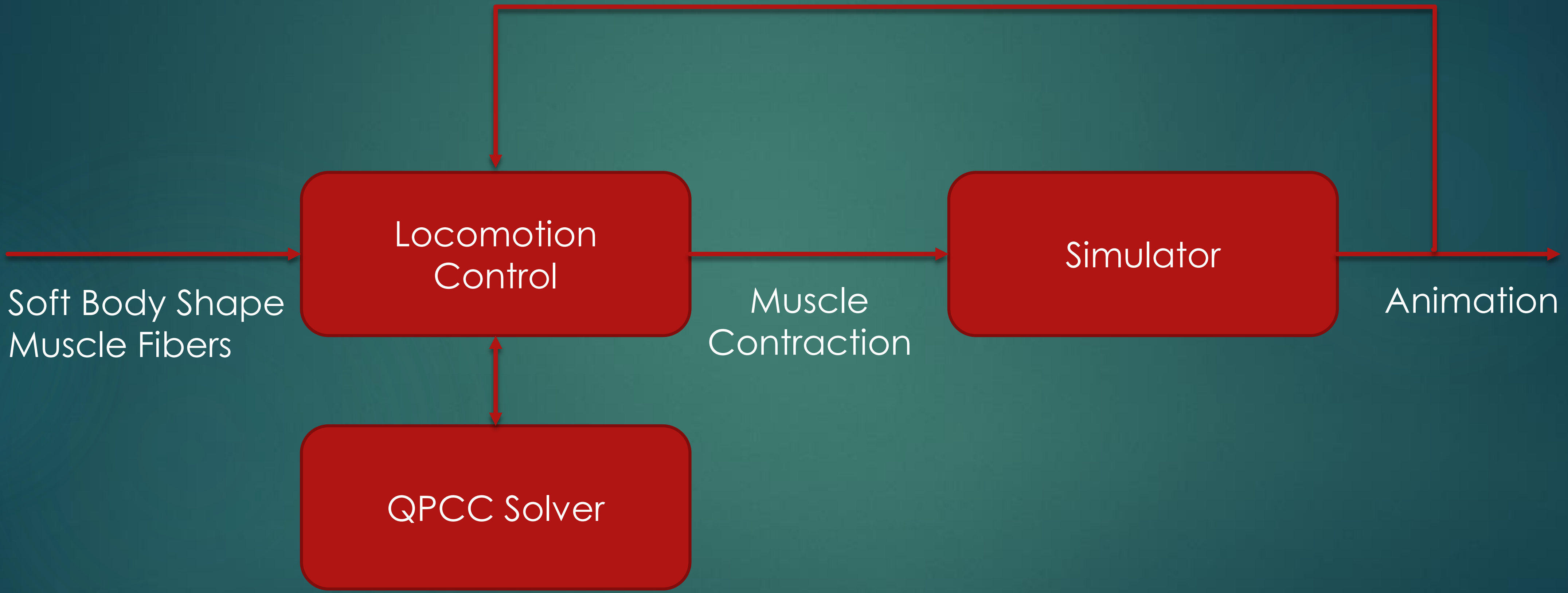
- Paper by Jie Tan, Greg Turk and C. Karen Liu presented in SIGGRAPH 2012 titled Soft Body Locomotion

- Source: http://www.cc.gatech.edu/~jtan34/project/softBodyLocomotion.html

# Motivation

- Wide variety of animals with no skeleton can be modelled using this technique
  - For example, slugs, starfish, earthworms
- Many animated characters move in a flexible manner that they seem to be boneless
- Animation principle of *Squash and Stretch* can be best seen with Soft Body Characters

# System Overview

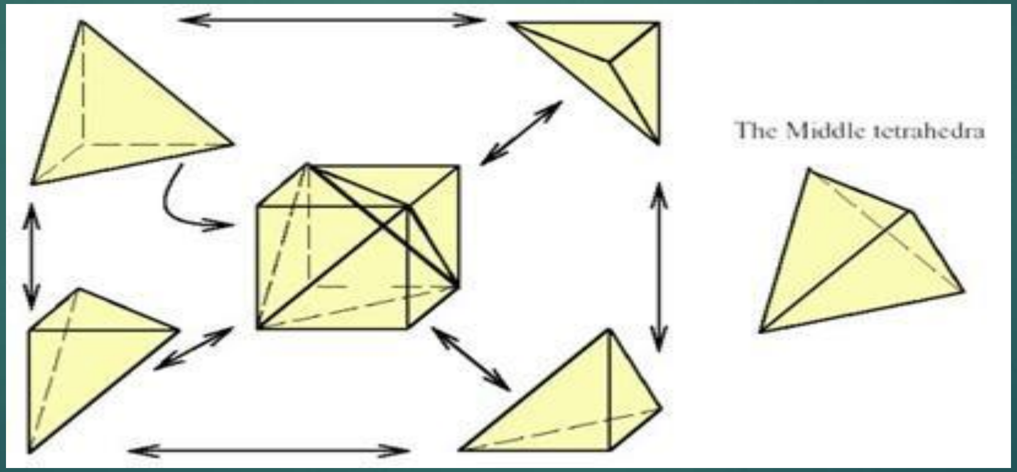


# Problem Components

- Modelling the Soft Body Character
- Modelling of Muscle Fibers
- Evaluation of Dynamic Equation of Motion
- Optimization Problem with muscle contractions and contact forces as parameters
- Rendering of Soft Body Character

# Modelling of Soft Body

- Body of character is modelled using a tetrahedral mesh
- Tetrahedral mesh is generated using TETGEN as also done in referred paper




Source: http://www.byclb.com/TR/Tutorials/volume_rendering/images/image012.jpg

# Modelling of Soft Body continued

- Input to TETGEN
  - Part 1 – Node List
    - First Line <# of points> <dimension> <number of attributes> <boundary marker>
    - Remaining Lines : list of points
      - <point #> <x> <y> <z> [attributes] [boundary markers]
  - Part 2 – Facet List
    - First Line <# of facets> <boundary marker>
    - Following Lines : list of facets where each facet has following format
      - <facet #>
      - <number of polygons>
      - For each polygon
        - <# of corners> <corner 1> <corner 2> …. <corner #>

# Modelling of Soft Body continued..

- Output of TETGEN
  - First Line
    - <# of tetrahedral> <# of nodes per tetrahedral> <# of attributes>
  - Remaining Lines
    - <tetrahedral #> <node> <node> …. [attributes]
  - This file acts as the model for Soft Body Character

# Dynamic Equation of Motion

- Dynamics of Soft Body is dependent on different kinds of forces
  - Muscle Force – Force exerted by muscle fibers on each vertex
  - Internal Elastic Force – Force due to stress and strain characteristics of material of body
  - Damping Force – It prevents high frequency changes in body position
  - External Force – Force exerted on the body by external agent
    - For example, gravity, friction etc.

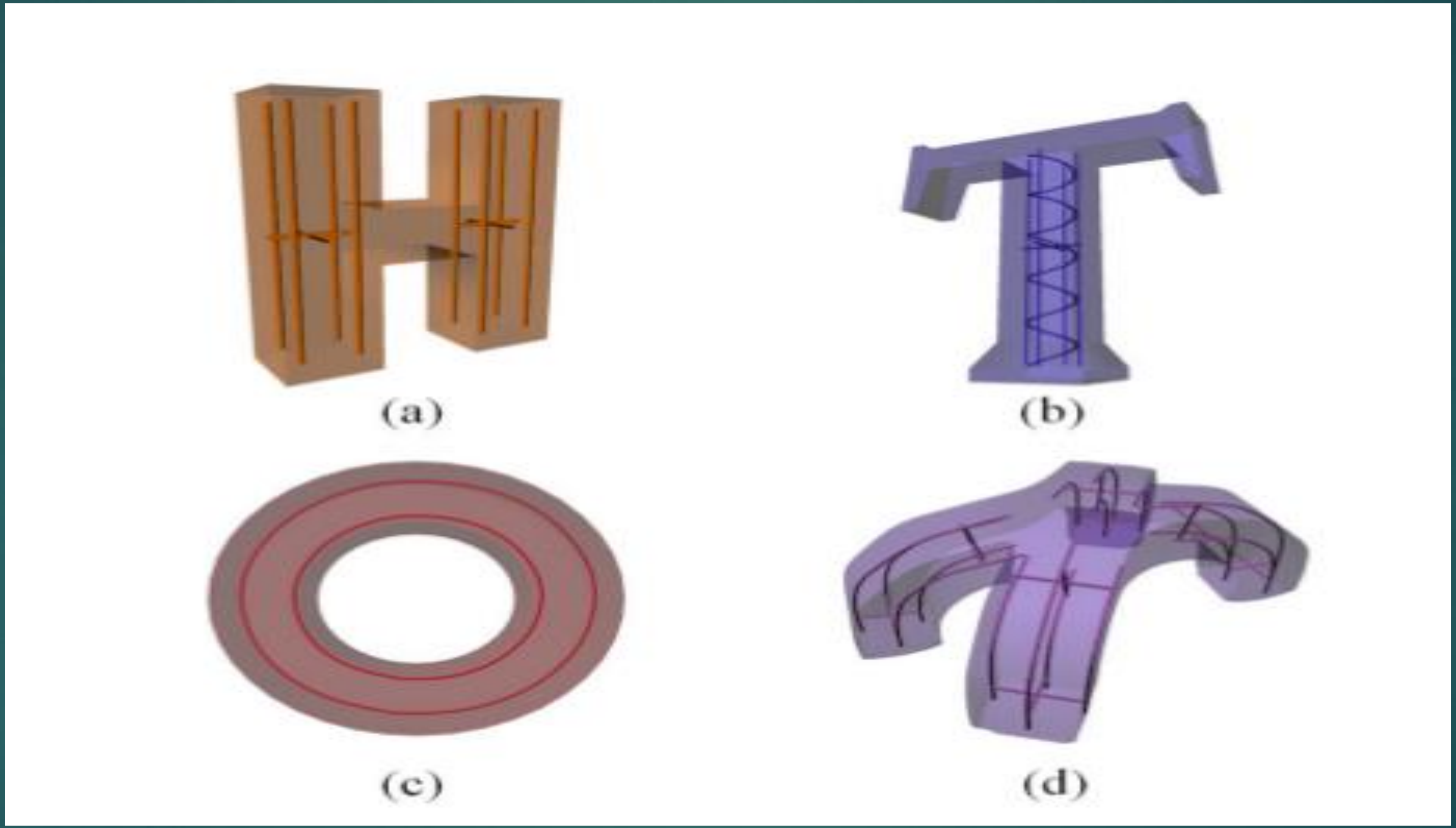
# Muscle Modelling

- Muscles are modelled as polygonal curves with small number of muscle segments
- Number of segments describes the number of independently contracting degrees of freedom
- Muscle segments cannot bend or elongate. They can only contract along their current direction
- Muscle contraction induces force which is modelled as a deformable spring

# Muscle Groups

- Longitudinal Muscles
  - These are linear muscle extending from one end of body to other
  - These control shortening and bending of soft body
- Radial Muscles
  - These are short muscles that span a cross-section of the soft body
  - Their contraction along with volume preservation causes elongation of the body
- Helical Muscles
  - These muscles wrap around the body in a helical shape
  - Their contraction cause the soft body to twist

# Muscle Groups continued..



Source: Jie Tan, Soft Body Locomotion, SIGGRAPH 2012

# Muscle Forces

- Force due to contraction of a muscle segment is given by

$$f = k(l_d - l)$$

- For each muscle, we calculate the force by each segment of that muscle using their respective contractions and contraction direction

- These small forces are then added vectorially to get net force exerted by the muscle fiber

# Muscle Forces continued..

- Each tetrahedral element is affected by multiple muscles
- To calculate force on single element we take weighted sum of forces from each muscle fiber influencing it
- Weight given to a muscle fiber is inversely proportional to shortest distance of the center of element from the muscle fiber
- Normalization of weights is done within a muscle group so that their effects are distinctly visible

# Muscle Forces continued..

- Now we need to calculate muscle force on each vertex of the body
  - For each face, force on tetrahedral element is multiplied with area weighted face normal
  - This force is then evenly distributed among vertices of that face
- This finally gives us muscle force component on each vertex of the Soft Body

# Finite Element Model

- FEM assumes that the body is divided into many small elements, in our case it's the tetrahedral elements
- Using FEM we calculate for each element a stiffness matrix depending upon material properties and displacement of each vertex with respect to initial position
- FEM then integrates these elemental stiffness matrix to form a global stiffness matrix for the entire body
- Global stiffness matrix is then used to calculate internal elastic force exerted on the body

# Finite Element Model continued..

- For each element using its initial position we calculate a 12x12 stiffness matrix K using Lame’s coefficient
- This result has its origins in Solid Mechanics and Theory of Vibrations
- Now force on each element is calculated using the stiffness matrix and displacement from initial position
- Now we distribute the force exerted on an element to all its vertices and hence we get Global force on each vertex of the soft body

# Optimization Function

- New position of the body at each time step is function of muscle lengths and contact forces
- We formulate an objective function on next state of soft body having muscle fiber segment lengths and contact forces as parameters
- Constraints on optimization are
  - Muscle segment length : $0.5\ l_0 <= l_d <= l_0$
  - Contact Forces : Controls contact force when in conditions of sliding or lifting. These enforce a valid contact of body with the ground

# Optimization Function Controllers

- We use two levels of controllers for formulating the objective function
- Momentum Control
  - Important for maintaining balance and locomotion
- Base Control
  - Controls the size of contact area
  - Important for balancing of soft body

# Optimization Function Controllers

- Momentum Control
  - For Linear Momentum
    - This calculates the difference between change in linear momentum when calculated for each element and aggregated with change in linear momentum calculated under assumption that entire body mass is present at Center of Mass
  - For Angular Momentum
    - This calculates the difference between change in angular momentum when calculated for each element and aggregated with desired change in angular momentum
    - Desired angular momentum change is predefined in the system

# Optimization Function Controllers

- Base Control
  - By squashing and stretching soft body can change its contact area to maintain its balance
  - For this, we define an objective function which controls the projected base area by matching its change rate to desired rate
  - Desired rate of change in contact area is pre-defined in the system
  - Projected area is calculated by projection of pre-defined base area on ground

# Optimization Constraints

- Muscle Segment Length
  - $0.5\, l_0 <= l_d <= l_0$
- Complementarity Constraints from Contact Forces
  - Normal force from ground and vertical velocity are complementary
    - Both cannot have non zero values simultaneously
    - This determines condition for contact breaking at contact points
  - Tangential velocity and Tangential Force are complementary
    - When frictional force is dominant velocity is zero while when velocity is non zero frictional force is overcome by external force
    - This determines the static or sliding condition at the contact points

# QPCC for Optimization Solving

- QPCC stands for Quadratic Program with Complementarity Constraints
- Objective function that we formulated for Locomotion and Base Control is a quadratic problem with complementary constraints
- Since our constraints are pairable paper makes use of a new algorithm to solve QPCC problem
- This algorithm assumes a solver for Quadratic Optimization Problem

# QPCC Algorithm

- First it generates an initial guess which is consistent with the complementary conditions
- Now it generates a Boolean array which specify for each complementary condition which part is set to boundary condition.
- Now we can get a linear constraint out of each pair of complementary condition by selecting non boundary condition
- After this we can get a quadratic problem by replacing complementarity conditions with linear constraints

# QPCC Algorithm continued..

- This quadratic problem is then fed to QP solver which minimizes the objective function and returns the minimized value and parameter values at minima
- Now we check if any of the complementary condition needs to be inverted, this happens at contact breaking / joining or sliding / stopping conditions
- This can result in another QP problem with different linear constraints due to inversion of complementary condition
- Thus we form a list of QP problems

# QPCC Algorithm continued..

- Now we solve each QP problem in the list using the QP solver and get minima and corresponding parameter values
- Now we find the QP solution with which we achieved the minimum value for objective function and return its corresponding parameter values as solutions to the QPCC problem
- This QPCC problem is solved at each time step to get muscle segment lengths and contact forces which then are used in dynamic equation of motion to get next position of soft body

# Expected Results

- We aim to successfully simulate
  - Balancing
  - Jumping
  - Sliding
  - Rolling

  motions for soft body characters

"

Thank You

"

Any Questions ??

# Appendix

- Dynamic Equation of Motion

$$M\ddot{p} = f_x + f_e + f_d + f_m$$

- Internal Elastic Force

$$\hat{f}_e = -\hat{B}^T \hat{D} \hat{B}(\hat{p} - \hat{R}\hat{x})$$

- Damping Force

$$f_d = -(\mu M + \lambda K)\dot{p}$$

- Force equation after substitution

$$M\ddot{p} = f_x - K(p - Rx) - (\mu M + \lambda K)\dot{p} + A(l_d - l)$$

$$\tilde{M}\dot{p}^{n+1} = M\dot{p}^n + \Delta t(f_x^n - K(p^n - Rx^n) + A(l_d - l^n))$$
$$p^{n+1} = p^n + \Delta t \dot{p}^{n+1}$$

# Appendix

- Linear Momentum Control

$$G(l_d, f_\perp, f_\parallel) = \|\dot{L}(\dot{p}^{n+1}, \dot{p}^n) - \bar{\dot{L}}\|^2$$

- Angular Momentum Control

$$G(l_d, f_\perp, f_\parallel) = \|\dot{H}(\dot{p}^{n+1}, \dot{p}^n, \dot{p}^n) - \bar{\dot{H}}\|^2$$

- Base Control

$$G(l_d, f_\perp, f_\parallel) = \|\dot{A}(\dot{p}^{n+1}, \dot{p}^n) - \bar{\dot{A}}\|^2$$

- QPCC objective function and constraints

$$\min_{\mathrm{l}_d, \mathrm{f}_\perp, \mathrm{f}_\parallel, \lambda} G(\mathrm{l}_d, \mathrm{f}_\perp, \mathrm{f}_\parallel)$$

subject to

$$0.5\mathrm{l}_0 \leq \mathrm{l}_d \leq \mathrm{l}_0$$

$$0 \leq \begin{bmatrix} \mathrm{f}_\perp \\ \mathrm{f}_\parallel \\ \lambda \end{bmatrix} \perp \begin{bmatrix} \mathrm{N}^T \dot{\mathrm{p}}^{n+1} \\ \mathrm{D}^T \dot{\mathrm{p}}^{n+1} + \mathrm{E}\lambda \\ \mu \mathrm{f}_\perp - \mathrm{E}^T \mathrm{f}_\parallel \end{bmatrix} \geq 0$$