Universität Stuttgart

Benjamin Maier, Nehzat Emamy

Parallelizing a Multiscale Model Of a Skeletal Muscle

Abteilungsklausur 2017

March 9, 2017

Simulation of Large Systems
Institute for Parallel and Distributed Systems
University of Stuttgart

IPVS

SGS

SimTech Cluster of Excellence

Overview

Overview

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult

- Need for simulation

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult

- Need for simulation

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult

- Need for simulation

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult
- Need for simulation

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult
  - Involves tests on human or animals
  - Only for small components
  - Hard measurements of inner quantities $\rightsquigarrow$ EMG
- Need for simulation
  - Submodels for different phenomena
  - Integrated simulation of muscle physiology - "overall human model"

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult
  - Involves tests on human or animals
  - Only for small components
  - Hard measurements of inner quantities $\rightsquigarrow$ EMG
- Need for simulation
  - Submodels for different phenomena
  - Integrated simulation of muscle physiology - "overall human model"

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult
  - Involves tests on human or animals
  - Only for small components
  - Hard measurements of inner quantities $\rightsquigarrow$ EMG
- Need for simulation
  - Submodels for different phenomena
  - Integrated simulation of muscle physiology - "overall human model"

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult
  - Involves tests on human or animals
  - Only for small components
  - Hard measurements of inner quantities $\rightsquigarrow$ EMG
- Need for simulation
  - Submodels for different phenomena
  - Integrated simulation of muscle physiology – "overall human model"

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult
  - Involves tests on human or animals
  - Only for small components
  - Hard measurements of inner quantities $\rightsquigarrow$ EMG
- Need for simulation
  - Submodels for different phenomena
  - Integrated simulation of muscle physiology - "overall human model"

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult
  - Involves tests on human or animals
  - Only for small components
  - Hard measurements of inner quantities $\rightsquigarrow$ EMG
- Need for simulation
  - Submodels for different phenomena
  - Integrated simulation of muscle physiology - "overall human model"

Motivation

The musculoskeletal system

- Fitness for different tasks: endurance, fine control, strength



- Enabled by same muscle tissue
- Complex system: skeletal muscles, bones, tendons, motor neurons

Understanding the system

- Experiments difficult
  - Involves tests on human or animals
  - Only for small components
  - Hard measurements of inner quantities $\rightsquigarrow$ EMG
- Need for simulation
  - Submodels for different phenomena
  - Integrated simulation of muscle physiology - ”overall human model”

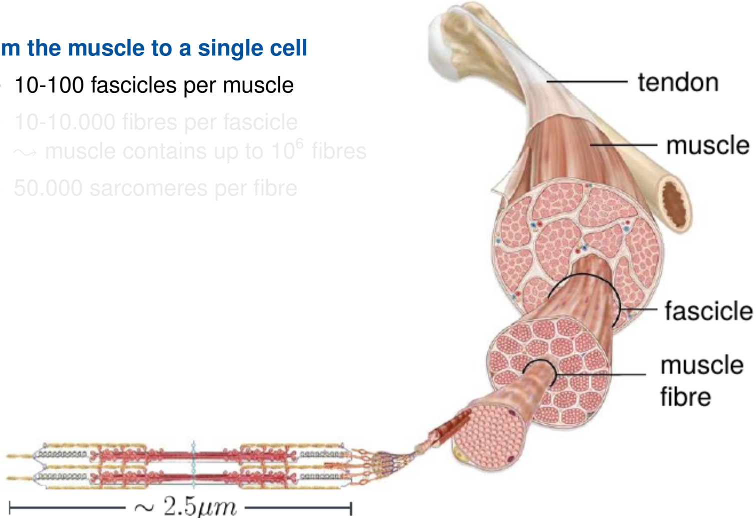
Physiological Introduction

1. Motivation
2. **Physiological Introduction**
3. The Mathematical Model
4. The Computational Framework OpenCMISS

Physiological Introduction

From the muscle to a single cell

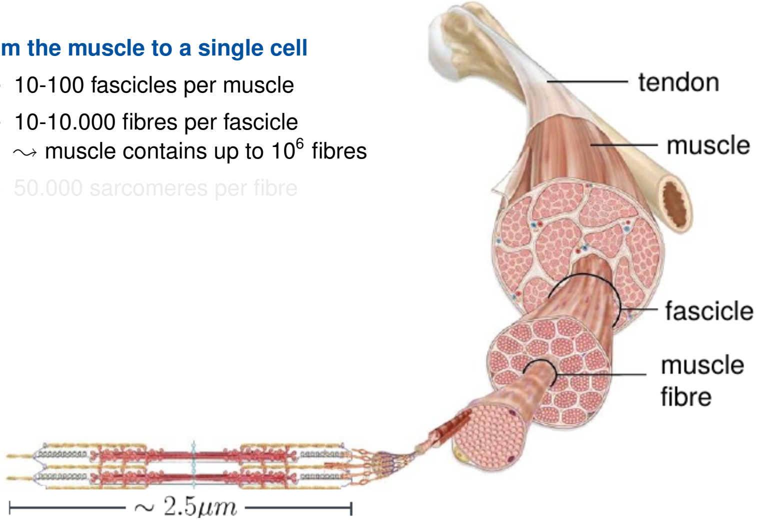
- 10-100 fascicles per muscle
- 10-10.000 fibres per fascicle
  ⇝ muscle contains up to 10^6 fibres
- 50.000 sarcomeres per fibre



Physiological Introduction

From the muscle to a single cell

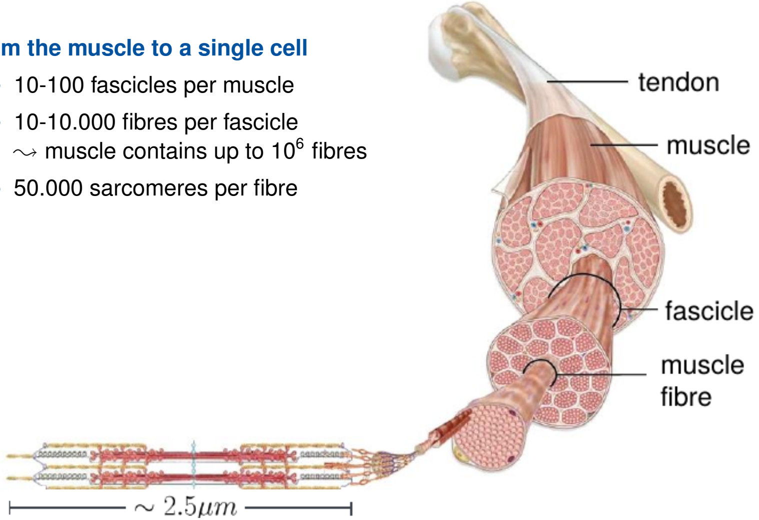
- 10-100 fascicles per muscle
- 10-10.000 fibres per fascicle
  $\rightsquigarrow$ muscle contains up to 10^6 fibres
- 50.000 sarcomeres per fibre



Physiological Introduction

From the muscle to a single cell

- 10-100 fascicles per muscle
- 10-10.000 fibres per fascicle
  $\rightsquigarrow$ muscle contains up to 10^6 fibres
- 50.000 sarcomeres per fibre

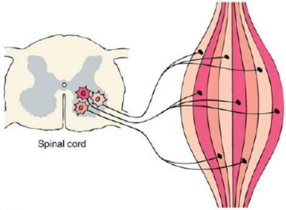


Physiological Introduction

Activation of muscle fibres

- Origin of stimulation is motor neuron pool in spinal cord
- Transport of electrical signals via nerves
- Activation level proportional to frequency of ”spike trains”

- Fibres are adressed as motor units (MU), all fibres of a MU are excited simultaneously
- Exponential MU distribution
- Action potential travels along muscle fibre
- Neighbouring fibres are not electrophysically coupled

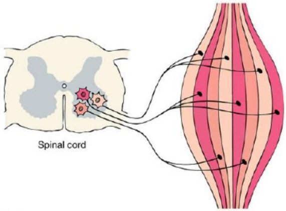


Fibre

Physiological Introduction

Activation of muscle fibres

- Origin of stimulation is motor neuron pool in spinal cord
- Transport of electrical signals via nerves
- Activation level proportional to frequency of "spike trains"
- Fibres are adressed as motor units (MU), all fibres of a MU are excited simultaneously
- Exponential MU distribution
- Action potential travels along muscle fibre
- Neighbouring fibres are not electrophysically coupled

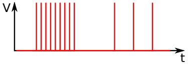


Fibre

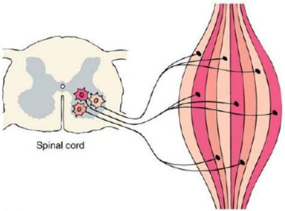
Physiological Introduction

Activation of muscle fibres

- Origin of stimulation is motor neuron pool in spinal cord
- Transport of electrical signals via nerves
- Activation level proportional to frequency of ”spike trains”



- Fibres are adressed as motor units (MU), all fibres of a MU are excited simultaneously
- Exponential MU distribution
- Action potential travels along muscle fibre
- Neighbouring fibres are not electrophysically coupled

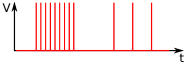


Fibre

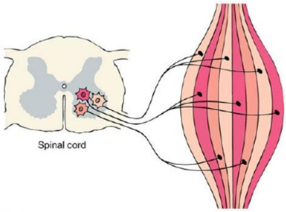
Physiological Introduction

Activation of muscle fibres

- Origin of stimulation is motor neuron pool in spinal cord
- Transport of electrical signals via nerves
- Activation level proportional to frequency of ”spike trains”



- Fibres are adressed as motor units (MU), all fibres of a MU are excited simultaneously
- Exponential MU distribution
- Action potential travels along muscle fibre
- Neighbouring fibres are not electrophysically coupled

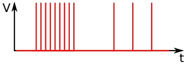


Fibre

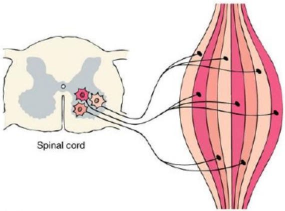
Physiological Introduction

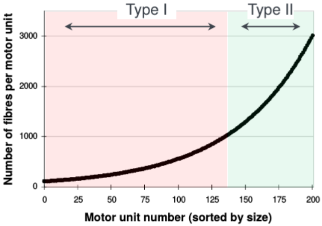
Activation of muscle fibres

- Origin of stimulation is motor neuron pool in spinal cord
- Transport of electrical signals via nerves
- Activation level proportional to frequency of ”spike trains”



- Fibres are adressed as motor units (MU), all fibres of a MU are excited simultaneously
- Exponential MU distribution
- Action potential travels along muscle fibre
- Neighbouring fibres are not electrophysically coupled



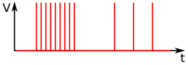


Fibre

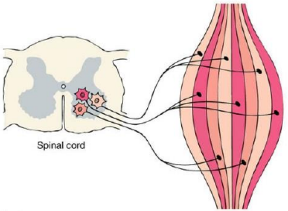
Physiological Introduction

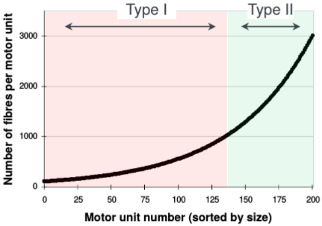
Activation of muscle fibres

- Origin of stimulation is motor neuron pool in spinal cord
- Transport of electrical signals via nerves
- Activation level proportional to frequency of ”spike trains”



- Fibres are adressed as motor units (MU), all fibres of a MU are excited simultaneously
- Exponential MU distribution
- Action potential travels along muscle fibre
- Neighbouring fibres are not electrophysically coupled



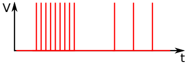


Fibre

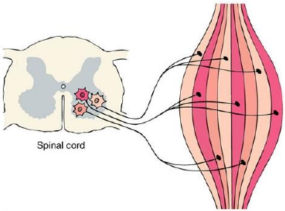
Physiological Introduction

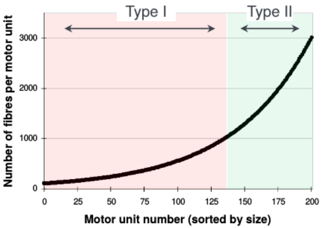
Activation of muscle fibres

- Origin of stimulation is motor neuron pool in spinal cord
- Transport of electrical signals via nerves
- Activation level proportional to frequency of ”spike trains”



- Fibres are adressed as motor units (MU), all fibres of a MU are excited simultaneously
- Exponential MU distribution
- Action potential travels along muscle fibre
- Neighbouring fibres are not electrophysically coupled



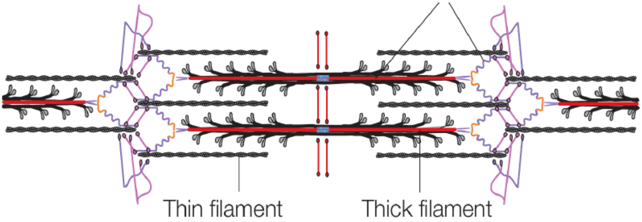


Fibre

Physiological Introduction

Force generation

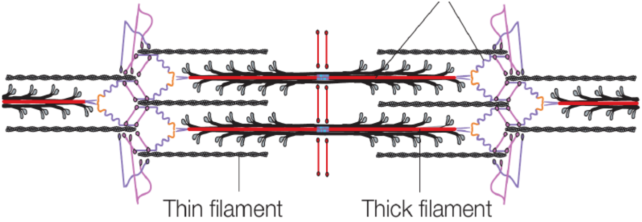
- Contractile unit: sarcomere
- Crossbridges attach and detach to thin filament
- Passive behaviour: hyperelastic



Physiological Introduction

Force generation

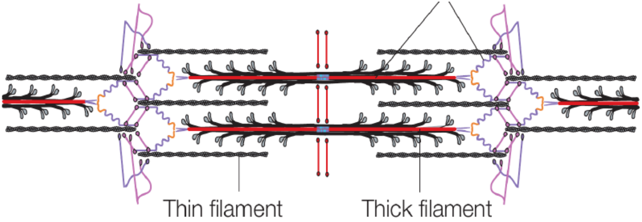
- Contractile unit: sarcomere
- Crossbridges attach and detach to thin filament
- Passive behaviour: hyperelastic



Physiological Introduction

Force generation

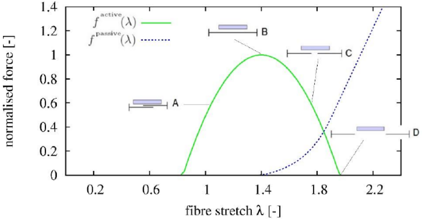
- Contractile unit: sarcomere
- Crossbridges attach and detach to thin filament
- Passive behaviour: hyperelastic



Physiological Introduction

Force generation characteristics

- Force-length relationship

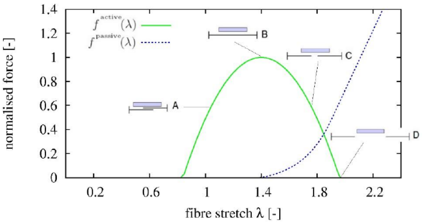


- Force-velocity relationship

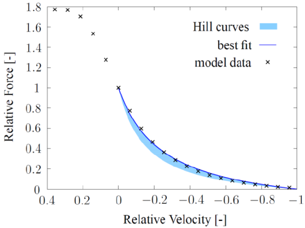
Physiological Introduction

Force generation characteristics

- Force-length relationship



- Force-velocity relationship

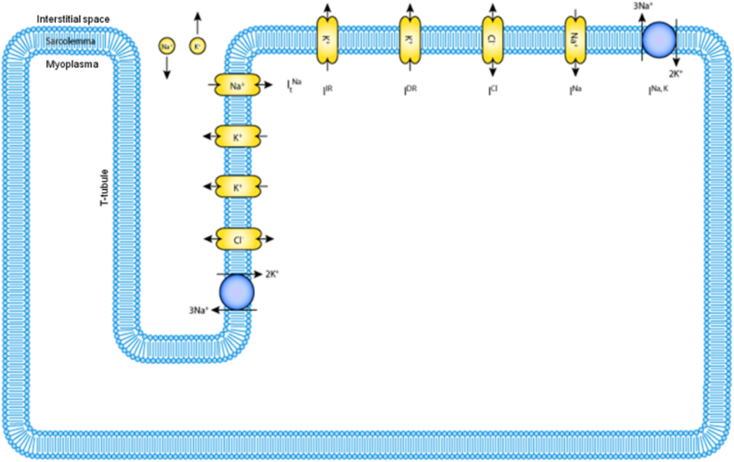


The Mathematical Model

Model on cellular level

Ion currents

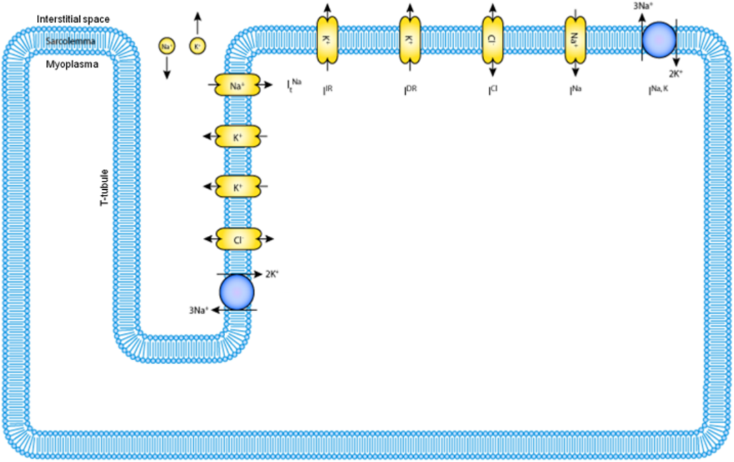
- Ionic currents between intracellular and extracellular space
- Different ions pass through dedicated channels and carry charge



Model on cellular level

Ion currents

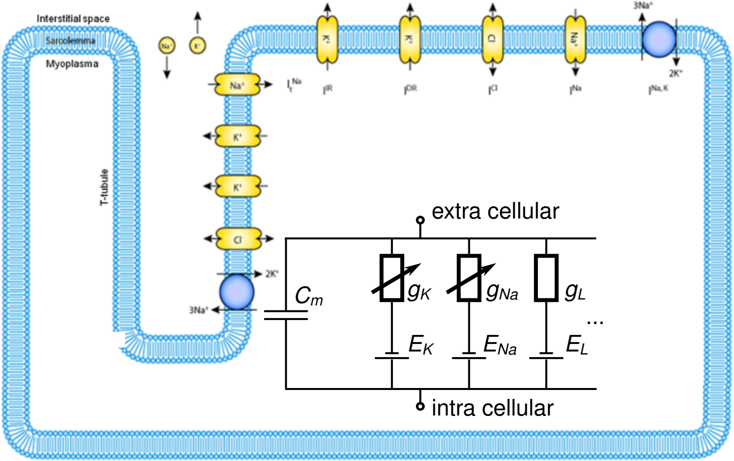
- Ionic currents between intracellular and extracellular space
- Different ions pass through dedicated channels and carry charge



Model on cellular level

Ion currents

- Ionic currents between intracellular and extracellular space
- Different ions pass through dedicated channels and carry charge



Model on cellular level

Ion currents

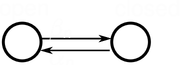
- Ionic currents between intracellular and extracellular space
- Different ions pass through dedicated channels and carry charge

$$\begin{aligned} \frac{\mathrm{d}V_m}{\mathrm{d}t} &= -\frac{1}{C_m} \cdot \big(I_{ionic}(V_m) - I_{stim}/A\big) \\ I_{ionic} &= I_{Cl} + I_{IR} + I_{DR} + I_{Na} + I_{NaK} \\ I_{Cl} &= g_{Cl}(V_m) \cdot J_{Cl}(Cl_i, Cl_o, V_m), \end{aligned} \tag{1}$$

where $g_i(V_m)$ ... conductance of ion channel $i \in \{Cl, Na, ...\}$

Conductances

- Channels open on molecular level



$$g_{Na} = \bar{g}_{Na}\, m^3\, h$$

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \alpha_m\,(1-m) + \beta_m\, m, \qquad \frac{\mathrm{d}h}{\mathrm{d}t} = \alpha_h(1-h) + \beta_h\, h \tag{2}$$

Model on cellular level

Ion currents

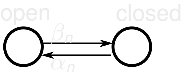
- Ionic currents between intracellular and extracellular space
- Different ions pass through dedicated channels and carry charge

$$
\begin{aligned}
\frac{\mathrm{d} V_m}{\mathrm{d} t} &= -\frac{1}{C_m} \cdot \left(I_{ionic}(V_m) - I_{stim}/A \right) \\
I_{ionic} &= I_{Cl} + I_{IR} + I_{DR} + I_{Na} + I_{NaK} \\
I_{Cl} &= g_{Cl}(V_m) \cdot J_{Cl}(Cl_i, Cl_o, V_m),
\end{aligned}
\tag{1}
$$

where $g_i(V_m)$ ... conductance of ion channel $i \in \{Cl, Na, ...\}$

Conductances

- Channels open on molecular level



$$g_{Na} = \bar{g}_{Na}\, m^3\, h$$

$$\frac{\mathrm{d} m}{\mathrm{d} t} = \alpha_m (1 - m) + \beta_m\, m, \qquad \frac{\mathrm{d} h}{\mathrm{d} t} = \alpha_h (1 - h) + \beta_h\, h \tag{2}$$

Model on cellular level

Ion currents

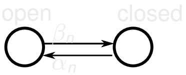
- Ionic currents between intracellular and extracellular space
- Different ions pass through dedicated channels and carry charge

$$\begin{aligned} \frac{\mathrm{d}V_m}{\mathrm{d}t} &= -\frac{1}{C_m} \cdot \left(I_{ionic}(V_m) - I_{stim}/A\right) \\ I_{ionic} &= I_{Cl} + I_{IR} + I_{DR} + I_{Na} + I_{NaK} \\ I_{Cl} &= g_{Cl}(V_m) \cdot J_{Cl}(Cl_i, Cl_o, V_m), \end{aligned} \tag{1}$$

where $g_i(V_m)$ ... conductance of ion channel $i \in \{Cl, Na, ...\}$

Conductances

- Channels open on molecular level



$$g_{Na} = \bar{g}_{Na}\, m^3\, h$$

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \alpha_m (1 - m) + \beta_m\, m, \qquad \frac{\mathrm{d}h}{\mathrm{d}t} = \alpha_h (1 - h) + \beta_h\, h \tag{2}$$

Model on cellular level

Ion currents

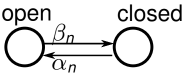
- Ionic currents between intracellular and extracellular space
- Different ions pass through dedicated channels and carry charge

$$\begin{aligned} \frac{\mathrm{d}V_m}{\mathrm{d}t} &= -\frac{1}{C_m} \cdot \left(I_{ionic}(V_m) - I_{stim}/A\right) \\ I_{ionic} &= I_{Cl} + I_{IR} + I_{DR} + I_{Na} + I_{NaK} \\ I_{Cl} &= g_{Cl}(V_m) \cdot J_{Cl}(Cl_i, Cl_o, V_m), \end{aligned} \tag{1}$$

where $g_i(V_m)$ ... conductance of ion channel $i \in \{Cl, Na, ...\}$

Conductances

- Channels open on molecular level



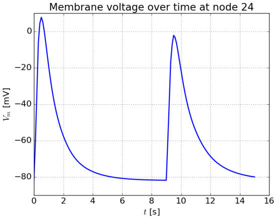
$$g_{Na} = \bar{g}_{Na}\, m^3\, h$$

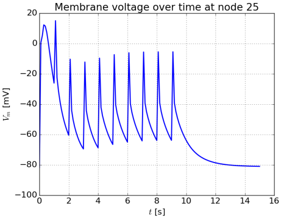
$$\frac{\mathrm{d}m}{\mathrm{d}t} = \alpha_m\,(1-m) + \beta_m\, m, \qquad \frac{\mathrm{d}h}{\mathrm{d}t} = \alpha_h\,(1-h) + \beta_h\, h \tag{2}$$

Model on cellular level

Behaviour of electrodynamics model

- Fatigue is included in the model





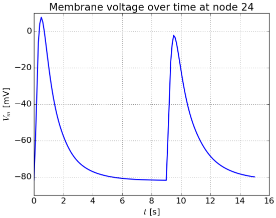
Force generation

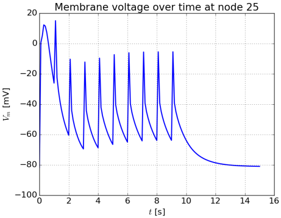
- State model with transition parameters

Model on cellular level

Behaviour of electrodynamics model

- Fatigue is included in the model





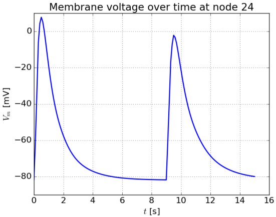
Force generation

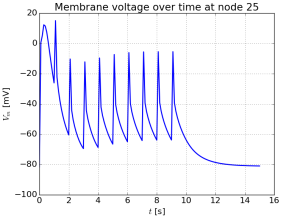
- State model with transition parameters

Model on cellular level

Behaviour of electrodynamics model

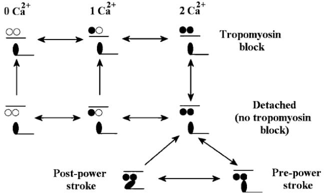
- Fatigue is included in the model





Force generation

- State model with transition parameters



Model on cellular level

Model overview

- System of ordinary differential equations with 56 variables
- Input: I_{Stim}, Output: F

Two compartment model

$I_T = (v_s - v_t)/R_a 10^3\ \mu A/cm^2$ (access current)

$\frac{dv_s}{dt} = -\left(I_s^{ionic} + I_T\right)/C_m$ mV/ms (sarcolemma membrane voltage)

$\frac{dv_t}{dt} = -\left(I_t^{ionic} - I_T/\gamma\right)/C_m$ mV/ms (t - tubule membrane voltage)

$\left[\frac{dh_s}{dt}, \frac{dm_s}{dt}, \frac{dn_s}{dt}, \frac{dh_s^K}{dt}, \frac{dS_s}{dt}, I^{Cl}, I^{IR}, I^{DR}, I^{Na}, I^{NaK}\right] = Z_s\left(v_s, h_s, m_s, n_s, h_s^K, S_s, K_i, K_e, Na_i, Na_e\right)$

$I_s^{ionic} = I^{Cl} + I^{IR} + I^{DR} + I^{Na} + I^{NaK} - I_{app}(t; t_{app})\ \mu A/cm^2$ (sarcolemma current)

$\left[\frac{dh_t}{dt}, \frac{dm_t}{dt}, \frac{dn_t}{dt}, \frac{dh_t^K}{dt}, \frac{dS_t}{dt}, I_t^{Cl}, I_t^{IR}, I_t^{DR}, I_t^{Na}, I_t^{NaK}\right] = Z_t\left(v_t, h_t, m_t, n_t, h_t^K, S_t, K_i, K_t, Na_i, Na_t\right)$

$I_t^{ionic} = I_t^{Cl} + I_t^{IR} + I_t^{DR} + I_t^{Na} + I_t^{NaK} \mu A/cm^2$ (t - tubule current) (1)

$\frac{dK_i}{dt} = -f_T\left(I_t^{IR} + I_t^{DR} - 2I_t^{NaK} + I_{rest}^K\right)/(1000F\xi) - \left(I^{IR} + I^{DR} - 2I^{NaK} + I_{rest}^K\right)/(1000F\xi_2)$ mM/ms (intracellular $[K^+]$)

$\frac{dK_t}{dt} = \left(I_t^{IR} + I_t^{DR} - 2I_t^{NaK} + I_{rest}^K\right)/(1000F\xi) - (K_t - K_e)/\tau_K$ mM/ms (t - tubule $[K^+]$)

$\frac{dK_e}{dt} = \left(I^{IR} + I^{DR} - 2I^{NaK} + I_{rest}^K\right)/(1000F\xi_3) + (K_t - K_e)/\tau_{K_2}$ mM/ms (interstitial $[K^+]$)

$dNa_i/dt = -f_T\left(I_t^{Na} + 3I_t^{NaK} + I_{rest}^{Na}\right)/(1000F\xi) - \left(I^{Na} + 3I^{NaK} + I_{rest}^{Na}\right)/(1000F\xi_2)$ mM/ms (intracellular $[Na^+]$)

$dNa_t/dt = \left(I_t^{Na} + 3I_t^{NaK} + I_{rest}^{Na}\right)/(1000F\xi) - (Na_t - Na_e)/\tau_{Na}$ mM/ms (t - tubule $[Na^+]$)

$dNa_e/dt = \left(I^{Na} + 3I^{NaK} + I_{rest}^{Na}\right)/(1000F\xi_3) + (Na_t - Na_e)/\tau_{Na_2}$ mM/ms (interstitial $[Na^+]$)

where the function Z_s is defined by

$\left[\frac{dh}{dt}, \frac{dm}{dt}, \frac{dn}{dt}, \frac{dh^K}{dt}, \frac{dS}{dt}, I^{Cl}, I^{IR}, I^{DR}, I^{Na}, I^{NaK}\right]$

$= Z_s(V, h, m, n, h^K, S, K_i, K_o, Na_i, Na_o)$

$\alpha_h = \bar{\alpha}_h \exp(-(V - V_h)/K_{\alpha_h})\ ms^{-1}$

$\beta_h = \bar{\beta}_h/(1 + \exp(-(V - V_h)/K_{\beta_h}))\ ms^{-1}$

$\alpha_m = \bar{\alpha}_m(V - V_m)/(1 - \exp(-(V - V_m)/K_{\alpha_m}))\ ms^{-1}$

$\beta_m = \bar{\beta}_m \exp(-(V - V_m)/K_{\beta_m})\ ms^{-1}$

$\alpha_n = \bar{\alpha}_n(V - V_n)/(1 - \exp(-(V - V_n)/K_{\alpha_n}))\ ms^{-1}$

$\beta_n = \bar{\beta}_n \exp(-(V - V_n)/K_{\beta_n})\ ms^{-1}$

$h_\infty^K = 1/(1 + \exp((V - V_{h_K^\infty})/A_{h_K^\infty}))$

$S^\infty = 1/(1 + \exp((V - V_{S^\infty})/A_{S^\infty}))$

$\tau_{h^K} = 1000 \times \exp(-(V + 40)/25.75)$ ms

$\tau_S = 8571/(0.2 + 5.65((V + V_\tau)/100)^2)$ ms

$a = 1/(1 + \exp((V - V_a)/A_a))$ (2)

$E_K = RT/(+F)\log(K_o/K_i)$ mV (K^+ Nernst potential)

$E_{Cl} = E_K(Cl^-$ Nernst potential)

$E_{Na} = RT/(+F)\log(Na_o/Na_i)$ mV (Na^+ Nernst potential)

$Cl_i = (128 + 5 \times 5.7)$
$/(5 + \exp((-FE_K)/(RT)))$ (Intracellular $[Cl^-]$)

$Cl_o = (128 + 5 \times 5.7) - 5Cl_i$ (Extracellular $[Cl^-]$)

$J_{Cl} = V(Cl_i - Cl_o\exp((FV)/(RT)))/(1 - \exp((FV)/(RT)))$

$J_K = V(K_i - K_o\exp(-(FV)/(RT)))/(1 - \exp(-(FV)/(RT)))$

$J_{Na} = V(Na_i - Na_o\exp(-(FV)/(RT)))/(1 - \exp(-(FV)/(RT)))$

$g_{Cl} = \bar{g}_{Cl} a^4\ mS/cm^2$

$K_R = K_o\exp(-\delta E_K F/(RT))$ mM

$\bar{g}_{IR} = G_K K_R^2/(K_K + K_R^2)\ mS/cm^2$

$y = 1 - (1 + (K_S/(S_i^2\exp(2(1-\delta)VF/(RT)))) \times (1 + K_R^2/K_K))^{-1}$

$g_{IR} = \bar{g}_{IR} y\ mS/cm^2$

$g_{DR} = \bar{g}_K n^4 h^K\ mS/cm^2$

$g_{Na} = \bar{g}_{Na} m^3 hS\ mS/cm^2$

$I_{Cl} = g_{Cl}(J_{Cl}/45)\ \mu A/cm^2$ (Cl^- current)

$I_{IR} = g_{IR}(J_K/50)H(J_K/50)\ \mu A/cm^2$ (K^+ IR current)

$I_{DR} = g_{DR}(J_K/50)\ \mu A/cm^2$ (K^+ DR current)

$I_{Na} = g_{Na}(J_{Na}/75)\ \mu A/cm^2$ (Na^+ current)

$H(t) = \begin{cases} 1, t > 0 \\ 0, t < 0 \end{cases}$ (3)

$\sigma = \frac{1}{7}(\exp(Na_o/67.3) - 1)$

$f_1 = (1 + 0.12\exp(-0.1VF/(RT)) + 0.04\sigma\exp(-VF/(RT)))^{-1}$

$\bar{I}_{NaK} = F\bar{J}_{NaK}/((1 + K_{mK}/K_o)^2 \times (1 + K_{mNa}/Na_i)^3)\mu A/cm^2$

$I_{NaK} = \bar{I}_{NaK} f_1\ \mu A/cm^2$ (Na-K exchanger)

$\frac{dh}{dt} = \alpha_h(1 - h) - \beta_h h$ (Na inactivation gating variable)

$\frac{dm}{dt} = \alpha_m(1 - m) - \beta_m m$ (Na activation gating variable)

$\frac{dn}{dt} = \alpha_n(1 - n) - \beta_n n$ (K-DR activation gating variable)

$\frac{dh^K}{dt} = (h_\infty^K - h^K)/\tau_{h^K}$ (K-DR inactivation gating variable)

$\frac{dS}{dt} = (S_\infty - S)/\tau_S$ (Na very slow inactivation gating variable)

and the function Z_t is defined by

$\left[\frac{dh_t}{dt}, \frac{dm_t}{dt}, \frac{dn_t}{dt}, \frac{dh_t^K}{dt}, \frac{dS_t}{dt}, I_t^{Cl}, I_t^{IR}, I_t^{DR}, I_t^{Na}, I_t^{NaK}\right] = Z_t(v_t, h_t, m_t, n_t, h_t^K, S_t, K_i, K_t, Na_i, Na_t)$

$\left[\frac{dh_t}{dt}, \frac{dm_t}{dt}, \frac{dn_t}{dt}, \frac{dh_t^K}{dt}, \frac{dS}{dt}, I^{Cl}, I^{IR}, I^{DR}, I^{Na}, I^{NaK}\right] = Z_s(v_t, h_t, m_t, n_t, h_t^K, S_t, K_i, K_t, Na_i, Na_t)$

$\eta_{Cl} = 0.1$ (Cl channel density in t-tubules relative to sarcolemma)

$\eta_{IR} = 1.0$ (K-IR channel density in t-tubules relative to sarcolemma)

$\eta_{DR} = 0.45$ (K-DR channel density in t-tubules relative to sarcolemma)

$\eta_{Na} = 0.1$ (Na channel density in t-tubules relative to sarcolemma)

$\eta_{NaK} = 0.1$ (Na/K exchanger density in t-tubules relative to sarcolemma)

$I_t^{Cl} = \eta_{Cl} I^{Cl}\ \mu A/cm^2$

$I_t^{IR} = \eta_{IR} I^{IR}\ \mu A/cm^2$

$I_t^{DR} = \eta_{DR} I^{DR}\ \mu A/cm^2$

$I_t^{Na} = \eta_{Na} I^{Na}\ \mu A/cm^2$

$I_t^{NaK} = \eta_{NaK} I^{NaK}\ \mu A/cm^2$

PDE model

$\bar{G}_L = G_L \tau f_T$ mS/cm

$D_s = 1000 \times (R_s/(2(R_i + R_e)))/C_m$ cm^2/ms

$D_t = \bar{G}_L/(C_m f_T/\xi)$ cm^2/ms

$\left[\frac{dh_s}{dt}, \frac{dm_s}{dt}, \frac{dn_s}{dt}, \frac{dh_s^K}{dt}, \frac{dS_s}{dt}, I^{Cl}, I^{IR}, I^{DR}, I^{Na}, I^{NaK}\right] = Z_s(v_s, h_s, m_s, n_s, h_s^K, S_s, K_i, K_e, Na_i, Na_e)$

$I_s^{ionic} = I^{Cl} + I^{IR} + I^{DR} + I^{Na} + I^{NaK} - I_{app}(t; t_{app})\ \mu A/cm^2$ (sarcolemma current)

$\left[\frac{dh_t}{dt}, \frac{dm_t}{dt}, \frac{dn_t}{dt}, \frac{dh_t^K}{dt}, \frac{dS_t}{dt}, I_t^{Cl}, I_t^{IR}, I_t^{DR}, I_t^{Na}, I_t^{NaK}\right] = Z_t(v_t, h_t, m_t, n_t, h_t^K, S_t, K_i, K_t, Na_i, Na_t)$

$I_t^{ionic} = I_t^{Cl} + I_t^{IR} + I_t^{DR} + I_t^{Na} + I_t^{NaK}\ \mu A/cm^2$ (t-tubule current)

$\frac{\partial v_t(r,x)}{\partial t} + \frac{I_t^{ionic}}{C_m} = D_t\left(\frac{\partial^2 v_t}{\partial r^2} + \frac{1}{r}\frac{\partial v_t}{\partial r}\right)$ mV/ms (t-tubule membrane voltage)

$\left.\frac{\partial v_t}{\partial r}\right|_{r=0} = 0$

$\frac{dO_1}{dt} = k_L/fC_1 - k_{Lm}fO_1 + 4k_C/fO_0 - fk_{Cm}O_1 - 3k_C/fO_1 + 2fk_{Cm}O_2$

$\frac{dC_2}{dt} = 3k_C C_1 - 2k_{Cm}C_2 - k_L/f^2C_2 + k_{Lm}f^2O_2 - 2k_C C_2 + 3k_{Cm}C_3$

$\frac{dO_2}{dt} = 3k_C/fO_1 - 2fk_{Cm}O_2 + k_L/f^2C_2 - k_{Lm}f^2O_2 - 2k_C/fO_2 + 3fk_{Cm}O_3$

$\frac{dC_3}{dt} = 2k_C C_2 - 3k_{Cm}C_3 - k_L/f^3C_3 + k_{Lm}f^3O_3 - k_C C_3 + 4k_{Cm}C_4$

$\frac{dO_3}{dt} = k_L/f^3C_3 - k_{Lm}f^3O_3 + 2k_C/fO_2 - 3fk_{Cm}O_3 - k_C/fO_3 + 4fk_{Cm}O_4$

$\frac{dC_4}{dt} = k_C C_3 - 4k_{Cm}C_4 - k_L/f^4C_4 + k_{Lm}f^4O_4$

$\frac{dO_4}{dt} = k_C/fO_3 - 4fk_{Cm}O_4 + k_L/f^4C_4 \text{-} k_{Lm}f^4O_4$

$I_T = 1000 \times [v_s(x) - v_t(r = R_s, x)]/R_a = \bar{G}_L \left.\frac{\partial v_t}{\partial r}\right|_{r=R_s}$ $\mu A/cm^2$ (access current)

$\frac{\partial v_s(x)}{\partial t} + \frac{I_s^{ionic}}{C_m} = D_s\frac{\partial^2 v_s}{\partial x^2}$, mV/ms (sarcolemma membrane voltage)

$\left.\frac{\partial v_s}{\partial x}\right|_{x=0} = 0, \left.\frac{\partial v_s}{\partial x}\right|_{x=l_s} = 0$

$\frac{\partial K_t(r,x)}{\partial t} = D_K\left(\frac{\partial^2 K_t}{\partial r^2} + \frac{1}{r}\frac{\partial K_t}{\partial r}\right) + (I_t^{IR} + I_t^{DR} - 2I_t^{NaK} + I_{rest}^K)/(1000F\xi)$

$\left.\frac{\partial K_t}{\partial r}\right|_{r=0} = 0$

$K_t(r = R_s, x) = K_e(x)$ mM/ms (t-tubule $[K^+]$)

$\frac{\partial Na_t(r,x)}{\partial t} = D_{Na}\left(\frac{\partial^2 Na_t}{\partial r^2} + \frac{1}{r}\frac{\partial Na_t}{\partial r}\right) + (I_t^{Na} + 3I_t^{NaK} + I_{rest}^{Na})/(1000F\xi)$

$\left.\frac{\partial Na_t}{\partial r}\right|_{r=0} = 0$

$Na_t(r = R_s, x) = Na_e(x)$ mM/ms (t-tubule $[Na^+]$)

$\frac{\partial K_e(x)}{\partial t} = (I^{IR} + I^{DR} - 2I^{NaK} + I_{rest}^K)/(1000F\xi_3)$ mM/ms (extracellular $[K^+]$)

$\frac{\partial Na_e(x)}{\partial t} = (I^{Na} + 3I^{NaK} + I_{rest}^{Na})/(1000F\xi_3)$ mM/ms (extracellular $[Na^+]$)

$\frac{\partial K_i(r,x)}{\partial t} = -f_T(I_t^{IR} + I_t^{DR} - 2I_t^{NaK} + I_{rest}^K)/(1000F\xi)$ mM/ms (intracellular $[K^+]$)

$\frac{\partial Na_i(r,x)}{\partial t} = -f_T(I_t^{Na} + 3I_t^{NaK} + I_{rest}^{Na})/(1000F\xi)$ mM/ms (intracellular $[Na^+]$)

Ca^{2+} release from RyR's

$k_C = 0.5\alpha_1\exp((v_t - \nabla/(8K))$ (activation rate between closed states)

$k_{Cm} = 0.5\alpha_1\exp(-(v_t - \nabla/(8K))$ (inactivation rate between closed states)

$\frac{dC_0}{dt} = -k_L C_0 + k_{Lm}O_0 - 4k_C C_0 + k_{Cm}C_1$ ($C_0 - C_4$ are RyR closed states)

$\frac{dO_0}{dt} = +k_L C_0 - k_{Lm}O_0 - 4k_C/fO_0 + fk_{Cm}O_1$ ($O_0 - O_4$ are RyR open states)

$\frac{dC_1}{dt} = +4k_C C_0 - k_{Cm}C_1 - k_L/fC_1 + k_{Lm}fO_1 - 3k_C C_1 + 2k_{Cm}C_2$

Model on cellular level

Model overview

- System of ordinary differential equations with 56 variables
- Input: I_{Stim}, Output: F

Two compartment model

$$I_T = (v_s - v_t)/R_a 10^3\ \mu A/cm^2 \text{ (access current)}$$
$$\frac{dv_s}{dt} = -\left(I_s^{ionic} + I_T\right)/C_m \text{ mV/ms (sarcolemma membrane voltage)}$$
$$\frac{dv_t}{dt} = -\left(I_t^{ionic} - I_T/\gamma\right)/C_m \text{ mV/ms (t - tubule membrane voltage)}$$
$$\left[\frac{dh_s}{dt}, \frac{dm_s}{dt}, \frac{dn_s}{dt}, \frac{dh_s^K}{dt}, \frac{dS_s}{dt}, I^{Cl}, I^{IR}, I^{DR}, I^{Na}, I^{NaK}\right] = Z_s\left(v_s, h_s, m_s, n_s, h_s^K, S_s, K_i, K_e, Na_i, Na_e\right)$$
$$I_s^{ionic} = I^{Cl} + I^{IR} + I^{DR} + I^{Na} + I^{NaK} - I_{app}(t; t_{app})\ \mu A/cm^2 \text{ (sarcolemma current)}$$
$$\left[\frac{dh_t}{dt}, \frac{dm_t}{dt}, \frac{dn_t}{dt}, \frac{dh_t^K}{dt}, \frac{dS_t}{dt}, I_t^{Cl}, I_t^{IR}, I_t^{DR}, I_t^{Na}, I_t^{NaK}\right] = Z_t\left(v_t, h_t, m_t, n_t, h_t^K, S_t, K_i, K_t, Na_i, Na_t\right)$$
$$I_t^{ionic} = I_t^{Cl} + I_t^{IR} + I_t^{DR} + I_t^{Na} + I_t^{NaK} \mu A/cm^2 \text{ (t - tubule current)} \tag{1}$$
$$\frac{dK_i}{dt} = -f_T\left(I_t^{IR} + I_t^{DR} - 2I_t^{NaK} + I_{rest}^K\right)/(1000F\xi) - \left(I^{IR} + I^{DR} - 2I^{NaK} + I_{rest}^K\right)/(1000F\xi_2) \text{ mM/ms (intracellular } [K^+])$$
$$\frac{dK_t}{dt} = \left(I_t^{IR} + I_t^{DR} - 2I_t^{NaK} + I_{rest}^K\right)/(1000F\xi) - (K_t - K_e)/\tau_K \text{ mM/ms (t - tubule } [K^+])$$
$$\frac{dK_e}{dt} = \left(I^{IR} + I^{DR} - 2I^{NaK} + I_{rest}^K\right)/(1000F\xi_3) + (K_t - K_e)/\tau_{K_2} \text{ mM/ms (interstitial } [K^+])$$
$$dNa_i/dt = -f_T\left(I_t^{Na} + 3I_t^{NaK} + I_{rest}^{Na}\right)/(1000F\xi) - \left(I^{Na} + 3I^{NaK} + I_{rest}^{Na}\right)/(1000F\xi_2) \text{ mM/ms (intracellular } [Na^+])$$
$$dNa_t/dt = \left(I_t^{Na} + 3I_t^{NaK} + I_{rest}^{Na}\right)/(1000F\xi) - (Na_t - Na_e)/\tau_{Na} \text{ mM/ms (t - tubule } [Na^+])$$
$$dNa_e/dt = \left(I^{Na} + 3I^{NaK} + I_{rest}^{Na}\right)/(1000F\xi_3) + (Na_t - Na_e)/\tau_{Na_2} \text{ mM/ms (interstitial } [Na^+])$$

where the function Z_s is defined by

$$\left[\frac{dh}{dt}, \frac{dm}{dt}, \frac{dn}{dt}, \frac{dh^K}{dt}, \frac{dS}{dt}, I^{Cl}, I^{IR}, I^{DR}, I^{Na}, I^{NaK}\right]$$
$$= Z_s(V, h, m, n, h^K, S, K_i, K_o, Na_i, Na_o)$$
$$\alpha_h = \bar{\alpha}_h \exp(-(V - V_h)/K_{\alpha_h})\ ms^{-1}$$
$$\beta_h = \bar{\beta}_h/(1 + \exp(-(V - V_h)/K_{\beta_h}))\ ms^{-1}$$
$$\alpha_m = \bar{\alpha}_m(V - V_m)/(1 - \exp(-(V - V_m)/K_{\alpha_m}))\ ms^{-1}$$
$$\beta_m = \bar{\beta}_m \exp(-(V - V_m)/K_{\beta_m})\ ms^{-1}$$
$$\alpha_n = \bar{\alpha}_n(V - V_n)/(1 - \exp(-(V - V_n)/K_{\alpha_n}))\ ms^{-1}$$
$$\beta_n = \bar{\beta}_n \exp(-(V - V_n)/K_{\beta_n})\ ms^{-1}$$
$$h_\infty^K = 1/(1 + \exp((V - V_{h_K^\infty})/A_{h_K^\infty}))$$
$$S^\infty = 1/(1 + \exp((V - V_{S^\infty})/A_{S^\infty}))$$
$$\tau_{h^K} = 1000 \times \exp(-(V + 40)/25.75)\ ms$$
$$\tau_S = 8571/(0.2 + 5.65((V + V_\tau)/100)^2)\ ms$$
$$a = 1/(1 + \exp((V - V_a)/A_a)) \tag{2}$$

$$E_K = RT/(+F)\log(K_o/K_i)\ mV\ (K^+ \text{ Nernst potential})$$
$$E_{Cl} = E_K(Cl^- \text{ Nernst potential})$$
$$E_{Na} = RT/(+F)\log(Na_o/Na_i)\ mV\ (Na^+ \text{ Nernst potential})$$
$$Cl_i = (128 + 5 \times 5.7)/(5 + \exp((-FE_K)/(RT))) \quad (\text{Intracellular } [Cl^-])$$
$$Cl_o = (128 + 5 \times 5.7) - 5Cl_i \quad (\text{Extracellular } [Cl^-])$$

$$J_{Cl} = V(Cl_i - Cl_o\exp((FV)/(RT)))/(1 - \exp((FV)/(RT)))$$
$$J_K = V(K_i - K_o\exp(-(FV)/(RT)))/(1 - \exp(-(FV)/(RT)))$$
$$J_{Na} = V(Na_i - Na_o\exp(-(FV)/(RT)))/(1 - \exp(-(FV)/(RT)))$$
$$g_{Cl} = \bar{g}_{Cl} a^4\ mS/cm^2$$
$$K_R = K_o\exp(-\delta E_K F/(RT))\ mM$$
$$\bar{g}_{IR} = G_K K_R^2/(K_K + K_R^2)\ mS/cm^2$$
$$y = 1 - (1 + (K_S/(S_i^2\exp(2(1-\delta)VF/(RT)))) \times (1 + K_R^2/K_K))^{-1}$$
$$g_{IR} = \bar{g}_{IR} y\ mS/cm^2$$
$$g_{DR} = \bar{g}_K n^4 h^K\ mS/cm^2$$
$$g_{Na} = \bar{g}_{Na} m^3 hS\ mS/cm^2$$
$$I_{Cl} = g_{Cl}(J_{Cl}/45)\ \mu A/cm^2\ (Cl^- \text{ current})$$
$$I_{IR} = g_{IR}(J_K/50)H(J_K/50)\ \mu A/cm^2\ (K^+ \text{ IR current})$$
$$I_{DR} = g_{DR}(J_K/50)\ \mu A/cm^2\ (K^+ \text{ DR current})$$
$$I_{Na} = g_{Na}(J_{Na}/75)\ \mu A/cm^2\ (Na^+ \text{ current})$$
$$H(t) = \begin{cases} 1, t > 0 \\ 0, t < 0 \end{cases} \tag{3}$$
$$\sigma = \frac{1}{7}(\exp(Na_o/67.3) - 1)$$
$$f_1 = (1 + 0.12\exp(-0.1VF/(RT)) + 0.04\sigma\exp(-VF/(RT)))^{-1}$$
$$\bar{I}_{NaK} = F\bar{J}_{NaK}/((1 + K_{mK}/K_o)^2 \times (1 + K_{mNa}/Na_i)^3)\mu A/cm^2$$
$$I_{NaK} = \bar{I}_{NaK} f_1\ \mu A/cm^2 \text{ (Na-K exchanger)}$$

$$\frac{dh}{dt} = \alpha_h(1 - h) - \beta_h h \text{ (Na inactivation gating variable)}$$
$$\frac{dm}{dt} = \alpha_m(1 - m) - \beta_m m \text{ (Na activation gating variable)}$$
$$\frac{dn}{dt} = \alpha_n(1 - n) - \beta_n n \text{ (K-DR activation gating variable)}$$
$$\frac{dh^K}{dt} = (h_\infty^K - h^K)/\tau_{h^K} \text{ (K-DR inactivation gating variable)}$$
$$\frac{dS}{dt} = (S_\infty - S)/\tau_S \text{ (Na very slow inactivation gating variable)}$$

and the function Z_t is defined by

$$\left[\frac{dh_t}{dt}, \frac{dm_t}{dt}, \frac{dn_t}{dt}, \frac{dh_t^K}{dt}, \frac{dS_t}{dt}, I_t^{Cl}, I_t^{IR}, I_t^{DR}, I_t^{Na}, I_t^{NaK}\right] = Z_t(v_t, h_t, m_t, n_t, h_t^K, S_t, K_i, K_t, Na_i, Na_t)$$
$$\left[\frac{dh_t}{dt}, \frac{dm_t}{dt}, \frac{dn_t}{dt}, \frac{dh_t^K}{dt}, \frac{dS}{dt}, I^{Cl}, I^{IR}, I^{DR}, I^{Na}, I^{NaK}\right] = Z_s(v_t, h_t, m_t, n_t, h_t^K, S_t, K_i, K_t, Na_i, Na_t)$$
$$\eta_{Cl} = 0.1 \text{ (Cl channel density in t-tubules relative to sarcolemma)}$$
$$\eta_{IR} = 1.0 \text{ (K-IR channel density in t-tubules relative to sarcolemma)}$$
$$\eta_{DR} = 0.45 \text{ (K-DR channel density in t-tubules relative to sarcolemma)}$$
$$\eta_{Na} = 0.1 \text{ (Na channel density in t-tubules relative to sarcolemma)}$$
$$\eta_{NaK} = 0.1 \text{ (Na/K exchanger density in t-tubules relative to sarcolemma)}$$
$$I_t^{Cl} = \eta_{Cl} I^{Cl}\ \mu A/cm^2$$
$$I_t^{IR} = \eta_{IR} I^{IR}\ \mu A/cm^2$$
$$I_t^{DR} = \eta_{DR} I^{DR}\ \mu A/cm^2$$
$$I_t^{Na} = \eta_{Na} I^{Na}\ \mu A/cm^2$$
$$I_t^{NaK} = \eta_{NaK} I^{NaK}\ \mu A/cm^2$$

PDE model

$$\bar{G}_L = G_L \tau f_T \quad mS/cm$$
$$D_s = 1000 \times (R_s/(2(R_i + R_e)))/C_m \quad cm^2/ms$$
$$D_t = \bar{G}_L/(C_m f_T/\xi) \quad cm^2/ms$$
$$\left[\frac{dh_s}{dt}, \frac{dm_s}{dt}, \frac{dn_s}{dt}, \frac{dh_s^K}{dt}, \frac{dS_s}{dt}, I^{Cl}, I^{IR}, I^{DR}, I^{Na}, I^{NaK}\right] = Z_s(v_s, h_s, m_s, n_s, h_s^K, S_s, K_i, K_e, Na_i, Na_e)$$
$$I_s^{ionic} = I^{Cl} + I^{IR} + I^{DR} + I^{Na} + I^{NaK} - I_{app}(t; t_{app})\ \mu A/cm^2 \text{ (sarcolemma current)}$$
$$\left[\frac{dh_t}{dt}, \frac{dm_t}{dt}, \frac{dn_t}{dt}, \frac{dh_t^K}{dt}, \frac{dS_t}{dt}, I_t^{Cl}, I_t^{IR}, I_t^{DR}, I_t^{Na}, I_t^{NaK}\right] = Z_t(v_t, h_t, m_t, n_t, h_t^K, S_t, K_i, K_t, Na_i, Na_t)$$
$$I_t^{ionic} = I_t^{Cl} + I_t^{IR} + I_t^{DR} + I_t^{Na} + I_t^{NaK}\ \mu A/cm^2 \text{ (t-tubule current)}$$
$$\frac{\partial v_t(r,x)}{\partial t} + \frac{I_t^{ionic}}{C_m} = D_t\left(\frac{\partial^2 v_t}{\partial r^2} + \frac{1}{r}\frac{\partial v_t}{\partial r}\right) \text{ mV/ms (t-tubule membrane voltage)}$$
$$\left.\frac{\partial v_t}{\partial r}\right|_{r=0} = 0$$

$$\frac{dO_1}{dt} = k_L/fC_1 - k_{Lm}fO_1 + 4k_C/fO_0 - fk_{Cm}O_1 - 3k_C/fO_1 + 2fk_{Cm}O_2$$
$$\frac{dC_2}{dt} = 3k_C C_1 - 2k_{Cm}C_2 - k_L/f^2C_2 + k_{Lm}f^2O_2 - 2k_C C_2 + 3k_{Cm}C_3$$
$$\frac{dO_2}{dt} = 3k_C/fO_1 - 2fk_{Cm}O_2 + k_L/f^2C_2 - k_{Lm}f^2O_2 - 2k_C/fO_2 + 3fk_{Cm}O_3$$
$$\frac{dC_3}{dt} = 2k_C C_2 - 3k_{Cm}C_3 - k_L/f^3C_3 + k_{Lm}f^3O_3 - k_C C_3 + 4k_{Cm}C_4$$
$$\frac{dO_3}{dt} = k_L/f^3C_3 - k_{Lm}f^3O_3 + 2k_C/fO_2 - 3fk_{Cm}O_3 - k_C/fO_3 + 4fk_{Cm}O_4$$
$$\frac{dC_4}{dt} = k_C C_3 - 4k_{Cm}C_4 - k_L/f^4C_4 + k_{Lm}f^4O_4$$
$$\frac{dO_4}{dt} = k_C/fO_3 - 4fk_{Cm}O_4 + k_L/f^4C_4\text{-}k_{Lm}f^4O_4$$

$$I_T = 1000 \times [v_s(x) - v_t(r = R_s, x)]/R_a = \bar{G}_L\left.\frac{\partial v_t}{\partial r}\right|_{r=R_s}\ \mu A/cm^2 \text{ (access current)}$$
$$\frac{\partial v_s(x)}{\partial t} + \frac{I_s^{ionic}}{C_m} = D_s\frac{\partial^2 v_s}{\partial x^2}, \text{ mV/ms (sarcolemma membrane voltage)}$$
$$\left.\frac{\partial v_s}{\partial x}\right|_{x=0} = 0, \left.\frac{\partial v_s}{\partial x}\right|_{x=l_s} = 0$$
$$\frac{\partial K_t(r,x)}{\partial t} = D_K\left(\frac{\partial^2 K_t}{\partial r^2} + \frac{1}{r}\frac{\partial K_t}{\partial r}\right) + (I_t^{IR} + I_t^{DR} - 2I_t^{NaK} + I_{rest}^K)/(1000F\xi)$$
$$\left.\frac{\partial K_t}{\partial r}\right|_{r=0} = 0$$
$$K_t(r = R_s, x) = K_e(x) \text{ mM/ms (t-tubule } [K^+])$$
$$\frac{\partial Na_t(r,x)}{\partial t} = D_{Na}\left(\frac{\partial^2 Na_t}{\partial r^2} + \frac{1}{r}\frac{\partial Na_t}{\partial r}\right) + (I_t^{Na} + 3I_t^{NaK} + I_{rest}^{Na})/(1000F\xi)$$
$$\left.\frac{\partial Na_t}{\partial r}\right|_{r=0} = 0$$
$$Na_t(r = R_s, x) = Na_e(x) \text{ mM/ms (t-tubule } [Na^+])$$
$$\frac{\partial K_e(x)}{\partial t} = (I^{IR} + I^{DR} - 2I^{NaK} + I_{rest}^K)/(1000F\xi_3) \text{ mM/ms (extracellular } [K^+])$$
$$\frac{\partial Na_e(x)}{\partial t} = (I^{Na} + 3I^{NaK} + I_{rest}^{Na})/(1000F\xi_3) \text{ mM/ms (extracellular } [Na^+])$$
$$\frac{\partial K_i(r,x)}{\partial t} = -f_T(I_t^{IR} + I_t^{DR} - 2I_t^{NaK} + I_{rest}^K)/(1000F\xi) \text{ mM/ms (intracellular } [K^+])$$
$$\frac{\partial Na_i(r,x)}{\partial t} = -f_T(I_t^{Na} + 3I_t^{NaK} + I_{rest}^{Na})/(1000F\xi) \text{ mM/ms (intracellular } [Na^+])$$

Ca^{2+} release from RyR's

$$k_C = 0.5\alpha_1\exp((v_t - \nabla/(8K)) \text{ (activation rate between closed states)}$$
$$k_{Cm} = 0.5\alpha_1\exp(-(v_t - \nabla/(8K)) \text{ (inactivation rate between closed states)}$$
$$\frac{dC_0}{dt} = -k_L C_0 + k_{Lm}O_0 - 4k_C C_0 + k_{Cm}C_1 \text{ } (C_0 - C_4 \text{ are RyR closed states})$$
$$\frac{dO_0}{dt} = +k_L C_0 - k_{Lm}O_0 - 4k_C/fO_0 + fk_{Cm}O_1 \text{ } (O_0 - O_4 \text{ are RyR open states})$$
$$\frac{dC_1}{dt} = +4k_C C_0 - k_{Cm}C_1 - k_L/fC_1 + k_{Lm}fO_1 - 3k_C C_1 + 2k_{Cm}C_2$$

Propagation of electrical signals

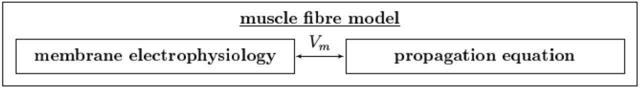
Overview

- Model the propagation of electric stimuli along the muscle fibres
- Extend the existing local model
- Current can not only pass through membrane but also flow along the fibre

Propagation of electrical signals

Overview

- Model the propagation of electric stimuli along the muscle fibres
- Extend the existing local model

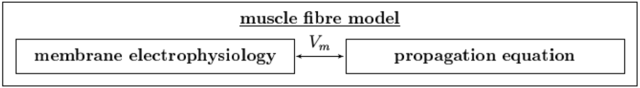


- Current can not only pass through membrane but also flow along the fibre

Propagation of electrical signals

Overview

- Model the propagation of electric stimuli along the muscle fibres
- Extend the existing local model

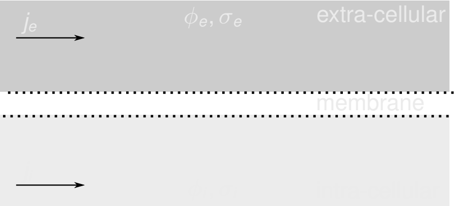


- Current can not only pass through membrane but also flow along the fibre

Propagation of electrical signals

Model setting

- 2 domains: intra-, extracellular
- Homogenised: domains occupy the same space
- Electric potentials ϕ_i, ϕ_e, conductivities σ_i, σ_e, membrane voltage $V_m = \phi_i - \phi_e$



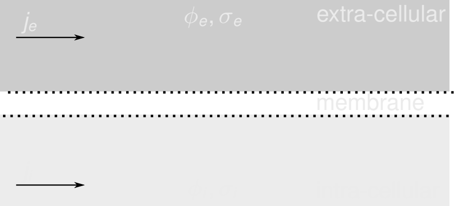
- Current density described by potential:

$$j_e = -\sigma_e \frac{\partial \phi_e}{\partial x}, \quad j_i = -\sigma_i \frac{\partial \phi_i}{\partial x}$$

Propagation of electrical signals

Model setting

- 2 domains: intra-, extracellular
- Homogenised: domains occupy the same space
- Electric potentials ϕ_i, ϕ_e, conductivities σ_i, σ_e, membrane voltage $V_m = \phi_i - \phi_e$



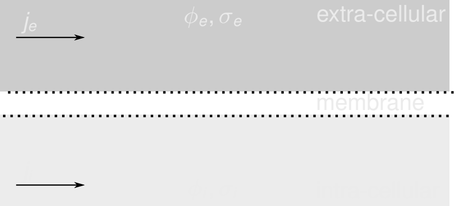
- Current density described by potential:

$$j_e = -\sigma_e \frac{\partial \phi_e}{\partial x}, \quad j_i = -\sigma_i \frac{\partial \phi_i}{\partial x}$$

Propagation of electrical signals

Model setting

- 2 domains: intra-, extracellular
- Homogenised: domains occupy the same space
- Electric potentials ϕ_i, ϕ_e, conductivities σ_i, σ_e, membrane voltage $V_m = \phi_i - \phi_e$



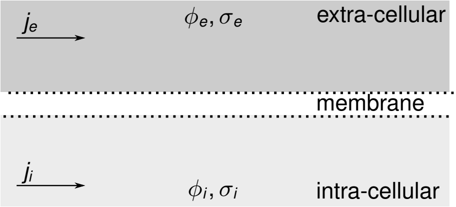
- Current density described by potential:

$$j_e = -\sigma_e \frac{\partial \phi_e}{\partial x}, \quad j_i = -\sigma_i \frac{\partial \phi_i}{\partial x}$$

Propagation of electrical signals

Model setting

- 2 domains: intra-, extracellular
- Homogenised: domains occupy the same space
- Electric potentials ϕ_i, ϕ_e, conductivities σ_i, σ_e, membrane voltage $V_m = \phi_i - \phi_e$

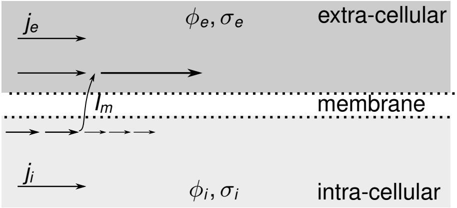


- Current density described by potential:

$$j_e = -\sigma_e \frac{\partial \phi_e}{\partial x}, \quad j_i = -\sigma_i \frac{\partial \phi_i}{\partial x}$$

Propagation of electrical signals

Model setting



- Conservation of charges: local changes in current density affect other domain

$$\frac{\partial}{\partial x} j_i = -\frac{\partial}{\partial x} j_e$$

$$\Leftrightarrow \quad \frac{\partial}{\partial x}\left(\sigma_i \frac{\partial \phi_i}{\partial x}\right) = -\frac{\partial}{\partial x}\left(\sigma_e \frac{\partial \phi_e}{\partial x}\right) \quad = I_m$$

$$= A_m\left(C_m \frac{\partial V_m}{\partial t} + I_{ionic}(V_m)\right)$$

Propagation of electrical signals

- Substitute $V_m = \phi_i - \phi_e$ to eliminate ϕs:

$$\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial V_m}{\partial x}\right) = A_m\left(C_m \frac{\partial V_m}{\partial t} + I_{ionic}(V_m)\right), \qquad \text{with } \sigma_{eff} := \sigma_i || \sigma_e$$

Numerical treatment

- Equation solved for $\dfrac{\partial V_m}{\partial t}$:

$$\frac{\partial V_m}{\partial t} = -\frac{1}{C_m} I_{ionic}(V_m) + \frac{1}{A_m C_m}\frac{\partial}{\partial x}\left(\sigma_{eff}\frac{\partial}{\partial x} V_m\right)$$

- Employ finite differences in t and Godunov operator splitting:

$$\frac{V_m^* - V_m^{(k)}}{\Delta t} = -\frac{1}{C_m} I_{ionic}(V_m^{(k)}),$$

$$\frac{V_m^{(k+1)} - V_m^*}{\Delta t} = \frac{1}{A_m C_m}\frac{\partial}{\partial x}\left(\sigma_{eff}\frac{\partial}{\partial x} V_m^{(k+1)}\right),$$

Propagation of electrical signals

- Substitute $V_m = \phi_i - \phi_e$ to eliminate ϕs:

$$\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial V_m}{\partial x}\right) = A_m\left(C_m\frac{\partial V_m}{\partial t} + I_{ionic}(V_m)\right), \qquad \text{with } \sigma_{eff} := \sigma_i || \sigma_e$$

Numerical treatment

- Equation solved for $\frac{\partial V_m}{\partial t}$:

$$\frac{\partial V_m}{\partial t} = -\frac{1}{C_m} I_{ionic}(V_m) + \frac{1}{A_m C_m}\frac{\partial}{\partial x}\left(\sigma_{eff}\frac{\partial}{\partial x}V_m\right)$$

- Employ finite differences in t and Godunov operator splitting:

$$\frac{V_m^* - V_m^{(k)}}{\Delta t} = -\frac{1}{C_m} I_{ionic}(V_m^{(k)}),$$

$$\frac{V_m^{(k+1)} - V_m^*}{\Delta t} = \frac{1}{A_m C_m}\frac{\partial}{\partial x}\left(\sigma_{eff}\frac{\partial}{\partial x}V_m^{(k+1)}\right).$$

Propagation of electrical signals

- Substitute $V_m = \phi_i - \phi_e$ to eliminate ϕs:

$$\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial V_m}{\partial x}\right) = A_m\left(C_m\frac{\partial V_m}{\partial t} + I_{ionic}(V_m)\right), \qquad \text{with } \sigma_{eff} := \sigma_i||\sigma_e$$

Numerical treatment

- Equation solved for $\dfrac{\partial V_m}{\partial t}$:

$$\frac{\partial V_m}{\partial t} = -\frac{1}{C_m}\,I_{ionic}(V_m) + \frac{1}{A_m\,C_m}\,\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial}{\partial x}\,V_m\right)$$

- Employ finite differences in t and Godunov operator splitting:

$$\frac{V_m^* - V_m^{(k)}}{\Delta t} = -\frac{1}{C_m}\,I_{ionic}(V_m^{(k)}),$$

$$\frac{V_m^{(k+1)} - V_m^*}{\Delta t} = \frac{1}{A_m\,C_m}\,\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial}{\partial x}\,V_m^{(k+1)}\right),$$

Propagation of electrical signals

- Substitute $V_m = \phi_i - \phi_e$ to eliminate ϕs:

$$\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial V_m}{\partial x}\right) = A_m\left(C_m\,\frac{\partial V_m}{\partial t} + I_{ionic}(V_m)\right), \qquad \text{with } \sigma_{eff} := \sigma_i || \sigma_e$$

Numerical treatment

- Equation solved for $\dfrac{\partial V_m}{\partial t}$:

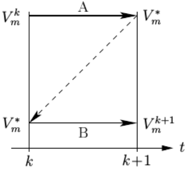
$$\frac{\partial V_m}{\partial t} = -\frac{1}{C_m} I_{ionic}(V_m) + \frac{1}{A_m\,C_m}\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial}{\partial x} V_m\right)$$

- Employ finite differences in t and Godunov operator splitting:

$$\frac{V_m^* - V_m^{(k)}}{\Delta t} = -\frac{1}{C_m} I_{ionic}(V_m^{(k)}),$$

$$\frac{V_m^{(k+1)} - V_m^*}{\Delta t} = \frac{1}{A_m\,C_m}\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial}{\partial x} V_m^{(k+1)}\right),$$

$$V_m^{(k+1)} = V_m^* + \Delta t \cdot \frac{1}{A_m\,C_m}\frac{\partial}{\partial x}\left(\sigma_{eff}\,\frac{\partial}{\partial x} V_m^{(k+1)}\right)$$



Continuum mechanics model

Basics in continuum mechanics

- Consider muscle in reference configuration, $\mathbf{X} \in \mathcal{B}_0$ and in deformed state at time t, $\mathbf{x} \in \mathcal{B}_t$
- Mapping is called placement function $\mathbf{x} = \chi(\mathbf{X}, t)$
- Deformation gradient: $\mathbf{F} = \dfrac{\partial \chi(\mathbf{X}, t)}{\partial \mathbf{X}}$, right Cauchy-Green tensor: $\mathbf{C} = \mathbf{F}^\top \mathbf{F}$
- Strain: Green-Lagrangian strain tensor: $\mathbf{E} = 1/2(\mathbf{C} - \mathbf{I})$
- Stress: 2nd Piola-Kirchhoff stress tensor: $\mathbf{S} = (\det \mathbf{F})\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}, \quad \mathbf{t} = \mathbf{T}\,\mathbf{n}$
- Constitutive assumpions + balance laws to get relation between stresses ($\mathbf{S}$) and strains ($\mathbf{E}$)

Continuum mechanics model

Basics in continuum mechanics

- Consider muscle in reference configuration, $\mathbf{X} \in \mathcal{B}_0$ and in deformed state at time t, $\mathbf{x} \in \mathcal{B}_t$
- Mapping is called placement function $\mathbf{x} = \chi(\mathbf{X}, t)$
- Deformation gradient: $\mathbf{F} = \dfrac{\partial \chi(\mathbf{X}, t)}{\partial \mathbf{X}}$, right Cauchy-Green tensor: $\mathbf{C} = \mathbf{F}^\top \mathbf{F}$
- Strain: Green-Lagrangian strain tensor: $\mathbf{E} = 1/2(\mathbf{C} - \mathbf{I})$
- Stress: 2nd Piola-Kirchhoff stress tensor: $\mathbf{S} = (\det \mathbf{F})\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}, \quad \mathbf{t} = \mathbf{T}\,\mathbf{n}$
- Constitutive assumpions + balance laws to get relation between stresses ($\mathbf{S}$) and strains ($\mathbf{E}$)

Continuum mechanics model

Basics in continuum mechanics

- Consider muscle in reference configuration, $\mathbf{X} \in \mathcal{B}_0$ and in deformed state at time t, $\mathbf{x} \in \mathcal{B}_t$
- Mapping is called placement function $\mathbf{x} = \chi(\mathbf{X}, t)$
- Deformation gradient: $\mathbf{F} = \dfrac{\partial \chi(\mathbf{X}, t)}{\partial \mathbf{X}}$, right Cauchy-Green tensor: $\mathbf{C} = \mathbf{F}^\top \mathbf{F}$
- Strain: Green-Lagrangian strain tensor: $\mathbf{E} = 1/2(\mathbf{C} - \mathbf{I})$
- Stress: 2nd Piola-Kirchhoff stress tensor: $\mathbf{S} = (\det \mathbf{F})\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}, \quad \mathbf{t} = \mathbf{T}\,\mathbf{n}$
- Constitutive assumpions + balance laws to get relation between stresses ($\mathbf{S}$) and strains ($\mathbf{E}$)

Continuum mechanics model

Basics in continuum mechanics

- Consider muscle in reference configuration, $\mathbf{X} \in \mathcal{B}_0$
  and in deformed state at time t, $\mathbf{x} \in \mathcal{B}_t$
- Mapping is called placement function $\mathbf{x} = \chi(\mathbf{X}, t)$
- Deformation gradient: $\mathbf{F} = \dfrac{\partial \chi(\mathbf{X}, t)}{\partial \mathbf{X}}$, right Cauchy-Green tensor: $\mathbf{C} = \mathbf{F}^\top \mathbf{F}$
- Strain: Green-Lagrangian strain tensor: $\mathbf{E} = 1/2(\mathbf{C} - \mathbf{I})$
- Stress: 2nd Piola-Kirchhoff stress tensor: $\mathbf{S} = (\det \mathbf{F})\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}, \quad \mathbf{t} = \mathbf{T}\,\mathbf{n}$
- Constitutive assumpions + balance laws to get relation between stresses ($\mathbf{S}$) and strains ($\mathbf{E}$)

Continuum mechanics model

Basics in continuum mechanics

- Consider muscle in reference configuration, $\mathbf{X} \in \mathcal{B}_0$ and in deformed state at time t, $\mathbf{x} \in \mathcal{B}_t$
- Mapping is called placement function $\mathbf{x} = \chi(\mathbf{X}, t)$
- Deformation gradient: $\mathbf{F} = \dfrac{\partial \chi(\mathbf{X}, t)}{\partial \mathbf{X}}$, right Cauchy-Green tensor: $\mathbf{C} = \mathbf{F}^\top \mathbf{F}$
- Strain: Green-Lagrangian strain tensor: $\mathbf{E} = 1/2(\mathbf{C} - \mathbf{I})$
- Stress: 2nd Piola-Kirchhoff stress tensor: $\mathbf{S} = (\det \mathbf{F})\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}, \quad \mathbf{t} = \mathbf{T}\,\mathbf{n}$
- Constitutive assumpions + balance laws to get relation between stresses ($\mathbf{S}$) and strains ($\mathbf{E}$)

Continuum mechanics model

Basics in continuum mechanics

- Consider muscle in reference configuration, $\mathbf{X} \in \mathcal{B}_0$ and in deformed state at time t, $\mathbf{x} \in \mathcal{B}_t$
- Mapping is called placement function $\mathbf{x} = \chi(\mathbf{X}, t)$
- Deformation gradient: $\mathbf{F} = \dfrac{\partial \chi(\mathbf{X}, t)}{\partial \mathbf{X}}$, right Cauchy-Green tensor: $\mathbf{C} = \mathbf{F}^\top \mathbf{F}$
- Strain: Green-Lagrangian strain tensor: $\mathbf{E} = 1/2(\mathbf{C} - \mathbf{I})$
- Stress: 2nd Piola-Kirchhoff stress tensor: $\mathbf{S} = (\det \mathbf{F})\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}, \quad \mathbf{t} = \mathbf{T}\,\mathbf{n}$
- Constitutive assumpions + balance laws to get relation between stresses ($\mathbf{S}$) and strains ($\mathbf{E}$)

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, "ground matrix" modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\,(tr(\mathbf{C})\mathbf{I} - \mathbf{C})$$

- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

$$\mathbf{S}_{ani} = b_1\,(\lambda_f^{d_1-2} - \lambda_f^{-2})\,\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, "ground matrix" modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\,(tr(\mathbf{C})\mathbf{I} - \mathbf{C})$$

- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

$$\mathbf{S}_{ani} = b_1\,(\lambda_f^{d_1 - 2} - \lambda_f^{-2})\,\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, "ground matrix" modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\,(tr(\mathbf{C})\mathbf{I} - \mathbf{C})$$

- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

$$\mathbf{S}_{ani} = b_1\,(\lambda_f^{d_1-2} - \lambda_f^{-2})\,\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, "ground matrix" modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\,(tr(\mathbf{C})\,\mathbf{I} - \mathbf{C})$$

- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

$$\mathbf{S}_{ani} = b_1\,\left(\lambda_f^{d_1-2} - \lambda_f^{-2}\right)\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, "ground matrix" modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\,(tr(\mathbf{C})\mathbf{I} - \mathbf{C})$$

- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

$$\mathbf{S}_{ani} = b_1\,(\lambda_f^{d_1-2} - \lambda_f^{-2})\,\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, "ground matrix" modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\,(tr(\mathbf{C})\mathbf{I} - \mathbf{C})$$

- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

$$\mathbf{S}_{ani} = b_1\,(\lambda_f^{d_1 - 2} - \lambda_f^{-2})\,\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, "ground matrix" modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\left(tr(\mathbf{C})\,\mathbf{I} - \mathbf{C}\right)$$

- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

$$\mathbf{S}_{ani} = b_1\left(\lambda_f^{d_1-2} - \lambda_f^{-2}\right)\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, ”ground matrix” modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\left(tr(\mathbf{C})\,\mathbf{I} - \mathbf{C}\right)$$

- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

$$\mathbf{S}_{ani} = b_1\left(\lambda_f^{d_1-2} - \lambda_f^{-2}\right)\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Model assumptions of muscle

- Incompressibility, i.e. $\det(\mathbf{F}) = 1$
- Quasi-static condititons: no inertial terms
- No external body forces: no gravity, no contact e.g. to bones
- Isothermal conditions: no balance of energy
- Stress tensor can be additively composed into active and passive part:

$$\mathbf{S} = \mathbf{S}_{pass} + \mathbf{S}_{act} - p\,\mathbf{C}^{-1}$$

Passive stresses

- Modelled as fibre reinforced material with isotropic and anisotropic part:

$$\mathbf{S}_{pass} = \mathbf{S}_{iso} + \mathbf{S}_{ani}$$

- Isotropic part, "ground matrix" modelled by *Mooney-Rivlin* material:

$$\mathbf{S}_{iso} = 2\,c_{10}\,\mathbf{I} + 2\,c_{01}\left(tr(\mathbf{C})\,\mathbf{I} - \mathbf{C}\right)$$

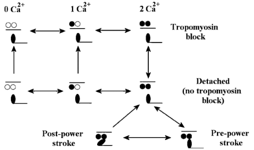
- Anisotropic part acts in fibre direction $\mathbf{a}_0$ with fibre stretch λ_f:

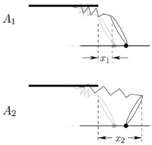
$$\mathbf{S}_{ani} = b_1\left(\lambda_f^{d_1-2} - \lambda_f^{-2}\right)\mathbf{a}_0 \otimes \mathbf{a}_0$$

Continuum mechanics model

Active stresses

- From crossbridge dynamics to a stress tensor field
  ℓ_S: sarcomere length, f_s: stimulation frequency
  $\lambda_f = \ell_S / \ell_S^0$: fibre stretch





$$\frac{\partial x_1}{\partial t} = -\left(f_0 \frac{[D_2]}{[A_1]} + h' \frac{[A_2]}{[A_1]}\right) x_1 + h' \frac{[A_2]}{[A_1]} (x_2 - x_0) + \frac{\dot{\ell}_S}{2}$$

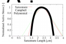
$$\frac{\partial x_2}{\partial t} = -h_0 \frac{[A_1]}{[A_2]} (x_2 - (x_1 + x_0)) + \frac{\dot{\ell}_S}{2}.$$

$$f_0 = \bar{f}\left(1 + \frac{[A_1]}{T_{tot}}\left[\exp\left(\frac{x_1}{x_0}(\nu - 1)\right) - 1\right] + \frac{[A_2]}{T_{tot}}\left[\exp\left(\frac{x_2}{x_0}(\nu - 1)\right) - 1\right]\right),$$

$$g_0 = \bar{g} \exp\left(\vartheta (x_2 - x_0)^2\right),$$

$$B(f_s, \dot{\ell}_s) = [A_1]\, x_1 + [A_2]\, x_2$$

Λ :



$$\gamma = \Lambda(\ell_S) \frac{B(f_s, \dot{\ell}_S) - B(0,0)}{B(f_s^{max}, 0) - B(0,0)}$$

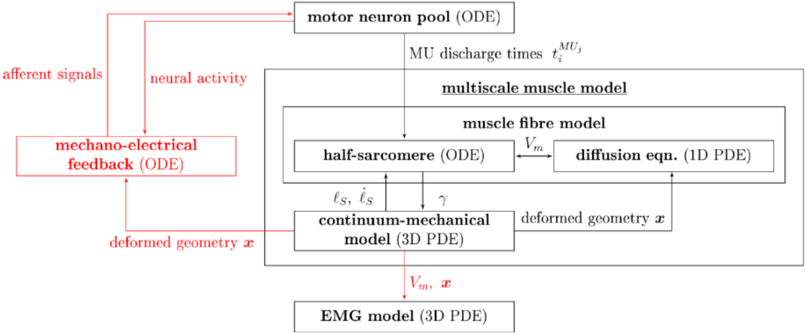
$$\bar{\gamma} = \Gamma(\gamma)$$

$$P^{act} = P^{max}\, \bar{\gamma}(f_s, \lambda_f, \dot{\lambda}_f)$$

$$\mathbf{S}_{act} = \lambda_f^{-1}\, P^{act}\, \mathbf{a}_0 \otimes \mathbf{a}_0$$

The Mathematical Model

The full multiscale model



Timestepping scheme

Discretization and solver types

- ODE: finite differences, forward Euler
- 1D PDE: 1D finite elements, Crank-Nicholson ($\theta = \frac{1}{2}$) $\rightsquigarrow$ nonlinear system, solved using newton search, linear system in each step: GMRES (PETSc)
- 3D PDE: 3D finite elements, backtracking line search (PETSc)
- The solvers have different time steps according to characteristic sizes of the scales
- Typical values: $h^{ODE} = 10^{-4} ms, \quad h^{DEQ} = 50 \cdot 10^{-4} ms, \quad h^{CMM} = 10^{-1} ms$

Timestepping scheme

Discretization and solver types

- ODE: finite differences, forward Euler
- 1D PDE: 1D finite elements, Crank-Nicholson ($\theta = \frac{1}{2}$) $\rightsquigarrow$ nonlinear system, solved using newton search, linear system in each step: GMRES (PETSc)
- 3D PDE: 3D finite elements, backtracking line search (PETSc)
- The solvers have different time steps according to characteristic sizes of the scales
- Typical values: $h^{ODE} = 10^{-4} ms$, $\quad h^{DEQ} = 50 \cdot 10^{-4} ms$, $\quad h^{CMM} = 10^{-1} ms$

Timestepping scheme

Discretization and solver types

- ODE: finite differences, forward Euler
- 1D PDE: 1D finite elements, Crank-Nicholson ($\theta = \frac{1}{2}$) $\rightsquigarrow$ nonlinear system, solved using newton search, linear system in each step: GMRES (PETSc)
- 3D PDE: 3D finite elements, backtracking line search (PETSc)

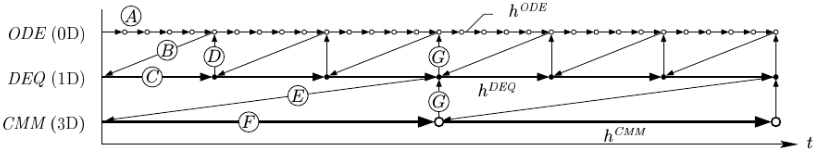
- The solvers have different time steps according to characteristic sizes of the scales

- Typical values: $h^{ODE} = 10^{-4} ms, \quad h^{DEQ} = 50 \cdot 10^{-4} ms, \quad h^{CMM} = 10^{-1} ms$

Timestepping scheme

Discretization and solver types

- ODE: finite differences, forward Euler
- 1D PDE: 1D finite elements, Crank-Nicholson ($\theta = \frac{1}{2}$) $\rightsquigarrow$ nonlinear system, solved using newton search, linear system in each step: GMRES (PETSc)
- 3D PDE: 3D finite elements, backtracking line search (PETSc)
- The solvers have different time steps according to characteristic sizes of the scales

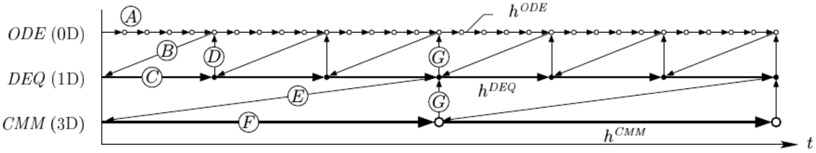


- Typical values: $h^{ODE} = 10^{-4} ms$, $h^{DEQ} = 50 \cdot 10^{-4} ms$, $h^{CMM} = 10^{-1} ms$

Timestepping scheme

Discretization and solver types

- ODE: finite differences, forward Euler
- 1D PDE: 1D finite elements, Crank-Nicholson ($\theta = \frac{1}{2}$) $\rightsquigarrow$ nonlinear system, solved using newton search, linear system in each step: GMRES (PETSc)
- 3D PDE: 3D finite elements, backtracking line search (PETSc)
- The solvers have different time steps according to characteristic sizes of the scales



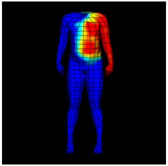
- Typical values: $h^{ODE} = 10^{-4} ms, \quad h^{DEQ} = 50 \cdot 10^{-4} ms, \quad h^{CMM} = 10^{-1} ms$

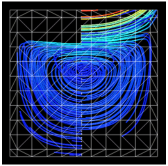
The Computational Framework OpenCMISS

The computational framework OpenCMISS

History

- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

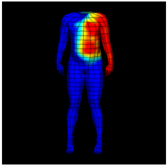
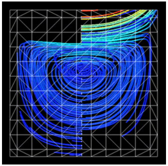




The computational framework OpenCMISS

History

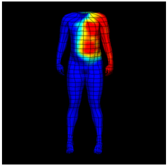
- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

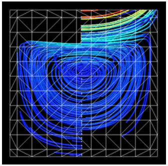
  

The computational framework OpenCMISS

History

- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

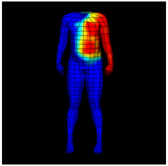


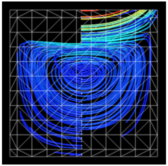


The computational framework OpenCMISS

History

- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

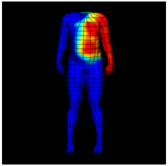


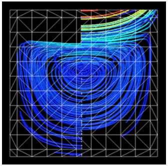


The computational framework OpenCMISS

History

- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

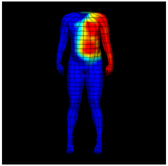


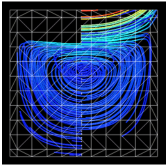


The computational framework OpenCMISS

History

- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

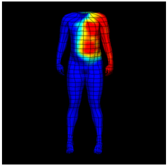


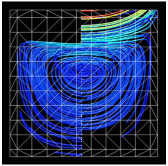


The computational framework OpenCMISS

History

- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

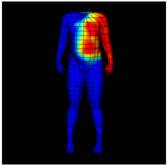


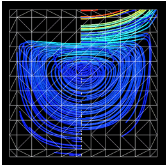


The computational framework OpenCMISS

History

- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

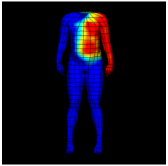


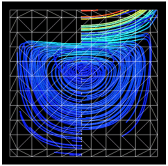


The computational framework OpenCMISS

History

- Framework for simulation of coupled multiphysics simulations for bioengineering problems
- Predecessor CMISS: Continuum Mechanics, Image analysis, System identification and Signal processing
- Development began in 1980 in Auckland, NZ
- First application: heart and lungs
- Programming language: Fortran 90
- Redevelopment since 2005: OpenCMISS
- Objectives: become open source, support MPI parallelisation, adapt new evolving model encoding standards
- Main development is driven by Auckland, more developers and users in the UK, Norway, Spain, Germany, China
- Number of lines of code today (only core): 300k fortran, 2k C++

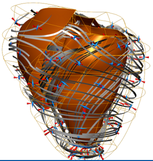


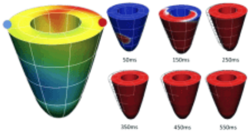


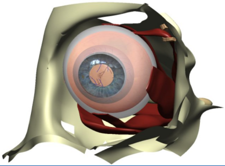
The computational framework OpenCMISS

Features

- Finite elements calculations in different dimensions
- Different general components from which the desired problem can be build: regions with possibly multiple overlayed meshes, basis functions, interpolation methods, solvers
- The structure of control loops can be arbitrarily composed, subloops with fixed time stepping, load increment stepping, conditional break
- Parallelization requirements like ghost cells can be automatically created, MPI communication is transparent from user
- Some external dependencies can be interfaced: PETSc (solvers), SUNDIALS (for DAE solver), Hypre (preconditioners), SLEPc (eigenproblems), ScaLAPACK (linear algebra)
- The XML-based description standards *CellML* and *FieldML* are supported.
- Offline-visualization tools



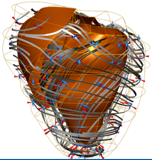


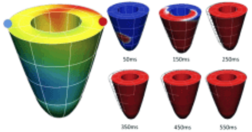


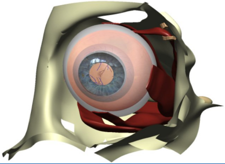
The computational framework OpenCMISS

Features

- Finite elements calculations in different dimensions
- Different general components from which the desired problem can be build: regions with possibly multiple overlayed meshes, basis functions, interpolation methods, solvers
- The structure of control loops can be arbitrarily composed, subloops with fixed time stepping, load increment stepping, conditional break
- Parallelization requirements like ghost cells can be automatically created, MPI communication is transparent from user
- Some external dependencies can be interfaced: PETSc (solvers), SUNDIALS (for DAE solver), Hypre (preconditioners), SLEPc (eigenproblems), ScaLAPACK (linear algebra)
- The XML-based description standards *CellML* and *FieldML* are supported.
- Offline-visualization tools



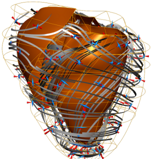


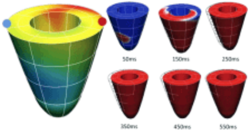


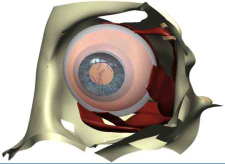
The computational framework OpenCMISS

Features

- Finite elements calculations in different dimensions
- Different general components from which the desired problem can be build: regions with possibly multiple overlayed meshes, basis functions, interpolation methods, solvers
- The structure of control loops can be arbitrarily composed, subloops with fixed time stepping, load increment stepping, conditional break
- Parallelization requirements like ghost cells can be automatically created, MPI communication is transparent from user
- Some external dependencies can be interfaced: PETSc (solvers), SUNDIALS (for DAE solver), Hypre (preconditioners), SLEPc (eigenproblems), ScaLAPACK (linear algebra)
- The XML-based description standards *CellML* and *FieldML* are supported.
- Offline-visualization tools



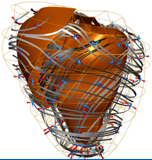
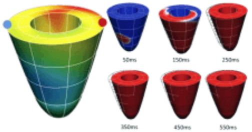
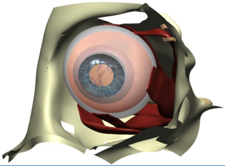




The computational framework OpenCMISS

Features

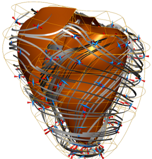
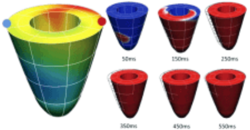
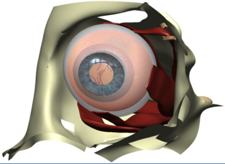
- Finite elements calculations in different dimensions
- Different general components from which the desired problem can be build: regions with possibly multiple overlayed meshes, basis functions, interpolation methods, solvers
- The structure of control loops can be arbitrarily composed, subloops with fixed time stepping, load increment stepping, conditional break
- Parallelization requirements like ghost cells can be automatically created, MPI communication is transparent from user
- Some external dependencies can be interfaced: PETSc (solvers), SUNDIALS (for DAE solver), Hypre (preconditioners), SLEPc (eigenproblems), ScaLAPACK (linear algebra)
- The XML-based description standards *CellML* and *FieldML* are supported.
- Offline-visualization tools

The computational framework OpenCMISS

Features

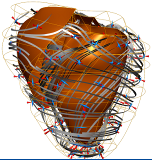
- Finite elements calculations in different dimensions
- Different general components from which the desired problem can be build: regions with possibly multiple overlayed meshes, basis functions, interpolation methods, solvers
- The structure of control loops can be arbitrarily composed, subloops with fixed time stepping, load increment stepping, conditional break
- Parallelization requirements like ghost cells can be automatically created, MPI communication is transparent from user
- Some external dependencies can be interfaced: PETSc (solvers), SUNDIALS (for DAE solver), Hypre (preconditioners), SLEPc (eigenproblems), ScaLAPACK (linear algebra)
- The XML-based description standards *CellML* and *FieldML* are supported.
- Offline-visualization tools

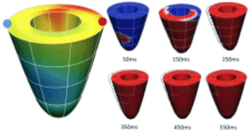




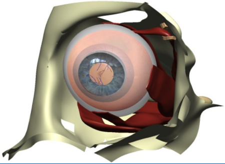
The computational framework OpenCMISS

Features

- Finite elements calculations in different dimensions
- Different general components from which the desired problem can be build: regions with possibly multiple overlayed meshes, basis functions, interpolation methods, solvers
- The structure of control loops can be arbitrarily composed, subloops with fixed time stepping, load increment stepping, conditional break
- Parallelization requirements like ghost cells can be automatically created, MPI communication is transparent from user
- Some external dependencies can be interfaced: PETSc (solvers), SUNDIALS (for DAE solver), Hypre (preconditioners), SLEPc (eigenproblems), ScaLAPACK (linear algebra)
- The XML-based description standards *CellML* and *FieldML* are supported.
- Offline-visualization tools



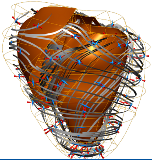


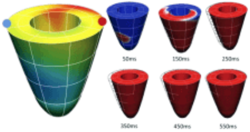


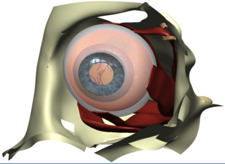
The computational framework OpenCMISS

Features

- Finite elements calculations in different dimensions
- Different general components from which the desired problem can be build: regions with possibly multiple overlayed meshes, basis functions, interpolation methods, solvers
- The structure of control loops can be arbitrarily composed, subloops with fixed time stepping, load increment stepping, conditional break
- Parallelization requirements like ghost cells can be automatically created, MPI communication is transparent from user
- Some external dependencies can be interfaced: PETSc (solvers), SUNDIALS (for DAE solver), Hypre (preconditioners), SLEPc (eigenproblems), ScaLAPACK (linear algebra)
- The XML-based description standards *CellML* and *FieldML* are supported.
- Offline-visualization tools



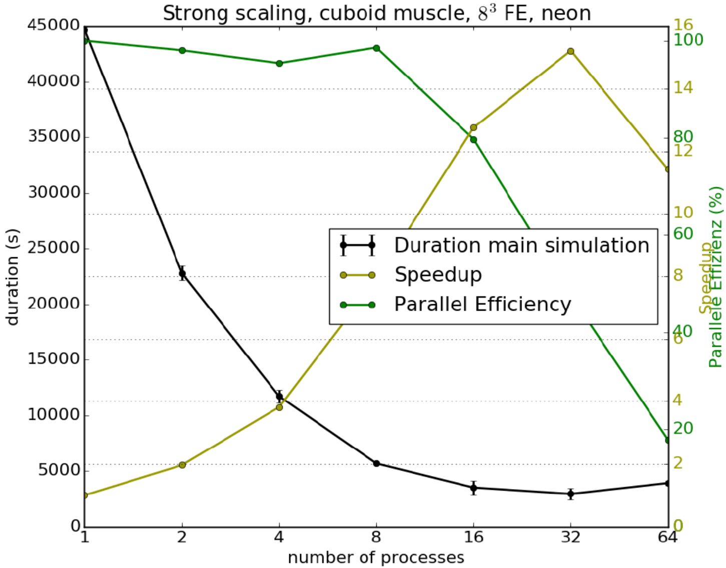




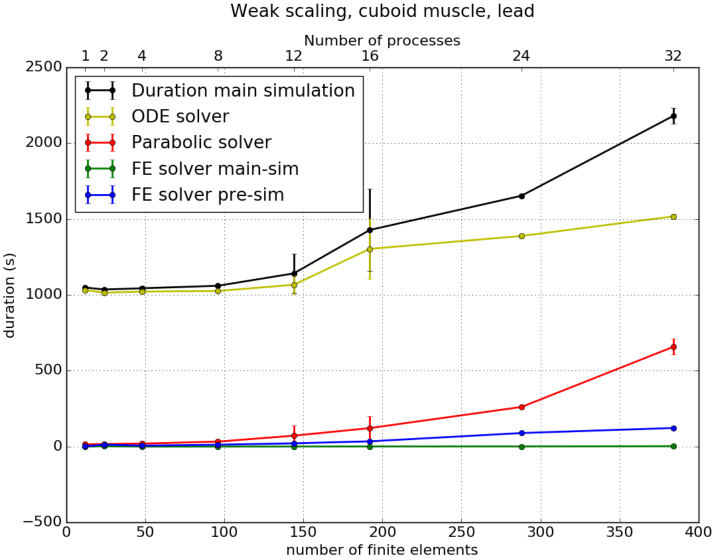
The computational framework OpenCMISS

Simulation

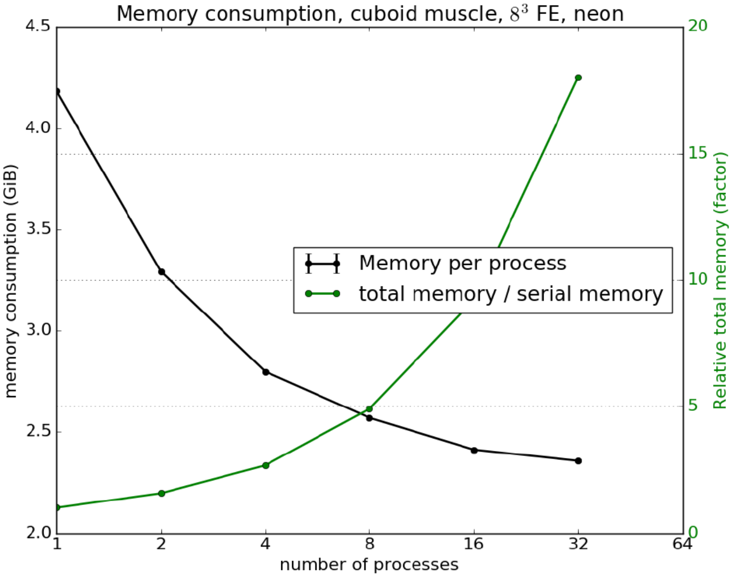
Scaling Properties



Scaling Properties



Scaling Properties



Thank you for your attention!