Guided Learning of Control Graphs for Physics-Based Characters

Libin Liu\textsuperscript{1} Michiel van de Panne\textsuperscript{1} KangKang Yin\textsuperscript{2}

\textsuperscript{1}The University of British Columbia

\textsuperscript{2}National University of Singapore
Why Physics-based Characters?
Existing Solutions

[Coros et al. 2010]  ...  [Lee et al. 2010]

[Al Borno et al. 2013]  ...  [Liu et al. 2012]

[Tan et al. 2014]  ...
Our Goal

Automated Learning Framework

Physics-based controller

Motion Graph

Control Graph
Pipeline

Motion Clip

Open-loop Control

Closed-loop Control

Guided Learning
Key Intuition

Multiple Open-loop Solutions

Regression

Feedback Policy
Key Intuition

Multiple Open-loop Solutions

Guided Learning

Feedback Policy
Key Intuition

Multiple Open-loop Solutions

Guided Learning

Feedback Policy
Key Intuition

Multiple Open-loop Solutions

Guided Learning

Feedback Policy
Control Fragment

A short control unit:

• $\delta t \approx 0.1$ seconds long
• Open-loop control segment $\hat{m}$
• Linear Feedback policy $\pi$

$C : \{\delta t, \hat{m}, \pi\}$
Controller

A chain of control fragments
Open-loop Control Trajectory

\[ C_1 \xrightarrow{\delta t} C_2 \]

\[ \hat{m}_1, \hat{m}_2, \hat{m}_3, \hat{m}_4, \hat{m}_5, \hat{m}_6 \]

Target Poses/Joint Angles

PD-servos
Simulator

Simulation States
Open-loop Control Trajectory

Target Poses/Joint Angles

Simulation States

Simulation States
Open-loop Control Trajectory

\[ \Delta t \]

Target Poses/
Joint Angles

\[ \pi_1, \pi_2, \pi_3, \pi_4, \pi_5, \pi_6 \]

Simulation
States

\[ s_0, s_1 \]

perturbation

\[ C_1, C_2 \]
Feedback Policy

\[ a_1 = M_1 s_1 + \hat{a}_1 \]

\[ \pi_1 \]

Target Poses/ Joint Angles

Feedback Actions

Simulation States

\( \delta t \)

offset

t

\( t \)

15
Feedback Policy

\[ \delta t \]

Target Poses/Joint Angles

Feedback Actions

Simulation States

\[ s_0 \]

\[ s_1 \]

\[ a_1 \]

\[ a_2 \]

\[ \pi_1 \]

\[ \pi_2 \]

\[ s_2 \]

\[ s_e \]
Feedback Policy

\[ C_1 \xrightarrow{\delta t} C_2 \]

\[ \pi_1 \]

\[ \pi_2 \]

Target Poses/
Joint Angles

Feedback
Actions

Simulation
States

\[ \m_1 \xrightarrow{\delta t} \m_2 \xrightarrow{\delta t} \m_3 \xrightarrow{\delta t} \m_4 \xrightarrow{\delta t} \m_5 \xrightarrow{\delta t} \m_6 \]

\[ s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow s_4 \rightarrow s_5 \rightarrow s_6 \rightarrow s_e \]
Learning of Control Fragments

What do we need to learn?

- Open-loop control trajectory $\hat{m}$
  - SAMCON [Liu et al. 2010, 2015]
- Feedback policy $\pi$
  - Guided SAMCON
SAmpling-based Motion CONtrol [Liu et al. 2010, 2015]
• Motion Clip → Open-loop control trajectory
SAMCON

Reference Trajectory

State

\( \delta t \)

\( \delta t \)

\( \delta t \)

\( \delta t \)

time
Sampling & Simulation
Sample Selection

Actions (PD-control Targets)

State

Reference Trajectory

time

\[ \delta t \]

\[ \delta t \]

\[ \delta t \]

\[ \delta t \]
SAMCON Iterations

Actions
(PD-control Targets)

State

Reference Trajectory

dt

time

α

Constructed Open-loop Control Trajectory
Sample Distribution

\[ a \sim \mathcal{N}(\mu, \Sigma) \]
SAMCON Family

Original SAMCON
[Liu et al. 2010]
SAMCON Family

Original SAMCON
[Liu et al. 2010]

Improved SAMCON
[Liu et al. 2015]
SAMCON Family

Original SAMCON
[Liu et al. 2010]

Improved SAMCON
[Liu et al. 2015]

Guided SAMCON

\[ \mu = 0 \]

\[ \mu = \mu_k \]
SAMCON Family

Original SAMCON
[Liu et al. 2010]

Improved SAMCON
[Liu et al. 2015]

Guided SAMCON

\[ \mu = 0 \]

\[ \mu = \mu_k \]

\[ \mu = M_k s + \hat{a}_k \]
SAMCON Family

Original SAMCON
[Liu et al. 2010]

Improved SAMCON
[Liu et al. 2015]

Guided SAMCON

\[
\mu = 0 \\
\mu = \mu_k \\
\mu = M_k s + \hat{a}_k
\]
SAMCON Family

Original SAMCON
[Liu et al. 2010]

Improved SAMCON
[Liu et al. 2015]

Guided SAMCON
Policy Update
Policy Update

Regression
Guided Learning Iterations

Regression

Guided SAMCON
Guided Learning Iterations

Regression

Guided SAMCON

Regression
Example: Cyclical Motion

\[ C_k: \{ \tilde{m}_k, \delta t, \pi_k \} \]

\[ \{ C_1, C_2, C_3, C_4, C_1, C_2, C_3, C_4, C_1, C_2, C_3, C_4, \ldots \} \]

\[ \text{SAMCON} \]
Example: Cyclical Motion

\[ \pi_1, \pi_2, \pi_3, \pi_4 \]

\[ \mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \mathcal{C}_4, \mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \mathcal{C}_4, \ldots \]

SAMCON
Spin Kick

Back Flip
(with pause)

Funky Walk

Back Flip
(with small hop)
Control Graph

A graph whose nodes are control fragments
Control Graph

A graph whose nodes are control fragments
Converted from a motion graph
Learning of Control Graphs

Random Walk ($W$):

$$\{c_1, c_2, c_3, c_1, c_2, c_3, c_4, c_5, c_6, \ldots \}$$
Learning of Control Graphs

Random Walk ($\mathcal{W}$):

$$\{c_1, c_2, c_3, c_1, c_2, c_3, c_4, c_5, c_6, \ldots \}$$
Performance

Offline learning
• 1 second reference motion -> \( \approx \) 1 hour learning
• 20-core computer

Runtime
• >10x realtime
• ODE, time step: 5ms
Conclusion

Learning Pipeline

Motion Clip

Motion Graph

- Stand
- Walk
- Run
- Kip-up
- Flip

Run

Stand

Walk

Kip-up

Flip
Limitations & Future Work

Active human-object interaction
• basketball, soccer

Multi-character interaction
• dancing, boxing, martial arts

Motion Planning