UT Austin Villa 2014: RoboCup 3D Simulation League Champion via Overlapping Layered Learning

Patrick MacAlpine, Mike Depinet, and Peter Stone

Department of Computer Science, The University of Texas at Austin

January 28, 2015
Layered Learning:
Hierarchical machine learning paradigm that enables learning of complex behaviors by incrementally learning a series of sub-behaviors. Higher layers directly depend on the learned lower layers.

Sequential Layered Learning (SLL): Freeze parameters of layer after learning before learning of the next layer (P. Stone, 2000)

Layered Learning:
Hierarchical machine learning paradigm that enables learning of complex behaviors by incrementally learning a series of sub-behaviors. Higher layers directly depend on the learned lower layers.

PROBLEMS:
Sequential Layered Learning: Can be too limiting in the joint behavior policy search space

Concurrent Layered Learning: The increased dimensionality can make learning harder or intractable
Layered Learning:
Hierarchical machine learning paradigm that enables learning of complex behaviors by incrementally learning a series of sub-behaviors. Higher layers directly depend on the learned lower layers.

SOLUTION:
Overlapping Layered Learning: Tradeoff between freezing or keeping open previous learned behaviors

Optimizes “seam” or overlap between behaviors: keeps some parts of previously learned layers open during subsequent learning
Overlapping Layered Learning:

Keeps some, but not necessarily all, parts of previously learned layers open during learning of subsequent layers.

Patrick MacAlpine (2015)
Overlapping Layered Learning:
Keeps some, but not necessarily all, parts of previously learned layers open during learning of subsequent layers.

Combining Independently Learned Behaviors: Two or more behaviors learned independently and then combined by relearning subset of behaviors’ parameters
Overlapping Layered Learning:
Keeps some, but not necessarily all, parts of previously learned layers open during learning of subsequent layers.

Combining Independently Learned Behaviors: Two or more behaviors learned independently and then combined by relearning subset of behaviors’ parameters
Partial Concurrent Layered Learning: Only part, but not all, of a previously learned layer’s behaviors are left open
Overlapping Layered Learning:
Keeps some, but not necessarily all, parts of previously learned layers open during learning of subsequent layers.

Combining Independently Learned Behaviors: Two or more behaviors learned independently and then combined by relearning subset of behaviors’ parameters
Partial Concurrent Layered Learning: Only part, but not all, of a previously learned layer’s behaviors are left open
Previous Learned Layer Refinement: After a layer is learned and frozen, and then a subsequent layer is learned, part of all of the previous layer is unfrozen
RoboCup 3D Simulation Domain

- Teams of 11 vs 11 autonomous robots play soccer
- Realistic physics using Open Dynamics Engine (ODE)
- Simulated robots modeled after Aldebaron Nao robot
- Robot receives noisy visual information about environment
- Robots can communicate with each other over limited bandwidth channel
Learned Layers

- **19 learned behaviors** for standing up, walking, and kicking (more than 3X behaviors of previous layered learning systems)
  - CILB, PCLL, PLLR
- **Over 500 parameters optimized** during the course of learning using CMA-ES algorithm
  - frozen, open
Learn four different walk parameter sets for four different subtasks

- Going to a target
- Sprinting forward (+/- 15° of current heading)
- Positioning around the ball when dribbling
- Approaching the ball to kick it

Learn fixed kick

Combine kick with walk through overlapping behavior layer
Sequential Layered Learning of Walk Behaviors

Red ‘T’ = *GoToTarget* parameters, yellow ’S’ = *Sprint* parameters

- Optimizing parameters for omnidirectional walk engine (step height, frequency, balance, etc.)
- Agent rewarded for distance traveled toward magenta target
- First *GoToTarget* layer optimized and frozen, then *Sprint* layer learned through *sequential layered learning*
Without Layered Learning

Attempt to transition between *Dribble* walk parameters (red 'D') and *Fast* walk parameters (yellow 'F')

- Unstable when not using layered learning to learn transition between walks
Kick_Long_Primitive Optimization

- Optimize joint positions that make up a series of fixed frame poses for executing kicking motion
- Kick ball from fixed standing position
- Reward for kick distance and accuracy

Patrick MacAlpine (2015)
Kick_Long_Behavior Optimization

- Approach ball and kick it
- Reward for kick distance and accuracy
- Relearning overlap kick parameters for positioning and stability with walk (combining independently learned behaviors)

Patrick MacAlpine (2015)
Layered Learning Paradigm Comparisons

Learning the Kick_Fast_Behavior

- **Concurrent Layered Learning** struggles learning kick and approach at same time
- **Sequential Layered Learning** difficulty learning kick in presence of walk approach
- **Overlapping Layered Learning (CILB)**, where walk approach and kick are learned independently in isolation and then combined, performs the best
Dribbling and Kicking the Ball

Red 'T' = *GoToTarget* parameters, yellow 'S' = *Sprint* parameters, cyan 'P' = *Positioning* parameters, orange 'A' = *Approach* parameters
Kickoffs are indirect (can’t score with a single kick)
Learn touch and fixed kick behaviors independently
Combining touch and kick by relearning positioning parameters (combining independently learned behaviors) and also learning new timing parameter (partial concurrent layered learning)
Kickoff Fail

Robots interfere with each other when trying to learn a kick with a touch
Repetition on Different Robot Types

Type 0: Standard Nao model
Type 1: Longer legs and arms
Type 2: Quicker moving feet
Type 3: Wider hips and longest legs and arms
Type 4: Added toes to foot
Repetition on Different Robot Types

Type 0: Standard Nao model
Type 1: Longer legs and arms
Type 2: Quicker moving feet
Type 3: Wider hips and longest legs and arms
Type 4: Added toes to foot

<table>
<thead>
<tr>
<th>Opponent</th>
<th>Type 0</th>
<th>Type 1</th>
<th>Type 2</th>
<th>Type 3</th>
<th>Type 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apollo3D</td>
<td>1.788</td>
<td>1.907</td>
<td>1.892</td>
<td>1.524</td>
<td>2.681</td>
</tr>
<tr>
<td>AustinVilla2013</td>
<td>0.950</td>
<td>0.858</td>
<td>1.152</td>
<td>0.613</td>
<td>1.104</td>
</tr>
<tr>
<td>FCPortugal</td>
<td>2.381</td>
<td>2.975</td>
<td>3.331</td>
<td>2.716</td>
<td>3.897</td>
</tr>
</tbody>
</table>

Learning paradigms display good effectiveness and generality
Repetition on Different Robot Types

Type 0: Standard Nao model
Type 1: Longer legs and arms
Type 2: Quicker moving feet
Type 3: Wider hips and longest legs and arms
Type 4: Added toes to foot

<table>
<thead>
<tr>
<th>Opponent</th>
<th>Type 0</th>
<th>Type 1</th>
<th>Type 2</th>
<th>Type 3</th>
<th>Type 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apollo3D</td>
<td>1.788</td>
<td>1.907</td>
<td>1.892</td>
<td>1.524</td>
<td>2.681</td>
</tr>
<tr>
<td>AustinVilla2013</td>
<td>0.950</td>
<td>0.858</td>
<td>1.152</td>
<td>0.613</td>
<td>1.104</td>
</tr>
<tr>
<td>FCPortugal</td>
<td>2.381</td>
<td>2.975</td>
<td>3.331</td>
<td>2.716</td>
<td>3.897</td>
</tr>
</tbody>
</table>

Learning paradigms display good effectiveness and **generality**

**Computation per type**

≈ 700k parameter sets evaluated
≈ 1.5 years compute time (≈ 5 days on distributed computing cluster)
RoboCup 2014

Won competition with **undefeated** record: outscored opps 52–0

<table>
<thead>
<tr>
<th>Opponent</th>
<th>Avg. Goal Diff.</th>
<th>Record (W-L-T) %</th>
</tr>
</thead>
<tbody>
<tr>
<td>BahiaRT</td>
<td>2.075 (0.030)</td>
<td>990-0-10</td>
</tr>
<tr>
<td>FCPortugal</td>
<td>2.642 (0.034)</td>
<td>986-0-14</td>
</tr>
<tr>
<td>magmaOffenburg</td>
<td>2.855 (0.035)</td>
<td>990-0-10</td>
</tr>
<tr>
<td>RoboCanes</td>
<td>3.081 (0.046)</td>
<td>974-0-26</td>
</tr>
<tr>
<td>FUT-K</td>
<td>3.236 (0.039)</td>
<td>998-0-2</td>
</tr>
<tr>
<td>SEU_Jolly</td>
<td>4.031 (0.062)</td>
<td>995-0-5</td>
</tr>
<tr>
<td>KarachiKoalas</td>
<td>5.681 (0.046)</td>
<td>1000-0-0</td>
</tr>
<tr>
<td>ODENS</td>
<td>7.933 (0.041)</td>
<td>1000-0-0</td>
</tr>
<tr>
<td>HfutEngine</td>
<td>8.510 (0.050)</td>
<td>1000-0-0</td>
</tr>
<tr>
<td>Mithras3D</td>
<td>8.897 (0.041)</td>
<td>1000-0-0</td>
</tr>
<tr>
<td>L3M-SIM</td>
<td>9.304 (0.043)</td>
<td>1000-0-0</td>
</tr>
</tbody>
</table>
RoboCup 2014

**Won competition with **undefeated** record: outscored opps 52–0**

<table>
<thead>
<tr>
<th>Opponent</th>
<th>Avg. Goal Diff.</th>
<th>Record (W-L-T) %</th>
</tr>
</thead>
<tbody>
<tr>
<td>BahiaRT</td>
<td>2.075 (0.030)</td>
<td>990-0-10</td>
</tr>
<tr>
<td>FCPortugal</td>
<td>2.642 (0.034)</td>
<td>986-0-14</td>
</tr>
<tr>
<td>magmaOffenburg</td>
<td>2.855 (0.035)</td>
<td>990-0-10</td>
</tr>
<tr>
<td>RoboCanes</td>
<td>3.081 (0.046)</td>
<td>974-0-26</td>
</tr>
<tr>
<td>FUT-K</td>
<td>3.236 (0.039)</td>
<td>998-0-2</td>
</tr>
<tr>
<td>SEU_Jolly</td>
<td>4.031 (0.062)</td>
<td>995-0-5</td>
</tr>
<tr>
<td>KarachiKoalas</td>
<td>5.681 (0.046)</td>
<td>1000-0-0</td>
</tr>
<tr>
<td>ODENS</td>
<td>7.933 (0.041)</td>
<td>1000-0-0</td>
</tr>
<tr>
<td>HfutEngine</td>
<td>8.510 (0.050)</td>
<td>1000-0-0</td>
</tr>
<tr>
<td>Mithras3D</td>
<td>8.897 (0.041)</td>
<td>1000-0-0</td>
</tr>
<tr>
<td>L3M-SIM</td>
<td>9.304 (0.043)</td>
<td>1000-0-0</td>
</tr>
</tbody>
</table>

- Across **11,000 games** played won all but 67 games which ended in ties *(no losses)*
Summary

- Introduced three paradigms for Overlapping Layered Learning
  - Combining Independently Learned Behaviors, Partial Concurrent Layered Learning, Previous Learned Layer Refinement

- Showed effectiveness of Overlapping Layered Learning for learning complex behaviors in the RoboCup 3D simulation domain
  - Learned 19 behaviors while optimizing over 500 parameters
- Demonstrated generality of Overlapping Layered Learning to multiple robot models
  - Able to successfully learn behaviors for 5 different robot types
Summary

- **Introduced three paradigms** for Overlapping Layered Learning
  - Combining Independently Learned Behaviors, Partial Concurrent Layered Learning, Previous Learned Layer Refinement

- **Showed effectiveness** of Overlapping Layered Learning for learning complex behaviors in the RoboCup 3D simulation domain
  - Learned 19 behaviors while optimizing over 500 parameters
Summary

- **Introduced three paradigms** for Overlapping Layered Learning
  - Combining Independently Learned Behaviors, Partial Concurrent Layered Learning, Previous Learned Layer Refinement

- **Showed effectiveness** of Overlapping Layered Learning for learning complex behaviors in the RoboCup 3D simulation domain
  - Learned 19 behaviors while optimizing over 500 parameters

- **Demonstrated generality** of Overlapping Layered Learning to multiple robot models
  - Able to successfully learn behaviors for 5 different robot types
More Information

UT Austin Villa 3D Simulation Team homepage:
www.cs.utexas.edu/~AustinVilla/sim/3dsimulation/
Email: patmac@cs.utexas.edu

Highlights from 2014 Final vs RoboCanes (University of Miami)
Need to Optimize Hand-tuned Behaviors

- Walk designed and hand-tuned to work on the actual Nao robot
- Provides a slow and stable walk
CMA-ES (Covariance Matrix Adaptation Evolutionary Strategy)

- Evolutionary numerical optimization method
- Candidates sampled from multidimensional Gaussian and evaluated for their fitness
- Weighted average of members with highest fitness used to update mean of distribution
- Covariance update using evolution paths controls search step sizes
Walk_PositionToDribble Optimization

Red 'T' = GoToTarget, yellow 'S' = Sprint, cyan 'P' = Positioning parameters

- Dribble ball toward goal for 15 seconds from multiple starting points around ball
- Reward for distance ball dribbled toward goal
Walk_ApproachToKick Optimization

Red ‘T’ = GoToTarget, yellow ‘S’ = Sprint, orange ‘A’ = Approach parameters

- Approach position relative to ball to execute kick
- Penalized for time taken to reach point to execute kick from

Patrick MacAlpine (2015)
KickOff_Touch_Primitive

- Touch ball only once and move ball as little as possible
● Long accurate kick that travels far in the air
### Impact of Overlapping Layered Learning

1000 games vs. top 3 teams from 2013

<table>
<thead>
<tr>
<th>Team</th>
<th>Average Goal Difference</th>
<th>No Kickoff</th>
<th>Dribble Only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apollo3D</td>
<td>2.726 (0.036)</td>
<td>2.059 (0.038)</td>
<td>1.790 (0.033)</td>
</tr>
<tr>
<td>AustinVilla2013</td>
<td>1.525 (0.032)</td>
<td>1.232 (0.032)</td>
<td>0.831 (0.023)</td>
</tr>
<tr>
<td>FCPortugal</td>
<td>3.951 (0.049)</td>
<td>3.154 (0.046)</td>
<td>1.593 (0.028)</td>
</tr>
</tbody>
</table>

- **No Kickoff**: On kickoff, kick ball deep into opponent's end
- **Dribble Only**: No kicking
### Impact of Overlapping Layered Learning

1000 games vs. top 3 teams from 2013

<table>
<thead>
<tr>
<th>Opponent</th>
<th>Full Team</th>
<th>No Kickoff</th>
<th>Dribble Only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apollo3D</td>
<td>2.726 (0.036)</td>
<td>2.059 (0.038)</td>
<td>1.790 (0.033)</td>
</tr>
<tr>
<td>AustinVilla2013</td>
<td>1.525 (0.032)</td>
<td>1.232 (0.032)</td>
<td>0.831 (0.023)</td>
</tr>
<tr>
<td>FCPortugal</td>
<td>3.951 (0.049)</td>
<td>3.154 (0.046)</td>
<td>1.593 (0.028)</td>
</tr>
</tbody>
</table>

**No Kickoff:** On kickoff, kick ball deep into opponent’s end  
**Dribble Only:** No kicking
Related Work