On the Role of Tracking in Stationary Environments

Rich Sutton
Anna Koop
David Silver

University of Alberta

with thanks to Mark Ring and Alborz Geramifard

Modern ML is focused on convergence to a static solution

- We usually assume learning is complete and over by the time the system is in normal operation
- In this sense we are more concerned with *learned* systems than with *learning* systems
- Even in reinforcement learning
  - where learning could be continual
  - still we focus on convergence to a static optimum

Converging vs Tracking

- Converging:
  - approaching a static best solution
- Tracking:
  - chasing an ever-changing best solution

Outline

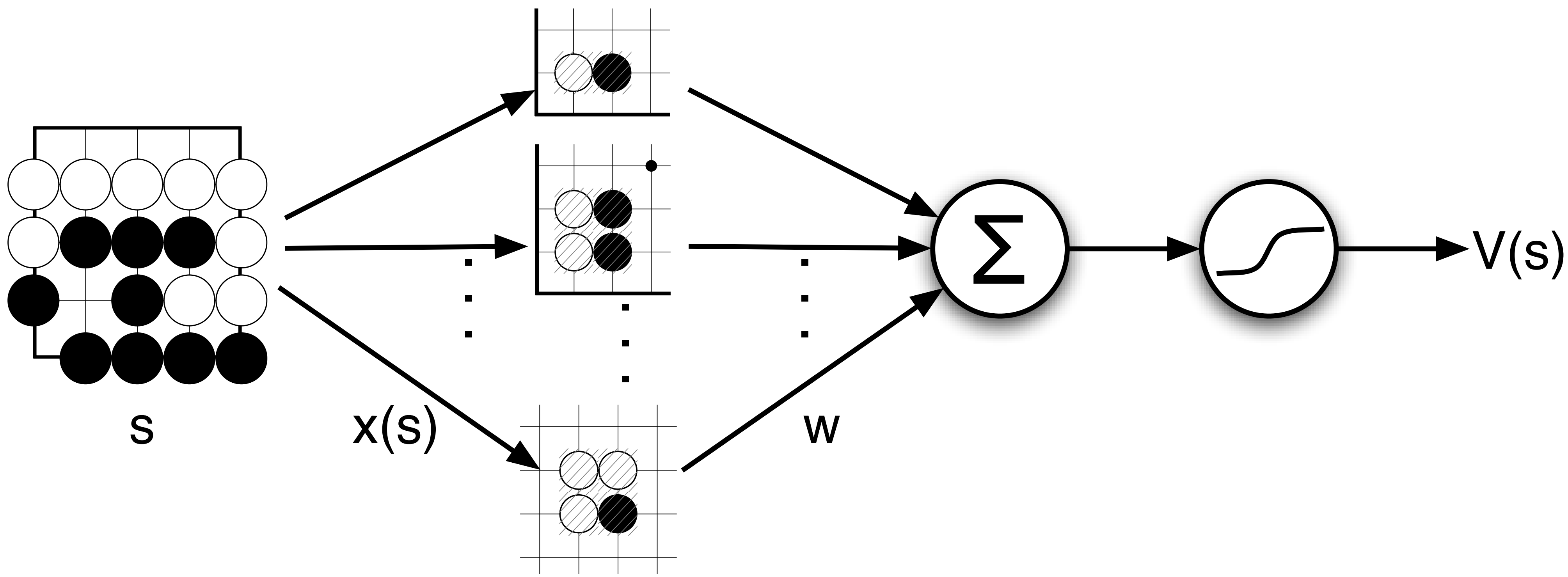
- Tracking wins in Computer Go
- Tracking wins in the Black & White world
- Tracking may revolutionize transfer learning

Computer Go has come of age

- It is now suitable for use as a challenging yet workable ML testbed

RLGO

(Silver, Sutton & Müller, 2007)

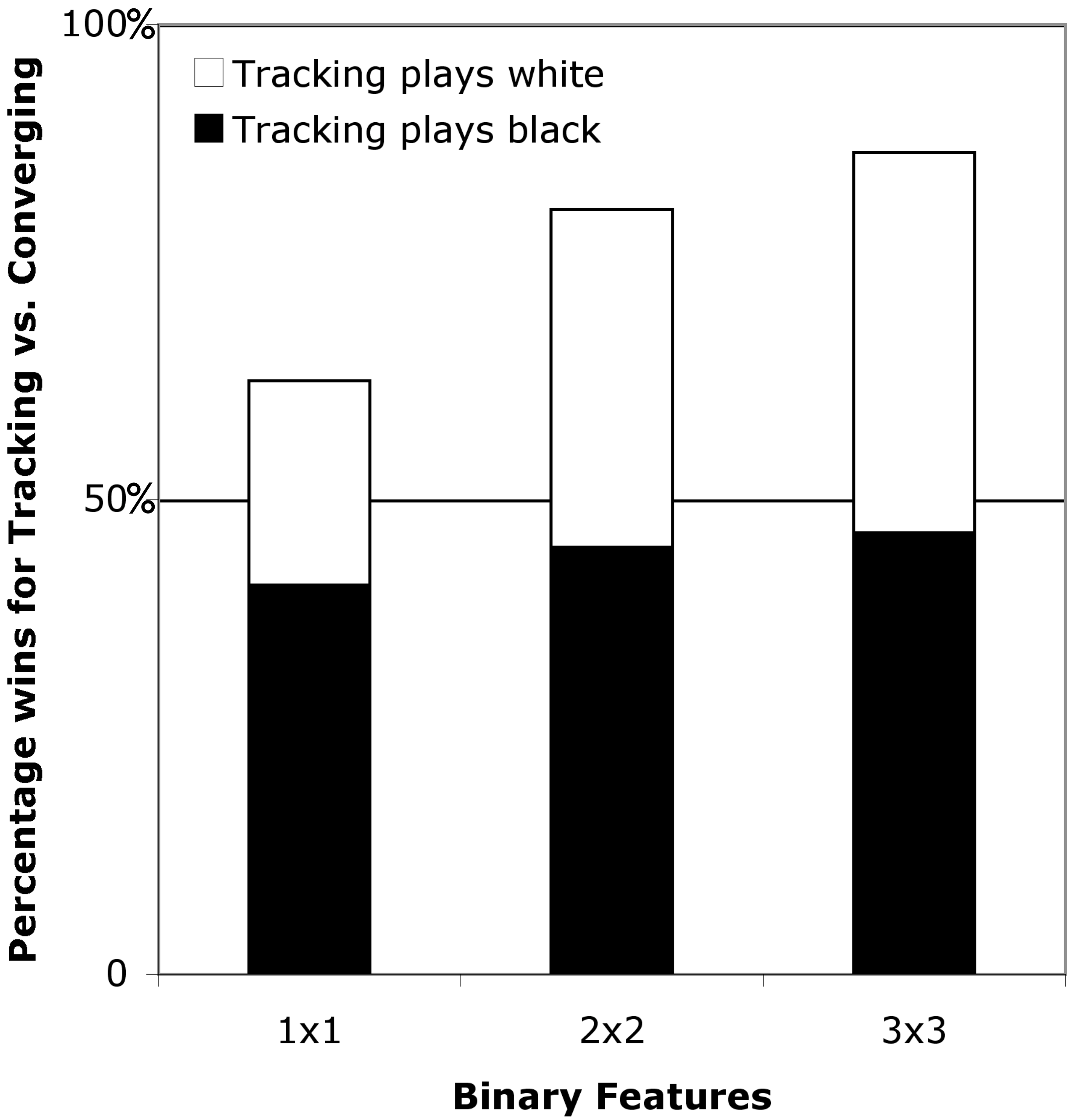


- Best static evaluation function in 9x9 Go without prior knowledge
- A component of the world's best Computer Go player, MoGo (Gelly & Silver, 2007)

Tracking vs converging in Go

- Converging player:
  - Self-play TD learning for 250,000 games
  - Final value function used for play (greedy)
- Tracking player:
  - Each game starts with random value function
  - For each position encountered, apply self-play TD learning for 10,000 possible continuations
  - Current value function used for play
  - fast and practical

Computer Go results



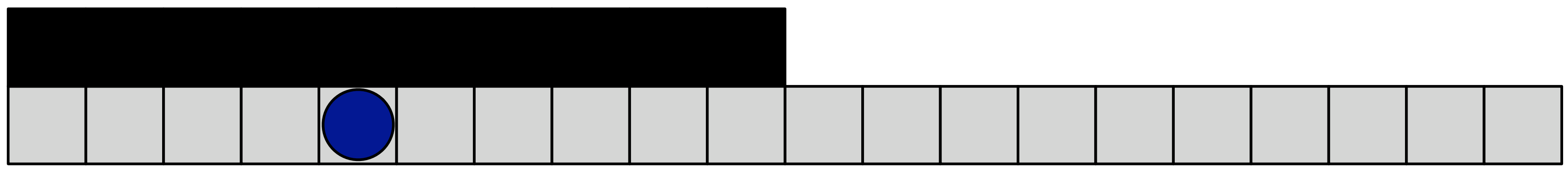
- Tracking player plays 100 games as white and 100 games as black
- Tracking player wins significant majority of games
- Advantage is greater with larger-template features

Tracking wins on a stationary problem

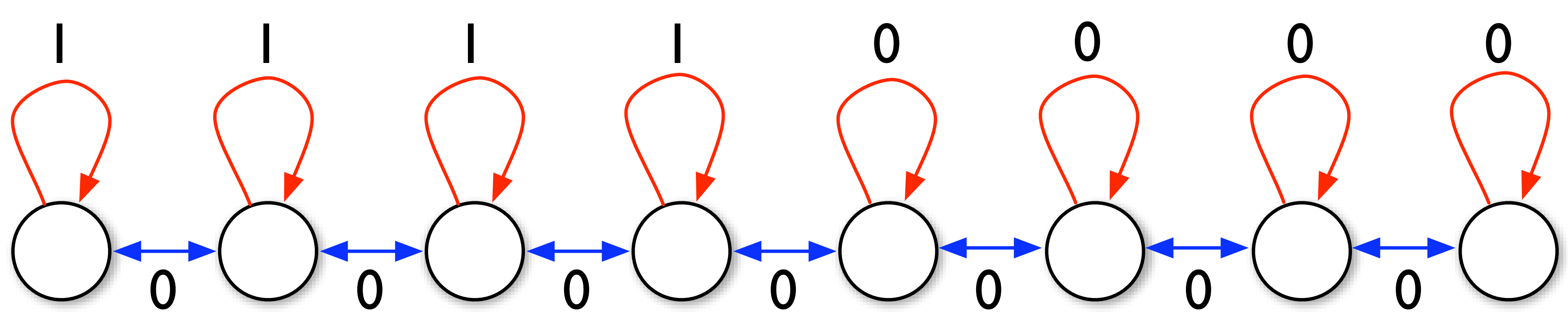
9x9 Go results
(not in paper)

- Tracking player beats all handcrafted Go programs
- Against 9x9 GnuGo:
  - converging player wins 5%
  - tracking player wins 57%
- Tracking player beats all converging Go programs
  - higher rated than NeuroGo (Enzenberger 2003)
- Tracking algorithms now dominate this domain

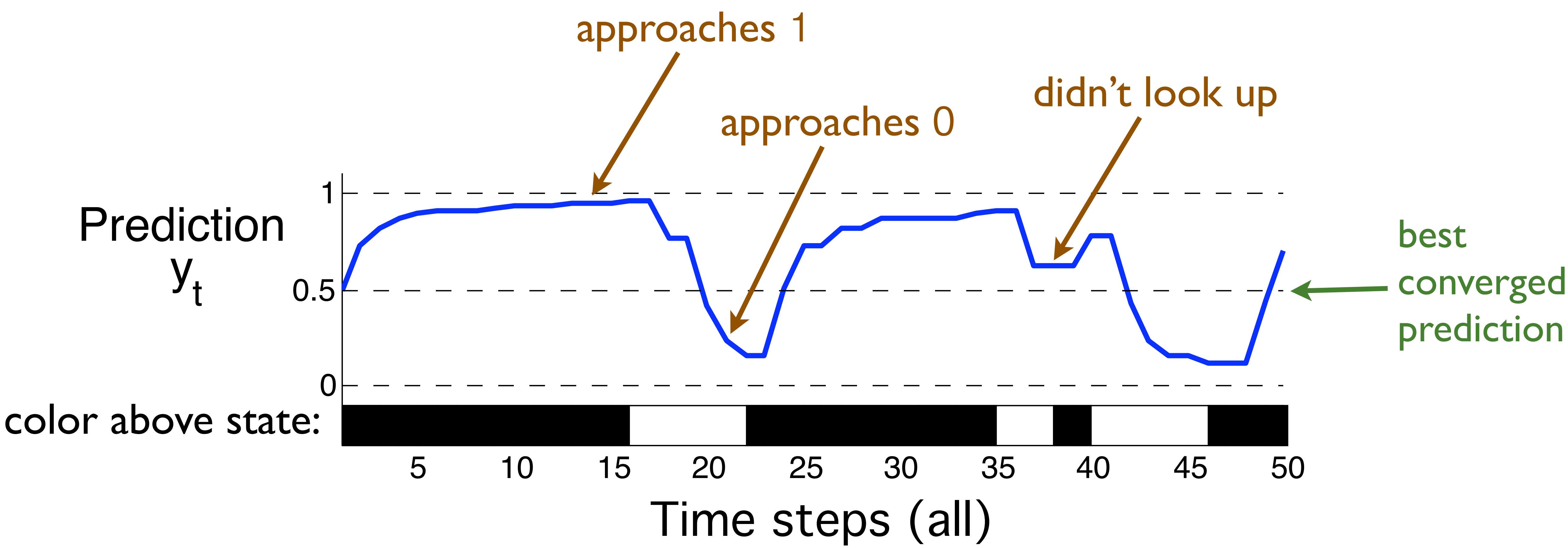
Black & White world



- States are indistinguishable
- Agent wanders back and forth
- Occasionally looks up $(p = 0.5)$
- Predict probability of seeing 1

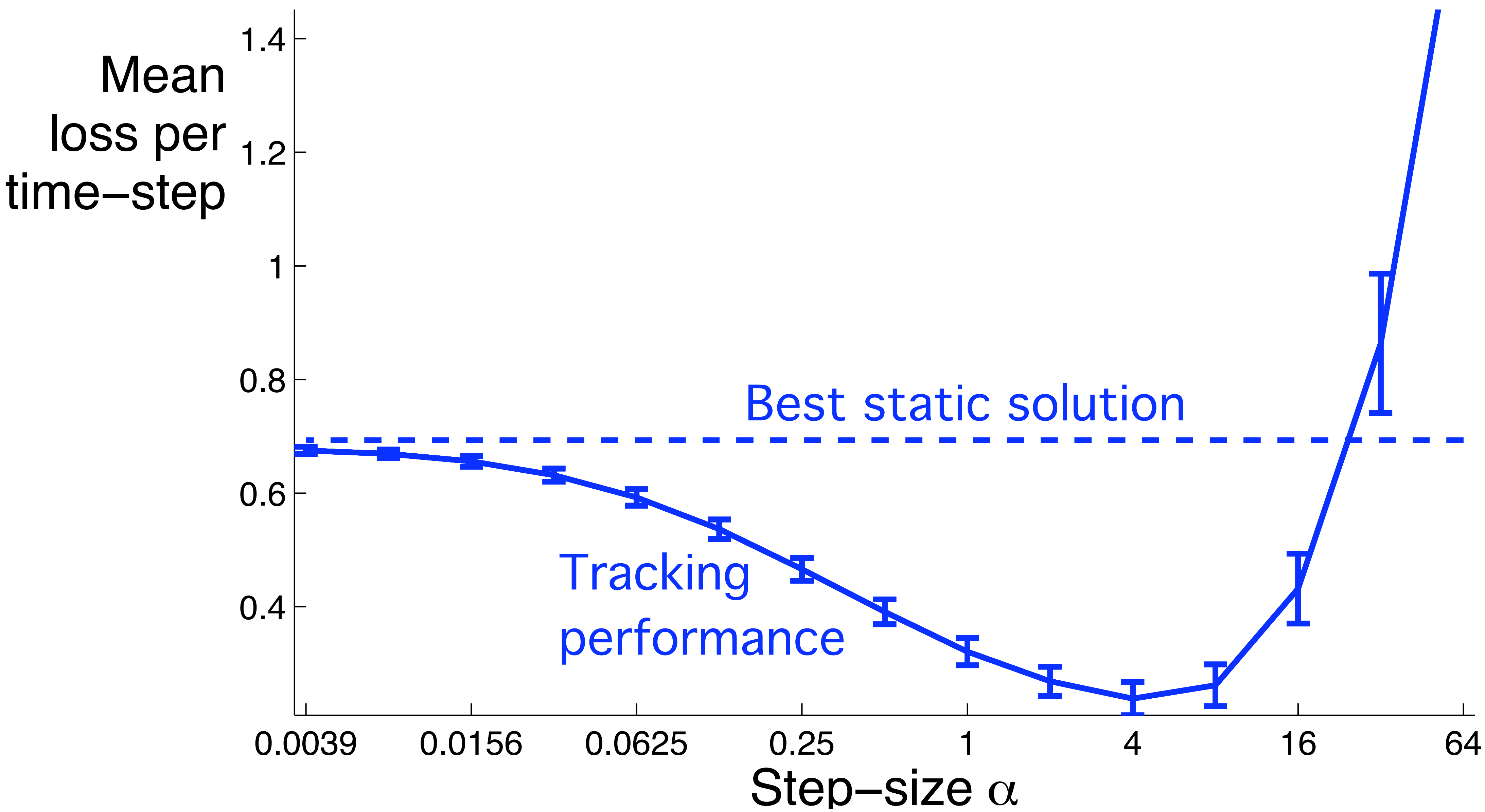


B&W: Sample trajectory



Tracking should be better than always predicting 0.5

B&W world results



Tracking is up to 3 times better than converging

B&W learning details

- Learn only when "looking up"
- Learn a single weight w_t
- Logistic semi-linear prediction

$$y_t = \frac{1}{1 + e^{-w_t x_t}} \qquad x_t = 1$$

- Log loss wrt observed target z_t

$$L_t = -z_t \log(y_t) - (1 - z_t)\log(1 - y_t)$$

- Gradient descent learning algorithm

$$w_{t+1} = w_t + \alpha(z_t - y_t)x_t$$

Conclusion: Tracking systems perform better

- In Computer Go, in B&W world, in Mountain Car (Alborz Geramifard)
- Tracking wins wherever there is
  - limited function approximation
  - temporal coherence
  - more state in the world than in your function approximator
- Tracking is a method, not a problem

Second conclusion: Tracking could revolutionize transfer learning

- Tracking involves continual, repeated learning
- Thus there is an opportunity for transfer learning methods
  - such as feature selection, learning-to-learn, meta-learning, and discovery of structure/representations/options...
- To have dramatic performance benefits
- Thus removing the need for multiple tasks

Example of transfer in tracking: Incremental delta-bar-delta (IDBD)

- A meta-learning method for automatically setting step-size parameters based on experience
- An incremental form of hold-one-out cross validation
- Originally proposed for supervised learning (Sutton, 1981; Jacobs, 1988; Sutton, 1992)
- Extended to TD learning (Utgoff, Schraudolf)
- Here extended to the semi-linear case

Incremental delta-bar-delta

$$\Delta w = \alpha * \text{error}$$

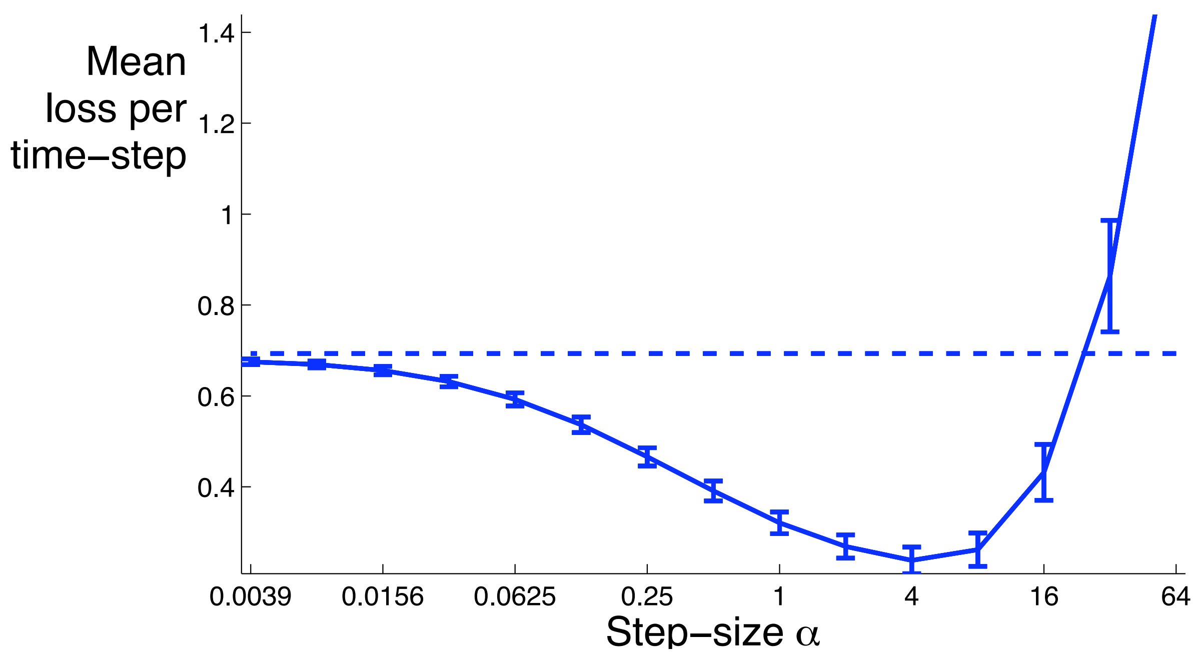
$$\Delta\alpha \propto \Delta w * \overline{\Delta w}$$

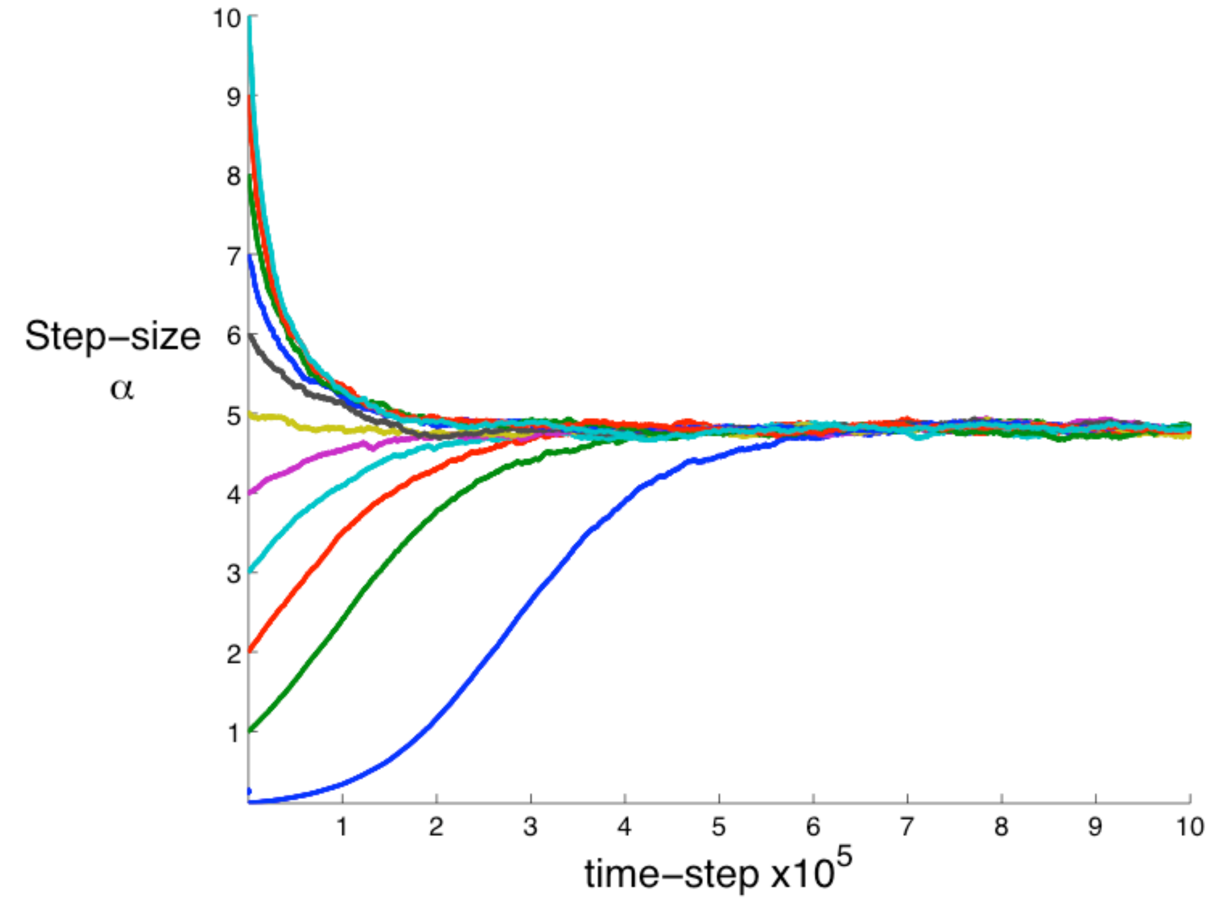
average Δw in the recent past

Algorithm 1 Semi-linear IDBD

Initialize h_i to 0, w_i and β_i as desired, $i = 1..n$
for each time step t **do**
  $y \leftarrow \frac{1}{1+e^{-w^T x}}$
  $\delta \leftarrow z - y$
  for each $i = 1..n$ **do**
    $\beta_i \leftarrow \beta_i + \mu \delta x_i h_i$
    $\alpha_i \leftarrow e^{\beta_i}$
    $w_i \leftarrow w_i + \alpha_i \delta x_i$
    $h_i \leftarrow h_i[1 - \alpha_i (x_i)^2 y(1 - y)] + \alpha_i \delta x_i$
  end for
end for

IDBD meta-learning on the B&W world





Without IDBD, the best fixed step-size is α ≈ 5

IDBD learns α ≈ 5 for a wide range of meta-step-size parameters

We can use a stationary tracking task to show the benefits of meta-learning

Final conclusion: Tracking rocks!

- Tracking systems can perform better
- Tracking shows off the benefits of meta-learning without multiple tasks
- Tracking recognizes the temporal structure of life/learning
- Tracking may be the way of the future for ML