

A Reinforcement Learning Approach for Solving Chess Endgames

Zacharias Georgiou (S3009874)

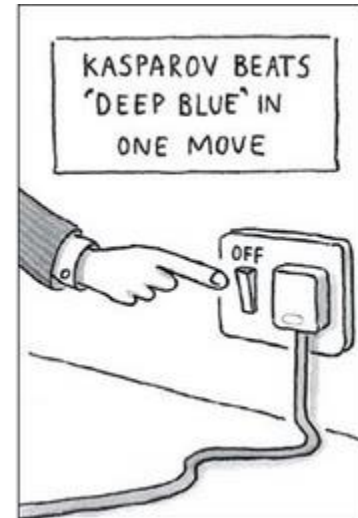
Evangelos Karountzos (S2705532)

Yaroslav Shkarupa (S3017125)

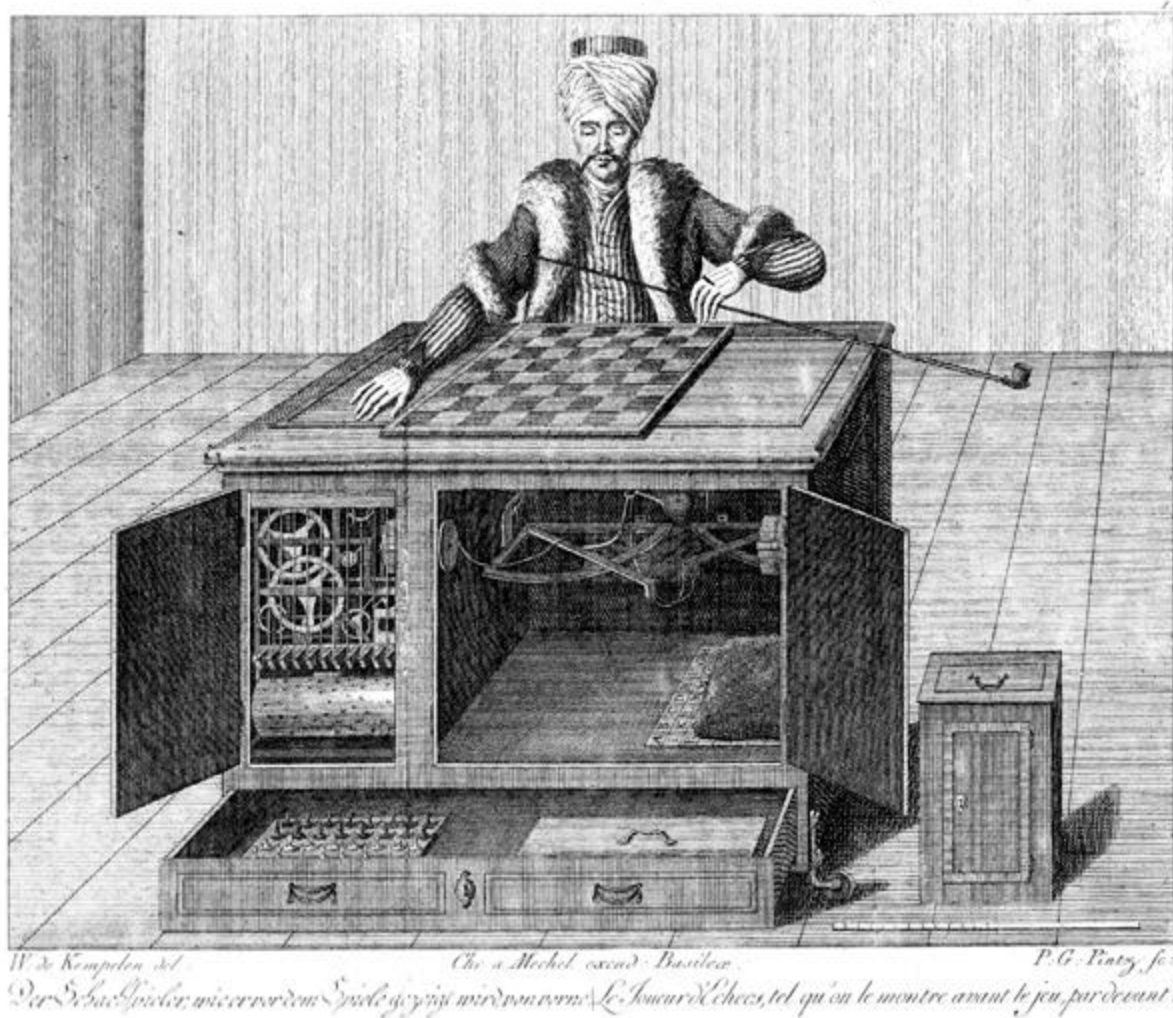
Matthia Sabatelli (S2847485)

Introduction

- Chess and AI
- Chess endgames
- Our project
 - Reinforcement Learning
 - Q-Learning
- Results
- Final demo



Chess and AI



The Turk first “AI” chess player

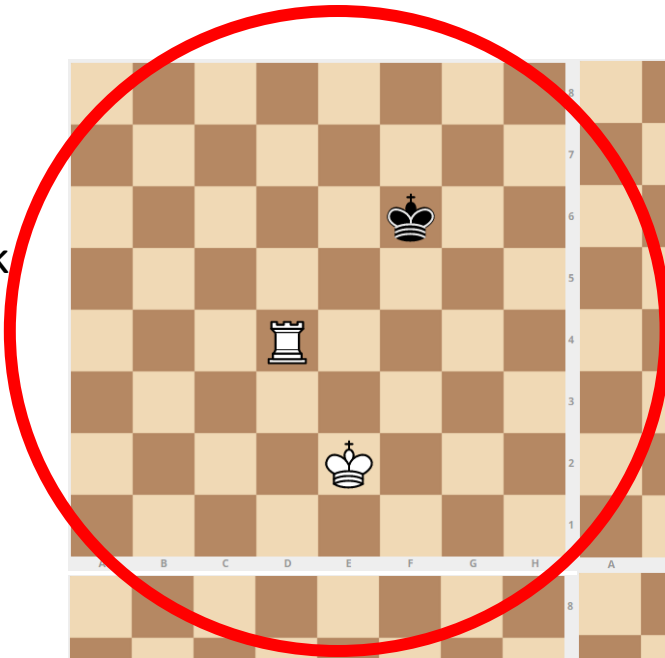
Chess and AI

Deep Blue computer
Winning Kasparov

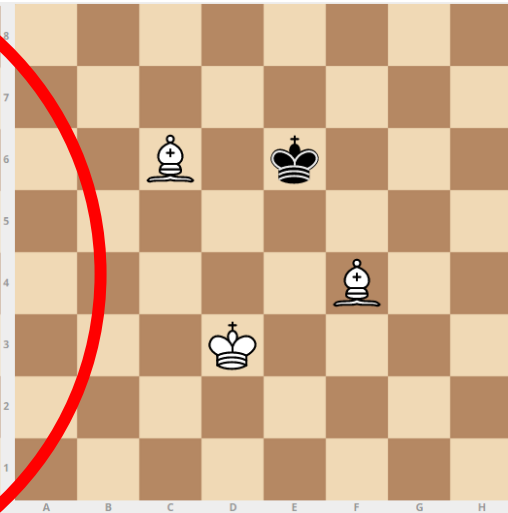


Chess endgames

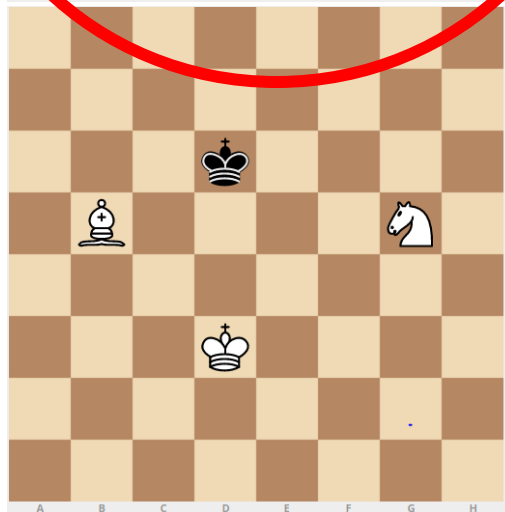
King-Rook
Vs
King



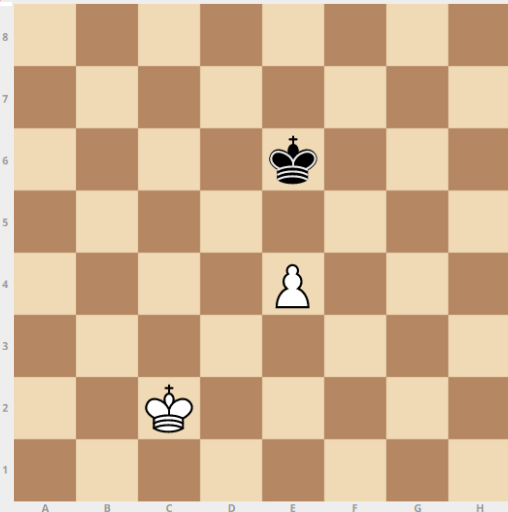
King-Bishop-Bishop
Vs
King



King-Bishop-Knight
Vs
King



King-Pawn
Vs
King



Our Project

- Machine Learning Method: Reinforcement Learning
- Learning Algorithm: Q-Learning
 - Starting from a random position
 - Policy: Random exploration
 - Creation of a basic memory (with all the possible states $2 \cdot 64^3$)
 - Training of the agent

$$Q_{t+1}(s_t, a_t) = \underbrace{Q_t(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha_t(s_t, a_t)}_{\text{learning rate}} \cdot \left(\underbrace{R_{t+1}}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \underbrace{\max_a Q_t(s_{t+1}, a)}_{\text{estimate of optimal future value}} - \underbrace{Q_t(s_t, a_t)}_{\text{old value}} \right)$$

**BLACK
TURN**

OTHER STATE
 $Q < Q(\text{CHEKMATE})$

CHEKMATE POSITION
Reward = 1



$$Q(S1, CM) \leq Q(OS) * \gamma$$

$$Q(S1, CM) \leq Q(CM) * \gamma$$

**WHITE
TURN**



STATE 1

STATE 2

$$Q(S3, S1) \leq \max(Q(S1, CM), Q(S1, OS)) * \gamma$$

**BLACK
TURN**

STATE 3



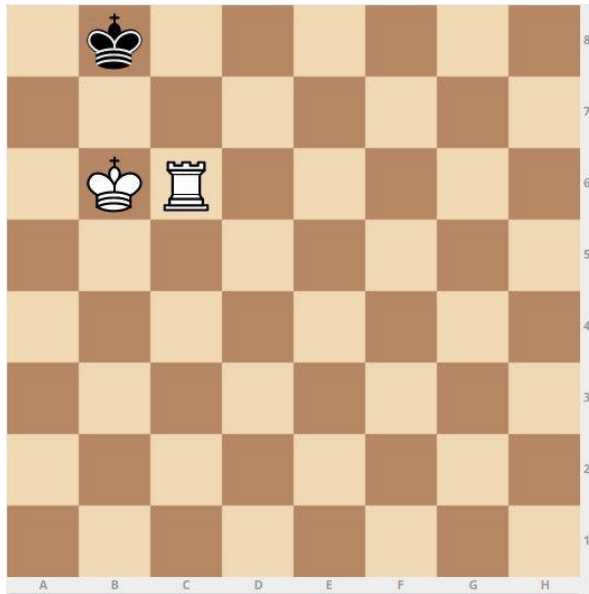
$$Q(S4, S3) \leq \min(Q(S3, S1), Q(S3, S2)) * \gamma$$

**WHITE
TURN**

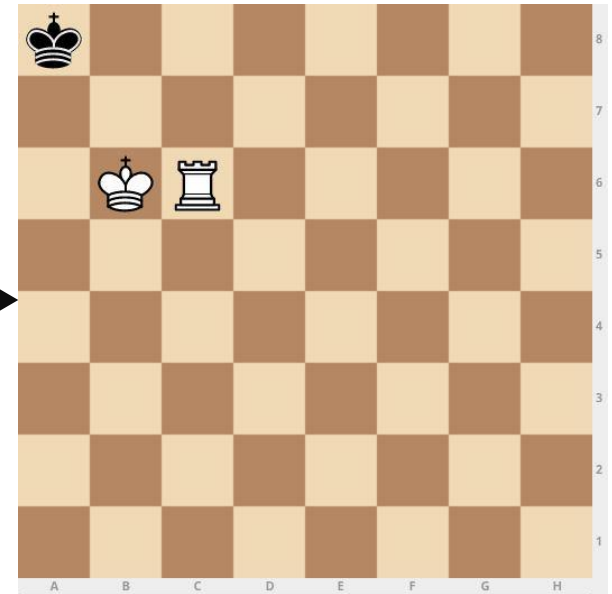
STATE 4

Playing process (after training)

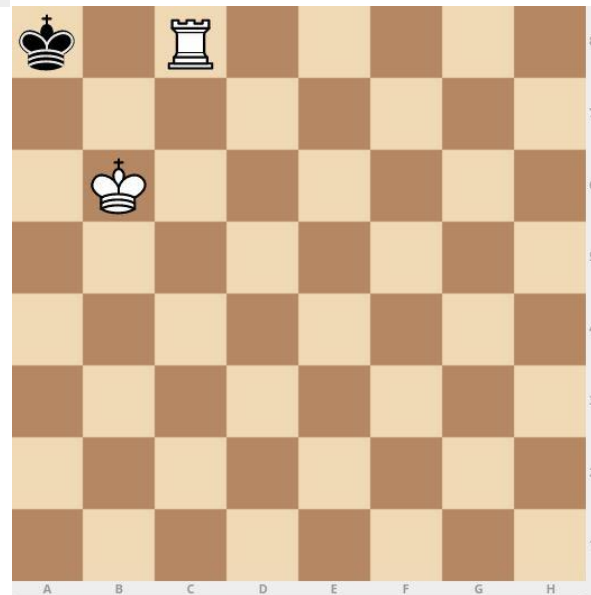
1. Black plays



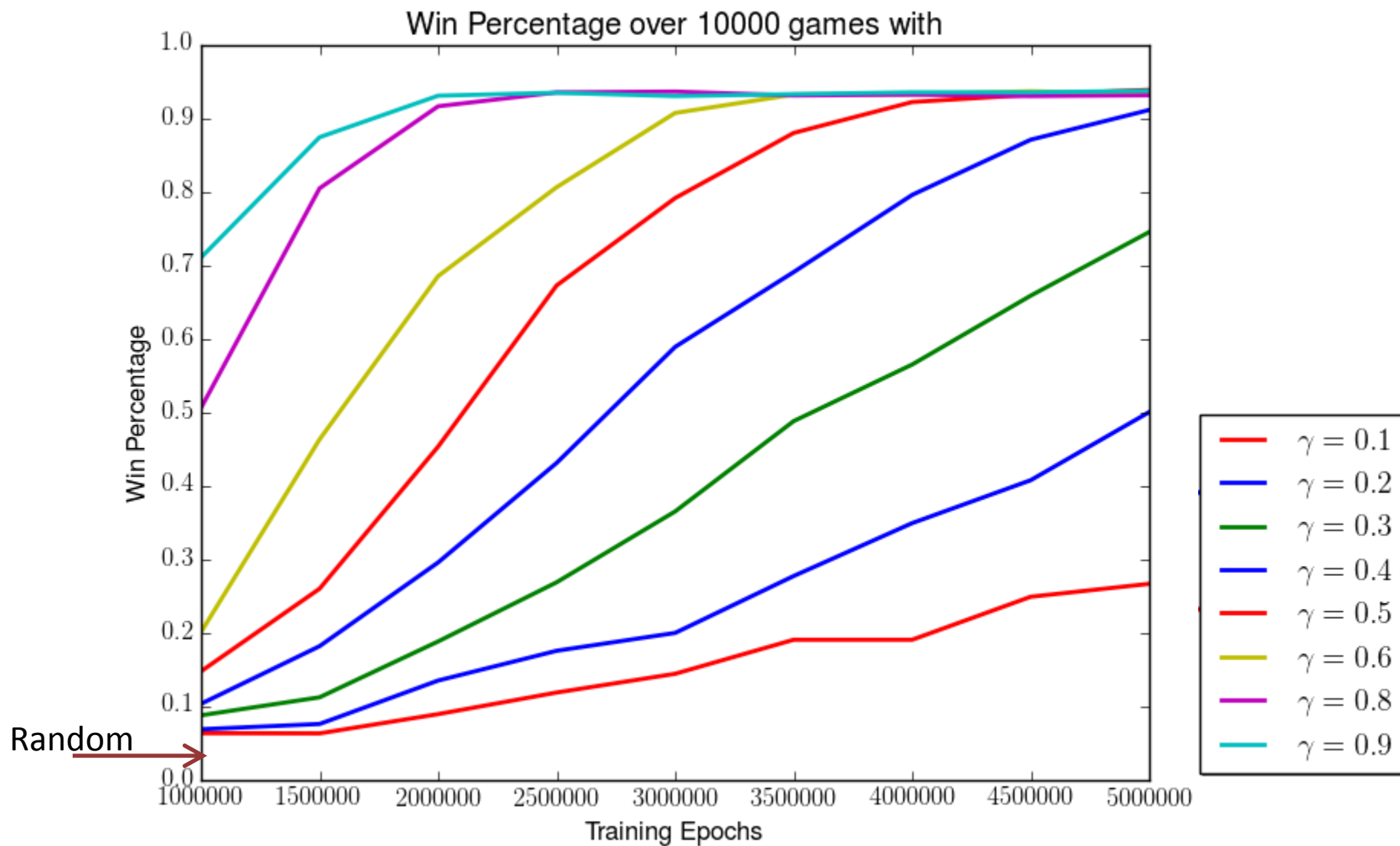
2. White plays



3. End



Results



Expansion, restrictions and improvements

- How can we expand?
 - Try different algorithms and compare them
 - Try different endgames
- Restrictions
 - Too big data set ($2 \cdot 64^3$)
 - Approximately 30.5 million states with 4 pieces
- Improvements
 - TD learning where you only save states and not actions
 - Multithreading
 - Data compression

LITTLE KNOWN CHESS FACTS #2

THE WHITE AND BLACK KINGS ARE,
IN SECRET, *DEEPLY* IN LOVE.

Thank you

