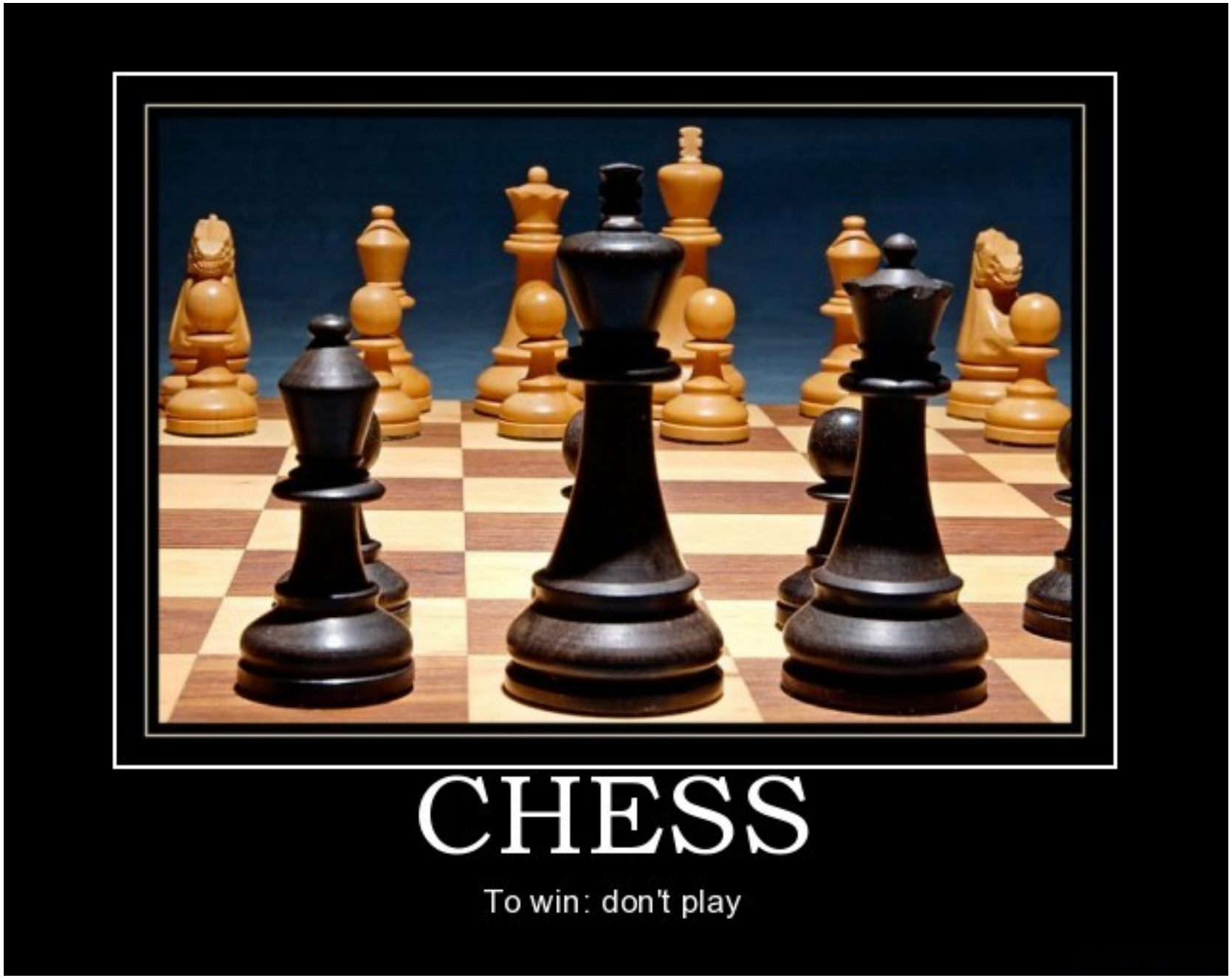
Chess AI improvement through an evolutionary approach

Cédric Guillot

CPSC 565

April 9, 2013


CHESS
To win: don't play

1. Introduction

2. Implementation / Strategy
   - Tools
   - Architecture
   - Board evaluation / parameters evaluation

3. Results
   - Set-up
   - Evolved AI

4. Future work

Chess AI history

- 1951 - Alan Turing develops on paper the first program capable of playing a full game of chess
- 1956 - John McCarthy invents the alpha-beta search algorithm
- 1957 - First practical chess program, Alex Bernstein and a team of Russian programmers
- 1981 - Cray Blitz becomes the first computer to gain a master rating (2200 ELO)
- 1997 - Deep Blue wins a six-game match against Garry Kasparov
- Today - Computers have reached 3250 ELO ratings

1 Introduction

2 Implementation / Strategy
- Tools
- Architecture
- Board evaluation / parameters evaluation

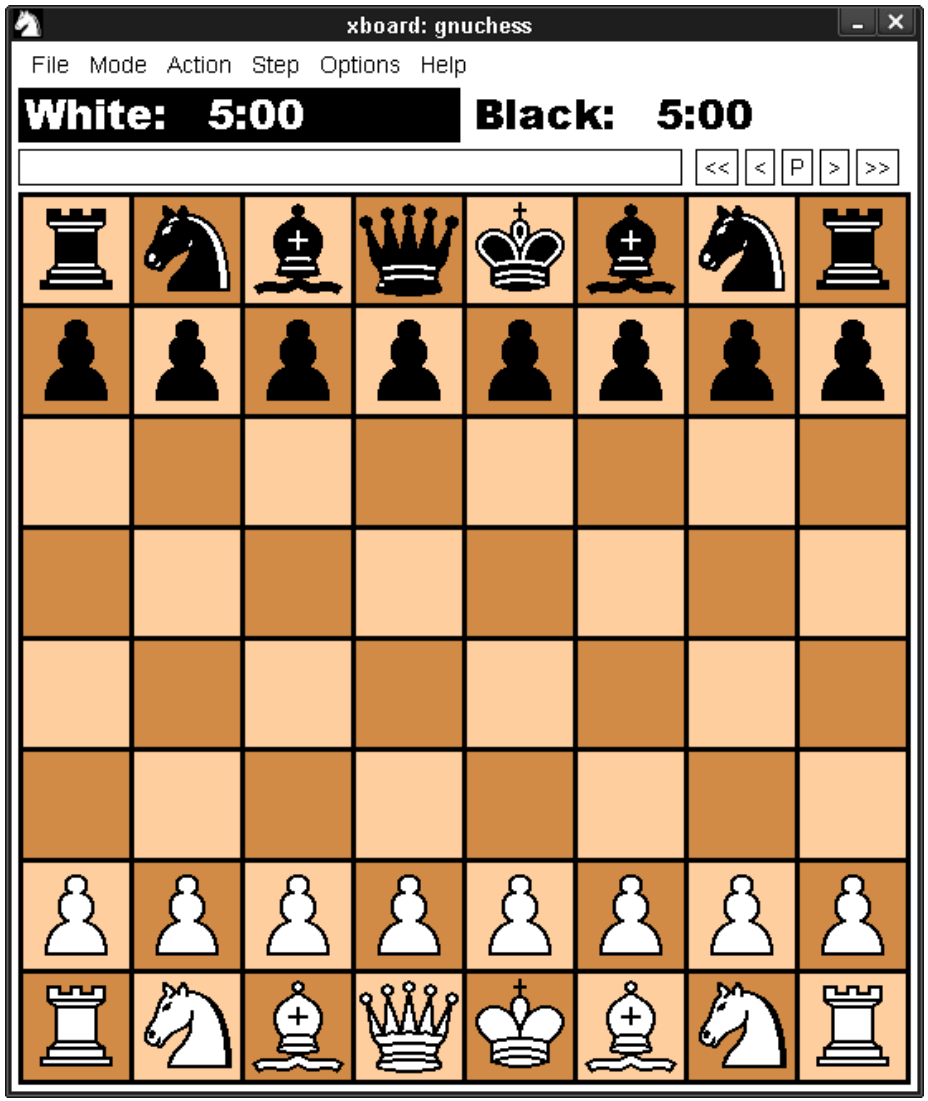
3 Results
- Set-up
- Evolved AI

4 Future work

Tom Kerrigan's Simple Chess Program (TSCP)

- Chess engine used for playing all the games
- Written in 1997
- Negamax algorithm for the AI

GUI: GNU Xboard



Architecture

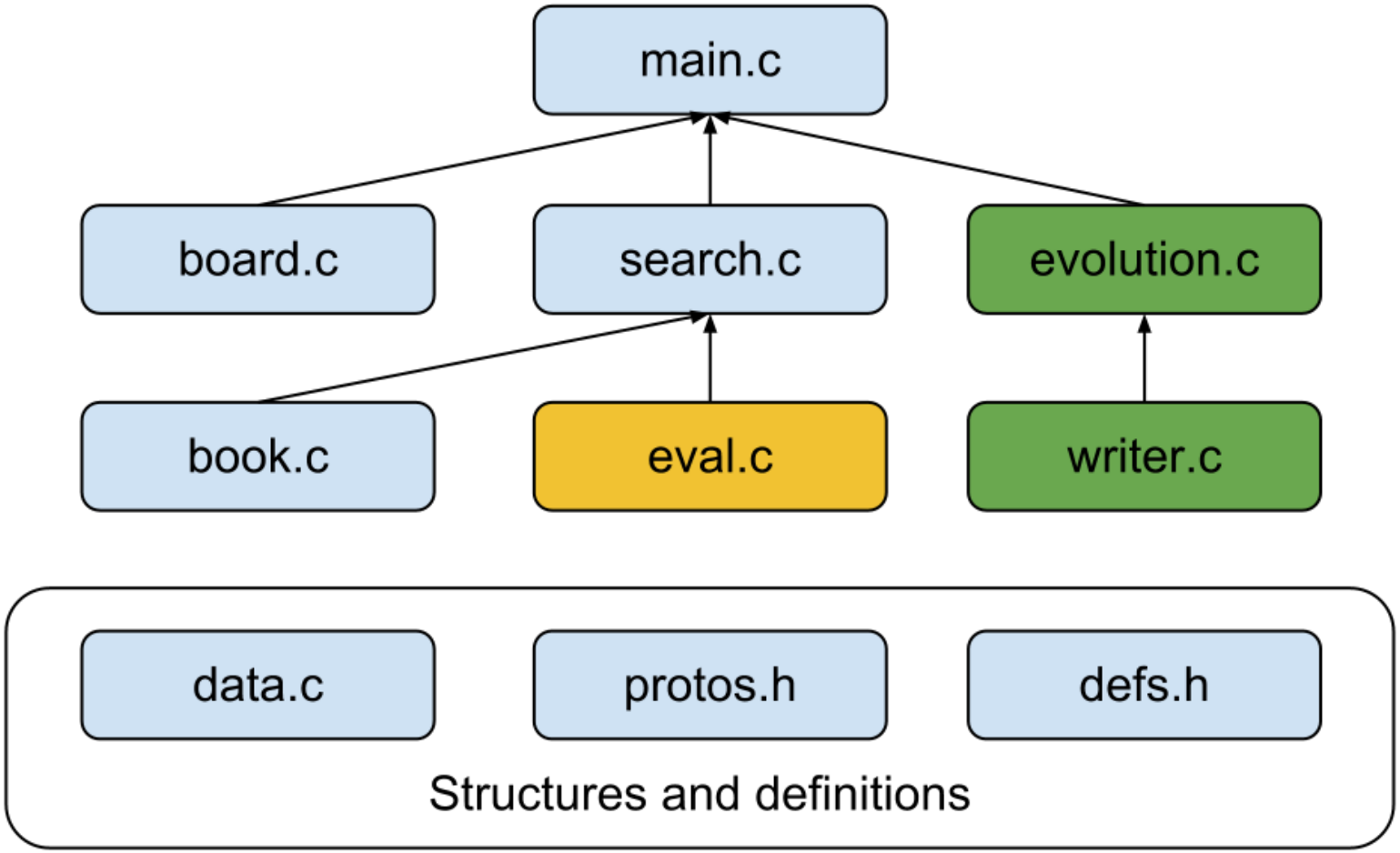


Figure: TSCP and evolution algorithm plugin architecture

Board evaluation / parameters evaluation

- Board evaluated at each move during the game, using the pieces values
- Individuals of the same generation compete against each other
  - $\frac{n(n-1)}{2}$ games per generation
  - one game as white, one as black
- Point system: 0 for loss, 3 for win, 1 for stalemate or draw

1. Introduction

2. Implementation / Strategy
   - Tools
   - Architecture
   - Board evaluation / parameters evaluation

3. Results
   - Set-up
   - Evolved AI

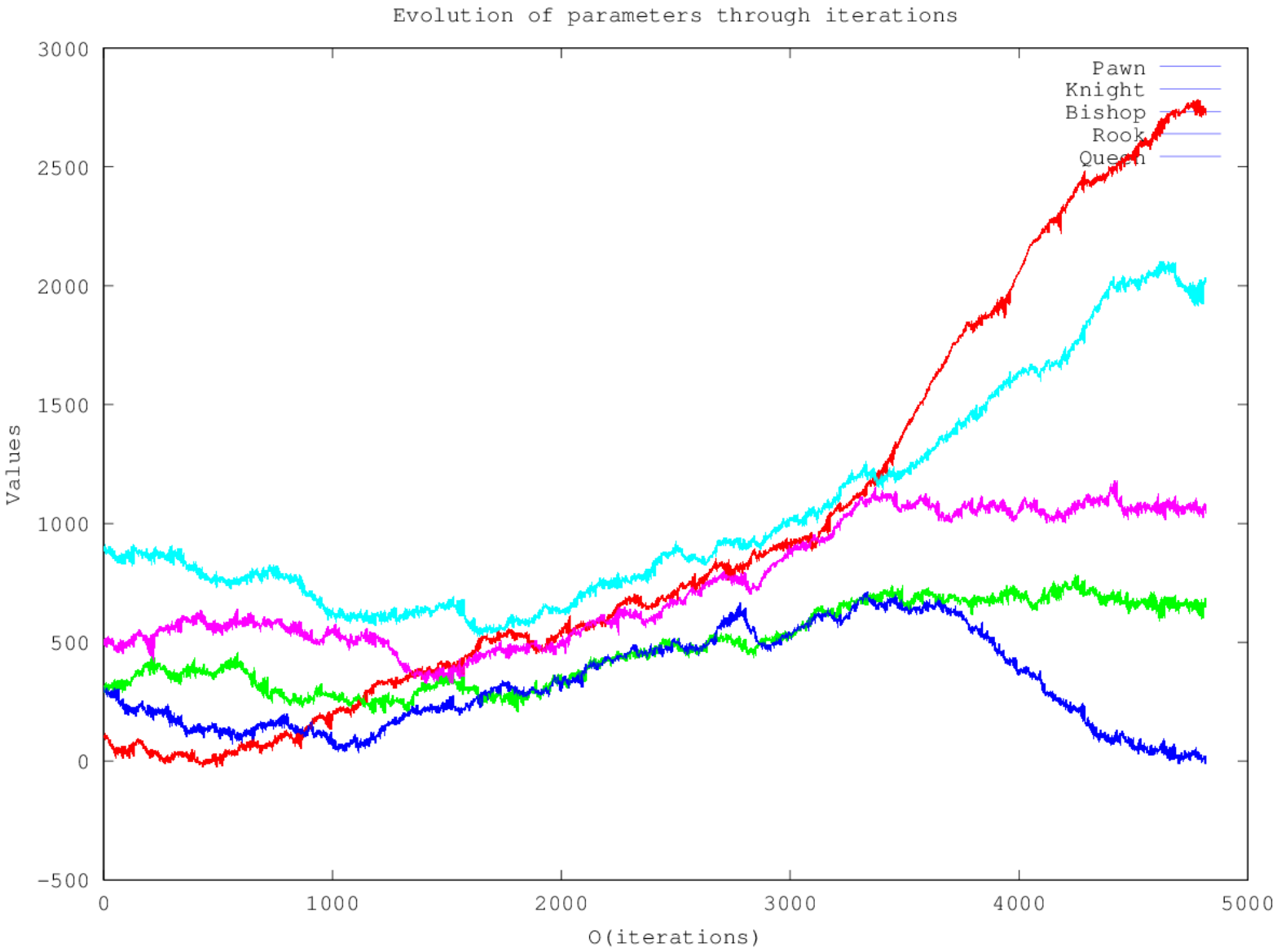
4. Future work

Set-up

- Search depth: n = 1
- One day and a half running on a standard laptop
- Optimized parameters: pawn(100), knight(300), bishop(300), rook(500) and queen(900) values
- Evolution strategy parameters: $\mu = \frac{1}{2}\lambda = 4$
- Static strategy parameters: RAND(-15, 15)

Evolved AI

- Original AI against human
- Evolved AI


Evolution of parameters through iterations
Pawn
Knight
Bishop
Rook
Queen
Values
O(iterations)
3000
2500
2000
1500
1000
500
0
-500
0
1000
2000
3000
4000
5000

1 Introduction

2 Implementation / Strategy
- Tools
- Architecture
- Board evaluation / parameters evaluation

3 Results
- Set-up
- Evolved AI

4 Future work

Future work

- Stabilize the algorithm
  - Boundaries for values
- Evolving strategy parameters

References

- Hallam Nasreddine, Hendra Suhanto Poh and Graham Kendall : Using an Evolutionary Algorithm for the Tuning of a Chess Evaluation Function Based on a Dynamic Boundary Strategy http://red.cs.nott.ac.uk/ gxk/papers/ieeecis2006.pdf
- DAVID B. FOGEL, FELLOW, IEEE, TIMOTHY J. HAYS, SARAH L. HAHN, AND JAMES QUON : A Self-Learning Evolutionary Chess Program http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1360168
- Graham Kendall, Glenn Whitwell : An Evolutionary Approach for the Tuning of a Chess Evaluation Function using Population Dynamics http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=934299

Questions

Any questions or suggestions?