

An Apology for Chess

Thomas Desautels

Gatsby Unit

13 November, 2014

The Indictment

- “Chess is too difficult to be a game and not serious enough to be a science or an art.” – Napoleon Bonaparte
- “[Chess] is a foolish expedient for making idle people believe they are doing something very clever, when they are only wasting their time.” – George Bernard Shaw, *The Irrational Knot*
- “Chess is not a game but a disease.” – Sir Henry Campbell-Bannerman
- “... as elaborate a waste of human intelligence as you could find anywhere outside an advertising agency.” – Raymond Chandler, *The Long Goodbye*

Quotes from <https://sites.google.com/site/renochessclub/sayings>

The Odds Against Success

- It takes many years of study and more or less total devotion to chess to become a world-class player.
- Even if you're the best human, you still will get thrashed by a computer.
 - Hikaru Nakamura (Currently #9 human by rating) & Rybka vs. Stockfish: 0+ 2- 2=¹

¹<http://www.chess.com/news/stockfish-outlasts-nakamura-3634>

Where Does That Leave Us?

Two obvious conclusions:

- Chess is a pointless waste of your time, since being the “best” is impossible; or
- Chess is worth playing if something about it makes you happy.

Perhaps unsurprisingly, I’m going to argue in favor of the second.

Decision Trees, Heuristics, and Computer Chess

- Fully deterministic, sequential, no hidden information.
- In principle, could be min-maxed to death and solved.
- Due to high branching factor and search depth, use:
 - Opening books
 - Deep, well-pruned search trees for move evaluation, using carefully-tuned heuristics for positional evaluation
 - Exact tables for solving simplified positions perfectly.

Put this together well enough, with enough computing power, and you can beat the best humans.

OK...

Lots of problems have such properties, and watching a computer solve them is boring.
I want to *play*, and I want to appreciate the experience. What's there to appreciate?

Human Executable Algorithms in Chess

Don't laugh! Algorithms (and algorithmic proofs!) are elegant.

You can often solve seemingly difficult situations by steering the game towards the many basic situations you have mastered. [IM Jeremy Silman]

A rather familiar idea: reduce difficult problems to the known solutions of a set of simple problems.

Importantly, these simple problems often have algorithmic solutions which are *iterative* and *monotonic*.

Posed More Mathematically

Does there exist a sequence of moves (and in particular, an iterative algorithm!) which either:

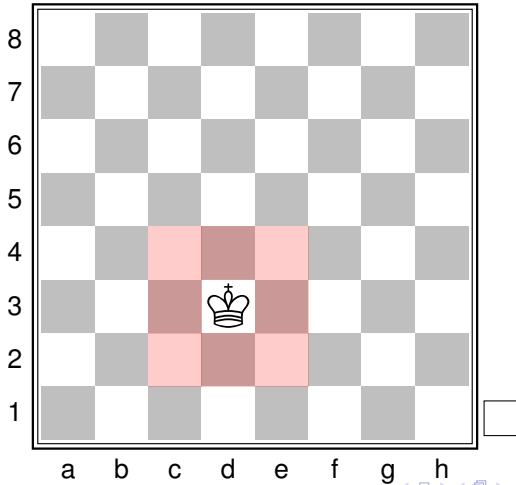
- 1 produces a checkmate; or
- 2 forces a simplification to a state in which there exists such a solution?

Our agenda

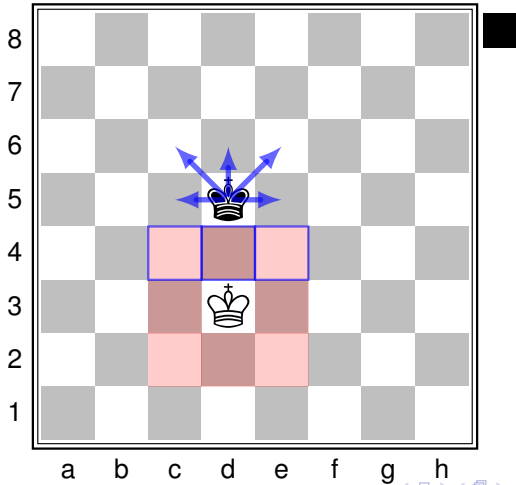
- Maintaining the Opposition
- Using the Opposition to advance the king
- Using the Opposition to advance a central pawn and promote

All in the endgame.

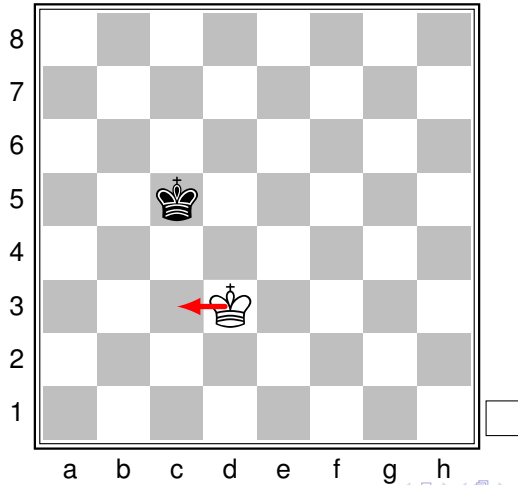
How Kings Move



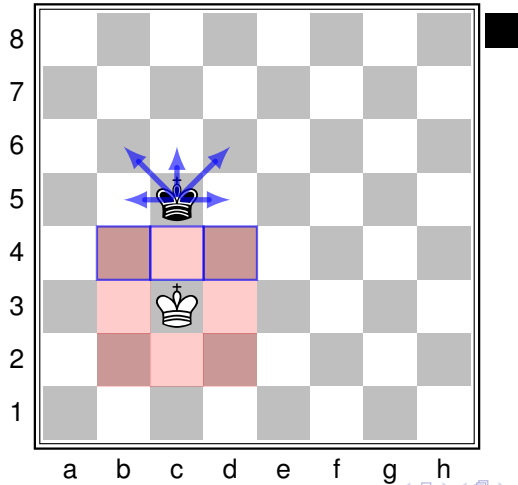
Two Kings in Opposition



Maintaining the Opposition



The Opposition has been Maintained



Algorithm 1

MaintainOpposition

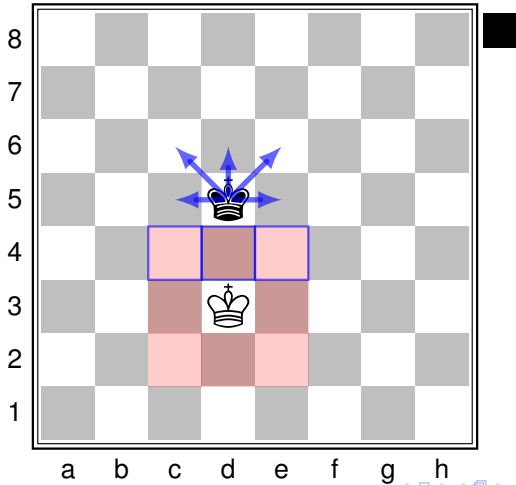
for $t=1:\infty$ **do**

$\text{MoveVector} = \text{BKPos}(t) - \text{WKPos}(t) - (0,2)$

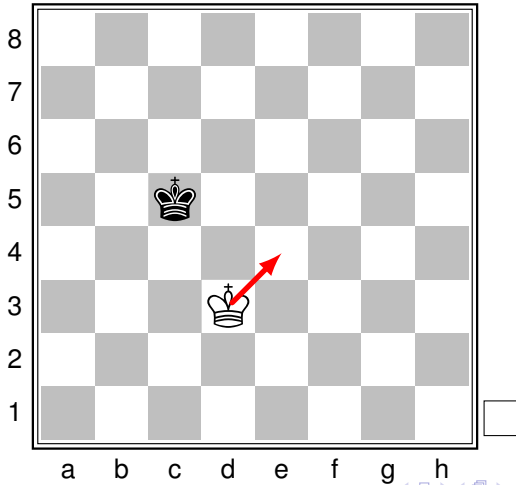
$\text{WKPos}(t+1) = \text{WKPos}(t) + \text{MoveVector}$

end for

Using the Opposition to Advance the King



An Opportunity to Advance

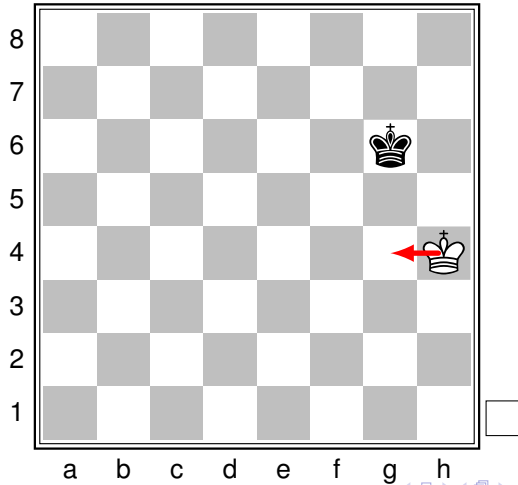


Algorithm 2

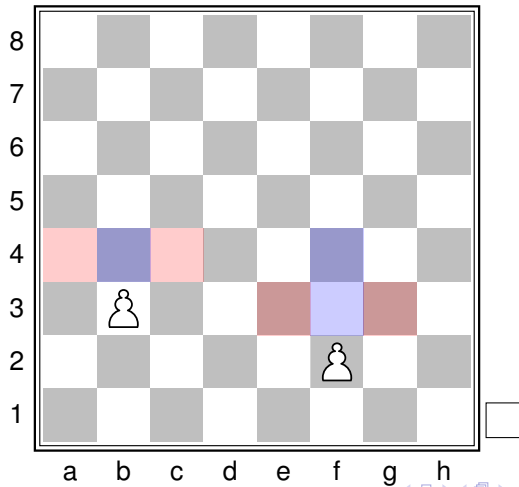
AdvanceKingWithOpposition

```
while WKPos(t),y < 8 do  
  if BKPos(t),y - WKPos(t),y <= 1 then  
    WKPos(t+1) = WKPos(t) + (0,1)  
  else  
    if BKPos(t) = WKPos(t) + ({-1,0,1},3) then  
      WKPos(t+1) = WKPos(t) + ({-1,0,1},1)  
    else  
      if BKPos(t) = WKPos(t) + ({-3,-2,-1,1,2,3},2) then  
        Try: WKPos(t+1) = WKPos(t) + ({-1,0,1,-1,0,1},1)  
        Catch: WKPos(t+1) = WKPos(t) + ({ , ,-1,1, , },0)  
      end if  
    end if  
  end if  
end while
```

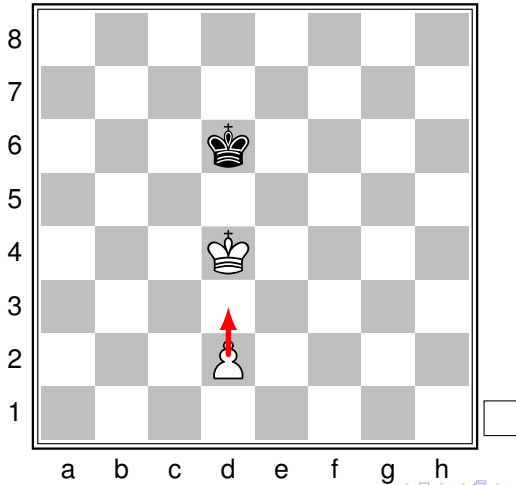
The Catch Condition: Hitting The Edge of the Board



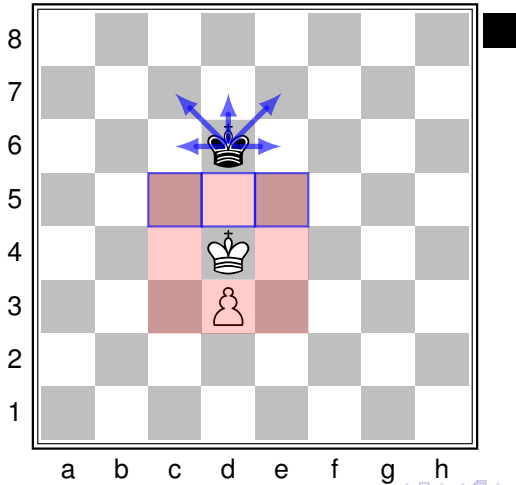
How Pawns Move



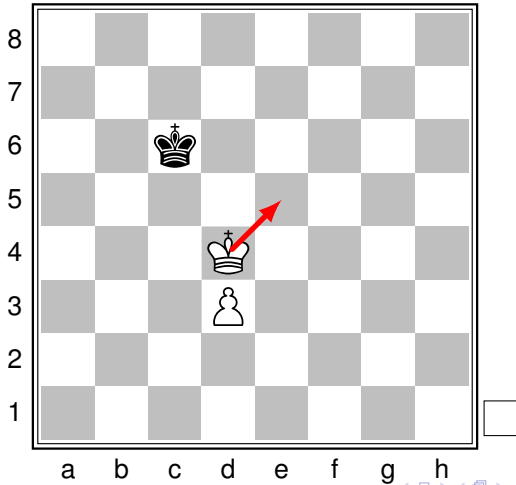
Using the Opposition to Advance And Promote



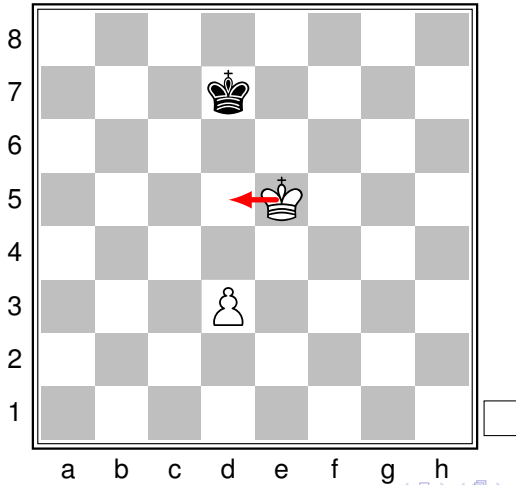
Using the Opposition to Advance And Promote II



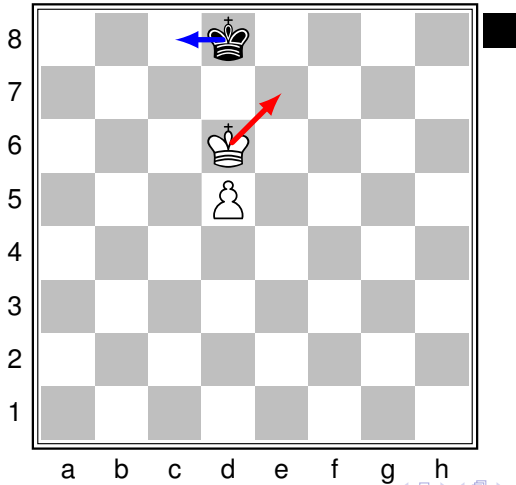
Using the Opposition to Advance And Promote III



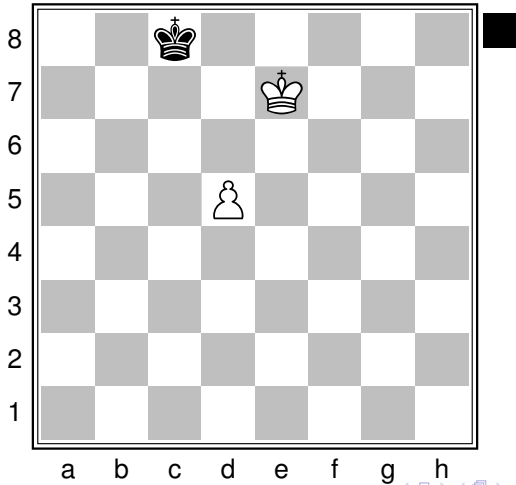
Using the Opposition to Advance And Promote III



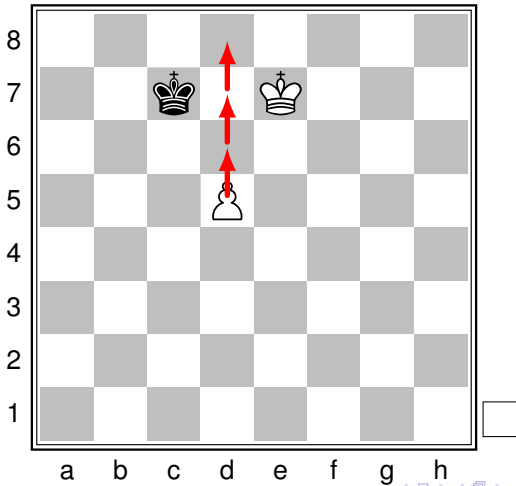
Using the Opposition to Advance And Promote IV



Using the Opposition to Advance And Promote V



Using the Opposition to Advance And Promote VI



(Fuzzy Pseudocode for) Algorithm 3

AdvanceCentralPawnAndPromote

```
while NotPromoted do  
  if not BHasOpposition and PromPathNotSafe then  
    if SafeToAdvanceKing then  
      AdvanceKing  
    else  
      if PawnMoreThan2RanksBack then  
        AdvancePawn  
      else  
        TakeOpposition  
      end if  
    end if  
  else  
    AdvancePawn  
  end if  
end while
```

Summary

- Chess has elegant, iterative algorithms in the endgame, with proven results.
- A good strategy is to simplify to a won endgame (if you're winning) or a drawn endgame (if you're losing).
- From such a point, sequentially apply these algorithms to produce the desired result.