Going Deep
An Approach to studying Search vs. Knowledge

Algorithmics lunchtalk
J. Renze Steenhuisen
February 29, 2008
Some Quote...

“Remarkable follow-up research on previous work done on deep-search behaviour using chess programs was published in 2005 by Steenhuisen...” (Guid and Bratko, 2007)
A Chess Position
Question: Search vs Knowledge

- How do chess programs work?
  - 'Brute force’ searching
  - Heuristic evaluation

- How to improve the playing strength?
  - Increase search depth
  - Increase evaluation quality
Computer Chess
Computer Chess
Computer Chess
Computer Chess
## Example Output

<table>
<thead>
<tr>
<th>Move</th>
<th>Time</th>
<th>Analysis</th>
<th>Moves</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0.04</td>
<td>2.77</td>
<td>1. c6 Kd6 2. Bxc7+ Nxc7 3. Kf3 g4+</td>
</tr>
<tr>
<td>6-→</td>
<td>0.06</td>
<td>2.77</td>
<td>1. c6 Kd6 2. Bxc7+ Nxc7 3. Kf3 g4+</td>
</tr>
<tr>
<td>7</td>
<td>0.07</td>
<td>2.73</td>
<td>1. c6 Kd6 2. Bxc7+ Nxc7 3. Kf3 g4+</td>
</tr>
<tr>
<td>7</td>
<td>0.14</td>
<td>2.75</td>
<td>1. Kf3 gxh4 2. gxh4 Nd4+ 3. Ke3 Nxd5+</td>
</tr>
<tr>
<td>7-→</td>
<td>0.16</td>
<td>2.75</td>
<td>1. Kf3 gxh4 2. gxh4 Nd4+ 3. Ke3 Nxd5+</td>
</tr>
<tr>
<td>8</td>
<td>0.24</td>
<td>2.72</td>
<td>1. Kf3 Nd4+ 2. Ke3 Nce6 3. hxf5 fxe5</td>
</tr>
<tr>
<td>8</td>
<td>0.32</td>
<td>3.02</td>
<td>1. Bxc7 Nxc7 2. g4 hxe4 3. hxf5 fxg5</td>
</tr>
<tr>
<td>8-→</td>
<td>0.32</td>
<td>3.02</td>
<td>1. Bxc7 Nxc7 2. g4 hxe4 3. hxf5 fxg5</td>
</tr>
<tr>
<td>9</td>
<td>0.36</td>
<td>2.87</td>
<td>1. Bxc7 Nxc7 2. g4 gxh4 3. gxh5 f5</td>
</tr>
<tr>
<td>9-→</td>
<td>0.45</td>
<td>2.87</td>
<td>1. Bxc7 Nxc7 2. g4 gxh4 3. gxh5 f5</td>
</tr>
<tr>
<td>10</td>
<td>0.53</td>
<td>2.78</td>
<td>1. Bxc7 Nxc7 2. g4 gxh4 3. gxh5 f5</td>
</tr>
<tr>
<td>10-→</td>
<td>0.81</td>
<td>2.78</td>
<td>1. Bxc7 Nxc7 2. g4 gxh4 3. gxh5 f5</td>
</tr>
</tbody>
</table>
Question: Search vs Knowledge

• How do chess programs work?
  • ‘Brute force’ searching
  • Heuristic evaluation

• How to improve the playing strength?
  • Increase search depth
  • Increase evaluation quality
Increase Search Depth

- MiniMax → Alpha-Beta Search → PVS
- Transposition tables
- Search extensions and reductions
- Multi-threading and/or parallelism
- etc.
But playing strength? ...

Experiments for measuring performance increase when searching deeper:

- Self Play
- Go Deep
Self Play

EXPERIMENT:
Match two otherwise identical programs, with one search to depth $d$ and the other to depth $d + 1$.

RESULTS:
Increasing the search depth results in higher rating by 200-250 ELO-points per ply.

(Thompson, 1982; Condon and Thompson, 1983; Szabo and Szabo, 1988; Berliner et al., 1990; Mysliwietz, 1994; Scheaffer, 1996; Junghanns et al., 1997)
Self Play

(Thompson, 1982; Condon and Thompson, 1983; Heinz, 2001)
Go Deep

- $B(d)$ denotes the best move of iteration $d$
- Best-change rate $BC(d)$ is the probability that $B(d) \neq B(d - 1)$
- Fresh-best rate $FB(d)$ is the probability that $B(d) \neq B(j), \forall j < d$
Go Deep: Example

6  0.04  2.77  1. c6  Kd6  2. Bxc7+  Nxc7  3. Kf3  g4+
6-> 0.06  2.77  1. c6  Kd6  2. Bxc7+  Nxc7  3. Kf3  g4+
7  0.07  2.73  1. c6  Kd6  2. Bxc7+  Nxc7  3. Kf3  g4+
7  0.14  2.75  1. Kf3  gxh4  2. gxf4  Nd4+  3. Ke3  Nxd5+
7-> 0.16  2.75  1. Kf3  gxh4  2. gxf4  Nd4+  3. Ke3  Nxd5+
8  0.24  2.72  1. Kf3  Nd4+  2. Ke3  Nce6  3. hxg5  fxg5
8  0.32  3.02  1. Bxc7  Nxc7  2. g4  hgx4  3. hxg5  fxg5
8-> 0.32  3.02  1. Bxc7  Nxc7  2. g4  hgx4  3. hxg5  fxg5
9  0.36  2.87  1. Bxc7  Nxc7  2. g4  gxh4  3. gxh5  f5
9-> 0.45  2.87  1. Bxc7  Nxc7  2. g4  gxh4  3. gxh5  f5
10  0.53  2.78  1. Bxc7  Nxc7  2. g4  gxh4  3. gxh5  f5
10-> 0.81  2.78  1. Bxc7  Nxc7  2. g4  gxh4  3. gxh5  f5
Go Deep

Newborn’s hypothesis (Newborn, 1985):

\[ RI(d + 1) = \frac{BC(d + 1)}{BC(d)} \cdot RI(d) \]  \hspace{1cm} (1)
Go Deep

(Hyatt and Newborn, 1997; Heinz, 1998)
Go Deep

Crafty v19.3

Probability vs Ply depth (d)
Go Deep

![Graph showing probability vs. ply depth](image)

- Title: Go Deep
- Subtitle: Graph showing probability vs. ply depth
- X-axis: Ply depth (d)
- Y-axis: Probability
- Data source: Crafty v19.3
Go Deep

- $B(d)$ denotes the best move of iteration $d$
- Best-change rate $BC(d)$ is the probability that $B(d) \neq B(d - 1)$
- Fresh-best rate $FB(d)$ is the probability that $B(d) \neq B(j), \forall j < d$
Go Deep

Crafty v19.3

Ply depth (d)

Probability

Best Change

Fresh Best
Heinz’ hypothesis (Heinz, 1998):

\[
\frac{\text{BC}(d + 1)}{\text{BC}(d)} \approx \frac{\text{FB}(d + 1)}{\text{FB}(d)}
\]
Go Deep

![Graph showing the comparison of BC(d+1)/BC(d) and FB(d+1)/FB(d) with respect to search depth (d). The x-axis represents search depth (d), and the y-axis represents the rating increase factor. The red line represents BC(d+1)/BC(d), and the blue line represents FB(d+1)/FB(d). The graph shows fluctuations in the rating increase factor as the search depth increases.]
Go Deep

![Graph showing the ratio of BC(d+1)/BC(d) and FB(d+1)/FB(d) over search depth (d).]
Go Deep

![Graph showing the relationship between Rating Increase Factor and Search depth (d).](image)

- **Crafty: BC(d+1)/BC(d)**
- **Crafty: FB(d+1)/FB(d)**
- **Fritz: RI(d+1)/RI(d)**
A Look into the Future

Some *odd* results:

- Fit through Self-Play data: zero at 18 ply
- Fit through Best-Change data: zero at 28 ply
Go Wide on Best Change

![Graph showing probability against ply depth for different engines: Diep, Crafty, Rybka, Shredder.](image-url)
Go Wide on Fresh Best

![Graph showing the probability of different programs at various ply depths]

- Diep
- Crafty
- Rybka
- Shredder

Probability vs. Ply depth (d)
Increase Evaluation Quality

- Material score
- Piece-Square bonus
- Pawn structure
- King safety
- Material imbalance
- etc.
But playing strength? ...

Experiments for measuring performance increase with improved evaluation quality:
- ... nothing ... yet!
Aspiration

EXPERIMENT:
Measure the evaluation differences when searching to increasing depths.
## Aspiration: Example

<table>
<thead>
<tr>
<th>Move</th>
<th>Time</th>
<th>Score</th>
<th>Moves</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0.04</td>
<td>2.77</td>
<td>1. c6 Kd6 2. Bxc7+ Nxc7 3. Kf3 g4+</td>
</tr>
<tr>
<td>6-→</td>
<td>0.06</td>
<td>2.77</td>
<td>1. c6 Kd6 2. Bxc7+ Nxc7 3. Kf3 g4+</td>
</tr>
<tr>
<td>7</td>
<td>0.07</td>
<td>2.73</td>
<td>1. c6 Kd6 2. Bxc7+ Nxc7 3. Kf3 g4+</td>
</tr>
<tr>
<td>7-→</td>
<td>0.14</td>
<td>2.75</td>
<td>1. Kf3 gxh4 2. gxh4 Nd4+ 3. Ke3 Nxd5+</td>
</tr>
<tr>
<td>7-→</td>
<td>0.16</td>
<td>2.75</td>
<td>1. Kf3 gxh4 2. gxh4 Nd4+ 3. Ke3 Nxd5+</td>
</tr>
<tr>
<td>8</td>
<td>0.24</td>
<td>2.72</td>
<td>1. Kf3 Nd4+ 2. Ke3 Nce6 3. hxg5 fxg5</td>
</tr>
<tr>
<td>8-→</td>
<td>0.32</td>
<td>3.02</td>
<td>1. Bxc7 Nxc7 2. g4 hxg4 3. hxg5 fxg5</td>
</tr>
<tr>
<td>8-→</td>
<td>0.32</td>
<td>3.02</td>
<td>1. Bxc7 Nxc7 2. g4 hxg4 3. hxg5 fxg5</td>
</tr>
<tr>
<td>9</td>
<td>0.36</td>
<td>2.87</td>
<td>1. Bxc7 Nxc7 2. g4 gxh4 3. gxh5 f5</td>
</tr>
<tr>
<td>9-→</td>
<td>0.45</td>
<td>2.87</td>
<td>1. Bxc7 Nxc7 2. g4 gxh4 3. gxh5 f5</td>
</tr>
<tr>
<td>10</td>
<td>0.53</td>
<td>2.78</td>
<td>1. Bxc7 Nxc7 2. g4 gxh4 3. gxh5 f5</td>
</tr>
<tr>
<td>10-→</td>
<td>0.81</td>
<td>2.78</td>
<td>1. Bxc7 Nxc7 2. g4 gxh4 3. gxh5 f5</td>
</tr>
</tbody>
</table>
Aspiration

[Graph showing the relationship between window size and percentage for different depths in Crafty v19.3.]
Aspiration
Aspiration

Crafty v19.3

depth 2  depth 6  depth 10  depth 14  depth 18
depth 3  depth 7  depth 11  depth 15
depth 4  depth 8  depth 12  depth 16
depth 5  depth 9  depth 13  depth 17
Conclusions

- Chess is a cool research topic
- Many open issues
- Doing sound research is time consuming but worth the while
- Visualisation of your results is important
Research Agenda

- Test Newborn’s hypothesis (i.e., do self-play with Crafty v19.3)
- Relate evaluation quality with playing strength
- Develop new search algorithm which takes accuracy into account
References

- ...
Going Deep
An Approach to studying Search vs. Knowledge

Algorithmics lunchtalk
J. Renze Steenhuisen
February 29, 2008
Deep Search Experiment

![Graph showing the relationship between probability and search depth (d). The graph includes a red line for BC(d), a green line for Exponential fit to BC(d), and a blue line for Linear fit to BC(d).]