Game Playing in the Real World

Next time: Knowledge Representation

Reading: Chapter 7.1-7.3

What matters?

- Speed?
- Knowledge?
- Intelligence?
  - (And what counts as intelligence?)
- Human vs. machine characteristics

- The decisive game of the match was Game 2, which left a scare in my memory … we saw something that went well beyond our wildest expectations of how well a computer would be able to foresee the long-term positional consequences of its decisions. The machine refused to move to a position that had a decisive short-term advantage – *showing a very human sense of danger*. I think this moment could mark a revolution in computer science that *could earn IBM and the Deep Blue team a Nobel Prize*. Even today, weeks later, no other chess-playing program in the world has been able to evaluate correctly the consequences of Deep Blue's position. (Kasparov, 1997)

Quotes from IEEE article

- Why, then, do the very best grandmasters still hold their own against the silicon beasts?
- *The side with the extra half-move* won 3 games out of four, corresponding to a 200-point gap in chess rating – roughly the difference between a typically grandmaster (about 2600) and Kasparov (2830)
- An opening innovation on move nine gave Kasparov not merely the superior game but *one that Fritz could not understand*
- Kasparov made sure that *Fritz would never see the light at the end of that tunnel by making the tunnel longer.*

Deep Blue – A Case Study

Goals

- Win a *match* against human World Chess Champion
- Under regulation time control
  - No faster than 3 min/move

Background

- Early programs emphasized emulation of human chess thought process
- 1970-1980: emphasis on hardware speed
  - Chess 4.5
  - Belle (1st national master program, early 80s)
  - mid-1980s: Cray Blitz, Hitech
- 1986-1996
  - Deep Thought, Deep Thought II
  - 1988: 2nd Fredkin Intermediate Prize for Grandmaster level performance
- 1996: Deep Blue
  - New chess chip, designed over a 3 year period

Problems to Overcome

- Gaps in chess knowledge
- Adaptation
- Speed

Design Philosophy

- Integrate the maximally possible amount of software-modifiable chess knoweldge on the chess chip

Deep Blue System Architecture

- Chess chip
  - Searched 2-2.5 million chess positions/second
- IBM RS/6000 SP supercomputer
  - Collection of 30 workstations (RS 6000 processors)
  - Each processor controlled up to 16 chess chips
  - 480 chess chips total
- Maximum speed: 1 billion chess positions/second
- Sustained speed: 200 million chess positions/second

Search

Software/hardware mix:

- 1st 4 plies: 1 workstation node
- 2nd 4 plies: in parallel over 30 workstations nodes
- Remaining plies: in hardware

50 nanoseconds in a 0.6-micron, three-metal

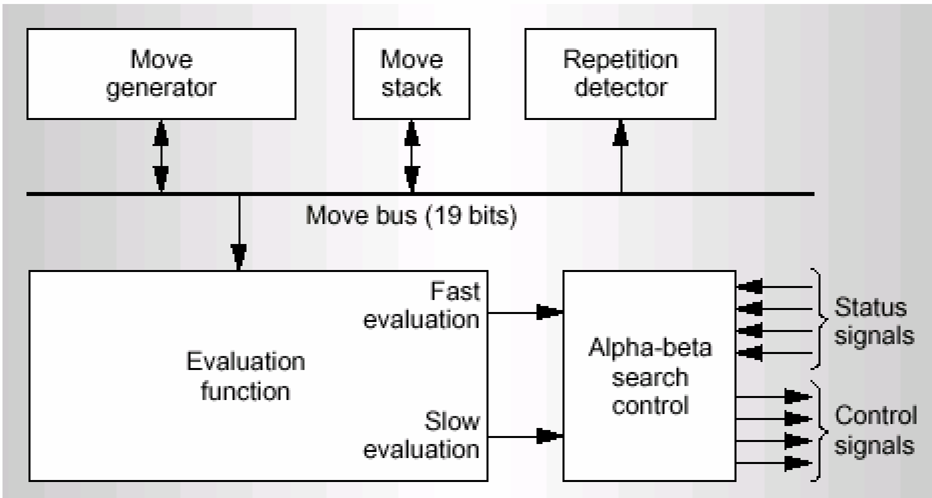


Figure 1. Block diagram of the chess chip.

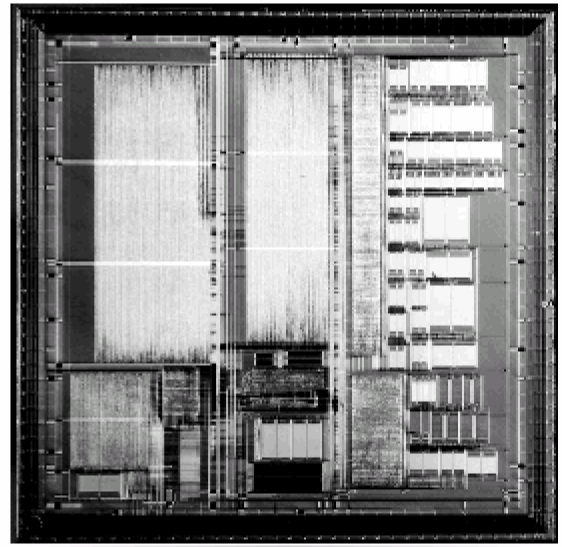


Figure 2. Die photo of the chess chip.

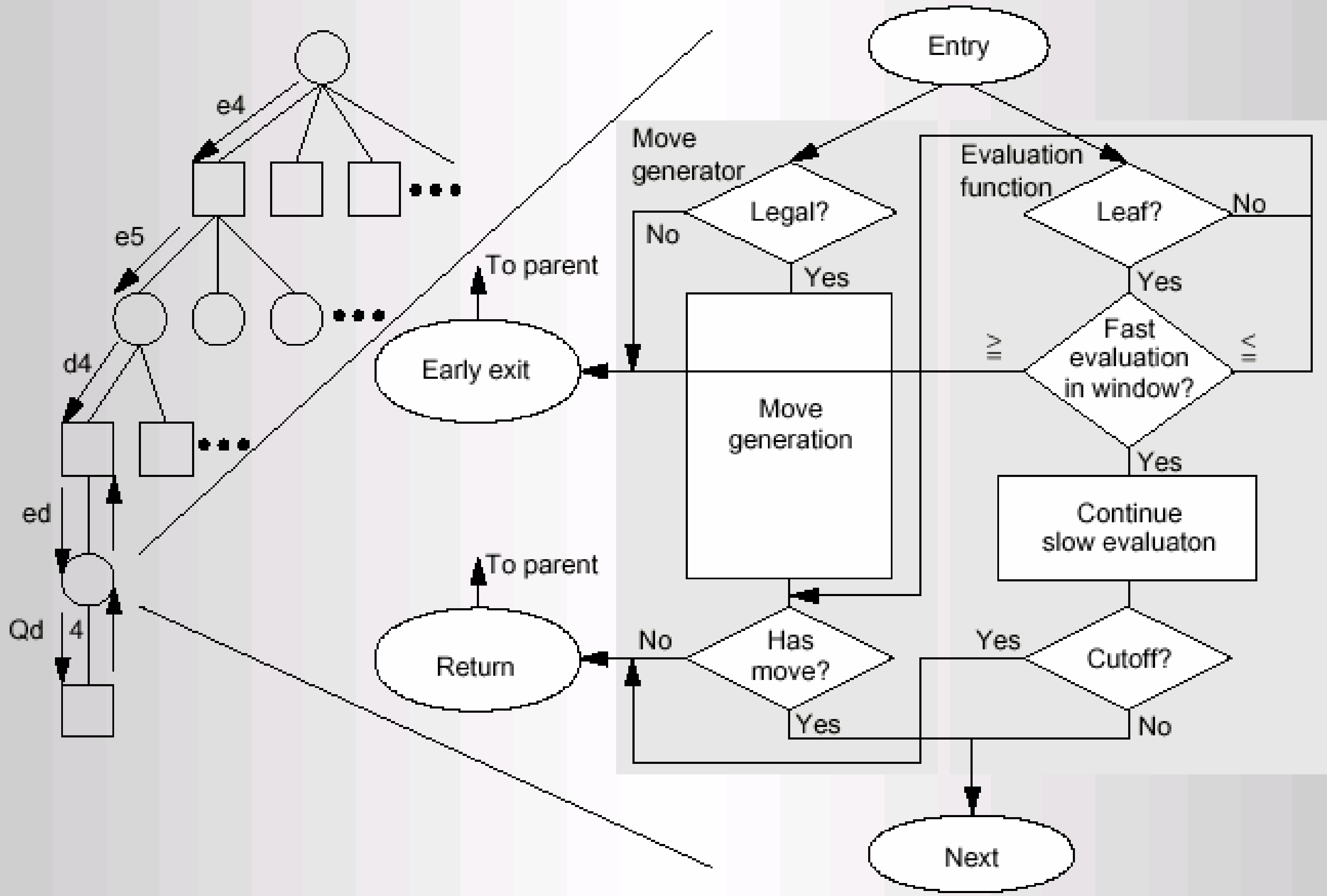


Figure C. A chess chip's basic search algorithm: search tree (left), flow chart (right).

Evaluation Functions

- An ideal evaluation function would rank terminal states in the same way as the true utility function; but must be *fast*
- Typical to define features, & make the function a linear weighted sum of the features

Chess

- F_1=number of white pieces
- F_2=number of black pieces
- $F_3=F_1/F_2$
- F_4=number of white bishops
- F_5=estimate of “threat” to white king

Weighted Linear Function

- $Eval(s) = w_1 F_1(s) + w_2 F_2(s) + \ldots + w_n F_n(s)$
    - Given features and weights
- Assumes independence
- Can use *expert knowledge* to construct an evaluation function
- Can also use *self-play* and *machine learning*

Evaluation functions in Deep Blue

- Opening moves
- Evaluation during a game
  - 8000 feature evalution
  - E.g., square control, pins, x-rays, king safety, pawn structure, passed pawns, ray control, outposts, pawn majorigy, rook on the 7th blockade, restraint, color complex, trapped pieces,…..
  - Weighted non-linear function
- End games
  - Large endgame databases of solved positions, with 5 or 6 pieces.

Using expert decisions

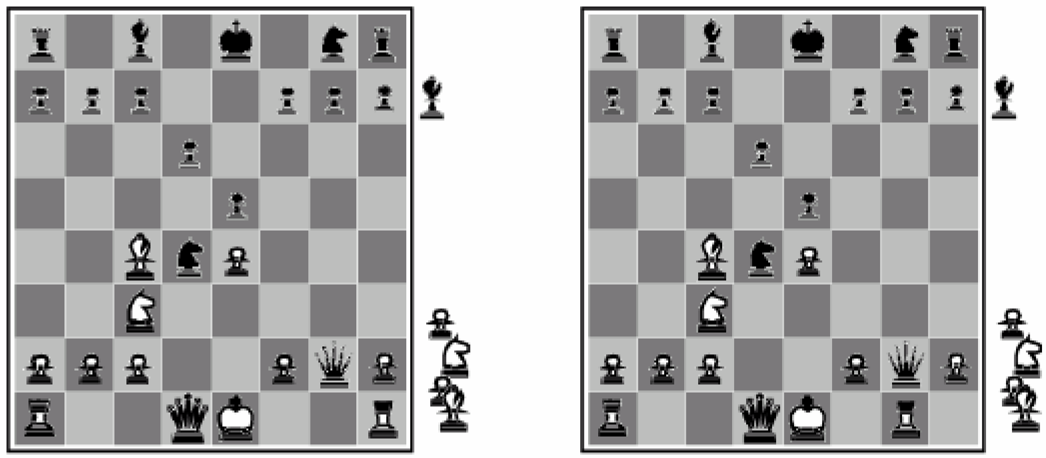
- Database of 4000 opening moves
- Extended book
  - 700,000 Grandmaster games
  - For each of the first 30 or so moves, compute an evaluation for each move that has been played
  - Scores are used as evaluation fns to bias the search

A Sample of Evaluation features

- The number of times a move has been played
- Relative number of times a move has been played
- Strength of the players that played the moves
- Recentness of the move
- Results of the move
- Commentary on the move
- Game moves vs. commentary moves

Quiescence Search

A fixed cut-off depth is not very robust, and can lead to problems. Example:



(a) White to move (b) White to move

- evaluate states that are unlikely to have wild swings in value; identify instability, e.g. with ``capture positions'' in chess

Impact

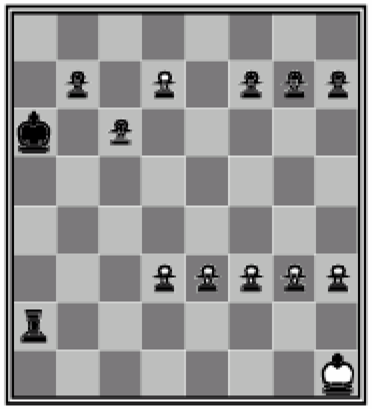
- Program with quiescence search matches a program without quiescence search but searching 4 plies deeper
- QS increases #positions searched 2-4 times
- 4 more plies of search is up to a thousand times increase

Quiescence Search in Deep Blue

- 1997 match
  - Software search extended to about 40 plies along forcing lines
  - Nonextended search researched about 12 plies

Horizon Effect: *Extensions*

- Opponent has a move that is *ultimately unavoidable*, but search cannot tell because of *depth limit*
  - use heuristics to do search in particular situations (e.g. on promising branches)



Black to move

Figure 6.9 The horizon effect. A series of checks by the black rook forces the inevitable

Quotes from IEEE article

- Why, then, do the very best grandmasters still hold their own against the silicon beasts?
- *The side with the extra half-move* won 3 games out of four, corresponding to a 200-point gap in chess rating – roughly the difference between a typically grandmaster (about 2600) and Kasparov (2830)
- An opening innovation on move nine gave Kasparov not merely the superior game but *one that Fritz could not understand*
- Kasparov made sure that *Fritz would never see the light at the end of that tunnel by making the tunnel longer.*

Discussion

- How intelligent are chess playing programs?
- How like humans are programs?
- How to explain the 1997 match vs the 2002 match with Fritz?
- Is chess playing a solved problem?
- What would be some next directions?

Applying game playing to other types of games

- Imperfect knowledge
  - Bridge
- Games with a random element
  - Backgammon