**1/26/18**

09-Adversarial-Search-III

2 kinds of evaluations:

* static eval (heuristics)
* dynamic eval - look ahead is involved
  + mini max
    - alpha beta pruning
      * don't bother with the parts of the tree that don’t affect the results

Dynamic programming

* use hashing

**Zobrist Hashing:**

* avoid recomputing values for some states by saving their values
  + especially those within 3 - 4 ply of the current state
* Use hash table to save: [state, value, ply-used]
* for hashing function use Zobrist hashing function
  + for each piece on the board - **exclusive-or** the current key with a pre-generated random number.
  + generate child hash from parent
    - take position of change, hash the position’s value (exclusive or this too)
    - take position of destination, hash the position’s value (exclusive or this)
  + hash values for similar boards are very different
  + hash values can be efficiently computed with an incremental approach

**(Slide 4 - Pseudo Code for zobrist hashing)**

**Game Playing Issues**

* Representing moves: a (Source, Destination) approach works for some games when the squares on the board have been numbered.
* Source: The number of the square where a piece is being moved from.
* Destination: The number of the square where the piece is being moved to.

(For Othello, only the destination is needed.)

* Opening moves:
  + Some programs use an “opening book”
  + Some competitions require that the first 3 moves be randomly selected from a set of OK opening moves, to make sure that players are “ready for anything”
* Regular maximum ply are typically 15-20 for machines, with extra ply allowed in certain situations.
* Static evaluation functions in checkers or chess may take 15 to 20 different features into consideration.

**Scoring Polynomial**

* f ( s ) = a1 ADV + a2 APEX + a3 BACK + . . . + a16 THRET
* There are 16 terms at any one time.
* coefficients of some terms can be changed to make them more important
  + coefficients are powers of 2
  + ordered so no two are equal at any time
* They are automatically selected from a set of 38 candidate terms.
* 26 of them are described in the following 3 slides. **(8-10**)
* program keeps track of all values to see if they correlate with improvement in game position
  + if so, value goes up. if not, it goes down