Defeating Mastermind

By Justin Dowell

MATH 3210

Overview

- What is Mastermind?
- Sample Game
- Algorithms in General
- The First Guess and Optimization
- Specific Algorithms
- Games as a Whole
- Review
- Future Work

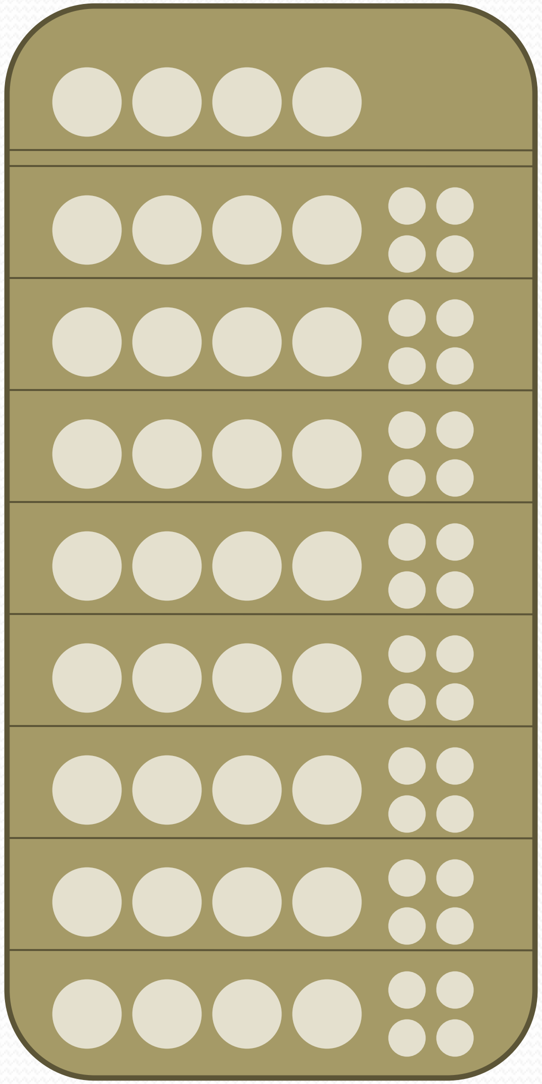
What is Mastermind?

- A code-breaking game for two players, Codemaster and Codebreaker
- Played on a board with eight rows of four holes; the guessing space for the Codebreaker
- One extra row shielded from Codebreaker keeps Codemaster’s hidden code
- Code & Guesses are made with four pegs
- Pegs can be any of any six colors

What is Mastermind?

- To the side of the guessing rows, Codemaster's reply section exists. Also four holes.
- Replies are given with 'keys':
  - Black key for a correct color peg in correct place
  - White keys for a correct color peg in wrong place
  - No key for an incorrect color peg
- Codebreaker's goal is to crack Codemaster's code before running out of room.

Sample Game: The Board



Hidden Code

Guess 8

Reply to Guess 8

Guess 7

Reply to Guess 7

Guess 6

Reply to Guess 6

Guess 5

Reply to Guess 5

Guess 4

Reply to Guess 4

Guess 3

Reply to Guess 3

Guess 2

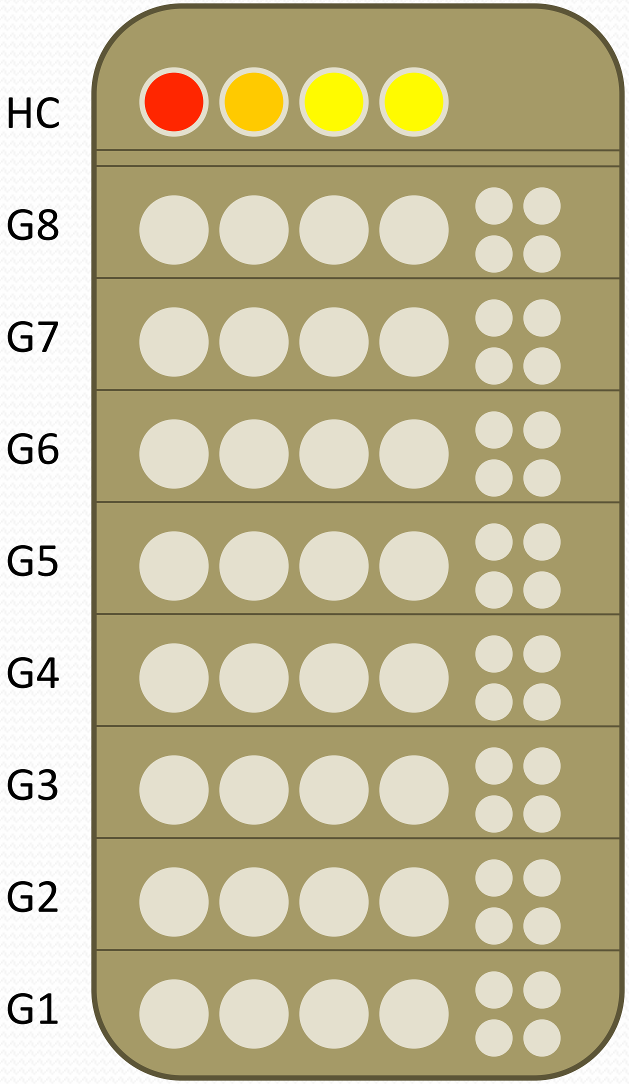
Reply to Guess 2

Guess 1

Reply to Guess 1

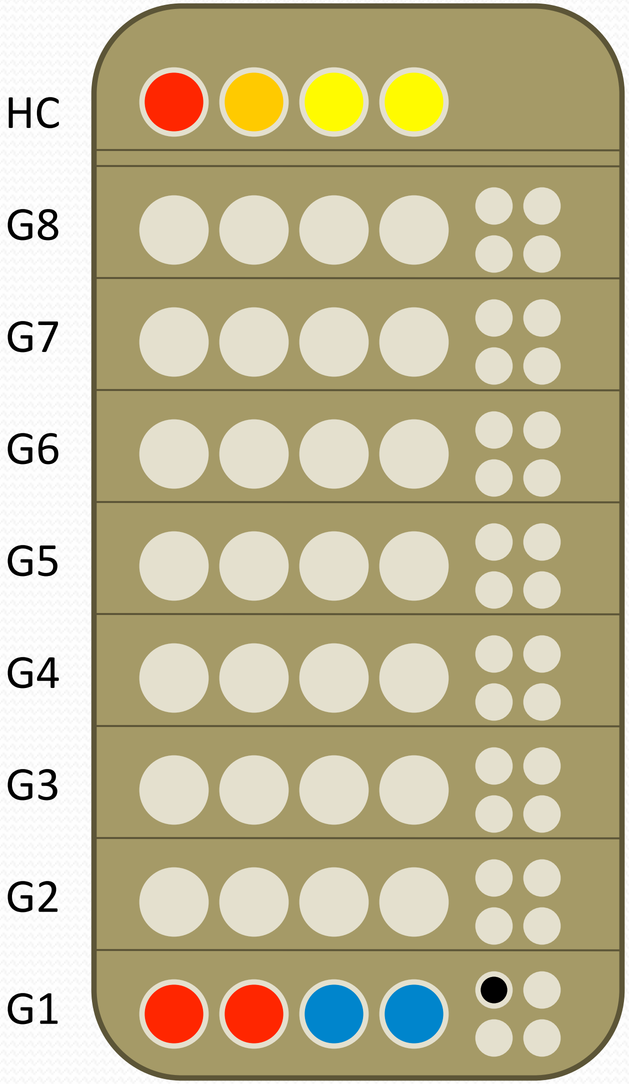
Sample Game: Turn 0

- Prior to the first turn, the Codemaster sets up hidden code, ROYY



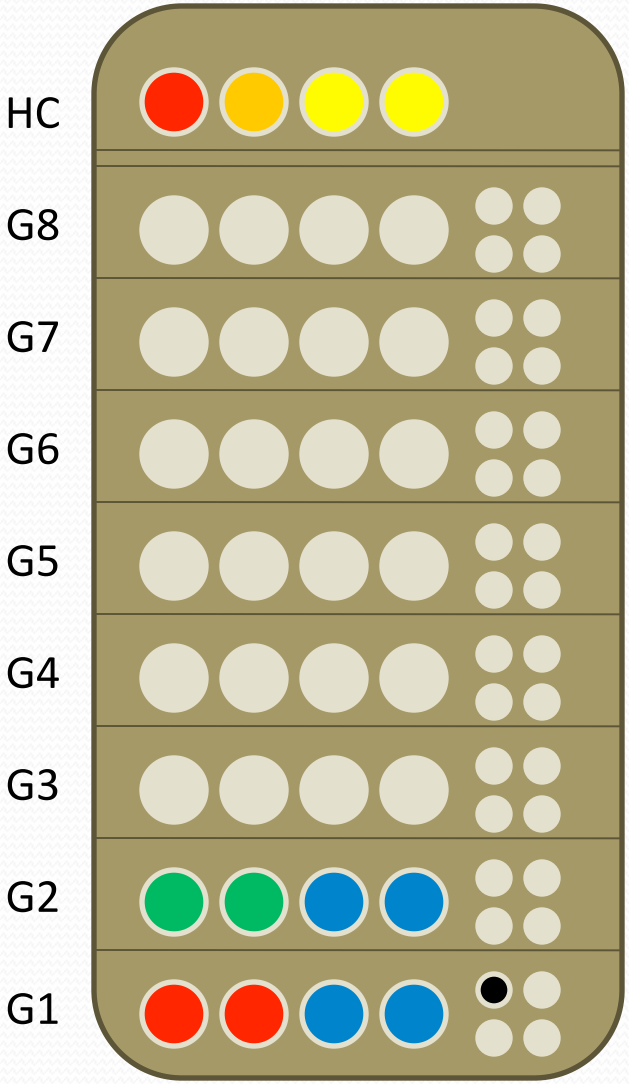
Sample Game: Turn 1

- Codebreaker’s guess: RRBB
- Reply: 1 Black, 0 White
- So there’s a red or blue in the code, but only one. “Which one is it?”, Codebreaker wonders.



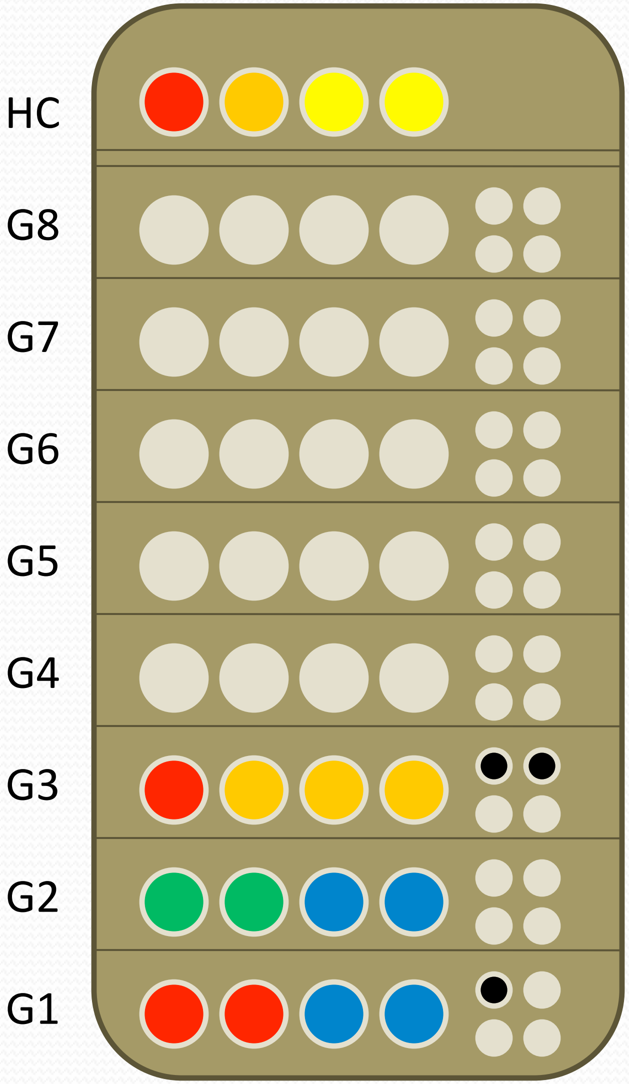
Sample Game: Turn 2

- Codebreaker’s guess: GGBB
- Reply: 0 Black, 0 White
- So there are no greens or blues in the code. That also means there’s one red, based on the first guess. “Which place is it in?”, Codebreaker wonders.



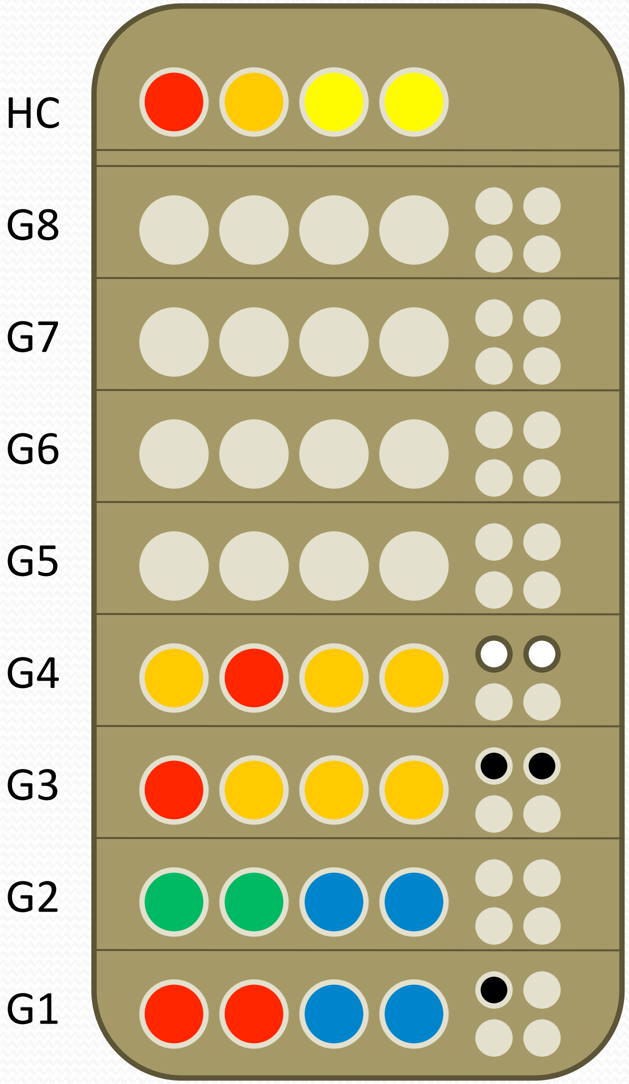
Sample Game: Turn 3

- Codebreaker's guess: ROOO
- Reply: 2 Black, 0 White
- "Now I know the red peg is first, and there's one orange peg in the code. Where does the orange peg go, though?"



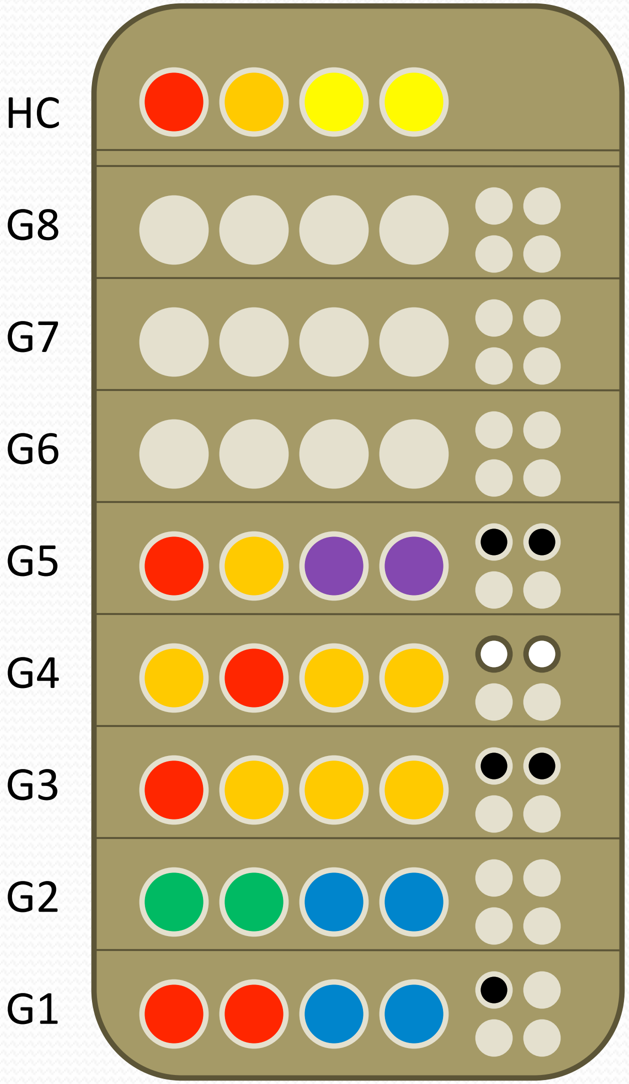
Sample Game: Turn 4

- Codebreaker’s guess: OROO
- Reply: 0 Black, 2 White
- When the correct orange is moved, it will earn a white peg.
- The Codebreaker moves the known red peg out of place, and finds the correct orange.



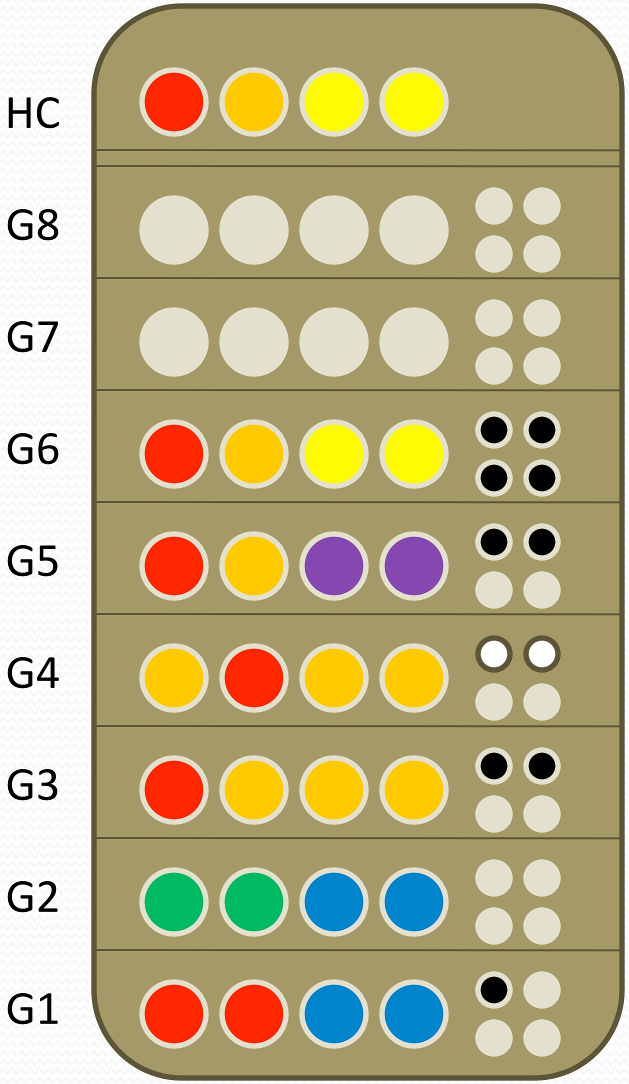
Sample Game: Turn 5

- Codebreaker’s guess: ROPP
- Reply: 2 Black, 0 White
- The Codebreaker tries to learn the color of the remaining two pegs. Only violet and yellow remain.
- …And after this guess, only yellow. “The code is ROYY.”



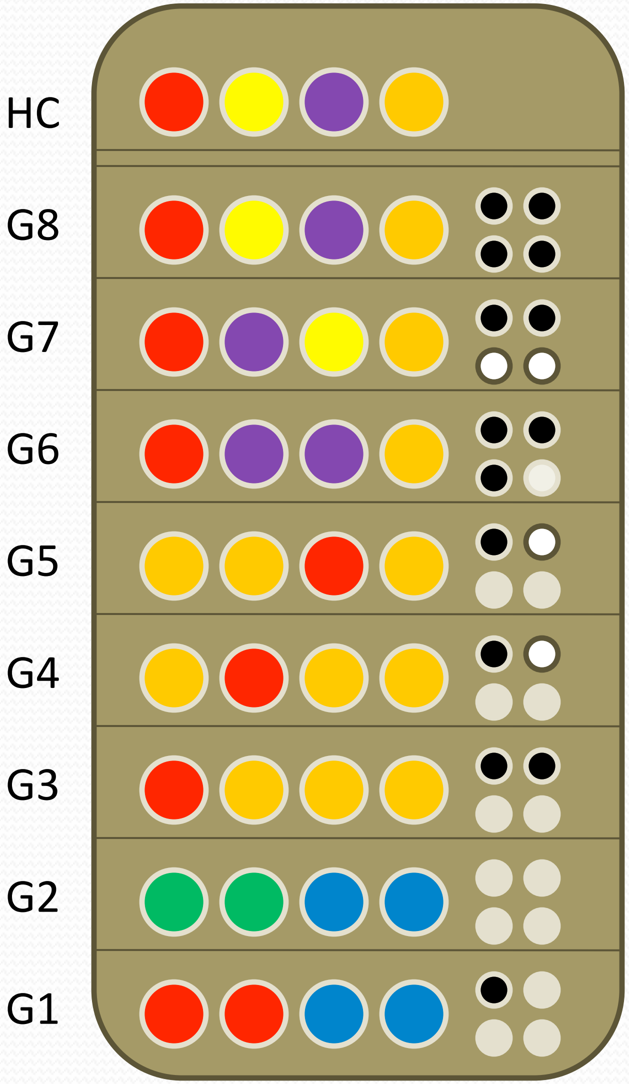
Sample Game: Turn 6

- Codebreaker's guess: ROYY
- Reply: 4 Black, 0 White
- The Codebreaker wins.
- It was lucky the orange was found fast.
- It was also a good thing the last two pegs weren't different colors, or it could have gone all the way to the last move.



Sample Game: What if…?

- It could have been a close call.
- Surely there’s some way for a computer to figure out good guesses?



Algorithms in General

- All algorithms for suggesting a ‘best’ Mastermind guess operate using similar principles.
- For each possible guess:
    - For each code that could still be the hidden code:
        - Calculate what the reply would be if this guess was guessed and this code was actually the hidden code.
    - Tally how many times each reply appears.
    - Rate the guess based on an algorithm-specific statistic about the different tallies.
    - If it’s better than the last guess, keep it!

The First Guess and Optimization

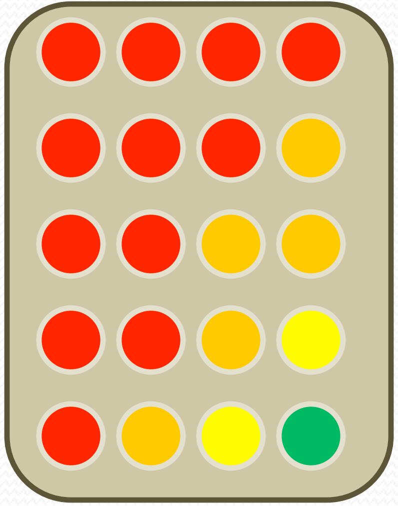
- Using brute force, the best first guess can be very expensive to calculate.
- 6^4 Guesses * 6^4 Codes = 1296^2 Replies
- That’s 1679616 replies to check! Larger games get even more expensive.
- How can this be improved?

The First Guess and Optimization

- As far as judging the worth of the first move, there isn't any statistical difference between one color to the next.
  - (RRRR = GGGG)
- Position doesn't really matter either.
  - (RROO = OORR, VVBY = VBYV)
- Only the unique patterns make any difference.
- What is actually obtained is based on the individual game, but the statistical worth of some guesses is identical before guessing.

The First Guess and Optimization

- Color and position don’t matter…
- …So there are really only 5 first guesses.
  - ‘AAAA’ (RRRR, OOOO, etc.)
  - ‘AAAB’ (RRRO, YYBY, etc.)
  - ‘AABB’ (RROO, VGGV, etc.)
  - ‘AABC’ (RROY, BBOR, etc.)
  - ‘ABCD’ (ROYG, GYRO, etc.)



- The same cannot be done for guesses after the first, as the game now contains additional information.

The First Guess and Optimization

Reply tallies

Guess	AAAA	AAAB	AABB	AABC	ABCD
(0, 0)	625	256	256	81	16
(0, 1)	0	308	256	276	152
(0, 2)	0	61	96	222	312
(0, 3)	0	0	16	44	136
(0, 4)	0	0	1	2	9
(1, 0)	500	317	256	182	108
(1, 1)	0	156	208	230	252
(1, 2)	0	27	36	84	132
(1, 3)	0	0	0	4	8
(2, 0)	150	123	114	105	96
(2, 1)	0	24	32	40	48
(2, 2)	0	3	4	5	6
(3, 0)	20	20	20	20	20
(4, 0)	1	1	1	1	1

Specific Algorithms

- MiniMax (Knuth 1977)
  - Bases value of a guess on the largest reply tally, with larger being worse.
  - In other words, the guess that does the best job in the worst case (pessimistic).
  - 5 turn maximum to solve the code.
  - High average number of turns.

Specific Algorithms

- MaxEnt (Bestavros and Belal 1986)
  - Bases value of a guess on Shannon Entropy.
  - Looks for maximum possible probable loss (in bits of information) of remaining hidden codes.
  - Maximum 6 turns.
  - Lower average than MiniMax.

Specific Algorithms

- Irving (Irving 1979)
  - Bases value of a guess on expected size of remaining hidden codes.
  - Squares the size of a reply tally, multiplies by probability of getting that reply, and summates for all replies.
  - Maximum 6 turns.
  - Lower average than MiniMax.

Specific Algorithms

- MaxParts (Kooi 2005)
  - Bases value of a guess on number of different replies possible.
  - Very simple, does surprisingly well.
  - Maximum 6 turns.
  - Lower average than most other algorithms.
  - First guess is 'lucky' (AABC instead of ABCD).

Specific Algorithms

- WideDev (Dowell 2009)
  - Bases value of a guess on number of different replies possible, then lowest standard deviation of tallies.
  - Maximum 6 turns.
  - Lower average than most other algorithms.
  - Suprisingly, does worse than MaxParts, but is more ‘stable’.

Specific Algorithms

- LongRect (Dowell 2009)
  - Bases value of a guess on number of different replies possible, then the largest reply tally, with larger reply tallies being worse.
  - MaxParts + MiniMax.
  - Maximum 6 turns.
  - Lower average than most other algorithms.
  - Like MinDev, does worse than MaxParts, but is more ‘stable’.

Games as a Whole

- To obtain statistics for how well an algorithm performs overall, it is necessary to run the algorithm to the end for each and every hidden code.
- The number of turns for each game is recorded, and statistics can be derived.

Games as a Whole

The algorithms in comparison:

Algorithm Name	#Guesses	Mean	StdDev	Max Turns
MiniMax	5778	4.458	0.607	5
MaxEnt	5723	4.416	0.631	6
Irving	5696	4.395	0.619	6
MaxParts	5668	4.373	0.649	6
WideDev	5685	4.387	0.627	6
LongRect	5681	4.383	0.623	6
OptDepth	????	4.341	?	5
OptMean	????	4.340	?	6

Review

- What is Mastermind?
- Sample Game
- Algorithms in General
- The First Guess and Optimization
- Specific Algorithms
- Games as a Whole

Future Work

- Score-based algorithm (based on black and white keys most probably received).
- Genetic Algorithm to solve the code.
- Checking more than one move ahead.
- Algorithm that combines all possible measurements and alters coefficients to converge on optimal strategy.
- The big question: Can optimal guesses be found without looking all the way through the game?

References

- Knuth, D.E. (1976). The computer as Mastermind, *Journal of Recreational Mathematics,* 9(1), 1–6.
- Irving, R.W. (1978-79). Towards an optimum Mastermind strategy, *Journal of Recreational Mathematics,* 11(2), 81–87.
- Koyama, K., and Lai, T.W. (1993). An optimal Mastermind strategy, *Journal of Recreational Mathematics,* 25, 251–256.
- Bestavros, A., and Belal, A., (1986). MasterMind: A Game of Diagnostic Strategies, *Bulletin of the Faculty of Engineering*, Alexandria University, Alexandria, Egypt.
- Heeffer, A., and Heeffer, H., (2007). NEAR-OPTIMAL STRATEGIES FOR THE GAME OF LOGIK, *ICGA Journal*. doi: 10.1.1.71.9209
- Kooi, B., (2005). Yet another mastermind strategy, *ICGA Journal,* 28(1), 13–20.

Questions?

The End!