Report: Build an Adversarial Game Playing Agent

I Selected: Advanced Heuristic
100 Matches Command: python run_match -f -r 100 -o MINIMAX

	A	B	C	D	E	F	G
			d = depth		**Winning Rate of Matches**		**Remark**
	Algorithm	**Heuristic**	**Parameter**	**Opponent**	**200**		
	AlphaBeta(Baseline)	#own_moves - #opp_moves	a=1 b=1 d=3	Minimax	41.20%		
	AlphaBeta	liberties of liberties	a=1 b=2 d=2	Minimax	45.80%		
	AlphaBeta	liberties of liberties	a=1 b=2 d=3	Minimax	61.20%		
	AlphaBeta	liberties of liberties	a=1 b=2 d=4	Minimax	56.30%		
	AlphaBeta	liberties of liberties + collapse	a=1 b=2 d=2	Minimax	41.50%		
	AlphaBeta	liberties of liberties + collapse	a=1 b=2 d=3	Minimax	62.50%		**Best**
	AlphaBeta	liberties of liberties + collapse	a=1 b=2 d=4	Minimax	55%		

Q1. What features of the game does your heuristic incorporate, and why do you think those features matter in evaluating states during search?

- I increase the importance of the opponent's liberties and collapsing liberties.
- This heuristic tend to make my agent to aggressive and decreasing opponent’s liberties.
- I force to select center position that maximize the liberties at game started.

Q2. Analyze the search depth your agent achieves using your custom heuristic. Does search speed matter more or less than accuracy to the performance of your heuristic?

- Search depth is proportional to decision speed.
- Increasing search depth doesn't guarantee the increase of winning rate, But decreasing drops the winning rate.
- Larger the search depth, Slower speed of decision
- **I think it's okay if the search speed is not too slow.**