

Set #1

Match #	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	Random	9	1	10	0	8	2	9	1
2	MM_Open	5	5	7	3	6	4	7	3
3	MM_Center	7	3	9	1	9	1	10	0
4	MM_Improved	10	0	8	2	8	2	7	3
5	AB_Open	6	4	4	6	4	6	3	7
6	AB_Center	7	3	5	5	5	5	5	5
7	AB_Improved	6	4	5	5	5	5	6	4
Win Rate:		71.4%		68.6%		64.3%		67.1%	

I find the values of legal player moves and opponent moves . then I use the following formula to calculate score

$(\text{player_multiplier} * \text{player_moves}) - (\text{opponent_multiplier} * \text{opponent_moves})$

Player_multiplier and opponent_multiplier are coefficient values that are passed in the dunctions body.

Coefficient values

custom_score

Player_multiplier = 0

opponent_multiplier = 2

custom_score_2

Player_multiplier = 1

opponent_multiplier = 1

custom_score_3

Player_multiplier = 2

opponent_multiplier = 1

Set#2

Match #	Opponent	AB_Improved		AB_Custom		AB_Custom_2		AB_Custom_3	
		Won	Lost	Won	Lost	Won	Lost	Won	Lost
1	Random	7	3	9	1	9	1	8	2
2	MM_Open	7	3	9	1	7	3	8	2
3	MM_Center	6	4	8	2	8	2	9	1
4	MM_Improved	6	4	7	3	9	1	8	2
5	AB_Open	6	4	5	5	5	5	6	4
6	AB_Center	6	4	7	3	4	6	6	4
7	AB_Improved	4	6	6	4	4	6	5	5
Win Rate:		60.0%		72.9%		65.7%		71.4%	

Algorithm

```
(player_moves**player_multiplier) - (opponent_moves**opponent_multiplier)
```

Instead of multiplying , we use the value to the power of coefficient.

It has better performance.

```
custom_score
```

```
Player_multiplier = 0
```

```
opponent_multiplier = 2
```

```
custom_score_2
```

```
Player_multiplier = 1
```

```
opponent_multiplier = 1
```

```
custom_score_3
```

```
Player_multiplier = 3
```

```
opponent_multiplier = 0
```

Overall, I recommend custom_score because of its higher win rate performance. It seems like an emphasis on maximizing the impact of opponent moves makes the difference.