Computer and Human Preference Divergences at Chess

Southern Economic Association Annual Conference 2014, Atlanta, GA

Kenneth W. Regan[1]
University at Buffalo (SUNY)

Nov. 24, 2014

[1]Joint work with Tamal Biswas and Jason Zhou. Websites:
http://www.cse.buffalo.edu/~regan/
http://www.cse.buffalo.edu/~regan/chess/fidelity/

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.
2. Hence **cheating** with computers during human games has emerged. . .

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.
2. Hence **cheating** with computers during human games has emerged. . . and **cheating detection** has become a need.

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.
2. Hence **cheating** with computers during human games has emerged... and **cheating detection** has become a need.
3. Desirable to measure more than raw performance.

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.
2. Hence **cheating** with computers during human games has emerged...and **cheating detection** has become a need.
3. Desirable to measure more than raw performance.
4. Other axis is **Cognitive Style**—how do computers' differ from humans'?

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.
2. Hence **cheating** with computers during human games has emerged. . . and **cheating detection** has become a need.
3. Desirable to measure more than raw performance.
4. Other axis is **Cognitive Style**—how do computers' differ from humans'?
5. Can we discriminate it? How much does it add to significance of cheating tests?

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.
2. Hence **cheating** with computers during human games has emerged. . . and **cheating detection** has become a need.
3. Desirable to measure more than raw performance.
4. Other axis is **Cognitive Style**—how do computers' differ from humans'?
5. Can we discriminate it? How much does it add to significance of cheating tests?
6. Implications for **Computerized Agents**, including PDAs.

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.
2. Hence **cheating** with computers during human games has emerged. . . and **cheating detection** has become a need.
3. Desirable to measure more than raw performance.
4. Other axis is **Cognitive Style**—how do computers' differ from humans'?
5. Can we discriminate it? How much does it add to significance of cheating tests?
6. Implications for **Computerized Agents**, including PDAs.
7. "Isomorphism" to Multiple-Choice Testing and other econo/psycho-metrics relevance.

Outline

1. Computer chess *engines* running on mass-market hardware can slay even the best human players.
2. Hence **cheating** with computers during human games has emerged... and **cheating detection** has become a need.
3. Desirable to measure more than raw performance.
4. Other axis is **Cognitive Style**—how do computers' differ from humans'?
5. Can we discriminate it? How much does it add to significance of cheating tests?
6. Implications for **Computerized Agents**, including PDAs.
7. "Isomorphism" to Multiple-Choice Testing and other econo/psycho-metrics relevance.
8. Discussion and applicability of model to the other papers.

A Predictive Analytic Model

1. Domain: A set of decision-making situations t.
   Chess game turns

A Predictive Analytic Model

1. Domain: A set of decision-making situations t.
   Chess game turns
2. Inputs: Values v_i for every option at turn t.
   Computer values of moves m_i

A Predictive Analytic Model

1. Domain: A set of decision-making situations t.
   Chess game turns
2. Inputs: Values v_i for every option at turn t.
   Computer values of moves m_i
3. Parameters: $s, c, \dots$ denoting skills and levels.
   Trained correspondence to chess Elo rating E

A Predictive Analytic Model

1. Domain: A set of decision-making situations t.
   Chess game turns
2. Inputs: Values v_i for every option at turn t.
   Computer values of moves m_i
3. Parameters: $s, c, \ldots$ denoting skills and levels.
   Trained correspondence to chess Elo rating E
4. Defines *fallible agent* $P(s, c, \ldots)$.

A Predictive Analytic Model

1. Domain: A set of decision-making situations t.
   Chess game turns
2. Inputs: Values v_i for every option at turn t.
   Computer values of moves m_i
3. Parameters: $s, c, \dots$ denoting skills and levels.
   Trained correspondence to chess Elo rating E
4. Defines *fallible agent* $P(s, c, \dots)$.
5. Main Output: Probabilities $p_{t,i}$ for $P(s, c, \dots)$ to select option i at time t.

A Predictive Analytic Model

1. Domain: A set of decision-making situations t.
   Chess game turns
2. Inputs: Values v_i for every option at turn t.
   Computer values of moves m_i
3. Parameters: $s, c, \dots$ denoting skills and levels.
   Trained correspondence to chess Elo rating E
4. Defines *fallible agent* $P(s, c, \dots)$.
5. Main Output: Probabilities $p_{t,i}$ for $P(s, c, \dots)$ to select option i at time t.
6. Derived Outputs:
   - Aggregate statistics: *move-match* MM, *average error* AE, ...
   - Projected confidence intervals for those statistics.
   - "Intrinsic Performance Ratings" (IPR's).

Elo Rating System

- Points are (ideally) zero-sum: what P gains O loses.

Elo Rating System

- Points are (ideally) zero-sum: what P gains O loses.
- Only rating differences matter—absolute numbers have no intrinsic meaning.

Elo Rating System

- Points are (ideally) zero-sum: what P gains O loses.
- Only rating differences matter—absolute numbers have no intrinsic meaning. Yet my work argues no significant "inflation."

2800	World champ: Carlsen peak 2881, now 2863
2700	"Super-GM"
2600	"Strong GM"
2500	Grandmaster (GM)
2400	International Master (IM) (KWR, D. Levy, H. Berliner)
2200	National Master, 30,000 worldwide
2000	Expert
⋮	⋮
1000	Class E, "bright beginner"
⋮	⋮
600?	True beginner with "sight of the board"?

The Data

- Over 2 million moves of 50-PV data: approaching 200GB

The Data

- Over 2 million moves of 50-PV data: approaching 200GB
- Over 30 million moves of Single-PV data: about 35 GB

The Data

- Over 2 million moves of 50-PV data: approaching 200GB
- Over 30 million moves of Single-PV data: about 35 GB
- = over 100 million pages of text data at 2k/page.
- All taken on two quad-core home-style PC's. Is this "Big Data"?

The Data

- Over 2 million moves of 50-PV data: approaching 200GB
- Over 30 million moves of Single-PV data: about 35 GB
- = over 100 million pages of text data at 2k/page.
- All taken on two quad-core home-style PC's. Is this "Big Data"?

Two Modes of Operation

1. To test games by player P for cheating:

Two Modes of Operation

1. To test games by player P for cheating:
   - Regress **on large data** to set params $s, c, \ldots$ for Elo rating of P.

Two Modes of Operation

1. To test games by player P for cheating:
   - Regress **on large data** to set params $s, c, \ldots$ for Elo rating of P.
   - Use $s, c, \ldots$ to generate projections and confidence intervals for tests ("MM" and "AE" tests) from analysis **of player's games**.

Two Modes of Operation

1. To test games by player P for cheating:
    - Regress **on large data** to set params $s, c, \ldots$ for Elo rating of P.
    - Use $s, c, \ldots$ to generate projections and confidence intervals for tests ("MM" and "AE" tests) from analysis **of player's games**.
    - So far independent of moves played.

Two Modes of Operation

1. To test games by player P for cheating:
   - Regress **on large data** to set params $s, c, \ldots$ for Elo rating of P.
   - Use $s, c, \ldots$ to generate projections and confidence intervals for tests ("MM" and "AE" tests) from analysis **of player's games**.
   - So far independent of moves played.
   - Compare acutal results from moves played.

Two Modes of Operation

1. To test games by player P for cheating:
   - Regress **on large data** to set params $s, c, \ldots$ for Elo rating of P.
   - Use $s, c, \ldots$ to generate projections and confidence intervals for tests ("MM" and "AE" tests) from analysis **of player's games**.
   - So far independent of moves played.
   - Compare acutal results from moves played.

2. To compute "Intrinsic Performance Rating" (IPR) for P:

Two Modes of Operation

1. To test games by player P for cheating:
   - Regress **on large data** to set params $s, c, \ldots$ for Elo rating of P.
   - Use $s, c, \ldots$ to generate projections and confidence intervals for tests ("MM" and "AE" tests) from analysis **of player's games**.
   - So far independent of moves played.
   - Compare acutal results from moves played.

2. To compute "Intrinsic Performance Rating" (IPR) for P:
   - Regress **on** P**'s games**—i.e. **on small data**—to get $s_P, c_P, \ldots$

Two Modes of Operation

1. To test games by player P for cheating:
   - Regress **on large data** to set params $s, c, \dots$ for Elo rating of P.
   - Use $s, c, \dots$ to generate projections and confidence intervals for tests ("MM" and "AE" tests) from analysis **of player's games**.
   - So far independent of moves played.
   - Compare acutal results from moves played.

2. To compute "Intrinsic Performance Rating" (IPR) for P:
   - Regress **on** P**'s games**—i.e. **on small data**—to get $s_P, c_P, \dots$
   - Apply $s, c, \dots$ to "Virtual Standardized Test" (same 8,316 positions for everyone, results agree with whole training set to 4 places).

Two Modes of Operation

1. To test games by player P for cheating:
   - Regress **on large data** to set params $s, c, \ldots$ for Elo rating of P.
   - Use $s, c, \ldots$ to generate projections and confidence intervals for tests ("MM" and "AE" tests) from analysis **of player's games**.
   - So far independent of moves played.
   - Compare acutal results from moves played.

2. To compute "Intrinsic Performance Rating" (IPR) for P:
   - Regress **on** P**'s games**—i.e. **on small data**—to get $s_P, c_P, \ldots$
   - Apply $s, c, \ldots$ to "Virtual Standardized Test" (same 8,316 positions for everyone, results agree with whole training set to 4 places).
   - Score mapped to Elo scale, to get IPR $\pm$ *error*.

Two Modes of Operation

1. To test games by player P for cheating:
   - Regress **on large data** to set params $s, c, \ldots$ for Elo rating of P.
   - Use $s, c, \ldots$ to generate projections and confidence intervals for tests ("MM" and "AE" tests) from analysis **of player's games**.
   - So far independent of moves played.
   - Compare acutal results from moves played.

2. To compute "Intrinsic Performance Rating" (IPR) for P:
   - Regress **on** P**'s games**—i.e. **on small data**—to get $s_P, c_P, \ldots$
   - Apply $s, c, \ldots$ to "Virtual Standardized Test" (same 8,316 positions for everyone, results agree with whole training set to 4 places).
   - Score mapped to Elo scale, to get IPR $\pm$ *error*.
   - Error of measurement, not confidence test.

Examples

2014 World Championship Match

- Anand, 2785 ± 145
- Carlsen, 2920 ± 135
- Combined, 2850 ± 100,

Screening test:

72	67.8%	0.055	202	Tal, Mihail	Marseille1989cat11
73	67.8%	0.078	121	Morphy, Paul	MorphyParisOffhand1858
74	67.7%	0.032	297	Shirov, A.	SofiaMTel2009cat21

Full test: Morphy at 2350 projected to match almost 60%, and full test actual is less ("regression to mean"), so not significant.

Separating Skill Assessment From Prediction

Separating Skill Assessment From Prediction

- Engines work by **iteratively deepened search**.

Separating Skill Assessment From Prediction

- Engines work by **iterated deepened search**.
- Some moves' values "*swing*" markedly down (a "trap") or up (a "hidden resource").

Separating Skill Assessment From Prediction

- Engines work by **iteratively deepened search**.
- Some moves' values "*swing*" markedly down (a "trap") or up (a "hidden resource").
- New "Depth" paper shows strong effect of *swing* on human probabilities.

Separating **Skill Assessment** From **Prediction**

- Engines work by **iteratively deepened search**.
- Some moves' values "*swing*" markedly down (a "trap") or up (a "hidden resource").
- New "Depth" paper shows strong effect of *swing* on human probabilities.
- Computers largely immune to effect, especially in fixed-depth play.

Separating Skill Assessment From Prediction

- Engines work by **iteratively deepened search**.
- Some moves' values "*swing*" markedly down (a "trap") or up (a "hidden resource").
- New "Depth" paper shows strong effect of *swing* on human probabilities.
- Computers largely immune to effect, especially in fixed-depth play.
- Explains 'strange' 58%–42% law for human preference of first-listed of two moves given equal value at highest depth, conditioned on one of them having been played.

Separating Skill Assessment From Prediction

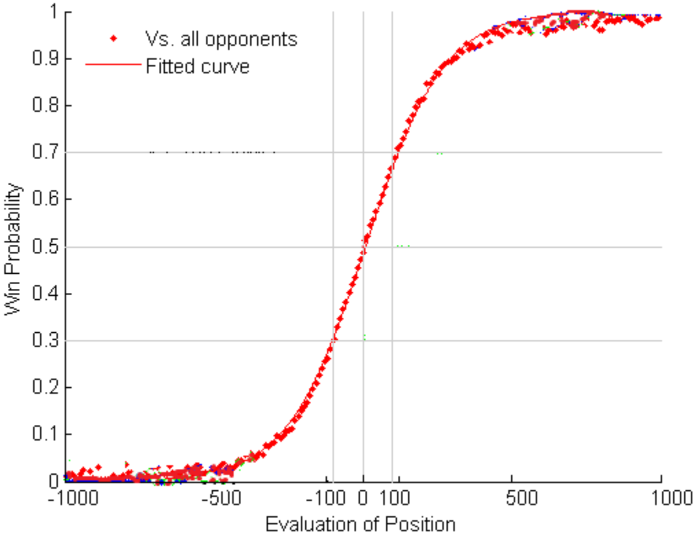
- Engines work by **iteratively deepened search**.
- Some moves' values "*swing*" markedly down (a "trap") or up (a "hidden resource").
- New "Depth" paper shows strong effect of *swing* on human probabilities.
- Computers largely immune to effect, especially in fixed-depth play.
- Explains 'strange' 58%–42% law for human preference of first-listed of two moves given equal value at highest depth, conditioned on one of them having been played.
- First-listed move higher-valued at lower depths; moves sort is *stable*.

Separating Skill Assessment From Prediction

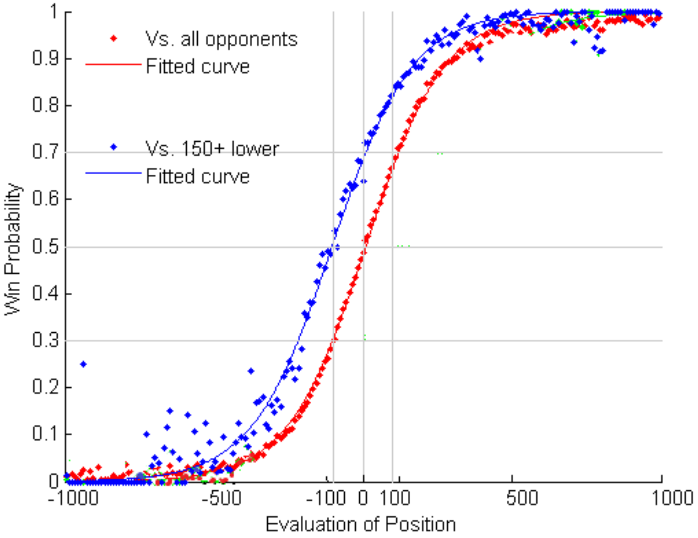
- Engines work by **iteratively deepened search**.
- Some moves' values "*swing*" markedly down (a "trap") or up (a "hidden resource").
- New "Depth" paper shows strong effect of *swing* on human probabilities.
- Computers largely immune to effect, especially in fixed-depth play.
- Explains 'strange' 58%–42% law for human preference of first-listed of two moves given equal value at highest depth, conditioned on one of them having been played.
- First-listed move higher-valued at lower depths; moves sort is *stable*.

Use values at all depths to **predict***; use highest-depth values to* **assess**.

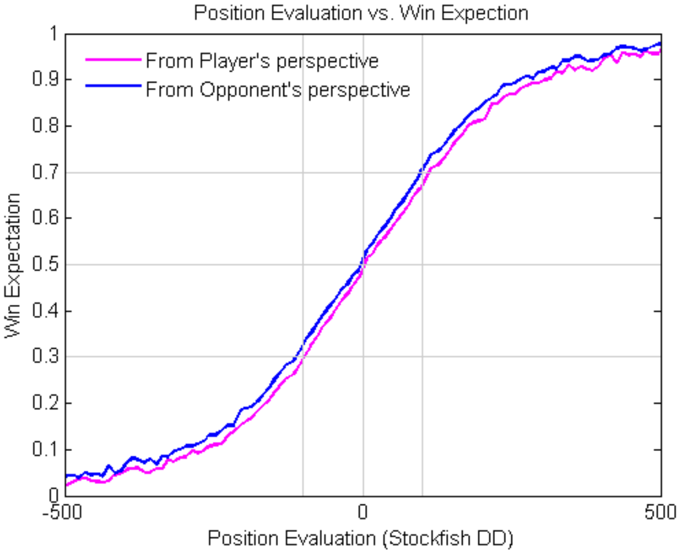
Win % Expectation Curve



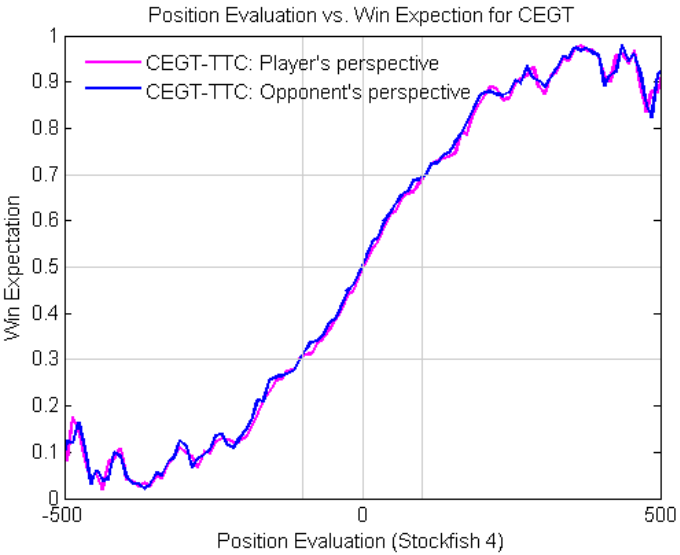
And When You're Higher Rated



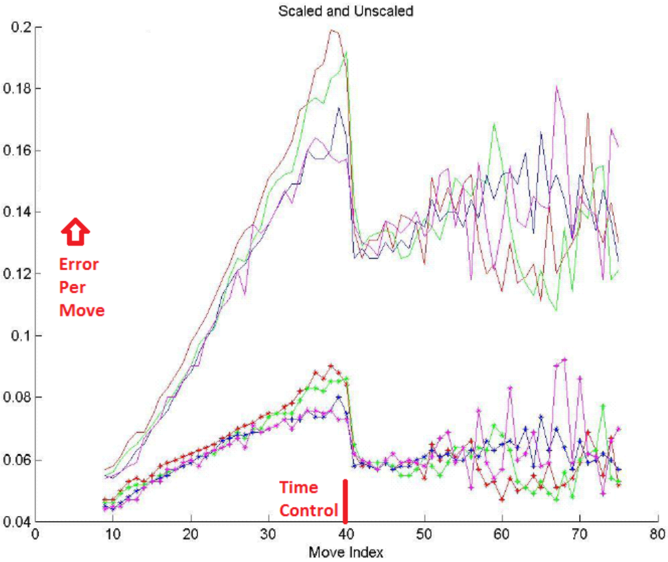
Would You Like it to be Your Move?



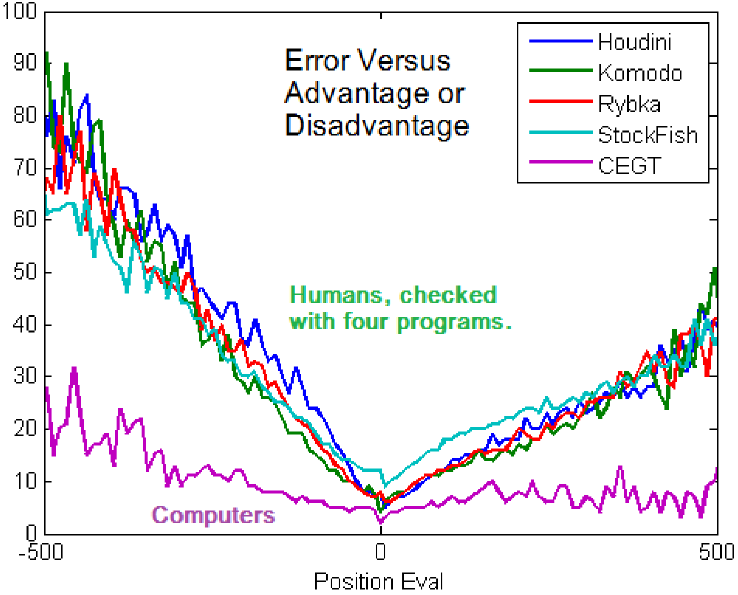
Effect Absent in Computer Play



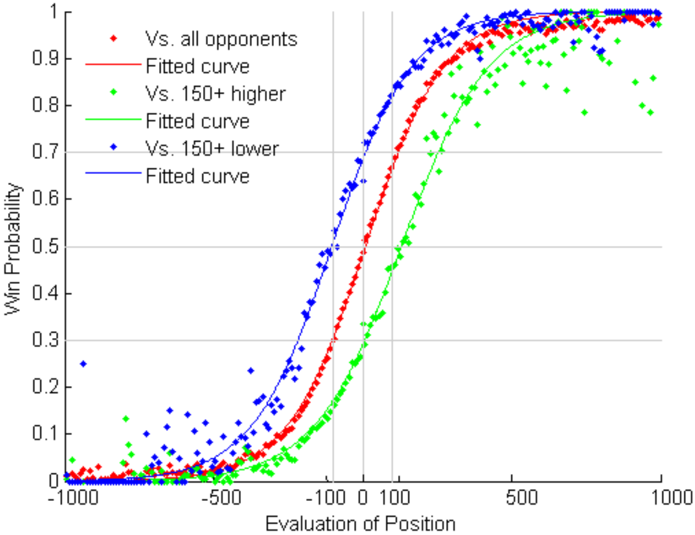
Managing a Time Budget



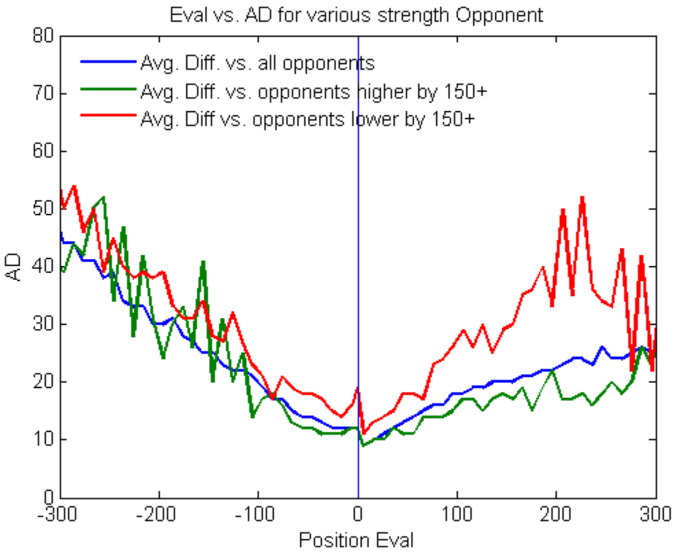
Minding Nickels and Dimes



Are We Psychological or Rational?

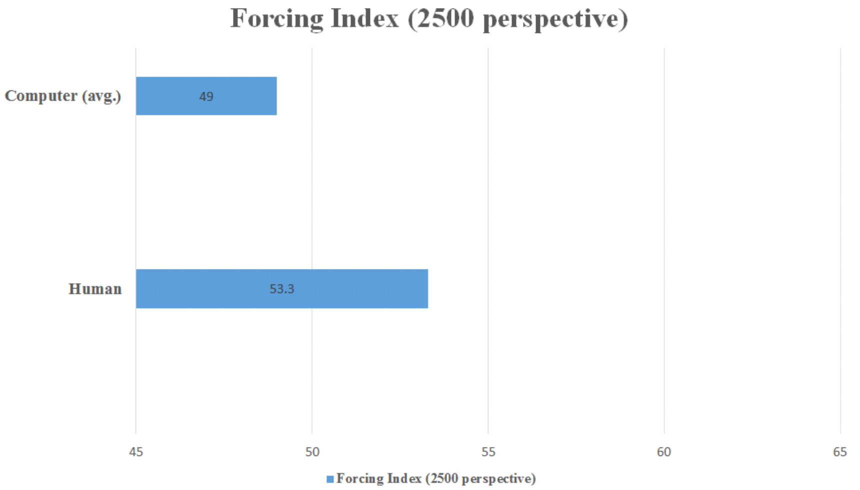


Some Evidence for Psychological

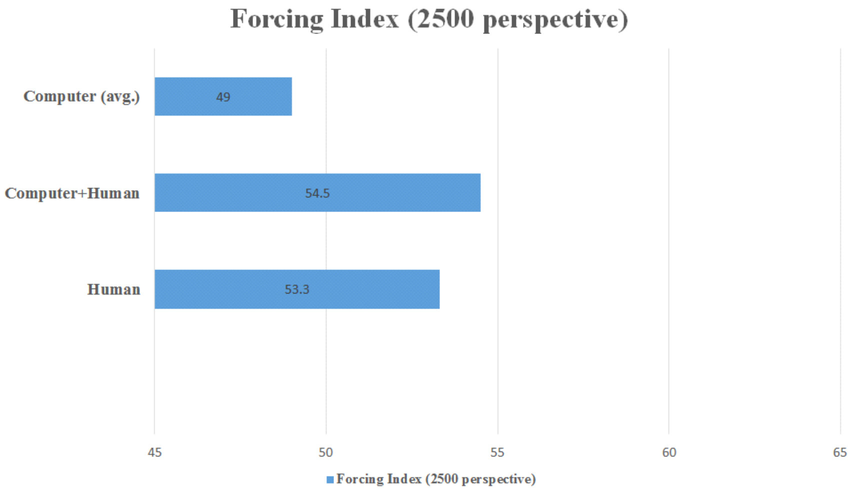


Minima stay at 0.

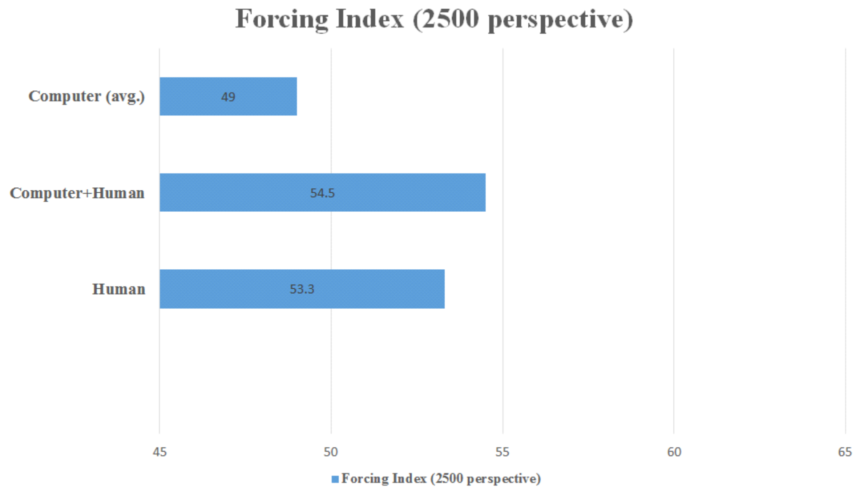
Degrees of Forcing Play



Add Human-Computer Tandems

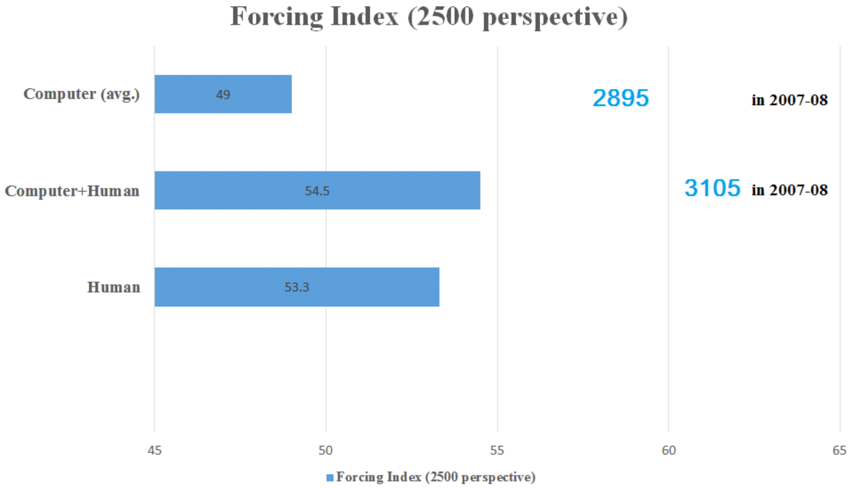


Add Human-Computer Tandems



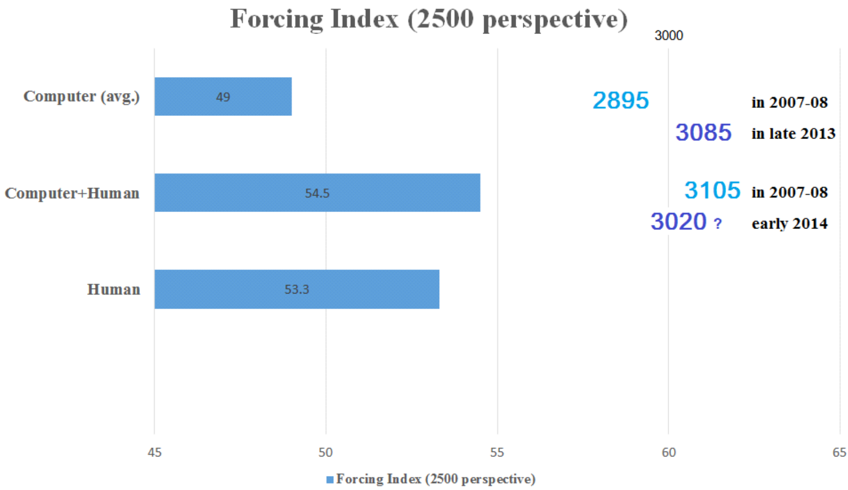
Evidently the humans called the shots. How was the quality?

2007–08 Freestyle Performance



Adding 210 Elo was significant. Forcing but good teamwork.

2014 Freestyle Tournament Performance



2014: tandems marginally better W-L, but quality not clear...

Summary For Us and PDAs

Summary For Us and PDAs

1. PDAs pick up every little difference: “**Forest** *and* **Trees**”

Summary For Us and PDAs

1. PDAs pick up every little difference: "**Forest** *and* **Trees**"
2. We should avoid overconfidence...

Summary For Us and PDAs

1. PDAs pick up every little difference: "**Forest** *and* **Trees**"
2. We should avoid overconfidence. . . and take counsel when "down."

Summary For Us and PDAs

1. PDAs pick up every little difference: "**Forest** *and* **Trees**"
2. We should avoid overconfidence... and take counsel when "down."
3. **Look before we Leap...**

Summary For Us and PDAs

1. PDAs pick up every little difference: "**Forest** *and* **Trees**"
2. We should avoid overconfidence... and take counsel when "down."
3. **Look before we Leap... Don't rush in...**

Summary For Us and PDAs

1. PDAs pick up every little difference: "**Forest** *and* **Trees**"
2. We should avoid overconfidence... and take counsel when "down."
3. **Look before we Leap... Don't rush in... Measure risks.**

Summary For Us and PDAs

1. PDAs pick up every little difference: "**Forest** *and* **Trees**"
2. We should avoid overconfidence... and take counsel when "down."
3. **Look before we Leap... Don't rush in... Measure risks.**
4. Even at a purely calculational pursuit like chess, *our brains still contribute.*

Summary For Us and PDAs

1. PDAs pick up every little difference: “**Forest** *and* **Trees**”
2. We should avoid overconfidence. . . and take counsel when “down.”
3. **Look before we Leap. . . Don’t rush in. . . Measure risks.**
4. Even at a purely calculational pursuit like chess, *our brains still contribute*. (2014: maybe)

Summary For Us and PDAs

1. PDAs pick up every little difference: "**Forest** *and* **Trees**"
2. We should avoid overconfidence... and take counsel when "down."
3. **Look before we Leap... Don't rush in... Measure risks.**
4. Even at a purely calculational pursuit like chess, *our brains still contribute*. (2014: maybe)
5. **Main takeaway:**

Summary For Us and PDAs

1. PDAs pick up every little difference: “**Forest** *and* **Trees**”
2. We should avoid overconfidence... and take counsel when “down.”
3. **Look before we Leap... Don’t rush in... Measure risks.**
4. Even at a purely calculational pursuit like chess, *our brains still contribute*. (2014: maybe)
5. **Main takeaway:**

It should be natural to program PDAs so they enhance our freedom rather than constrain it.

Summary For Us and PDAs

1. PDAs pick up every little difference: "**Forest** *and* **Trees**"
2. We should avoid overconfidence... and take counsel when "down."
3. **Look before we Leap... Don't rush in... Measure risks.**
4. Even at a purely calculational pursuit like chess, *our brains still contribute*. (2014: maybe)
5. **Main takeaway:**

It should be natural to program PDAs so they enhance our freedom rather than constrain it.

This could be the beginning of a beautiful relationship...