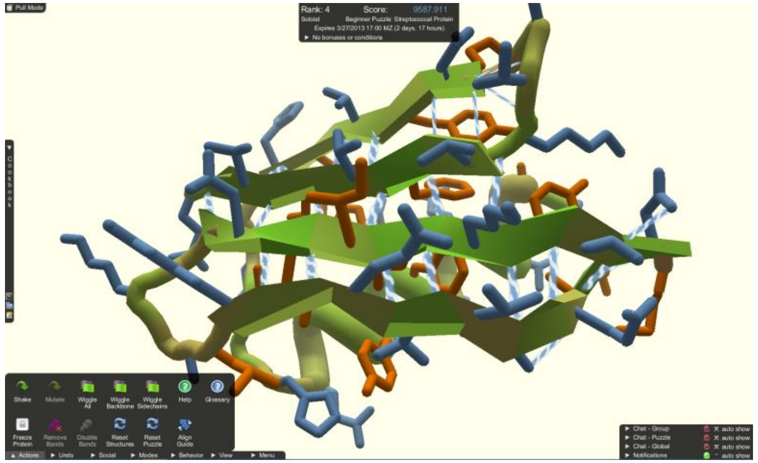
An Online System for Player-vs-Level Matchmaking in Human Computation Games

Anurag Sarkar and **Seth Cooper**

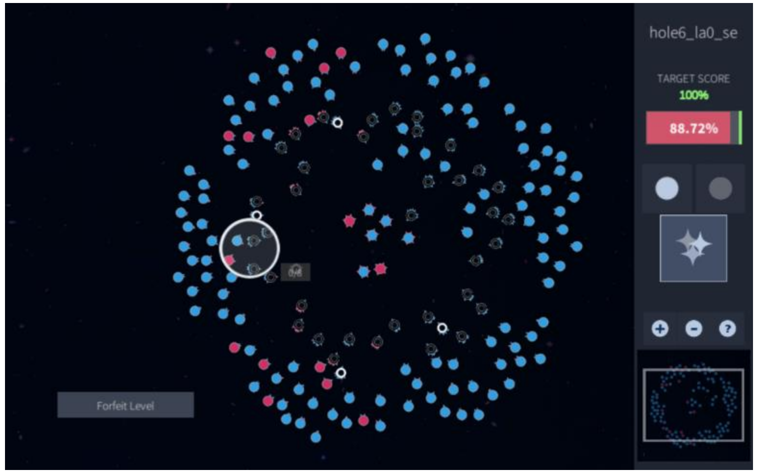
Northeastern University

Background

- Human computation games (HCGs) model real world problems to help solve them through players



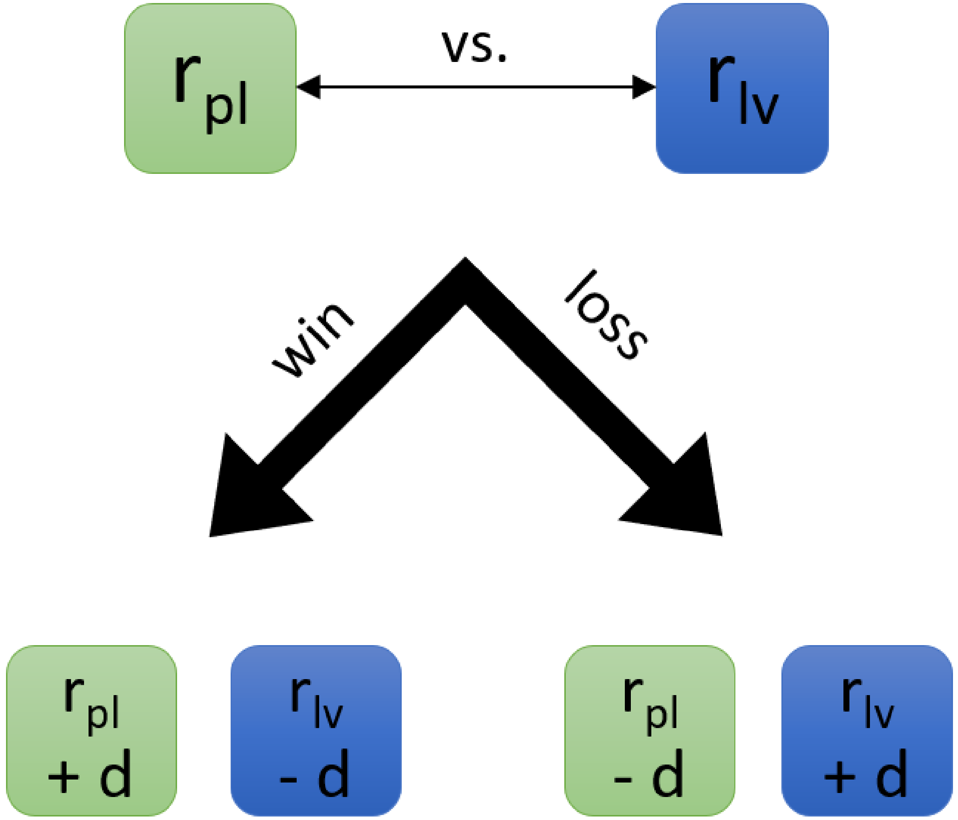
Foldit



Paradox

Background

- Human computation games (HCGs) model real world problems to help solve them through players
- Prior works did dynamic difficulty adjustment (DDA) in HCGs using rating systems and skill chains, framing DDA as PvL matchmaking
  - Player rating → player ability
  - Level rating → level difficulty



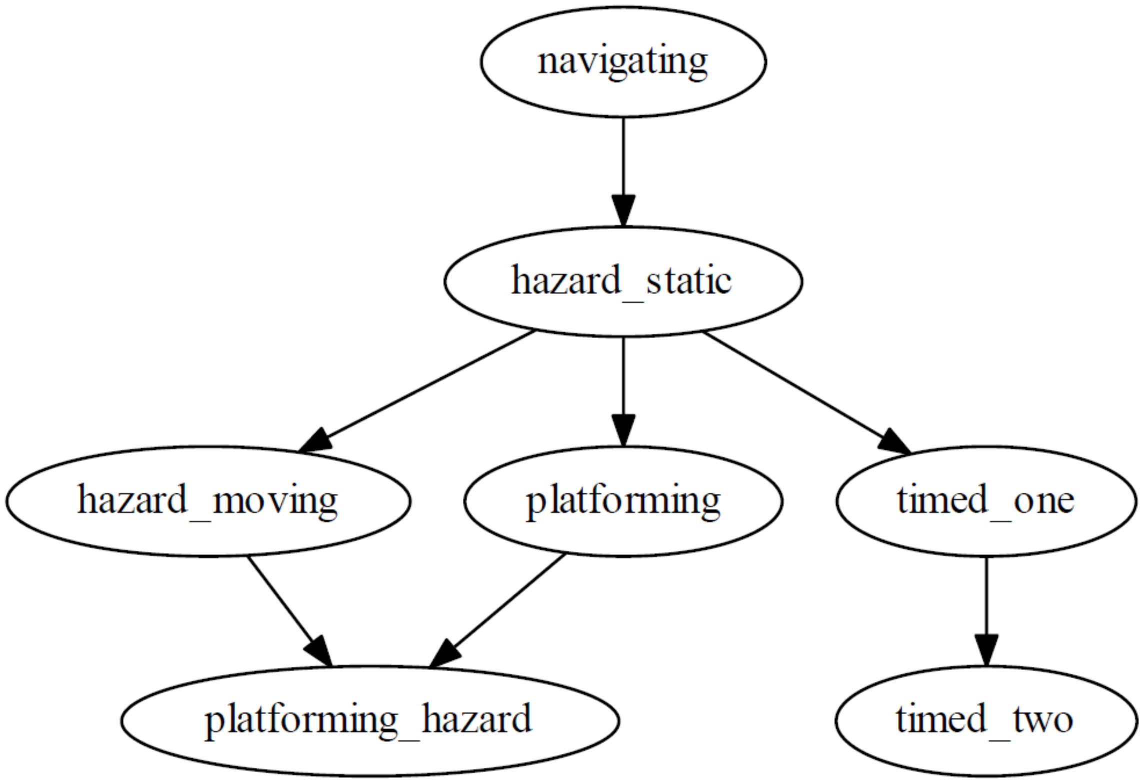
Background

- Human computation games (HCGs) model real world problems to help solve them through players

- Prior works did dynamic difficulty adjustment (DDA) in HCGs using rating systems and skill chains, framing DDA as PvL matchmaking
  - Player rating → player ability
  - Level rating → level difficulty

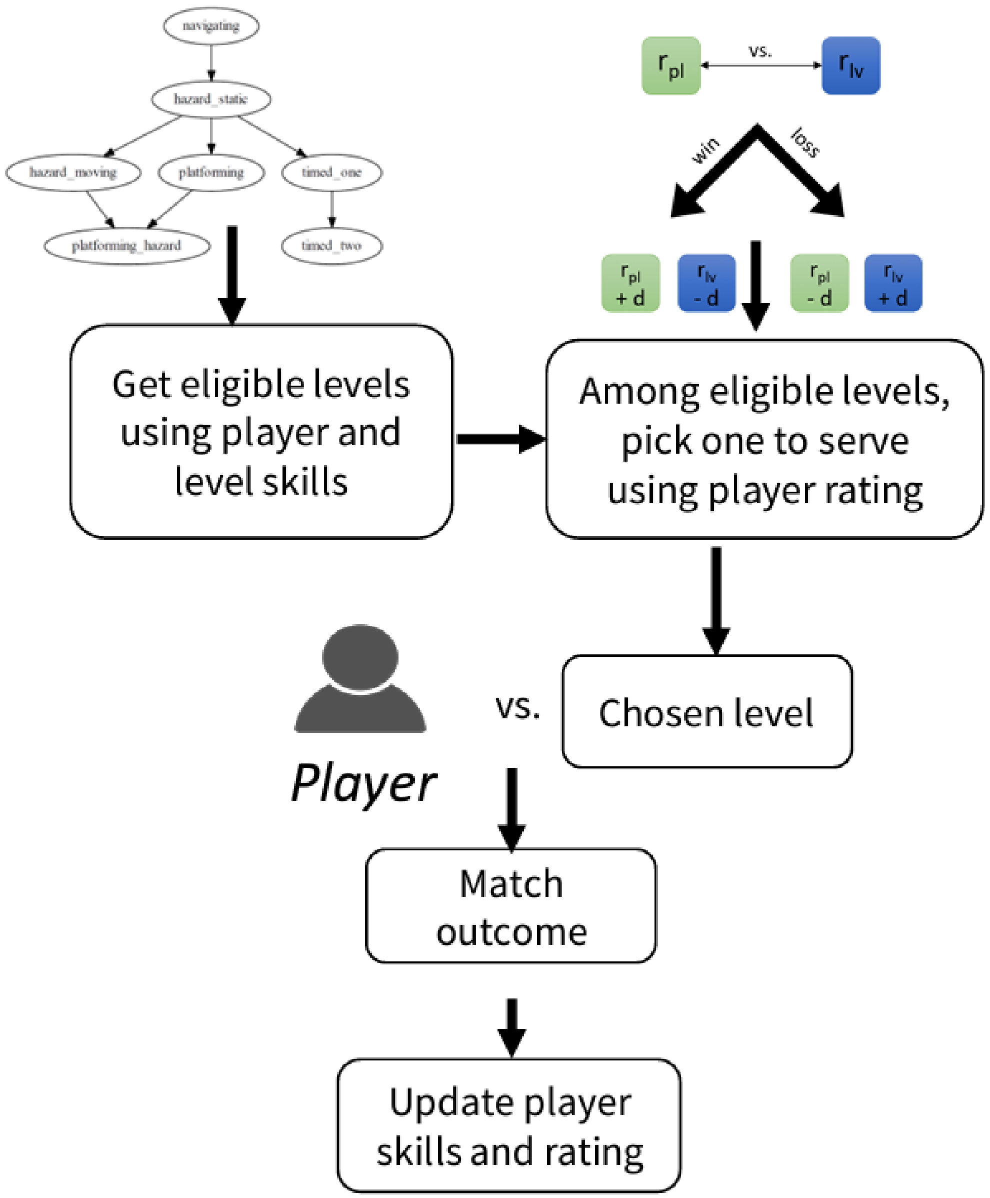
- Skill chains
  - Define the order of player skill acquisition during gameplay

  - Can be used to define level progressions of varying difficulty



DDA Model

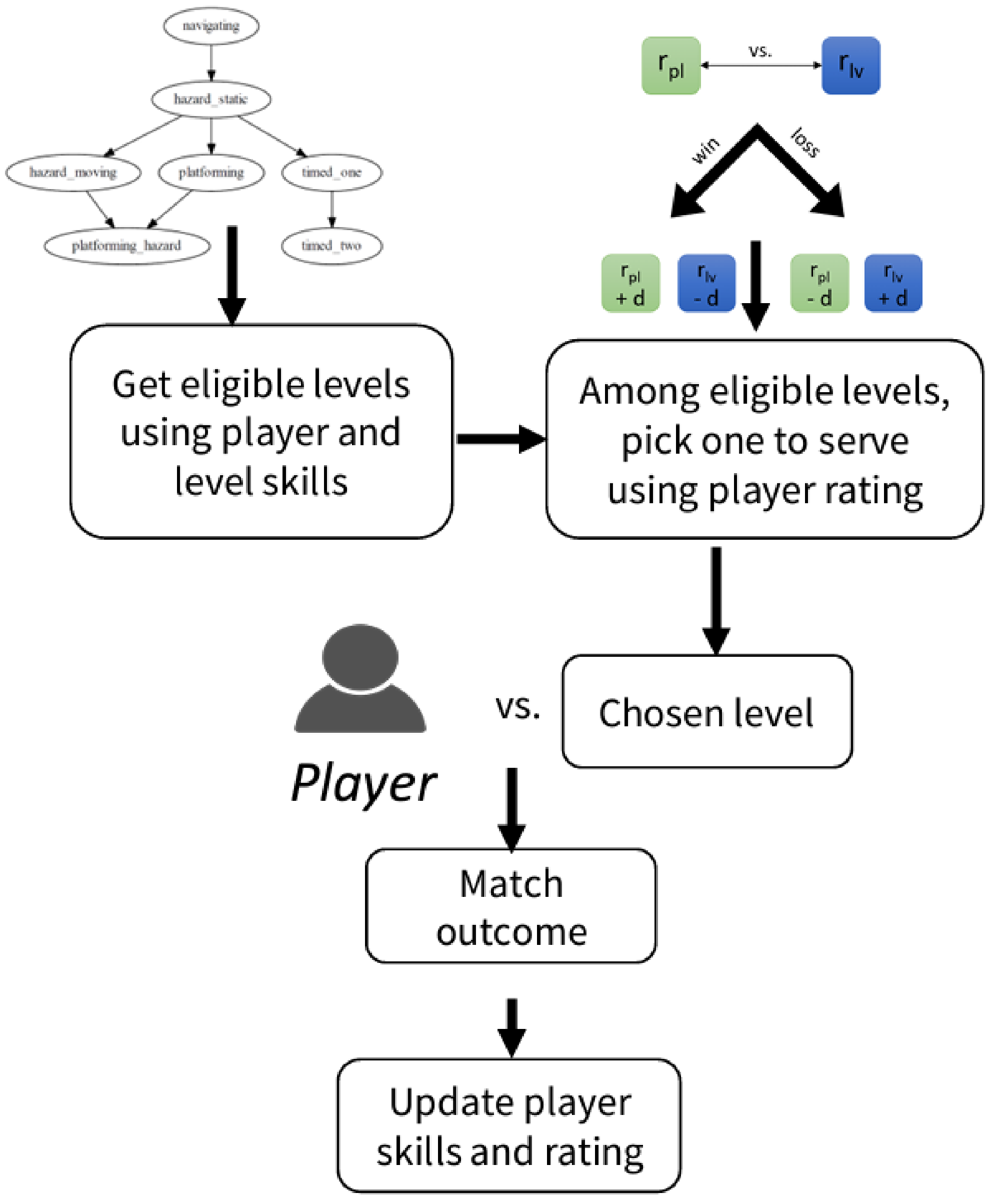
- Two step DDA process:
  - Skill Chain - determine set of eligible levels based on player's acquired skills and skills required by levels
  - Rating System – from among the eligible levels serve the best match



DDA Model

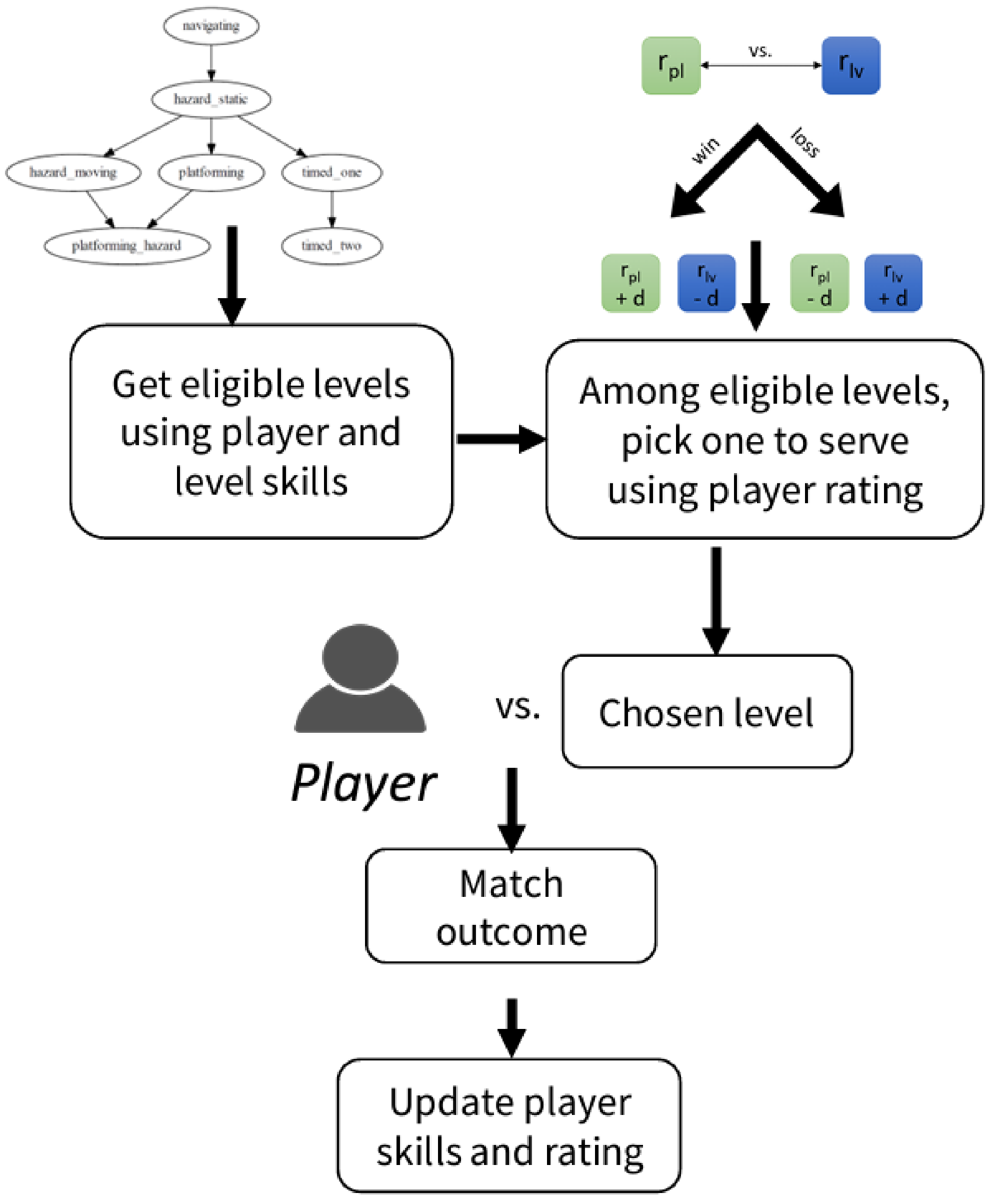
- Problems
  - To avoid cold-start, requires collecting level ratings via random playthroughs in prior *off-line* phase and keep them fixed during *on-line* matchmaking phase

    --- updating online creates skewed distribution of harder levels mainly being attempted by advanced players



DDA Model

- Problems
  - To avoid cold-start, requires collecting level ratings via random playthroughs in prior *off-line* phase and keep them fixed during *on-line* matchmaking phase
    --- updating online creates skewed distribution of harder levels mainly being attempted by advanced players
  - Each level requires a target score threshold for determining win/loss fixed across all players



Approach

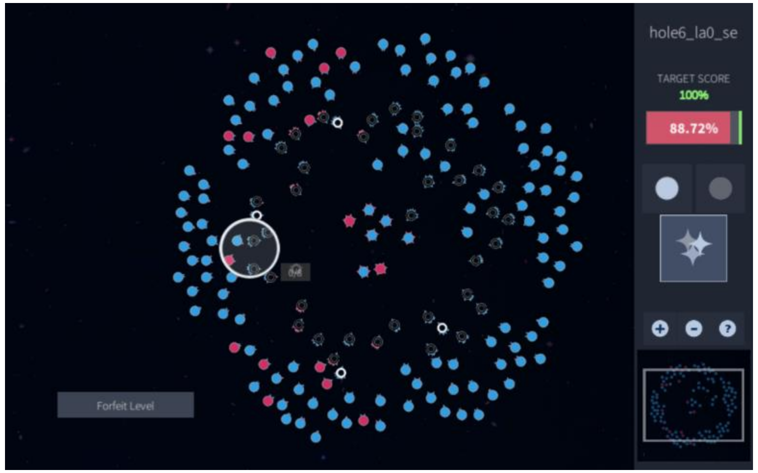
- Two extensions to DDA model:
  - ε-greedy matchmaking to address cold-start issue with level ratings
  - Incorporate rating arrays from prior work for dynamic win/loss thresholds

Approach

- Two extensions to DDA model:
  - ε-greedy matchmaking to address cold-start issue with level ratings
  - Incorporate rating arrays from prior work for dynamic win/loss thresholds

- Games

Iowa James



Paradox

Algorithm 1 ϵ-greedy matchmaking

Input: all_levels
Output: $level$
$level = \varnothing$
$candidates$ = { levels from all_levels player hasn't completed or just played in the previous match }
if $random < \epsilon_1$ **then**
  $level \leftarrow$ random choice from $candidates$
else
  $candidates \leftarrow$ remove ineligible levels based on player's current skills from $candidates$
  if $random < \epsilon_2$ **then**
    $level \leftarrow$ random choice from $candidates$, weighted inversely by number of playthroughs
  else
    $level \leftarrow$ best match from $candidates$, as determined by rating system comparing player and level ratings
  end if
end if
return $level$

ε-Greedy Experiment

- For each game, recruited players using Amazon Mechanical Turk

- Players randomly assigned to one of 4 settings:
  - $\varepsilon_1 = \varepsilon_2 = 0$ (original DDA model)
  - $\varepsilon_1 = \varepsilon_2 = 0.1$
  - $\varepsilon_1 = \varepsilon_2 = 0.2$
  - $\varepsilon_1 = \varepsilon_2 = 1$ (random)

- Variables
  - *Play Time*
  - *Levels Completed*
  - *Levels Lost*
  - *Level Rating Error*
    - (mean square error between level ratings in matches vs. final level ratings in random condition)

ε-Greedy Experiment

- ε = 0.1 (random) had lowest *Level Rating Error*
- ε = 0.1 and ε = 0.2 had lower *Level Rating Error* than ε = 0 (original model) in both games
- *Levels Completed* not significantly lower for ε= 0.1 and ε= 0.2 compared to original model
- Takeaway: ε-greedy approach produces more accurate level ratings compared to the original model while still performing useful level assignment for matchmaking

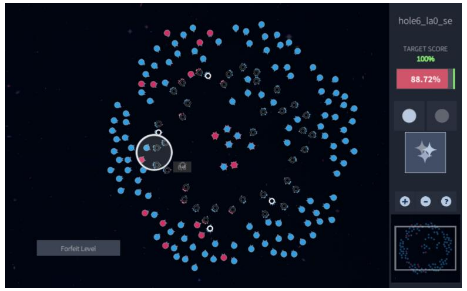
Iowa James

Variable	$\epsilon = 0$ (orig.)	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 1$ (random)
Play Time ($p = .56$)	224.75	246.1	347.58	510.97
Levels Completed ($p = .03$)	2^a	1^{ab}	1^{ab}	0^b
Levels Lost ($p = .15$)	3	3	4.5	4
Level Rating Error ($p < .01$)	207.3^a	162.45^b	176.61^{ab}	93.64^c

Paradox

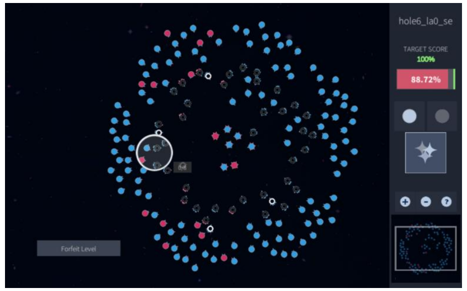
Variable	$\epsilon = 0$ (orig.)	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 1$ (random)
Play Time ($p = .51$)	653.08	620.24	290.62	749.07
Levels Completed ($p < .01$)	4^a	4^a	3.5^a	1^b
Levels Lost ($p = .14$)	2	1	2	2
Level Rating Error ($p < .01$)	181.56^a	126.87^b	88.81^c	73.22^d

Rating Arrays



1700

Single Level Rating



0%	→	305
10%	→	929
20%	→	1140
30%	→	1280
40%	→	1395
50%	→	1500
60%	→	1605
70%	→	1720
80%	→	1860
90%	→	2071

Array of Level Ratings

Rating Array Experiment

- Players recruited using Mechanical Turk, but only used *Paradox* for this experiment
- Two conditions – rating arrays and no rating arrays (single rating per level)
- Used $\varepsilon = 0.1$ since this had best results in prior experiment
- Variables – *Play Time, Levels Completed, Levels Lost*
- Tracked high score for each of the 19 non-tutorial levels

Rating Array Experiment

- Significantly more *Levels Completed* in the array condition
- High score evaluation for 19 non-tutorial levels:
  - Array > Non-Array: 5
  - Non-Array > Array: 8
  - Tie: 6 (in each case, array found high score in fewer matches)
  - Differences in high scores not significant *(p = .5)*
- Takeaway: Rating arrays leads to players completing more levels while producing similar high scores as no-array condition

Variable	Array	No-Array
Play Time ($p = .3$)	529.16	411.39
Levels Completed ($p < .01$)	4^a	3^b
Levels Lost ($p = .77$)	2	2

Conclusion

- We presented an online version of an existing DDA system for HCGs
- Using ε-greedy matchmaking helped address cold-start issues related to level ratings
- Demonstrated first online use of rating arrays which led to players completing more levels

Future Work

- Automatically infer skill chains
- Probabilistic modeling of player skill acquisition
- Test rating arrays with other HCGs as well as educational games

Future Work

- Automatically infer skill chains
- Probabilistic modeling of player skill acquisition
- Test rating arrays with other HCGs as well as educational games

Contact
Anurag Sarkar
Northeastern University
sarkar.an@northeastern.edu