Intelligent Computational Media

Course overview, 2021

Prof. Perttu Hämäläinen

Aalto University

Materials

- Github: https://github.com/PerttuHamalainen/MediaAI/README.md, https://github.com/PerttuHamalainen/MediaAI/Syllabus.md
- Twitter: https://twitter.com/aaltomediaai

Structure

- First days: lectures, some programming exercises (neural network principles & architectures, optimization)
- Rest of the days: most of the time for hands-on exercises and individual/group work, possibly a short lecture each day

Links to lecture slides and exercises:

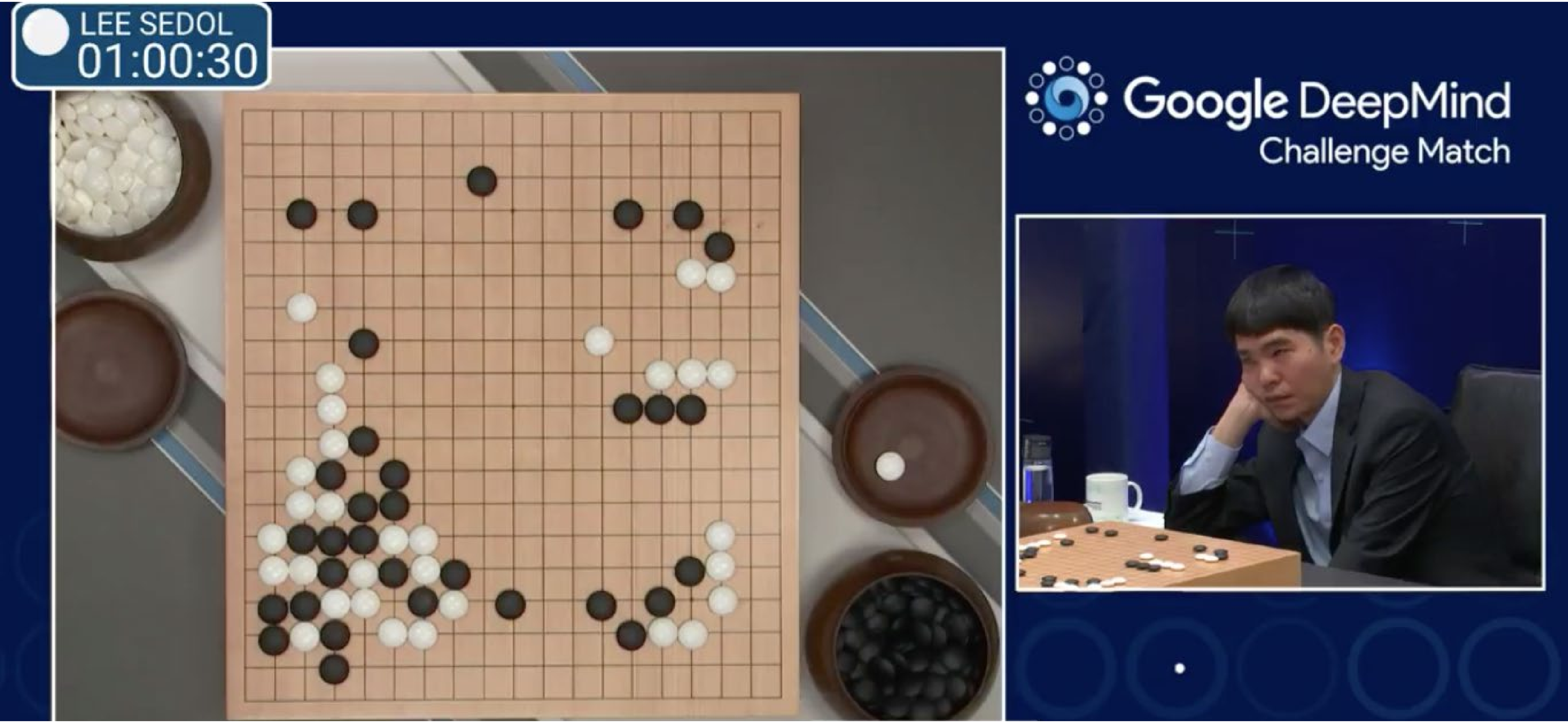
https://github.com/PerttuHamalainen/MediaAI/Syllabus.md

Learning goals

- Understand how common AI algorithms & tools work
- Understand what the tools can be used for
- Get hands-on practice of designing, implementing and/or using the tools

In practice

- Common workflow: Identify the model you need (e.g., StyleGAN), find an open source implementation (install on your computer or find something that runs in Google Colab), collect training data and process it into the format expected by the code, train
- Skills needed: a bit of Python to understand and modify ML code. Scraping data from the Internet and processing it if needed (e.g., image resizing & cropping). Python is also great for the scraping and processing.
- If you don’t want to code: RunwayML (I can’t help you with that, but feel free to try)



Co-creating with AI – competition is futile

Gatys et al. (2015) Neural Algorithm for Artistic Style

A Neural Algorithm of Artistic Style

Leon A. Gatys,[1,2,3*] Alexander S. Ecker,[1,2,4,5] Matthias Bethge[1,2,4]

[1]Werner Reichardt Centre for Integrative Neuroscience
and Institute of Theoretical Physics, University of Tübingen, Germany
[2]Bernstein Center for Computational Neuroscience, Tübingen, Germany
[3]Graduate School for Neural Information Processing, Tübingen, Germany
[4]Max Planck Institute for Biological Cybernetics, Tübingen, Germany
[5]Department of Neuroscience, Baylor College of Medicine, Houston, TX, USA
*To whom correspondence should be addressed; E-mail: leon.gatys@bethgelab.org

In fine art, especially painting, humans have mastered the skill to create unique visual experiences through composing a complex interplay between the content and style of an image. Thus far the algorithmic basis of this process is unknown and there exists no artificial system with similar capabilities. However, in other key areas of visual perception such as object and face recognition near-human performance was recently demonstrated by a class of biologically inspired vision models called Deep Neural Networks.[1,2] Here we introduce an artificial system based on a Deep Neural Network that creates artistic images

The First Trailer for 'Loving Vincent,' an Animated Film Featuring 12 Oil Paintings per Second by Over 100 Painters

by Christopher Jobson on February 26, 2016

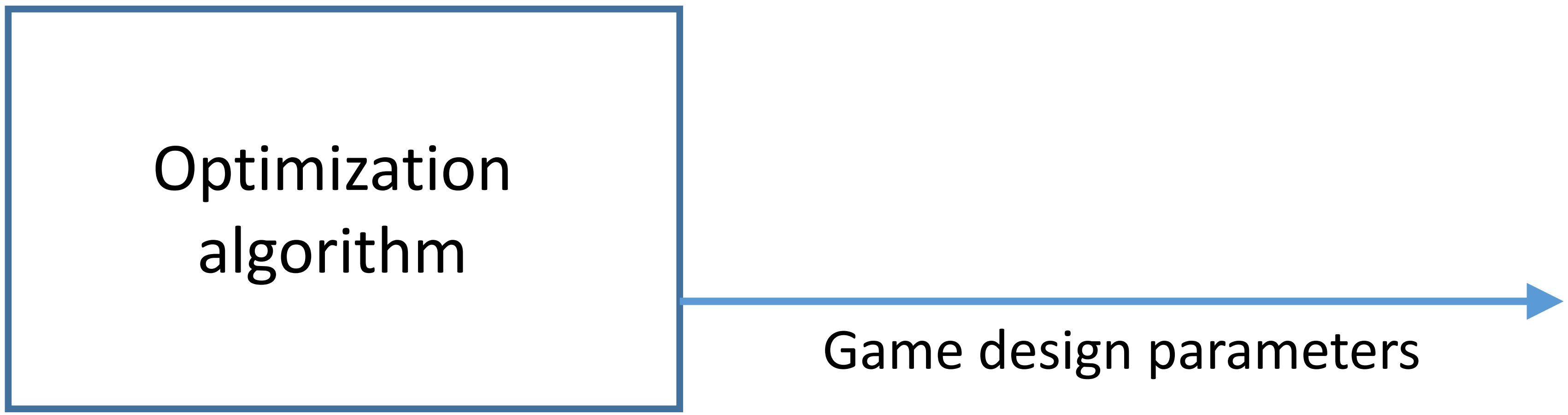
AI and media

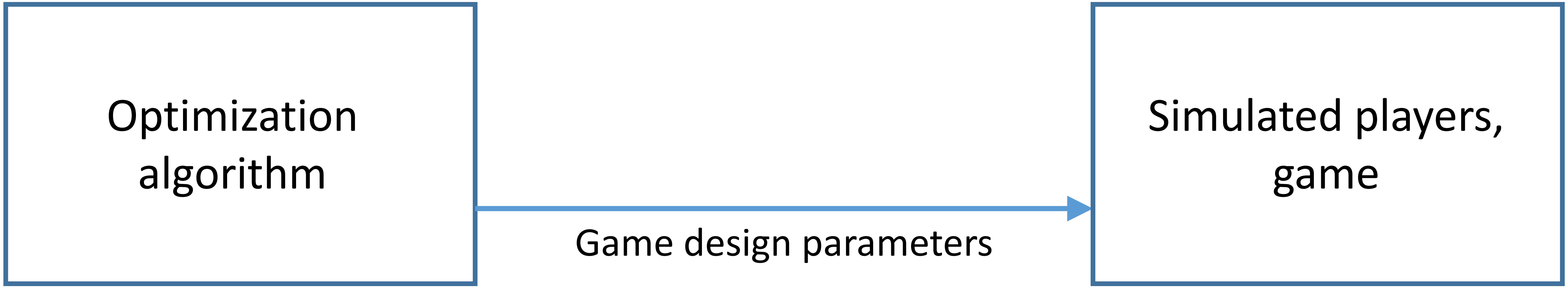
- Media creation (content design, with or without human assistance)
- Media adaptation (e.g., style transfer, adapting a game to support player needs and motivations)
- Media use (testing, collaboration & competing with humans, e.g., using game playing agents)
- User modeling: biomechanical, perceptual, emotion, experience (models of users yield predictions and estimates of human behavior and experience, which enables and informs all of the above)

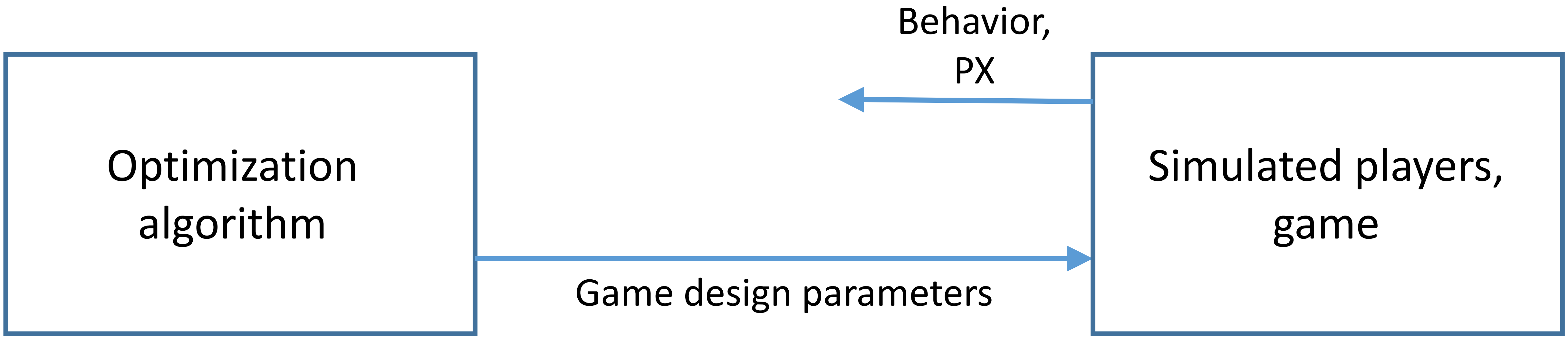
AI players for playtesting?

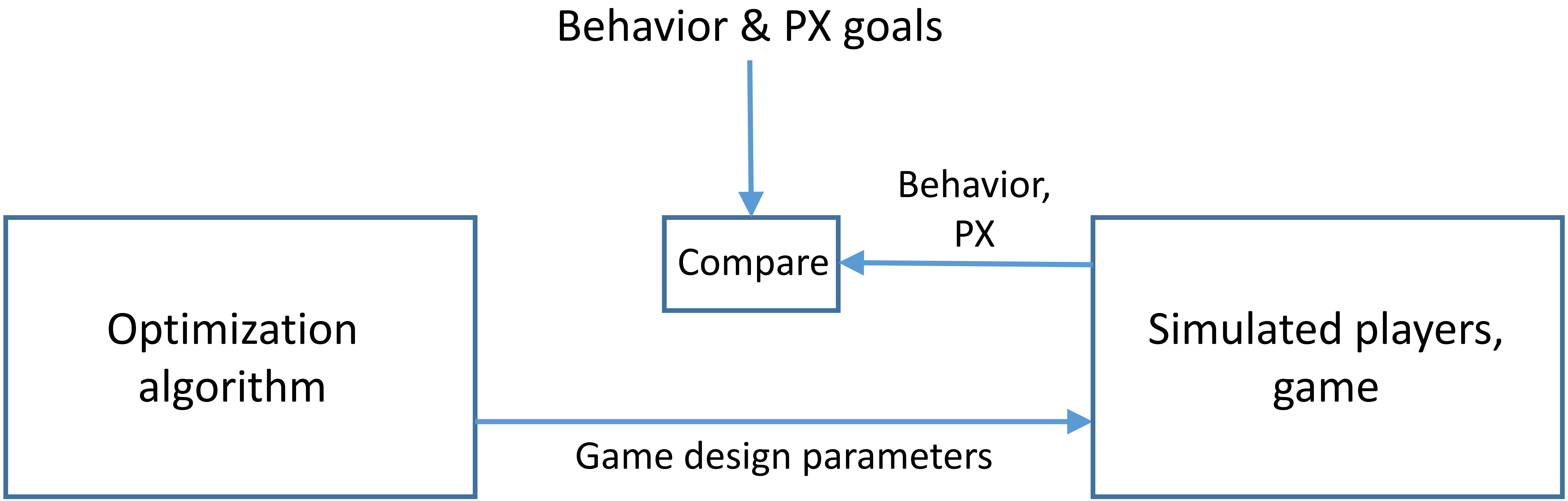
Automated game design

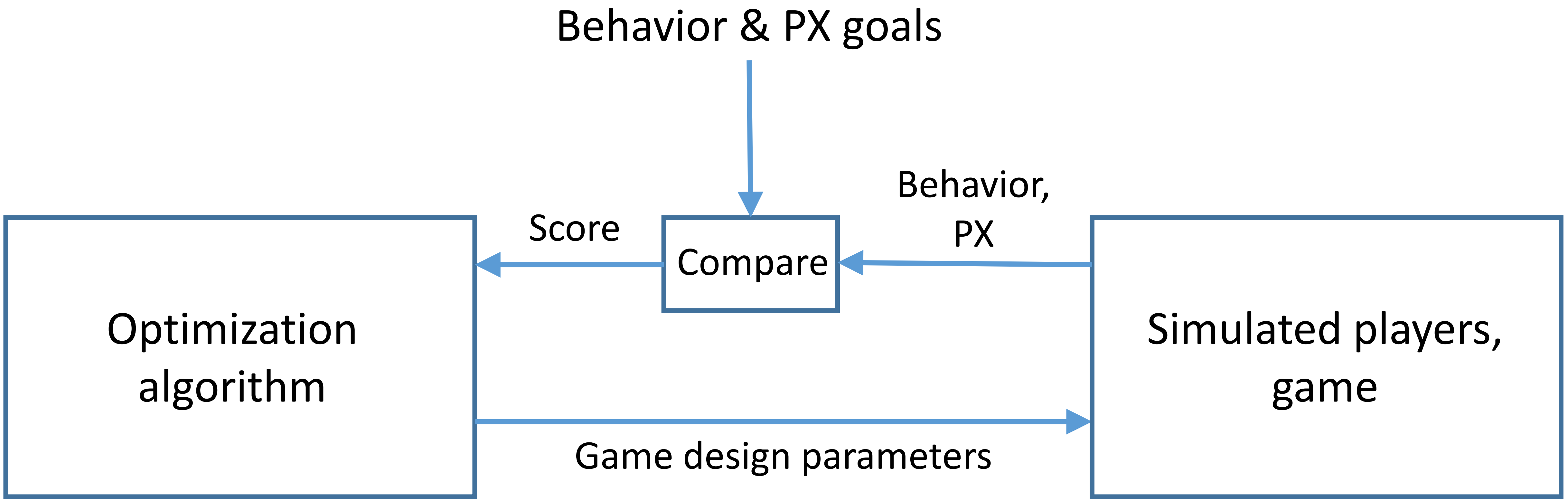
Optimization algorithm


Optimization algorithm
Game design parameters


Optimization algorithm
Simulated players, game
Game design parameters


Behavior,
PX
Optimization
algorithm
Simulated players,
game
Game design parameters


Behavior & PX goals
Compare
Behavior,
PX
Optimization
algorithm
Simulated players,
game
Game design parameters


Behavior & PX goals
Score
Compare
Behavior,
PX
Optimization
algorithm
Simulated players,
game
Game design parameters

An Experiment in Automatic Game Design

Julian Togelius and Jürgen Schmidhuber

Abstract—**This paper presents a first attempt at evolving the rules for a game. In contrast to almost every other paper that applies computational intelligence techniques to games, we are not generating behaviours, strategies or environments for any particular game; we are starting without a game and generating the game itself. We explain the rationale for doing this and survey the theories of entertainment and curiosity that underly our fitness function, and present the details of a simple proof-of-concept experiment.**

Keywords: game design, evolutionary design, entertainment metrics

I. Introduction

Can computational intelligence (CI) help designing games? One is tempted to answer "Yes, obviously, the whole field of Computational Intelligence in Games (CIG) is devoted to this, isn't it?"

However, the majority of CIG research is concerned with learning to play particular games as well as possible. There interest from game developers in learning to play the game better *per se*.

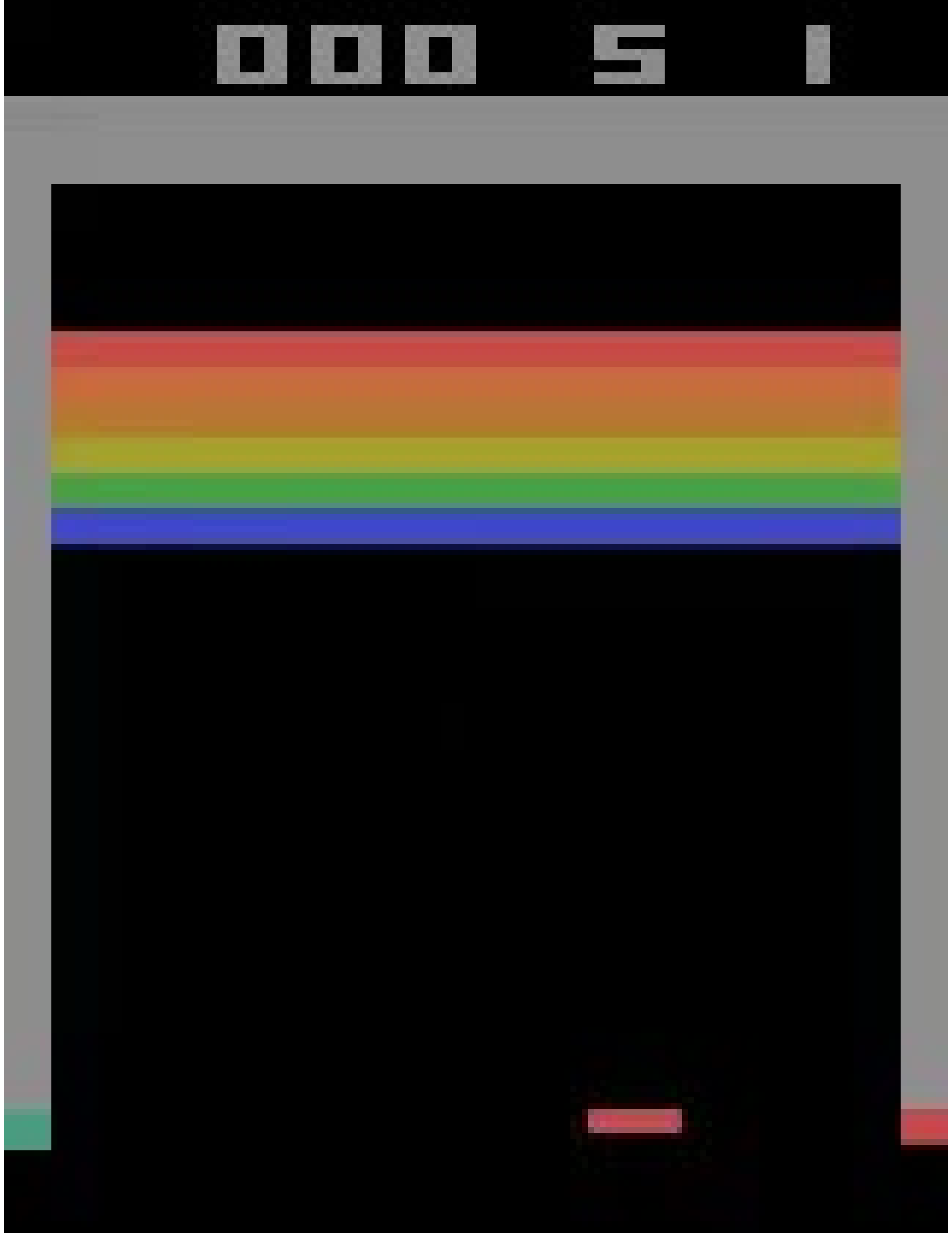
Now, there is certainly other research being carried out in the CIG field that is more directly relevant to real game development (and often dependent on research done in learning to play games, which thus becomes indirectly relevant to game development). For example, we have CI techniques proposed to generate NPC controllers that play interestingly as opposed to just well [1], [2]; CI techniques for automatically finding exploits/bugs in games [3]; CI techniques for modelling the behaviour of human players[4], [5]; CI techniques for making NPCs trainable by human players [6]; and techniques for generating the content of a game, such as tracks, levels or mazes [4], [7].

While the above techniques all represent relevant research directions for game design, they all assume that there is a game there to begin with. Before we let CI loose on

Why is it hot now?

Mnih et al. 2015:

Human-level control of Atari games using Deep Reinforcement Learning



Reinforcement learning?

- Initially, random exploration
- Repeat actions that yield rewards

Predicting Game Difficulty and Churn without Players

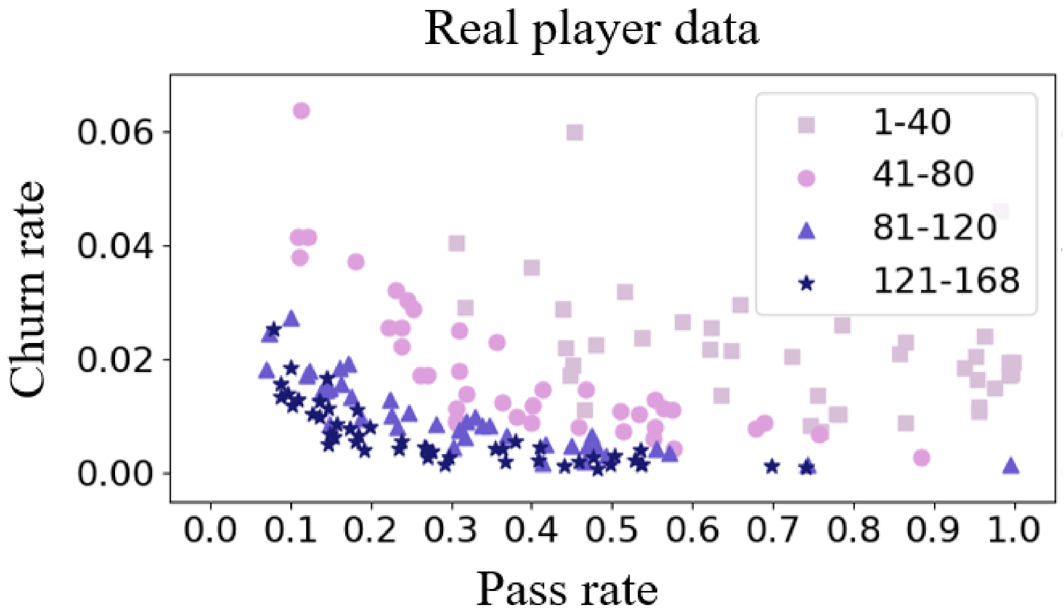
with AI Players

Dataset: 95k players, 168 game levels

Shaghayegh Roohi[2,1], Asko Relas[1], Jari Takatalo[1], Henri Heiskanen[1], Perttu Hämäläinen[2]
1) Rovio Entertainment, 2) Aalto University

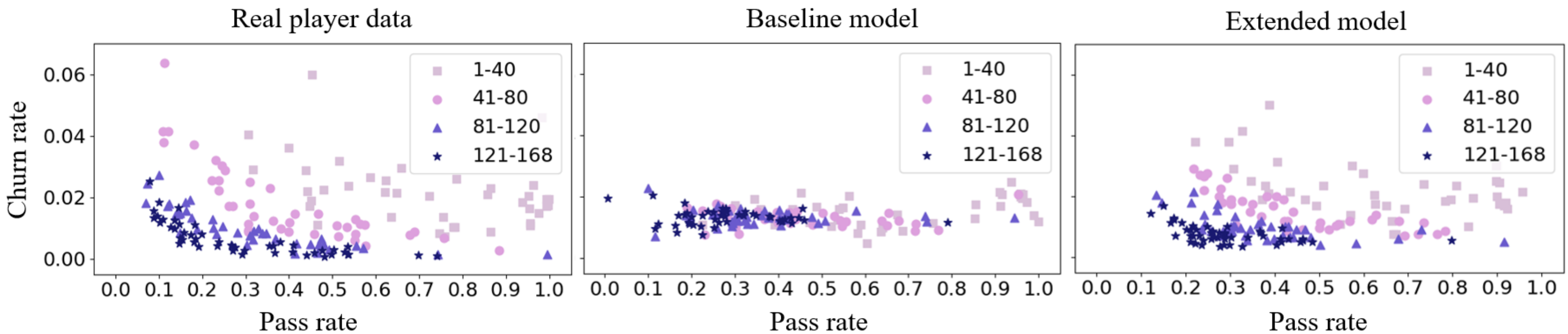
Problem: predict observed player data using simulation

- Data: Pass and churn rates of 168 Angry Birds Dream Blast levels, 95k players
- Pass rate: 1 divided by how many tries a level requires, on average
- Churn rate: probability of a player not returning to the game after the level for 7 days

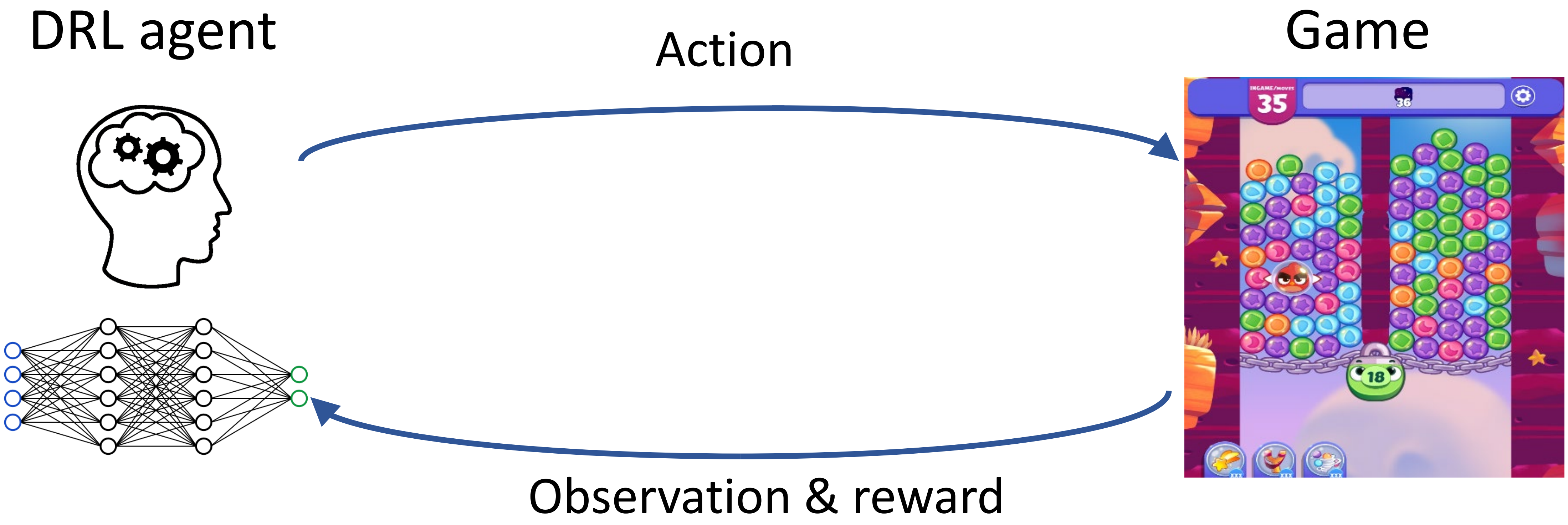


Contribution

- Model the relation between level pass and churn rates by combining
  1. AI game playing using Deep Reinforcement Learning (DRL)
  2. Simulation of how the player population evolves over the game levels, based on simple computational models of skill, persistence, and boredom
- Human-like data emerges from our simulations

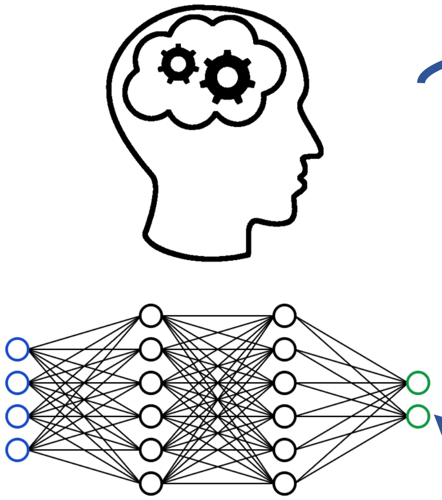


Game-playing AI



Game-playing AI

DRL agent

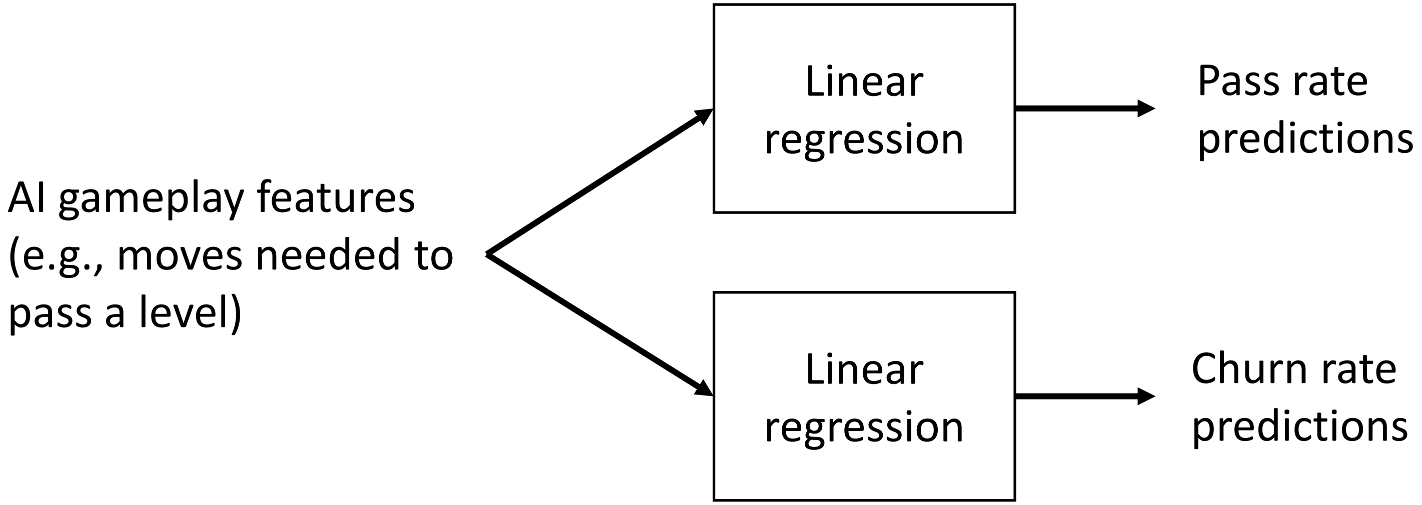


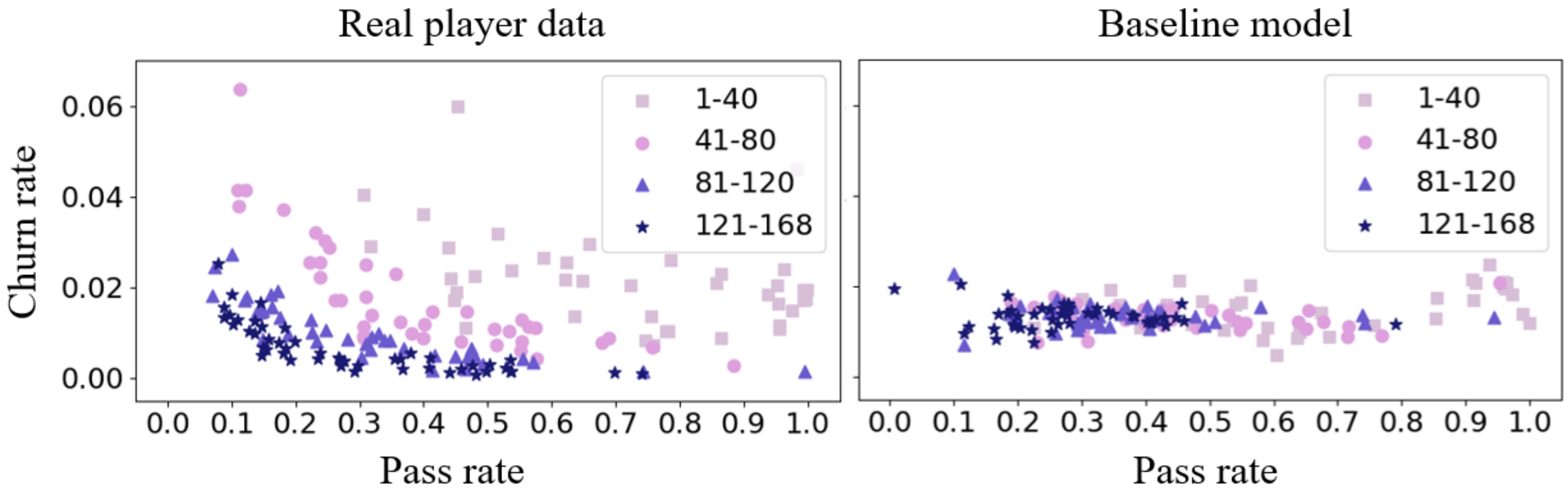
Action

Observation & reward

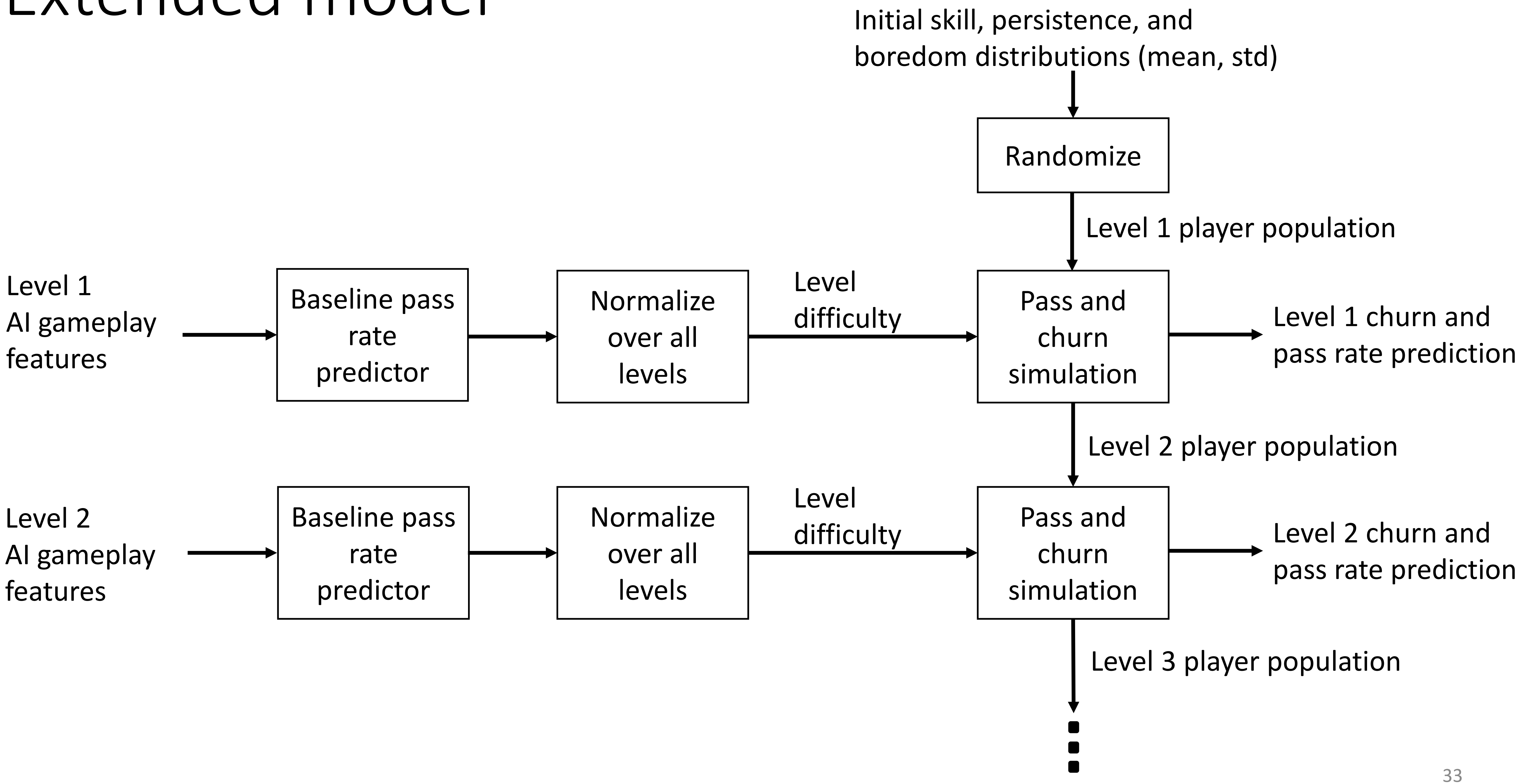
Game

Baseline model

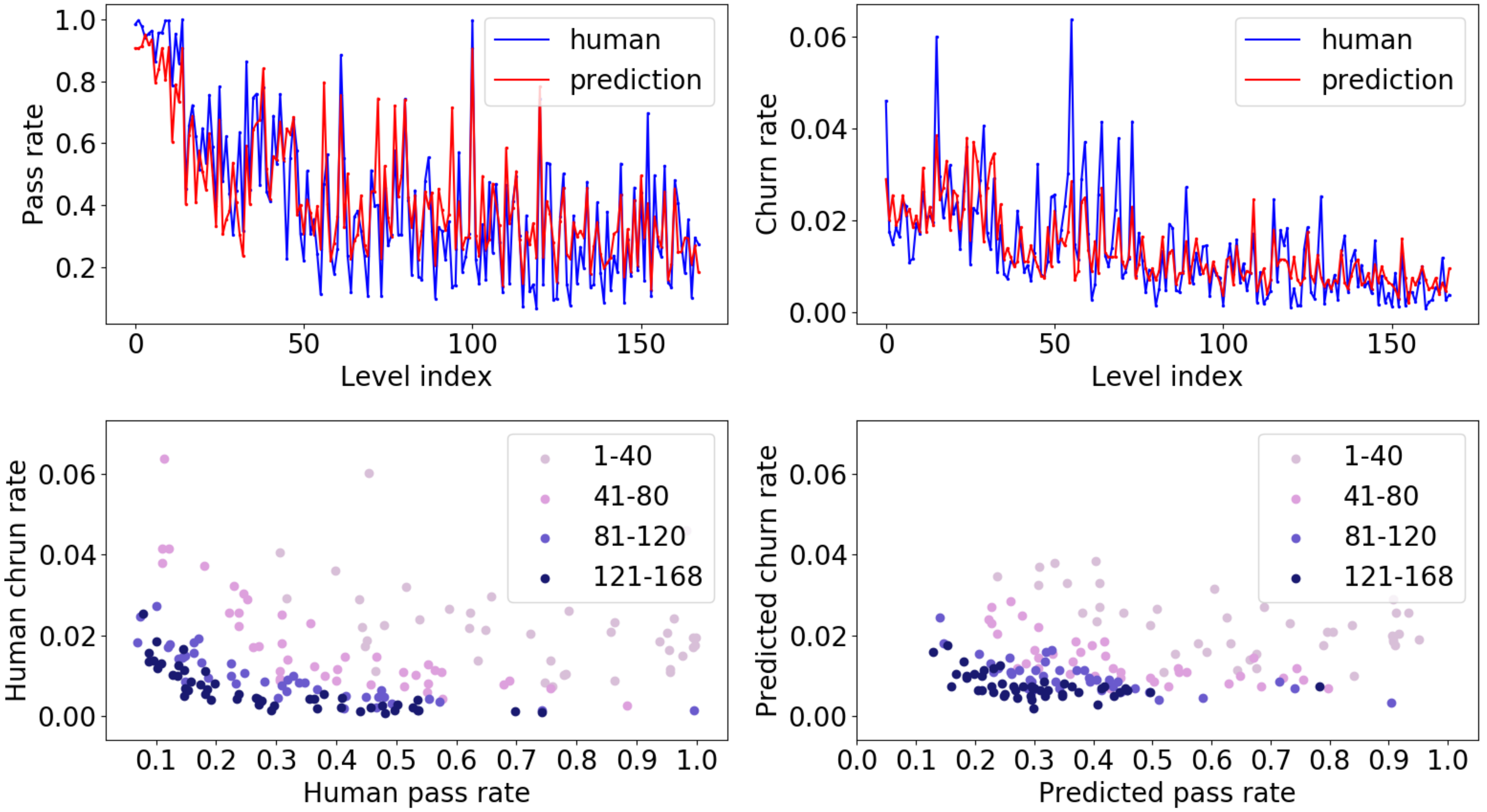




Extended model



Extended model results



Pedagogical approach of the course

- The slides and talks have minimal math, focus on visualizations and hands-on practice
- Everything in Github: Lecture slides, exercises, extra material

Idea → Math → Code → Research paper → Idea

Idea → Code → Math → Research paper → Idea

Kill Math

Bret Victor / April 11, 2011

The power to understand and predict the quantities of the world should not be restricted to those with a freakish knack for manipulating abstract symbols.

When most people speak of Math, what they have in mind is more its mechanism than its essence. This "Math" consists of assigning meaning to a set of symbols, blindly shuffling around these symbols according to arcane rules, and then interpreting a meaning from the shuffled result. The process is not unlike casting lots.

This mechanism of math evolved for a reason: it was the most efficient means of modeling quantitative systems given the constraints of pencil and paper. Unfortunately, most people are not comfortable with bundling up meaning into abstract symbols and making them dance. Thus, the power of math beyond arithmetic is generally reserved for a clergy of scientists and engineers (many of whom struggle with symbolic abstractions more than they'll actually admit).

We are no longer constrained by pencil and paper. The symbolic shuffle should no longer be taken for granted as the fundamental mechanism for understanding quantity and change. Math needs a new

Alan Kay: *Doing With Images Makes Symbols*

Jacques Hadamard, the famous French mathematician, in the late stages of his life, decided to poll his 99 buddies, who made up together the 100 great mathematicians and physicists on the earth, and he asked them, "How do you do your thing?" They were all personal friends of his, so they wrote back depositions. Only a few, out of the hundred, claimed to use mathematical symbology at all. Quite a surprise. All of them said they did it mostly in imagery or figurative terms. An amazing 30% or so, including Einstein, were down here in the mudpies [doing]. Einstein's deposition said, "I have sensations of a kinesthetic or muscular type." Einstein could feel the abstract spaces he was dealing with, in the muscles of his arms and his fingers...

The sad part of [the doing -> images -> symbols] diagram is that every child in the United States is taught math and physics through this [symbolic] channel. The channel that almost no adult creative mathematician or physicist uses to do it... They use this channel to communicate, but not to do their thing. Much of our education is founded on those principles, that just because we can talk about something, there is a naive belief that we can teach through talking and listening.

Up and Down the Ladder of Abstraction

A Systematic Approach to Interactive Visualization

Bret Victor / October, 2011

use arrow keys

"In science, if you know what you are doing, you should not be doing it. In engineering, if you do not know what you are doing, you should not be doing it. Of course, you seldom, if ever, see either pure state."

—Richard Hamming, *The Art of Doing Science and Engineering*

How can we design systems when we don't know what we're doing?

The most exciting engineering challenges lie on the **boundary of theory and the unknown**. Not so unknown that they're hopeless, but not enough theory to predict the results of our decisions. Systems at this boundary often rely on *emergent behavior* — high-level effects that arise

How do we explore? If you move to a new city, you might learn the territory by walking around. Or you might peruse a map. But far more effective than either is *both together* — a street-level experience with higher-level guidance.

Likewise, the most powerful way to gain insight into a system is by *moving between levels of abstraction*. Many designers do this instinctively. But it's easy to get stuck on the ground, experiencing concrete systems with no higher-level view. It's also easy to get stuck in the clouds, working entirely with abstract equations or aggregate statistics.

This interactive essay presents the **ladder of abstraction**, a technique for

Passing the course

- Project work, either design or tech. **BASICALLY: Learn to do something interesting with Unity Machine Learning Agents or Google Colab**
- Example tech project: implement an AI method such as Monte Carlo Tree Search for your game, or integrate some existing tool.
- Example design project: learn to use an existing tool such as Unity Machine Learning Agents, Pix2Pix, style transfer, create some new prototype or experiment using it
- I’ll talk to everyone personally to comment/approve the project

Passing the course

- Submit a report to get the credits, via MyCourses
- Deadline April 1st
- Include:
  - Names of the students in the team
  - What was each student’s starting knowledge
  - What did you create: 1 page text + images, link to video if possible
  - How did it work out / what were the results
  - If purely conceptual work or essay, 5-10 pages text & images (sketches, storyboards...)
  - What each student learned