ARTIFICIAL INTELLIGENCE AND ITS APPLICATIONS

# SERIOUS GAMES AND INTERACTIVE SIMULATION

GIUSEPPE TURINI

Kettering UNIVERSITY

# TABLE OF CONTENTS

# INTRODUCTION

**Serious games** are games that have an *"…explicit and carefully thought-out educational purpose and are not intended to be played primarily for amusement…"*, but this *"does not mean that serious games are not, or should not be, entertaining…"*

CLARK C. ABT. SERIOUS GAMES. 1970.

**Real-time interactive simulators** are serious games consisting in 2D/3D graphics-oriented simulations that can realistically replicate real-world events, while providing a high degree of interactivity and *"real-time"* system reactions.

**Note:** In most cases there is not a clear distinction between a serious game and a real-time interactive simulation. For example: a serious game can integrate (and usually does) elements of real-time interactive simulation, and viceversa.

# A SERIOUS GAME EXAMPLE


See: *"Vital Signs: Emergency Department"* medical interactive trainer by BreakAway Games

# REAL-TIME INTERACTIVE SIMULATION

A real-time interactive simulation is a digital simulation, usually graphics-oriented, designed to be interactive and performed with discrete-time. In particular:

- These are **graphics-oriented** simulators, including 2D or 3D interactive visualization.
- These systems can **realistically simulate** natural phenomena and real-life events.
- These simulators are **highly interactive**, including complex user interfaces.
- These softwares can react in real-time, so **performance optimization** is critical.
- These simulators integrate elements of: **game development, AI, HCI, networking**, etc.
- These systems can be designed for **single-player or multi-player** environments.

Real-time interactive simulation involves a multidisciplinary approach, drawing from: computer science, computer engineering, mathematics, and physics.

# REAL-TIME INTERACTIVE SIMULATION (2): DISCRETE TIME

In a real-time interactive simulator the simulation is performed with discrete-time.

Usually, the simulation discrete-time step is constant, that is: time moves forward in steps of equal duration, and this is commonly known as fixed time-step simulation.

However, in some cases, a variable time-step simulation can also be used.

Fixed time-step simulation is preferrable in some situations, whereas variable time-steps simulations provide better results in others.

Example: In an interactive video game integrating some physics simulation, the graphics visualization runs at a variable time-step (frame rate) while the physics updates run at a fixed time-step (usually ~50 Hz).

Example: In a real-time interactive simulator integrating physics and haptics, the physics updates run at a fixed time-step (usually ~50 Hz) while the haptic rendering runs at a different fixed time-step (usually ≥1 KHz).

# REAL-TIME INTERACTIVE SIMULATION (3): ONLINE OR OFFLINE

To solve the mathematical equations at a given time, each simulation variable is solved successively as a function of variables of the previous time-step.

In a discrete-time simulation, the time required to solve all mathematical equations at a give time may be shorter or longer than the duration of the simulation time-step.

If the computing time to solve all mathematical equations (update time) is shorter than the simulation time-step, the simulation is called accelerated simulation (not real-time).

If the computing time to solve all mathematical equations (update time) is longer than the simulation time-step, the simulation is called offline simulation (not real-time).

If the computing time to solve all mathematical equations (update time) equals the simulation time-step, the simulation is called real-time simulation (or online simulation).

# REAL-TIME INTERACTIVE SIMULATION (4): MULTITHREADING

In a real-time interactive simulator the simulation is usually performed in a multi-threaded environment (*i.e.*, multiple software modules executed concurrently), allowing the decoupling of simulation tasks with different requirements.

In particular:

- Multithreading allows **optimal use of multi-core CPUs**.
- The **graphics rendering** thread usually runs at a variable frequency (frame-rate, >30 Hz), depending on the content visualized, and that should be constantly maximized.
- The **physics simulation** thread usually runs at a constant frequency (physics updates, >50 Hz), depending on the mathematical equations solved.
- The **collision detection** thread is usually integrated in the physics thread.
- The **haptics feedback** thread usually runs at high constant frequency (>1 kHz), to provide appropriate tactile/force feedback.

# REAL-TIME INTERACTIVE SIMULATION (5): -IN-THE-LOOP

The capability to solve simulation equations in real-time enables different types of testing:

- Hardware-in-the-Loop (HIL): A testing technique for complex real-time embedded hardware modules, by enabling their interactions with a simulated system.

  For example: An hardware-in-the-loop test can simulate a car engine interacting with the ECU (engine control unit) with real inputs/outputs to enable the ECU testing.

- Human-in-the-Loop: A testing technique for complex real-time systems, by using interactive simulation involving always a human user that influences its outcomes.

  For example: A human-in-the-loop test can simulate an airplane cockpit interacting with the pilot with real inputs/outputs to enable the pilot testing/evaluation/training.

- Software-in-the-Loop (SIL): A testing technique for software algorithms, by simulating their interaction with a software environment/platform.

  For example: A software-in-the-loop test can simulate a self-driving car interacting with an AI algorithm with real inputs/outputs to enable the testing of its code.

# GAME ENGINES

Today, most serious games and real-time interactive simulators are developed by using game engines, and less frequently they are implemented completely from scratch.

A game engine is a software framework mainly designed for the development of video games. These are some of the functionalities/modules typically included in a modern game engine:

- A 2D/3D **editor** to configure the game content.
- **Code libraries** to support game programming/scripting.
- A **rendering engine** (renderer) to visualize 2D/3D graphics.
- A **physics engine** responsible for rea-time interactive physics simulation.
- A **collision detection** module to manage collision events and collision responses.
- Then other modules for: sounds, animation, VR-AR, AI, networking, etc.

These are the main game engines on the market today:

- **Unreal**, **Unity**, Godot, Amazon Lumberyard, CryEngine, etc.

# GAME ENGINES (2)

Modern game engines are development platforms for real-time interactive visual applications, including: 2D/3D graphics, VR-AR integration, AI modules, etc.


**See:** The *"Unity Editor"* Software Part of the Unity Game Engine

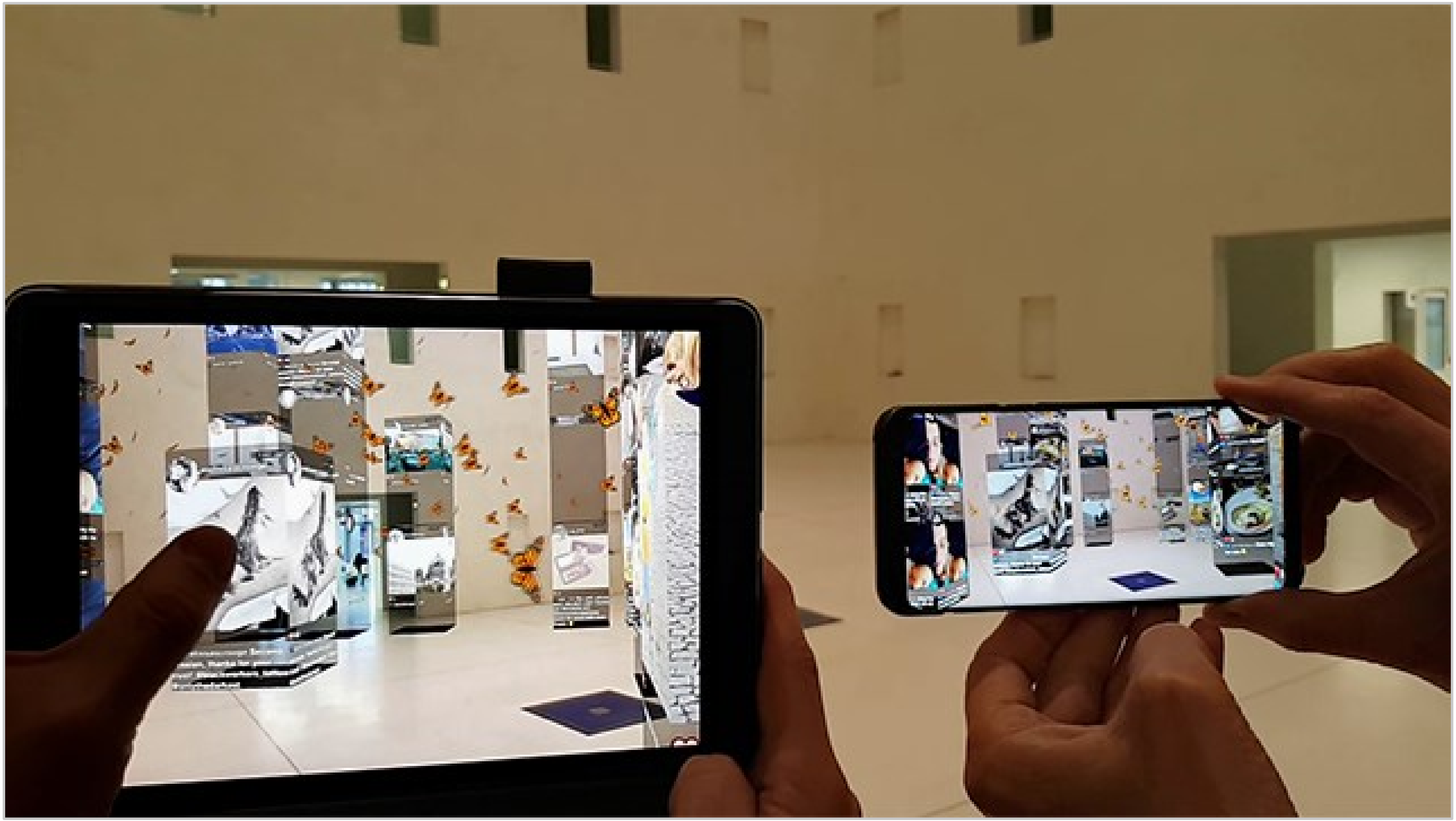
# APPLICATIONS

These are just some examples of applications of serious games to different fields:

- Art games.
- Business simulation games.
- Educational games.
- Fitness games.
- Serious games in healthcare.
- Serious games in security and safety.
- Scientific interactive simulators.

# ART GAMES

An *"art game"* is a serious game used to create interactive and multimedia art. These games use aesthetics, atmosphere, and interactivity to encourage player creativity.



**See:** *"10.000 Moving Cities"*, an Aumented Reality Multiplayer Art Game by Marc Lee

# BUSINESS SIMULATION GAMES

Serious games that are economic simulators (economic simulation games, tycoon games) that focus on the (micro) management of economic processes.


**See:** [*"Sim City 4"*, a City-Building Business Simulation Game by Electronic Arts]

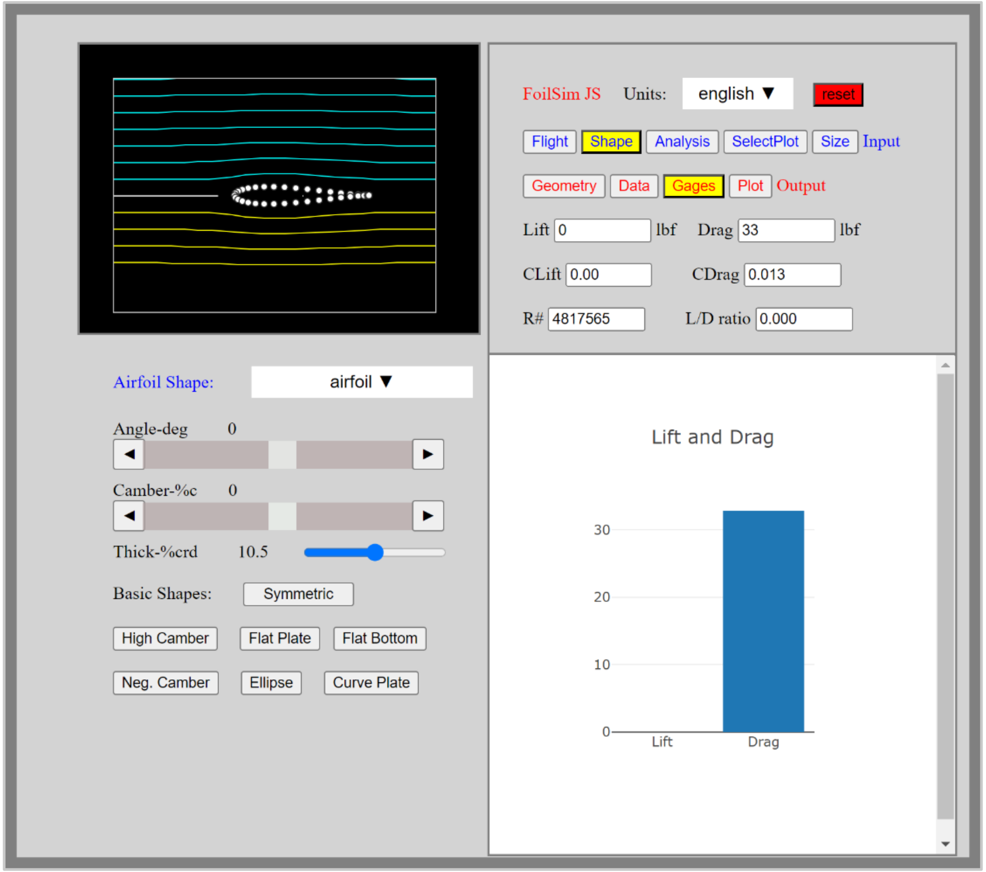
# EDUCATIONAL GAMES

Serious games can provide players with the opportunity to gain experience in terms of both theory and practice, and they can also be used in university curricula to consolidate learning.


See: *"Rocksmith"*, an Educational Game to Learn to Play Guitar by Ubisoft

# EDUCATIONAL GAMES (2)

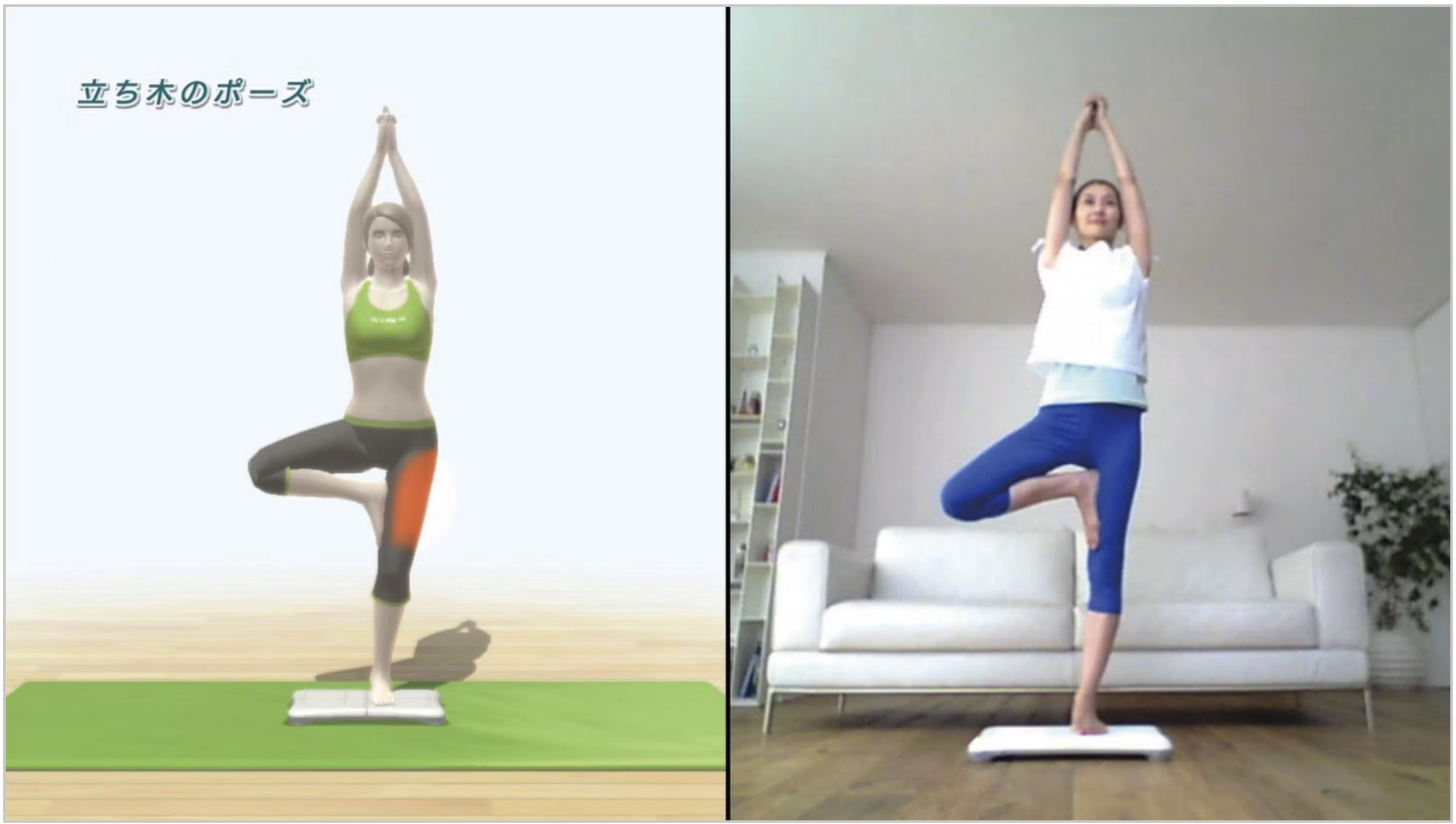
Real-time interactive simulations can be integrated in education programs to support traditional teaching tools and enhance the learning experience of students.




**See:** *"Beginners Guide to Aerodynamics"*, Interactive Simulations by NASA

# FITNESS GAMES

Serious games can integrate a form of exercise (exergames, or gamercises), targeting coordination, muscles, etc.; and can also be used for training/evaluating elite athletes.




See: *"Wii Fit U"*, an Exergame by Nintendo for the Wii U Console

# FITNESS GAMES (2)

Serious games can also be used as training tools for athletes.

**See:** "QB Sim Academy" Simulation-Based Training for Quarterbacks by Sports VTS

# SERIOUS GAMES IN HEALTHCARE

Healthcare serious games can be: (1) training systems for doctors, (2) motivational tools and treatment elements for patients, (3) supporting educational platforms for medical students.


See: *"Touch Surgery"* interactive surgical simulator by Medtronic

# SERIOUS GAMES IN HEALTHCARE (2)

Real-time interactive simulations can also serve as training tools for medical staff.


**See:** *"Laparo S"* the Most Realistic Laparoscopy Interactive Simulator by Virta Med

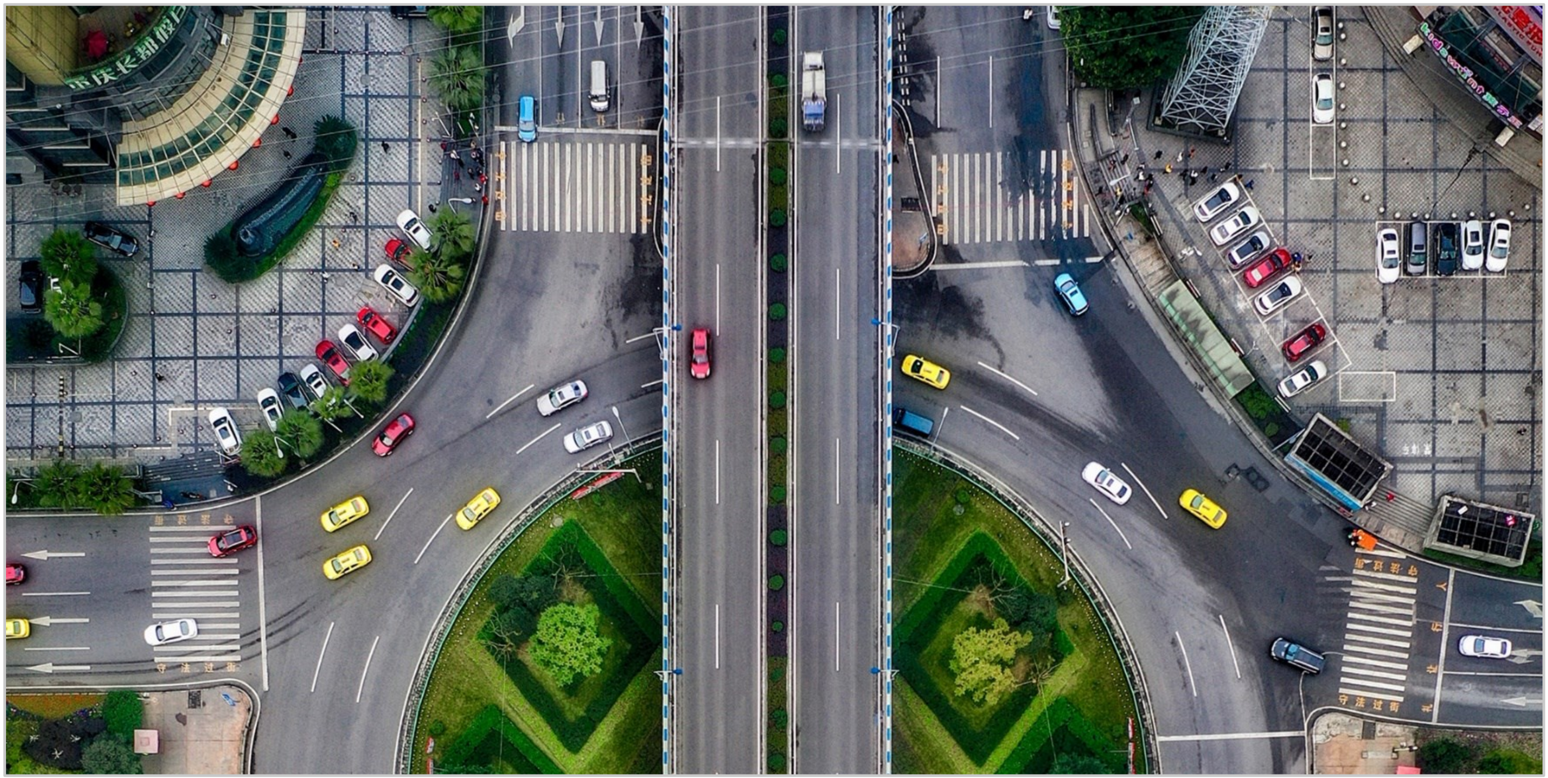
# SERIOUS GAMES IN SECURITY AND SAFETY

Serious games are frequently used in security (disaster control teams, defense, fire brigades, police forces, crisis centres) to simulate critical scenarios (natural disasters, acts of terrorism, danger prevention, emergency care).


**See:** Korea Tech Today - *"ETRI Develops VR-Based Firefighting Training Simulator"*

# SCIENTIFIC INTERACTIVE SIMULATORS

Serious games can be simulation systems for scientific research, allowing the configuration of experiments and the relative data collection to study different processes or scenarios.



See: *"PTV Vissim"*, a Multimodal Traffic Simulation Software by PTV Group

# SCIENTIFIC INTERACTIVE SIMULATORS (2)

Real-time interactive simulations can also serve as supporting tools for scientific research.

See: *"CARLA: An Open Urban Driving Simulator"* by Dosovitskiy et al.

# SCIENTIFIC INTERACTIVE SIMULATORS (3)

Real-time interactive simulations can also serve as supporting tools for scientific research.


**See:** [*"Foldit"*, an Online Video Game About Protein Folding by the University of Washington]

# AI AND SERIOUS GAMES

**Game AI:** Usually, AI in video games is applied by using techniques part of a distinct AI subfield called Game AI (different from *"true AI"*). Game AI uses a mix of: computer science, control theory, robotics, computer graphics, to achieve *"intelligent"* behaviors in non-player characters and enemies. Game AI is focused on improving gameplay and player experience, rather than the learning of the machine.

Nowadays, *"true AI"* is also applied in serious games and real-time interactive simulators for:

- **Assessment:** Assessing learning, players, and programs in educational games.
- **Decision-Making:** Evaluating current player/game status, and decide action.
- **Design and Validation:** Designing game elements and validating game tasks.
- **Pathfinding:** Determining the *"best"* route between 2 game locations.
- Team Strategy: Deciding the group tactic to coordinate a collaborative action.
- **Player and Experience Modeling:** Modeling player and experience in real-time.
- **Procedural Content:** Automating the creation of game content.

# ASSESSMENT

Game-based assessment is used in educational games with 3 main goals:

- Assisting learning (formative assessment).
- Assessing the player capabilities.
- Evaluating educational programs.

**Game Learning Analytics:** Collecting, analyzing, and extracting information (off-game and in-game) by using an educational serious game.

The AI techniques used for game-based assessment include:

- Linear regression.
- Decision trees.
- Neural networks.
- Bayesian networks.

# DECISION-MAKING

**Decision Tree:** Data structure representing a prediction model based on a set of decision rules. It can be used for: diagnosis, training/learning, prediction, rehab, etc.

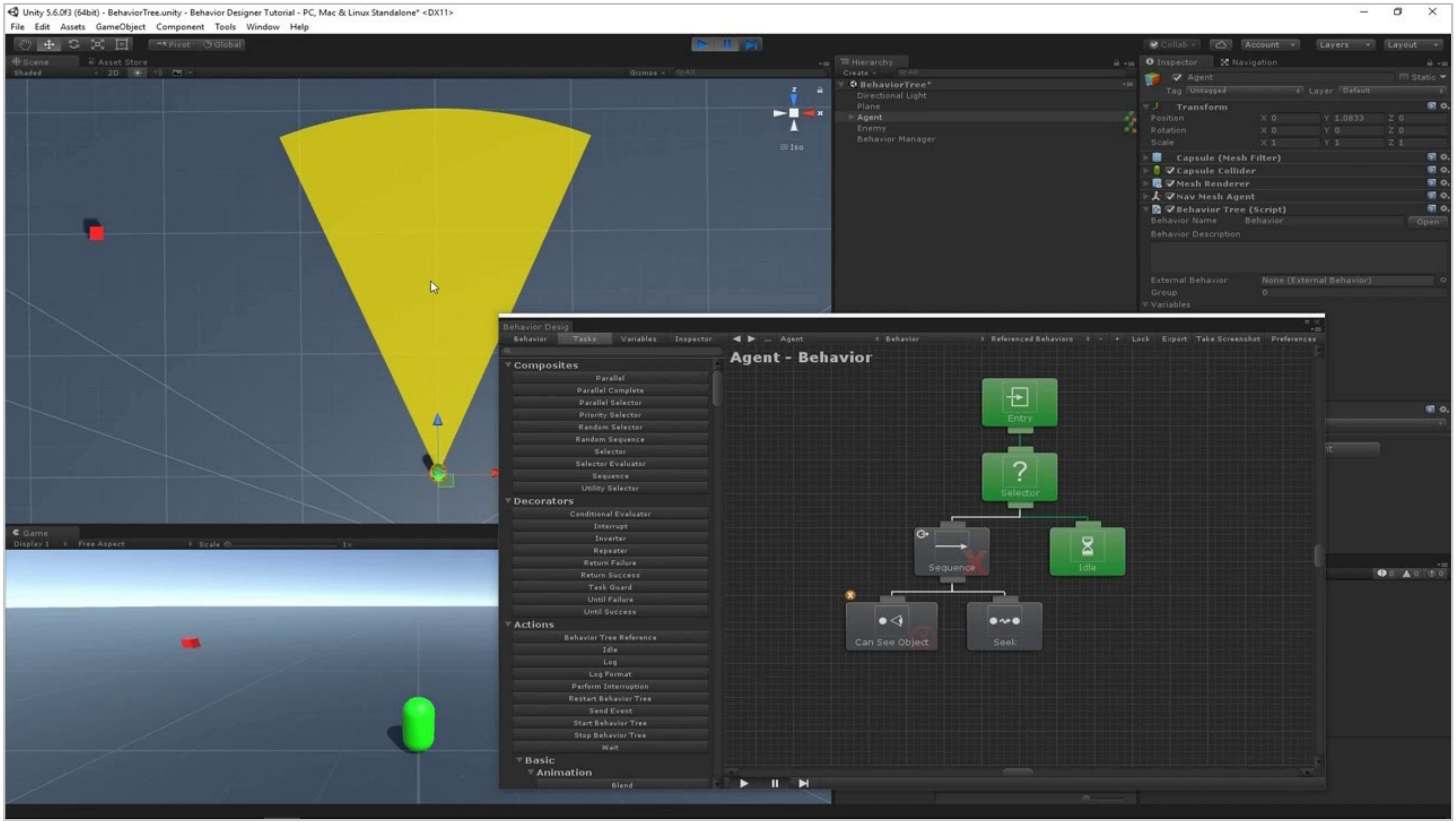
**Fuzzy Logic:** Set of mathematical techniques designed to cope with gray areas. It can be used for: teaching/training, prediction, rehab, support, etc.

**Finite State Machine (FSM):** A mathematical model of computation including states and state-state transitions. At any give time, the FSM is in exactly 1 state, and can switch from the current state to another state by using a transition triggered by events.

**Monte Carlo Tree Search (MCTS):** A heuristic search algorithm for decision processes. In video games, MCTS is used to solve the game tree focusing on the most promising moves weighted considering *"playouts"*.

# DECISION-MAKING (2)

AI decision trees can be used in game development to create complex game agents.




See: *"Behavior Designer"* a Visual Editor for Decision Trees for Unity by Opsive

# DESIGN AND VALIDATION

**Game Design:** Designing content, rules, and mechanics of a game.

**Game Validation:** Verifying and balancing game tasks and their difficulty.

In recent years, AI started being used game design, and these are just some examples:

- Machine learning to create images from text, allowing fast design of game images.
- AI tools for creating different game assets (game arts, icons, logos, etc.).
- AI-powered software to create game dialogues using realistic voices.
- AI methods can be used to create procedural game environments (forests, mazes, etc.)

Different data-driven AI techniques can also be applied to game validation, in particular:

- Unsupervised machine learning.

# DESIGN AND VALIDATION (2)

Game design can use AI procedural content generation to create complex game content.


See: Unreal Engine 5 – Artificial Intelligence in Unreal Engine

# PATHFINDING

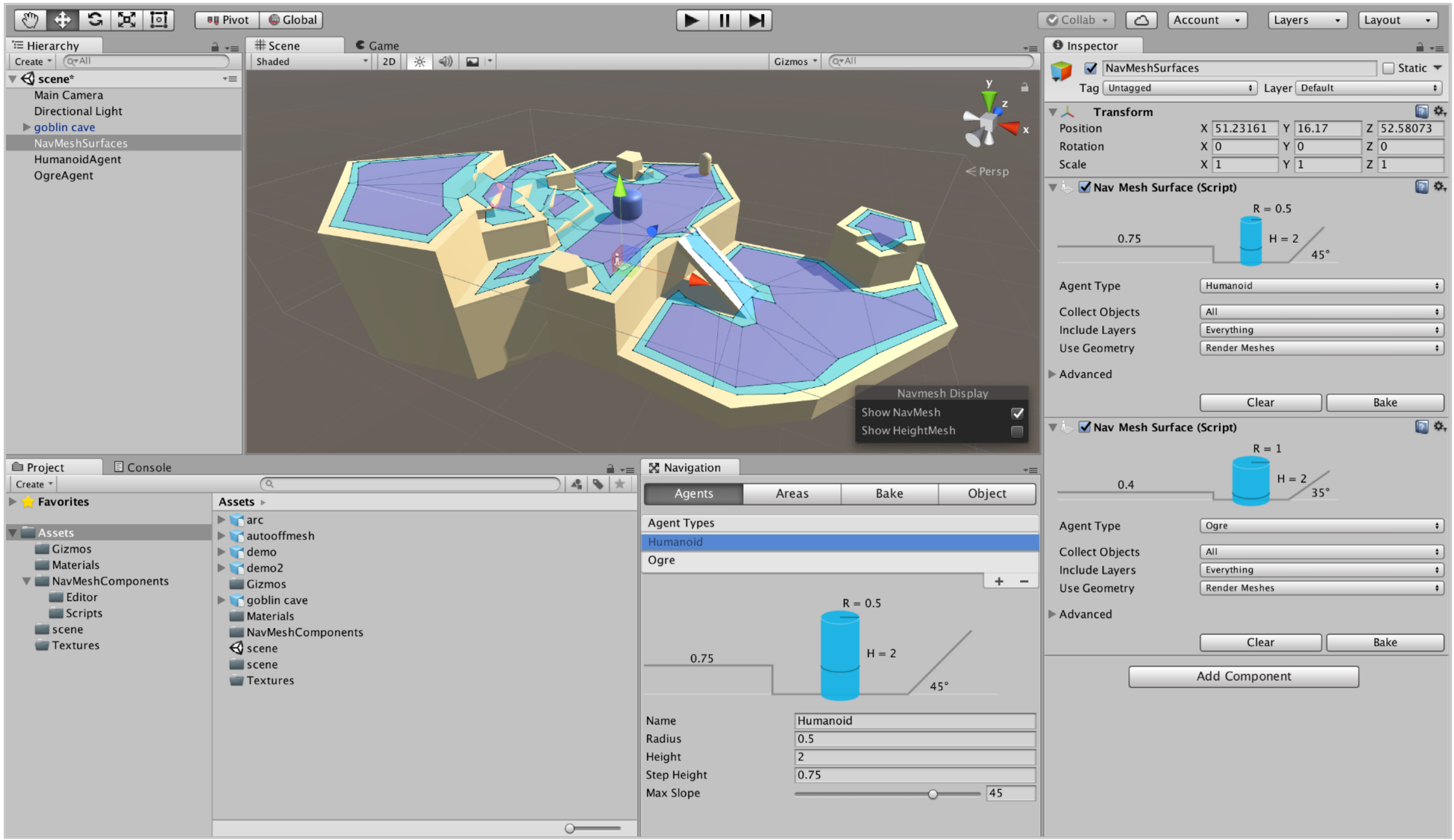
**Grid-Based:** The map/terrain is represented by a uniform structured grid (2D/3D) that supports the use of specific pathfinding algorithms (A*, IDA*, etc.).

**Navigation Mesh:** The map/terrain is represented by a preprocessed *"patchwork"* made by irregular polygons (navigation mesh), with each polygon representing a map area with different *"walking"* characteristics (slow, fast, non-walkable, no jumping, etc.). Then AI algorithm can use this mesh to determine *"best"* paths.

**Waypoints:** Special map locations, usually manually configured, that AI algorithms can use to facilitate the determination of the *"best"* path between 2 locations.

# PATHFINDING (2)

AI navigation meshes can be used for pathfinding tasks on an unstructured game map.




See: "*NavMeshes*" the Navigation Meshes in Unity

# PLAYER AND EXPERIENCE MODELING

AI can be used for **player experience modeling (PEM)** at runtime, that is: to dynamically customize the game in real-time considering player ability and emotions. For example:

- Predicting player decisions and actions.
- Evaluating in real-time the engagement of a player.
- Optimizing in real-time the game design.

PEM can be based on a variety of: player data, inputs, and physiological signals.
These are just some example of unobtrusive PEM-related AI components:

- Facial emotion recognition in real-time by using **artificial emotional intelligence (AEI)**.
- Adaptation and assessment (TwoA) using **AI to balance game difficulty** in real-time.
- Stealth assessment powered by **AI for unobtrusive formative assessment** to automatically evaluate player knowledge and skills without the need for explicit tests.

# PLAYER AND EXPERIENCE MODELING (2)

**Player Modeling:** Creating computational models to detect, predict, and describe the human player behaviors. These models can be built using static (player age) or dynamic player information (gameplay style).

There is a multitude of AI algorithms used for player modeling, these are just some examples:

- Supervised/unsupervised learning.
- Probabilistic algorithms.
- Evolutionary methods (genetic algorithms).
- Reinforcement learning.

# CURRENT CHALLENGES

It is clear that both serious games and real-time interactive simulators can benefit from the application of AI techniques. However, in addition to the general challenges AI is facing, there are some specific challenges to apply AI in serious games, for example:

- Game design is a complex process involving both artistic and technical aspects, and it is difficult to describe it so that AI methods can be properly applied.
- The validation of a serious game in respect to traditional methods is a challenging task. Nowadays, most serious games include individual non-AI-based validation; so, there is a lot of work to be done before we can use AI-based general validation.
- Most AI methods requires large amount of data in input. Unfortunately, most serious games suffer from data scarcity (because of a small player population); so, any AI technique used is affected by this issue.

# FUTURE DEVELOPMENTS

The current benefits that AI is providing to both serious games and real-time interactive simulators could lead to promising future innovations, such as:

- AI could power systhetic data generation capable of simulating different human behaviors as well as a variety of environmental scenarios for serious games.
- Serious games and AI algorithms could be used to support the data sharing needed by different disciplines as computational social science.
- To study causality different scientific disciplines (social science, computer science, etc.) use a variety of methods: from randomized controlled trials, to predictive models. AI and serious games could allow the integration of different approaches into a single system to improve this kind of scientific research studies.

# ABOUT THE AUTHOR

Name: Giuseppe Turini, Associate Professor of Computer Science

Website: sites.google.com/view/turinig

Email: gturini@kettering.edu

Education: M.Sc. in Computer Science @ University of Pisa (Italy)
Ph.D. in Health Technologies (Comp. Eng.) @ University of Pisa (Italy)

Past Labs: EndoCAS Research Center @ University of Pisa (Italy)
Visual Computing Lab @ CNR of Pisa (Italy)

Teaching: Computing and Algorithms I-II-III, Computer Graphics
Elements of Game Design, Virtual Reality, Advanced Game Development

Research: Computational Geometry, Interactive Simulation, Computer-Assisted Surgery
VR-AR, HCI-HMI, Serious Games.

# REFERENCES

- These slides are available online at: github.com/turinig/cetl-prof-dev


- *"Serious Games"* by Abt. 1970.
- *"The What, Where and Why of Real-Time Simulation"* by Belanger et al. 2010.
- *"Serious Games and Edutainment Applications"* by Ma et al. 2011.
- *"A Review of AI in Serious Games for Public Health"* by Rafiq et al. 2021.
- *"Game AI Revisited"* by Yannakakis. 2012.
- *"Serious Games and AI"* by Perez et al. 2023.

- Healty Simulation - "*Serious Games*"
- Springer - JCSG Serious Games Conference Proceedings 2022
- Springer - GALA Games and Learning Conference Proceedings 2022
- IEEE - SeGAH Serious Games for Health Conference Proceedings 2023