Player-Customized Puzzle Instance Generation for Massively Multiplayer Online Games

Alexandru Iosup (Delft University of Technology, the Netherlands)

A.Iosup@tudelft.nl

Massively Multiplayer Online Games

• **Large-scale distributed application**
Real-time virtual world simulations entertaining millions of players spread across the world [1,2].

• **Tight Quality-of-Service constraints**
Responsiveness: 10-35 updates/second. **Content: users demand customized content on-the-spot, or quit**.

• **Fashion-driven: one quitter triggers an avalanche**
An MMOG may lose 20% of its players in just a few days [2].

• Bottom line: **Produce content or die!**

What is MMOG Content?

• **Content immerses the players into the Virtual World**
Trees, Houses (objects on which players can act), Maps (topological connection between trees), **Puzzles (logical challenges that players solve)**, Quests (suites of puzzles).
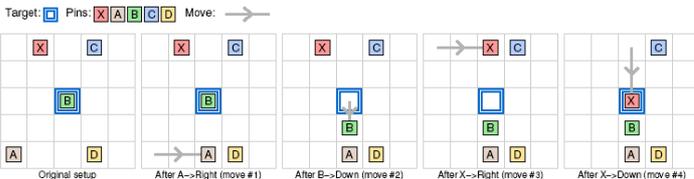

Target: Pins: X A B C D Move:

| Original setup | After A-->Right (move #1) | After B-->Down (move #2) | After X-->Right (move #3) | After X-->Down (move #4) |

Figure 2: A LunarLockout puzzle instance and its best solution.

Puzzle Content Generation for MMOGs

• **Balance puzzle difficulty (challenge #1)**
Match puzzle instances given to players to player solving ability

• **Puzzle freshness (challenge #2)**
Give players puzzle instances that lead to fresh experience.

• **Scalability (challenge #3, addressed previous work [3])**
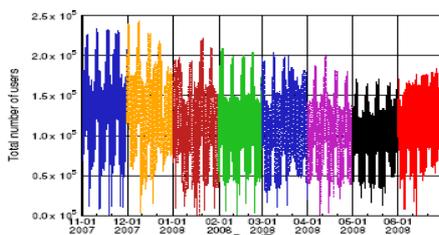Hundreds of thousands of concurrently active players

Figure 2: The number of concurrently active players over time in RuneScape, the second most popular MMORPG after WoW [2].

Current Content Generation Approach

• **Teams of human designers**

• **Pro: balanced content**

• **Cons: difficult to manage, costly, not enough new content for each player (mismatch challenge #2), does not scale (mismatch challenge #3).**

Our Approach: Dynamic Puzzle Instance Generation

• **Automate puzzle instance generation on on-demand resources (grids, clouds, …)**

• **Pro: scalability [3]**

• How to achieve:
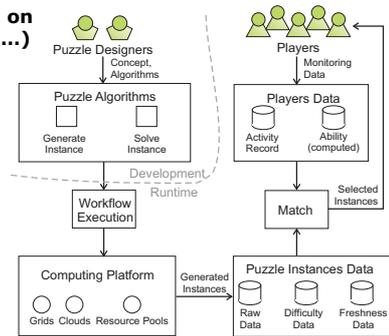 - **Puzzle difficulty vs solving ability**
  - **Puzzle freshness**

Figure 3: Our approach for dynamic puzzle instance generation.

Puzzle Difficulty vs Solving Ability/Puzzle Freshness

• **Consider puzzle-agnostic and puzzle-specific metrics**
Puzzle-agnostic metrics are general to all puzzles, but not very expressive.
Puzzle-specific metrics apply to a few or just one puzzle, but are very expressive.

• Puzzle difficulty vs Solving ability: we design a **history-based mechanism for evaluating the solving ability of the players**
Player skill level and leveling time. Future work: to evaluate matching accuracy.

• **Puzzle difficulty vs Solving ability**
  - Solution size = the number of moves in the optimal solution (agnostic)
  - Joker uses = Number of times the Joker (X) pin is moved (specific)

• **Puzzle freshness**
  - Each pin is different (agnostic), different animations for each direction (specific)

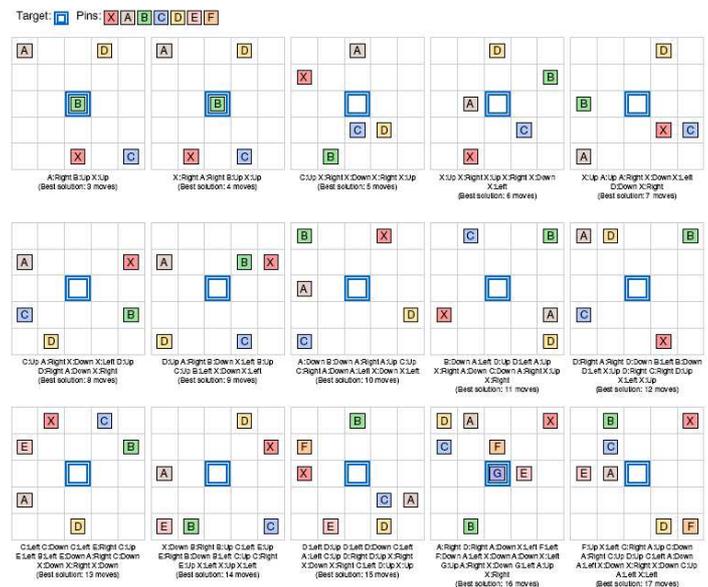## Our Results: Automatic Puzzle Instance Generation

Figure 4: Auto-generated Puzzle instances. **Puzzle: Can you assess the puzzle difficulty and freshness? Hint: Use the metrics introduced in this poster.**

References

[1] B. S. Woodcock, "An analysis of mmog subscription growth," Report, 21 Edition. [Online] Available: http://www.mmogchart.com , Jun 2006.
[2] V. Nae, A. Iosup, S. Podlipnig, R. Prodan, D.H.J.Epema, T. Fahringer, Efficient Management of Data Center Resources for Massively Multiplayer Online Games, In the ACM/IEEE SuperComputing Conference on High Performance Networking and Computing (SC'08), pp.10-17, Nov 2008.
[3] A. Iosup. POGGI: Puzzle-based Online Games on Grid Infrastructures. In Euro-Par, LNCS, pages 390–403, 2009. **Best paper award!**

Google: "Iosup"
http://www.pds.ewi.tudelft.nl/~iosup/

TUDelft
Technische Universiteit Delft