IMGD 1001:
Game Balance

Robert W. Lindeman

Associate Professor

Interactive Media & Game Development

Department of Computer Science

Worcester Polytechnic Institute

gogo@wpi.edu

Outline

- Gameplay (done)
- Level Design (done)
- Game Balance (next)

Mini-Outline

- Broadly, game balance includes:
  - Player-Player (next)
  - Player-Gameplay
  - Gameplay-Gameplay

Player/Player Balance (1 of 2)

- Players should have “fair” chance of winning
  - Advantage should only be for skill
  - Any luck should be infrequent, minor, and equal to both
- Ex: *Virtua Fighter*
  - Say, Sarah Bryant beats Lion every time
  - Does that mean the game is unbalanced?
    - Not necessarily, look more closely
- Suppose a friend said she could beat everyone as Sarah Bryant all the time.
  - Would only be a problem if a beginner as Sarah always beat an expert as Lion
  - What if you could choose your character? Sarah versus Sarah?

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Player/Player Balance (2 of 2)

- Allow victory to be decided by *skill* and *judgment*
- Avoid results caused by a stroke of luck
  - Right from the start or magnified as game progresses (ex: start close to gold mine provides escalating advantage)
- Simplest way is to have *symmetry*
  - Same weapons, maneuvers, hit points (sports do this – teams are nearly always symmetric)
  - But note, this is not always the most interesting way.
  - You may want different moves for fighters, say. (More later)

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Symmetry Example

- Two heroes square off for a duel, poised in kung fu stances. Both are equally matched.
- They wait for an advantage.
- Hours pass...Days pass...
- Breeze comes by, flicks spec of dust in one’s eye
- Blinks, frowns, then bows
- We know result without fighting → tiny asymmetry was enough to decide the outcome!
- If breeze or dust decided the game, is that ok?
  - No...you’d want your money back!
- Don’t want to decide by factors out of user control
  - Keep symmetric

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Symmetry

- □ Symmetry is fine in abstract games
  - ■ Ex: *chess, basketball*
- □ In realistic games, it would be a problem
  - ■ Ex: *U.S. versus Iraq*, game symmetry would be bothersome since it wouldn't be realistic
- □ While symmetry is easy, it's kind of an insult
  - ■ Ex: *LOTR BfME,* Wargs are the same as horses...but Wargs can bite in book/movie!
- □ Better is *functional* symmetry that is not obvious

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Symmetry in Level Design

- Can avoid obvious symmetry
    - Ex: each player has impassible region on flank (but water for one, mountain range for another)
        - Knights and soldiers can’t cross
        - Later on, advanced units can cross
        - Choice of unit depends upon barrier
            - Mountaineers to storm, ships to cross sea
            - Or bluff, and then go up the middle
- Players can choose asymmetric start location
    - Should not be deciding factor
        - Ex: you choose downwind port, so you lose (like dust in eye)
    - Avoid making start location critical decision
    - Ex: potential mines in many spots, so not critical

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Symmetry in Game Design (1 of 2)

- Make all choices for players functionally the same
  - Ex: *Warcraft 2* – humans have griffons and orcs have dragons; both flying toughies.
- But even slight differences make it interesting
  - Ex: *Warcraft 2* – orc player's runes explode, making use in mountain passes good
- "Just broken" asymmetry easier to manage than total asymmetry (can compensate for it)

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Symmetry in Game Design (2 of 2)

- Making choices for players different, yet balanced is tougher
- Ex: *Starcraft*: Protoss, Zergs, Terrans – all very different (Same with *Command and Conquer: Generals*)
  - Imagine the hours of playtesting!
  - Recommended only for deep pockets
  - Starcraft is often a “benchmark” against which to judge other RTS game balance
- Also, if re-creating historical simulation, tradeoff between fairness and authenticity
  - Ex: *Conquistadors vs. Aztecs* – Aztecs are doomed, but may be no fun. Not symmetric

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Mini-Outline

- Broadly, game balance includes:
  - Player-Player
  - Player-Gameplay (next)
  - Gameplay-Gameplay

Player/Gameplay Balance: Introduction (1 of 3)

- Remember that the business is about *interactivity*
  - Think about the player's relationship to the game
  - Ex: If you had to "tune" the TV every time you changed the channel, you wouldn't do it much
  - Likewise, the player should not struggle for a small reward
- Ex: *Baldur's Gate*
  - Attributes are 3-18
  - Why?
  - Can re-roll if don't like your numbers.
  - So, re-roll until all 18s.
  - Test of endurance!

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Player/Gameplay Balance: Introduction (2 of 3)

- Player/Gameplay balance entails balancing challenges against player's improvement curve
  - (We talked about this previously, see Gameplay slides with graphs)

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Player/Gameplay Balance: Introduction (3 of 3)

- Often, games use difficulty settings (player manually selects)
  - Still challenge to make the "Normal" level right
- Compromises
  - Could ask the player up front some questions, then recommend a setting
    - Ex: Have you played FPS before?
  - Could have the player do a tutorial level, then recommend a setting

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Sub-Outline

- Again, true balance is an art, but three guidelines that can help
  1) Reward the player
  2) Let the machine do the work
  3) Make a game that you play *with*, not *against*

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Reward the Player

- The player will have to learn, and will make mistakes
  - Discouraging!
- Want to offset this with a reward when they do something right
  - Ex: *Virtua Fighter*, takes longer to learn complicated moves
  - Sarah's backflip. Reward comes from seeing flip (eye candy) and punch in kidneys (payoff)
- Best used when it expands game options
  - Ex: "Now with backflip, I can see new use for reverse punch"
- Better to reward the player for doing something *right* than to punish them for doing something *wrong*
  - Punishment makes players not want to play

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Let the Machine do the Work

- The interface should show the player the world and let him/her manipulate it
- The computer is a tool to take care of a wide-range of tedious tasks
  - If tasks are not fun, don't make the player do them
- Blur of boundary between chore and game feature
  - RPGs could provide graph paper so player can manually draw map as they explore... but is that fun?
  - Ex: In *D&D*, can tell D.M. "We go back to the dungeon entrance". Easy, fun. What if a game makes player walk back over the map that has been seen? Boring, not fun.
  - Ex: *Myst* provides lightning bolt move to avoid tedium
  - Other examples?
- Also, if option is no-brainer, then AI should take care of it!

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Mini-Outline

- Broadly, game balance includes:
  - Player-Player
  - Player-Gameplay
  - Gameplay-Gameplay (next)

Gameplay/Gameplay Balance: Introduction (1 of 2)

- □ Consider Warcraft 2, with dozens of units.
  - ■ Nearly perfectly balanced
    - □ No unit costs so much that you don’t want it
    - □ No unit is too weak that you can do without it
- □ Either they got lucky *or* did **lots** of play testing (probably the latter)
- □ Strong Rock-Paper-Scissors relationship
  - ■ Have to play all units, none are dispensable

Gameplay/Gameplay Balance: Introduction (2 of 2)

- □ Challenges when balancing aspects of gameplay?
  - ■ You want a variety of interesting choices, rather than a single, dominant choice
  - ■ The best choices depend upon choices of other players (or on AI)
  - ■ As a designer, it's not easy to see how frequently different choices will be worth making, but you need to know this to balance game
- □ Sounds like catch-22? Can use simple concepts to make first guess
  - ■ Then lots of play testing to fine tune! ☺

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Game Balance (1 of 3)

- Establish the value of each game choice
- For game balance, each choice must
  - not be reducible to a simple value (otherwise, it's easy to determine if it dominates or is dominated

    or
  - factors must even out
- Example where things even out: Pirate game
  - Dreadnoughts > Galleons > Brigantines
  - All have identical functions
  - If Dreadnoughts are 2x more power, then (for balance) Galleons should take ½ time to spawn, so will have 2 Galleons for each Dreadnought

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Game Balance (2 of 3)

- Example where it doesn't even out: *Starcraft*
  - Mutalisks fly over any terrain, but cannot fight other fliers
  - Wraiths are not as tough, but can attack other fliers
  - Observers can see enemy, but not fight
  - → There is no expression for values since different things!
- Another example, in the Pirate game
  - Instead of spawn rate, compensate by making Dreadnoughts slowest, Brigantines fastest
  - → Getting more interesting gameplay, but what about balance?

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Game Balance (3 of 3)

- □ Use weights to get an average set combining all factors based on perceived importance
- □ Then, adjust component values so all units are useful
  - ■ How to adjust? Lots of play testing!
- □ Often need tools so level designers can balance

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Combinatorial Explosions

- □ How many components should there be to make the game interesting?
  - ■ Too few? Then becomes trivial (Ex: in Hastings, only way to change power base is to put infantry on hill)
  - ■ Too many? Then too hard to have skilled play
- □ Rule of thumb: N factors that could modify core mechanics, and each is boolean (hill or not, rain or not, ...) → 2^N possible combinations...explodes rapidly
  - ■ Remember, N=24 gives about 16 million combinations!
  - ■ Err on the side of caution

"In Populous (EA god-game), should have lots of characters or half-dozen? Noticed would be easier to understand game experience with few, versatile units rather than many specific ones."
Richard Leinfellner (executive in charge of *Bullfrog*)

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

Design Scalability

- Intransitive designs are inflexible
  - If there is a balanced relationship and we remove one, we will have a dominant strategy
  - Ex: RPS and remove R...always choose S!
- If project lead says you are behind schedule, so don't include 5th orc type,

  → Elegant design falls like a house of cards!
- But it is relatively easy to add components
  - Doesn't have to be symmetrical, can be redundant or useful in only a few cases
    - Ex: scout, or special spell
- Lesson
  - If you are going to scale, scale ***up*** not ***down***

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

A Game Balance Checklist (1 of 3)

- *Player-Player*
  - Ensures game is fair
  - Especially important for multiplayer games
  - Symmetry works for this, but asymmetry may be needed or more appealing (try "just broken")
  - Make sure any asymmetry doesn't magnify imbalance as game progresses
- *Golden rule*: a player should never be put in an unwinnable situation through no fault of their own

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

A Game Balance Checklist (2 of 3)

- *Player-Gameplay*
  - Ensures player never becomes frustrated
  - Continually brings player back for more
  - Interface should not present obstacles
  - Small rewards are needed to guide player
    - Ex: Fancy animation or new powers
  - The best rewards *widen options*
- *Golden rule*: The game should be fun to learn as well as to play, and it should be *more* fun the more you master it

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris

A Game Balance Checklist (3 of 3)

- *Gameplay-Gameplay*
  - Ensures no element redundant or useless
  - Can do briefly by making factor table for each attribute
    - Make sure each unit is best at something
  - Each component dynamically best, not statically so
  - Oblige player to alter tactics
  - Don't have to have every component equally useful
  - Cost, availability, and ease of use should reflect value
  - Get right through play testing
- *Golden rule*: all options in game must be worth using sometime, net cost of each option must be on par with payoff

Based on Chapter 5, *Game Architecture and Design*, by Rollings and Morris