Balance

Arno Kamphuis

Game Design 2010-2011

Today

- Practical assignment
- What is balance?
- Can it be proved?
- How can it be achieved?
- Different things to balance
  - Player vs. Player
  - Player vs. Game play
  - Game play vs. Game play
- Different types of balance
  - Static and Dynamic balance

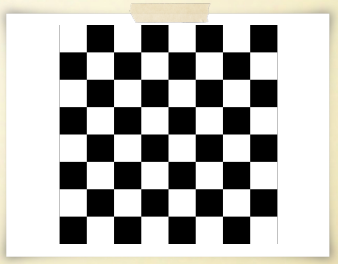
Practical assignment

Remember deadline

Today at 17h00!!

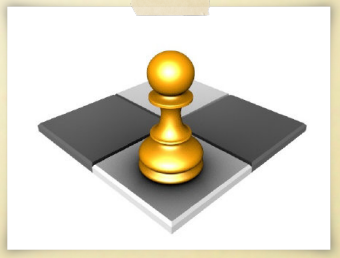
Design a board game

- Bounds of the task:
  - Checker board 8 by 8
  - Pawns
  - Dice (simple 6-sided)
- Create a rule set
- Determine win/loose situation
- 'Prove' that it is balanced



Deliverable

- Write 3 documents
  - Design document (max. 2000 words)
  - 1-page rule book (max. 500 words)
  - 2-page player manual (max. 750 words)



What is balance?

- Hard to describe
- No clear definition
- Depends on many factor, including
  - The player !!!
- The book states

"A balanced game is one where the main determining factor for the success of the player is the skill level of that player."

Other definitions

"Game balance is a concept describing fairness or balance of power in a game between multiple players or strategic options."

"When players have multiple options or routes to victory, most or all of these options are about equally effective and/or feasible."

What to balance?

- Are you balancing against:
  - The player
  - The game play
- Are you balancing it so that it
  - is a 'fair' game?
  - provides a consistent experience to the player no matter what his/her ability?

How to achieve balance?

- Traditionally (and still most often used)

  "The game is played, the game is tweaked, the game is played, the game is tweaked, and then finally, when time runs out, the game is released."

- Theoretically difficult problem
  - n-dimensional with huge n
  - Game theory can only solve small problems

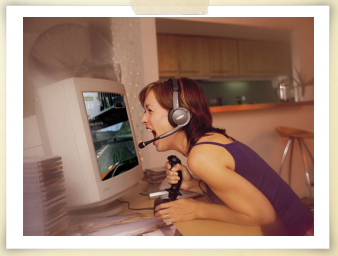
Analysis in different ways

- Classes of balance
  - Static balance
  - Dynamic balance
- Types of balance
  - Player vs. player
  - Player vs. game play
  - Game play vs. game play

Types of balance

- Player vs. Player
  - Players must have an equal chance of winning
- Player vs. Game play
  - The game play should help the player play the game
- Game play vs. Game play
  - Features must be balanced against each other
  - Each feature must be useful in certain situations

Player vs. Player



Player vs. Player

- Each player must have an equal a priori chance of winning
  - Even when using different equipment
- No special advantages, only skill counts
  - Use leagues when there are different types of players
  - Handicap in golf

Player vs. Player

- Luck
  - There can be luck, but it must be equally distributed over the players
  - Luck should never be too crucial
  - Bad examples:
    - One player starts close to resources or in a better protected part of the game world
    - There is one special weapon and the first person to stumble upon it has a huge advantage
- Also relevant for computer players
  - Perceived fairness

Player vs. Player...But how?

Player vs. Player

- Exact symmetry
- Broken symmetry
- Complete asymmetry

Exact symmetry

- Like in chess
- A bit boring
- Looks unrealistic
- Still often not really balanced because of player preferences

Broken symmetry

- Even though the features look different, they are effectively the same
  - functional symmetry
  - river versus mountain range (with ships versus tunneling equipment)
- Just broken symmetry is relative easy to balance and often aesthetic appealing
  - Different units but similar effects

Complete asymmetry

- Create completely different units/features/rules for the different players
- It is almost impossible to balance
  - Make sure it does not show up during the lifetime of the game (impossible)
  - Avoid the need to patch later (almost impossible)
  - Much, much testing is required!!

Remember Player Preferences

- Do not force one playing style on players
- A feature is only good in the hands of somebody who can use it
- There can be differences in the effectiveness of features for different players
- These should preferably be matched by other inconveniences (price, learning curve, ...)

Player vs. Game play

Player vs. Game play

- The most important aspect of the game is the player!
  - Keep him in mind when designing the game
  - The player should have fun
- If the player loses it must be his own fault, not the systems fault
  - It matters here what the player thinks, not what is actually the case

Game play is not the opponent

- The *game play* should not be considered the opponent
  - Like not being able to find the words in a text adventure
  - Or not being able to remember the right keystrokes
  - Or not being able to get the right view on what is happening
  - Or not being able to find the correct pixel to click on

Play the game

- You should play WITH a game, not AGAINST it
  - No progress by trial-and-error
    - No invisible death traps
  - Give clues on what to do (but they might be concealed)
  - Give hints on what the effects of certain actions are
  - Avoid stagnation
- A game should be fun to learn and fun to play

Save a Game?

- When can the player save the game
  - Any moment
  - At certain points (how far apart)
  - A limited number of times
  - Too few leads to frustration
  - Too many leads to trial and error
- How many different save slots

The curse of Save Game

- Saving has become a game play feature
  - Trial and error
  - To mask bad game design
  - Can be solved by better balancing and better level design

Game play vs. Game play

Game play vs. Game play

- All game play aspects must be balanced with respect to each other
  - Rules
  - Features
  - Level design
  - We will mainly discuss features here
- No dominant strategies
  - Keeps game play interesting
- All features should be useful
  - Leads to enough variation

Transitive Relationships

- A beats B beats C beats ...
- Examples:
  - Weapons with different strengths
  - Power ups with increasing power
- Easy to design
- Easily leads to dominant strategies
- Must be balanced by indirect (shadow) costs

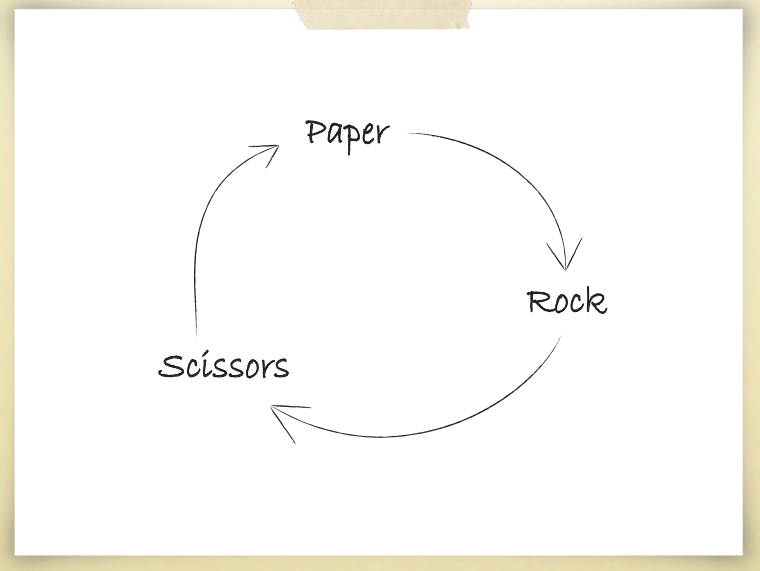
Shadow Costs

- Features in a game have shadow costs
- Costs that are not directly related to the use
- Examples:
  - Feature can be used a limited number of times
  - Feature can only be used once in a while
  - Feature costs more money to produce
  - Feature appears only later in the game
  - Feature is only available by making moves that are difficult to master
- Shadow costs can be used to balance features or to stimulate certain uses

Cyclic (intransitive) relationships

- Example: Fighting game
- Three moves: Leg sweep, Kick, Stomp
- Rock-Paper-Scissors (RPS) relationship between moves
  - Leg beats Kick
  - Kick beats Stomp
  - Stomp beats Leg
- No move is dominant

Rock Paper Scissors



Cyclic relationships

Net Payoff Matrix

- For each interaction write down the relative benefit (or loss, or cost)
- Simple: +1, -1, 0

Cyclic relationships

	Leg	Kick	Stomp
Leg	0	+1	-1
Kick	-1	0	+1
Stomp	+1	-1	0

Cyclic relationships

Disadvantage: there is a simple optimal strategy:

- Randomly choose a move
- No other strategy can beat this
- Not much fun

Cyclic relationships

We can add costs

- Each move costs energy: Leg 1, Kick 2, Stomp 3
- When you lose an exchange it costs 5 energy
- You can again make the relative benefit matrix, e.g. Leg-Kick give -1+2+5 = 6
- It costs:
  - You lose 1 for the move
  - The opponent loses 2 for the move
  - The opponent loses 5 because you win the exchange
- Now what is the optimal strategy?

Cyclic relationships

	Leg	Kick	Stomp
Leg	0	+6	-3
Kick	-6	0	+6
Stomp	+3	-6	0

Cyclic relationships

- Let l, k, s be the frequency of the moves
- Let L, K, S be the net payoffs of the different moves
- $L + K + S = 0$, that is: zero sum game, both players use same strategy so nobody wins
- $L = K = S = 0$, otherwise you would simply increase the frequency of the move with higher net payoff (so your strategy was not optimal)

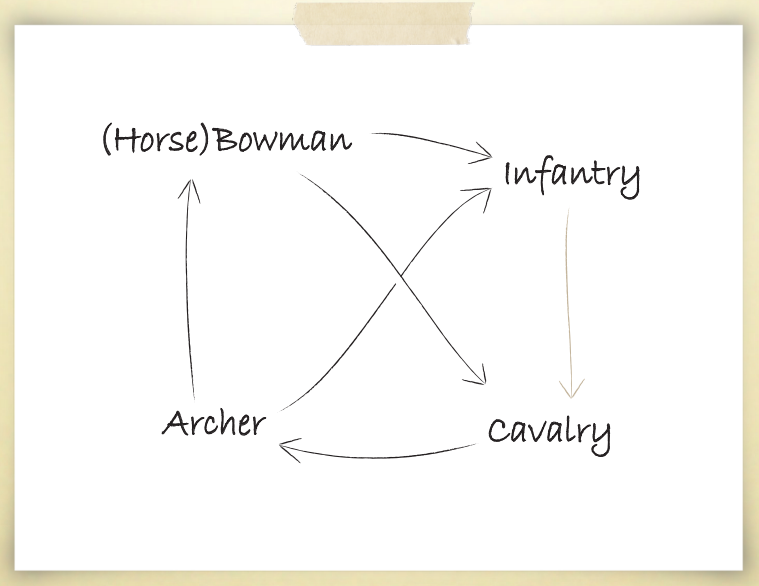
Cyclic relationships

- From the table it follows that:
  - $L = 6k - 3s$
  - $K = 6s - 6l$
  - $S = 3l - 6k$
- This leads to $l : k : s = 2 : 1 : 2$
  - 40% Leg sweeps
  - 20% Kicks
  - 40% Stomps
- You should not do anything predictable so random with these average frequencies

Cyclic relationships

- There is one optimal strategy which is not too much fun
- Better:
  - Multiple optimal strategies
  - Strategies that are difficult to find
  - Learning and anticipating what your opponent will do (e.g. through clues)

More attributes



More attributes

	B	I	C	A
bowmen (B)	0	+1	+1	-1
infantry (I)	-1	0	+1	-1
cavalry (C)	-1	-1	0	+1
archer (A)	+1	+1	-1	0

More attributes

- The relations should make sense in the game such that the user naturally understands them)
- There are multiple optimal solutions (that is good)
- You can win without bowman or infantry (which is bad)
- So change the shadow costs
  - make them cheaper to create
  - make them move faster
  - make them appear later

What to do?

- Best choice should depend on earlier choices
  - A fast weak ship needs other weapons than a slow strong ship (if you designed the right opponents)
- How to find out what is good you need to
  - Draw payoff diagrams
  - Try to compute some information about strategies
  - Make a special testing program in which you can easily change the parameters
  - Test, test, test!!!!!!!!!!!!
- All options in the game should be worth using sometimes and the cost must be in balance with the payoff you get when using it

Dynamic Balancing

- Balance the game during game play
- Change the parameters of the game to continuously provide interesting play
- Examples:
  - The effect of a weapon wears off when used too often
  - Enemies get more powerful or clever if you beat them too easy
  - Cars adapt their speed to the players speed
- Should be as invisible as possible

Emergence

- Emergence occurs when a set of simple rules leads to (possibly unexpected) complex results
- Emergence can lead to very satisfying game play
- Emergence can easily break the balance in a game

Feedback

- During a game the balance between players changes
- **No feedback**: Being ahead does not influence your abilities
  - Examples are many sport games
  - Very balanced
- **Positive feedback**: Being ahead gives you advantages
  - Examples:
    - Chess
    - Car racing
    - Many other games
  - Rather natural
  - Can lead to frustrating game play
  - Limit the amount of positive feedback

Feedback

- **Negative feedback**: Being ahead gives you disadvantages
  - Examples:
    - Maintaining a bigger army costs more resources
    - You have to feed enemy prisoners
    - 8-ball pool: The more you get ahead the less choice options you have
  - Can lead to longer, interesting game play
    - Also against slightly weaker opponents
  - The game might never finish
- Introduce randomness
  - Keeps a game interesting even for weaker opponents

Next time

Storyline

Chapter 4 of the book