Design of computer video games

7. Game balancing

Boyan Bontchev

Agenda

- Introduction to game balancing - definitions and basic concepts
- Well-balanced games of PvP type
- Well-balanced games of PvE
- Avoiding dominant strategies
- Using chance for rewarding skillful play
- Keeping the player in the *flow* state of peak enjoyment
- Using positive feedback and/or combined feedback
- Examples

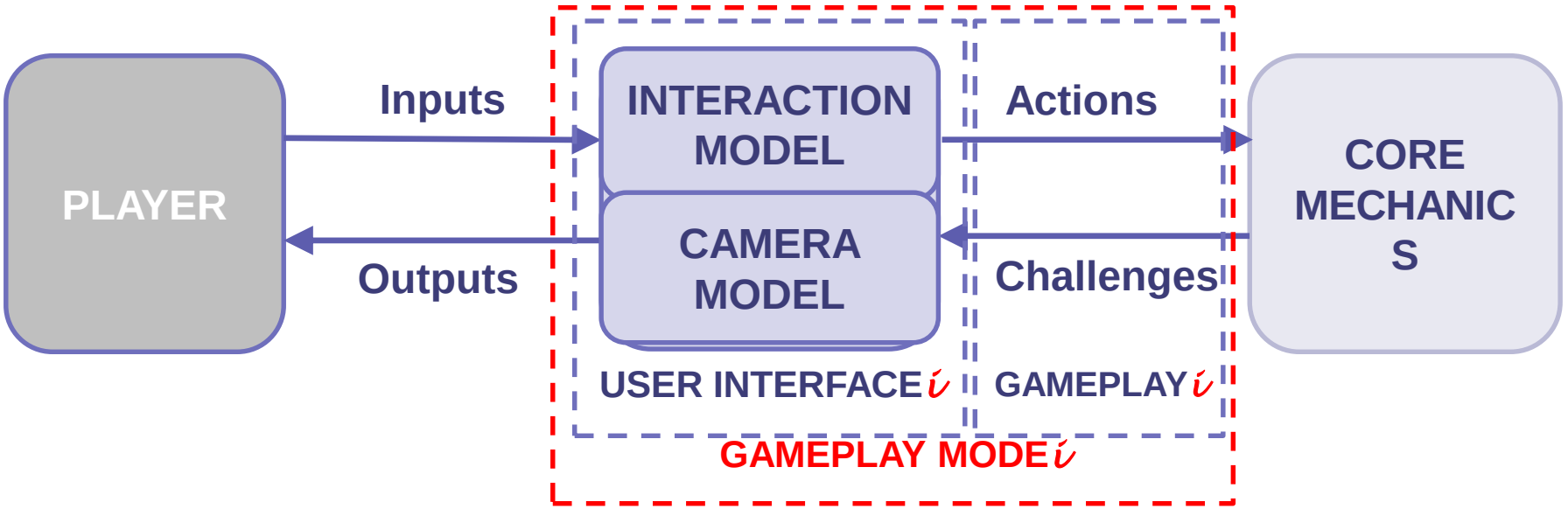
References

- Adams, E. Fundamentals of Game Design, Third Edition, Pearson Education, Inc., ISBN-13: 978-0-321-92967-9, 2014
- Salen, K., Zimmerman, E . Rules of Play - Game Design Fundamentals, MIT Press Cambridge, Massachusetts London, England, ISBN 0-262-24045-9, 2004.
- Fullerton, T., Swain C., Hoffman, S. S. Game Design Workshop: A Playcentric Approach to Creating Innovative Games, Second Edition, Elsevier, ISBN 978-0-240-80974-8, 2008.
- Adams, E., Dormans, J. Game Mechanics: Advanced Game Design, New Riders Games, ISBN-13: 978-0-321-82027-3, 2012.
- Smith, M., Queiroz, C. Unity 4.x Cookbook, Packt Publishing, Birmingham B3 2PB, UK, ISBN 978-1-84969-042-3, 2013.
- Other references are cited in slides

From Module 5

What is the first designer job?

- RULE: First job is to design the *primary* gameplay mode



- Only AFTER the *primary* gameplay mode – details of its UI
- Gameplay modes *typically* share a number of UI features, so it's best to define all the modes before you begin UI

Better gameplay – true fun

From Module 6

- **Gameplay comes first**
- **Get a feature right or leave it out -** it is far worse to ship a game with a broken feature than to ship a game with a missing feature
- **Design around the player** - you must examine *every* decision from the player's point of view
- **Know your target audience** - different groups of players want different things - for any group, *know* what they want and what they think is fun, and then provide it
- **Abstract or automate parts of the simulation that *aren't* fun**
- **Strive for harmony, elegance, and beauty**

What is game balancing?

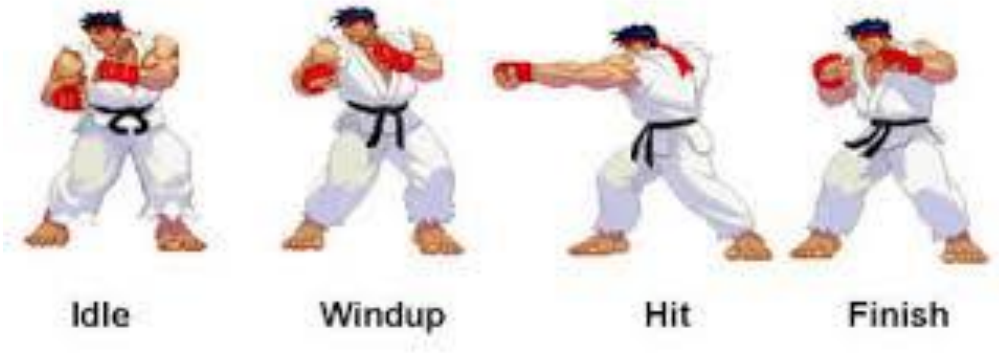
- A balanced game:
  - fair to the player (or players)
    - neither too easy
    - nor too hard
  - makes the skill of the player the most important factor in determining his/her success
- Depends upon game type:
  - Player versus Player(s) (PvP) – against one or more opponents
  - Player versus Environment (game world) (PvE)

Features of a well-balanced game (both PvP and PvE)

- **The game provides meaningful choices –**
  - if the player chooses one from several possible strategies for solving the game's challenges, no strategy should be *dominant* i.e. much more effective than the others – otherwise he/she will never try less effective ones
  - if such a *dominant strategy* exists, it indicates a poorly balanced game
- **The role of chance is not so great that player skill becomes irrelevant** - a better player should be *statistically* more successful than a poor one

Features of a well-balanced game (only PvP) 1/2

- **The players perceive the game to be fair** - the exact meaning of fairness varies among different players



- **Any player who falls behind early in the game gets a reasonable opportunity to catch up again before the game ends** - *early in the game* and *a reasonable opportunity* vary depend on how long the designer expects a game to last

Features of a well-balanced game (only PvP) 2/2

- **The game seldom or never results in a stalemate,** particularly among players of unequal ability:
  - a stalemate disappoints players as their efforts produce no victory
  - if stalemates occur frequently *among players of unequal ability*, the game violates the principle that player skill should influence the outcome more than any other factor.
- The player perceives the game to be fair:
  - in a PvE game, the player's perception of fairness involves a number of factors and
  - is complicated by the absence of an opponent

- The game's level of difficulty must be consistent - the perceived difficulty may be low or high but should not change suddenly, especially within a single game level

Dominant strategy 1/2

- *Strategy* - a plan for playing a game, usually according to a principle or approach that the player believes is likely to produce success
  - aggressive
  - defensive
- *Dominant strategy* - a strategy that reliably produces the best outcome a player may achieve, no matter what his/her opponent does
  - undesirable because once a player discovers it, he/she never has any reason to use any other strategy
  - makes all other choices pointless and thus *limits the fun* the player can have with such a game

Dominant strategy 2/2

- *Dominant strategy* - a strategy that reliably produces the best outcome a player may achieve, no matter what his/her opponent does
  - doesn't mean that the player who uses it always dominates her opponent - he/she can still lose through bad luck
  - dominant strategy is even worse in asymmetric games - one player may use it but another player may not!
  - strategies that avoid loss or prevent an opponent from scoring points can also qualify as dominant
    - e.g., using endless delaying tactics for a limited time to preserve their lead

Example of dominant strategy

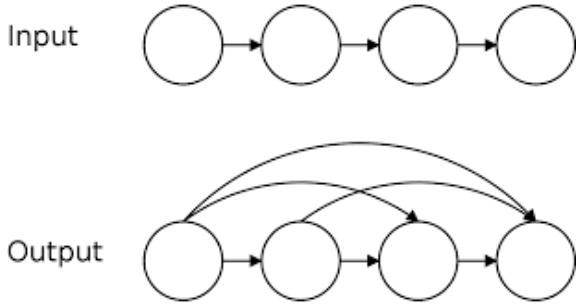
The best-known dominant strategy in any PvP video game is the tank rush in Westwood's *Command & Conquer: Red Alert.*

- An experienced player playing as the Soviet side can produce many tanks in the early part of the game,
- then use those tanks to attack massively the unprepared enemy
- This almost always produces a victory.

Discussion

- Pro’s and con’s of dominant strategies?
- How dominant strategies do appear - badly designed characters can also result in dominant strategies?
- How to prevent dominant strategies?

Transitive relationships among player's options

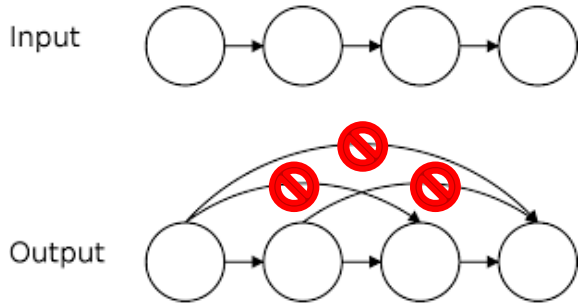


- Transitive relationships
- In games:
  - If option A is better than option B, and option B is better than option C,
  - then players will never use option C
  - (option A becomes a dominant strategy)
- Solution: to correct this imbalance, you may impose ***direct costs*** on using each strategy and so give players a reason to consider the (formerly) weaker strategies as well. Example:
  - option 1: BMW – the fastest, the most expensive
  - option 2: VW – the modest in terms of speed and price
  - option 3: Dachia – the slowest, the cheapest
  - You can use transitive relationships to upgrade a player's powers during his/her progress through the game (let start with Dachia…)

Direct costs vs *shadow costs*

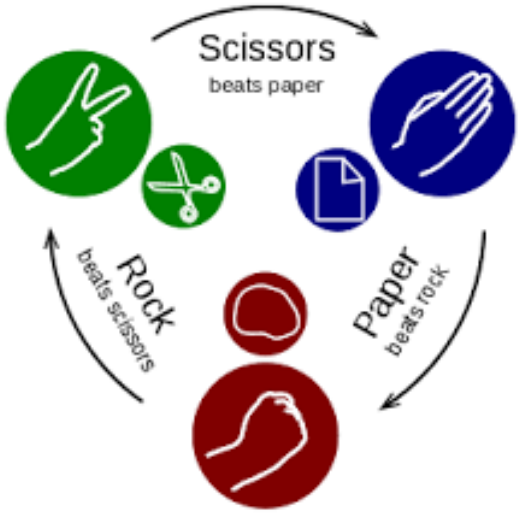
- Setting up ***direct costs*** that exactly counter the advantages of certain choices does balance the game, but
- such a clear and obvious balancing mechanism produces a rather simple game
- To create a more interesting choice for the player, you can instead impose shadow costs.
  - ***Shadow costs*** are secondary, or hidden, costs that lie behind the apparent costs of goods or services.
  - Example:
    - option 1: BMW – the fastest, the most expensive *BUT* with a smaller fuel tank that requires the car to stop to refuel more often in the road race
    - option 2: VW – the modest in terms of speed and price (and fuel tank)
    - option 3: Dachia – the slowest, the cheapest (but with the greatest fuel tank)
  - You can hide a shadow cost completely by building it into the mechanics and not documenting it in the game's manual

Intransitive relationships among player's options



- Intransitive relationships

- Example - rock-paper-scissors (RPS or Rochambeau)



- Result: a balanced, three-way intransitive relationship

Orthogonal unit differentiation

- Units should not only differ in the magnitude of their power at performing one task, as BMW and VW did, but also should display entirely different qualities
- Orthogonality in games:
  - ideally, each type of unit a player can control/command in a game should be orthogonally different from all the others
  - the more diverse the types of challenges in your game, the easier you will find it to create orthogonally differentiated unit types
- Orthogonally differentiated units also help to prevent dominant strategies from arising if you define the victory condition in such a way that the player must use a variety of different units in order to win the game.

Source: Smith, Harvey. Orthogonal Unit Differentiation, Game Developers' Conf., 2003

Using chance in video games 1/2

- **Use chance sparingly (**if chance is not to play a larger role in the game**):**
  - chance should affect only a minority of the player actions that lead to victory and
  - the majority of actions should depend on skill.
- **Use chance in frequent challenges with small risks and rewards** rather than infrequent challenges with large risks and rewards. Example: in poker, chance plays a large role in each hand, but smart players don't bet large amounts

Using chance in video games 2/2

- **Allow the player to choose actions to use the chances (odds) to his advantage**
- **Allow the player to decide how much to risk:**
  - allow the player to choose how much he places at risk at frequent intervals
  - with this choice, you give the player more control over his resources and so tend to reward skill
- **Don’t use chance to determine large issues in the game** unless the player *explicitly* chooses to take large risks

Making PvP games to be *fair*

Players generally consider a PvP game to be *fair* if they believe:

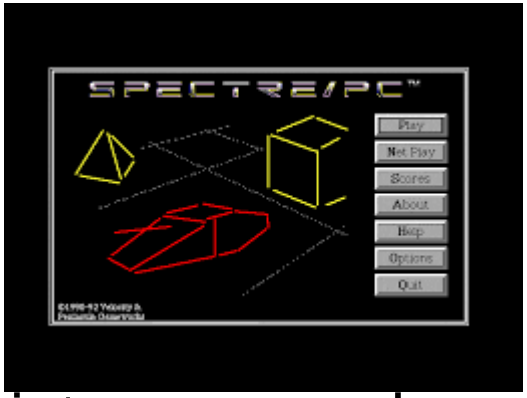
1. the rules give each player an equal chance of winning when play begins and
2. the rules do not give advantage or disadvantage to players unequally during the game in ways that they cannot influence or prevent apart from the operation of chance (in moderation).

Fairness in PvP games

- In **symmetric** PvP games (chess, Monopoly) –
  - each player begins with the same resources, has the same options available to her, faces the same challenges, and tries to achieve the same victory condition
  - *dominant strategies appear seldom* – but in single-player PvP game with a simple artificial opponent, the player can cheat it
  - players hate to discover that the AI is cheating against them (if AI plays by different rules, the game turns to be assymetric)
- In **asymmetric** PvP games (Fox and Geese) –
  - you must test the mechanics for each type of competitor against every other possible type of competitor in order
  - *to make sure that none has a dominant strategy* that confers an advantage over all his opponents.

Point distribution

- The point distribution system:
  - even being fair, it doesn't absolutely guarantee that no dominant strategy will emerge
  - existing risk - one particular combination of features may be superior to any other combination
- Example: *Spectre* allows players to design a tank by assigning points to three attributes:
  - speed
  - armor (defensive strength), and
  - shot power (offensive strength)



- Each player gets the same number of points, so none has a built-in advantage, but each can construct a tank that matches his own preferred style of play

Balanced asymmetric game: *StarCraft*

- *StarCraft* offers the most well-balanced combination of asymmetric features in any war game available, The game offers players a choice of three races:
  - Terrans (or humans)
  - Protoss
  - Zerg

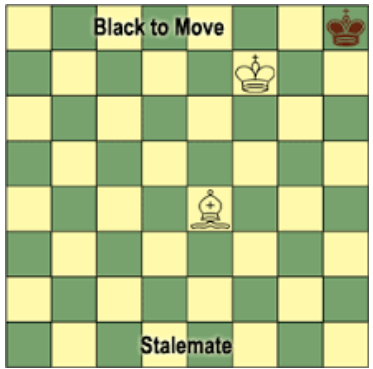
- their costs tend to balance their abilities
- The balance in *StarCraft* makes use of both
  - direct costs (computed from the amount of raw material required to build a unit) and
  - shadow costs — hidden weaknesses

Making PvE games *fair* – 1/2

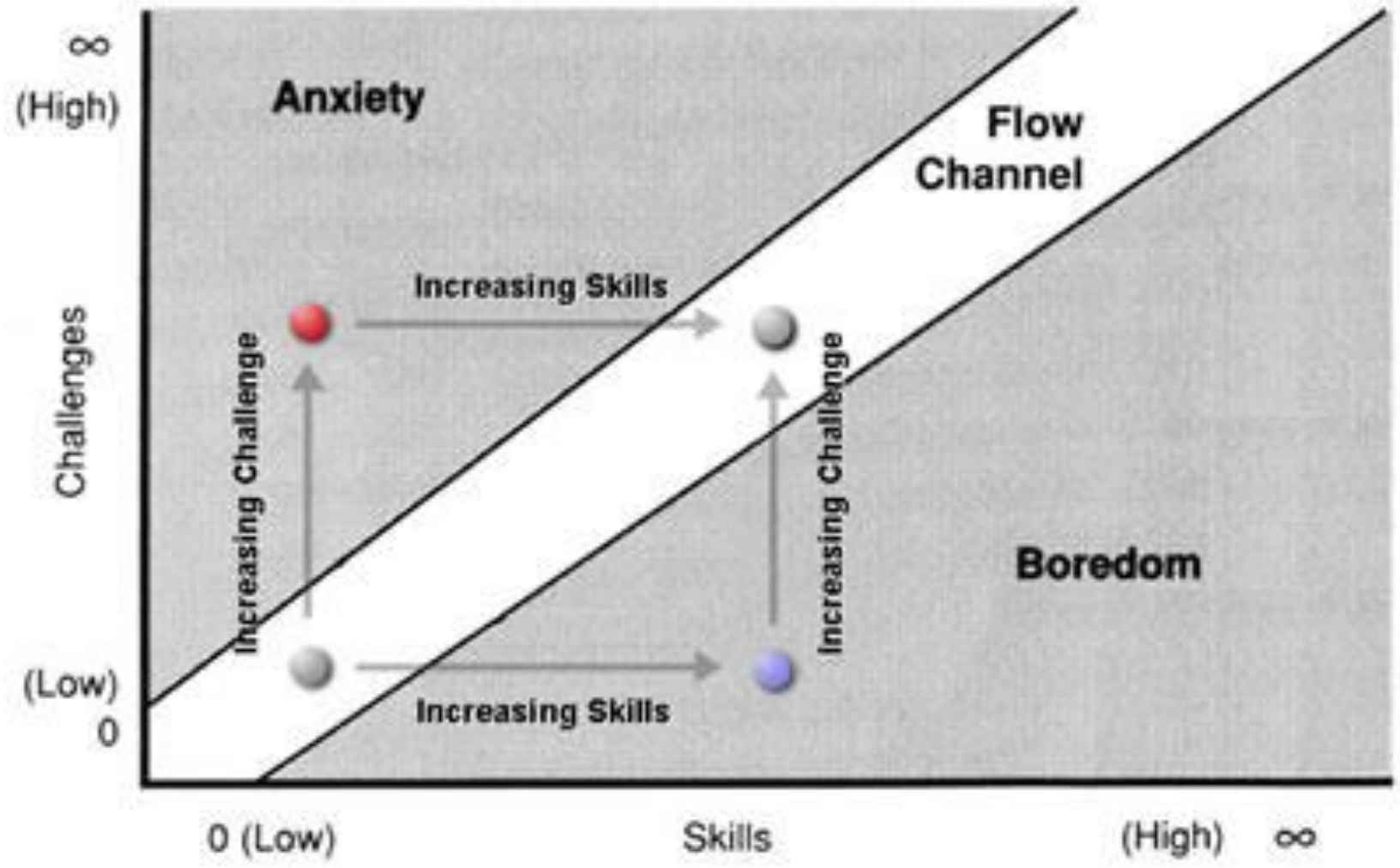
- **The game should offer the player challenges at a consistent maximum level of difficulty,** with no sudden spikes
- **The player should not suddenly lose the game without warning and through no fault of his own**
  - *learn-by-dying* design is considered unfair
  - *The Immortal* (an old video game) requires the player to learn by dying, but allows players to restart the game indefinitely without having to start over at the beginning

Making PvE games *fair* – 2/2

- **A stalemate should not occur** - e.g. when the player is locked in one tunnel and cannot escape
- **The game doesn't ask the player to make critical decisions without adequate Information**
- **All the factual knowledge required to win the game should be contained within the game** (contra-examples?)
- **The game should not require the player to meet challenges not normally presented in the game's genre** (such as a formal logic puzzle in a flight simulator) – in a hybrid genre, you must make this clear before the player starts to play

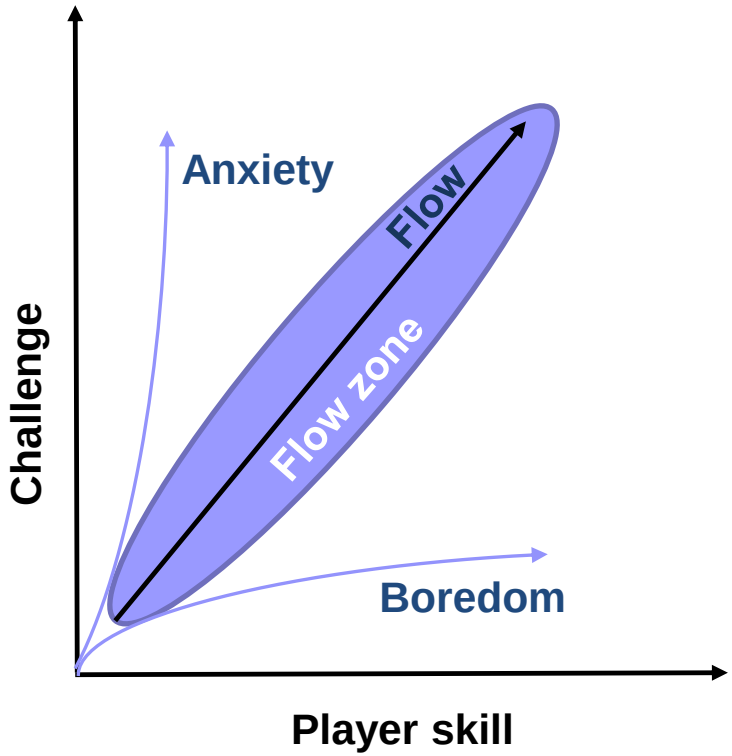


Balance between difficulty and ability

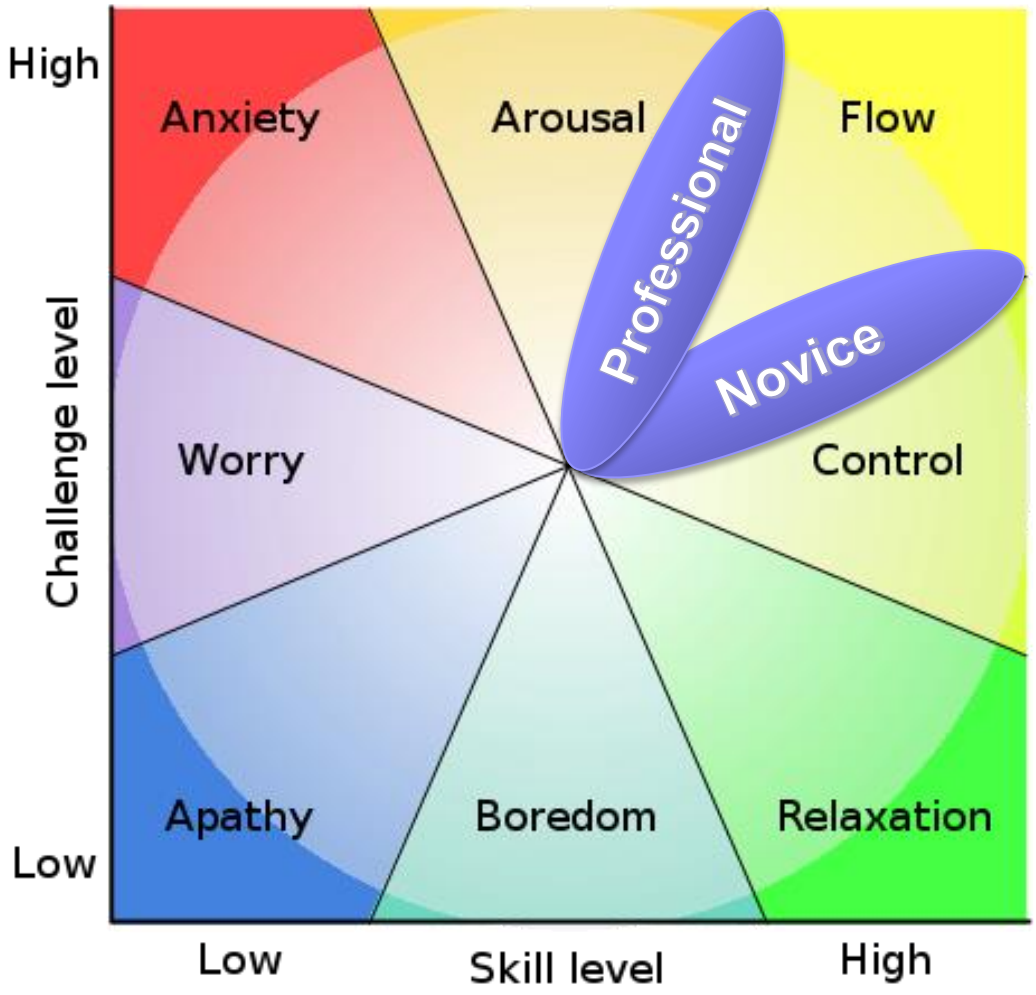


Flow

- Mihaly Csikszentmihalyi (1990) developed the fundamental concept of **flow** as a process of optimal human experience. In digital games - a balance between the inherent challenge of the game activity and the player's ability required for its execution.



Flow and psychic state

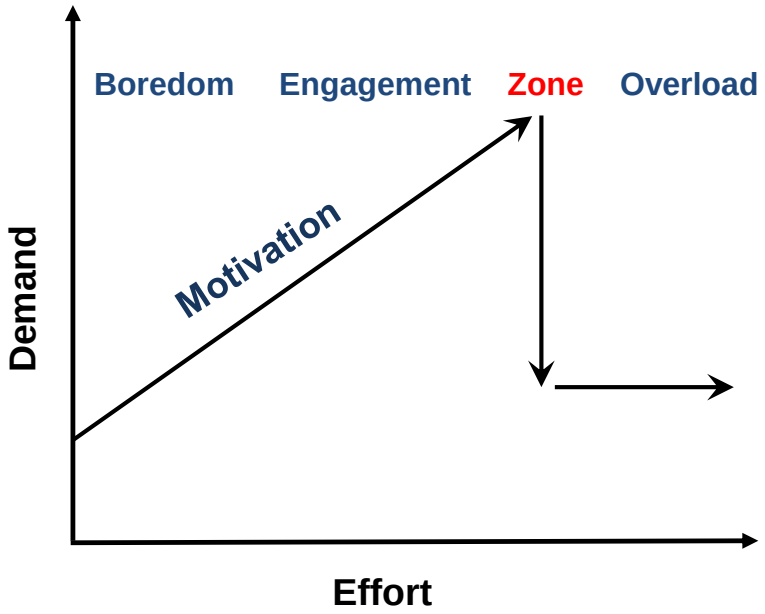


GameFlow

- Based on the flow theory, Gilleade and Dix (2004) stressed the importance of three issues:
  - motivation of the users: why they want to play
  - experience and skills: how able are they to play
  - detection: how to identify when change is necessary
- Sweetser and Wyeth (2005) presented GameFlow model for assessing player enjoyment incl.:
  - concentration
  - challenge
  - skills
  - control
  - clear goals
  - feedback
  - immersion, and
  - social interaction.

Motivational intensity

The dynamic relation between the effort and the demand level was described as motivational intensity (Fairclough and Gilleade, 2012) going through boredom (low effort because of low demand), engagement (rising effort due to increasing demand), zone (peak of achievable effort at the highest level of demand), and overload – low effort due to excessive demand.



Types of difficulty

- ***Absolute*** *difficulty* of a challenge - the amounts of intrinsic skill required to meet the challenges and stress of the challenge *compared to the trivial case*
- ***Relative*** *difficulty* of a challenge - difficulty relative to the player's power to meet that challenge. When level designers build challenges into the game world, they must also take into account the power provided to the player to meet those challenges
- ***Perceived*** *difficulty* of a challenge - the difficulty that the player actually senses, and the type we are most concerned with - consists of the relative difficulty minus the player's experience at meeting such challenges.

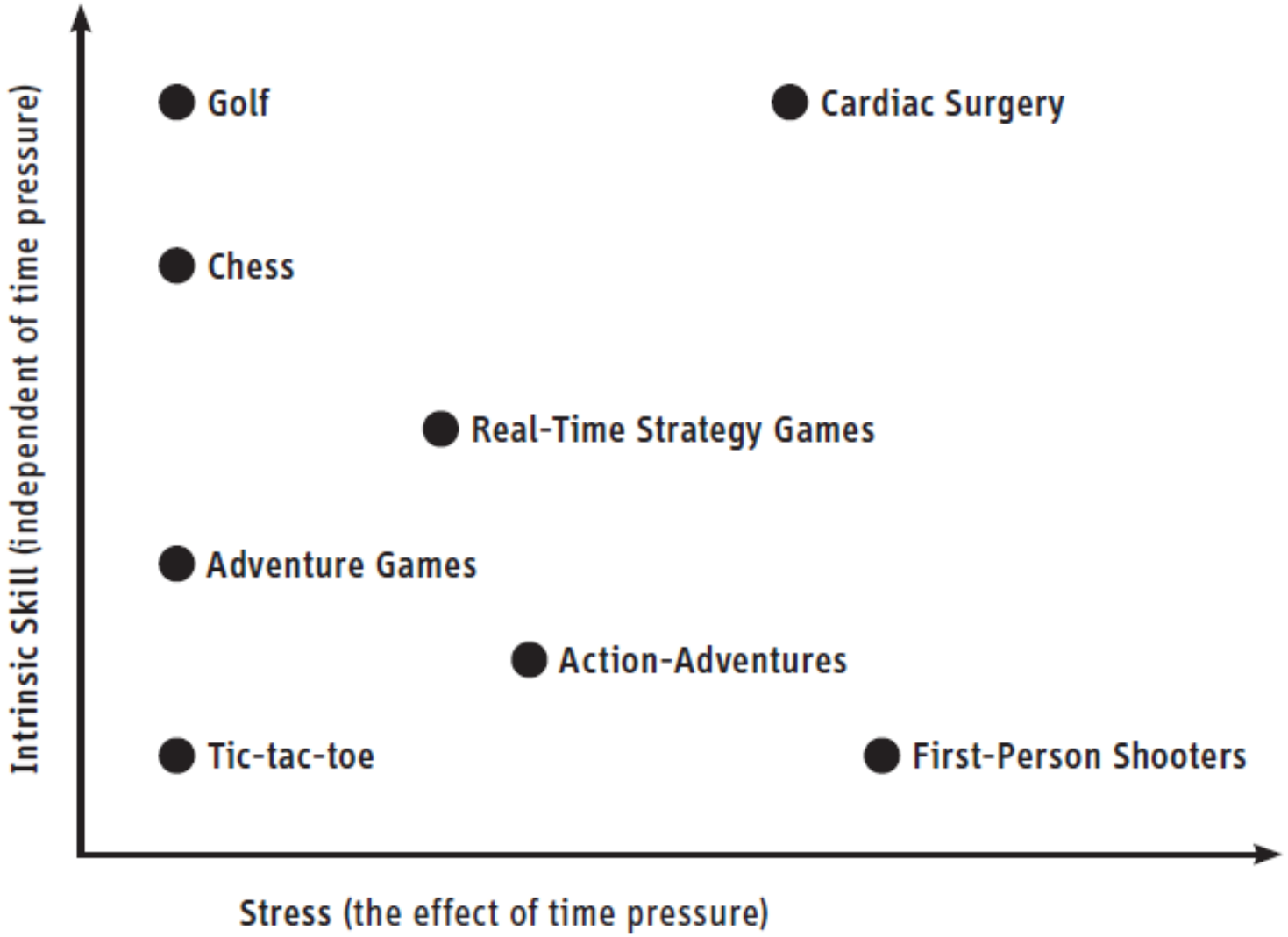
Absolute difficulty depends on:

From Module 6

- *intrinsic skills required* – needed to surmount the challenge *if you give the player an unlimited amount of time to do it. Varies for:*
  - *Physical tasks – archer game, shooting*
  - *Mental tasks – sudoku, trivia*
  - *Combined tasks*
- *stress* - measures how a player perceives the effect of time pressure on his ability to meet a challenge requiring a given level of intrinsic skill. Stress in:
  - Tetris – to complete the task under time pressure
  - Golf - ?

Intrinsic skills vs stress for tasks

From Module 6



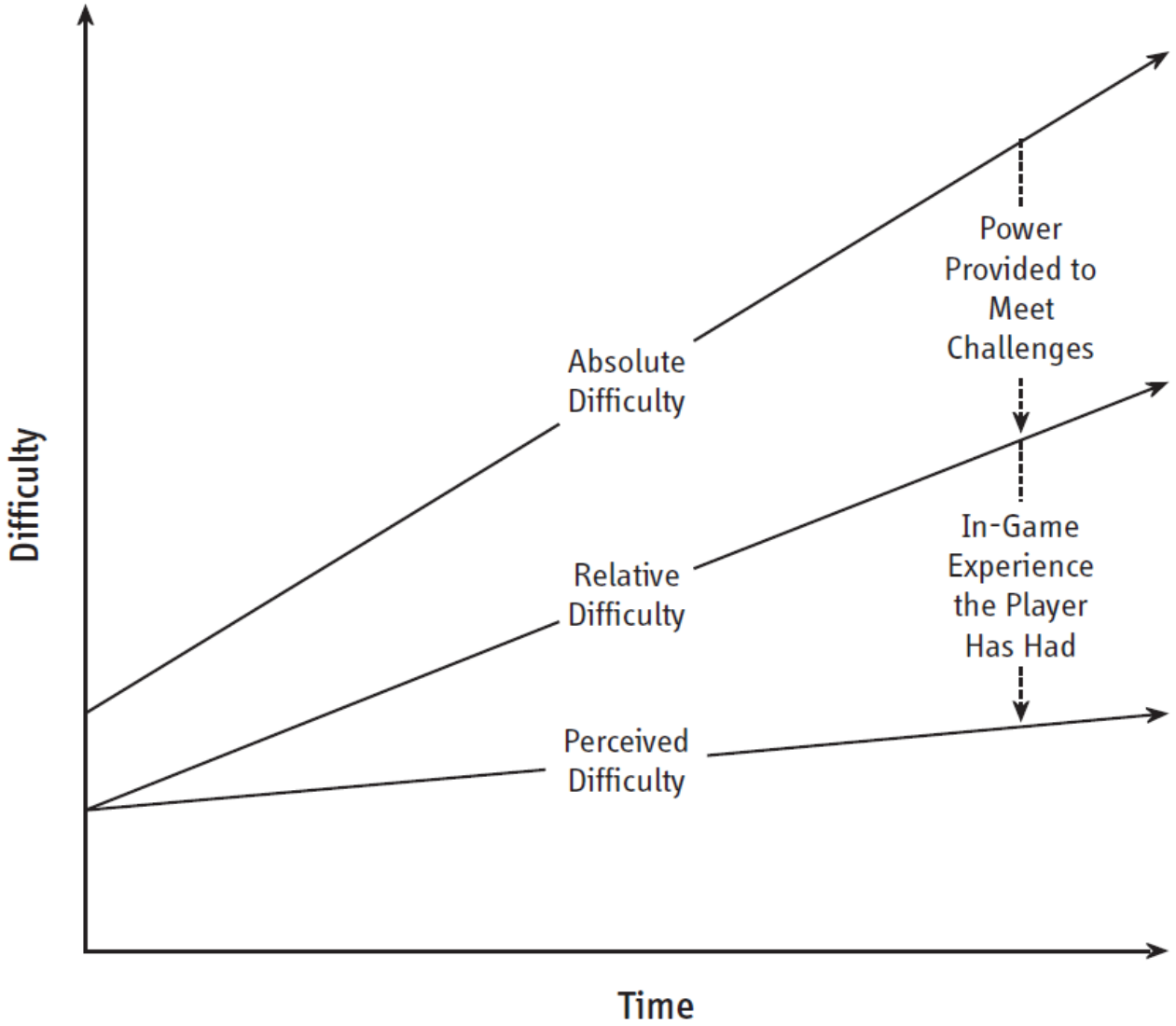
Key factors of perceived difficulty

The designer controls four key factors that create perceived difficulty:

- intrinsic skill required,
- stress,
- power provided by the game, and
- in-game experience.

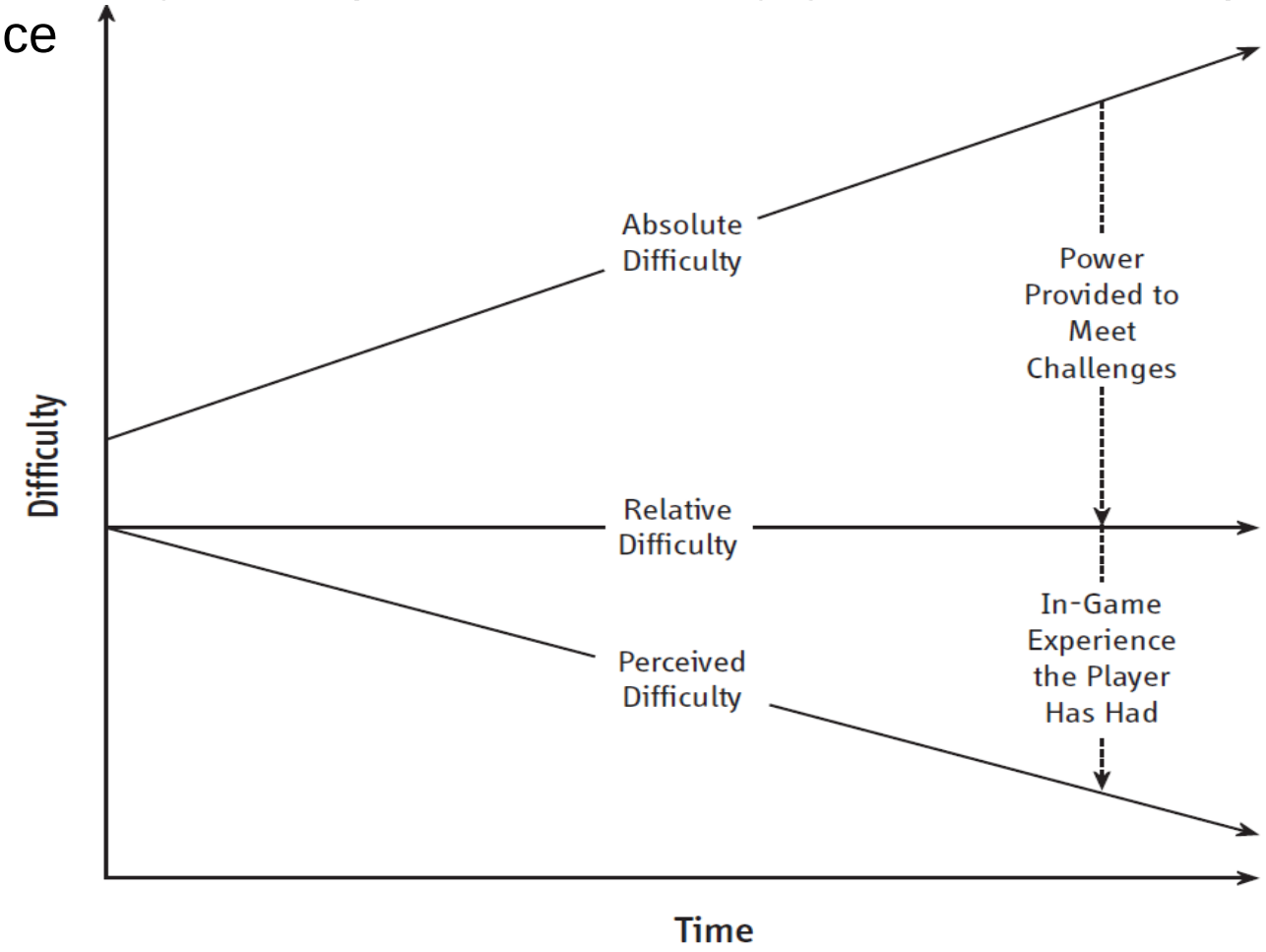
- The major factors the designer cannot control are:
  - previous experience and
  - native talent

Absolute, relative, and perceived levels of difficulty

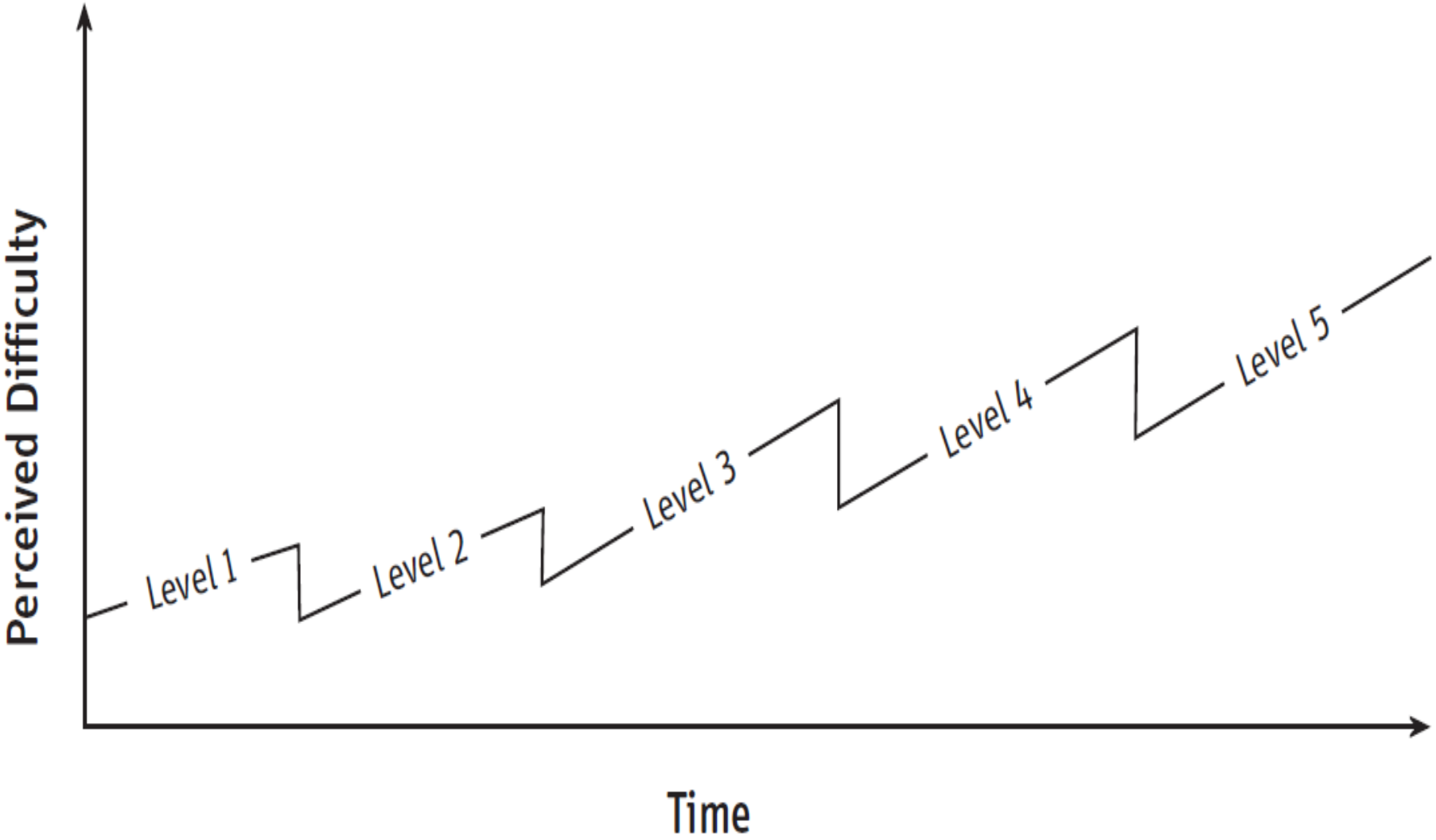


Flat relative difficulty

- If the available power grows at exactly the same rate as the absolute difficulty goes up, the relative difficulty will be a flat line
- When relative difficulty is flat, perceived difficulty goes down as the player gains experience



Perceived difficulty progression across multiple game levels



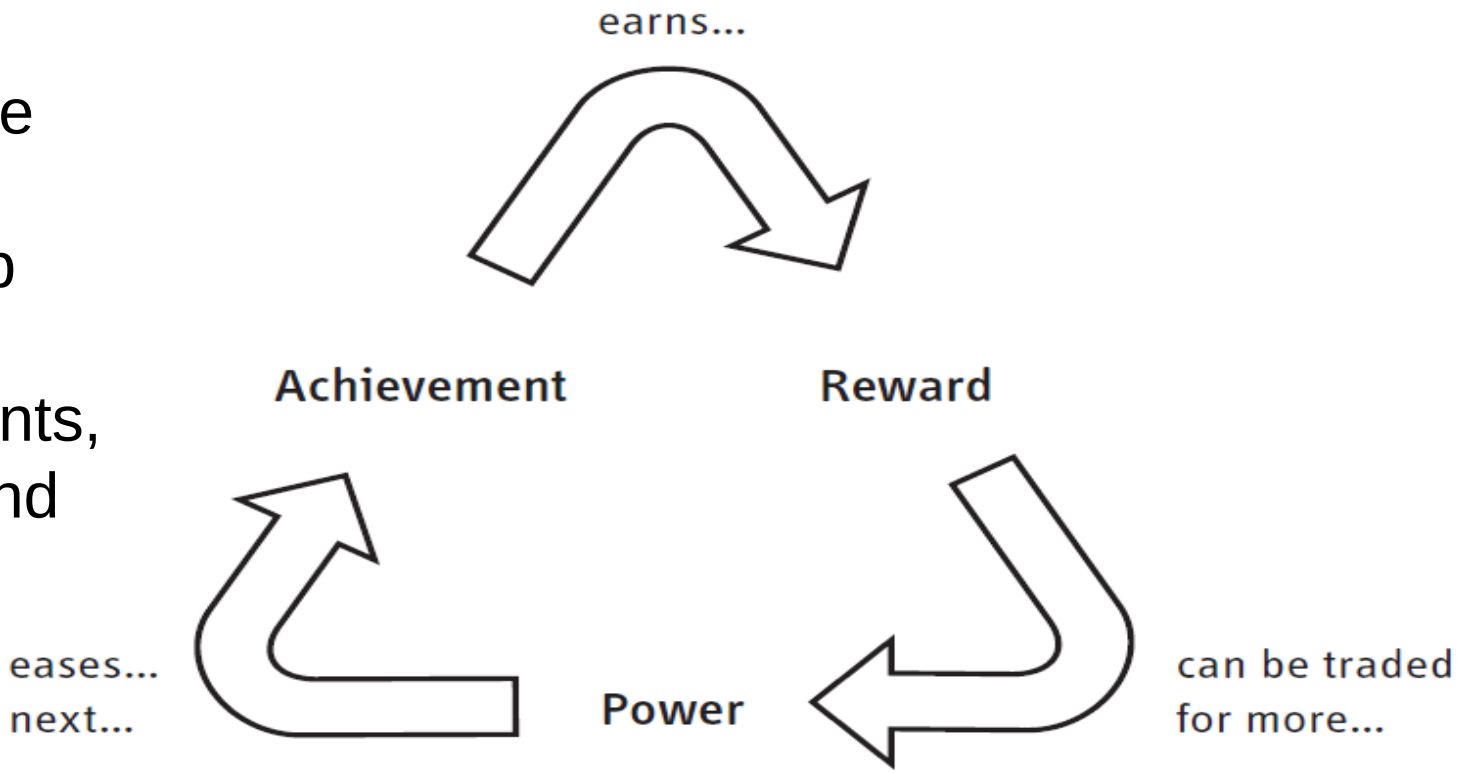
DDA

- DDA (dynamic difficulty adjustment) - adjustment of game difficulty level for balancing challenge with player skills
- DDA is implemented within both the mini-games embedded into the educational maze, as follows:
  - For the 3D matching game - the linear and angular velocity of the flying textual objects (i.e., the quiz answers) is adjusted according the emotional state of the individual player, therefore, changing dynamically the difficulty in reading and selection of quiz answers;
  - For the 3D zoom ordering game - the linear velocity of the flying images is adjusted according the emotional player state. In this way, the game difficulty is dynamically changed, i.e. the difficulty of realizing the correct image which should be placed at a given position.

Positive feedback

Positive feedback occurs when a player's achievement causes changes to the state of the game that make a subsequent achievement easier, which in turn makes further achievements easier still, and so on…

The positive feedback relationship among achievements, rewards, and power



Positive feedback – why and how to

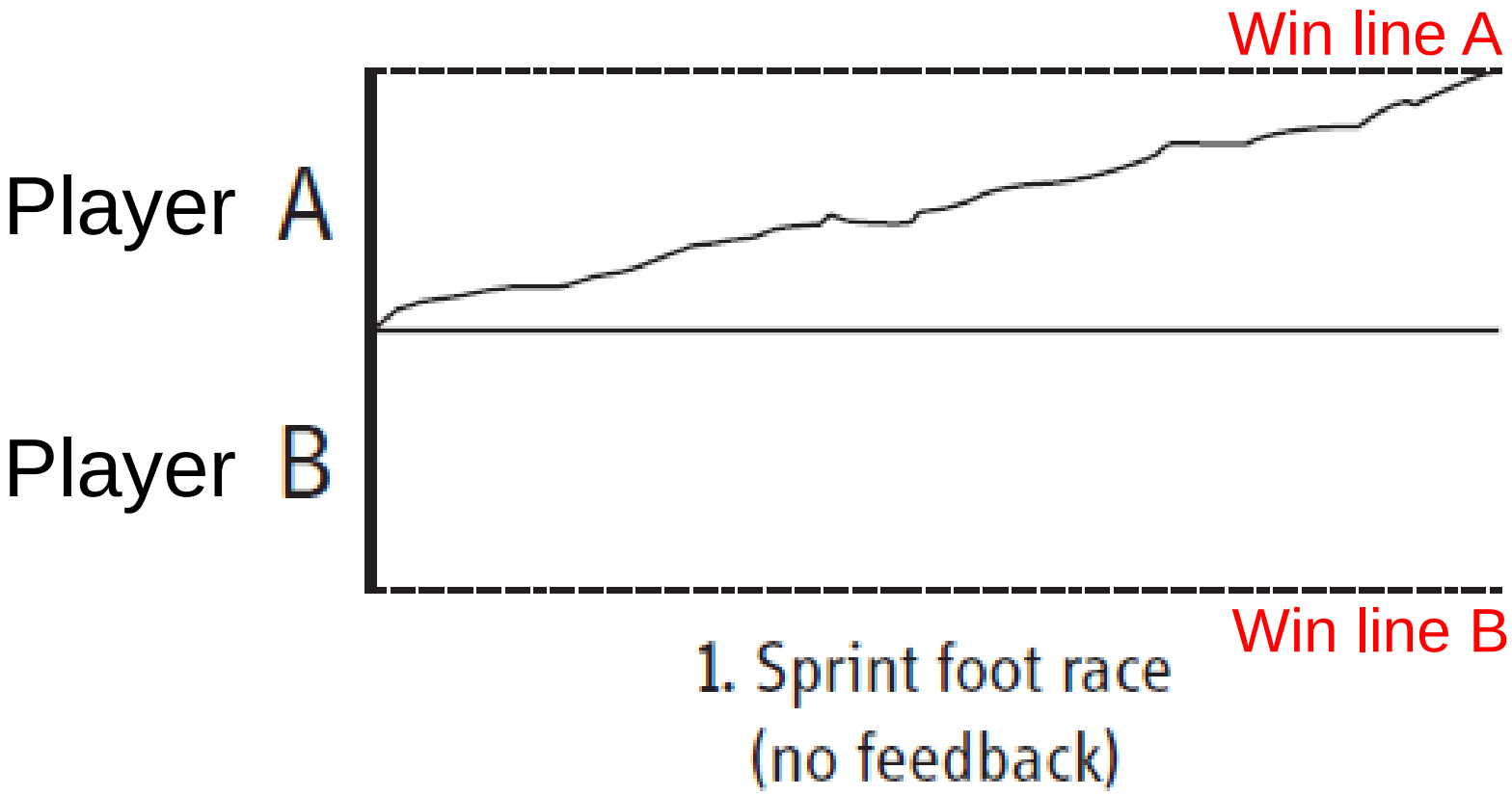
Benefits:

- discourages stalemate
- rewards success

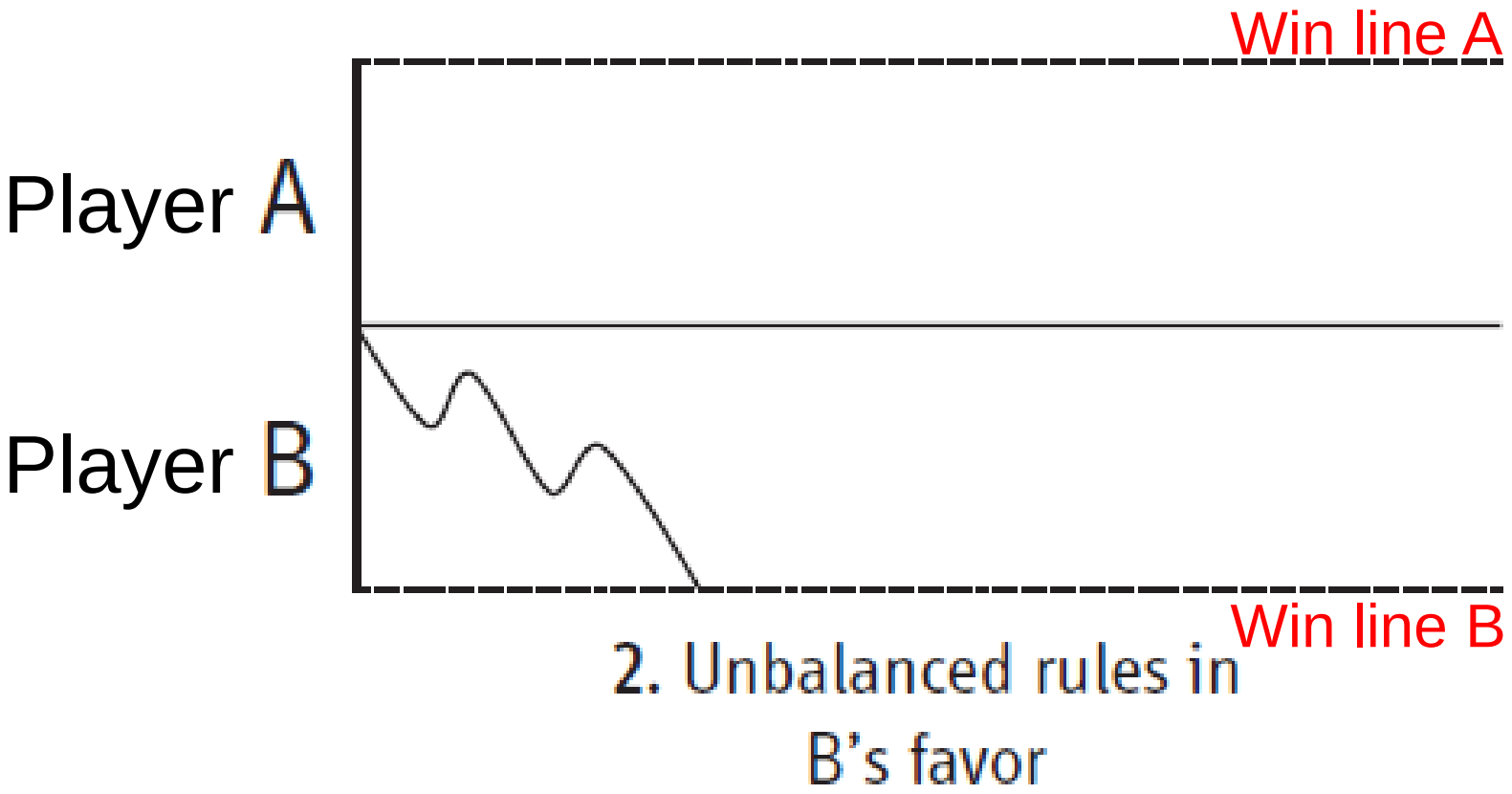
How to control it:

- Don’t provide too much power as a reward for success
- Introduce (some!) negative feedback
- Raise abs difficulty as the player proceeds
- Allow collaborating forces against the leader - *Diplomacy*
- Define victory in terms unrelated to the feedback cycle – like checkmate in chess
- Use the effects of chance to reduce the size of the player’s rewards

Feedback examples 1/6

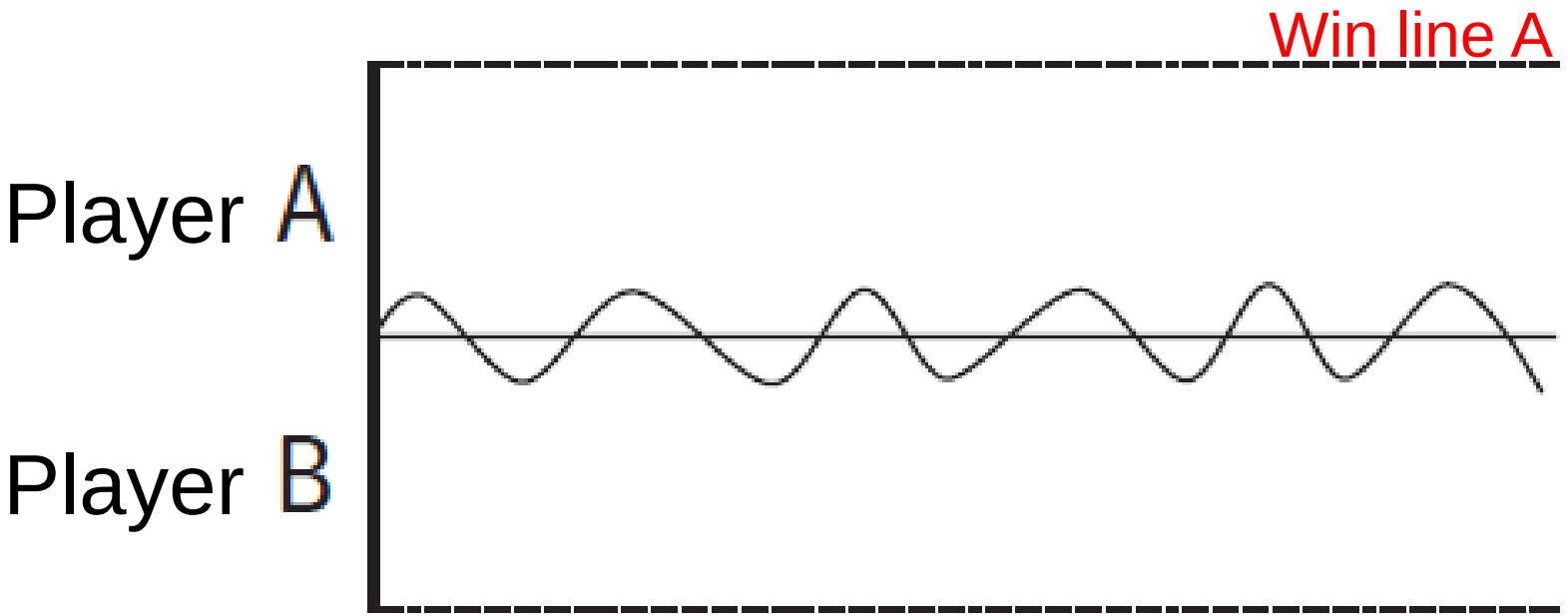


Feedback examples 2/6



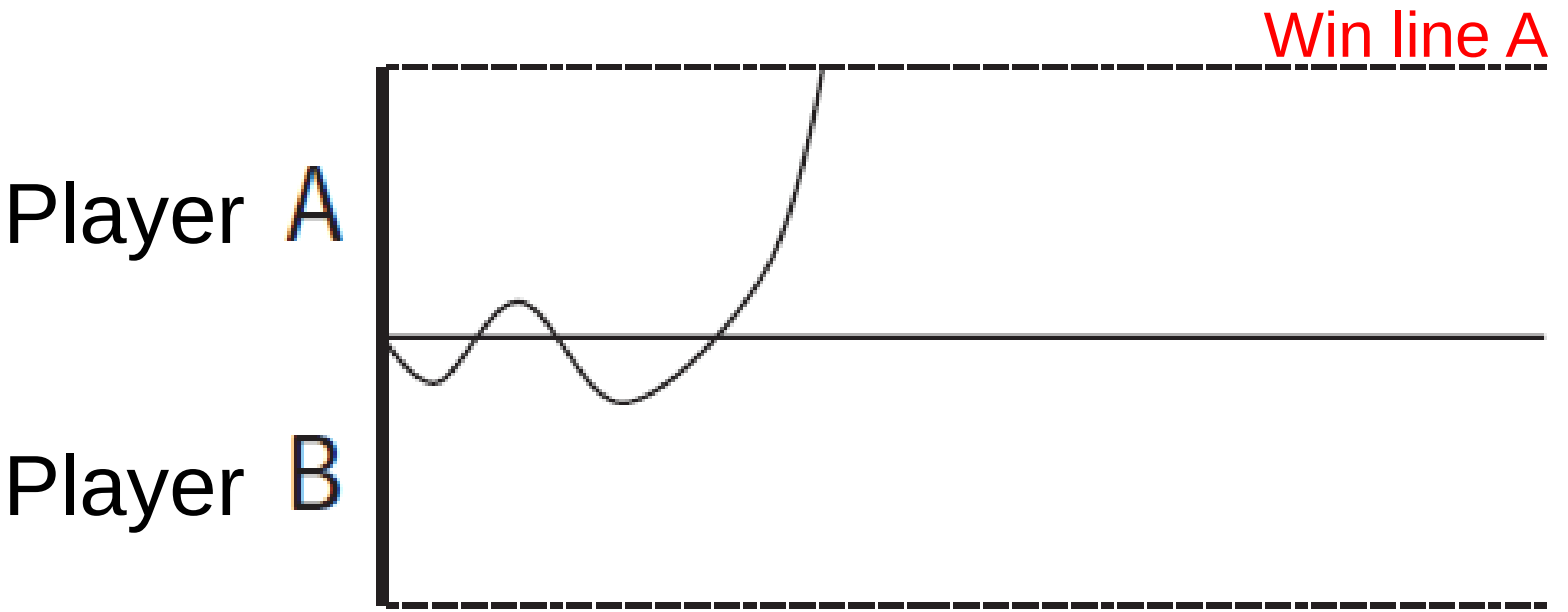
2. Unbalanced rules in B's favor

Feedback examples 3/6



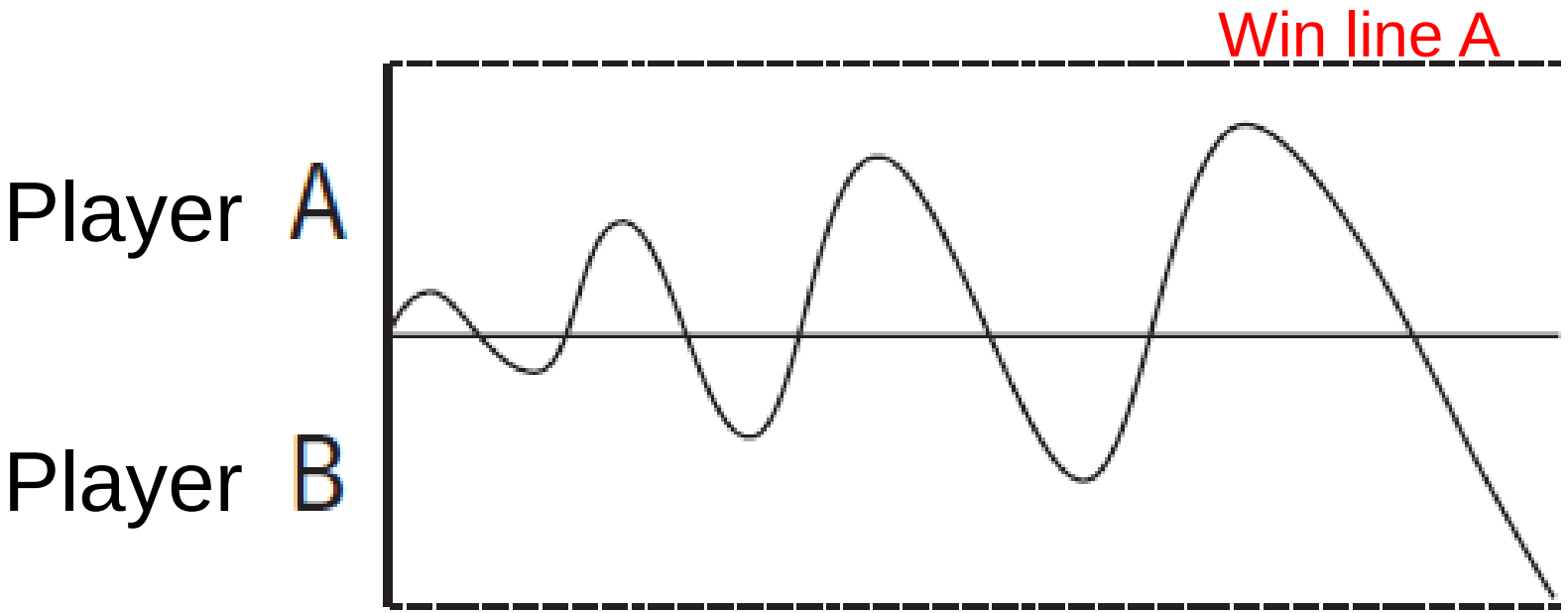
3. Stalemate (insufficient feedback to produce victory)

Feedback examples 4/6



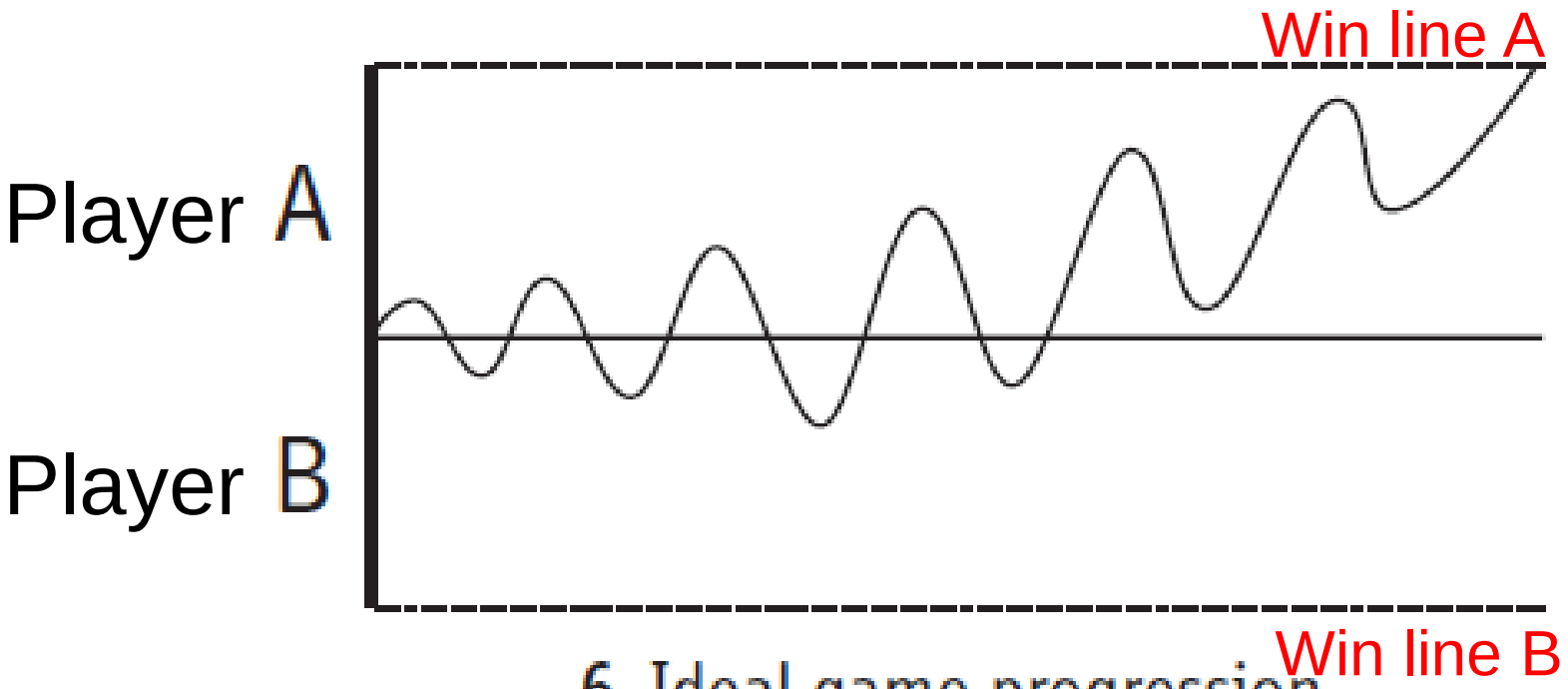
4. Balanced rules, but feedback operates too fast

Feedback examples 5/6



5. Wild swings in the lead
(powerful *negative* feedback)

Feedback examples 6/6



6. Ideal game progression (lead changes hands, better player wins eventually)

Other balance considerations

- Avoid stagnation in PvE games - when the game leaves the player in a position in which he simply does not know what to do next
- Avoid trivial solutions like:
  - any gameplay decision that has no real effect on the game world, or
  - any decision that requires the player to pick from a slate of options that includes only one reasonable option
- Separate the code from the data, and:
  - Modify only one parameter at a time
  - When modifying parameters, make big adjustments, not small ones
  - Keep records
  - Use (pseudo-)random numbers

Conclusions

- Avoid dominant strategies
- Use chance in such a way that your game rewards skillful play
- Keep the player in the *flow* state of peak enjoyment
- Use positive feedback and/or combined feedback
- Adjust the core mechanics of your game to produce a challenging yet enjoyable experience


Въпроси