

Balance of Power

Mechanics and Assets for Flow Maximization

Content

1. Introduction.....	2
2. The “Flow” state.....	2
3. Dynamics That Maintain Balance	2
4. In-Game Asset & Mechanic-driven Dynamics	4
a) Minimum Power Level.....	5
b) Communicating via Interface	5
c) Visual / Audio Language	5
d) Gradual Introduction of Mechanics	7
e) Diminishing Returns	7
f) Soft Caps.....	8
g) Maximum Power Level.....	8
h) Fallback Options	9
i) Stable Balance	9
j) High Cooldown / Limited Use Options	10
k) Average Result / Safety Net	11
Note: Recovery Timing	11
5. Meta-Game Mechanics	12
a) Saving mechanisms both manually and automatically	12
b) Offering a change of difficulty levels.....	12
6. Addendum – Analyzed Games	13
7. Contact Data.....	14

1. Introduction

This article is about what options are available to designers and artists to ensure a certain level of game balance, allowing the player to enter the Flow state.

Game balance and the Flow it allows the player to achieve is important because if a game that is about challenge is unbalanced, the player will become either frustrated or bored, which will overshadow all the other experiences and emotions the game might otherwise evoke. So in order to determine if the other parts of the game work well, balance has to be taken care of first.

Example:

A game like “Far Cry” might offer the player the joy of experiencing the visual qualities of a tropic island, but she won’t be able to focus on those perceptions while she is under constant stress because the game is too difficult for her.

2. The “Flow” state

Since this lecture is about how to raise the chances of the player reaching the state of “Flow”, let’s first have a look at what “Flow” actually is:

Flow is the mental state of operation in which the person is fully immersed in what he or she is doing, characterized by a feeling of energized focus, full involvement, and success in the process of the activity. (Mihály Csíkszentmihályi)

There are several conditions that have to be fulfilled for Flow to happen, among them being “Clear Goals”, “Feedback” and “Control”. The absence of those would cause the player to reflect about what he as a human being is doing, which destroys Flow.

In addition to those other conditions there is the one this article is about: *“Balance between ability level and challenge”*.

In the following chapters, a number of ways to use assets and certain mechanics that promote the state of “Flow” will be discussed in detail.

3. Dynamics That Maintain Balance

A comfortable challenge curve should directly follow the player’s skill (ability level), slightly oscillating constantly.

Keeping the difficulty curve like that is not trivial. The player constantly uses a wide array of games to achieve his goals, including but not limited to:

- Motoric abilities (hand/eye coordination, reflexes, strength)
- Cognitive skills (willpower, perception, combinatorics, deduction, multi-tasking)
- Social skills (making friends, solving conflict, empathy)
- Knowledge (math, physics, gaming standards, designer’s knowledge)

The interesting thing about “Knowledge” is that it may substitute many other skills if it is communicated in time

Example:

Knowing the answer to a logic puzzle will ensure success just as fine as actually solving it once it is encountered does. Telling the player what he will have to do, and getting him to train it a few times, will ensure that even players with low stress tolerance will be able to perform well under heavy action.

The main problem is that in many games the player's success in any situation not only relies on his use of skills right in that moment, but also access to vital information – usually communicated by interface and in-game assets – and on his performance in the past – usually represented by accumulated power. Unfortunately, in most games that kind of power tends to grow exponentially.

Example:

In many RPGs (e.g. D&D-based games, Diablo, MMOs), the player's base stats grow exponentially if only for the reason to make him feel powerful when checking out the stats screen or easily dispatching formerly tough enemies. Additional synergetic effects of staying power, damage output and control options usually raise the player's power exponentially as well, though.

RTS games (like Starcraft, C&C Generals) are the prime example for exponential growth, though. Not only does resource gathering tend to grow exponentially, since for each raise in production power, it gets easier to buy additional production power. The strength of an army tends to rise exponentially as well. Purchasing two units of a kind rarely ever is more than twice as expensive as buying one (considering production prerequisites it usually gets even cheaper), but their combined strength is at least three times as high (depending on the focus fire factor).

In FPS games the player's power might upset the balance because he managed to never use any of his options in any situation where that option wasn't the most efficient and effective choice available, leading to a too high power level since the game would not be balanced around that most optimal case.

If left unregulated, this quickly leads to an imbalance between the player's skill (ability level) and the challenge. Once the balance is gone, it probably won't normalize by itself unless you have taken precautions.

The imbalance might favor the player, which is bad, or favor whatever challenges him, which is worse. No matter what, the balance needs to be restored as quickly as possible to prevent deadlocks and the other negative effects like the loss of all other experiences.

For the challenge progression to unfold correctly (and thus favor "Flow"), the power levels of player and challenger alike must remain at all times. The right mechanics can all but guarantee that.

4. In-Game Asset & Mechanic-driven Dynamics

A suitable mechanic needs to satisfy a number of criteria:

First and foremost it needs to work without the cooperation of the player. If it only works with his cooperation, he will make use of it if it works in his favor, but will always ignore it if it doesn't, even if that would raise his long-term ability to have fun. The player would never willingly give up power if the game apparently is based on gathering it, no matter how bad the boredom may be.

Example:

In Half-life2, wall-mounted devices exist to raise the player's health and armor values back to normal values. No player would ever use those devices in order to lower his health/armor value to make the next passage of the game more exciting, unless there was an obvious trade-off or benefit for it.

The next condition is that it should not overly violate the player's expectations of that game's experience.

Example:

In World of Warcraft, Blizzard implemented the concept of "Rested XP", which guaranteed the player bonus experience for defeating monsters relative to how long they had been logged out of the time when they first logged back in. At the beginning the feature was explained as "after logging in, you get as much XP as you should, but after some time you grow tired and start getting less". Players didn't like that, so they changed it to "after logging in, you get more XP for some time, until it returns to normal values again". Players liked that a lot better, even though the XP numbers had not changed at all – they only changed the description in order to better fit the players' expectations.

Last the mechanics should also work well in multiplayer environments to eliminate the necessity of separate multiplayer mechanics which the players would have to learn separately.

Example:

In a singleplayer-only game it might be sufficient to regulate the balance by adjusting the strength of the challenges along the way, but in a multi-player environments that will fail if the challenge is actually another player, since he surely won't back down when the game nicely asks him to.

There are many (combinable) mechanics that fulfill those requirements. To maximize the effect of the mechanics listed below, the challenger should be subject to the same set of rules as the player. This has the added benefit of a more transparent gameplay and enables the player to perceive solutions for challenges more easily.

Also the techniques for transferring knowledge to the player should be used to spare the player constant saving/reloading or seeking that kind of knowledge on the internet.

a) Minimum Power Level

To have a level of power that cannot be fallen below is of most use in games where frequent setbacks are part of the gameplay. That minimum power level should be high enough to ensure that the player can still engage in meaningful conflict and stay competitive.

This of course only works if the power level in question is not the primary one, meaning it is not responsible for almost losing in the first place.

Example:

In most kinds of multiplayer action games, the player cannot lose his most basic set of options, e.g. a standard weapon (Half-Life: Crow Bar, Unreal: Energy Pistol), no matter how often he is defeated.

There are also several strategy games in which the player continually receives a basic income in resources or troops, no matter how badly he plays (Company of Heroes: Manpower resource). This allows her to play an important role even after showing bad performance for several minutes.

Of course it is important that the minimum level is of secondary nature, e.g. ammunition in an FPS instead of actual health points, or resources instead of troops in an RTS.

b) Communicating via Interface

Giving the player all the necessary information via the game interface is not the most elegant, but usually the most transparent solution.

In games where forcefully keeping the player away from challenges he is not yet ready for is not a great option, simply clearly stating that fact is a great way of protecting Flow (e.g. in MMOs where the player might go everywhere, monster levels should be shown). Depending on how well the interface fits the overall setting/genre, the immersion loss caused by it might be considered too steep a price, though.

Example:

HUDs that tell the player what he needs to know are a standard nowadays except in the most realistic games (e.g. simulation-like games) and are found in every genre (the selection circle / crosshair showing an entity's alignment in almost every game, the level display in most MMOs etc.).

c) Visual / Audio Language

If possible, as much information as possible should not only be delivered by the interface, but also by the game world itself. This includes the use of graphics and acoustics.

By properly tuning the look and sound of a challenge, be it the respective models, animations or textures, sound FX or music, it is possible to shape the expectations of the player. Once a certain expectation has been established (e.g. "green things are mean things"), it is important to stay consistent, building a visual and acoustical language that remains valid for the rest of the game.

Unless there is a very important message in doing so, the game – and thus also its established language – should never mislead the player or disappoint his expectations. Once the trust has been lost, the visual or acoustical language and all the work that went into it are worthless.

While graphics in general are better suited for transporting huge amounts of information when compared to acoustical means, they also depend on the player actually looking at those graphics. Audio language has some benefits in comparison.

Not only is hearing directly tied into the player's brain, since humans perceive acoustical information on a much more raw and less "interpreted" level than visual information, which makes it possible to very quickly transport crucial balancing information by e.g. putting the player into a specific emotional state using music. This makes it much easier to very quickly transport crucial balancing information by putting the player into a specific emotional state.

In addition to that, acoustical input is also almost impossible to ignore, since there is no such thing as "accidentally not listening to that sound right then", since few players will play your game without sound.

Even more helpful is that it is very easy to refocus the player's center of attention using sound without the player being able to resist it due to hardwired behavioral circuits in the brain.

Example:

In lots of games, the properties of the game world are of major importance. Since most worlds are textured anyway, this is a great way of communicating the most important properties (e.g. in Half-Life 2, small wood planks are always breakable, but corrugated metal sheets never are yet can be ripped off with the gravity gun etc.).

Prop models are also a great way of foreshadowing future encounters and their challenge level (e.g. in WC3 one would find hostile-looking props near the entrances of enemy bases, and in Half-Life 2 Combine checkpoints always had the same look and prop-usage).

One of the most famous examples of visual language is most likely the "Big Boss Door" usually found in all Zelda games.

Complex yet important states like "in combat" are easily communicated by clever use of audio (the combat music and "end of combat" jingle in Gears of War or Mass Effect demonstrate this nicely). Different set-ups of instruments can be used to communicate different challenge levels long before balance, and thus Flow, may have reached critical levels, giving the player a great opportunity to prepare himself.

Even more complex concepts that are not totally time-critical can be communicated by clever use of voice-overs as seen (or rather heard) in Brutal Legend, where the protagonist keeps talking to himself during boss fights offering the player insight on how to possibly beat the encounter.

d) Gradual Introduction of Mechanics

When complex new mechanics are to be introduced, it is often possible to introduce them gradually in several steps. Those steps might optimally include a first passive safe encounter of the mechanic, a passive challenging encounter of the mechanic, the first active use of the mechanic by the player in a safe environment and finally the active use of the mechanic by the player while under pressure.

Additions to the mechanic should be introduced in the same way. That way the designer can be reasonably sure that the player is not overwhelmed by too much information, preventing him from losing track of the situation, which might have threatened the balance.

Example:

In Half-Life 2, the Gravity Gun, and later its upgraded version, are each introduced in a multi-step process as described above.

In RTS games, new units or buildings are usually introduced by offering the player a few of them to play around with in a not-that-dangerous situation before they are really needed and permanently are unlocked.

e) Diminishing Returns

Just as the Maximum Power Cap, a degenerating “Return of Investment” greatly benefits singleplayer-modes, because of the usually huge difference in power between the challenger and the player in the early stages. The higher the ROI loss is, the faster a gap in power can be closed. Overdoing it might limit the player’s feeling for power accumulation and deny crucial positive emotional feedback, though.

Of course, there is no difference between a player that is just starting out and one that has just been completely devastated by the challenger, so this also greatly helps when recovering from a weak phase. It thus works for multiplayer games just as well.

Even though diminishing returns systems might be most easily implemented as a single steady function determining the pay-off of certain actions, effort should be put into finding a more immersive and transparent solution that is more readable to the player than a usual equation is.

Example:

The “upkeep” mechanic found in several RTS games (Warcraft 3, Sins of a solar empire) is an excellent example of how gaining resources can be subject to diminishing returns, but the experience point gain per specific monster in many MMOs (e.g. World of Warcraft) also diminishes as the player levels up.

A very harsh diminishing returns concept can often be found in FPS games, e.g. Quake, where each type of weapon may only be picked up once by each player. This makes sure that not only the strong players cannot hoard resources by picking up the respawning weapons over and over, but also they cannot deny those resources from other players no matter what they do.

f) Soft Caps

Another way of stabilizing power gain is to make recurring loss and recovery more efficient than raising the current maximum cap permanently.

Such a dynamic could be realized by implementing mechanics that cause high initial costs – possibly of a highly limited resource that cannot be “farmed” – for raising the power soft cap. This encourages (or even requires) the player to work with smaller amounts of power, which among other positive aspects (performance etc.) lowers the probability of dangerously high power concentrations. Of course this also is true for the challenger.

Example:

Especially older games (and nowadays titles with less focus on realism) commonly feature soft caps or one-time investments. Many classics limited the player's options until a certain “price” had been paid, unlocking one of those options forever (e.g. buying a greater bomb bag or quiver in the Zelda series), but recently especially RTS games have kept using the mechanic (usually found in form of global upgrades, e.g. the “relic” level troops/tanks in the Warhammer 40k RTS, or a higher overall troop limit in other games).

In those games, playing without those options for some time until the necessary resources could be accumulated without major problems (and thus not upset the balance earlier) was usually a good idea.

g) Maximum Power Level

Restricting the maximum available power is especially useful in certain situations where the gap in power is meant to be huge by design. In such a scenario, the player usually starts at the bottom and works his way up to the top. This is most common in singleplayer modes.

To prevent the balance from quickly leaving the intended interval – most likely by an additional power-gain of the challenger - a hard cap might be established. Of course, this also works to keep the player from amassing huge amounts of power in relative safety in order to gain a sure victory.

The cap, or some kind of sub-cap, should be reached after a relatively short (depending on the encounter-length) amount of time.

Maximum caps also provide a nice unofficial goal for the player, giving him feedback on how well he is doing and serving as a great progress indicator, and thus add nicely to the overall drama curve as well as the pacing.

The maximum of power that may theoretically be gained is not suitable as a cap, as there is no guarantee that the player will reach it in time. That kind of power ceiling is more of a progress meter for the player.

Example:

Just about every game is rife with power caps. Usually they come in form of health and ammo limits in action (Half-Life 2) and role-playing games (Zelda series) and population caps in more strategic games (Starcraft, Company of Heroes).

An effective way of assuring a certain level of balance is keeping the power level of the challenger relative to the player's. To prevent the player from feeling a loss of achievement, it is very useful to severely limit that dynamic adjustment using tight maximum (and minimum) power levels. This of course does not work in games where both the player and the challenger are governed by the same rules (e.g. RTS), but works quite well in others, e.g. RPGs (Mass Effect, Fallout 3).

h) Fallback Options

To ease the short-term challenge after the player faced a set-back, it is useful to implement mechanics that enable player to retreat to a state he knows is safe. This lowers the difficulty for a while, allowing the player to relax a bit and return to a state of mind where he might be able to learn from his mistakes.

This is also a good time to change the skill set that is required from the player a bit, since he does not seem to be too good at what he was doing before the setback. That way the player has to earn the regain of his power and feels satisfied by that.

Also this usually buys the other mechanics some time to kick in.

Additionally this creates the option for the player to achieve "incremental success", beating a challenge one step after the other, usually "paying" for that with additional game time or lower scores.

Example:

In almost all of the more recent FPS games, the player character has the option to take cover frequently and enjoys the benefits of automatically regenerating health at the same time (Halo 3, Call of Duty, Gears of War). This allows the less skilled players to beat a challenge in multiple runs.

Mechanics like that can traditionally be found in RTS games, where withdrawing to your well-protected base is possible in almost every title (Warcraft series, Starcraft series, Command & Conquer series, Age of Empires series, Company of Heroes).

i) Stable Balance

In many kinds of games, it is possible to facilitate situations that are inherently stable, as long as nobody decides to make a move. Either the designer specifically set up a situation in which all involved parties are in a state that stabilizes the Status Quo, or the actors pursued that state themselves since that is where they are strongest.

Whoever now attempts to shift that balance weakens his own position temporarily, because he left his personal state of stability. This offsets the advantage of seizing the initiative, which at the same time lowers the overall pacing. Generally a pacing that is a bit too slow is easier for the players to deal with than a too fast one, since it does not make learning more difficult.

Example:-

In most games, an actor's state primarily consists of his internal possible actions and what his environment provides him with. This is usually the case in RPG, RTS and FPS games.

In all those genres, the actors try to take up a position that best fits their possible actions, or the other way around. This typically leads to a situation where all actors occupy different positions, each having adapted his possible actions accordingly. Whoever now makes a move into the opponent's terrain severely weakens himself.

This is commonly found in RTS, represented by mechanics as supply lines and choosing terrain favoring the own army (e.g. Starcraft: Zerglings are strong in open field, weak in choke points).

In FPS games, this commonly is encountered in situations like the sniper taking up a position that allows him to cover a large region of the map that is far away from him, whereas somebody equipped with a shotgun would seek short narrow spaces that allow him to maximize the weapon's strengths (wide spread, high power) without having to deal with its weaknesses (very short range)

An inversely-working version of this mechanic is sometimes found in Beat'em Up games, where attacking (instead of waiting for the enemy to attack) is the standard action. In some of those games, standing around and letting down all defenses leads to filling up a gauge used for powerful special moves (Street Fighter, Capcom vs. SNK).

j) High Cooldown / Limited Use Options

Another means of stabilizing critical situations is creating high-powered options that are severely restricted in their usage. The limitations – which are necessary to restrict likely usage to critical moments – are most often found in either the frequency or the number of total uses.

It is a good idea to have those high-powered options boost the primary power of the player (or prevent further losses) or lower that of the challenger, which should be what is directly determining success or failure in critical situations. In order to prevent the player from simply using those options as soon as they are available, which is tedious as well as does not help the overall balance, the power boost granted by those options should be cheaply available in times of low stress, making it unviable to use this mechanic as a common means of power-gain.

Also the frequency with which the player might actually need those options – meaning he is about to fail – needs to be rare enough to make waiting for the option to become available again before continuing in the game inefficient and unviable.

Example:

Especially RPGs – traditionally not the most transparent of games – have always featured one-shot options to bring the player back from the brink of defeat or ensure victory in an otherwise even match. The most common examples are health or buff potions or powerful “daily use” spells and abilities (WoW, D&D games), as opposed to “resting” as the ordinary way to restore that power.

In the last years, high-cooldown support options have become commonplace in RTS-games as well, though – usually they enable the player to further customize his play-style without providing him with constantly available tactics (C&C Generals: support drops, Company of Heroes: General Powers).

In action games, one-time-options are usually found in form of super-bombs or equivalents that help the player without demanding too much skill, should he have misjudged a situation (R-Type)

k) Average Result / Safety Net

In many games, a single challenge might very well be decided within a few seconds. To prevent Flow breaks caused by “accidents” that might be perceived as random or unwarranted by the player and lead to frustration, mechanics that provide the player with a “second chance” or a “win 3 out of 5”-type challenge might be implemented.

Those mechanics might take the average player performance into account or delay defeat just long enough for the player to prove his skill once more.

A highly important factor is if, and how, past successes, draws and failures influence your chances in future challenges. While raising the chances of success to a certain degree stabilizes Flow, but also reduces the feeling of accomplishment, reducing them is risky balance-wise.

Example:

In several FPS games where there is only one entity representing the player, and that entity might very well end up in situations where a single hostile action might potentially lead to a defeat, there are mechanics in place to ensure that no single enemy action might ever do that (Bioshock: the player cannot fall below 1 hit points from a single attack).

Beat’em-up games, just like many traditional sports, usually feature a “best of X” mechanic, making sure that a single defeat does not cost the player all his progress while maintaining Flow at the same time (e.g. Street Fighter 4).

In RTS games, the map usually contains enough resources for the player to start several attacks instead of just one – if there are enough resources for 5 attacks, on hardest difficulty all might have to succeed, while on the lowest a single one might do it (Starcraft, Warcraft 3, Command & Conquer).

Also one regularly finds “ticket systems” which are similar to the “best of X” mechanic, but less discrete. Examples can be seen in Battlefield 2 or Company of Heroes.

Note: Recovery Timing

However the recovery of power may be implemented, the duration of the process and its frequency should be tightly linked to the actual encounter length. If that is not the case, reloading will very often be the better alternative time-wise, which will destroy the immersion and Flow completely.

5. Meta-Game Mechanics

a) Saving mechanisms both manually and automatically

Should all the systems fail to keep the player in the game, which certainly happens every now and then, there should be additional fallbacks available for the player.

The least frequent one, which is found in most games nowadays, usually is an auto-save feature, preferably with rotating save-games (meaning keeping older auto-saves, as found in the FPS “Prey”).

This should be supplemented by rotating quick saves (as found in Half-Life 2) and – depending on the game and tech – check-points (saves that do not survive a game reset etc., as in Call of Duty 4).

Example:

The FPS “Prey” keeps all save-games that have ever been made – quick saves, auto saves and regular saves. This has the added benefit of offering the player a sense of “history” as well, but of course comes at the price of disc space.

Half-Life 2 is one of the few games featuring rotating quick-saves instead of a single one, which drastically lowers the chances of the player unknowingly saving his progress in a deadlock situation (e.g. with 1 health point left 3 seconds before being hit by a grenade).

Call of Duty 4 is tightly packed with checkpoints that make sure the player cannot lose more than a few minutes of gameplay.

b) Offering a change of difficulty levels

If the player either repeatedly fails or overwhelmingly succeeds at a challenge, he might have chosen the wrong difficulty setting to begin with. This can be corrected by offering a more fitting difficulty setting at the appropriate time (done in Warcraft 3, where it is the only way to access the “easy” difficulty).

Example:

Warcraft 3, among other games, offers the player to lower the difficulty setting after he has failed a mission a few times.

6. Addendum – Analyzed Games

The following games have been selected as great examples for specific mechanics/assets (listed by mechanic, followed by games where it can be observed):

- Maximum Power Level
 - Half-Life 2 (ammunition)
 - Starcraft (troops)
 - Zelda (bombs, arrows, hearts, rupee limit)
- Minimum Power Level
 - Unreal (unlimited ammo gun)
 - Company of Heroes (manpower reinforcements)
 - Halo (shield energy)
- Diminishing Returns
 - Warcraft 3 (gold/wood income)
 - World of Warcraft (experience points for monster of certain level)
- One-Time Investments
 - Zelda (larger bomb bag, bigger quiver)
 - Starcraft (armor and weapon damage)
 - Company of Heroes (special abilities for specific unit type)
- Fallback Options
 - Call of Duty 4 / Gears of War / Halo (cover & regenerative health)
 - Warcraft 3 / C&C series (base defenses & enough resources for multiple attacks)
- High Cooldown / Limited Use Options
 - C&C Generals (support powers)
 - Call of Duty 4 / Half-Life 2 (grenades)
 - World of Warcraft (potions and skills with high cooldown)
 - Company of Heroes (Generals powers)
- Inherently Stable Situations
 - Starcraft / Warcraft 3 / Company of Heroes (specific unit's strengths / weaknesses depending on terrain)
 - Call of Duty 4 Multiplayer (weapon / perk load-out vs. level layout)
- Average Result / Safety Net
 - Bioshock (no one-hit kills)
 - Company of Heroes (unlimited resources & strong base defenses & upkeep)
 - Warcraft 3 (lots of resources & strong base defenses & upkeep)
- Auto-Saves
 - Half-Life 2, Prey, Warcraft 3
- Quick-Saves
 - Half-Life 2
- Checkpoints
 - Call of Duty 4
- Difficulty Adjustment Offer
 - Warcraft 3 (easy difficulty available after a few failures)

- Visual Language
 - Half-Life 2 (destruction / usage of gravity gun possible depending on texture, e.g. wood vs. metal)
 - Zelda series (boss doors)
- Audio Language
 - Gears of War / Mass Effect (combat music & end-of-combat sound)
 - Brutal Legend Trailers (hints offered by protagonist)

7. Contact Data

André Beccu
Am Ziegelhof 12
77694 Kehl
Germany

Phone: 0049 / 151 – 230 20 845

Private Mail: andre.beccu@web.de

Company Mail: abeccu@spellbound.de