

# Complexity Progression in Turn-Based Strategy Games

Passarella Valeria

## Introduction

One of the most attractive and captivating elements of games is the clear sense of progression and growth they provide, not only in regards to the player characters' abilities, but also in the ever-growing and increasingly complex challenges they will have to face: to prevent the experience from becoming stale and unrewarding, the game's difficulty should rise steadily according to the player's progress.

Strategy games are particularly effective at conveying a sense of scale and player growth, by gradually granting the player more control over units, upgrades, resources, and territories. But when it comes to the opposing forces, the enemy units, how can that same sense of natural growth be communicated? How can they still pose a threat to the player as they become more powerful and experienced? The most obvious answer is to simply scale the enemy's statistics accordingly - a big enough number and the player is sure to falter eventually - but this alone doesn't solve the problem of continued player engagement, as it's rarely a satisfying challenge to overcome.

Instead, the answer should be looked for elsewhere, in other aspects of gameplay that may not strictly relate to the enemies themselves, but that can strengthen them and increase the game's complexity all the same. By making use of secondary or unexpected elements to create new types of conflict, the player's experience can be kept fresh, challenging, surprising, and exciting even hours into the game. Here, I would like to explore some of these possible options, specifically in regards to turn-based, grid-based strategy games.

## Referenced Titles

- *Pokémon Conquest* - Tecmo Koei, Nintendo (2012)

Playing as a warlord, the objective is to conquer every kingdom in the region, by defeating each respective enemy warlord and defending against potential attacks to one's own territories. The player can recruit new warriors to their army after defeating them in battle, who can then be deployed in the various kingdoms under the player's rule.

Up to six warriors can be used in battles. Each warrior has a special ability and the option to hold an additional item, but can only choose a single pokémon out of their team to carry into battle. The warrior's pokémon act as the actual units the player

controls in the arena. Each pokémon has a single move, an ability, and a set of stats that can be increased by battling. Warriors can acquire new pokémon any time one is encountered in the wild.

The player wins a battle by defeating all of the enemy army's pokémon and, in turn, loses by allowing all of their own army's pokémon to faint. Winning a battle results in the player successfully conquering or defending the chosen kingdom, while a loss results in the kingdom being lost. When the player loses their last kingdom, they get a game over.

- *Into The Breach* - Subset Games (2018)

The player takes control of a squad of giant mechs tasked with defending a group of islands from an invading species of aliens called *Vek*. The four islands can be tackled in any order the player chooses, but a minimum of two have to be completed in order to unlock the final battle. Islands are reclaimed by winning a series of encounters, and then defeating the respective boss.

A squad is composed of only three mechs, which may be upgraded throughout the play session. As long as a mech survives the encounter, all damage done to it will be healed at the end of the battle. Instead, the player's most valuable resource is *Grid Energy*: one unit of it is lost any time a *Vek* manages to damage a civilian building during a battle, and allowing it to be depleted completely results in a game over.

As such, the main objective in any encounter is not to eliminate all *Vek* units, but rather to minimise *Grid damage* each turn. Additionally, for every battle the player is given a small list of missions, additional challenges that grant a number of rewards upon being successfully completed. Mission rewards are essential to the player's progression and survival, but succeeding in them requires the player to pay attention to multiple different objectives, which may even come into conflict with each other.

- *Mario + Rabbids Kingdom Battle* - Ubisoft (2017)

A group composed of Mario characters and their respective Rabbid counterparts travels through a twisted and absurd land with the objective of restoring its order, fighting groups of opposing Rabbids along the way.

The player can choose three out of the available characters to create their party, and each character can then be customised through an array of abilities and weapons. During a battle, any player unit can perform up to three actions each turn: move, attack, and use an ability. During the movement phase, the active range of a unit can be greatly expanded by jumping on another player unit, and small additional damage can be inflicted to enemies by dashing through them. Both player and enemy units can hide behind cover to significantly reduce their chances of being hit by opposing attacks.

Winning an encounter allows the player to proceed in the overworld. Damage inflicted to player units carries over between encounters until their health is restored at specific checkpoints. Depleting a unit's health during a battle renders it unusable until it is healed again, and another unit has to be used in its place.

# Why “Complexity”?

Throughout this document, I will be referring to a game’s “complexity” level rather than its “difficulty”. This is because, although related, such terms are not interchangeable.

Judging a game’s difficulty with precision is not always possible, at least not objectively. Turn-based strategy games don’t require dexterity skills such as a short reaction time or precise aim: parameters like timing and accuracy can’t be measured, and the game speed is always set by the player (depending on how quickly they get through their turn). The player is free to approach their turn in any way they’d like, for as long as they’d like, and as such, the extent to which they will struggle with a game will be greatly affected by their own skills, aptitudes, previous experience, and current awareness.

It is possible to make a strategy game (almost) objectively easier, by building a more flexible and forgiving system (for example, by adding the option to undo a move); but reliably making it harder is much less straightforward. An enemy with a larger health pool or damage range does make the game more punishing and time-consuming, but these factors alone might result in an unpleasant, unexciting, and unfair experience more often than in a captivating challenge. It’s possible that the latter may come to be after enough balancing, but hand-crafting the perfect difficulty curve in this context would be a highly specific and laborious process.

Instead, I find that concentrating on affecting the game’s complexity would yield more positive and flexible results. By focusing on a game’s systems and the interactions between them, and further developing them in order to drive the game’s progression, the player is continuously provided with an increasing depth of gameplay: such a system is not only more likely to engage the player, especially if their interaction with it is aptly rewarded, but it also gives them more elements to consider during their turn, and thus also affecting the game’s difficulty in a way that’s more natural for the genre.

# Methods to increase complexity

## Changes to enemy units

- Statistics increase

While not a perfect solution, increasing the enemy units' statistics (such as HP, Atk, and Def) to match those of the player can be an effective tool to create an immediate sense of progression. The player will easily recognise to be up against more powerful foes, and will thus adjust their strategy accordingly. However, as mentioned in the introduction, this method doesn't intrinsically make the game more challenging, only more time-consuming; thus, it isn't recommended to rely solely on it.

- Additional enemy types

Another quickly recognisable way of increasing the potential threat posed by the enemy units is to introduce variations to enemy types. They may range from simply a stronger version of a previously established unit to, more often, entirely new units. New enemy types will normally be equipped with different moves, additional abilities or side effects to their moves, or both.

They have to be distinct in their visuals, so that a player may be able to tell at first glance in what way they differ from the units they were already familiar with.



Left to right, a Ziggy and a Smasher from *Mario + Rabbids Kingdom Battle*. Both are common enemies, introduced early on, and they serve as counterparts to each other: the Ziggy hides behind cover and shoots from afar but doesn't have much HP, while the Smasher roams around the arena, attacks with a powerful melee weapon, and has a good amount of HP. All of these characteristics are made unmistakably clear in their

designs; from visuals alone, the player is able not only to differentiate them, but also to predict their play styles even before being formally introduced to them.

Additional enemy units should be vastly different from all preexisting units: not in statistics, but especially in moves and behaviour. There should be little overlap between the different enemy types, and the player should be required to adopt different strategies in order to defeat each one. By ensuring each enemy type creates a unique and specific type of conflict, they are all made valuable from a game design perspective, and all units are prevented from being perceived as replaceable or redundant. This also allows for the creation of more complex challenges for the player, as they will have to prioritise certain units and carefully choose their moves depending on the situation.

- **Buffs to enemy units**

Temporary power-ups are an additional method to strengthen enemy units. They take the form of buffs to the enemy statistics, or additional effects to an enemy's actions, differing from the previous methods in that they're usually active for a limited number of turns or as long as a certain condition is being met. They are most often initiated by a special type of enemy unit or a non-attacking object on the arena, making the destruction of their sources a priority, and in turn diverting the player's attention and forces away from the regular enemy units.

An interesting example is the *Psion* unit from *Into The Breach*. *Psion* units do not attack, but instead provide all other enemy units with an additional effect as long as they stay alive: these effects include health regeneration, damage resistance, or the ability to explode upon being defeated. Depending on the circumstances and the specific type of *Psion* present, its defeat may be a trivial, optional matter, or essential to a favourable outcome of the battle. It's up to the player to decide whether to divert away from the fight part of their very limited numbers to get rid of it, or instead concentrate on counteracting its effects for the remaining turns.



The *Blood Psion* (left) restores one unit of health to all enemy units at the end of the player's turn, while the *Shell Psion* (right) provides an extra unit of defence to all enemy units. Both of these effects are indicated on Vek units by a small, unique icon.

## Changes to the arena

- Increased size

Increasing the size of the arena doesn't only create a natural sense of progression, but it also requires the player to keep under control a bigger area, and a bigger number of enemies. They must be aware of what goes on right in front of their units, but also of whatever may be approaching from afar, necessitating an increased level of attention and care in regards to their actions. However, too big of an arena or too little a number of enemies, and the battle may be perceived by the player as dragging out unnecessarily.

The player units' movement range is usually unaffected by this change, meaning they will take a greater number of turns to traverse the entire arena. Because of this, if they are unable to cover that distance through their attacks while the enemies are, they will be forced to receive an unavoidable amount of damage before being able to react. This is usually undesirable, as inherently frustrating for the player: placement of enemy units should be carefully considered as to reduce it or avoid it entirely.



Levels in *Mario + Rabbids Kingdom Battle* increase significantly in size right from the early hours of the game, to the point that player units become barely visible in the arena preview at the start of the battle.

- Normally inaccessible areas

The number of tiles that units can move through rarely corresponds precisely to the size of the arena.

Tiles may be inaccessible because blocked by non-player, non-enemy units that function as cover, requiring them to be destroyed before any other unit can occupy the underlying space. Cover blocks damage in part or entirely, and thus is often an essential element of both the player's and the enemy's strategies. Usually, attacking cover damages it and eventually destroys it: it allows the player to buy time and divert enemy attacks, but they can't rely on it for the entirety of the battle. Reversely, the player may choose to intentionally destroy enemy cover to free up enemy units to their attacks, and an additional space for their own units to move through.



Cover being destroyed in *Mario + Rabbids Kingdom Battle*.

Alternatively, inaccessible areas may be occupied by special environmental tiles, such as water or lava. These tiles often have additional effects, and can only be accessed by special types of units. As such, causing any other type of unit to find itself on a special environmental tile (usually through the *pushing* effect of an attack) inflicts additional damage to it, and can also apply an additional negative status effect. These effects apply to both player and enemy unit alike, making them not just an obstacle for the player, but also a tool at their disposal.

In *Pokémon Conquest*, *lava* and *water* tiles can only be crossed, respectively, by *fire-type* and *water-type* pokémon, and those with *flying* abilities, making units that possess such traits much more valuable in tight arenas where such tiles significantly restrict normal movement.



The *fire-type* pokémon *Tepig* standing on a lava tile, and the *water-types* *Wooper* and *Magikarp* standing on water tiles, which are normally inaccessible to other types of units. Exception to this is *Drifloon*, in the far right image, who is a *flying-type*, and thus able to move over any type of tile.

In *Into The Breach*, only the units with the *Flying* ability can cross *water*, *lava*, and *chasm* tiles, and any other unit is immediately killed upon entering them: this makes such tiles a greatly effective strategy to eliminate the more resistant enemies.

Additionally, units with the *Massive* ability, such as the player's mechs, survive being pushed into water, but cannot perform any attacks while *submerged*.



The *Triptych Mech*, a player unit, has the *Massive* ability: despite being *submerged*, that is, standing on a *water* tile, it was not killed; only prevented from attacking for the turn (but not from moving).



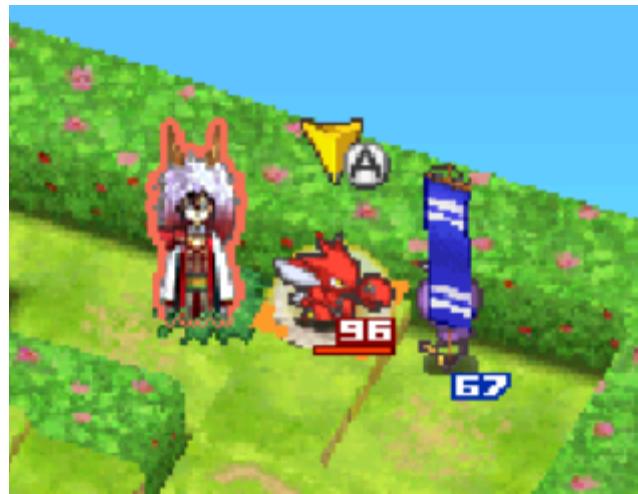
The same is true for Vek units, such as this *Firefly Leader* (purple): it too has the *Massive* ability, and so pushing it into a *water* or *A.C.I.D.* tile doesn't instantly kill it, but it does cancel its attack for the upcoming turn.

## • Hazards

Hazard tiles are another type of special tile that affect an arena's navigability. They are accessible to normal units, but inflict either damage or a negative passive effect to the unit that lands on them. Like inaccessible areas, they affect all units regardless of their alliance, and some specific abilities may be able to evade their effects. They take effect every turn, or just once per instance.

Hazards in *Pokémon Conquest* are mostly unpredictable events: stepping on an unmarked tile may reveal a *hidden trap*, which prevents a unit from moving for a number of turns, and *falling rocks* or *thunder* will occasionally strike random tiles in certain

arenas. This makes it impossible for the player to avoid them, utilise them to their advantage, or factor them into their strategy, rendering them more annoying than anything else.



An enemy unit stuck in a trap in *Pokémon Conquest*, unable to move or attack.

By contrast, in *Into The Breach*, tiles where hazards will be present are marked clearly, like everything else in the game. This way, the player is able to build their strategy for the turn around them, turning a potential danger into an advantage. They take the form of *wildfires*, which inflict small damage to a unit every turn, *smoke*, which prevents a unit from attacking, *artillery strikes*, which immediately kill any unit present on the affected tiles, or natural events such as *tidal waves*, *volcanic eruptions*, and *earthquakes* which turn the marked tiles into inaccessible areas (respectively, *lava*, *water*, and *chasm* tiles).



An *artillery strike* in *Into The Breach*: affected tiles are marked during the player's turn, and are damaged at the start of the enemy's turn.

## Changes to gameplay

- Special win conditions

Additional complexity may be achieved by directly altering the necessary win conditions, requiring the player to reconsider their approach to the battle and develop a new plan of action. Such a method is most effective later on into the game, once the player has already developed a consistently-reliable strategy: at that point, switching things up and forcing them to behave differently will offer a more memorable challenge.

It's important, however, that the alternative win conditions don't stray too far from the original mechanics of the game, or they may come off as confusing, awkward, and out of place. The game system should remain intact and unchanged, and the focus should be merely shifted to some secondary element that would otherwise be of minimal importance.

In *Mario + Rabbids Kingdom Battle*, the player will occasionally be tasked with escorting a non-attacking unit through the arena, instead of defeating all enemies like is most often the case. They must thus prioritise its movement and provide for its defence, making sure to clear the path ahead and avoid excessive exposure.

In this game, moving through the arena is a fundamental part of winning battles, and so this special win condition does not feel especially different.



An escort mission in *Mario + Rabbids Kingdom Battle*.

A battle in *Pokémon Conquest* is normally won by defeating all enemy pokémon, but certain areas have special requirements for victory: the *Greenleaf*, *Illusio*, and *Viperia* battlefields require all banners to be captured by the player, while that of *Pugilis* to defend the banners for five turns straight by making use of its unique layout.

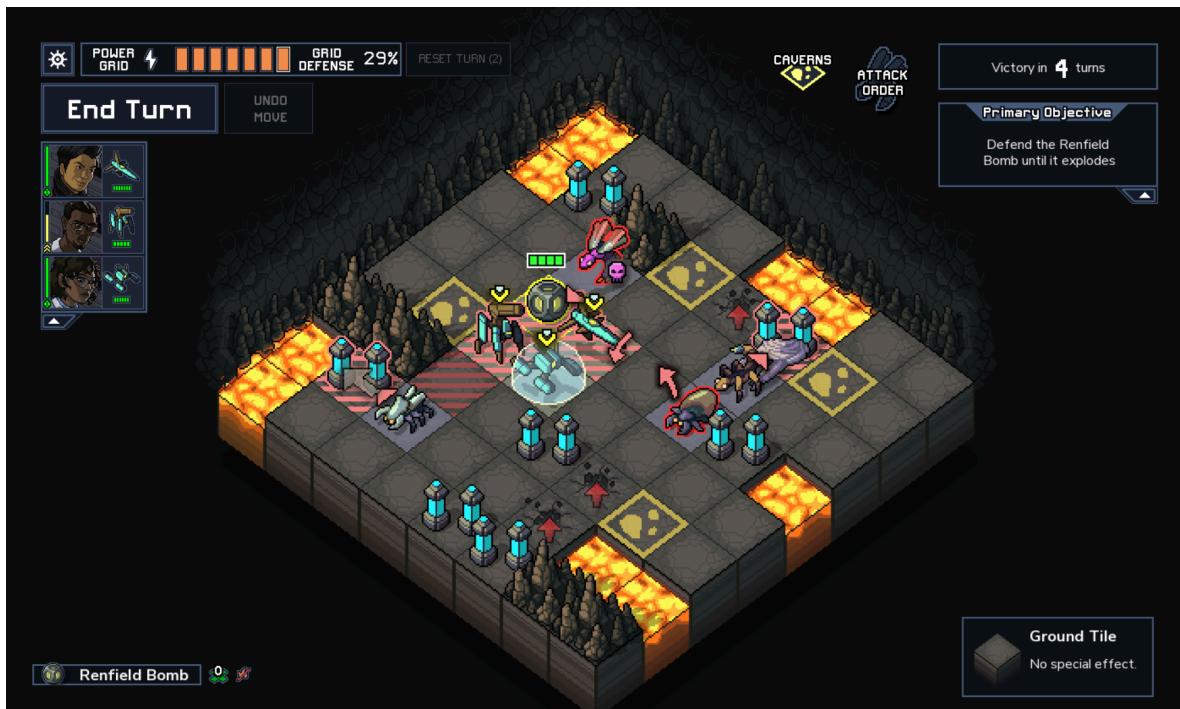
Holding ground and defending certain areas is an effective strategy to make full use of one's entire army to defeat the enemy, and so this special win condition feels like a natural evolution of the concept.



Battles in *Pugilis* are won by capturing all four banners and then protecting them.

Every battle in *Into The Breach* is accompanied by a number of additional missions, which encourage the player to pursue different objectives throughout a single battle and perform actions they normally may not. Though optional, their rewards may be of absolute importance to a player; the player must decide what to prioritise and what they can afford to miss out on, increasing significantly every encounter's complexity. Moreover, during the final battle, the player is given an additional non-attacking unit to defend for every remaining turn, alongside the regular mechs and *Grid Energy*; this makes careful management of all three resources a necessity, and significantly increases tension.

Much of the game's core difficulty centres around defending different resources and making calculated sacrifices, so even though the player is unexpectedly introduced to the new unit, it still feels perfectly in line with their previous experience.



The final phase of the final battle in *Into The Breach*. Note the *Primary Objective* requiring the player to protect the *Reinfield Bomb* (highlighted), while *Grid energy* is still able to be depleted and mechs are still vulnerable to enemy damage.

- Gimmicks & minibosses

Another way to temporarily switch up gameplay is the inclusion of minibosses, a gimmick restricted to a particular arena, or a combination of the two.

Often, these tend to deviate from regular gameplay significantly, as their objective is to be a unique stand-out moment, a short difficulty spike or a sort of refreshing break. Because of this, they are hardly used more than once or twice, making them inelegant solutions.

This method differs from special win conditions in that the player doesn't have to behave any differently to obtain a victory: they may not engage with the gimmick at all, and they might approach the miniboss like they would any other enemy.

In *Mario + Rabbids Kingdom Battle* progress is often punctuated by a fight with a new miniboss, all of which have unique abilities or moves.

In *Pokémon Conquest* many battlefields have unique additional gimmicks, which never appear anywhere else. The enemy army will use them to their own advantage, but interaction with them is not necessary for the player's victory. The *Pokemari balls* in *Chrysalia*, the rolling rocks in *Cragspur*, and the random events in *Yaksha* fall under this category.



The battlefield of *Chrysalia*. Attacking the *Pokemari balls* in the arena will cause them to bounce away, damaging any pokémon unit along their paths.

# Applying the methods

A game isn't required nor expected to use all of the methods discussed above; instead, it should include only the ones that best fit its identity, theme, and gameplay specifics.

Regardless, each of these methods affects the game's complexity in different ways, and it's possible to deduce a general order in which they are usually introduced.

In most cases, the first element to be affected will be the arena: it's the common ground, and so any change will affect both the enemy and the player. This should avoid giving the enemy a perceived unfair advantage before the player is able to get used to the game's systems. By simply increasing the arena size or adding some inaccessible areas, and not directly affecting the enemy's power, the player is free to experiment and familiarise themselves with the new changes without the risk of serious repercussions. However, more complex changes to the area, such as hazards, should not be introduced at this time.

An increase of the enemy's statistics will naturally follow soon, and it often persists until the final stages of the game. New enemy types will be added not long after, usually once the player has proved to have acclimated to the basic gameplay, as to avoid creating an environment that's too punishing. However, presenting a new enemy type early on will immediately communicate to the player that the game will change as they proceed: this should discourage them from feeling infallible, and create excitement at the prospect of discovering new types of units.

Afterwards, once the player has demonstrated to be confident and capable of overcoming the more basic challenges, more radical changes may be added, such as arena hazards or more threatening types of enemies. Soon after, gameplay changes may follow. At this point, most of the game's complexity progression will be driven by such additions: new changes may be made at a higher rate up until the midpoint of the game, and will then slow down considerably.

Significant gameplay changes are also common during the game's final stages: bosses or special win conditions are effective ways to create tension and spectacle for a memorable final send-off for the player.

## Conclusion

When attempting to increase difficulty in turn-based strategy games, directly altering the game's complexity instead often results in a more captivating and challenging experience for the player, rather than one that's simply more time-consuming or irritating.

The methods outlined in this document are not the only possible design choices to increase complexity; they are, however, the most common in this specific genre, proving themselves effective design tools.

Like all tools, they should not be used mindlessly, instead closely considering whether their inclusion does benefit the specifics of the game, and add to its core gameplay or thematic elements. If harnessed effectively, they can help to create an engaging sense of progression, and memorable trials for the player to overcome.