4 Games as Systems

If chapter 3 dealt with systemic ingredients of games, this chapter deals with emergent systemic phenomena: properties of games that are more large-scale, and that typically emerge from the rules rather than being included within them. The sections in this chapter look at games as an abstract whole rather than looking at concrete individual elements.

We begin with a discussion of abstract games (like rock-paper-scissors) and how they may map onto more concrete games or parts of those games. We then discuss the overall flow of the game: first from the point of view of the ebb and flow of victory, whether accelerating (“snowballing”) or decelerating (“catch-up”), then from the point of view of complexity (does the game become more complex over its course, less complex, or does the complexity shift in some more complicated way?). Finally, we talk about game balance, and how the strategic choice in the game—that is, the experienced complexity of it—can collapse if the balance is wrong.

4.1 Characteristic: Abstract Subgames and Essential Games

Games are often disguised versions of other simpler games, or have other simpler games within them. An example of the former is when one says of a political game “it’s really just the chip-taking game.” An example of the latter is a rock-paper-scissors relationship between units in an RTS: perhaps melee beats ranged beats flying, but flying beats melee because the melee units can’t hit back. These abstract games may be good or bad for the game in question (depending on how they are integrated into the game, and whether they support or simply overwrite the other elements of the game), but recognizing them is always a help for understanding the game.
Explicit Subgames

Sometimes a game is quite explicitly made up of distinct subgames. *Mario Party* and *Wario Ware* are collections of minigames. In the boardgame *Titan*, the main game involves traveling around the board building up an army, but when two armies fight, a tactical subgame with its own distinct rules is fought on a separate map. In some live-action roleplaying games rock-paper-scissors is used explicitly as a conflict resolution mechanic.

Games with subgames fall into three rough categories:

1. *Games that contain subgames* *Titan*, most MMOs (buying from vendors, e.g., is a subgame), most *Final Fantasy* games, and many other examples.
2. *Games that are subgame collections* *Mario Party*, *Cranium*, *Wario Ware*, *Barbu*, dealer's choice poker.
3. **Staged games** Games that fall into stages so distinct, they are essentially different games. Examples include bridge (bidding and *play*), *Magic* drafting (the initial draft of cards for your deck, followed by playing with the deck you drafted), or Skitub (accumulate cards in the trick-taking phase, then attempt to void your hand in the rummylike phase).

There is a continuum between subgames on the one hand and metagame activities on the other, especially with staged games. Bidding and play in bridge definitely feel like subgames of one game. *Magic* drafting, *Magic* constructed (deckbuilding and play), *Warhammer* (army selection, painting, and play), and *D&D* (character creation and play) feel successively more like a game activity with an associated metagame activity. But for any of these games, one can think of them as a (smaller) game with a pregame activity, or a (larger) game that’s made up of subgames. Note that each of the subgames of a given larger game will tend to have its own characteristics: length of playtime, interactivity, costs, rewards, and so on, and thus may appeal to different parts of the larger game audience.

**Implicit Subgames**

Although some games have explicit subgames, it’s more common, as in the melee-ranged-flying RTS example, for the subgames to be implicit. Another example, found in many TCGs and RTS games, is early-game units losing to midgame units losing to late-game units. Showing up a little later with a bigger army can win you the game; planning to show up much later with a huge army is bad for your chances, because you will lose to a fast army before your huge army is built. A related RTS rock-paper-scissors is the rush-eco-turtle (aka rush-boom-turtle) relationship, where a fast attack beats an economic buildup, which in turn defeats a defensive strategy.

These rock-paper-scissors subgames do not determine the outcome of the game all by themselves; they are merely one element. Enough other things are going on that it is easy to miss the relationship of the elements in the subgame, because winning or losing a particular subgame can easily be washed out by losing or winning other subgames. But to some extent winning or losing the game as a whole can be thought of as winning or losing enough of the various subgames. Many elements of design can be broken down by looking at the subgames as well—for example, a balance problem in one of the subgames (like rush being too good against turtling, or ranged beating both air and melee) can spill out to break the game as a whole.

Another abstract game that appears as part of many games is the “resource commitment game,” which can be modeled as follows: two players each have 10 points, which they simultaneously distribute into three buckets A, B, and C. Then they compare, and whoever wins two out of three is the winner. So for example if one player chooses 7–2–1 and the other chooses 4–3–3, the latter player wins the game since she won both B and C. Any military game where multiple areas are important and it makes sense to split forces potentially has this subgame. At a tactical level, *Diplomacy* is very much like this (at a strategic level, it is of course fundamentally political). Of course, in many military games having the largest army after the clash is most important, in which case each player brings his entire force, and there’s no such army splitting and thus no abstract subgame, at least on this axis. Similarly any game where a player distributes points when designing a character, starship, or other game unit potentially has this abstract subgame. Again, this subgame can disappear if it’s in the player’s interest to invest all or almost all of his points in a single area. If a designer wants to preserve this subgame, it’s important to watch out for this problem.
One can think of sports and other physical games as having physical skill subgames as well. For example, running a sprint, or throwing and catching, can be thought of as subgames of football. Note that many computer games (but few boardgames and card games) have some physical subgames, typically involving reaction time and eye-hand coordination, and some physical games (pool, rifle shooting, darts) have relatively few of the physical skills we associate with sports (again, mainly eye-hand coordination is involved).

**Essential Game**

If the game can be viewed as a disguised version of some abstract game, we'll refer to that abstract game as the "essential game," as in "this game is essentially a chip-taking game." Essential games are most often found in multiplayer games with a lot of targeted interaction: the voting games and chip-taking games discussed earlier.

Note that when people say some game is "really just X" they usually mean to be dismissive. But having an essential game is not necessarily bad. What is important is that the essential game does not make all the other game mechanics superfluous, but instead integrates them into some kind of cohesive whole. Overly political games can fail on this account: how good you are at the various submechanics of the game may not matter, and instead victory is determined solely by the politics. Very often, the essential game of politics can be hard to uncover due to the slow but interesting process of the development of positional and directional heuristics. A game that is essentially a chip-taking game at a perfect level of play can in practice be far from that. Here casual play explains why there are so few "classic" games with a chip-taking essence relative to newer ones. Over time a player community will often be able to uncover the essential game, perhaps leading to its decline if that underlying game is unsatisfying, but a casually played game only a few years old is unlikely to encounter this sort of problem.

**Exercise 4.1:** Why do designers consistently add RPS (rock-paper-scissors) subgames to trading card games or miniatures games?

**Exercise 4.2:** Give some examples of implicit RPS in football.

**Exercise 4.3:** Give some examples of explicit subgames in football.

**Exercise 4.4:** Give some examples of computer games that contain explicit subgames (other than those mentioned in the text).

**Exercise 4.5:** What are some of the drawbacks of explicit subgames? Some of the benefits?

**Exercise 4.6:** Would you expect to see more RPS in a strategy game or a physical skill game? Why? Give some examples of RPS in physical skill games.

**Exercise 4.7:** What are some ways a turn-based game (without simultaneous turns) can include RPS?

**Exercise 4.8:** Would you expect to see more RPS in a simultaneous-turn game or a (nonsimultaneous) turn-based game? Give some examples of RPS in turn-based games. What about real-time computer games?

**4.2 Characteristic: Snowball and Catch-Up**

Many games, especially multiplayer games, have “catch-up features”: features whose purpose is to help
losing players catch up, such as the shells (missiles) in *Mario Kart* that let you shoot at the drivers ahead of you. And many games naturally have a tendency to “snowball”: once you start winning, you win more and more due to your initial advantage, such as the ability of a winning *Monopoly* player to use her money to buy even more advantage. In *Chutes & Ladders*, one can think of the chutes and the ladders as a catch-up feature—the viewpoint is slightly problematic given that it is not obvious the chutes hurt the leader more than the other players, or that the ladders help the loser more than the other players (more on this later), but surely if one player were twenty squares ahead she would vote to get rid of all the chutes and ladders if she could, and the players who were far back would vote against her. And any political game tends naturally to exhibit catch-up in the form of “pick on the leader” and sometimes snowballing as well in the form of “eliminate the weak.”

![Figure 4.2](iStockphoto.com)

But if one thinks about catch-up and snowball features a bit more, it becomes quite tricky. Suppose you’re way ahead of me. But the game has a lot of catch-up features. Then I still have a chance to win. Well, then perhaps you are not so far ahead of me after all. (For example, a twenty-square lead in *Chutes & Ladders* might be basically the same as a five-square lead in the equivalent game without the chutes and the ladders.) Or, if the game has a lot of snowball features, then you are even further ahead of me than it seems. In either case, how meaningful are the ideas of “catch-up” and “snowball” at all? If we both understand the game well, we understand that you are however far ahead of me you truly are, and “catch-up” and “snowball” are illusory.
There is partial truth to this idea that catch-up and snowball are illusory. But it will take us a fair amount of untangling to sort out what is really going on. We will start by looking at some more examples of snowball and catch-up. Then we will give a more precise way of defining the terms, look at how those more precise concepts reveal the illusion, and examine how that illusion relates to the perceived realities of games. Armed with our new (and hopefully more enlightened) viewpoint, we will look at a number of issues surrounding snowball and catch-up.

![Diagram](image)

**Figure 4.3**

Is A really ahead? And if so, by how much?

**More on Snowballing**

Any game in which your score (either in the sense of official score that determines the winner, i.e., victory points, or “unofficial” score in the sense of a simple and easy to use metric, such as money) equals your power tends to have snowballing. *Monopoly* is a classic example; more money helps you to make even more, until you are unstoppable. In no-limit poker, having a high score definitely increases your power. Chess can also be viewed this way, if you think of “score” as your lead in pieces. On the flipside, in go a territorial lead does not typically help you make even more territory. Players often trade away territory for power, in the hope that that power will allow them to get even more territory later in the game.

In most sports, having a lot of points doesn’t really help you score more. You can’t “spend” a touchdown you made in football to get a new player. Boxing is one of the few exceptions (and indeed boxing is also exceptional in having an explicit snowball-handling mechanism: early ending of fights by the referee). In a PvP computer game with experience points and leveling up, snowball is
the rule: win a few fights early, and you will be higher-level and win even more fights later on (\textit{Defense of the Ancients} being an extreme example). In an RTS like \textit{Starcraft}, if one thinks of army size, or army size plus economy size, as a kind of score, then one certainly sees snowballing.

Another natural effect that increases snowballing in some multiplayer games arises from players’ reaction to randomness (although randomness itself is generally a catch-up feature; see below). If a game has a lot of uncertainty, knocking out a losing player can be beneficial to the winning players. His elimination means he doesn’t have a chance to randomly get lucky later and defeat one of the leaders. Poker is a common example: players who are ahead in no-limit poker are typically happy to knock out a player in a worse position.

Snowballing is often considered bad by designers and players. Partly this is just a natural feeling of unfairness: why reward the player who is already winning? A more sophisticated point of view is to think of it as a problem in logical elimination. Nobody thinks it “unfair” simply because a game ends and someone wins it. What’s bad is when someone who has little or no chance to win is forced to continue playing for a long time before the game is over. So a snowball feature that directly leads to the end of the game might be a fine thing (and in fact might not be perceived as a snowball feature at all, but simply as a mechanism for determining when the game is over). Less appealing is a snowball feature that pushes the game to a state where the winner has an even higher chance of winning, with the game still nowhere near its conclusion.

One way to limit snowballing is to unlink power and score, say by adding victory points that are used to determine the winner but that cannot be spent or otherwise used during the game. Sports and European boardgames commonly use this technique. The price of limiting snowballing in this way is often a more complicated game (gold and victory points as in many European boardgames rather than simply dollars as in \textit{Monopoly}). Race games have this feature naturally: you typically can’t “spend” your lead in the race to buy anything, so your lead in the race is simply a specialized and intuitive kind of victory point. In fact, many physical races have something like the opposite of spendable victory points in that not only can’t the leader “spend” her lead, she is often harmed by her lead position due to air resistance and the ability of nonleaders to draft.

Although designers largely tend to look for ways to limit snowballing, there are often good reasons to increase it. If games are dragging on too long, or the game suffers too much from player-elimination issues, adding snowballing effects can be helpful. Rewards for knocking out a player (as in \textit{Risk}) are one example. These are strictly speaking snowball effects—they make losers lose by even more—but it’s often better to have a 0 percent chance to win (and thus be able to go get a cup of coffee) than to have a 1 percent chance to win (and thus be forced to remain in the game with almost no chance for victory). Games that use the “eliminate the winners” mechanic (e.g., a footrace, or the card game known variously as Asshole, President, or \textit{Dai Hin Min}) are similarly snowball games—once you drop out, you are no longer in any danger of being the loser.

More on Catch-Up

The dynamic of the rich getting richer means snowballing tends to appear naturally in games. By contrast, catch-up features are more often deliberately included, and less often appear as natural outcomes of game features put in for other reasons.

There are countless examples of deliberately added catch-up features in games. The shells in \textit{Mario}
Kart fire forward, so the person in the lead can’t make use of them, but people who are behind can fire on those ahead of them. And the Spiny Shell specifically homes in on the person in first place. Some racing games go so far as to speed up the car of anyone who is behind. In Warcraft III, the upkeep tax on large armies is a catch-up feature limiting the snowballing effect of large-army dominance.

As mentioned above, any game with voting or other political features will typically thereby have catch-up, sometimes to the point of making “in the lead” a meaningless concept for much of the game. If someone has a clear lead, it is in the interests of all the other players to stop him. Note that this kind of catch-up is especially agential—one playgroup may think it is fine to pick on the leader, but another may impose limits on it (similarly for political snowballing—various groups will be for or against knocking out weak players, say). But politics is surely a net catch-up feature, since picking on the leader in some form is all but universal in political games.

In general, randomness may be thought of as a kind of catch-up feature. Although a random event may not differentially help losing players over winning players, change in the game state is still appreciated more by the players who are losing than by those who are winning. Reset buttons are one example: if a player is losing in a race game, she is happy to play a card that says “everyone goes back to Start,” or that scrambles everyone’s position around randomly. Note that although scrambling everyone’s positions at random is in some sense “treating all players equally,” it can only help the person in last place and only hurt the person in first. Even a seemingly equitable random jolt like “each player rolls two dice and moves forward that many spaces” is probably better for someone who is behind.

Expansion and Contraction of Win Probabilities

Saying that “each player moves forward a random number of spaces” is a catch-up feature is perhaps counterintuitive (although the fact that the player who’s behind is in favor of it is evidence in favor of this viewpoint). And we still haven’t addressed the question of how meaningful catch-up is once you take it into account: if I’m way behind you in Chutes & Ladders, but I might roll a 4 and land on a ladder and wind up ahead of you, then I’m not really so far behind you after all, am I? To address these issues, let’s try to be a bit more precise about what catch-up and snowball really mean by creating a simple mathematical model of the progress of a game over time.

At any moment in a game, we can write down each player’s chance to win. Typically those chances will start out more or less equal (for a fair game), change somewhat over the course of time, and then gradually shift toward 1 (for the winner) and 0 (for everyone else). If we write the various chances for each player in a row, say for a four-person game that lasts ten turns, we might see something like (0.25, 0.25, 0.25, 0.25) at the start of turn 1, (0.3, 0.2, 0.15, 0.35) on turn 4, and (0.9, 0.03, 0.03, 0.04) on turn 9. We’ll call this list of numbers a state vector. Note that the sum of the numbers is always 1.

If there’s no chance involved at all (i.e., the game is completely determined), then the vector will look like (0, 0, 0, 1, 0)—all 0s for the players who have no chance and 1 for the player who is certain to win.

In a two-player game, if I am 70 percent likely to win at a certain point (perhaps it’s a simple race game and I am eight squares ahead), and then later I am only 60 percent likely to win (perhaps
you've rolled well and I'm only five squares ahead now, then you have caught up. If instead later I am 95 percent likely to win, then that's a snowball situation relative to the earlier game state.

What we're really looking at is the spread of the state vector: as it spreads out, the game is snowballing toward its conclusion. If the player who is behind catches up, the vector will be less spread out. The standard way of defining spread is by the variance: the expected sum-of-squares deviation from the average. The average is just the sum of the values divided by \( n \), so for a state vector \( (p_1, \ldots, p_n) \) the variance is

\[
\frac{(p_1 - 1/n)^2 + \cdots + (p_n - 1/n)^2}{n}.
\]

This number represents how far the state \( (p_1, \ldots, p_n) \) is from the "most caught up state" \( (1/n, \ldots, 1/n) \). Naturally, the state \( (1/n, \ldots, 1/n) \) has the smallest possible variance, namely 0. The largest possible variance belongs to vectors like \( (0, 1, \ldots, 0) \)—the most extreme snowball states.

So we'll define a catch-up event as one that decreases the variance of the state vector, and a snowball event as one that increases the variance.

It should be mentioned that ending state vectors like \( (0, 1, \ldots, 0) \) represent a game with a unique winner. Some games, however, end in draws, thus ending with minimum rather than maximum variance. This also happens in games with scaled victory conditions, for example poker, where the ending state of the players can be any redistribution of the total buy-in one likes.

Note that hurting the player in the lead and helping the player who is behind are exactly the same thing in this model: if we go from \( (0.7, 0.3) \) to \( (0.6, 0.4) \) we have both hurt the leader and helped the follower. Any increase in one player's chances must represent a decrease in the chances of some other player(s). Similarly, in a snowball event, the leader does better as the person behind does worse.

Thinking about games in this manner abstracts away a great many features, but a surprising amount of the flow of the game can be read from the time history of a player's chances to win. Some examples:

Here's a three-person game that progressed in a typical fashion (figure 4.4). One player (the thick black line) started out with a \( 1/n = 1/3 \) chance to win, fell a bit behind, started winning, and then continued to widen his lead until the end of the game.
Figure 4.4

Typical win

In the two-player game represented in figure 4.5 our player took a modest early lead, but eventually lost the game.

Figure 4.6 shows what a player’s history might look like in a highly political game (perhaps the chip-taking game), where none of the early choices matter because an apparent lead results in getting “picked on.” The player represented by the thick black line may have gained a lot of points early on, or he may not have, but none of that affected his chances to win. Each player had about as good a chance to win as any of the others throughout most of the game. The winner of this game was determined entirely at the end.
Figure 4.5
Typical loss

```
1
2/3
1/3
```

Time

Figure 4.6
Only the end matters

This same graph might also describe a game with extremely strong catch-up features—for example, a race game where cars behind the leader are given a significant speed boost.

```
1
2/3
1/3
```

Time

Figure 4.7
Effective elimination

In this unhappy game (figure 4.7), our player (the solid black line) was beaten down about halfway
through the game. At that point, she had almost no chance to win, but she wasn’t actually eliminated until the very end of the game. (Perhaps it was a race game in which she was so far behind that she had no real chance to win.)

**Catch-Up: What Is Apparent and What Is Real?**

One might ask the question of what part of catching up is real and what part is illusory. Imagine a game with an easy first-order state heuristic like a race. The game has a “catch-up mechanic” that helps people who are behind. Players enjoy playing because they feel like they can catch up if they fall back and remember pleasurably the number of great comebacks they have seen. But if players understand how this mechanic works, they should adjust their heuristics and when they evaluate their chances to win (i.e., use their new state heuristics) they will find there is no catching up from behind in the race—in fact there is no falling behind and never was, at least to the extent they once believed. It just means that a large lead on score really represents a small lead in chance to win.

Real catch-up features can come in two types. The first type is features that put some limitations on how big the variance can get (until the very end) and how fast it can get there. The second type is features that tend to reduce the variance in ongoing games by having events that either end the game in favor of the leader or reduce the variance going forward. For a simple example, consider a duel, with two duelists shooting in turn, each with a 60 percent chance to hit (which ends the duel). At the start, the first shooter has a greater than 0.6 chance to win, but if he misses, the other shooter is now ahead. Many games have features that work in this manner: the player who is ahead must press his advantage and attempt to win quickly lest the other player catch up.

It’s true that for games with a unique winner, it is common that catch-up is the apparent catch-up that comes from imperfect state heuristics, not necessarily “true catch-up” (if one thinks of “true catch-up” as variance control). But that’s okay. It is important to remember that since it is rare for players to have perfect state heuristics, it may be true that there is no reason to draw a distinction between a “true” versus “apparent” catch-up feature. When a catch-up feature is put in, by and large what is happening, as we have stated, is partially actual variance control and partially catch-up relative to a particular heuristic—for example, the lead in *Mario Kart*. The feature’s effect is one of muddying the heuristics, but as long as those heuristics don’t change, for all practical purposes the effect is real—the player who thinks he is far behind thinks he is catching up. The danger lies in players developing new heuristics, perhaps seeing that there is no catch-up but instead only a nonintuitive ranking of the leaders, and placing themselves back into the state the designer was attempting to avoid—namely player dissatisfaction with their ability to come back from behind.

Catch-up features can still do good things:

- Sometimes skill can be used to apply the catch-up feature if you are behind (or avoid it if you are ahead).
- Catch-up features allow a nice first-order heuristic (score/position without the catch-up feature considered) and a more advanced second-order heuristic. Since climbing the heuristic tree is a big part of the enjoyment of games, that’s no small thing.
- Catch-up features keep more players in the game in the sense that they have a reasonable chance to win. In other words, catch-up features slow down the spread of the state vector. The catch-up feature may also put a cap on the variance of the vector until just before the end. (To see this is true, just
consider a game with some catch-up feature, and delete that feature partway through the game; now the players' chances to win are as far apart as they would seem with the simpler heuristic, which is further apart than they would be with the more complex heuristic, i.e., with the catch-up feature implemented.) In fact this may be the most important true catch-up function—that at no time until the very end will any player have too great a chance to win. Features like this often work better if player heuristics do not fully take this lack of variance into account, so that players feel exciting “comebacks” are common.

- Catch-up features with the “maybe it fires, maybe it doesn’t” coin-flip type event (the missile that hits or misses) make the point lead more random-seeming, make it change more, and make the typical game more exciting for most players. Pushed to extremes, this can backfire with some sophisticated players, who realize that the point lead is such a bad indication of the true game state that they lose interest in it—a race where nobody cares who is in the lead is probably not an enjoyable game. Worst of all is if there is no useful state heuristic left whatsoever.

- Catch-up features can have the real quality that at any time the state vector has too high a variance, the game will either end or shrink the variance. This effect is very important since generally the harshest problem of having a small chance to win isn’t losing itself, but rather playing a game where one has too low a chance of winning. There are catch-up features that consistently deal with this problem by ending games or making them fairer. Either way is often a gain for the player behind, especially in a noncompetitive game.

Despite all these positive features of catch-up, it is worth remembering that it is often easy to see the bad in snowballing, but not the good (control of game length, better player-elimination characteristics), and the good in catch-up features, but not the bad (poor positional heuristics, relative irrelevance of early-game choices, opportunity for kingmaking because the last-place person has more power, overly long games).

Miscellaneous Catch-Up Topics

Now with more perspective under our belts, we can tackle several issues involving catch-up and snowballing.

Randomness and Catch-Up

Stirring the pot (e.g., resetting everyone's scores to the starting score, or adding a random amount to everyone's score) is somewhat like what people tend to think of as catch-up, and somewhat not: it doesn’t differentially help losers and hurt winners, but it tends in practice to hurt winners and help losers simply because the winners are ahead and the losers are behind. Events of this kind tend to be “catch-up events” in the sense of decreasing the variance in the state vector when they happen (compared to the variance of the vector when they fail to happen). So one can think of random elements in a game as being in themselves a kind of catch-up feature—if a game has a lot of randomness, you are probably not as far ahead as the score (or other first-order state heuristic) might indicate.

Random features in a game often give rise to “press-your-luck” situations: cases where a player can choose to make the game more random or less. Typically a player chooses to make things more random when behind, less random when ahead. Hail Mary passes in football, going for three-point shots in basketball, guessing in Clue, or pushing for a risky Yahtzee combination are all examples, but
perhaps the ultimate example of a press-your-luck game is Sid Sackson’s boardgame *Can’t Stop*.

**Over-Catch-Up**

Sometimes catch-up features are so strong that it is better to be second, or at the very least it does not hurt to be second. Race games with lots of ways to hurt the leader, shoot the person in front of you, speed up if you are behind, and so on, can have this problem. Highly political games tend to be this way, due to the “pile on the leader” tendency almost all playgroups have. Over-catch-up tends to be frustrating—people want to pump up their score, or get ahead in the race, and they do not want to be punished for it. In theory, jockeying for second (or should it be third?) and then jumping to win at the end can be a reasonable game, but in practice it is not much fun if it happens all the time. And games of this type tend to have all the play choices other than those near the very end of the game be irrelevant to the outcome.

A game with this attribute will generally become less fun when players realize it.

**Catch-Up in Very Long-Running Games**

Catch-up can take certain unusual forms in games that go on for a very long amount of time—typically one and a half player games (like single-player RPGs) or MMOs. Long one and a half player games tend to have a great deal of catch-up: if one thinks of the basic metric of player level compared to stage in the game (for an RPG), then grinding is a mechanic that lets the player freely “catch up” anytime she wants. The same applies to an MMO.

The reason very long games require catch-up is that if a game lasts ten hours (say), then without some form of catch-up, a losing player would be clearly losing for the last several hours, which is just too long to be in a state of all-but-certain loss. Of course, if the catch-up features are extremely strong, one gets the problems one often sees in these genres: choices in the early stages of the game may not matter very much, or the player may be discouraged on realizing that a painstaking and tedious method of play is most likely to guarantee victory. One common attempt to solve the problem of catch-up in very long games is to use dynamic difficulty adjustment. This basically amounts to catching up the player invisibly whenever she falls behind, and catching up the AI if the player moves ahead. The problem is that it is rather like your spouse cheating on you: arguably fine if you know nothing about it, but liable to make you feel bad if you do find out about it, which eventually you will (at least in the case of games, given the Internet). Players who are trying to play well want to feel that if they do play well, they will be rewarded. This feeling is hard to come by if the game tries to ensure equal outcomes regardless of player skill.

Even for games with a short game length, one can think of the ongoing metagame as a corresponding game with very long game length. This very long game has some of the same issues—for example, if a dozen people all learn to play chess together, as time goes on their skill levels will spread apart. After a number of months, some of the players may be in a permanently winning or losing state. Those players have (all but) won or lost the have-the-most-skill metagame, leading to a bad play experience. Better players in the group teaching weaker ones provides a sort of catch-up feature, arguably analogous to level grinding (the weaker players are spending more time improving their skills, the stronger players are spending less).
Targeting and Catch-Up

A catch-up feature may hit various targets: it may hurt the leader specifically, the player of your choice, the guy in front of you, a random player, everyone but you, everyone in front of you, and so on (likewise for catch-up features that help a player, although “help yourself” is by far the most common sort). Depending on whom the catch-up feature targets, different gameplay effects can occur. We’ll give just a few examples of problems that can arise, especially if the catch-up feature is too strong.

Hurting the leader often tends to lead to over-catch-up and a “play for second” style of game. Each player hurts the leader, nobody else gets hurt, and thus the lead cycles regularly, but having the lead isn’t necessarily meaningful. There are no choices, so the only skill increase comes from the disguise of the first-order “who’s ahead” heuristic, a heuristic that may be so heavily damaged as to be almost useless.

Hurting your choice of player tends to lead to highly political games. As with any targeting mechanic, carried to an extreme it may result in a chip-taking game.

Hurting someone near you (in whatever sense the game defines “near”: a player sitting adjacent to you in a boardgame, a car driving near you in a racing game) can be good in that it is less political, although of course it does represent some diminishing of player choice. However, such a mechanic—say the Green Shell in Mario Kart (which, being unaimed, typically is used against players who are close)—may not give large-scale catch-up. Instead, it may cause clumping: groups of players who are close together keep shooting each other, forming clumps, but one clump can’t affect another far-off clump (although occasionally a player will break away from one clump and push ahead or fall behind until pulled into the orbit of another clump). In this sense Mario Kart is almost exactly like a large bicycle race, with the Green Shell playing the same role as drafting: something that pulls together nearby vehicles but does not affect faraway ones. They are a catch-up feature within a given clump, but less so when viewed from the point of view of the race as a whole.17

Conclusion: Limitations and Effects of Catch-Up

The limitations presented on catch-up are interesting for both player and designer. The most critical of these is the idea that catch-up relative to a fixed state heuristic is real even when the situation relative to winning percentage is not. Any time a player is in a game that he perceives to have a catch-up feature, there is some indication that his state heuristics may be insufficient and there is a possible gain in strategic understanding to be had by altering them. Similarly the designer needs to take care that adding a catch-up feature relative to a state heuristic continues to serve the basic intention of keeping players excited and hopeful without eliminating some core element of the game. It is probably true that you don’t want to naively add a catch-up feature to maintain player hope at the expense, for example, of actually wanting to be in the lead in a race, or gather the most power in a political game. While the simple state heuristic of the lead being good in a race is arguably more important than keeping all players involved, there is often a lot of leeway for players maintaining naive state heuristics depending on the player audience. It is much more likely that adding a catch-up feature to a race for children or casual players will be seen as a true catch-up feature relative to the lead heuristic, rather than causing a shift in player heuristics to devalue the lead in favor of a more complicated formula. Even for a more hardcore game, features like dynamic difficulty adjustment may
cause players to believe they “caught up” due to good or lucky play when in fact the existence of the feature meant they were never really behind. This situation can break down in a game meant to be highly replayable as players refine their state heuristics. Few players are interested in a game where they see they always have a 50 percent, or even 95 percent, chance of winning no matter what they do.

True catch-up in the sense of limiting state vector variance has an important place as well. Very often this will achieve the goal of continuing player involvement while maintaining clean first-order heuristics. The difficulty here lies in the potential to disenfranchise competitive players who may feel slighted they can only be a limited amount better than a truly bad opponent. Again the audience is the key. A feature that limits a player’s downside to 1 percent of the leader’s chance to win may not go far enough to keep many people interested in the game. Conversely, a feature that sets that limit at 40 percent may scare away more competitive players.

Catch-up features that either end games or make them fairer can work especially well in achieving the basic goals for player hope in a long game. They may have the tendency, however, to create a lack of control over the length of the game, since by their nature they achieve their leveling by ending some games early. Still, it is encouraging for many to know that there are no bounds on how good they can get at the game while at the same time worse players will never have to be in a game they feel they can’t win for long.

Exercise 4.9: Give some examples of pressing your luck in baseball and hockey.

Exercise 4.10: Give some examples of pressing your luck in a chess tournament.

Exercise 4.11: What types of audiences would be more interested in games with catch-up features?

Exercise 4.12: Do games with catch-up features tend to have poor or good state heuristics? Why?

Exercise 4.13: What are the risks of dynamic difficulty adjustment? Do all the risks go away if no players know that the difficulty is being adjusted?

Exercise 4.14: For a game to have a “true” catch-up feature what needs to happen? How might this be beneficial for the player audience? (Hint: Think about the elimination qualities inherent in such a game.)

4.3 Characteristic: Complexity Tree Growth and Game Arc

Game Complexity Trees

One can think of a game as a series of choices. In fact, the game designer Sid Meier famously defined a game as “a series of interesting choices.” There are certainly games with no interesting choices, and in fact examples of games without any choices, but these tend to be limited to the sphere of gambling. For logical completeness, one would typically consider all possible choices at any given node (decision point). But from the point of view of human players, what matters is the number of meaningful choices. “Meaningful” is of course an inherently agential concept: for the exact same game state, a beginning player might be choosing at random (no meaningful choices), an intermediate player might feel pressure to examine a great many choices, and an expert might need to examine only a few. Imagine, for example, the number of meaningful choices for players of various skill levels at tic-tac-toe or chess.
Exercise 4.15: How many meaningful choices are there for a beginner at tic-tac-toe? For an expert? What about a beginner at chess? An expert?

Exercise 4.16: Pick two different games and give a reasonable estimate of how many choices there are at any point. Do this for both beginner and expert players.

If the number of choices is too low, you hardly have a game at all (Conway's Life, or tic-tac-toe or Nim between expert players) or you have a boring one from a strategic point of view. If the count is too high, you may have a game that is not fun for most people (go, or an RTS).

We speak of the game tree as being sparse if there are relatively few choices, and bushy if there are relatively many. Of course these terms are not absolute, but relative (say to the number of choices likely to be enjoyed by a particular audience, or compared to other games in the same genre, or compared to some particular other game). Often we compare the sparseness or bushiness of the game tree at one point in the play of the game, say the opening, with another point, say the endgame.

Bushiness and sparseness are largely systemic—how many legal actions are there at a certain point in the game tree? But they are also somewhat agential, because players will not necessarily consider all possible choices, but will prune the game tree by eliminating certain choices from consideration. Pruning can come from eliminating duplicate moves (i.e., recognizing symmetries) or from quickly recognizing some moves as bad ones. Pruning can also come from the recognition that some choices are effectively though not exactly the same—for instance, the exact spot to stand on a large playfield.

Games that are very sparse or very bushy can be unsatisfying to players. Too sparse, and players may not feel they have any interesting choices to make—in the extreme case, where a player always sees a single correct move (as in Nim or tic-tac-toe, say), one might hardly feel there is a game there at all. Too bushy, and the player may feel overwhelmed by all the choices and may feel they are left making moves essentially at random (as happens to beginners in go, or in building certain complex
RPG characters.

The Game Arc

The game arc is how the bushiness of the tree fluctuates throughout the course of the game. Commonly the tree will start out sparse, get bushier, and then become sparse again. This is basically a good thing—it provides the game with good pacing, and a kind of narrative arc. A typical example would be Monopoly: initially one simply rolls the dice and makes relatively few decisions (it’s often right just to buy whatever one can at the start, and houses, hotels, and mortgaging properties are not relevant in the early game). In the midgame one is deciding on trades, deciding how to invest in one’s properties, and in general making most of the decisions one makes in the game. Toward the end things typically settle down again, and one rolls to see who lands on whose developed properties, along perhaps with continuing to develop the properties one has decided to focus on. An extreme example would be the card game Oh Hell, where the expanding and contracting of the game tree is built in directly, as the hand size (and thus the bushiness of the game tree) changes from round to round.

One can use diagrams showing the relative number of choices available over the course of the game as a visual aid to help one understand the game arc. So, for example, a classic game arc like that of Monopoly, where one starts out with a modest number of choices, those choices increase, and then finally the choices decrease until the game ends, would look something like the arc shown in figure 4.9.

Note the rough resemblance to pictures of a narrative arc, with an opening, a climax, and a denouement. Many, perhaps most, boardgames fall into this pattern. An RTS can as well, with basic opening build patterns, a complex middle game, and relatively fewer choices at the end (perhaps because a max tech level has been reached, or perhaps the definitive battles have been fought and a mopping-up phase has been reached).

![Figure 4.9](image)

Classic game arc
Even a crossword puzzle might fall into this pattern, with some easy clues answered early on, then a number of tough clues that don’t cross many other completed answers needing to be tackled, with the last few clues coming fairly quickly as the final portions of the board are filled out. Similar remarks can be made for card solitaire.

Sometimes a particular game, say of chess, may end suddenly, before the bushiness decreases. In this case the overall game arc rises as the bushiness increases, and then suddenly terminates, with no “calm” phase at the end (figure 4.10).

Beginners may experience the game arc somewhat differently. To them, the opening phases may seem quite confusing—they haven’t learned the standard opening patterns, and so may need to think as hard about the opening as about the middle game, leading to an arc somewhat like the one in figure 4.11.

One might think of the arc in figure 4.11 as representing a beginner’s chess game, say. There are slightly fewer possibilities in principle for the opening moves (certain pieces simply cannot be moved because they are blocked). Still, there are many moves to consider—for instance, opening rook pawn moves forward two—that an expert can simply ignore. If one thinks of go, the situation for the beginner is even worse. Even the theoretical number of opening moves is larger, and the end does not collapse much either—in fact, when the game is over is not terribly clear, giving rise perhaps to a diagram as in figure 4.12.

![Figure 4.10](image)

A game ending suddenly
Games in which one places a large army on the field that is then gradually destroyed (*Risk, Warhammer, Myth*) often start at maximal complexity and gradually decrease (figure 4.13).

Some mini games or wargames have rules for reinforcements, which is a way of getting closer to the classic game arc: more choices for later on, with a somewhat simpler starting position.

Sports often have a more or less steady game arc. Basketball is not noticeably more or less complex...
in the first quarter than in the third (perhaps slightly more right at the end due to management of the clock) (figure 4.14).

Any game with very short atoms will also tend to have a flat arc like the sports one. That’s because there’s not much room for an arc inside such a short atom, and each atom being the same means the overall game arc is flat. Another example of this flatness of arc can be seen in a tournament (thought of as a game in and of itself): what a player is doing in the first hour of the tournament is not that different from what she’s doing in the fifth hour.\textsuperscript{22} Flatness of arc might be one of the few disadvantages to the otherwise strong game feature of small atoms.

For a paper RPG or an MMO, one has to decide what time period one is looking at, since the game length is not well defined. Looking at, say, an MMO over the lifetime of a single character, one might say the game arc gradually increases in bushiness (higher-level characters have more options and more difficult challenges) until raiding starts. In most cases, raiding involves fewer real choices for an individual character, and indeed the game arc can sometimes get very sparse. The raid leader, however (who is managing the raid as a whole) is typically engaging in a very complex task. His game arc might show an increase in bushiness.

![Complexity progression in a miniatures battle](image)

\textbf{Figure 4.13} 
Complexity progression in a miniatures battle
Figure 4.14
Sports—roughly constant complexity

Figure 4.15
MMO—lifetime of one character

Thinking of a single MMO raid as one event, it often follows something like the classic arc: not too many decisions early on (gather, buff everyone), some choices further on culminating in a boss fight, after which things are quickly over.

The one-session measure coming from a raid is probably the more important one: that's where the narrative-like satisfaction of increasing tension followed by a denouement is at its best.

As mentioned above, some games, like miniatures games and chess, violate the usual game arc
pattern in that they start with a great many pieces on the board and then gradually shrink as those pieces are destroyed—in other words, their game tree appears to start maximally bushy and then decrease. Sometimes, however, the early phases of the game have fewer choices than it might appear: in a minis game, the pieces start far apart, and the early moves may consist largely of closing the distance, which is relatively simpler than the midgame. (Although this is true for average minis players, for very expert players the early maneuvers are extremely complex and telling, so the game arc pattern of maximal and decreasing bushiness is probably closer to the truth.) In chess, the mobility of pieces is restricted early on, and opening patterns are available to tell you what to do, thereby narrowing the meaningful choices.

![Graph showing bushiness over time.]

Figure 4.16

MMO—a single raid

Many, perhaps most, games are like chess in that the opening moves become at least partially standardized over time. For example, think of the standardized opening build patterns in an RTS. These openings have the effect of pruning the game tree during the early game for nonbeginners, but beginners are often more at sea in the early game. Once again this points out the value of good zero-level heuristics, for without them beginners will have a worse game arc than more experienced players.

Some games, like RTS games and trading card games, have a very explicit early arc built in: many game options are simply unavailable early on and must be slowly enabled. Games of this sort may have the later arc extremely bushy, and sudden endings are possible: a Magic or a Warcraft III game may end suddenly, with the game tree still uncollapsed.

One large and important category of games that do not follow the standard game arc well is sports (and computer games based on the sports model, like Mario Kart or Street Fighter). Although there is sometimes a collapse at the very end, due to time running out or to the outcome being inevitable, by and large the sorts of decisions players make at the beginning of a soccer game, auto race, or tennis match are similar to the ones they make during the middle or toward the end.

Exercise 4.17: Think about the complexity tree for Nim. What does it look like for a beginner? For an
expert? For an intermediate player?

**Exercise 4.18:** Draw some parallels between the “classic” game complexity arc and the standard three-act narrative structure common in plays and films.

**Exercise 4.19:** What drawbacks for the player might a game with many options have?

**Exercise 4.20:** Give an example of a game where the number of meaningful options increases with player skill.

**Exercise 4.21:** Construct an example where adding an option to the game decreases the number of meaningful options.

### 4.4 Characteristic: Game Balance and Strategic Collapse

Often players and designers worry about game balance or criticize a game for being unbalanced. Lack of balance can be thought of as a problem with the complexity tree: the bushiness of the tree is much less than intended because a small number of strategies are much better than the others. Phrased another way, the game seems to be offering a large number of interesting choices, but on closer examination it turns out only a few of them are viable, so the fun of the game is not realized. Players just use those few choices again and again. The heuristics of the game have been reduced from the potentially rich and satisfying collection that might have existed to a very simple heuristic: play only with the much better strategies.

Phrasing the problem in terms of complexity trees lets us make some additional observations we might otherwise have missed. First of all, saying the tree’s bushiness is less than “intended” raises the question of intended by whom. The answer is some mixture of designer intent and genre convention (player expectation). For example, in an RTS players expect that almost every unit has some use. If one can always achieve victory by building nothing but one particular unit, that game is considered unbalanced. In fact, players and designers often feel so strongly about this problem that they will refer to the game (or the unit) as “broken,” meaning that once you discover the strategy of just building that unit, the game is no longer working as intended, to the point where it is analogous to a broken machine. If a unit in an RTS is so bad that one never builds it, that also fails to meet players’ expectations (although the game as a whole is rarely declared “broken” merely because of that—a single bad unit does not cause the decision tree to collapse the way a single very good unit can, so the rest of the game remains reasonable). Most RTS games have a modest number of units, and the conventions of the genre are such that players expect, or at least hope, that all will be useful, at least in some situations. Designers generally try to adhere to this expectation, so the “intention” is indeed some mix of player expectation and designer intent.
In an MMO, on the other hand, there are an enormous number of items, and the genre convention is that most of them are not that good compared to the best items. Because this is expected, players and designers do not usually consider it a problem that many items are so bad that one would never want to use them. However, items that are so good that they are the only thing one would consider using (in that particular item slot) are sometimes considered “broken.” The ideal, perhaps, is that of all the items for a given slot, there will be a handful that are reasonable to decide among, and a large number that are not worth considering. The end result is that the question “What item do I put in this slot?” has a reasonable bushiness, at least for those in the know. (Those not in the know are probably not choosing from the entire set of possible items, but just from some smaller subset available at their level of play, so the bushiness of the decision node is reasonable for them as well.\textsuperscript{24}) Again, what is “intended” is a mix of player expectation and designer intent.

In other words, the amount of balance expected in RTS units and in MMO items are examples of standards. Of course other genres have widely differing standards of balance—for example, those in Magic or other trading card games.

Saying imbalance is a problem with the decision tree’s bushiness also reminds us that imbalance must be relative to some set of heuristics, because the tree’s bushiness for a given player is dependent on the heuristics that player is using. Thus balance is not only an expert phenomenon. It is perfectly possible for an imbalance to occur at the level of beginner heuristics: some strategy may be
executable by a beginner, and all but unbeatable by the things a beginner can do in response. One example might be the strategy of rushing in an RTS where rush defense requires at least an intermediate level of skill. An imbalance at the beginner level is bad—it may cause the beginner to quit before gaining a higher level of skill—but at least there is some way out, namely to become a more expert player. Moreover, small advantages of one strategy (or unit, or item, etc.) over another tend to be more telling at an advanced level, so serious imbalances are more likely to occur there. All this means designers tend to worry more about imbalance at the expert level, which is quite natural. But a bit of worrying at the beginner level is also appropriate.

**Options Offered, Options Received**

Good game designers have some idea of the choices players might make, and they try to offer enough choices to make the game fun and interesting.

But because of the risk of imbalance, there is a paradox that offering more options does not necessarily lead to the player having more options. At least for players in the know, the real options are the “good” ones, and adding an additional bad one doesn’t really increase the available options. Adding one that’s much better than the others is even worse—it destroys all the previous options as viable choices. Think for example of an MMO where one is faced with a choice of six character classes: add a new class that is not as good, and the expert player still has six choices. Add a new class that is better, and she now has one choice (i.e., no choice at all). Beginners might still have choices to make, but they will not be so happy when they eventually learn they made the “wrong” choice. And they surely will learn if the game has multiplayer play, although a beginner who has made a “bad” choice might never know in a single-player game (assuming he never goes online for hints or tips and tricks, and he does not learn too much over the course of the game). But even then the designer is left with a quandary: Are the other parts of the game (e.g., the monster difficulty) balanced for the player making the “right” choice (destroying the poor soul who’s chosen unwisely) or for the player making the “wrong” choice (boring the player who has chosen well)?

Balance problems can occur at the strategic (large-scale) level, like picking a character class or a style of Magic deck. They can also occur at the tactical (small-scale) level, like picking an item for your character or a card for your deck. And imbalances don’t have to be item-based: they can also be choices in game actions you take, such as disc jumping in Tribes or rushing in a badly balanced RTS. For another example, expert players in tournament no-limit poker can find themselves playing against less skilled players who go all in when they have a good hand and drop out otherwise. If this strategy works too well, then the rich strategic area of postflop play suffers—and for that reason there are experts who prefer pot-limit poker, which they would view as “more balanced.”

Just as adding an extra choice (such as an unbalanced RTS unit or RPG character class) can actually reduce the real choices in the game, eliminating a choice can sometimes increase the number of real choices. In Magic, a low-level example would be banning a specific overpowered card; a high-level example would be the color wheel itself (which restricts the cards you can play with in any given deck, but increases the number of viable decks overall).

**Kinds of Imbalance**

A gameplay choice can fail to be balanced in roughly three different ways, in increasing order of
severity:

1. It can be too weak. As discussed above, sometimes this is okay, or even desirable (to keep the decision node at a reasonable size). What’s important are not violating standards for no reason and having the right number of choices remaining when the weak ones are discarded.

2. It can be too strong, to the point where it drives out other choices of an equivalent type, but not all choices in the game. An example might be a flying unit in an RTS that’s better than all the other flying units. If you build a flying unit, you’ll build this one, but you still have the choice of whether to build a flying unit, and when you build nonflyers you have choices of which ones to build (assuming they are still balanced with each other). In an MMO, a helmet that’s better than any other helmet (at the level cap, say) might drive out all your other helmet choices, but you still have choices for all your other item slots (and choices of character skills, which abilities to use in combat, and so on). In Magic, there might be some two-mana green creature card that every green deck will want four copies of, but there are still all the nongreen decks, and even the green ones have lots of other cards to choose from for the other slots.

3. It can be so strong that it drives out a huge number of other choices, perhaps to the point of determining the winner of the game. In a badly balanced RTS, you might have a single unit that’s so good, your best strategy is to build nothing but that one unit. In an MMO, the “Helmet of Always Winning” would be so powerful you wouldn’t care about your other equipment choices—or a spell that did far too much damage might be the only spell you ever bothered to cast. In Magic, a blue card so powerful that everyone played blue, and games hung on who drew that particular card first, and the other cards in the deck all went toward supporting that card, would turn a game of making choices in deckbuilding into a game with one choice (i.e., no choices): play the broken blue card.

Games with Explicit Costing Systems

Some games have explicit systems built into them that require a player to pay an in-game cost in order to use some object in the game (by “objects” we mean a collection from which a player chooses—cards in a deck, spells an MMO character can cast, units in an RTS, equipable items in an RPG).26

Such games tend to have two properties: there are many objects in the game, and players choose from among those objects as part of creating their strategy—they create a deck, or a character, or an army. Typical examples are trading card games, plastic or metal miniatures games, RTS games, and MMOs. Such games are a fairly modern phenomenon, perhaps because to reach a large enough player base they more or less require mass production and large amounts of leisure time. The earliest examples are probably military simulation wargames, which date back only to the eighteenth century.27

Games such as these, with their large number of objects and player freedom to choose among them, present special balance problems. In chess, the queen is better than the other pieces, but each player only has one and is not allowed to bring more. If a player is allowed to bring as many queens as he wants, something must offset the queen’s power. That something is the explicit costing system.

So the first advantage of the explicit system is that it allows units to be of different power levels without the game being unbalanced—the game simply needs to charge more for the more powerful objects. This cost differential is what allows a game like Starcraft to have both battlecruisers and
marines, even though the former are much more powerful than the latter on a unit-by-unit basis.

A second advantage of an explicit costing system is as a balance aid to the designer. By separating cost from effect, the designer can choose the effect she wants based on criteria other than balance, and then tweak the cost separately until the balance is right. This is much easier than trying to tweak a whole number of different effects on an object until that object is perfectly in balance with other comparable objects.

On the downside, the explicit costing system typically adds to the complication of the game. And it never works perfectly in a game with a very large number of objects. Such games always need adjustments if the competitive pressure is large, lest the game suffer some amount of strategic collapse. There are always cards in Magic that need banning, and there are always balance fixes needed in a large MMO. A game with fewer units, like an RTS, can eventually reach a more or less stable state, but even that takes an enormous amount of time and energy if one desires a level of balance suitable for competitive play.

Implicit Subgames as Balance Mechanisms

No one needs to worry about strategic collapse in rock-paper-scissors. All three choices will be constantly viable. Even if one were to construct a version of RPS where Rock had a 10 percent chance of beating paper, the game would still contain all three choices as viable options, albeit ideally chosen at different rates than the standard game. There are many balanced implicit subgames whose existence can help to guarantee a minimum degree of choice for the player. These can be built in somewhat explicitly (e.g., color A beats color B beats color C beats color A in a trading card game) or can be included more subtly as a general strategic truism (e.g., ranged units beat slow armored units, which beat fast melee units, which beat ranged units). By building in many copies of these uncollapsible games, the designer is able to set maximum levels of strategic collapse. As in political games, which tend also to be inherently balanced, the art of the designer is in making these subgames not overly obvious or dominant so there is still much play left in climbing the heuristic tree.

Exercise 4.22: For a general manager in professional sports, what prevents strategic collapse in terms of team construction (i.e., think of constructing the best possible team roster as a game in itself—what prevents strategic collapse in this game)?

Exercise 4.23: What are some rules and restrictions added to basketball after 1945 intended to prevent strategic collapse?

Exercise 4.24: What are some drawbacks of eliminating certain cards, miniatures, or digital objects from a game’s play environment, when done to prevent strategic collapse?

Exercise 4.25: Does every character class in an RPG need to be played equally to avoid strategic collapse? In a game with four character classes, what usage data for the classes played should cause a designer to worry? A game with ten character classes? With fifty?