Preemptive Nav: Setup

(Forsenlawless)

Resources

The practice map:
https://www.mediafire.com/file/lzbecct99g8p22i/Lama%2527s_Practice_Map.zip/file
Bolan's stronghold practice mod: https://github.com/mjtb49/StrongholdTrainer/releases
Logwet's blinded mod: https://github.com/logwet/blinded/releases/tag/v2.4.3

Introduction and settings

Most of the time, preemptive is very simple: the biggest orange + green spike in any stronghold will be the portal room (with the correct settings). However, this will not work all the time, and for some hardware it is a poor recommendation. The purpose of this doc is to outline a more precise method for all hardware.

These settings/considerations are essential:

- In the practice map, use 8 render distance, 150% entity distance, and any FOV.
- *In real strongholds*, use 8rd, 50% entity distance and 30 FOV.
- Make sure multiple spawners aren't on the screen at once.
- If you stand too close to any spawners mobs will spawn; an easy way to eliminate the undesirable mobs is by using /kill @e[type=minecraft:mob name]
- Before you break a block, a highlight appears around it indicating you can break it. Make sure this doesn't happen when you're doing preemptive.

How to use the practice map

1. By standing on the corresponding blue lines, measure each of the three glass cages with root.gameRenderer.level.entities.

2. For each cage, write down the *left column* % values (*not the right column as you would for mapless*) of both "blockentities" (the orange) and "unspecified" (the green). We call these three *pure readings.*

   Most often, these pure readings will be the way you identify the portal room. In practice, when both blockentities and unspecified % match with what you have measured, it is a reliable signal. They don't have to match identically and rarely will - expect %s that are slightly higher or lower than your test values.

3. If your *center blockentity* + *unspecified %* is approx equal to 50%, follow the method here. It is better. You can also try this method without the required hardware, although it is less likely to work.
   https://docs.google.com/document/d/10U2dP9V2PAPyleQ0jSIheIR_ZIBnpgbE-4iWdC5P-Xk

4. If not, spawn a single skeleton inside the *leftmost* cage and perform another measurement. This time, *add* the % value of blockentities to the % value of unspecified, and note the result. Do the same with 2, 3, 4 and 5 skeletons. These are our *mob readings.*

   In a real stronghold, if E is equal to 1 use the mob reading for one skeleton, and *a valid signal is anything approx equal to or higher than the mob reading in the practice map*. The same applies for E equal to 2, 3, 4 or 5. Keep in mind that despite this, for pure readings, valid signals are only those *approx equal* to the practice map readings.

5. Measure the spawner false positives behind the cow pen. You will notice that (with the exception of the zombie spawner) the added value of blockentities and unspecified is higher than any of the 3 cages. So, if the values you notice are “too high” in a real run, it is likely a false positive. (However, note that false positives can overlap with the silverfish spawner - so if there is a spike nowhere else, it’s best to assume the false positive does in fact overlap.)

For the curious

Why are there cows in the practice map?

No this is not an epic prank, mobs that aren't rendered but still exist in the world affect the values of gameRenderer. So to closely approximate realistic values of pie we need some mobs in our map.

Why skeletons?

Obviously, we're trying to account for the effect all mobs have on gameRenderer, so we choose skeletons to simplify the test, because it seems their values are fairly representative of a lower bound for the possible effect of mobs.
Actually, spiders measure less than skeletons, so this isn't true; but it seems that many spiders alone reduce the reading for spawners so much as for them to be indistinguishable from chests. This means that by following the skeleton method you can miss portal rooms, but it is very rare indeed that you will only have many spiders and no other mobs in your vision.

Why are the chests slightly moved in the left and right cages?

Because why not? No, we need to account for what I will call (with flair) chest relativity, which is perhaps the most bizarre part of all of this. Basically, depending on where chests are relative to the spawner, the values of unspecified and blockentities will differ. It's strange, and I can't think of any convincing explanation for this other than potentially the spawner being loaded first or the chest being loaded first affecting pie.

Does this detect chests?

No. Chests consistently measure lower than even a single spawner, which is why this works.

Then why doesn't this work sometimes?

If you incorporate everything, you will get a high success rate - in about 85% of strongholds, it will correctly identify the portal room (does not mean it will identify the optimal path to take at this rate). In the 15% or so failure cases, there are a few common reasons why it fails, listed here in order of frequency:

- Mineshaft bait: A false signal from a cave spider spawner. If the stronghold clearly intersects with a visible mineshaft in the first room, best practice is to skip preemptive nav.
- Dungeon bait
- No-chunked: Sometimes the chunk where the portal room is will fail to load, even if it is within range.
- Out of range: Very rarely, the portal room will be too far to be detected.