Pokemon Nuzlocking Pivot Chains

Trevor Squires

July 22, 2025

1 Introduction

Pokemon Nuzlocking is a popular variant of the franchise aimed at creating more of an artificial challenge for what would otherwise be a relatively easy game. Nuzlocking goes by many forms, but often centers around adding additional rules to an ordinary mainstream Pokemon game that signficantly increase the difficulty. Although these are self-imposed rules and entirely optional to the player purely for the sake of having fun, there have been some 'standardized' rulesets developed among the community. The style recently exploded in popularity and has spawned many fan made games, content, and problems to solve.

The problem we seek to analyze in this document is directly the result of two rules that will appear in virtually any Nuzlocke variant:

1. Once a Pokemon faints, it cannot be revived and must sit on the sidelines for the remainder of the game
2. Pokemon battles must be played on set mode - not switch

For the uninitiated, Pokemon battles involve multiple trainers sending out Pokemon of their own to face off against each other. It is turned based style gameplay with each Pokemon getting an opportunity to do some action on its turn. The goal of the Pokemon battle is to knock out (or faint) all of the opposing Pokemon. Whoever does that first, is considered to be the winner.

These two Nuzlocke rules add significant difficulty to battling. For one, Pokemon fainting is a common occurence in battle. In fact, it has to happen for the battle to conclude. In a standard 6v6 Pokemon battle, at minimum 6 Pokemon must die. An 'even' fight might see 8-10 total faints. Futhermore, there are often 50+ total battles to partake in during a Pokemon playthrough. If, on average, three of your Pokemon are fainting (and therefore cannot be used again) in every battle, then you're going to go through hundreds of Pokemon. This is of course problematic when there are only maybe 50 possible encounters in the game. Pokemon Nuzlocking is primarily a game about preserving your Pokemon across all the battles.

The second rule makes that preservation particularly difficult. When the game is played on set mode, switching your Pokemon can only be done using your Pokemon's 'action' for that turn, i.e. it wastes a turn. Pokemon battles are a game of matchups. Pokemon A might be strong against Pokemon B, but is weak to Pokemon C. Pokemon B might be quite strong into Pokemon C, however. So rather than filling your team with the strongest, it's more a game of positioning yourself into good matchups. When a Pokemon faints an opposing Pokemon, a replacement is allowed to be sent in. Most often, this replacement is often one that is strong against the one who fainted the original Pokemon. Thus, it is often smart to switch out of this now-advantageous matchup. The only consequence of doing so is that you allow for a 'free' turn to the opponent while you make the switch. Minimizing the damage you take during this free switch turn is a crucial part of keeping your team healthy and thus a crucial part of nuzlocking.

2 The Art of the Pivot

One of the cleanest and most natural ways to minimize damage is to take advantage of the type chart. Each Pokemon has one or two types and each attack has an associated type with it. There is a type interaction chart that determines a damage multiplier when type A attacks type B. For example, if water attacks fire,

the damage multiplier is 2x. In the reverse situation, however, the damage multiplier is 0.5x. Thus, a very natural way to reduce the damage taken during a repositioning turn is to make sure you reposition into a type that is going to take as little damage as possible during the switch turn. This sounds (and can be) difficult in practice, but we can abuse the often deterministic decision-making of the opponent. Without going into too much details, we'll assume that the opponent is always going to go for the highest damaging move against our Pokemon.

Let's see how this can be abused in an example. Assume we are in a Pokemon battle where our rock type Pokemon has just fainted an opposing Pokemon and a dragon/ground type is coming in to replace it. We have a Pokemon of type water/fairy left that we can switch to which is able to kill the dragon/ground with a fairy type move (since it will do 2x damage), but if we switch in while the opponent is using a ground type move, we will take too much damage and faint. If the opponent was instead using a dragon type move, we could come in freely since that move does 0x damage to water/fairy. So instead of switching directly to our water/fairy, we need to switch to something that

1. Takes reduced or zero damage from ground type moves
2. Takes additional damage from dragon type moves so that we bait a dragon type move next turn

By switching to a dragon/flying type first, we take 0 damage from the incoming ground type attack, then switch into our water/fairy to again take 0 damage from the now dragon type attack. Our dragon/flying type Pokemon is our **pivot**. They did not take part in this fight, but existed to manipulate the opponents decision-making into a favorable one for us.

Pivots are an essential to difficult nuzlockes because they enable you to leverage the type chart to a great advantage. Even average Pokemon are quite powerful in the right matchup, and pivots help you position into those matchups.

3 Resistance Complete Pivots

Unfortunately, not every pivot is always available to the player at every point in nuzlockes. Although in theory there may exist a perfect pivot to what you are trying to accomplish, it may just not be feasible given the resources at your disposal. Futhermore, some pivots require hefty investment. As we saw in the above example, to defeat the dragon/ground type, we not only needed a Pokemon with a good matchup into it (our water/fairy), we also needed a pivot to get into the good matchup. If all 6 of your opponents Pokemon require a dedicated pivot of your own, you're quickly going to run out of space in your limited 6 slots of Pokemon you can bring. Thus, there exists the need for role compression among pivots. Ideally, you want Pokemon that can not just be used a pivots, but also take advantage of other Pokemon as pivots and be used themselves.

Perhaps the strongest form of role compression is a pair of Pokemon that completely cover each other's weaknesses. For example, if Pokemon A were weak to water and fire, but resisted grass a perfectly complementary Pokemon B would be one that is weak to grass but resisted water and fire. This way, if Pokemon A is ever facing a bad matchup, Pokemon B can easily switch in and take reduced damage and vice versa. A popular nuzlocke content creator, Pokemon Challenges, refers to this concept as resistance complete. A pair of resistance complete Pokemon can be extremely valuable to a nuzlocke team composition because they offer a perfect complementary mix of offense and defense without additional help.

However, resistance complete need not be limited to only two Pokemon. We say that a set S of Pokemon is **resistance complete** if for any Pokemon A in the set, there exists another Pokemon B that takes reduced damage from all of Pokemon A's weaknesses and there is a Pokemon C for which Pokemon A resists all of Pokemon C's weaknesses. This is merely a fancy way of saying any Pokemon in the set can be safely switched out of and in to. For example, dragon/ground + flying/water + normal/steel is a resistance complete set of size three. Dragon/ground can switch into normal/steel to cover its ice, dragon, and fairy weaknesses. Normal/steel can switch into flying/water to cover its ground, fire, and fighting weaknesses. Flying/water can switch into dragon/ground to cover its rock and electric weaknesses. The trio offers perfect coverage and can generate a type advantage on nearly half of all attack types.

Building a team around a resistance complete core can check off a lot of team readiness boxes and is a great way construct a robust team.

4 Computing Resistance Complete Sets*

This section is entirely optional for those only interested in the results. While resistance complete pairs are easy to compute, looking at entire sets is a combinatorial nightmare. For just pairs, there are only around 12k type pairings to look at and these can easily be iterated over. However, there are nearly 700k sets of size 3, 26M of size 4, you get the picture. Manual enumeration is not going to be very feasible. Instead, I built a mixed integer program that can compute all the resistance complete pivot chains up to size 4 in only a few seconds.

4.1 Model Formulation

To identify viable *resistance complete sets* - sets of typings that mutually reinforce each other's resistances - we formulate a feasibility-based mixed-integer program (MIP). The search space is first pruned to include only those typings with at least one mutual coverage partner. That is, there are many dual typings that don't have any typings who completely resist them. Thus, they can be excluded from this analysis as they will never be part of any set.

Decision Variables

Let $\mathcal{T}$ denote the set of valid typings under consideration. Also let us define the sets $B(j)$ to be the set of types that completely resist all of types j's weaknesses. For each typing $t \in \mathcal{T}$, we define a binary decision variable:

$$x_t = \begin{cases} 1 & \text{if typing } t \text{ is included in the pivot chain,} \\ 0 & \text{otherwise.} \end{cases}$$

Model

$$\begin{aligned}
&\text{Minimize:} && 0 \quad \text{(dummy objective)} && && \\
&\text{Subject to:} && \sum_{t \in \mathcal{T}} x_t = k && && \text{(chain length)} \\
& && x_i \leq \sum_{j \in B(i)} x_j && \forall i \in \mathcal{T} && \text{(coverage requirement)} \\
& && x_i \leq \sum_{j : i \in B(j)} x_j && \forall i \in \mathcal{T} && \text{(mutual coverage)} \\
& && x_i = 0 && \forall i \text{ with no mutual partners} && \text{(prune isolated typings)}
\end{aligned}$$

Explanation of Constraints

- **Coverage constraints:** If a typing i is selected, at least one of its "resistor" buddies must also be selected. This ensures that each included typing is protected by another.
- **Mutual coverage constraints:** A typing i can only be included if it contributes to protecting at least one other typing in the set. This ensures two-way synergy rather than one-sided dependencies.
- **Isolation pruning:** Typings that cannot contribute to or benefit from mutual coverage are explicitly excluded from selection.

This formulation allows us to exhaustively enumerate all *resistant complete sets* of a specified size k, where mutual type synergy is both necessary and sufficient.

5 Results

Due to the vastness and decaying relevance of pivot sets larger than 4, I only chose to solve this model for pivot sets of size 2, 3, and 4. One could go higher, but the insights drawn are not that great above 4. In this section, we briefly summarize the results.

There are

- 25 size 2 resistance complete sets
- 96 size 3 resistance complete sets
- 787 size 4 resistance complete sets

with a few standouts shown below in Table 1

Table 1: Most frequently appearing dual typings in pivot chains of length 2–4.

Typing	Size 2 Set	Size 3 Set	Size 4 Set
Flying/Water	4	40	438
Ground/Water	4	30	333
Bug/Steel	5	18	172
Dragon/Flying	6	17	157
Dragon/Steel	3	11	151
Electric/Steel	2	13	125
Dragon/Ground	1	13	118
Grass/Steel	3	8	113
Fire/Flying	2	10	101
Dragon/Fighting	0	12	92

Table 1 shows the top 10 most appearing dual types. We notice that common defensive types like flying, steel, and water are quite prominent. Futhermore, dragon appears to be a frequently occuring type due to its many resistances and being covered completely by steel. Probably most notable is the exact typings at the top of the list correspond with many of the classic defensive Pokemon. Gyrados, Gastrodon/Quagsire, and Scizor are all acknowledged as very reliable nuzlocke Pokemon, and this analysis suggests its justifiably so due to their appearance in many resistance complete sets.

Another interesting quirk is that the top four dual typings all have quadruple weaknesses. This suggests (and gives credit to the belief) that having such a critical weakness is not necessarily a bad thing. The existence of a 4x weakness likely gives a Pokemon the opportunity to resist many other types and that 4x weakness can even be beneficial when serving as a pivot. As long as you're getting something in return for having the 4x weakness, it appears to be quite an advantage in a nuzlocke.

6 The Pivot Chain Explorer

Since it would be impossible for me to list all of the thousand or so resistance complete pivot chains and even more impossible for anyone to sift through them all, I created a hopefully simple to use tool to help aid your nuzlocke/analysis. The link to the tool can be found on my webpage.

The Pivot Chain Explorer allows one to search for all possible pivot chains of a particular length that include particular types. Simply select the type 1 and type 2 of the type you wish to include in the chain, add it to the running list of types to search for, specify the chain length you want to search in, and then press search. A few notes about using the tool:

1. All types are encoded as dual types. So monotypes much be searched by putting them in type1 and type2, e.g. fire/fire.
2. There is no way to remove a type from the running list. To do so, you must start over.

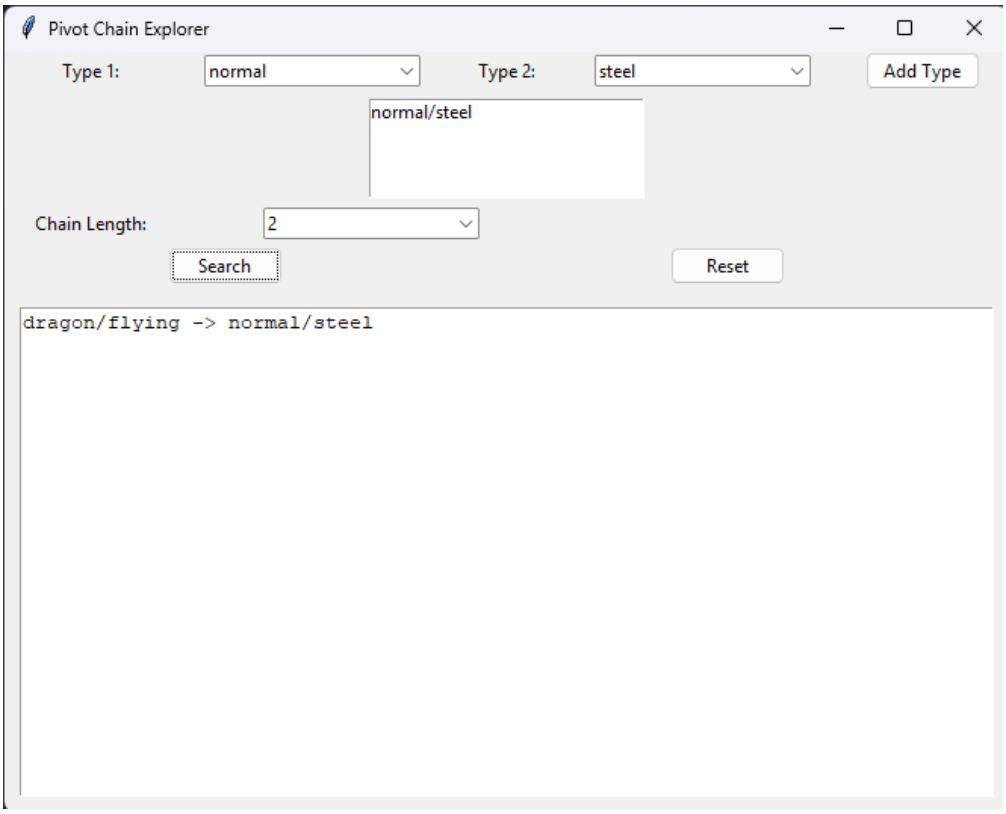

Figure 1: Example use of the pivot chain explorer

3. No type is necessary to be added. If you search without a type list, it will return all pivot chains of the given chain length.

4. Chain length is not necessary to be specified. If you search without a chain length, it will return all pivot chains that contain the type you're searching for.

5. If you do not specify a type nor a chain length, it will return all thousand or so resitance complete pivot chains.

6. The types within a chain are not ordered in any way. You may have to reason yourself as to how you would actually pivot through these types.