New Weave Damage Formulas for Pokemon Go

eric654256
Elite4Insider
https://www.reddit.com/r/elite4insider/

January 2019

Abstract

In Pokemon Go, each Pokemon has two moves: a fast move and charge move. Each move does a specific damage and takes a length of time. It is quick to calculate which moves does the most damage per second. Making this more complicated, fast moves fill the energy bar and the charge move uses the energy bar. There is a well known Weave damage formula that calculates the damage/second for both moves considering energy use. But it doesn't consider extra parts of battling: energy gained from damage taken, dodging, energy lost due to a full charge bar, and others. The equations introduced here are thought to include all known battle details.

1 Introduction

Determining the best Pokemon in Pokemon Go can be subjective. The mechanics of battling has largely been figured out, but there is no best Pokemon for all situations. But to find the top Pokemon for particular situation, there are two general approaches: formulas and simulations.

Simulations can be quite accurate. Given the level of understanding of battle mechanics, they can give every good results, but they can take a long time time to calculate. And even if the source code is open source, they can be difficult to read and understand. Two popular simulators are [1] and [2].

Presented here is an update to the well known Weave formula [3]. The original Weave formula is extremely simple and easy to calculate. It provides a great starting point for any selection of Pokemon and is derived in section 4.1. For Trainer Battles, this is all you need and a longer explanation is below.

Unfortunately, Raid and Gym battles have different battle mechanics. The original formula suffers from some flaws so many people have updated the formula or created simulators. This version of the formula is thought to be the most complete version that includes all known details about battle mechanics.

In addition, this methodology is fast. There is a reference implementation in JavaScript that calculates all attacking Pokemon and move set combinations (3600+) against any defender. Even within the browser of your phone, this page calculates quickly. The reference implementation is available at [4] so it can be reviewed if there is interest.

Also discussed here are other metrics: Damage Per Second (DPS), Total Damage Output (TDO), Time To Faint (TTF) and Damage Output At X (DO@X). This paper introduces a new metric DO@X that is the damage output after X seconds, which is a generalization of both DPS and TDO. This metric can be adjusted between more damage or more time for your individual needs.

2 Battle Description

Each battle has an attacker, the player, and a defender, the computer. Each Pokemon has a fast move and a charge move. The fast move fills the energy bar and the charge move uses the energy bar. The damage taken from the opponent also fills the energy bar.

2.1 Defender Move Duration

The defender does not attack continuously. There is a 1.5 - 2.5 second delay between attacks [5]. So the move duration is increased by 2 seconds on average. Unfortunately there is an exception where the defender's first two attacks happen without a delay. When calculating a long running average such as this one, the first two attacks become less important so it disappears in the long run. The start of the battle needs to be handled differently and there is more about it in the discussion, section 9.

2.2 Energy Gained from Opponent Damage

As mentioned, energy is gained from damage taken [5]. This means that the attacker gets additional energy from the defender's attacks. With more energy, the attacker can use its charge move more often, usually increasing its damage per second (DPS). So the defender now takes more damage, increasing it's energy. The defender can use it's charge move more often, usually increasing it's DPS. This increases the attacker's energy gained and cycle begins again. This can be calculated back and forth, and it converges in just a few iterations. But it can also be solved closed form, and the solution is provided in section 6.1.

2.3 Energy Lost Due to Full Energy Bar

A charge move is only possible with a charge bar is complete and a portion of the energy from the last fast move is wasted if filling the whole bar completely, This always happens when there one long bar, and less frequently when there are two or three. The original weave formula doesn't handle this. People have suggested to calculate the number of fast moves required to fill the charge bar, then round up. However, given that energy is gained from damage taken, it is impossible to tell how much is wasted on the last fast move. I propose losing half of the last move's energy because of energy gained from the opponents damage.

The defender (computer) doesn't always attack once the charge bar is filled. There is a 50% chance of performing a charge move once it is possible [6]. For charge move with one bar, additional fast moves are wasted. For charge moves with 2 or 3 bars, there is a smaller chance of wasting a fast move.

In addition to the 50% chance, the defender's charge move is planned on the previous move [6]. For 1 bar charge moves, there is an additional move wasted. For 2 or 3 bar charge moves, there is a lower chance of filling the bar completely and wasting moves.

Complete details about full energy bars can be found in section 5.

2.4 Damage Window Start

When performing a charge move, the energy is not used immediately when the charge button is pressed. All moves have a "wind-up" time before the HP bar is reduced and Damage Window Start is when the energy is used. Any damage taken before this time does not convert into energy if the energy bar is full.

2.5 Attacker Dodging

The attacker (human player) has the ability to dodge the defender's (computer) attacks by swiping left or right. If the dodge is successful, only 25% of the damage is done [5]. Intuitively, dodging will decrease the attacker's DPS since they are dodging instead of attacking. Also, since more time is spent dodging, a larger proportion of energy will be gained from the opponent's damage.

2.6 Non-constant DPS

Calculating an average DPS implies that DPS is constant throughout the battle. However, this isn't exactly true. The DPS of the fast move is different from the DPS of the charge move. Also, there is a delay between attacks for defenders. More about this in the discussion in section 9.

3 Damage Calculation

3.1 Move Damage

Each move has a power. The damage that the move does is [7].

$$Damage = Floor\Big(\frac{1}{2}\ Power\ \frac{Attack}{Defense}\ STAB\ Weather\ Friend\ Effectiveness\Big) + 1$$

$Power$ = the move's power.

$Attack = (BaseAttack + AttackIV) * CPM$ of the Pokemon doing damage

$Defense = (BaseDefense + DefenseIV) * CPM$ of the Pokemon taking damage

CPM = CP modifier, this is the number that increases when Pokemon increase level

$STAB$ = Same Type Attack Bonus: If the attack move is the same type as the attacking Pokemon, there is a 1.2 STAB bonus

$Weather$ = If the move's type matches the weather, there is a 1.2 bonus

$Friend$ = If there is a friend in the battle, there is a 1.03, 1.05, 1.07, or 1.10 bonus depending on the highest friend level.

$Effectiveness$ = Attack moves of a specific type are better or worse against types of Pokemon taking the damage. This table can be found thru a quick internet search.

Notice that attack and defense depend on CPM, the Pokemon's level. Because of flooring (rounding down), the damage will suddenly jump as the Pokemon increases level. Suppose we have two of Pokemon with the same level. They have damage of 5.1 and 5.9 before flooring. At this specific level, they do the same damage after flooring. But as we increase the two Pokemons' levels, the Pokemon with 5.9 damage would do 6 damage at a lower level than the other Pokemon. If we look across all levels, the Pokemon with 5.9 damage would more damage at most levels. To reflect this difference, $Floor() + 1$ is replaced with adding 0.5 because it will give a better ranking for Pokemon.

$$Damage = \frac{1}{2}\ Power\ \frac{Attack}{Defense}\ STAB\ Weather\ Friend\ Effectiveness + 0.5$$

4 Weave Damage

4.1 Original Formula

Each fast and charge move has a damage P_f and P_c over a duration T_f and T_c. Damage per second (DPS) is simply $\frac{P_f}{T_f}$ and $\frac{P_c}{T_c}$. But fast moves give an amount of energy E_f and charge moves use an amount of energy E_c. To properly calculate DPS, we need to make sure that the "energy in" is equal to the "energy out".

How many fast moves N_f and how many charge moves N_c do we do?

$$E_f N_f = E_c N_c$$

This equation has a simple solution $N_f = E_c$ and $N_c = E_f$ which results in

$$E_f E_c = E_c E_f$$

Now we can calculate DPS, which is simply damage divided by time. Since we have the number of fast and charge moves that solve the energy equation, we can calculate it DPS as:

$$\frac{damage}{time} = \frac{P_f N_f + P_c N_c}{T_f N_f + T_c N_c} = \frac{P_f E_c + P_c E_f}{T_f E_c + T_c E_f}$$

With this intuition, we will explore an alternate formula. We can also change the energy equation to just

$$E_f N = E_c$$

Now, N is the number of fast moves per single charge move. This gives $N = \frac{E_c}{E_f}$ and an equivalent DPS equation

$$\frac{damage}{time} = \frac{P_f N + P_c}{T_f N + T_c} = \frac{P_f \frac{E_c}{E_f} + P_c}{T_f \frac{E_c}{E_f} + T_c}$$

This is the same DPS equation, but the energy equation is just one equation and one variable with only one solution. This form and interpretation is more useful and we will use this form.

As a reminder, defenders have a delay between attacks, which is 2 seconds on average. It doesn't change the energy equation, but it changes their DPS which is now:

$$\frac{damage}{time} = \frac{P_f N + P_c}{(T_f + 2)N + (T_c + 2)} = \frac{P_f \frac{E_c}{E_f} + P_c}{(T_f + 2)\frac{E_c}{E_f} + (T_c + 2)}$$

4.2 Trainer Battles

Gamepress talks about trainer battle mechanics [8]. Regarding calculating DPS and TDO, the important parts are:

1. No Energy is gained from opponent damage. This is the most simplifying change to battles.
2. Charge moves take no time.When a charge move happens, the opponent cannot attack. And after the charge move completes, both players can attack immediately. This means charge moves take no time.
3. Move duration has been changed to a number of turns, each turn taking 0.5 seconds.
4. Move power and energy has changed for trainer battles.
5. The energy bar can hold up to 100 energy. The highest energy cost for a charge move is 80. So if you always use a charge move immediately, you'll always have at least 20 energy free. The highest energy gain for a fast move is 12, so energy bar will never be full.

All of these changes make trainer battles very simple to calculate. The original weave damage formula will be accurate, if shields are not used. To calculate with shields, we could possibly assume a worst case and both shields are used in the battle. Formulas has not been developed for this yet.

4.3 Gym (and Raid) Battles

For battles at a gym, the original battle mechanics didn't change. The original move statistics also didn't change. To calculate DPS, TDO, DO@20 for gym battles, use all the adjusments that follow. The final formula is in 7.3, but consider reading all preceeding the details to understand the formula.

5 Energy Lost Due to Full Energy Bar

Charge moves can consist of 1, 2 or 3 energy bars which always add up to 100. When there is one energy bar, some energy from fast moves is always lost when trying to fill the bar completely. For 2 or 3 energy bars, it is still possible but much less likely.

5.1 Attacker

When considering the energy also gained by the opponent's damage, it is difficult to know how much energy of the last fast move will be wasted. Since the defender attacks have a uniform 1.5 to 2.5 second delay between attacks, we can estimate losing energy from 0.5 fast moves on average. So now the attacker energy and DPS equation would look like this:

$$E_f^*(N^* - 0.5) = E_c^* \qquad (energy)$$

$$N^* = E_c^*/E_f^* + 0.5$$

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^*} = \frac{P_f^* \left(\frac{E_c^*}{E_f^*} + 0.5\right) + P_c^*}{T_f^* \left(\frac{E_c^*}{E_f^*} + 0.5\right) + T_c^*}$$

Asterisk (*) denotes the constants/variables for attackers and not defenders.

For 2 or 3 energy bars, it is reasonable to assume that no energy will ever be lost. Given a realistic amount of dodging, a charge move will be performed before the energy bar is filled. We introduce another constant L^* that represents this difference. The final DPS^* formula is

$$DPS^* = \frac{P_f^* \left(\frac{E_c^*}{E_f^*} + L^*\right) + P_c^*}{T_f^* \left(\frac{E_c^*}{E_f^*} + L^*\right) + T_c^*}$$

$$L^* = \begin{cases} 0.5 & if\ attacker\ has\ 1\ charge\ energy\ bar \\ 0 & if\ attacker\ has\ 2\ or\ 3\ charge\ energy\ bars \end{cases}$$

5.2 Defender

Defenders are controlled by the computer and have random behavior. First, when the defender has the energy required for a charge move, it only does a charge move 50% of the time. The number of attempts until a charge move is a Geometric Distribution. So on average, it does its charge move on the second opportunity and it loses 1 additional fast move of energy. Second, the defender decides the next move before the previous move, so it loses another fast move of energy. In total, it loses 2.5 fast moves of energy if there is one charge bar. The defender energy and DPS equation would look like this:

$$E_f(N - 2.5) = E_c$$

$$N = E_c/E_f + 2.5$$

$$DPS = \frac{P_f N + P_c}{(T_f + 2)N + (T_c + 2)} = \frac{P_f \left(\frac{E_c}{E_f} + 2.5\right) + P_c}{(T_f + 2)\left(\frac{E_c}{E_f} + 2.5\right) + (T_c + 2)}$$

For the defender with 2 charge bars, energy will be lost if the first charge move is not performed before the 2nd charge bar is also filled. Let it take $N = \frac{E_c}{E_f}$ fast moves to do a charge move. When there is enough energy, the defender performs the charge move with probability $p = \frac{1}{2}$. Since the charge move occurrence follows a Geometric distribution, the probability of losing energy is

$$p_{energy\ lost} = 1 - C.D.F. = (1-p)^N = \frac{1}{2}^N$$

The defender also decides its move on the previous move. So the probability is now

$$p_{energy\ lost} = \frac{1}{2}^{N-1}$$

If the defender has 3 charge bars, the probability is

$$p_{energy\ lost} = \frac{1}{2}^{2N-1}$$

Once we start losing energy, we will lose 1.5 moves on average. We lose 0.5 moves for filling the bar completely and we lose 1 additional move because of the 50% chance of performing the charge move. Also, it is possible for $\frac{E_c}{E_f} < 1$ so we put a maximum value for L. Putting it all together, the defender's DPS is

$$DPS = \frac{P_f\left(\frac{E_c}{E_f} + L\right) + P_c}{(T_f+2)\left(\frac{E_c}{E_f} + L\right) + (T_c+2)}$$

$$L = \begin{cases} 2.5 & if\ defender\ has\ 1\ charge\ energy\ bar \\ min\left(2.5, 1.5\ \frac{1}{2}^{\frac{E_c}{E_f}-1}\right) & if\ defender\ has\ 2\ charge\ energy\ bars \\ min\left(2.5, 1.5\ \frac{1}{2}^{2\frac{E_c}{E_f}-1}\right) & if\ defender\ has\ 3\ charge\ energy\ bars \end{cases}$$

6 Energy Gained from Opponent Damage

6.1 Simple Formulas with No Adjustments

In addition to energy gained from fast moves, half of the damage taken by the opponent attacks is gained as energy. This makes the equations more complicated since the attacker and defender energy equations depend on each other. Ultimately, we still just need to set energy in equal to energy out. The additional energy gained is the opponent's DPS multiplied by the duration of the attacks: $\frac{1}{2}\ Opponent\ DPS\ *\ My\ Attack\ Time$

Constants/Variables with an asterisk * are the attacker's. The defender's constants/variables have no asterisk.

First the DPS equations, without any other adjustments.

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^*} \qquad (attacker)$$

$$DPS = \frac{P_f N + P_c}{(T_f+2)N + (T_c+2)} \qquad (defender)$$

Then the energy in/out equations

$$E_f^* N^* + \frac{1}{2} DPS \left(T_f^* N^* + T_c^* \right) = E_c^* \qquad (attacker)$$
$$E_f N + \frac{1}{2} DPS^* \left((T_f + 2) N + (T_c + 2) \right) = E_c \qquad (defender)$$

The energy equations are just two equations with two variables (N, N^*) which can be solved. After solving for N and N^*, we can then calculate DPS and DPS^*. The details are in the appendix.

$$N^* = \frac{W\ N + X}{Y\ N + Z}$$

$$W = E_c^*(T_f + 2) - \frac{1}{2} P_f T_c^*$$
$$X = E_c^*(T_c + 2) - \frac{1}{2} P_c T_c^*$$
$$Y = E_f^*(T_f + 2) + \frac{1}{2} P_f T_f^*$$
$$Z = E_f^*(T_c + 2) + \frac{1}{2} P_c T_f^*$$

$$N = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

(The plus solution is almost always the correct one)

$$A = \frac{1}{2}(P_f^* W + P_c^* Y)(T_f + 2) + E_f (T_f^* W + T_c^* Y)$$
$$B = \frac{1}{2}\Big((P_f^* W + P_c^* Y)(T_c + 2) + (P_f^* X + P_c^* Z)(T_f + 2) \Big) - \Big(E_c (T_f^* W + T_c^* Y) - E_f (T_f^* X + T_c^* Z) \Big)$$
$$C = \frac{1}{2}(P_f^* X + P_c^* Z)(T_c + 2) - E_c (T_f^* X + T_c^* Z)$$

To verify your implementation, it is a good idea to recalculate both energy equations using N and N^* and verify that the "energy in" is equal to the "energy out" for each.

6.1.1 What If N is Negative?

Taking a closer look look at the energy equations, what does it mean if N is negative? N is the defender's number of fast moves per charge move. So the energy gained by the opponent's damage is enough to do a charge move every time and no energy is needed from fast moves. To handle this, we set $N = 0$ and solve the new equations. The details are in the appendix.

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^*} \qquad (attacker)$$
$$DPS = \frac{P_c}{T_c + 2} \qquad (defender)$$

$$N^* = \frac{E_c^*(T_c + 2) - \frac{1}{2} P_c T_c^*}{E_f^*(T_c + 2) + \frac{1}{2} P_c T_f^*}$$

Double check the attacker energy equation to validate your implementation

$$E_f^* N^* + \frac{1}{2} DPS \left(T_f^* N^* + T_c^* \right) = E_c^* \qquad (attacker\ energy)$$

Also check the defender energy inequality that we are getting more energy in than energy out.

$$\frac{1}{2} DPS^* \ (T_c + 2) > E_c \qquad (defender\ energy)$$

6.1.2 What If N^* is Negative?

This is similar to the situation above, but for the attacker. We set $N^* = 0$ and solve. The solution is below and the details are in the appendix.

$$DPS^* = \frac{P_c^*}{T_c^*} \qquad (attacker)$$

$$DPS = \frac{P_f N + P_c}{(T_f + 2)N + (T_c + 2)} \qquad (defender)$$

$$N = \frac{E_c T_c^* - \frac{1}{2} P_c^* (T_c + 2)}{E_f T_c^* + \frac{1}{2} P_c^* (T_f + 2)}$$

Double check the defender energy equation

$$E_f N + \frac{1}{2} DPS^* \left((T_f + 2)N + (T_c + 2) \right) = E_c \qquad (defender\ energy)$$

Check the attacker energy inequality too.

$$\frac{1}{2} DPS \ T_c^* > E_c^* \qquad (attacker\ energy)$$

6.1.3 What If Both N and N^* are Negative?

For the reasons stated above, we just set $N = 0$ and $N^* = 0$. This immediately solves

$$DPS^* = \frac{P_c^*}{T_c^*} \qquad (attacker)$$

$$DPS = \frac{P_c}{T_c + 2} \qquad (defender)$$

And we should check the energy inequalities for both attacker and defender.

$$\frac{1}{2} DPS \ T_c^* > E_c^* \qquad (attacker\ energy)$$

$$\frac{1}{2} DPS^* \ (T_c + 2) > E_c \qquad (defender\ energy)$$

6.1.4 Validating the four solutions

Also to validate your implementation, exactly one of the previous four sections must occur.

1. From section 6.1, $N > 0$ and $N^* > 0$
2. From section 6.1.1, $N^* > 0$ and defender energy inequality is true
3. From section 6.1.2, $N > 0$ and attacker energy inequality is true
4. From section 6.1.3, both attacker and defender inequalities are true

6.2 Fast Move Energy Lost Due to Full Energy Bar

6.2.1 Attacker

If the charge move is one long energy bar, then 0.5 fast moves are lost for the attacker when trying to fill that bar completely. For 2 or 3 smaller energy bars, we can reasonably assume no energy is ever lost.

$$L^* = \begin{cases} 0.5 & if\ attacker\ has\ 1\ charge\ energy\ bar \\ 0 & if\ attacker\ has\ 2\ or\ 3\ charge\ energy\ bars \end{cases}$$

We subtract this amount from the energy gained from fast moves. The energy gained is now

$$E_f^*(N^* - L^*)$$

6.2.2 Defender

For defenders, 2.5 fast moves are lost if there is one long energy bar. Let it take N fast moves to perform one charge move. For 2 energy bars, the probability of losing energy is $\frac{1}{2}^{N-1}$. For 3 energy bars, the probability is $\frac{1}{2}^{2N-1}$. For 2 or 3 energy bars, 1.5 fast moves are lost.

$$L = \begin{cases} 2.5 & if\ defender\ has\ 1\ charge\ energy\ bar \\ min\Big(2.5, 1.5\ \frac{1}{2}^{N-1}\Big) & if\ defender\ has\ 2\ charge\ energy\ bars \\ min\Big(2.5, 1.5\ \frac{1}{2}^{2N-1}\Big) & if\ defender\ has\ 3\ charge\ energy\ bars \end{cases}$$

Similar to the attacker, we subtract this from the energy that the defender gains from fast moves. We have a similar equation for the defender

$$E_f(N - L)$$

6.3 Opponent Damage Energy Lost Due to Full Energy Bar

Similar to the previous section, some energy from opponent damage isn't gained because the energy bar is full. And similar to the previous section, we subtract the time lost from those moves.

$$\frac{1}{2}DPS\ \Big(T_f^*(N^* - L^*) + T_c^*\Big) \qquad (attacker)$$

$$\frac{1}{2}DPS^*\ \Big((T_f + 2)(N - L) + (T_c + 2)\Big) \qquad (defender)$$

6.4 Damage Window Start

All moves have a "wind up" before any damage is done, but only some are noticeable. The damage is done V seconds after the start of the move and is called the Damage Window Start. This is important because the energy for the charge move isn't used until the damage starts [9]. This means, any energy gained from the opponent damage is wasted if the energy bar is full. We introduce an constant I^* defined as

$$I^* = \begin{cases} 1 & if\ attacker\ has\ 1\ charge\ energy\ bar \\ 0 & if\ attacker\ has\ 2\ or\ 3\ charge\ energy\ bars \end{cases}$$

I is the number of moves lost per fast/charge move cycle. To update the equations, we subtract $I^*V_c^*$ from the attacker time for energy gained from defender damage damage.

For the defender, I is defined using a similar methodology as L. However, because $N < 1$ is possible, we add a maximum probability.

$$I = \begin{cases} 1 & if\ defender\ has\ 1\ charge\ energy\ bar \\ min\Big(1, \frac{1}{2}^{N-1}\Big) & if\ defender\ has\ 2\ charge\ energy\ bars \\ min\Big(1, \frac{1}{2}^{2N-1}\Big) & if\ defender\ has\ 3\ charge\ energy\ bars \end{cases}$$

Also, we include the 2 second delay between attacks and use $V_c + 2$. Combined, we subtract $I\ (V_c + 2)$ seconds from the defender time.

6.5 Energy Gained from Opponent Damage, Rounded Up

The energy gained is also rounded up, i.e. $ceiling()$. Similar to changing $floor() + 1$ for damage done, replacing $ceiling()$ with +0.5 will give better rankings.

$$ceiling\Big(\frac{1}{2}damage\Big) \approx \frac{1}{2}damage + 0.5$$

The $\frac{1}{2}damage$ is already included into the energy equations above. To add 0.5 for every opponent's move, lets introduce a new equation, the time equation:

$$T_f^* N^* + T_c^* = K\Big((T_f + 2)N + (T_c + 2)\Big)$$

This is the time for one attacker's fast/charge move cycle to equal K number of defender's fast/charge move cycles. In other words, K is the number of attacker's cycles per defender's cycles.

$$K = \frac{T_f^* N^* + T_c^*}{(T_f + 2)N + (T_c + 2)}$$

The attacker energy equation is one cycle, during which there are K defender cycles. These K defender cycles each have N fast moves and 1 charge move. Each move gains 0.5 energy from replacing ceil(). So the additional energy gained is as follows. The defender energy equation is similar.

$$K(0.5N + 0.5) \Rightarrow 0.5K(N + 1) \qquad (attacker)$$
$$\frac{1}{K}(0.5N^* + 0.5) \Rightarrow \frac{0.5}{K}(N^* + 1) \qquad (defender)$$

Finally, as in the other sections, not all moves will gain energy because the energy bar might be full. We calculate the ratio of moves without a full energy bar

$$\frac{N^* - L^* + 1}{N^* + 1} \quad (attacker)$$

$$\frac{N - L + 1}{N + 1} \quad (attacker)$$

Multiplying it to the previous equations gives us

$$\begin{aligned} 0.5K(N+1)\frac{N^* - L^* + 1}{N^* + 1} & \qquad (attacker) \\ \frac{0.5}{K}(N^*+1)\frac{N - L + 1}{N + 1} & \qquad (defender) \end{aligned}$$

6.6 Equations with All the Adjustments

Now, lets combine all of these ideas together and see the complete equations. We solve these two equations for N and N^*

$$E_f^*(N^* - L^*) + \frac{1}{2}DPS\left(T_f^*(N^* - L^*) + T_c^* - I^*V_c^*\right) + 0.5K(N+1)\frac{N^* - L^* + 1}{N^* + 1} = E_c^* \qquad (attacker)$$

$$E_f(N - L) + \frac{1}{2}DPS^*\left((T_f+2)(N - L) + (T_c+2) - I\ (V_c+2)\right) + \frac{0.5}{K}(N^*+1)\frac{N - L + 1}{N + 1} = E_c \qquad (defender)$$

Where we define:

$$K = \frac{T_f^* N^* + T_c^*}{(T_f+2)N + (T_c+2)}$$

$$I^* = \begin{cases} 1 & if\ attacker\ has\ 1\ charge\ energy\ bar \\ 0 & if\ attacker\ has\ 2\ or\ 3\ charge\ energy\ bars \end{cases}$$

$$I = \begin{cases} 1 & if\ defender\ has\ 1\ charge\ energy\ bar \\ min\left(1, \frac{1}{2}^{N-1}\right) & if\ defender\ has\ 2\ charge\ energy\ bars \\ min\left(1, \frac{1}{2}^{2N-1}\right) & if\ defender\ has\ 3\ charge\ energy\ bars \end{cases}$$

$$L^* = \begin{cases} 0.5 & if\ attacker\ has\ 1\ charge\ energy\ bar \\ 0 & if\ attacker\ has\ 2\ or\ 3\ charge\ energy\ bars \end{cases}$$

$$L = \begin{cases} 2.5 & if\ defender\ has\ 1\ charge\ energy\ bar \\ min\left(2.5, 1.5\ \frac{1}{2}^{N-1}\right) & if\ defender\ has\ 2\ charge\ energy\ bars \\ min\left(2.5, 1.5\ \frac{1}{2}^{2N-1}\right) & if\ defender\ has\ 3\ charge\ energy\ bars \end{cases}$$

We solve for N and N^* to eventually calculate these values.

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^*} \qquad (attacker)$$

$$DPS = \frac{P_f N + P_c}{(T_f + 2)N + (T_c + 2)} \qquad (defender)$$

Unfortunately, L and I are non-linear functions of N which makes these equations extremely difficult to solve. To solve these nonlinear equations, we can use Broyden's Method [11]. Details are in the appendix.

After solving, double check both energy equations to verify your solution

6.6.1 What If N is Negative?

Set $N = 0$ and solve the attacker energy equation for N^*. Similar to above, this equation is not linear in N^* so we use Secant Method [12] to find the solution. Details are in the appendix.

After finding a solution, double check the attacker energy equation.

The defender energy equation becomes an inequality and check that also.

$$E_f(N - L) + \frac{1}{2} DPS^* \left((T_f + 2)(N - L) + (T_c + 2) - I\ (V_c + 2) \right) + \frac{0.5}{K}(N^* + 1)\frac{N - L + 1}{N + 1} > E_c$$

6.6.2 What If N^* is Negative?

Set $N^* = 0$ and solve the defender energy equation for N. As above, this equation is not linear in N so we use Secant Method [12] to find the solution. Details are in the appendix.

After finding a solution, double check the defender energy equation.

The attacker energy equation becomes an inequality and check that also.

$$E_f^*(N^* - L^*) + \frac{1}{2} DPS \left(T_f^*(N^* - L^*) + T_c^* - I^* V_c^* \right) + 0.5K(N + 1)\frac{N^* - L^* + 1}{N^* + 1} > E_c^*$$

6.6.3 What If Both N and N^* are Negative?

Set $N^* = 0$ and $N = 0$. Both attacker and defender energy equations become inequalities and check those.

$$E_f^*(N^* - L^*) + \frac{1}{2} DPS \left(T_f^*(N^* - L^*) + T_c^* - I^* V_c^* \right) + 0.5K(N + 1)\frac{N^* - L^* + 1}{N^* + 1} > E_c^*$$

$$E_f(N - L) + \frac{1}{2} DPS^* \left((T_f + 2)(N - L) + (T_c + 2) - I\ (V_c + 2) \right) + \frac{0.5}{K}(N^* + 1)\frac{N - L + 1}{N + 1} > E_c$$

6.6.4 Validating the four solutions

Also to validate your implementation, exactly one of the previous four sections must occur.

1. From section 6.6, $N > 0$ and $N^* > 0$
2. From section 6.6.1, $N^* > 0$ and defender energy inequality is true
3. From section 6.6.2, $N > 0$ and attacker energy inequality is true

4. From section 6.6.3, both attacker and defender inequalities are true

When validating this part of your solution, sometimes section 6.6 will not have a solution and fail to converge. However, one of the other three will be satisfied.

7 Attacker Dodging

7.1 Defender's Adjusted Damage with Attacker Dodging

The defender will do less damage if the attacker dodges the defender's attacks. First, lets assume the attacker is always doing a move when the defender starts its attack. The defender's move delay is randomly 1.5 - 2.5 seconds, uniformly likely, so assume that we are uniformly likely to be at any point in the attacker's move. When the defender does a move, the attacker can dodge it if he/she finishes their own move (and start dodging) before the defender's damage window starts at time V [10]. Also, we throw in a delay for human reaction time (500ms). Given attacker's fast or charge move, the probability of dodging the defender's move is

$$\begin{aligned} H_{ff} &= If\ attacker\ is\ doing\ fast\ move,\ probability\ of\ dodging\ defender's\ fast\ move && (1)\\ &= max\Big(0, min\Big(1, \frac{V_f - 500ms}{T_f^*}\Big)\Big) && (2) \end{aligned}$$

$$\begin{aligned} H_{cf} &= If\ attacker\ is\ doing\ charge\ move,\ probability\ of\ dodging\ defender's\ fast\ move && (3)\\ &= max\Big(0, min\Big(1, \frac{V_f - 500ms}{T_c^*}\Big)\Big) && (4) \end{aligned}$$

$$\begin{aligned} H_{fc} &= If\ attacker\ is\ doing\ fast\ move,\ probability\ of\ dodging\ defender's\ charge\ move && (5)\\ &= max\Big(0, min\Big(1, \frac{V_c - 500ms}{T_f^*}\Big)\Big) && (6) \end{aligned}$$

$$\begin{aligned} H_{cc} &= If\ attacker\ is\ doing\ charge\ move,\ probability\ of\ dodging\ defender's\ charge\ move && (7)\\ &= max\Big(0, min\Big(1, \frac{V_c - 500ms}{T_c^*}\Big)\Big) && (8) \end{aligned}$$

Next, we need to know how often the attacker will be doing a fast or a charge move.

$$Probability\ of\ attacker\ performing\ charge\ move = \frac{T_c^*}{T_f^* N^* + T_c^*}$$

$$Probability\ of\ attacker\ performing\ fast\ move = 1 - \frac{T_c^*}{T_f^* N^* + T_c^*}$$

Putting these together gives us the attacker's probability of dodging a defender's given move.

$$J_f = Probability\ of\ dodging\ defender's\ fast\ move \tag{9}$$
$$= \frac{T_c^*}{T_f^* N^* + T_c^*} H_{cf} + (1 - \frac{T_c^*}{T_f^* N^* + T_c^*}) H_{ff} \tag{10}$$

$$J_c = Probability\ of\ dodging\ defender's\ charge\ move \tag{11}$$
$$= \frac{T_c^*}{T_f^* N^* + T_c^*} H_{cc} + (1 - \frac{T_c^*}{T_f^* N^* + T_c^*}) H_{fc} \tag{12}$$

So now the defender's damage for its two moves are now

$$\begin{aligned}
P_{f-dodge} &= P_f \Big[0.25 * J_f + 1.00 * (1 - J_f) \Big] \\
&= P_f \Big[1 - 0.75 J_f \Big] \\
&= P_f \Big[1 - 0.75 \Big(\frac{T_c^*}{T_f^* N^* + T_c^*} (H_{cf} - H_{ff}) + H_{ff} \Big) \Big] \\
&= P_f - 0.75 P_f \frac{T_c^*}{T_f^* N^* + T_c^*} (H_{cf} - H_{ff}) - 0.75 P_f H_{ff} \\
&= Q \frac{T_c^*}{T_f^* N^* + T_c^*} + R \\
Q &= -0.75 P_f (H_{cf} - H_{ff}) \\
R &= P_f - 0.75 P_f H_{ff}
\end{aligned}$$

$$\begin{aligned}
P_{c-dodge} &= P_c \Big[0.25 * J_c + 1.00 * (1 - J_c) \Big] \\
&= P_c \Big[1 - 0.75 J_c \Big] \\
&= P_c \Big[1 - 0.75 \Big(\frac{T_c^*}{T_f^* N^* + T_c^*} (H_{cc} - H_{fc}) + H_{fc} \Big) \Big] \\
&= P_c - 0.75 P_c \frac{T_c^*}{T_f^* N^* + T_c^*} (H_{cc} - H_{fc}) - 0.75 P_c H_{fc} \\
&= S \frac{T_c^*}{T_f^* N^* + T_c^*} + U \\
S &= -0.75 P_c (H_{cc} - H_{fc}) \\
U &= P_c - 0.75 P_c H_{fc}
\end{aligned}$$

7.2 New Energy Equations with Attacker Dodging and Energy Gained from Opponent Damage, No Other Adjustments

Each dodge takes 0.5 seconds. Suppose for the moment, there is no waiting to dodge, or spam dodging. If the defender does N fast attacks per charge attack, then the attacker takes at least $0.5N + 0.5$ seconds to dodge through one cycle of the defender's fast and charge attacks. For one cycle of the attacker's fast/charge attacks, let there be K cycles of the defender's fast/charge attacks. The attacker's DPS and energy equation becomes

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^* + K(0.5N + 0.5)}$$

$$E_f^* N^* + \frac{1}{2} DPS\,\Big(T_f^* N^* + T_c^* + K(0.5N + 0.5)\Big) = E_c^*$$

But a new variable K has been introduced so we need a third equation. We can solve for K by looking at the time taken by the attacker's energy equation. The time that the attacker took for one fast/charge cycle (and dodging K fast/charge cycles from the defender) equals the time the defender took for those K fast/charge cycles.

$$T_f^* N^* + T_c^* + K(0.5N + 0.5) = K\Big((T_f + 2)N + (T_c + 2)\Big)$$

This equation gives us three equations (two energy equations and one time equation) and three variables (N, N^*, and K). For generality, 0.5 has been replaced with G_f^* and G_c^*. This is to allow for both time spent dodging and time spent waiting to dodge. So if you spam dodge multiple times, or pause attacking and wait for the yellow flash, set G_f^* and G_c^* appropriately. If the attacker is not dodging fast moves, set $H_{ff} = 0$, $H_{cf} = 0$, $G_f^* = 0$.

First the DPS equations. We use the defender's new $P_{f-dodge}$ and $P_{c-dodge}$

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)} \qquad (attacker)$$

$$DPS = \frac{P_{f-dodge} N + P_{c-dodge}}{(T_f + 2)N + (T_c + 2)} = \frac{\Big(Q\frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big)N + S\frac{T_c^*}{T_f^* N^* + T_c^*} + U}{(T_f + 2)N + (T_c + 2)} \qquad (defender)$$

Then the energy equations

$$E_f^* N^* + \frac{1}{2} DPS\,\Big(T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)\Big) = E_c^* \qquad (attacker)$$

$$E_f N + \frac{1}{2} DPS^*\,\Big((T_f + 2)N + (T_c + 2)\Big) = E_c \qquad (defender)$$

And the time equation

$$T_f^* N^* + T_c^* + K(G_f^* N + G_c^*) = K\Big((T_f + 2)N + (T_c + 2)\Big)$$

We can substitute DPS^*, DPS and K into the energy equations and get two equations and variables, N^* and N. The solution is below and the details are in the appendix

$$N = \frac{W\ N^* + X}{Y\ N^* + Z}$$

$$W = E_c T_f^* - \frac{1}{2} P_f^* (T_c + 2 - G_c^*)$$

$$X = E_c T_c^* - \frac{1}{2} P_c^* (T_c + 2 - G_c^*)$$

$$Y = E_f T_f^* + \frac{1}{2} P_f^* (T_f + 2 - G_f^*)$$

$$Z = E_f T_c^* + \frac{1}{2} P_c^* (T_f + 2 - G_f^*)$$

$$N^* = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

(the plus solution is almost always the correct one)

$$A = \frac{1}{2}(RT_f^* W + UT_f^* Y) + E_f^* \Big((T_f + 2 - G_f^*) W + (T_c + 2 - G_c^*) Y \Big)$$

$$B = \frac{1}{2} \Big((QT_c^* + RT_c^*) W + RT_f^* X + (ST_c^* + UT_c^*) Y + UT_f^* Z \Big)$$

$$-E_c^* \Big((T_f + 2 - G_f^*) W + (T_c + 2 - G_c^*) Y \Big) + E_f^* \Big((T_f + 2 - G_f^*) X + (T_c + 2 - G_c^*) Z \Big)$$

$$C = \frac{1}{2} \Big((QT_c^* + RT_c^*) X + (ST_c^* + UT_c^*) Z \Big) - E_c^* \Big((T_f + 2 - G_f^*) X + (T_c + 2 - G_c^*) Z \Big)$$

Double check the attacker and defender energy equation, as well as the time equation, using K solved below

$$K = \frac{T_f^* N^* + T_c^*}{(T_f + 2) N + (T_c + 2) - G_f^* N - G_c^*}$$

7.2.1 What If N is Negative?

Set $N = 0$ and solve the attacker equation

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^* + K G_c^*} \qquad (attacker)$$

$$DPS = \frac{S \frac{T_c^*}{T_f^* N^* + T_c^*} + U}{T_c + 2} \qquad (defender)$$

$$K = \frac{T_f^* N^* + T_c^*}{T_c + 2 - G_c^*}$$

$$N^* = \frac{E_c^* (T_c + 2 - G_c^*) - \frac{1}{2}(ST_c^* + UT_c^*)}{E_f^* (T_c + 2 - G_c^*) + \frac{1}{2} UT_f^*}$$

Double check the attacker energy equation and time equation

$$E_f^* N^* + \frac{1}{2} DPS \ (T_f^* N^* + T_c^* + K G_c^*) = E_c^*$$

$$T_f^* N^* + T_c^* + K G_c^* = K (T_c + 2)$$

Check the defender energy inequality

$$\frac{1}{2} DPS^* \ (T_c + 2) > E_c$$

7.2.2 What If N^* is Negative?

Set $N^* = 0$ and solve the defender equation

$$DPS^* = \frac{P_c^*}{T_c^* + K(G_f^* N + G_c^*)} \qquad (attacker)$$
$$DPS = \frac{(Q+R)N + S + U}{(T_f+2)N + (T_c+2)} \qquad (defender)$$

$$K = \frac{T_c^*}{(T_f+2)N + (T_c+2) - G_f^* N - G_c^*}$$

$$N = \frac{E_c T_c^* - \frac{1}{2} P_c^* (T_c + 2 - G_c^*)}{E_f T_c^* + \frac{1}{2} P_c^* (T_f + 2 - G_f^*)}$$

Double check the defender energy equation and the time equation

$$E_f N + \frac{1}{2} DPS^* \Big((T_f+2)N + (T_c+2) \Big) = E_c$$

$$T_c^* + K(G_f^* N + G_c^*) = K\Big((T_f+2)N + (T_c+2) \Big)$$

Check the attacker inequality

$$\frac{1}{2} DPS \Big(T_c^* + K(G_f^* N + G_c^*) \Big) > E_c^*$$

7.2.3 What If Both N and N^* are Negative?

Set $N^* = 0$ and $N = 0$. The DPS equations are immediate.

$$DPS^* = \frac{P_c^*}{T_c^* + K G_c^*} \qquad (attacker)$$
$$DPS = \frac{S+U}{T_c+2} \qquad (defender)$$

$$K = \frac{T_c^*}{T_c + 2 - G_c^*}$$

Double check the time equation

$$T_c^* + K G_c^* = K(T_c + 2)$$

It is important to check the energy inequalities. They can fail.

$$\frac{1}{2} DPS\ (T_c^* + K G_c^*) > E_c^* \qquad (attacker)$$
$$\frac{1}{2} DPS^* (T_c + 2) > E_c \qquad (defender)$$

7.2.4 Validating the four solutions

Also to validate your implementation, exactly one of the previous four sections must occur.

1. From section 7.2, $N > 0$ and $N^* > 0$
2. From section 7.2.1, $N^* > 0$ and defender energy inequality is true
3. From section 7.2.2, $N > 0$ and attacker energy inequality is true
4. From section 7.2.3, both attacker and defender inequalities are true

7.3 Equations with All the Adjustments

Now, lets add all the previously discussed adjustments. Solve the two energy equations for N and N^*

$$E_f^*(N^* - L^*) + \frac{1}{2}DPS\,\Big(T_f^*(N^* - L^*) + T_c^* - I^*V_c^* + K(G_f^*N + G_c^*)\Big) + 0.5K(N+1)\frac{N^* - L^* + 1}{N^* + 1} = E_c^* \qquad (attacker)$$

$$E_f(N - L) + \frac{1}{2}DPS^*\,\Big((T_f + 2)(N - L) + (T_c + 2) - I\,(V_c + 2)\Big) + \frac{0.5}{K}(N^* + 1)\frac{N - L + 1}{N + 1} = E_c \qquad (defender)$$

Where we define

$$K = \frac{T_f^*N^* + T_c^*}{(T_f + 2)N + (T_c + 2) - G_f^*N - G_c^*}$$

$$I^* = \begin{cases} 1 & if\ attacker\ has\ 1\ charge\ energy\ bar \\ 0 & if\ attacker\ has\ 2\ or\ 3\ charge\ energy\ bars \end{cases}$$

$$I = \begin{cases} 1 & if\ defender\ has\ 1\ charge\ energy\ bar \\ min\Big(1, \frac{1}{2}^{N-1}\Big) & if\ defender\ has\ 2\ charge\ energy\ bars \\ min\Big(1, \frac{1}{2}^{2N-1}\Big) & if\ defender\ has\ 3\ charge\ energy\ bars \end{cases}$$

$$L^* = \begin{cases} 0.5 & if\ attacker\ has\ 1\ charge\ energy\ bar \\ 0 & if\ attacker\ has\ 2\ or\ 3\ charge\ energy\ bars \end{cases}$$

$$L = \begin{cases} 2.5 & if\ defender\ has\ 1\ charge\ energy\ bar \\ min\Big(2.5, 1.5\ \frac{1}{2}^{N-1}\Big) & if\ defender\ has\ 2\ charge\ energy\ bars \\ min\Big(2.5, 1.5\ \frac{1}{2}^{2N-1}\Big) & if\ defender\ has\ 3\ charge\ energy\ bars \end{cases}$$

We solve for N, N^* to eventually calculate DPS with the defender's new $P_{f-dodge}$ and $P_{c-dodge}$

$$DPS^* = \frac{P_f^*N^* + P_c^*}{T_f^*N^* + T_c^* + K(G_f^*N + G_c^*)} \qquad (attacker)$$

$$DPS = \frac{P_{f-dodge}N + P_{c-dodge}}{(T_f + 2)N + (T_c + 2)} \qquad (defender)$$

Similar to the previous equations without dodging, L and I are non-linear functions of N which makes these equations extremely difficult to solve. Finally $P_{f-dodge}$ and $P_{c-dodge}$ also contain N^* making the equations difficult to solve.

To solve the two non-linear equations, we use Broyden's Method [11] and the details are in the appendix.

After solving for N and N^*, double check that the two energy equations are correctly solved.

7.3.1 What If N is Negative?

Set $N = 0$ and solve the attacker energy equation. Since the attacker energy equation is not linear in N^*, we use Secant Method [12] to solve the equation.

After solving, double check the attacker energy equation and check the defender energy inequality

$$E_f(N-L) + \frac{1}{2}DPS^* \Big((T_f+2)(N-L) + (T_c+2) - I\ (V_c+2)\Big) + \frac{0.5}{K}(N^*+1)\frac{N-L+1}{N+1} > E_c$$

7.3.2 What If N^* is Negative?

Set $N^* = 0$ and solve the defender energy equation. Since the defender energy equation is not linear in N, we use Secant Method [12] to solve the equation.

After solving, double check the defender energy equation and check the attacker energy inequality

$$E_f^*(N^*-L^*) + \frac{1}{2}DPS\ \Big(T_f^*(N^*-L^*) + T_c^* - I^*V_c^* + K(G_f^*N + G_c^*)\Big) + 0.5K(N+1)\frac{N^*-L^*+1}{N^*+1} > E_c^*$$

7.3.3 What If Both N and N^* are Negative?

Set $N^* = 0$ and $N = 0$. The DPS equations are immediate.

Check both energy inequalities.

$$E_f^*(N^*-L^*) + \frac{1}{2}DPS\ \Big(T_f^*(N^*-L^*) + T_c^* - I^*V_c^* + K(G_f^*N + G_c^*)\Big) + 0.5K(N+1)\frac{N^*-L^*+1}{N^*+1} > E_c^*$$

$$E_f(N-L) + \frac{1}{2}DPS^* \Big((T_f+2)(N-L) + (T_c+2) - I\ (V_c+2)\Big) + \frac{0.5}{K}(N^*+1)\frac{N-L+1}{N+1} > E_c$$

7.3.4 Validating the four solutions

Also to validate your implementation, exactly one of the previous four sections must occur.

1. From section 7.3, $N > 0$ and $N^* > 0$
2. From section 7.3.1, $N^* > 0$ and defender energy inequality is true
3. From section 7.3.2, $N > 0$ and attacker energy inequality is true
4. From section 7.3.3, both attacker and defender inequalities are true

When validating this part of your solution, it possible that section 7.3 will not have a solution and fail to converge. However, one of the other three sections should be satisfied.

8 Finding the Best Pokemon

8.1 Total Damage Output (TDO)

Calculating a Pokemon's damage per second (DPS) is only half of the problem. Pokemon also need to survive long enough to do that damage. To estimate this, we can create a generic opponent (gym defender), with generic moves and calculate DPS^* and DPS for you (attacker) and the computer (defender). Then, calculate the HP^* for your Pokemon and we can calculate the amount of damage before your Pokemon faints.

$$HP^* = (BaseStamina + StaminaIV) * CPM \qquad (attacker)$$

$$HP = 2 * (BaseStamina + StaminaIV) * CPM \qquad (defender)$$

$$TDO^* = DPS^* \frac{HP^*}{DPS}$$

For ranking purposes, it does not matter what is used for the defender's DPS. If the defender's DPS changes, the attacker's TDO^* will change, but they all change proportionally and the ranking will stay the same.

Also, TDO doesn't consider the survivability (HP) of the computer (defender). If there is a maximum HP for the computer, then TDO will be capped and all high level Pokemon would be equivalent.

8.2 Damage Output at X seconds (DO@X)

This calculates the amount of damage done by X seconds. For Pokemon that faint before X seconds, this is their TDO^*. For others, this is DPS^* times X.

$$DO@X^* = DPS^*\ min(X, \frac{HP^*}{DPS})$$

This metric is particularly useful because it is a generalization of DPS and TDO.

$$DPS = DO@1$$

$$TDO = DO@999$$

If the calculated ranking seems too weak and Pokemon won't survive long, increase the seconds. If the ranking doesn't seem to do enough damage, decrease the seconds.

The current raid system has raid bosses with lots of hit points. Because of the time limit, defeating a raid bosses means doing enough damage before time is up. Tier 1 and 2 raid bosses are relatively easy to defeat alone. Tier 3 raid bosses are a harder to defeat alone. Tier 4 and 5 are impossible to defeat alone, i.e. a single person cannot do enough damage within the time limit.

For higher tier raids, high DPS Pokemon may faint before doing enough damage to defeat the raid boss. High TDO Pokemon may not do damage fast enough and run out of time before defeating the raid boss. The best Pokemon do enough damage over a specific period of time. To measure this, you can calculate $DO@X$.

The rankings of other websites seem to match the rankings of DO@20 or DO@25. But this metric is flexible. If you find that you are not doing enough damage, try DO@15 or DO@10 and setup multiple battle parties in order to re-enter the raid quickly. If you are limited in your Pokemon and they are not lasting long enough, try DO@30 or higher.

8.3 Time To Faint (TTF)

We can also calculate who should win the battle on average by calculating the HP for the attacker and defender, then time to faint (TTF). Whoever faints first will lose the battle.

$$TTF^* = HP^*/DPS \qquad (attacker)$$
$$TTF = HP/DPS^* \qquad (defender)$$

When the battle begins, the defender immediately attacks twice without any 2 second delay for either. We can subtract that in the attacker's TTF. However, it only happens once for raid battles, so it is left out of TTF in the current reference implemenation for now.

$$TTF^* = HP^*/DPS - 4 \qquad (attacker)$$

Next, fast moves and charge moves have different DPS. A simple division will not give the best TTF. Fortunately, when a charge move happens, the fast/charge cycle completes and the calculated average DPS will equal the observed DPS. We just need to figure out how many fast/charge cycles happen and then how the last cycle plays out.

$$Attacker\ survived\ fast/charge\ cycles\ from\ Defender = floor\Big(\frac{HP^*}{P_{f-dodge}N + P_{c-dodge}}\Big)$$

$$Attacker's\ Remaining\ HP^* = HP^* - floor\Big(\frac{HP^*}{P_{f-dodge}N + P_c}\Big)(P_f N + P_{c-dodge})$$

$$Attacker\ survived\ fast\ moves\ from\ Defender's\ last\ cycle = \frac{Remaining\ HP^*}{P_{f-dodge}}$$

Now, we need to see if these are too many fast moves, and we faint the attacker with our charge move. If the number of fast moves is more than N, then we complete one additional cycle instead.

$$Adjusted\ Cycles\ Surived\ from\ Defender = \begin{cases} Cycles\ survived + 1 & fast\ moves\ survived > N \\ Cycles\ survived & otherwise \end{cases}$$

$$Adjusted\ Fast\ Moves\ Surived\ from\ Defender = \begin{cases} 0 & fast\ moves\ survived > N \\ Fast\ moves\ survived & otherwise \end{cases}$$

Now we just add up the two sections of time. We can also subtract 4 seconds because the defender does not wait before its first and second moves. The reference implementation does not do this because it only happens for the first attacker during a Raid.

$$Attacker's\ final\ TTF = (Adjusted\ Cycles\ Surived\ from\ Defender)\Big((T_f + 2)N + (T_c + 2)\Big)$$
$$+ (Adjusted\ Fast\ Moves\ Surived\ from\ Defender)\ (T_f + 2)$$

The defender's TTF is slightly different, because the attacker time also includes time spent dodging. For simplicity, dodge time is split proportionally to each move.

$$Attacker\ dodge\ time\ split\ per\ fast\ move = \frac{T_f^*}{T_f^* N^* + T_c^*} K(G_f^* N + G_c^*)$$

$$Adjusted\ attacker\ fast\ move\ time = T_f^* + \frac{T_f^*}{T_f^* N^* + T_c^*} K(G_f^* N + G_c^*)$$

This allows us to use very similar equations as above.

$$Defender\ survived\ fast/charge\ cycles\ from\ Attacker = floor\Big(\frac{HP}{P_f^* N^* + P_c^*}\Big)$$

$$Defender\ remaining\ HP = HP - floor\Big(\frac{HP}{P_f^* N^* + P_c^*}\Big)(P_f^* N^* + P_c^*)$$

$$Defender\ survived\ fast\ moves\ from\ Attacker's\ last\ cycle = \frac{Remaining\ HP}{P_f^*}$$

Again, we check the number of fast moves survived to see if the attacker's charge move would faint the defender

$$Adjusted\ Cycles\ Surived\ from\ Attacker = \begin{cases} Cycles\ survived + 1 & fast\ moves\ survived > N^* \\ Cycles\ survived & otherwise \end{cases}$$

$$Adjusted\ Fast\ Moves\ Surived\ from\ Attacker = \begin{cases} 0 & fast\ moves\ survived > N^* \\ Fast\ moves\ survived & otherwise \end{cases}$$

The total TTF is the time spent in the survived fast/charge cycles and the time survived in the last cycle.

$$Defender's\ final\ TTF = (Adjusted\ Cycles\ Surived\ from\ Attacker)\Big(T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)\Big) +$$

$$(Adjusted\ Fast\ Moves\ Surived\ from\ Attacker)(Adjusted\ attacker\ fast\ move\ time)$$

9 Discussion

9.1 Not Constant DPS

These equations calculate long running DPS averages. There are various parts of battling that do not follow this.

1. Fast moves and charge moves each have their own DPS. Charge moves usually have higher DPS than fast moves. So the observed DPS will be less than the calculated DPS when fast moves happen, then match the calculated DPS after each charge move.

2. For Pokemon that don't even last until a single charge move, the calculated DPS will be especially high because DPS of fast moves is lower than DPS of charge moves. This situation is poorly estimated and we can avoid recommending these short lived Pokemon by calculating DO@5. This metric is described in section 8.2.

3. When starting a battle, the computer will immediately attack twice. This is not handled in the average DPS since this DPS is a long running average and the first two attacks would eventually average out.

All of these can be handled when calculating time-to-faint (TTF). While they all would technically alter the DPS, a close approximation can be made through TTF which is described in section 8.3.

9.2 Defender Attack Delay

The defender has a delay of 1.5 - 2.5 seconds between attacks. If we assume the delay is uniformly distributed, the delay across n attacks is a Irwin-Hall distribution and the variance is $\frac{n}{12}$. Below is a table of a hypothetical 1 second attack, and cumulative delays and a confidence interval. For practical purposes, the CDF of the Irwin-Hall distribution is complicated. For $n >= 15$, a normal approximation of the confidence interval is extremely close and is used below.

Number Of Attacks	**Time Spent Attacking (sec)**	**Delay Time Average (sec)**	**Total Average Time (sec)**	**95% Confidence Interval**
1	1	2	3	2.525 - 3.475
2	2	4	6	5.224 - 6.776
3	3	6	9	8.031 - 93969
5	5	10	15	13.746 - 16.254
10	10	20	30	28.219 - 31.781
15	15	30	45	42.764 - 47.236
20	20	40	60	57.418 - 62.582
30	30	60	90	86.838 - 93.162
40	40	80	120	116.349 - 123.651
50	50	100	150	145.918 - 154.082

10 Appendix

10.1 Solving Equations from Section 6.1

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^*} \qquad (attacker)$$

$$DPS = \frac{P_f N + P_c}{(T_f + 2)N + (T_c + 2)} \qquad (defender)$$

$$E_f^* N^* + \frac{1}{2} DPS \left(T_f^* N^* + T_c^* \right) = E_c^* \qquad (a1:\ attacker\ energy)$$

$$E_f N + \frac{1}{2} DPS^* \Big((T_f + 2)N + (T_c + 2) \Big) = E_c \qquad (d1:\ defender\ energy)$$

Solve two energy equations for N and N^*

$$E_f^* N^* + \frac{1}{2} \frac{P_f N + P_c}{(T_f + 2)N + (T_c + 2)} \left(T_f^* N^* + T_c^* \right) = E_c^* \qquad (substitute\ DPS\ into\ a1)$$

$$E_f^* N^* \Big((T_f + 2)N + (T_c + 2) \Big) + \frac{1}{2} (P_f N + P_c) \left(T_f^* N^* + T_c^* \right) = E_c^* \Big((T_f + 2)N + (T_c + 2) \Big)$$

$$E_f^* N^* \Big((T_f + 2)N + (T_c + 2) \Big) + \frac{1}{2} (P_f N + P_c)\ T_f^* N^* = E_c^* \Big((T_f + 2)N + (T_c + 2) \Big) - \frac{1}{2} (P_f N + P_c)\ T_c^*$$

$$N^* = \frac{E_c^* \Big((T_f + 2)N + (T_c + 2) \Big) - \frac{1}{2} (P_f N + P_c)\ T_c^*}{E_f^* \Big((T_f + 2)N + (T_c + 2) \Big) + \frac{1}{2} (P_f N + P_c)\ T_f^*}$$

$$N^* = \frac{\Big(E_c^*(T_f+2) - \frac{1}{2}P_f T_c^*\Big)N + E_c^*(T_c+2) - \frac{1}{2}P_c T_c^*}{\Big(E_f^*(T_f+2) + \frac{1}{2}P_f T_f^*\Big)N + E_f^*(T_c+2) + \frac{1}{2}P_c T_f^*}$$

$$N^* = \frac{W\ N + X}{Y\ N + Z}$$

$$W = E_c^*(T_f+2) - \frac{1}{2}P_f T_c^*$$
$$X = E_c^*(T_c+2) - \frac{1}{2}P_c T_c^*$$
$$Y = E_f^*(T_f+2) + \frac{1}{2}P_f T_f^*$$
$$Z = E_f^*(T_c+2) + \frac{1}{2}P_c T_f^*$$

$$E_f N + \frac{1}{2}\frac{P_f^*\frac{W\ N+X}{Y\ N+Z} + P_c^*}{T_f^*\frac{W\ N+X}{Y\ N+Z} + T_c^*}\Big((T_f+2)N + (T_c+2)\Big) = E_c \qquad (\textit{substitute } DPS^* \textit{ and } N^* \textit{ into d1})$$

$$\frac{1}{2}\Big(P_f^*\frac{W\ N+X}{Y\ N+Z} + P_c^*\Big)\Big((T_f+2)N + (T_c+2)\Big) = \Big(T_f^*\frac{W\ N+X}{Y\ N+Z} + T_c^*\Big)\Big(E_c - E_f N\Big)$$

$$\frac{1}{2}\Big(P_f^*(W\ N+X) + P_c^*(Y\ N+Z)\Big)\Big((T_f+2)N + (T_c+2)\Big) = \Big(T_f^*(W\ N+X) + T_c^*(Y\ N+Z)\Big)\Big(E_c - E_f N\Big)$$

$$\frac{1}{2}\Big((P_f^*W + P_c^*Y)N + P_f^*X + P_c^*Z\Big)\Big((T_f+2)N + (T_c+2)\Big) = \Big(E_c - E_f N\Big)\Big((T_f^*W + T_c^*Y)N + T_f^*X + T_c^*Z\Big)$$

$$\frac{1}{2}\Big[(P_f^*W + P_c^*Y)(T_f+2)N^2 + \Big((P_f^*W + P_c^*Y)(T_c+2) + (P_f^*X + P_c^*Z)(T_f+2)\Big)N + (P_f^*X + P_c^*Z)(T_c+2)\Big]$$
$$= -E_f(T_f^*W + T_c^*Y)N^2 + \Big(E_c(T_f^*W + T_c^*Y) - E_f(T_f^*X + T_c^*Z)\Big)N + E_c(T_f^*X + T_c^*Z)$$

This results in the quadratic equation, and quadratic formula. However, we almost always take the "plus" solution. The "minus" solution usually results in division by zero when calculating DPS for both attacker and defender.

$$AN^2 + BN + C = 0$$

$$N = \frac{-B + \sqrt{B^2 - 4AC}}{2A}$$

$$A = \frac{1}{2}(P_f^*W + P_c^*Y)(T_f+2) + E_f(T_f^*W + T_c^*Y)$$

$$B = \frac{1}{2}\Big((P_f^* W + P_c^* Y)(T_c + 2) + (P_f^* X + P_c^* Z)(T_f + 2)\Big) - \Big(E_c(T_f^* W + T_c^* Y) - E_f(T_f^* X + T_c^* Z)\Big)$$

$$C = \frac{1}{2}(P_f^* X + P_c^* Z)(T_c + 2) - E_c(T_f^* X + T_c^* Z)$$

10.1.1 What If N is Negative?

Set $N = 0$ and solve the attacker energy equation

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^*} \qquad (attacker)$$

$$DPS = \frac{P_f N + P_c}{(T_f + 2)N + (T_c + 2)} = \frac{P_c}{T_c + 2} \qquad (defender)$$

Solve attacker energy equation

$$E_f^* N^* + \frac{1}{2} DPS \left(T_f^* N^* + T_c^*\right) = E_c^* \qquad (attacker\ energy)$$

$$E_f^* N^* + \frac{1}{2}\frac{P_c}{T_c + 2}\left(T_f^* N^* + T_c^*\right) = E_c^* \qquad (substitute\ DPS)$$

$$\left(E_f^* + \frac{1}{2}\frac{P_c}{T_c + 2}\ T_f^*\right) N^* + \frac{1}{2}\frac{P_c}{T_c + 2} T_c^* = E_c^*$$

$$N^* = \frac{E_c^* - \frac{1}{2}\frac{P_c}{T_c+2} T_c^*}{E_f^* + \frac{1}{2}\frac{P_c}{T_c+2}\ T_f^*}$$

$$N^* = \frac{E_c^*(T_c + 2) - \frac{1}{2} P_c T_c^*}{E_f^*(T_c + 2) + \frac{1}{2} P_c T_f^*}$$

Check the defender energy inequality

$$\frac{1}{2} DPS^*\ (T_c + 2) > E_c \qquad (defender\ energy)$$

10.1.2 What If N^* is Negative?

Set $N^* = 0$ and solve the defender energy equation

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^*} = \frac{P_c^*}{T_c^*} \qquad (attacker)$$

$$DPS = \frac{P_f N + P_c}{(T_f + 2)N + (T_c + 2)} \qquad (defender)$$

Solve defender energy equation.

$$E_f N + \frac{1}{2} DPS^* \Big((T_f + 2)N + (T_c + 2)\Big) = E_c \qquad (d1:\ defender\ energy)$$

$$E_f N + \frac{1}{2}\frac{P_c^*}{T_c^*}\Big((T_f+2)N + (T_c+2)\Big) = E_c \qquad (substitute\ DPS^*)$$

$$\Big(E_f + \frac{1}{2}\frac{P_c^*}{T_c^*}(T_f+2)\Big)N + \frac{1}{2}\frac{P_c^*}{T_c^*}(T_c+2) = E_c$$

$$N = \frac{E_c - \frac{1}{2}\frac{P_c^*}{T_c^*}(T_c+2)}{E_f + \frac{1}{2}\frac{P_c^*}{T_c^*}(T_f+2)}$$

$$N = \frac{E_c T_c^* - \frac{1}{2}P_c^*(T_c+2)}{E_f T_c^* + \frac{1}{2}P_c^*(T_f+2)}$$

Check the attacker energy inequality

$$\frac{1}{2}DPS\ T_c^* > E_c^* \qquad (attacker\ energy)$$

10.1.3 What If Both N and N^* are Negative?

Set $N^* = 0$ and $N = 0$

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^*} = \frac{P_c^*}{T_c^*} \qquad (attacker)$$

$$DPS = \frac{P_f N + P_c}{(T_f+2)N + (T_c+2)} = \frac{P_c}{T_c+2} \qquad (defender)$$

Check both energy inequalities

$$\frac{1}{2}DPS\ T_c^* > E_c^* \qquad (attacker\ energy)$$

$$\frac{1}{2}DPS^*\ (T_c+2) > E_c \qquad (defender\ energy)$$

10.2 Solving Equations from Section 6.6

As mentioned earlier, the two energy equations are non-linear and difficult to solve algebraically. There are multiple ways to solve non-linear equations. In this case, Broyden's Method [11] is used because of its simplicity and speed. Broyden's Method is a generalization of Secant Method [12]. For that reason, consider reading the next two sections first (10.2.1, 10.2.2), and then come back to this section.

$$E_f^*(N^* - L^*) + \frac{1}{2}DPS\ \Big(T_f^*(N^* - L^*) + T_c^* - I^* V_c^*\Big) + 0.5K(N+1)\frac{N^* - L^* + 1}{N^* + 1} - E_c^* = 0 \qquad (attacker)$$

$$E_f(N - L) + \frac{1}{2}DPS^*\ \Big((T_f+2)(N-L) + (T_c+2) - I\ (V_c+2)\Big) + \frac{0.5}{K}(N^*+1)\frac{N - L + 1}{N+1} - E_c = 0 \qquad (defender)$$

We start by selecting a starting point $[N_0^* = 0, N_0 = 0]$ and calculating the attacker and defender energy equations, giving us $[ae_0, de_0]$. Next, we select a second point $[N_1^* = E_c^*/E_f^* + 0.5, N_1 = E_c/E_f + 2.5]$ and calculate the energy equations, giving us $[ae_1, de_1]$.

Similar to the Secant Method, we calculate a slope between these two points. When there is more than one variable, the slope is the Jacobian Matrix.

$$J = \begin{bmatrix} \frac{\partial attacker}{\partial N^*} & \frac{\partial attacker}{\partial N} \\ \\ \frac{\partial defender}{\partial N^*} & \frac{\partial defender}{\partial N} \end{bmatrix}$$

Similar to the Secant Method, we use Newton's finite differences to estimate the Jacobian Matrix. To estimate the partial derivatives, we calculate the energy equations for two points: $[N_1^*, N_0]$ and $[N_0^*, N_1]$. Use the point $[N_1^*, N_0]$ to calculate attacker and defender energies of $[ae_{N^*}, de_{N^*}]$. Use the second point $[N_0^*, N_1]$ to get $[ae_N, de_N]$. The estimated Jacobian matrix is now

$$J = \begin{bmatrix} \frac{ae_{N^*} - ae_0}{N_1^* - N_0^*} & \frac{ae_N - ae_0}{N_1 - N_0} \\ \\ \frac{de_{N^*} - de_0}{N_1^* - N_0^*} & \frac{de_N - de_0}{N_1 - N_0} \end{bmatrix}$$

We invert the matrix into:

$$J^{-1} = \frac{1}{det} \begin{bmatrix} \frac{de_N - de_0}{N_1 - N_0} & -\frac{ae_N - ae_0}{N_1 - N_0} \\ \\ -\frac{de_{N^*} - de_0}{N_1^* - N_0^*} & \frac{ae_{N^*} - ae_0}{N_1^* - N_0^*} \end{bmatrix}$$

$$det = \frac{ae_{N^*} - ae_0}{N_1^* - N_0^*} \frac{de_N - de_0}{N_1 - N_0} - \frac{ae_N - ae_0}{N_1 - N_0} \frac{de_{N^*} - de_0}{N_1^* - N_0^*}$$

For convienience the inverted matrix will be noted as

$$J^{-1} = \begin{bmatrix} j_{1,1} & j_{1,2} \\ \\ j_{2,1} & j_{2,2} \end{bmatrix}$$

With the inverted matrix, we can take step in the direction of that slope and calculate the next $[N_2^*, N_2]$

$$\boldsymbol{n}_2 = \boldsymbol{n}_1 - J^{-1}\boldsymbol{e}_1$$

$$\begin{bmatrix} N_2^* \\ \\ N_2 \end{bmatrix} = \begin{bmatrix} N_1^* \\ \\ N_1 \end{bmatrix} - \begin{bmatrix} j_{1,1} & j_{1,2} \\ \\ j_{2,1} & j_{2,2} \end{bmatrix} \begin{bmatrix} ae_1 \\ \\ be_1 \end{bmatrix}$$

$$N_2^* = N_1^* - j_{1,1} * ae_1 - j_{1,2} * de_1$$
$$N_2 = N_1 - j_{2,1} * ae_1 - j_{2,2} * de_1$$

and calculate the attacker and defender energy equations $[ae_2, de_2]$.

After this second step, we need to get the new slope between $[N_1^*, N_1]$ and $[N_2^*, N_2]$. But we do not have to calculate the new Jacobian Matrix from scratch. We can use the Sherman-Morrison formula to update the inverse of the Jacobian (J^{-1}) directly.

$$J_{new}^{-1} = J^{-1} + \frac{\Delta\boldsymbol{n} - J^{-1}\Delta\boldsymbol{e}}{\Delta\boldsymbol{n}^T J^{-1}\Delta\boldsymbol{e}}\Delta\boldsymbol{n}^T J^{-1}$$

$$= \begin{bmatrix} j_{1,1} & j_{1,2} \\ j_{2,1} & j_{2,2} \end{bmatrix} + \frac{\begin{bmatrix} N_2^* - N_1^* \\ N_2 - N_1 \end{bmatrix} - \begin{bmatrix} j_{1,1} & j_{1,2} \\ j_{2,1} & j_{2,2} \end{bmatrix}\begin{bmatrix} ae_2 - ae_1 \\ de_2 - de_1 \end{bmatrix}}{\begin{bmatrix} N_2^* - N_1^* & N_2 - N_1 \end{bmatrix}\begin{bmatrix} j_{1,1} & j_{1,2} \\ j_{2,1} & j_{2,2} \end{bmatrix}\begin{bmatrix} ae_2 - ae_1 \\ de_2 - de_1 \end{bmatrix}}\begin{bmatrix} N_2^* - N_1^* & N_2 - N_1 \end{bmatrix}\begin{bmatrix} j_{1,1} & j_{1,2} \\ j_{2,1} & j_{2,2} \end{bmatrix}$$

$$= \begin{bmatrix} j_{1,1} & j_{1,2} \\ j_{2,1} & j_{2,2} \end{bmatrix} + \frac{\begin{bmatrix} N_2^* - N_1^* \\ N_2 - N_1 \end{bmatrix} - \begin{bmatrix} j_{1,1}(ae_2 - ae_1) + j_{1,2}(de_2 - de_1) \\ j_{2,1}(ae_2 - ae_1) + j_{2,2}(de_2 - de_1) \end{bmatrix}}{\begin{bmatrix} N_2^* - N_1^* & N_2 - N_1 \end{bmatrix}\begin{bmatrix} j_{1,1}(ae_2 - ae_1) + j_{1,2}(de_2 - de_1) \\ j_{2,1}(ae_2 - ae_1) + j_{2,2}(de_2 - de_1) \end{bmatrix}}\begin{bmatrix} (N_2^* - N_1^*)j_{1,1} + (N_2 - N_1)j_{2,1} \\ (N_2^* - N_1^*)j_{1,2} + (N_2 - N_1)j_{2,2} \end{bmatrix}^T$$

$$= \begin{bmatrix} j_{1,1} & j_{1,2} \\ j_{2,1} & j_{2,2} \end{bmatrix} + \frac{\begin{bmatrix} N_2^* - N_1^* - j_{1,1}(ae_2 - ae_1) - j_{1,2}(de_2 - de_1) \\ N_2 - N_1 - j_{2,1}(ae_2 - ae_1) - j_{2,2}(de_2 - de_1) \end{bmatrix}}{denom}\begin{bmatrix} (N_2^* - N_1^*)j_{1,1} + (N_2 - N_1)j_{2,1} \\ (N_2^* - N_1^*)j_{1,2} + (N_2 - N_1)j_{2,2} \end{bmatrix}^T$$

$$\begin{aligned} denom = (N_2^* - N_1^*)&\Big(j_{1,1} * (ae_2 - ae_1) + j_{1,2} * (de_2 - de_1)\Big) \\ +(N_2 - N_1)&\Big(j_{2,1} * (ae_2 - ae_1) + j_{2,2} * (de_2 - de_1) \end{aligned}$$

$$j_{1,1\ new} = j_{1,1} + \frac{\Big(N_2^* - N_1^* - j_{1,1}(ae_2 - ae_1) - j_{1,2}(de_2 - de_1)\Big)\Big((N_2^* - N_1^*)j_{1,1} + (N_2 - N_1)j_{2,1}\Big)}{denom}$$

$$j_{1,2\ new} = j_{1,2} + \frac{\Big(N_2^* - N_1^* - j_{1,1}(ae_2 - ae_1) - j_{1,2}(de_2 - de_1)\Big)\Big((N_2^* - N_1^*)j_{1,2} + (N_2 - N_1)j_{2,2}\Big)}{denom}$$

$$j_{2,1\ new} = j_{2,1} + \frac{\Big(N_2 - N_1 - j_{2,1}(ae_2 - ae_1) - j_{2,2}(de_2 - de_1)\Big)\Big((N_2^* - N_1^*)j_{1,1} + (N_2 - N_1)j_{2,1}\Big)}{denom}$$

$$j_{2,2\ new} = j_{2,2} + \frac{\Big(N_2 - N_1 - j_{2,1}(ae_2 - ae_1) - j_{2,2}(de_2 - de_1)\Big)\Big((N_2^* - N_1^*)j_{1,2} + (N_2 - N_1)j_{2,2}\Big)}{denom}$$

With the updated Jacobian matrix inverse, we repeat the process and calculate the next N_3^* and N_3 and following ae_3 and de_3. Repeat until (N^*, N) stops changing or both ae and de are close enough to zero.

It is possible that the two equations do not have a solution and they will fail to converge. In this case, one of the other three situations will be satisfied.

10.2.1 What If N is Negative?

In this case, we set $N = 0$ and find a N^* that satisfies the attacker energy equation. As mentioned earlier, the attacker energy equation is non-linear and extremely difficult to solve algebraically. With a single equation and single variable, we can use Secant Method [12] to find the root of the equation.

$$E_f^*(N^* - L^*) + \frac{1}{2}DPS\left(T_f^*(N^* - L^*) + T_c^* - I^*V_c^*\right) + 0.5K(N+1)\frac{N^* - L^* + 1}{N^* + 1} - E_c^* = 0$$

Secant method requires to starting points to calculate a slope. We can start with points $N_0^* = 0$ and $N_1^* = E_c^*/E_f^* + 0.5$. Use the two points to calculate the attacker energy equation and we get ae_0 and ae_1. The next point is defined by the equation

$$N_2^* = N_1^* - ae_1\frac{N_1^* - N_0^*}{ae_1 - ae_0}$$

Then we can calculate the next ae_2 and calculate the next N_3^*. This repeats until convergence when N^* stops changing or ae is close to zero.

After convergence, check that the defender energy inequality is true. If not, then continue with the other three situations.

10.2.2 What If N^* is Negative?

In this case, we set $N^* = 0$ and find a N that satisfies the defender energy equation.

$$E_f(N - L) + \frac{1}{2}DPS^*\left((T_f+2)(N-L) + (T_c+2) - I\ (V_c+2)\right) + \frac{0.5}{K}(N^*+1)\frac{N - L + 1}{N + 1} - E_c = 0$$

We use Secant method here and the procedure is similar as the previous section. We start with $N_0 = 0$ and $N_1 = E_c/E_f + 2.5$ and use them to calculate the defender energy equation resulting in de_0 and de_1. The next point is defined as

$$N_2 = N_1 - de_1\frac{N_1 - N_0}{de_1 - de_0}$$

We can calculate de_2 and then N_3 and repeat until convergence. Convergence is when N stops changing or de is close to zero.

10.2.3 What If Both N and N^* are Negative?

There are no equations to solve here, but set $N = 0$ and $N^* = 0$ and double check the energy inequalities to see if this is the satisfied solution from the 4 possibilities. If so, then just calculate DPS and DPS^* using $N = 0$ and $N^* = 0$

10.3 Solving Equations from Section 7.2

First the DPS equations. We use the defender's $P_{f-dodge}$ and $P_{c-dodge}$ formulas

$$DPS^* = \frac{P_f^*N^* + P_c^*}{T_f^*N^* + T_c^* + K(G_f^*N + G_c^*)} \qquad (attacker)$$

$$DPS = \frac{\Big(Q\frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big)N + S\frac{T_c^*}{T_f^* N^* + T_c^*} + U}{(T_f + 2)N + (T_c + 2)} \qquad (defender)$$

Then the energy equations

$$E_f^* N^* + \frac{1}{2}DPS\,\Big(T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)\Big) = E_c^* \qquad (a1:\ attacker)$$

$$E_f N + \frac{1}{2}DPS^*\,\Big((T_f + 2)N + (T_c + 2)\Big) = E_c \qquad (d1:\ defender)$$

And the time equation

$$T_f^* N^* + T_c^* + K(G_f^* N + G_c^*) = K\Big((T_f + 2)N + (T_c + 2)\Big)$$

$$K = \frac{T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)}{(T_f + 2)N + (T_c + 2)} \qquad (time\ alternate\ t1)$$

$$K = \frac{T_f^* N^* + T_c^*}{(T_f + 2)N + (T_c + 2) - G_f^* N - G_c^*} \qquad (time\ alternate\ t2)$$

Solve

$$E_f N + \frac{1}{2}\frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)}\,\Big((T_f + 2)N + (T_c + 2)\Big) = E_c$$
$$(substitute\ DPS\ into\ d1)$$

$$E_f N + \frac{1}{2}(P_f^* N^* + P_c^*)\frac{1}{K} = E_c \qquad (simplify\ using\ t1)$$

$$E_f N + \frac{1}{2}(P_f^* N^* + P_c^*)\frac{(T_f + 2)N + (T_c + 2) - G_f^* N - G_c^*}{T_f^* N^* + T_c^*} = E_c \qquad (substitute\ using\ t2)$$

$$E_f N(T_f^* N^* + T_c^*) + \frac{1}{2}(P_f^* N^* + P_c^*)\Big((T_f + 2)N + (T_c + 2) - G_f^* N - G_c^*\Big) = E_c(T_f^* N^* + T_c^*)$$

$$E_f N(T_f^* N^* + T_c^*) + \frac{1}{2}(P_f^* N^* + P_c^*)\Big((T_f + 2 - G_f^*)N + T_c + 2 - G_c^*\Big) = E_c(T_f^* N^* + T_c^*)$$

$$E_f N(T_f^* N^* + T_c^*) + \frac{1}{2}(P_f^* N^* + P_c^*)(T_f + 2 - G_f^*)N = E_c(T_f^* N^* + T_c^*) - \frac{1}{2}(P_f^* N^* + P_c^*)(T_c + 2 - G_c^*)$$

$$N = \frac{E_c(T_f^* N^* + T_c^*) - \frac{1}{2}(P_f^* N^* + P_c^*)(T_c + 2 - G_c^*)}{E_f(T_f^* N^* + T_c^*) + \frac{1}{2}(P_f^* N^* + P_c^*)(T_f + 2 - G_f^*)}$$

$$N = \frac{\Big(E_c T_f^* - \frac{1}{2}P_f^*(T_c + 2 - G_c^*)\Big)N^* + E_c T_c^* - \frac{1}{2}P_c^*(T_c + 2 - G_c^*)}{\Big(E_f T_f^* + \frac{1}{2}P_f^*(T_f + 2 - G_f^*)\Big)N^* + E_f T_c^* + \frac{1}{2}P_c^*(T_f + 2 - G_f^*)}$$

$$N = \frac{W\ N^* + X}{Y\ N^* + Z}$$

$$W = E_c T_f^* - \frac{1}{2} P_f^* (T_c + 2 - G_c^*)$$
$$X = E_c T_c^* - \frac{1}{2} P_c^* (T_c + 2 - G_c^*)$$
$$Y = E_f T_f^* + \frac{1}{2} P_f^* (T_f + 2 - G_f^*)$$
$$Z = E_f T_c^* + \frac{1}{2} P_c^* (T_f + 2 - G_f^*)$$

$$E_f^* N^* + \frac{1}{2} \frac{\Big(Q \frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big) N + S \frac{T_c^*}{T_f^* N^* + T_c^*} + U}{(T_f + 2) N + (T_c + 2)} \Big(T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)\Big) = E_c^*$$
$$(substitute\ DPS\ into\ a1)$$

$$E_f^* N^* + \frac{1}{2} \Big[\Big(Q \frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big) N + S \frac{T_c^*}{T_f^* N^* + T_c^*} + U\Big] K = E_c^* \qquad (simplify\ using\ t1)$$

$$E_f^* N^* + \frac{1}{2} \Big[\Big(Q \frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big) N + S \frac{T_c^*}{T_f^* N^* + T_c^*} + U\Big] \frac{T_f^* N^* + T_c^*}{(T_f + 2) N + (T_c + 2) - G_f^* N - G_c^*} = E_c^*$$
$$(substitute\ K\ using\ t2)$$

$$E_f^* N^* + \frac{1}{2} \Big[\Big(Q \frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big) \frac{W\ N^* + X}{Y\ N^* + Z} + S \frac{T_c^*}{T_f^* N^* + T_c^*} + U\Big] \frac{T_f^* N^* + T_c^*}{(T_f + 2) \frac{W\ N^* + X}{Y\ N^* + Z} + (T_c + 2) - G_f^* \frac{W\ N^* + X}{Y\ N^* + Z} - G_c^*} = E_c^*$$
$$(substitute\ N)$$

$$\frac{1}{2} \Big[\Big(Q \frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big) \frac{W\ N^* + X}{Y\ N^* + Z} + S \frac{T_c^*}{T_f^* N^* + T_c^*} + U\Big] (T_f^* N^* + T_c^*)$$
$$= (E_c^* - E_f^* N^*) \Big((T_f + 2) \frac{W\ N^* + X}{Y\ N^* + Z} + (T_c + 2) - G_f^* \frac{W\ N^* + X}{Y\ N^* + Z} - G_c^*\Big)$$

$$\frac{1}{2} \Big[\Big(Q T_c^* + R(T_f^* N^* + T_c^*)\Big) \frac{W\ N^* + X}{Y\ N^* + Z} + S T_c^* + U(T_f^* N^* + T_c^*)\Big]$$
$$= (E_c^* - E_f^* N^*) \Big((T_f + 2) \frac{W\ N^* + X}{Y\ N^* + Z} + (T_c + 2) - G_f^* \frac{W\ N^* + X}{Y\ N^* + Z} - G_c^*\Big)$$

$$\frac{1}{2} \Big[\Big(Q T_c^* + R(T_f^* N^* + T_c^*)\Big)(W\ N^* + X) + \Big(S T_c^* + U(T_f^* N^* + T_c^*)\Big)(Y\ N^* + Z)\Big]$$
$$= (E_c^* - E_f^* N^*) \Big((T_f + 2 - G_f^*)(W\ N^* + X) + (T_c + 2 - G_c^*)(Y\ N^* + Z)\Big)$$

$$\frac{1}{2}\Big[\Big(QT_c^* + RT_c^* + RT_f^*N^*\Big)(W\ N^* + X) + \Big(ST_c^* + UT_c^* + UT_f^*N^*\Big)(Y\ N^* + Z)\Big]$$
$$= (E_c^* - E_f^*N^*)\Big[\Big((T_f + 2 - G_f^*)W + (T_c + 2 - G_c^*)Y\Big)N^* + (T_f + 2 - G_f^*)X + (T_c + 2 - G_c^*)Z\Big]$$

$$\frac{1}{2}\Big[(QT_c^* + RT_c^*)W\ N^* + RT_f^*W\ {N^*}^2 + (QT_c^* + RT_c^*)X + RT_f^*XN^*$$
$$+(ST_c^* + UT_c^*)Y\ N^* + UT_f^*Y\ {N^*}^2 + (ST_c^* + UT_c^*)Z + UT_f^*ZN^*\Big]$$
$$= E_c^*\Big((T_f + 2 - G_f^*)W + (T_c + 2 - G_c^*)Y\Big)N^* + E_c^*\Big((T_f + 2 - G_f^*)X + (T_c + 2 - G_c^*)Z\Big)$$
$$-E_f^*\Big((T_f + 2 - G_f^*)W + (T_c + 2 - G_c^*)Y\Big){N^*}^2 - E_f^*\Big((T_f + 2 - G_f^*)X + (T_c + 2 - G_c^*)Z\Big)N^*$$

This results in the quadratic equation, and quadratic formula. However, we almost always take the "plus" solution. The "minus" solution usually results in division by zero.

$$A{N^*}^2 + BN^* + C = 0$$

$$N^* = \frac{-B + \sqrt{B^2 - 4AC}}{2A}$$

$$A = \frac{1}{2}(RT_f^*W + UT_f^*Y) + E_f^*\Big((T_f + 2 - G_f^*)W + (T_c + 2 - G_c^*)Y\Big)$$
$$B = \frac{1}{2}\Big((QT_c^* + RT_c^*)W + RT_f^*X + (ST_c^* + UT_c^*)Y + UT_f^*Z\Big)$$
$$-E_c^*\Big((T_f + 2 - G_f^*)W + (T_c + 2 - G_c^*)Y\Big) + E_f^*\Big((T_f + 2 - G_f^*)X + (T_c + 2 - G_c^*)Z\Big)$$
$$C = \frac{1}{2}\Big((QT_c^* + RT_c^*)X + (ST_c^* + UT_c^*)Z\Big) - E_c^*\Big((T_f + 2 - G_f^*)X + (T_c + 2 - G_c^*)Z\Big)$$

10.3.1 What If N is Negative?

Set $N = 0$ and solve the attacker equation

$$DPS^* = \frac{P_f^*N^* + P_c^*}{T_f^*N^* + T_c^* + K(G_f^*N + G_c^*)} = \frac{P_f^*N^* + P_c^*}{T_f^*N^* + T_c^* + KG_c^*} \qquad (attacker)$$
$$DPS = \frac{\Big(Q\frac{T_c^*}{T_f^*N^* + T_c^*} + R\Big)N + S\frac{T_c^*}{T_f^*N^* + T_c^*} + U}{(T_f + 2)N + (T_c + 2)} = \frac{S\frac{T_c^*}{T_f^*N^* + T_c^*} + U}{T_c + 2} \qquad (defender)$$

The time equation

$$T_f^*N^* + T_c^* + KG_c^* = K(T_c + 2)$$
$$K = \frac{T_f^*N^* + T_c^* + KG_c^*}{(T_c + 2)}$$

$$K = \frac{T_f^* N^* + T_c^*}{T_c + 2 - G_c^*}$$

Solve the attacker energy equation

$$E_f^* N^* + \frac{1}{2} DPS\ (T_f^* N^* + T_c^* + K G_c^*) = E_c^*$$

$$E_f^* N^* + \frac{1}{2} \frac{S \frac{T_c^*}{T_f^* N^* + T_c^*} + U}{T_c + 2}\ (T_f^* N^* + T_c^* + K G_c^*) = E_c^* \qquad (substitute\ DPS)$$

$$E_f^* N^* + \frac{1}{2}\Big(S \frac{T_c^*}{T_f^* N^* + T_c^*} + U\Big) K = E_c^* \qquad (simplify\ using\ time\ equation)$$

$$E_f^* N^* + \frac{1}{2}\Big(S \frac{T_c^*}{T_f^* N^* + T_c^*} + U\Big) \frac{T_f^* N^* + T_c^*}{T_c + 2 - G_c^*} = E_c^* \qquad (substitute\ K)$$

$$\frac{1}{2}\Big(S \frac{T_c^*}{T_f^* N^* + T_c^*} + U\Big)(T_f^* N^* + T_c^*) = (E_c^* - E_f^* N^*)(T_c + 2 - G_c^*)$$

$$\frac{1}{2}\Big(S T_c^* + U(T_f^* N^* + T_c^*)\Big) = (E_c^* - E_f^* N^*)(T_c + 2 - G_c^*)$$

$$\frac{1}{2}(S T_c^* + U T_f^* N^* + U T_c^*) = E_c^*(T_c + 2 - G_c^*) - E_f^* N^*(T_c + 2 - G_c^*)$$

$$E_f^* N^*(T_c + 2 - G_c^*) + \frac{1}{2} U T_f^* N^* = E_c^*(T_c + 2 - G_c^*) - \frac{1}{2}(S T_c^* + U T_c^*)$$

$$N^* = \frac{E_c^*(T_c + 2 - G_c^*) - \frac{1}{2}(S T_c^* + U T_c^*)}{E_f^*(T_c + 2 - G_c^*) + \frac{1}{2} U T_f^*}$$

Check the defender energy inequality

$$\frac{1}{2} DPS^*\ (T_c + 2) > E_c$$

10.3.2 What If N^* is Negative?

Set $N^* = 0$ and solve the defender equation

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)} = \frac{P_c^*}{T_c^* + K(G_f^* N + G_c^*)} \qquad (attacker)$$

$$DPS = \frac{\Big(Q \frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big) N + S \frac{T_c^*}{T_f^* N^* + T_c^*} + U}{(T_f + 2) N + (T_c + 2)} = \frac{(Q + R) N + S + U}{(T_f + 2) N + (T_c + 2)} \qquad (defender)$$

The time equation

$$T_c^* + K(G_f^* N + G_c^*) = K\Big((T_f+2)N + (T_c+2)\Big)$$
$$K = \frac{T_c^* + K(G_f^* N + G_c^*)}{(T_f+2)N + (T_c+2)}$$
$$K = \frac{T_c^*}{(T_f+2)N + (T_c+2) - G_f^* N - G_c^*}$$

Solve the defender energy equation

$$E_f N + \frac{1}{2} DPS^* \ \Big((T_f+2)N + (T_c+2)\Big) = E_c$$

$$E_f N + \frac{1}{2}\frac{P_c^*}{T_c^* + K(G_f^* N + G_c^*)}\Big((T_f+2)N + (T_c+2)\Big) = E_c \qquad (substitute\ DPS)$$

$$E_f N + \frac{1}{2}\frac{P_c^*}{K} = E_c \qquad (simplify\ with\ time\ equation)$$

$$E_f N + \frac{1}{2} P_c^* \frac{(T_f+2)N + (T_c+2) - G_f^* N - G_c^*}{T_c^*} = E_c \qquad (substitute\ K)$$

$$E_f T_c^* N + \frac{1}{2} P_c^* \Big((T_f+2)N + (T_c+2) - G_f^* N - G_c^*\Big) = E_c T_c^*$$

$$E_f T_c^* N + \frac{1}{2} P_c^* (T_f + 2 - G_f^*)N + \frac{1}{2} P_c^* (T_c + 2 - G_c^*) = E_c T_c^*$$

$$N = \frac{E_c T_c^* - \frac{1}{2} P_c^* (T_c + 2 - G_c^*)}{E_f T_c^* + \frac{1}{2} P_c^* (T_f + 2 - G_f^*)}$$

Check the attacker inequality

$$\frac{1}{2} DPS \ \Big(T_c^* + K(G_f^* N + G_c^*)\Big) > E_c^*$$

10.3.3 What If Both N and N^* are Negative?

Set $N^* = 0$ and $N = 0$. The DPS equations are immediate.

$$DPS^* = \frac{P_f^* N^* + P_c^*}{T_f^* N^* + T_c^* + K(G_f^* N + G_c^*)} = \frac{P_c^*}{T_c^* + K G_c^*} \qquad (attacker)$$
$$DPS = \frac{\Big(Q\frac{T_c^*}{T_f^* N^* + T_c^*} + R\Big)N + S\frac{T_c^*}{T_f^* N^* + T_c^*} + U}{(T_f+2)N + (T_c+2)} = \frac{S+U}{T_c+2} \qquad (defender)$$

The time equation

$$T_c^* + KG_c^* = K(T_c + 2)$$
$$K = \frac{T_c^*}{T_c + 2 - G_c^*}$$

It is important to check the energy inequalities. They can fail.

$$\frac{1}{2}DPS\ (T_c^* + KG_c^*) > E_c^* \qquad (attacker)$$
$$\frac{1}{2}DPS^*(T_c + 2) > E_c \qquad (defender)$$

10.4 Solving Equations from Section 7.3

We need to solve the two energy equations for N^* and N

$$E_f^*(N^* - L^*) + \frac{1}{2}DPS\ \left(T_f^*(N^* - L^*) + T_c^* - I^*V_c^* + K(G_f^*N + G_c^*)\right) + 0.5K(N+1)\frac{N^* - L^* + 1}{N^* + 1} = E_c^* \qquad (attacker)$$
$$E_f(N - L) + \frac{1}{2}DPS^*\ \left((T_f + 2)(N - L) + (T_c + 2) - I\ (V_c + 2)\right) + \frac{0.5}{K}(N^* + 1)\frac{N - L + 1}{N + 1} = E_c \qquad (defender)$$

As a reminder, the DPS equations also change

$$DPS^* = \frac{P_f^*N^* + P_c^*}{T_f^*N^* + T_c^* + K(G_f^*N + G_c^*)} \qquad (attacker)$$
$$DPS = \frac{P_{f-dodge}N + P_{c-dodge}}{(T_f + 2)N + (T_c + 2)} \qquad (defender)$$

As mentioned in previous text, these is very difficult to solve algebraically. We can use Broyden's Method to solve it. Follow the procedure described in detail in section 10.2 but use the Energy and DPS equations from this section which include adjustments for dodging

10.4.1 What If N is Negative?

Set $N = 0$ and solve N^* in the attacker energy equation. We can use the Secant Method which is described in section 10.2.1. Remember to use the equations from this section instead.

10.4.2 What If N^* is Negative?

Set $N^* = 0$ and solve N in the defender energy equation. We can use the Secant Method which is described in section 10.2.2. Remember to use the equations from this section instead.

10.4.3 What If Both N and N^* are Negative?

There are no equations to solve here, but set $N = 0$ and $N^* = 0$ and check the energy inequalities to see if this is the satisfied solution from the 4 possibilities. If so, then just calculate DPS and DPS^* using $N = 0$ and $N^* = 0$

Table 1: Move notation

Move	Damage	Energy	Duration	Damage Window Start
Attacker Fast	P_f^*	E_f^*	T_f^*	V_f^*
Attacker Charge	P_c^*	E_c^*	T_c^*	V_c^*
Defender Fast	P_f	E_f	T_f	V_f
Defender Charge	P_c	E_c	T_c	V_c

10.5 Variables

$Damage =$ Damage done (3.1)

$Energy =$ Energy gained if fast move, Energy used if charge move (4.1)

$Duration =$ Length of time for the move (4.1)

$Damage\ Window\ Start =$ How long after move starts when the damage is done. (6.4)

$DPS^* =$ Damage per second of the attacker (4.1)

$DPS =$ Damage per second of the defender (4.1)

$N^* =$ Number of attacker's fast moves per one attacker charge move (4.1)

$N =$ Number of defender's fast moves per one defender charge move (4.1)

$L^* =$ Number of moves lost by attacker after getting to one charge move of energy (5)

$L =$ Number of moves lost by defender after getting to one charge move of energy (5)

$I^* =$ Probability of attacker losing energy after getting to one charge move of energy (6.4)

$I =$ Probability of defender losing energy after getting to one charge move of energy (6.4)

$Q, R, S, U =$ Constants related to dodging (7.1)

$P_{f-dodge} =$ Damage from the defender's fast move if the attacker is dodging (7.1)

$P_{c-dodge} =$ Damage from the defender's charge move if the attacker is dodging (7.1)

$K =$ Number of defender fast/charge cycles for one attacker fast/charge cycle (7.2)

$G_f^* =$ Time spent by attacker dodging defender's fast moves (7.2)

$G_c^* =$ Time spent by attacker dodging defender's charge moves (7.2)

$TDO =$ Total damage output (8.1)

$DO@X =$ Damage output after X seconds (8.2)

$TTF =$ Time to faint (8.3)

References

[1] `https://pokemongo.gamepress.gg/gobattlesim`

[2] `https://www.pokebattler.com/`

[3] https://web.archive.org/web/20160815210025/http://pokemongo.gamepress.gg/optimal-moveset-explanation

[4] https://elite4insider.com/

[5] https://pokemongo.gamepress.gg/gym-combat-mechanics

[6] https://pokemongo.gamepress.gg/new-discoveries-theory-battle-mechanics

[7] https://pokemongo.gamepress.gg/damage-mechanics

[8] https://pokemongo.gamepress.gg/pokemon-go-trainer-battle-guide

[9] https://pokemongo.gamepress.gg/how-calculate-comprehensive-dps

[10] https://pokemongo.gamepress.gg/how-to-dodge

[11] https://en.wikipedia.org/wiki/Broyden's_method

[12] https://en.wikipedia.org/wiki/Secant_method