Differential equations for mint, burn and transfer:

CADcad consists of a simulated development of the token economy over time. In order for CADcad to know how to simulate the system, it is necessary to specify according to which mathematical relation the economy changes over time. Given that CADcad proceeds through iteration it is useful to conceive of time in a discrete fashion, namely dividing it into specific units of time, as the transition from one steady state to the next. To capture the dynamic evolution of the token economy, we used differential equations describing the transition between states.

By *State* we mean a set of objects which fully describe the token economy at a certain point in time. By *Economy* we mean relative credit/debt position of each actor in the Trojan DAO together with a measure describing the value of a credit accumulated in the Trojan DAO token economy.

Note that in our model we have assumed a flat and constant bonding curve for simplicity. In absence of a redistributive tax, the system would therefore trade tokens for ETH at a fixed rate, without any direct privilege to first adopters. However, given that for every new entry in the system part of the tokens are distributed to existing users, the initial adopters still benefit from the fact they invested first, and therefore are incentivized to onboard new members.

A vector $\boldsymbol{x}(t)$ of $N(t)$ components was used to model $N(t)$ existing Trojan members in time t. The DAO pool $D(t)$ can be considered to be an additional actor in the system, collecting redistributive taxes and transaction taxes in order to affect the distribution of credit in the Trojan economy. On top of these distributional measures, we have the reserve $R(t)$ containing the Ether which backs the tokens. Therefore, we can uniquely describe a state t of the system with the triple $(\boldsymbol{x}, D, R)(t)$.

Given the blockchain nature of the economy, each state in time $t+1$ can be deterministically derived from the knowledge of the state in time t, together with the information concerning all transactions between token holders together with all the burn/mint action affecting shares and/or token in time t. Therefore, defining $b(t)$ as the amount of Ether provided in a *mint* token event, $c(t)$ as the number of tokens burned in a burn token event, $(\boldsymbol{x}, D, R)(t+1)=f((\boldsymbol{x}, D, R)(t); (b, c)(t))$, meaning that, knowing $b(t)$ and $c(t)$ we can uniquely determine the state in $t+1$.

For the purpose of simplicity in running the code, we assumed to transition from a state to the next when any action is carried out, namely mint/burn tokens and mint/burn shares. This demands to model the random process according to which events happen in time and how preceding events influence the likelihood of future ones, e.g. if there have been many burn shares it is more likely for new burn shares to happen, given the bank run effect.

We break down the following scenarios into their differential equations:

Mint token:

Investors provide ETH in exchange for tokens. The amount of tokens they get is worth *less* than their investment in ETH, since two fees are applied, a DAO tax of 2%, which goes to replenish the 'DAO pool', and a redistribution tax of 1%, which is redistributed to all existing token holders in proportion to the proportion of tokens held by each holder.

$$\Delta D = \theta_1 b_t + \frac{\theta_2 b_t D_t}{k M_t}$$

$$\Delta R = b_t$$

$$\Delta x_i = x_{t,i} \frac{b_t \theta_2}{k M_t}$$

$$\Delta x_j = \frac{b_t}{k}(1 - \theta_1 - \theta_2)$$

Where *D* is the "DAO pool", *R* is the bonding curve reserve and x is the number of tokens of an individual, *i* being any generic member in the DAO apart from the member which is minting, namely *j*. $\theta 1$ is the DAO pool tax and $\theta 2$ is the redistributive tax.

Burning tokens:

Token holders can burn their tokens to get back an equivalent amount of ETH from the token reserve minus a "DAO tax" of 3%

$$\Delta D = c_t \theta_3$$
$$\Delta R = -kc_t(1 - \theta_3)$$
$$\Delta x_i = 0$$
$$\Delta x_j = -c_t$$

Where *c* is the number of tokens burned by the user, *i* is a generic member and *j* is the member which is burning the token. $\theta 3$ is the DAO tax on burning.

Minting of shares:

Members willing to acquire a stake in the Trojan DAO by supplying work/artistic creation or financial means can acquire shares allowing them to participate in the governance of the DAO:

$$\Delta x_j = -d$$
$$\Delta x_i = 0$$
$$\Delta y_j = z$$
$$\Delta y_i = -zy_{t,i}$$
$$\Delta B = d$$

Where *z* is the number of shares assigned to the contributor by the existing members according to a social decision function *f*, such that $z = f(W, E)$, where *W* is the amount of tokens offered as Tribute and *E* is a measure of the value of the contribution. The function *f* is decided by the Trojan DAO members in an arbitrary fashion. In the above equations, *d* is the deposit required by the Trojan Bank to those willing to acquire a share. *B* is the amount of tokens inside the Trojan Bank. And *y* is the number of shares of the DAO, *i* being the existing members and *j* being the member which is joining or is increasing their ownership of shares.

In the case of financial distribution, equation $\Delta B = d$ is to be substituted by equation $\Delta B = d+K$, where *K* is the number of tokens the financial contributor is offering as "tribute" to the Trojan Bank.

Burning of shares (Ragequitting):

When DAO member has a stake in the system they can decide to liquidate part or all of their share position by transforming the shares into TROJ tokens. This decision by a generic user *j* to do so increases his token position by an amount proportional to the relative amount shares burnt *ht* on the total existing shares *Qt* and to the sum of the Trojan Bank and the Bonding Curve Reserve contract. The total tokens in the Trojan Bank, therefore, decrease proportionally to the relative fraction of overall shares that were burnt.

$$\Delta x_j = \frac{h_t}{Q_t}(B_t + D_t)$$

$$\Delta x_i = 0$$

$$\Delta y_j = -h_t$$

$$\Delta y_i = 0$$

$$\Delta B = \Delta D = -\frac{h_t}{Q_t}$$

$$\Delta Q = -h_t$$