

# 0SumBot

very tiny documentation

Aly Shmahell

”aly.shmahell@gmail.com”

Department of Computer Science, University of L’Aquila

December 18, 2017

## **Abstract**

0SumBot can be described as a sum of two varied approaches to create a bot capable of playing, and sometimes, preferably most times, winning the PlanetWars contest.

In this effort The need to optimize parameters lead to the creation of 0SumBot, 1SumBot, G0Bot and G0GA.

## 0.1 General Description

0SumBot depends on the following in-game variables: Planet.NumShips(), Planet.GrowthRate(), PlanetWars.Distance(Planet,otherPlanet).

In order to achieve victory first we set a series of optimized versions of these parameters and compare them to the in-game parameters and then we set (Base, Target, FleetSize) based on the results.

One way of comparison is by calculating the localMinima of distance between base and enemy planets, and basing our choice of target on the result.

Another method that is followed is comparing each planet directly to optimized parameters to formulate a plan of attack.

Methods of comparison are not totally exclusive, and we usually use them entwined.

1SumBot is an alteration of 0SumBot that includes a localMinima for (Base, Enemy), (Base, Neutral) and (Base, Ally), then changes the strategy to include these additional parameters.

G0Bot is a copy of 0SumBot used for the G0GA genetic algorithm.

G0GA is a genetic algorithm designed for the purpose of fine-tuning the parameters of G0Bot for a more efficient battle strategy.

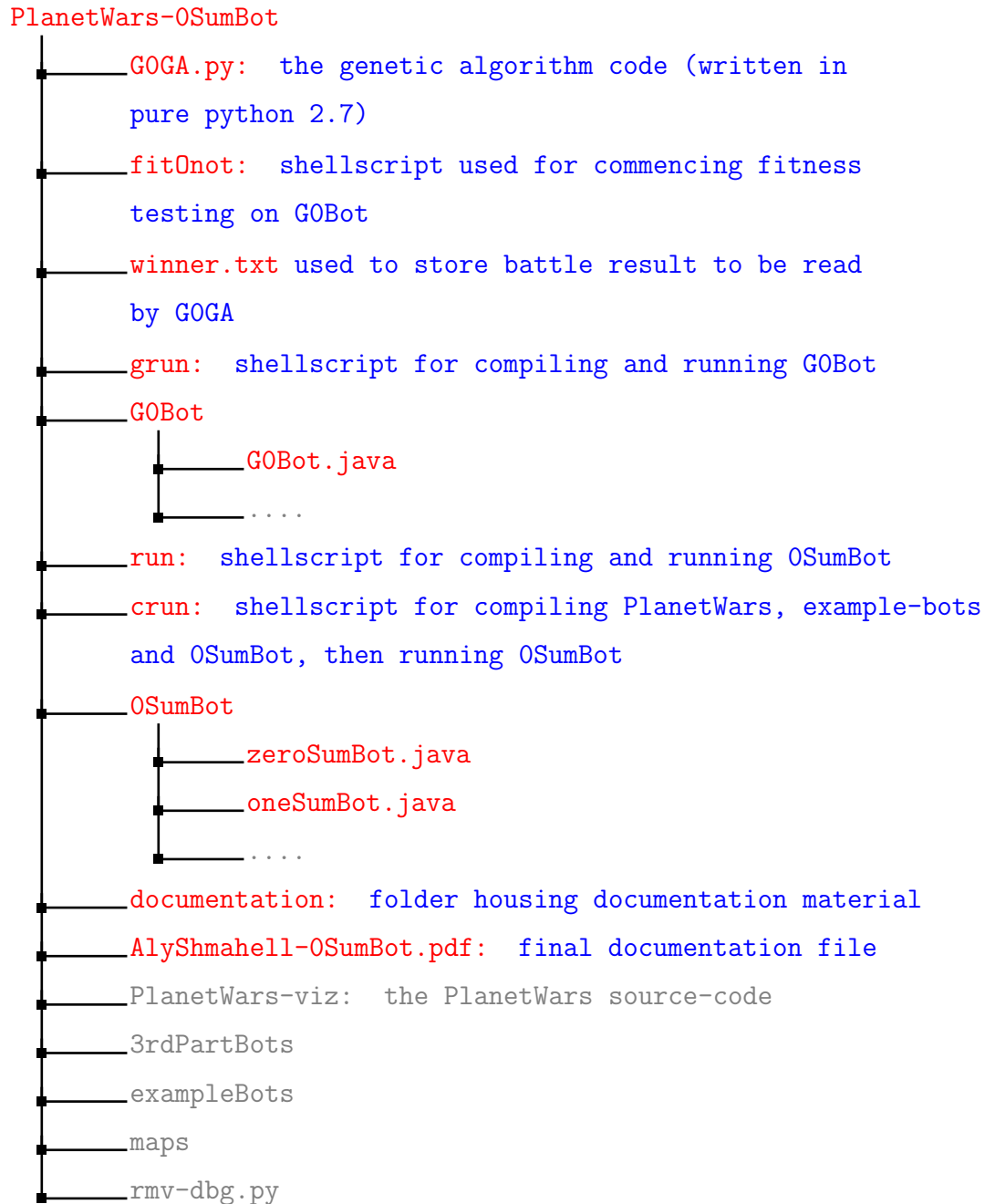
## 0.2 Notes on the bots

In order to avoid player timeout, make sure to check if planet is null before you make any operation involving that selected planet.

For example, when calculating local minima in 0SumBot it is essential to perform the following check beforehand:

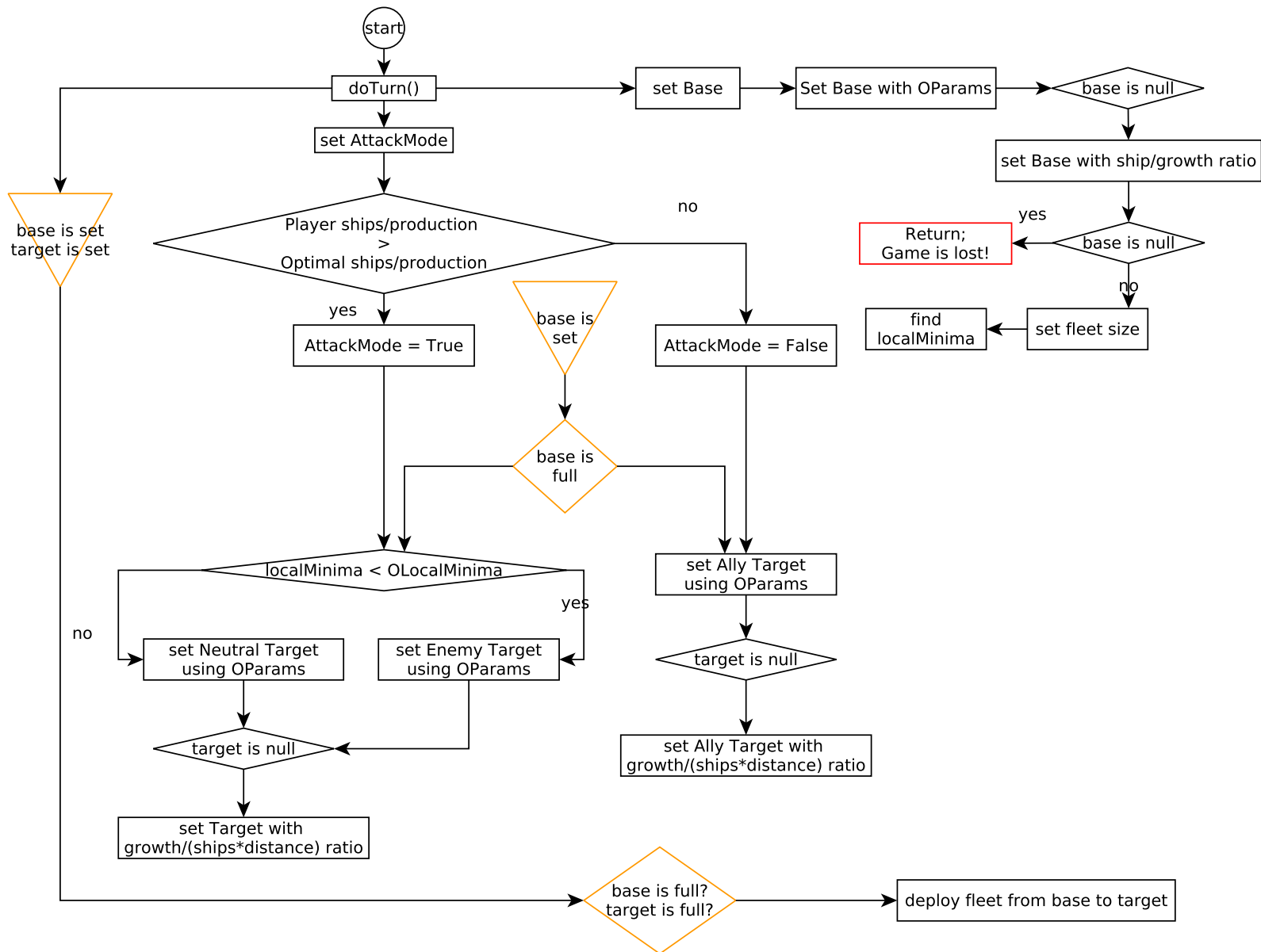
```
if (closestEnemy==null)
    return;
localMinima = pw.Distance(base,closestEnemy);
```

## 0.3 file description



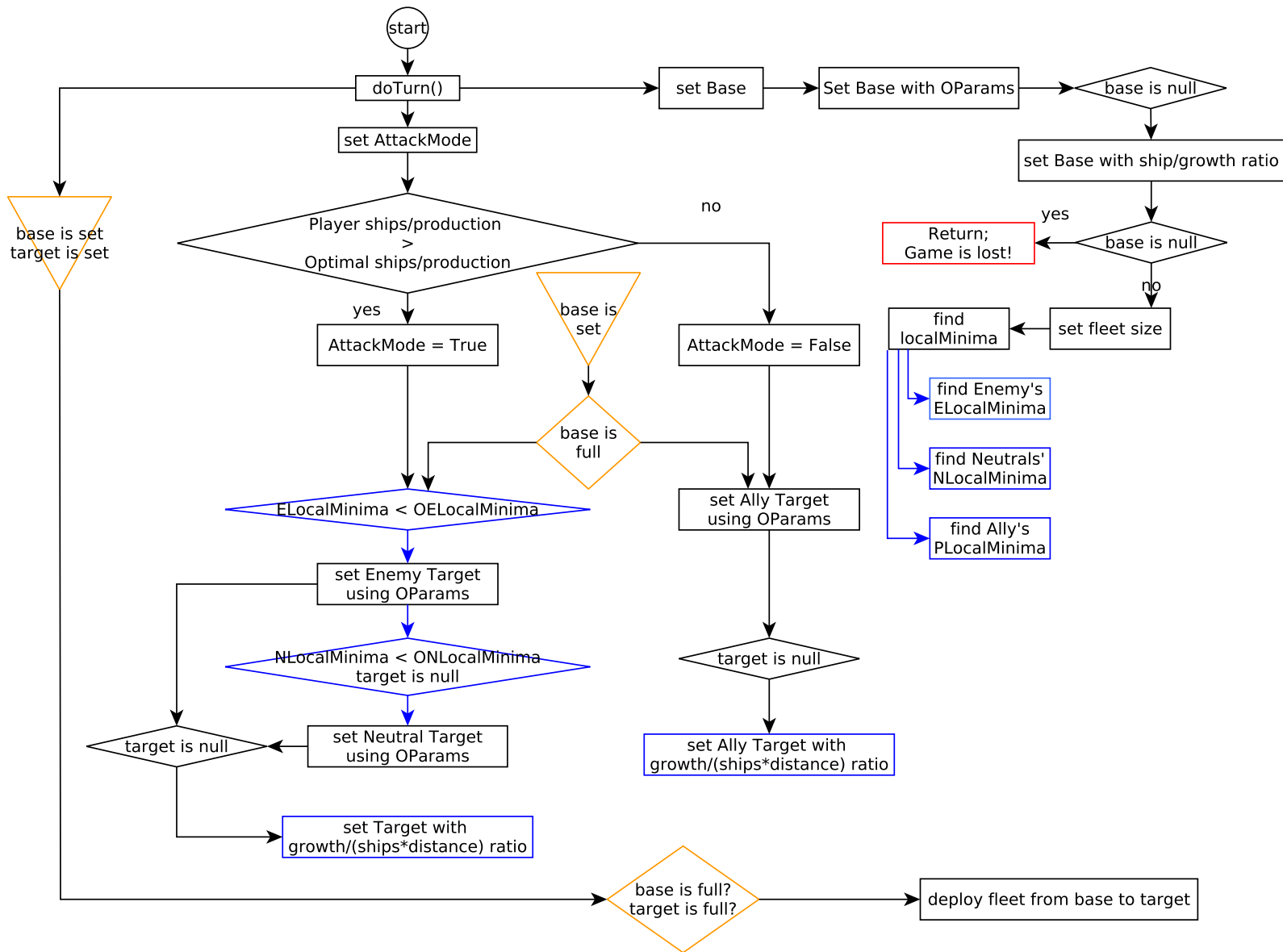
## 0.4 OSumBot

OSumBot Work-Flow



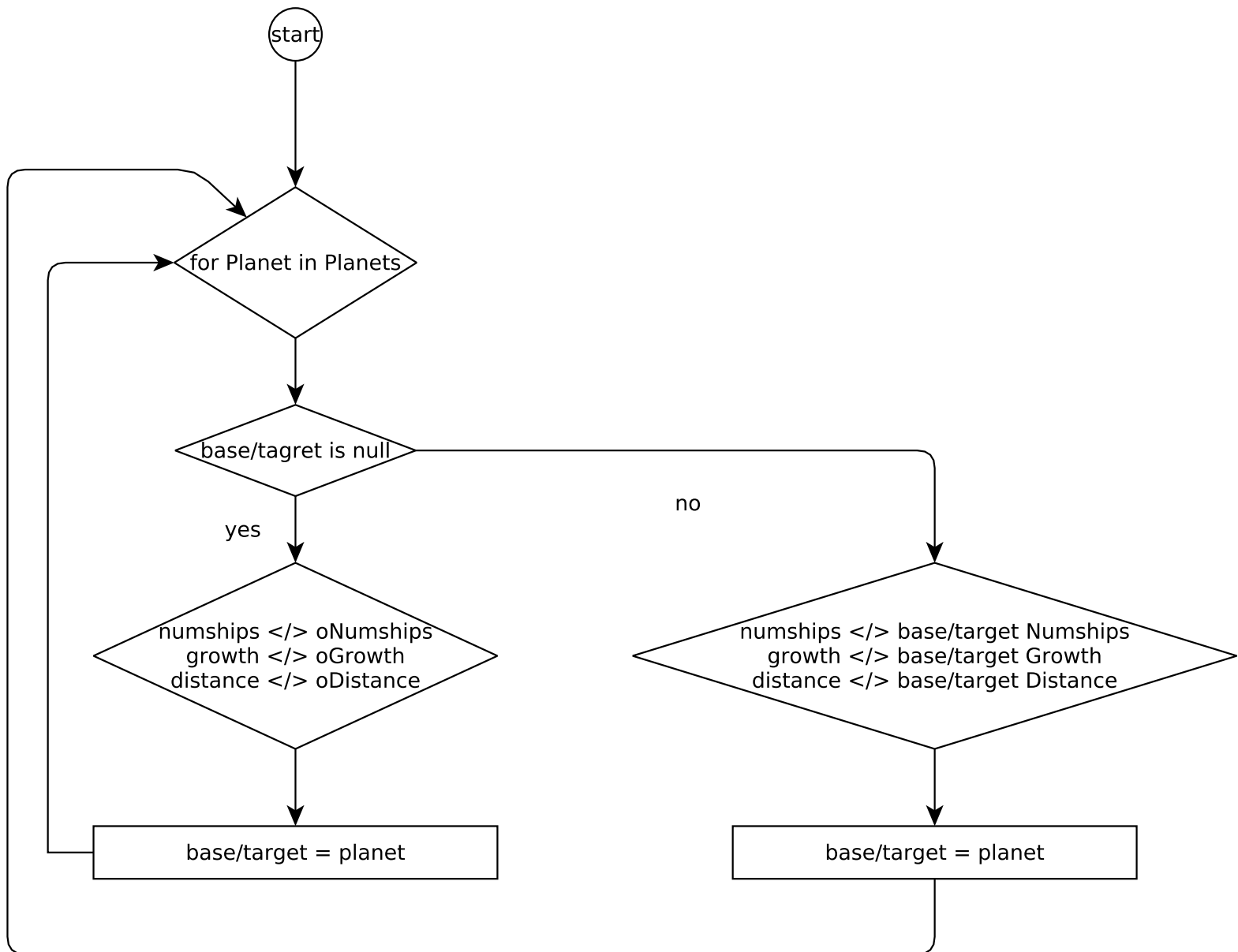
## 0.5 1SumBot

1SumBot Work-Flow: Changes from 0SumBot are in blue color.



## 0.6 Choosing planet according to optimized parameters

Choosing Base/Target Planet with Optimized Parameters



## 0.7 G0GA

G0GA is a steady-state 4-k tournament genetic algorithm improvised to evolve G0Bot.

Written in python 2.7, it performs fitness calculation by means of multiple bot simulations aided by helper bash scripts.

The GA performs parameter passing to the bot and to the helper scripts by means of line replacement directly on the target files (parameter injection).

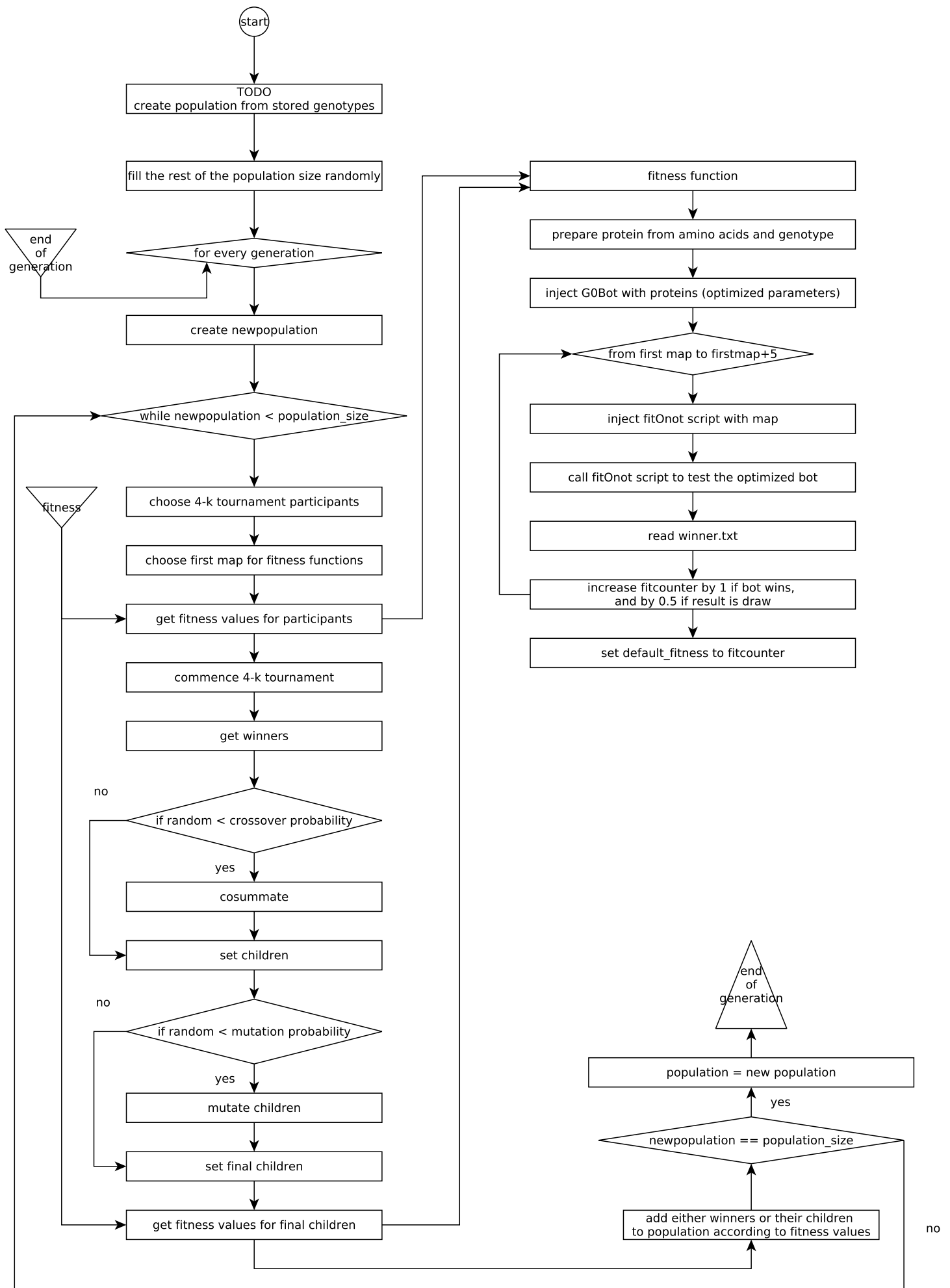
### 0.7.1 Notes on the GA

Always make sure to run as few calls to fitness as possible, in order to reduce simulation time and therefor decrease computation time.

This can be achieved by:

- Inside the generation generator :
  - Assigning fitness values to fitness variables and reusing them.
  - Assigning fitness values only when needed.
  - Only assigning fitness values when no change on the genotype is due later.
- Inside the individual:
  - Assigning fitness value to internal `default_fitness` variable.





## 0.8 What's next?

- Adding memory capability to G0GA, so that it records the best genotypes from the current run session in order for it to use them in creating individuals for future run sessions.
- Using Linear Regression in G0Bot and in G0GA to increase performance.
- Redefining battle strategy in G0Bot based on fuzzy functions.
- Improving the map shuffling process in G0GA.
- Accessing and use more in-game variables in G0Bot, like Fleet variables for example.

## 0.9 Code Repository

A Github repository exists for this project under:

`https://github.com/AlyShmahell/PlanetWars-0Sum`