Overview

Thursday, November 3, 2022 1:09 PM

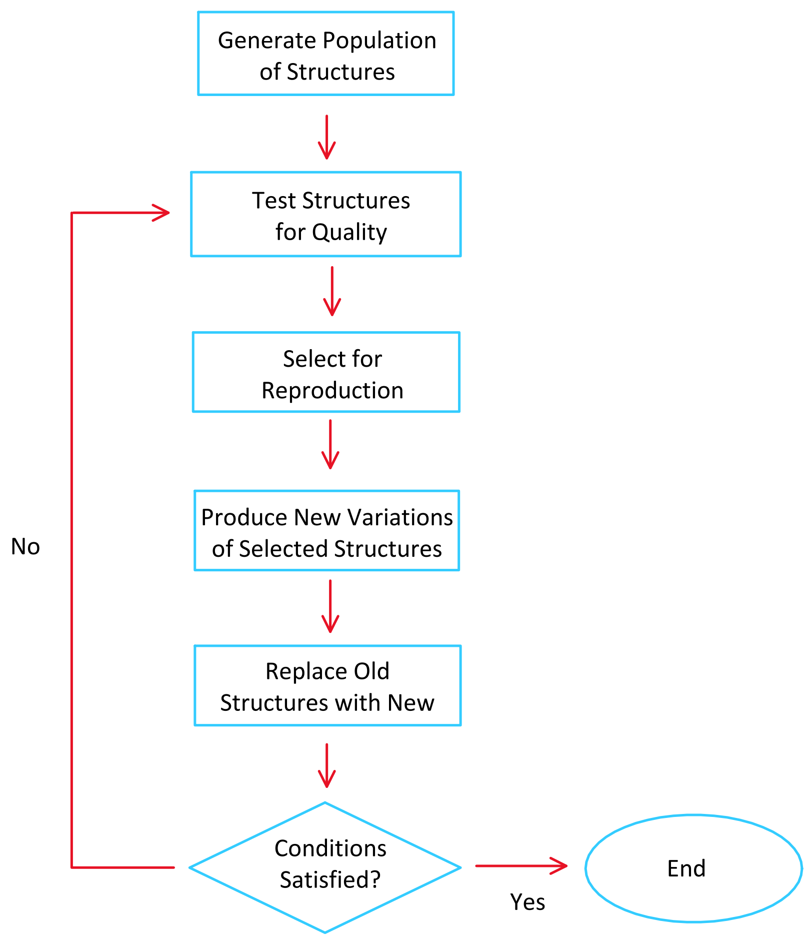
Founder: John Holland 1970's

Advantages:

- Ignores discontinuities
- Since it uses random methods, it has good exploration vs exploitation
- Intuitive methods based on theories in biology
- Object-oriented creatures
- Embarrassingly parallel

Disadvantages:

- Computationally intensive



Optimization: the search for the "best" solution for a problem
Fitness function: evaluates how close a given solution is to the optimum solution of the problem
Decision variables: independent variables to be manipulated when searching for the optimum solution
Constraint: restriction placed on decision variables or solution
Feasible solution: solution that does not violate any constraints
Search space: all values that the decision variables can assume
Mutation: (unary variation) changing a single structure
Crossover: (binary variation) exchanging material between structures
Penalty function: reduces the value of the fitness function when a constraint is violated
Niching: reducing fitness of population members in proportion to the number of other solutions that are the same. This makes solutions less good once they have been discovered by several other members of the population. This maintains population diversity, allowing to reach a global optima
Fitness landscape: graph of the fitness function of the entire space of data structures that are members of a population

Gene: a single encoded parameter
Chromosome: A set of genes making up an individual in a population
Simple Genetic Algorithm (SGA)
Steady State Genetic Algorithm (SSGA): No generations. Only the best fit individuals are added back into the population.

Selection Methods

Thursday, November 17, 2022 5:08 PM

Bad selection can result in premature convergence at a local optima

n-tournament: the Population is shuffled randomly and divided into small groups. The most fit individuals in each small group are chosen tobe parents.

Advantages:

- The most fit genes are guaranteed to survive
- Efficient and simple

1. Choose some number of individuals randomly from a population (with or without replacement)
2. Select the best individual from this group
3. Repeat as often as desired (usually until the mating pool is filled)

Roulette wheel/Proportionate: chooses parents in proportion to their fitness.

Probability of selection = gene fitness / sum of population fitness

$p(i) = \frac{f(i)}{\Sigma f}$

- Slow

Roulette Wheel Parent Selection

1. **Sum the fitnesses of all the population members; call the result total fitness.**
2. **Generate n, a random number between 0 and total fitness.**
3. **Return the first population member whose fitness, added to the fitnesses of the preceding population members, is greater than or equal to n.**

Examples of Roulette Wheel Parent Selection

Chromosome	1	2	3	4	5	6	7	8	9	10
Fitness	8	2	17	7	2	12	11	7	3	7
Running Total	8	10	27	34	36	48	59	66	69	76

Random Number	23	49	76	13	1	27	57
Chromosome Chosen	3	7	10	3	1	3	7

Figure 1.5: Examples of roulette wheel parent selection. The first table shows the fitness of ten chromosomes and the running total of fitness. The second table shows the chromosome that would be chosen by the roulette wheel algorithm using these fitness values for each of seven randomly generated numbers.

From "Handbook of Genetic Algorithms" by Lawrence Davis 1991

Rank selection: ordered by fitness and selected by rank (1-least fit, n-most fit

Probability of selection = gene rank/sum of population ranks

1. Rank population members
2. Generate random number between 0 and population size
3. Return the member whose rank equals the random number

Linear Ranking and binary tournaments have been shown to have identical performance

Absolute fitness replacement: replacing the least fit members of the population

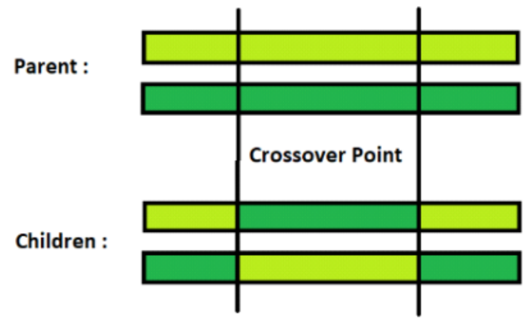
Locally elite replacement: The two parents and the two children are examined in the two most fit are put into the population in the slots occupied by the parents

Goldberg 1991:
https://www.cse.unr.edu/~sushil/class/gas/papers/Select.pdf

Crossover Types

Thursday, November 17, 2022 5:25 PM

n-point: n random points are chosen along a chromosome string and genetic material is exchanged between two parents



Uniform: Each gene is selected randomly from the genes of the parents (idea: have a probability based on fitness instead of random). Uses a binary mask, which is like a coin flip. Masks are selected every time crossover happens. Each mask value is random (value of .5, which makes it uniform)

Adaptive Methods:

"This can be achieved in a brute force manner by repeated calls to crossover until non-clones are produced, or in a more sophisticated fashion, such as Booker's reduced surrogate approach." - Dejong K, Spears WM. An analysis of the interacting roles of population size and crossover in genetic function optimization. 1990.

n-point having the point location as encoded material

Genetic material can determine what method to use! Pretty cool

Also:
Random elite: each child is compared with a random member of the population and is replaced only if it is least as fit

p_c is typically between .5 and 1.0

Mutation and Population Size

Thursday, November 17, 2022 5:45 PM

Mutation rate can theoretically reduce population size by increasing the number of new cases, but this risks the possibility of losing a good solution

Mutation typically occurs during reproduction

Real number creep: If a function is likely continuous, then it is good to jump around the current solution in small, random increments
Too small of a population size can introduce genetic drift

50 seems to be a decent start (number of bit lengths?)

p_m (probability of mutation) is typically between .001 and .005

D. Thierens, "Adaptive mutation rate control schemes in genetic algorithms," Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600), 2002, pp. 980-985 vol.1, doi: 10.1109/CEC.2002.1007058.

Multi-Objective Optimization

Wednesday, August 23, 2023 3:33 PM

Subscripts

i	gene or decision variable index
j	chromosome index

Superscripts

n	GA generation or population index

Each chromosome has a fitness vector $\mathbf{F}_j^n(x_1, x_2, \ldots, x_n)$

Given:

If a lower x_1 and x_2 is better, then

$\mathbf{F}_1^n$ $(x_1, x_2)= (2,10)$ -> non-dominated

$\mathbf{F}_2^n$ $(x_1, x_2)= (4,6)$ -> non-dominated

$\mathbf{F}_3^n$ $(x_1, x_2)= (8,4)$ -> non-dominated

$\mathbf{F}_4^n$ $(x_1, x_2)= (9,5)$ -> dominated

$\mathbf{F}_5^n$ $(x_1, x_2)= (7,8)$ -> dominated

Goldberg Ranking:

1. Identify all non-dominated individuals in population
2. Assign them a rank of 1
3. Temporally remove from population
4. Repeat previous steps on the remaining set of non-dominated individuals until all have been assigned a rank (next round gets assigned a rank of 2)

This scheme assures equal reproductive potential.

Goldberg, D. E.: Genetic Algorithms in Search, Optimization and Machine Learning. AddisonWesley, Reading, MA, 1989,

Bin/Block Ranking:

1. Use greedy selection until the number of points reaches a certain threshold value
2. Bins are "constructed" by dividing the design space into (M_{seg})^N_O =5^2=25 bins equally sized
3. If a bin is "full", or has a number of points belonging to it that is higher than some predetermined value, block all selections from that bin.
4. Randomly choose a nondominated chromosome from the accumulation file?

Pareto Optimal Set/Front

Friday, November 18, 2022 6:07 PM

"Includes optimal solutions for each of the individual objectives, as well as a range of tradeoff solutions in between, which are themselves optimal solutions. Providing a range of solutions to a multi-objective optimization problem is a powerful approach because it allows the designer to see the effect of decision variable variation on the design space in the form of optimal tradeoffs. Thus, the designer can choose individual objective weighting factors after their full influence is quantitatively known."

Genetic Algorithms Applied to Multi-Objective Aerodynamic Shape Optimization, Terry L. Holst

Pareto Optimality

Preventing Premature Convergence

Saturday, December 3, 2022 10:55 AM

Goal: Maintain population diversity and selection pressure

Genetic Drift: The tendency to converge on a single solution as the diversity of a population decreases
Fitness sharing: a type of niching, where the fitness of each individual is scaled based on its proximity to others.
Deceptive: A class of challenging problems for conventional genetic algorithms which usually mislead the search to some local optima rather than the global optimum.
Homogeneous vs. **Heterogenous**

From <https://ieeexplore.ieee.org/document/4630900>

From <https://stackoverflow.com/questions/37836751/what-are-fitness-sharing-and-niche-count-in-evolutionary-computation>

Review of 24 different approaches:
Pandey, Hari Mohan et al. “A comparative review of approaches to prevent premature convergence in GA.” *Appl. Soft Comput.* 24 (2014): 1047-1077.

From <https://www.semanticscholar.org/paper/A-comparative-review-of-approaches-to-prevent-in-GA-Pandey-Chaudhary/590a7e80ad9fd564e79129acd978bd9d02fe710b>

Determine convergence by subtracting avg fitness from max fitness. If it is low, then solution is likely well converged
Or by finding avg difference between genes

Metaselection algorithm: To share time among several evolving populations and delete older ones. Complex. Use if no progress has been made for several generations, or stopping conditions have not been met

Encoding

Saturday, December 3, 2022 11:23 AM

"The hardness of a problem is inherently related to the way we represent the problem. Therefore, the difficulty level of the problem can increase or decrease based on the representation" -
Pandey, Hari Mohan et al. “A comparative review of approaches to prevent premature convergence in GA.” *Appl. Soft Comput.* 24 (2014): 1047-1077.

Also, check out
Malik, Ms. Shikha and Mr. Sumit Wadhwa. “Preventing Premature Convergence in Genetic Algorithm Using DGCA and Elitist Technique.” (2014).

Talk about masking parameters

Penalty Functions

Friday, December 16, 2022 3:44 PM

1. Reject all unfeasible solutions (or a percentage of them) [not really a penalty function]
2. Repair unfeasible solutions by finding the closest feasible one
3. Superiority of feasible solutions over the unfeasible

Hyperparameter Control

Friday, December 16, 2022 3:47 PM

"The parameter control is a challenge in evolutionary computing for increasing the performance and obtaining a “universal” genetic algorithm, to be able to solve problems without manual parameter settings. The goal is to find not only the proper adjustments, but to do this in an efficient way." -
R. Maniu and L. A. Dumitru, "Genetic algorithm - adaptive crossover based on solution distribution in search space," *2017*
From <https://ieeexplore.ieee.org/document/7975084>

Deterministic: This rule modifies the strategy parameter in a fixed, predetermined (i.e., user-specified) way without using any feedback from the search.
Adaptive: Algorithm parameters are modified based on feedback about algorithm evolution
Static: Parameters are preset and do not change
Self-adaptive: Hyperparameters themselves are genetic material changed through evolution (Ex. genetically encoded mutation rates per individual)

Electrostatic Sieve Optimization Setup

Tuesday, December 20, 2022 4:15 PM

1. Need to figure parameters ✓
2. File I/O ✓
3. Fitness and param array ✓
4. set up hyperparameters + end condition ✓
5. run init pop. ✓
6. Fitness Function ✓
7. selection
8. crossover
9. mutation

$$\text{fitness} = \%\,\text{classified} \cdot \frac{1}{\text{abs}(\text{max\_ited\_class} - \text{mean\_class})}$$

Parameters:

Number of electrodes (change thickness, pitch, to be constrained by a penalty or basic functions

(also blah location as function to keep in same place)

Potential (float, not constrained [yet])

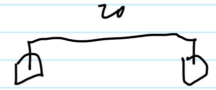
Phases (2,3,4) (Integer)

Frequency (make sure time is consistent)

ambient or continuous injection (binary)

$$1 + \frac{\text{size} - \text{thick}}{\text{pitch}} \qquad \text{rand 1st}$$

$$\text{size} = \text{pitch} \cdot (\text{num} - 1) + \text{thickness} \qquad 1-10 \text{ integer}$$



thick — num_cells

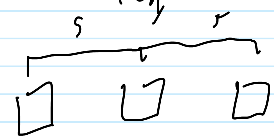
$$\text{pitch}: \frac{\text{size} - \text{thic}}{\text{num} - 1}$$

then check if size is over constrained

$$\frac{size \cdot thick}{num - 1}$$

$$n\_it = \frac{tot\_time}{dt}$$

$$dt = \frac{1}{freq} \cdot \frac{1}{n\_fall} \cdot \frac{1}{n\_it\_per\_phase}$$



pitch: size

Parallel GA

Wednesday, August 23, 2023 4:13 PM

Ridiculously parallel; a maximum speedup of the population size/number of processing elements (PEs) can be seen

Worker-master:
Master controls reproduction, selection, and genetic operators. Each worker process evaluates a single (or more) function evaluation.

This…

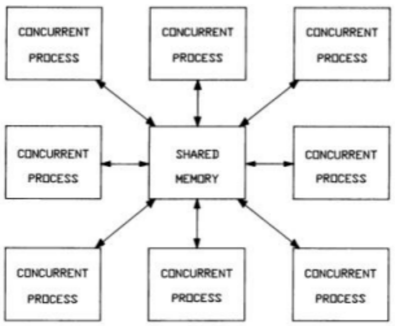
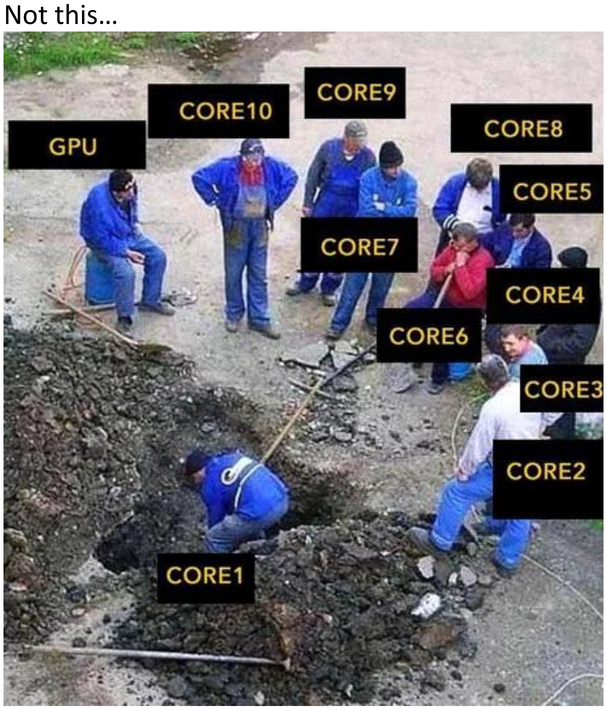


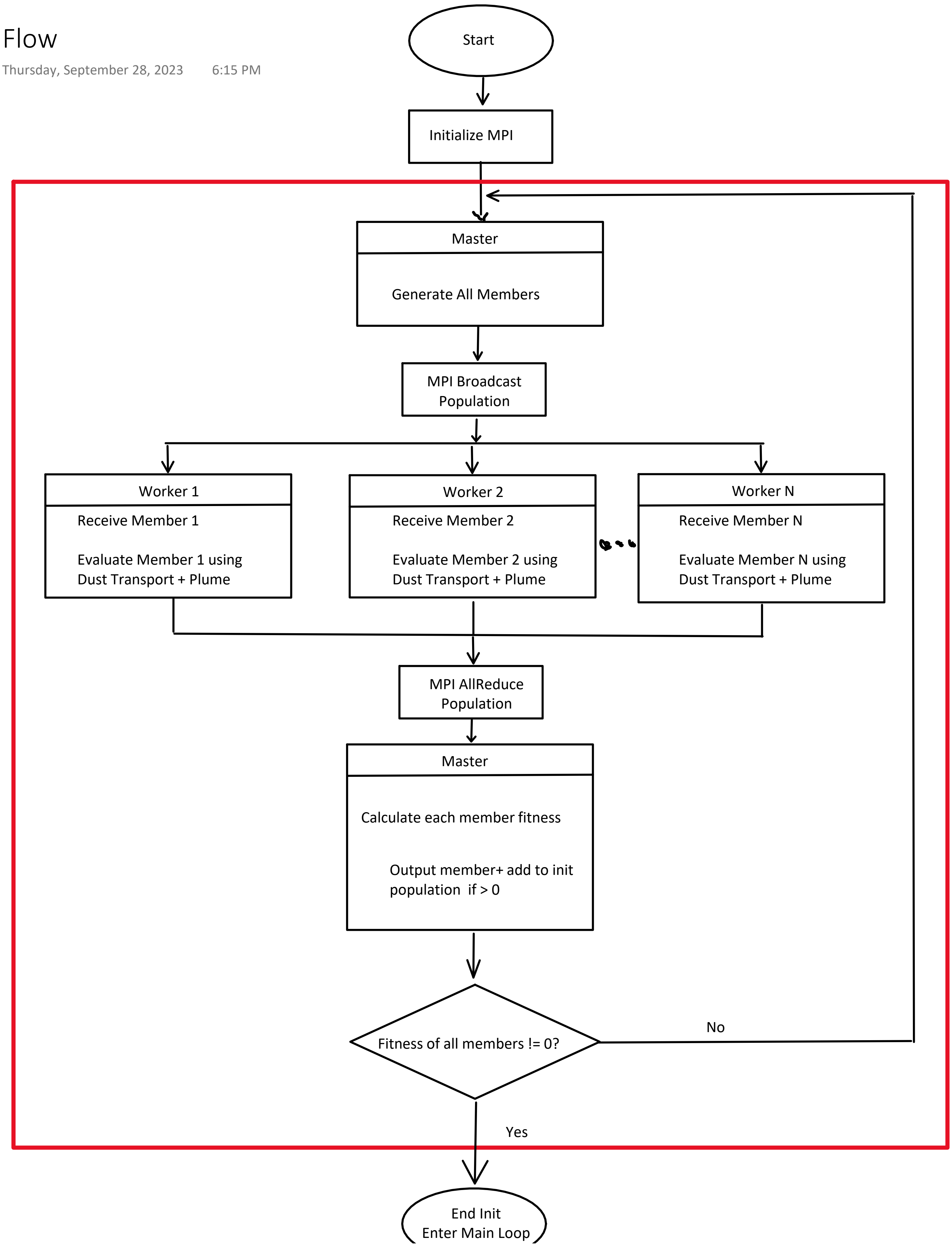
FIGURE 5.35 Schematic of an asynchronous concurrent genetic algorithm.

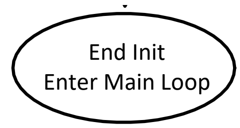
Not this…



Flow

Thursday, September 28, 2023 6:15 PM




End Init
Enter Main Loop