

THESIS PREPARATION

IT UNIVERSITY OF COPENHAGEN, AUTUMN 2018

RICHARD BÁNYI

1 INTRODUCTION

Intelligent agents are defined as entities that carry out some set of operations on behalf of a user or a program with some degree of independence or autonomy, and in so doing, employ some knowledge or representation of the user's goals or desires.¹ The field of artificial intelligence has been trying for decades to create machines that display human-level intelligence.

2 RELATED WORK

Donec euismod iaculis pretium. Donec non massa elit. Phasellus sagittis magna et maximus dictum. Duis quis ullamcorper orci. Mauris interdum, elit eu tincidunt tempor, lectus mi venenatis purus, quis posuere tellus ex in magna. Phasellus tincidunt nibh eu tortor semper, et varius justo vulputate. Nullam dictum congue lacinia. Maecenas sagittis nulla quis leo fringilla viverra. Proin eget egestas nisl. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum a interdum tellus, a hendrerit ligula. Duis ut risus ut lacus maximus euismod. Integer quis justo sit amet sapien accumsan rutrum nec nec dolor. Aliquam laoreet scelerisque ante, quis hendrerit ipsum viverra tempor.

3 ALGORITHMS

3.1 Evolutionary Algorithms

Evolutionary algorithms focus on global optimization problems inspired by biological evolution. EA are population based, meta heuristic search procedures that incorporate genetic operators. Algorithm maintains a population of candidate solutions which is subjected to natural selection and mutation². In each generation, a set of offspring is generated by applying bio operators such as *mutation*, *crossover*, *selection*. Each generation, the fitness of every individual in the population is evaluated. More fit individuals are stochastically selected from the current population, and each individual's *genome* is modified (recombined or randomly mutated) to form a new generation. The algorithm terminates when either the maximum number of generations has been produced, or fitness level has been reached for the population.

3.2 Non-Dominated Sorting Genetic Algorithm II

Multi-objective optimization is concerned with optimization problems which involves several objective functions to be optimized simultaneously. For example an objective for a problem may require to minimize cost while maximizing profits, thus these two contradicting objectives can't be optimized simultaneously. Maximizing one objective leads to weakening the other objective. Therefore there is no single feasible solution, but instead a set of *Pareto* optimal solutions. A

¹https://en.wikipedia.org/wiki/Autonomous_agent

²https://en.wikipedia.org/wiki/Evolutionary_algorithm

solution is called *nondominated*, if none of the objective functions can be improved in value without degrading some of the other objective values ³.

Example of pareto front can be seen in figure 1. The dark blue circles represents the set of *pareto optimal solutions*, *nondominated* solutions, that are not dominated by another feasible solution. Circle C is not on the Pareto frontier because it is dominated by both A and B.

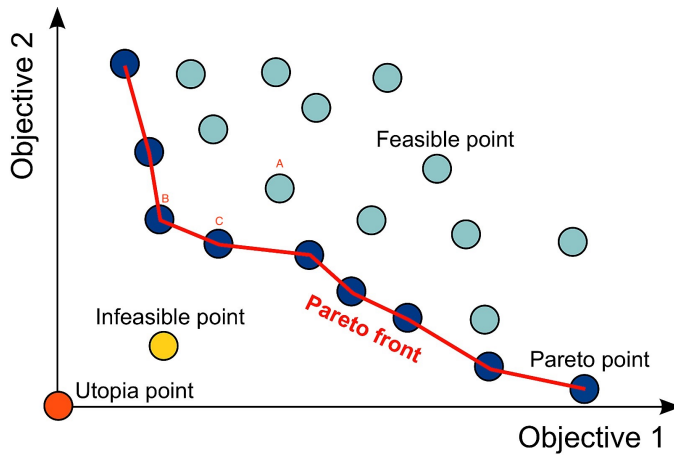


Fig. 1. Pareto Front

Evolutionary algorithm such as the *NSGA-II*⁴ is a standart approach in solving multi-objective optimization problem. The algorithm is similiar to a classical evolutionary algorithm. However the selection mechanism is different. The algorithm applies a pareto-based ranking scheme. This means that a rank is assigned to each individual based on nondominance - individuals that are not dominated by another get highest rank. Similiarly as in EA, parents produce a new offspring population using genetic operators (i.e. crossover and mutation).

³https://en.wikipedia.org/wiki/Multi-objective_optimization

⁴<https://ieeexplore.ieee.org/document/996017>

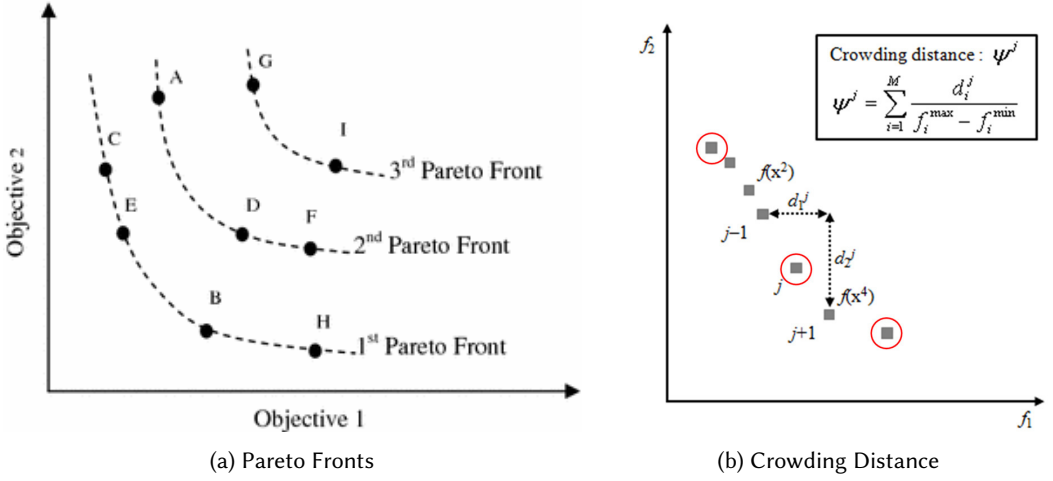


Fig. 2. Pareto Fronts and Crowding Distance.

Parents and offsprings are combined together and are sorted into different fronts, determined by their ranking. Then a secondary sorting strategy is applied by calculating the crowding distance for each individual within their fronts. The best individuals are either in the lower fronts (assuming that the objective values are to be minimized) or in the same front with higher crowding distance. The crowding distance is calculated as the sum of the cubic distance of the objective values between the neighbouring solutions of the individual. This ensures to maintain diversity and spread of solutions - crowded solutions areas are less preferred than sparsely crowded solutions in the solution area. Next generation is created by copying the best solutions (*elitism*) or the first N individuals from the population of parents and offsprings. A more detailed flow diagram of the sorting procedure is depicted in the following figure 3.

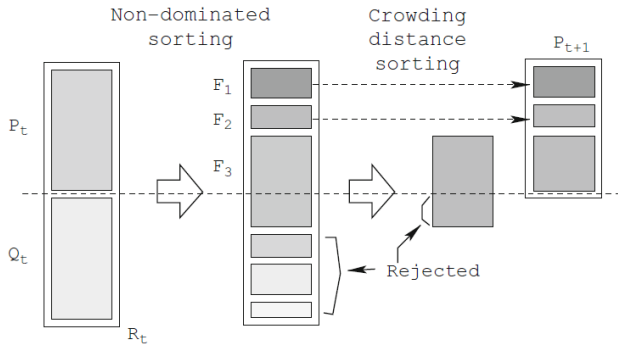


Fig. 3. Sorting procedure in NSGA-II

The implementation of the NSGA-II algorithm is provided by a DEAP⁵ evolutionary computation framework.

⁵<https://deap.readthedocs.io/en/master/>

3.3 NEAT

Donec euismod iaculis pretium. Donec non massa elit. Phasellus sagittis magna et maximus dictum. Duis quis ullamcorper orci. Mauris interdum, elit eu tincidunt tempor, lectus mi venenatis purus, quis posuere tellus ex in magna. Phasellus tincidunt nibh eu tortor semper, et varius justo vulputate. Nullam dictum congue lacinia. Maecenas sagittis nulla quis leo fringilla viverra. Proin eget egestas nisl. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum a interdum tellus, a hendrerit ligula. Duis ut risus ut lacus maximus euismod. Integer quis justo sit amet sapien accumsan rutrum nec nec dolor. Aliquam laoreet scelerisque ante, quis hendrerit ipsum viverra tempor.

3.4 HyperNEAT

Donec euismod iaculis pretium. Donec non massa elit. Phasellus sagittis magna et maximus dictum. Duis quis ullamcorper orci. Mauris interdum, elit eu tincidunt tempor, lectus mi venenatis purus, quis posuere tellus ex in magna. Phasellus tincidunt nibh eu tortor semper, et varius justo vulputate. Nullam dictum congue lacinia. Maecenas sagittis nulla quis leo fringilla viverra. Proin eget egestas nisl. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum a interdum tellus, a hendrerit ligula. Duis ut risus ut lacus maximus euismod. Integer quis justo sit amet sapien accumsan rutrum nec nec dolor. Aliquam laoreet scelerisque ante, quis hendrerit ipsum viverra tempor.

3.5 Multi-Spatial Substrates

Donec euismod iaculis pretium. Donec non massa elit. Phasellus sagittis magna et maximus dictum. Duis quis ullamcorper orci. Mauris interdum, elit eu tincidunt tempor, lectus mi venenatis purus, quis posuere tellus ex in magna. Phasellus tincidunt nibh eu tortor semper, et varius justo vulputate. Nullam dictum congue lacinia. Maecenas sagittis nulla quis leo fringilla viverra. Proin eget egestas nisl. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vestibulum a interdum tellus, a hendrerit ligula. Duis ut risus ut lacus maximus euismod. Integer quis justo sit amet sapien accumsan rutrum nec nec dolor. Aliquam laoreet scelerisque ante, quis hendrerit ipsum viverra tempor.