An Analysis of MultiObjective Evolutionary Techniques Applicable to Weather Routing

Joanna Szlapczynska[1] and Roman Smierzchalski[2]

[1] Gdynia Maritime University, Gdynia, Poland, e-mail: asiasz@am.gdynia.pl
[2] Gdynia Maritime University, Gdynia, Poland, e-mail: roms@am.gdynia.pl

Abstract. This document describes possible application of MultiObjective Evolutionary Algorithms (MOEAs) to a problem of finding the most suitable vessel route taking into account changeable weather conditions and several navigational constraints. The paper includes a review of MOEA-specific techniques and presents a proposal of their utilization in a weather routing evolutionary algorithm.

1 Introduction

A problem of finding the most suitable vessel route taking into account changeable weather conditions and navigational constraints e.g. landmasses is referred to as a weather routing problem. Such a problem is mostly considered for ocean-going ships where adverse weather conditions may impact both, often contradictory, economic and security aspects of voyage. Most of recent scientific researches in weather routing focus on shortening the passage time or minimization of fuel consumption only. However, modern MultiObjective Evolutionary Algorithms offer a method of dealing with such contradictory goals utilizing a well-known concept of Pareto-optimal sets.

One of the first weather routing approaches was a minimum time route planning based on weather forecasted data. Proposed by R.W. James in 1957 [3] an isochrone method, where recursively defined time-fronts are geometrically determined, was in wide use through decades. In late seventies based on the original isochrone method the first computer-aided weather routing tools were developed. Numerous improvements to the method were proposed since early eighties, with [2], [8] among others. Nonetheless, even the improved method has been displaced with time by genetic algorithms. Evolutionary approach as a natural successor of genetic approach has become popular in the last two decades and has been successfully applied to anti-collision manoeuvre modelling [6]. Modern weather routing tools also utilize evolutionary algorithms instead of the deprecated isochrone time-fronts. However, due to multiobjective nature of weather routing it is recommended to introduce some state-of-the-art multiobjective methods to the process of route finding.

This paper is organized as follows: section 2 presents a review of techniques utilized by MultiObjective Evolutionary Algorithms (MOEA) along with the basic concept of Pareto optimal sets. Section 3 describes a framework of weather routing evolutionary algorithm with possible MOEA extensions. Finally, section 4 summarizes the material presented.

2 MOEA Review

MultiObjective Evolutionary Algorithms (MOEA) have been growing in popularity since its inception in mid-1980s. In general, MOEAs extend the functionality of regular single-objective evolutionary algorithms providing a method of dealing with multiple and often conflicting criteria. However, one should be aware that “MOEA” term refers to some algorithmic framework rather than a specific ready-to-use solution or algorithm. Thus, already known MOEA techniques together with suitable design recommendations should be applied prior to building a problem-oriented multiobjective evolutionary application.

2.1 MOEA Classification

A decision maker is often involved with the final decision when a multiobjective problem is considered. It is simply because a single output solution should be chosen from a set of uniformly valid optimal solutions. From the decision maker’s point of view if improving in one objective results in detriment to the other, some compromise between the objectives is required. In result, the final solution of the multiobjective optimization problem consists of optimization and decision processes which interfere or follow each other. MOEAs are classified based on the order of optimization and decision processes. According to [7] there are three distinctive MOEA groups recognized, namely:

- “a priori” preference, where the decision maker combines all the objectives into a single scalar function;
- progressive preference, where the decision making and optimization processes alternate;
- “a posteriori” preference, where the found set of Pareto optimal solutions is presented to the decision maker who selects the final solution from the set provided.

Since the early 1990s, Pareto-based solutions have been the most extensively researched among the other MOEAs. Following the MOEA citation research presented in [7] it appears that the “a posteriori” approaches are the most popular nowadays. Thus, this paper focuses on the issues related to Pareto-based MOEA only.

2.2 Pareto Optimality Concept

In general, finding a solution in a multiobjective optimization problem is a complex task based on finding a trade-off between often incommensurable and competing objectives. The trade-off denotes that further improvement in one objective deteriorates the other. Thus, it is unlikely that a solution to a multi-objective problem would be a single optimal one, it is rather a set of equally optimal solutions.

An underlying definition for Pareto-optimal sets is the notion of Pareto-dominance. A particular solution x is said to dominate a solution y if the former performs better than the latter for at least one objective and performs no worse than the latter in all the other objectives.

Mathematically, the concept of Pareto-optimality is defined as follows. Let us consider a multi-objective optimization problem with m decision variables and n objectives. Without loss of generality the optimization problem can be expressed as given by equation 1:

$$u = F(x) = \{f_1(x), f_2(x), f_3(x), \ldots, f_n(x)\} \rightarrow min \qquad (1)$$

where $x = \{x_1, x_2, \ldots, x_m\}$ is a vector of decision variables and $u = \{u_1, u_2, \ldots, u_n\}$ is a performance vector associated with an x vector. A particular solution x with associated performance vector u is said to dominate another solution y with performance vector v ($x \prec y$) if the performance vectors u and v meet the equations 2-3:

$$x \prec y \Leftrightarrow u \prec v \qquad (2)$$

$$u \prec v \text{ iff } [\forall \mathrm{i} \in \{1,\ldots,n\}, u_i \leq v_i] \wedge [\exists i \in \{1,\ldots,n\} : u_i < v_i] \qquad (3)$$

A solution $x \in \Omega$ is Pareto optimal with respect to Ω if and only if there is no such $x' \in \Omega$ for which $v \prec u$ when (equations 4-5):

$$u = F(x) = \{f_1(x), f_2(x), f_3(x), \ldots, f_n(x)\} \qquad (4)$$

$$v = F(x') = \{f_1(x'), f_2(x'), f_3(x'), \ldots, f_n(x')\} \qquad (5)$$

It must be stressed out, though, that the Pareto optimality is always considered with respect to the Ω set which is assumed to be equal to the entire decision variable space unless otherwise specified. A Pareto-front PF^* is a set of points in the problem's criterion space corresponding to the Pareto-optimal set P^*. Formal definitions of P^* (equation 6) and PF^* (equation 7) are provided by the following formulas:

$$P^* := \left\{ x \in \Omega \,\middle|\, \neg \exists x' \in \Omega : F(x') \prec F(x) \right\} \qquad (6)$$

$$PF^* := \left\{ u = F(x) = \{f_1(x), \ldots, f_n(x)\} \,\middle|\, x \in P^* \right\} \qquad (7)$$

In MOEA literature P^* and PF^* sets are sometimes referred to as P_{true} and PF_{true} respectively. Further in the paper, if a solution x covers y it refers to the situation that x dominates y or these solutions are equal.

2.3 Basic MOEA Techniques

As mentioned earlier, MOEA should be considered rather as an algorithmic framework than as an explicit algorithm or solution. In result, prior to MOEA implementation, it is necessary to gather information about basic MOEA techniques, their advantages, disadvantages and utilization possibilities. The core set of basic MOEA techniques according to [5], [7] and [9] is described in the following subsections. A few of the techniques are already known from single-objective evolutionary applications.

Secondary population. Numerous MOEA publications ([5], [7], [9] among others) stress the importance of introduction a secondary population into MOEA design. Such a population is an additional population maintained throughout MOEA execution time, collecting all Pareto optimal solutions found so far during the search process. Its main goal is to preserve all desirable solutions throughout the generation process. In accordance with Pareto notation introduced in subsection 2.2, the secondary population is termed $P_{known}(t)$, where t denotes current generation number. Similarly, a current set of Pareto optimal solutions determined at the end of each generation with respect to the current MOEA generational population is termed $P_{current}(t)$. It is assumed, though, that $P_{known}(0)$ is an empty set and P_{known} without *t* annotation stands for the final set of Pareto optimal solutions collected before MOEA termination. Several strategies of second population storage exist. The most obvious and commonly used is the strategy of adding $P_{current}(t)$ to $P_{known}(t)$ at the end of each generation *t* (equation 8):

$$P_{known}(t) = P_{current}(t) \cup P_{known}(t-1) \tag{8}$$

The set of $P_{known}(t)$ must be periodically checked against obsolete Pareto solutions as Pareto optimality should always be evaluated within current Ω set. The simplest policy does not assume explicit copying $P_{known}(t)$ solutions back into the next population. However, other strategies exist where the secondary population participates in a tournament selecting next generations or is directly inserted into the next mating population. As suggested in [7], no secondary population strategy can be considered as the most useful or the most profitable for any particular purpose. Further discussion on the secondary population strategies can be found in [7].

Multiobjective ranking. Multiobjective evolutionary approach enforces that some transformation of the performance vector into a scalar fitness value is necessary. This transformation is achieved by means of a multiobjective ranking, often also referred to as Pareto ranking. In general, there are four basic ranking methods. All these methods are based on an assumption that preferred Pareto optimal solutions are ranked the same value whereas other solutions are assigned some less desirable rank value.

A ranking technique proposed by Goldberg [1] assumes that the population at each generation is checked for nondominated solutions. All the solutions are given rank 0 and removed from the population. Then again nondominated solutions are found in the shrunken population, given rank 1 and removed from the population. This process continues with increasing rank value at each step until all solutions have been ranked. In result a series of nondominated fronts is created.

A modification of the previous proposal was suggested by Fonseca and Fleming [5]. A rank assigned to the solution is based in this approach on the number of individuals by which the solution is dominated. As a result, all solutions with nondominated performance vectors receive rank 0.

A simplified ranking scheme was proposed by Van Veldhuizen [7]. Here a rank value equals to either 0 or 1 depending on whether the solution is nondominated or not. Thus there is no further differentiation between dominated solutions.

Another ranking policy was proposed by Zitzler and Thiele [9]. The twofold ranking policy incorporates information from the secondary population $P_{known}(t)$. During the first step each solution $i \in P_{known}(t)$ is given a real value rank $s_i \in [0;1)$. The s_i number (equation 9), called strength, is proportional to the number of population members $j \in P(t)$ for which $i \prec j$. Let *n* denote the number of individuals in t-th population *P(t)* covered by *i* and *N* denote the size of *P(t)*, then:

$$s_i = \frac{n}{N+1} \tag{9}$$

Following procedure step assigns a ranking value *rank*$_j$ to each solution in the t-th population $j \in P(t)$ calculated by summing the strengths of all solutions from the secondary population $i \in P_{known}(t)$ that cover j. The final rank is increased by 1 to guarantee that solutions in the secondary population have better rank than those in *P(t)* (equation 10):

$$rank_j = 1 + \sum_{i,\ i \prec j} s_i \tag{10}$$

All presented ranking strategies differ in terms of computational complexity, which impacts MOEA's total execution time. Detailed discussion on the best and worse case complexity can be found in [7]. Zitzler & Thiele approach, according to [7], involve some more overhead mostly due to utilization of the secondary population and additional comparisons required. Nonetheless, the last scheme is the only one that actively takes advantage of the secondary population technique.

Niching and fitness sharing. The term *niching* refers to the process of clustering in either solution-space or criterion-space. In this process clusters consist of groups formed by some individuals selected from the entire population. Niching is primarily aimed at finding and maintaining multiple optima. In result, this technique should assure a good spread of discovered solutions and prevent MOEA algorithm from being swamped by solutions with identical fitness. *Fitness sharing* is the most popular realization of the niching technique. It is based on an assumption that individuals in a particular niche share available resources. Thus, the more individuals are located in the vicinity of a certain individual, the more its fitness value is deteriorated. The vicinity is most often determined by a distance measure d(i,j) and specified by niche radius σ_{share}. The distance function d(i,j) operates either in solution-space (the parameters manipulated by the evolutionary algorithm) or criterion-space (the results corresponding to the chosen parameters), resulting in appropriate type of fitness sharing.
The basic approach (known from the single-objective EA) to fitness sharing was presented by Goldberg in [1]. Most of its implementations follow the assumption that fitness sharing is applied only to equally ranked individuals. A niche count, factor that reduces individual's fitness, is a sum of all values found by pairwise comparisons of the individual with all the other ones. Equation 11 is used to calculate a single niche count element. Required niche size is calculated by equation 12. A detailed discussion on various fitness sharing approaches is provided in [7].

$$sh(d) = \begin{cases} 1 - \left(\dfrac{d}{\sigma_{share}} \right)^{\alpha} & d < \sigma_{share} \\ 0 & otherwise \end{cases} \tag{11}$$

$$\sigma_{share} = \frac{\sqrt{\sum_{k=1}^{p} \left(x_{k,\max} - x_{k,\min} \right)^2}}{2\sqrt[p]{q}} \tag{12}$$

where:

sh(d) – single contribution to the individual's share count,
d – distance over some norm (usually Euclidean distance),

σ_{share} – niche size,
α – shaping parameter,
p – number of decision variables within the solution x,
$x_{k,max/min}$ – maximum/minimum value of k-th decision variable,
q – required number of niches.

Mating restrictions. The idea behind restricted mating is to prevent or minimize offspring, so-called *lethals*, created by recombination of chromosomes from different niches. Such individuals can lead to degradation of MOEA performance. To remedy the problem some restrictions to mating might be introduced providing a distance metric and a maximum distance value σ_{mate} for which mating is still permitted. The most popular solution for mating restriction is, according [5] and [7], introduction of the fitness sharing niche radius σ_{share} into the problem and setting $\sigma_{mate}=\sigma_{share}$. However, it is questioned ([7], [9]) whether such restriction policy is indeed a compulsory MOEA component, especially when there is no quantitative evidence of its benefits [7].

3 A proposal of Weather Routing Evolutionary Algorithm with MOEA Extensions

Criteria set. The weather routing criteria can be divided into two separate subsets, namely:

- economic criteria subset;
- safety criteria subset.

Primary goal of the former is to assure that total costs of the voyage remains as low as possible. That is both passage time as well as fuel consumption should be minimized. Safety criteria are represented by vessel traffic intensity and degree of constraint violation. The higher traffic intensity the higher collision risk, thus this criterion is also to be minimized. Likewise, violation of safety constraint should be minimized, however some violations i.e. land crossings are unacceptable for a route. Unfortunately safety issues might be in contrary to the economic goals, especially when the shortest route crosses a high vessel traffic intensity area or land. Thus an equilibrium between economic and safety criteria is to be found resulting in possibly inexpensive and safe route.

Having analyzed these facts a goal function for the evolutionary algorithm might be proposed as follows (equations 13-17):

$$f\ (t_r, v_{fc}, i_{traff}, r) = \{f_{economy}(t_r, v_{fc}), f_{safety}(i_{traff}, r)\} \longrightarrow min \tag{13}$$

$$f_{economy}(t_r, v_{fc}) = \{f_{passage\_time}(t_r), f_{fuel\_consumption}(v_{fc})\} \tag{14}$$

$$f_{safety}(i_{traff}, r) = \{f_{traffic\_intensity}(i_{traff}), f_{constraint\_violation}(r)\} \tag{15}$$

$$f_{passage\_time}(t_r) = t_r; f_{fuel\_consumption}(v_{fc}) = v_{fc}; f_{traffic\_intensity}(i_{traff}) = i_{traff} \tag{16}$$

$$f_{constraint\_violation}(r) = \sum_i p_i r_i \tag{17}$$

where

t_r – passage time for the route [h],

v_{fc} – total fuel consumption for the route [t],
i_{traff} – vessel traffic intensity (dimensionless),
p_i – penalty imposed when the i-th constraint is not met (dimensionless),
r_i – boolean value depicting whether the i-th constraint is not met (dimensionless).

Constraints. All limitations to the problem domain in weather routing are purely navigational. Landmasses which cannot be crossed constitute the prime constraint. Even a small violation of the constraint results in a route unacceptable from navigational standpoint. Along with an assumption that land shore does not change its shape during a route execution this constraint is assumed static. However, other navigational constraints exist that do not fall into category of static ones, namely ice phenomenon and tropical cyclones. Available information about ice and cyclones is mostly derived from forecasted, that is probablistic, data. Moreover, both ice concentrations as well as a centre of a tropical depression change with time. Thus these constraints are assumed fuzzy dynamic.

Chromosome structure. Values to be sought in weather routing are:

- a set of waypoints given by their geographical coordinates;
- velocity of the ship between any two consecutive waypoints, assumed constant on a sector between two waypoints.

In order to suitably represent given vales a chromosome's structure is proposed as follows. A chromosome is an ordered set of threesome values (x_i; y_i; v_i) where (x_i; y_i) represent geographical coordinates (longitude, latitude) of i-th waypoint and v_i represents ship's velocity between (i-1)-th and i-th waypoint. The first element in a chromosome has its velocity value undefined.

Initial population. Having determined all the basic routes, namely the orthodrome and isochrone route, it is possible to build the initial population for the weather routing evolutionary process. The population should consist of avg. 50 individuals, each being a random mutation of the basic routes. Also pure orthodrome and isochrone route should belong to the initial population. In addition to that, it is worth considering whether some other routes optimizing one of the other criteria (fuel consumption, vessel traffic intensity, degree of constraints violation) should also be included in the initial population.

Specialized operators. There are several specialized "genetic" operator required, customized to the established chromosome structure. According to the description of route finding problems provided by Michalewicz ([4]), following operators will be implemented in the weather routing evolutionary algorithm:

- crossing;
- mutation;
- insertion;
- deletion;
- smoothing;
- exchange of sub-routes.

Stop condition. In the end of each generation an increase of fitness function will be determined. Whenever the increase will be satisfactorily small (smaller than some ε value), the evolutionary process will be terminated.

Additional applicable MOEA techniques. According to the review given in section 2, following MOEA techniques will be applied to the weather routing evolutionary algorithm:

- secondary population with its basic strategy $P_{known}(t) = P_{current}(t) \bigcup P_{known}(t-1)$ **;**
- multiobjective ranking by Zizler and Thiele, actively utilizing data from the secondary population;
- criterion-space fitness sharing with Goldberg's niche count.

Mating restrictions is of arguable benefit to MOEA implementation [7], thus no such restrictions are planned to be introduced to the proposed algorithm.

4 Summary

This paper presents a review of available techniques taken from MultiObjective Evolutionary Algorithms (MOEAs). The MOEA techniques together with basic evolutionary approach are then applied to the problem of vessel route finding for changeable weather conditions, referred to as weather routing. Evolutionary algorithm for weather routing is presented along with selected MOEA techniques. Description provided refers to a proposal of the evolutionary algorithm for weather routing, not implemented yet. Yet conclusions and comments to the assumptions given should be presented when the algorithm is finally implemented.

Bibliography

[1] Goldberg, D.E. *Genetic Algorithms in Search, Optimization and Machine Learning,* Addision Wesley, Reading, 1987.

[2] Hagiwara, H. *Weather routing of (sail-assisted) motor vessels,* PhD thesis. Delft, 1989.

[3] James, R.W. *Application of wave forecast to marine navigation*, US Navy Hydrographic Office, Washington, 1957.

[4] Michalewicz, Z. *Genetic Algorithms + Data Structure = Evolution Programs*, Springer-Verlag Berlin Heidelberg, 1996.

[5] Purshouse, R.C., Fleming, P.J. *The MultiObjective Genetic Algorithm applied to benchmark problems – an analysis*, Research Report No. 796, University of Sheffield, 2001.

[6] Smierzchalski, R., Michalewicz, Z. Modeling of a Ship Trajectory in Collision Situations at Sea by Evolutionary Algorithm*, IEEE Transactions on Evolutionary Computation* No. 3 Vol. 4, pp. 227-241, 2000.

[7] Van Veldhuizen, D.A., Lamont, G.B. Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art, *Evolutionary Computation*, pp.125-147, Massachusets Institute of Technology, 2000.

[8] Wisniewski, B. *Methods of route selection for a sea going vessel (in Polish)* , Wydawnictwo Morskie, Gdańsk, 1991.

[9] Zitzler, E., Thiele, L. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach, *IEEE Transactions on Evolutionary Computation* No. 3 Vol. 4, pp. 257-271, 1999.