Ant Colony Optimization in Multiobjective Portfolio Selection

Karl Doerner[*] *Walter J. Gutjahr*[†] *Richard F. Hartl*[*] *Christine Strauss*[*] *Christian Stummer*[*]

[*] Department of Management Science, University of Vienna
Bruenner Str. 72, 1210 Vienna, Austria
Email: {karl.doerner, richard.hartl, christine.strauss, christian.stummer}@univie.ac.at

[†] Department of Statistics and Decision Support Systems, University of Vienna
Universitaetsstr. 5/3, 1010 Vienna, Austria
Email: walter.gutjahr@univie.ac.at

1 Introduction

Multiobjective decision-making and combinatorial optimization have been studied extensively over the past few decades (cf. [16], and [4] for bibliographies). Both fields play a decisive role in multiobjective combinatorial optimization, for which the class of (multiobjective) portfolio selection is of particularly high practical relevance (cf. [10] for a survey). Research and development (R&D) management provides an especially useful example: when large amounts of resources (see [14]) and, more importantly, a product's long-term commercial success are at stake it is crucial for a firm to determine the "best" subset of R&D projects out of dozens of proposals.

Support for making multiple objectives decisions is based on either (i) approaches aggregating different types of benefits (e.g., cash flow, sales or even intangibles such as image) in a unique overall objective function or (ii) approaches that (partially) determine the efficient (i. e., Pareto-optimal) portfolio candidates and then allow the decision-maker to interactively "move" in that solution space until a satisfactory alternative is found. While the former techniques often require extensive a priori preference information (e.g., weights, thresholds, marginal benefits or guidelines for benefit or resource substitution between different categories), those of the latter school regularly can do without such data. However, the process involved in identifying efficient portfolios is not trivial. While a brute-force complete enumeration procedure can determine them within acceptable time for comparatively small problems only, that task becomes increasingly demanding as the number of projects grows. In such complex problems, heuristic-based approaches can provide an attractive tradeoff between the quality of the approximation of a solution space and the computing capacity required to achieve this approximation. Accordingly, genetic algorithms, simulated annealing and tabu search have already been implemented for multiobjective combinatorial optimization problems during the last decade.

Our approach applies a constructive meta-heuristic, Ant Colony Optimization (ACO), and, thus, gives another evidence for its versatility. It is based on the Ant System – first introduced by [6] and [7] – that imitates the behavior shown by real ants when searching for food. Ants communicate information about food sources via the quantity of an aromatic essence called pheromone, which the ants secrete as they move along. Over time, the short direct paths leading from the nest to a food source are more frequented than longer paths. As a result, the direct paths are marked with more pheromone, which in turn attracts an ever increasing number of ants to follow these shorter routes and make the corresponding pheromone trails grow faster. Artificial ants not only imitate the behavior described, but also apply additional, problem-specific heuristic information. The Ant System has been applied to and provided solutions for various hard combinatorial optimization problems, e.g., [1], [2], [5], [8], [11], and [18], and a convergence proof for a generalized Ant System Algorithm has been established (cf. [12]). In order to meet multiobjective problem specific requirements the adapted ACO approach implements several pheromone matrices and random weights for their use. The lifespan concept and the pheromone decoding scheme are two more novel features which are necessary to model the portfolio selection process.

2 Problem Description

Our approach aims to determine efficient project portfolios. A portfolio is a subset of the set of N proposed projects; a project is characterized by its resource consumption and the benefits it provides. A portfolio is efficient (i. e., non-dominated or Pareto-optimal) in the sense that no other feasible project portfolio exists which promises higher benefits in at least one of the objectives and offers at least the same benefits in all objectives. In a subsequent phase of the decision-making process, these efficient portfolios serve as a basis for systems that allow decision-makers to interactively find their individual favorites (see [13] and [17] for examples).

Some interesting data might be neglected if benefits from different planning periods are aggregated to an overall net present value (see [15] for a discussion). For this reason, our approach considers the portfolios' benefits (e. g., cash flow or sales) separately for each period (e. g., one business year) and, thus, is characterized by a comparatively high number of objectives (e. g., three objective categories and five periods result in overall fifteen objectives). These objectives are evaluated by the sum of the benefit contribution of each project contained in the portfolio, with adjustments made for interrelation effects between the individual projects.

The objectives addressed above are subject to two types of constraints. The first group relates to limited resources (e. g., funds or manpower) or minimum benefit requirements both of which are evaluated per period, as well as by the sum of their net present values. The second group ensures that at most a maximum (at least a minimum, respectively) number of projects from the given subsets of projects exists for each feasible portfolio. For examples of the practical relevance of these constraints, it may be desirable to select a minimum number of projects that attract certain target groups or, from an R&D point of view, deal with emerging technologies. In addition, social resentments (which could affect a firm's image, for instance) or political changes impacting business conditions (e. g., taxes on raw materials) might also be considered. Finally, certain basic aspects of balancing a portfolio (e. g., ensuring that a minimum number of projects from each entity of the firm is included in any portfolio) can be handled by these constraints as well.

3 Solution Procedures

Two solution procedures are introduced in the following section. The first procedure described is a heuristic approach that relies on a random proportion rule, while Ant Colony Optimization for multiobjective portfolio selection is presented afterwards.

3.1 Heuristic Solution Procedure

The heuristic solution procedure consists of two consecutive phases: the initial construction phase involves the generation of a number of feasible portfolios, while the subsequent evaluation phase determines the potentially efficient portfolios among them.

In the construction phase, a random-proportional rule incorporates a measure that evaluates how well the project candidates fit into a partially constructed portfolio. An aggregated value of attractiveness $\eta_i(\Psi)$ is computed for each portfolio candidate i depending on the (partial) portfolio Ψ. This value is based on constraints and targets that can be categorized into four categories: maximum or minimum restrictions (e. g., upper/lower limit for number of projects of a certain project type in the portfolio), resource restrictions (e. g., maximum available workforce) and benefit restrictions (e. g., minimum profit expectations). If a maximum restriction or a resource restriction is violated, then the attractiveness value is set to zero. If the maximum restriction and the resource restriction is fulfilled, then the attractiveness value corresponds to the degree of fulfillment in the two remaining categories (i. e., minimum restrictions and/or benefit restrictions). A special case occurs when all restrictions are satisfied by including the considered project in the portfolio; in this case, the attractiveness value is set to one. After feasible portfolios are constructed, the potentially efficient ones (again, on the basis of Pareto-optimality) are determined by comparing them in pairs in the succeeding evaluation phase.

3.2 Ant Colony Optimization

In this section, we address the implementation of the ACO algorithm for the problem at hand (cf. [9]). The proposed ant system can be described by the algorithm given in Table 1.

```
procedure ACO (...) {
  Initialization of the ACO;
  for v:=1 to max_iterations {
    for Ant := 1 to Γ {
      determine the life span Ξ of the ant randomly on the interval [1..|N|];
      // The (maximum) number of projects an ant selects equals its life span Ξ.
      set Ψ={};
      determine the objective weight χ_k for each objective k randomly;
      ξ = Ξ;
      while ξ > 0 and ∃  η_i(Ψ) > 0 {
        select a project using formula (1) and add it to Ψ;
        update local pheromone information;
        decrement ξ;
      }
      check feasibility of portfolio Ψ;
      if portfolio Ψ is feasible {
        check potential efficiency of portfolio Ψ;
        if portfolio Ψ is potentially efficient {
          store portfolio Ψ;
    } } }
    for each objective k {
      determine best solution Ψ_k;
      update global pheromone information using Ψ_k;
} } }
```

Table 1: The ACO Procedure

In the initialization phase, Γ ants are generated each starting with an empty portfolio $\Psi = \{\}$. The life span Ξ and the objective weights χ are determined randomly for each ant. In the construction phase of the algorithm, each ant tries to construct a feasible portfolio Ψ by using a pseudo-random-proportional rule. After a portfolio has been constructed, its feasibility and efficiency is determined. Global pheromone update is performed by using the best solution Ψ_k of the current iteration for each objective k.

For each objective k the pheromone information is stored in a matrix τ, with the number of rows and columns corresponding to the projects. The value τ_{ij}^k represents the current pheromone information, i.e., the pheromone information with respect to objective k of including project i together with project j in the same portfolio. Let K denote the number of objectives, and N denote the set of project candidates.

Given the visibility, the pheromone information and the set of all feasible projects $\Omega = \{i \in N : \exists\, \eta_i(\Psi) > 0\}$, a feasible project i is selected for addition to the current portfolio Ψ according to a pseudo-random-proportional rule that can be stated as follows:

$$i = \begin{cases} arg\ max\{[(\sum_{k=1}^{K} \chi_k \cdot (\sum_{j \in \Psi} \tau_{ij}^k))]^\alpha \cdot [\eta_i(\Psi)]^\beta\} & \text{if } q \leq q_0 \\ \\ P & otherwise, \end{cases} \tag{1}$$

where q is a random number uniformly distributed in $[0..1)$, q_0 is a parameter $(0 \leq q_0 \leq 1)$ to be set by the user. P is a random variable selected according to the probability distribution given:

$$\mathcal{P}_i(\Psi) = \begin{cases} \frac{[(\sum_{k=1}^{K} \chi_k \cdot (\sum_{j \in \Psi} \tau_{ij}^k))]^\alpha \cdot [\eta_i(\Psi)]^\beta}{\sum_{h \in \Omega}\{[(\sum_{k=1}^{K} \chi_k \cdot (\sum_{j \in \Psi} \tau_{hj}^k))]^\alpha \cdot [\eta_h(\Psi)]^\beta\}} & \text{if } i \in \Omega \\ \\ 0 & otherwise, \end{cases} \tag{2}$$

Local updating is performed once an artificial ant has added a project to a portfolio. When an ant selects a project i, the amount of pheromone on these elements τ_{ij}^k of the pheromone matrix is decreased if project j is a project of the existing (partial) portfolio.

The global pheromone information is updated once each ant of the population has constructed a solution and the feasibility and efficiency have been determined. We use a pheromone update procedure in which only the best solution provided by an iteration is used for global updating (cf. [8]).

4 Numerical Analysis

The numerical example used in this study outlines a rather complex decision situation that does not permit any "intuitive" favoring of certain project combinations in advance. Our example considers twenty projects ($N = 20$), five planning periods, and three benefit categories ($K = 5 \cdot 3 = 15$). The projects vary substantially in both their potential benefits and the resources they require. Moreover, some projects focus on particular planning periods and/or benefit categories, while other projects provide average values. In addition to limited resources and minimum benefit requirements, ten supplementary constraints ensure that - to provide examples for a maximum and a minimum restriction - any feasible portfolio includes at most one of three projects pursuing the same goal, or at least two projects that help to diversify business. Finally, four interrelations are used to model synergism or cannibalism between projects.

The parameter settings of ACO chosen for the computational experiments ($\alpha = 1$, $\beta = 1$, $\rho = 0.1$, $\rho^{local} = 0.1$, $\tau_0 = 0.01$, $\Gamma = 10$) were taken from other applications in which they have proven to be advantageous (e. g. [8]). The only exception is parameter q_0; it is reduced from $q_0 = 0.9$ to $q_0 = 0.5$ because higher diversification is desirable for our application.

The problem described above was solved by means of complete enumeration, Monte Carlo Simulation, the heuristic described in Section 3.1, and ACO using the parameter settings outlined above. Computational runs with 100, 200, 300, 400, and 500 iterations were performed in order to provide an insight into how the solution quality for all three approaches develops. Complete enumeration shows that this problem has 138 efficient portfolios.

For the Monte Carlo Simulation, the heuristic, and the ACO Table 2 contains two quality measures averaged over 10 runs: For the problems considered here the most important criteria for the quality of a solution generated by a heuristic approach is the number of efficient portfolios. For large real-world problems it is not always sure, that any efficient portfolio can be found, it is important then to

	100 Iterations	200 Iterations	300 Iterations	400 Iterations	500 Iterations
Monte Carlo	0.1 (1.8)	0.4 (4.6)	0.3 (5.7)	0.2 (7.4)	10.9 (7.5)
Heuristic	76.4 (3.7)	97.9 (3.2)	111.1 (2.1)	117.3 (1.4)	124.3 (0.5)
ACO	87.4 (2.6)	110.3 (1.5)	119.7 (0.8)	122.4 (0.8)	128.0 (0.5)

Table 2: Computational Results

evaluate if the distance between the efficient set and its approximation is small enough [3]; in Table 2 the first value indicates the averaged number of portfolios appearing in the efficient set that are proven through complete enumeration to be actually efficient. Furthermore, a good approach should not propose dominated portfolios - as a consequence, the difference between the proposed portfolios and actually efficient ones should be kept to a minimum; the value in brackets indicates how many additional (dominated) portfolios have been proposed falsely as efficient ones by the approach. The computational results show that after the execution of 200 iterations ACO generates 110.3 efficient portfolios of the overall existing 138 efficient portfolios, whereas the use of a heuristic by itself only finds 97.9 portfolios on the average. The search for efficient portfolios on the basis of Monte Carlo Simulation does not prove successful even for a limited example, providing only 0.4 of the overall existing efficient portfolios on the average. Another indicator for the quality of ACO is the small percentage of all possible portfolios that the algorithm generates, e. g. only 0.5 per cent of the total search space is checked to find 128.0 out of 138 efficient portfolios in the version with 500 iterations. The results can be interpreted as good indications that ACO will generate satisfying results for problems that are too large to be enumerated completely.

Acknowledgements

The authors thank three anonymous referees for valuable comments on a previous version of this paper.

References

[1] A. Bauer, B. Bullnheimer, R.F. Hartl, and C. Strauss. Minimizing total tardiness on a single machine using Ant Colony Optimization. *Central European Journal of Operations Research*, 8:125–141, 2000.

[2] B. Bullnheimer, R.F. Hartl, and C. Strauss. An improved Ant System algorithm for the vehicle routing problem. *Annals of Operations Research*, 89:319–328, 1999.

[3] P. Czyzak and A. Jaszkiewicz. Pareto Simulated Annealing: A metaheuristic technique for multiple-objective combinatorial optimization. *Journal of Multi-Criteria Decisions Analysis*, 7:34–47, 1998.

[4] M. Dell'Amico, F. Maffioli, and S. Martello, editors. *Annotated Bibliographies in Combinatorial Optimization*. Wiley, Chichester, 1997.

[5] K. Doerner, R.F. Hartl, and M. Reimann. Cooperative Ant Colonies for optimizing resource allocation in transportation. In Boers et al., editors, *Applications of Evolutionary Computing, LNCS 2037*, Springer, Berlin, pages 70–79, 2001.

[6] M. Dorigo. Optimization, Learning and Natural Algorithms. Doctoral Dissertation, Politecnico di Milano, 1992.

[7] M. Dorigo, V. Maniezzo, and A. Colorni. Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics*, 26:29–41, February 1996.

[8] M. Dorigo and L.M. Gambardella. Ant Colony System: A cooperative learning approach to the travelling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1:53–66, January 1997.

[9] M. Dorigo and G. Di Caro. The Ant Colony Optimization meta-heuristic. In Corne, Dorigo, and Glover, editors, *New Ideas in Optimization*, McGraw-Hill, London, pages 11–32, 1999.

[10] M. Ehrgott and X. Gandibleux. A survey and annotated bibliography of multiobjective combinatorial optimization. *OR Spektrum*, 22:425–460, October 2000.

[11] L.M. Gambardella, E. Taillard, and G. Agazzi. MACS-VRPTW: A multiple Ant Colony System for vehicle routing problems with time windows. In Corne, Dorigo and Glover, editors, *New Ideas in Optimization*, McGraw-Hill, London, pages 64–76, 1999.

[12] W.J. Gutjahr. A graph-based Ant System and its convergence. *Future Generation Computing Systems*, 16:873–888, June 2000.

[13] V. Lotfi, T.J. Stewart, and S. Zionts. An aspiration-level interactive model for multiple criteria decision making. *Computers & Operations Research*, 19:671–681, October 1992.

[14] Organization for Economic Co-operation and Development. *Main Science and Technology Indicators 2000/1*. OECD, Paris, 2000.

[15] J.L. Ringuest and S.B. Graves. The linear R&D project selection problem: An alternative to net present value. *IEEE Transactions on Engineering Management*, 37:143–146, May 1990.

[16] R.E. Steuer, L.R. Gardiner, and J. Gray. A bibliographic survey of the activities and international nature of multiple criteria decision making. *Journal of Multi-Criteria Decision Analysis*, 5:195–217, September 1996.

[17] C. Stummer. A two-phase approach to group decision making in R&D project selection. In *Proceedings of the INFORMS-KORMS Seoul 2000 Conference*, pages 900–905, 2000.

[18] T. Stuetzle and M. Dorigo. ACO algorithms for the quadratic assignment problem. In Corne, Dorigo and Glover, editors, *New Ideas in Optimization*, McGraw-Hill, London, pages 33–50, 1999.