5.1 Greedy Randomized Adaptive Search Procedure

In section 3.1.1 we saw how randomization can make constructive search methods better than pure greedy methods, because the danger of being too eager is dampened a little.

We called this approach \textit{randomized adaptive search}. Now we want to explain a metaheuristic which is based on this idea. It is called \textit{Greedy Randomized Adaptive Search Procedure} (GRASP, cf [70, 170]).

As it is a constructive search method, it starts with an empty solution and adds solution elements to a partial solution until it is complete. It may be argued, this is exactly the way a greedy construction method works, but there are subtle differences, which are pointed out now.

Greedy methods do not perform search, they construct a single solution in an iterative fashion by evaluating all remaining solution elements and according to their performance add them to the partial solution. Elements are added as long as the solution is improved. If this is not the case anymore, the construction is stopped and the final solution is returned. Hence greedy methods do not perform search.

GRASP uses a so called restricted candidate list (RCL) which enables the algorithm to sample different solutions from the solution space and therefore perform a search. The main idea of the algorithm is to relax the condition of adding the best performing element. Instead, a list of promising candidate elements is built. From this list a random element is chosen and added to the current partial solution. This way a bias towards good solutions is given but at the same time different solutions can be constructed, as we have seen in section 3.1.1.

The argumentation given in [70] is the following: Through the use of a RCL variations in the solutions and their respective quality is imposed. Consider the length of the restricted candidate list is one, i. e. always the best performing element is added, then the same solution is constructed in every iteration. Hence the variance in solution quality is zero. By choosing a random element from the RCL different solutions...
are sampled, hence the final solution quality varies (to a certain degree). Thus some constructed solutions are worse and others are better than the mean solution quality.

In the upcoming section a more thorough discussion of the GRASP metaheuristic is given.

### 5.1.1 Main Components of Greedy Randomized Adaptive Search Procedures

Generally, GRASP searches by repeated solution construction, i.e. in every iteration GRASP starts with an empty solution, iteratively adds elements to the partial solution and returns a complete solution. Which is then improved using another search method.

![GRASP - Solution Processing](image)

A high level view of the general solution processing scheme is illustrated in figure 5.1. There it can be seen, the construction phase leads to a complete solution, which is then further optimized. For example in [70] they propose a neighborhood based search\(^1\). In the remainder of this section only the construction phase is further discussed.

During the solution construction phase only heuristic information is used, i.e. no information of the search history is incorporated by the construction.

Now the main components of GRASP are identified: It starts with an empty solution and evaluates all candidate elements according to a performance function for their influence on the quality of the current partial solution. Those elements are sorted in decreasing performance order in a so called restricted candidate list. From this sorted list, the best elements are added to the restricted candidate list from which then a random element is chosen and added to the partial solution.

From this description it is clear that the GRASP construction phase consists of the following elements:

- Candidate evaluation
- Restricted candidate list construction

\(^1\) A description of search methods able to improve single solutions can be found in the section 6.