Understanding Dynamic Optimization

# Logistical Application of "Greedy Algorithms"

For many years, mathematicians and computer science researchers have studied what has been named "greedy algorithms" for solving optimization problems. This paper analyses greedy algorithms and their principles as they apply to the optimization of logistical processes.

## Greedy Algorithms

An algorithm is a step-by-step recipe for solving a problem. More formally, it is a mathematical procedure often used to solve optimization problems in a finite number of steps. Most times the procedure involves the repetition of the same operation in each step to perform a set of choices.

While most solution algorithms make choices based on a global overview of all current and future possibilities aiming at reaching the single global optimum solution, **greedy algorithms** make choices that look best at that very moment. In other words, they make locally shortsighted choices believing that the choice will eventually lead to the global optimum solution.

When compared to algorithms that guarantee to yield a global optimum solution, greedy algorithms have several advantages: They are easier to implement, they require much less computing resources, they are much faster to execute. Their only disadva ntage being that they not always reach the global optimum solution; on the other hand, even when the global optimum solution is not reached, most of the times the reached sub-optimal solution is a very good solution.

For many things, not reaching the single optimum solution is not acceptable. For instance if a greedy algorithm were used to implement a "chess game", finishing second best would not be acceptable.

Mathematicians have spent considerably time studying under what conditions greedy algorithms yield global optimum solutions. Several classical optimization problems like *minimum spanning tree*, *shortest path from a single source*, and *optimal prefix codes for data compression* yield global optimum solutions using greedy algorithms. On the other hand, the *knapsack problem* does not. For a detail presentation of this issue, see "Introduction to Algorithms" by Thomas H. Cormen et al, chapters 17 and 24.

## The Logistical Application of Greedy Algorithms

Most logistical problems however are perfectly suited to solutions using greedy algorithms. This is primarily due to the nature of the problems themselves. First for most logistical solutions, the optimal solution is not "a fixed" point. The optimal solution changes through the course of the activity, trailers fail to arrive, expected goods are not available, conveyor jams limit productivity to unanticipated values. Conventional (non-greedy) solutions to these un-anticipated events is to either ignore them altogether or to take a "what if" approach. Conversely greedy algorithms follow a "what is" approach that neither ignores ("what is" is "what is") or attempts to predict "what if". What could be sacrificed using a


4221 Freidrich Lane Suite 150
Austin, TX 78744-1062
Phone 512.851.2377
Fax 512.851.2379
www.VARGOcompanies.com

"greedy algorithm" is when looking back at the entire events for a period it may be possible to find a bit better way to have solved the problem. Like human beings, computer programs also have 20-20 hindsight.

## Distribution Center Applications

The optimization problems that are normally found in distribution centers and the way that they are traditionally approached have some peculiar features that are interesting to analyze.

The problems can be very large. Using algorithms that guarantee global optimum solutions could take an astronomic number of iterations that even with today's fastest computers would take many years to execute. A good example of this type of problem is selecting orders for the day waves trying to minimize the number of active SKUs per wave.

Even if a global optimum solution is found at the beginning of the day, after running the optimization program all night, the chances of having the day going according to the global optimum solution are very slim. Distribution center processes are very dynamic and full of unplanned events. Equipment breaking, operators having a "bad hair" day and performing below their standard, last minute orders, inventory errors, straggler orders are part of the normal operation of a distribution center. They are the rule rather than the exception.

Most distribution centers that use optimization techniques to schedule their activities spend a very large amount of effort and resources in finding the optimum schedule. Then, in order to have the process conform to the optimum plan, large and costly controls are implemented to handle the normal exceptions and to bring the process back to the scheduled plan. The common result is that most of the benefits of the optimization are spent in the handling of exceptions. The saddest part of this issue is that the originally found global optimum solution is not any more the best solution for the new conditions of the system, leaving the distribution center spending effort and money in sticking to a non-optimal solution.

Greedy algorithms have features that play very well for distribution center applications. If the main disadvantage of greedy algorithms is that they do not guarantee yielding a global optimum solution, this may not be a big problem, or a problem at all, in a distribution center where the global optimum solution is continuously changing.

The inherent nature of greedy algorithm (look for what looks best at the moment) has the advantage of a short scope of data, which happens to be more reliable. The farther in the future that the calculation goes, the more likely the conditions are to change.

ADS engineers have used variations of greedy algorithms called "Dynamic Optimization" to implement processes in distribution centers for over thirty years. Systems using these techniques have an unmatched level of success. The advantages of greedy algorithms are perfectly suited to distribution center applications where predictability is difficult if not impossible. Optimization problems become more manageable, the faster optimization programs can run more or less continuously during the operation adapting in real-time, to the ever-changing operating conditions. In situations where there are absolutely no exceptions, while they **may** not yield the very best possible solution they do provide a very good one. However, in situations where un-anticipated events do occur in the process, solutions using greedy algorithms far exceed solutions provided by conventional systems.


4221 Freidrich Lane Suite 150
Austin, TX 78744-1062
Phone 512.851.2377
Fax 512.851.2379
**www.VARGOcompanies.com**
 