THE RISE OF Artificial Intelligence

Real-world Applications for Revenue and Margin Growth

Zbigniew Michalewicz
Leonardo Arantes
Matt Michalewicz

Published by Hybrid Publishers
Melbourne Victoria Australia

Hybrid Publishers,
PO Box 52, Ormond VIC Australia 3204.
www.hybridpublishers.com.au

NATIONAL LIBRARY OF AUSTRALIA

A catalogue record for this book is available from the National Library of Australia

First published 2021
Text design & layout by Midland Typesetters, Australia
Cover design by Marchese Design
ISBN 9781925736625 (p)
Printed by Tingleman Pty Ltd

Dedicated to the entrepreneurs, scientists, and business leaders that have paved the way for Artificial Intelligence over the decades past, and are paving the way for its future in the decades to come.

PREFACE

What This Book is About and How to Read It

"We're at the beginning of a golden age of AI. Recent advancements have already led to inventions that previously lived in the realm of science fiction—and we've only scratched the surface of what's possible."

Jeff Bezos, *Amazon CEO*

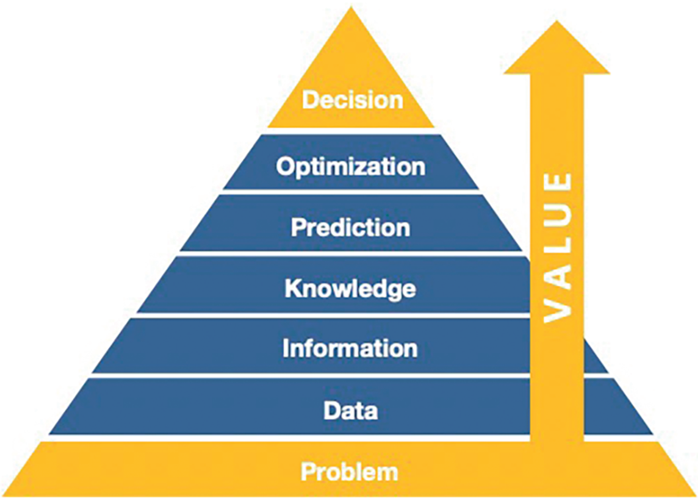
Few terms have captured our imagination in recent times like "Artificial Intelligence." And not just through sensationalized media articles about how AI will soon displace all jobs and rule the world, but also through movies, books, and television shows. It now seems that everyone "knows" about AI; that everyone has an opinion. And yet, in our experience, few people actually understand what Artificial Intelligence is and isn't, where the field came from and where it's heading, and how the technology can be harnessed to generate commercial outcomes.

Given the immense amount of disinformation and misunderstanding, we have written this book to demystify the subject of AI and explain it in simple language. Most importantly, we have written this book with the business manager in mind, someone interested in the topic from a real-world, commercial perspective—a perspective of how the technology can create value and increase competitiveness *today*, rather than what might happen in 25 years' time or how a superior intelligence might overcome the human race in the distant future. Such philosophical treatises are thought-provoking (to say the least) and the subject of many books published each year, but this isn't one of them. Instead, *The Rise of Artificial Intelligence* provides a commercial exploration of AI, with particular emphasis on how AI-based systems can improve decision making in organizations of all shapes and sizes.

As such, this book presents Artificial Intelligence through the lens of decision making for two reasons: First, because the world has reached a level of such unprecedented speed, complexity, and noise, that no one can assess and evaluate all the available data when making decisions; and secondly, because the decisions we make affect the outcomes we achieve. In other words, better business decisions lead to better business outcomes. Although Artificial

Intelligence can be applied to many areas besides decision making—such as automation and robotics, or image and speech recognition—these subjects don't feature heavily in the pages ahead except for Chapter 1, where we provide an overview of the research areas of AI. Ultimately, revenue and margin growth comes down to the decisions an organization makes (or doesn't make), and hence the application of AI to decision making is our primary focus.

To best present the concepts in this book, we've used a *problem-to-decision pyramid* to represent the continuum that exists in terms of an organization's ability to improve its decision making:



Each layer of this pyramid represents a step in the journey for improved decision making: the higher we go, the better our decisions (and the more value we can create). The structure of *The Rise of Artificial Intelligence* reflects the structure of this pyramid, with the first two parts of the book investigating each layer of the pyramid, and the last two parts illustrating the application of Artificial Intelligence to real-world problems for the purpose of generating revenue and margin growth.

Chapter 1 begins with a high-level overview of Artificial Intelligence—its history, areas of research, and current progress and challenges—before introducing the *problem-to-decision pyramid* in Chapter 2, which conceptualizes the journey from defining a problem to making a decision through the use of data, information, knowledge, prediction, and optimization. Chapter 3 concludes Part 1with an in-depth examination of a complex business problem set in the fast-moving consumer goods industry, which is used to explain the role of objectives, business rules and constraints, and the application of Artificial Intelligence algorithms for improved decision making.

This complex business problem of promotional planning and pricing is then used as a running example throughout Part II, which explores the inner workings of predictive models, optimization methods, and various learning algorithms. Because data and modeling form the basis of prediction and optimization, this part of the book opens with a chapter on data and modeling, along with a discussion of common issues such as data availability, completeness, and preparation. In Chapters 5 and 6 we review various AI and non-AI methods for predictive modeling and optimization, whereas in Chapter 7 we present adaptability and learning concepts—which together (i.e. prediction, optimization, and self-learning) comprise the backbone of any AI-based software system.

As an important aside, Chapters 4 through 7 represent the most technical material of the entire book, attempting to explain the innermost mechanics of several Artificial Intelligence algorithms such as neural networks and genetic programming. Although non-technical readers can easily progress through Part II to gain a deeper understanding of algorithms and models, readers without an interest in data, problem modeling, or how Artificial Intelligence algorithms work, can jump straight to Part III, which presents real-world applications of Artificial Intelligence.

The application areas in Part III explore the problem-to-decision pyramid in the context of real-world problems and business objectives, covering both the lower layers of the pyramid focusing on data and the analytical landscape of an organization (i.e. information and knowledge), as well as the upper layers of prediction, optimization, and self-learning, and how they're enabled by Artificial Intelligence methods. For ease of reading, we've divided Part III into three chapters, each being dedicated to a specific business function—in particular, *sales, marketing,* and *supply chain*. These case studies are based on an enterprise software platform called Decision Cloud®, which is a modularized, cloud-based platform that empowers staff to make better and faster decisions through the use of Artificial Intelligence.

And finally, Part IV concludes the book with common questions and concerns that organizations have on the application of Artificial Intelligence, such as: *"Would AI work for me?"* and *"Where should I start?"* These two chapters provide practical advice for selecting the right business problem, developing a business case, choosing a technology partner, as well as other topics such as digitalization and change management.

To improve the reader's understanding of the content, we've also created a set of supplementary videos that can be accessed at: www.Complexica.com/book/RiseofAI/. These videos bring to life the concepts presented in each chapter—for example, by providing a visual explanation of ant system algorithms in Chapter 1, the layers of the problem-to-decision pyramid in

Chapter 2, the workflow of promotional planning and pricing in Chapter 3, and so on. In these videos we're able to "show" concepts that can only be "told" within the confines of the printed page.

In terms of how to read this book or watch the videos, the ideal way is to progress sequentially from Chapter 1 to 12. For the less technically-inclined reader, however it's possible to jump around in any sequence that best satisfies curiosity and interest. For example, the reader might begin with an overview of Artificial Intelligence in Chapter 1, then progress to the application areas in Chapters 8, 9, and 10, before returning to Chapters 2 and 3 to better appreciate the problem-to-decision pyramid and the intricacies of solving complex business problems (after all, why are complex business problem so difficult to solve?). Alternatively, a reader might start with the application areas in Chapters 8, 9, and 10, then move back into Part II to better understand how algorithms and models work, before progressing to Part IV for practical advice for initiating an Artificial Intelligence project.

However, regardless of the reader's technical sophistication or their interest in the implementation aspects of AI-based software, it's highly recommended that everyone start with the first two chapters for an introduction into the world of Artificial Intelligence and an overview of basic concepts and terminology. From this perspective, the sequence of reading the remaining chapters is of far lesser importance.

Lastly, we'd like to say that the material presented in this book is the result of 40 years of first-hand Artificial Intelligence research within university settings, and more than twenty years of implementing AI-based enterprise software systems in many (often very large[1]) organizations across three continents. With that in mind, we'd like to thank everyone who made this book possible, with our special appreciation going to many Australian companies we collaborated with over the years in the application of Artificial Intelligence, such as PFD Foods, BHP Billiton, BMA, Pernod Ricard Winemakers, Lion Drinks, Bunzl, DuluxGroup, Rio Tinto, Metcash, Pfizer, Janssen, Haircare Australia, Fortescue Metals Group, CBH Group, Roy Hill, Glencore, Polyaire, Treasury Wine Estates, and Costa Group. Within these companies, we'd like to thank Chris Baddock, John Barakat, Renato Bellon, Simon Bennett, Damian Bourne, Warren Brodie, Michael Brooks, Pierre-Yves Calloc'h, Daryl Chim,

1 Our experiences of implementing enterprise-grade software based on the latest Artificial Intelligence algorithms and methods are based on many projects with global giants—such as BHP Billiton, General Motors, Bank of America, Pernod Ricard, Unilever, Air Liquide, Ford Motor Company, Glencore, Beiersdorf, Rio Tinto, and ChevronTexaco, among many others—as well as smaller companies that benefited from the research & development and innovation carried out by these larger organizations.

Richard Cohen, Jevan Dickinson, Andrew Endicott, Eglantine Etiemble, Scott Fellingham, Greg Feutrill, Garth Gauvin, Ward Gauvin, Scott Graham, Chris Green, Kylie Grigg, Richard Hansen, Mark Hayden, Kim Heatherton, Mark Ivory, James Jones, Mike Lomman, Brett McKinnon, Stuart McNab, Doug Misener, Luke Mitchell, Stephen Mooney, Aemel Nordin, Mark Powell, Rod Pritchard, Robin Pyne, Mathew Regan, Darryl Schafferius, Mark Shephard, Jon Simpson, Kerry Smith, Richard Taylor, Soner Teknikeller, Lance Ward, John Warda, and Joel Zamek.

We'd also like thank a few individuals who contributed to the content and ideas in this book, namely, Reza Bonyadi, Łukasz Brocki, Tom Heyworth, Xiang Li, Łukasz Olech, Ali Shemshadi, Larisa Stamova, Chris Zhu, as well as members of Complexica's scientific advisory board who we've worked with over the years: Reza Bonyadi, Łukasz Brocki, Longbing Cao, Raymond Chiong, Vic Ciesielski, Carlos Coello, Ernesto Costa, Kalyanmoy Deb, Kenneth De Jong, A.E. Eiben, Xiaodong Li, Masoud Mohammadian, Pablo Moscato, Frank Neumann, Zbigniew Raś, Markus Wagner, Thomas Weise, Adam Wierzbicki, and Mengjie Zhang.

And finally, it was a great pleasure to write about a topic that's been the central focus of our working lives for so many years, and we hope that readers enjoy this book as much as we enjoyed writing it. We believe that anyone in any organization who makes operational, tactical, or strategic decisions—whether on the factory floor or in the boardroom—will find this book valuable for understanding the science and technology behind better decisions. Enjoy!

Adelaide, Australia
March 2021

Zbigniew Michalewicz
Leonardo Arantes
Matt Michalewicz

TABLE OF CONTENTS

CHAPTER 10

Supply Chain

"Leaders win through logistics. Vision, sure. Strategy, yes.
But when you go to war, you need to have both toilet paper and
bullets at the right place at the right time. In other words,
you must win through superior logistics."

Tom Peters, *business author and speaker*

Supply chains are all about "supplying" the items we buy and consume each day through complex "chains," which move and process raw materials to make the final products we see on store shelves. Depending on the industry, a supply chain could be as straightforward as a few retail shops, warehouses, and trucks, or as complex as a sprawling network of mine sites and processing plants connected by rail and sea transport. From a higher perspective, however, all these supply chains are linked in one way or another through an intricate web of interactions. If we think about any common product—such as a bottle of wine or family car—we can trace the individual components of those products back through their respective supply chains, back through the trucks and ships that brought those products to the liquor store or dealership, back to the factories where those products were assembled, back through the transport network that brought those components to the factory, and so on.

In the case of wine, the grapes have to be grown first, before they begin their multi-month journey through harvesters, trucks, weighbridges, crushers, and other processing facilities to become the colorful accompaniment at our dinner table. And when it comes to cars, some components originate at mine sites, where the iron ore that will eventually become the car's frame and doors and hood is extracted from a pit. Each of these steps in the supply chain has its own challenges and complexities; for example, planning a mine site requires consideration of what grade of ore is required at what point in time, coupled with truck and digger availability, workforce rosters, maintenance schedules, and more—which is just the first step in the process—followed by the scheduling of trains that will transport the ore from various mine sites to the port, where the coordination of stackers and reclaimers happens to ensure that each ship is loaded on time. And after that, there is more, much more, as the ore arrives in another country and is heated by coal (which arrived at the furnace through a similarly complex supply chain) to become steel, which in turn is molded into the car's frame and doors and hood—components that represent just a handful of the 30,000 parts that make up the average car, each of which has their own supply chain from raw materials to finished part. On top of this, the automaker needs to predict consumer demand for its cars across different countries—a difficult problem in itself—all while the cars are being assembled and placed on ships for transport to those markets.

At every step, there is complexity, and the more steps we consider together, the more complex the problem becomes. This inherent complexity makes supply chain problems particularly well suited for the application of Artificial Intelligence and Decision Optimization Systems. Given the multi-component nature of supply chains (as discussed in Chapter 6.10), we'll present each part separately: First demand planning and inventory in Section 10.1, then production planning and scheduling in Section 10.2, followed by logistics and distribution in Section 10.3. At the highest level, these are the core components of a supply chain operation: predicting demand, planning and scheduling production (whether it be the production of iron ore from a mine or the assembly of cars in a factory), and then organizing logistics and distribution. Each component represents a complex business problem in itself, and together, an almost impossible challenge.

10.1 Demand Forecasting and Inventory Optimization

The holy grail of supply chain optimization is predicting what will be sold, in what quantity, where, and when—with 100% accuracy. An organization capable of doing that could run the leanest possible supply chain—with minimal inventory levels—while always satisfying customer demand and never stocking out. Unfortunately, such prediction accuracy is impossible to achieve, and for that reason, all organizations must carry some level of inventory to buffer against unexpected changes in demand.[1]

Despite the fact that organizations of all shapes and sizes have gorged themselves on demand planning software over the past few decades, demand forecasting still remains an unsolved problem within most organizations—"unsolved" in that the forecast error is still high, leading to stockouts, as well as excessive inventory levels and obsolescence. "We still carry a lot of

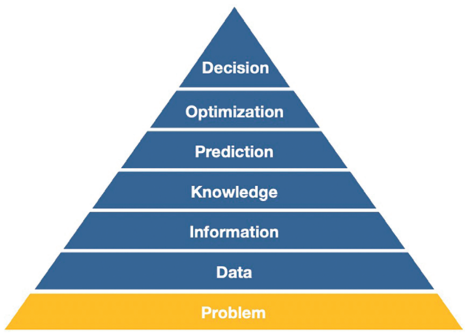
1 Another reason that organizations carry inventory is because the lead times on raw materials can also vary, so without a buffer of these inputs the manufacturing process can come to a halt. Such inventory is often classified as *raw materials inventory*—which could be the car frame, door, or hood—versus finished goods inventory, which is the finished (i.e. completely assembled) car itself.

inventory," these organizations will confess, "not as much as before, but still a lot. *Unfortunately, it's in the wrong place, at the wrong time."* Hence, not only are working capital and obsolescence costs still high, these organizations routinely suffer lost sales due to stockouts.

MAX Hardware found itself in precisely this situation. After evaluating many software systems for demand planning, the company purchased and implemented a system that best fit their operation from a features, functionality, and workflow perspective. Being a manufacturer and distributor of hardware products—such as nuts and bolts, screws, and various types of tools—the company sold its products through both retail chains and directly to the trade (i.e. electricians, plumbers, carpenters, etc.). Given MAX Hardware's extensive products range—in the thousands—along with significant lead times for certain raw materials and components, the company's flexibility was limited in ramping up production when inventory ran low. For this reason, improving demand forecast accuracy was of paramount importance—especially because trade customers couldn't wait for backordered products, so whatever MAX Hardware stocked out, these customers bought from competitors (sometimes leading to a permanent change in loyalty).

After the new demand planning system went live, however, it became apparent that forecasting accuracy was no better for a large number of products. Using these system-generated forecasts, MAX Hardware was still producing too much or too little of different product lines, creating excess inventory of some lines and shortages of others. As for products produced in the correct aggregate quantity (countrywide), they were frequently sent to the wrong distribution center and required expedited shipping to another distribution center to fulfill demand in that part of the country. Hence, the forecast for these products was accurate at the aggregate level, but highly inaccurate at the granular level of individual states or customer segments.

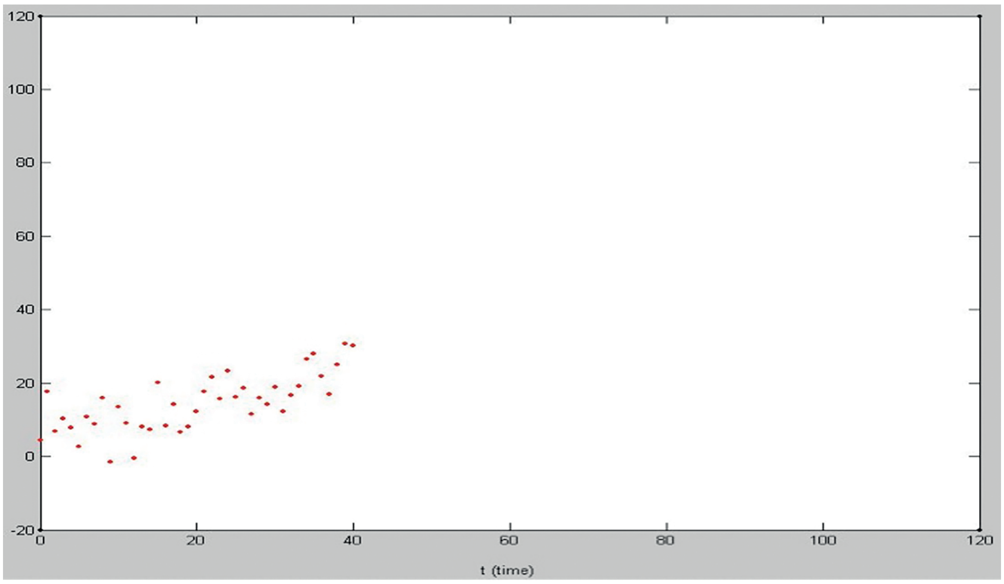
Because of these forecast accuracy issues, manual overrides became the norm at MAX Hardware as inventory managers and key account managers overrode the forecast in an attempt to improve its accuracy. In many cases, this made the situation worse, and a large amount of time was spent "playing around with the numbers," as management put it. Eventually, the forecasting function was removed from the system altogether, as MAX Hardware reverted to spreadsheets for generating a manual forecast—which meant that, in effect, the company had returned to the same state as before the demand planning system



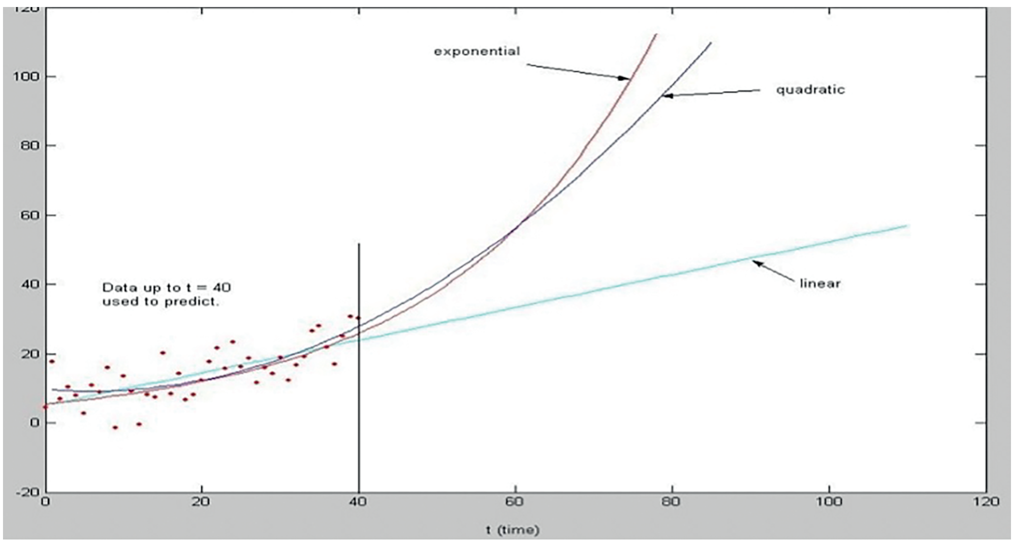
was implemented (i.e. using spreadsheets, gut feel, and manual processes to create the forecast). This situation continued for some time until management decided to take action and began searching for a system that could provide superior forecasting accuracy. As such, MAX Hardware defined their business problem and objective as:

Reduce inventory levels and stockouts through more accurate demand forecasting

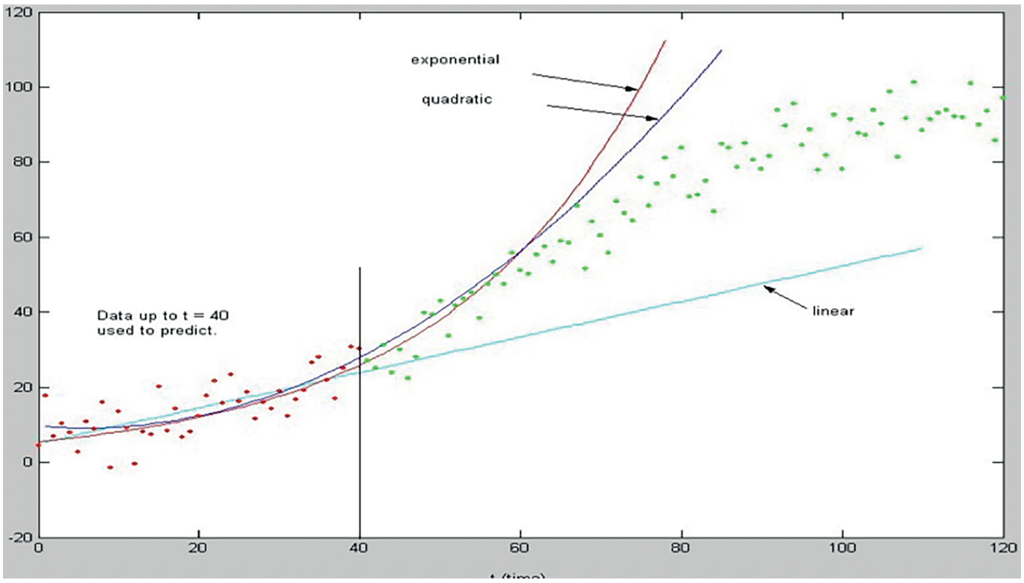
By going through the process of implementing the failed demand planning system, MAX Hardware realized that the vast majority of such systems were based on a standard set of statistical models that were configured in the same way: Namely, by taking the historical sales data for each product line—like the one shown below:



and finding the statistical model that best "fit" this data:

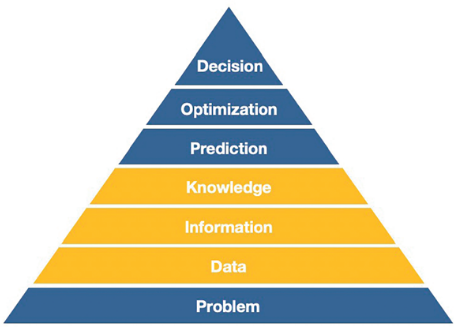


In the graph above, each curve is a statistical model, and each curve fits the data better than any other curve of its specific type. And so for the same historical data, we have three different predictions generated by three different models. But by only using internal data and standard statistical models, what inevitably happens is that the future turns out very differently to what these models predicted, thereby creating a forecast error of varying magnitude for different products:



Through this experience, MAX Hardware realized that demand forecasting was a scientific problem of selecting the most appropriate prediction method for the problem at hand and building a model (as discussed in Chapter 5), rather than a software problem of selecting the application with the most features and functionality. The real difficulty lay in predicting the future, which had to be addressed algorithmically within the selected demand planning system.

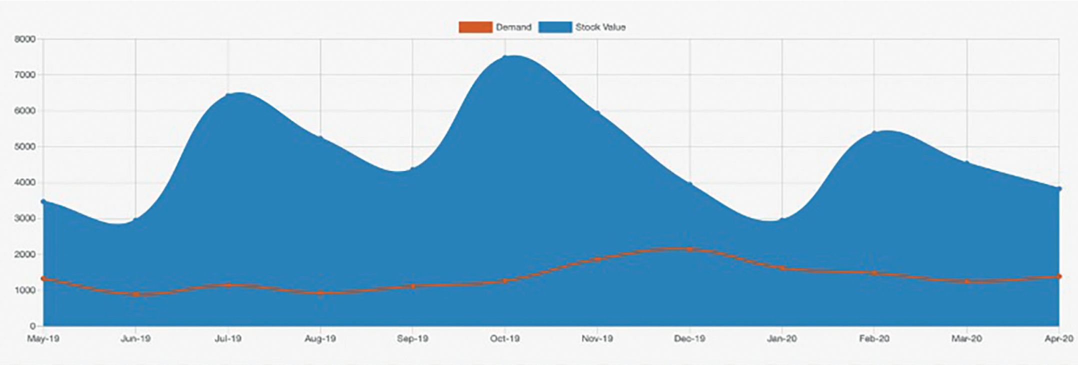
Data, information, & knowledge



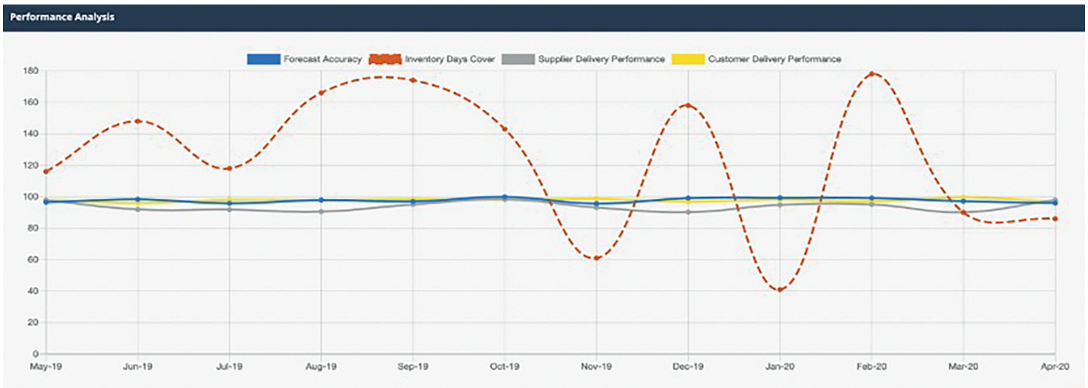
Like many other manufacturers in the building materials sector, MAX Hardware had a substantial amount of internal data, including:

- Historical sales by product by customer
- Historical pricing data by product by customer
- Historical inventory levels by product by week
- Historical forecasts created by inventory managers and by key account managers for retail chains

This data was used to provide inventory managers with a variety of reports and visualizations, including inventory levels against actual sales by product and time period:



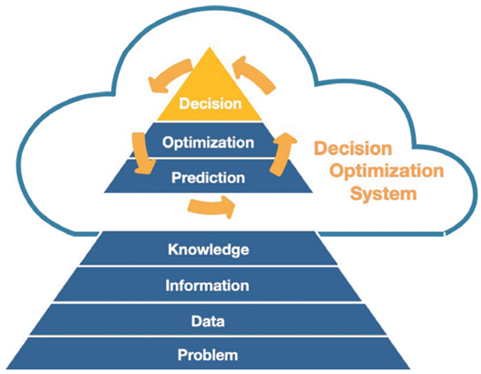
as well as historical performance on KPIs such as stockouts, customer fill rates, and inventory days cover:



Although these informational reports were plentiful, they didn't provide any predictive capabilities—only a rear-view mirror look at what happened in the past. To gain a better feel for future demand, MAX Hardware began experimenting with external data (such as building approvals and customer forecast data) in search of patterns that might repeat in the future. But such efforts were ad hoc, sporadic, and driven entirely by the analytical capabilities of the staff that undertook such analysis. For these reasons and others, the business case for a Decision Optimization System—one that would allow the company to hold the right inventory, at the right location, at the right time through improved demand forecasting accuracy—was created and endorsed by MAX Hardware.

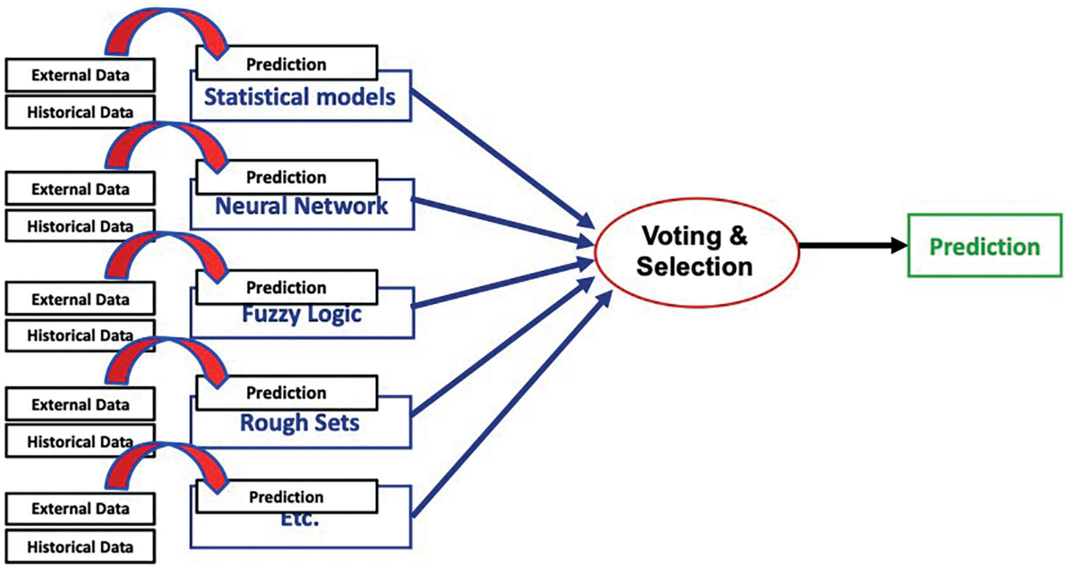
Decision Optimization System (prediction, optimization, & self-learning)

As is the case with many other complex business problems, MAX Hardware's business objective to simultaneously reduce inventory levels and stockouts was

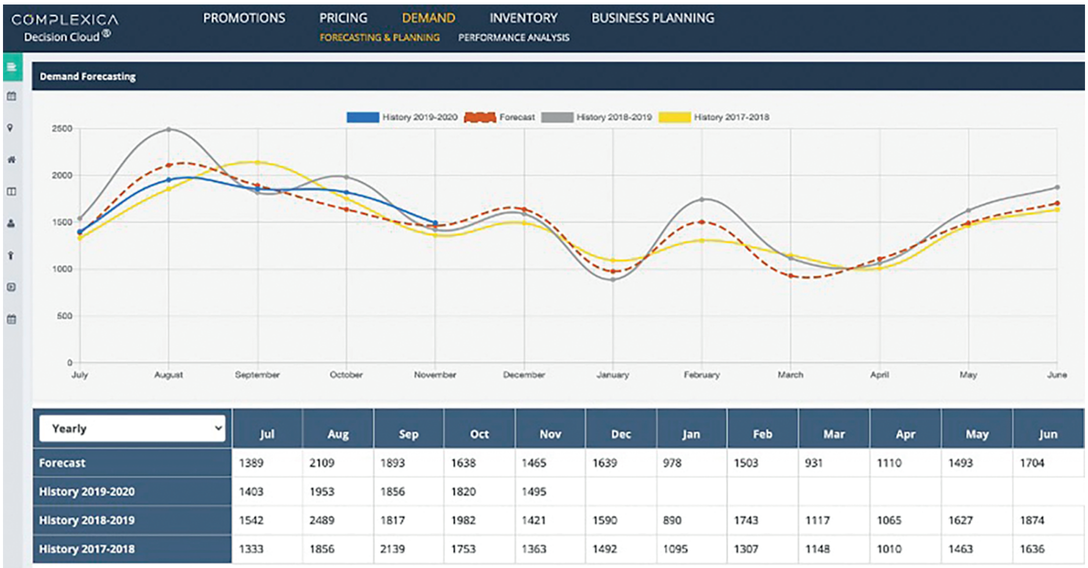


dependent on the accuracy of its prediction model—in this case, the accuracy of predicting future demand. Although it was still possible to improve inventory levels by using an inaccurate forecast (by holding a larger amount of safety stock for the most variable product lines and dynamically changing these safety stock levels throughout the year to account for seasonality and other demand effects), the largest benefit would accrue through improved forecast accuracy.

Knowing now that the problem of forecast accuracy was algorithmic in nature, the new Decision Optimization System had two fundamental differences from the first demand planning system: First, it used an ensemble model that combined statistical models with AI-based methods such as neural networks and fuzzy system (as discussed in Chapter 5); and second, each model was fed with both internal and external data to improve accuracy:

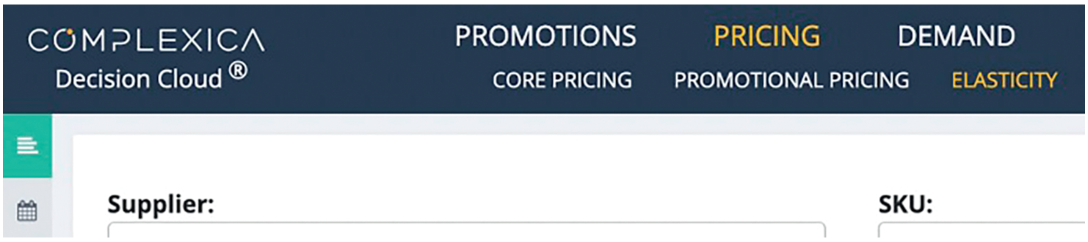


The ensemble model achieved a substantial increase in forecast accuracy over both the manual, spreadsheet methods, as well as the statistical models used by the failed demand planning system. This ensemble model became the prediction component of the Decision Optimization System, providing MAX Hardware with the most probable view of future demand by product, by distribution center, by time period, and in many cases, by customer:

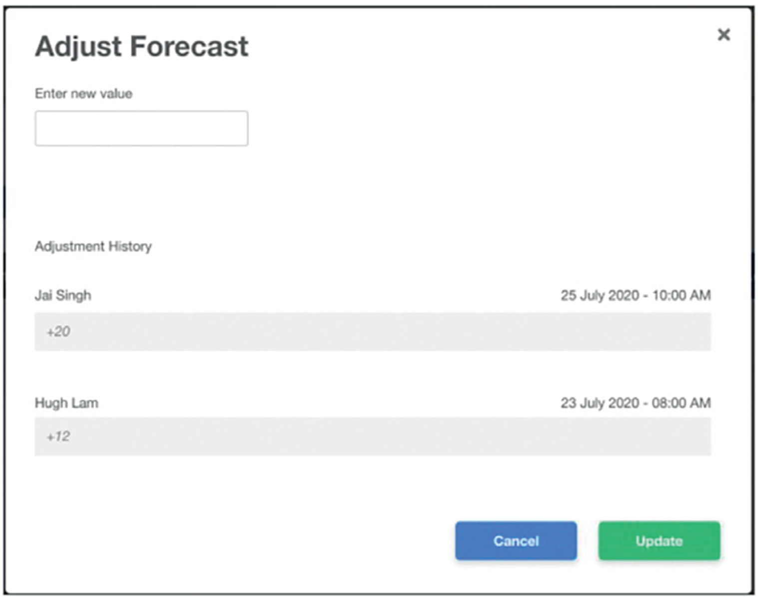


Yearly	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
Forecast	1389	2109	1893	1638	1465	1639	978	1503	931	1110	1493	1704
History 2019-2020	1403	1953	1856	1820	1495							
History 2018-2019	1542	2489	1817	1982	1421	1590	890	1743	1117	1065	1627	1874
History 2017-2018	1333	1856	2139	1753	1363	1492	1095	1307	1148	1010	1463	1636

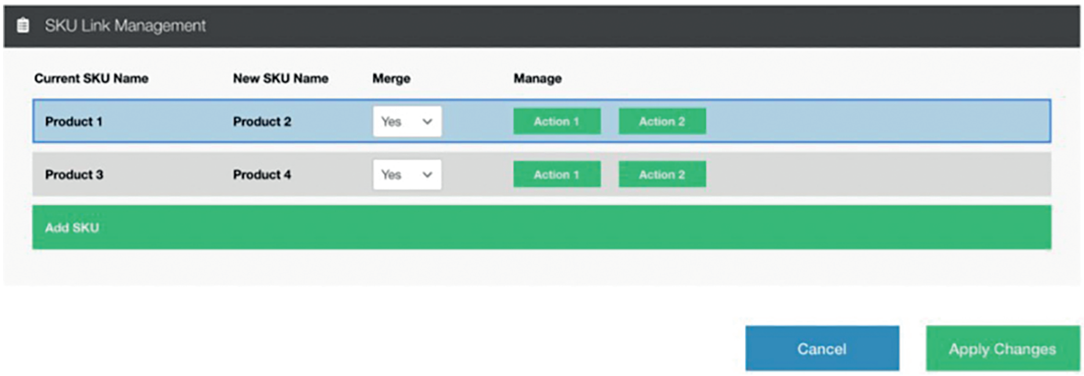
Once the Decision Optimization System was configured and implemented, it considered historical sales, customer forecasts, relevant external data, as well as promotional and pricing information for each product:



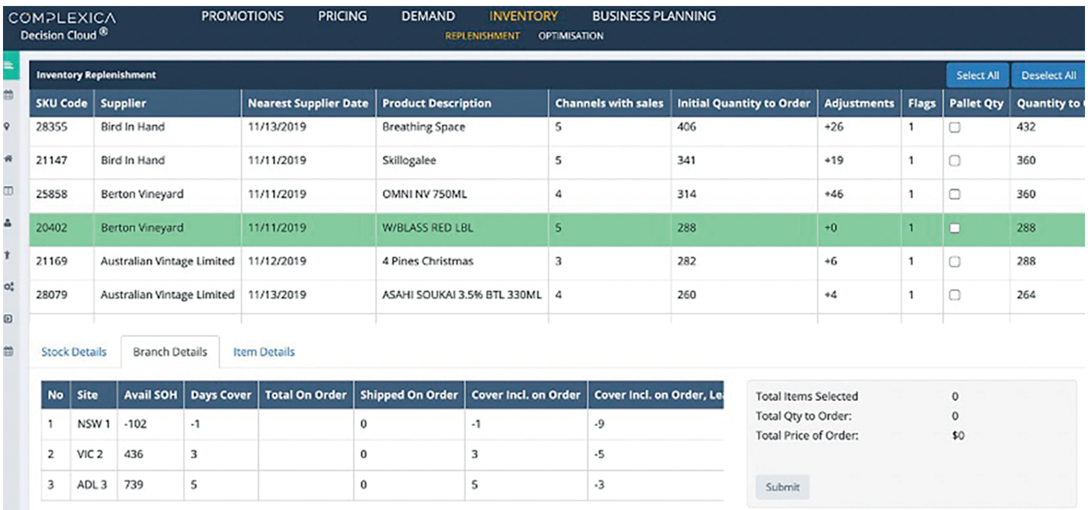
Although the demand forecast was generated by an ensemble model that used internal and external data, MAX Hardware still had the capability to override these forecasts:



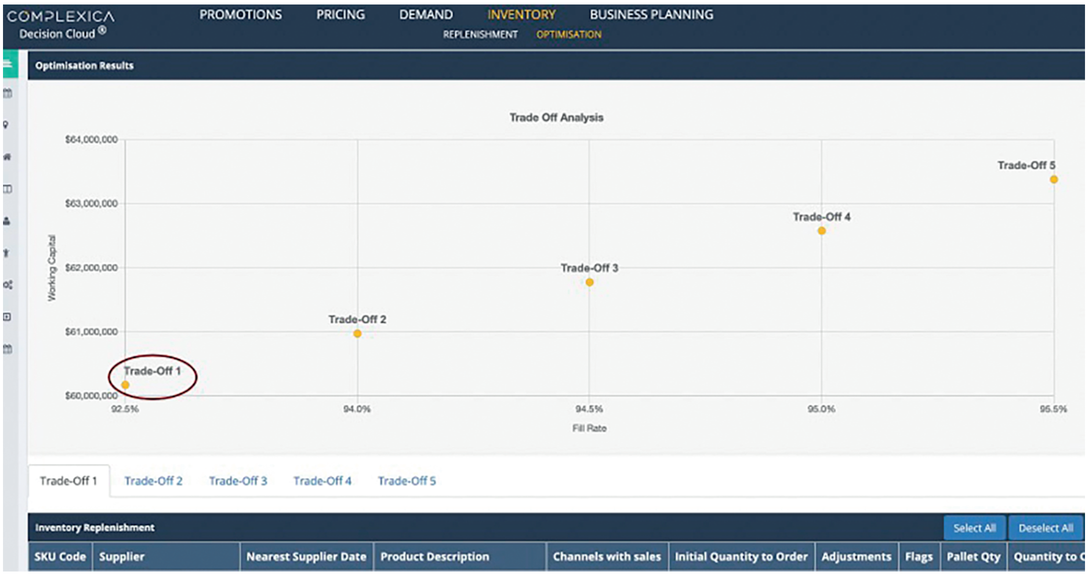
These overrides were captured in an audit log, and then analyzed by the Decision Optimization System to provide feedback on the effectiveness of each manual intervention (which in most cases were inferior to the system-generated forecasts). As for new products that lacked sales data, the Decision Optimization System used the historical sales data of similar products to estimate future demand:



The Decision Optimization System also allowed MAX Hardware to understand the trade-off between working capital levels and customer fill rates. This was done by defining working capital and fill rate targets, which could be set for all products and customers in aggregate, or broken down into individual targets for individual products, customers, distribution centers, and time periods, as shown below:



Once these targets were set, the Decision Optimization System would attempt to find a Pareto curve of optimized solutions (as discussed in Chapters 2.3 and 6.9) that illustrated the trade-off between inventory levels and customer fill rates:



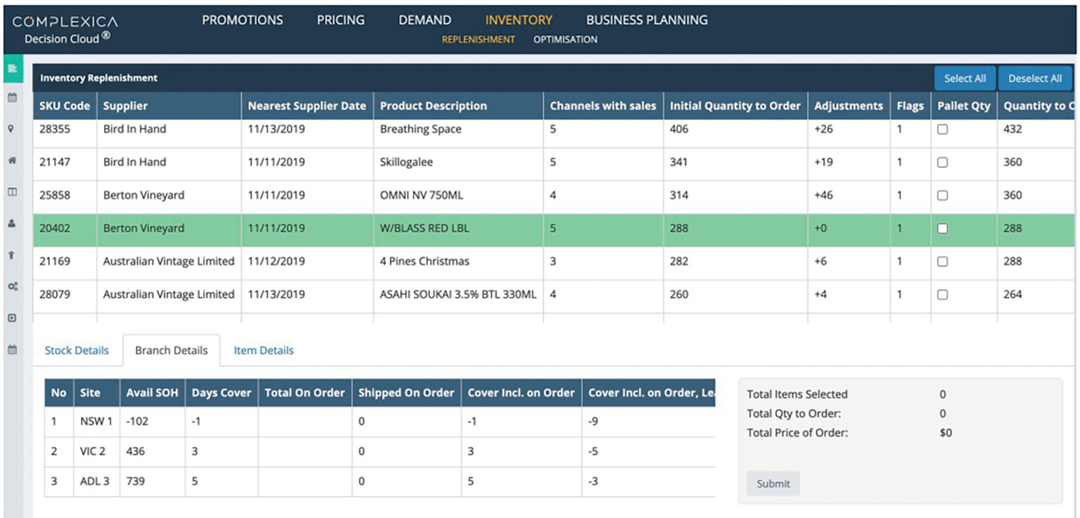
This advanced capability for multi-objective optimization allowed MAX Hardware to implement different inventory policies for different product lines, customers, distribution centers, and time periods, and was based on evolutionary algorithms (discussed in Chapter 6.7).

The reason that evolutionary algorithms were selected for optimization, is because this AI-based algorithmic method could simultaneously produce many potential solutions (i.e. a population of solutions)—hence it could generate a few final solutions at the end of a *single run*. Of course, these final solutions had to be substantially different from one another, because if the five solutions were quite similar (with just minimal differences), the usefulness of the trade-off results would be modest at the very best. To address this issue, the evaluation function of the evolutionary algorithm took into account the "uniqueness" of solutions: "similar" solutions were penalized, so they become less attractive as candidates for the next generation of solutions. Furthermore, the algorithm placed a premium on non-dominated solutions[2]—in other words, solutions where there was no single solution in the population of solutions better on all objectives (e.g. working capital and customer fill rates). Because of this evaluation function, the evolutionary algorithm improved the Pareto curve of solutions from one generation to the next, and the final result (consisting of several "best" solutions) was presented as a diverse set of possibilities that illustrated the trade-off between working capital and customer fill rates (as shown above).

And lastly, based upon the system-generated demand forecast and optimized inventory policies, the Decision Optimization System provided MAX Hardware with product replenishment recommendations that could be

2 See Chapter 6.9 for a full discussion on this topic.

reviewed/modified/accepted before being converted into production orders for finished goods, or purchase orders for raw materials:



The Decision Optimization System implemented by MAX Hardware provided a number of tangible benefits, including:

- Improved forecast accuracy, which was particularly important for hard-to-forecast product lines. On average, forecasting accuracy increased from approximately 64% to 89%, with many products exceeding 95%
- A 18% reduction in finished goods inventory
- A 43% reduction in stockouts, leading to a corresponding increase in customer fill rates (as measured by Delivery In Full, On Time metrics, "DIFOT")
- Less time and effort for inventory planning and replenishment, with some tasks being reduced from a few days to a few hours

MAX Hardware also realized additional benefits in metrics such as the cash-to-cash cycle time, stock turns, and customer loyalty, all of which contributed to the company's overall profitability and competitiveness.

10.2 Scheduling Optimization for Improved Asset Utilization, Throughput, and DIFOT

Every factory needs to plan and schedule its production, regardless of whether it's assembling cars, bottling wine, producing cardboard boxes, or extracting iron ore from a mine. These factories can be thought of as "nodes" within a supply chain, where raw materials and components go in one end and finished products emerge from the other. Many of these nodes are interconnected, where the output from one node is an input into the next. For example, the output from a mine could be iron ore or coal, which represents the raw material input into a steel-making factory. And in that steel-making factory, the finished sheets and slabs of steel become the raw material into the next manufacturing node, where the steel is formed into car components—and so on, with each node having its own demand forecasting, production planning, and scheduling process.

In this supply chain context, the words "planning" and "scheduling" are often used interchangeably, despite meaning very different things: Planning refers to *what* an organization will do, whereas scheduling refers to *when* an organization will do it. For this reason, planning is more macro and "higher-level" (i.e. deciding what products to produce each week or month, depending on the forecasted demand), while scheduling is more granular and exact. As a simple example, an airline might plan to provide 100 return flights between two cities for the month of May (based upon the forecast demand for travel)—which represents *what* the airline will do. This planning process is simpler than scheduling *when* these 100 round trips should occur: the exact time, crew, planes, maintenance, and so on. Hence, planning problems are usually easier to solve and optimize than scheduling problems.[3]

When it comes to production planning and scheduling within a factory, the same concepts apply. An automaker would first create a production plan for building a particular mix and volume of cars (again, based upon the forecast demand, production capacities, inventory levels at dealerships, as well as other considerations), and then use this plan to schedule the assembly of these cars (the exact components, production lines, and timing). Hence, the planning process is done at a higher, more macro level, whereas scheduling is granular and involves many complex details, such as the availability of input components and raw materials, labor constraints, production line availability, changeover times, maintenance schedules, and more.

In this case study we'll discuss CAST Metals, an organization with eight foundries spread across different locations, with each foundry operating several furnaces and casting machines. In a foundry operation, products are produced by melting metal inside a furnace and then pouring this heated liquid into a mold. Once the metal has cooled and solidified, the mold is removed to produce the final product (which could be a metal component for a railway network, automobile engine, pipe, or any number of other products). Creating and modifying a quarterly production plan of what products to produce was

---

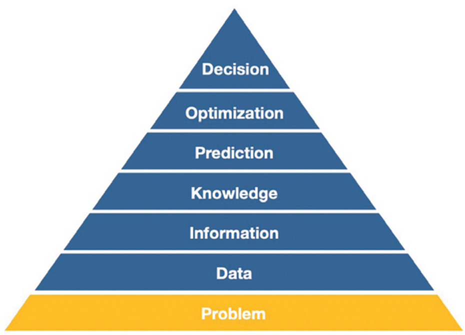
3 The predominant difficulty in planning problems lies in accurately predicting what will happen in the future; once this has been addressed, the planning process is usually straightforward. The difficulty in scheduling problems, on the other hand, is finding the schedule that maximizes or minimizes certain objectives—such as asset utilization or cost—from an almost infinite number of possible schedules.

part of CAST Metals' sales and operations planning ("S&OP")[4] process, in which confirmed and forecast demand was synchronized with manufacturing capacity and inventory levels. During this regular planning cycle, CAST Metals balanced production across its foundries by considering manufacturing capabilities and capacities, the location of its customers, transportation costs, as well as the overall production load and inventory across the network. Hence, demand forecasting, inventory management, and the global optimization of production across CAST Metals' eight foundries were addressed at the planning level, and were not a consideration for the scheduling process.

When the monthly production plan was converted into a weekly schedule—going down to hourly time buckets at the individual machine level—the objective was to meet customer due dates while simultaneously maximizing asset utilization and factory throughput. However, converting the higher-level plan into a detailed schedule was a complicated and difficult undertaking, requiring CAST Metals to consider many constraints and business rules for each individual foundry. Some of these constraints represented physical limitations (such as melting times and furnace capacities), while others represented operational business rules related to:

- Manufacturing some products during day shifts or night shifts
- Not manufacturing some products at the same time because of their similarity (making these products difficult to sort at the end of a production run)
- Operating certain casting machines on particular days (e.g. from Monday morning to Thursday evening)
- Using certain casting machines for particular products because of efficiency and tooling reasons

In addition to these business rules and constraints, the production schedule had to coordinate many independent processes, such as the preparation of cores and molds, pouring of molds, and the finishing of castings. There were also many relationships between various metal grades to consider, as well as the transition time for changing from one metal grade to another. Because of all these complexities, the result was substandard performance on metrics such as Delivery



4 Sales and Operations Planning is an integrated planning process for aligning and synchronizing various business functions of an organization.

In Full, On Time ("DIFOT") and Overall Equipment Efficiency ("OEE"), as well as excessive overtime labor due to last-minute schedule changes. Hence, CAST Metals defined their business problem and objective as:

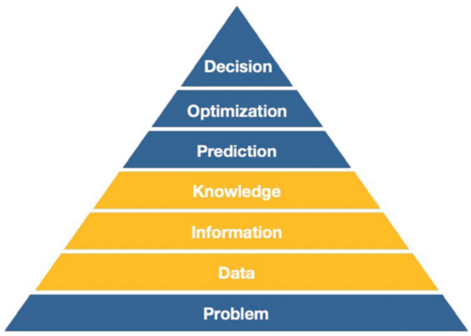
> *Simultaneously increase asset utilization, factory throughput, and customer service levels through optimized production scheduling*

To achieve this objective, CAST Metals decided to replace the manual, spreadsheet-based approach for production scheduling with a Decision Optimization System capable of:

- Converting production plans into detailed schedules that were optimized for asset utilization, labor costs, and customer service levels
- Dynamically "re-optimizing" the production schedule whenever circumstances changed (customer orders, machine failures, etc.)

Given the complexity of this business problem (i.e. an astronomical number of possible solutions, a dynamic environment with frequent changes, and many problem-specific constraints), Artificial Intelligence algorithms were the natural choice for optimization.

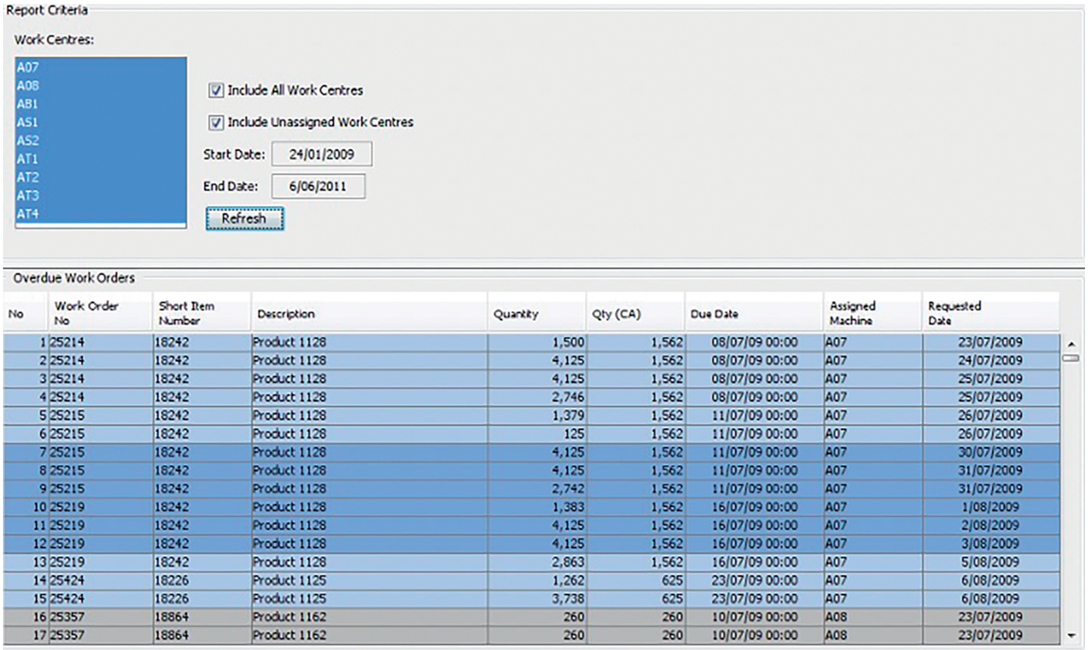
Data, information, & knowledge



When creating the monthly production plan or weekly schedule, CAST Metals had access to a variety of datasets, including:

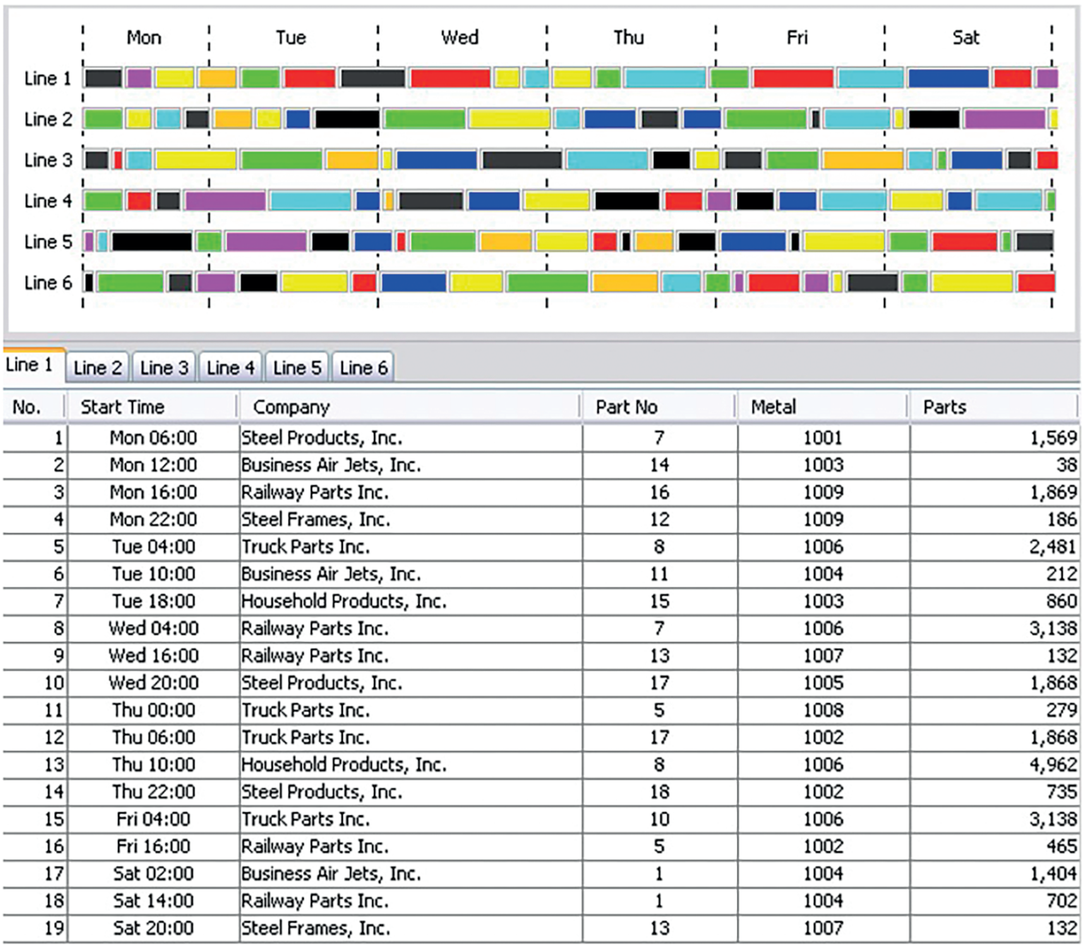
- Forecast orders by product
- Confirmed orders by customer
- Historical sales by product by customer
- Bill of materials for each product
- Product routing for each product for each foundry
- Historical customer service levels in terms of DIFOT metrics
- Historical factory performance levels in terms of asset utilization, maintenance schedules, labor costs (especially overtime), and breakdowns
- Historical inventory levels by product by week
- Current inventory levels
- Historical demand forecasts and their accuracy

This data was used to produce daily and weekly reports, such as inventory levels against actual sales by product, machine downtime, and overdue work orders, as shown below:



No	Work Order No	Short Item Number	Description	Quantity	Qty (CA)	Due Date	Assigned Machine	Requested Date
1	25214	18242	Product 1128	1,500	1,562	08/07/09 00:00	A07	23/07/2009
2	25214	18242	Product 1128	4,125	1,562	08/07/09 00:00	A07	24/07/2009
3	25214	18242	Product 1128	4,125	1,562	08/07/09 00:00	A07	25/07/2009
4	25214	18242	Product 1128	2,746	1,562	08/07/09 00:00	A07	25/07/2009
5	25215	18242	Product 1128	1,379	1,562	11/07/09 00:00	A07	26/07/2009
6	25215	18242	Product 1128	125	1,562	11/07/09 00:00	A07	26/07/2009
7	25215	18242	Product 1128	4,125	1,562	11/07/09 00:00	A07	30/07/2009
8	25215	18242	Product 1128	4,125	1,562	11/07/09 00:00	A07	31/07/2009
9	25215	18242	Product 1128	2,742	1,562	11/07/09 00:00	A07	31/07/2009
10	25219	18242	Product 1128	1,383	1,562	16/07/09 00:00	A07	1/08/2009
11	25219	18242	Product 1128	4,125	1,562	16/07/09 00:00	A07	2/08/2009
12	25219	18242	Product 1128	4,125	1,562	16/07/09 00:00	A07	3/08/2009
13	25219	18242	Product 1128	2,863	1,562	16/07/09 00:00	A07	5/08/2009
14	25424	18226	Product 1125	1,262	625	23/07/09 00:00	A07	6/08/2009
15	25424	18226	Product 1125	3,738	625	23/07/09 00:00	A07	6/08/2009
16	25357	18864	Product 1162	260	260	10/07/09 00:00	A08	23/07/2009
17	25357	18864	Product 1162	260	260	10/07/09 00:00	A08	23/07/2009

CAST Metals used these reports to balance urgent and overdue orders against run lengths and changeover times, with the output of this spreadsheet-based process being a day-by-day, line-by-line production schedule:



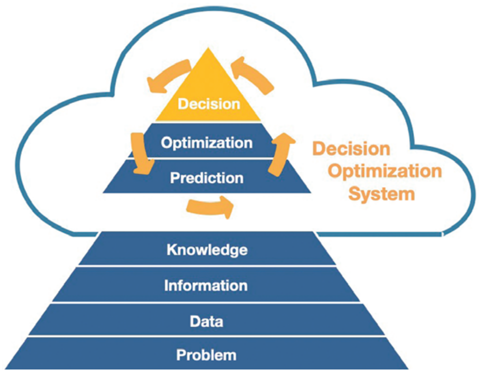
No.	Start Time	Company	Part No	Metal	Parts
1	Mon 06:00	Steel Products, Inc.	7	1001	1,569
2	Mon 12:00	Business Air Jets, Inc.	14	1003	38
3	Mon 16:00	Railway Parts Inc.	16	1009	1,869
4	Mon 22:00	Steel Frames, Inc.	12	1009	186
5	Tue 04:00	Truck Parts Inc.	8	1006	2,481
6	Tue 10:00	Business Air Jets, Inc.	11	1004	212
7	Tue 18:00	Household Products, Inc.	15	1003	860
8	Wed 04:00	Railway Parts Inc.	7	1006	3,138
9	Wed 16:00	Railway Parts Inc.	13	1007	132
10	Wed 20:00	Steel Products, Inc.	17	1005	1,868
11	Thu 00:00	Truck Parts Inc.	5	1008	279
12	Thu 06:00	Truck Parts Inc.	17	1002	1,868
13	Thu 10:00	Household Products, Inc.	8	1006	4,962
14	Thu 22:00	Steel Products, Inc.	18	1002	735
15	Fri 04:00	Truck Parts Inc.	10	1006	3,138
16	Fri 16:00	Railway Parts Inc.	5	1002	465
17	Sat 02:00	Business Air Jets, Inc.	1	1004	1,404
18	Sat 14:00	Railway Parts Inc.	1	1004	702
19	Sat 20:00	Steel Frames, Inc.	13	1007	132

Besides being labor intensive and time consuming, the scheduling process was inefficient for many other reasons, including:

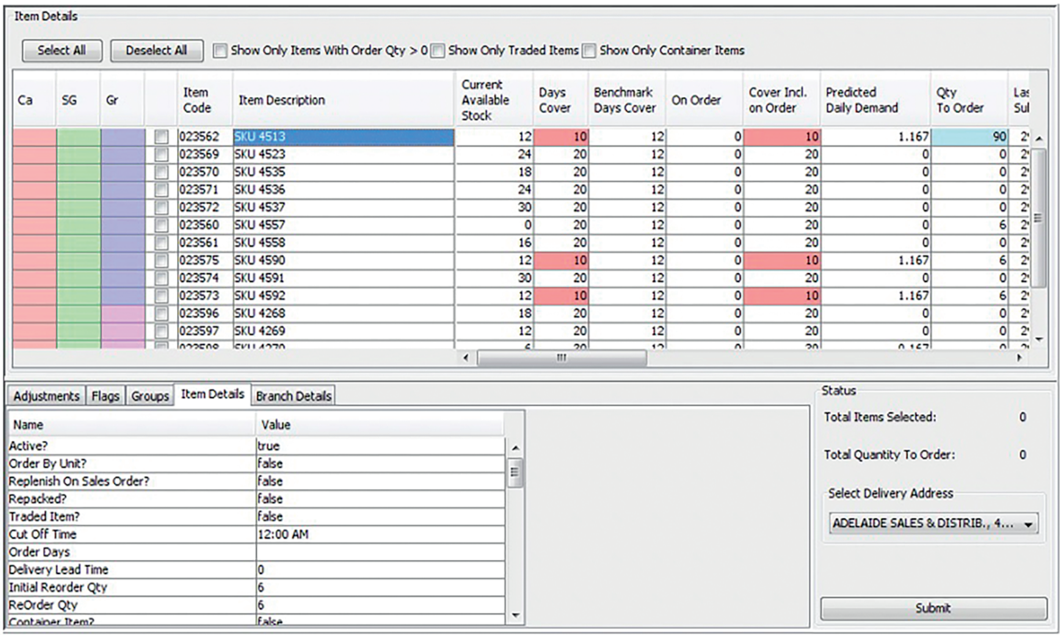
- The final schedule didn't consider many business rules and constraints, largely because these rules were in people's heads. Consequently, the production schedule was usually "un-executable," in that it omitted or abstracted certain variables such as maintenance, changeover times, differences in the defect rate between various machines, and variations in production run times. The production schedule was therefore seldom achieved, and schedule adherence was low.
- The production sequence was suboptimal, as no team of human experts could consider all possible scheduling combinations, which in turn led to factory performance issues.
- The schedule was static and disconnected from the factory floor, as there was no data feed from each machine to understand production from a "scheduled" vs. "actual" point of view. This meant that re-scheduling was a slow and painful process of first realizing that something had happened—such as machine failure, high defective rate, or some other event—followed by updating the spreadsheet-based schedule, before finally printing a new version and pushing it down to the factory floor (by which time it was again out of sync and not reflective of what was actually happening).

The business case for configuring and deploying a Decision Optimization System was based on achieving higher production volumes through each foundry (leading to greater revenue per site) and fewer late orders (leading to fewer financial penalties, greater customer satisfaction, and greater customer loyalty). Consideration was also given to potential future phases, where the Decision Optimization System could be extended to production planning and demand forecasting, thereby allowing CAST Metals to improve demand forecast accuracy and reduce inventory (as discussed in the previous case study), as well as globally optimize across all eight foundries to realize further efficiency gains.

Decision Optimization System (prediction, optimization, & self-learning)



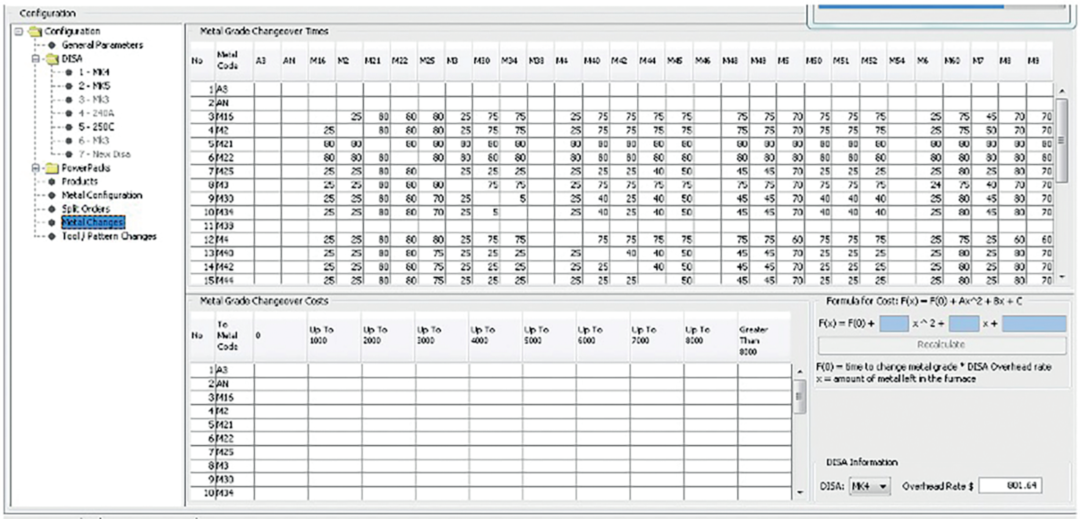
To enable optimized scheduling across its eight foundries, CAST Metals implemented a Decision Optimization System based on Artificial Intelligence methods for optimization. When converting the production plan into an executable schedule for each foundry, the Decision Optimization System considered the current inventory level of each product (as measured in days cover), as well as the designation of each order—namely, whether it was "make to stock" for replenishing inventory or "make to order" for a specific customer:



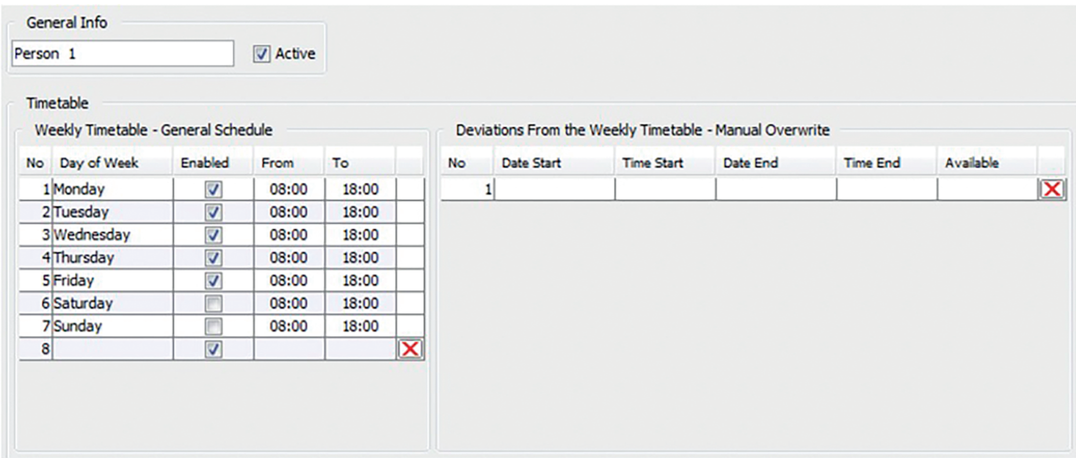
These inventory levels and designations impacted the prioritization of orders, with the Decision Optimization System placing more emphasis on orders where the product was being produced for a specific customer and no inventory existed for buffering the due date.

Another important consideration was the interplay between furnaces and casting machines, which represented the core scheduling issue. The primary objective was to optimize the distribution of production orders over some period of time in a way that maximized furnace utilization and machine throughput. Because the furnaces and machines worked together in the production process (first melting, then casting), the maximization of furnace utilization and production-line throughput had to be considered jointly. Secondary objectives included the maximization of DIFOT metrics and minimization of labor costs when optimizing the production schedule.

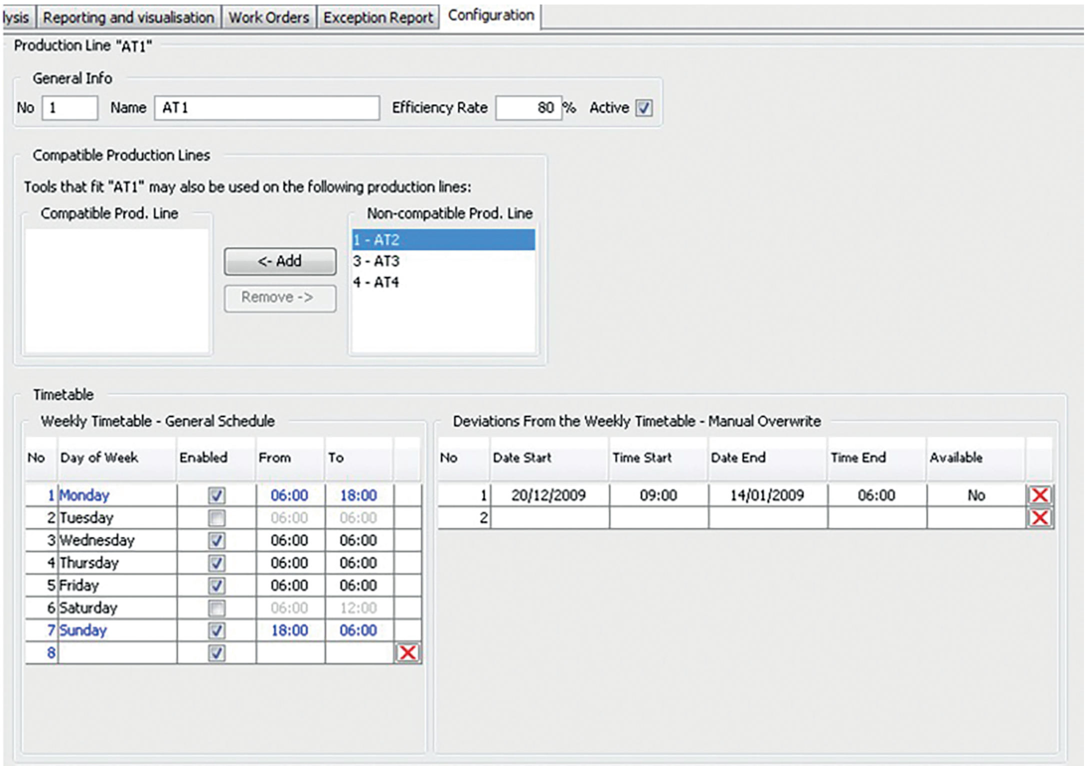
To generate a detailed schedule that was optimized (as well as realistic and executable), the Decision Optimization System held a variety of foundry- and machine-specific data that was referenced by the optimization model, such as changeover times between various metal grades:



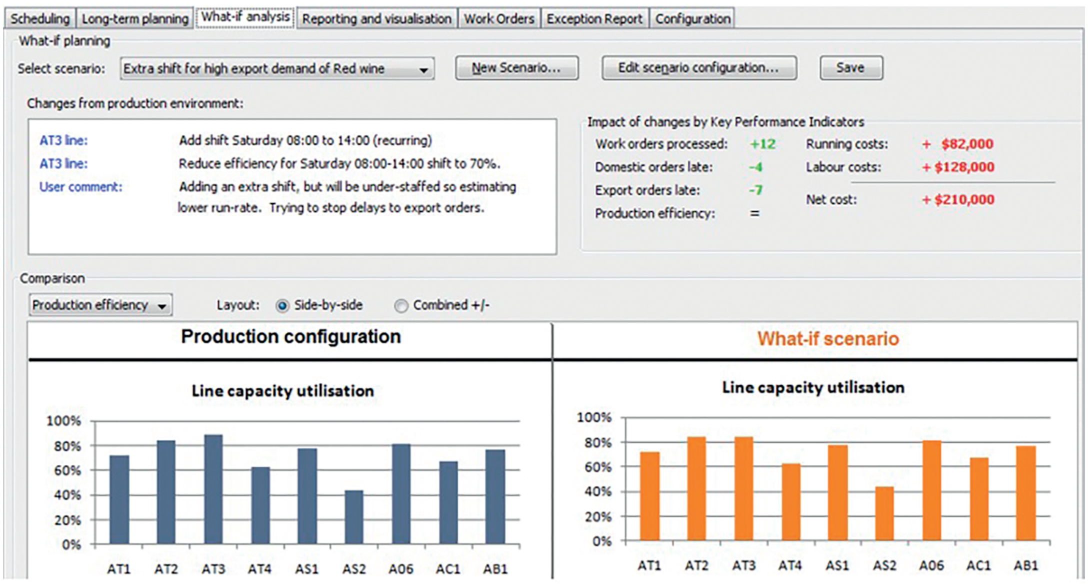
Within the optimization model itself, the approach for handling constraints was based on decoders, which separated between objectives and constraints (as discussed in Chapter 6.8). Using this approach, the optimization model used the constraints to "guide" the optimization process toward feasible schedules of higher quality. This constraint-handling approach also allowed for easy modification of business rules related to labor availability:



and machine availability:



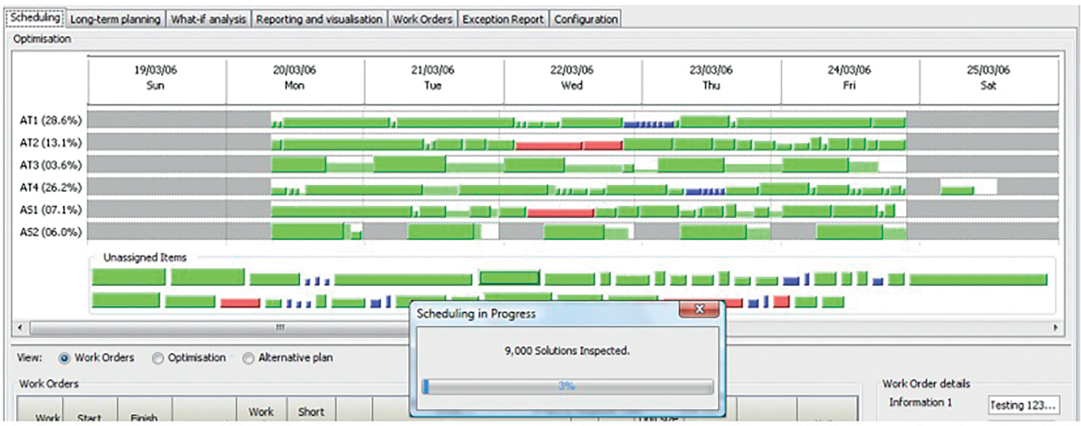
Both labor and machine capacities were treated as soft constraints, allowing the Decision Optimization System to flex production up and down as required. By modifying these constraints (or changing the capacity of the foundry in a more fundamental manner—for example, by adding another casting machine within the Decision Optimization System), CAST Metals could ask "what-if" questions and create alternate schedules, as shown below:



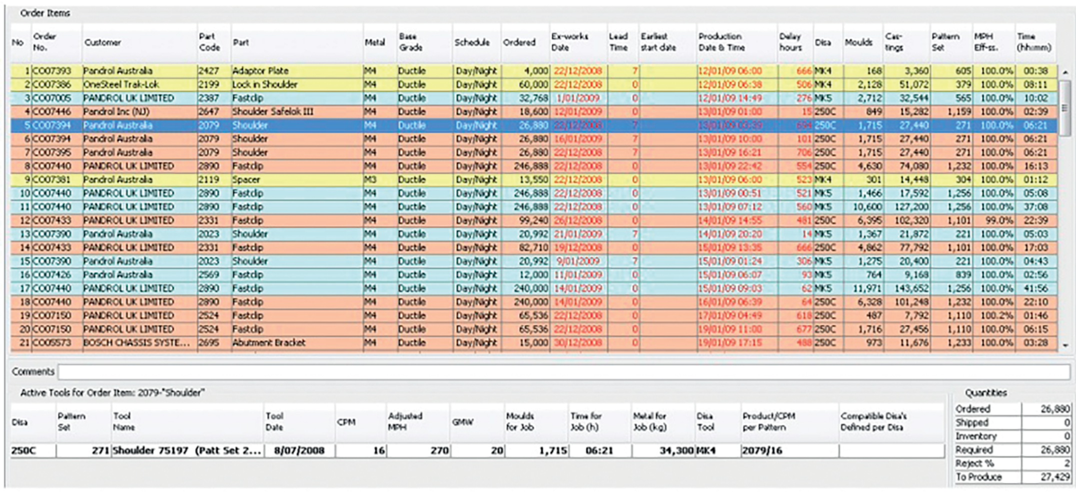
CAST Metals used this functionality to analyze a variety of "what-if" scenarios, including:

- Examining the effect of moving a production order forward or back, or from one machine to another
- Splitting large production orders into smaller work orders
- Examining the effect of constraining certain orders so they couldn't run in parallel
- Examining the effect of changes to the production calendar, furnaces, and production lines.

The most important output of the Decision Optimization System, however, was the production schedule itself. To generate feasible schedules right from the start of the optimization run, the system used a combination of evolutionary algorithms and simulated annealing (plus a decoder responsible for generating near-feasible solutions). Although the quality of the system-generated schedule improved as the optimization run progressed, CAST Metals could stop the run at any point and use the best available schedule rather than waiting to the end. This allowed for flexible usage and provided CAST Metals with ultimate control over the optimization process. The screen below shows an optimized schedule for a particular week, where each bar represents a production order:

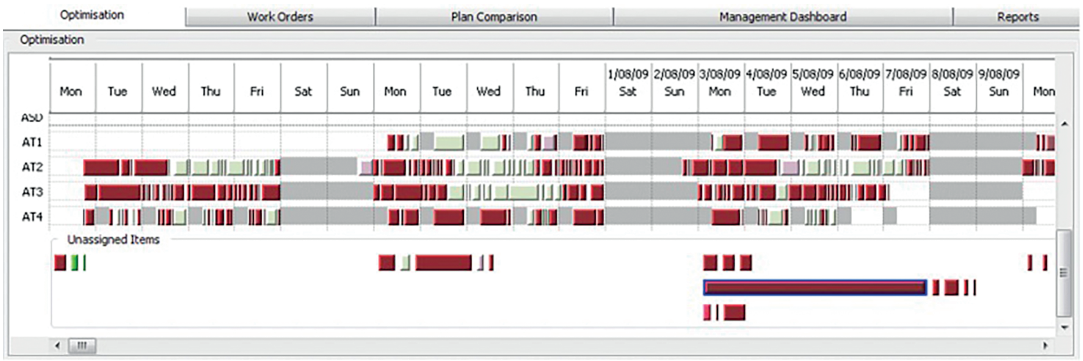


The details of each order could also be viewed by clicking on any bar, or by selecting the order from the items list:

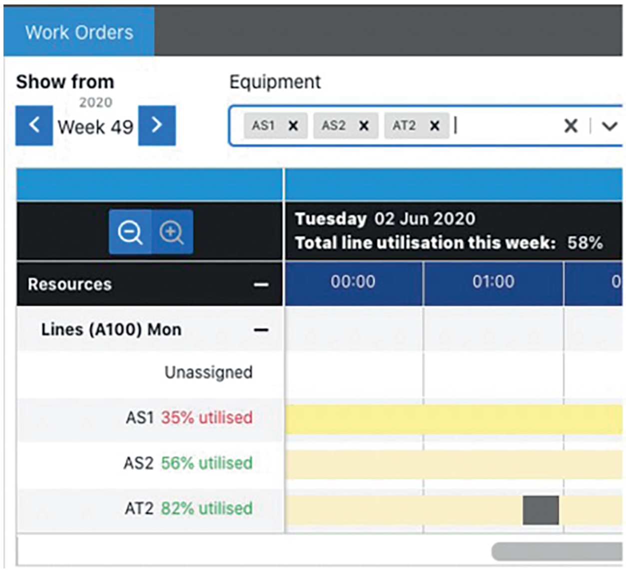


The Decision Optimization System also received a live data feed from each machine, allowing it to compare actual production against scheduled production in real time. Whenever the situation reached a point where the schedule was no longer feasible, the Decision Optimization System would flag that re-optimization was required based upon the current state of production in the foundry—in other words, that the current production schedule would no longer be met, and that re-optimization was required to re-align future production with the current reality on the factory floor.

This streaming machine data allowed for a real-time view into each foundry, providing CAST Metals with not only a "scheduled" versus "actual" perspective, but also an ability to dynamically re-optimize production whenever the unexpected occurred (which unfortunately was often). During this re-optimization process, if the Decision Optimization System ran out of capacity in the foundry to process all orders with hard due dates, it would flag these orders as "unassigned items," as shown below:



The Decision Optimization System also provided additional reporting on various utilization ratios, throughputs, and other KPIs in both graphical and numerical form. Such reporting was also displayed within the scheduling system itself, as shown below, allowing CAST Metals to evaluate the performance of each production schedule:



From the very start of the project, CAST Metals had a clear view of the KPIs it wanted to improve, which provided a baseline and benchmark for validating the performance of the Decision Optimization System. Also, given the scale of CAST Metals' manufacturing footprint, management knew that any improvement in these metrics would translate to a direct and significant improvement in financial performance of the entire business. The realization of these benefits, however, was dependent on CAST Metals successfully navigating two change management challenges:

- First, the spreadsheets that CAST Metals had built up over the years had to be replaced by the Decision Optimization System, which was a challenge in itself within each foundry. "But I've been using that spreadsheet for years," the production schedulers would complain. "No system can capture everything I've put into that spreadsheet!"
- And secondly, during the user acceptance testing ("UAT") phase of the project, CAST Metals encountered further resistance from end users because the Decision Optimization System was recommending schedules that "didn't look right."

On this second point, CAST Metal realized that "optimization projects" were very different from "automation projects," and thus required more significant change management. If CAST Metals had configured the Decision Optimization System to exactly replicate what end users did and generate schedules that "looked right" to everyone, then the only value of such a system would have been the time saved in generating these schedules. This would have become an automation project, because CAST Metals would be automating the scheduling process with the end result being exactly the same (only faster). This wasn't the outcome CAST Metals was seeking, so the Decision Optimization System wasn't configured to replicate what end users did, but rather, to generate optimized schedules that could improve various KPIs (and so by definition, these schedules had to be different to those being generated by end users up to that point). The only way the Decision Optimization System could create value was by recommending a *different* decision that led to a *different* result—in the case of CAST Metals, a different production schedule that led to improved asset utilization, throughput, and customer service levels). And because the system was recommending something different to what had been typically done in the past, change management was more challenging—"Hey! That doesn't look right to me," the production schedulers would say. "I would have done it differently."

The first change management challenge was addressed through extensive user training on the new Decision Optimization System, whereas the second challenge was addressed by educating end users on why the system was making certain recommendations. In addition to this education, the Decision Optimization System provided an explanation in natural language as to why a certain schedule or scheduling decision was optimal (i.e. "explainable AI," as discussed in Chapter 6.9). CAST Metals was able to successfully navigate these change management issues in large part because of strong executive sponsorship and leadership (which Chapter 11.3 explores in greater detail). Once the system was fully adopted, CAST Metals began executing the new schedules and realized an immediate improvement in manufacturing performance. Each foundry experienced a jump in DIFOT and asset utilization metrics, as well as reduced overtime labor requirements. The improvement within each foundry varied according to the capability of the production scheduler and the sophistication of their spreadsheets. In other words, the Decision Optimization System outperformed very capable staff with very sophisticated spreadsheets, but the outperformance was modest; as for average staff with basic spreadsheets, the improvement was pronounced.

And lastly, not only did CAST Metals know what was happening within each foundry in real time, but the company could now dynamically re-optimize and re-align the forward schedule with the production realities at each site, thereby running a continuously optimal manufacturing process.

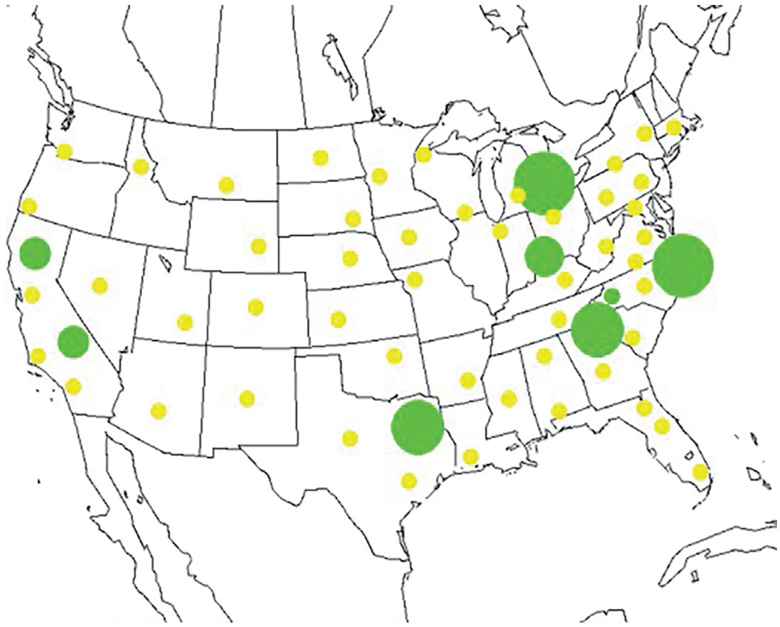
10.3 Logistics and Distribution Optimization

Logistics is the "connector" of a supply "chain," involving modes of transport (such as trucks, trains, and ships), as well as storage locations. Many logistics and distribution problems are "multi-nodal," in the sense that one truck needs to make many deliveries or stops (like the traveling salesperson problem in Chapter 2.1, which is representative of typical routing or journey planning problems). In multi-nodal problems, the optimization objective is to find the route that minimizes travel time and other cost metrics, while satisfying a number of hard and soft constraints (such a delivery times or slots). From that perspective, such problems are "one dimensional" and rarely encompass any prediction component other than a demand forecast used for load planning purposes.

Rather than concentrating on a standard logistics operation, this case study will explore a node-to-node distribution problem where the complexity arises not from the optimization challenge of finding the best route, but from the number of factors that impact the distribution plan (and which need to be considered during the optimization process, such as price changes, inventory levels, seasonality, and more), and the significant prediction problem that underpins the entire optimization result.

With this in mind, the case-study presented in this section is about GMAC, a car financing organization in the United States that leases around one million cars each year to consumers, organizations, and rental agencies.[5] When a car lease agreement expires—which could be from one to five years—the car is either returned to GMAC or purchased by the leasee (in either case, these cars are called *off-lease cars*). GMAC doesn't need to worry about the purchased off-lease cars, but it needs to sell the returned off-lease cars at one of many auction sites located across the United States. Each of these returned cars is different in its make, model, body style, trim, color, year, mileage, and damage level, and the overall number of cars leased each year translates into approximately 5,000 returned off-lease cars each day. The following figure illustrates a particular day, where green circles represent the returned off-lease cars and yellow circles represent the 50 auction sites at which GMAC sells its cars:

5 This case study is also covered in the article by Michalewicz, Z., Schmidt, M., Michalewicz, M., and Chiriac, C., called *A Decision-Support System based on Computational Intelligence: A Case Study*, IEEE Intelligent Systems, Vol. 20, No. 4, July–August 2005, pp. 44–9, which can be downloaded from: https://www.complexica.com/hubfs/case%20studies/Case_Study_An_Intelligent_Decision_Support_System.pdf.

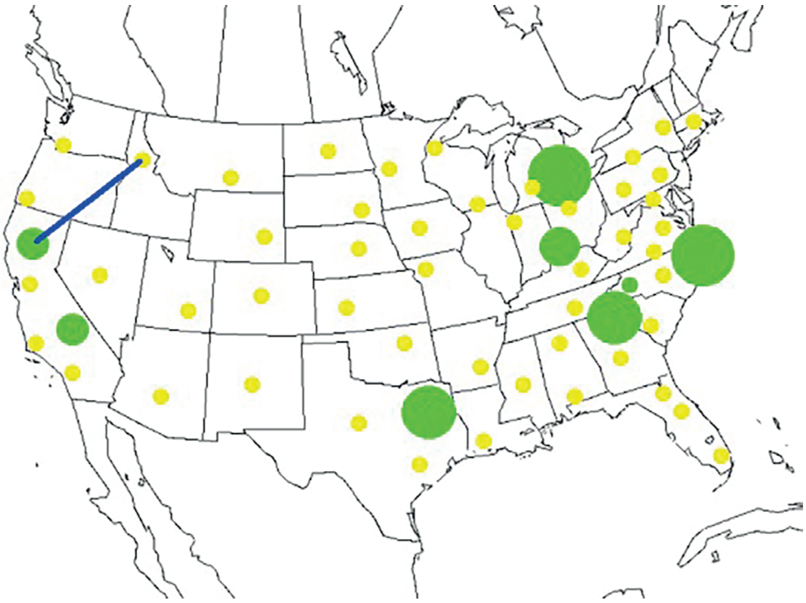


The larger the green circle, the more cars were returned at that particular location, with the sizes and locations of these circles varying from one day to the next (as different people and organizations return their cars at different locations). The yellow circles, on the other hand, represent the designated 50 auction sites where the returned off-lease cars are sold. The locations of these auction sites are fixed.[6]

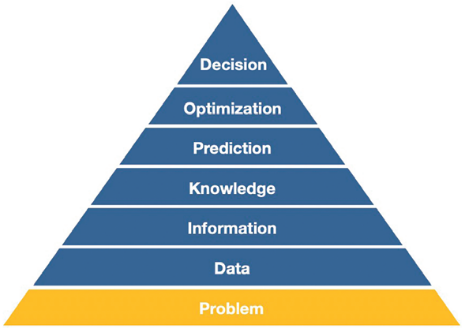
GMAC's task was to distribute the daily intake of approximately 5,000 cars to the 50 designated auction sites; in other words, to assign an auction site to each particular off-lease car. For example, if the first car is located at a dealership in Northern California, GMAC would consult some reports[7] on what the average sale price for that particular car is at each auction site (after adjusting for mileage, trim, damage level, etc.), and then ship the car to the auction site with the highest average sale price. Of course, GMAC also needed to estimate the transportation cost to each auction site (the longer the distance, the higher the cost, and longer transportation times resulted in higher depreciation costs and risks). Using this method, GMAC's decision for the first car could be visualized in the following way:

6 Although the locations of the 50 auction sites are fixed, GMAC may, from time to time, change the auctions it does business with by dropping some sites and adding new ones (thereby changing the location of the 50 yellow circles). This may happen if cars are routinely damaged at some sites, auction fees go up, or some other reason. However, these decisions raise several additional questions, such as: *How do we evaluate the monetary impact of dropping some sites and adding others*? and *Can we increase profits by replacing some auction sites with others*? We will address these important questions later in this section.

7 Many reports are available for estimating the auction price of cars, including *Black Book*, *Kelley Blue Book*, the *Manheim Market Report*, and others.



with the blue line representing the decision to ship the car from Northern California (green circle) to an auction site in Idaho (yellow circle). GMAC would then repeat this process for each car. Although straightforward, this approach for distributing off-lease cars didn't work very well, and led to a situation when GMAC didn't capture the full value of each off-lease car. Because the entire process was based on manual analysis and individual, car-by-car decisions, any small mistake that resulted in a net reduction of "only" $50 per car, would cost GMAC $250,000 in a single day!



As such, GMAC defined their business problem and objective as:

Maximize the aggregate resale value of all returned off-lease cars by optimizing the logistics and distribution to individual auction sites

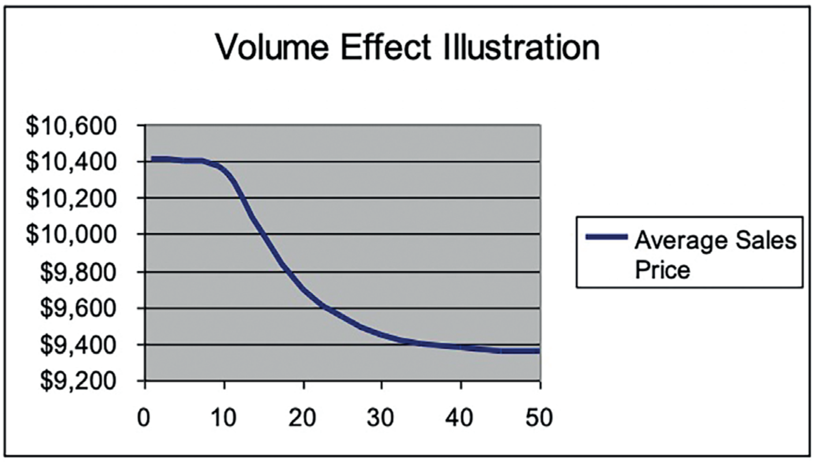
This was a difficult business problem to solve and objective to realize, because of the following reasons:[8]

1. *Number of possible solutions.* There were 50 possible solutions for each individual car, as GMAC can ship a car to any of the 50 auction sites; for two cars, there were 2,500 possible solutions (50 × 50); for three cars, 125,000 possible solutions (50 × 50 × 50), and so on. For 5,000

8 Recall our overview of complex business problems in Chapter 2, where we discussed the astronomical number of possible solutions, dynamic environments, and problem-specific constraints—all of which are present in this problem.

cars, however, there were approximately 50^{5000} possible solutions (50 multiplied by itself 5,000 times)! This was an overwhelming number (1 followed by 8,494 zeros) and no supercomputer could evaluate all these combinations in a billion human lifetimes. Nevertheless, GMAC had to make daily decisions for these cars, irrespective of how complex the problem was or the number of possible solutions.

2. *Transportation costs.* When GMAC shipped an entire truckload of cars from one location to another, it would realize a better price per car than when it shipped only one car (or a few cars), thereby lowering the overall logistics cost. This occurred because the cost of transport was primarily tied to individual trucks and drivers, with the number of cars on each truck being of secondary importance. Hence, the relationship between transportation cost and number of transported cars looked similar to the model presented towards the end of Chapter 6.1. Given this model, the cost for sending a single car from one location to another was $250, but the cost of sending two cars was $300 (reducing the cost per individual car to $150), with each additional car being $50. If a truck could hold 10 cars, then the transportation cost of a fully loaded truck was $700, or just $70 per car. But if GMAC needed to transport 11 cars, then a "jump" occurred in cost with $700 for the 10 cars on the first truck, and $250 for the single car on the second truck (for a total of $950).
3. *Volume effect.* Although GMAC wanted to send each car to the auction site where the highest price could be realized, sending too many cars of same color, make, and mileage to the same auction site would trigger the volume effect. For example, if GMAC sent 45 white Chevrolet Camaros to the same auction site (which might have all been returned from a rental agency on the same day), then these cars were likely to sell for the minimum opening price, because with 45 identical cars for sale, there wouldn't be enough buyers to bid the price up on each car (meaning there was a limit to how much supply could be absorbed by each site). On the other hand, if GMAC sent only five Chevrolet Camaros to the same auction site, then these five cars would fetch a higher price because the same number of buyers would be bidding on a smaller number of cars. To illustrate this point, the volume effect for a particular car at a particular auction site might be:



This graph illustrates the volume effect phenomenon, where GMAC could realize more money per car by selling *fewer similar cars*. In this example, the current average sale price for a particular car at a particular auction site might be approximately $10,400, and GMAC could realize this price by shipping up to seven cars to this location. However, if GMAC shipped 30 similar cars, then the average sale price per car would drop to $9,450. Note that the term "similar" could mean more than just the same make, model, or color. For example, many white compact cars of different makes and models often competed for the same buyers, thereby reducing the average sale price per car. Consequently, due to the volume effect, it wasn't effective for GMAC to consider one car at a time.

4. *Price depreciation and inventory holding costs*. To further complicate matters, every auction site had a set day for selling cars (e.g. every second Friday at 10 am). Because of this, if GMAC shipped 100 cars to an auction site and the delivery arrived one or two days *after* the auction day, then these cars would sit until the next sale day, incurring depreciation and holding costs. Because of this, GMAC needed to check the exact sale day and inventory levels across all 50 auction sites before making any new distribution decisions.
5. *Price changes*. Used car prices change over time, and these changes may be slow and subtle (over many years as consumer preferences change), sudden and dramatic (as was the case in March 2020 when the COVID-19 panic set in), or region specific (e.g. convertible cars become unpopular in northern states during the winter months, and consequently, they fetch a lower price—which is part of the "seasonality effect"). GMAC also had to deal with next year's models entering the market during August and September, causing older models to drop sharply in price (also part of the seasonality effect). During this time of year it was better to ship cars nearby and sell them quickly, rather

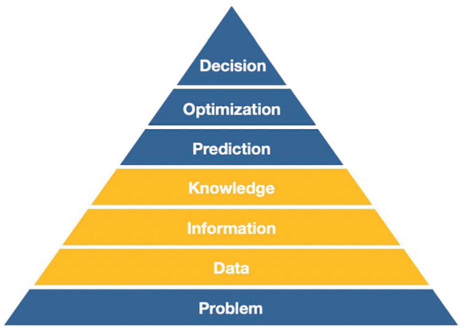
than shipping them longer distances to more lucrative auction sites. Additionally, new body style models are introduced every few years, causing an even bigger drop in price for the older body style.

Coming up with the daily decision of where to send the returned off-lease cars wasn't easy, as the decision needed to consider the above factors.

Furthermore, the process of transporting a car to a specific auction site could take up to two weeks, as the truck would have to drive to the pick-up location, load the car, pick up some additional cars (possibly somewhere close by), and then finally deliver the cars to the designated auction. Because of this, GMAC had to consider the sale price for each car a couple of weeks ahead of time. For example, for a car located in Jacksonville, Florida, GMAC might consider sending this car to an auction site in Georgia, Pennsylvania, or California. The price prediction for these three auction sites would be different, because GMAC would be predicting the sale price five days into the future for the Georgia auction site, ten days into the future for the Pennsylvania auction site, and fifteen days into the future for the California auction site. The differences in time were due to the transportation distance. However, to predict these prices, GMAC needed to consider the seasonality effect, price depreciation, volume effect, and inventory levels. In making the decision of Georgia vs. Pennsylvania vs. California, GMAC would also need to weigh the possibility of a better price in California against the higher transportation cost, higher depreciation, and higher overall risk.

These challenges were ideally suited for AI-based algorithms and the implementation of a Decision Optimization System, which would rely on advanced prediction, optimization, and self-learning capabilities to improve GMAC distribution decisions.

Data, information, & knowledge



GMAC maintained a historical collection of transactional sales data that could be visualized as a two-dimensional table representing off-lease cars sold at auction. One dimension of the table represented the number of records (cars), and the other dimension represented the characteristics of each car (e.g. VIN,[9] make, model, mileage, etc.):

9 VIN is an acronym for "Vehicle Identification Number," which is a string of 17 digits and letters that contains considerable information about a specific vehicle, (including country of origin, manufacturer, and model year).

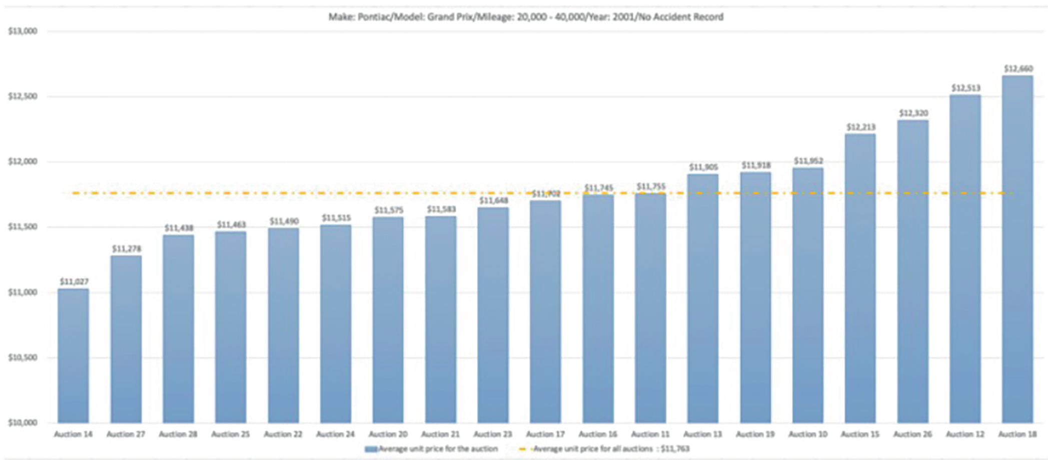
VIN	Type	Make	Model	Miles	Year	Color	Transmission	Body/Doors	Damage
2G1FP22P1P2100001	Rental	Chevy	S-10	34,983	2002	Silver	Manual	2D	$0
WB3PF43X8X9000331	Lease	Chevy	Cavalier	59,402	2001	Red	Automatic	2D Coupe	$0
4BBG38FJF04JDK000	Lease	Chrysler	Sebring	74,039	2000	Gray	Automatic	2D Coupe	$500
DJOW03FFU990SJ206	Lease	Ford	Escape	37,984	2001	Green	Manual	4D Sport	$250
JD8320DJ2094GK2X3	Rental	Ford	Focus	30,842	2001	Green	Manual	4D Sedan	$0
2JE9F0284JD0213M3	Lease	Isuzu	Rodeo	59,044	1999	White	Automatic	4D Sport	$250
4380JDDD9W02MD001	Rental	Jeep	Cherokee	48,954	2000	Black	Automatic	4D Sport	$500
490DK20285JF0209D	Rental	Mazda	626	38,943	2000	White	Automatic	4D Sedan	$0
10D92JD920KD00002	Lease	Nissan	Altima	39,488	2000	Black	Automatic	4D Sedan	$0
D920DKJ0284JJ9990	Rental	Nissan	Altima	23,584	1999	White	Manual	4D Sedan	$0
JD88D92JJD02K3361	Rental	Saturn	L	21,048	2001	White	Automatic	4D Sedan	$750
10DS0JJ20DXI00093	Lease	Suzuki	Vitara	15,849	2003	Yellow	Automatic	2D Sport	$0
21KD02KD0DJ920M27	Lease	BMW	Z3	49,858	2000	Blue	Manual	2.3 RSTR	$250
389DJ2DD298JWQ082	Lease	Ford	Explorer	42,893	2002	Green	Automatic	XLT 4WD	$0
108DJ2048FJJ20043	Rental	Ford	Mustang	20,384	2002	Red	Manual	GT	$0
DJC82002009DD2J04	Rental	Mercury	Frontier	27,849	2001	Silver	Automatic	SE-V6 Crew	$500
830DMM3029XMW0092	Lease	Honda	Accord	26,849	2002	Yellow	Automatic	EX V6	$0
CNEU200220CCI2202	Rental	Toyota	4Runner	33,483	2000	Silver	Automatic	SR5	$0
CNDJ2940JD88D2JD0	Lease	VW	Beetle	5,459	2003	Blue	Manual	GLS 1.8T	$0
1VC0CMEJ200V9EJJ1	Lease	Toyota	4Runner	81,837	2001	Silver	Automatic	LMTD 4WD	$250

This historical sales data contained the VIN, postal code of the auction site (ZIP), transaction date, and the sale price of each car:[10]

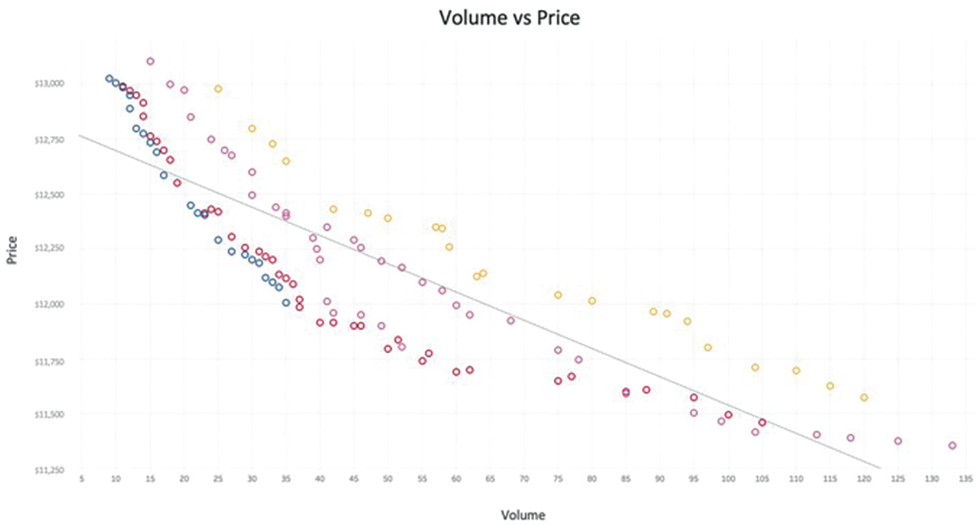
VIN	ZIP	Date	Price
39WWK93309KJ33012	28262	2.11.2004	$12,035
UDJ2293M99DL0K220	30334	2.11.2004	$15,600
4D09WJD92JE93H990	30334	2.11.2004	$10,590
KD37D92JF83NF8822	90012	3.11.2004	$9,265
NKI2389DD974F2235	28262	3.11.2004	$13,450
K29DH38FHW02HD923	48243	3.11.2004	$13,955
MDK293HFDWH299305	90012	4.11.2004	$12,495
28DN39FNDJW2N0024	90012	4.11.2004	$11,925
29H93NFI3HJF93F04	48243	4.11.2004	$11,396
ND920ENF1NAD02834	48243	5.11.2004	$9,835
D39DJ39EHQ8HH9335	28262	5.11.2004	$8,965
02UFIMF03JF9SH935	90012	5.11.2004	$13,960
D932NF93HG9057362	48243	5.11.2004	$8,830
00F8EB3IDNB293758	48243	8.11.2004	$7,920
IE038THJ203TH0234	28262	8.11.2004	$19,250
39FH324MV092HGM39	48243	8.11.2004	$22,640
F92N9F389FH120458	90012	8.11.2004	$13,580
F9485JG03H25495J5	30334	9.11.2004	$16,970
08GN94HJH03J49327	30334	9.11.2004	$14,320
F04JH402KG4509G45	48243	9.11.2004	$9,110

GMAC also possessed data for individual auction sites (e.g. the average number of participating auctioneers during different times of the year) and external data such as historical weather conditions at different auction sites during different sale days, historical petrol prices, color preferences in different areas of the United States, and so on. GMAC used this data to generate a variety of reports for the price difference between auction sites for the same off-lease car on attributes such as color, volume of cars sold at each auction site, number of auctioneers at different seasons at different auction sites, and so on. The following graph illustrates one such report: The price difference between auction sites for one particular make (Pontiac) and model (Grand Prix) with an odometer reading between 20,000 and 40,000 miles:

10 We could easily obtain the characteristics of each car by merging it with the previous table.

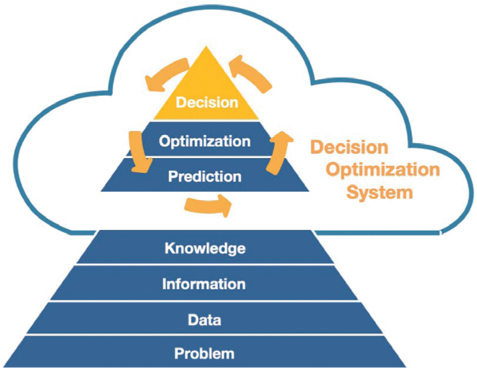


GMAC also studied the volume effect at different auction sites for various types of cars, and this knowledge was presented via graphs and other reports, like the one shown below (which shows the sale price of a Pontiac Grand Prix sold with a number of similar cars at the same auction site, with the different colors representing different odometer ranges—in this example, yellow circles correspond to the lowest odometer range of 0 to 10,000 miles:



Nonetheless, all this data, information, and knowledge were of limited assistance in helping GMAC make the best daily distribution decision, because even if GMAC had "perfect knowledge" and could accurately predict the price of *any* car at *any* auction site for *any* day, they still wouldn't know how to optimally distribute 5,000 cars because of all the complexities of this problem, such as logistics, price depreciation, inventory levels, volume effect, and so on. The number of possible distributions was simply too large to be evaluated in any reasonable amount of time, which drove the business case for an AI-based Decision Optimization System capable of increasing the aggregate resale value of returned off-lease cars.

Decision Optimization System (prediction, optimization, & self-learning)



For this particular logistics and distribution problem, the Decision Optimization System had to consider the characteristics of each car, characteristics of each auction site, transportation costs, volume effects, countrywide inventory (as well as cars in transit to various auction sites), price depreciation curves, and market-driven changes in price.

Before the predictive model was built, the data went through a data preparation process that included variable transformation and variable composition, data reduction and normalization, and the generation of missing values (as discussed in Chapter 4.2; also, for more information on this process, please watch the supplementary video at: www.Complexica.com/RiseofAI/Chapter4). GMAC also augmented its internal data with Black Book data (which provided regional sale prices, with each region containing several states and more than a dozen auction sites) and the Manheim Market Report (which reported on the average sale price of all cars sold at auctions owned by Manheim). The resulting prediction model was an ensemble based on decision trees (as discussed in Chapter 5.1) that generated sale price predictions in the following sequence of steps:

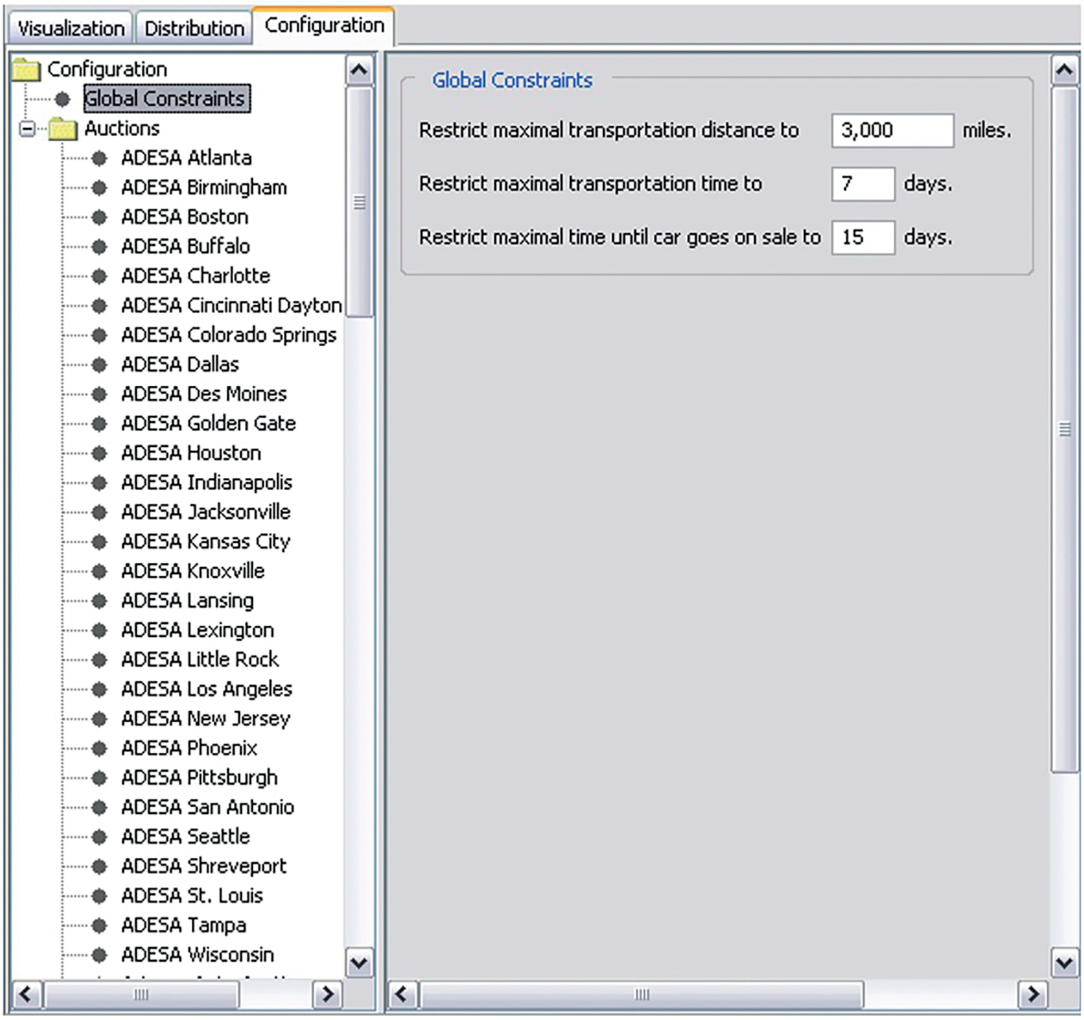
1. *Base price*. A predicted "base price" was generated based on the car's make, model, body style, and year.
2. *ZIP-based make/model adjustment*. Because some makes/models sold for a premium or discount in certain regions, the prediction model adjusted the base price for these specific makes/models in certain regions (e.g. Chevrolet Corvettes might sell for a $300 premium in Florida and California, and a $600 discount in Montana and Idaho).
3. *Car group/color adjustment*. Because some car groups/colors sold for a premium or discount irrespective of the region, the prediction model adjusted the base price for these specific car groups and colors (e.g. yellow Chevrolet Corvettes might sell for a $500 premium, while green ones for a $1,000 discount).
4. *Mileage adjustment*. The prediction model adjusted the base price for mileage and *model-year-age*, which was the age of a car according to its model year (i.e. when the 2005 Chevrolet Corvette became available in August 2004—which underwent a complete body style change—the

model-year-age of the 2004 Chevrolet Corvette became 1, as that model year was only one year old).

5. *Depreciation adjustment*. The prediction model adjusted the base price for daily depreciation, as calculated from the car's return date to its predicted sale day. Because the daily depreciation rate was higher in the summer months (preceding the introduction of new models), the depreciation rate increased from June onwards, reached its highest value in August, and then decreased to lower than average values for October, November, and December.
6. *Seasonality adjustment*. Because some makes/models sold for a premium or discount in certain regions at different times of the year, the prediction model adjusted the base price for these specific makes/models during certain seasons (e.g. convertible Chevrolet Corvettes may sell for a $1,800 discount in the northern states during the winter months).
7. *UVC adjustment*. The Universal Vehicle Code (UVC) component provided a more detailed car specification than the VIN, and in cases where the UVC was available, the prediction model adjusted the base price for additional options (e.g. the UVC might reveal that a specific Chevrolet Camaro is equipped with an upgraded suspension package).

For an average daily intake of off-lease cars, the ensemble model would predict each car's final auction price. However, if GMAC received a large number of similar cars on a particular day, then the predicted auction prices for these cars were adjusted further to account for the volume effect. For more information on the predictive model used by GMAC for this particular distribution problem, please watch the supplementary video at: www.Complexica.com/RiseofAI/Chapter5.

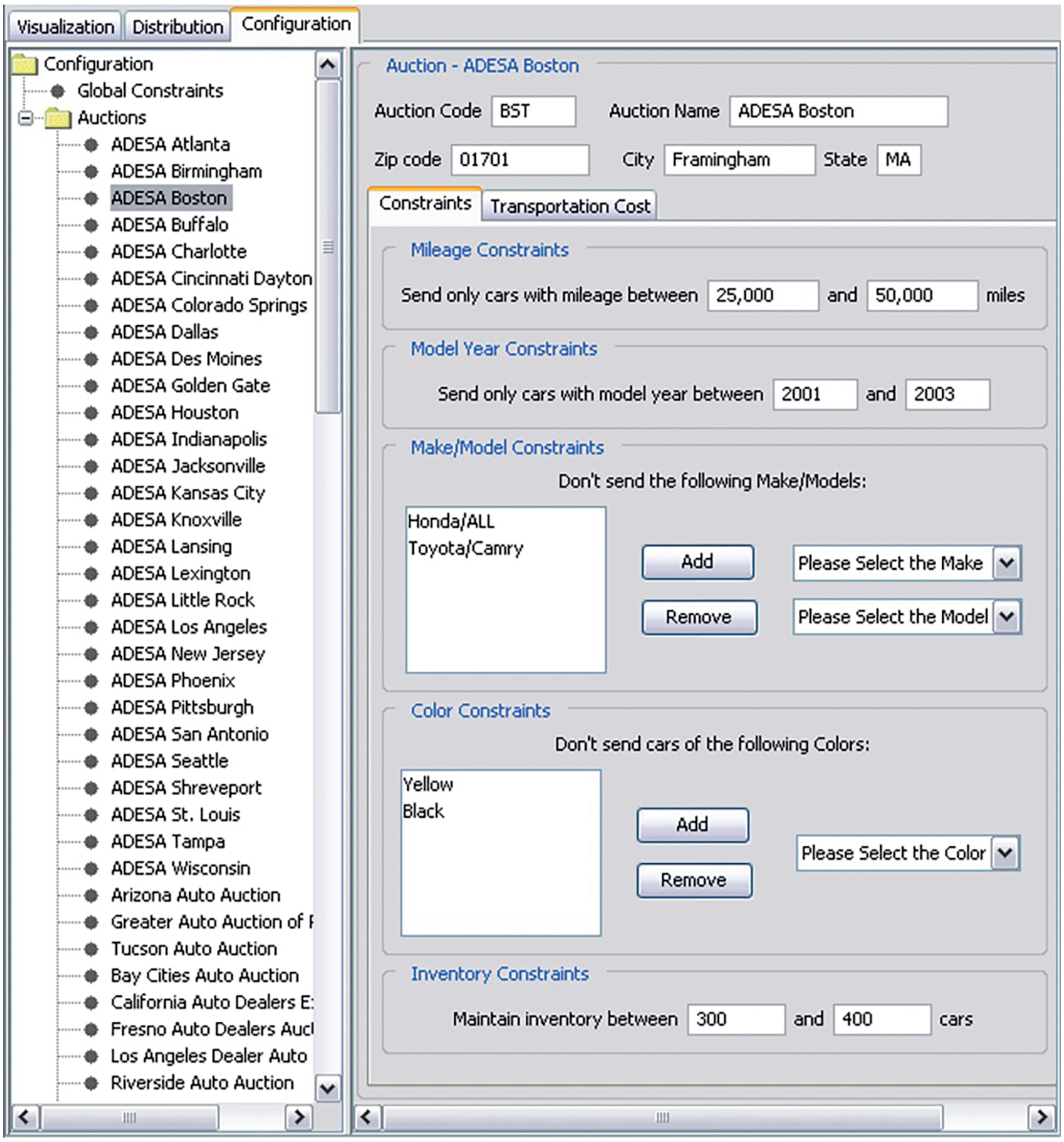
The Decision Optimization System also provided GMAC with the ability to add, modify, or delete various constraints and business rules. Constraints that were applied to all auction sites were regarded as global constraints, and an example of this was the "maximal transportation distance" constraint which limited the transportation distance of all cars—as shown in the screen below:



GMAC could also implement a large variety of local, auction-specific constraints within the Decision Optimization System, such as:

- *Mileage constraints*: which defined the upper and lower mileage of cars that could be shipped to a specific auction site. An example of this constraint would be "only ship cars that have between 30,000 and 70,000 miles to the ADESA Atlanta auction site."
- *Model year constraints*: which specified a range of model years that could be sent to a specific auction site. For example, GMAC could specify that a particular auction site could only accept model years between 2002 and 2004.
- *Make/model exclusion constraints*: which specified certain makes/models that were to be excluded from specific auction sites.
- *Color exclusion constraint*s: which specified certain colors that were to be excluded from specific auction sites.
- *Inventory constraints*: which specified the desired inventory level at each auction site. For example, GMAC could specify an inventory level between 600 and 800 cars for an auction site at any particular time.

The screen below shows the local constraints set for the “ADESA Boston” auction site:



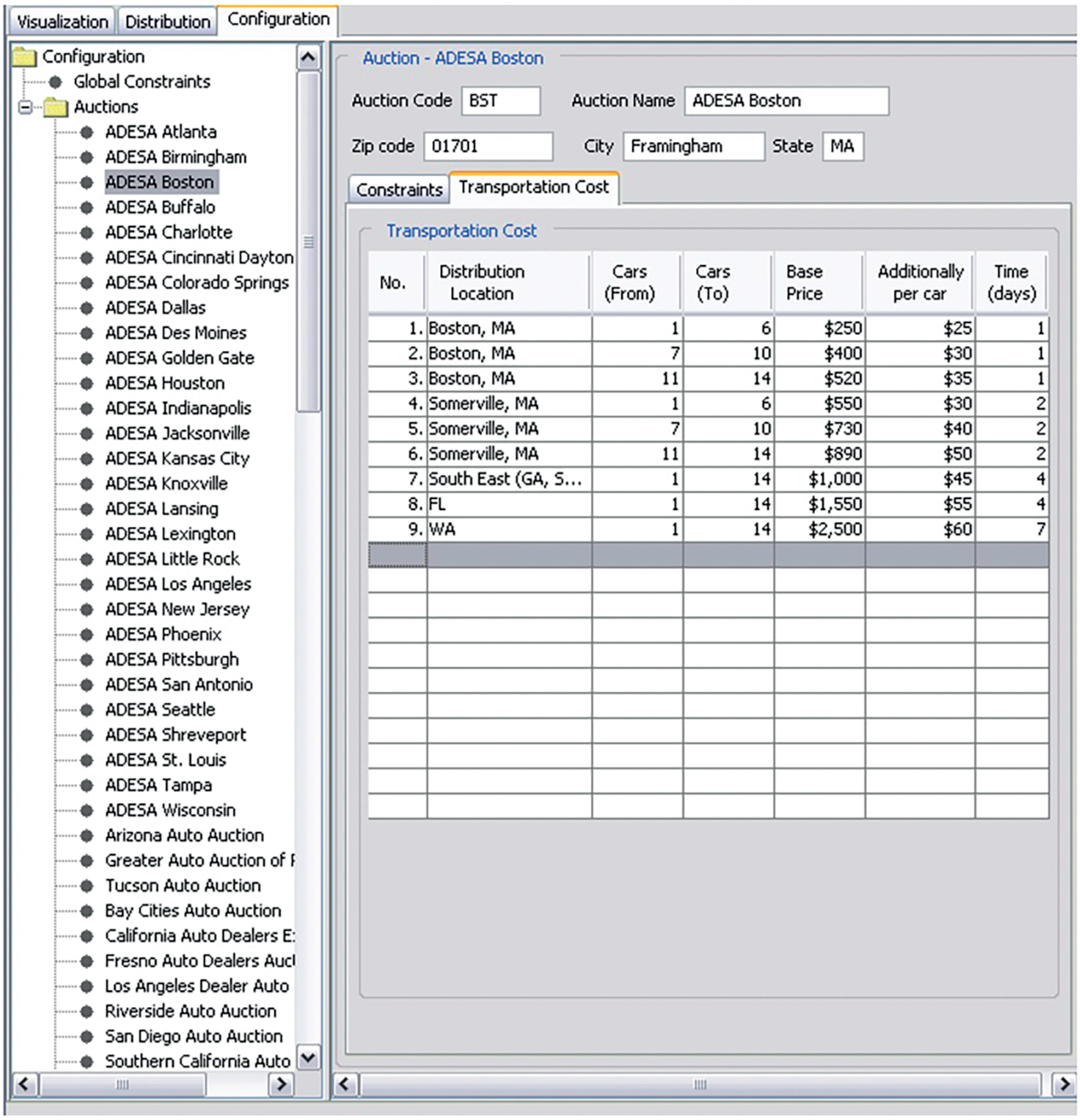
Each auction site could have different constraint settings, which represented the business rules that GMAC wanted to operate under for that particular site. For the ADESA Boston auction site, the constraints represented the following business rules (as shown above):

- “Send only cars with 25,000 to 50,000 miles”
- “Send only 2001, 2002, or 2003-year models”
- “Do not send any Honda or Toyota Camry cars”
- “Do not send any yellow or black cars”
- “Keep the inventory between 300 and 400 cars”

Except for the inventory constraint, all these constraints were defined as hard constraints. If the Decision Optimization System had to break a hard constraint, it would mark this recommendation with the notation "constraint violation." Inventory constraints, on the other hand, were defined as "soft" constraints and a penalty was assigned to solutions that violated these constraints. The penalty for violating a soft constraint would grow exponentially, and so instances where this constraint was violated in a significant way were rare. However, if the Decision Optimization System had to process a very large number of cars on a single day, then the inventory constraint might have been violated at almost every auction site. In such cases, the exponential penalty function would make these violations uniform. For example, in a case where all auction sites have a maximum inventory constraint of 300 cars but the current number of cars to be distributed would increase this inventory level to an average of 400 cars per auction, then the penalty for violating this soft constraint would be evenly distributed across all sites (so that they have the same degree of violation).

These constraints allowed GMAC to set various business rules (e.g. "do not send any red cars to Florida") within the Decision Optimization System, and so the configuration screen served as a link between GMAC and the system. GMAC could also use this configuration screen to investigate various "what-if" scenarios, such as "what would be the distribution of cars if we set the maximum transportation limit to 500 miles?" Because 300 auction sites were configured in the system and only 50 of them were "active," GMAC could activate or deactivate any auction site, and then re-run the optimization process to test a specific what-if scenario, such as "what would happen to the aggregate resale value of all cars if we used 60 auction sites instead of 50?"

GMAC could also use different what-if scenarios to investigate different transportation cost options available from different suppliers. The Decision Optimization System calculated the transportation cost from any distribution center to any auction site for any number of cars, and takes into account two factors that influenced this cost: (1) the distance between a distribution center and an auction site, and (2) the number of cars being shipped. The screen below shows the transportation costs for the ADESA Boston auction:

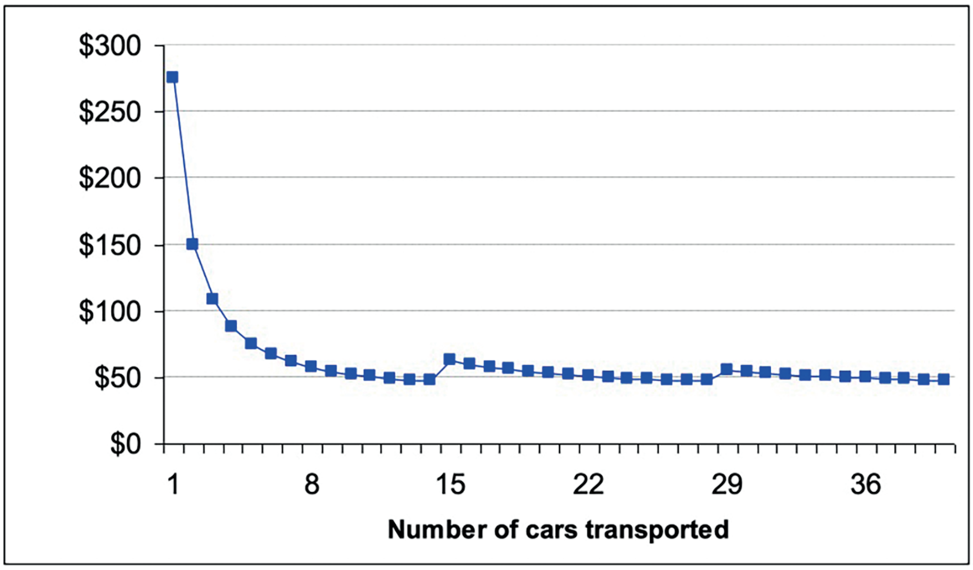


In this screen, the transportation cost is defined for cars sent to the ADESA Boston auction from five different locations.[11] The first two locations are defined by the cities Boston, MA and Somerville, MA; the third location is defined by a region containing the states Georgia, South Carolina, and North Carolina; while the fourth and fifth locations are defined by the states Florida and Washington, respectively. According to the transportation prices above, it would cost $250 to send a truck to Boston, MA, plus an additional $25 for each additional car. If GMAC wanted to ship six cars, then the transportation cost would be $400 ($250 + $25 × 6 = $400).[12] Also, row "No. 9"

11 The transportation cost was defined in terms of how much it would cost to ship a car (or group of cars) from a particular ZIP code, city, state, or region to the auction site.

12 If GMAC wanted to ship more than six cars, then the cost would be $400 for the first six cars ($250 plus $150 for six cars), plus $30 for each additional car. Hence, to ship 8 cars, the cost would be $400 for the first six cars, plus $60 for two additional cars, for a total of $460. Another price break occurred at the eleventh car, reducing the incremental cost per car to $35.

above defines the transportation cost between the ADESA Boston auction and the state of Washington. Because of the long distance (approximately 3,000 miles), it would cost $2,500 to ship a car to Washington, plus an additional $60 for each car on the same truck. Although the cost of shipping one car would be $2,560, the cost of shipping fourteen cars would be $3,340 ($2,500 + $60 × 14 = $3,340), or about $239 per car (which is ten times less!). As the following graph illustrates, the more cars transported from the same location, the smaller the transportation cost per car (in this particular case, cars that are transported from Boston, MA to the ADESA Boston auction):



In this graph, the average transportation cost per car decreases from $275 for one car to just $47 for fourteen cars. The graph also illustrates that the average transportation cost increases to about $62 when we need to transport fifteen cars (because an additional truck is needed for the extra car). After the fifteenth car, the average transportation cost goes down again, with smaller spikes when additional trucks are needed.

Besides these transportation costs, the Decision Optimization System also used inventory levels for each auction site to calculate several important parameters for the optimization process. One of these parameters was the volume effect, which was based on how many similar makes/models (or cars of the same color) were present at a specific auction site. Another important parameter was the anticipated sale date. If GMAC had 1,200 cars at a particular auction site (or in transit) and approximately 500 were sold during each auction session, then GMAC could assume that a car shipped today would be sold in the third auction session. Therefore, the Decision Optimization System needed to consider the additional depreciation and seasonality effect during this additional time. Once the system-generated distribution plan was approved, the auction inventory was updated with the new cars assigned to each auction. And lastly, the cars that had been recently sold at these auction sites were removed from inventory.[13]

The optimization model generated a variety of possible distribution plans that served as input to the prediction model. This input provided a destination assignment (i.e. auction site) for each off-lease car, which the prediction model used to generate a predicted sale price. The optimization model then summed all these predicted prices (i.e. the output data) to evaluate the quality of the distribution plan—the higher the sum of the predicted sale prices, the better the distribution plan. Hence, there was a strong relationship between the prediction and optimization models, as is the case within most Decision Optimization Systems.

The optimization model was comprised of several different AI-based algorithms that used different solution representations. For instance, evolutionary algorithms (see Chapter 6.7) used solutions based on indirect representation, where all available auction sites were sorted by *distance* from a particular car. In other words, auction 1 was the closest (distance-wise), auction 2 was the second closest, and so forth. Hence, each solution was represented by a vector of auction site indices (relative to a particular car), and the length of the vector was equal to the number of cars being distributed:

3	4	4	…	1	1

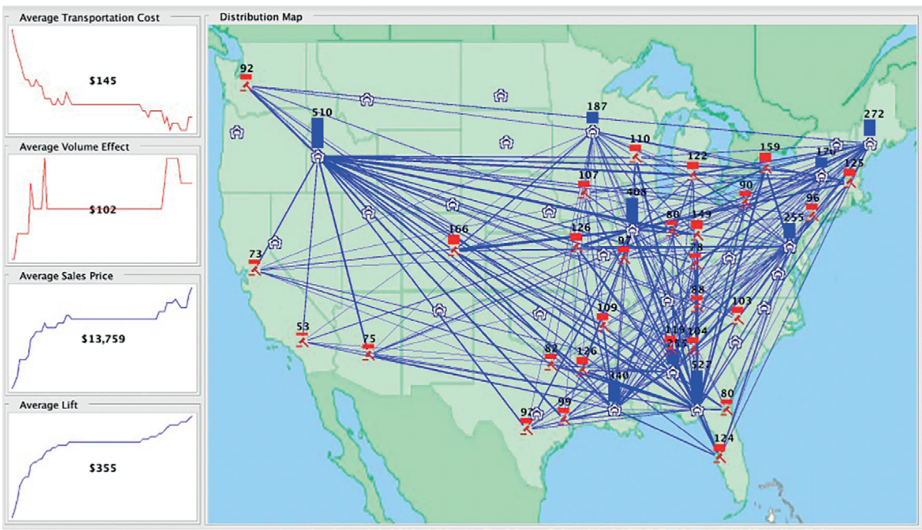
The vector above represents a solution where the first car is shipped to the third closest auction (for this particular car), the second car is shipped to the fourth closest auction (for this particular car), the third car is shipped also to the fourth closest auction (note, however, that the second and third car are most likely shipped to different auction sites, as the fourth closest auction for the second and third car need not be the same), and so on, with the last two cars being shipped to the closest auction sites. In this particular implementation of evolutionary algorithms, the optimization model applied the elitist strategy, which forced the best solution from one generation to the next, as well as various mutation and crossover operators that were discovered through experimentation. For additional information on the optimization model used within the GMAC Decision Optimization System, please watch the supplementary video at: www.Complexica.com/RiseofAI/Chapter6.

To enable learning within the Decision Optimization System, both the prediction and optimization models updated themselves with the arrival of

13 Data about sold cars was also used to tune the prediction model (explained later in this section).

new data. The prediction model contained numerous parameters (different values for various adjustments) that were automatically updated to capture changing trends in the used car marketplace at regular intervals (as discussed in Chapter 7.2), and in terms of optimization, each day brought a different "instance" of the same problem, as changes occurred in the number and type of cars to be distributed. For this reason, the optimization model was based on several optimization algorithms where each algorithm contained a few parameters that were adapted (as discussed in Chapter 7.3), and the usage of several optimization algorithms together generated a result that was better than the result of any single algorithm. For additional information on the learning components of this case study, please watch the supplementary video at: www.Complexica.com/RiseofAI/Chapter7.

The graphical user interface of the Decision Optimization System allowed GMAC to add, modify, or delete various constraints and business rules (as discussed earlier), as well as "visualize" the distribution plan. In the screen below, there are icons for each distribution center and each auction site, and four performance graphs. The white "horseshoe" icons represent distribution centers where off-lease cars are collected, cleaned, and conditioned for eventual sale at an auction site.[14] The red "hammer" icons represent auction sites, and the lines between the distribution centers and auction sites represent the volume of cars transported between these points (the thicker the line, the more cars are transported):



14 Only the largest leasing companies—such as GMAC—have such distribution centers. For leasing companies that do have them, an off-lease car is dropped off at a dealership, then shipped to the nearest distribution center for cleaning and conditioning, and then the Decision Optimization System ships the car to the best auction site. For leasing companies that don't have distribution centers, the car would be cleaned and conditioned at the dealership, and the Decision Optimization System would ship the car to the best auction site directly from the dealership.

The four graphs on the left-hand side display the optimization objectives:

- *Average Transportation Cost*. The Decision Optimization System calculates the total transportation cost and then displays the average cost per car.
- *Average Volume Effect*. The Decision Optimization System calculates the total lost revenue due to sending too many similar cars to the same auction sites and then displays the average value lost per car.
- *Average Sale Price*. The Decision Optimization System calculates the expected sale price for all the cars and then displays the average value per car.
- *Average (Net Sale Price) Lift*. This corresponds to the average "profit improvement" per car. The Decision Optimization System calculates this as the difference between the predicted average net sale price for the optimized solution (i.e. the sale price after subtracting all auction fees, transportation costs, etc.), and the predicted net sale price for the standard solution (which was based on expert rules that were developed by GMAC over the years).

In the screen above, the average transportation cost per car (first graph) has steadily decreased during the optimization run, while the average volume effect per car (second graph) has increased. The Decision Optimization System has chosen a distribution plan with a higher average volume effect, because it was more than offset by a lower average transportation cost and higher average sale price per car. This in turn resulted in a higher average (net sale price) lift per car (fourth graph).

Once the optimization process is complete, the Decision Optimization System generated an output file with the recommended distribution of cars, specifying the distribution center, recommended auction site, predicted sale price, transportation cost, and other data:

No.	Make	Model	Trim	Year	Distribution Location	Auction	Sales Price	Volume Effect	Distance	Transportation Cost	Net Price	Lift
1	Ford	F150	Base	2001	Augusta, ME	ADESA Buffalo	$9,452	$0	445	$180	$9,272	$113
2	Jeep	Grand Cherokee	Limited	2003	Albany, NY	ADESA Buffalo	$17,786	$0	241	$123	$17,663	$225
3	Toyota	Land Cruiser	VX	2002	Annapolis, MD	ADESA Buffalo	$31,662	$0	300	$153	$31,509	$318
	Total	**3**					**$19,633**	**$0**	**328**	**$152**	**$19,481**	**$218**
4	Dodge	Durango	Base	2002	Boise, ID	ADESA Seattle	$15,548	$94	385	$68	$15,480	$74
5	Dodge	Grand Caravan	SE	2002	Boise, ID	ADESA Seattle	$10,025	$61	385	$68	$9,957	$48
6	Ford	Expedition	Eddie Bauer	2003	Boise, ID	ADESA Seattle	$25,502	$154	385	$68	$25,434	$122
7	Ford	Expedition	Eddie Bauer	2003	Boise, ID	ADESA Seattle	$24,858	$150	385	$68	$24,790	$119
8	Ford	Mustang	GT	2002	Boise, ID	ADESA Seattle	$16,361	$99	385	$68	$16,293	$78
9	Ford	Mustang	Base	2001	Boise, ID	ADESA Seattle	$11,083	$67	385	$68	$11,015	$53
10	Honda	Accord	EX	1997	Boise, ID	ADESA Seattle	$5,334	$32	385	$68	$5,266	$26
11	Honda	Accord	LX	2003	Boise, ID	ADESA Seattle	$12,083	$73	385	$68	$12,015	$58
12	Honda	Accord	EX	2002	Boise, ID	ADESA Seattle	$12,054	$73	385	$68	$11,986	$58
13	Jeep	Grand Cherokee	Limited	2001	Boise, ID	ADESA Seattle	$12,373	$75	385	$68	$12,305	$59
	Total	**10**					**$14,522**	**$87**	**385**	**$68**	**$14,454**	**$69**
14	Dodge	Durango	Base	2003	Saint Paul, MN	ADESA St. Louis	$18,574	$0	478	$205	$18,369	$199
15	Dodge	Durango	Base	2003	Springfield, IL	ADESA St. Louis	$18,969	$76	109	$55	$18,914	$87
16	Dodge	Neon	HIGHLINE	2002	Springfield, IL	ADESA St. Louis	$7,497	$30	109	$55	$7,442	$22

An auction inventory report (below) was used to show inventory at each auction site, the number of cars being sent to each auction, the projected number of cars at each auction, and whether or not any inventory constraints are violated:

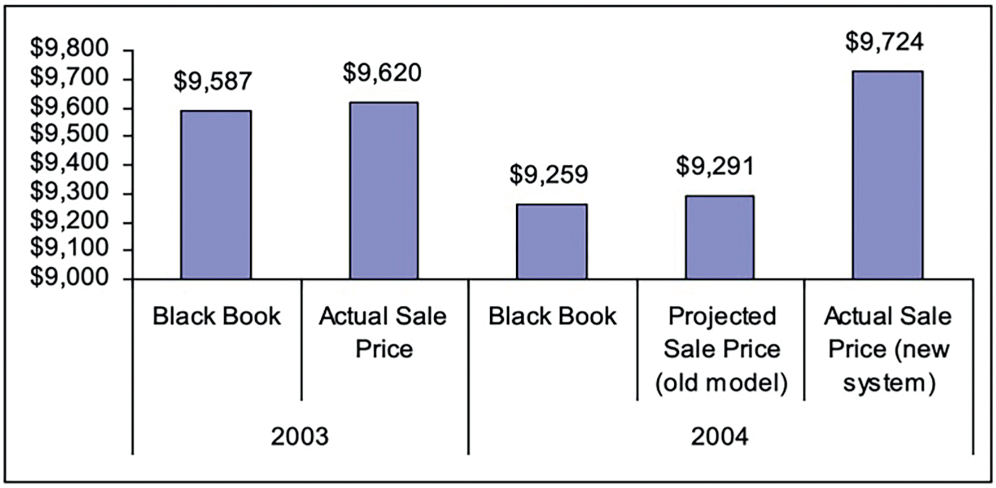
No.	Auction	Inventory	Distributed	Projected	Inventory Min.	Inventory Max.	Over(+)/ Under(-)
1	ADESA Atlanta	261	33	294	200	300	0
2	ADESA Birmingham	254	7	261	150	300	0
3	ADESA Boston	390	7	397	300	400	0
4	ADESA Buffalo	99	3	102	100	150	0
5	ADESA Charlotte	120	19	139	100	200	0
6	ADESA Cincinnati Dayton	123	8	131	100	200	0
7	ADESA Colorado Springs	297	0	297	150	300	0
8	ADESA Dallas	289	3	292	200	300	0
9	ADESA Des Moines	103	1	104	100	150	0
10	ADESA Golden Gate	141	10	151	100	150	+1
11	ADESA Houston	213	0	213	150	300	0
12	ADESA Indianapolis	135	2	137	100	150	0
13	ADESA Jacksonville	190	9	199	200	300	-1
14	ADESA Kansas City	185	16	201	150	300	0
15	ADESA Knoxville	204	2	206	150	300	0
16	ADESA Lansing	258	2	260	150	300	0
17	ADESA Lexington	103	1	104	100	200	0
18	ADESA Little Rock	207	3	210	150	300	0
19	ADESA Los Angeles	257	0	257	150	300	0
20	ADESA New Jersey	154	6	160	150	300	0
21	ADESA Phoenix	154	10	164	150	300	0
22	ADESA Pittsburgh	156	0	156	150	300	0
23	ADESA San Antonio	286	1	287	150	300	0
24	ADESA Seattle	224	10	234	150	300	0
25	ADESA Shreveport	162	6	168	150	300	0
26	ADESA St. Louis	182	6	188	100	200	0
27	ADESA Tampa	214	5	219	150	300	0
28	ADESA Wisconsin	173	2	175	100	200	0
	Total	**5,534**	**172**	**5,706**			

When used in a high-volume setting—where thousands of cars are returned off-lease each day—the Decision Optimization System generated a net profit lift in the hundreds of millions of dollars per year (by predicting the auction site at which GMAC could maximize the resale value of each car, and then optimizing the logistics). There were a few ways to validate this financial result:

- One way was by dividing the daily intake of returned off-lease cars into two equal groups with an almost identical division of makes/models. One group would be distributed using the manual method, whereas the Decision Optimization System would distribute the other group, and then the results would be compared.
- Another way was by using the manual method on selected days of the week (e.g. Mondays, Wednesdays, and Fridays) and the Decision Optimization System for the remaining days (e.g. Tuesdays and Thursdays). Again, the results could be compared when all cars were sold and the aggregate prices known.

- And the third way was by using the Decision Optimization System for one year and then comparing the average sale price with that of the previous year (before the system was implemented).

Using this last method (year-by-year comparison), the benchmark would need to be a trusted pricing source, like the Black Book price guide. GMAC applied this particular method by selecting a subset of cars with the same makes/models, year, trim, etc. and compared the average sale price of these cars with the average Black Book sale price for 2003 (before the Decision Optimization System was implemented). A chart depicting this comparison is presented below:



In this example, the average Black Book sale price for a particular mix of makes/models, year, trim, etc. in 2003 was $9,587 per car, and GMAC sold these cars for an average of $9,620 per car, or 0.344% higher than the Black Book sale price. The next step would be to compare the sale prices in 2004 (when the Decision Optimization System replaced the manual method of distributing cars) against the Black Book sale prices for that year. In this example, the average Black Book sale price was $9,259 per car in 2004, and the average actual sale price obtained by the system was $9,724. If the cars had been distributed using the manual method in 2004 (termed "old model" in the chart above), then GMAC would have attained similar results to those of the previous year (i.e. a 0.344% improvement over the Black Book benchmark, or an average of $9,291 per car). Using this approach, GMAC could credit the Decision Optimization System with the increased average sales price of $9,724 minus $9,291, or $433 per car. With one million cars being distributed on an annual basis, this result represented $433,000,000 in additional revenue, not to mention automation of a business process that required a substantial amount of time and human effort.

And lastly, as discussed in Chapter 6.10, the concept of global optimization was highly relevant to this distribution problem given the scale of the problem (e.g. million cars per year) and complexity (e.g. number of possible solutions, volume effect, price changes, transportation costs, etc.). A tempting approach for dealing with this scale and complexity would have been to break the overall problem into smaller "pieces"; for example, breaking up the United States into six regions that each dealt with a subset of the overall problem. This approach could have been taken further by breaking the problem apart into individual states or distribution centers, solving these individual pieces, and then "assembling" the pieces into an overall distribution plan. Although this would have made the problem easier to solve, the result would have been substantially inferior in comparison to solving the problem in its entirety (because the Decision Optimization System could consider and balance the volume effect, price changes, depreciation, and more across the entire United States, rather than being constrained to a particular state, site, or region). As this case study illustrates, the financial result of global optimization can be substantial, but the amount of complexity that needs to be addressed is equally substantial.

Index

M

N

O