The Future: Mathematical Programming 4.0

Jesus Velásquez-Bermúdez

Abstract. The Mathematical Programming (MP), and in general the mathematical modeling based on algebraic equations, requires new ideas to meet the challenges of the future, which already is bringing fundamental changes in the way of doing things, this is reflected in the new industrial revolution, called Industry 4.0. The short-term requirements in Mathematical Programming are: i) Standardization of Mathematical Programming Modeling (easy connection of multiple mathematical models), ii) Expert Optimization Systems (capitalization of the knowledge included in the results of the optimization), iii) Socialization of large-scale technologies to the community of mathematical modelers; it must be a basic knowledge not an expert knowledge, including teaching at graduate levels; the large-scale methodologies must be connected by to the model by parametrization, in a similar way that we, actually, connect the basic solvers, iv) Socialization of basic optimization in final users; this implied more end users of the optimization methodologies.

To developed new applications, according to the modern information technologies, it is necessary a new look of optimization according to the real-world technologies: Internet of Things (IoT), Industrial Internet of Things (IIoT), Smart Metering, Big Data and Robotization. A new environment to solve problems that must include the concept of Real Time Optimization (RTO) according to the Industry 4.0 and mainly with the cyber-physical systems controlled by algorithms based on advanced analytics and closely integrated with the internet.

This new environment may be called Mathematical Programming 4.0.

J. Velásquez-Bermúdez
DecisionWare - Do Analytics, Colombia
jesus.velasquez@decisionware.net

INDEX

1. Mathematical Modeling in the Future

The Mathematical Programming (MP), and in general the mathematical modeling based on algebraic equations, requires new ideas to meet the challenges of the future, which already is bringing fundamental changes in the way of doing things, this is reflected in the new industrial revolution, called Industry 4.0, that it is focused on the automation of manual human processes, leaving aside the cognitive activities. In the decade of the forties of the past century, the computation changed the calculation capacity of human that begins to use algorithms to solve increasingly complex mathematical problems.

While the algorithms will be a fundamental part of human knowledge, the future will be marked by new applications that enhance the ability to analyze and to generate knowledge from data online and speed-up: i) the programming development, ii) the deployment, and iii) the maintenance of real-life applications, using more effective mathematics methodologies as the artificial intelligence methodologies.

The change involves “think outside the box” that is far more than just another management cliché. This document contains several ideas about the future of the Mathematical Programming some of them implies a new paradigm in mathematical programing technologies.

The short-term requirements in Mathematical Programming are:

1. Standardization of Mathematical Programming Modeling (easy connection of multiple mathematical models)
2. Expert Optimization Systems (capitalization of the knowledge included in the results of the optimization)
3. Socialization of large-scale technologies to the community of mathematical modelers; it must be a basic knowledge not an expert knowledge, including teaching at graduate levels.
4. The large-scale methodologies must be connected by parametrization, in a similar way that we connect the basic solvers.
5. Socialization of basic optimization in final users; this implied more end user of the optimization methodologies. A new look of optimization according to the real-world technologies: i) Internet of Things (IoT), ii) Industrial Internet of Things (IIoT), iii) Smart Metering and Big Data, and vi) Robotization

2. Structured Mathematical Modeling (SMM)

Structured Mathematical Modeling (SMM) is defined, by the author, as a fundamental step in the process of socialization of the mathematical modeling, it is a necessity to ensure that the benefits arising from the applied mathematics extend to as many people as possible. This cannot be achieved, while the mathematical modeling is not within the reach of most professionals in engineering, economics and management sciences.

The main barriers must be overcome is the dependence of mathematical models from the Mathematical Programming technologies used to implement the models. The alternative is to normalize the formulation in such a way to ensure their portability between technological platforms. This standardization would allow professionals interested in the mathematical modeling the possibility of formulating their own models without to know in depth the syntax of a computer language; this fact would expand the number of mathematical modelers and diminish the level of expert knowledge required to formulate mathematical models.

An example of the current barrier is the case of a student who develops his thesis using a commercial high-performance optimization technology (IBM OPL, AMPL, AIMMS, GAMS, MODEL,...) owned by the University, or temporarily licensed by the supplier. The day after their degree, a trade barrier is created between the knowledge generated by the student in his thesis and its possible commercial use; since the student needs a formal commercial license to continue developing the thesis model or new models; then it depends on the economic condition, which is usually weak at the beginning of the professional practice. Fortunately, the producers of math technologies begin to offer their services in the cloud, which will facilitate the development of math models making use of limited licenses and their subsequent scaling to solve real-life problems in the producers of technologies servers.

The standardization process must define:

1. SMM Basic: the part of the mathematical modeling process that is included in the standard. This implies: i) regulate by a common agreement made by the representatives of all the mathematical modeling-related communities, and ii) that it should be “mandatory” for the industry and
2. SMM Advanced: the part of the process that is covered by the optimization companies, as a way of differentiation of products and services offered; it is not binding, but it is convenient for humanity.

In case that the mathematical modeling world community does not get in accordance with a global SMM, as it is suggested here, the ideas of this document can provide guidance for the organizations that management/produce large number of mathematical models, they can make an internal standard, which allows to capitalize the value added by the normalization.

2.1 Framework

2.1.1 Socialization and Standardization

Standardization of MP modeling is necessary to make easy, and sure, the connecting the mathematical models in an organization and between organizations; it is the key to facilitate the development and the implementation of the informatic technologies that underpin the enterprises digital transformation and to guarantee interoperability between different systems and solutions.

The adoption of recognized internationally standards facilitates to export the technology from the suppliers to the end users. Therefore, the Mathematical Programming community must collaborate in the elaboration of international standardization initiatives, coordinating the proposals and the needs of the: i) industrial companies, ii) producers of technologies, iii) advanced professionals, v) researchers and i) academic sector. It is the indicated way for society massively capture the wealth hidden in applied mathematics.

A clear example of growth and socialization of the methodologies and technologies was lived in information systems. The relational database (RDB) is a type of database (DB) that complies with the relational model, the most commonly used model for currently implement databases (DB). After being postulated their bases in 1970 by Edgar Frank Codd (from IBM), it becomes as a new paradigm in data base models. Codd showed the potential of the implementation of its model based on the expressive power of relational algebra.

The results of the process started by Todd were: i) the relational information systems, with its rules of implementation, and ii) SQL, Structured Query Language to interact with RDB; the benefits of standardization are resumed in the portability of existing information systems, which ensures that the end-user the control over the technologies that it has acquired (GNU General Public License or commercial), since there are rules that a dedicated professional can understand.

Before Todd, information systems technologies were up to the developer; this is the actual situation of math technologies. The owner of a mathematical model is the mathematical modeler that programs the computer codes of the algorithms, mainly when the solution is tailored to end user.

This standardization will be called Structured Mathematical Modeling (SMM). The possibility of standardizing MP technologies has been tested in the real-life by the models developed using OPTEX Expert Optimization System (Velasquez, 2019). The methodology of work implemented in OPTEX, for more than 20 years, guarantees that the mathematical models transcend to the mathematical modelers; ensuring the portability of implemented models that follow the OPTEX standardization. Today, using OPTEX, a non-expert user can use a text editor (like MS-Word) to fill standard templates to implement algorithms using an optimization, or general, technology (like GAMS, ILOG OPL, C^{++}, …) making use of advanced optimization methodologies, and their variations and improvements.

As an initial step, to define a SMM it is required to analyze the mathematical modeling process; then, to define the scope of standardization and the details that should be followed.

2.1.2 Mathematical Programming: A Natural Standard

The math is a natural standard for all professionals who need it for their professional practice. The formulation of a problem in algebraic terms, or in differential equations, allows professionals, from different cultures and with different languages, to express in a way such that all are understood through mathematics. Two algebraic formulations of the same problem can look different by the symbols used by the math modelers; but, if we build a map of the symbols all formulations should be equal.

MP meets the transitivity law, well known by mathematicians, indicating that if two objects of the same type are combined, the resulting object is of the same type. For example, an integrated model of the electricity-gas system is the union of the equations of the two individual systems (gas & electricity) plus the coordination constraints, that may include some new variables.

Another case is related to with Enterprise Wide Optimization, associate to the natural evolution of the business planning models that are link to the administrative functions of the organization. EWO involves dynamic mathematical models that evolve in the same way how the company evolves. These models can be built as a union of existing models that are associated, each of them, with a function within the organization, these models are: i) Supply Chain Management (SCM), ii) Sales & Operations Planning (S&OP) and iii) Integrated Business Planning (IBP). The figure 1, adapted from a diagram included in the digital paper "Beyond Supply Chain Optimization to Enterprise Optimization" (Shapiro J., 2006) allows to visualize the concepts expressed.

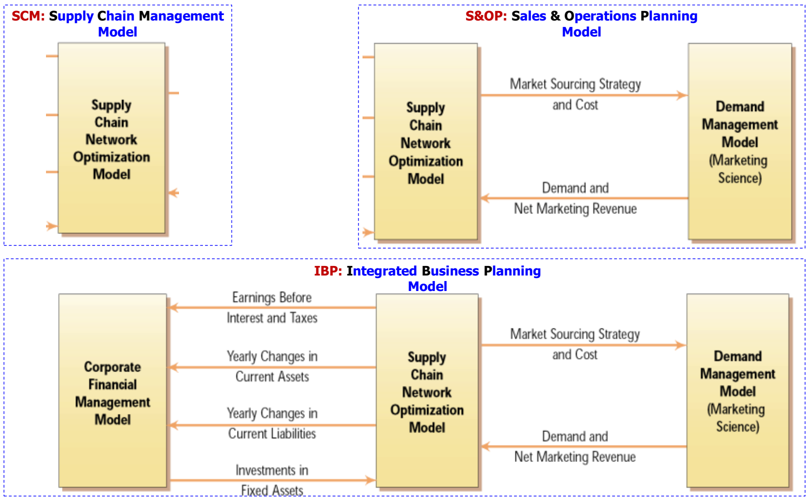


Figure 1. Enterprise Planning Model

To capitalize this advantage, SMM requires an appropriate computer technology that allows adding objects and produce a new object that can handle with the same technology; this is impossible for the case that we have two models in two different computer programs in any optimization technology.

A technology suitable for this is RDB, enabling to join tables whose results are still tables. If you check carefully, mathematical models can be arranged as data of an RDB, since they can be structured as a collection of related elements/objects/entities. Then the proposal for SMM is to use RDB to standardize the mathematical modeling.

Among the advantages of selecting the RDB as the base technology for the SMM are:

1. RDB is a socialized technology, there are multiple technologies RDB (commercial and GNU) available to modelers
2. RDB tables can handled using basic computer technologies such as spreadsheets (like MS-EXCEL), RTF files (Rich Text Files Format, like MS-Word), or CSV text files (Comma-Separated Values).
3. All “persons” are accustomed to fill templates and therefore they don’t require a special learning process.

This Mathematical Modeling Information System will be called MMIS.

Then, SMM is based on the vision that sees the MP as a standard that can be understood by any expert modeler, this standardization must be so solid that ensures that the binding of MP problems is a new problem of MP, for this a mathematical model must be conceived as the union of math components harmonically integrated. For example, a problem is a set of equations, a model a set of problems; and an equation the integration of formulas, variables, indexes, parameters and sets.

Another part of the process is related to the storage of the input/output data of the mathematical models. If the RDB are the standard of information systems, the logical thing is that the RDB are the standard for the organization and storage of Industrial Data Information System (IDIS).

2.1.3 Integration of Mathematical Methodologies

MP is normally linked to optimization or to equilibrium problems, or to the integration of these two methodologies under MPEC (Mathematical Programing with Equilibrium Constraints) models. However, this concept can be extended to all the mathematical methodologies which are supported in algebraic formulations, it is fundamental to integrate models from different methodologies.

For example, optimization supports, as a fundamental concept, the methodologies oriented to study stochastic processes; then, with the appropriate vision, the following methodologies are based in optimization problems that minimize an objective function (may be a loss/penalization function or a multicriteria objective function):

1. Machine Learning (ML):
   - Support Vector Machines (SVM), Support Vector Networks (SVN) and Support Vector Regression (SVR) (Cortes and Vapnik, 1995)
   - Reinforcement Learning: Markov Decision Process (MDP, Puterman 1994)
   - Clustering (Hansen and Jaumard 1997)
2. Advanced Probabilistic Models:
   - S-ARIMAX-GARCH (Box and Jenkis 1970, Engle 1982)
   - State Estimation: Kalman Filter (Kalman 1960)
   - Markovian/Semi-Markovian Process
   - Bayesian Inference
   - Constrained Econometric Models (Wolak 1989).
3. Ensembles of models using these methodologies

This can be a significant advantage, since it facilitates the modeling of variations around the basic methodologies. For example, to estimate models of "market share" for multiple vendors (V), acting in multiple regions (R), selling multiple product categories (C); it involves simultaneously solving V×R×C statistical models, with the following constraints: i) a common constraint that guarantee that the sum of the market shares in R×C markets must be equal to 1 and ii) that the market share of each vendor, in all R×C markets, must be greater or equal than zero. This case is call Simultaneous Constrained Least Squares and it easy to model using MP.

The union of mathematical methodologies (machine learning, advanced probabilistic models, optimization, …) around MP brings the great advantage of the ease of integration of algebraic formulations. Since the union of MP models produces a new MP, it is easy to mix the algebraic formulations of multiple methodologies in a single model.

2.1.3.1 Integration Statistical Models and Optimization

One example of integration statistical models and optimization may occur in the traditional Sales & Operations Planning (S&OP) process, showed in figure 2a, in which the substitution of demand tables by the algebraic equations of the statistical models (used to calculate the demand tables) allows that the demand is endogenously determined by S&OP model and not exogenously by end user, figure 2b.

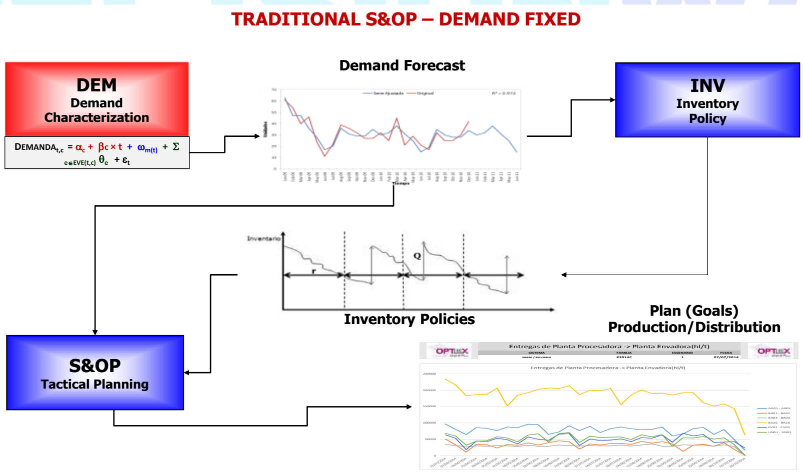


Figure 2a. S&OP Traditional Planning Process

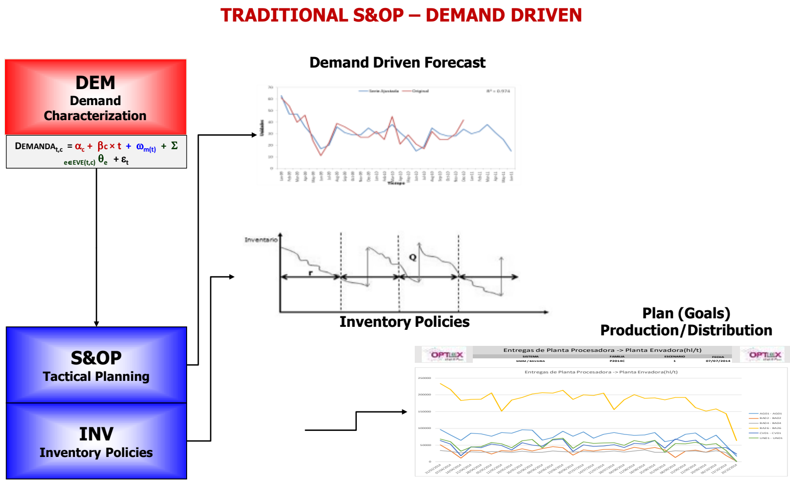


Figure 2b. S&OP Demand Driven Planning Process

The difference in this case is that the optimization model (S&OP plus inventory policy) determines the products to meet, according to their profitability. This type of modeling is called "demand driven" and is typical of the revenue management (optimal pricing) and marketing mix models.

2.1.3.2 Integration Machine Learning and Optimization

One example of machine learning models and optimization may occur in the case of a broker of transport services whose business is: i) to sell transportation services to the load generators, and ii) to purchase transportation services to the owners of the vehicles; a real case is the LTL (Less-Than-Truckload) market (Behrang, 2009).

In this case, a Decision Support Systems (DSS) for the broker should be the result of studying how should make offers of purchase and sale of transport services in the multiple hub in which he operates, so maximize your revenue. The historical data that has the broker are all offers made and the result of the same: success if achievement the business, failure otherwise. The broker must have the following models to fix his offers:

- Load Winners Offers, based on ML, determines the Support Vectors Machine (SVMs, Vapnik, 1995, Vapnik, 1998) to segment the price bids to load generators (SVM-DEM).
- Vehicle Winners Offers, determines the SVMs for segmented price offers to pay transporters (SVM-OFE).

Additionally, the broker must have a Revenue Management model (MAX-RM), linking the offers to LTL network operations to be carried out. The SVMs should be incorporated as constraints in RM, limiting the offers to those that are classified as successful. The figure 3 presents the flow of data over the three models (Velasquez 2019b).

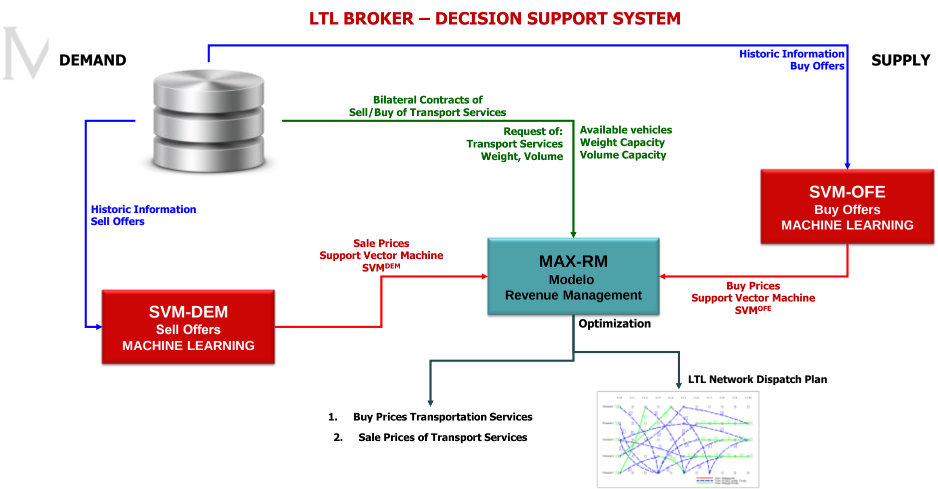


Figure 3. Revenue Management integrating Machine Learning and Optimization Models

2.1.4 Integration of Multiple Models and Problems

The evolution of Mathematical Programming involves the change of products which mathematical modelers delivered to enterprises. In the beginning the product was associated with a computer program that solved a problem specific and which was developed using low level computer languages, as FORTRAN or C, linked to an optimization library; considering the evolution of computing, today is clear that a mathematical model of a specific problem is simply a part of the product that require the end user.

Today, the product to deliver is DSS, may be using internet, composed of multiple models, which must share data among themselves, that should use advanced mathematical methodologies that can effectively use multiprocessing capabilities of the modern computers.

Due to the complexity of real systems, DSSs are composed of multiple mathematical models which are integrated through the data stream, thereby generating the information required by the decision maker to address all hierarchic levels: strategy, tactic, scheduling and real-time operations. The connection of data and models defines the decision-making chain, which supports the management productivity of organizations.

The different models must share information stored on a common database, coherent and standardized, to allow data integration along the decision-making chain, in which some of the outputs of a model becomes the inputs of the models of subsequent stages, so this coordinated effort guarantees the "sub-optimization" of the entire system; then, it is impossible to obtain with a holistic single model an “optimal” solution. Researchers and producers of technology solutions share this point of view.

The concept of SMM allows that a problem being part of several models and the constraints to be part of multiple problems and so on. This approach facilitates to build multi-problems DSS and to hand large-scale optimization models; since under the partition and decomposition scheme, a model consists of several coordinated problems whose solution is performed in accordance with an optimization methodology; like Benders Theory (1962), Lagrangean Relaxation (Kurt et al., 1985), Dantzig-Wolfe Decomposition (Dantzig and Wolfe, 1960) and/or Column Generation (Lubbecke, 2010) which can be integrated based on the concepts of Cross Decomposition (Van Roy, 1983).

2.1.5 IoT, IIoT and Dynamic of Smart Metering

Smart metering systems directly impact the use of optimization in real-life problems. The conventional is to think that a problem is solved at a given moment and turns to meet periodically every hour, every day,...; this presupposes that the information that is required to run the model is achieved from a run to the next, but the big-data generated by the smart-metering systems change completely the decision-making environment.

To view the impact, consider a problem easy to formulate as the VRP-TW (Vehicle Routing Problem with Time-Windows). Traditionally, a routing urban model runs n times in a day, and it assumes constant the expected travel times between each pair of nodes in the network; however, it is aware that in cities with congested traffic networks, this hypothesis does not correspond to reality. But trying to change this hypothesis involved complications since there were no organized measurement systems that could generate the information required to consider the travel times as a function of the time of departure of the vehicle from a specific node and linking such times to a specific path (sequence of streets) between two points on the network. The routing solution would be changed by the occurrence of exogenous events or when occurs the time limit for new routing; but keeps the hypothesis of time-independent travel times.

However, the situation today is totally different. There are intelligent big data measurement systems as Waze (the “world's largest” community-based traffic and navigation app). Waze provides/sells real-time traffic and road info; then, it is possible to constantly updated travel time between two points and the path associated with this time. Based on Waze measurements it is possible to define travel time function, $TT_d(t,_{n1},n_2)$, that indicates the average expected time for a trip from node n_1 to node $n_{2,}$ beginning at the moment t of the day d. This involves:

- The ability to resolve a new type of VRP problem with variable travel times, that it is not a trivial problem.
- Define models for re-routing and not for routing (starting from zero), since the optimization can be updated whenever an event occurs that changes the expected travel times, may be every second.

This new problem may be called VRP-TW-TDTT (Vehicle Routing Problem with Time-Windows and Time-Dependent Travel Times).

The processes enabled by intelligent measurement systems and by IoT (Internet of Things) and IIoT (Industrial Internet of Things) may be called Real-Time Optimization (RTO).

2.1.6 Problem Solution

To define a SMM it is necessary to specify the “standardized” process to follow in the solution of mathematical problems. The first thing is to define the inputs and the outputs of the mathematical model (figure 4). The inputs correspond to the values of sets and parameters which are read from the IDIS RDB, and outputs to the primal variables (activity values and dual variables of bound constrains) and the constraints (dual and slack variables) results of the solution of the mathematical model, that should be stored in the IDIS.

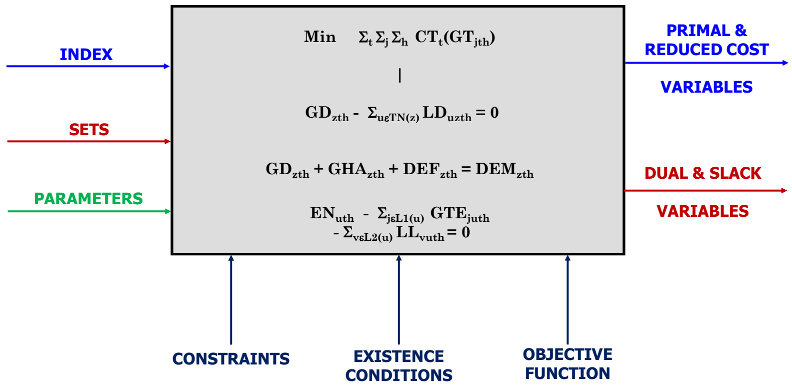


Figure 4. Input & Output Process

The simplest model that can be defined is an integrated problem that is solved without making cycles within the process, it may be conceptualized as:

1. Reading: Load parameters and sets from the RDB.
2. Pre-processing: calculation of sets and parameters from data read and/or other calculated data. In complex models (NP-hard) this activity may be associated with a domain reduction to facilitate the solution of the problem. Load the calculated values in the database.
3. Optimization: solution of the mathematical problem through an optimization solver
4. Post-processing: calculation of complementary results using the solution of the mathematical models
5. Storing: download the results in the RDB.

It is possible to define three levels of complexity to solve the model:

1. Integrated: The simplest model
2. Loop-Models: Models that require cyclic processes to be solved. There are several types of models that meet this feature:
    i) Parametric programing: the same problem solved for several values of their parameters, presumably based on a systematic variation.
    ii) Families of problems: solution to multiple problems, dependent of one or multiple indexes, that have the same structure, but varying data based on each levels of problems, e. g. DEA (Data Envelopment Analysis, Charnes et al., 1978).
    iii) Large-scale optimization: models solved using large-scale methodologies (partition/decomposition) that convert the integrated model into multiple smaller problems.
3. Parallelization: any of the cases of loop-models can be solved using multi-processing.

The figure 5 presents the optimization process.

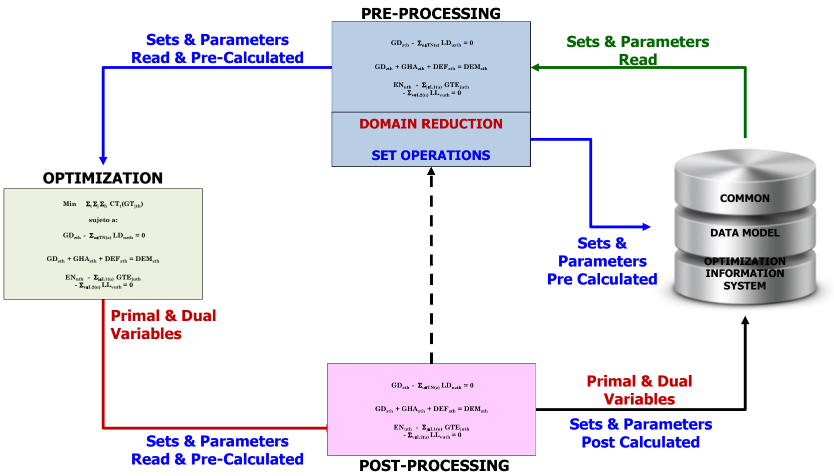


Figure 5. Optimization Process

2.1.7 Large Scale Optimization Methodologies

The concept of multi-problem model facilitates the implementation of Large-Scale Optimization Methodologies (LSOM) based on multi-level partition and decomposition, using Bender's Theory and/or Lagrangean Relaxation and/or other mathematical methodologies, like Matheuristics (heuristics based on MP).

The chapter J. F. Benders Theory, Variations and Enhancements (Velasquez 2019d) present very small summary of papers showing the gain in speed of the proper use of the improvements in Benders Theory. This leads to conclude that the point of reference to compare the speed of Mathematical Programming to solve complex problems are not the best basic solvers; the proper reference is the large-scale methodologies that make smart use of these solvers. Then, future of MP will be concentrated in multi-processing using LSOM rather than in the solution of basic problems; then, the research focus must be in generating effective computational codes to resolve such problems by making use of: i) computers with multiple CPUs and GPUs, and large RAM storages, and ii) computer grids. Thus, the no-expert modeler in this type of technology needs to access them. The research into solve optimization models in a GPU is the state-of-the-art (Meyer et al., 2017).

Considering that large scale technologies are the necessary complement to the basic optimization solvers (IBM CPLEX, GUROBI, XPRESS, …), since the union of the two powers (computers plus LSOM) allow to solve larger and more complex mathematical problems, SMM must incorporate, as part of its services, the automatic generation of computer algorithms using the variations and the improvements that have been developed by researchers. This may be in a second stage of SMM, or an option offered by the optimization technology companies.

OPTEX screenshot (figure 6) allows the parameterization of a model using the Benders Theory so that the end-user can make a research/study/project to determine which Benders methodology can be called the "best" for its specific problem; it shows that this approach is feasible (Velasquez 2019a).

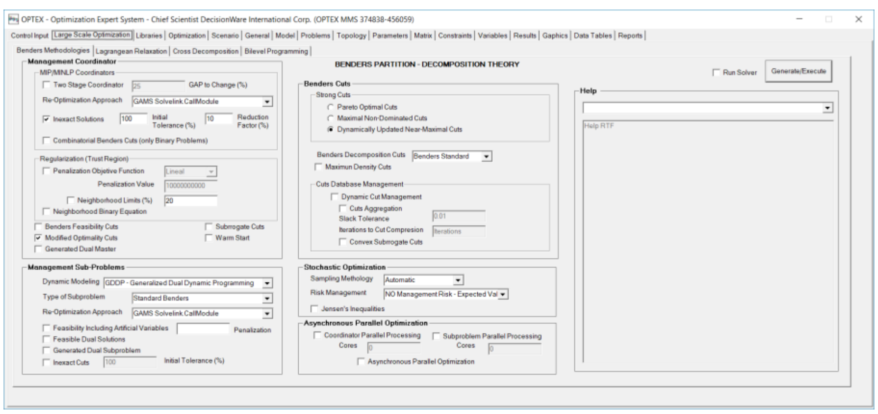


Figure 6. Benders Parametrization OPTEX Control Window

Therefore, it is a valid conclusion that education on optimization, whose upper limit is to use basic algorithms and implementation of integrated models is a matter of the past.

2.1.8 Stochastic Optimization

The power of computers, coupled with the power of LSOM, coupled with the power of the basic solvers change radically the environment of the mathematical modeler compared with the environment of the modelers of the past; then Stochastic Programming (SP) models should be common to math modelers and end-users of modern MP.

This leads to that the state-of-the-art of applied optimization solutions should migrate, massively, from deterministic optimization models to stochastic models. Stochastic optimization is there for many years, the first work, that the author knows, is the related with Modern Portfolio Theory (MPT), or mean-variance analysis, that was introduced by Economist Harry Markowitz in 1952 (for which he was awarded with the Nobel Prize in Economics).

Then, SMM must include as part of its services the modeling of Multi-Stage Stochastic Programming (MS-SP) that implies to handle random processes over the decision trees and solve problems with different types of objective functions, for example: i) expected value; ii) MiniMax or Maximin and iii) maximum regret; additionally, SMM must include several alternative to risk management; for example, Conditional-Value-at-Risk constraints (CVaR).

Therefore, it is necessary to normalize the process of conversion of deterministic (core) model into a stochastic model; this process may be automatic, in the sense that the user must only configure the conversion process and SMM generates the stochastic model from the deterministic formulation. For this, SMM can define a process that considers (figure 7):

1. Decision Tree, it may be generated using "split" variables with non-anticipative constraints, that is easiest way to express the decision tree.
2. Stochastic Process, the uncertainty dimensions must be defined by the users, considering which is the more convenient stochastic process. An easy way is to link random variables to parameters and/or sets of the core model, including indexes to handle each dimension of uncertainty. Currently the dimensions of uncertainty are included directly in the formulation of the problem implying that change the uncertainty dimensions involves changing the code of the program; This limits the correct use of the models, since in many cases it uses the model that is available and not the model that really requires the problem that is solving.
3. Risk Management: The biggest advantage of stochastic models is the inclusion of risk measures in the stochastic model. Nowadays, the risk measure most used is the CVaR (Conditional-Value-at-Risk, Rockafellar and Uryasev).
4. Solution Process: the solution of the stochastic model can be accomplished through direct solution of equivalent deterministic problem (the random variables are fixed during the optimization process) or a "real" stochastic model (the random variables change during the optimization process) using LSOM. Sampling methods that may be included in the SMM algorithms.

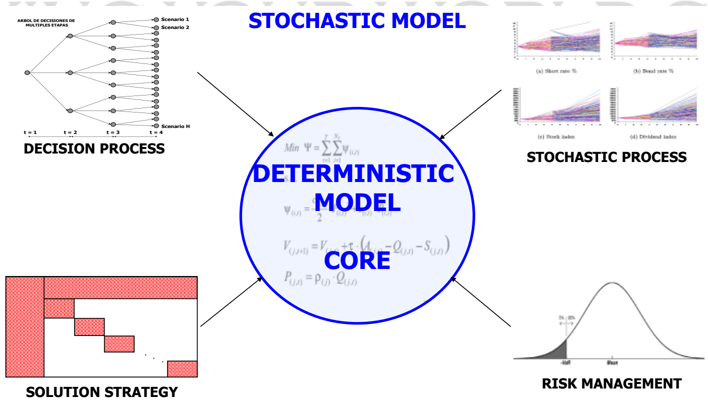


Figure 7. Stochastic Optimization Environment

Therefore, it is a valid conclusion that to use deterministic models, when uncertainty is an essential part of the decision-making process, is a matter of the past. More information about this topic can be read in Velasquez (2019d, 2019e).

3. SMM - Data Model: An Example

SMM standardizes the management of entities and relationships centered about its database algebraic language that must allow management of linear and non-linear equations.

Sort the elements that are part of a mathematical model around the concepts of RDB involves the need to structure the process of mathematical modeling in a way to store all elements in the tables of the SMM; this implies organize the mathematical model from an "universal" point of view of a relational information system; then, it is possible to affirm that the information system that supports SMM mathematical modeling is the first step towards normalization of the algebraic formulation and the use of mathematical models.

The implementation of the SMM can be arranged by stages (levels), at least two must be considered:

1. SMM Basic: includes entities related with: i) integrated basic models and ii) IDIS data model.
2. SMM Advanced: includes entities related with multi-problem models

3.1 Entities

The foundation of an RDB is the data model that is determined by the entities and the relationships between entities that will handle. Then the author proposes a possible SMM data model, which is supported in the design of MMIS of OPTEX, but it should be noted that it is only a proposal, which can be changed/adjusted in a broader discussion of the MP community.

The entities that are part of the SMM can be divided into three groups:

1. Formulation of Mathematical Models, entities used in the formulation of mathematical models, it must be included in SMM Basic.
2. Advanced Mathematical Modeling , entities used in the formulation of multi-problem mathematical models, it must be included in SMM Advanced.
3. IDIS Data Model, entities used in the formulation of data model of IDIS, it must be included in SMM Basic.

3.1.1 Formulation of Mathematical Models

The central entity/object of the algebraic formulation is the equation/constraint, it integrates the entities required to conceptualize an information system that store the elements/entities of a mathematical model, these entities must be at least: i) indexes, ii) sets, iv) parameters, and iv) variables.

Consider the following equation (called $PENV_{t,e,s}$) of balance production in packing plants of a S&OP distribution model:

$$\Sigma_{l \in LIE(e)} FEF_{l,s} \times PLE_{t,e,l,s} + \Sigma_{l \in LIE(e)} FEF_{l,s} \times PEE_{t,e,l,s} - \Sigma_{j \in ENP(e)} DPC_{t,e,j,s} = 0 \qquad \forall t \quad \forall e \in ENV \quad \forall s \in PEV(e) \qquad (1)$$

where

Indexes:

t Period
e Packing plant
l Packing line
s Final product
j Distribution Center

Sets:

$l \in LIE(e)$ Packing lines (l) in plant (e)
$j \in ENP(e)$ Distribution centers (j) connecting with packing plant (e)
$e \in ENV$ Packing plants
$s \in PEV(e)$ Final product (s) packing plant (e)

Parameters:

$FEF_{l,s}$ Efficiency in the process of packing the line (l) for the final product (s). Range: $0 \leq FEF_{e,l,s} \leq 1$.

Variables:

$PLE_{t,e,l,s}$ Production of final product (s), in ordinary time, in line (l) in plant (e) (boxes)
$PEE_{t,e,l,s}$ Production of final product (s), in overtime, in line (l) in plant (e) (boxes)

$DPC_{t,e,j,s}$ Dispatch of final product (s) from packing plant (e) to a distribution center (j) (boxes)

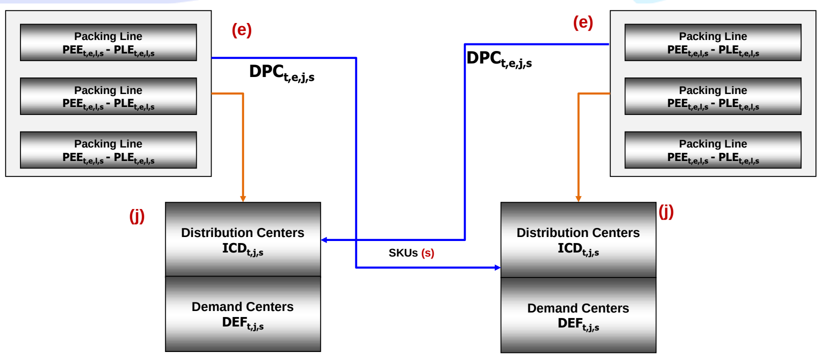
The algebraic formulation contains information of the five entities listed previously. As all RDB, the MMIS data-model requires: i) a master table and a relational key for each entity, and ii) as many secondary tables as relations between entities exist. Then, to storage the equation SMM requires the tables presented in table 1:

Table 1. Mathematical Modeling Information Systems (MMIS) – Basic Model Tables

Table	Entity	Comments – Main Information Stored	Relational key
MASTER TABLES			
INDEX	Index	Name of the indexes: t,e,l,s,j	COD_IND
SET	Set	Name of the sets: LIE, ENP, ENV, PEV	COD_SET
PARAMETER	Parameter	Name of the parameters: FEF	COD_PAR
VARIABLE	Variable	Name of the variables: PLE, PEE, DPC	COD_VAR
RESTRICTION	Restrictions	Name of the constraints: PENV	COD_RES
SECONDARY TABLES			
SET_IND	Set-Index	Independent indexes of the sets: LIE(e), ENP(e), PEV(e)	COD_SET COD_IND
PAR_IND	Parameter-Index	Indexes of the parameters: FEFl,s	COD_PAR COD_IND
VAR_IND	Variable-Index	Indexes of the variables: $PLE_{t,e,l,s}$, $PEE_{t,e,l,s}$, $DPC_{t,e,j,s}$	COD_VAR COD_IND
RES_IND	Restriction-Index	Indexes of the constraints: $PENV_{t,e,s}$	COD_RES COD_IND

To store mathematical models in an RDB, the entities must associate to codes like any RDB. This implies to adjust certain usual mathematical modeling that are closer to the academy than to the real-life deployments. Free the mind of these limits, facilitates that the MMIS handle hundreds of equations; this is important, since it is not surprising that a model for solving problems in the real-life has hundreds of generic equations. In the table, the names must be associate with the relational keys of MMIS.

Below, the tables used by OPTEX are presented (partially). The idea is that the reader visualizes the content and the structure of a MMIS. The formulation presented corresponds to the model PTALIN (figure 8); the formulation, the GAMS code and OPTEX tables, with a data set, may be may downloaded from Velasquez (2019c).



Variable	Existence Conditions	Description
$DFC_{t,j,s}$	$\forall t \ \forall j \in PUN \ \forall s \in PCA(j)$	Deficit of final product (unattended demand)
$DPC_{t,e,j,s}$	$\forall t \ \forall e \in ENV \ \forall j \in ENP(e) \ \forall s \in PPJ(e,j)$	Final product dispatch from packing plant to distribution center
$ICD_{t,j,s}$	$\forall t \ \forall j \in PUN \ \forall s \in PCA(j)$	Stocks of final product in distribution center
$PEE_{t,e,l,s}$	$\forall t \ \forall e \in ENV \ \forall l \in LIE(e) \ \forall s \in PRL(l)$	Final product production in overtime
$PLE_{t,e,l,s}$	$\forall t \ \forall e \in ENV \ \forall l \in LIE(e) \ \forall s \in PRL(l)$	Final product production in ordinary time

Figure 8. PTALIN Model – Visual Description

- **INDEX**

The INDEX table (table 2) must have at least five fields (name of the field between parenthesis); each index is associated to an entity in the IDIS; then it must be associate to a master table and a relational key. The indexes associate to time are universal; then, they don't be associate to a master table. Three index types must be considered: i) alphanumeric, ii) numeric, and iii) time. Fields COD_DB (master table) and COD_FIELD (key) are related with IDIS data model.

Table 2. Mathematical Modeling Information Systems (MMIS) – Table: INDEX					
Index (COD_IND)	IDIS Entity (DES_IND)	Alias (COD_INDA)	Type (COD_TIN)	Master Table (COD_DB)	Relational Key (COD_FIELD)
e	Packing plant		Alphanumeric	MAE_ENV	COD_ENV
j	Distribution Center	k	Alphanumeric	MAE_PUN	COD_PUN
k	Distribution Center	j	Alphanumeric	MAE_PUN1	COD_PUN1
l	Packing line		Alphanumeric	MAE_LIN	COD_LIN
q	Period	t	Time		DATE
s	Final product		Alphanumeric	MAE_PRF	COD_PRF
t	Period	q	Time		DATE

- **SET**

Sets may be divided into two groups: read and calculated. The entity contents in the set must be associated with the dependent index, and the indexes that "indexing" the set must be associated with independent indexes. The read sets must be linked to records in a table which can be filtered to define the appropriate sets. Calculated indexes must be associated with one operation. Field COD_DB is related with the data model of IDIS, the indexes related with the set are read in the field associated with the relational key. Then, table SET must have at least the six fields (table 3):

Table 3. Mathematical Modeling Information Systems (MMIS) – Table: SET				
READ SETS				
Set (COD_SET)	Description (DES_SET)	Dependent Index (COD_IND)	Table (COD_DB)	Filter (FILTER)
ENP	Distribution centers <- Packing plant | j∈ENP(e)	j	ENV_PUN	
ENV	Packing plants | e∈ENV	e	MAE_ENV	
EVL	Packing plants <-> Packing line | e∈EVL(l)	e	MAE_LIN	
LIE	Packing lines <-> Packing plant | l∈LIE(e)	l	MAE_LIN	
LPF	Packing line – Final Product | l∈LPF(s)	l	LIN_PRF	
PCA	Final product <-> Distribution Center | s∈PCA(j)	s	DEMCDA	
PRL	Final product <-> Packing line | s∈PRL(l)	s	LIN_PRF	
PUN	Distribution Center | j∈PUN	j	MAE_PUN	
CALCULATED SETS				
Set (COD_SET)	Description (DES_SET)	Dependent Index (COD_IND)	Operation (OPERATION)	
EPF	Packing Plant - Final product | e∈EPF(s)	e	$SU_{l \in LPF(s)}$ EVL(l)	
PEV	Final product <-> Packing plant | s∈PEV(e)	s	$SU_{l \in LIE(e)}$ PRL(l)	
PPJ	Product - Packing plant - Distribution Center | s∈PPJ(e,j)	s	PEV(e)∩PCA(j)	

The secondary table SET_IND stores the relationships between independent indexes and sets. Alternatively, to create a table, a field can be included in master table SET; however, for purposes of standardization and visualization of the relationships, seems best create the secondary table .

Table 4. MMIS - Table: SET_IND	
Set (COD_SET)	Independet Index (COD_IND)
ENP	e
EVL	l
LIE	e
LPF	s

Table 4. MMIS - Table: SET_IND	
Set (COD_SET)	Independet Index (COD_IND)
PCA	j
PRL	l
EPF	s
PEV	e

Table 4. MMIS - Table: SET_IND	
Set (COD_SET)	Independet Index (COD_IND)
PPJ	e
PPJ	j

- **PARAMETER**

Parameters may be divided into two groups: read and calculated; the read parameters must be linked to a table and a field in this table. Calculated parameters must be associated with one operation. Then, table PARAMETER must have at least the six fields (table 5). The function INV(x) is equal to 1/x.

Table 5. Mathematical Modeling Information Systems (MMIS) – Table: PARAMETER

Parameter (COD_PAR)		Read Parameters			
ID	Algebraic Expression	Description (DES_PAR)	Unit (COD_UNI)	Table (COD_DB)	Field (COD_FIELD)
BOCF	$BOCF_s$	Bottles per box	Bottles/Boxes	MAE_PRF	BOCU
CUPF	$CUPF_{e,s}$	Normal production cost per final product per box-unit	$/BoxU	ENV_PRF	CUPF
DMJS	$DMJS_{t,s,j}$	Demand of final product per distribution center	BoxU	DEMCDA	DEM
FCPF	$FCPF_s$	Boxes per pallet	Boxes/Pallet	MAE_PRF	FCPF
FCUF	$FCUF_s$	Boxes per box-unit	Boxes/BoxU	MAE_PRF	FCUF
FLPC	$FLPC_{e,j}$	Freight packing plant to distribution center per trip	$/Trip	ENV_PUN	FLE
ICD	$ICD_{j,s}$	Initial stock of final product in distribution center	BoxU	PUN_INI	ICD
VPBM	$VPBM_{l,s}$	Speed production bottles per minute	Bottles/Min	LIN_PRF	VPBM
Parameter		Calculated Parameters			
ID	Algebraic Expression	Description (DES_PAR)	Unit (COD_UNI)	Algebraic Formulae (FORMULA)	
BOCU	$BOCU_s$	Bottles per box-unit	Bot./BoxU	$INV(FCUF_s) \times BOCF_s$	
CUPE	$CUPE_{e,s}$	Overtime production cost per product per boxu	$/BoxU	$1.25 \times CUPF_{e,s}$	
DMJN	$DMJN_{t,s,j}$	Product demand per distribution center (negative)	BoxU	$- DMJS_{t,s,j}$	
FPCF	$FPCF_{e,j,s}$	Freight packing plant to distribution center per pallet	$/pallet	$0.025 \times FLPC_{e,j}$	
FPCT	$FPCT_{e,j,s}$	Freight packing plant to distribution center per boxu	$/BoxU	$INV(FCUF_s) \times FSCF_{e,j,s}$	
FPHC	$FPHC_{l,s}$	Production time per packing line per final product	Hours/BoxU	$INV(VPBH_{l,s}) \times BOCU_s$	
FSCF	$FSCF_{e,j,s}$	Freight packing plant to distribution center per box	$/Box	$INV(FCPF_s) \times FPCF_{e,j,s}$	
UREL	UREL	Availability of extra hours (overtime) per line	Hours	ULPR × HELM	
UROL	UROL	Availability of ordinary hours per line	Hours	ULPR × HOLM	
VPBH	$VPBH_{l,s}$	Speed production per hour	Bottles/Hour	$60 \times VPBM_{l,s}$	

The secondary table PAR_IND stores the relationships between indexes and parameters. Alternatively, to create a table, a field can be included in master table PARAMETER; however, for purposes of standardization and visualization of the relationships, seems best create the secondary table (table 6).

Table 6. MMIS - Table: PAR_IND

Parameter (COD_PAR)	Index (COD_IND)
BOCF	s
BOCU	s
CUPE	e
CUPE	s
CUPF	e
CUPF	s
DMJN	t
DMJN	s
DMJN	j
DMJS	t
DMJS	s

Table 6. MMIS - Table: PAR_IND

Parameter (COD_PAR)	Index (COD_IND)
DMJS	j
FCPF	s
FCUF	s
FLPC	e
FLPC	j
FPCF	e
FPCF	j
FPCF	s
FPCT	e
FPCT	j
FPCT	s

Table 6. MMIS - Table: PAR_IND

Parameter (COD_PAR)	Index (COD_IND)
FPHC	l
FPHC	s
FSCF	e
FSCF	j
FSCF	s
ICD	j
ICD	s
VPBH	l
VPBH	s
VPBM	l
VPBM	s

- **VARIABLE**

The table VARIABLE (table 7) must have at least the six fields, that storages information about: i) ID (code), ii) description, iii) units, iv) type (real, positive, integer, binary), v) lower bound and vi) upper bound. In this case, all variables have the same units (COD_UNI): BoxUni.

Table 7. Mathematical Modeling Information Systems (MMIS) – Table: VARIABLE					
ID (COD_VAR)	Algebraic Existence Conditions	Description (DES_VAR)	Type (TYPE)	Lower Bound (LOW_BOU)	Upper Bound (UPP_BOU)
DFC	$DFC_{t,j,s}$ $\forall t\ \forall j \in PUN$ $\forall s \in PCA(j)$	Deficit of final product (unattended demand)	Real$^+$	0	∞
DPC	$DPC_{t,e,j,s}$ $\forall t\ \ \forall e \in ENV$ $\forall j \in ENP(e)$ $\forall s \in PPJ(e,j)$	Final product dispatch from packing plant to distribution center	Real$^+$	0	∞
ICD	$ICD_{t,j,s}$ $\forall t\ \forall j \in PUN$ $\forall s \in PCA(j)$	Stocks of final product in distribution center	Real$^+$	0	∞
PEE	$PEE_{t,e,l,s}$ $\forall t\ \forall e \in ENV$ $\forall l \in LIE(e)$ $\forall s \in PRL(l)$	Final product production in overtime	Real$^+$	0	∞
PLE	$PLE_{t,e,l,s}$ $\forall t\ \forall e \in ENV$ $\forall l \in LIE(e)$ $\forall s \in PRL(l)$	Final product production in ordinary time	Real$^+$	0	∞

It is important to note that the conditions of existence of the variable is not an academic custom, but these conditions facilitate the formulation of models with generic equations that are adjusted in accordance with these existence conditions. In addition, the existence conditions reduce the domain of variables, decreasing the size of the models and avoid problems of unbounded solutions, due to have variables not controlled by any constraint.

The secondary table VAR_IND (table 8) stores the relationships between indexes and variables, and the existence condition for each index must be associated with a set.

Table 8. MMIS - Table: VAR_IND		
Variable (COD_VAR)	Index (COD_IND)	Set (COD_SET)
DFC	t	
DFC	j	PUN
DFC	s	PCA
DPC	t	
DPC	e	ENV
DPC	j	ENP
DPC	s	PPJ
ICD	t	
ICD	j	PUN

Table 8. MMIS - Table: VAR_IND		
Variable (COD_VAR)	Index (COD_IND)	Set (COD_SET)
ICD	s	PCA
PEE	t	
PEE	e	ENV
PEE	l	LIE
PEE	s	PRL
PLE	t	
PLE	e	ENV
PLE	l	LIE
PLE	s	PRL

- **RESTRICTION/EQUATION**

The table RESTRICTION (table 9) must have at least the six fields, that storages information about: i) ID (code), ii) description, iii) units, iv) type (≥, ≤, =, ::), v) RHS, and vi) LHS. In this case, all equations have the same units (COD_UNI): BoxUni.

Table 9. Mathematical Modeling Information Systems (MMIS) – Table: RESTRICTION						
ID (COD_RES)	Algebraic	Description (DES_RES)	Type (TYPE)	LHS (LHS)	RHS (RHS)	Equations (EQUATION)
CPHE	$CPHE_{t,e,l}$ $\forall t\ \ \forall e \in ENV$ $\forall l \in LIE(e)$	Packing Line Capacity - Overtime	$\leq$		UREL	$\Sigma_{s \in PRL(l)}$ $FPHC_{l,s} \times PLE_{t,e,l,s} \leq$ UREL
CPHO	$CPHO_{t,e,l}$	Packing Line Capacity -	$\leq$		UROL	$\Sigma_{s \in PRL(l)}$ $FPHC_{l,s} \times PLE_{t,e,l,s} \leq$

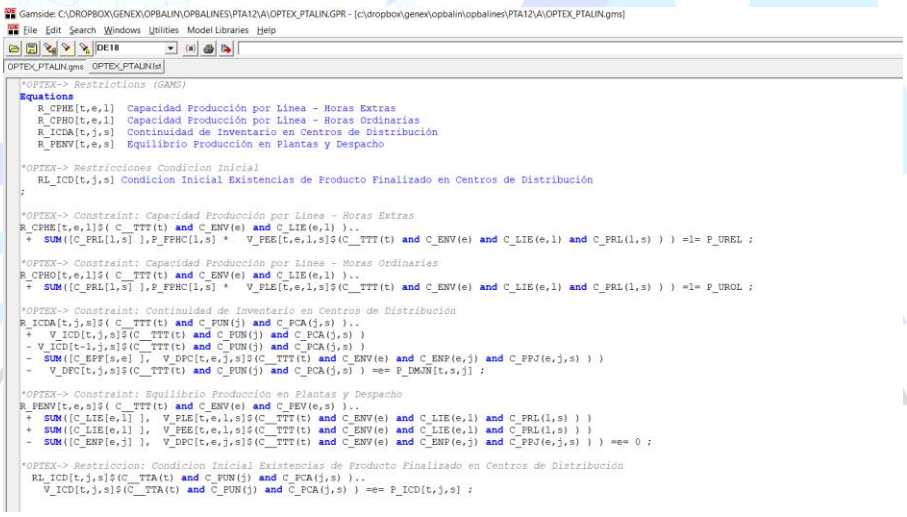
Table 9. Mathematical Modeling Information Systems (MMIS) – Table: RESTRICTION						
ID (COD_RES)	Algebraic	Description (DES_RES)	Type (TYPE)	LHS (LHS)	RHS (RHS)	Equations (EQUATION)
	$t \ \forall e \in ENV$ $\forall l \in LIE(e)$	Regular time				UROL
ICDA	$ICDA_{t,j,s}$ $\forall t \ \forall j \in PUN$ $\forall s \in PCA(j)$	Inventory continuity in distribution centers	=		DMJN	$ICD_{t,j,s} - ICD_{t-1,j,s} - \Sigma_{e \in EPF(s)} DPC_{t,e,j,s} - DFC_{t,j,s} = DMJN_{t,s,j}$
PENV	$PENV_{t,e,s}$ $\forall t \ \forall e \in ENV$ $\forall s \in PEV(e)$	Balance production in plants and dispatch	=		0	$\Sigma_{l \in LIE(e)} PLE_{t,e,l,s} + \Sigma_{l \in LIE(e)} PEE_{t,e,l,s} - \Sigma_{j \in ENP(e)} DPC_{t,e,j,s} = 0$

The secondary table RES_IND (table 10) stores the relationships between indexes and constraints, and the existence condition for each index must be associated with a set.

Table 10. MMIS - Table: RES_IND		
Restriction (COD_RES)	Index (COD_IND)	Set (COD_SET)
CPHE	t	
CPHE	j	PUN
CPHE	t	
CPHE	e	ENV
CPHE	l	LIE
CPHO	t	
CPHO	e	ENV

Table 10. MMIS - Table: RES_IND		
Restriction (COD_RES)	Index (COD_IND)	Set (COD_SET)
CPHO	l	LIE
ICDA	t	
ICDA	j	PUN
ICDA	s	PCA
PENV	t	
PENV	e	ENV
PENV	s	PEV

Based on the above tables, it is possible to define any mathematical problem using an algebraic formulation. The following GAMS screenshot (figure 9) presents the equations of the model PTALIN.



```
*OPTEX-> Restrictions (GAMS)
Equations
   R_CPHE[t,e,l]  Capacidad Producción por Linea - Horas Extras
   R_CPHO[t,e,l]  Capacidad Producción por Linea - Horas Ordinarias
   R_ICDA[t,j,s]  Continuidad de Inventario en Centros de Distribución
   R_PENV[t,e,s]  Equilibrio Producción en Plantas y Despacho

*OPTEX-> Restricciones Condicion Inicial
   RL_ICD[t,j,s] Condicion Inicial Existencias de Producto Finalizado en Centros de Distribución
;

*OPTEX-> Constraint: Capacidad Producción por Linea - Horas Extras
R_CPHE[t,e,l]$( C__TTT(t) and C_ENV(e) and C_LIE(e,l) )..
 +  SUM([C_PRL[l,s] ],P_FPHC[l,s] *   V_PEE[t,e,l,s]$(C__TTT(t) and C_ENV(e) and C_LIE(e,l) and C_PRL(l,s) ) ) =l= P_UREL ;

*OPTEX-> Constraint: Capacidad Producción por Linea - Horas Ordinarias
R_CPHO[t,e,l]$( C__TTT(t) and C_ENV(e) and C_LIE(e,l) )..
 +  SUM([C_PRL[l,s] ],P_FPHC[l,s] *   V_PLE[t,e,l,s]$(C__TTT(t) and C_ENV(e) and C_LIE(e,l) and C_PRL(l,s) ) ) =l= P_UROL ;

*OPTEX-> Constraint: Continuidad de Inventario en Centros de Distribución
R_ICDA[t,j,s]$( C__TTT(t) and C_PUN(j) and C_PCA(j,s) )..
 +   V_ICD[t,j,s]$(C__TTT(t) and C_PUN(j) and C_PCA(j,s) )
 - V_ICD[t-1,j,s]$(C__TTT(t) and C_PUN(j) and C_PCA(j,s) )
 -  SUM([C_EPF[s,e] ],  V_DPC[t,e,j,s]$(C__TTT(t) and C_ENV(e) and C_ENP(e,j) and C_PPJ(e,j,s) ) )
 -   V_DFC[t,j,s]$(C__TTT(t) and C_PUN(j) and C_PCA(j,s) ) =e= P_DMJN[t,s,j] ;

*OPTEX-> Constraint: Equilibrio Producción en Plantas y Despacho
R_PENV[t,e,s]$( C__TTT(t) and C_ENV(e) and C_PEV(e,s) )..
 +  SUM([C_LIE[e,l] ],  V_PLE[t,e,l,s]$(C__TTT(t) and C_ENV(e) and C_LIE(e,l) and C_PRL(l,s) ) )
 +  SUM([C_LIE[e,l] ],  V_PEE[t,e,l,s]$(C__TTT(t) and C_ENV(e) and C_LIE(e,l) and C_PRL(l,s) ) )
 -  SUM([C_ENP[e,j] ],  V_DPC[t,e,j,s]$(C__TTT(t) and C_ENV(e) and C_ENP(e,j) and C_PPJ(e,j,s) ) ) =e= 0 ;

*OPTEX-> Restriccion: Condicion Inicial Existencias de Producto Finalizado en Centros de Distribución
  RL_ICD[t,j,s]$(C__TTA(t) and C_PUN(j) and C_PCA(j,s) )..
    V_ICD[t,j,s]$(C__TTA(t) and C_PUN(j) and C_PCA(j,s) ) =e= P_ICD[t,j,s] ;
```

Figure 9. GAMS code of PTALIN Model – Equations - Generated by OPTEX

To have a model of optimization, it is necessary to include objective function, which may be associated with a variable defined in a constraint (as in GAMS) or it can be specify as a new entity with its own characteristics. In the case of including a new entity, the proposal is an objective function based on the summations of pairs of parameters multiplied by a variable plus the summations of another objective functions multiplied by a weighting factor. In this case the objective function (MCPD) is equal to the sum of three costs:

$$MCPD = CODI + COPR + DEFI \tag{2}$$

where

- Distribution Cost (CODI):

$$CODI = \Sigma_t\ \Sigma_{j\in PUN}\ \Sigma_{s\in PCA(j)}\ FPCT_{e,j,s} \times DPC_{t,e,j,s} \tag{3}$$

- Production Cost (COPR):

$$COPR = \Sigma_t\ \Sigma_{e\in ENV}\ \Sigma_{s\in PRL(l)}\ (\ CUPF_{e,s} \times PLE_{t,e,l,s} + CUPE_{e,s} \times PEE_{t,e,l,s}\ ) \tag{4}$$

- Deficit Cost (DEFI):

$$DEFI = \Sigma_t\ \Sigma_{j\in PUN}\ \Sigma_{s\in PCA(J)}\ \infty \times DFC_{t,j,s} \tag{5}$$

Three tables are required to store the above algebraic expressions (table 11):

Table 11. Mathematical Modeling Information Systems (MMIS) – Objective Function Tables

Table	Entity / Relations	Comments – Main Information Stored	Relational key
MASTER TABLE			
FUNC_OBJ	Objective function	Name of the objective functions: MCPD, CODI, COPR, DEFI	COD_FOB
SECONDARY TABLES			
VAR_FOB	Variable – Objective Function	Pair of parameters variables included in the objective function	COD_FOB COD_VAR COD_PAR
FOB_FOB	Objective Function – Objective Function	Objective functions included in the objective function	COD_FOB COD_FOB1

The table FUNC_OBJ has two fields (table 12):

Table 12. Mathematical Modeling Information Systems (MMIS) - Table: FUNC_OBJ

ID (COD_FOB)	Description (DES_FOB)
MCPD	Total Cost
CODI	Distribution Cost
COPR	Production Cost
DEFI	Deficit Cost

The contents of the secondary tables are presented below (table 13 and table 14):

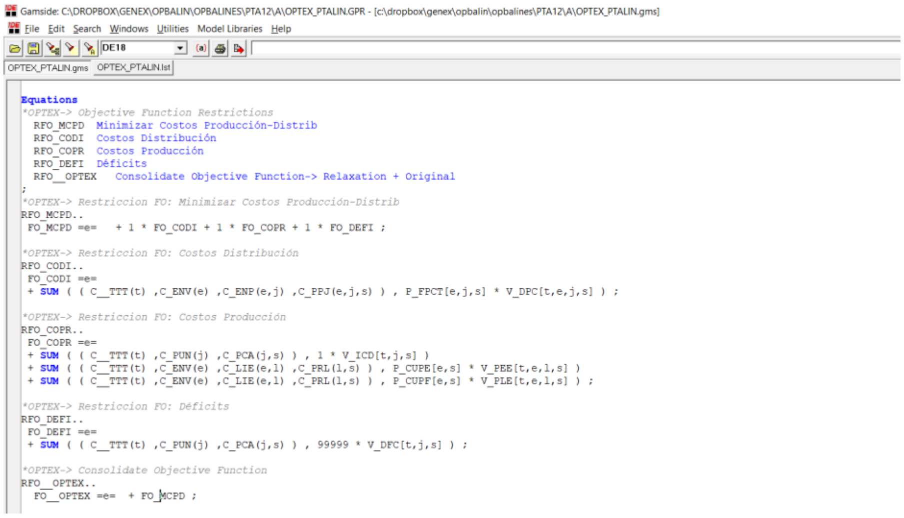
Table 13. MMIS - Table: FOB_FOB

Objective Function (COD_FOB)	Objective Function (COD_FOB1)	Weigth (WEIGTH)
MCPD	CODI	1
MCPD	COPR	1
MCPD	DEFI	1

Table 14. MMIS - Table: VAR_FOB

Objective Function (COD_FOB	Variable (COD_VAR)	Parameter (COD_PAR)
CODI	DPC	FPCT
COPR	PEE	CUPE
COPR	PLE	CUPF
DEFI	DEFI	99999

Based on the above tables, it is possible to define any mathematical problem using an algebraic formulation. The following GAMS screenshot (figure 10) presents the objective function equations of the model PTALIN.



```
Gamside: C:\DROPBOX\GENEX\OPBALIN\OPBALINES\PTA12\A\OPTEX_PTALIN.GPR - [c:\dropbox\genex\opbalin\opbalines\PTA12\A\OPTEX_PTALIN.gms]
File  Edit  Search  Windows  Utilities  Model Libraries  Help
DE18
OPTEX_PTALIN.gms  OPTEX_PTALIN.lst

Equations
*OPTEX-> Objective Function Restrictions
  RFO_MCPD  Minimizar Costos Producción-Distrib
  RFO_CODI  Costos Distribución
  RFO_COPR  Costos Producción
  RFO_DEFI  Déficits
  RFO__OPTEX   Consolidate Objective Function-> Relaxation + Original
;
*OPTEX-> Restriccion FO: Minimizar Costos Producción-Distrib
RFO_MCPD..
 FO_MCPD =e=   + 1 * FO_CODI + 1 * FO_COPR + 1 * FO_DEFI ;

*OPTEX-> Restriccion FO: Costos Distribución
RFO_CODI..
 FO_CODI =e=
 + SUM ( ( C__TTT(t) ,C_ENV(e) ,C_ENP(e,j) ,C_PPJ(e,j,s) ) , P_FPCT[e,j,s] * V_DPC[t,e,j,s] ) ;

*OPTEX-> Restriccion FO: Costos Producción
RFO_COPR..
 FO_COPR =e=
 + SUM ( ( C__TTT(t) ,C_PUN(j) ,C_PCA(j,s) ) , 1 * V_ICD[t,j,s] )
 + SUM ( ( C__TTT(t) ,C_ENV(e) ,C_LIE(e,l) ,C_PRL(l,s) ) , P_CUPE[e,s] * V_PEE[t,e,l,s] )
 + SUM ( ( C__TTT(t) ,C_ENV(e) ,C_LIE(e,l) ,C_PRL(l,s) ) , P_CUPF[e,s] * V_PLE[t,e,l,s] ) ;

*OPTEX-> Restriccion FO: Déficits
RFO_DEFI..
 FO_DEFI =e=
 + SUM ( ( C__TTT(t) ,C_PUN(j) ,C_PCA(j,s) ) , 99999 * V_DFC[t,j,s] ) ;

*OPTEX-> Consolidate Objective Function
RFO__OPTEX..
  FO__OPTEX =e=  + FO_MCPD ;
```

Figure 10. GAMS code of PTALIN Model – Objective Function - Generated by OPTEX

It is important to note, that the GAMS program associated with PTALIN was generated by OPTEX from the tables described as SMM-Basic. OPTEX can write the PTALIN model in many optimization technologies (AMPL, IBM OPL, MOSEL, GMPL , C, …). This checks that standardization is feasible and produces added value for mathematical modelers, and therefore for the end users.

3.1.2 Advanced Mathematical Modeling

Advanced Mathematical Modeling entities are used in the formulation of multi-problem mathematical models. These entities don't have a universal definition, for this document we define the following entities.

- Problem: set/group of constraints
- Model: set/group of problems
- Decision Support System (DSS): set/group of models

The basic case is a model defined based on an integrated problem.

The above objects are critical to addressing large-scale problems by coordinating multi-problem loop-models, some models that meet this feature are:

1. Parametric programing: the same problem solved for several values of their parameters; for example, it is used to build pareto frontiers or convex hull representations of the optimal response of a part of a bigger problem.
2. Families of problems: solution to multiple problems, dependent of one or multiple indexes; for example, Data Envelopment Analysis (DEA, Charnes et al., 1978).
3. Large-scale optimization: models solved using large-scale optimization methodologies like Benders Partitioning, Lagrangean Relaxation, …

The tables required for advanced SMM are presented below (table 15).

Table 15. Mathematical Modeling Information Systems (MMIS) – Advanced Modeling Tables			
Table	Entity	Comments – Main Information Stored	Relational key
MASTER TABLES			
PROBLEM	Index	Name of the problems: PTALIN	COD_PRO
MODEL	Set	Name of the models: PTALIN	COD_MOD
DSS	Parameter	Name of the DSSs: PTALIN	COD_DSS
SECONDARY TABLES			
PRO_RES	Problem - Restrictions	Restrictions incorporated in a problem	COD_PRO, COD_RES
MOD_PRO	Model - Problems	Problems incorporated in a model	COD_MOD, COD_PRO
DSS_MOD	DSS - Models	Models incorporated in a DSS	COD_VAR, COD_MOD

In this case, the model PTALIN has only one problem. The tables filled for this case are:

- **PROBLEM**

Tables that define the problem PTALIN are (table 16 and table 17):

Table 16. Mathematical Modeling Information Systems (MMIS) Table: PROBLEM		
Problem (COD_PRO)	Description (DES_PRO)	Format (FORMAT)
PTALIN	Distribution of products (drinks)	LP Linear Programming

Tabla 17. Mathematical Modeling Information Systems (MMIS) - Table: PRO_RES	
Problem (COD_PRO)	Restricction (COD_RES)
PTALIN	CPHE
PTALIN	CPHO
PTALIN	ICDA
PTALIN	PENV

Tables that define the model PTALIN are (table 18 and table 19):

Table 18. Mathematical Modeling Information Systems (MMIS) Table: MODEL		
Model (COD_MOD)	Description (DES_MOD)	Type (TYPE)
PTALIN	Distribution of products (drinks)	Integrated

Table 19. Mathematical Modeling Information Systems (MMIS) - Table: MOD_PRO	
Model (COD_MOD)	Problem (COD_PRO)
PTALIN	PTALIN

Keep in mind that there are multiple ways in which the problems that are part of a model, therefore more information is required for a multi-problem case. For example, the Benders' standard methodology has two problems: a coordinator and a subproblem, the tables should be able to store this information according to the role of the problems in the optimization process.

- **DSS**

Tables that define the decision support system PTALIN are (table 20 and table 21):

Table 20. Mathematical Modeling Information Systems (MMIS) Table: DSS	
DSS (COD_DSS)	Description (DES_DSS)
PTALIN	Distribution of products (drinks)

Table 21. Mathematical Modeling Information Systems (MMIS) - Table: DSS_MOD	
DSS (COD_DSS)	Model (COD_MOD)
PTALIN	PTALIN

The importance of DSS tables is the possibility of selecting a set of models, within all the models that may reside in an SMM, which are associated with a specific reason; for example, the sets of models that a math modeler has been sold to an end user.

3.1.3 IDIS Data Model

IDIS data model entities are used to define the tables of IDIS. From the point of view of mathematical modeling this part can be considered as complementary, not necessary, and therefore is not required to be part of the SMM; however, an application oriented to an end-user (a company) must have, in accordance with the current standards, an RDB to store input and output data of the mathematical models (figure 11), for this reason is provided in this document.

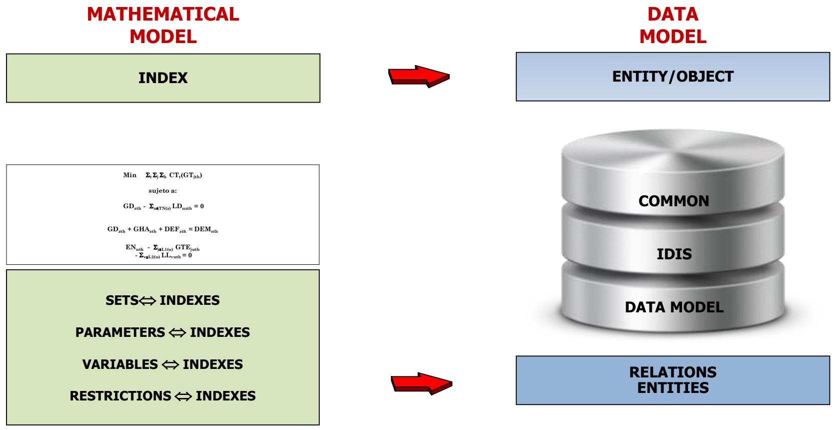


Figure 11. Optimization Model → Industrial Data Information System Data Model

It can be said that the data model of the IDIS is strongly linked to the structure of the mathematical models.

1. The indexes of the models correspond to IDIS entities; then, each index, except the time, must have a master table and a relational key
2. Relationships are defined by the indexes associated with: i) read datasets (independent indexes plus the dependent index), ii) read parameters, iii) variables and iv) restrictions.

Following these rules, it is possible to generate automatically a “draft” of the RDB required by the end user to store data related to its mathematical problems.

Below, the tables that integrate IDIS for PTALIN model are presented, including their relationship with read sets and read parameters (table 22 and table 23).

- **MASTER TABLES**

Table 22. Industrial Data Information System (IDIS) – Master Tables

Table	Description	Relational Key	Secondaries Keys	Sets	Parameters
MAE_ENV	Packing Plants	COD_ENV	COD_FAB	ENV	
MAE_LIN	Packing Lines	COD_LIN	COD_ENV	LIE(e), EVL(l)	
MAE_PRF	Final Products	COD_PRO	COD_TPR		
MAE_PUN	Distribution Centers	COD_PUN		PUN	

- **SECONDARY TABLES**

Table 23. Industrial Data Information System (IDIS) – Secondary Tables

Table	Description	Secondaries Keys	Sets	Parameters
DEMCDA	Distribution center - Demand	DATE, COD_PRF, COD_PUN	PCA(j)	$DMJS_{t,s,j}$
ENV_PRF	Packing plant - Final product	COD_ENV, COD_PRF		$CUPF_{e,s}$
ENV_PUN	Freight packing plant -> Distribution center	COD_ENV, COD_PUN	ENP(e)	$FLPC_{e,j}$
LIN_PRF	Packing line - Final product	COD_LIN, COD_PRF	PRL(l), LPF(s)	$VPBM_{l,s}$
PUN_INI	Distribution center – Initial stock	COD_PRF, COD_PUN		$ICD_{j,s}$

3.2 SMM Disadvantages

The disadvantage of SMM is the resistance to the paradigm shift.

Mentally, the mathematical modelers have been educated to link a mathematical model to an optimization technology using a computer program that they must write. In an optimization project, once the math modeler has an idea about the mathematical model, the modeler begins to order all his concepts around a computer program in a specific technology. This process in most cases does not have rules that guide the programming; the result is clear, a mathematical model "smelted" in a computer program using a specific optimization technology, that can hardly be understood by another math modeler and that it creates a barrier to its easy integration with new models.

In many courses in the universities, the professor teaches optimization technologies rather than mathematical modeling. Unfortunately, as time passes, the knowledge is accumulated in a specific technology; then, the resistance to change increases.

The first step to change to the SMM paradigm implies differentiate the mathematical model of the optimization technology.

3.3 SMM Advantages

Below, the advantages of SMM are presented:

1. Differentiate the mathematical model of the Mathematical Programming technology.
2. In the early stages of a project, the focus is the model, not the optimization technology.
3. MMIS, as any RDB is organized and normalized. This ensures control of models developed for companies, since any connoisseur of mathematical modelling can know the structure of the reference model developed using.
4. SMM facilitates teamwork, since tables can reside on any client-server RDB and therefore modelers groups can work concurrently (in many remote places, using the internet); the team must compliance the internal rules of the organization/enterprise to give names to the codes of the mathematical elements. The services of the RDB-server can be used to ensure the coherence of the RDB based on the validation of the referential integrity, as it is done in any RDB. This avoids errors and that the math modeler imprints with his "brand" the models that he builds.
5. Given that SMM is a data-model implemented in tables (tables 1 to 21), it is easy to make models (simples or complex), since, the data-model may be developed in spreadsheets or RTF files. The figure 12 presents the process implemented in OPTEX to facilitate development from MS-WORD and/or MS-EXCEL filling templates, the process is: i) algebraic model in the blackboard/paper, ii) algebraic formulation in templates in MS-WORD, iii) algebraic formulation in tables in MS-EXCEL, iv) algebraic formulation in CSV format an ANSI standard, v) import the CSV files to OPTEX-MMIS, vi) write computer algorithm of the model in an optimization technology.

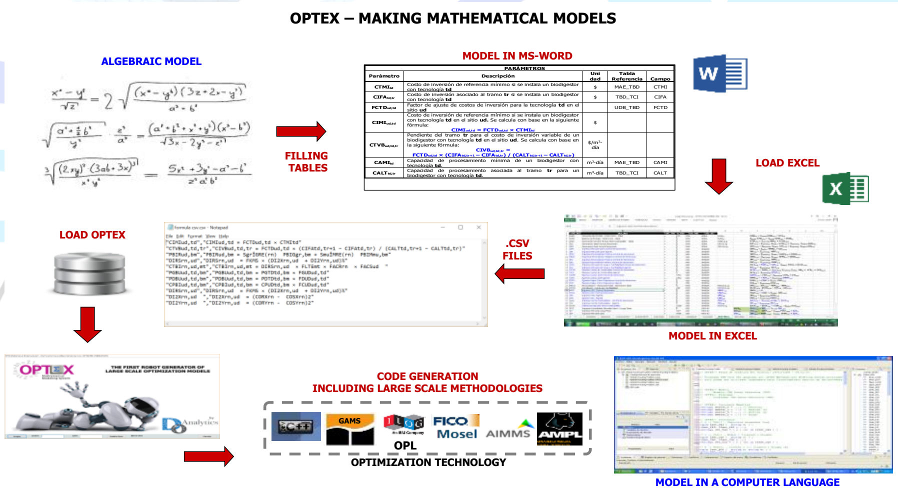


Figure 12. OPTEX Workflow to Make an Optimization Model

6. If the SMM becomes a global standard, the structure of the tables would be known; then, any interested person/company may develop a "handler" for your own service, or to buy/sell it in the market of optimization technologies. The current producers of optimization technologies should develop their own "handlers" to ensure the import/export of the mathematical models to SMM standards.
7. SMM may act as an optimization expert system that capture knowledge and experience storing the mathematical components that work correctly in the MMIS, so that it is not necessary to rewrite them. Then, build a mathematical model may be as build a "LEGO" building, selecting the proper constraints in the constraint's "store".

8. Visualize the mathematical model as a set of entities and relationships associated to an RDB facilitate to revise the models. The following shell window (figure 13) displays all existing relationships between the indexes and the entities used in the basic mathematical formulation, this allows the mathematical modeler to explore all the elements of a mathematical model and to verify the consistency of the content of the MMIS.

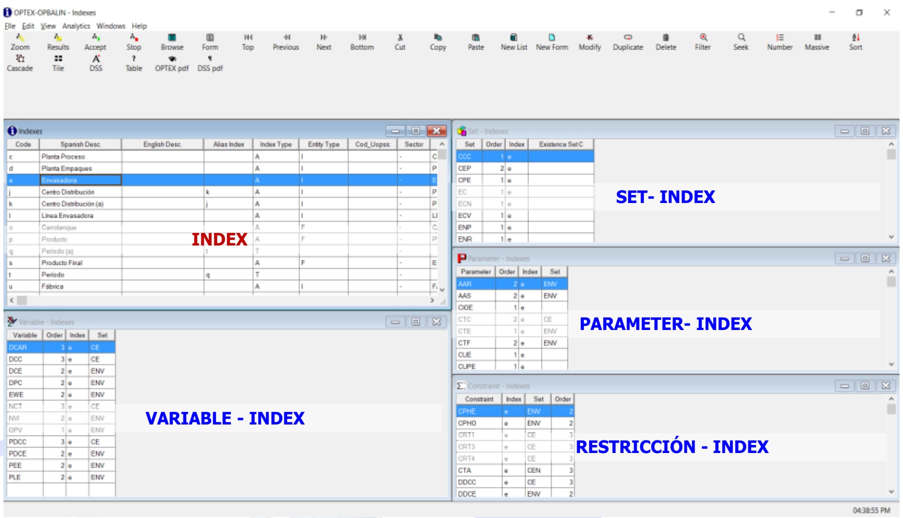


Figure 13. INDEX Shell Window Generated by OPTEX

9. If the MMIS is built respecting the rules of referential integrity of an RDB, the resulting IDIS the most suitable link to interconnect the models of decision-making chain associated with the DSS. The figure 14 shows the integration of models of a DSS for the electric sector as part of the SMM approach.

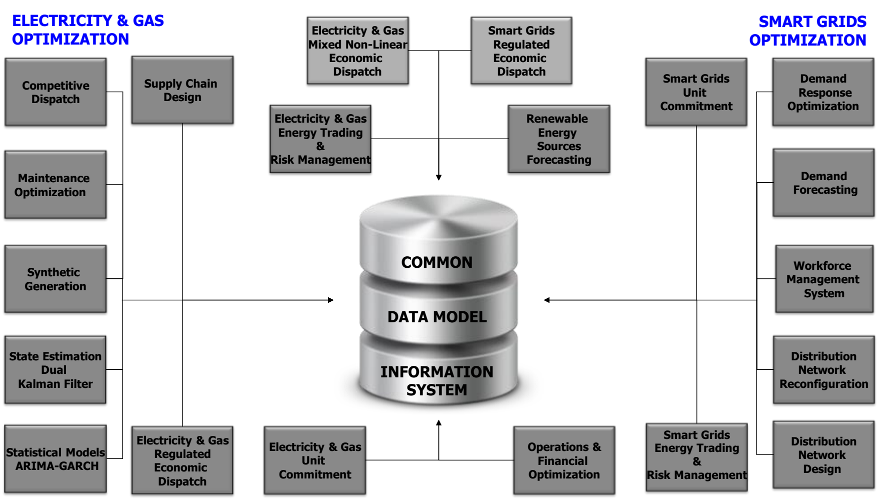


Figure 14. Optimization Models Linked by the Industrial Data Information System

4. Smart Algorithms that Make Advanced Analytical Algorithms

4.1 Advanced Analytics Professionals

A robot is an artificial agent, meaning it acts instead of a person, doing things. Robots are usually machines controlled by a computer program or by electronic circuitry. The robot can be a physical mechanical mechanism and/or a virtual software system.

Consistent with the development of the Artificial Intelligence (AI), automation has come to stay in the field of the Advanced Analytics (the commercial evolution of Operation Research), where analysts and modelers will receive help of robots to do their job; Davenport and Harris (2007), in their seminal book "Competing on Analytics", displayed three types of professionals involved with Analytics: i) Amateur, ii) Semi-professional and iii) Professional, twelve years after the publication of the book, comes a new type of professional: the "robotizer", professionals that make algorithms which, in turn, make advanced analytical algorithms, this speed-up the process of use of Advanced Analytics for those organizations that believe in it, and therefore opening more gap with those who do not believe.

It is evident that the robotics has already advanced in the process of using advanced analytic tools; analysts of data (input and output) of the mathematical models gradually were replaced by automatic processes (robots) that replace them; the reasons are: speed and accuracy. This is not new, Revenue Management systems many years working in an automated manner in relation to models that explain the elasticity demand-price and its subsequent application to optimization models, this is due to the number of models implementing pricing models in stores with large number of SKUs. The solution: computers dedicated to run mathematical models all time and checking and validate its results; it must be a standard in short term.

4.2 Industry 4.0 Revolution

“Industry 4.0 is a name given to the current trend of automation and data exchange in manufacturing technologies. It includes cyber-physical systems, the IoT/IIoT, cloud computing and cognitive computing. Industry 4.0 is commonly referred to as the fourth industrial revolution. Industry 4.0 fosters what has been called a "smart factory". Within modular structured smart factories, cyber-physical systems monitor physical processes, create a virtual copy of the physical world and make decentralized decisions. Over the IoT/IIoT, cyber-physical systems communicate and cooperate with each other and with humans in real-time both internally and across organizational services offered and used by participants of the value chain” (Wikipedia).

The Cognitive Robot (CR) is fundamentals for Industry 4.0, it is based on concepts of AI, that writes advanced analytics algorithms that are required for the digital transformation of enterprises, CR automatically linking them to the enterprise information system; in summary, CR is a skilled robot that creates robots for complex processes using advanced mathematical methodologies (state-of-the-art). This robotization process is at the highest level of automation, because it does not replace manual human work but supports the construction of robots replacing human cognitive tasks, related to the optimization modeling of stochastic processes and/or business/industrial processes.

CR increases productivity of mathematical modeler; understanding productivity such as: make more models in less time and ensuring the quality of the produced algorithms. To develop CR is necessary the Structured Mathematical Modeling (SMM), this makes CR independent of industrial mathematical technologies. As well as in the manual work robots enhance human ability, in the cognitive process, robots promote knowledge, systematized the cognitive tasks that are repetitive, like: i) write programs (in at least one optimization technology), ii) check data of IDIS, iii) data analytics, iv) check the mathematical formulation store in MMIS, … , all these free of errors. Then CR speed-up the development times; changes in a model that works properly are implemented in minutes/hours.

There are at least two ways to orient these robots:

i) To select an algorithm from a set of prototype algorithms as the best (but somebody writes the prototype algorithm). This is the way selected for many informatic tools used in Artificial Intelligence (this is the case of Python, that facilitates the access to many algorithm libraries) ; and

ii) To write the algorithms directly; it is necessary for if we need to developed algorithm for more oriented algebraic modeling, like Mathematical Programming. This document is oriented to this type of algorithms.

5. Optimization Knowledge Expert Systems

In artificial intelligence, an Expert System (ES) is a computer system that power and help the decision-making ability of a human expert. An ES is a knowledge-based system that uses a knowledge-based architecture where the knowledge base represents facts about the world.

The inference engine is an automated reasoning system that evaluates the current state of the knowledge-base, applies relevant rules, and then asserts new knowledge into the knowledge base. The inference engine may also include abilities for explanation, so that it can explain to a user the chain of reasoning used to arrive at a conclusion by tracing back over the chain of rules that resulted in the assertion.

An important aspect of modeling of complex systems is the presumption that optimization should be done in a single pass, in which the optimization model starts from zero and reaches the optimum in one step; in a lot of cases, the time available is insufficient to solve the complex problem with the required precision. This was valid when processing capacity and RAM and disk capacities were a scarce resource, it is not true today. Today it is possible to have idle computer processing capability, or can be rented at a low cost, this implies the change of the concept of starting from scratch, to pre-preprocessing before during the time the mathematical model is not required.

The basic idea is that scheduling, routing or real-time optimization applications never optimize from scratch, perhaps the first time, but once launched an optimization application can be conceived as a permanent process of re-optimization which can occur at any time. This fact implies the need to create an RDB to store the optimization results for use in the future we called this function as Optimization Expert System (OES).

OES is the way to capitalize of the experience acquired in previous optimization process, so that each new optimization involved in process starts reading/uploading the knowledge stored in database. The database may be in a disk or in RAM memory, and the "next" optimization will occur in the next second or in the next day or … ; it is a general concept. OES may contains information for, at least, three ways:

1. Start points: selecting the start point based on the history of runs and considering the differences with the new run. The easiest idea is to use the optimum point of the last run as a starting point for the next. This is already used in many matheuristics used in complex real-life problems. Lara et al. (2018) realized experiments to test the speed-up of the warm start.
2. Cutting planes that constraint the optimal-feasible zone based on previous runs of a model, it is applicable for many LSOM.
3. Optimal convex (or non-convex) hull that synthesize the optimal behavior of parts (sub-systems) of a complex system, making the model more "light".

Bellow, the two last alternatives are presented.

5.1 Cutting Planes

This implies to include in OES an RDB to manage the cuts produced by a LSOM, it may imply the following process: i) cleaning database eliminating cuts, ii) aggregate cuts and iii) subrogate cuts. Rules of management of the database depends on the LSOM used and it is a matter of research, having as reference the rules implemented in actual algorithms to prevent unlimited explosion of unnecessary cuts.

Optimization processes should involve the experience gained in past optimizations; for example, BT cutting planes generated until an optimization k may include as initial cutting planes in the next optimization, or re-optimization, k+1. But, a good alternative may be to include an only one subrogate cut that is equivalent to all cuts used in the last iteration of the run k, following the theory developed by Greenberg and Pierskalla (1970, Velasquez 1986). The figure 15 describes the process.

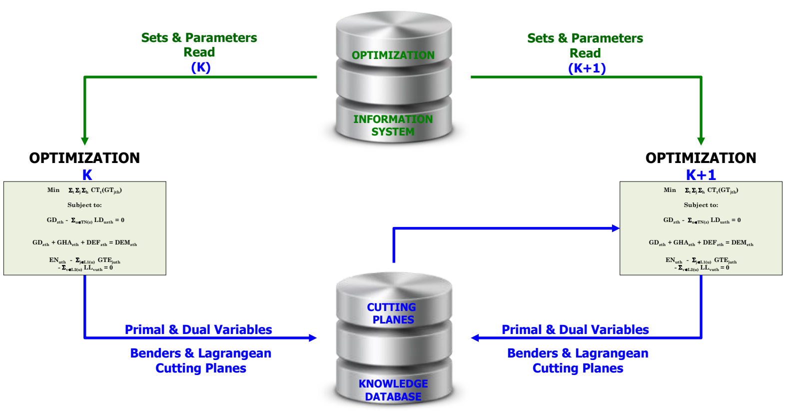


Figure 15. Management Cuts – Warm Start

5.2 Convex Hull

Optimal Convex (or non-convex) Hull (OCH) is oriented to synthesize the optimal behavior of parts (sub-systems) of a complex system, and to replace the equations of the subsystems in original model by the equations that represent the OCH. The figure 16 shows the approach that includes:

1. The model of the subsystem

2. The generation of the OCH and store the results in the OES-RDB.
Include the OCH equation in the original model and solve it

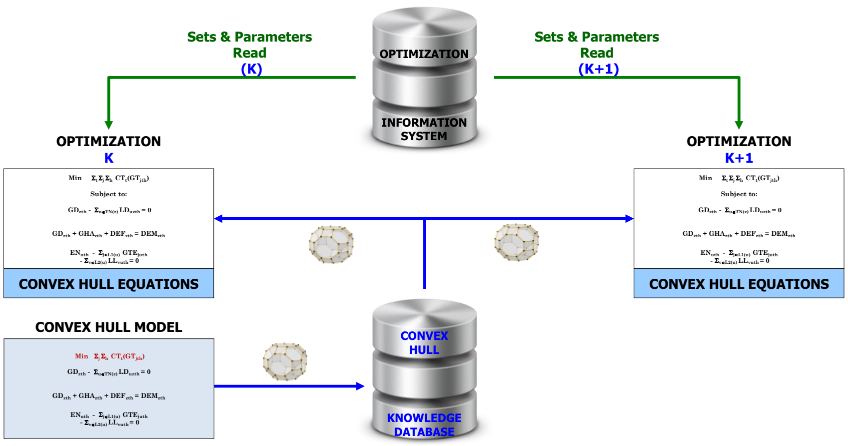


Figure 16. Management Convex Hulls

An example can be the "real time" optimization of pipeline operations involving the integration of pumping stations (where must be selected the pumping patterns, parallel or serial, and the technical specifications of the operation: pressure and flow) and displacement of multiple types of oils along the pipeline (where energy losses have a non-linear behavior with respect to the speed), which corresponds to a non-linear non-convex mixed problem of high mathematical complexity.

The equations related to the pumping stations may be replaced by the non-convex hull (NCH) that represents the optimal operating conditions for “all”' combinations of pressure and flow for each type of oil (API). Then the energy (HP) required can be pre-calculated and storage before running the scheduling/re-scheduling problem, this implies to define a table with values of non-convex hull, HP=NCH(Q, P, API). The figure 17 shows the grid associated with NCH of a pumping station for a type of oil. The equations that represent the NCH correspond to an interpolation in two dimensions, due to the non-convexity, binary variables are required for their correct representation.

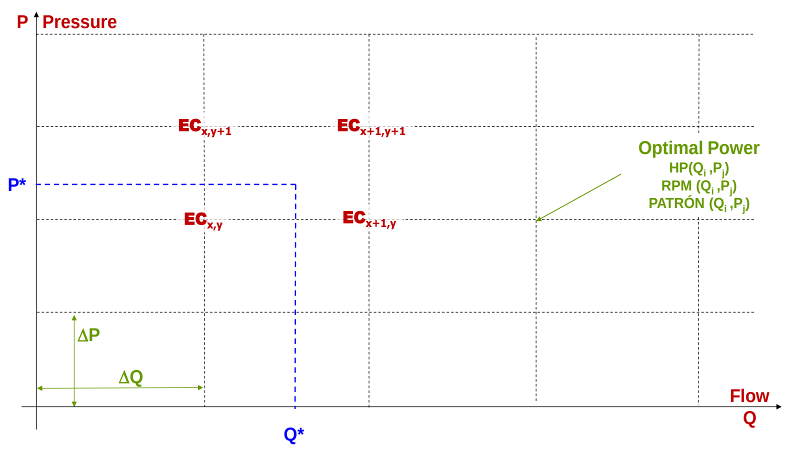


Figure 17. Convex Hull - Pumping Station Operation

The new problem is lighter than the integrated problem and it can meet an optimal solution in reasonable times for the real-time optimization problem. The calculation of the non-convex hulls can be done during the idle time of the computer system. More information of this case can be found in the paper Oil Pipelines Real-Time Optimization (Velásquez, 2019b).

6. Asynchronous Parallel Optimization

Asynchronous Parallel Optimization (APO, Velasquez, 1995, 1997) is defined as act of solve an optimization problem using multiple cores in a computer, or in a grid of computers, using the moderns multiprocessing environments; joining the decomposition and the partition large-scale theories (Benders Theory & Lagrangean Relaxation) that permits structure complex multilevel mathematical models; these systems are characterized by the set of problem families that they include. For this section it is convenient to define two terms:

1. Partition: the action of dividing a problem into two subproblems establishing a hierarchical relationship between them.
2. Decomposition: the action of dividing a problem in to multiple subproblems with the same level in a hierarchical scale.

These definitions are valid in this section, but the reader is caution that they are not universal definitions.

6.1 Implementation of Parallel Optimization

First of all, the modeler should be aware that to develop this type of application requires knowledge about the formalities which must be considered to implement applications of parallel computing of any kind; for example, the fundamentals of the implementation of a DCS (Distributed Control Systems) or of a SCADA (Supervisory Control And Data Acquisition) may help, these computer industrial systems implies the permanent communication between many task that altogether assume the integrated management and control of the industrial system in a multi-tasking environment.

The possibility of parallelism in BT is directly associated with application of decomposition to the problem. We will consider two cases: the application to problems: i) two levels and ii) multiple levels. In the figure 18, BT parallel applies to decoupled cuts and to unified cuts (Velasquez 2019d), but it does not apply to standard Benders cuts. In the case of decoupled cuts, the problems may belong to the same dimension (e. g. time, random scenarios) or to different dimensions (e. g. oil, gas and electricity)

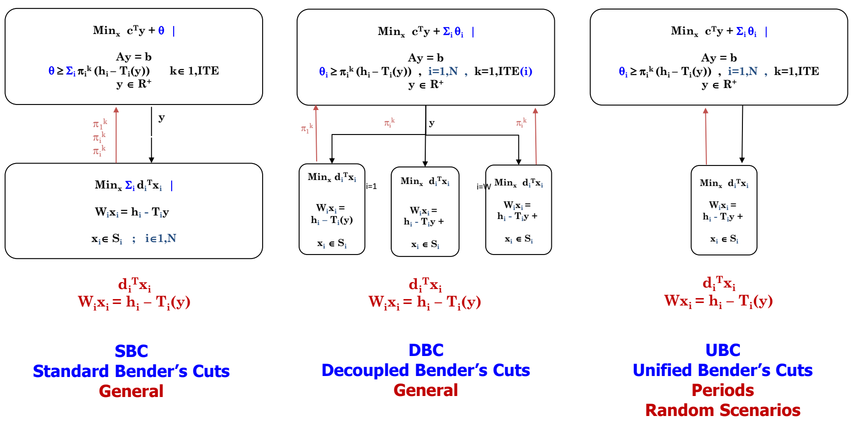


Figure 18. Benders Decomposition Type of Cuts

6.2 Framework

Joining the decomposition theory and the multilevel partition theory can be structured complex multilevel models

1. Interconnected Electrical Systems: integrated by the hydraulic, electric and the gas sectors
2. Integrated Energy Systems: integrated by electric, oil, coal, gas, consumer and external sectors
3. Global Multi-Business Industrial Supply Chains: multi-echelon supply chains integrated by factories, distributions centers and market located in multiple regions/countries for many complementary products. An example is the oil supply chain integrated by exploration, production, transportation and refining sectors.

Another reason to break down systems is related to the functions of the decisions; an example is:

1. Strategic: related with the expansions of the supply chain, in the long term.
2. Tactical: related with the plan (goals) of the operations of the supply chain, in the medium term.
3. Operations: related with the actions and the technical specifications (specs) of each installation/area of the supply chain, in the short term.

In this case, models of different hierarchy must be interconnected to coordinate the correct evaluation of the projects and their subsequent execution. The links may be marginal costs (dual variables) and/or border conditions (primal variables). These systems are characterized by the set of families of problem that they include. In the next diagram each color represents a type of optimization problem related with a physical installation and/or decision level.

The following diagram presents a multilevel decomposition and partition scheme. The process followed is:

1. At the top level is the partition by the functionality of decisions: it corresponds to investments in expansion and simulation of operational decisions, which depend on multiple scenarios of the decision-making environment.
2. In the next step the system decomposition is done by random scenarios. This gives rise to a two-stage stochastic optimization model, known as L-Shape (Van Slyke and Wets, 1969)
3. Given that the system is multi-sector (for example, the energy sector: electricity, oil, gas and biofuels) it is possible to decompose the system in as many subproblems as couples <sectors-scenarios> exists. The Benders cuts may be decoupled, solving separate each subproblem.
4. To decrease the size of the subproblems, it is possible to make a new decomposition, in this case by zones.
5. Finally, a new decomposition can be based on the periods of the planning horizon.

These systems are characterized by the set of problem families that they include; each color represents a type of optimization problem related with a physical installation and/or decision level(figure 19); this concept will be used later. The main advantage partition/decomposition approach is related to the abundance of opportunities for parallelism and atomization of the mathematical problem, involving several alternatives to address the mathematical problem solution; as a counterpart, the selection of the optimal alternative is a new research topic that must face the mathematical modeler.

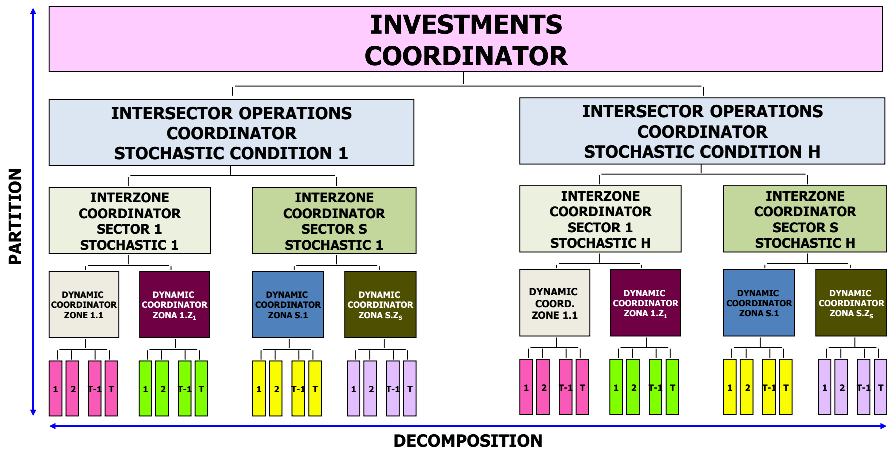


Figure 19. Multilevel Benders System

6.3 Optimization Database

A fundamental point of the process of parallel optimization is the communications between the tasks which can be:

1. Messages: between tasks using peer-to-peer approach or any type of messages services, like internet services.
2. Database: all the information is stored in RAM or in disk; the tasks access to the database to upload the data produced for other task and to /download the results. To inform the tasks the availability of new information, one of the alternatives, is to implement a system semaphores that make that tasks access the data base the information when it is ready. This is the approach suggested here.

The figure 19 shows an example of four processors used in parallel optimization, three assigned to subproblems and one to the coordinator; the communication of tasks is done by a database in RAM memory (ideal for speed, but it may also be on a disk), which we will call “Optimization Database”.

The concept of a database as a way of data exchange is widely used, as example is the case of the DCSs or the SCADAs that share a database in memory, where all the measurements carried out in the industrial system are stored, this database usually is referred to as Real-Time Database. In the case of optimization, data producers are the optimization problems which, in turn, are provided of data produced by other problems. The generated data depend on: i) the type of problem, ii) the large-scale methodology and iii) the partition/decomposition scheme.

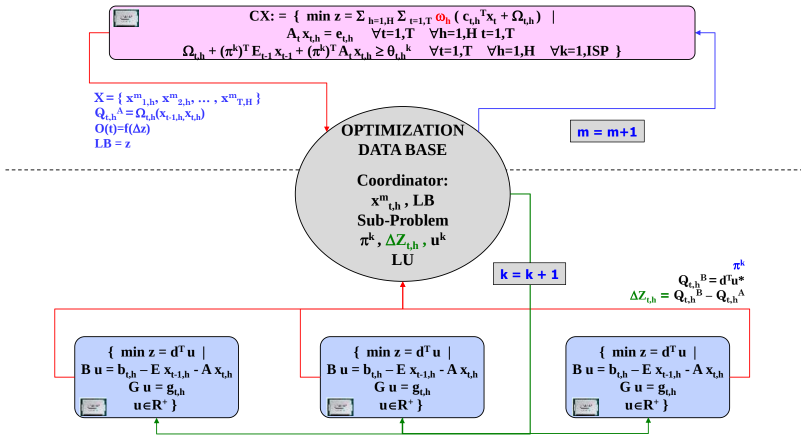


Figure 20. Parallel Optimization

In the case of BT, the subproblems produce dual variables and receive primal variables; if the large-scale methodology is LR (Lagrangean Relaxation), the subproblems receive dual variables and generate primal variables. If methodology implies generalized cuts, such as Generalized Benders Decomposition (Geoffrion, 1972) the subproblems produce dual and primal variables. In multilevel BT, the problems of intermediate levels receive primal variables from the upper level and dual variables from the lower level.

Van Roy (1983) lays down the principles for use simultaneous large scale such as BT and LR methods. An example of DC can be the partition and decomposition of the coordinator of the multilevel model analyzed in the previous section. In that case, it is possible to apply RL divided the BT coordinator (which defines investments) in two problems: a Lagrangean relaxed coordinator and a subproblem that can be decomposed in many slave-subproblems; in the case of a multi-sectoral model, each subproblem represents the investments in a sector. To solve the model all subproblems, coordinator and slaves, must interchange information based on primal and dual variables that generates cutting planes.

In general, the optimization knowledge must store results of the primal and dual variables of the optimization problems, linked to the loop iterations (figure 21). This approach is generic, independent of the content of the problems; so, it can be generalized to many cases in such a way of handling standardized schemas of parallelization.

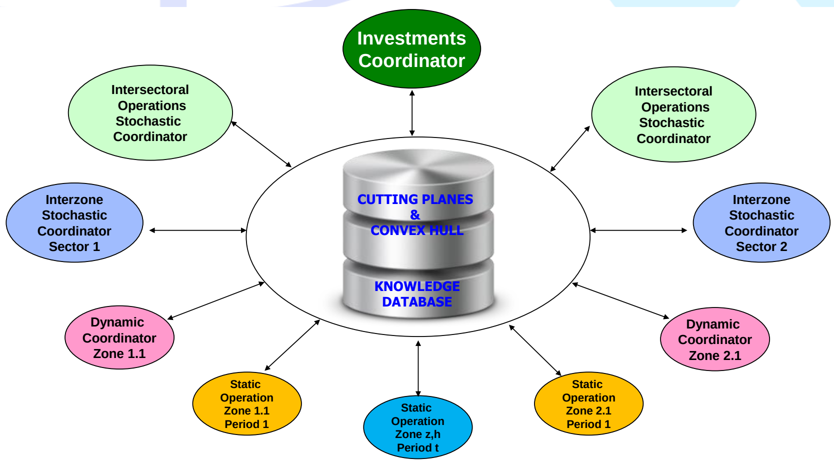


Figure 21. Optimization Knowledge Expert System

7. Real-Time Distributed Optimization

Real-Time Distributed Optimizacion is the distribution of the optimization process in many agents that act simultaneous and independently when they received information from the its exogenous environment. The process can be summarized in the following steps:

1. From a top-down analysis mathematical is possible to construct mathematical or logical rules of interaction

between multiple agents (representing each part of the system), which can represent the "state of the reality",

2. Starting from the math/logic rules, following an approach bottom-up, is possible to build segmented/atomized models of the real-world.

Using asynchronous optimization processing, it is possible to define the actions of an agent that keep the system on the "optimality path" in a cyber-physical system.

7.1 Distributed Optimization as An Artificial Smart Neural Net

Making a parallel with neural nets, the concept of problem family can be assimilated to a "neuron" class; and the optimization process can be defined as a complex communications system between "smart neurons". Basic neurons (perceptron, figure 22) are based on perceptions that are added and processed in order to explain the behavior of the system based on adjustment of observations to history, but without ability to internally process the signals receiving.

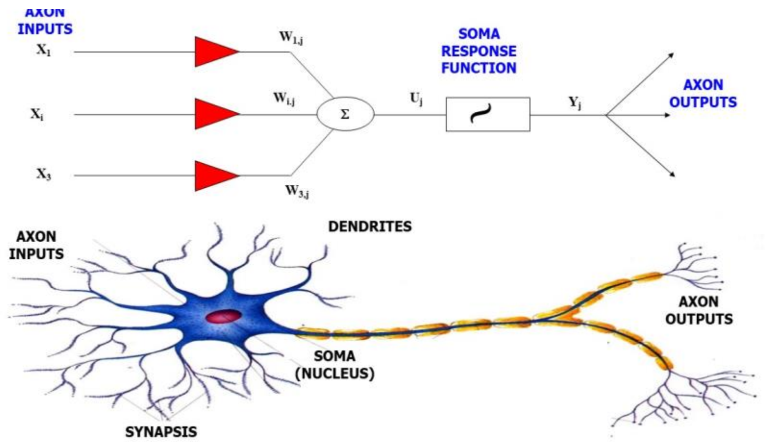


Figure 22. Structure of a Basic Artificial Neuron

If we define a type of problem such as a smart neuron (figure 23), which has autonomous capability, based on "universal" mathematical laws, to process the inputs from the environment (primal variables in the case of BT and dual variables in the case of RL) and produce the necessary information for another type of neuron processing, then we have a new type of neural net: a smart neural net, that is based in mathematical laws and not only in the perception of the historic data.

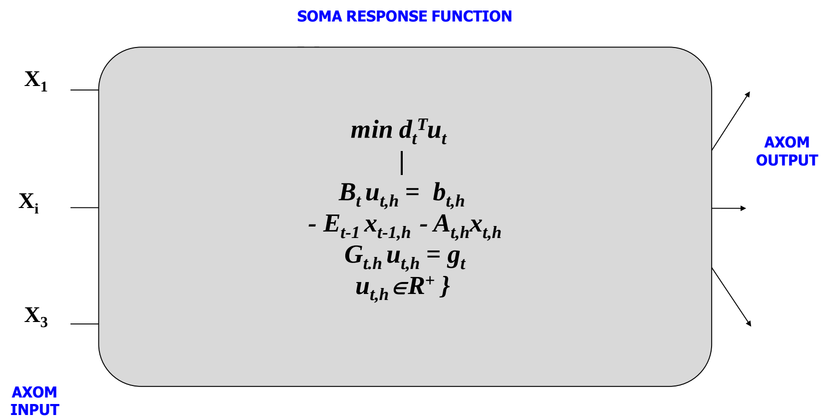


Figure 23. Structure of a Smart Artificial Neuron

As was noted previously, different types of problems that make up a system based on multilevel partition and decomposition of physical systems. This means that the structure of the smart neural network corresponds one-to-one with the parts of the physical system and it is not the result of an empirical process in which many structures are tested as part of the analytical work oriented to determine the best structure that represents/perceives the system. For example, in a supply chain system (figure 24), all the relations between neurons are defined by the partition/decomposition protocols (figure 25).

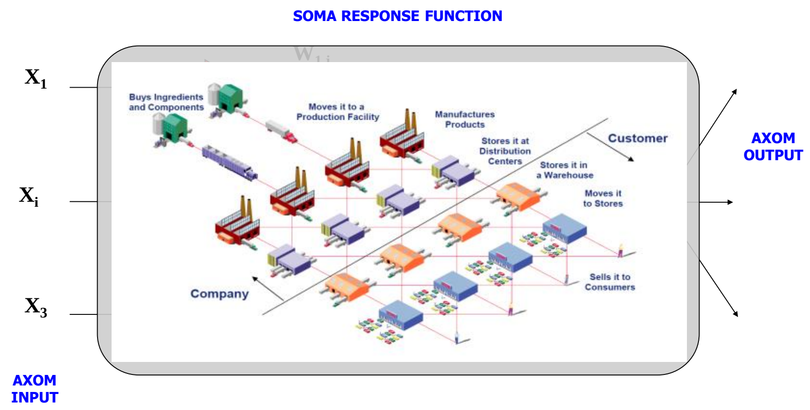


Figure 24. A Supply Chain Model

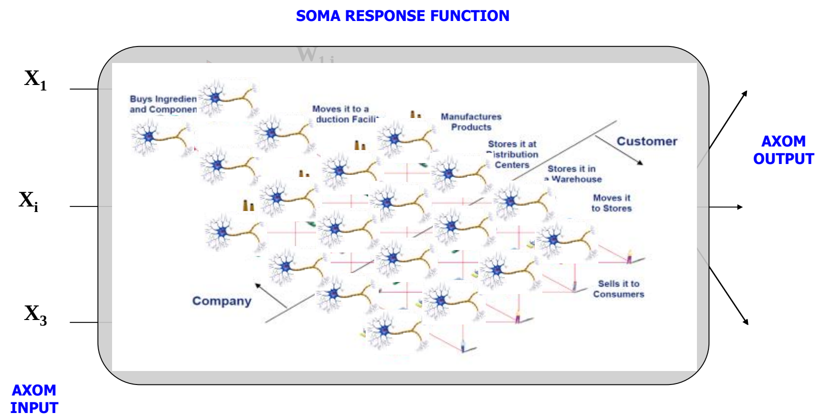


Figure 25. A Supply Chain Model as A Smart Neuron Network

Perceptions-based neural nets are appropriate for describing static processes, e.g. the recognition of a pattern (image, letter, object,...), intelligent neural nets can handle dynamic processes, using the mathematical laws that support them.

7.2 Real-Time Distributed Optimization

In many systems, the traditional optimization is based on the synchronized use of optimization models that run periodically (hourly, daily, weekly, monthly, quarterly, …) and whose information is broadcasted to all components of the system, so, that they will act autonomously until the next time that the mathematical model will be run. Real-Time Distributed Optimization (RT-DO) is topic open to researchers in Mathematical Programming and it is the link to coordinate the cyber-physical systems

New technologies and the large amount of data generated permanently (big-data) change this view to an optimization that should be based on events: the models will be run, autonomously, when it is necessary, by events. This implies that each component of the system must know which information that it needs to take of the available measuring systems (smart metering) and what is the information that it should be provided so that other components of the system can make its decisions oriented to keep system in the “optimality path”. We considered three cases.

7.2.1 Industrial Supply Chain

An example of distributed optimization may be a company that owns N production plants, each plant witn various production lines (figure 26) standard BT, whereby the subproblem is decomposed into N subproblems, one for each plant. It is easy to check that the Benders cuts, decoupled, represents a function of production for each subproblem (Velásquez 2019d). This information can be stored in the optimization database and serves to make runs at any time for adjusting the plan/schedule, “automatically”.

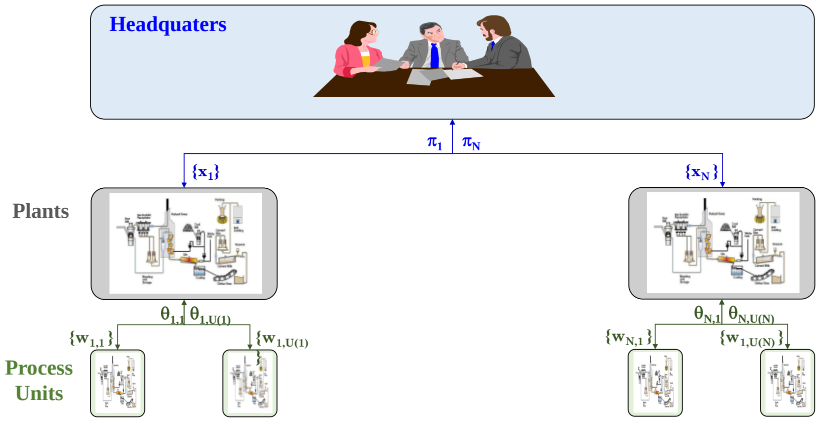


Figure 26. Benders Multi-Plant Industrial System

Consider examples for two cases:

1. A plant out of operation, due to a catastrophic event (tsunami, landslide, strike, fire,...). The re-optimization may be considered, as warm start points, the cutting planes (production functions) of the remaining N-1 plants that must be stored in OES-RDB;
2. A plant undergoes changes in its industrial infrastructure, leaves a production unit (for example, corrective maintenance). In this case, during the event, the production function represented by the cutting planes ceases to be valid. Then, the history of primal variables sent by the coordinator can be extracted from the last OES-RDB and from this information to quickly build an adjusted production, and thus make the whole industrial system optimization. If the problem associated with each plant is decomposed in multiple process units, the optimization of the affected plant could also achieve simply by removing the affected process unit.

This analysis can be extended to "any" system, to facilitate the reoptimizing process.

7.2.2 Smart Grids

An example of the need of RT-DO is the optimization of the performance of actual intelligent power supply networks (smart grids). To achieve the optimality of the new systems of electricity, ideally it is necessary to optimize simultaneously, in an integrate model, all the smart grids that make up the power system. In practice, this is impossible to achieve, the reasons are non-enumerable; however, it is possible, based on the study of the type of power components (smart grids) to establish the communication rules between components to achieve optimality. The figure 27 illustrates the concepts.

In public systems, the most complex part of this process may be the agreement between the parties oriented to act cooperatively to maximize the social surplus. In a private company, this may be easier. It should be noted that the two previous process cannot implement if the multi-plant, or the smart grid, problem is solved in an integrated model. This is another advantage of LSOM: a better understanding of the functioning of techno-socio-economic systems.

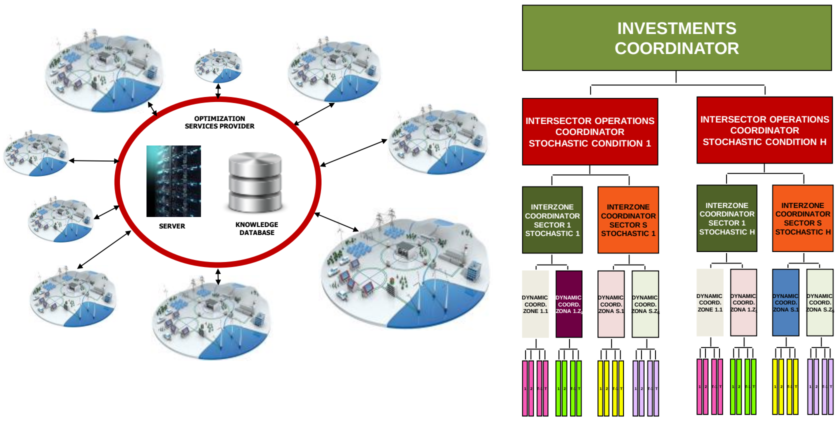


Figure 27. A Network of Smart Grids Optimization using LSOM

7.2.3 Routing

The VRP-TW-TDTT, described in a previous section, is another example of RT-DO. In this case the process of a logistics operator which provides its services in a city congested traffic could be made as follows (figure 28):

1. At a certain moment of the day, a central dispatcher plans integrated all routes to comply with the delivery and/or the pick-ups of the day; this implies to solve the problem VRP-TW-TDTT optimizing an objective function according to the business rules of the enterprise. This process would be carried out centrally to generate the route (sequence of clients) and the path (sequence of streets) of each vehicle that will be used that day, each route must be loaded in a mobile smart device (SD) in the vehicle.

2. During the route, when finish the service in a node, the mobile device requests (to Waze) the information of the estimate travel times that need to end the route; with the new information the SD can solve a Travelling Salesman Problem (TSP) and send the information to the central dispatcher. The starting point must be the last route assigned to the vehicle.

3. The central dispatcher decides when is necessary to solve the full VRP-TW-TDTT to reoptimize all routes, respecting constraints of the vehicles that are in operation; which limits certain operations, specific for each type of problem VRP.

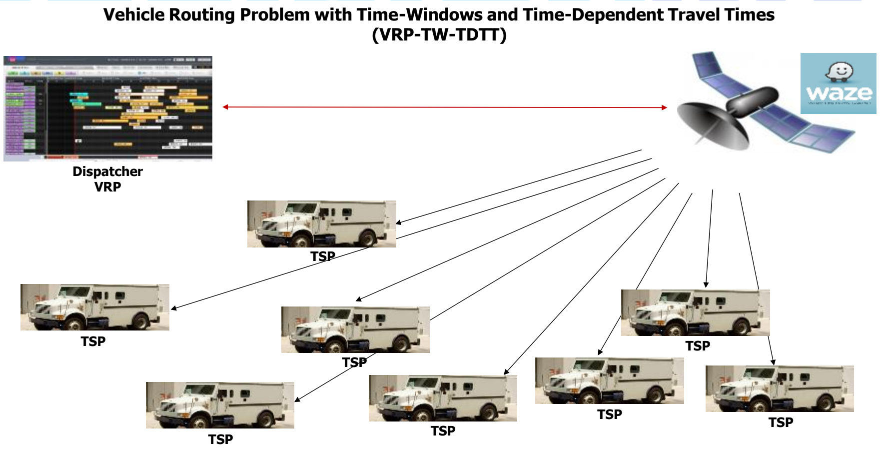


Figure 28. Distributed Real Time Optimization VRP-TW-TDTT

This approach is consistent, since one of the ways of solving the VRP model is view it as a set of TSP problems, one for each vehicle, and solve it with a column generation scheme.

References

1. Behrang, H. (2009). Dynamic Decision Making for Less-Than-Truckload Trucking Operations. Doctoral Thesis University of Maryland, 2009;
2. Benders, J.F. (1962). Partitioning procedures for Solving Mixed Variables Programming Problems. Numer. Math 4, 238-252.
3. Box, G. and Jenkins, G. (1970). Time Series Analysis: Forecasting and Control. San Francisco: Holden-Day.
4. Charnes, A. Cooper, W. and Rhodes, E. (1978). Measuring the Efficiency of Decision-Making Units., European Journal of Operational Research 2, 429-444.
5. Codd, E. F. (1970). .Relational Completeness of Data Base Sublanguages. Database Systems: 65–98. CiteSeerX 10.1.1.86.9277.
6. Cortes, C. and Vapnik, V, (1995). .Support-vector networks. Machine Learning. 20 (3): 273–297. CiteSeerX 10.1.1.15.9362. doi:10.1007/BF00994018.
7. Dantzig, G. B. and Wolfe, P. (1960). Decomposition Principle for Linear Programs. Operations Research, 8(1):101-111,1960.
8. Davenport, T. H. and Harris, J. G. (2007). Competing on Analytics: The New Science of Winning. 1st Edition Harvard Business School Press (2007)
9. Engle, R. F. (1982). Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation». Econometrica 50 (4): 987-1007. JSTOR 1912773.
10. Geoffrion, A. (1972). Generalized Benders Decomposition. Journal of Optimization Theory and Applications. October 1972, Volume 10, Issue 4, pp 237–260
11. Greenberg H. and Pierskalla, W.P. (1970). .Subrogate Mathematical Programming. Operations Research 18 (1970), 924-939
12. Hansen, P and Jaumard, B. (1997). Cluster Analysis and Mathematical Programming. Mathematical Programming · October 1997 DOI: 10.1007/BF02614317 · Les Cahiers du GERAD. G–97–10
13. Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems, Transactions of the ASME - Journal of Basic Engineering Vol. 82: pag. 35-45 (1960).
14. Kurt, O. J, Mikael, N. and Smeds P. (1985). Variable Splitting: A New Lagrangean Relaxation Approach to Some Mathematical Programming Models. University of Linköping, Department of Mathematics, 1985.
15. Lara, C. L., Mallapragada, D., Papageorgiou, D., Venkatesh, A., and Grossmann, I. (2017). .MILP Formulation and Nested Decomposition Algorithm for Planning of Electric Power Infrastructures. http://egon.cheme.cmu.edu/Papers/Lara_Grossmann_Nested_ElecPowerPlan.pdf
16. Lubbecke, M. (2010). Column Generation. Wiley Encyclopedia of Operations Research and Management Science, edited by James J. Cochran Copyright © 2010 John Wiley & Sons, Inc.
17. Markowitz, H.M. (1952). Portfolio Selection. The Journal of Finance. 7 (1): 77–91, 1952. doi:10.2307/2975974. JSTOR 2975974.
18. Meyer, X. , Albuquerque, P. and Chopard, B. (2017). Linear Programming on a GPU: A Case Study. Chapter 10 in book: Designing Scientific Applications on GPUs, Publisher: Chapman and Hall/CRC (2017)
19. Puterman, M. (1994). Markov Decision Processes - Discrete Stochastic Dynamic Programming. John Wiley & Sons, Inc., New York, NY.
20. Rockafellar, R. T. and Uryasev, S. (2000). Optimization of Conditional Value-at-Risk. Journal of Risk.
21. Shapiro J. (2006). .Beyond Supply Chain Optimization to Enterprise Optimization., https://mthink.com/legacy/www.ascet.com/content/pdf/ASC3_wp_shapiro.pdf
22. Van Roy, T. (1983). Cross Decomposition for Mixed Integer Programming., Mathematical Programming (1983) 25: 46. https://doi.org/10.1007/BF02591718
23. Van Slyke R. and Wets R. (1969). L-shaped Linear Programs with Applications to Optimal Control and Stochastic Programming. SIAM Journal on Applied Mathematics 17, 638-663.
24. Vapnik, V. (1995). The Nature of Statistical Learning Theory. Springer-Verlag, New York, 1995.
25. Vapnik, V. (1998). Statistical Learning Theory. John Wiley and Sons, Inc., New York, 1998.
26. Velásquez, J. M. (1995). OEDM: Optimización Estocástica Dinámica Multinivel. Teoría General. Revista Energética No. 13 (http://www.doanalytics.net/Documents/OEDM.pdf).
27. Velasquez, J. (1986). PDS: Primal-Dual Subrogate Algorithm for Nonlinear Programming. http://www.doanalytics.net/Documents/DW-DT-013-PDS.pdf
28. Velasquez, J. (1997). Asynchronous Parallel Optimization for Expansion and Operation of Multisectorial Industrial Systems. 34th European Conference on Operational Research held in Barcelona (Spain).

29. Velasquez, J. (2019a). OPTEX – Optimization Expert System. https://www.linkedin.com/pulse/optex-optimization-expert-system-new-approah-make-models-velasquez/
30. Velasquez, J. (2019b). Oil Pipelines Real-Time Optimization. https://www.linkedin.com/pulse/oil-pipelines-real-time-optimization-jesus-velasquez/
31. Velasquez, J. (2019c). PTALIN - A Distribution Tactical Model. http://www.doanalytics.net/Documents/DW-DT-PTALIN.rar
32. Velasquez, J. (2019d). .J. F. Benders Theory, Variations and Enhancements. Chapter in Large Scale Optimization in Supply Chain & Smart Manufacturing: Theory & Real-Life Application. Published by Springer
33. Velasquez, J. M. (2019e). Stochastic Optimization: Fundamentals, Chapter in the Book Large Scale Optimization Applied to Supply Chain & Smart Manufacturing: Theory & Real-Life Applications. Springer (2019)
34. Wikipedia. https://en.wikipedia.org/wiki/Industry_4.0
35. Wolak, F. (1989). Testing Inequality Constraints in Linear Econometric Models. Journal of Econometrics 41 (1989) 205-235. North-Holland

Subject Index