Overview

SAS/OR 13.1 includes new features and enhancements to current features in optimization, discrete-event simulation, and constraint programming. Appearing alongside ongoing improvements in the performance of the linear, mixed integer, quadratic, and general nonlinear optimization solver algorithms, these changes expand the range of problems you can address, make it easier to use the SAS/OR modeling and solution methods, deepen integration with other SAS analytic capabilities, and more fully utilize your available computational resources.

Highlights of Enhancements

Highlights include the following:

- **PROC OPTMODEL** adds:
  - Direct access to network optimization and analysis algorithms (experimental)
  - Parallel execution of solver invocations in a FOR loop
  - Support for function definition via PROC FCMP in Base SAS
- **PROC OPTLSO** adds:
  - Multiobjective optimization
  - Support for the use of array-structured data in function definition (with PROC FCMP)
- The mixed integer linear programming (MILP) solver adds the option to execute in parallel on multiple computational cores (experimental).
- **SAS Simulation Studio** adds:
  - Support for custom block icons
  - Improvements to the simulation clock display
  - Enhancements to the Submodel block interface
  - Other interface improvements

The CLP Procedure

The CLP procedure uses constraint logic programming methods to solve general and scheduling-oriented constraint satisfaction problems and it can also solve optimization problems. In SAS/OR 13.1, the OBJECTIVE statement, specifying an objective function to be added to a constraint satisfaction problem, is now production status. This statement enables PROC CLP to solve optimization problems that have intricate logical constraints.

The OPTLSO Procedure

The OPTLSO procedure uses local and global search methods to solve optimization problems without making simplifying assumptions about the nature or behavior of the objective or constraint functions. In SAS/OR 13.1, PROC OPTLSO enables you to specify more than one objective by using the OBJECTIVE= option in the PROC OPTLSO statement. For multiple objectives, PROC OPTLSO returns a set of Pareto-optimal points, constituting an efficient frontier. This feature expands the range of problems that can be addressed using PROC OPTLSO and enables you to explore the trade-offs that can exist between your identified objective functions. The definition of the Pareto-optimal set is always based on the values of the objective functions, and it also includes feasibility considerations when constraints are present. You can limit the size of this set by specifying the PARETOMAX= option in the PROC OPTLSO statement.

The OPTMODEL Procedure

The OPTMODEL procedure provides an interactive environment in which you can build and solve a wide range of optimization models. In SAS/OR 13.1, PROC OPTMODEL adds the network solver (experimental), providing direct access to the set of eleven network analysis and optimization algorithms also accessible through PROC OPTNET. This makes it easier to solve network-oriented problems with PROC OPTMODEL, especially as a component of a larger solution or other analytical process. The SOLVE WITH NETWORK statement invokes the network solver. Unlike other solvers that PROC OPTMODEL uses, the network solver operates directly on arrays and sets. You do not need to explicitly define variables, constraints, and objectives to use the network solver. PROC OPTMODEL declares the appropriate objects internally as needed. You specify the names of arrays and sets that define your network-structured inputs and outputs as options in the SOLVE WITH NETWORK statement. In addition to input and output, options in the SOLVE WITH NETWORK statement define processing and diagnostic controls and specify the algorithm to execute.
PROC OPTMODEL also now enables you to run optimization solver invocations in parallel, in iterations of a concurrent FOR loop that you specify using the COFOR statement. The COFOR statement operates in the same manner as the FOR statement, except that in a COFOR statement PROC OPTMODEL can execute the SOLVE statement concurrently with other statements. The execution of the COFOR substatement is interleaved between loop iterations so that other iterations can be processed while an iteration waits for a SOLVE statement to finish executing. Multiple solvers can run concurrently. PROC OPTMODEL manages this interleaving so that in many cases a FOR loop can be replaced by a COFOR loop to achieve concurrency with minimal or no other changes to the code. For problems in which the SOLVE statement execution accounts for the majority of the time needed to execute a loop iteration, the COFOR statement can significantly reduce overall execution time.

Finally, in SAS/OR 13.1 PROC OPTMODEL can call functions and subroutines that have been defined and compiled using PROC FCMP in Base SAS. This capability enables reuse of previously defined functions and subroutines, deepening integration with other SAS analytic procedures from which they can also be called. You can use an FCMP function anywhere a function is otherwise permitted in PROC OPTMODEL. You can use the CALL statement in PROC OPTMODEL to call FCMP subroutines. An FCMP subroutine can return data by updating PROC OPTMODEL parameters (numeric and string) that are passed as arguments in the corresponding CALL statement and declared using the OUTARGS statement in the PROC FCMP subroutine definition. In addition to numeric and string parameters, you can pass PROC OPTMODEL arrays to PROC FCMP functions and subroutines that accept matrix arguments.

**Linear and Nonlinear Optimization**

In SAS/OR 13.1, the concurrent solve capability (specified using the ALGORITHM=CONCURRENT option in the PROC OPTLP statement or in the SOLVE statement in PROC OPTMODEL) for linear and nonlinear optimization is now production. This feature executes all available solution algorithms in parallel, as permitted by the number of computational cores available. The first algorithm to terminate returns its solution.

**Mixed Integer Linear Optimization**

In SAS/OR 13.1, the mixed integer linear programming (MILP) solver adds the ability (experimental) to execute the branch-and-cut solution algorithm in parallel on multiple computational cores (single-machine mode). To enable parallel processing of the branch-and-cut algorithm, you need to specify the PARALLEL=1 option in the PROC OPTMILP statement or in the SOLVE WITH MILP statement in PROC OPTMODEL.

**SAS® Simulation Studio 13.1**

SAS Simulation Studio 13.1, a component of SAS/OR 13.1 for Windows environments, makes several enhancements to its graphical simulation modeling and analysis interface. You can now specify a custom image to replace the default icon for any block instance; the new Visual tab in each block’s Block Properties dialog box enables you to specify a custom image and customize the block label. The Simulation Clock display has been augmented to graphically indicate the status (running, paused, or stopped/completed) of a model.

The Definition view for a Submodel window adds its own block template that appears when you place the mouse pointer over a small image of the template near the top of the Submodel window. This feature makes it easier for you to add blocks to the definition of a submodel. In SAS Simulation Studio 13.1, it is also easier to edit expressions, thanks to the addition of a dedicated Expression window that opens from many Block Properties dialog boxes for blocks in which expressions such as attribute rules can be specified. In addition, the calculation of statistics in the Resource Stats Collector block has been improved and expanded. Finally, SAS Simulation Studio 13.1 streamlines the method of specifying a DataStreamDescription factor or an InStreamPolicy port value for a Numeric Data Source block in order to control the probability distribution that is sampled in the block.

**For More Information**

For more information, ask your organization’s SAS representative to contact the SAS Customer Interaction Center at 1.800.727.0025.