MODELING PROCESS AND FORM FOR PROCESS PLANT PIPE ROUTING

H. Craig Howard, Ph.D.[1]
Senior Knowledge Engineer
Design Power, Inc.

Abstract

The Design Power Auto Router rapidly generates conceptual pipe routes using an integrated object model built from process schematics along with the geometry and layout of equipment and pipe supports. The software reasons about physical and topological relationships between the components to position nozzles on the equipment and to route pipelines between the nozzles while avoiding obstacles, minimizing unsupported pipe lengths, and placing pipeline components such as tees, reducers, flow meters, and control valves. The router reduces the time required for conceptual routing from weeks or months to a few hours to set up the routing model and a few seconds or minutes to generate a set of routes. With this kind of improvement in speed, plant designers can produce more accurate and detailed estimates, or explore many more layouts.

INTRODUCTION

> "Piping constitutes 25 to 35 percent of the material costs of a process plant, requires 30 to 40 percent of the erection labor, and consumes 40 to 48 percent of the total engineering hours." [Rase 1963]

That statement is still as true as it was in 1963. In the U.S., medium to large engineering firms do most plant design. Total design billings by the top 500 design firms was $27.95 billion (ENR 1995). Thirty-six percent (36%) of the total billing was in industrial process, petrochemical and power industries. These industries are the main industries involved in piping design. Using the lower piping engineering estimate, 40%, these numbers tell us that the U.S. design firms spent at least $3.9 billion in piping design last year.

The design of process plants (refineries, chemical synthesis, power production, etc.) is driven by concerns of process integrity, safety, and constructibility. Current design tools for process plant design are broken into little islands of automation—programs that rely on manual integration to complete the design cycle and human

1 10200 N. DeAnza Blvd., Cupertino, CA 95014, USA; e-mail: hch@dp.com; phone: 408-366-6600; FAX: 408-366-6607.

knowledge of the design interdependencies. There are discrete tools for process simulation, process and instrumentation diagramming, facility layout, piping design, and cost estimation. These tools are typically closed systems with at best limited facilities for customization and intelligence. With the complexity of the individual steps and the difficulty of integration, there is little opportunity for a downstream design task to provide feedback to influence upstream design operations within the same project.

Plant Design Software

An integrated process model supported in a knowledge-based engineering environment can provide the basis for automating the integration between existing tools, providing rapid feedback from downstream tasks and checking designs for operability and safety in ways that current software cannot. Furthermore, an organization's expertise can be added to the environment as rules to maintain the unique and valuable advantages that design firms accumulate over time. In an industry where a design or design-build firm may generate many proposals in order to get a few projects, the ability to rapidly proceed through an entire design to get good cost estimates may provide a significant edge.

At Design Power, we have developed applications for process plant design including process synthesis, process safety analysis, plant layout, and pipe routing. This paper focuses on the last area, describing the Design Power Auto Router™ software that will be released as a commercial product for conceptual pipe routing in fourth quarter of 1995. The core of each plant design application is an integrated process model that combines the conceptual process views that engineers usually capture in schematic drawings and the three-dimensional spatial views that would otherwise be trapped in a series of two-dimensional geometric drawings.

The reasoning for process design tasks uses object representations for the components (equipment, pipelines, pipeways, obstacles, etc.) and relationships between those components (pipelines *connect* equipment, a pump *is part of* a process group, a tank *is located in* a layout region) to automate tasks such as layout and routing, and to provide a basis for user interaction with those tasks. The environment maintains the dependencies between design decisions so that the effects of changes can be automatically propagated throughout the design; e.g., if an equipment item is moved, then pipelines connected to that item or in its immediate vicinity may need to be rerouted. The results of the conceptual routing can easily be exported from the Auto Router's integrated object model into a detailed piping system to complete the piping design.

The Auto Router is implemented in libraries (knowledge bases) on top of Design Power's Design++® development environment.[2] (See Figure 1.) The Auto Router uses a CAD system (MicroStation in the initial version) as a vehicle for three-

---

2 Design++™ is a knowledge-based engineering design system that provides object-oriented representation and a design rule language with dependency tracking to provide model-based reasoning with a tight coupling to multiple CAD systems (including MicroStation and AutoCAD).

dimensional visualization and manipulation of the evolving design. This paper outlines the workflow and describes the object representation and reasoning in the Auto Router in the context of a small process plant example.

WORKFLOW

The first step in the workflow for Auto Router is to take the information found in a process design schematic and use the process model builder to create corresponding objects (see Figure 2). The next step is to combine the information that an engineer has generated in a plot study, in which equipment and structures have been located on the site with the process model that was just created.

With the completed process model, the Auto Router can locate the nozzles using a set of intelligent nozzle placement rules. Then the system is ready to route the pipes. The routing includes the placement of a variety of pipeline components such as control valves, reducer, flow meters, and tees. Once the pipes have been routed, the model contains all the information needed to generate a list of material quantities. However, if the user doesn't like the results, she can change the layout or process model, try some different options, and then reroute the pipes.

Initially, the pipes are displayed as simple lines representing the bottoms of the pipes on the horizontal segments and the centers of the pipes on the vertical pipes. At any point, the routed pipes can be represented as three-dimensional extruded shapes including the pipeline components to produce hidden line or rendered views.

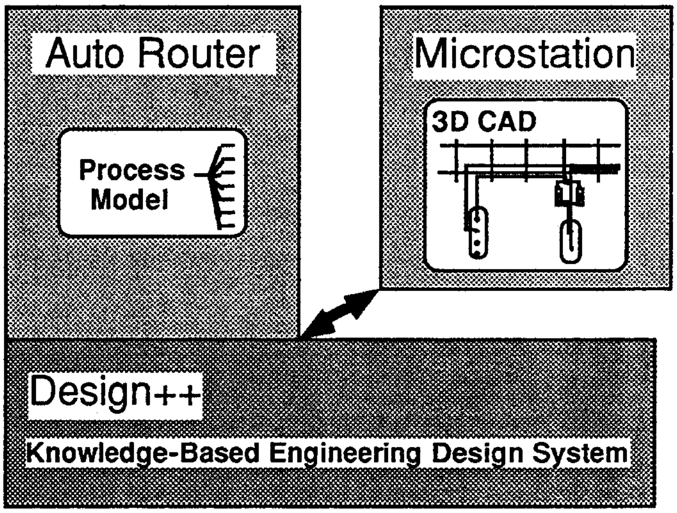

Figure 1: Auto Router software architecture

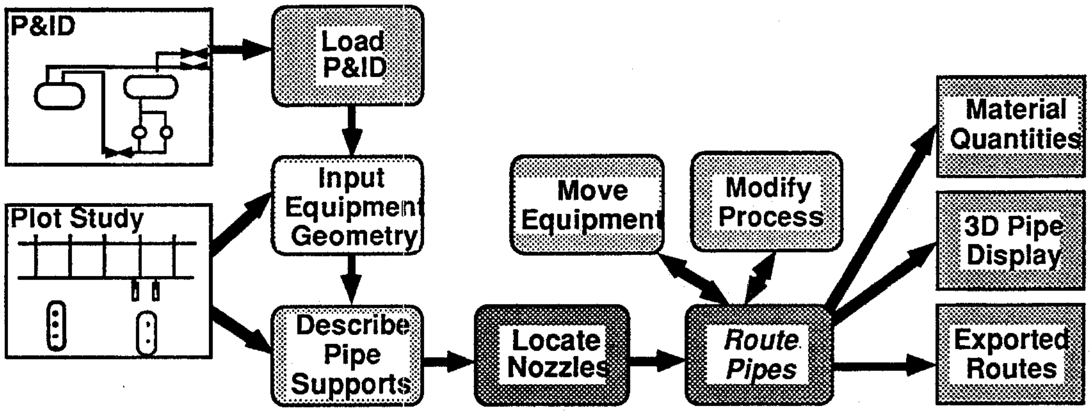

Figure 2: Workflow in Process Plant Design System

BUILDING THE PROCESS MODEL

The Auto Router requires the information that typically is communicated in a Process Flow Diagram (PFD) or in a Process and Instrumentation Diagram (P&ID). Frequently, the PFD spent some time in computer form as input to or output from one or more process simulation programs. We may also have an equipment list generated as part of early cost estimates. The P&ID is typically composed and exchanged as a schematic drawing. In one fashion or another, those sources of data are be combined into a list of equipment, a list of nozzles attached to the equipment, a list of logical pipelines with properties (size, material, commodity, and insulation), and a connectivity list describing which pipelines are connected to which nozzles and other pipelines.

The process modeler associates the input with classes defined in the plant design libraries. *Libraries* in Design++ contain hierarchies of classes that define the inheritance of associated attributes, behavior (rules), and geometric representation. A portion of the Auto Router's equipment library is shown in Figure 3. To represent a particular design, the classes are instantiated in a Design++ *model*, which contains instances arranged in an aggregation hierarchy. The model created by the process model builder as part of the demonstration case for this paper is shown in Figure 4.

The root object in the model hierarchy is the project, and it has several parts for representing the logical process model, the other physical objects on the site (e.g., pipeways, pedestals, and other obstacles), and some specialized reasoning agents (e.g., process model builder and router) that contain rules for managing the tasks performed by the software. The process model is divided into units according to logical divisions in the process design. Each unit has a set of equipment and pipelines, identified in the model display by their tags (unique descriptive identifiers). Each piece of equipment may have one or more nozzles as parts; their tags are constructed from the equipment tag and a nozzle label. Pipelines have a 3DLINE object as a separate part to support the geometry for the three-dimensional extruded representations of pipes that we will see later. The pipeline may also have a number of pipeline components (i.e., tees, reducers, flow meters, and control valves) positioned along the pipe's route.

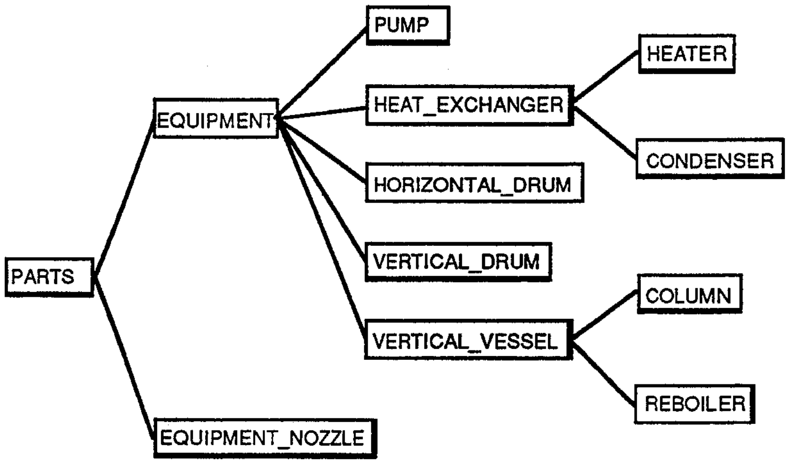

Figure 3: Classes in Equipment library

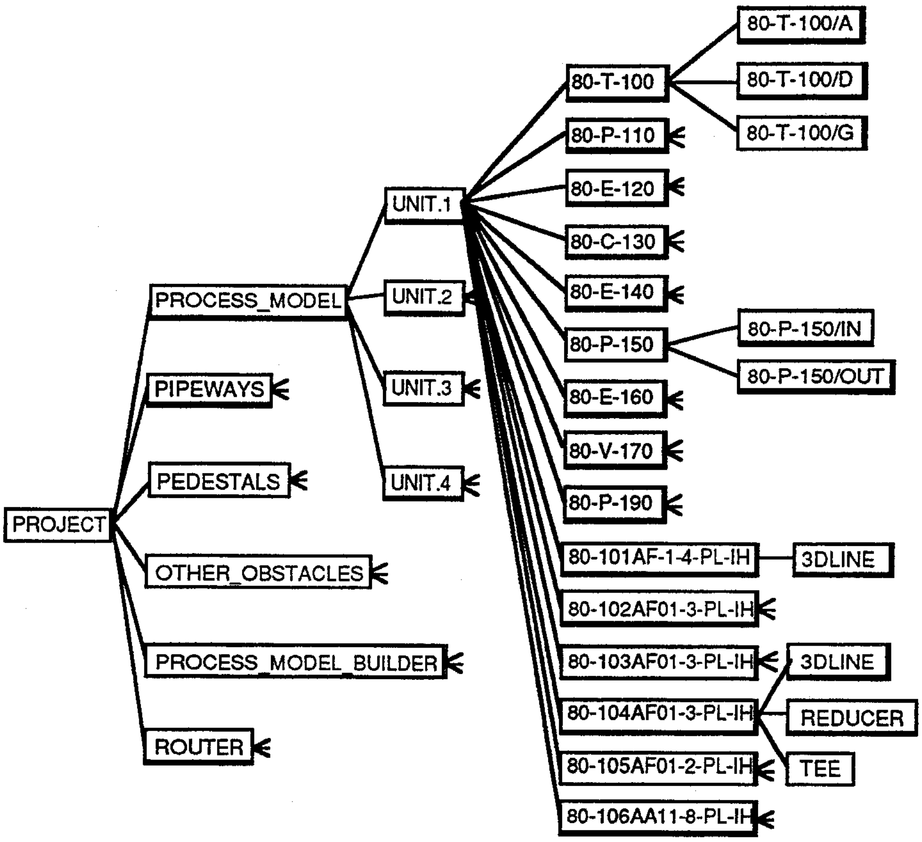

Figure 4: Part of model aggregation hierarchy for demonstration case

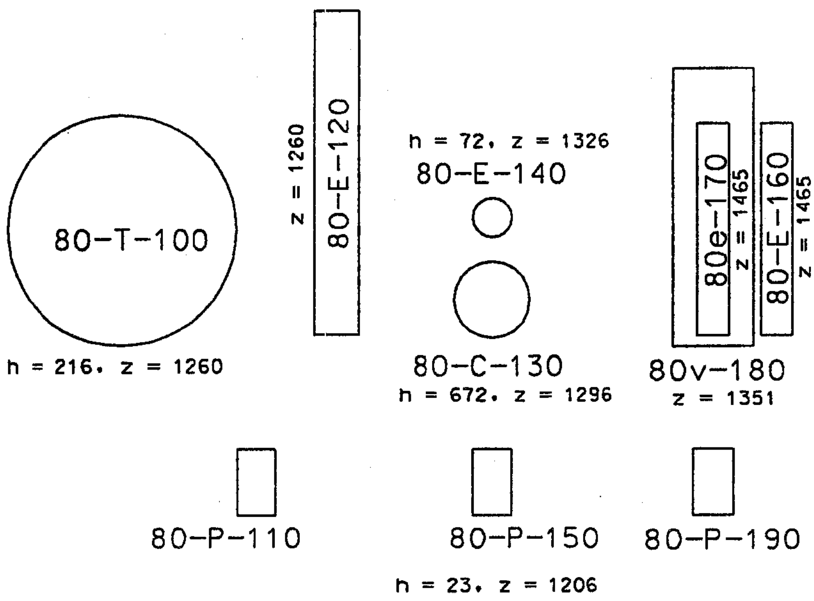

Figure 5: CAD input from plot study

Adding Geometry

To complete the process model, we need to add geometry. In the Auto Router, the geometry can be acquired from an existing two- or three-dimensional CAD drawing with a simple interface in the CAD system that allows a user to link existing drawing elements with process model objects. The interface extracts the available information from the drawing element (shape, dimensions, location, and orientation) and allows the user to supplement that data as necessary; for example, a user would need to add information about height and elevation to a two-dimensional drawing. The user will also need to add the objects that weren't in the process model: pipeways, equipment supports, buildings, previously placed equipment in retrofit projects, etc. Those steps can easily be accomplished using the CAD interface.

The CAD input from the plot study for our demonstration case is shown in Figure 5. The layout designer has generated a two-dimensional drawing and annotated the information for the z-dimensions. The labels on the shapes correspond to the process model tags shown in Figure 4. By linking the 2D shapes to the process model and supplying the z-dimensions as described above, the model has sufficient information to generate a 3D representation of the plant. (You can skip ahead to Figure 8 to see what the completed plant looks like.) The demonstration case has a small distillation process, including a vertical drum that serves as the feed storage tank (80-T-100), a distillation column (80-C-130), a reboiler (80-E-140) for the column, an input heater for the column (80-E-120), a horizontal drum that serves as an output storage vessel (80-V-180), paired condensers for the output streams from the column (80-E-170 and 80-E-160), and a trio of pumps (80-P-110, 80-P-150, and 80-P-190) along the

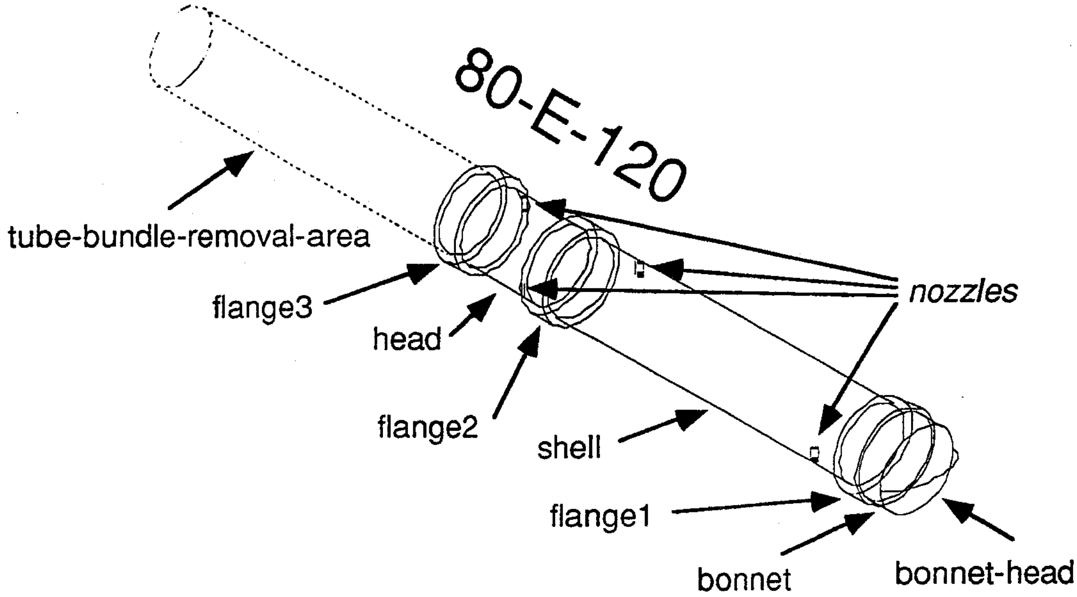

Figure 6: A Heat Exchanger showing elements and nozzles

pipeway. As part of the layout design, the user defines a main pipeway of steel beams and columns to support the long pipe runs. To finish the physical model, the user creates a pedestal frame under the elevated output vessel and paired condensers.

Parametric Representation of Equipment

In a highly specialized domain such as plant design, it is difficult to anticipate every type of object, especially every type of equipment. That is why the Auto Router includes a flexible method for defining the geometry and other specialized knowledge about equipment classes. At the point when a piper is starting the conceptual design, he wants to deal with a very idealized representation of the equipment such as the rectangles and circles in Figure 5. In the Auto Router, a minimal set of geometry input can be expanded using a parametric language that allows the piper to describe more complex representations for common equipment classes. Figure 6 shows an isometric view of the 80-E-120 heat exchanger, which was generated from the corresponding rectangular shape in Figure 5. The Auto Router includes an equipment assembly language that allows the piping designer to represent the complex geometry of this shell and tube exchanger including both its physical parts and its access spaces (e.g., the tube-bundle-removal-area) as annotated in Figure 6 (excluding the nozzles). There is also a special element type for obstacles that define the space around this equipment in which pipelines cannot be routed unless they originate at a nozzle within that space.

Placing nozzles

The final pieces of geometry required for pipe routing are the nozzle locations on the equipment. Each equipment class includes a set of nozzle placement rules that use the

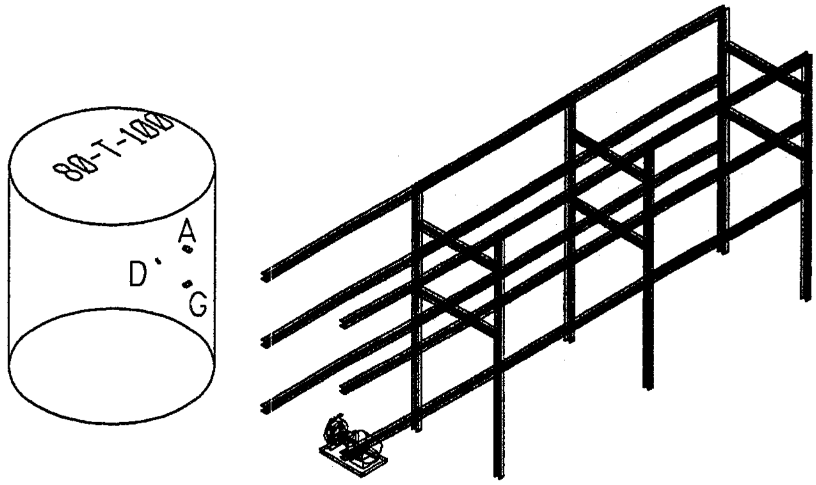

Figure 7: Nozzle placement on the vertical drum

information such as the flow direction through the nozzles (inlet or outlet), the commodity in the connected pipeline, and the location of the other end of the pipeline to intelligently place the nozzle at a reasonable default location. Figure 6 shows the default nozzle locations for a heat exchanger: the tube inlet and outlet are located on the top and bottom in the middle of tube section; the cold input pipeline with process liquid goes on the bottom of the shell near the base of the exchanger and the hot outlet pipeline goes on the top of the shell near the head.

Nozzle placement on other types of equipment may be more complex and variable. For instance, consider the vertical drum with its three nozzles shown in Figure 7. Inlet nozzles (A and D) are located near the top of the tank, and outlet nozzles (G) are located near the bottom of the tank. Furthermore, the first inlet nozzle (A) is oriented toward the pipeway (it is likely connected to a pipeway pipe). The next inlet nozzle (D) is offset 20 degrees from the previous one, and that offset is towards the other end of the pipeline to which D is connected. Since that pipeline runs along the pipeway off the lower left of the figure, D is oriented to the left (or clockwise) from A. The outlet nozzle (G) is oriented parallel to the pipeway on the side toward the other end of its pipeline, which flows to the pump shown in the figure. Therefore, G is located on the right side of the tank. The Auto Router includes a nozzle placement language that allows pipers to specify rules such as these in a form that closely follows the English description.

AUTO-ROUTING PIPE

Now that we have the process information, the equipment locations, the pipe supports, and the nozzle placements, the Auto Router is ready to route pipes. The router looks for good routes for those pipes and also places a number of pipeline components such as tees, reducers, and control valves. The Auto Router reasons about where the equipment is, where the pipelines start at the nozzles, about avoiding obstacles, where pipelines should connect, and minimizing unsupported pipe lengths. It tries to find the shortest routes, turns with right angle bends wherever possible, maintains appropriate piping levels, and uses the pipeways for long pipe runs.

This small demonstration problem has 26 pipes. For an experienced piper, the conceptual routing for this problem would likely take 3-5 days. For the Auto Router software, that task takes about 20 to 30 seconds on a SUN SPARCstation 2, roughly one second per pipe. Figure 8 show plan and isometric views of the routed model including representations of the pipes in their final form as three-dimensional extruded shapes. Since the Auto Router produces its display in MicroStation, the user has access to the complete capabilities of the CAD system for viewing and rendering.

Since this router took us only a few seconds to generate a result for this layout, a user can easily modify the problem and reroute to compare results for the piping. Those modifications can be performed directly in the CAD interface—dragging equipment to new locations, repositioning nozzles, adding or removing obstacles, selecting process components to delete, etc.

Estimating Quantities

With all the routing details represented in our model, it is a trivial process to generate the list of material quantities. Table 1 gives the results output by the Auto Router for the pipe routes shown in Figure 8. In the table, we have information about the pipeline, its identifiers (the unit id and the line id), the pipe class (which gives us material properties), the commodities in the pipes (e.g., "PL" is a process liquid), the nominal diameter in inches, the length of the pipe in feet, the number of elbows, and the number of tees. We could easily combine these results with cost information or weight information to give us a reasonable aggregate value for the routing for comparison or for cost estimation.

Unit ID	Line ID	Class	Commodity	Diameter	Length	Elbows	Tees
60	100	AF01	PL	4.0	103.7	8	0
80	101	AF01	PL	4.0	21.3	4	0
80	102	AF01	PL	3.0	55.0	8	0
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
80	108	AA11	PV	6.0	6.8	3	0
TOTALS					1240.0	86	6

Table 1: Material quantities list for demonstration case (abbreviated)

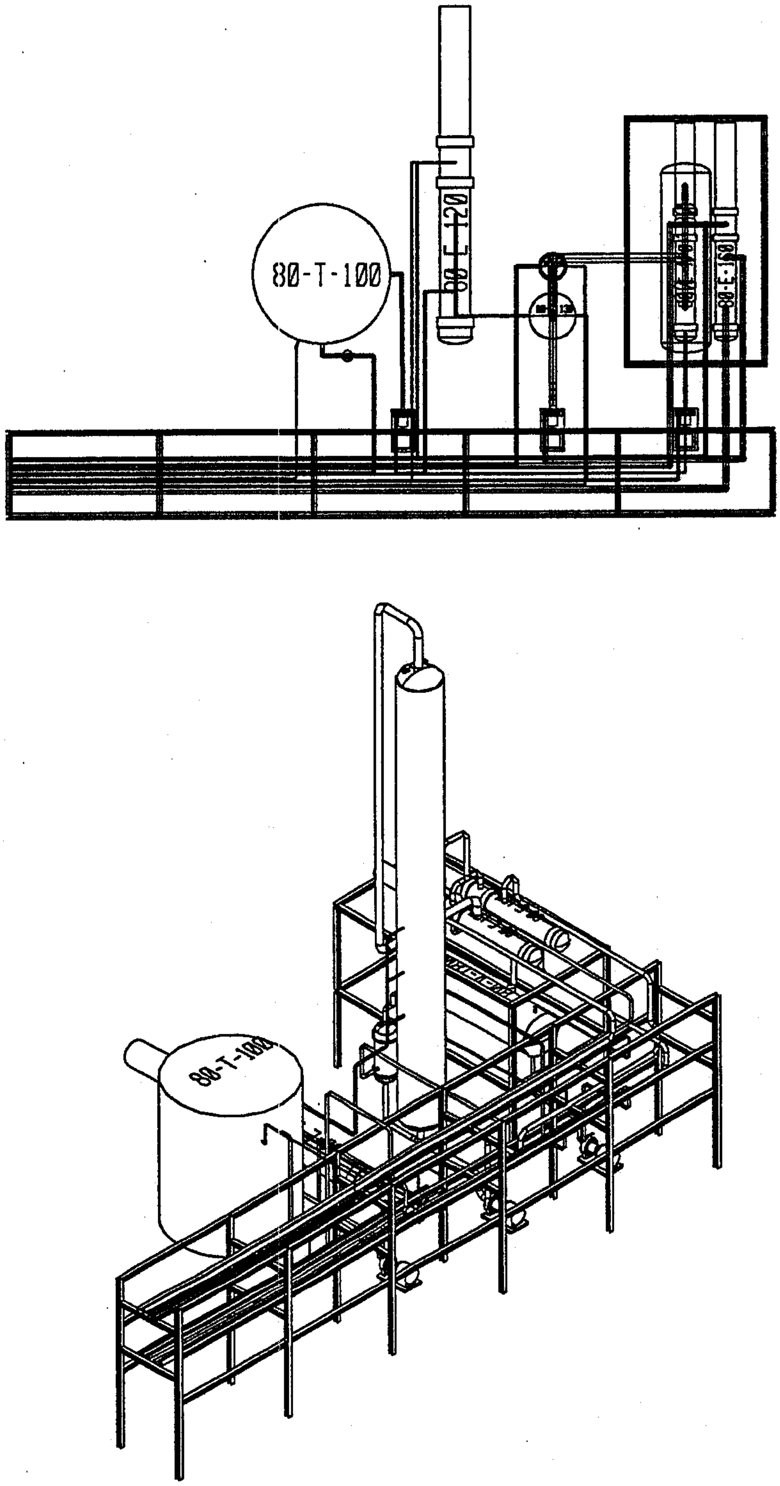

Figure 8: Plan and isometric view of routing results

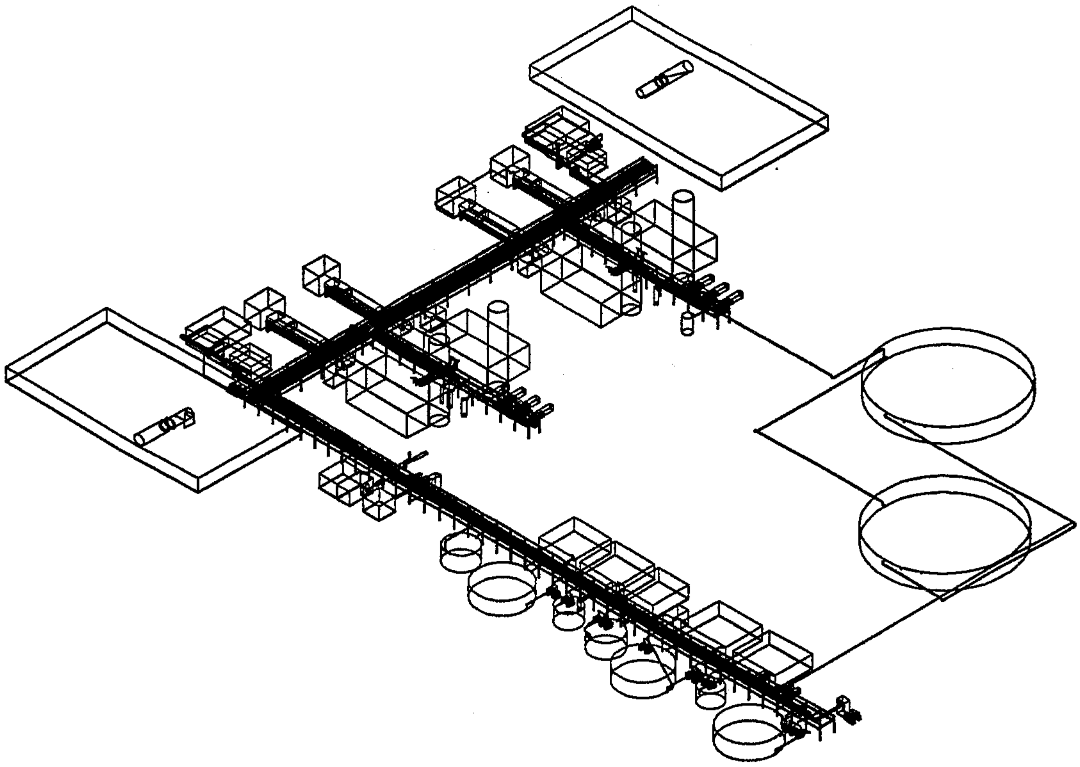

Figure 9: Layout and pipe routes for a large co-generation facility (courtesy Fluor Daniel, Inc.)

Performance

This small demonstration case provided a good context for discussing the Auto Router representation and reasoning, but its real value is in solving much larger problems. The Auto Router scales up nicely to handle those real problems. The software has been tested on a variety of plant layouts. For instance, the large co-generation facility shown in Figure 9 has 144 process pipelines (many of which are header lines that are split into branches at paired equipment), four pipeways, and almost 100 pieces of equipment. The Auto Router running on a Sun SPARCstation 2 produced a set of routes for that plant in just under 6 minutes.

Because the spatial reasoning required for routing is very complex, the first implementation of the Auto Router aims for a 90-95% solution, i.e., finding good routes for almost all pipes, which is adequate for conceptual piping for cost estimation or layout evaluation. Furthermore, it informs the user about the pipes with which it has problems, allowing the user to explore possible layout and connectivity problems that could make routing more difficult. The Auto Router will also provide manual routing adjustment tools to let the piper try various alternatives to tweak a set of routes.

CONCLUSION

The Auto Router combines object representation and reasoning with off-the-shelf CAD to automate the early phases of piping design. Its main purpose is to provide the plant layout engineer and piping engineer with the means to rapidly evaluate the feasibility and piping costs of a proposed layout. However, the conceptual pipe routes generated by the Auto Router also shave weeks from the overall piping design schedule by providing feasible starting points for detailed piping design.

The Auto Router is highly interactive and customizable. A piper can add, delete, or move equipment, nozzles, structures, or other obstacles at any time using the CAD interface. The connectivity can be changed as well—pipelines can be added, deleted, and reconnected. Pipers can add new equipment classes by specifying parametric descriptions for the equipment geometry, default nozzle placement, and preferred pipe routing in the equipment's vicinity. The user may also define the priority order in which pipes are routed and their preferred placements on pipeways.

A key feature of this kind of automation is the immediate CAD feedback. With this kind of approach, CAD shifts its role from being strictly a drawing tool to being a user interface (showing intermediate results and supporting object selection and manipulation) and a report generator (for external communication through traditional design drawings). The implementation of the Auto Router on top of Design++ makes that flexible CAD integration possible.

ACKNOWLEDGMENTS

The Auto Router software was designed and implemented by a dedicated team, including Matti Katajamaki, Alan Axworthy, Dr. Janet Murdock, and Ulf Strom, with support from the rest of the Design Power crew. That effort was further supported by Kevin Lombardo, Brian Weir, and Tami Morse-McGill of Fluor Daniel, Inc.

REFERENCES

[Rase 1963] Rase, Howard F., *Piping Design for Process Plants*, Robert E. Krieger Publishing Co., Malabar, Florida, 1963.

[ENR 1995] *Engineering News Record*, "The Top 500 Design Firms", April 3, 1995, p. 34-42.