A DISCRETE DETERMINISTIC PIPELINE FLOW SIMULATOR WITH ONLINE SCHEDULER INTERFACE TO SOLVE DYNAMIC BATCH SCHEDULING PROBLEMS

Charles A. DeFelice
Cities Service Company, Tulsa, Oklahoma

CASE OVERVIEW

Simulations of pipeline systems are undertaken for a variety of reasons: to evaluate system performance under varying operating conditions, to aid in pumping cost minimization, to analyze proposed system modifications, and most often to predict the results of actual daily operation. In large pipeline systems which operate at high flow rates, continuous hydraulic models are required to provide the precision needed in determining expected valve switching times, surge activity, and line pressures. In smaller pipelines where the flow is considerably slower, the operating characteristics are relatively stable with system changes occurring more slowly. In operating environments such as this, the line activity can adequately be monitored and the only real need is for timing information concerning product movements. This need can be met with a less sophisticated discrete approximated simulation model. Such a model is a component in a pipeline scheduling system designed for Badger Pipeline. It is necessary to set a framework for the system description by first outlining the nature of the pipeline's operation and the manual scheduling system that was replaced.

Badger Pipeline Characteristics

The Badger Pipeline (Figure 1), unlike many refined product pipelines, does not consist of long mainlines, breakout storage tanks, and feeder lines. It is a closed system of twelve interconnected line segments. The configuration includes three origin pump stations whose independent flows must be coordinated in the utilization of common line segments to seven destinations. Two connected refineries and three other pipeline systems provide an average of four million barrels of refined product to be delivered through 280 miles of single and parallel lines varying in size from 8" to 16" inside diameter.

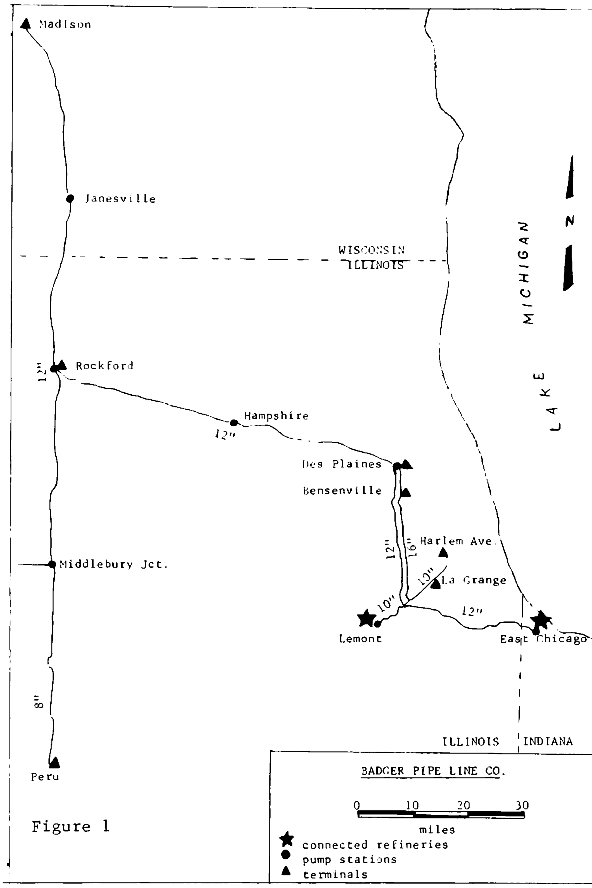


Figure 1

Pipeline Operation

The line segments forming the pipeline network are the lengths of pipeline which are separated by product delivery points or pipeline junction points. The line segments are always filled to capacity with sequences of identifiable product volumes. As new product volumes are pumped into the pipeline network, the existing product linefill is forced through a designated route of line segments that link an origin pumping station to a destination receiving station. The pumping process results in (1) the addition of new product volume to the first line segment sequence in the route, (2) the movement of the existing route sequence through the route segments, and (3) the delivery of the displaced volume from the last line segment sequence in the route.

Each product volume entering the system has a prescribed route of line segments to follow from origin to destination. Movement along any line segment is made possible only when the segment is part of a concatenation of segments beginning at an origin station which satisfies the flow requirements of each volume in the resulting route sequence. The nature of the pipeline's operation dictates that for any existing system linefill, there will be only one route of line segments leading from each origin which will satisfy all the current routing requirements. This flow route is found by tracing the volume requirements from the origin through the network until a volume is found at its destination. During flow, the linking line segments conducting the origin flow through the network will be required to change pattern in response to evolving destination and routing requirements. With multiple flows possible in a network, normal flow pattern changes can cause conflict in the use of common line segments. The resolution of these conflicts through flow timing control and the control over the initial origin pump sequences will determine the flow interaction over the schedule range. Badger's network is capable of maintaining three independent flows and has 23 distinct origin-destination flow routes possible.

Badger Pipeline's operation is based on three pumping cycles per month; each of the origins will pump to the destinations all the product volumes nominated by shippers for movement during requested ten day intervals. Monthly nominations are received from shippers requesting specific origin-destination-cycle pumpings of various products in given volumes (batches). Three factors must be taken into consideration when determining each of the origin's batch pumping sequences: (1) Shippers' timing requests, (2) Product interface requirements, and (3) Standing linefill constraints. The shippers' timing requests include the cycle nominations, special due-date requests, and desirable contiguous sourcing of volumes. Product interface requirements are sequencing rules designed to minimize the product degradation resulting from normal mixing during flow. The standing linefill constraint refers to maintaining a single product grade in the line segments at the times that they are not flowing to prevent a total mixing of products.

Manual Scheduling System

Under the manual scheduling system, the schedulers would collect a month's batch nominations and perform the following functions:

1. Convert the nominations to a standard identification form.
2. Determine initial feasible origin pump sequences which would result in flow patterns satisfying the shippers' requests, product interface requirements, and standing linefill requirements.
3. Determine from rate tables, the pump unit combinations over the range of the schedule which would result in sufficient flow rates to complete the schedule in the required time.
4. Calculate the batch movements to derive individual pumping and receiving times.
5. Supply tentative timing information to the shippers, origins, destinations, and dispatchers in the form of hand-typed communications.
6. Update the schedule as changes occur and notify the applicable parties.

The schedulers would require roughly eight days to produce the first tentative schedule. Because of the dynamic nature of the shippers' product, volume, and timing requests, they would expend most of their remaining time making daily revisions to the schedules. The time consuming function of hand-calculating the revisions prohibited the timely reporting of updates, the examination of alternative pumping sequences, and the analysis of pump unit efficiencies. It was determined that a computerized operation system incorporating flexibility, responsiveness, and ease of manipulation would be required to produce effective schedules in a timely manner.

COMPUTERIZED PIPELINE OPERATION SYSTEM

A timesharing computer system, including a pipeline flow simulator, was implemented to meet Badger's needs. The computerized operation system (Figure 2) was designed to be executed totally in IBM/TSO foreground and can be divided into three functional areas: (1) Batch nomination sequencing, (2) Flow simulation, and (3) Report processing.

Batch Nomination Sequencing

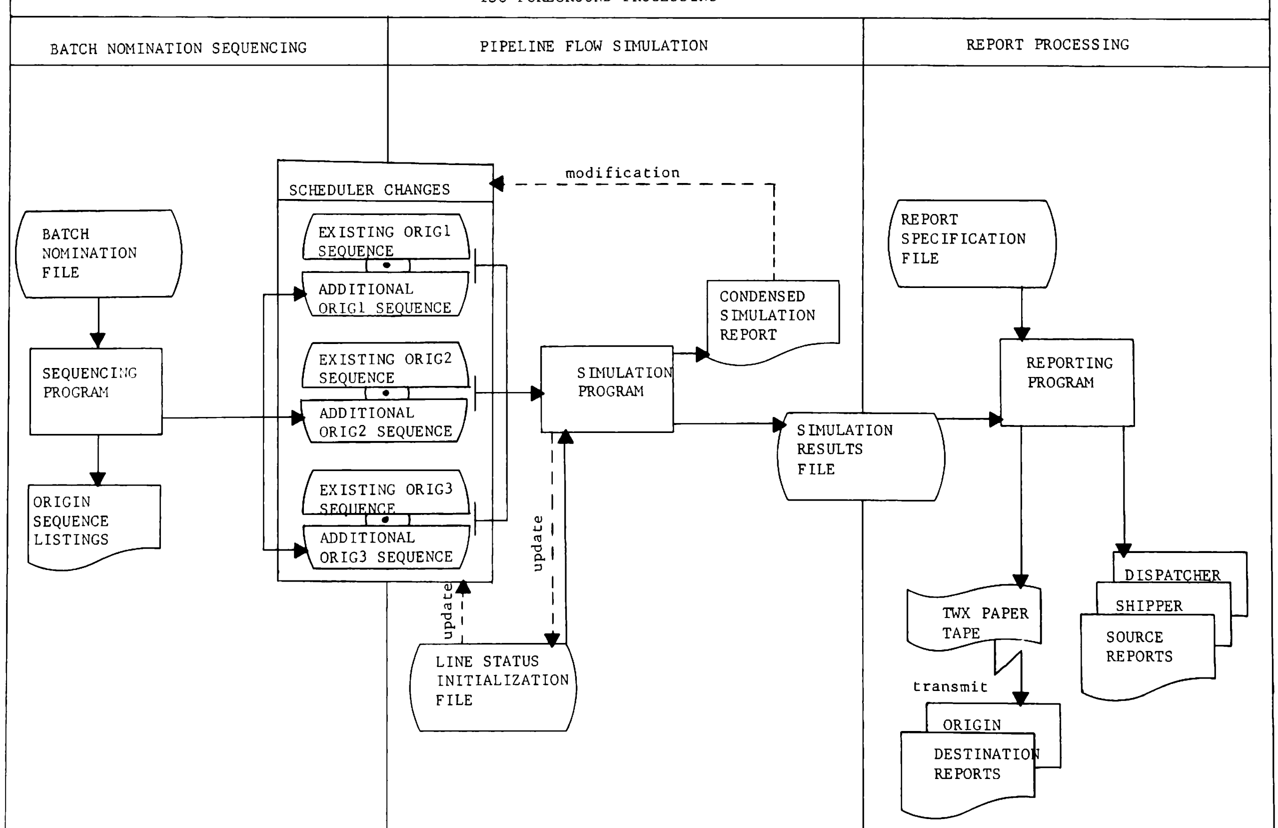
The first phase of the system develops initial feasible pump sequences for each of the origins from the month's batch nominations. The batch numbers on the nomination request forms (Figure 3) are constructed with codes indicating the shipper, origin, source, product, grade, cycle requested, destination, and cosignee of the volumes specified. The batch nominations are keyed into a data file as they are received, and the sequencing program is run after all of the requests are received.

The pumping sequences produced for each origin-cycle are developed from historically derived rules for producing good feasible schedules. The procedure employed is based on breaking the total origin-destination volumes into destination-product-grade pumping slugs that meet the product sequencing and linefill constraints. The exact sequence of slugs will depend on the characteristics of the total cycle volumes, but will result in an efficient utilization of the common line segments among the origins. Within each slug, the individual batches are sequenced according to shipper and source rules designed to facilitate contiguous pumpings.

A formatted file containing the sequenced batch nominations is generated for each of the origins. Each of the output files is concatinated to the respective origin

Figure 2

TSO FOREGROUND PROCESSING



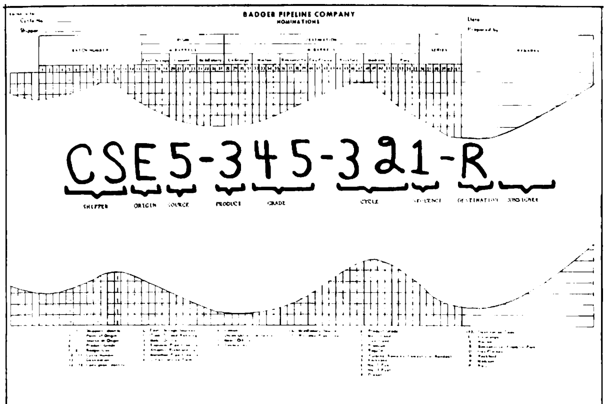


Figure 3

sequence file to provide continuous input to the flow simulator. A copy of the sequences is provided for scheduler review along with cycle summaries of gross product movements to aid the schedulers in determining the necessary flow rates.

Flow Simulation

Once the new batch sequences have been added to the existing files, the schedulers make any initial modifications to the pumping sequences they desire. The simulator had to have the flexibility to allow the schedulers complete control over the flow timing. This control is provided through macro "batch-commands" which can be coded with the batch information appearing in the sequence files to activate flow rate, flow delay, or flow interaction events. These variable events initiate the flow timing changes in the simulation when the coded batch reaches the discrete point in the network designated by the command. After the scheduler has encoded the desired control information, the flow simulator is executed producing a report of gross schedule timing information. The scheduler will alternate modifying the pump sequence files, providing better flow interaction or accounting for nomination changes, and executing the simulator.

Report Processing

The report phase of the system is executed when the simulation output satisfies the scheduler. He is able to select the reports (Figure 4) which are produced and the time period covered in each. All of the reports are generated from a simulation output file which contains the complete and partial batch pumping and receiving times, as well as origin flow rate information. Shipper, source, and dispatcher reports are distributed while the origin and destination reports are created on paper tape with the necessary TWX control bytes for automatic transmission to the stations.

PIPELINE FLOW SIMULATOR

The pipeline flow simulator used in the operation system is based on a deterministic model with discrete events, and utilizes an asynchronous event-advanced clocking mechanism. This simplistic model design was chosen over a more sophisticated one because the added precision was not required. The results of the model chosen are well within the tolerance levels of the manual system.

Simulation Control

The simulator's action is governed by two elements: (1) The origin pump sequences provided, which dictate the fixed event changes in flow patterns over time, and (2) The variable events, input as batch-commands, which affect the timing of the flow pattern changes. The batch-commands at the disposal of the schedulers provide the

DISPATCHER REPORT

LEMONT DISPATCHERS SCHEDULE DATE 12/02/75

SCHEDULE 12/02/75 TO 12/10/75

RATE	TRAN	BATCH NUMBER	VOLUME	TIME	RATE CHNG LOC	BATCH NUMBER	VOLUME
	STRT	TXL3-410-341D	99.0	12/01 2113			
	STRT	MBL2-700-341B	20.0	12/02 1935			
	STRT	TXL3-701-341D	14.0	12/03 0002			
3800	CH R		6.5	12/03 0141	BV	MBL2-700-341B	20.0
	STRT	MBL2-700-341W	25.0	12/03 0324			

DESTINATION REPORT

BENSENVILLE FROM TULSA DATE 12/02/75
SCHEDULE 12/02/75 TO 12/10/75

BATCH NUMBER	VOLUME	START	FINISH	CONSIGNEE
MBL2-700-341B	20.0	12/03 0141	12/03 0657	
MBE4-500-341B	10.0	12/03 1759	12/03 2142	
MBE5-340-341B	40.0	12/06 0749	12/06 2109	
MBL2-340-341B	30.0	12/09 1814	12/10 0045	

FOR MOBIL OIL CORP TANKAGE

BATCH NUMBER	VOLUME	START	FINISH	CONSIGNEE
MBL2-700-341B	20.0	12/03 0141	12/03 0657	

SOURCE REPORT

TO ATLANTIC RICHFIELD DATE 12/02/75
BADGER PIPE LINE SCHEDULE FOR 12/02/75 TO 12/10/75

PUMP DATE	BATCH NUMBER	VOLUME M BBLS	TERMINAL DELIVERY
12/02	ARE4-500-341D	8.7	ATLANTIC RICHFIELD

ORIGIN REPORT

LEMONT FROM TULSA DATE 12/02/75
PUMP SCHEDULE 12/02/75 TO 12/10/75

BATCH NUMBER	VOLUME	START	FINISH
TXL3-410-341D	99.0	12/01 2113	12/02 1935
MBL2-700-341B	20.0	12/02 1935	12/03 0001
TXL3-701-341D	14.0	12/03 0002	12/03 0324
MBL2-700-341W	25.0	12/03 0324	12/03 1001

SHIPPER REPORT

TO CITIES SERVICE OIL
BADGER PIPE LINE SCHEDULE FOR 12/02/75 TO 12/10/75

PUMP DATE	BATCH NUMBER	VOLUME M BBLS	VOLUME M BBLS	DELIVERY DATE	TERMINAL
12/03	CSL2-700-341W	10.0	10.0	12/05 - 12/06	MADISON
12/04 - 12/05	CSL1-010-341D	15.0	15.0	12/05	DES PLA
12/08	CSL2-330-341W	20.0	20.0	12/11	MADISON
	CSL2-330-341D		7.2	12/10	DES PLA
12/09	CSL2-330-341D	50.0			
	CSL2-330-341D		42.8	12/09	DES PLAI

FROM UNION OIL OF CALIF LEMONT

PUMP DATE	BATCH NUMBER	VOLUME M BBLS
12/04 - 12/05	CSL1-010-341D	15.0

Figure 4

flexibility and control to emulate any flow activity eminating from the origins. They are keyed into the sequence files following the batches which are designated to initiate the flow controlling process. The format of the batch-commands (Figure 5) consists of an event code, an event point, and flow control information. The event processing occurs at the time the controlling batch's simulated movement reaches the designated discrete point in the network. The event processing results in an immediate system change when the event is a flow rate change, flow preemption, or a flow strip, and will additionally result in a pending system change when the event is a relative time flow delay, calendar time flow delay, or a matching condition delay. "Dummy" batch control is provided for use when discrete point batch arrivals do not approximate the necessary controlling process. The dummy batches initiate system change via time delay only.

The particular origin flow pattern affected by batch-command processing is the one on which the controlling batch resides at the time it arrives at the event point. The WA command will result in a flow delay for a designated time period beginning at the controlling batch's arrival. The FW command will cause a flow delay from the time of the controlling batch's arrival until a designated calendar time. The WA command is usually used to make schedule timing adjustments while the FW command is used to initiate fixed startups. The CO command is used to delay flow for an unknown period of time until a releasing AP command is processed in the course of another flow. This matching condition delay is primarily used to postpone eminent flow pattern changes until the common line segments required for use are freed by another origin.

The PO command is used to terminate conflicting flow or to delay conflicting flow pattern changes until a preemptive flow is complete. The preempt command is used in situations where the normal cycle operation is interrupted to permit a special requested delivery. The CR command is used to approximate, in a stepped average fashion, the flow rate changes resulting from changing product linefill, destinations, or pump unit combinations. The ST command is used to alter the flow stream characteristics when two destinations are to receive product from the same origin flow. The dummy batch control is generally used in conjunction with the delay commands to terminate flows during holidays, line repair, etc.

Simulation Methodology

The simulator initializes line status, flow status, and pending flow alterations from a system status file which is automatically updated at the time special program runs are made to eliminate historical portions of the sequence files. The simulation procedure employed is:

1. Determine the next scheduled event occurrence (the completion of a flow delay or a discrete point arrival of a batch).
2. Update the clock and the segment positions of all the batches to this time.
3. Update the system and/or create pending events for any batch-commands designated for execution at the batch arrival point.
4. Process any released conditional events and update accordingly.
5. Determine any change in flow pattern necessitated by the batch's current position and flow requirements and update accordingly.
6. Write any relevant timing or flow rate change information to the simulation output file.
7. Access the batch information from a sequence file for any origin requiring a batch for pumping.
8. Repeat the procedure until schedule completion.

As each batch enteres the simulation, the route of line segments which it is required to take is determined from the codes in its batch identification. The line segment end points in the determined route represent the points in the pipeline network at which the batch can affect a flow pattern change. As the simulation events occur and require batch movement, the barrel distance from the next possible event point is decremented. The barrel displacement in each of the line segments will vary according to the season of the year (average temperature).

Registers are maintained for each origin indicating the flow pattern active and the flow rate to aid in determining the batch movements and system changes required. Registers are also maintained at each destination and junction indicating the active batch and the barrels remaining to pass. These registers are required for determining the occurrences of line injections and partial receiving times.

Flow Processing Example

Actual simulation processing is dependent on the flow interaction inherent in the three sequence files and the initial network linefill. A complete processing example of this is difficult to follow, therefore, the illustration limits itself to the processing of one batch's movement. Reference will be made to Figure 5 for the example illustration.

Assume that the following batch information is next in origin point 2's sequence file when a batch is required for pumping.

CSE4-345-161R 0200
AP07a CO07b CR072800

The simulator determines from the batch codes that a batch of 20,000 barrels is required to travel route segments 4, 5, 7, and 9 to destination point 9. In actual processing, the batch's movement along these routes can be delayed at any time by flow preemptions, dummy batch control, and delay command processing initiated by controlling batches residing on flow patterns including the batches position. For example simplicity, the illustration is presented as if these did not occur.

The batch's movement along segment 4 cannot be interrupted since the segment is necessarily part of any flow eminating from origin 2. The particular flow route on which the batch resides is dictated by the network linefill and will change as required without delaying the batch's movement. The rate at which it travels toward point 5 (or any point) is determined by the CR command processing along the flow route. When the batch reaches point 5, the simulator will recognize that the flow must continue along segment 5 to satisfy the batch's routing requirement. If the flow at this time includes segment 2, the simulator will transfer flow to a flow pattern which includes segment 5 and which satisfies all the requirements of the batches within. Since there was no CR command for processing at point 5, to indicate a switch in destinations, no pattern change was required. During the time that the batch is passing through point 5 (or when it passes through point 7), it is subject to being split by another flow's injection. After the batch passes point 5, two things can occur: (1) Any batch arriving at point 5 from segment 1, which requires flow on segment 5, will cause the flow pattern to shift to origin 1 (it is assumed that this does not occur), and (2) Any following batch, on the existing flow route, which arrives at point 5 and requires delivery to destination points 3 or 4 will transfer the flow pattern and halt the flow on segment 5 (it is assumed that this does not occur). In situation 2, the batch's movement will not resume until another batch arrives at point 5 which requires flow on segment 5. The batch's arrival at point 6 will activate a simulator check to insure that segment 7 is on the current flow rate; if the preceding batch was delivering into point 6, an appropriate pattern change

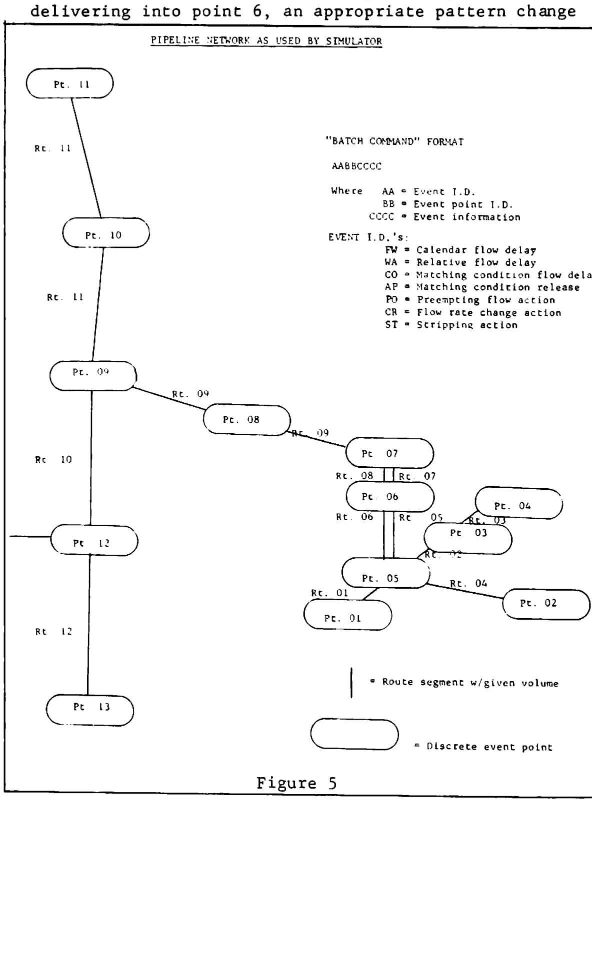


Figure 5

would take place to include segment 7. Any following batches arriving for delivery to point 6 from segment 5 will halt the batch's flow to point 7 during the delivery time. When the batch arrives at point 7, the simulator processes the batch commands. If the matching AP (from another flow) has not been processed, the flow will halt until it has been. The flow will resume at the new flow rate indicated after the simulator insures that segment 9 is in the flow pattern. Once the batch has passed point 7, it becomes a candidate for flow patterns including segments 6 and 8, and can be delayed by following batches with a delivery point 7. On the batch's arrival at point 9, a pattern switch into point 9 will be made if the preceding batch was not delivering there.

CONCLUSION

The discrete pipeline flow simulator has proved to be a viable component in the computerized scheduling system designed for Badger Pipeline. It provides the opportunity to quickly generate effective operational schedules in a fraction of the time it took under the manual system. It is currently planned to enhance the operation system with a pump unit optimization program. This program would determine the pump unit combinations which would minimize the electrical charges and at the same time complete the pumping sequences provided in the required time. Since the flow rates would be determined over the schedule range, the simulator requirement for the CR command could be eliminated through a direct system interface.