WATER RESOURCES RESEARCH, VOL. 44, W02426, doi:10.1029/2007WR006192, 2008

Shortcomings of linear programming in optimizing river basin allocation

Nesa Ilich[1]

Received 22 May 2007; revised 12 November 2007; accepted 10 December 2007; published 19 February 2008.

[1] Numerous computer models for river basin planning and management have been developed and used extensively since the mid-1970s. Early developments have relied on the use of network flow algorithms (NFA), due mainly to higher execution speed than the standard Simplex solvers. However, subsequent efforts to include proper modeling of hydraulic and hydrologic constraints introduced iterative schemes into the NFA-based models, which diminished the initial advantages in execution speed and which also caused concerns over the accuracy of the convergence schemes. Hence full-blown commercial linear programming (LP) solvers were introduced as a replacement to the iterative solution strategy of the NFA approach. This paper demonstrates one possible failure to solve a simple allocation problem using the NFA-based model and shows how this problem can be solved using the standard LP approach. It then identifies cases when even a full-blown LP approach cannot properly model two critical aspects of river basin management, one related to reservoirs with multiple outflows and the other one related to modeling of hydrologic channel routing. For NFA-based models the failures are the result of the inability to include relationships between flows on different model components directly into the search process. For the models based on LP solvers, the failures are caused by the fact that integrated reservoir outflow capacity between the starting and the ending storage levels is assumed over the entire length of the assumed time step, while the actual outflow can only take place during the portion of the time step when the storage level is above the invert of the outlet structure.

Citation: Ilich, N. (2008), Shortcomings of linear programming in optimizing river basin allocation, *Water Resour. Res.*, *44*, W02426, doi:10.1029/2007WR006192.

1. Introduction

[2] River basin management models introduced the need to model different allocation priorities and deficit-sharing policies among various water users. In such models the available river flows could completely bypass the upstream users and be allocated to a downstream user, or the other way around. These models require a complex set of rules that account for every possible combination of supply and demand conditions (hence the term "rule based" models), or a built-in optimization solver that treats the allocation problem as a mathematical program. Initial review of reservoir operation models for basin planning purposes was compiled by *Yeh* [1985] and later updated by *Wurbs* [1993] and *Labadie* [2004]. The primary focus of this paper is a large group of deterministic linear programming (LP)-based optimization models that found widespread application in practice.

[3] Some modeling standards do exist in the water resources sector nowadays. For example, the use of HEC-RAS for river hydraulics analyses has become part of standard curricula at many engineering schools in North America and overseas. However, in the area of basin allocation modeling, there is neither a universally accepted model nor a general agreement on the minimum mandatory technical specifications for it in terms of its capabilities, solution techniques, limitations, and a number of benchmark test problems that would allow commercial models to be verified and compared by potential users. This is not so in the operations research field, where any vendor offering a new LP solver to the market is greeted by over 30 tough and well-established benchmark test problems with known solutions that the new vendor would be expected to match. This prevents the vendors of LP solvers to market their products without first publicly demonstrating their capabilities. In spite of numerous papers on optimization in river basin planning and water resources in general, established and widely acceptable benchmark problems are still nonexistent. Once the standard set of mandatory constraints are established and generally agreed upon, testing of other vendors' solutions could be made transparent to anyone.

[4] The priority of supply was initially aimed at representing the water licensing system still in use in North America from where most of the early model development originates. Water licensing systems can be represented by an LP formulation. Hence the early efforts focused on implementation of efficient LP solvers with typical objectives of finding the minimum cost flow in a network. One such

[1]Department of Civil Engineering, University of Calgary, Calgary, Alberta, Canada.

0043-1397/08/2007WR006192$09.00

specialized solver was the Out-of-Kilter algorithm [*Fulkerson*, 1961]. Consider a network consisting of ordered pairs (i, j) of arcs A and a total of N nodes i, j. The minimum cost flow problem is defined as the following linear program:

$$\text{Minimize} \quad \sum_{(i,j)\in A} c_{ij}x_{ij} \qquad \forall i,\ j \in N \tag{1}$$

subject to

$$\sum_i x_{ij} - \sum_i x_{ji} = 0 \qquad \forall j \in N \tag{2}$$

$$0 \le l_{ij} \le x_{ij} \le u_{ij} \qquad \forall (i,\ j) \in A, \tag{3}$$

where c_{ij}, l_{ij}, x_{ij}, and u_{ij} are the cost per unit of flow, lower bound, flow, and the upper bound on flow along an arc (i, j), respectively, while constraint (2) represents the mass balance at each node. Early river basin planning models which utilized the Out-of-Kilter algorithm were define SIMYLD [*Evenson and Moseley*, 1970], ARSP [*Sigvaldason*, 1976], MODSIM3 [*Labadie et al.*, 1986], WASP [*Kuczera and Diment*, 1988], DWRSIM [*Chung et al.*, 1989], CRAM [*Brendecke*, 1989], KCOM [*Andrews et al.*, 1993], and WRMM [*Ilich et al.*, 2000b]. Most of these models are still in use, and some early versions have evolved to more sophisticated phases, e.g., DWRSIM has now been replaced with CALSIM [*Draper et al.*, 2004], which uses a mixed-integer program (MIP) solver, and it is still maintained and used actively by the California Department of Water Resources. In some models, such as the SIMYLD or the WRMM, the original version of the Out-of-Kilter algorithm has been replaced with alternative variants which were proven to be significantly faster such as the SUPERK algorithm of *Barr et al.* [1974] or, as in the case of the MODSIM model, the Relax4 network flow solver of *Bertsekas and Tseng* [1988]. These models have been used extensively in river basin planning studies. Nonlinear constraints have been handled by using successive iterations within a time step until a desired convergence is achieved. This is done by initially guessing the flow bounds u_{ij}, solving the minimum cost flow problem, evaluating the network flow solution against the assumed bounds, resetting the bounds to new values based on the previous solution, and reiterating if necessary until the assumed bounds and the network flow solution were within a reasonable tolerance limit. Although the use of iterations is openly acknowledged in the accompanying documentation of some of the models such as MODSIM, ARSP, or REALM [*Department of Sustainability and Environment*, 2006], no information is available on the success of convergence or the impact of iterations on the quality of the converged solutions. The iterative process is typically associated with resetting the reservoir outflow limits, diversion flows at unregulated weirs where maximum flow diversion is a function of the available flow in the river, irrigation return flows, hydropower output or canals losses; in general, in situations where flow in one component depends on the flow in another component. The resulting solution from one iterative call of the solver becomes the starting point for the next iteration. As demonstrated on a simple test problem in this paper, there is no guarantee that this iterative process must converge to the optimal solution that can be obtained using a full LP formulation with piecewise linearization of non-network constraints, since such problem formulation does not require iterations.

[5] To avoid iterations, non-network and segmented nonlinear constraints were included directly into the MIP formulation in models such as CALSIM, RIVERWARE [*Zagona et al.*, 2001], HEC-FCLP [*Needham et al.*, 2000], or OASIS [*Randall et al.*, 1997]. Some of these models are also capable of optimizing allocation over single or multiple time steps, although there is no universally accepted methodology on how to utilize multiple time step solutions for the development of practical short-term operating rules. Development of such methodology is outside of the scope of this paper.

[6] Results of the models listed above have been used mainly in river basin planning studies, and they were dependent on the nature of the historic hydrologic time series that was typically used as input. The fact that the historic series will never repeat itself in the same fashion, along with doubts that the historic series is a legitimate representative of the current and future conditions, introduces the need to consider uncertainty of reservoir inflows into modeling. Various ways have been proposed for this, but one of the most promising is implicit stochastic optimization, where stochastic hydrologic time series are first produced as an alternative to the historic series and used as hydrologic inputs [*Koutsoyiannis and Economou*, 2003]. Also, innovations in algorithmic developments that may enable generation of stochastic series with shorter time step and simultaneously preserve multiple autocorrelations and cross correlations of arbitrary time lag n have also emerged recently [*Ilich and Despotovic*, 2007]. Whether historic or stochastic series are used as hydrologic input is transparent to the optimization process, which treats either historic or synthetic series as known inputs. A recognized difficulty is to relate the results of planning models to the real-time operation where inflows are not known in advance. *Nalbantis and Koutsoyiannis* [1997] favor development of parametric rules as a simple guideline that operators can understand and follow, such as, for example, the target elevation that should be reached for each reservoir at the end of a given time step. Another way to tackle the issue of hydrologic uncertainties may be to develop short-term operational model based on the following concept: (1) Develop and use stochastic hydrologic series as hydrologic input; (2) use stochastic hydrologic series obtained in point 1 to conduct basin-wide multiple time step optimization in order to form a database of "perfect" solutions of reservoir releases and diversion flows; (3) use pattern matching techniques based on a range of options, starting from the simplest based on multiple regression in combination with reservoir operating zones [*Ilich et al.*, 2000a] to the latest developments in artificial intelligence such as the artificial neural networks or support vector machines that can "learn" from the database of optimal solutions developed in step 2; and (4) verify the proposed short-term operational model by applying it using a discrete set of short time steps with historic data. Note that the potential benefits of the short-term models developed in this fashion can be assessed by comparing the model output for the recent historic series of inflows with the historic operation of the system, and

Table 1. Input Data for Test Problem 1

Volume, 1000 m^3	Elevation, m	Outflow, m^3/s	Elevation, m
0.000	1653.54	0.000	1660.00
772.030	1656.00	1.850	1661.00
1960.51	1659.00	3.250	1662.00
2412.63	1660.00	4.364	1663.00
2892.74	1661.00		
3400.83	1662.00		
3936.90	1663.00		

comment on the added benefits arising from the use of the model. Again, in this approach, step 2 involves application of deterministic optimization models considered in this paper. There may be other strategies for generating short-term reservoir operating rules. Which one will be the most successful is subject to future research and practice. However, all this is outside of the scope of this paper. This paper deals with difficulties within the realm of the deterministic LP framework, which has so far been the most frequently used approach in practice (the majority of the referenced models used by various agencies are LP-based). This raises the importance of the ability to solve deterministic optimization problems properly.

[7] Test Problem 1 demonstrates a failure to find accurate solutions on sample problems using iterative calls to network flow solvers. Modified Test Problem 1 includes discussion on how iterations can be avoided by adding segmented linearized reservoir outflow constrained to fit a more general formulation within a wider LP context, as well as a case where even this formulation fails. Finally, Test Problem 2 discusses implications of including hydrologic channel routing as a constraint into a linear program, followed by conclusions and recommendations. Appendix A includes technical discussion on the necessity to use mixed integer programming, backed by a numerical example.

2. Test Problem 1

[8] This test problem demonstrates breakdown of an iterative scheme on modeling a reservoir with two distinct outflow structures. Reservoir outflows are limited by the capacity of the outlet structure (a weir or an orifice). They are a function of the average available storage over a given time step and the geometry of the structure that determines the flow area. Reservoir elevation determines the maximum possible outflow at each point during a given time period. However, reservoir elevation also changes during the time period as a result of the overall mass balance of inflows and outflows. Also, many reservoirs have more than one outlet structure which introduces additional complexity during simultaneous operation. The average outflow capacity over a time step is the integrated average from the beginning to the end of the calculation time step. An NFA-based model must "guess" the final elevation for a time step in order to numerically integrate the average outflow capacity and thus estimate the upper bound on reservoir outflow; hence the need to resort to iterative calls of the network flow solver, such that the guessed value is improved from iteration to iteration until it is sufficiently close to the calculated value. Iterative procedures may fail to converge to a sensible solution in complex systems with multiple reservoirs and multiple outlet structures.

[9] The sample test run for an NFA-based models is presented as an introduction to its related problem for a full LP formulation presented in this paper. The volume versus elevation and the outflow versus elevation relationships for this problem are given in Table 1, and the modeling schematic representing this problem is shown in Figure 1. The input data include reservoir average weekly inflow of 10 m^3/s, municipal demand of 3.25 m^3/s, irrigation demand of 12 m^3/s, and starting reservoir elevation of 1662.0 m. There is one reservoir with two outflows, one for municipal water supply through an orifice and the other one for irrigation supply through a large capacity bottom outlet. Since the bottom outlet is capable of emptying the reservoir within a time step of 1 week, it does not require inclusion of the outflow versus elevation curve. That is not the case with the municipal supply outlet, which is given the highest allocation priority, followed by irrigation and storage with their respective cost factors of 100, 10, and 1. These cost factors can be viewed as penalties per unit flow of deficit, where deficit is the deviation from the target demand and the objective function is converted to a goal programming formulation by replacing x_{ij} with $(u_{ij} - x_{ij})$ in expression (1), or they can be viewed as payoff factors if the minimization problem (1)–(3) is converted to maximization by reversing the signs of the cost factors. The following paragraphs analyze the results of NFA solution to this problem in an iterative manner using the weekly calculation time step:

[10] 1. The initial outflow capacity for orifice outflow is set to the initial reservoir level of 1662 m, which corresponds to 3.25 m^3/s. For ease of demonstration, this outflow capacity is set equal to the municipal demand.

[11] 2. The solution derived by the model significantly depletes the storage due to the relatively large downstream water demands compared with the available storage and inflow.

[12] 3. The model then evaluates the solution obtained in step 2 to check compliance with the constraints. It calculates

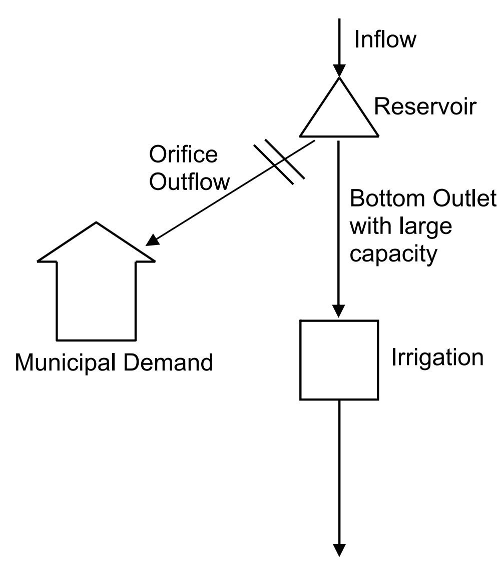

Figure 1. Problem 1 modeling schematic.

Table 2. Iterative Reservoir Outflow Solutions for Test Problem 1

Iteration	Municipal Supply, m^3/s	Irrigation Supply, m^3/s
1	3.250	12.000
2	0.975	12.000
3	0.971	12.000
4	0.969	12.000

the average orifice outflow capacity based on time-integration of reservoir outflows for the entire week by starting from elevation 1662 m and calculating new elevation (and the corresponding maximum outflow) at the end of each time increment assuming steady state inflow of 10 m^3/s and outflows which are 12 m^3/s for the bottom outlet and the minimum of 3.25 m^3/s and the limits imposed by the outflow versus elevation curve given in Table 1. To achieve reasonably accurate time integration of outflow capacity, the model uses a sequential series of time increments equal to 1/30th of the week (5.6 hours) and updates the reservoir elevation at the end of each time increment.

[13] 4. The new outflow capacity of 0.975 m^3/s obtained in step 3 is checked with the assumed outflow capacity of 3.25 m^3/s. Since there is a large difference between the two, the process is repeated starting from step 1 and assuming the average outflow capacity over a time step of 0.975 m^3/s. Steps 2–4 are thus repeated in an iterative manner until the assumed outflow capacity approximately equals the outflow derived by integration of outflow capacity in step 3.

[14] The final solution after several iterations converges to municipal outflow of 0.969 m^3/s, irrigation supply of 12 m^3/s, and the ending reservoir level of 1658.103 m, which corresponds to the storage deficit of 3.850 m^3/s (this is the flow required to get the storage back to the full supply level of 1663 m within a 7-day period, and hence it is expressed in the units of flow, as are all other components to conform to the requirements of the network flow algorithms). For arbitrarily assumed respective cost of deficit per unit of flow of 500, 10, and 1 for municipal, irrigation and storage component, the objective function (expressed in the goal programming formulation as the total cost of deficit) of this solution is

$$\begin{aligned}\text{total cost} &= 500(3.25 - 0.969) + 10(12 - 12) + 1(3.86)\\ &= 1144.36\end{aligned} \tag{4}$$

[15] The assumed costs can be viewed as a loss of revenue in monetary terms per unit of deficit flow, or they can be a subset of a larger set of priorities if the entire modeling schematic is considered as a small part of a larger system. Whichever case is considered is not essential for further analyses. Deficits are calculated in brackets as the difference between the stated target and the achieved supply. It is easy to see that the above solution is far from the best. A much better (and optimal) solution can be obtained by assuming that the reservoir remains at its starting level during the entire time interval. This would result in 3.25 m^3/s allocated to municipal demand and 6.75 m^3/s to irrigation, while the storage level remains unchanged from its starting level of 1662 m, which gives a smaller storage deficits equivalent to a weekly flow of 0.89 m^3/s. Storage deficit is evaluated as the difference between the target volume at full supply level storage capacity and the ending volume for a week, divided by the length of the time step, hence the units of flow. The corresponding value of the objective function is then

$$\begin{aligned}\text{total cost} &= 500(3.25 - 3.25) + 10(12 - 6.75) + 1(0.89)\\ &= 53.39\end{aligned} \tag{5}$$

This solution is superior to the one found by the iterative approach, as it fully meets the municipal demand which has the highest priority. The NFA solver can only take into account a fixed value of the flow upper bound in each iteration, be it 3.25 m^3/s in the first iteration or 0.969 m^3/s in the final, while the lowest pricing vector on storage ensures that either municipal demand or irrigation draw the reservoir down. Network flow solvers cannot address inherent relationship between two or more network flow variables. In this case emptying reservoir storage for irrigation affects the available supply to the municipality. Table 2 shows the results of all iterations until convergence is achieved. This problem was tested using the NFA version of the WRMM model as well as the version of the MODSIM model used by USBR. Both models converged to the same solution.

[16] An attempt to increase the storage penalty above that of the irrigation penalty for the storage segment below the invert of the orifice is not a good solution, since it would render a large segment of storage inaccessible to irrigation at all times, which may not be desirable. Any attempt to shorten the time step will make no difference. Under the NFA framework, reservoir storage will be selfishly depleted by the downstream irrigation component regardless of the time step length. Yet storage depletion that encroaches on supply to the municipal demand component is exactly what should be prevented to achieve the best solution. This failure is not the result of an unusual choice of the input data. To demonstrate this, Table 3a shows the NFA solutions of the same problem for a variety of inflows, ranging from 2 m^3/s to 14 m^3/s in

Table 3a. NFA Solutions of Test Problem 1 for a Range of Inflows

Test Run	Inflow, m^3/s	Week Ending Storage, m	Municipal Supply, m^3/s	Irrigation Supply, m^3/s	Storage Deficit Penalty	Municipal Supply Deficit Penalty	Irrigation Supply Deficit Penalty	Total Penalty
1	2.0	1653.54	0.511	7.112	6.51	1369.50	48.88	1424.89
2	4.0	1653.54	0.511	9.112	6.51	1369.50	28.88	1404.89
3	6.0	1653.54	0.511	11.112	6.51	1369.50	8.88	1384.89
4	8.0	1655.47	0.623	12.000	5.51	1313.50	0.00	1319.01
5	10.0	1658.10	0.969	12.000	3.86	1140.50	0.00	1144.36
6	12.0	1659.92	1.693	12.000	2.58	778.50	0.00	781.08
7	14.0	1661.19	2.683	12.000	1.57	283.50	0.00	285.07

Table 3b. Correct Solutions of Test Problem 1 for a Range of Inflows

Test Run	Inflow	Week Ending Storage, m	Municipal Supply, m^3/s	Irrigation Supply, m^3/s	Storage Deficit Penalty	Municipal Supply Deficit Penalty	Irrigation Supply Deficit Penalty	Total Penalty
1	2.0	1661.19	2.682	0.000	1.57	284.00	120.00	405.57
2	4.0	1662.00	3.250	0.750	0.89	0.00	112.50	113.39
3	6.0	1662.00	3.250	2.750	0.89	0.00	92.50	93.39
4	8.0	1662.00	3.250	4.750	0.89	0.00	72.50	73.39
5	10.0	1662.00	3.250	6.750	0.89	0.00	52.50	53.39
6	12.0	1662.00	3.250	8.750	0.89	0.00	32.50	33.39
7	14.0	1662.00	3.250	10.750	0.89	0.00	12.50	13.39

increments of 2 m^3/s solved for a single time step using the same starting reservoir level, and Table 3b shows the correct solutions obtained using the full LP formulation which is detailed in the following.

3. Modified Test Problem 1

[17] A method for linearization of reservoir outflow constraints within a classical LP framework relies on the use of MIP solvers, as outlined in the following. Also included is a discussion on the accuracy of this representation for a system of reservoir with two outflows presented in the Test Problem 1 above. Single time step optimization is chosen for demonstration purposes as it is easier to follow and verify. Failure to deliver accurate solution in single time step optimization (STO) mode cannot be fixed by mere introduction of multiple time step optimization (MTO), since the nature of LP constraints in STO and MTO is identical.

[18] As part of setting up the constraint matrix, reservoir storage is segmented into the zones that correspond to the number of segments in the outflow versus storage curve as shown in Figure 2. For a given time interval, storage can be converted to the units of flow by being divided with the length of the time step t. Hence for any starting and ending storage (Vs/t and Ve/t) within a given storage segment, the maximum outflow capacity $Q\max(o)$ can be related to average storage over the time interval using the slope S of the segment in the linearized outflow curve as:

$$Q\max(o) = \frac{1}{S}\cdot\frac{1}{2}\left(\frac{Vs}{t}+\frac{Ve}{t}\right). \tag{6}$$

[19] The term $Q\max(o)$ represents the upper bound on flow in the outlet structure channel for any storage in the respective reservoir storage zone. Hence the constraint for limiting the outflow from a single reservoir zone can be written as

$$Q(o) \leq \frac{1}{S}\cdot\frac{1}{2}\left(\frac{Vs}{t}+\frac{Ve}{t}\right). \tag{7}$$

[20] When solving an individual time step, both $Q(o)$ and Ve/t are decision variables and only the initial storage Vs/t is known. The above relationship imposes a limit on the outflow capacity related to one storage zone. If the ending reservoir elevation stays within the same zone as the initial, the maximum outflow capacity would correspond to capacity for the middle storage between the initial Vs/t and the ending storage Ve/t.

[21] When multiple reservoir zones i are used, the total integrated outflow must conform to

$$Qt(o) \leq \sum_{i=1}^{n}\frac{1}{Si}\cdot\frac{1}{2}\left(\frac{Vs(i)}{t}+\frac{Ve(i)}{t}\right). \tag{8}$$

[22] Term $Vs(i)$ represents the starting volume at the beginning of the time step for each zone i. The sum of the initial volume in all zones i is the starting volume at the beginning of time step t. The above approach has initially been proposed and adopted by *Windsor* [1973] as a representation of reservoir outflow constraints. In addition to the above expression, which represents reservoir outflow limitation as a function of segmented storage and the outflow curve capacity, the other simultaneous constraint that must be satisfied is the mass balance of inflows, outflows, and storage change.

[23] The above LP formulation is only valid if reservoir zones are filled from bottom to top and emptied from top to

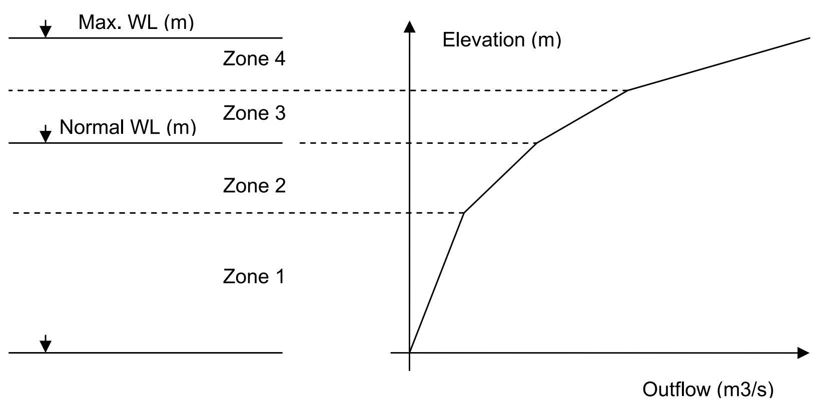

Figure 2. Reservoir zones as a function of the linear approximation of the outflow versus elevation curve.

bottom. While penalty settings may help achieve this sequence of filling and emptying in some simple cases, they are unable to guarantee it in every instance. A failure to fill storage zones from top to bottom using only penalty factors is demonstrated on a simple problem in Appendix A. To overcome this, a binary variable associated with each reservoir zone is introduced and the problem is reformulated as a mixed-integer program (MIP) such that a complete formulation also includes the binary variables in the set of decision variables. The downside is that MIP solvers require significantly larger computational effort. However, the inclusion of binary variables is necessary if the model is to guarantee the proper sequence of filling or emptying of storage, resulting in solutions that are physically possible. Otherwise the reservoir inflows may be routed downstream through the upper storage zones, which have sufficient outflow capacity, leaving the lower storage zones empty. This situation could be triggered by a high-priority demand downstream of the reservoir, when the reservoir has insufficient elevation to support the adequate outflow capacity.

[24] Binary variables are integers with possible values of 0 and 1. Most commercial LP solvers recognize binary variables as a distinct category, which eliminates the need to specify the variable bounds. Assuming Z_i are the binary variables associated with reservoir zones i, and U_i are the upper bounds in the units of flow for the respective zones, where flow represents storage of each reservoir segment divided by the length of the time interval, it is possible to define the additional constraints as follows:

$$U_i Z_{i+1} \leq X_i \leq U_i Z_i \qquad i = 1,\ n - 1. \tag{9}$$

[25] Variable X_i represents storage in zone i. Reservoir zone i is full if $X_i = U_i$. The working of expression (9) is demonstrated for the first two zones starting from the bottom: (1) Storage is in the bottom zone, which requires that $Z_1 = 1$ and $Z_2 = 0$, hence $0 \leq X_1 \leq U_1$; (2) when storage is in the first zone above the bottom zone, i.e., $Z_1 = 1$, $Z_2 = 1$, and $Z_3 = 0$, expression (9) implies that four conditions must be met simultaneously: $X_1 \leq U_1 Z_1$, which becomes $X_1 = U_1$ when the bottom zone is full; $U_1 Z_2 \leq X_1$, which also becomes $U_1 = X_1$ for $Z_2 = 1$; $X_2 \leq U_2 Z_2$, which works the same as step 1 above written for zone 2; and $U_2 Z_3 \leq X_2$, which becomes $0 \leq X_2$.

[26] Of all the above conditions, the first two ensure that $X_1 = U_1$ before X_2 can take on any value above zero, which is equivalent to forcing the storage to fill the bottom zone first before filling the upper zone. In other words, the above set of constraints ensures that if there is any water in zone 2 above the bottom (i.e., if $Z_2 = 1$ and $0 < X_2$), the storage zone 1 must be filled first (i.e., $X_1 = U_1$ and $Z_1 = 1$). The same considerations are extended to each set of two subsequent zones by replacing 1 with index i and 2 with $i + 1$.

[27] The previous Test Problem 1 can be solved successfully with the above approach, giving the correct solution where the initial reservoir level of 1662 m remains unchanged. However, the problem becomes more difficult to solve, as each binary variable adds two more rows to the constraint matrix as per expression (9), and each control structure adds one more row to the solution matrix required to incorporate expression (8). Worst of all, large problems will take considerably more computational effort to solve, extending execution times by a factor of 100 or more compared with simulations that do not require the use of mixed integer LP formulation.

[28] Does the above scheme always work with adequate accuracy? Surprisingly, the answer is no. Consider, for example, a Modified Test Problem 1 obtained by switching the priorities between the municipal demand and irrigation (i.e., giving irrigation demand a higher priority than municipal). The best solution is now the one already given for the original Test Problem 1 using the NFA solver, which remains unchanged if the priorities between municipal demand and irrigation are swapped. Paradoxically, the NFA algorithm works for this priority system since the iterative integration of outflow capacity for the municipal supply can properly assess the zero outflow capacity for the portion of the time step spent below the invert of the outlet structure. The full MIP formulation cannot do this and it consequently breaks down, as demonstrated below. Expression (8) imposes a limit on reservoir outflows with accuracy which depends on the following: (1) Starting and the ending reservoir levels for a time interval: When the starting level is above the invert of the outlet structure and the ending level is below it (or vice versa) for one time step, significant inaccuracy may emerge; and (2) reservoir zones should be sized to approximately equal volumes (while providing reasonable linear segmentation of the outflow versus elevation curve), such that when reservoir elevation crosses several zones within a time step, the integrated average is approximately equal to the arithmetic average. Since the model works with averaged flows per time step, the integrated average approaches arithmetic average only if equal amount of time was spent in each of the zones that were emptied (or filled) within a given time step. This condition is usually not met 100% all the time, since the starting and the ending elevations may not be exactly at a boundary of one or two zones, but it may help reduce possible inaccuracies associated with an elevation drop (or rise) over two or more zones.

[29] Failure of MIP to deliver a reasonably accurate solution to modified Test Problem 1 with swapped priorities between irrigation and municipal water requirements is directly linked to condition 1 above. It is a drastic example of failure and as such deserves to be dealt with in more detail. The principal difficulty with the outflow constraint represented by expression (8) is that it assumes that outflow is happening over the entire time step t. However, if the ending (or starting) elevation is below the invert of the outlet structure, only a portion of the time step will be spent above the invert, which means that the outflow will not be possible during the entire time step. This is what constraint (8) fails to address.

[30] The weekly MIP solution for Modified Test Problem 1, which differs from the original Test Problem 1 only by swapped priorities between the municipal demand and irrigation, delivered 12 m^3/s to irrigation and 1.625 m^3/s to the municipal demand. To demonstrate failure, Table 4 shows the solutions of modified Test Problem 1 using a series of seven sequential daily time step solutions. To make the daily time step simulation transparent to weekly time step, inflow was kept constant at 10 m^3/s for each day of the week, and so were the irrigation and municipal water requirements at 12 m^3/s and 3.25 m^3/s, respectively, while

Table 4. Modified Test Problem 1 With Daily Solutions

Time, days	Reservoir Elevation, m	Irrigation Supply, m^3/s	Instantaneous Outflow Capacity, m^3/s	Average Daily Outflow,[a] m^3/s
0	1662.00	12	3.250	
1	1661.20	12	2.692	2.971
2	1660.57	12	1.592	2.142
3	1660.10	12	0.615	1.104
4	1659.70	12	0.090	0.353
5	1659.32	12	0.000	0.045
6	1658.93	12	0.000	0.000
7	1658.49	12	0.000	0.000

[a]Seven-day average is 0.945.

the ending reservoir elevation for one day was used as the starting elevation for the following day. Note that the reservoir elevation drops each day, such that on day 4 it shows a very small outflow of 0.045 m^3/s, and for days 5, 6, and 7 there is no municipal supply at all since the storage level has dropped below the invert of the outlet structure. If the proposed linearization scheme is to work well, then the weekly time step simulation should deliver weekly flows equal to the average of the daily flows obtained for the first seven days, i.e., the municipal demand should receive 0.945 m^3/s, which is the daily flow average shown in Table 3a and which is close to the correct solution of 0.969 m^3/s (the best solution of modified Test Problem 1 is in fact the same as the solution obtained from the NFA-based model for the original Test Problem 1). However, an attempt to produce a weekly simulation with the same reservoir inflow and the same downstream water requirements allocated 12 m^3/s to irrigation and 1.625 m^3/s to the municipality, almost double than the average of the daily flow solutions of 0.945 m^3/s. This is because integration of the outlet structure capacity by expression (8) could not account for any other time step length than the assumed length of seven days. Yet the daily flow solution shows that the integrated average should be calculated only over the first 4 days. In other words, the length of the time step for the constraint (8) should have been 4 days, not 7. The difficulty is that in cases such as this one the "suitable" time step length is not known beforehand to allow proper setting of constraint (8).

[31] One possible remedy to this problem is to introduce a reduction factor f that represents the fraction of the time step for which the elevation is above the invert of the outlet structure. This factor should multiply the right-hand side of expression (8). When f is equal to 1 the reservoir level does not cross the invert during a time step; otherwise f is set to the value given by expression (10):

$$f = \frac{V_s - V_{inv}}{V_s - V_{end}}. \tag{10}$$

[32] In expression (10) Vs represents the starting storage at the beginning of the simulated time step, $Vinv$ corresponds to the storage at the invert of the outlet structure, and $Vend$ represents the storage at the end of the simulated time step. Expression (10) de facto defines a ratio between the storage change above the invert of the outlet structure and the total storage change for a time step. This ratio represents the fraction of the time step spent above the invert of the outlet structure. For example, modified Test Problem 1 would yield the f factor of 0.567 calculated using volumes in 1000 m^3 as

$$f = \frac{3400.83 - 2412.63}{3400.83 - 1658.41}. \tag{11}$$

This would reduce the outflow capacity to 0.921 m^3/s (=0.567 × 1.625), a much closer value to the daily integrated solution of 0.945 m^3/s in Table 3a than the value of 1.625 m^3/s obtained without using the adjustment factor f.

[33] In a single time step solution mode, Vs and $Vinv$ are fixed; however, $Vend$ is a decision variable. Therefore introduction of the term f as a multiplier on the right-hand side of expression (8) generates a mixed integer nonlinear constraint, and takes the entire problem out of the scope of the classical linear programming. The above adjustment factor can also be incorporated in multiple time step optimization by utilizing the value of the binary variable Z_i of the zone immediately above the invert of the outlet structure. For reasons of simplicity and without loss of generalization, assume that the starting and ending elevations are within the first zone above or below the invert. Expression (10) can then be rewritten in the following form:

$$f = \frac{V_s - V_{inv}}{V_s - V_{inv} + (V_{inv} - V_{end})(Z_i^{t-1} - Z_i^t)^2}. \tag{12}$$

[34] Variables Z_i^{t-1} and Z_i^t in expression (12) represent the value of the binary variable associated with the zone immediately above the invert of the outlet structure at two subsequent time steps: $(t - 1)$ and t, respectively. In any time step that ends with full or partial storage in this zone, the value of Z_i is set to 1; otherwise, if the zone is empty at the end of a time step, Z_i is set to 0. The settings are ensured by satisfying constraints (9). When reservoir storage starts and ends above the invert within a given time step, the difference between the two binary variables in expression (12) is zero, and the value of f is reduced to 1. Conversely, when the storage level crosses the invert during the time step (in either direction), the squared difference of the two binary variables is 1, thus reducing expression (12) to expression (10). It should be noted that in multiple time step optimization, both V_s and V_{end} in expression (12) are variables. Hence, in multiple time step optimization, the reservoir outflow constraints would contain products of quadratic and binary terms, which makes them unsuitable for LP implementation. The general form for defining upper bound on reservoir outflow channel can thus be written as

$$Qt(o) \le f \sum_{i=1}^{n} \frac{1}{Si} \cdot \frac{1}{2} \left(\frac{Vs(i)}{t} + \frac{Ve(i)}{t} \right), \tag{13}$$

where factor f is defined by (12) while the rest of the above expression conforms to (8). Index i in expression (13) is the counter for storage zones from bottom zone to top zone while t is the length of the simulated time step. In conclusion, calculation of the right-hand side of expression (13) provides the upper limit of reservoir outflow over a time step t. When the starting and ending storage volumes are both above the invert of the outlet structure, factor f in expression (13) takes the value of 1, and expression (13) is

Table 5a. Cylindrical Storage and Linear Outflow Curve

Volume, 1000 m^3	Elevation, m	Outflow, m^3/s	Elevation, m
0.000	1653.54	0.000	1660.00
3936.90	1663.00	4.364	1663.00

the same as expression (8). If, however, the reservoir level crosses the invert of the outlet structure during the calculation time step *t*, factor *f* provides an estimate of the fraction of the time step *t*, which is spent above the invert of the outlet structure and reduces the outflow capacity accordingly. Introduction of factor *f* brings in higher accuracy, but at the expense of requiring application of nonlinear solution techniques. However, introduction of factor *f* allows the users to continue to use weekly time steps, since the use of shorter (e.g., daily) time steps lowers execution efficiency and introduces other difficulties discussed later in this paper.

[35] To dispel any notion that this problem has anything to do with the number of points used in the outflow versus elevation curve or in the volume versus elevation curve, simpler variants of both the original problem 1 and modified problem 1 are solved assuming a cylindrical shape of storage, where the rate of elevation drop is constant for a fixed outflow rate, and a linear outflow versus storage relationship is assumed. Although in reality outflow is not a linear function of storage, this can be assumed for demonstration purposes. Benefits of such an assumption are that only two points on the curve are sufficient to define it, and the effects of the curvature are eliminated. The starting storage level for both problems is 1662 m, and the assumed linear relationships for volume versus elevation and outflow versus elevation are given in Table 5a, while Table 5b provides the results. As earlier, the NFA solution allocates to the municipal supply 0.733 m^3/s (this can be compared with 0.969 m^3/s obtained earlier for the original problem) while the best solution is the same as before (3.25 m^3/s). The principal reason for failure is the same; that is, the NFA solution procedure is unable to evaluate the effect that the bottom outlet releases for irrigation have on municipal supply. Modified test problem 1 also shows similar failure, with a deceiving value of the objective function which is lower than what it should be (22.1 instead of 28.6). The reason for this is because municipal supply gets 1.455 m^3/s while the maximum achievable is only 0.733 m^3/s for the given starting and ending storage levels, as seen in the correct solution column. Again, the failure here is inherent in the linearization scheme of the outlet curve, which is based on the assumption that average outflow from the reservoir estimated between the starting and the ending elevations will take place over the entire time step, while in reality it can only take place for a portion of the time step when elevation is above the invert of the outlet structure.

[36] One approach to resolving this issue could be to shorten the time step *t*. If the calculation was conducted with a series of seven daily time steps, the results would be more accurate, as attested by the average of daily reservoir outflows in Table 4. Again, the problem would still persist for one of the seven days when the invert is crossed, but the total volumes in error would be reduced due to shorter time step length of 1 day. However, the use of daily time step poses a number of problems. To begin with, the basic assumption in LP-based models is that water is available from any storage to any demand in the system within the calculation time step *t*. Assuming an average water velocity of 0.5 m/s, this would restrict the length of the total river system to about 43 km downstream of any supply reservoir. If larger systems with multiple reservoirs are to be analyzed, the model must include ability to account for channel storage change and travel time in the system. The most suitable way to do this within the mathematical programming framework is to introduce hydrologic routing equations as additional constraints in the model. As section 4 will show, this is fraught with dangers and difficulties that have yet to be addressed successfully.

4. Inclusion of Hydrologic Channel Routing Into an LP Formulation

[37] There are many advantages of introducing shorter time steps in basin allocation models. Simulations with monthly time steps may give overly optimistic results due to underestimating the amount of reservoir spills which happen during short runoff events that are not seen in the monthly data. Alberta Environment, a provincial government water management agency in western Canada, which has maintained and used the WRMM model since the early 1980s, has long abandoned the use of monthly time steps, since it was established that reservoir spills obtained from monthly simulations were on average close to 30% lower than in weekly simulations for the same scenarios. This should not come as a surprise, since the use of average monthly flows removes the peaks and provides "flat" reservoir inflows that are much easier to manage by the

Table 5b. Model Outputs for Cylindrical Storage and Linear Outflow Curve

	Test Problem 1			Modified Test Problem 1		
Component	NFA Solution	Correct Solution	Penalty per Unit of Deficit Flow	MIP Solution	Correct Solution	Penalty per Unit of Deficit Flow
Reservoir elevation, m	1658.02	1662.47		1656.98	1658.02	
Irrigation supply, m^3/s	12.000	6.428	10.0	12.000	12.000	100.0
Municipal supply, m^3/s	0.733	3.250	100.0	1.455	0.733	10.0
Storage deficit, m^3/s	3.421	0.366	1.0	4.143	3.421	1.0
Total penalty	1261.9	56.1		22.1	28.6	

model. A sizeable runoff event that lasted for a 1 or 2 weeks is typically lost when flows are averaged over a month. If such an event was modeled with a full starting storage with a weekly time step, a significant portion of the runoff would spill. Such magnitude of spills is typically not seen in simulations with monthly time steps.

[38] A move to shorter (e.g., daily) time steps is therefore beneficial. However, the total travel time of water from the most upstream storage to the most downstream users in medium and larger river basins is well beyond 1 day, and typically larger than a week. With an average water velocity of 0.5 m/s, any river exceeding the length of 300 km may have travel time longer than 1 week, especially during the low-flow season. It is therefore not surprising that many model vendors have added channel routing capabilities to their models, typically by using the Muskingum or Muskingum-Cunge methods since they are given as a linear form of channel inflows and outflow. Test Problem 2 and the discussion that follows will argue the following two points: (1) Hydrologic channel routing cannot be included in the LP-based models for a single time step optimization without violating the assumption of "demand driven reservoir releases," which is the basic premise on which these models were built and on which they operate on a steady-state basis; and (2) inclusion of channel routing in multiple time step optimization framework may often require nonlinear representation of routing coefficients, since routing coefficients should be updated when channel flows change from low- to high-flow seasons. This in turn may require a shift from LP framework in search for other suitable solution techniques.

[39] The following is a review of the routing capabilities of some of the models that are currently in use. Some models such as MODSIM or REALM offer Muskingum routing capabilities while they can only handle single time step optimization, as detailed in their online user manuals. They are therefore likely to run into the difficulties demonstrated by Test Problem 2 presented below. Models such as RIVERWARE and ARSP offer channel routing only as a refinement of their steady state simulation based on a daily time step [*Boroughs and Zagona*, 2002]. *Braga and Barbosa* [2001] report on inclusion of channel routing into multiple time step optimization using an advanced Network Simplex Solver [*Ahuja et al.*, 1993] that can handle non-network side constraints required for inclusion of channel routing, but they omit discussing the impact of assuming a fixed value for travel time constant K in Muskingum routing equation, which can cause significant inaccuracies as initially reported by *Ponce and Yevjevich* [1978] and confirmed subsequently by many other researchers. Calibration of the Muskingum routing coefficients in their work was demonstrated by fitting a single event hydrograph.

[40] One of the most versatile and flexible models presently available is OASIS. This model has the capability to include Muskingum channel routing in either single or multiple time step optimization framework. However, routing is rarely used in MTO, as acknowledged by the OASIS model vendors from Hydrologic, Inc. Instead, single daily time step simulation within an LP framework and with channel routing is used on most studies [*Susquehanna River Basin Commission*, 2006], which makes the modeling vulnerable to the difficulty demonstrated by Problem 2. The vendors of OASIS recognized the difficulty presented in Test Problem 2 in their ongoing work (personal communication, 2007) and pointed out that a perfect solution for this has yet to be found. Consequently, their modeling efforts focused on the development of heuristics rules and user-defined formulas for reservoir releases, all of which tend to diminish the role of the LP solution engine that is supposed to drive reservoir releases. Similar solution strategy was resorted to recently by Alberta Environment, while attempting to run 6-hourly time steps with channel routing within an LP framework. The routing scheme could only work with a fixed set of prescribed user-defined reservoir releases or fixed-target reservoir levels that had to be enforced. Also, experience from Alberta Environment pointed out that successful seasonal channel routing requires dynamic updates of the channel routing coefficients for seasonal flow changes. Without dynamic updates, hydrologic routing could not work for a range of channel flows that start at 20 m^3/s and reach over 500 m^3/s within a 4-month period. Consequently, this issue deserves attention, given the current widespread practice of using channel routing within a single time step optimization framework. The purpose of Test Problem 2 is to alert the user community of bad practice of modeling channel routing within LP framework with a single time step optimization. As demonstrated in the following, MTO requires automatic updates of the routing coefficients which can no longer be assumed constant when the flow change is significant. This violates the assumption of constant routing coefficients that is the basis for LP formulation.

[41] In a daily time step simulation mode, release from storage at time t is made to meet demands that are lagged several days later, due to the size of the basin being modeled and the associated travel times between the storage reservoir and demand nodes. Hence a combination of MTO and adequate channel routing is required to account for flow attenuation. Without MTO, the LP solver would force higher outflows from the reservoir than necessary in order to reduce the travel time and attempt to meet a downstream demand within a short routing time step, when in fact this demand should have been met by storage releases made in one or more of the preceding days. It should also be noted that to run a medium-sized network on a daily basis with 500 arcs in MTO over a 70-year period would involve 500 × 365 × 70 = 12.775 million variables! Problems with lengthy computer execution times, truncation errors, troubleshooting, and/or debugging of infeasible solutions probably explain why using MTO with routing is not the norm in practice.

[42] Channel routing methods suitable for LP implementation include Muskingum, Muskingum-Cunge, or the Williams routing equation built into the SSARR model. While they differ in the way the routing coefficients are calculated and updated, their suitability for LP implementation is due to explicit formulation of routed channel flows in a linear form:

$$Q_{j+1}^{t+1} = C1Q_j^t + C2Q_j^{t+1} + C3Q_{j+1}^t. \tag{14}$$

[43] Indices t and $t + 1$ in expression (14) represent time, while indices j and $j + 1$ represent upstream and downstream end of a channel, respectively. To include the channel routing formulation given by expression (14) above in the LP matrix of constraints, the users have to define the

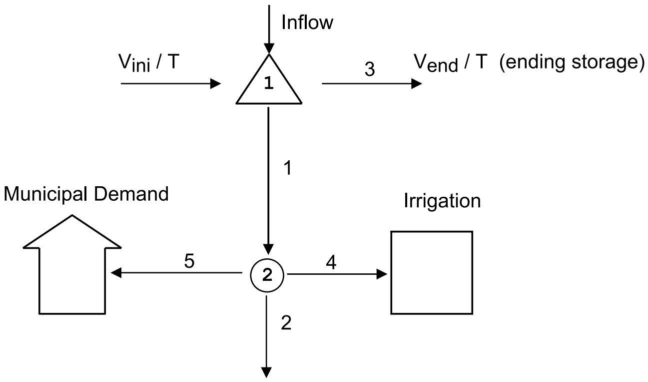

Figure 3. Modeling schematic for Test Problem 2.

reference flow rate for which the routing coefficients C_i are determined. However, the use of a single reference flow value for defining coefficients C_i causes inaccuracies, as reported by *Ponce and Yevjevich* [1978]. As a result, the current state of the art hydrologic models such as HEC-HMS [*Hydrologic Engineering Center*, 2006] and HSPF [*U.S. Environmental Protection Agency*, 2006] all utilize updated values of coefficients C_i based on the flows from previous time steps, or by using the average of the three flow rates on the right-hand side of expression (14). An attempt to include dynamic updates of the values of coefficients C_i in the LP framework for multiple time step optimization would introduce nonlinear constraints in the solution matrix since coefficients C_i would automatically take the form of $C_i(Q)$. This would constitute departure from LP. To the best of the author's knowledge, there has been no reported attempt so far on the efforts to include updated values of C_i as nonlinear constraints in the multiple time step optimization framework. Yet the errors associated with relying on the fixed reference flows and the use of constant coefficients C_i have been well documented. They can be substantial, especially for long-term simulations where channel flows vary from dry to wet seasons by an order of magnitude.

5. Test Problem 2

[44] Test problem 2 depicted in Figure 3 consists of a simple system with one reservoir, two channels, and two downstream withdrawals. Only the upstream channel 1 located between the reservoir and the withdrawals is modeled using hydrologic channel routing. Channel 2 is needed to handle possible system spills which are sometimes inevitable. The storage versus elevation curve for reservoir 1 is the same as in the previous example (Table 1), and the initial elevation is set to 1662.92 m. Two 6-hourly time steps are solved, the first one being a steady state solution that provides the initial channel flows for the routing equation (14). Reservoir inflow is set to 2 m^3/s, and municipal demands (link 5) are set to 1 m^3/s for both time steps, respectively, while irrigation demand (link 4) is set to 1 m^3/s for the first and 2 m^3/s is set for the second time step. Assume that the routing coefficients C_i in this example were estimated as $C_1 = C_2 = 0.1523$ and $C_3 = 0.6954$. These estimates were obtained using the linear variant of the Williams equation employed by the SSARR routing model, and the actual details related to how they were obtained are not essential for this demonstration. It should be noted that the sum of all three routing coefficients is 1, which is a standard expectation for a linear routing model. Assume that the model solves the maximum flow problem, i.e., it maximizes the sum product of flows times priorities, where the priority vector has the weight factors of 1, 10, and 1000 for a unit of flow assigned to storage, irrigation, and municipal demand, respectively, while the channel flow weights are set to zero. There are no outlet capacity constraints assumed in this example, and hence the outflow limits are equal to the channel flow limits which are set sufficiently high at 50 m^3/s. The initial steady state solution is easy to see: Inflow of 2 m^3/s is routed through the reservoir into channel 1 and it is split at the end of channel 1 evenly to irrigation (channel 4) and the municipal demand, since they both require 1 m^3/s in the first 6 hourly time step. This steady state solution provides the flow of 2 m^3/s at the beginning and at the end of channel 1. Hence the values of Q_j^t and Q_{j+1}^t in equation (14) are initially set to 2 m^3/s. Together with the routing coefficients already defined, equation (14) becomes

$$Q_2 = 0.1523 \cdot 2 + 0.1523 \cdot X_1 + 0.6954 \cdot 2. \tag{15}$$

[45] The storage balance for reservoir 1 equates the sum of the initial storage $Vini/T$ plus the inflow of 2 m^3/s to the sum of the outflow (variable X_1) and the ending storage (variable X_3 also expressed in the units of flow). The initial volume that corresponds to the elevation of 1662.92 m is 3,900,000 m^3 as per volume versus elevation curve in Table 1. Hence the term $Vini/T$ is evaluated as 3,900,000/(6 × 3600) = 180.5556 m^3/s for the 6-hourly time step. Using the weights listed above and the mass balance equation as constraints, the LP can be formulated as in term so flow maximization as

$$\text{Maximize } 1 \cdot X_3 + 10 \cdot X_4 + 1000 \cdot X_5 \tag{16}$$

subject to

$$X_1 + X_3 = 180.5556 + 2 \tag{17}$$

$$Q_2 - X_4 - X_5 - X_2 = 0. \tag{18}$$

[46] After substituting term Q_2 by the right-hand side of equation (14) and moving the constant terms on the right-hand side, equation (18) is transformed to

$$0.1523 \cdot X_1 - X_4 - X_5 - X_2 = -1.6954 \tag{19}$$

[47] The upper bounds on variables X_1, X_2, X_3, X_4, and X_5 are 50, 50, 250, 2, and 1, respectively. For this formulation, LP solver gives the following solution: X_1 = 8.5657; X_2 = 0; X_3 = 173.99; X_4 = 2; and X_5 = 1. All water demands are met, which should come as no surprise given that there is an upstream reservoir. However, to meet the increase in demand on two subsequent time steps from 1 to 2 m^3/s for the irrigation block, the model had to release a much higher flow (8.5657 m^3/s) from storage! This is a consequence of the fact that with flows around 3 m^3/s it takes around 17 hours for water to reach the irrigation block from the reservoir. However, the model finds the

reservoir outflow that will, after routing, provide an increase of 1 m^3/s at the downstream end of channel 1 within the 6-hourly time step. Hence the model decides to "flood" the downstream channel to shorten the travel time such that the flow increase of 1 m^3/s will be available within the 6-hour period. This is not the way the reservoir would actually be operated. Rather, the operator would make storage releases in advance taking into account the travel time to the destination points. Hence, to properly account for channel routing in this example, multiple 6-hour time step optimization should be conducted for three time steps simultaneously. In other words, two more 6-hourly time steps should be added to the system such that they precede the above time step. The model should then find moderate reservoir releases that after being routed through channel 1 over the next two 6-hourly time steps, provide sufficient increase of flow to meet the demand in the third time step, without the need to cause any unnecessary spills. The demands at the end of channel 1 should not be modeled at all in the first two time steps, since storage releases are not able to reach the downstream end of channel 1. Elaborate details on how multiple time step optimization should be setup to achieve this are beyond the scope of this paper, and they have already been covered elsewhere in the literature [*Braga and Barbosa*, 2001]. The main purpose of this test problem was to demonstrate that channel routing cannot work within the LP framework using a single time step solution unless the system is so small that the entire travel time is shorter than the length of the time step required for routing, which is typically between 6 and 24 hours (the shorter the time step, the more accurate the routing transformation). This severely restricts the size of the systems that can be modeled as LP with channel routing. Multiple time step optimization would only solve the above riddle for fixed values of routing coefficients that must be known in advance for each time step. Yet, for larger flow variations between wet and dry seasons, the routing coefficients must be updated as a function of channel flows, which brings the entire LP framework in question even within the multiple time step optimization framework.

[48] There are two possible remedies to this situation. One is to consider nonlinear programming solution techniques that would allow automatic dynamic updating of routing coefficients during the search process. Once such attempt has been made [*Ilich and Simonovic*, 2001], with a search engine based on a hybrid between network flow and genetic algorithms, resulting in an efficient search conducted directly on the network and being restricted only to the feasible region. This solver has been applied [*Ilich et al.*, 2000a; *Ilich*, 2001], and its current improvements are related mainly to the development of ability to conduct automatic self-calibration of its search parameters. However, as any other nonlinear search engine, this one is also unable to guarantee finding the global optimum, in spite of the fact that its search proceeds in parallel from all corners of the feasible region, which gives it a good statistical chance of converging to a global optimum. The principal advantage of LP and its variants such as quadratic programming is a guarantee that global optimum will be found.

[49] The other possible remedy is to try to develop a complete linearization of the channel routing equation. This would involve splitting flow in the channel into a number of segments (i.e., treating each channel as a set of parallel subchannels) and devising an algorithm that would define fixed routing coefficients for each parallel subchannel. The sum of routed flow in all parallel subchannels would then have to equal the routed flow in a physical channel. If this could be done, linear programming formulation would still be possible for multiple time step optimization. This is a hopeful approach since it is known that routing coefficients change slowly with the change of channel flow, and they can have representative values for a limited flow range within a channel.

6. Other Nonlinear Components

[50] Detailed treatment of the hydropower component has not been included in this paper, although most models referenced in this paper claim to have some capability to model hydropower generation, a claim that is highlighted by the vendors of RIVERWARE or OASIS. It is sufficient to notice that hydropower components introduce nonlinear terms both in the constraints and in the objective function. Most efforts to linearize them so far have resorted to iterative calls of LP solvers, without explicit critical reviews of the possible lack of accuracy that iterations may cause. Hydropower generation is described by the following equation:

$$P = QH\eta(Q, H), \tag{20}$$

where P is the average power over a time step, Q is the average flow through the turbines, H is the average net head over a time step, and η is the efficiency factor, which depends on both flow and net head. In fact, there are two efficiency functions, one associated with the efficiency of the turbine and the other one associated with efficiency of the generator. Those two efficiency functions usually do not peak for the same choice of Q and H. Of the other two terms, net head H is often difficult to evaluate dynamically during the solution process. It can be approximated as the average upstream storage elevation over a time step reduced by hydraulic losses through the diversion tunnel, and then also reduced by the average water elevation immediately downstream of the hydropower plant over the calculation time step. The water surface elevation downstream of the plant may be governed by a flow rating curve, or by the pool elevation of a downstream reservoir if there are two sequential reservoirs forming a cascade. In either case, both downstream and upstream elevations are a function of the reservoir mass balance, where the flow variable Q is an important constituent. Hence the net head is a function of reservoir inflows, outflows, starting elevation, the shape of the storage capacity curve for one or more reservoirs, and possibly hydraulic losses through the diversion tunnel or the flow rating curve downstream of the plant. Most of these constituents are interconnected and change dynamically during the search procedure. Last but not least, maximum flow through the turbine Q is limited by two nonlinear functions of the net head, one related to the turbine capacity and the other to the capacity of the generator.

[51] Most of the referenced models claim the ability to optimize hydropower using LP with iterations. Possible

failures of LP solvers to deliver high-quality solutions in the field of hydropower modeling and the adequate remedies for such failures have yet to be fully addressed in a comprehensive manner.

7. Conclusions and Recommendations

[52] This paper examines the shortcomings of LP-based models in river basin management and planning to address hydraulic and hydrologic constraints found in river basin networks. In particular, the paper focuses on the limitations of LP to optimize river basin allocation problems that involve multiple reservoir outflows and channel routing, and claims that neither of these limitations can be overcome in a satisfactory manner within the LP framework. The limitations are demonstrated using numerical examples with sufficient input data to be independently verified. A recommendation is made to use LP techniques with caution and be aware of the possible problems that may arise even when full LP application models are considered. A suggestion is also made to encourage researchers and practitioners in this field to define a set of desired technical specifications for river basin allocation models, and work jointly to define a standard set of test problems for model verification that could be used for models that utilize either LP or various non-LP solution strategies. In closing, future research in model development should attempt to address issues raised in this paper.

Appendix A

[53] Here we demonstrate the need to use binary variables for modeling of reservoir operation within the LP framework using linearized outlet structure constraints. The test problem presented here is first solved without binary variables, and the solution is analyzed. The problem involves a single reservoir with two outlets, with low starting elevation such that reservoir refill is required to reach the desired outflow through the outlet structure. The input data are the same as in Test Problem 1, except that municipal demand is set to 4 m^3/s and irrigation requirement is set to 8 m^3/s, and the starting storage elevation is set to 1656 m. Reservoir storage is divided into four zones, with penalties of 1, 2, 3, and 4 assigned to the zones in the top to bottom order. These penalties should ensure that the bottom zone is filled first, followed by the zone above it, and so on. It also ensures that the top zone is emptied first at times of reservoir release, followed by the zone below it, etc. This penalty scheme works well when there are no side constraints that interfere with it, as is the case in this example. Figure A1 shows the arc representation of all components. There are six variables in the problem. Variable X1 represents supply to irrigation, variable X2 is municipal supply (which is only possible through an outlet structure with outlet versus elevation curve given in Table 1), and the four reservoir zones X3 through X6 represent reservoir storage at the end of calculation time step. Table A1 provides the upper bounds in units of flow and penalty factors chosen for this test problem. The upper bounds for storage zones were read for the four elevations given in the outflow versus elevation curve (1660, 1661, 1662, and 1664 m), converted to incremental storage (or storage of a single zone), and divided by the length of the week in seconds to obtain storage zones in the units of flow. For example, for the invert elevation 1660 m, the corresponding storage is 2,412,630 m^3. When divided by the length of the week in seconds (86,400 $\times$ 7), it gives 3.989 m^3/s, which is the upper bound of the bottom zone 4. For zones 3, 2, and 1, it is necessary to first calculate incremental storage. For example, for zone 3 the storage is

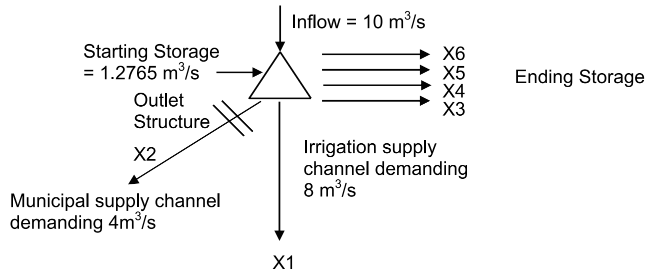

Figure A1. Layout of system variables.

$$2892740 - 2412630 = 480110 \text{ m}^3. \quad (A1)$$

[54] When divided by the length of the week, this gives 0.79383 m^3/s, as seen in Table A1 for variable X4. To evaluate slope S for the first storage segment, divide incremental storage of 0.70383 m^3/s by the incremental change in outflow capacity, which is (1.85 − 0) for the first segment of the outlet curve given in Table 1. Consequently, S_1 = 0.79383/1.85 = 0.38045, and the term $1/(2S_1)$ for the first zone above the invert becomes 1.1652. Similar terms for the other two zones above invert are derived in the same way to give 0.83324 and 0.62841. We now focus on expression (8) in the main text. In it, the outflow capacity on the right-hand side is calculated as the average of the starting and ending outflow capacity for a given time step. Since the starting storage elevation is 1656 m, which is below the invert, the initial outflow capacity is zero. In other words, term $Vs(i)/t$ for all zones i above the invert is zero, since none of the zones above the invert have any storage at the beginning of the time step. The only other term on the right-hand side of expression (8) is $Ve(i)/t$, which is represented by variables X4 through X6, since X3 represents storage zone below the invert that is only accessible through the bottom outlet for irrigation supply. Move the remaining right-hand side terms of expression (8) to the left side:

$$Qt(o) - \sum_{i=1}^{n} \frac{1}{Si} \cdot \frac{1}{2} \left(\frac{Ve(i)}{t} \right) \leq 0. \quad (A2)$$

[55] We can now formulate the allocation program as an LP using flow maximization objective:

$$\text{Maximize} \quad 10X_1 + 100X_2 + 4X_3 + 3X_4 + 2X_5 + 1X_6 \quad (A3)$$

subject to

$$X_1 + X_2 + X_3 + X_4 + X_5 + X_6 = 10 + 1.2765 \quad (A4)$$

$$X_2 - 1.16523X_4 - 0.83324X_5 - 0.62841X_6 \leq 0 \quad (A5)$$

$$0 \leq X_i \leq Ui, \quad (A6)$$

where Ui are the upper bounds on Xi defined in Table A1, while 1.2765 m^3/s in the mass balance equation (A4)

Table A1. Upper Bounds and Penalties for the Decision Variables

Variable	Upper Bound	Penalty Factor
X1	8.0000	10
X2	4.0000	100
X3	3.9891	4
X4	0.7938	3
X5	0.8401	2
X6	0.8864	1

represents the available storage of 772,030 m^3 expressed in the units of flow at the beginning of the weekly step. The above problem can be solved using a spreadsheet solver. The solution is X_1 = 6.5742; X_2 = 2.182; X_3 = 0; X_4 = 0.7938; X_5 = 0.8401; and X_6 = 0.8863. To maximize outflow to the municipal demand X_2, the model has moved all storage to the upper zones which are above the invert, and left the storage in the zone below the invert empty (X_3 = 0). It did not help that the bottom zone has the highest penalty of 4 compared with other storage zones. It can be emptied through the bottom outlet for irrigation supply, which has a demand on it with a penalty of 10. There is no question that this solution makes no physical sense, since storage must be filled from bottom to top. Binary variables are the only mechanism that can prevent this from happening. They make the problem much more difficult to solve, but they guarantee that situations such as this cannot happen. Four binary variables (X_7 through X_{10}) are introduced into this problem. Their penalty factors are all zero. However, they are linked to the storage zone bounds by additional seven constraints that are added to the problem:

$$X_3 - 3.98914\ X_7 \leq 0 \tag{A7}$$

$$X_4 - 0.79383\ X_8 \leq 0 \tag{A8}$$

$$X_5 - 0.84010\ X_9 \leq 0 \tag{A9}$$

$$X_6 - 0.88636\ X_{10} \leq 0 \tag{A10}$$

$$-X_3 + 3.98914\ X_8 \leq 0 \tag{A11}$$

$$-X_4 + 0.79383\ X_9 \leq 0 \tag{A12}$$

$$-X_5 + 0.84010\ X_{10} \leq 0. \tag{A13}$$

[56] With the above additional constraints, the model is solved for all 10 variables. The solution is X_1 = 2.58508; X_2 = 2.182; X_3 = 3.98914.; X_4 = 0.7938; X_5 = 0.8401; X_6 = 0.8863; X_7 = 1; X_8 = 1; X_9 = 1; and X_{10} = 1. It can be noted that there are no gaps in filling storage zones from bottom to top in this solution. Also, the model correctly decided to maximize storage over the calculation time step at the expense of reducing supply to irrigation, since that is the only way of maximizing municipal supply, which is constrained by the available storage and outlet structure flow limits. If this problem is solved using any NFA solver, the supply to irrigation would be much higher and the solution would be far from optimal, for the same reason already demonstrated earlier in Test Problem 1. The only difference between the original Test Problem 1 and this test problem is that in this case storage undergoes refill, as opposed to Test Problem 1 where storage was drawn down.

[57] **Acknowledgments.** The Water Resources Management Model used for the above test runs is a public domain modeling tool owned and maintained by Alberta Environment, a provincial government water management agency in Canada. The contribution of Leslie Stillwater of USBR, who volunteered to independently verify the NFA results of Test Problem 1 using the MODSIM model, is truly appreciated.

References

Ahuja, R. K., T. L. Magnanti, and J. B. Orlin (1993), *Network Flows*, Prentice-Hall, Upper Saddle River, N. J.

Andrews, E. S., F. I. Chung, and J. B. Orlin (1993), Multilayer, priority-based simulation of conjunctive facilities, *J. Water Resour. Plann. Manage.*, *118*(1), 32–53.

Barr, R. S., F. Glover, and D. Klingman (1974), An improved version of the out-of-kilter method and comparative study of computer codes, in *Mathematical Programming 7*, pp. 60–86, North-Holland, Amsterdam.

Bertsekas, D. P., and P. Tseng (1988), Relaxation methods for minimum cost ordinary and generalized network flow problems, *Oper. Res.*, *36*(1), 93–114.

Boroughs, C. B., and E. A. Zagona (2002), Daily flow routing with the Muskingum-Cunge method in the Pecos River RiverWare model, paper presented at the Second Federal Interagency Hydrologic Modeling Conference, Subcomm. on Hydrol., Las Vegas, Nev, 28 July to 1 Aug.

Braga, B., and P. F. S. Barbosa (2001), Multiobjective real-time reservoir operation with network flow algorithm, *J. Am. Water Resour. Assoc.*, *37*(4), 837–852.

Brendecke, C. M. (1989), Network models of water rights and system operations, *J. Water Resour. Plann. Manage.*, *115*(5), 684–696.

Chung, F. I., M. C. Archer, and J. J. DeVries (1989), Network flow algorithm applied to California aqueduct simulation, *J. Water Resour. Plann. Manage.*, *115*(2), 131–147.

Department of Sustainability and Environment (2006), Resource allocation model (REALM), Victoria, Australia. (Available at http://www.dse.vic.gov.au/DSE/wcmn202.nsf/)

Draper, A. J., A. Munévar, S. K. Arora, E. Reyes, N. L. Parker, F. I. Chung, and L. E. Peterson (2004), CalSim: Generalized model for reservoir system analyses, *J. Water Resour. Plann. Manage.*, *130*(6), 480–489.

Evenson, D. E., and J. C. Moseley (1970), Simulation/optimization techniques for multi-basin water resource planning, *Water Resour. Bull.*, *6*(5), 725–736.

Fulkerson, D. R. (1961), An out-of-kilter method for minimal cost flow problems, *SIAM J. Appl. Math.*, *9*(1), 18–27.

Hydrologic Engineering Center (2006), HEC-HMS user manual, U.S. Army Corps of Eng., Davis, Calif. (Available at http://www.hec.usace.army.mil/software/hec-hms/)

Ilich, N. (2001), The benefits of replacing LP solvers in basin allocation models with a generalized non-linear evolutionary network flow solver (SFEP), paper presented at the 2001 World Environmental and Water Resources Congress, Am. Soc. of Civ. Eng., Orlando, Fla., 20–24 May.

Ilich, N., and J. Despotovic (2007), A simple method for effective multi-site generation of stochastic hydrologic time series, *J. Stochast. Environ. Res. Risk Assess.*, *22*(2), 265–279, doi:10.1007/s00477-007-0113-6.

Ilich, N., and S. P. Simonovic (2001), An evolution program for non-linear transportation problems, *J. Heuristics*, *7*, 145–168.

Ilich, N., R. Drury, P. Godman, and S. P. Simonovic (2000a), Operation of Bighorn/Brazeau Hydro Power System of Transalta Utilities Corporation, paper presented at the XIII International Conference on Computational Methods in Water Resources, Univ. of Calgary, Calgary, Alberta, Canada, 25–29 June.

Ilich, N., S. P. Simonovic, and M. Amron (2000b), The benefits of computerized real-time river basin management in the Malahayu Reservoir System, *Can. J. Civ. Eng.*, *27*(1), 55–64.

Koutsoyiannis, D., and A. Economou (2003), Evaluation of the parameterization-simulation-optimization approach for the control of reservoir systems, *Water Resour. Res.*, *39*(6), 1170, doi:10.1029/2003WR002148.

Kuczera, G., and G. Diment (1988), General water supply system simulation model: WASP, *J. Water Resour. Plann. Manage.*, *114*(4), 365–382.

Labadie, J. W. (2004), Optimal operation of multireservoir systems: A state-of-the-art review, *J. Water Resour. Plann. Manage.*, *130*(2), 93–111.
Labadie, J. W., D. A. Bode, and A. M. Pineda (1986), Network model for decision-support in municipal raw water supply, *Water Resour. Bull.*, *22*(6), 927–940.
Nalbantis, I., and D. Koutsoyiannis (1997), A parametric rule for planning and management of multiple reservoir systems, *Water Resour. Res.*, *33*(9), 2165–2177.
Needham, J. T., D. W. Watkins Jr., and J. Lund (2000), Linear programming for flood control in the Iowa and Des Moines rivers, *J. Water Resour. Plann. Manage.*, *126*(3), 118–127.
Ponce, V. M., and V. Yevjevich (1978), Muskingum-Cunge method with variable parameters, *J. Hydraul. Div. Am. Soc. Civ. Eng.*, *104*(HY12), 1663–1667.
Randall, D., L. Cleland, C. S. Kuehne, G. W. Link, and D. P. Sheer (1997), Water supply planning simulation model using mixed-integer linear programming ''engine,'' *J. Water Resour. Plann. Manage.*, *123*(2), 116–124.
Sigvaldason, O. T. (1976), A simulation model for operating a multi purpose multireservoir system, *Water Resour. Res.*, *12*(2), 263–278.
Susquehanna River Basin Commission (SRBC) (2006), Conowingo Pond management plan, *Publ. 242*, Harrisburg, Pa. (Available at www.srbc.net)
U.S. Environmental Protection Agency (2006), Hydrologic Simulation Program—FORTRAN (HSPF) user manual, Washington, D. C. (Available at http://www.epa.gov/ceampubl/swater/hspf/)
Windsor, J. S. (1973), Optimization model for operation of flood control systems, *Water Resour. Res.*, *9*(5), 1219–1226.
Wurbs, R. (1993), Reservoir system simulation and optimization models, *J. Water Resour. Plann. Manage.*, *119*(4), 455–471.
Yeh, W. W.-G. (1985), Reservoir management and operations models: A state-of-the art review, *Water Resour. Res.*, *21*(12), 1797–1818.
Zagona, E. A., T. J. Fulp, R. Shane, T. Magee, and H. M. Goranflo (2001), RiverWare: A generalized tool for complex reservoir system modeling, *J. Am. Water Resour. Assoc.*, *37*(4), 913–929.

N. Ilich, Department of Civil Engineering, University of Calgary, 7128 5th Street NW, Calgary, Alberta, Canada T2K 1C8. (nilich@optimal-solutions-ltd.com)