

Last Resort Algorithms for Optimization of Water Resources Systems

Slobodan P. Simonovic
Natural Resources Institute &
Dept. of Civil and Geological Engineering
University of Manitoba

Floods, droughts, water scarcity and water contamination are some among many water problems that are present today and will be even more noticeable in the future. In the past, many traditional optimization algorithms have been used for management of complex water resources systems in order to provide an improved basis for decision making. Their application to water resources planning, management and operations has been established as one of the most important advances made in the field of water resources engineering¹⁹. However, the complexity of real water resources management problems today exceeds the capacity of traditional optimization algorithms. Two problems are selected to illustrate the point: (a) a real-time hydropower operation problem; and (b) a water resources network flow problem. In the first case, the search algorithm known as 'simulated annealing' has been discussed in the context of real-time hydropower optimization. In the second case, an original evolution algorithm, named a strongly feasible evolution program, is presented for non-linear optimization of water resources network flows. Both of them are named 'last resort algorithms' due to the fact that they represent the last attempts at solving very complex water resources management problems.

Real-time operation of multiple reservoir systems for hydropower generation

In general, linear (LP), non-linear programming (NLP) and dynamic programming (DP) models are used for solution of many problems in the field of water resources^{19,11,18}. In attempting to solve a multi-period and multiple reservoir operations for hydropower generation problem, we need to keep in mind the enormity of the problem formulation and the computational power required to solve the problem in real-time. Traditional optimization algorithms still

Des algorithmes de dernier recours pour l'optimisation de systèmes de ressources hydriques

Slobodan P. Simonovic
Natural Resources Institute &
Dept. of Civil and Geological Engineering
University of Manitoba

Inondation, sécheresse, pénurie et contamination ne sont que quelques-uns des nombreux problèmes associés à l'eau auxquels nous sommes confrontés aujourd'hui et qui n'iront qu'en s'amplifiant dans les années futures. Dans le passé, divers algorithmes d'optimisation classiques ont été utilisés afin d'améliorer le processus de prise de décision dans la gestion de systèmes complexes de ressources en eau. Leur utilisation pour la planification, la gestion et l'exploitation des ressources hydriques a été regardée comme l'un des avancements les plus importants dans le domaine du génie hydro-économique¹⁹. Cependant, la complexité des problèmes réels de gestion des ressources en eau excède aujourd'hui la capacité des algorithmes d'optimisation traditionnels. Nous avons choisi deux problèmes pour illustrer ce point : (a) un problème d'exploitation de l'énergie hydraulique en temps réel et (b) un problème de réseau de ressources en eau. Dans le premier cas, nous examinons l'algorithme de recherche dit du "recuit simulé" dans une perspective d'optimisation de l'énergie hydraulique en temps réel. Dans le second cas, nous présentons un algorithme évolutif original, appelé "programme évolutif fortement réalisable" (SFEP), pour l'optimisation non linéaire d'un réseau de ressources en eau. Ces deux algorithmes sont appelés "algorithmes de dernier recours" parce qu'ils constituent les dernières tentatives de résolution de problèmes très complexes de gestion des ressources hydriques.

L'exploitation en temps réel de systèmes à réservoirs multiples pour la production d'énergie hydroélectrique

En règle générale, on utilise des modèles de programmation linéaire, de programmation non linéaire et de programmation dynamique pour résoudre une grande partie des problèmes dans le secteur des ressources en eau^{19,11,18}. Quand on essaie de résoudre un problème lié à l'exploitation de systèmes à plusieurs réservoirs pour la production d'énergie hydroélectrique sur plusieurs périodes, il ne faut pas perdre de vue la grande complexité de l'énoncé du problème et la puissance de calcul

suffer from one of these limitations: (a) computational intractability; (b) requirement of calculation of derivatives; and (c) need for too many assumptions for the problem to fit in a standard form of the optimization technique (ranging from linearization of the objective function and constraints to incorporating problem specific information into the formulation).

In the case of real-time operation of multiple reservoir systems for hydropower generation, the computational time required to solve the problem is more than the actual time within which a decision is required for implementation. For example, an hourly scheduling problem for a weekly time horizon and four reservoirs with a total of 168 time intervals (hours) formulated as the mixed-integer non-linear program (MINLP) and solved using the GAMS optimization solver³ failed to produce the results after four hours of real-time computation on the powerful SUN workstation¹⁶. This can be explained by the limited capabilities of the optimization algorithm and the complexity associated with the problem formulation. The MINLP formulation suffers from the combinatorial explosion problem. On the other side, the use of the DP algorithm has limitations due to the curse of dimensionality. From the experimentation with both MINLP and DP formulations, it is concluded that an hourly scheduling problem for a time horizon of a week cannot be solved within the time limit of one hour.

As the last resort approach, the simple combination of simulation and search algorithms has been used in addressing the same scheduling problem. Search techniques range from simple hill climbing heuristics to complex tabu search methods. In many search algorithms, simulation becomes a perfect tool in obtaining the performance measure of each solution obtained through perturbation of decision variables. Simulated annealing is used as a search method guided by a stochastic criterion that helps in surpassing the local optimum. The technique derives its name from the 'annealing' process generally used in the glass making industry and metallurgical processes⁸. It has been used in solving large-scale combinatorial problems like travelling salesman or circuit wiring. The technique uses an imperfect analogy between the process of

nécessaire pour résoudre le problème en temps réel. Les algorithmes d'optimisation classiques présentent toujours l'une ou l'autre des restrictions suivantes : (a) intractabilité de calcul, (b) nécessité de calculer les dérivées et (c) nécessité d'énoncer un trop grand nombre d'hypothèses pour que l'on puisse adapter le problème à une forme courante de la technique d'optimisation (de la linéarisation de la fonction objectif et des contraintes jusqu'à l'incorporation de données spécifiques au problème dans l'énoncé).

En ce qui concerne l'exploitation en temps réel de systèmes à réservoirs multiples pour la production d'énergie hydroélectrique, le temps de calcul requis pour résoudre le problème est supérieur au temps réel durant lequel une décision doit être prise. Ainsi, un problème d'ordonnement horaire pour un horizon prévisionnel d'une semaine et quatre réservoirs totalisant 168 intervalles de temps (heures), énoncé sous la forme d'un programme non linéaire en nombres mixtes et résolu à l'aide du solveur d'optimisation GAMS³, n'a pas réussi à produire les résultats escomptés après quatre heures de calcul en temps réel sur le puissant poste de travail SUN¹⁶. Cet échec s'explique par les capacités limitées de l'algorithme d'optimisation et par la complexité associée à l'énoncé du problème. L'énoncé du programme non linéaire en nombres mixtes souffre du problème d'explosion combinatoire. Par ailleurs, on ne peut utiliser sans restrictions l'algorithme de programmation dynamique à cause des problèmes de dimensionnalité. À la lumière des essais réalisés avec des algorithmes de programmation non linéaire en nombres mixtes et de programmation dynamique, nous en sommes venus à la conclusion qu'il était impossible de résoudre un problème d'ordonnement horaire pour un horizon prévisionnel d'une semaine dans un délai d'une heure.

En dernier recours, nous avons donc utilisé la simple combinaison d'algorithmes de simulation et de recherche pour résoudre ce même problème d'ordonnement. Les techniques de recherche employées vont de simples heuristiques d'escalade à des méthodes plus complexes de recherche avec tabous. Dans bon nombre d'algorithmes de recherche, la simulation constitue un outil idéal qui fournit la mesure du rendement de chaque solution obtenue au moyen d'une perturbation des variables de décision. On utilise le recuit simulé comme méthode de recherche fondée sur un critère stochastique qui aide à dépasser l'optimum local. Cette technique tire son nom du procédé de "recuit" généralement employé dans l'industrie du verre et en métallurgie⁸. On l'utilise pour résoudre des problèmes

an imperfect analogy between the process of cooling in metals to a low energy state and optimization of complex functions. In annealing, a metal or alloy is slowly cooled at each intermediate temperature until some kind of equilibrium is achieved. Higher temperatures correspond to greater kinetic energy. The best stable structures are obtained at lower temperatures. However, the rapid cooling of metal can result in a brittle structure. Optimization can follow the logic of the annealing process, maximizing the strength, minimizing the brittleness, by generating a structure with least energy¹². The generalized simulated annealing algorithm contains the following steps: (i) selection of variables that influence the system the most; (ii) initialization of variables and random perturbation of their values; (iii) calculation of the performance measure (*energy*) using a simulation model; (iv) IF [$energy_{new} < energy_{old}$] THEN accept the move; (v) ELSE accept/reject based on a stochastic criterion; (vi) repeat steps (ii)-(v) until stopping criterion is met; and (vii) store the best solution obtained so far. Steps (i) to (iii) are problem specific, while better efficiency can be achieved by modifying steps (iv) to (vi). So, simulated annealing is a heuristic algorithm for obtaining good (not necessarily global) optimal solutions to complex optimization problems.

Problem presentation is quite important for the implementation of the simulated annealing algorithm. For the real-time operation of the multiple reservoir system for hydropower generation problem, release (or discharge) at each of the plants is used as the state (control) variable and the cost function of power generation is defined as the performance measure. Problem specific information related to efficiency-discharge curves is used in reducing the range of the values for the control variables. The complete model¹⁷ consists of two modules: (a) annealing algorithm; and (b) simulation model. Module (a) uses steps (i), (ii) and (iv) to (vii) of the generalized annealing algorithm, and step (iii) is executed using the simulation model. The following calculations are carried out by the simulation model:

1. calculation of storage states based on the release values for all the time periods;
2. calculation of average storage from initial

combinatoires à grande échelle comme le problème du voyageur de commerce ou celui du câblage de circuits. Cette technologie repose sur une analogie imparfaite entre le processus de refroidissement des métaux à un état de faible énergie et l'optimisation de fonctions complexes. Le recuit consiste à refroidir lentement le métal ou l'alliage à chaque température intermédiaire jusqu'à ce qu'un certain équilibre soit atteint. Les températures plus élevées correspondent à une énergie cinétique supérieure. C'est aux températures plus basses qu'on obtient les structures les plus stables. Toutefois, un refroidissement rapide du métal produit parfois une structure cassante. L'optimisation peut suivre la logique du procédé de recuit, c'est-à-dire maximiser la puissance et minimiser la fragilité en produisant une structure à énergie plus faible¹². L'algorithme généralisé du recuit simulé inclut les étapes suivantes : (i) sélectionner les variables qui influent le plus sur le système; (ii) initialiser les variables et réaliser une perturbation aléatoire de leurs valeurs; (iii) calculer la mesure du rendement (*énergie*) à l'aide d'un modèle de simulation; (iv) SI l'*énergie* diminue, ALORS accepter le mouvement; (v) SINON, l'accepter ou le rejeter en se basant sur un critère stochastique; (vi) répéter les étapes (ii)-(v) jusqu'à ce qu'on obtienne un critère d'arrêt et (vii) garder la meilleure solution obtenue jusque-là. Les étapes (i) à (iii) sont spécifiques au problème, alors qu'il est possible d'atteindre une plus grande efficacité en modifiant les étapes (iv) à (vi). Par conséquent, le recuit simulé est un algorithme heuristique qui permet d'obtenir de bonnes solutions optimales (pas nécessairement globales) pour les problèmes d'optimisation complexes.

La formulation du problème est importante pour la mise en oeuvre de l'algorithme du recuit simulé. Dans le cas du problème d'exploitation en temps réel d'un système à réservoirs multiples pour la production d'énergie hydroélectrique, le débit (ou décharge) à chacune des usines correspond à la variable d'état (de contrôle), et la fonction de coût de la production d'énergie est définie comme mesure du rendement. On utilise les données spécifiques au problème qui se rapportent aux courbes rendement-débit pour réduire l'ensemble de valeurs pour les variables de contrôle. Le modèle complet¹⁷ comporte deux modules : (a) l'algorithme du recuit et (b) le modèle de simulation. Le module (a) comprend les étapes (i), (ii) et (iv) à (vii) de l'algorithme généralisé du recuit, et l'étape (iii) est exécutée à l'aide du modèle de simulation. Le modèle de simulation effectue les calculs suivants :

1. calcul des états de retenue en fonction des valeurs de débit pour toutes les périodes;

- and final values;
3. selection of tailwater curves based on the discharge and the average forebay and the downstream forebay elevations;
 4. calculation of head required for hydropower generation using average forebay elevation and the tailwater elevation;
 5. check of all constraints (power production, maximum and minimum allowable storage, and other);
 6. evaluation of infeasibilities based on the constraint violations from step 5; and
 7. reporting of the solution.
2. calcul de la retenue moyenne à partir des valeurs initiales et finales;
 3. sélection des courbes d'eau d'aval en fonction du débit et de l'élévation moyenne des réservoirs et de l'élévation des réservoirs en aval;
 4. calcul de la hauteur de chute requise pour produire l'énergie hydroélectrique en fonction de l'élévation moyenne des réservoirs et du niveau aval;
 5. vérification de toutes les contraintes (production d'énergie, retenue maximale et minimale admissible et autres);
 6. évaluation des non-réalisabilités en fonction des violations des contraintes de l'étape 5 et
 7. présentation de la solution.

Assessment of the simulated annealing optimization is done using a four-reservoir system on the Winnipeg River. Hourly scheduling is considered based on the weekly targets. The objective is to minimize the generation cost for all time periods (168 hourly intervals) while meeting the system demand. The results obtained corroborate the fact that near optimum generating schedule can be obtained within a very short period of time (much smaller than one hour). Problem specific information can be quite useful in improving the search and further reducing computational time.

Non-linear optimization of water resources networks

The most common approach to optimization of water resources networks is based on using the cost-capacitated network representation of a river basin and by solving the corresponding linear minimum cost flow problem. Some of the models which utilized these concepts are SIMYLD⁴, ACRES¹⁵, MODSIM3¹⁰, and WASP⁹. Virtually all models still rely on linear programming (LP), mainly since the objective function related to water licensing priorities and other allocation objectives is linear. The only exception to this is the hydropower production. Typically, most of the above models use the Out-of-Kilter algorithm². The main deficiencies in using LP for network flow optimization are: (i) water allocation models based on LP solvers are unable to incorporate a non-linear change of flow along a river or canal reach without an iterative procedure; (ii) the Out-of-Kilter and

L'évaluation de l'optimisation par recuit simulé se fait à partir d'un système à quatre réservoirs sur la rivière Winnipeg. On étudie l'ordonnancement horaire en fonction des objectifs hebdomadaires. L'objectif est de minimiser les coûts de production pour toutes les périodes (168 intervalles horaires) tout en répondant à la demande du système. Les résultats obtenus corroborent le fait qu'il est possible d'obtenir un horaire de production quasi-optimal en un délai très court (beaucoup plus court qu'une heure). Les données spécifiques au problème peuvent se révéler très utiles pour accélérer la recherche et réduire encore davantage le temps de calcul.

Optimisation non linéaire de réseaux de ressources hydriques

La méthode d'optimisation de réseaux de ressources hydriques la plus courante consiste à utiliser la représentation d'un bassin hydrographique sous forme de réseau avec coûts et capacités et à résoudre le problème linéaire de flux des coûts minimal. Les modèles SIMYLD⁴, ACRES¹⁵, MODSIM3¹⁰ et WASP⁹ reposent sur ces concepts. Presque tous les modèles sont encore fondés sur la programmation linéaire, essentiellement parce que la fonction objectif relative aux priorités des permis d'exploitation hydraulique et aux autres objectifs d'affectation est linéaire. La seule exception à cette règle est la production d'énergie hydroélectrique. En règle générale, la plupart des modèles ci-dessus emploient l'algorithme des arcs non conformes². Les principales lacunes liées à l'utilisation de la programmation linéaire pour l'optimisation de réseaux sont les suivantes : (i) les modèles d'affectation des ressources hydriques basés sur des solveurs de programmation linéaire sont incapables d'incorporer un changement de débit non linéaire sur un tronçon de rivière ou de canal sans un

other LP solvers assume instantaneous availability of water from any potential source (inflow or reservoir) to any existing user in the network; and (iii) non-linearities associated with the flow bounds are handled by applying successive iterations within a time step if necessary. It should be noted that each time an iteration is performed, a slightly different problem is submitted to the optimizer resulting in a new solution that becomes the starting point for the next iteration. There is no guarantee that this process will result in convergence to the global optimum even when the objective function is convex. The problem being solved is non-linear in terms of its flow bounds, and the guessing process solves successive linear approximations of a non-linear problem.

There were some attempts to solve water resources network flow problems using non-linear and dynamic programming too. Lack of ability to distinguish between a local and a global optimum in the first case and intensity of computational requirements in the latter motivated the search for new approaches which would combine efficiency and ability to find the global optimum. Evolutionary programs as probabilistic optimization algorithms based on the similarities with the biological evolutionary process are especially promising. In this concept, a population of individuals, each representing a search point in the space of feasible solutions, is exposed to a collective learning process which proceeds from generation to generation. The population is arbitrarily initialized and subjected to the process of selection, recombination and mutation through stages known as generations, such that the newly created generations evolve towards more favourable regions of the search space. The progress in the search is achieved by evaluating the fitness of all individuals in the population, selecting the individuals with the highest fitness value and combining them to create new individuals with increased likelihood of improved fitness. Back and Schwefel¹ identified three main streams of evolutionary algorithms that have emerged in the last three decades: evolution strategies (ES); evolutionary programming (EP); and genetic algorithms (GA). The current state of the art in the field of evolution algorithms can be described as follows: (a) there is no general algorithm applicable to all problems; (b) the efficiency

processus itératif; (ii) l'algorithme des arcs non conformes et les autres solutionneurs de programmation linéaire présument d'une disponibilité instantanée de l'eau de toute source potentielle (apport d'eau ou réservoir) vers n'importe quel utilisateur existant dans le réseau et (iii) pour tenir compte des non-linéarités associées aux bornes sur les flots, on procède au besoin à des itérations successives à l'intérieur d'un intervalle. Il convient de noter que chaque fois qu'on procède à une itération, un problème légèrement différent est soumis au logiciel d'optimisation qui produit une nouvelle solution, celle-ci devenant alors le point de départ de l'itération suivante. Il n'est pas garanti que ce procédé puisse mener à une convergence vers l'optimum global, même quand la fonction objectif est convexe. Le problème à résoudre est non linéaire au plan de ses bornes sur les flots et le processus de choix aléatoire résout des approximations linéaires successives d'un problème non linéaire.

Certains ont tenté de résoudre des problèmes de réseau de ressources hydriques par la programmation non linéaire et aussi par la programmation dynamique. Or, l'impossibilité de distinguer entre un optimum local et un optimum global dans le premier cas et la complexité des calculs dans le second ont conduit à la recherche de nouvelles méthodes qui seraient efficaces et permettraient aussi de trouver l'optimum global. Les programmes évolutifs, utilisés comme algorithmes d'optimisation probabilistes fondés sur des analogies avec le processus évolutif biologique, sont particulièrement prometteurs. En vertu de ce concept, une population d'individus, dont chacun représente un point de recherche dans l'espace des solutions réalisables, est exposée à un processus d'apprentissage collectif transmis de génération en génération. La population est initialisée et soumise arbitrairement au processus de sélection, de recombinaison et de mutation à travers des stades appelés générations, de telle manière que les générations nouvellement créées évoluent vers des régions plus favorables de l'espace de recherche. Dans le cadre de cette recherche, on évalue l'adaptation de tous les individus de la population, on sélectionne les individus présentant la valeur d'adaptation la plus élevée et on les combine pour créer de nouveaux individus qui ont plus de chances de mieux s'adapter. Back et Schwefel¹ ont regroupé en trois catégories principales les algorithmes évolutifs élaborés au cours des trois dernières décennies : les stratégies d'évolution, la programmation évolutive et les algorithmes génétiques. Voici l'état actuel des réalisations dans le domaine des algorithmes évolutifs : (a) il n'existe pas

varies as a function of problem size and complexity; (c) most evolution algorithms converge to an optimal point from inside and outside of the feasible region; (d) evolution algorithms do not take into account shape or gradient of the objective function; and (e) evolution algorithms may require calibration of the search parameters to ensure efficient convergence.

Some applications of evolution algorithms in the water resources field include pipeline optimization^{5,6} and water supply networks design¹⁴. In order to address general requirements of the water resources network flow problem, an original strongly feasible evolution program (SFEP) has been developed⁷ and tested on a number of previously published transportation problems. Both transportation and minimum cost network flow problems share similar constraints, associated with (i) minimum and maximum flow along an arc; (ii) minimum and maximum flow through a node; and (iii) a continuity equation for each node. SFEP can handle non-linearities in the constraints and in the objective function. The solution process is based on a recombination operator in which all parents in a small mating pool have equal chance of contributing their genetic material to offspring. The main contributions of this approach are in the massive parallel initialization procedure which creates only feasible solutions with simple heuristic rules, and in the gene therapy procedure which fixes defective genes ensuring that the offspring resulting from the recombination is always feasible. Both procedures utilize the properties of network flows.

SFEP uses a floating-point representation to describe all individuals (feasible solutions) which evolve in the solution procedure. The algorithm is divided into five steps: initialization, evaluation, selection, recombination and gene therapy. Of those, initialization, recombination and gene therapy are of particular interest.

Initialization. The process of initialization takes advantage of the relationship between the number of dependent and independent variables describing the network problem. This means that values can be assigned to independent variables in a random manner and

d'algorithme général applicable à tous les problèmes, (b) l'efficacité varie en fonction de la taille du problème et de sa complexité, (c) la plupart des algorithmes évolutifs convergent vers un point optimal de l'intérieur et de l'extérieur de la région réalisable, (d) les algorithmes évolutifs ne prennent pas en compte la forme ou le gradient de la fonction objectif et (e) avec les algorithmes évolutifs, il est parfois nécessaire de procéder à un calibrage des paramètres de recherche pour assurer une convergence efficace.

Dans le domaine des ressources en eau, on utilise les algorithmes évolutifs entre autres pour l'optimisation de pipelines^{5,6} et pour la conception de réseaux d'approvisionnement en eau¹⁴. Pour satisfaire aux impératifs généraux du problème de réseau de ressources en eau, on a conçu un programme évolutif fortement réalisable (SFEP)⁷ original qu'on a vérifié sur un certain nombre de problèmes de transport publiés antérieurement. Les problèmes de transport et de réseau économique partagent des contraintes similaires : (i) flot minimal et maximal sur un arc, (ii) flot minimal et maximal à travers un sommet et (iii) équation de continuité pour chaque sommet. Le programme évolutif fortement réalisable peut traiter les non-linéarités dans les contraintes et dans la fonction objectif. Le processus de résolution est fondé sur un opérateur de recombinaison en vertu duquel tous les parents faisant partie d'une petite génération intermédiaire ont une chance égale de transmettre leur matériel génétique à leurs descendants. Les particularités majeures de cette méthode sont le processus d'initialisation massivement parallèle qui engendre seulement des solutions réalisables à partir de règles heuristiques simples, et la thérapie génique qui, en reconstruisant les gènes déficients, assure que les descendants engendrés par la recombinaison sont toujours réalisables. Ces deux méthodes font appel aux propriétés des réseaux.

Le programme évolutif fortement réalisable utilise une représentation à virgule flottante qui permet de décrire tous les individus (solutions réalisables) qui évoluent dans la procédure de résolution. L'algorithme se divise en cinq étapes : l'initialisation, l'évaluation, la sélection, la recombinaison et la thérapie génique. Nous nous intéresserons plus particulièrement à l'initialisation, à la recombinaison et à la thérapie génique.

L'initialisation. Le processus d'initialisation exploite la relation entre le nombre de variables dépendantes et indépendantes représentant le problème de réseau. Ainsi, on peut attribuer au hasard des valeurs aux variables indépendantes et recalculer ensuite les

then the dependent variables can be recalculated to insure compliance with the constraints. A maximum spanning tree solution is used to define a set of dependent flow variables. Flows along the arcs that do not belong to the maximum spanning tree are the independent variables. Some rules should be followed in relationship to the upper and lower limits of the independent variables. Rather than allocating values to independent variables in a completely random manner, a more practical approach can be used based on the knowledge of the objective function. In the process of assigning values to independent variables, the maximum flow-minimum cut theorem from network flow theory can be of assistance. Cut is defined here as a set of arcs which isolates a given set of nodes from other nodes in the network. To generalize, minflow and maxflow variables are defined and used in the assignment process. The process is finalized by solving the dependent variables to ensure feasibility. This process is repeated 500 to 1000 times to create the initial population. An *allocate* function is defined to perform the assignment process. It can be a simple uniform guess between zero and the current limit. However, a simple uniform guess is often not the best choice. Therefore, by inspecting the shape of the objective function for each decision variable, a much better guess can be made regarding its value, with a higher likelihood of hitting the corners of the feasible region that are essential for accelerating further search. So, the use of simple heuristic rules can significantly increase the chances of generating some good solutions in the initial population.

Evaluation. Evaluation of the initial population involves the calculation of the objective function (fitness value) for each of them.

Selection. A small fraction of the individuals with the best fitness value are selected to join the mating pool.

Recombination. The recombination operator is the main engine of the solution algorithm. It is modelled after the natural process known in biology as *crossover*. It involves two individuals creating a new organism by passing some randomly chosen genetic material from one parent and some from the other. In our problem, the genetic material is a string of

variables dépendantes pour assurer qu'elles satisfont aux contraintes. On utilise un arbre de recouvrement maximal pour définir un ensemble de variables de flot dépendantes. Les flots sur les arcs qui n'appartiennent pas à l'arbre maximal correspondent aux variables indépendantes. Il faut observer certaines règles en rapport avec les bornes supérieure et inférieure des variables indépendantes. Plutôt que d'attribuer des valeurs aux variables indépendantes de façon totalement aléatoire, on peut avoir recours à une approche plus pratique basée sur l'information de la fonction objectif. Au moment d'attribuer des valeurs aux variables indépendantes, le théorème flot maximal-coupe minimale de la théorie des réseaux peut être utile. Ici, la coupe correspond à un ensemble d'arcs qui isole un ensemble donné de sommets des autres sommets du réseau. Pour généraliser, des variables de flot minimal et de flot maximal sont définies et utilisées dans le processus d'affectation. On termine en résolvant les variables dépendantes de manière à garantir la réalisabilité. On répète cette procédure 500 à 1000 fois pour créer la population initiale. On définit une fonction d'*affectation* pour exécuter l'attribution. Il peut s'agir d'un simple choix aléatoire uniforme entre zéro et la limite actuelle. Néanmoins, un simple choix aléatoire uniforme n'est souvent pas le meilleur choix. Par conséquent, en explorant la forme de la fonction objectif pour chaque variable de décision, on peut mieux estimer sa valeur, et on a de meilleures chances d'atteindre les coins de la région réalisable qui permettent d'accélérer toute recherche subséquente. Ainsi, l'utilisation de simples règles heuristiques peut augmenter considérablement les chances de produire de bonnes solutions dans la population initiale.

L'évaluation. L'évaluation de la population initiale repose sur le calcul de la fonction objectif (degré d'adaptation) pour chaque individu.

La sélection. On sélectionne un petit pourcentage d'individus parmi ceux présentant la meilleure valeur d'adaptation pour les inclure dans la génération intermédiaire.

La recombinaison. L'opérateur de recombinaison, qui est le moteur principal de l'algorithme de solutions, s'inspire du processus biologique naturel appelé *croisement*. Il met en jeu deux individus qui créent un nouvel organisme en transmettant du matériel génétique provenant d'un parent et de l'autre et choisi de façon aléatoire. Dans notre problème, le matériel génétique est une chaîne de nombres qui correspondent aux valeurs des variables de décision constituant la solution réalisable retenue pour la

numbers which are values of decision variables forming the feasible solution selected for mating. In most technical applications of GAs, the practice is to break the solution string in only one or two points and conduct mutual replacement by exchanging the partial string segments. In SFEP, each of the chosen parents is allowed randomly to contribute their genetic material to offspring. In the genetic makeup of a new individual, some parents may contribute more of their genetic material than the others, but this is done randomly without the introduction of any bias. The approach adopted here differs from the commonly accepted wisdom that relies on some form of bias among selected parents, allowing the best parents to pass their genetic material to offspring more often.

Gene therapy. There are two possible violations of feasibility of the new individual created by recombination. Firstly, the sum of flows along one path connecting multiple nodes, or along different arcs leaving from or leading to one node, may be exceeded. Secondly, the sum of all independent variables may be insufficient. To achieve the two fixes in an easy and efficient manner, the entire mating process is carried out in a similar way as the initialization, which insures the feasibility of independent variables. The gene therapy is therefore a monitoring and adjustment procedure, which quickly fixes any individual violations of feasibility that may occur in the mating process. After finishing one iteration of the gene therapy, it is still necessary to recalculate the dependent variables and the objective function of the new individual. The new selection proceeds immediately for each individual. If the individual has better fitness value than the worst parent in the mating pool, it will be placed in the mating pool in its appropriate position, pushing all parents with less favourable fitness down by one place and pushing the worst parent out of the mating pool. Total lifetime of one individual is only a function of its fitness value and the fitness value of other individuals in the mating pool.

Implementation of SFEP generated better solutions for all functions used by Michalewicz¹³. The procedure seems to offer a good potential for further development into an efficient solver for a large class of network flow problems with

reproduction. Dans la plupart des applications techniques des algorithmes génétiques, la pratique courante consiste à rompre la chaîne solution en seulement un ou deux points et à effectuer une substitution mutuelle en échangeant des segments de chaîne partiels. Dans un programme évolutif fortement réalisable, chacun des parents choisis est autorisé au hasard à transmettre son matériel génétique à ses descendants. Dans le profil génétique d'un nouvel individu, certains parents transmettent une plus grande part de leur matériel génétique que d'autres, mais cela se fait de façon aléatoire, non biaisée. L'approche adoptée ici se distingue en cela de la pratique courante en vertu de laquelle on introduit un biais dans le choix des parents, afin de permettre aux meilleurs parents de transmettre plus souvent leur matériel génétique à leurs descendants.

La thérapie génique. Il existe deux violations possibles de la réalisabilité du nouvel individu créé par recombinaison. D'abord, on peut excéder le nombre de flots sur un chemin reliant plusieurs sommets ou sur différents arcs partant d'un sommet ou menant à un sommet. Deuxièmement, la somme de toutes les variables indépendantes peut être insuffisante. Pour corriger facilement et efficacement ces deux problèmes, on peut mener le processus de reproduction d'une manière très semblable à l'initialisation, ce qui assure la réalisabilité des variables indépendantes. La thérapie génique est donc une procédure de suivi et de mise au point qui permet de corriger rapidement toute violation individuelle de réalisabilité pouvant se produire durant la reproduction. Une fois terminée une première itération de la thérapie génique, il est encore nécessaire de recalculer les variables dépendantes et la fonction objectif du nouvel individu. La nouvelle sélection commence immédiatement pour chaque individu. Si l'individu présente une meilleure adaptation que le parent le plus faible de la génération intermédiaire, il sera inclus dans la génération intermédiaire, dans sa position appropriée; tous les parents dotés d'une valeur d'adaptation moins favorable perdront ainsi une place et le parent le plus faible sera exclu de la génération intermédiaire. La durée de vie totale d'un individu est uniquement fonction de sa valeur d'adaptation et de la valeur d'adaptation des autres individus dans la génération intermédiaire.

La mise en œuvre du programme évolutif fortement réalisable a permis de produire de meilleures solutions pour toutes les fonctions utilisées par Michalewicz¹³. Cette méthode semble avoir tout le potentiel pour devenir un solveur efficace dans une large éventail

non-linear objective functions and constraints. devenir un solutionneur efficace dans un large éventail de problèmes de réseau comportant des fonctions objectif et des contraintes non linéaires.

References / Références

1. Back, T., and H.P. Schewel, (1993) "An Overview of Evolutionary Algorithms for Parameter Optimization", *Evolutionary Computation*, 1(1), 1-23.
2. Barr, R.S., F. Glover, and D. Klingman, (1974) "An improved version of the Out-Of-Kilter method and comparative study of computer codes", *Mathematical Programming*, 7(1), 60-68.
3. Brooke, A., D. Kendrick, and A. Meeraus, (1996) *GAMS: A User's Guide*, 286 pp.
4. Evanson, D.E., and J.C. Moseley, (1970) "Simulation/Optimization Techniques for Multi-Basin Water Resources Planning", *Water Resources Bulletin*, 6(5), 725-736.
5. Goldberg, D., (1987) "Genetic Algorithms in Pipeline Optimization", *ASCE Journal of Computing in Civil Engineering*, 1(2), 128-141.
6. Ilich, N., and S.P. Simonovic, (1998) "An Evolution Program for Pipeline Optimization", *ASCE Journal of Computing in Civil Engineering*, 12(4) , 232-240.
7. Ilich, N., and S.P. Simonovic, (2000) "An Evolution Program for Non-Linear Transportation Problem", to appear in *Journal of Heuristics*.
8. Kirkpatrick, S., C.D. Gelatt, and M.P. Vecchi, (1983) "Optimization by simulated annealing", *Science*, (220)4598, 671-680.
9. Kuczera, G., and G. Diment, (1988) "General Water Supply System Simulation Model: WASP", *ASCE Journal of Water Resources Planning and Management*, 114(4), 365-382.
10. Labadie, J.W., D.A. Bode, and A.M. Pineda, (1986) "Network Model for Decision-Support in Municipal Raw Water Supply", *Water Resources Bulletin*, 22(6), 927-940.
11. Mays, L.W., and Y.K. Tung, (1992) *Hydrosystems engineering and management*, McGraw-Hill Book Co., Inc., New York.
12. Mehrota, K., K.M., Chilukuri, and S. Ranka, (1997) *Elements of Artificial Neural Networks*, The MIT Press, Cambridge, Massachusetts, 344 pp.
13. Michalewicz, Z., (1994) *Genetic Algorithms + Data Structures = Evolutionary Programs*, Springer-Verlag, KG, Berlin. Germany.
14. Savic, D.A., and G.A. Walters, (1997) "Evolving Sustainable Water Networks", *Hydrological Sciences Journal*, 42(4), 549-564.
15. Sigvaldason, O.T., (1976) "A Simulation Model for Operating a Multi Purpose Multireservoir System", *Water Resources Research*, 12(2), 263-278.
16. Teegavarapu, R.S.V., and S.P. Simonovic, (2000) "Short-Term Operation Model for Coupled Hydropower reservoirs", to appear in *ASCE Journal of Water Resources Planning and Management*.

17. Teegavarapu, R.S.V., and S.P. Simonovic, (2000a) "Optimal Operations of Multiple Reservoir Systems Using Simulated Annealing", submitted to *Water Resources Research*.
18. Wurbs, R., (1993) "Reservoir-system simulation and optimization models", *ASCE Journal of Water Resources Planning and Management*, 116(1), 52-70.
19. Yeh, W.W-G. (1985), "Reservoir Management and Operations Models: A State-of-the-art Review", *Water Resources Research*, 21(12), 1797-1818.