Solution of Optimization Continuous Centre Location Problems as Web Service

Elena Ivanova, Todor Stoilov

Abstract: *Version of the facility location problem is considered: given a set of positions, the goal is to find one or more centre points that minimize the maximum distance to a given set of points. Such kinds of problems arise in a lot of practical applications in different fields of study: management, economy, production planning, etc. The paper particularly addresses developments, related to the design of Web service for solving optimization continuous location problems.*

Key words: *Continuous location problem, Non-differentiable optimization, Cutting plane method.*

INTRODUCTION

The location problem of facilities is considered in the following form: given a set of client positions, the goal is to find one or more centre points that minimize the maximum distance to a given set of points. Facility location problems arise in a wide set of practical applications in different fields of study: management, economy, production planning and many others [5]. In general there are two types of location problems: continuous, where the decision variables can assume real values, and discrete, where the decision variables can take discrete values, such as the integers. The paper is concerned with continuous location problems.

Web services provide standard means of interoperating between different software applications, running on a variety of platforms and/or frameworks. The Web service as a programmable application logic is accessible using standard Internet protocols. Like components, Web services [6] represent functionalities that can be easily reused without knowing how the service is implemented.

The paper particularly addresses developments, related to the design of Web service for solving optimization continuous location problems.

THE CONTINUOUS LOCATION PROBLEM

Since the problem of finding multiple centres can be transformed into a single centre problem with the cost of increasing the dimension of the positions, the paper deals with the single continuous centre location problem. It is stated as follows.

There is a set M of *m* positions (locations) embedded in the $\mathbf{R}^n$ space. The problem is to find a centre point, minimizing the maximum distance to a given set of points.

Let M = $\{y_1, y_2, \ldots, y_m\}$, $y_i \in \mathbf{R}^n$, i = 1, 2, …, m, is the set of locations. The objective function for the facility allocation problem is as following ($\|.\|$ is the Euclidean norm):

$$(1)\ F(x) = \max_{y \in M} \|y - x\|$$

Thus the problem of finding a central point can be formally written as:

$$(2) \min_{x \in R^n} F(x) .$$

On the fig.1 is represented the graphics of the objective function F(x), defined by (1), for x=(x1,x2) є $\mathbf{R}^2$, and the set M, consisting of four points{(0,0),(1,0),(0,1),(1,1)}.

It is not difficult to prove that the function F(x), defined by (1), is convex. Since F(x) is convex and it is defined over the open set $\mathbf{R}^n$, the function F(x) from (1), is continuous [7]. It can be shown that the optimization problem (2) has unique solution.

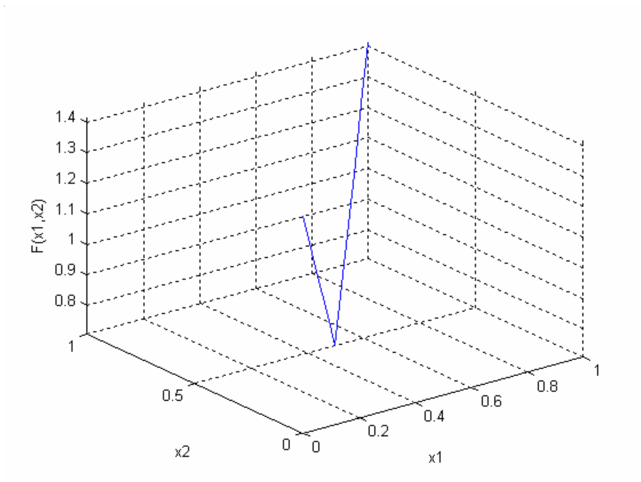

Fig.1. the function F(x)

Although the objective function F(x) from (1) is continuous, it is not differentiable. For that reason the methods of non-linear and nondifferentiable optimization are used for solving the problem (2).

Subgradient Method

This is the simplest and probably the most used method for nondifferentiable convex optimization. It is quite similar to the classical Steepest Descent method [1, 2, 3]. The main difference between these two methods consists in replacing gradients at Steepest Descent method with subgradients at Subgradient method. The Subgradient algorithm is represented on fig. 2. Hereafter the notation $\partial F(x)$ concerns the subdifferential of the function F at x, so "$s \in \partial F(x)$" means that s is a subgradient of F at x.

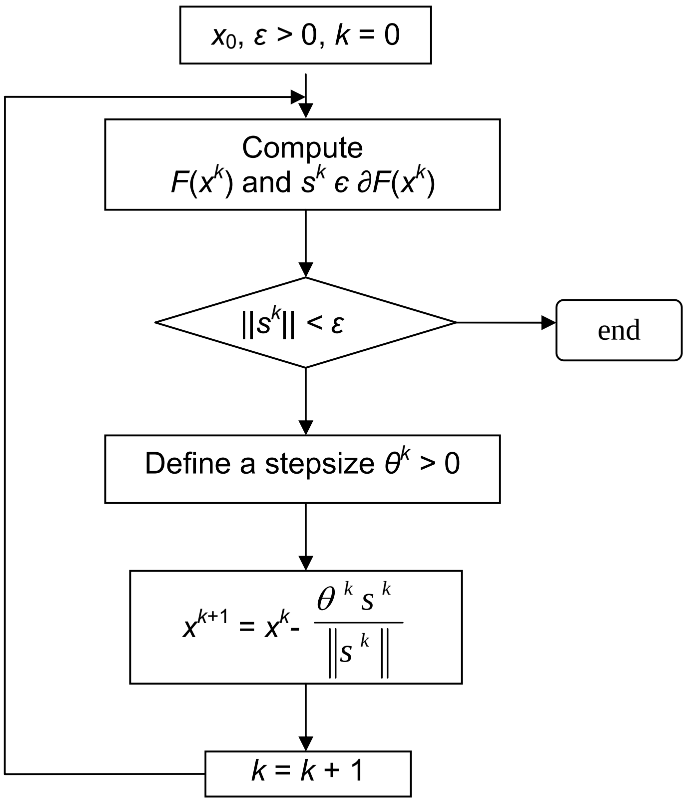

Fig.2. Subgradient algorithm

The choice of the step sizes *{θ^k}* is crucial for the performance of this algorithm. There are appropriate choices which guarantee convergence of the numerical calculations. The details can be seen in [1, 2].

Although the Subgradient method is very simple and quite easy to implement, it has several disadvantages. First of all, the stopping criterion in step 3 may never be reached, because the method works with subgradient, which is not guaranteed to be a descent direction as the gradient is in the classical Steepest Descent method. Secondly, the method ignores any past information about the trial approximations of the solution. For that reason a lot of iterations can be performed before the evaluations attempt the necessary accuracy.

Cutting plane method

In order to avoid such kind of problems, Cutting plane method was utilized to solve the optimization problem (2). There are available several cutting plane algorithms [4, 5]. The differences between these algorithms are mainly in way of choosing the next trial point. For solving the continuous location problem Kelley-Cheney-Goldstein's [5] method was used.

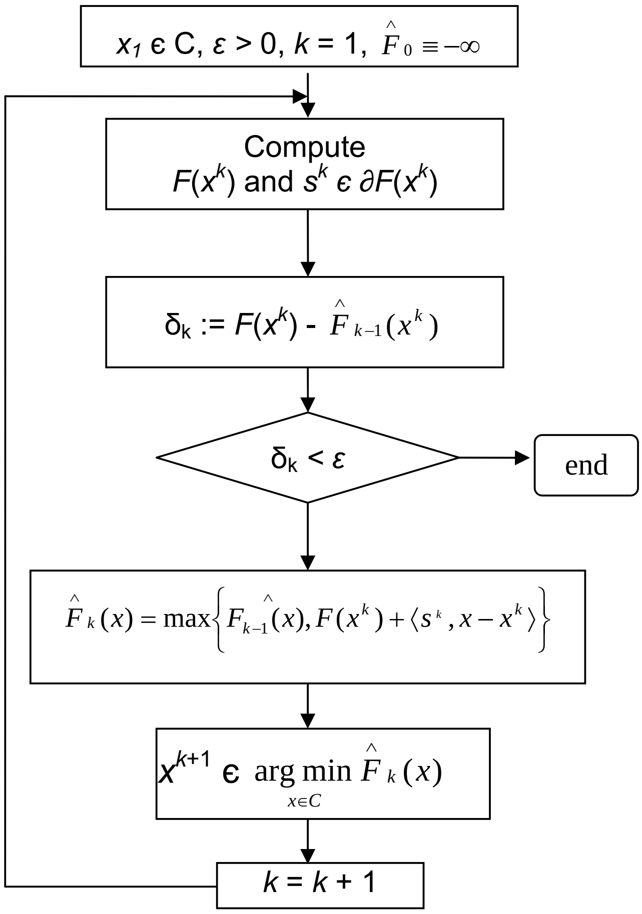

Fig.3. Cutting plane algorithm

The information from the previous iterations can be used to build descent directions and a linear model of the objective function. Consider the bundle of information:

B = {x^k, F (x^k), $s^k \in \partial F$ (x^k)}, k = 1, 2, …, L, the following piecewise-linear approximation of F can be derived [3]:

$$(3)\ \hat{F}_L(x) = \max_{k=1..L} F(x^k) + \langle s^k, x - x^k \rangle ,$$

where $\langle s^k, x^k \rangle$ is a scalar product of the vectors s^k and x^k. The sequence of functions $\hat{F}_L$, defined by (3), is called cutting-plane model of the function F. It is clear that $\hat{F}_L(x) < \hat{F}_{L+1}(x)$ for all $x \in \mathbf{R}^n$. As s^k is a subgradient of F(x) at x^k, we have F(x) ≥ $F(x^k) + \langle s^k, x - x^k \rangle$ for all $x \in \mathbf{R}^n$. Having in mind this and the construction of $\hat{F}_L$, we have that $\hat{F}_L(x) <$ F(x) for all $x \in \mathbf{R}^n$. The cutting plane model can be represented by linear constraints, so optimizing the model reduces to a LP. A local LP is solved each iteration. In order for the method to be well-defined, a compact C has to be specified (because a local LP is available). The Cutting plane algorithm is represented on fig. 3.

IMPLEMENTING THE CUTTING PLANE ALGORITHM

The design of the optimization program follows the approach, described in [1]. The components of the program are of the two main categories: (U) – user and (A) – algorithm.

(U) – user: these parts of the program characterize the concrete problem which is about to be solved. They depend on the user who poses the problem and wants to find the solution. This part consists of two subcomponents. The first (U0) refers to the choice of the initial point x_1 and the tolerance ε. The second (U1) refers to the properties of the objective function. In the curent case the number and the coordinates of the position y_i from the set M are configured here.

(A) – algorithm: the second major category includes the properties of the optimization algorithm itself – the set of rules/steps for building the iterative sequence { x_k }.

The formal architecture of the optimization program is depicted on fig.4.

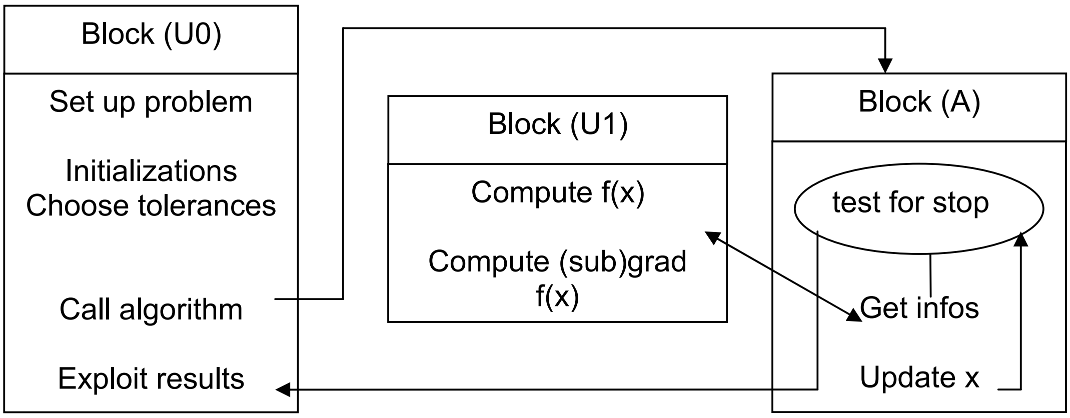

Fig.4 Architecture of the optimization program

The software implementation consists of several programm modules. The module CVector is common for all the other components and implements multidimensional vector. The next auxiliary module is Bundle and represents the set B = {x^k, F (x^k), $s^k \in \partial F(x^k)$}, k = 1, 2, …, L. CPAlgorithm performs the steps of the Cutting Plane method. The main solver consists of several parts: Initialization, Call Algorithm and Display the results. The Call Algorithm component invokes CPAlgorithm which acts together with module Oracle. Oracle evaluates the properties of the objective function. The configuration of these modules is shown on fig.5. The solver is implemented in Visual C++.

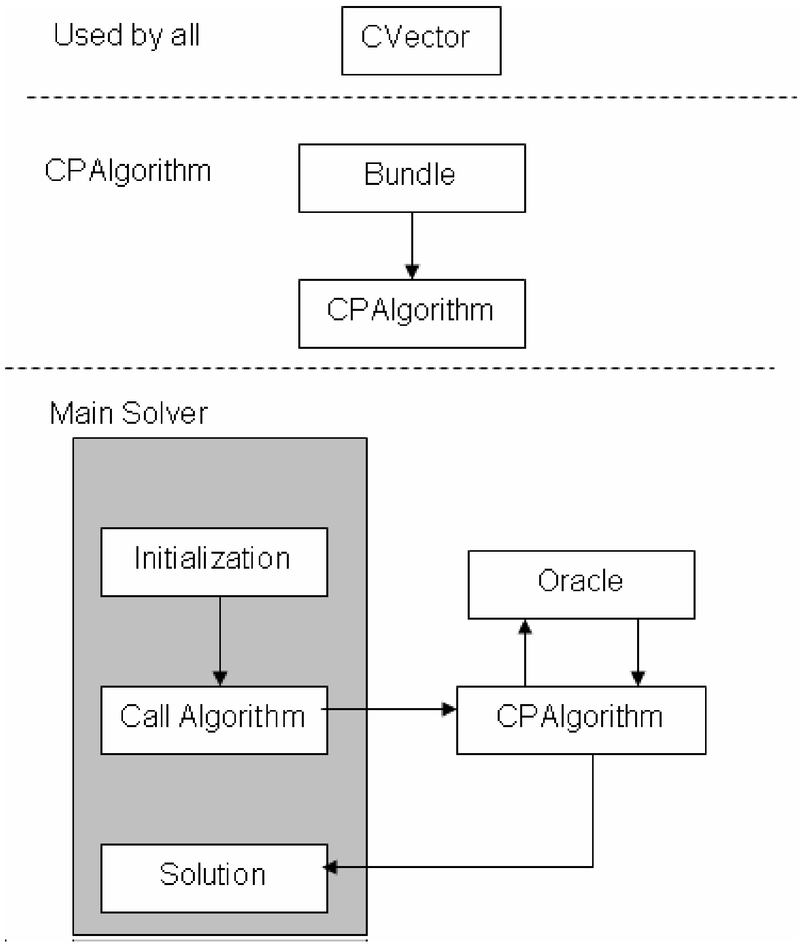

Fig.5 The scheme for Cutting plane implementation

DESIGN AND IMPLEMENTATION OF WEB SERVICE INTERFACE

The optimization program, described above, was equipped with a suitable interface, so that it can be discovered and invoked through the network by different client applications. Thus the optimization solver was developed as a service. The architecture of the whole application can be represented with the two layers model from fig.6. The external layer is the communication one, which receives and sends back massages, and the internal is the executive layer, where the core service operation is performed.

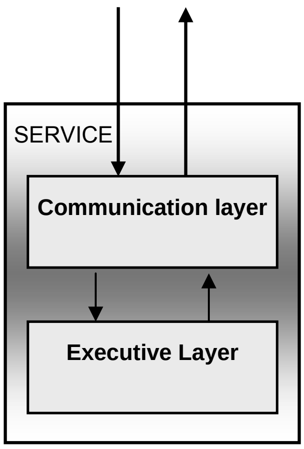

Fig.6. Two layers model of a service

For implementation of communication layer, SOAP protocol was used, applied in the Web Services protocol stack [6]. Web services provide standard means of interoperability between different software applications, running on a variety of platforms and/or frameworks. The Web service as a programmable application logic is accessible using standard Internet protocols. Like components, Web services represent functionalities that can be easily reused without knowing how the service is implemented. Particularly SOAP stands for Simple Object Access Protocol and it is a lightweight, XML-based messaging protocol that contains an envelope, header, and body, designed to exchange information in a decentralized, distributed environment [6]. The construction of the communication layer of the discussed service is shown on fig. 7.

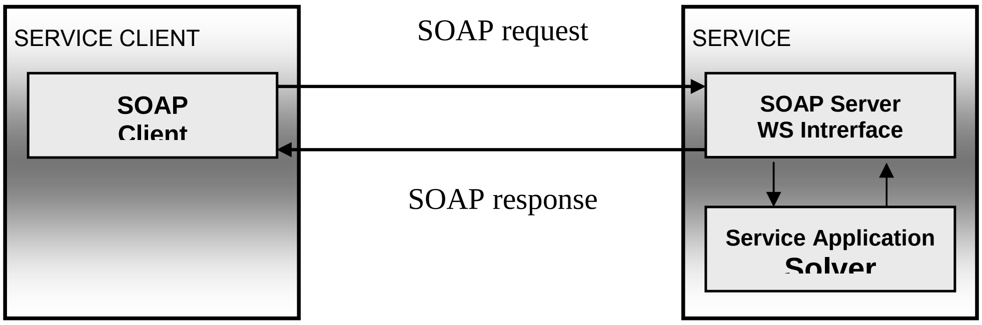

Fig.7. Communication Layer: SOAP Client/Server

CONCLUDING REMARKS

Continuous location optimization problem has been described. Two nondifferentiable optimization methods, Subgradient and Cutting plane, were implemented for the solution of the problem. Due to disadvantages of the Subgradient method, the Cutting plane algorithm was chosen for the core service implementation. The optimization program was written on C and was developed as a Web Service. The Web service is PHP based and the communication functionalities support the SOAP standard.

REFERENCES

[1] Jean – Baptiste Hiriart – Urruty and Claude Lemaréchal, *Convex analysis and Minimization Algorithms I and II,* Springer-Verlag Berlin Heidelberg 1993

[2] N. Shor, *Minimization methods for non-differentiable functions*, Springer-Verlag, Berlin, 1985.

[3] Alexandre Belloni, Introduction to Bundle Methods, Lecture Notes for IAP 2005 Course, available at http://web.mit.edu/belloni/www/

[4] S. Boyd and L. Vandenberghe, Localization and Cutting-plane Methods, Lecture Notes, 2003, available at http://www.stanford.edu/class/ee392o/

[5] Olivier PETON, *THE HOMOGENEOUS ANALYTIC CENTER CUTTING PLANE METHOD, PhD Thesis, University of Geneve, 2002*

[6] http://www.w3.org/

[7] Кендеров П., Г. Христов, А. Дончев, *Математическо оптимиране*, Университетско издателство "Климент Охридски", София, 1989

ABOUT THE AUTHOR

Assistant Prof. Elena Ivanova, Institute of Computer and Communication Systems, Bulgarian Academy of Sciences, Phone: (359 2) 979 2774, fax: (359 2) 72 39 05; E-mail: e_ivanova@hsh.iccs.bas.bg

Prof. Todor Stoilov, D.Sc., PhD, Institute of Computer and Communication Systems, Bulgarian Academy of Sciences, Phone: +359 2 73 78 20, E-mail: todor@hsi.iccs.bas.bg