Cairo University
Institute of Statistical Studies and Research

The 47th Annual

Conference

on Statistics,

Computer Sciences,

and Operation Research

Operations Research

24-27, Dec . 2012

Index
Operations Research

Approximate Techniques for Solving Multi-Objective problems.

Abd Elrazik M.M.(1), Zein el Dean R.(2), Gadallah M.H.(3)

Abstract. In this paper an efficient and effective technique of using surrogate approximations is developed to explore the design space and capture the Pareto frontier during multi- objective optimization. The method employs design of experiments and meta-modeling technique (Response surface Methodology) to sample the design space, construct global approximations from the sample data, and quickly explore the design space to obtain the Pareto frontier with specifying weights for the objectives. To demonstrate the method, the technique is applied on benchmarking of problems from the literature. A comparison study is done on the different type of multi-objective problems with two objectives for one or two variables.

Keywords: surrogate approximations, Pareto frontier, multi-objective optimization, approximation models, response surface methodology and weighted method.

1. Introduction

Many real-world problems are multi objective problems, often requiring tradeoffs between disparate and conciliating objectives. The perverseness of these tradeoffs in engineering design has given rise to radicand vast array of approaches of multi-objectives and multi- criteria optimization, [16]. In today's simulation based design environment, the design of large scale engineering systems, such as automobiles, aircraft or underwater vehicles often involves years of effort. Optimum designs are preferred in this cost-conscious world. In order to obtain the optimized designs, some optimization of the system should be performed. However, optimization of the multidisciplinary systems would take a long time to accomplish. Multidisciplinary Design Optimization (MDO) is usually cost-intensive procedure due to the large number of high fidelity function calls. Optimization costs can be reduced by using the approximation technique .This means the approximation technique reduced the cost and time to obtain the optimum solution for multi-objective problems [14]. There are many researchers used a several types of approximations techniques to solve many types of multi-objective problems, such as Response Surface Methodology (RSM), Kriging Methodology, Surrogate Approximation, Taylor series expansion, Non-Uniform Rational B-spline (NURBS), ...etc, to capture the Pareto frontier during multi-objective optimization, with minimum cost, during short time. For examples, [6] studied New method For Generating The Pareto Surface In Nonlinear Multi- criteria Optimization Problems , [15] discussed Kriging Interpolation on High-Performance Computers, [21]. compared between

---

1,2. Department of Operations Research, Institute of Statistical Studies & Research, Cairo University, Egypt.

3. Faculty of Engineering, Cairo University, Egypt.

Response surface and Kriging models for multidisciplinary design optimization ,[20]. studied Global Optimization of Problems with Disconnected Feasible Regions via Surrogate Modeling, [17] using surrogate model for traffic optimization of congested networks an analytic queuing network approach, Gad allah discussed optimization formulation based on limited data and RSM an approximation,[5] studied reference set metrics for multi-objective algorithms and [22] discussed a bi-objective optimization model to eliciting Decision maker's preferences for the promethee II method. The structure of this paper is as follows: Section (2) presents multi-objective optimization. Section (3) presents Approximation technique .Section (4) provides the solution methodology. Section (5) provides computational analysis contains two examples solved by Response Surface Methodology with weighted method.

2. multi-objective optimization

A multi-objective optimization problem consists of more than one objective to be optimized with some additional constraints specified. The formulation of the multi-objective optimization problem is given as follows:

$$\begin{array}{lll} \text{Minimize / Maximize} & F_m(x) & m = 1,2,\dots,M. \\ \text{Subject to} & g_k(x) & k = 1,2,\dots,K. \\ & x_i^{(L)} \leq x_i \leq x_i^{(U)} & i = 1,2,\dots,n. \end{array}$$

Where,

$F(x) = (f1, f2, \dots, fm)$ are a vector of objective functions.

$g_k(x)$ are called constraint functions which specify the different constraints that must be satisfied.

$x = \{x_1, x_2, \dots, x_n\}$ are an $N - tuple$ solution vector in X, a finite set of feasible solutions.

$x_i^{(L)}$ and $x_i^{(U)}$ are the lower and upper bounds respectively that bound the decision variable and hence constitute the decision space.

3. Approximation technique

Many researchers have recognized that engineering design involves multiple and often conflicting criteria or objectives (e.g., mass, stiffness, stress, deformation, stability in a structure problem). The solution set of a multi criteria design problem usually includes a large number of points, which are referred to as Pareto (efficient, non-dominated) decisions. In the multi-objective optimization, no solution optimizing all objective functions simultaneously exists in general. Instead, Pareto optimal solutions, which are "efficient" in terms of all objective functions, are introduced, therefore, the need to decide a final solution among Pareto optimal solutions taking into account the balance among objective functions, which is called "trade-off analysis". The engineering use efficient and effective approximation techniques to solve the conflict in objectives to explore the design space and capture the Pareto frontier during multi-objective optimization [2]. Response Surface Methodology (RSM) is the approximation technique which used in this peper.RSM collection of statistical and mathematical techniques useful for developing, improving, and optimizing processes, [19].The most extensive applications of RSM are in the particular situations where several input variables potentially influence some performance measure or quality characteristic of the process. Thus, performance measure or quality characteristic is called the **response**. The input variables are sometimes called **independent variables** and they are subject to the control of the scientist. The field of response surface methodology consists of the experimental strategy for exploring the space of the process or independent variables empirical statistical modeling to develop an appropriate approximate relationship between the yield and the process variables and optimization methods for finding the values of the process variables that produce desirable values of the response. In general, the relationship is: $y=f(\xi_1, \xi_2 ... \xi_k) +\varepsilon$ (1). Where y the response , $\xi_1, \xi_2 ... \xi_k$ independent variables, f is unknown and perhaps very complicated and ε is a term that represents other sources of variability not accounted for in f. Usually, ε includes effects such as measurement(statistical) error in the response, often assuming a normal distribution with mean zero and variance σ^2. Then $E(y) = \eta = E[f(\xi_1, \xi_2 ... \xi_k)] + E(\varepsilon) = f(\xi_1, \xi 2 ... \xi_k)$ (2).The variables $\xi_1, \xi_u ... \xi_k$ in Equation (2) are usually called the **natural variables,** because they are expressed in the natural units of measurement, such as degrees Celsius, pounds per square inch, etc. It is convenient to transform the natural variables to **coded variables** $x_1, x_2 ... x_k$, which are usually defined to be dimensionless with mean zero and the same standard deviation. In terms of the coded variables, the response function (2) will be written as:$\eta = f(x_1, x_2 ... x_k)$(3).Because the form of the true response function f is unknown, this must approximate it. Successful use of RSM is critically dependent upon the experimenter's ability to develop a suitable approximation for f. usually, a low-order polynomial in some relatively small region, the independent variable space is appropriate. In many cases, either a **first-order** or a **second order** model is used. The first-order model is appropriate when the experimenter is interested in approximating the true response surface over a relatively small region of the independent variable space in a location where there is little curvature in f. For the case of two independent variables, the first-order model in terms of the coded variables: $\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2$ (4).The form of the first-order model in Equation (4) is sometimes called main effects model, because it includes only the main effects of the two variables $x_1$and x_2.If there is an **interaction** between these variables, it can be added to the model easily as follows: $\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_1 x_2$ (5).This is the first-order model with interaction. Adding the interaction term introduces curvature into the response function. Often the curvature in the true response surface is strong enough that the first-order model (even with the interaction term included) is inadequate. A second-order model will likely be required in these situations. For two variables, the second-order model is: $\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_1 x_1^2 + \beta_2 x_2^2 + \beta_{12} x_1 x_2$ [6].This model would likely be useful as an approximation to the true response surface in a relatively small region. The second-order model is widely used for several reasons: [1]the model is very flexible. It can take a wide variety of functional forms, so it will often work well as an approximation to the true response surface. [2] it is easy to estimate the parameters (the β's).The method of least squares can be used for this purpose.[3] there is considerable practical experience indicating that second-order models work well in solving real response surface problems. The general motivation for a polynomial approximation for the true response function f is based on the Taylor Series Expansion around the point $x_{10}, x_{20}, \ldots x_{k0}$ .Finally, let's note that there is a close connection between RSM and linear regression analysis. For example, consider the model $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k + \varepsilon$ (7).The β's are a set of unknown parameters. This must collect data on the system studied to estimate the unknown parameters. In general, polynomial models are linear functions of the unknown β's; and the technique is referred to as linear regression analysis. **Advantages of RSM**: Well established, easy to implement, Better performance is expected for low order non-linear response functions and Best suited for small-scale applications (k<10).**disadvantages of RSM:** Less efficiency is expected when applied to problems with highly non-linear and irregular performance. Higher order polynomials can be adopted. In this case, a large number of training points is needed to accurately capture the function behavior and estimate a large number of parameters. In addition, instabilities and false optima may appear [1].

4. Solution Methodology

In this work the Pareto frontier is not approximated directly as in previous re-Searchers but, a comparison between optimal solution in RSM with weighted method and Reference optimal solution is done. Classify solutions according to the capability of approximation technique employed (best, agree and poor) solution with reference optimal solutions model. The method starts with converting the multi-objective problems with two objectives for one or two variables only into single objective using a weighted regression technique. The result of is manipulated by using response surface methodology through finding f^*_{min} at every weigh w ($0 \leq w \leq 1$), and generate the parito solution by RSM .The pseudo code of the main steps for the suggested algorithm is illustrated as follows:

1. Change the multi-objective model to single objective model by weighed method.
   (f_1) first objective. (f_2) second objective.

   $f^* = w_i f_1 + (1-w_i) f_2$ single objective model by weighed method.

   Where w_i ... weighed values from zero to 1.
2. Select min f^* at every weight from (zero to 1), with intervals (0.1).
3. Determine every x_1, x_2 (variables in f_1, f_2) at every f^* min.
4. Calculates f_1, f_2 by the value of x_1, x_2.
5. Draw the relation between f_1, f_2 in real case.
6. Draw the relation between f_1, f_2 using Response Surface Method.
7. Compare between the results in steps 5, 6 in the value of f^*.
8. Calculate the different between f^* in real case and f^* in RSM by the formula $\Delta_{Diff}\% = [F^*_{min}(RSM) - F^*_{min}(real)] / F^*_{min}(RSM)$.

5. Computational Analysis

In this section the proposed methodology is applied on seventeen benchmarking problems. Also, two problems are solved illustrate the proposed algorithm, Table (1) shows the summary of the problems tested and optimal solution by reference , Table (2) shows the summary of the test problems function using RSM, reported solutions by reference.

-Summary of problems tested and optimal solution by reference

Problem no.	Range of Variables		Optimal Solution by reference	
problem-01	x_1=[-10,10]	x_2=[-10,10]	f(x*)= 0.115	[4]
problem-02	x=[0,1]		f(x*)= 0.0145	[8]
problem-03	x_1=[0,1]	x_2=[0,1]	f(x*)= 0.0196	[3]
problem-04	x_1=[-80,1]	x_2=[0,5]	f(x*)= 0.056	[10]
problem-05	x_1=[0,5]	x_2=[0,3]	f(x*)= 2.682	[23]
problem-06	x=[0,1]		f(x*)= 0.000	[18]
problem-07	x=[0,1]		f(x*)= 0.000	[18]
problem-08	x=[-5,5]		f(x*)= 1.000	[18]
problem-09	x_1=[0.4 1.6]	x_2=[2.0,5.0]	f(x*)= 16.10	[23]
problem-10	x_1=[-5,5]	x_2=[-5, 5]	f(x*)= 0.002	[13]
problem-11	x_1=[0.4,1.6]	x_2=[2.0,5.0]	f(x*)= 3.560	[24]
problem-12	x_1=[0,5]	x_2=[0,3]	f(x*)= 0.700	[23]
problem-13	x_1=[-15,30]	x_2=[-15,30]	f(x*)= 36.320	[7]
problem-14	x_1=[-10,10]	x_2=[-10,10]	f(x*)= 87.235	[11]
problem-15	x_1=[-10,10]	x_2=[-10,10]	f(x*)= 1.685	[11]
problem-16	x_1=[-10,10]	x_2=[-2, 4]	f(x*)= 0.00	[12]
problem-17	x_1=[-9,9]	x_2=[-2, 4]	f(x*)= 0.00	[12]

Table (1) Summary of problems tested.

- Table (2) gives summary of test problems function using RSM, reported solutions by reference. The measure is developed by $\Delta_{diff\%}$. They represent deviations from optimum solution by reference and the optimum solution by RSM.

Problem no.	Without Response Surface Methodology	With Response Surface Methodology	Reference[#]	Δ_{diff} %	Notes
	f(x*)	f(x*)	f(x*)	(f*res-f*real)/f*res%	
problem-01	0.101	0.101	0.115	-0.13861%	**
problem-02	0.171	0.024	0.0145	$3.958*10^{-3}$%	⊛
problem-03	-1.000	-1.000	0.0196	1.019%	**
problem-04	-5. 0871	-4.966	0.056	1.0112%	**
problem-05	1.000	1.000	2.682	-1.682%	**
problem-06	0.000	0.000	0.000	0.000%	*
problem-07	-1.000	-0.879	0.000	1.000%	**
problem-08	0.500	0.500	1.000	-1.000%	**
problem-09	16.10	16.10	16.10	0.000%	*
problem-10	0.002	0.002	0.002	0.000%	*
problem-11	3.560	3.560	3.560	0.000%	*
problem-12	0.703	-0.966	0.700	1.072%	**
problem-13	36.451	36.451	36.320	$3.593*10^{-3}$%	⊛
problem-14	87.226	87.226	87.235	$-1.031*10^{-4}$%	**
problem-15	1.689	1.689	1.685	$2.368*10^{-3}$%	⊛
problem-16	-196.0	-196.0	0.000	1.000%	**
problem-17	-6.818	-6.818	0.000	1.000%	**

Table (2) summary of test problems function using RSM, reported solutions by reference.

We divided the problems into three classes according to optimal solution: 1- RSM best than Original solution (**). 2- RSM agrees with Original solution (*). 3- Original solution best than RSM (⊛).

5.1 RSM best than Original solution.

The list of problems where RSM technique best than the Original solutions in the optimal solution: These problems are 1, 3, 4, 5, 7, 8, 12, 14, 16 and 17 respectively. A full list of these problems is given in References. This means that RSM best in 10 problems from 17 problems. We selected one problem (problem # 5) to explain how the solutions by RSM best than the original solutions, it will be explain in section (5).

5.2 RSM agrees with Original solution.

The list of problems where RSM techniques agree with the Original solutions in the optimal solution: These problems are 6, 9, 10 and 11 respectively. A full list of these problems is given in References. This means that RSM agree with 4 problems from 17 problems. We are selected one problem (problem # 9) to explain how the solutions by RSM agree with the original solutions; it will be explain in section (5).

5.3 Original solution best than RSM.
The list of problems where Original solutions best than RSM in the optimal solution: These problems are 2, 13 and 15 respectively. A full list of these problems is given in References. This means that original solution best in 3 problems from 17 problems.

But the $\Delta_{diff\%}$ for the three problems is very narrow between Original solutions and RSM solutions as shown:

Problem-2---------------------------------- 0.003958 ≈ 0.004.

Problem-13-------------------------------- 0.003593 ≈ 0.004.

Problem-15-------------------------------- 0.002368 ≈ 0.002.

5.4 We select two problems (problem # 5 and problem # 9) from table (1) to discuss the steps of solutions of approximate techniques (Response Surface Methodology) with weighted method in different classes:
1- Problem # 5 RSM best than Original solution.
2- Problem # 9 RSM agrees with Original solution.

1- Problem 5:

$$F_1(X) = (X_1 + X_2 + 7.5)^2 + (X_2 - X_1 + 3)^2/4$$

Minimize:

$$F_2(X) = (X_1 - 1)^2/4 + (X_2 - 4)^2/2$$

Subject to: $g_1(X) = 2.5 - (X_1 - 2)^3/2 - X_2 \geq 0$

$$g_2(X) = 3.85 + 8(X_2 - X_1 + 0.65)^2 - X_2 - X_1 \geq 0$$

Where : $0 \leq X_1 \leq 5$

$$0 \leq X_2 \leq 3$$

The optimal solutions are:

F* min = 2.682 by krigining method

F* min = 1,000 by RSM with w = 0.0

Sequential of solution:

1- Determent the feasible performance space by RSM as fig (2).

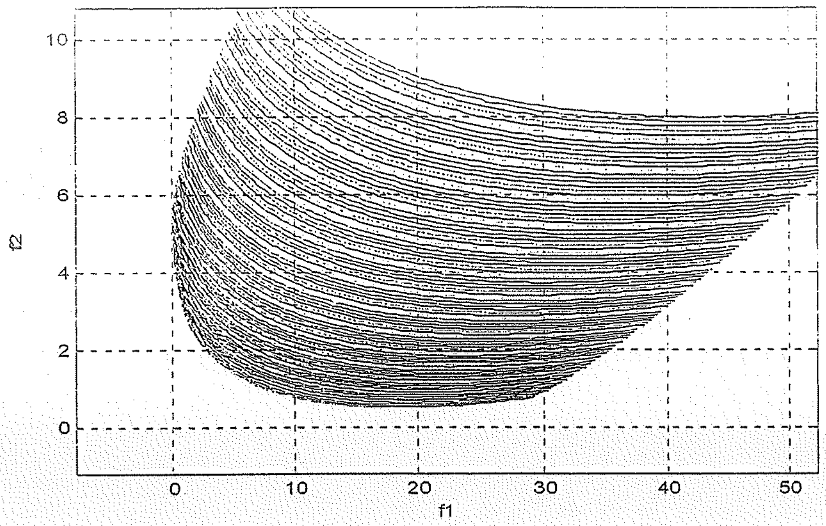

Fig (2) feasible performance space by RSM for # 5.

2- Apply RSM with weighted method to determent F^*_{min} at every weight (w), (where $0 \le w \le 1$ and the interval of w is 0.1) by drawing as fig (3).

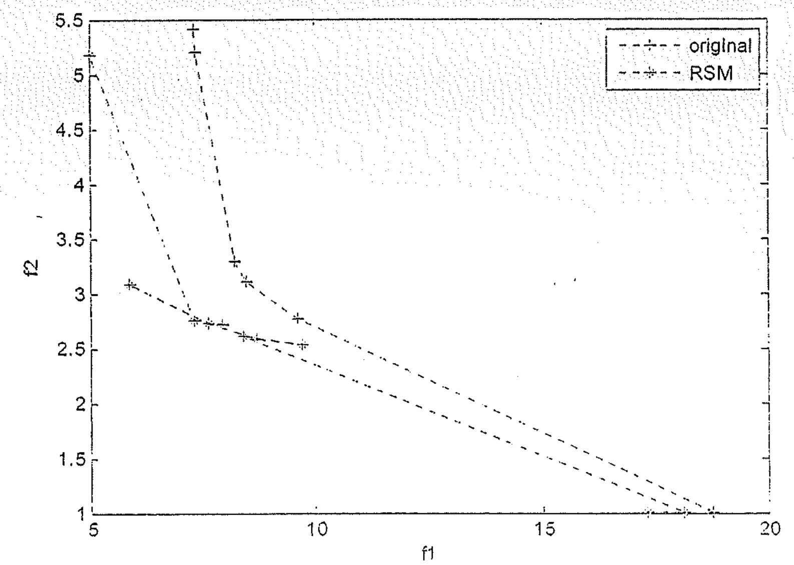

Figure (3) Pareto frontier by RSM with weighted method for problem #5.

3- Determent F* minimum calculated by RSM at every weight in # 5 and select the minimum value of F*, this value is the optimal solution of # 5 as table (3).

W	$F^*_{without\ RSM}$	F^*_{RSM}
0.0	1.000	1.000
0.1	2.776	2.713
0.2	4.140	3.768
0.3	4.718	4.685
0.4	5.249	5.044
0.5	5.739	5.166
0.6	6.228	4.759
0.7	6.686	6.140
0.8	6.897	6.877
0.9	7.096	6.832
1.0	7.283	5.015

Table (3) F^*_{min} at every weight with RSM , without RSM for problem #5.

2- Problem 9:

Minimize: $f_1=(x_1-2)^2+(x_2-1)^2$

$f_2 = x_1^2 +(x_2-6)^2$

Subject to. : $g_1=x_1-1.6 \leq 0$

$g_2=0.4-x_1 \leq 0$

$g_3=x_2-5 \leq 0$

$g_4=2- x_2 \leq 0$

Where: $0.4\leq X_1 \leq 1.6$

$2.0\leq X_2 \leq 5.0$

The optimal solutions are:

F* min = 16.10 by krigining method

F* min = 16.10 by RSM with w = 0.0

Sequential of solution:

1- Determent the feasible performance space by RSM as fig (4).

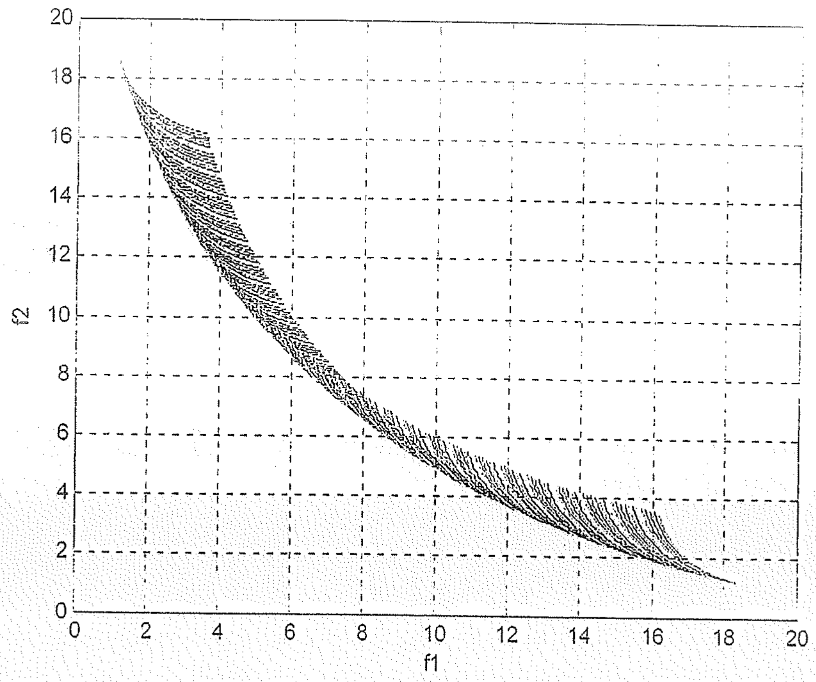

Fig (4) feasible performance space by RSM for # 9

2- Apply RSM with weighted method to determent F^*_{min} at every weight (w), (where $0 \leq w \leq 1$ and the interval of w is 0.1) by drawing as fig (5).

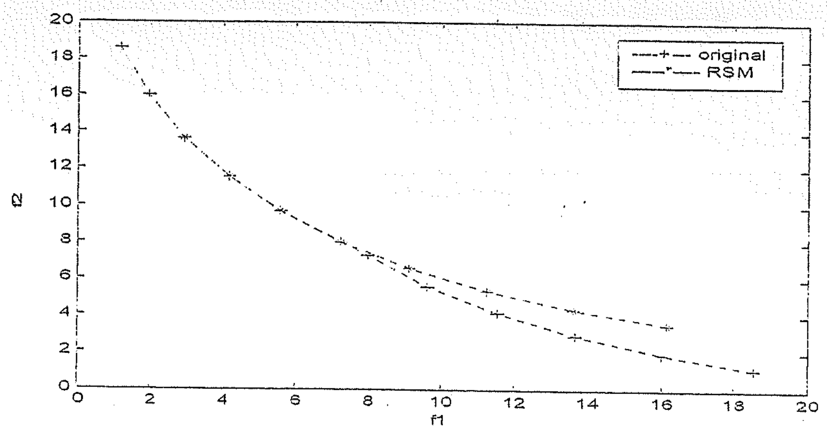

Figure (5) Pareto frontier by RSM with weighted method for problem #9

3-Determent F* minimum calculated by RSM at every weight in # 9 and select the minimum value of F* , This value is the optimal solution of # 9 in table(4).

W	$F^*_{without\ RSM}$	F^*_{RSM}
0.0	16.10	16.10
0.1	16.56	17.10
0.2	17.58	18.10
0.3	18.36	18.99
0.4	19.30	19.93
0.5	20.10	20.30
0.6	21.10	21.20
0.7	22.20	22.50
0.8	22.96	23.40
0.9	24.66	25.63
1.0	26.63	27.63

Table (4) F^*_{min} at every weight with RSM , without RSM for problem #9

6- Conclusion

We develop a new technique based on approximation techniques Response Surface Methodology (RSM) and weighted method to determent the optimal solution for multi-objectives problems in short time and low cost .This method is applied to Seventeen problems got from literature. The problems have two objectives with one or two variables. After solve seventeen problems we observed that:

- RSM best than original solution in 10 problems out of 17 problems.
- RSM agrees with original solution in 4 problems out of 17 problems.
- Original solution best than RSM in 3 problems out of 17 problems, but the differences is very narrow from original solution; we used additional sample points until get the best optimal solution. Results are inferior due to several possible reasons.

References

[1] **Ahmed, M.Y. , Qin, N. (2009),** "*Comparison of Response Surface and Kriging Surrogates in Aerodynamic Design Optimization of Hypersonic Spiked Blunt Bodies*", 13th International Conference, (2009), on Aerospace Sciences &Aviation Technology.

[2] **Buchanan, C. (2007),** "*Techniques for solving Nonlinear Programming Problems with Emphasis on Interior Point Methods and Optimal Control Problems*", Master of Philosophy, Vol. 3, pp. 115 – 136.

[3] **Carlos, A.C., Cortes, N.C. (2002),** "*Solving Multi-objective Optimization Problems Using an Artificial Immune System* "Applied Mathematics and Computation, Vol. 3, pp. 35 – 75.

[4] **Christopher, C. (2004),** "*Smart Pareto Filte Obtaining A Minimal Representation f Multi-objective Design Space* ", Applied Mathematics and Computation,Vol . 36. No. 6, pp. 721-740.

[5] **Chilukuri, K. M., Kishan, G. M. (2011)** "*Reference set Metrics for multi-objective Algorithms*" Semccopart 1, LNCS 7076, pp. 72 – 84.

[6] **Das, I., Dennis, J.E.(1996),** " *Normal-Boundary Intersection : A New Method For Generating The Pareto Surface In Nonlinear Multicriteria Optimization Problems* " , AIChE Journal, Vol. 1, No. 9, pp. 21 – 29.

[7] **Douglas A. G. Vieira, Ricardo L. S. Adriano, João A. Vasconcelos, and Laurent K.(2004),** " *Treating Constraints as Objectives in Multi-objective Optimization Problems Using Niched ParetoGenetic Algorithm*" ,IEEE , Vol. 40, No. 2, pp.201-230.

[8] **Freschi, F. , Repett, M.(2006),"** *VIS An artificial immune network for multi-objective optimization* ",In Proceeding of the IEEE International Conference on Evolution Computation, Vol. 38, pp. 975-996.

[9] **Gadallah , Mohamed, H. (2009) ,** "*Optimization Formulation Based on Limited Data and RSM An Approximation*", proceeding of the world congress on Engineering 2009 Vol. I,WCE 2009, July 1-3, (2009) London,U.K

[10] **Gao, X., Bingzhen , C. (2008),** " *Multi-objective optimization for the periodic operation of the naphtha pyrolysis process using a new parallel hybrid algorithm combining NSGA-II with SQP*", Computers and ChemicalEngineering32(2008),pp.38-65.

[11] **Jouni, Lampinen (2000),"** *Multi-objective Nonlinear Pareto-Optimization*", Lappeenranta University of technology, Laboratory of Information Processing, A Pre-Investigation Report (2000).

[12] **G.,Francisco, G.V.,Dieter,N.,Jan,J.,Ruckmann.(2007),** "*Pathfollo- wing methods for nonlinear multiobjective optimization problems*" International Journal of Management Science and Engineering Management,Vol.2(2007)No.3,pp.163-177.

[13] **Karakasis, M.K.(2006),** " *On the use of metamodels-assisted, multi-objective evolutionary algorithms* " Optimization"andEngineering,Vol.38,No.8,pp.941-957.

[14] **Kathleen, M. C., Kamneva, N.Y., Reminga, J.(2004),** "*Response Surface Methodology*", CASOS TechnicalReport,CMU-ISRI-Vol.1,pp.14-29.

[15] **Kerry, K.E., Hawick, K.A. (1998),** "*Kriging Interpolation on High-Performance Computers*", Technical ReportDHPC-035,1998,Vol.1401,pp.429-438.

[16] **Marler,R.T. ,Jasbirs, S.A.(2005),** " *Function –Transformation methods for multi –objective optimization*" AiaaJournal,Vol.37,No.6,pp.551– 570.

[17] **Osorio, C., Bierlaire, M.(2009),** "*A Surrogate Model for Traffic Optimization of Congested Networks an Analytic Queuing Network Approach*",AIAAJournal,Vol.1,pp.14– 39.

[18] [18] **Parsopoulos, K. E., Vrahatis, N.M.(2002),** "*Recent Approaches To Global Optimization Problems throughParticleSwarmOptimization*",NaturalComputing1,Vol.2,pp.35–45.

[19] **Raymond, M. H., Montgomery, D.C. (2002),** "*Response Surface Methodology: process and product optimization using designed experiment*", A Wiley- Interscience Publication, 2002, Vol. 1, No. 7, pp.53–78.

[20] **M., Papalambros, P., Goovaerts, P.(2002),** "*Global Optimization of Problems with Disconnected Feasible Regions via Surrogate Modeling*", 9th AIAA/ISSMO Symposium on Multi-Disciplinary Analysis andOptimization,Atlanta,Georgia,AIAA-Vol.3,No.1, pp. 74 – 89.

[21] **Simpson, T. W., Mauery, T. M., Korte, J. J. (1998),** "*Comparison Of Response Surface And Kriging Models For Multidisciplinary Design Optimization*", Paper No. AIAA-98-4755, Vol. 1, No. 7, pp. 381-391.

[22] **Stefan, E.A. (2011),** "*A bi-objective optimization model to eliciting Decision maker's preferences for the PROMETHEE II method*" computer & Decision Engineering (CODE) Department Université Libre de Bruxelles,Belgium(ULB).Vol.4,pp.98–113.

[23] **Wilson,B.,Daved,c.,Timothy,W.(2001),"***Efficient Pareto Frontier Exploration using Surrogate Approximations* ",Kluwer Academic Publishers Manufactured in The Netherlands, Optimization and Engineering,Vol.2,pp.31-50.

[24] **Yusheng, l. , georges, M. F. (2003),** " *Minimum Effort Approximation of the pareto Space of Convex Bi-Criteria Problems optimization and engineering*" ASME Transactions, Journal of Mechanical design ,Vol. 4, No 2. pp. 231 – 261.

Correlation Based Feature Selection for Spam Email Using Quantum Bio Inspired Estimation of Distribution Algorithm

Omar S. Soliman[1] and Aliaa Rassem[2]

Abstract

Spam or junk mails introduce many problems; they consume internet users' time, fill up a space on servers and sometimes allow for security threats. With the increasing number of spam mails, email classification becomes necessary where many spam detection models have been developed. Email feature selection becomes also a necessary preprocessing step before spam detection due to the high dimensionality of data that may affect the classification accuracy. Correlation based feature Selection (CFS) is one of the feature selection techniques that is based on features correlations in filtering good features. Bio inspired optimization (BIO) has been recently applied to solve complex problems in computer science and operations research fields. This paper introduces quantum vaccine immune clonal algorithm with estimation of distribution algorithm (QVICA- with EDA), a BIO algorithm, that helps in a faster and more efficient search in complex features spaces. The proposed algorithm integrates both quantum computing concepts and vaccination process with the immune clonal selection and EDA sampling to help CFS method in selecting the best features. The algorithm is implemented and evaluated using SpamBase dataset and is compared with the GA search algorithm. The obtained results showed the ability of QVICA-with EDA to obtain better feature sub- sets with fewer length, higher fitness values and with higher power of class predictability.

Keywords: Correlation Based Feature Selection (CFS), Quantum Computing, Vaccine Principles, Immune Clonal Algorithm, Estimation of distribution Algorithm (EDA)
.

[1,2] Faculty of Computers and Information, Cairo University, 5 Ahmed Zewal Street, Orman, Giza, Egypt.

1 Introduction

E-mail is one of the many technological developments that have influenced our lives. However, the downside of this sporadic success is the constantly growing size of unfiltered e-mail messages received by recipients. Thus, E-mail classification becomes a significant and growing problem. E-mail classification tasks are often divided into several sub-tasks. First, Data collection and representation of e-mail messages, second, e-mail feature selection and feature dimensionality reduction for the remaining steps of the task. Finally, the e-mail classification phase of the process. Spam is defined as junk or unwanted e-mail message that is delivered by the internet mail service. Spam e-mails can also clog up the e-mail system by filling-up the server disk space when sent to many users from the same organization. Moreover, virus attachments, spyware agents and phishing have become the most serious security threats to individuals and businesses recently. Many spam detection models have been proposed to solve such problems. The goal of these models is to distinguish between spam and legitimate mail messages [E]. A number of spam detection models using machine learning techniques have been proposed. As the amount of spam has been increased tremendously using bulk mailing tools, spam detection techniques should counteract with it. To cope with this, parameters optimization and feature selection have been used to reduce processing overheads while guaranteeing high detection rates. From the feature selection perspective, one of the specific problems that decrease accuracy of spam and non-spam emails classification is high data dimensionality. Therefore, Feature selection is used to find out only important features of audit data and eliminate irrelevant features for the reduction of dimensionality [22], [25].

The aim of this paper is to develop a quantum bio inspired estimation of distribution algorithm for correlation based feature selection to effectively select the relevant email features for spam detection. The algorithm is applied over the spambase benchmark dataset and compared with GA algorithm to evaluate its effectiveness. The rest of this paper is organized as follows: Section 2 introduces related work in the field of feature selection. Section 3 introduces the proposed algorithm introduced in this paper. Where experimental results and discussion are are presented in section 4. The last section is devoted to conclusions and further works.

2 Related Works and Background

Quadratic TF * IDF (Term Frequency *Inverse Document Frequency) mutual information method is a statistical method for evaluating the importance of a word for a document set or a document. It was applied to make the feature set obtain a better ability of classification. It showed a rise in the precision of classification to some degree and more classification efficiency in text classification [19]. A filter based probabilistic feature selection method, distinguishing feature selector (DFS), for text classification was proposed and compared with well-known filter approaches including chi square, information gain, Gini index and deviation from Poisson distribution. Experimental results indicated that DFS offered a competitive performance in terms of classification accuracy, dimension reduction rate and processing time [20]. The genetic algorithm (GA) was applied also in feature selection to decrease the number of useless features in the high-dimensional email body and subject. A GA feature selector with the Multi-Layer Perceptron (MLP) classifier were able to decrease the data dimensionality and increase the spam detection rate as compared against other classifiers such as SVM and Naive Bayes [22]. GA was applied again with SVM in the (GA-SVM) feature selection technique. GA is applied to optimize the feature subset selection and classification parameters for SVM classifier. It eliminates the redundant and irrelevant features in the dataset, and thus reduces the feature vector dimensionality d rastically. This helps SVM to select optimal feature subset from the resulting feature subset. This hybrid system showed a significant improvement over SVM in terms of classification accuracy as well as the computational time in the face of a large dataset [23]. A local concentration (LC) based feature extraction approach ispired from the human immune system was introduced for anti-spam that extracted local concentration features for messages. It proved to be robust against messages with variable message length. It also had better performance in terms of accuracy and F1 measure comparing with the global concentration based approach [24]. A spam detection model using random forest (RF) was developed with the help of parameters optimization and features selection. A variable importance of individual feature was provided for feature selection to select the important features on a scale between 0 and 1. The model was able to detect spam with low processing overheads and high detection rates [25]. A method of Bi-Test that utilizes binomial hypothesis testing was applied as feature selection technique for spam to estimate whether the probability of a feature belonging to the spam satisfies a given threshold or not.The method was compared with four famous feature selection algorithms (information gain, v2- statistic, improved Gini index and Poisson distribution). The experiments show that Bi-Test performs significantly better than v2-statistic and Poisson distribution, and produces comparable performance with information gain and improved Gini. Moreover, it executes faster than the other four algorithms [26].

2.1 Correlation based Feature Selection (CFS)

Correlation based Feature Selection (CFS), developed by Hall (1999), is a simple filter algorithm that ranks feature subsets according to a correlation based heuristic evaluation function. The bias of the evaluation function is toward subsets that contain features that are highly correlated with the class and uncorrelated with each other. Irrelevant features should be ignored because they will have low correlation with the class. Redundant features should be screened out as they will be highly correlated with one or more of the remaining features. CFS has two main phases, the first is calculating the matrix of feature-feature and feature-class correlations. The second phase is a search procedure that is applied to explore the features space and get the optimal subset. To examine all possible subsets and select the best is prohibitivedue to large features space. Various heuristic search strategies such as, best first, are often applied to search the features space in reasonable time. The CFS correlation based heuristic evaluation function is defined as in equation 1 [3], [5], [13].

$$M_s = kr_{cf}/\sqrt{k + k(k-1)}r_{ff} \quad (1)$$

Equation 1 is Pearsons correlation where all variables have been standardized. M_s is the heuristic merit of a feature subset S containing k features, r_{cf} is the mean feature-class correlation and r_{ff} is the average feature–feature intercorrelation. The numerator of this equation can be thought of as providing an indication of how predictive of the class a set of features is; and the denominator of how much redundancy there is among the features [5], [11].

2.2 Quantum Inspired Evolutionary Algorithms and Estimation of Distribution Algorithms

Immune clonal algorithm (ICA) is an optimization algorithm that simulated the human immune system where antibodies are used as candidates solutions and cloning and mutataion are the evolution operators. Quantum inspired evolutionary algorithms (QIEAs) were introduced in the 1990s to get advantage of the quantum computing to increase EAs performance. Quantum inspired ICA (QICA) inpires the quantum concepts of quantum bits, superposition and obseration (for more details about quantum properties, see [10]). Using inspired quantum properties, QICA can reduce the computataional time and maintain higher performance. The vaccine operator was added to the ICA to help in diversifying the solutions through the search space. The vaccinatation process may take a long time so the estimation of distribution algorithms (EDAs) are used to sample large number of vaccines in a reduced time. Estimation of Distribution Algorithms (EDAs) are population based algorithms with a theoretical foundation on probability theory. They can extract the global statistical information about the search space from the search so far and builds a probability model of promising solutions [5]. Unlike genetic algorithms (GAs), the new individuals in the next population are generated without crossover or mutation operators. They are randomly reproduced by a probability distribution estimated from the selected individuals in the previous generation.

3 Proposed Algorithm

The effectiveness of correlation based features selection method (CFS) depends on the effectiveness of the search algorithm and its ability to examine the most number of possible feature subsets to select the most representative one. As in equation 1, the fitness of any candidate subset is evaluated by dividing the subset class prediction power with the features interactions so only relevant and non redundant features are favored for selection. The proposed quantum bio inspired estimation of distribution algorithm algorithm for CFS, it is based on the quantum vaccined immune clonal algorithm, and EDA. The schema of the proposed algorithm is shown in figure 1.

As shown in figure 1, the algorithm has two main stages. First Phases is about data preprocessing and the second is the CFS with its two phases.

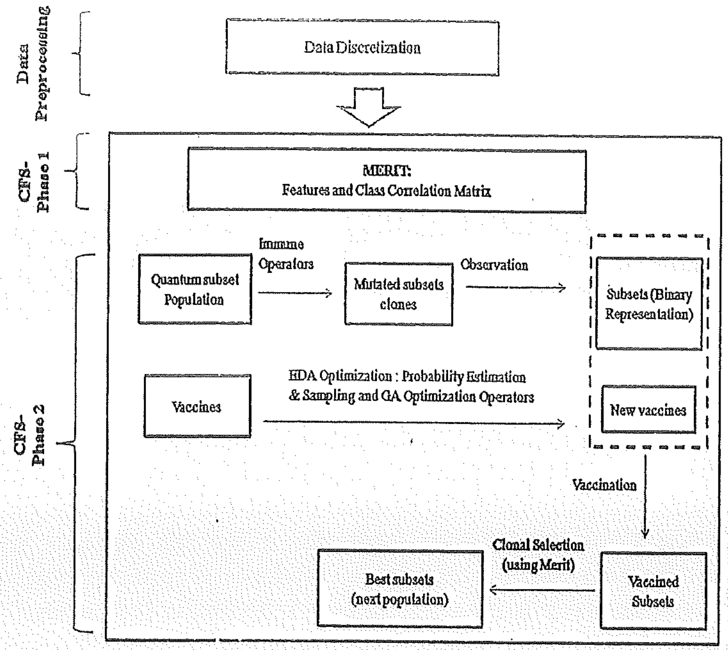

Figure 1: The schema of proposed algorithm

The details of proposed algorithm are described as follows:

- **Data Preprocessing Phase**

  This phase is concerned with data preprocessing of the spambase dataset; it is responsoble for data discretization so the data can be processed by the CFS. Also, all features are normalized in this phase to be handled equally as follows:

  - **Data Discretization**: As many filter methods work on discrete data and as KDD data set has some continuous features, a discretization step is the second step for the preprocessing of the data. Equal frequency discretization (EFD) is the used discretization method with k, number of intervals, equals 10.

- CFS Phases

  after the dataset is ready for the feature selection process where all features have discrete numeric values, the two CFS phases are implemented as follows,

  - **Phase 1: Correlations Matrix**: a matrix of feature-feature correlations and feature-class correlations is computed from the features' values. This matrix is the evlautaion criteria of any possible feature subset. It is the base of subsets selection in the second phase.
  - **Phase 2: Search Technique**: The QVICA-V with EDA is applied for this phase to search, rank and select better subsets. This technique integrates the quantum computing and immune clonal selection principles with the vaccination and EDA sampling mechansim to improve the solutions fitness and the degree of diversity. Two populations, one for solutions (feature subsets) and one for vaccines are initialized first in the quantum bit representation. After initilization, two parallel flows take place for the evolution process where one of them is over the vaccines population and the other over the subsets population.
    - **First flow**: The vaccines population is divided into two sub populations. Genetic operators are used to evolve the first sub population by the crossover and mutataion operators where EDA is applied on the second subpopulation. It estimates a Probability model that represents the fittest vaccines then new vaccines are sampled from this model. Fittest vaccines are the farthest vaccines with higher distance values from subsets so as higher space exploration degree is ensured.
    - **Second flow**: Quantum subsets, candidate problem solutions, are evolved by the clonal and quantum mutataion operators then observed into more subsets. The final subsets are vaccined by the new generated vaccines then the immune clonal selection took place to select best subsets as the next population where the phase is repeated until the number of iterations is reached [8].

Table 1: SpamBase dataset Attributes Description

Attributes Range	Values Range	Description
1 to 48	[0,100]	percentage of words in the e-mail that match WORD
49 to 54	[0,100]	percentage of characters in the e-mail that match CHAR
55 to 57	[1,...]	the length of sequences of consecutive capital letters.
58	0,1	denotes whether the e-mail was considered spam (1) or not (0)

4 Experiemntal Results

Many experiments have been done to test the QVICA-with EDA search technique against the GA search. All the experimenst are done on the benchmark dataset in the spam mails field is the spambase dataset. The Spambase dataset is an e-mail message collection containing 4601 messages, being 1813 (39) marked as spam messages, was created by Hopkins et al.. The collection of spam e-mails came from individuals who had filed spam and the collection of non-spam e-mails came from filed work and personal e-mails [25]. There are 58 attributes where 57 of them are continuous and the last one is the nominal class which denotes whether the e-mail was considered spam (1) or not (0). Most of the attributes indicate whether a particular word or character was frequently occuring in the e-mail where table 1 shows a detailed description of each attribute.

Different parameters settings are tested through three experiments, where the number of the subsets population size is set to 1000 with fixed number of iterations of 1000. The initial quantum population size is set to 5 with clone scale of 4 in addition to crossover and mutation probabilities settings. Three values for mutation probabilty are set, low value of 0.25, average value of 0.5 and high value of 0.8. Three evaluation measures are used the base of the comparison in this paper. Theses measures include the best subset fitness obtained, the average feature- feature (f-f) correlation and the average

Table 2: Best Feature Subset, with its number of selected features, and its Fitness Value using GA and QVICA-with EDA Search

	Exp.	Best fitness value	Features Number	Selected features
GA Search	1	0.445	17	1,3,6,8,12,13,17,18,19,21,28,32,34, 40,44 ,49 ,50
	2	0.4536	16	1,2,3,5,6,9,12,16,19,21,23,28,37,42,43,50
	3	0.4479	15	1,3,10,12,18,19,21,23,24,28,29,35,36,38 ,56
QVICA-with EDA Search	1	0.5494	4	3,12,19,21
	2	0.5178	6	3,5,12,18,19,21
	3	0.4954	7	1,3,12,19,21,51,57

features-class (f-c) correlation of the best subset. Table 2 shows the first evaluation measure, the best features subset obtained at end of each experiment with the number of features it includes and its fitness value (CFS evaluation function score) for GA and proposed algorithm.

As shown in table 2, the QVICA-with EDA search method outperformed the GA search in getting optimal subsets with higher fitness values of CFS merit function. The largest difference obtained of the fitness values is in the first case, where the mutataion probabailty has small value where the lowest difference obtained in the third experiemnt. The proposed algorithm is also more effective than GA in searching for the relevant and non redundant features which is clear from the number of features obtained from both of them at each experiment. QVICA-with EDA is able to get features subset with fewer length, which means that it eliminates more irrelevant and redundant features than GA.

More tests are done where two additional evaluation measures are calculated for both algorithms which are the average feature-feature (f-f) correlation and average feature-class (f-c) correlation values of the best subset. CFS aims to both lower the first measure, to ensure redundancy elimination and to maximize the second measure for higher class predictability power. Table 3 shows the values of these measures recorded for both algorithm through the five experiments.

Table 3 proves the ability of QVICA-with EDA search method of main-

Table 3: Average (f-f) and (f-c) Correlations of the Best Subset of QVICA-with EDA and GA Search

Exp.	Average features Correlations		Average Class correlations	
	GA	QVICA-with EDA	GA	QVICA-with EDA
1	0.9929	0.9976	0.4435	0.5489
2	0.9932	0.994	0.4521	0.5165
3	0.9921	0.9906	0.4463	0.4934

taining the CFS second goal as in all experiments, its selected subset has features with higher class predicability than the GA (as listed in the last two columns of the table). For features interactions, both algorithms seem to behave the same and achieve similar results. Therfore, the proposed algorithm is not able to improve the correlations values, but it still meets the first goal through obtaining the same results of the GA search.

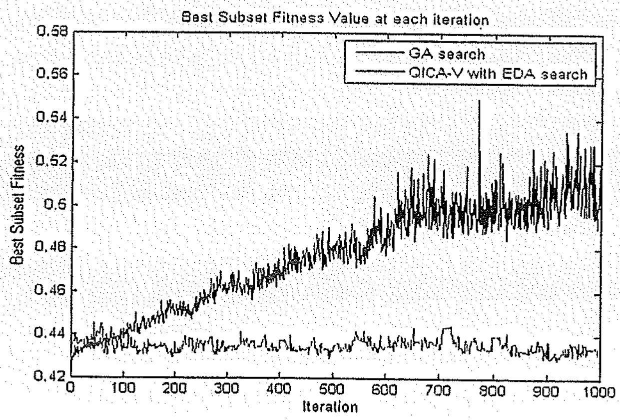

Figure 2: Best Subset fitness value found by QVICA-with EDA and GA searches at each iteration at experiment 1

For more analysis regarding the performance of the proposed algorithm,

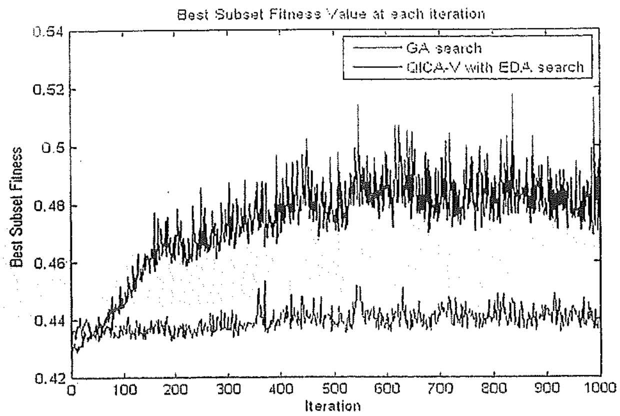

Figure 3: Best Subset fitness value found by QVICA-with EDA and GA searches at each iteration at experiment 2

the best subset fitness value obtained at each experiment's iterations are visualized for both algorithms in figures. 2, 3 and 4. In all of these figures, the proposed algorithm has an increasing curve of the best fitness value through all iterations where GA has a random behavior. The increasing curve shows how the whole subset population in QVICA-with EDA moves towards the best solution due to the EDA sampling that always sample the best solutions found so far at each iteration. A small degree of randomization also appears in the curve and is gained from both the vaccination and quantum observation process. This randomization means that no only the search direction is towards optimality but also there is a space at each iteration for searching more areas in the features space to explore new unknown subsets. On the other hand, the GA search strategy appears to be highly random according to the GA operators but it fails to reach or get closer to optimality regions which is clear from the maximum values obtained by both algorithms. Comparing figures . 2 and 4, the algorithm's search strategy takes more iterations till achieving a stable behavior in the first experiment than the third one. High mutataion probabailty value in the third experiemnt hepls in exploring more areas in the features space and thus finds new subsets that hurries in achieveing staibilty around the optimility region. Low values for the mutataion probabailties has also an advantage that appears in fig 2 where the best subset in the population reaches higher values of fitness (with respect to the values in fig 4).

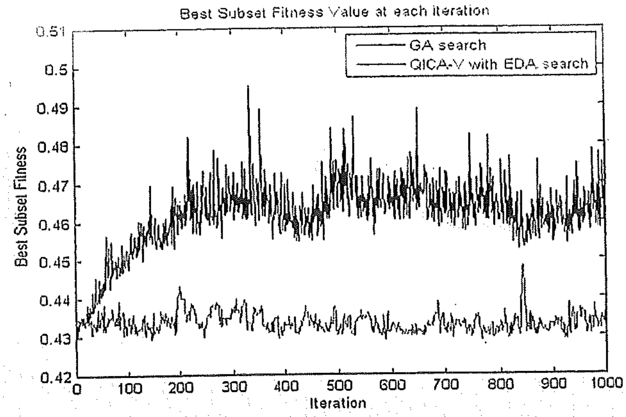

Figure 4: Best Subset fitness value found by QVICA-with EDA and GA searches at each iteration at experiment 3

5 Conclusions

This paper introduced a new CFS method based on a quantum bio inspired estimation of distribution algorithm to effeciently select more relevant email features to improve the spam mails classficiation accuracy. The algorithm that combines quantum computing (QC) concepts with vaccination principles, the immune clonal selection and estimation of distribution algorithm (EDA) sampling. The quantum properties of q-bits representation, quantum mutation and observation with the EDA sampling were utilized for improving the performance of the search and in reducing the computation time. The proposed algorithm employed as a search technique for CFS to find the optimal subsets of the features space. It was implemented and evaluated using benchmark dataset SpamBase dataset, and compared with results obtained using GA algorithm. Results showed that it was capable to obtain better feature subsets with fewer length, higher fitness values and reducing computation time. For future work we intend to apply the proposed search algorithm to a real application; more experiments and more comparative study with the most recent Machine learning algorithms.

References

[1] Olusola, A., Oladele, A. and Abosede, D.: Analysis of KDD 99 Intrusion Detection Dataset for Selection of Relevance Features. Proceedings of the World Congress on Engineering and Computer Science, vol. I, 20-22, (2010).

[2] Cantu-Paz, E.: Feature Subset Selection by Estimation of Distribution Algorithms. Proceedings of the Genetic and Evolutionary Computation Conference (GECCO2002), 303-310, (2002).

[3] Hall, M.: Correlation-based feature selection for machine learning. PhD Thesis., Department of Computer Science, Waikato University, New Zealand, (1999).

[4] KDD99 archive: The Fifth International Conference on Knowledge Discovery and Data Mining. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[5] Hall, M. and Smith, L.: Feature Selection for Machine Learning: Comparing a Correlation based Filter Approach to the Wrapper. Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference, 235-239, (1999).

[6] Hoque, M., Mukit, Md. and Bikas, Md.: An Implementation of Intrusion Detection System Using Genetic Algorithm. International Journal of Network Security and Its Applications (IJNSA), vol. 4, no. 2, 109-120, (2012).

[7] Alomari, O. and Othman, Z.: Bees Algorithm for feature selection in Network Anomaly detection. Journal of Applied Sciences Research 8 (3),1748-1756, (2012).

[8] Soliman, O. S., Rassem, A.: A Bio Inspired Estimation of Distribution Algorithm for Global Optimization. T. Huang et al. (Eds.): ICONIP 2012, Part III, LNCS 7665, 645-652, (2012).

[9] Srinivasu, P., Avadhani, P., Satapathy, S. and Pradeep, T.: A Modified Kolmogorov-Smirnov Correlation Based Filter Algorithm for Feature Selection . Advances in Intelligent and Soft Computing, vol. 132, 819-826, (2012).

[10] Niu, Q., Zhou, T. and Ma, S.: A Quantum-Inspired Immune Algorithm for Hybrid Flow Shop with Make span Criterion. Journal of Universal Computer Science, vol. 15, no. 4, 765-785,(2009).

[11] Boln-Canedo, V., Snchez-Maroo, N. and Alonso-Betanzos, A. :Feature selection and classification in multiple class datasets: An application to KDD Cup 99 dataset. Expert Systems with Applications 38 (5), 5947-5957, (2011).

[12] He, X., Zeng, J., Xue, S. and Wang, L.: An New Estimation of Distribution Algorithm Based Edge Histogram Model for Flexible Job-Shop Problem. Communications in Computer and Information Science, vol. 158, part 5, 315-320,(2011).

[13] Saeys, Y., Inza2, I. and Larranaga, P.: A review of feature selection techniques in bioinformatics. Bioinformatics 23 (19), 2507-2517, (2007).

[14] Saeys, Y., Degroeve, S., Aeyels, D., Van de Peer, Y. and Rouze, P.: Fast feature selection using a simple estimation of distribution algorithm: a case study on splice site prediction. Bioinformatics, vol. 19, 179-188,(2003).

[15] Saeys, Y., Degroeve, S. and Van de Peer, Y.: Feature Ranking Using an EDA-based Wrapper Approach. Studies in Fuzziness and Soft Computing, vol. 192, 243-257, (2006).

[16] Yang, Y. and Webb, G.: A Comparative Study of Discretization Methods for Naive-Bayes Classifiers.Proceedings of Pacific Rim Knowledge Acquisition Workshop, 159-173, (2002).

[17] Chunga, Y. and Wahid, N.: A hybrid network intrusion detection system using simplified swarm optimization (SSO). Applied Soft Computing 12(9), 3014-3022, (2012).

[18] Zhu, Z., Ong, Y. and Dash, M.: Wrapper-Filter Feature Selection Algorithm Using A Memetic Framework. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions 37(1), 70-76,(2007).

[19] Gong, S. , Gong, X. and Wang, Y. : Study of E-mail Filtering Based on Mutual Information Text Feature Selection Method. Advances in Intelligent and soft Computing, vol. 127, 33-39, (2012).

[20] Uysal, A. and Gunal S. :A novel probabilistic feature selection method for text classification. Knowledge-Based Systems, (2012).

[21] Beiranvand, A. , Osareh, A. and Shadgar, B. :Spam Filtering By Using a Compound Method of Feature Selection. Journal of Academic and Applied Studies 2(3) , 25-31, (2012).

[22] Behjat, A.R. : GA-based feature subset selection in a spam/non-spam detection system. Computer and Communication Engineering (ICCCE), 675 - 679, (2012).

[23] Temitayo, F., Stephen, O. and Abimbola, A. :Hybrid GA-SVM for Efficient Feature Selection in E-mail Classification. Computer Engineering and Intelligent Systems, vol. 3, no.3, (2012).

[24] Zhu, Y. and Tan, Y. :A Local-Concentration-Based Feature Extraction Approach for Spam Filtering. IEEE transactions on information forensics and security, vol. 6, no. 2, (2011).

[25] Lee, S., Kim, D. and Park, J. : Cost-Sensitive Spam Detection Using Parameters Optimization and Feature Selection. Journal of Universal Computer Science, vol. 17, no. 6, (2011).

[26] Yang, J. ,Liu, Y., Liu, Z., Zhu, X. and Zhang, X. :A new feature selection algorithm based on binomial hypothesis testing for spam filtering. Knowledge-Based Systems, vol. 24 , 904914, (2011).

A hybrid classification system for Hepatitis C Virus

Omar S. Soliman1 and Eman AboElhamd2

Abstract:
This paper proposes a decision support system for hepatitis C Virus (HCV) diagnosis and treatment. The proposed system integrates principle component analysis (PCA) algorithm and support vector machine (SVM). The PCA algorithm is used for selecting features vector, the SVM algorithm with different kernel functions is used for patient's classification. The proposed system starts with data preprocessing phase using local mean algorithm. This is followed by features extraction phase using PCA algorithm then classification using SVM algorithm. The proposed system is implemented and evaluated using two different data sets, one is the benchmark HCV data set from UCI repository of machine learning databases and the other is a real data set from Egyptian National Liver Institute, and compared with different classifier systems. The obtained results showed the effectiveness of the proposed system in achieving higher classification accuracy with lower processing time.

Keywords: Hepatitis C Virus (HCV), Artificial Neural Network (ANN); Support Vector Machine (SVM), Radial Basis Function (RBF).

1. Introduction:

HCV is a chronic disease, in some cases, those with cirrhosis will go on to cause liver failure, liver cancer or life-threatening esophageal and gastric varices.
Nowadays, people who have been affected from hepatitis virus are found all over the world. Liver's swelling and redness without indicating any particular reason is referred to as "Hepatitis". Hepatitis virus arises due to one of three viruses; Hepatitis A, hepatitis B, and hepatitis C are the prominent viruses. Hepatitis C virus is the most severe type. It assaults the liver and caused swelling and redness in liver. We could found that large number of inhabitants who had infected from this virus said that they even didn't feel any indication by which they recognized that they have been infected from such severe virus, most of them come to know about their state during their usual health check-up. Hepatitis C virus will also be transmitted to the new born baby if the mother is tainted from this virus. In this virus, if the patient is having cirrhosis then the possibility of liver cancer raises. According to the professionals, the rate of death from hepatitis will be three times more in the next twenty years.

Fibrosis liver disease could be considered as a deadly disease that adversely affects the lives of many people, which is the final result of injury to the liver. Accurate assessment of the degree of fibrosis is important clinically, especially when treatments aimed at reversing fibrosis are being evolved. Liver biopsy has been considered to be the gold standard to assess fibrosis. However, liver biopsy being invasive and is not favored by patients or physicians, alternative approaches to assess liver fibrosis have assumed great importance. Moreover, therapies aimed at reversing the liver fibrosis have also been tried lately with variable results. So, the aim of this paper is to develop a classification system for HCV.

1;2 Faculty of Computers and Information, Cairo University, 5 Ahmed Zewal Street, Orman, Giza, Egypt.

The rest of the paper is organized as follows; section 2 describes the problem background and related works. The proposed system is introduced in section 3 and experimental results are presented in section 4. Where, the last section is devoted to conclusions and further research.

2. PROBLEM BACKGROUND AND RELATED WORKS:

HCV is a major public health problem all over the world, so, lots of work had been done in the area to help in diagnosis and treatment of this disease. Artificial Neural Network (ANN) is one of the most common methods in this area. An automatic diagnosis system for hepatitis C virus which integrates PCA and ANN for classification of HCV and deals with mixture of feature extraction and classification is proposed in [1].In [2],an Expert system for the diagnosis of the Hepatitis B virus disease was proposed; they employed the generalized regression neural network to classify the patient in two categories: Infected or immune mentioning the causes and severity. They showed that Generalized regression ANN will be the best suitable ANN for hepatitis B diagnosis and helps in reducing extra time consumption in treatment. Even if there is any number of missing parameters in blood test. Also, in [3] an ANN algorithm was proposed to Predict Cirrhosis in Patients with Chronic Hepatitis B Infection with Seven Routine Laboratory Findings. They showed the ability of ANN to predict the presence or absence of cirrhosis in patients with chronic hepatitis B, and showed that if the ANN value of certain laboratory manifestations is negative, there will be a 95% chance of having a negative liver biopsy. In [4], an ANN algorithm was introduced as an aided non-invasive grading evaluation of hepatic fibrosis by duplex ultrasonography. This model had good sensitivity and specificity in quantitative diagnosis of hepatic fibrosis or liver cirrhosis. It may help non-invasive grading diagnosis of liver fibrosis in clinical practice. A comparison between Back-propagation and Naive Bayes Classifiers was introduced in to diagnose hepatitis disease was introduced in [5].A decision support model to predict 3-month mortality risk of acute-on-chronic hepatitis B liver failure using ANN was introduced in [6]. In [7], they proposed ANN system and multiple logistic regression (MLR) analysis based on clinical factors to predict a 6-year incidence of metabolic syndrome (MetS) including the insulin resistance index that calculated by homeostasis model assessment (HOMA-IR). In [8], they presented a framework to establish a prediction system for the personalized treatment of chronic hepatitis C virus usingalogistic regression model for personalized therapy based on age, viral genotype, initial viral load, aspartate aminotransferase/alanine aminotransferase ratio, α-fetoprotein levels, and IL28B single nucleotide polymorphism genotype. They showed that the personalized approach to therapy may better inform treatment decisions and reduce incidence of side effects and antiviral resistance.

An algorithm based on decision trees was proposed to determine the outcome of patients with acute liver failure, which highlight most common causes of hepatitis, these may be useful to determine the indication for liver transplantation [9]. Also, in

[10]A prediction model of hepatitis C patients to drug treatment using decision tree learning algorithm was proposed and used to predict with high probability. A decision-tree analysis also, was proposed as predictive model for pretreatment prediction of anemia progression by pegylated interferon alpha-2b plus ribavirin combination therapy in chronic hepatitis C infection. This model was useful for predicting the probability of severe anemia, and has the potential to support clinical decisions regarding early dose reduction of ribavirin [11]. In [12], a soft computing classification criterion was proposed to design and develop user friendly diagnostic system for Hepatitis B. They utilized clinical data of the patients and the opinions of medical experts to design the diagnostic system. In [13], they introduced the usage of a modern information system using medical informatics technologies to support in the task of diagnosis and efforts entreating to overcome the problems related to the liver disease. A hybrid model using Particle Swarm Optimization and Case Based Reasoning for hepatitis disease diagnose was proposed in [14]. The CBR was used for feature extraction method. Where the PSO was used to assemble a decision making system based on the selected features and diseases recognized.

In [15],an intelligent HCV diagnosis system using Principle component analysis and least square support vector machine classifier for hepatitis diagnosis. A machine learning model for hepatitis disease diagnosis that hybridizes SVM and simulated annealing (SA) was proposed in [16]. In [17], they introduced an intelligent system using multiple linear regressions (MLR) and SVM for developing quantitative structure activity relationship (QSAR) model for HCV.A machine learning methods to predict HCV non-structural proteins 5B polymerase inhibitors was proposed [18]. In [19], they developed decision model for chronic hepatitis C virus infection that uses a Markov model to estimate lifetime costs and quality-adjusted life-years in two treatment strategies (a standard-duration therapy and truncated therapy). In [20], they proposed a specific HCV evolution and response to the combined interferon and ribavirin therapy based on Bayesian networks (BN), Linear projection (LP) and Self-Organizing tree (SOT) models. The BN models employed to predicted therapy outcomes, gender and ethnicity, where LP and SOT models used to confirm associations of the HVR1 and NS5A structures with response to therapy and demographic host factors predicted by BN. In [21], they applied the serum Fourier Transform infrared spectroscopy for noninvasive assessment of hepatic fibrosis in patients with chronic hepatitis C. This work strongly suggests that the serum from CHC patients exhibits infrared spectral characteristics, allowing patients with extensive fibrosis to be differentiated from those with no hepatic fibrosis. In [22], a genetic polymorphism in IL28B and viral factors was proposed as a pre-treatment prediction of response to pegylated interferon plus ribavirin for chronic hepatitis C. The IL28B polymorphism and mutations of HCV were significant pre-treatment predictors of response to PEG-IFN/RBV. The decision model, including these host and viral factors may support selection of optimum treatment strategy for individual patients. In [23], Data Mining techniques were applied to reveal complex interactions of the risk factors and clinical feature profiling associated with the staging of Non-hepatitis B virus/non-hepatitis C virus related hepatocellular carcinoma. Data mining disclosed complex interactions of the risk factors and clinical feature profiling associated with the staging of NBNC-HCC.

Also, data mining techniques were used for building a simple and readily available factors model for the identification of patients at high risk of hepatocellular carcinoma. The model allows physicians to identify patients requiring HCC surveillance and those who benefit from IFN therapy to prevent HCC [24].In [25], a decision tree model that includes age, gender, AFP, platelet counts, and GGT was proposed for predicting the probability of response to therapy with peg-interferon plus ribavirin and has the potential to support clinical decisions regarding the selection of patients for therapy.

2.1 Liver fibrosis stages:

The medical attributes that provide information to triggers a certain degree of liver fibrosis were presented in [26]. These attributes include the stiffness indicators from Fibroscan, standard hematological and biochemical exams. There are five possible fibrosis categories. These categories and their number of representatives are defined as:

- F0 (no fibrosis) – 29 examples;
- F1 (portal fibrosis without septa) – 227 examples;
- F2 (portal fibrosis and few septa) – 164 examples;
- F3 (numerous septa without cirrhosis) – 87 examples;
- F4 (cirrhosis) – 215 examples.

2.2Principal Component Analysis:

PCA is an attributes selection technique which is considered as one of the most prevalent and useful statistical method for dimensionality reduction. This method transforms the original data in to new dimensions [1]. The new variables are formed by taking linear combinations of the original variables of the form:

$$H_1 = b_1{}' = b_{11}K_1 + b_{12}K_2 + \cdots + b_{1m}K_m \qquad (1)$$

$$H_2 = b_2{}' = b_{21}K_1 + b_{22}K_2 + \cdots + b_{2m}K_m \qquad (2)$$

$$H_p = b_p{}' = b_{p1}K_1 + b_{p2}K_2 + \cdots + b_{pm}K_m \qquad (3)$$

In matrix style, we can write H = B.K, where K are known as the loading parameters. The new axes are attuned such that they are orthogonal to one another with utmost expand of information.

$$var(H_i) = b_i{}' \sum b_i, \mathrm{I} = 1, 2 \ldots \mathrm{p} \qquad (4)$$

$$cov(H_i, H_j) = b_i{}' \sum b_j \;, \mathrm{I} = 1, 2 \ldots \mathrm{p} \qquad (5)$$

K_1 is the first principal component holding the prime variance.

As the direct computation of matrix B is not achievable. So, in feature transformation, the first step is to ascertain the covariance matrix U which can be expressed as:

$$U_{m\times n} = \frac{1}{m-1}\left[\sum_{i=1}^{m}(K_i - \overline{K})'.(K_i - \overline{K})\right], \ \ where\ \overline{K} = \left(\frac{1}{m}\right)\sum_{i=1}^{m} X_i \qquad (6)$$

The next step is to determine the eigen values for the covariance matrix U. Eventually, a linear transformation is defined by n eigen vectors match up to n eigen values from a m-dimensional space to n-dimensional space ($n<m$). Principal axes are also referred to as eigen vectors $E_1, E_2, \dots, E_m$ correspond to eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$. Generally, the first few principal components hold most of the information. Analysis of variances proportion represents the total number of principal components that should be retained from the dataset.

Algorithm 1: PCA

Step1: Recover basis: Calculate $XX^T = \sum_{i=1}^{t} x_i x_i^T$ and let U = eigenvectors of XX^T corresponding to the top d eignvalues.
Step2: Encode training data: $Y = U^T$ where Y is a $d \times t$ matrix of encoding of the original data.
Step3: Reconstruct training data: $X = UY = UU^T X$
Step4: Encode test example: $y = U^T x$ where y is a $d - dimensionalencodingofx$.
Step5: Reconstruct test example: $\hat{x} = Uy = UU^T x$.

SVM is a supervised learning method that can be applied to classification or regression. Classification is achieved by a linear or nonlinear separating surface in the input space of the dataset. SVM delivers state-of-the-art performance in real world applications such as text categorization, hand-written character recognition, image classification, bio-sequences analysis. It is established as one of the standard tools for Machine Learning and data mining. SVM employs kernel to map the input data into some much higher dimensional feature space implicitly in which data becomes linear separable. The linear decision boundary is drawn in a manner that the margin, minimum distance between training examples to the boundary, is maximized. The SVM approach seeks to find the optimal separating hyper plane between classes by focusing on the training cases that are placed at the edge of the class descriptors. These training cases are called support vectors. Training cases other than support vectors are discarded. This way, not only is an optimal hyperplane fitted, but also less training samples are effectively used; thus high classification accuracy is achieved with small training sets [27].

Given a training dataset with n samples $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ where x_i is a feature vector in a v-dimensional feature space and with labels $y_i \in -1,1$ belonging to either of two linearly separable classes C1 and C2. Geometrically, the SVM modeling algorithm finds an optimal hyperplane with the maximal margin to separate two classes, which requires solving the optimization problem:

$$\text{Max} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{j=1}^{n} \alpha_i \alpha_j \; y_i y_j . K(x_i, x_j) \tag{7}$$

$$\text{S.T.} \sum_{i=1}^{n} \alpha_i \; y_i \, , 0 \leq \alpha_i \leq C \tag{8}$$

Where α_i is the weight assigned to training samplex_i.

If > 0, x_i is called support vector C is a regulation parameter used to trade-off the training accuracy and the model complexity K is a kernel function, which is used to measure the similarity between two samples. The SVM algorithm provides a choice of four kernel types: Linear, Polynomial, Radial Basis Function, and Sigmoid. These kernel functions are defined as:

$$\text{Linear Kernel Function: } K(x_i, x_j) = x^T{}_i x_j \tag{9}$$

$$\text{Polynomial Kernel Function: } K(x_i, x_j) = (\gamma x^T{}_i x_j + r)^d, \gamma > 0 \tag{10}$$

$$\text{Sigmoid Kernel Function: } K(x_i, x_j) = \tan\,(\gamma x^T{}_i x_j + r) \tag{11}$$

Algorithm 2: SVM:

Step1: Choose a kernel function

Step2: Choose a value for C.

Step3: Solve the quadratic programming problem.

Step4: Construct the discriminate function from the support vector.

3. Proposed Method:

The proposed classification system for HCV diagnosis and treatment is composed of three main phases including the data pre-processing, features extraction and classification phase. These phases are described in algorithm- 3, where, its conceptual scheme is showed in Fig.1

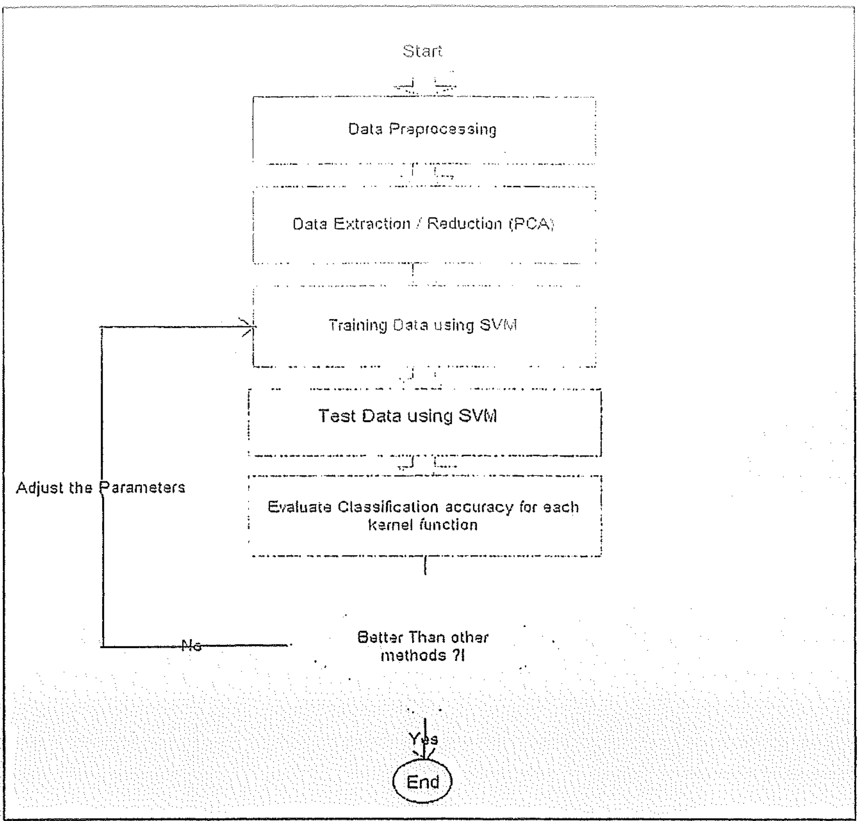

Fig. 1: Flow chart for our proposed method

Algorithm 3: Proposed System:

Phase 1: Data Pre-processing: The aim of this phase to clear data from noise and complete missing data using Local mean algorithm 4.

Phase 2: Data Extraction: The purpose of this phase is to extract the most effective HCV features vector using PCA algorithm 5.

Phase 3: Data Classification. The aim of this phase is to classify the HCV patient into one of 2 classes (either Live or Die) based on his level of fibrosis and tests results. SVM with different kernel functions algorithm 2 was used for this phase.

Algorithm 4: Local Mean Method:

Step 1: Determine the number of cells that we will calculate the average for (n).

Step 2: Given an empty cell that has an initial value (U = 0).

Step 3: Calculate the average of its n previous cells using equation (12).

$$U_t = \frac{U_{n-1} + U_{n-2} \dots U_0}{n} \qquad (12)$$

Step 4: Return number in the empty cell.

4. Experimental Results:

4.1 Data Sets:

The proposed system is implemented and evaluated using two different data sets, one is the benchmark HCV data set from UCI repository of machine learning databases and the other is a real data set from Egyptian National Liver Institute. The UCI dataset contains 154 HCV patients record and has 22 attributes table 2. These attributes contains 12 binary and 10 attributes with discrete values; there are 32 cases out of 154 that die due to HCV. The second National Institute of liver dataset Contains 101 records and has 27 attributes(7 binary and 20 with discrete values) table 3.

Table 2: The set of attributes for UCI data set.

Attribute No	Variable	Values
1	Fibroses Type	F0, F1, F2, F3, F4
2	Class	Die, Live
3	Age	10, 20, 30, 40, 50, 60, 70, 80
4	Gender	Male, Female
5	STEROID	No, Yes
6	ANTIVIRALS	No, Yes
7	FATIQUE	No, Yes
8	MALAISE	No, Yes
9	ANOREXIA	No, Yes
10	LIVER BIG	No, Yes
11	LIVER FIRM	No, Yes
12	PLEEN PALPABLE	No, Yes
13	SPIDERS	No, Yes
14	ASCITES	No, Yes
15	VARICES	No, Yes
16	BILIRUBIN	0.39, 0.80, 1.20, 2.0, 3.0, 4.0
17	ALK PHOSPHATE	33, 80, 120, 160, 200, 250
18	SGOT	13, 100, 200, 300, 400, 500
19	ALBUMIN	2.1, 3.0, 3.8, 4.5, 5.0, 6.0
20	PROTIME	10, 20, ... , 90
21	HISTOLOGY	No, Yes
22	Weight	67,75,85,90,100,110

Table 3: The set of attributes for National Liver Institute dataset.

Attribute No.	Variables	Values
1	Class	Die, Live
2	Age	20,30,40,50,59
3	Gender	Male, Female
4	Glu-Sc	8, 67, 80, 81 ...,217
5	Creat-Sc	0.4, 0.6, 0.8, ...,1.45
6	Alb-Sc (alkaline phosphatase)	0.6, 1.3, 3.7, 4.5, 5.2 ..., 40
7	AlkPh-Sc	12, 49, 51, 70, 142 ... 701

8	AST-Sc	5, 7, 14, 22, 30, 169 ...175
9	ASTUNL-Sc	12, 30, 40, 50, 133 ...152
10	ALT-Sc	7, 17, 22, 25, 30, 95..., 141
11	ALTUNL-Sc	12, 32, 40, 44, ..., 90
12	T.Bil-Sc	0.3, 0.41, 0.5, 0.6, 0.9..., 10
13	WBC-Sc	3.5, 3.6, 4, 5.6, ..., 7.6
14	Hb-Sc (Hemoglobin)	10.1,11.5, 13, 15.6, ..., 17.3
15	Plt-Sc	11.4, 150, 163, 268 ..., 315
16	Pt.time-Sc	1.2, 1.9, 11, 12.8, 13,...,17.7
17	INR-Sc(blood test)	1, 1.45, 1.58, 1.6, ..., 2
18	%PT-Sc	60, 75, 80, 92, 100 ...,120
19	Preg-Sc	N.A, NEG
20	HbsAg-Sc	POS, NEG
21	Anti-HCV-Sc	POS, NEG
22	ANA-Sc	POS, NEG
23	TSH-Sc	0.3, 0.4, 1.1, 1.5, 2.7 ..., 5.4
24	HCV-RNA-Sc	14, 88, 2018, 76, 54, 87..,850
25	AFP-Sc	0.8, 1.7, 4, 6.7, 20 ..., 150
26	Biopsy-Stage-Sc	1/8, 2/8 , 6/8, ... 11/8
27	Biopsy-Grade	0/6, 1/6, 2/6, ..., 5/6

4.2. Results and Discussion:

In the pre-processing phase, the local mean algorithm 4 used to clear noisy of HCV patients' data records and handle missing data. The PCA algorithm 1 is used to extract features vector. For UCI dataset, the most effective and reduced features in treatment are 6 out of 22 as reported in table 4. For National Institute for liver datasets only 6 out of 27 features were selected and reported in table5.The SVM with different kernel function is used to classify HCV patients into two classes Die or Live. For both data sets 75% of records used in the training where 25% used for testing the proposed system.

Table 4: Extracted features from PCA Algorithm for UCI dataset

Selected Variables	Variable Description
Age	The smallest value of the age is 7 and the largest is 78
Bilirubrin	It is the yellow breakdown product of normal hemecatabolism which is found in hemoglobin, a principal component of red blood cells.
ALK Phosphate	It's an enzyme which exists in the blood; It responsible for removing phosphate groups from many types of molecules.
SGOT	It's an enzeme which works as indicator for liver health.
Albumin	Any protein that is soluble in water and also in moderately concentrated salt solutions.
PROTIME	It's predecessor of thrombin which is produced in the liver.
Class	Die/ Live

Table 5: Extracted attributes from PCA Algorithm for National Institute Data set

Selected Variables	Variable Description
Age	The smallest value of the age is 20 and the largest is 59
INR-Sc	This is the international normalized ratio which measures the extrinsic pathway of coagulation. This test is also called "ProTime INR" and "INR PT". They are used to determine the clotting tendency of blood, in the measure of warfarin dosage, liver damage, and vitamin K status.
AST-Sc	Aspartate transaminase, is usually used to detect liver damage. It is often ordered in conjunction with another liver enzyme, alanine aminotransferase (ALT), or as part of a liver panel to screen for and/or help diagnose liver disorders.
Alb-Sc	Any protein that is soluble in water and also in moderately concentrated salt solutions.
AlkPh-Sc	It's an enzyme which exists in the blood; It responsible for removing phosphate groups from many types of molecules.
Pt-Time-Sc	Partial Thromboplastin time, is a performance indicator measuring the efficacy of both the "intrinsic" and the common coagulation pathways and determine the clotting tendency of blood.
Class	Die/ Live

The classification results are recorded at each iteration, and the average error for each kernel function is calculated over 79 independent experiments.Fig.2,visualizes the classification error of each kernel function for UCI dataset. As shown in fig. 2 the Sigmoid and RBF kernel function have minimum errors.

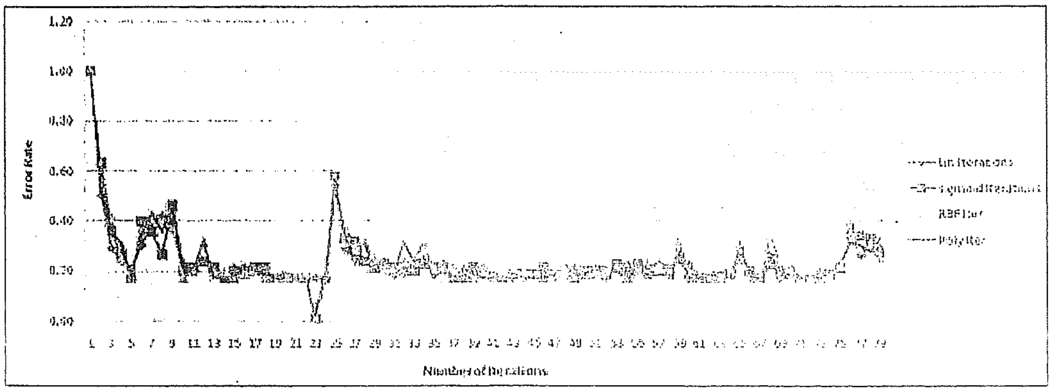

Fig. 2: Classification Error of each kernel for UCI dataset

For National Institute of liver data set the classification error is averaged and visualized in fig. 3. As shown in fig. 3 Polynomial kernel function gives the minimum error which is 0.118, followed by RBF, Sigmoid kernels. Where the error for linear kernel was the worst with value 0.127.

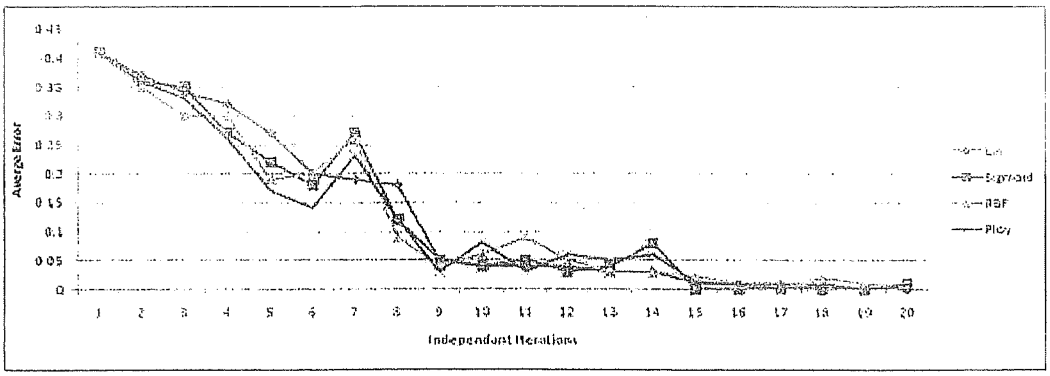

Fig. 3: Classification Error of each kernel for National Institute of liver data set

4.3. Performance Evaluation:

In order to evaluate the performance of the proposed system, the classification accuracy for each kernel function is calculated using equation 12.

$$\text{Accuracy} = \frac{\textit{Number of True Positives} + \textit{Number of True Negatives}}{\textit{True Posotives} + \textit{True Negatives} + \textit{False Positive} + \textit{False Negatives}} \qquad (12)$$

The classification accuracy for both data sets are calculated and visualized in fig. 4 and fig.5. Fig. 4 shows the classification accuracy for UCI dataset, as shown in Fig. 4, the highest classification accuracy was obtained from Sigmoid and RBF kernels which is 0.9934 followed by linear then Polynomial kernel function. Fig. 5 shows the visualized accuracy of each kernel functions National Institute of liver data set. As shown in fig. 5 the Sigmoid kernel function has the highest accuracy which .999, followed by polynomial and linear kernels. RBF kernel has the lowest accuracy.

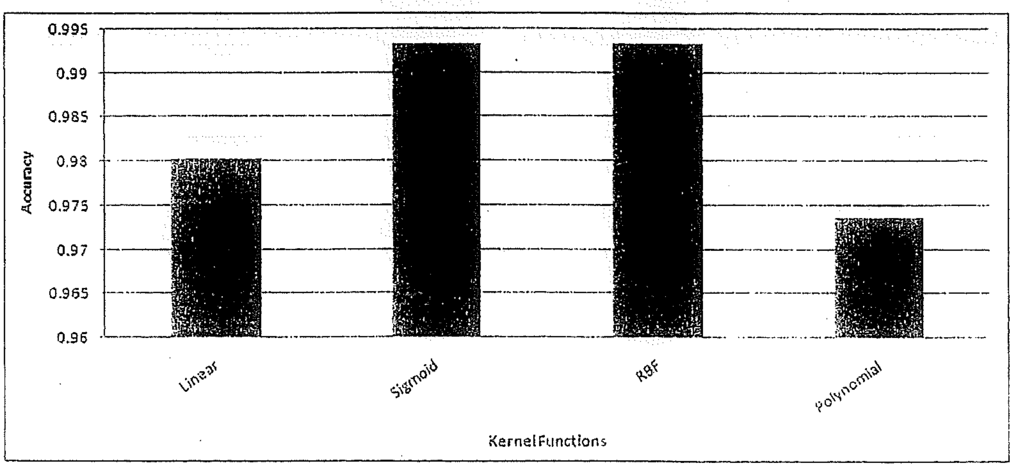

Fig. 4: Classification Accuracy of each Kernel Function for UCI dataset

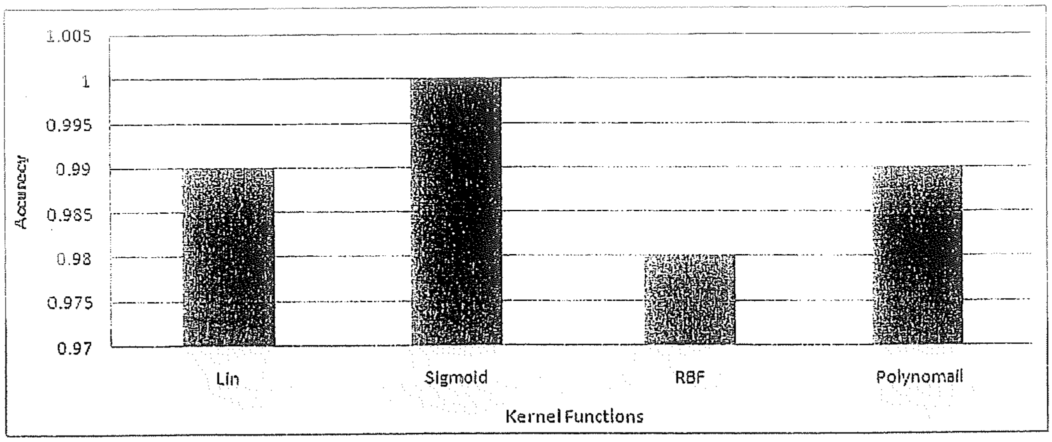

Fig. 5: The accuracy of each kernel functions National Institute of liver data set

The proposed algorithm is compared with 4 different classifier algorithms including ANN [1], Cooperative Co-Evolutionary Algorithm (CCEA) [26],ANN by using duplex ultrasonogaphy [4] and Hybrid Case Based Reasoning and Particle Swarm Optimization (CBR - PSO) [14]. The comparison results are reported in table 6. As reported in table 6 the proposed system has the highest classification accuracy.

Table 6: Classification Accuracy for different classifier systems:

Used Method	Classification Accuracy (%)
PCA + ANN	89.6
CCEA	65.55
ANN by using duplex ultrasonogaphy	88.3
CBR - PSO	94.58
Proposed algorithm (SVM)	99.34

5. Conclusions and future work:

This paper introduced a classification system for HCV patients that integrate PCA and SVM. It is composed of three main phases include pre-processing, features extraction and classification phase. The PCA algorithm employed to extract the most effective HCV features that support in diagnoses and treatment. The SVM algorithm used to classify HCV patients into two classes Die or Live, different kernel functions were implemented such as Linear, Sigmoid, RBF and Polynomial. The proposed algorithm was implemented on different data sets (UCI repository Data set and National Liver Institute). The obtained results showed that the maximum accuracy obtained from both Sigmoid and RBF kernel function which is 0.9934 in UCI data set and it was 1 for National Liver Institute data set. The proposed system was compared with different classifiers; the comparison results showed that it has the highest classification accuracy. For future work other classification algorithm such as Cooperative Co-Evolutionary algorithm can be used.

6. References:

[1] Tahseen A. Jilani, H.Yasin, M. Mohammad, "PCA-ANN for Classification of Hepatitis-C Patients", International Journal of Computer Applications, Volume 14,No.7, Pages 0975 – 8887, February 2011.

[2] D. Panchal, S.Shah, "An Expert System for Hepatitis B Diagnosis Using Artificial Neural Networks", International Conference & Workshop on Recent Trends in Technology, (TCET) 2012 Proceedings published in International Journal of Computer Applications (IJCA),icwet2012 - Number 11, PP. 34-38,March2012.

[3] P. Vahdani, S. Moayed, Z. Aminzadeh, M. Reza, sh. Gharibzadeh, G. Vahdani, S. Fekri, P. Eftekhari "Using Artificial Neural Network to Predict Cirrhosis in Patients with Chronic Hepatitis B Infection with Seven Routine Laboratory Findings", 9(4),PP271-275,2009.

[4] L.Zhang, Q.Li, Y.Duan, G.Yan, Y.Yang, R.Yang, "Artificial neural network aided non-invasive grading evaluation of hepatic fibrosis by duplex ultrasonography", BMC Medical Informatics and Decision Making,ISSN 1472-6947,June 2012.

[5] B. KARLIK, "Hepatitis Disease Diagnosis Using Back propagation and the Naive Bayes Classifiers", Journal of Science and Technology,Vol 1, No 1,PP.2233 - 0054,2011.

[6] H. Zheng, Q. Shi, F. Lin, D. Xiao, L. Chen, Y. Liu, C. Fan, P. Chen. "A model to predict 3-month mortality risk of acute-on-chronic hepatitis B liver failure using artificial neural network" ,Journal of Viral Hepatitis, Vol. 5, No. 4, PP. 539-55, AUG 2012.

[7] H.Hirose, T.Takayama, Sh. Hozawa, T.Hibi,L. Saito, "Prediction of metabolic syndrome using artificial neural network system based on clinical data including insulin resistance index and serum adiponectin", Official Jornal of the International Aids Socity, PP 1051-1056, November 2011.

[8] H.Ochi, C. Nelson, H. Abe, Y. Hayashida, T. Uchiyama, N. Kamatani, Y. Nakamura, K. Chayama, "Toward the Establishment of a Prediction System for the Personalized Treatment of Chronic Hepatitis C", The Journal of Infectious Diseases,Major article, 205 (2),PP.204-210,2012.

[9] N.Nakayama, M. Oketani, Y. Kawamura, M. Inao, S. Nagoshi, K.Fujiwara, H. Tsubouchi, S. Mochida, "Algorithm to determine the outcome of patients with acute liver failure: a data-mining analysis using decision trees", Journal of Gastroenterology, Volume 47, Number 6, PP.664-677,2012.

[10] Y. Kawamura, Sh. Takasaki, M. Mizokami, "Using decision tree learning to predict the responsiveness of hepatitis C patients to drug treatment", Journal of Gastroenterology, Issue: 6, Volume 2, PP. 98–102, 2012.

[11] N.Hiramatsu, M. Kurosaki, N. Sakamoto, M. Iwasaki, M. Sakamoto, Y. Suzuki, F.Sugauchi, A.Tamori, S.Kakinnuma, K. Matsuura, "Pretreatment prediction of anemia progression by pegylated interferon alpha-2b plus ribavirin combination therapy in chronic hepatitis C infection: decision-tree analysis",Journal of Gastroenterology, Issue: 9,Volume 46, Number 9,PP.1111-1119,2011.

[12] P.Srivastava, A.Srivastava, R.Sirohi, "Soft Computing Tools and Classification Criterion for Hepatitis B ", International Journal of Research and Reviews in Soft and Intelligent Computing (IJRRSIC) Vol. 2, No. 2, ISSN: 2046-6412,June 2012.

[13] A. TSAKONA, K. PASCHALI, D. TSOLIS, G. SKAPETIS, "HEPATIC LIVER DISEASES – METHODS FOR DIAGNOSIS AND MEDICAL INFORMATICS FOR TREATMENT SUPPORT", JOURNAL OF MEDICAL INFORMATICS & TECHNOLOGIES, Vol. 17/2011, ISSN 1642-6037,2011.

[14] M. Neshat, M. Sargolzaei, A. Nadjaran, A. Masoumi, "Hepatitis Disease Diagnosis Using Hybrid Case Based Reasoning and Particle SwarmOptimization", International Scholarly Research Network ISRN Artificial Intelligence, Volume 2012, 6 pages, May 2012.

[15] D.Çalişir, E. Dogantekin, "A new intelligent hepatitis diagnosis system: PCA–LSSVM", Expert Syst. Appl, Volume 38, Issue 8, PP. 10705–10708, August 2011.

[16] J.Salimi, M. Hossein, K. Mozafari," Hepatitis disease diagnosis using a novel hybrid method based on support vector machine and simulated annealing (SVM-SA)", Comput Methods Programs Biomed PY,SN - 18727565, October 2011.

[17] E.Pourbasheer, S.Riahi, M. Reza, P.Norouzi, "QSAR study of C allosteric binding site of HCV NS5B polymerase inhibitors by support vector machine", Molecular Diversity, Issue: 3, Volume: 15, PP. 645-653, 2011.

[18] L. Weil, "Prediction of Hepatitis C Virus Non-Structural Proteins 5B Polymerase Inhibitors Using Machine Learning Methods",ActaPhysico-ChimicaSinica,Vol.27, Issue(06), PP. 1407-1416, 2011.

[19] F. Gellad, MPH, J. Muir, MHS, G. McHutchison, Sievert, I. Sharara, A. Brown, R. Flisiak, M. Jacobson, D. Kershenobich, P. Manns, A. Schulman, D. Reed, "Cost-Effectiveness of Truncated Therapy for HepatitisC Based on Rapid Virologic Response",Jornal of Viral Hepatitis,Vol. 15, No. 5,PP.1098-3015,August 2012.

[20] Lara, J, "Host-specific HCV evolution and response to the combined interferon and ribavirin therapy", Jornal of Virology, September 2012, volume 86, issue 18,PP.8211-8224.

[21] E.Scaglia, G.Sockalingum, J. Schmitt, C.Gobinet, N. Schneider, M.Manfait, "Noninvasive assessment of hepatic fibrosis in patients with chronic hepatitis C using serum Fourier transform infrared spectroscopy", Analytical and Bioanalytical Chemistry, Issue: 9, Volume: 401,PP.2919:2925,2011.

[22] M. Kurosaki, Y. Tanaka, N. Nishida, N. Sakamoto, N. Enomoto, M. Honda, M. Sugiyama, K. Matsuura, F. Sugauchi, Y. Asahina, M. Nakagawa, M. Watanabe, M. Sakamoto, Sh. Maekawa, A. Sakai, Sh. Kaneko, K. Ito, N. Masaki, K. Tokunaga, N. Izumi, M. Mizokami, "Pre-treatment prediction of response to pegylated-interferon plus ribavirin for chronic hepatitisC using genetic polymorphism in IL28B and viral factors", BMC Infectious Diseases, PP. 439–448, March 2011.

[23] T. Kawaguchi, T. Kakuma, H. Yatsuhashi, H. Watanabe, H. Saitsu, K. Nakao, A. Taketomi, S. Ohta, A. Tabaru, K. Takenaka, T. Mizuta, K. Nagata, Y. Komorizono, K. Fukuizumi, M. Seike, Sh. Matsumoto, T. Maeshiro, H. Tsubouchi, T. Muro, O. Inoue, M. Akahoshi, M. Sata, "Data mining reveals complex interactions of risk factors and clinical feature profiling associated with the staging of non-hepatitis B virus/non-hepatitis C virus-related hepatocellular carcinoma", Hepatology Research , Volume 41, Issue 6, PP. 564–571, June 2011.

[24] M.Kurosaki, N. Hiramatsu, M. Sakamoto, Y. SuzukiM. Iwasaki, A. Tamori, K. Matsuura, S. Kakinuma, F. Sugauchi, N. Sakamoto, M. Nakagawa, N. Izumi, "Data mining model using simple and readilyavailable factors could identify patients at high risk for hepatocellular carcinoma in chronic hepatitis C", Journal of Hepatology, PP. 602–608, March 2012.

[25] M. Kurosaki, N. Sakamoto, M. Iwasaki, M. Sakamoto, Y. Suzuki, N.Hiramatsu, F.Sugauchi, H.Yatsuhashi, N. Izumi, "Pretreatment prediction of response to peginterferon

plus ribavirin therapy in genotype 1 chronic hepatitis C using data mining analysis", Journal of Gastroenterology, Volume 46, Number 3, PP. 401-409,2011.

[26] C.Stoean, R.Stoean, M.Lupsor, H.Stefanescu, R.Badea, "Feature selection for a cooperative coevolutionary classifier in liverfibrosis diagnosis", Journal of Elsevier, PP. 238–246,16 February2011.

[27] C.J. Burges," A tutorial on support vector machines for pattern recognition", Data Mining and Knowledge Discovery, vol.2, No.2, pp.121-167,1998

[28] Z. Jiang, K. Yamauchi, K. Yoshioka, et al., "Support vector machine-basedfeature selection for classification of liver fibrosis grade in chronic hepatitis C",Journal of Medical Systems,Volume: 30, Issue: 5, PP.389-394, 2006.

A Comparative Study for Approximation Techniques to Nonlinear Optimization Problems

[1] Gamal Eldin A. Mohamed , [2] Mohamed H. Gadallah, [3] Ramadan Z. el Dean

ABSTRACT

Constrained nonlinear programming problems often arise in many engineering applications and much of today's engineering analysis consists of running complex computer codes. The use of statistical techniques to build approximations of expensive computer analysis codes pervades much of today's engineering design. These approximations are used to replace the actual expensive computer analyses. In this paper, we use Response Surface methodology (RSM), and Kriging Method. Both methods are applied to thirty widely used different classes' of single objective optimization problems. We compared the results of both response surface methodology and Kriging model with the solutions of the original. Generally speaking, the Kriging method is able to return almost the same solution as the original model optimum for majority of problems. Response Surface Methodology comes next to Kriging is finding the optimum to the original model.

Keywords: Approximation Techniques, Response Surface Methodology, Kriging Model.

1. Introduction

Current engineering analyses rely heavily on running expensive and complex computer codes. Despite the steady and continuing growth of computing power and speed, the complexity of these codes maintains pace with computing advances. Approximation techniques may be applied to build computationally inexpensive models. Response-Surface Models are frequently utilized to construct surrogate approximations; however, they may be inefficient for systems having large number of design variables. The primary focus of this work is a thorough research into the approximation techniques for solving nonlinear single objective problems unconstraint and constraints equality and inequality.

[1] Studied parameter identification techniques for expensive-to-simulate models and presented a new strategy to address the optimization problem with limited budget. Based on Kriging, this approach computes an approximation of the probability distribution for the optimal parameter vector and selects the next simulation to be conducted so as to optimally reduce the entropy of this distribution. A continuous-time state-space model is used to illustrate the method.

[2] Compared the global accuracy of different strategies to build response surfaces by varying sampling methods and modeling techniques. Aerodynamic shape optimization is employed as an application. For comparison, a robust strategy using a new efficient initialization technique followed by a gradient optimization is used. First, a study of different sampling methods

1,3- Department of Operations Research, Institute of Statistical Studies & Research, Cairo University , Cairo. Egypt
2- Faculty of Engineering, Cairo University, Cairo, Egypt.

proves that including 'a posteriori' information on the function to sample distribution can improve accuracy over classical space filling methods like Latin Hypercube Sampling. Second, Kriging and gradient enhanced Kriging are compared on two to six dimensional test cases.
[3] Demonstrated the time efficiency and accuracy of ordinary Kriging through the use of fast matrix vector products combined with iterative methods. Methods based on the fast Multiple methods and nearest neighbor searching techniques are used for implementation of the fast matrix-vector products.
[4] Studied optimization of nonlinear programming problems in which the objective functions and / or some constraints are represented by noisy implicit black box functions. The black box modules are substituted by meta-models based on Kriging interpolation that assumes that the errors are not independent but function of the independent variables. A Kriging meta-model uses a non-Euclidean measure of the distance that avoids sensitivity to the units of measure. It includes adjustable parameters that weight the importance of each variable getting a good model representation. It allows calculating errors that can be used to establish stopping criteria and provide a solid base to deal with 'possible infeasibility' due to inaccuracies in the representation of objective function and constraints. The algorithm continues with a refining stage and successive bound contraction in the domain of independent variables with or without Kriging recalibration until an acceptable accuracy is achieved.

There are many types of approximations used to solve many types of nonlinear optimization problems, such as Response Surface Methodology (RSM), Kriging Methodology, Surrogate Approximation, Taylor series expansion, Non-Uniform Rational B-spline (NURBS), ...etc. These approximations are then used in place of the actual analysis codes, offering the following benefits:

- They yield insight into the relationship between output responses, y and input design variables, **x**.
- They provide fast analysis tools for optimization and design space exploration since the inexpensive-to-run approximations are used in lieu of the expensive-to run computer analyses.
- They facilitate the integration of discipline dependent analysis codes.

This paper proposes two approximation techniques, namely Response Surface Methodology (RSM) and Kriging to engineering optimization problems. Both methods are applied to thirty widely used classes' of single objective optimization problems cited from literature.
The reset of the paper is organized as follow: in section 2, Classification of Approximations used in this search, in section 3, tested problems, in section 4, result analysis, in section 5, conclusions.

2. Classification of Approximations used in this search

2.1 Response Surface Methodology [RSM]

The Response Surface Methodology is a type of optimization that applies an approximation technique to the objective and other functions of an optimization problem. Response Surface Methodology (RSM) is not an algorithm, it is a set of tools grouped together to analyze a problem. RSM is a collection of statistical and mathematical techniques useful for developing, improving and optimizing processes, [5].
The Response surface model can be represented by (1):

$$y(x) = f(x) + \varepsilon \qquad (1)$$

where y(x) is the unknown function of interest, f(x) is a known polynomial function of x, and ε is random error which is assumed to be normally distributed with mean zero and variance σ^2. The individual errors, ε_i, at each observation are also assumed to be independent and identicall distributed. The polynomial function, f(x), used to approximate y(x) is typically a low order polynomial which is assumed to be either linear, Eqn. (2), or quadratic, Eqn. (3). Given a response y and a vector of independent factors x influencing y, the relationship between y and x is:

$$\hat{y}\,(\mathrm{x}) = \beta_0 + \sum_{i=1}^{k} \beta_i x_i \qquad (2)$$

$$\hat{y}\,(\mathrm{x}) = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_i^2 + \sum_{i=1}^{k} \sum_{j=,i<j}^{k} \beta_{ji}\, x_i x_j \qquad (3)$$

The parameters β_0, β_i, β_{ii}, and β_{ji}, of the polynomials in Eqns. (2) and (3) are determined through least squares regression which minimizes the sum of the squares of the deviations of predicted values $\hat{y}$ (x), from the actual values y(x). The coefficients of Eqns. (2) and (3) are used to fit the model and can be found using least squares regression given by the following Eqn:

$$\beta = [X'X]^{-1}X'y \qquad (4)$$

where **X** is an $n \times p$ the design matrix of sample data points, **X'** is its transpose, and **y** is a column vector an $n \times 1$, containing the values of the response at each sample point. It is to see that **X'X** is a $p \times p$ symmetric matrix and **X'y** is a $p \times 1$ column vector.

2.2 Kriging Approximation

Kriging Models have their origins in mining and geostatistical applications that deal with spatially and inter-correlated data. some statisticians develop Design and Analysis of Computer Experiments (DACE) for deterministic computer-generated data based on the Kriging method. Kriging is a useful method for developing metamodels for product design optimization, [5]. Kriging postulates a combination of a global model plus departures as represented by (5):

$$\mathrm{y}\,(\mathbf{x}) = \mathrm{f}(\mathbf{x}) + \mathrm{Z}(\mathbf{x}) \qquad (5)$$

where y(**x**) is the unknown function of interest, f(**x**) is a known (usually polynomial) function of **x**, and Z(**x**) is the realization of a stochastic process with mean zero, variance σ^2, and non-zero covariance.
The covariance matrix of Z(**x**) is given by Eqn. (6).

$$\mathrm{Cov}\,[Z(\mathbf{x}^i), Z(\mathbf{x}^j)] = \sigma^2\, \mathbf{R}([\mathrm{R}\,(\mathrm{x}^i, \mathrm{x}^j)]) \qquad (6)$$

In Eqn. (6), **R** is the correlation matrix, and R (x^i, x^j) is the correlation function between any two of the n_s sampled data points x^i and x^j. **R** is a ($n_s \times n_s$) symmetric matrix with ones along the diagonal. In this work, we employ a Gaussian correlation function of the form:

$$\mathrm{R}\,(\mathrm{x}^i, \mathrm{x}^j) = \exp\,[-\textstyle\sum_{k=1}^{n_{dv}} \theta_k\, |x_k^i - x_k^j|^2\,] \qquad (7)$$

where n_{dv} is the number of design variables, θ_k are the unknown correlation parameters used to fit the model, and the x_k^i and x_k^j are the k^{th} components of sample points x^i and x^j, and k is the number of design parameters.
Predicted estimates, $\hat{y}(x)$, of the response $y(x)$ at untried values of x are given by:

$$\hat{y} = \hat{\beta} + r^T(x)\, R^{-1}(y - f\hat{\beta}) \qquad (8)$$

where y is the column vector of length n_s which contains the sample values of the response, and f is a column vector of length n_s which is filled with ones when f(x) is taken as a constant. In Eqn. (8), $r^T(x)$ is the correlation vector of length n_s between an untried x and the sampled data points $\{x^1, ..., x^{ns}\}$:

$$r^T(x) = [R(x,x^1), R(x,x^2), ..., R(x,x^{ns})]^T \qquad (9)$$

The mean parameter β is evaluated by minimizing the sum of squares of error using the equation:

$$\hat{\beta} = (f^t R^{-1} f)^{-1} f^t R^{-1} y \qquad (10)$$

3. Solution Using RSM and Kriging Methods

Response surface and kriging models in equations (3) and (8) respectively are applied to approximate the original model at certain points in the search space. Solutions are compared for different classes of problems and results are concluded.

3.1 Tested problems

We tend to classify solutions according to the capability of approximation technique employed. Whenever, solutions devised significantly, we further study the problem and propose solution outlets. Response surface and kriging methods are applied to thirty widely used different classes' of single objective optimization problems.
Summary of problems tested.

Problem- 1 (Levy No. 5) (Parsopoulos & Vrahatis, 2002)

$$\text{Min } f(x) = \sum_{i=1}^{5}[i\cos((i-1)x_1+i)] \quad \sum_{j=1}^{5}[j\cos((j+1)x_2+j)] + (x_1+1.42513)^2 + (x_2+0.80032)^2$$

$$x \in [10,-10]$$

The global solution is $x^* = (-1.3068, -1.4248)$ $\quad f(x^*) = -176.1375$

Problem- 2 (Levy No. 3) (Parsopoulos & Vrahatis, 2002)

$$\text{Min } \quad f(x) = \sum_{i=1}^{5}[i\cos((i-1)x_1+i)] \quad \sum_{j=1}^{5}[j\cos((j+1)x_2+j)]$$

$$x \in [10,-10]$$

The global solution is $x^* = (-1.3068, -1.4248)$ $\quad f(x^*) = -176.542$

Problem- 3 (LevyNo.8) (Parsopoulos & Vrahatis, 2002)

$$\text{Min } f(x) = \sin^2(\pi y_1) + \sum_{i=1}^{n-1}(y_i-1)^2\ [1+10\sin^2(\pi y_{i+1})] + (y_n-1)^2$$

where $\quad y_i = 1 + (x_i-1)/4\ , \quad i = 1, \dots, n$

$x_i \in [-10, 10]$ for $i = 1, 2, 3$

The global solution is $x^* = (1, 1, 1)$ $f(x^*) = 0$

Problem- 4 Freudenstein-Roth function, (Parsopoulos & Vrahatis, 2002)

$$\text{Min} \quad f(x) = [-13 + x_1 + ((5 - x_2)\, x_2 - 2)\, x_2]^2 + [-29 + x_1 + ((x_2 + 1)\, x_2 - 14)\, x_2]^2$$

$x_i \in [-5,5]$ for $i = 1, 2$

The global solution is $x^* = (5, 4)$ $f(x^*) = 0$

Problem- 5 Goldstein-Price function, (Wang, 2003)

$$\text{Min } f(x) = [1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)]\,[30 + (2x_1 - 3x_2)^2 \quad (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$$

$x \in [-2,2]$

The global solution is $x^* = (0, -1)$ $f(x^*) = 3$

Problem - 6 Six-Hump Camel Back Branin , (Wang, 2003)

$$\text{Min } f(\mathbf{x}) = 4x_1^2 - 2.1x_1^4 + 1/3\, x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$$

$6 \le x_1 \le 2.4, \quad -0.8 \le x_2 \le 1.2$

The global solution is $x^* = (0.089, 0.713)$ $f(\mathbf{x}^*) = -1.031628$

Problem- 7 Branin function, (Villemonteix, et al., 2008)

$$\text{Min} \quad f(\mathbf{x}) = (x_2 - 5.1/4\pi^2 x_1^2 + 5/\pi\, x_1 - 6)^2 + 10(1 - 1/(8\pi))\cos(x_1) + 10 + 0.5x_1$$

$-5 \le x_1 \le 10, \quad 0 \le x_2 \le 15$

The global solution is $x^* = (-3.2, 12.3)$ $f(\mathbf{x}^*) = -1.17$

Problem- 8 The Brain-Hoo function, (Wang, 2003)

$$\text{Min } f(\mathbf{x}) = (x_2 - 5/4\pi^2 x_1^2 + 5/\pi\, x_1 - 6)^2 + 10(1 - 1/(8\pi))\cos(x1) + 10$$

$-5 \le x_1 \le 10, \quad 0 \le x_2 \le 15$

The global solution is $x^* = \{(-\pi, 12.3), (-\pi, 2.25), (-3\pi, 2.25)\}$

$f(\mathbf{x}^*) = -1.0316$

Problem -9 (Wang, 1998)

$$\text{Min} \quad f(x) = \sin(x) + \sin(2x/3)$$

$3.1 \le x \le 20.4$

The global solution is $x^* = 17.0393$ $f(x^*) = -1.90596$

Problem- 10 Six hump camelback function, (Sasena-2002)

$$\text{Min} \quad f(x) = (4 - 2.1x_1^2 + x_2^3/3)\, x_1^2 + x_1x_2 + (-4 + 4x_2^2)\, x_2^2$$

$x \in [-10, 10], \quad i = 1, 2$

The global solution is $x^* = (-0.0898, 0.7127)$ $f(x^*) = -1.0316$

Problem -11 (Wang, 1998)

$$\text{Min} \quad \sin(x) + \sin(10x/3) + \log(x) - 0.84x$$

$2.7 \le x \le 7.5$

The global solution is $x^* = 5.2023$ $f(x^*) = -4.601308$

Problem -12 (Wang, 1998)

Min $100(x_2 - x_1^2)^2 + (1-x_1)^2$

$0 \le x_1 \le 3,\quad 0 \le x_2 \le 3$

The global solution is $x^* = (1, 1)$ $f(x^*) = 0$

Problem -13 Rastrigin function, (Wang 2003)

Min $x_1^2 + x_2^2 - \cos 18x_1 - \cos 18x_2$

$-1 \le x_1 \le 1,\quad -1 \le x_2 \le 1$

The global solution is $x^* = (0, 0)$ $f(x^*) = -2$

Problem -14 (Parsopoulos & Vrahatis, 2002)

Min $f(x) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$

$0 \le x_1 \le 4,\quad 0 \le x_2 \le 3$

The global solution is $x^* = (3, 2)$ $f(x^*) = 0$

Problem -15 (Parsopoulos & Vrahatis, 2002)

Min $f(x) = (9x_1^2 + 2x_2^2 - 11)^2 + (3x_1 + 4x_2^2 - 7)^2$

$0 \le x_1 \le 4,\quad 0 \le x_2 \le 3$

The global solution is $x^* = (1, 1)$ $f(x^*) = 0$

Problem -16 (Parsopoulos & Vrahatis, 2002)

Min $f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$

$-1 \le x_1 \le 1,\quad -1 \le x_2 \le 1\quad -1 \le x_3 \le 1,\quad -1 \le x_4 \le 1$

The global solution is $x^* = (0, 0, 0, 0)$ and $f(x^*) = 0$

Problem- 17 (Sasena, 2002)

Min $f(x) = -\sin(x) - \exp(x/100) + 10$

$x \in [0,10]$

The global solution is $x^*=7.8648$ $f(x^*) =7.9182$

Problem-18 (Zhang, et al. 2004)

Min f(x) = exp (-x)*sin (2*pi*x);

$x \in [-1.5,1]$

The global solution is $x^* = -1.2959$ $f(x^*) = -3.5034$

Problem- 19 Mystery function, (Sasena, 2002)

Min $f = 2 + 0.01(x_2 - x_1^2)^2 + (1 - x_1)^2 + 2(2 - x_2)^2 + 7 \sin 0.5x_1 \sin 0.7x_1x_2$

$x_i \in [0, 5]\quad \forall i = 1, 2$

The global solution is $x^* = (2.5044, 2.5778)$ $f(x^*) = -1.4565$

Problem- 20 (Wang, 1998)

Min $-0.0201\, x_1^4 x_2 x_3^2/10^7$

Subject to $x_1^2 x_2 \le 675$

$x_1^2 \; x_3^2 \le 4190000$

$0 \le x_1 \le 36$, $0 \le x_2 \le 5$, $0 \le x_3 \le 125$

The global solution is $x^* = (16.51, 2.477, 124)$ $f(x^*) = -5.6848$

Problem- 21 (Wang, 1998)

Min $\log(1 + x_1^2) - x_2$

Subject to $(1 + x_1^2)^2 + x_2^2 - 4 = 0$

$x_1=[-1;1]$ $x_2=[0;2]$

The global solution is $x^* = (0, \sqrt{3})$ $f(x^*) = -3$

Problem -22 (Wang, 1998)

Min $-12x_1 - 7x_2 + x_2^2$

Subject to $-2\, x_1^4 + 2 - x_2 = 0$

$x_1=[0,2]$ $x_2=[0,3]$

The global solution is $x^* = (0.71751, 1.470)$ $f(x^*) = -16.73889$

Problem -23 (Zhang et al., 2000)

Min $f(x) = (x_1 - 4.0)^3 + (x_1 - 3.0)^4 + (x_2 - 5.0)^2 + 10.0$

Subject to $g(x) = -x_1 - x_2 + 6.45 \le 0$

$1 \le x1 \le 10$, $1 \le x2 \le 10$

The global solution is $x^* = (1.21280, 5.23742)$ $f(x^*) = -1.39378$

Problem -24 Gomez test function, (Sasena, 2002)

Min $f(x) = (4-2.1\, x_1^2 + (1/3)\, x_1^4)\, x_1^2 + x_1 x_2 + (-4+4\, x_2^2)\; x_2^2$

Subject to $g(x)$: $-\sin(4\pi x_1) + 2(\sin 2\pi x_2)^2 \le 0$

$-1 <= x_i <= 1$, $i=1, 2$.

The global solution is $x^* = (0.1093, -0.6234)$ $f(x^*) = -0.9711$

Problem -25 (Sasena-2002)

Min $f(x) = -(x_1 - 1)^2 - (x_2 - 0.5)^2$

Subject to $g1(x)$: $(x_1 - 3)^2 + (x_2 + 2)^2\, e^{-x_2^7} - 12 \le 0$

$g2(x)$: $10x_1 + x_2 - 7 \le 0$

$g3(x)$: $(x_1 - 0.5)^2 + (x_2 - 0.5)^2 - 0.2 \le 0$

$xi \in [0, 1]\ \forall i$

The global solution is $x^* = (0.2017, 0.8332)$ $f(x^*) = -0.7483$

Problem -26 Three-hump camelback function, (Sasena-2002)

Min $f(x) = 2x_1^2 - 1.05\, x_1^4 + 1/6\, x_1^6 - x_1 x_2 + x_2^2$

Subject to $g(x)$: $(3 - x_1)^2 + (1 - x_2)^2 - 3 \leq 0$

$x_1 \in [-3, 3]$, $x_2 \in [-1.5, 1.5]$

The global solution is $x^* = (1.7476, 0.8738)$ $f(x^*) = 0.2986$

Problem- 27 Constrained mystery function. (Sasena, 2002)

Min $f(x) = 2 + 0.01(x_2 - x_1^2)^2 + (1 - x_1)^2 + 2(2 - x_2)^2 + 7 \sin 0.5x_1 \sin 0.7x_1x_2$

Subject to $g(x)$: $-\sin(x_1 - x_2 - \pi/8) \leq 0$

$x_i \in [0, 5]\ \forall i$

The global solution is $x^* = (2.7450, 2.3523)$ $f(x^*) = -1.1743$

Problem- 28 Reverse constrained mystery function. (Sasena-2002)

Min $f(x) = -\sin(x_1 - x_2 - \pi/8)$

Subject to $g(x)$: $-1 + 0.01(x_2 - x_1^2)^2 + (1 - x_1)^2 + 2(2 - x_2)^2 + 7\text{-}\sin 0.5x1 \sin 0.7x_1x_2 \leq 0$

$x_i \in [0, 5]\ \forall i$

The global solution is $x^* = (3.0716, 2.0961)$ $f(x^*) = -0.5503$

Problem -29 (Wang, 1998)

Min $x_1^2 - 2.1x_1^4 + x_1^6/3 + x_1x_2 - 4x_2^2 + 4x_2^4$

Subject to $-2x_1^4 + 2 - x_2 \geq 0$

$(x_2 + 2)^2 - x_1 - x_1^2 \geq 0$

$(x_1 + 1) + (x_1 - 1)^2 - x_2 - 6 \leq 6$

$-1 \leq x_1 \leq 4$, $-10 \leq x_2 \leq 10$

The global solution is $x^* = (-0.0898420, 0.712656)$ $f(x^*) = -1.031628$.

Problem -30 (Wang, 1998)

Min $\sin(\pi x_1/12) \cos(\pi x_2/16)$

Subject to $4x_1 - 3x_2 = 0$

$0 \leq x_1 \leq 20$; $0 \leq x_2 \leq 20$,

The global solution is $x^* = (9, 12)$ $f(x^*) = -0.5$

3.2 Solutions Strategy

A solutions strategy is developed in this paper as shown in figure 1 the strategy depends on 2 solution procedures: Kriging and Response Surface Modeling. The ideal situation is when both approximations yield identical optimum solutions to other solution techniques. Certain problems yield good approximations using RSM, others yield good approximations using Kriging method and some cannot be solved using either method. Whenever this is the case, we propose several modifications to the problem prior to approximation. Figure 1 gives the flow chart of the solution strategy.

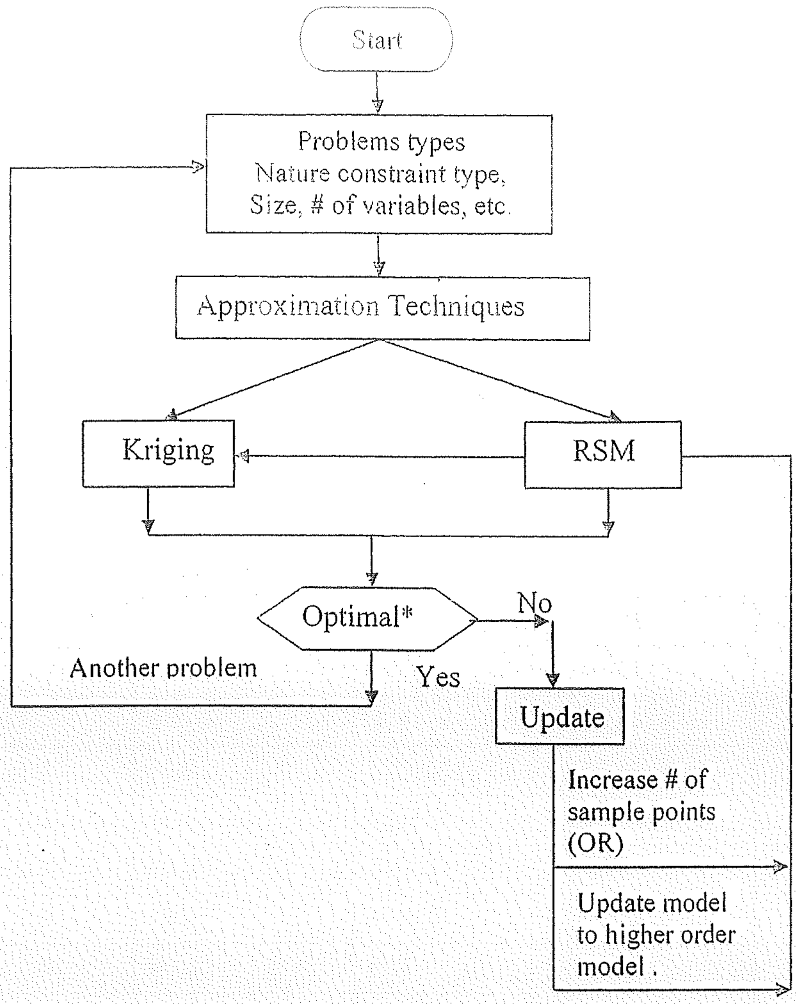

Figure 1: Flowchart of the solution strategy by RSM & Kriging.

Table 1 gives summary of test problems using RSM, Kriging versus reported solutions by reference. A Matlab code is developed as part of this paper for optimization purposes.

Two measures are developed: Δ_1 and Δ_2. They represent deviations from optimum solution using RSM and Kriging methods respectively.

* Means both approximations yield identical optimum solutions to other solution techniques.

Table 1: Summary of test problems using RSM and Kriging

Problem #	MATLAB	Response Surface Methodology	Kriging Methodology	Reference[#]	Δ_1	Δ_2
	f(x*)	f(x*)	f(x*)	f(x*)	(f*-f*res)/f*%	(f*-f*kg)/f*%
01	-176.114+	-9.1095	-164.4267+	−176.138 [6]	94.8%	6.649%
02	-176.517+	-0.109	-165.691+	−176.542 [6]	99.94%	6.15%
03	0.00017■	0.2081■	0.3567■	0.0 ■ [6]	-	-
04	0.00047+	-650	0.0 +	0.0 + [6]	-	0.0%
05	3.0063+	-6894.8604	2.9777+	3.0 + [7]	109%	0.74%
06	-1.03156■	-1.0316■	-1.0298■	−1.0316■ [7]	0.0029%	0.26%
07	-1.18576+	9.4684	-1.161+	−1.17 + [9]	909%	0.769%
08	0.398+	9.9457	0.3997+	0.3979 + [7]	-2400%	-0.45%
09	-1.9059+	-1.2724	-1.8992+	-1.9059+ [8]	33%	0.35%
10	-1.0323■	-0.8859■	-1.0233■	−1.0316■ [9]	14.12	0.8%
11	-4.6013■	-4.3141■	-4.542■	-4.6013■ [8]	6.2%	1.3%
12	0.0+	-312.6274	0.0+	0.0 + [8]	-	0.0%
13	-1.9997+	-0.0656	-2.0+	-2.0 + [7]	97%	0.0%
14	0.0+	1.3748	0.2502+	0.0 + [6]	-	-
15	0.0003+	-0.6096	0.0+	0.0 + [6]	-	0.0%
16	0.0■	0.0■	0.0■	0.0 ■ [6]	0.0%	0.0%
17	7.9182■	7.9047■	7.922■	7.9182 ■[9]	0.17%	-0.0005%
18	-3.5342■	-3.4914■	-3.5149■	-3.5034 ■ [10]	0.34%	-0.32%
19	-1.45645+	1.3518	-1.3778+	-1.4565 +[9]	192%	5.3%
20	-5.62601+	-1.5938	-5.4494+	-5.6848 + [8]	72%	4.0%
21	-2.9952■	-2.9982■	-2.9425■	-3.0 ■ [8]	0.06%	1.92%
22	-16.7304■	-16.723■	-16.413■	-16.739 ■[8]	0.09%	1.9%
23	-.3915++	-1.3915++	-0.75	-1.394 ++[11]	0.16%	46%
24	-0.9709■	-0. 9642■	-0.9683■	-0.971■ [9]	0.71%	0.28%
25	-0.6872■	-0.6872■	-0.6378■	−0.748 ■ [9]	8.1%	14.7%
26	0.2987■	0.2972■	0.2993■	0.2986 ■ [9]	0.47%	-0.23%
27	-1.1662+	2.5959	-1.0420+	−1.1743+ [9]	321%	11.27%
28	-0.57065+	-1.0019	-0.6582+	−0.5503 + [9]	-82%	-19.6%
29	-1.03136■	-1.0351■	-0.9912■	-1.03168■ [8]	0.33%	3.9%
30	-0.5■	-0.4763■	− 0.5■	− 0.5 ■ [8]	4.74%	0.0%

Δ_1 = Difference of optima (RSM) %. Δ_2 = Difference of optima (kriging) %.
■ ≡ RSM, Kriging Agree with Original solution + ≡ Kriging Agree with Original solution
++ ≡ RSM Agree with Original solution

3.3 Classification of problems

We divided the problems into three classes according to:
1- Agreement of RSM, Kriging and Original solution ,in problems (3, 6, 10, 11, 16, 17, 18, 21, 22, 24, 25, 26, 29 and 30)respectively. This means that both RSM and Kriging agree in 14 out of 30 problems.
2- Response Surface Solutions different from the original solutions, in problems (1,2,4,5,7,8,9,12,13,14,15,19,20,27,and28)
3- Kriging solutions different from the original solutions in problem 23.

4. Result analysis

These approximations models represent a cheap and reliable alternative for expensive high fidelity computational models. These methods are applied to thirty different optimization problems cited from literature. Some problems agreed in solutions to both Approximations.

Response Surface Models are applied to approximate the original model; we solved thirty different classes of test problems selected from the literature, and results of Response Surface Models and the original models are compared, we observed that:
- RSM and nonlinear optimization agree in 15 problems out of 30 problems.
- Fifteen problem solutions are different from the original solutions. Results are inferior due to several possible reasons. Features of these functions include Cosine, multiplication and additive, Cosine, and additive functions, Sine and additive function, Sine, multiplication functions with constraints and Polynomial function & exponential functions.

Kriging method is applied to approximate the original model; we solved thirty different test problems cited from literature. Results are compared with optima cited from literature. Generally,
- Twenty-nine problems are satisfied.
- Only one problem returned Results different from the original solution

5. Conclusion

Nonlinear engineering systems can be modeled via approximation techniques such as Response Surface Methodology (RSM) and Kriging techniques. These approximations models represent a cheap and reliable alternative for expensive high fidelity computational models. These methods are applied to thirty different optimization problems cited from literature. Some problems agree in solution to both approximations. Twenty-nine problems out of 30 problems agree in solutions using Kriging method, and fifteen problems out of 30 problems agree in solutions using RSM. The Kriging method is able to return almost the same solution as the original model optimum for majority of problems.

References

[1] Villemonteix, J., Vazquez, E., Walter, E., "Identification of expensive to-simulate parametric models using Kriging and Stepwise Uncertainty Reduction". hal-00252148, version 1 - 13, 2008.

[2] Laurenceau, J., Sagaut, P., "Building Efficient Response Surfaces of Aerodynamic Functions with Kriging and CoKriging", AIAA Journal, Vol. 46, No 2, 2008, pp. 498-507.

[3] Vikas, N. M. C., Duraiswami, R. R. David, M. M.," Efficient Kriging via Fast Matrix-Vector Products", Aerospace Conference, 2008 IEEE, 2008, pp. 1-7.

[4] Caballero, J. A., Grossmann, I. E., "An Algorithm for the Use of Surrogate Models in Modular Flow sheet Optimization", AIChE Journal, Vol. 54, 2008, pp. 2633-2650.

[5] Simpson, T. W., Mauery, T. M., Korte, J. J., "Comparison Of Response Surface And Kriging Models For Multidisciplinary Design Optimization", Paper No. AIAA-98-4755, Vol. 1, 1998, pp. 381-391.

[6] Parsopoulos, K. E., Vrahatis, N.M., "Recent Approaches to Global Optimization Problems through Particle Swarm Optimization", Natural Computing 1: 2002, pp. 235–306.

[7] Wang, G.G., "Adaptive Response Surface Method using Inherited Latin Hypercube Design Points", Transactions of the ASME, Journal of Mechanical Design, Vol. 125, 2003, pp. 210-220.

[8] Wang, H., "Global Optimization of Nonconvex Factorable Programs with Applications to Engineering Design Problems", Master's Thesis, Virginia Polytechnic Institute and State University, 1998.

[9] Sasena, M., Papalambros, P., Goovaerts, P., "Global Optimization of Problems with Disconnected Feasible Regions via Surrogate Modeling", 9th AIAA/ISSMO Symposium on Multi-Disciplinary Analysis and Optimization, Atlanta, Georgia, AIAA-2002-5573, 2002.

[10] Zhang, C., Sheng, Q., June, R.O., "Notes on the Convergence and Applications of Surrogate Optimization", Pomona, CA, USA, 2004, pp. 1–9.

[11] Zhang, J., Margaret, M.W., Chen, W., "Local Approximation of the Efficient Frontier in Robust Design", Transactions of ASME, Journal of Mechanical Design, Vol. 122, 2000, pp. 232-236

The linear Goal Programming problem In Rough Environment

Maaty, M.A[1] and Farahat , F.A.*[2]

Abstract:

Goal Programming (GP) has many significant advances in recent years, particularly in the area of intelligent modeling and solution analysis as it is powerful and effective methodology for the modelling, solution, and analysis of problems having multiple and conflicting objectives. In this paper, goal programming in rough environment is investigated when roughness exists on the system constraints or goal constraints or both. Also the solutions of these kinds of problems are presented.

Key words: Rough set, goal programming, rough goal programming, rough function.

1. Introduction:

Goal programming (GP) is a tool that has been proposed as a model and approach for the analysis of problems involving multiple, conflicting objectives with different measures units. Goal programming (GP), firstly introduced by Charnes and Cooper in the early 1960's [10].This technique has been applied in different areas to support the decision maker such as multiobjective decision making (MODM) problem [14]. For solving goal programming problem (GPP), it is necessary to establish a hierarchy of importance among these goals so that lower priority goals are considered only after higher priority ones are satisfied. Since it is not always possible to achieve every goal according to the decision maker desires, GP attempts to reach a satisfactory solution. So that GP is a good tool to enable the decision maker to realize the balance between the available resources and the desired goals. The objective function of GPP focused on minimizing the deviation between the available resources and the required goals. is one of the main approaches for solving multicriteria decision making problems. The solution depends on the metrics used for the deviations and the weighting method of the different goals. There are two common weighting methods. The first one is the fixed ordering of goals. In practice, this is implemented by searching a lexicographic minimum of the ordered deviation vector [11]. The second one is the use of weights on goals and the minimization of the weighted sum of goal deviations [12].Sometimes, the minimization of the maximum deviation is also used as in [13,15].

This paper is organized as follows: In Section2, the basic concepts of goal programming problems.Insection3, the concepts of rough set. In Section 4, the study

[1,2] Higher technological institute, 10th of Ramadan city, Egypt

addresses the concept of rough goal programming. Section 5, includes conclusion and recommendations for further studies.

2. The Goal Programming Problem:

Goal programming is considered as one of the three main approaches for solving multicriteria decision making problem, the other two approaches are vector optimization problem (VOP) and the Interactive approach. In GP decision makers must initially give priorities for their goals .The GP gives an optimal solution. The GP can deal with the objectives in different measures units. To solve the linear goal programming problem we can use the modified simplex method which is an algebraic procedure. It was developed by Charnes and Cooper. And we can use the graphical method is based on drawing the feasible region by using a given set of constraints (goals) if it was in two dimensions, the optimal solution can be reached by identifying the optimal extreme point according to specified condition and stated priority structures. There are many formulas for GP, but let us use the following form:

$$\min Z(d^-, d^+) = \sum_{i=1}^{m} P_i\,(w_i^- d_i^- + w_i^+ d_i^+).$$

Subject to
$$G_i(x) + d_i^- - d_i^+ = b_i\,, i = 1, \ldots, m \qquad (1)$$
$$X \in S$$
$$d_i^-, d_i^+ \geq 0$$

The objective function seeks to minimize the deviations from the preferred aspiration levels. X is a variable that is under control of the decision maker and has an impact on the problem solution .A deviation variable reflects the underachievement (negative deviation and denoted by d_i^-) or overachievement (positive deviation and denoted by d_i^+) of goal i .A goal programming is composed of m functions in n decision variables and $2m$ deviation variables. Any set of nonnegative X, d_i^-, d_i^+ values that satisfy the system constraints constitute a feasible solution. If $(n + 2k) - k$ of the variables (any of the variables X, d_i^-, d_i^+ are set to zero) and the resulting set of m objectives are solved, the resultant solution is termed as a basic solution. The m variables that are not set to zero are the basic variables while the $n+k$ variables that are set to zero are the non-basic variables. An implementable solution is a feasible solution in which all objectives are satisfied. The goal programming achievement function is an ordered vector of a dimension equal to the number (k) of preemptive priorities within the problem and expresses the level of achievement of each objective set within a given priority. The solution to a given goal programming model is considered an optimal solution if for this solution the corresponding value of objective function is the same or preferred to the value of objective function for any other feasible solution. S, $G_i(x)$ $i = 1, ..., m$ are the system constraints and goal constraints of the problem respectively.

3. Rough set theory (RST):

Rough set theory (RST) is a new mathematical theory introduced by Zdzislaw Pawlak [7, 8, 9, and 10] in the early 1980s to deal with vagueness or uncertainty. RST expresses vagueness by employing a boundary region of a set not by means of membership function [6, 11]. Some elements have the same information, are indiscernible in view of the available information (appear as the same) and cause roughness. The basic concept of RST is the approximation of indefinable sets or concepts through definable sets. Rough sets meaning approximation sets are built on ordinary sets.

Let U be a non-empty finite set of objects, called the universe, and $E \subseteq U \times U$ be an equivalence relation on$U$. The ordered pair $A = (U, E)$ is called an approximation space generated by E on U. The equivalence relation$E$ generates a partition $U/E = \{Y_1, Y_2, ..., Y_m\}$ where $Y_1, Y_2, ..., Y_m$ are the equivalence classes (also called elementary sets) of the approximation space A satisfy $Y_i \subseteq U, Y_i \neq \emptyset, Y_i \cap Y_j = \emptyset \ (i \neq j)$, and $\cup_i^m Y_i = U$. In rough set theory, any subset $X \subseteq U$ is described by the elementary sets of A, and the two sets $X_* = \cup\{Y_i \in U/E \mid Y_i \subseteq X\}$

$$X^* = \cup\{Y_i \in U/E \mid Y_i \cap X \neq \emptyset\}$$

are called the lower and the upper approximations of X ,respectively[1].Therefore $X_* \subseteq X \subseteq X^*$. The difference between the upper and the lower approximations is called the boundary of X and is denoted by $X_{BN} = X^* \sim X_* = \{x \mid x \in X^*, x \notin X_*\}$. The set X is called exact (crisp) in A if $X_{BN} = X^* \sim X_* = \emptyset$, otherwise the set X is inexact (rough) in A. Shortly, a rough set in A is the family of all subsets of U having the same lower and upper approximations. An underlying notion of rough set theory is the indiscernibility of objects. By modeling indiscernibility as an equivalence relation, one can partition a finite universe of objects into pair-wise disjoint subsets. The partition provides a granulated view of the universe. An equivalence class is considered as a whole, instead of many individuals. The empty set, equivalence classes and unions of equivalence classes form a system of definable subsets under indiscernibility. All subsets not in the system are consequently approximated through definable sets. For simplicity we deal with upper and lower approximation as single crisp sets. The lower approximation set and the upper approximation set are deterministic sets. Lower approximation set contains all classes of elements which surely belong to set X, upper approximation set contains all classes of elements which possibly and surely belong to X set where $X_* \subseteq X \subseteq X^*$.

Definition Rough Function:

The rough function $f(x)$ is a function can't be defined or formatted on a crisp mathematical formula but can be defined by two functions $(f_*(x), f^*(x))$ such that $f_*(x) \leq f(x) \leq f^*(x)$ where these functions surrounding it from above and down.

The mathematical programming in rough environment is introduced in [1, 2, 4]. It is classified into three classes on the basis of where roughness is found [1]; 1st Class: mathematical programming problems with rough feasible set and crisp objective function; 2nd Class: mathematical programming problems with crisp feasible set and rough objective function; 3rd Class: mathematical programming problems with rough feasible set and rough objective function [1]. The solutions of the mathematical programming in rough environment are formed into solution sets.

4. The linear Rough goal programming:

Roughness in goal programming may be found in the constraints. So the rough goal programming problem can be classified into three classes the first class deals with roughness in system constraint only, the second class deals with roughness in goal constraints only and the last one deals with roughness in both system and goal constraint.

4.1. 1st class: The Roughness found in The System Constraints only:

A rough goal mathematical programming problem of the 1st class takes the following form where system constraints are a rough set.

$$\min Z(d^-, d^+) = \textstyle\sum_{i=1}^{m} P_i \, (w_i^- d_i^- + w_i^+ d_i^+).$$

Subject to

$$G_i(X) + d_i^- - d_i^+ = b_i\,, \qquad i = 1, \dots, m \qquad (2)$$

$$X \in S,$$

$$d_i^-, d_i^+ \geq 0.$$

Where

$$S_* \subseteq S \subseteq S^*.$$

First, solve the following problem

$$\min Z(d^-, d^+) = \sum_{i=1}^{m} P_i\,(w_i^- d_i^- + w_i^+ d_i^+),$$

Subject to
$$G_i(X) + d_i^- - d_i^+ = b_i\,,\, i = 1, \dots, m, \qquad (2\text{-a})$$
$$X \in S^*,$$
$$d_i^-, d_i^+ \geq 0.$$

Let us denote the optimal solution by

$$\left(X', \overline{d_i^-}, \overline{d_i^+}\right), \qquad i = 1, \dots, m,$$

This solution is surely optimal if $X' \in S_*$ and possibly optimal if $X' \notin S_*$.

Second we will solve the following problem,

If we don't get the surly optimal solution

$$\min Z(d^-, d^+) = \sum_{i=1}^{m} P_i\,(w_i^- d_i^- + w_i^+ d_i^+).$$

Subject to
$$G_i(X) + d_i^- - d_i^+ = b_i\,,\, i = 1, \dots, m \qquad (2\text{-b})$$
$$X \in S_*$$
$$d_i^-, d_i^+ \geq 0$$

Then we get solution $\left(X'', \overline{\overline{d_i^-}}, \overline{\overline{d_i^+}}\right)$

Then the optimum value is between:

$$\begin{pmatrix} w_1^- \overline{d_1^-} + w_1^+ \overline{d_1^+} \\ w_2^- \overline{d_2^-} + w_2^+ \overline{d_2^+} \\ \cdot \\ \cdot \\ \cdot \\ w_m^- \overline{d_m^-} + w_m^+ \overline{d_m^+} \end{pmatrix} \leq \begin{pmatrix} w_1^- d_1^- + w_1^+ d_1^+ \\ w_2^- d_2^- + w_2^+ d_2^+ \\ \cdot \\ \cdot \\ \cdot \\ w_m^- d_m^- + w_1^+ d_1^+ \end{pmatrix} \leq \begin{pmatrix} w_1^- \overline{\overline{d_1^-}} + w_1^+ \overline{\overline{d_1^+}} \\ w_2^- \overline{\overline{d_2^-}} + w_2^+ \overline{\overline{d_2^+}} \\ \cdot \\ \cdot \\ \cdot \\ w_m^- \overline{\overline{d_m^-}} + w_m^+ \overline{\overline{d_m^+}} \end{pmatrix}$$

Example1:

TopAd, a new advertising agency with 10 employees, has received a contract to promote a new product. The agency can advertise by radio and television. It has ability to use 20 or 30 employs. The following table gives the number of people reached by each type of advertisement and the cost and labor requirements.

	Data/min advertisement	
	Radio	*Television*
Exposure	4	8
Cost	8	24
Assigned(case1 use 20 employees)	2	5
Assigned(case2 use 30 employees)	3	5

The contract prohibits TopAd from using more than 5 minutes of radio advertisement on casee1also it prohibits TopAd from using more than 7 minutes of radio advertisement on case 2. Additionally, radio and television advertisements need to reach at least 45 million people. TopAd has a budget goal of $100000 for the project. How many minutes of radio and television advertisement should use?

The linear rough goal programming problem can be written as:

$$\min 2\, d_1^- + d_2^+$$

Subject to

$$4x_1 + 8x_2 + d_1^- - d_1^+ = 45$$
$$8x_1 + 24x_2 + d_2^- - d_2^+ = 100$$
$$X \in S \qquad X = (x_1, x_2)$$
$$x_1, x_2, d_1^-, d_1^+, d_2^-, d_2^+ \geq 0$$

where

$$S_* \subseteq S \subseteq S^* \tag{2-c}$$
$$S^* = \{X \in R^2 | 6x_1 + 10x_2 \leq 60 \, , x_1 \leq 7\}$$
$$S_* = \{X \in R^2 | 4x_1 + 10x_2 \leq 40 \, , x_1 \leq 5\}$$

If $X \in S^*$ we get $x_1 = 6.88$, $x_2 = 1.88$, $d_1^- = 2.5$, $d_1^+ = 0$, $d_2^- = 0$, $d_2^+ = 0$.

And $\bar{z} = 5$,

(6.88,1.88) $\notin S_*$ so it's possibly optimal solution.

If $X \in S_*$ we get $x_1 = 5$, $x_2 = 2$, $d_1^- = 9$, $d_1^+ = 0$, $d_2^- = 0$, $d_2^+ = 0$.

And $\bar{z} = 18$, which is a solution.

4.2. 2nd class: The Roughness exists in The Goal Constraints:

A rough goal mathematical programming problem of the 2nd case takes the following form

$$\min Z(d^-, d^+) = \textstyle\sum_{i=1}^{m} P_i (w_i^- d_i^- + w_i^+ d_i^+).$$

Subject

$$G_i(x) + d_i^- - d_i^+ = b_i \qquad , i = 1, \dots, m$$
$$X \in S \tag{3}$$
$$d_i^-, d_i^+ \geq 0$$

Where

$$G_{i*}(x) \leq G_i(x) \leq G_i^*(x) \quad , i = 1, \dots, m$$

First, solve the following problem,

$$\min Z(d^-, d^+) = \textstyle\sum_{i=1}^{m} P_i (w_i^- d_i^- + w_i^+ d_i^+).$$

Subject to

$$G_i^*(x) + d_i^- - d_i^+ = b_i \, , i = 1, \dots, m \tag{3-a}$$
$$X \in S$$
$$d_i^-, d_i^+ \geq 0$$

The solution obtained

$$\left(\overline{x^*}, \overline{d_i^-}, \overline{d_i^+}\right) \qquad , i = 1, \dots, m$$

It is called surely optimal solution if

$$G_{i*}(\overline{x^*})+,\overline{d_i^-}-\overline{d_i^+}=b_i \quad ,i=1,\dots,m$$

If it is not satisfied the solution called possibly optimal solution then we solve the following problem

$$\min Z(d^-,d^+)=\textstyle\sum_{i=1}^{m} P_i\,(w_i^-d_i^-+w_i^+d_i^+).$$

Subject to

$$\begin{aligned} G_{i*}(x)+d_i^- - d_i^+ &= b_i\ ,i=1,\dots,m \\ X &\in S \\ d_i^-,d_i^+ &\geq 0 \end{aligned} \qquad (3\text{-b})$$

The solution obtained

$$\left(\overline{x_*},\overline{\overline{d_i^-}},\overline{\overline{d_i^+}}\right)\ ,i=1,\dots,m$$

Then the optimum value is between:

$$\begin{pmatrix} w_1^-\overline{d_1^-}+w_1^+\overline{d_1^+} \\ w_2^-\overline{d_2^-}+w_2^+\overline{d_2^+} \\ \cdot \\ \cdot \\ \cdot \\ w_m^-\overline{d_m^-}+w_m^+\overline{d_m^+} \end{pmatrix} \leq \begin{pmatrix} w_1^-d_1^-+w_1^+d_1^+ \\ w_2^-d_2^-.+w_2^+d_2^+ \\ \cdot \\ \cdot \\ \cdot \\ w_m^-d_m^-+w_1^+d_1^+ \end{pmatrix} \leq \begin{pmatrix} w_1^-\overline{\overline{d_1^-}}+w_1^+\overline{\overline{d_1^+}} \\ w_2^-\overline{\overline{d_2^-}}+w_2^+\overline{\overline{d_2^+}} \\ \cdot \\ \cdot \\ \cdot \\ w_m^-\overline{\overline{d_m^-}}+w_m^+\overline{\overline{d_m^+}} \end{pmatrix}$$

Example2: consider the following

$$minz=p_1d_1^+ + p_2d_2^-$$

Subject to

$$\begin{aligned} &G_1(x_1,x_2)\leq 1, \\ &G_2(x_1,x_2)\geq 32, \\ &x_1+x_2\leq G_1(x_1,x_2)\leq 5x_1+3x_2, \\ &2x_1+x_2\leq G_2(x_1,x_2)\leq 8x_1+5x_2, \\ &x_1+4x_2\leq 4, \\ &2x_1+3x_2\leq 6, \\ &x_1,x_2,d_1^-,d_1^+.d_2^-,d_2^+\geq 0\ . \end{aligned} \qquad (3\text{-c})$$

First we will solve:

$$minz=p_1d_1^+ + p_2d_2^-$$

Subject to

$$\begin{aligned} &5x_1+3x_2+d_1^- - d_1^+ = 1, \\ &2x_1+x_2+d_2^- - d_2^+ = 32, \\ &x_1+4x_2\leq 4, \\ &2x_1+3x_2\leq 6, \\ &x_1,x_2,d_1^-,d_1^+,d_2^-,d_2^+\geq 0 \end{aligned} \qquad (3\text{-d})$$

Then we get $x_1=0.2, x_2=0, d_1^-=0, d_1^+=0\ .\ d_2^-=30.6, d_2^+=0, \bar{z}=30.6$.

Second we will solve:

$$minz=p_1d_1^+ + p_2d_2^-$$

Subject to

$$\begin{aligned} &x_1+x_2+d_1^- - d_1^+ = 1, \\ &8x_1+5x_2+d_2^- - d_2^+ = 32, \\ &x_1+4x_2\leq 4, \\ &2x_1+3x_2\leq 6, \end{aligned} \qquad (3\text{-e})$$

$x_1, x_2, d_1^-, d_1^+, d_2^-, d_2^+ \geq 0$

Then we get $x_1 = 3, x_2 = 0, d_1^- = 0, d_1^+ = 2, d_2^- = 8, d_2^+ = 0, \bar{z} = 10.$

$\therefore$the optimum value is in between 10, 30.6

4.3. 3rd class: The Roughness exists in both Goal and System Constraints:

A rough goal mathematical programming problem of the 3rd class takes the following form:

$$\min Z(d^-, d^+) = \sum_{i=1}^{m} P_i\,(w_i^- d_i^- + w_i^+ d_i^+),$$

Subject to
$$G_i(x) + d_i^- - d_i^+ = b_i \qquad , i = 1, \dots, m,$$
$$X \in S. \qquad (4)$$
$$d_i^-, d_i^+ \geq 0 \quad ,$$

where
$$G_{*i}(x) \leq G_i(x) \leq G_i^*(x),$$
$$S_* \subseteq S \subseteq S^*.$$

First solve the following problem:

$$\min Z(d^-, d^+) = \sum_{i=1}^{m} P_i\,(w_i^- d_i^- + w_i^+ d_i^+),$$

Subject to
$$G_i^*(x) + d_i^- - d_i^+ = b_i\,, i = 1, \dots, m, \qquad (4\text{-a})$$
$$X \in S^*$$
$$d_i^-, d_i^+ \geq 0$$

The obtained solution:

$$\left(X', \overline{d_i^-}, \overline{d_i^+}\right) \qquad , i = 1, \dots, m$$

It is the surly feasible optimal solution if $X' \in S_*$ and possibly feasible optimal solution if$X' \notin S_*$.

It is also called the surely optimal solution if

$$G_{*i}(X') + \overline{d_i^-} - \overline{d_i^+} = b_i\,, i = 1, \dots, m$$

If it is not satisfied then we solve the following problem

$$\min Z(d^-, d^+) = \sum_{i=1}^{m} P_i\,(w_i^- d_i^- + w_i^+ d_i^+).$$

Subject to
$$G_{*i}(x) + d_i^- - d_i^+ = B_i\,, i = 1, \dots, m \qquad (4\text{-b})$$
$$X \in S_*$$
$$d_i^-, d_i^+ \geq 0$$

Then we get

$$\left(X'', \overline{\overline{d_i^-}}, \overline{\overline{d_i^+}}\right) \,, i = 1, \dots, m$$

Then the optimum value is between:

$$\begin{pmatrix} w_1^- \overline{d_1^-} + w_1^+ \overline{d_1^+} \\ w_2^- \overline{d_2^-} + w_2^+ \overline{d_2^+} \\ \cdot \\ \cdot \\ \cdot \\ w_m^- \overline{d_m^-} + w_m^+ \overline{d_m^+} \end{pmatrix} \leq \begin{pmatrix} w_1^- d_1^- + w_1^+ d_1^+ \\ w_2^- d_2^- + w_2^+ d_2^+ \\ \cdot \\ \cdot \\ \cdot \\ w_m^- d_m^- + w_1^+ d_1^+ \end{pmatrix} \leq \begin{pmatrix} w_1^- \overline{\overline{d_1^-}} + w_1^+ \overline{\overline{d_1^+}} \\ w_2^- \overline{\overline{d_2^-}} + w_2^+ \overline{\overline{d_2^+}} \\ \cdot \\ \cdot \\ \cdot \\ w_m^- \overline{\overline{d_m^-}} + w_m^+ \overline{\overline{d_m^+}} \end{pmatrix}$$

Example3: consider

$$minz = p_1 d_1^+ + p_2 d_2^-$$

Subject to

$$G_1(x_1, x_2) \le 1,$$
$$G_2(x_1, x_2) \ge 32,$$
$$x_1 + x_2 \le G_1(x_1, x_2) \le 5x_1 + 3x_2, \qquad (4\text{-c})$$
$$2x_1 + x_2 \le G_2(x_1, x_2) \le 8x_1 + 5x_2,$$
$$X \in S \qquad X = (x_1, x_2)$$

where $S_* \subseteq S \subseteq S^*$

$$S^* = \{X \in R^2 | 6x_1 + 10x_2 \le 60 , x_1 \le 7\},$$
$$S_* = \{X \in R^2 | 4x_1 + 10x_2 \le 40 , x_1 \le 5\}$$
$$x_1, x_2, d_1^-, d_1^+, d_2^-, d_2^+ \ge 0$$

First we solve

$$minz = p_1 d_1^+ + p_2 d_2^-$$

Subject to

$$5x_1 + 3x_2 + d_1^- - d_1^+ = 1,$$
$$2x_1 + x_2 + d_2^- - d_2^+ = 32, \qquad (4\text{-d})$$
$$6x_1 + 10x_2 \le 60 ,$$
$$x_1 \le 7,$$
$$x_1, x_2, d_1^-, d_1^+, d_2^-, d_2^+ \ge 0$$

Then we get $x_1 = 0.2, x_2 = 0, d_1^- = 0, d_1^+ = 0, d_2^- = 31.6, d_2^+ = 0 , \bar{z} = 31.6$

Second we solve $minz = p_1 d_1^+ + p_2 d_2^-$

Subject to

$$x_1 + x_2 + d_1^- - d_1^+ = 1,$$
$$8x_1 + 5x_2 + d_2^- - d_2^+ = 32, \qquad (4\text{-e})$$
$$4x_1 + 10x_2 \le 40 ,$$
$$x_1 \le 5,$$
$$x_1, x_2, d_1^-, d_1^+, d_2^-, d_2^+ \ge 0$$

Then we get $x_1 = 4, x_2 = 0, d_1^- = 0, d_1^+ = 3, d_2^- = 0, d_2^+ = 0, \bar{z} = 3$

∴the optimum value is in between 3, 31.6

5. Conclusion

In this article the linear goal programming problem in rough environment when roughness found in the system constraints or goal constraints or both are introduced. Illustrative examples are introduced.

References:

[1] M. S. Osman. E. F. Lashein. E. A. Youness , T. E. M. Atteya, Mathematical Programming in Rough Environment. Optimization: A Journal of Mathematical Programming and Operations Research, Volume 60(2011), Issue 5, Pages 603 - 611.

[2] E. A. Youness, Characterizing solutions of rough programming problems European Journal of Operational Research 168(3): 1019-1029. 2006.

[3] M. S. Bazaraa, H. D. Sherali, C. M. Shetty, Nonlinear Programming: Theory and Algorithms. Wiley-Interscience; 3ed edition, 2006.

[4] L. Baoding, Theory and Practice of Uncertain Programming, First Edition, Physica-Verlag, Heidelberg, 2002.

[5] Z. Pawlak: Rough sets, International Journal of Computer and Information Sciences (5)(1982)341–356.

[6] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Boston, 1991.

[7] Z. Pawlak. Rough Sets, Rough Relations and Rough Functions. Fundamental Inform. 27(2/3) (1996): 103-108.

[8] Z. Pawlak. Vagueness, A rough set view, In: Structures in Logic and Computer Science. Springer (1997) pp.106-117.

[9] Lin Feng , Tianrui Li , Da Ruan , Shirong Gou,A vague-rough set approach for uncertain knowledge acquisition, Knowledge-Based Systems 24 (2011) 837–843.

[10] A. Charnes, W.W. Cooper "Management Models and Industrial Applications of Linear Programming" vols. 1&2, John Wiley and Sons, New York, 1961.

[11] Y. Ijiri, "Management Goals and Accounting for Control", North-Holland, Amsterdam, 1965.

[12] A. Charnes, W.W. Cooper, "Goal programming and multi-objective optimization" European Journal of Operational Research 1 (1) (1977) 39–45.

[13] R.B. Flavell, A new goal programming formulation, Omega 4 (1976) 731–732.

[14] V. Chankong. V.V. Haimes" Multiobjective Decision Making: Theory and Methodology", North-Holland, Amsterdam, 1983.

[15]Ozden Ustun. "Multi-choice goal programming formulation based on the conic scalarizing function" Applied Mathematical Modelling 36 (2012) 974–988.

Characterizing Solution of Complex Programming Based on Lexicographic Order

Hamiden Abd Elwahed[1]* Mahmoud M. El-Sherbiny[2]

Abstract

This paper aims to introduce a suggested technique for characterizing the solution of complex linear (non-linear) programming problems based on the lexicographic order. The previous work of handling the complex programming problem is reviewed and compared with the suggested technique presented in this paper. In order to illustrate and clarify the suggested technique numerical examples are presented in the sake of this paper.

Keyword: Complex programming; linear; non-linear programming; lexicographic order; Kuhn-Tucker Conditions

1. Introduction

Mathematical programming, with its countless applications in economy, science, technology, ect. has been generalized in different directions. One of these directions involves the n-dimensional complex space C^n, instead of the n-dimensional real space R^n. Mathematical programming in complex space originated by Levinson [1] generalized Farkas' theorem to the complex space and gave duality theorems for a particular case of the complex linear optimization problem. Mand and Hanson [2], generalized P. Walfe's duality from the optimization in real space to the optimization in complex space. Ben Israel [3] formulated the pair of dual problems for the linear optimization in complex space and gave duality theorems analogous to these from the linear optimization in real space. In [4], a Kuhn-Tucker constraint qualification is given and a Kuhn-Tucker theorem for the general optimization problem in complex space is proved, while Abrams [5] presented the sufficient conditions for a point to be an optimal solution of an optimization problem in complex space, saddle point theorems and duality theorems are investigated. Youness and Elbrobsy [6] formulated the problem in complex space taking into account the two terms (real and imaginary) of the objective function. In this paper, a suggested technique is presented to characterize the solution of complex nonlinear programming via lexicographic order.

The remaining part of the paper is organized as follows: in section 2, the model of complex programming problem is presented. In section 3, the suggested technique for characterizing the solution of the problem is given. Finally in section 4 numerical examples are presented in order to illustrate the suggested technique.

---

1. *Operations Research Dept., Institute of Statistical Studies and Research (ISSR), Cairo University, Egypt. Email:* hamiden_2008@yahoo.com

2. *Operations Research Dept., Institute of Statistical Studies and Research (ISSR), Cairo University, Egypt. Email:* m_sherbiny@yahoo.com

Problem formulation and concept

The model of complex programming problem can be presented as in (1).

$$\left.\begin{aligned} & Min\, f(x) = u(x) + iv(x) \\ & s.t. \\ & x \in M = \{x \in R^n : g_r(x) = l_r(x) + ih_r(x) \\ & \qquad\qquad \le b_r + ib_r',\, r = 1,\dots,m;\, i = \sqrt{-1}\} \end{aligned}\right\} \quad (1)$$

Where $u, v : R^n \to R; l_r, h_r : R^m \to R, r = 1,\dots, m$ are convex functions on M.

Definition 2.1[7]:

Lexicographic order of two complex numbers $z_1 = a_1 + ib_1$ and $z_2 = a_2 + ib_2$ is $z_1 \le z_2$ *Iff* $a_1 \le a_2$ and $b_1 \le b_2$.

In [1], the problem (1) has been transformed into (2) by ignoring the imaginary part of the complex in the objective function and considered the solution of the problem (2) as the solution of the complex problem presented in (1).

$$\left.\begin{aligned} & Min\, u(x) \\ & s.t. \\ & \quad l_r(x) \le b_r, r = 1,\dots, m; \\ & \quad h_r(x) \le b_r', r = 1,\dots, m; \end{aligned}\right\} \quad (2)$$

While in [6], Youness and Elbrollosy handled the problem (1) by transforming the problem into two sub-problems (3) and (4), taking into consideration the two parties of the complex in the objective function. Therefore, the solution of the complex problem presented in (1) can be considered as a point in the intersection of both sub problems (3) and (4).

$$\left.\begin{aligned} & Min\, u(x) \\ & s.t. \\ & \quad l_r(x) \le b_r, r = 1,\dots, m; \\ & \quad h_r(x) \le b_r', r = 1,\dots, m; \end{aligned}\right\} \quad (3)$$

and

$$\left.\begin{aligned} & Min\, v(x) \\ & s.t. \\ & \quad l_r(x) \le b_r, r = 1,\dots, m; \\ & \quad h_r(x) \le b_r', r = 1,\dots, m; \end{aligned}\right\} \quad (4)$$

2. Suggested technique

In order to handle the complex problem presented in (1), this problem converted into (5) by ignoring the imaginary part in the objective function and considering the solution of imaginary part v^*,which is found by solving (4).

$$\left.\begin{aligned} & Min\ u(x) \\ & s.t. \\ & \quad l_r(x) \leq b_r, r = 1,\dots,m; \\ & \quad h_r(x) \leq b_r^{'}, r = 1,\dots,m; \\ & \quad v(x) \leq v^* \end{aligned}\right\} \qquad (5)$$

Where v^* is the solution of (4) .

Definition 2.2[8]:

A point x^* is said to be an optimal solution of problem (5) if and only if $u(x^*) \leq u(x)$, for all x .

3. Numerical examples

In this section, two numerical examples are presented in order to demonstrate and clarify the suggested technique developed in the paper.

Consider the following:

Example 1:

$$\left.\begin{aligned} & Min\ f(x) = (x_1 - 2x_2) + i(-2x_1 - x_2) \\ & s.t. \\ & \quad (-x_1 + 3x_2) + i(x_1 + 3x_2) \leq 21 + 27i, \\ & \quad (4x_1 + 3x_2) + i(3x_1 + x_2) \leq 45 + 30i \end{aligned}\right\} \qquad (6)$$

The above mentioned problem based on definition (2.1) can be re-written as in (7) and the suggested technique will be applied.

$$\left.\begin{aligned} & Min\ f(x) = (x_1 - 2x_2) + i(-2x_1 - x_2) \\ & s.t. \\ & \quad -x_1 + 3x_2 \leq 21, \\ & \quad 4x_1 + 3x_2 \leq 45, \\ & \quad x_1 + 3x_2 \leq 27, \\ & \quad 3x_1 + x_2 \leq 30. \end{aligned}\right\} \qquad (7)$$

Based on the suggested technique, the above mentioned problem can be represented as in (8).

$$\left.\begin{array}{l} Min(x_1 - 2x_2) \\ s.t. \\ \quad -x_1 + 3x_2 \le 21, \\ \quad 4x_1 + 3x_2 \le 45, \\ \quad x_1 + 3x_2 \le 27, \\ \quad 3x_1 + x_2 \le 30, \\ \quad 2x_1 + x_2 \ge 21 \end{array}\right\} \qquad (8)$$

By solving (8), the optimal solution is $x^*= (9, 3)$, which coincides with the solution found by [7] .Therefore, we can complain that the solution found by the suggested model (5) is equivalent to the solution given by the intersection with the solutions of models (3) and (4), respectively.

Example 2:

$$\left.\begin{array}{l} Min\, f(x)=(3x_1 + x_2) + i(5x_1 - 11x_2) \\ s.t. \\ \quad (x_1^2 + x_2^2) + i(x_1 - x_2) \le 5 + i \end{array}\right\} \qquad (9)$$

The above mentioned problem based on definition (2.1) can be re-written as in (10) and the suggested technique will be applied.

$$\left.\begin{array}{l} Min\ (3x_1 + x_2) + i(5x_1 - 11x_2) \\ s.t. \\ \quad x_1^2 + x_2^2 \le 5, \\ \quad x_1 - x_2 \le 1 \end{array}\right\} \qquad (10)$$

Referring to the suggested technique, Problem (10) is written as in (11).

$$\left.\begin{array}{l} Min\ 3x_1 + x_2 \\ s.t. \\ \quad x_1^2 + x_2^2 \le 5, \\ \quad x_1 - x_2 \le 1, \\ -5x_1 + 11x_2 \ge 21 \end{array}\right\} \qquad (11)$$

By solving the model (11) the feasible area is unbounded. So , infinite number of solutions can be obtained, this result coincides with the solution found by [6].Therefore, we can complain that the solution found by the suggested model (5) is equivalent to the solution given by the intersection with the solutions of models (3) and (4), respectively.

5. Conclusion

This paper aims to introduce a suggested technique for characterizing the solution of complex linear (non-linear) programming problems based on the lexicographic order. The WINQSB computer package has been used for solving the illustrated numerical examples to obtain the results. The suggested technique is compared with the previous work of handling the complex programming problem and the results are coincided with their. The suggested technique needs less of time and effort comparing with the technique illustrated in [6] , while the solution found through solving one model instead of solving two models as suggested in [6].

References

[1] Levinson N., Linear programming in complex space, J. Math. Anal. Appl., 24(1966)44-62.

[2] Mand B., and Hasson M. A., Quadratic programming in complex space, J. Math. Anal. Appl, 20(1967)507-524

[3] Ben – Israel A., Linear equations and inequalities on finite dimensional, real or complex, vector space: A unified theory, J. Math. Anal. Appl., 2(27) (1969)367-389.

[4] Abrams R. A., and Ben – Israel A., Nonlinear programming in complex space: Necessary conditions. SIAMJ. Control, 4 (1979) 606 – 620.

[5] Abrams R. A., Nonlinear programming in complex space, Ph.D. Thesis in Applied Mathematics. North Western University, Evanston. Illinois, 1969.

[6] Youness E. A., and Elbrollosy M. E., Extension to necessary optimality conditions in complex programming, J. Appl. Math. Compu., 254(2004)229-237.

[7] Gabr.K.M, El –Banna. A. A, Youness. A. E, and Mekawy. M. I, A Study on fuzzy complex programming problems, Ph.D. Thesis in Pure Mathematics (Operations Research). Faculty of Science, Tanta University, 2012.

[8] Simmons D. M., Nonlinear Programming for Operations Research, Prentice – Hall, Inc., Englewood Cliffs, N. J., 1975.

Characterization of Basic Notions in Parametric Rough Convex Programming I

(Parameters in the constraints and Roughness in the objective function)

M.S.A.Osman[a] M.H.Eid[b] M.A.Elsisy[c]

Abstract

Parametric studying of a programming problem often yields a new insight into the programming problem at hand. The behavior of optimal solutions set is studied to detect the change of this behavior if the input data are changed. In this paper questions concerning how the optimal solutions set changes with respect to perturbations in the rough data are investigated. So it presents characterization of basic notions in parametric rough convex programming when parameters in the constraints and roughness exist in the objective function.

Key words: Rough function, rough nonlinear programming problem, the solvability set, the stability set of the first kind ,the stability set of the fifth kind .

I. Introduction

The mathematical programming in rough environment is introduced in [1,4,11]. It is classified into three classes on the basis of where roughness is found [4]; 1st Class: mathematical programming problems with rough feasible set and crisp objective function; 2nd Class: mathematical programming problems with crisp feasible set and rough objective function; 3rd Class: mathematical programming problems with rough feasible set and rough objective function. So the nonlinear programming problem in a rough environment is called rough nonlinear programming problem (PNLPP). PNLPP can be defined according to roughness which may exist in objective functions and/or constraints. There are two solutions sets of the PNLPP; surely optimal solutions set and possibly optimal solutions set in the 2nd class. This paper investigates the characterization of basic notions in parametric rough convex programming when parameters in the constraints and roughness in the objective function. The study of rough set theory (RST) and rough functions is essential to these parametric studying of PNLPP.

This paper is organized as follows: Section 2 (basic concepts) introduces the

[a]Al Asher University, Al Asher of Ramadan, Egypt.

[b]Faculty of Engineering (Shoubra), Benda University, Benda, Egypt.

[c]Faculty of Engineering (Benda), Benda University, Benda, Egypt. (elsisy.mohamed@yahoo.com)

nonlinear programming problem, briefly reviews some basics of rough function and rough nonlinear programming problem. In Section 3, the study addresses characterization of basic notions in parametric rough convex programming when parameters in the constraints and roughness in the objective function. Section 4 includes conclusion and recommendations for further studies.

2 Basic Concepts:

Rough set theory (RST) is a new mathematical theory introduced by Pawlak [7,8,9,10] in the early 1980s to deal with vagueness or uncertainty. RST expresses vagueness by employing a boundary region of a set and not by means of membership function [1,3,9]. The basic concept of RST is the approximation of indefinable sets or concepts through definable sets. Rough sets meaning approximation sets are built on ordinary sets. The rough function is a new concept of functions based on the rough set theory. It is needed in many applications to defined functions when there are rough data presented to model these functions. So it is a theoretical basis for rough mathematical programming [4,8,9]. As shown in [4], let U be the universe set ($U \neq \emptyset$) and let F be a sequence of functions $f_0(x), f_1(x), \dots, f_k(x)$ such that

- $f: U \to R$ for every $f \in F$.
- $f_0(x), f_1(x), \dots, f_k(x)$ are continuous on U.
- $f_0(x) \leq f_1(x) \leq \cdots \leq f_k(x)$ for every $x \in U$.

Suppose that $V = U \times R$, then the ordered pair $A = (V, F)$ will be called the approximation space generated by F on set V. In fact, every non-empty sequence F is a discretization on V, and generates a partition $V/R = \{C_0, C_1, C_2, \dots, C_{3k+2}\}$ on V, where sets $C_0, C_1, C_2, \dots, C_{3k+2}$ are the equivalence classes generated by F, and are defined as follows[4]:

$C_0 = \{(x,y) \in V | y < f_0(x)\}$

$C_1 = \{(x,y) \in V | y = f_0(x), y \neq f_1(x)\}$

$C_2 = \{(x,y) \in V | y = f_0(x) = f_1(x)\}$

$C_3 = \{(x,y) \in V | f_0(x) < y < f_1(x)\}$

$C_4 = \{(x,y) \in V | y = f_1(x), y \neq f_0(x), y \neq f_2(x)\}$

$C_5 = \{(x,y) \in V | y = f_1(x) = f_2(x) \neq f_0(x)\}$

$C_6 = \{(x,y) \in V | f_1(x) < y < f_2(x)\}$

$C_7 = \{(x,y) \in V | y = f_2(x), y \neq f_1(x), y \neq f_3(x)\}$

$C_{3k+2} = \{(x, y) \in V | y > f_k(x)\}$.

As seen in above description about rough function, it is defined by many functions where it intersects with some functions in some area and between some functions in other areas. So to simplify the rough function will be defined by two functions.

Definition 1 (Rough Function): The rough function $f(x)$ is a function that cannot be defined or formatted on a crisp mathematical formula but it can be defined by two functions $f_*(x), f^*(x)$ such that $f_*(x) \leq f(x) \leq f^*(x)$ where these functions surrounds it from above and down. The surrounding functions $[f_*(x), f^*(x)]$contain all surely and possibly values of rough function at any $\in X$. The surely values of $f(x)$ are the values of $f_*(x), f^*(x)$ at $x \in X$ such that $f_*(x) = f^*(x)$.

To show the solution of the rough function, let $f(x)$ be an arbitrary rough function in the approximation space $A = (V, F)$, then the points of V are classified into three classes with respect to $f(x)$, as follows:

a) A point $P = (\bar{x}, \bar{y}) \in V$ is surely optimal solution to the curve $y = f(x)$ if and only if $\bar{y} = f_*(\bar{x}) = f^*(\bar{x})$.

b) A point $P = (\bar{x}, \bar{y}) \in V$ is possibly optimal solution to the curve $y = f(x)$ if and only if $\bar{y} \in [f_*(x), f^*(x)]$.

c) A point $P = (\bar{x}, \bar{y}) \in V$ is not solution to the curve $y = f(x)$ if and only if $\bar{y} \notin [f_*(x), f^*(x)]$.

Definition 2 (Convex Upper Rough Function): The rough function $f(x)$is called an upper convex rough function on the set X if for any x_1, x_2in X and any $\alpha \in [0,1]$,

$$f^*(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f^*(x_1) + (1 - \alpha)f^*(x_2).$$

Definition 3 (Convex Lower Rough Function): The rough function $f(x)$is called a lower convex rough function on the set X if for any x_1, x_2in X and any $\alpha \in [0,1]$

$$f_*(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f_*(x_1) + (1 - \alpha)f_*(x_2).$$

Definition 4 (Convex Rough Function): The rough function $f(x)$is called a convex rough function on the set X if its lower and upper approximation functions are convex functions.

As known the nonlinear programming problem (NLPP) is defined as a problem to optimize objective function under a given constraints and one of these functions is nonlinear or all [2]. NLPP is to determine a vector $\bar{x} = (x_1, \dots, x_n)^T$ that solves the following problem:

$$\min f(x)$$

subject to $\quad x \in X$

where $\quad X = \{x \in \mathrm{R}^n | g_i(x) \le 0 \quad i = 1,2, \dots, l\}$

x is a vector of design variables. $f\ (x): R^n \ \to R$ is called objective function, X is the feasible region.

The rough nonlinear programming problem when roughness in the objective function and a feasible region is a crisp set can be defined as following:

$$\min f(x)$$

subject to $\quad x \in X \qquad (1)$

where $\quad f_*(x) \le f(x) \le f^*(x)$

$f_*(x)$ and $f^*(x)$ are surrounding $f(x)$ from down and above respectively. X is the feasible region. To solve these problem, firstly solve the following problem:

$$\min f_*(x)$$

subject to $\quad x \in X$

The optimal solutions set is defined as:

$$C = \{\bar{x} \in X | f_*(\bar{x}) = \min_{x \in X} f_*(x)\}.$$

The optimal value of the lower approximation function is

$$f_* = \min_{x \in X} f_*(x)$$

$$C_1 = \{\bar{x} \in C | f^*(\bar{x}) = f_*\} = \begin{cases} \ne \emptyset \\ = \emptyset \end{cases}.$$

C_1 is surely optimal solutions set when $C_1 \ne \emptyset$. $C \sim C_1$ is a possibly optimal solutions set. When $C_1 = \emptyset$, the problem is resolved to get C_2 which it is an optimal solutions set of the upper approximation function.

$$C_2 = \{\bar{x} \in X | f^*(\bar{x}) = \min_{x \in X} f^*(x)$$

The optimal value for upper approximation function is

$$f^* = \min_{x \in X} f^*(x).$$

However the optimal value of the rough objective function is between the optimal value of the lower approximation function $f_*(x)$ and the optimal value of the upper approximation function $f^*(x)$.

$$f_* \le \min_{x \in X} f(x) \le f^*.$$

Remark: $\quad \min_{x \in X} f_*(x) \le \min_{x \in X} f(x) \le \min_{x \in X} f^*(x)$

$$f_* \leq f(x) \leq f^*$$

$$f_* = \min_{x \in X} f_*(x) \quad , \quad f^* = \min_{x \in X} f^*(x)$$

$$\text{If } \min_{x \in X} f_*(x) = \min_{x \in X} f^*(x) \quad \leftrightarrow \quad C_1 \neq \emptyset \ .$$

If $C_1 = \emptyset$ $\rightarrow$ The problem does not have a surely optimal solutions set.

Definition 5 (The surely optimal solutions set): If $C_1 \neq \emptyset$, then C_1 contains all surely optimal solutions, then it can be called the surely optimal solutions set

$$\dot{C}_1 = \{\bar{x} \in C | f^*(\bar{x}) = f_*\}$$

Definition 6 (The possibly optimal solutions set): $C \sim C_1$ contains possibly optimal solutions then it can be called the possibly optimal solutions set that is defined as

$$C \sim C_1 = \{x | x \in C \cap \ x \notin C_1\}.$$

3 The parameters found in the feasible region and roughness in the objective function:

Osman papers [5, 6] provide the basis of theorems and concepts of the parametric convex programming which are used in this paper, i.e. the notions of the set of feasible parameters, solvability set, the stability set of the first kind and the stability set of the fifth kind which are defined and analyzed qualitatively for the problem:

$$\min f(x) \tag{I}$$

subject to $\quad \mathfrak{M}(v) = \{x \in R^n | g_r(x) \leq v_r \ , r = 1,2, \ldots, l\}$

where $\quad f_*(x) \leq f(x) \leq f^*(x)$

$f^*(x); f_*(x); g_r(x), r = 1,2, \ldots, l$ are convex functions, possessing continuous first order partial derivatives on the n-dimensional vector space R^n and $v_r, r = 1,2, \ldots, l$ are arbitrary real numbers.

Definition 7 (The set of feasible parameters) [4]: The set of feasible parameters for problem (I) denoted by $\mathfrak{A}$ is defined

by: $$\mathfrak{A} = \{v \in {}'R^l | \mathfrak{M}(v) \neq \emptyset\} \tag{2}$$

where $'R^l$ is the l-dimensional vector space of parameters.

The set $\mathfrak{A}$ is nonempty, unbounded and moreover, if $v \in \mathfrak{A}$, then all v in the nonnegative orthant of the parametric vector space $'R^l$ with the origin at $v = \bar{v}$ belong to the set $\mathfrak{A}$. The set $\mathfrak{A}$ is convex and if there is $v \in \mathfrak{A}$ such that $\mathfrak{M}(v)$ is bounded, then $\mathfrak{A}$ is closed [4].

Definition 8 (The solvability set): The solvability set for problem (I) denoted by $\mathfrak{B}$, is defined by

$$\mathfrak{B} = \{v \in {}'R^l | m_{*opt}(v) \neq \emptyset \cap m^*_{opt}(v) \neq \emptyset\} \quad (3)$$

where $m_{*opt}(v), m^*_{opt}(v)$ are the set of all optimal points of problem (I), i.e.

$$\begin{aligned} m_{*opt}(v) &= \{x^* \in \mathfrak{M}(v) | f_*(x^*) = \min_{x \in \mathfrak{M}(v)} f_*(x)\} \\ m^*_{opt}(v) &= \{x^* \in \mathfrak{M}(v) | f^*(x^*) = \min_{x \in \mathfrak{M}(v)} f^*(x)\} \end{aligned} \quad (4)$$

If $m_{opt}(v) = m_{*opt}(v) \cap m^*_{opt}(v) = \{x^* \in \mathfrak{M}(v) | f_*(x^*) = f^*(x^*)\} \neq \emptyset$ (5)

where $f_*(x^*) = \min_{x \in \mathfrak{M}(v)} f_*(x), f^*(x^*) = \min_{x \in \mathfrak{M}(v)} f^*(x)$, then it is called the surely optimal solutions set. If $m_{opt}(v) = \emptyset$ the problem does not have a surely optimal solutions set but have only a possibly optimal solutions set

$$m_{popt}(v) = m_{*opt}(v) \cup m^*_{opt}(v).$$

If for one $v \in \mathfrak{B}$ it holds that the set $m_{opt}(v)$ is bounded, then $\mathfrak{B} = \mathfrak{A}$ where $m_{opt}(v)$ are given by (5).

Definition 9 (The stability set of the first kind): Suppose that $v \in \mathfrak{B}$ with a corresponding optimal solutions $\bar{x}, \bar{\bar{x}}$, of $f_*(x), f^*(x)$ respectively then the stability set of the first kind of problem (I) corresponding to $\bar{x}, \bar{\bar{x}}$ denoted by two sets $\mathfrak{S}_*(\bar{x}), \mathfrak{S}^*(\bar{\bar{x}})$

$$\mathfrak{S}_*(\bar{x}) = \{v \in {}'R^l | f_*(\bar{x}) = \min_{x \in \mathfrak{M}(v)} f_*(x)\}$$
$$\mathfrak{S}^*(\bar{\bar{x}}) = \{v \in {}'R^l | f^*(\bar{\bar{x}}) = \min_{x \in \mathfrak{M}(v)} f^*(x)\} \quad (6)$$

If $\bar{x} = \bar{\bar{x}}$, there are two cases. The first one $f_*(\bar{x}) = f^*(\bar{\bar{x}})$, it can be called a surely optimal solution. In these case, the stability set of the first kind of problem (I) is gotten. The intersection between $\mathfrak{S}_*(\bar{x}) \cap \mathfrak{S}^*(\bar{\bar{x}})$ keep a surely optimal solution at $\bar{x}$ is always generated. The second case $f_*(\bar{x}) \neq f^*(\bar{\bar{x}})$, then the optimal value of the rough function is between the lower- and upper- approximation function. So the stability set of the fifth kind can be defined and obtained.

Definition 10 (The stability set of the fifth kind): Suppose that $v \in m_{popt}(v)$ with optimal values $f_*(\bar{x}), f^*(\bar{\bar{x}})$ corresponding to the optimal value of the upper approximation function and the lower approximation function respectively, then the stability sets of the fifth kind of problem I corresponding to $f_*(\bar{x}), f^*(\bar{\bar{x}})$ denoted by $T^*(f^*(\bar{\bar{x}})), T_*(f_*(\bar{x}))$ respectively are defined by:

$$T_*(f_*(\bar{x})) = \{(x, v) \in R^n \times {}'R^l | \min_{x \in \mathfrak{M}(v)} f_*(x) = f_*(\bar{x})\}$$
$$T^*(f^*(\bar{\bar{x}})) = \{(x, v) \in R^n \times {}'R^l | \min_{x \in \mathfrak{M}(v)} f^*(x) = f^*(\bar{\bar{x}})\}$$

The intersection region between $\left(T^*(f^*(\bar{\bar{x}})) \cap T_*(f_*(\bar{x}))\right)(v)$ keep the value of the rough objective function between the optimal value of the upper approximation function and the optimal value of the lower approximation function however the parameters and variables are change.

The following steps lead us to obtain the stability set of the first kind & fifth kind when the parameters in the constraints and the roughness in the objective function:

Step 1: Formulate the problem and include all the parameters that need to be investigated.

Step 2: Start with any certain $v \in B$ and get the optimal solutions set C to the lower approximation function.

Step 3: (a) If there is $\bar{x} \in C_1$ then it is a surely optimal solution.

- Formulate the Kuhn Tucker conditions (KTC) for upper- and lower-approximation functions at $\bar{x}$.
- Get $\mathfrak{S}_*(\bar{x})$ that $\min_{\mathfrak{M}(v)} f^*(x)$ at $\bar{x}$.
- Get $\mathfrak{S}_*(\bar{x})$ that $\min_{x \in \mathfrak{M}(v)} f_*(x)$ at $\bar{x}$.
- Get $\mathfrak{S}_*(\bar{x}) \cap \mathfrak{S}^*(\bar{\bar{x}})$.

(b) Otherwise, $\bar{x} \in C(\, C_1 = \emptyset)$ then it is a possibly optimal solution, get an optimal solution $\bar{\bar{x}}$ of the upper approximation function.

- Formulate KTC of the problem for the upper approximation function and the lower approximation functions.
- Get T_* that keep $f_*(x)$ at $f_*(\bar{x})$ on $\mathfrak{M}(v)$.
- Get T^*that keep $f^*(x)$ at $f^*(\bar{\bar{x}})$ on $\mathfrak{M}(v)$.

Then $T^*(f(\bar{\bar{x}})) \cap T_*(f(\bar{x}))(v)$ keep the value of the rough objective function between the optimal value of the upper approximation function and the optimal value of the lower approximation function $f_*(\bar{x}) \leq f(x) \leq f^*(\bar{\bar{x}})$, however the parameters or variables are changes.

Example: $\min f(x)$

subject to
$$x^2 + y^2 \leq v_1$$
$$y - x \leq v_2$$

where
$$x^2 - 3y + y^2 + 1 \leq f(x) \leq x^2 - y + 2y^2 + 4$$

The Lagrangian of the problem can be stated according to the upper approximation function and lower approximation function respectively:

$$L^*(x,y,\lambda_1,\lambda_2,\mu_1,\mu_2) = x^2 - y + 2y^2 + 4 + \mu_1(x^2 + y^2 - v_1) + \mu_2(-x + y - v_2)$$

$$L_*(x,y,\lambda_1,\lambda_1,\mu_1,\mu_2) = x^2 - 3y + y^2 + 1 + \mu_1(x^2 + y^2 - v_1) + \mu_2(-x + y - v_2)$$

$$KTC^* \longrightarrow \begin{cases} (4+2\mu_1)y + \mu_2 - 1 = 0 \\ (2+2\mu_1)x - \mu_2 - 1 = 0 \\ \mu_1(x^2+y^2-v_1) = 0 \\ \mu_2(-x+y-v_1) = 0 \\ x^2+y^2 \le v_1 \\ -x+y \le v_2 \end{cases}, \quad KTC_* \longrightarrow \begin{cases} (2+2\mu_1)y + \mu_2 - 3 = 0 \\ (2+2\mu_1)x - \mu_2 - 1 = 0 \\ \mu_1(x^2+y^2-v_1) = 0 \\ \mu_2(-x+y-v_1) = 0 \\ x^2+y^2 \le v_1 \\ -x+y \le v_2 \end{cases}$$

The feasible region $\mathfrak{M}(4,2) = \{(x,y) \in R^2 | x^2 + y^2 \le 4 , y - x \le 2\}$.

The set of feasible parameters for the example is defined by:

$$\mathfrak{A} = \{v \in {}'R^l | v_1 \ge 0, v_2 \ge 0\} \cup \{v \in {}'R^l | v_1 \ge 0, v_2 \le 0, v_2^2 \le 2v_1\}.$$

$$\min_{x\in\mathfrak{M}(4,2)} x^2 - 3y + y^2 + 1 = f_*\left(0,\frac{3}{2}\right) = -\frac{5}{4}$$

$$\min_{x\in\mathfrak{M}(4,2)} x^2 - y + 2y^2 + 4 = f^*\left(0,\frac{1}{4}\right) = \frac{31}{8}$$

$$-\frac{5}{4} \le \min_{x\in\mathfrak{M}(4,2)} f(x) \le \frac{31}{8}$$

$$T_*(-\frac{5}{4}) = \left\{(x,y,v_1,v_2) \in R^2 \times R^2 \middle| v_1 \ge \frac{9}{4}, v_2 \ge \frac{3}{2}, x = 0, y = \frac{3}{2}\right\}$$

$$T^*(\frac{31}{8}) = \left\{(x,y,v_1,v_2) \in R^2 \times R^2 \middle| v_1 \ge \frac{1}{16}, v_2 \ge \frac{1}{4}, x = 0, y = \frac{1}{4}\right\}$$

$$\left(T_*\left(-\frac{5}{4}\right) \cap T^*\left(\frac{31}{8}\right)\right)(v) = \left\{v \in R^2 | v_1 \ge \frac{9}{4}, v_2 \ge \frac{3}{2}\right\}$$

4. Conclusion

This paper discusses the parametric study of the PNLPP when the objective function is a rough function and parameters found in a crisp feasible region set. Also, the paper addresses new concepts including the solvability set, the stability set of the first kind and the stability set of the fifth kind on rough environment. The surely- and possibly- optimal solutions sets of PNLPP in the 2nd Class are characterized. Future studies may focus on the parametric study of the PNLPP when roughness and parameters exist on both the objective function and feasible region.

References

[1] Baoding, L. (2002). Theory and Practice of Uncertain Programming. (1st ed).

Physica-Verlag, Heidelberg.

[2] Bazaraa, M. S., Sherali, H., &Shetty, C. M. (2006). Nonlinear Programming: Theory and Algorithms. (3rd ed). Wiley-Inter science.

[3] Feng, L., Li, T., Ruan, D., and Gou, .S. (2011). A vague-rough set approach for uncertain knowledge acquisition. Knowledge-Based Systems 24. (Pages 837–843).

[4] Osman, M. S., Lashein, E. F., Youness, E.A, and Atteya, T. E. M. (2011). Mathematical Programming in Rough Environment, Optimization: A Journal of Mathematical Programming and Operations Research, Volume 60. Issue 5 (Pages 603 – 611).

[5] Osman. M. S. A.,(1977) Qualitative analysis of basic notions in parametric convex programming I (Parameters in the constraints Applications of Mathematics, Vol. 22 , No. 5, 318—332.

[6] Osman. M. S. A., (1977) Qualitative analysis of basic notions in parametric convex programming. II (Parameters in the objective function) Applications of Mathematics, Vol. 22, No. 5, 333—348.

[7] Pawlak, Z. (1982). Rough sets, International Journal of Computer and Information Sciences (5) (Pages 341–356).

[8] Pawlak, Z. (1991). Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Boston.

[9] Pawlak, Z. (1996). Rough Sets, Rough Relations and Rough Functions. Fundamental Inform. 27(2/3) (Pages 103-108).

[10] Pawlak, Z. (1997). Vagueness: A rough set view. Structures in Logic and Computer Science. Springer (Pages 106-117).

[11] Youness, E.A. (2006). Characterizing solutions of rough programming problems. European Journal of Operational Research 168(3) (Pages 1019-1029).

Characterization of Basic Notions in Parametric Rough Convex Programming II

(Parameters in the objective function and Roughness in the feasible region)

M.S.A.Osman[a] M.H.Eid[b] M.A.Elsisy[c]

Abstract

Parametric studying of a programming problem often yields a new insight into the programming problem at hand. The behavior of optimal solutions set is studied to detect the change of this behavior if the input data are changed. In this paper questions concerning how the optimal solutions set changes with respect to perturbations in the rough data are investigated. It presents characterization of basic notions in parametric rough convex programming when parameters in the objective function and roughness in the feasible region.

Key word: Rough set, rough nonlinear programming problem, the solvability set, the stability set of the first kind, the stability set of the fifth kind.

I. Introduction

The mathematical programming in rough environment is introduced in [1, 4, 11]. It is classified into three classes on the basis of where roughness is found [4]; 1st Class: mathematical programming problems with rough feasible set and crisp objective function; 2nd Class: mathematical programming problems with crisp feasible set and rough objective function; 3rd Class: mathematical programming problems with rough feasible set and rough objective function. So the nonlinear programming problem in a rough environment is called rough nonlinear programming problem (RNLPP) and can be defined according to roughness which may exist in objective functions and/or constraints. There are two solutions sets of the RNLPP; surely optimal solutions set and possibly optimal solutions set in the 1st Class. This paper investigates the Characterization of basic notions in parametric rough convex programming when parameters in the objective functions and roughness in the feasible region. The study of rough set theory (RST), the rough optimality and rough functions is essential to the parametric studying of RNLPP.

[a]Al Asher University, Al Asher of Ramadan, Egypt.

[b]Faculty of Engineering (Shoubra), Benda University, Benda, Egypt

[c]Faculty of Engineering (Benda), Benda University, Benda, Egypt.(elsisy.mohamed@yahoo.com)

This paper is organized as follows: Section 2 (Basic Concepts) introduces the nonlinear programming problem and briefly reviews some basics of rough set theory. In Section 3, the study addresses characterization of basic notions in parametric rough convex programming when parameters in the objective functions and roughness in the feasible region. Section 4 includes conclusion and recommendations for further studies.

2 Basic Concepts:

Rough set theory (RST) is a new mathematical theory introduced by Pawlak [7, 8, 9, 10] in the early 1980s to deal with vagueness or uncertainty. RST expresses vagueness by employing a boundary region of a set and not by means of membership function [1,3,9]. The basic concept of RST is the approximation of indefinable sets or concepts through definable sets. Rough sets meaning approximation sets are built on ordinary sets. So it can be described by two sets the lower and upper approximation set which are deterministic sets. Lower approximation set, X_* is defined as the biggest deterministic set contains on the rough set which contains all of elements which surely belong to set X. The upper approximation set, X^* is defined as the smallest deterministic set contains the rough set which contains all elements which possibly and surely belong to rough set X where $X_* \subseteq X \subseteq X^*$. The difference between the upper and the lower approximations is called the boundary of rough set. It is denoted by $X_{BN} = X^* \sim X_* = \{x | x \in X^*, x \notin X_*\}$. The set X is called exact (crisp) in the available information if $X_{BN} = X^* \sim X_* = \emptyset$, otherwise the set X is inexact (rough).

The nonlinear programming problem (NLPP) is defined as a problem to optimize an objective function under given constraints and the function and/or constraints are nonlinear [2]. NLPP is to determine a vector $\bar{x} = (x_1, \dots, x_n)^T$ that solves the problem:

$$\min f(x)$$

subject to $\quad x \in X$

where $\quad X = \{x \in \mathrm{R}^n | g_i(x) \leq 0 \quad i = 1,2, \dots, l\}$

x is a vector of design variables. $f(x): R^n \to R$ is called the objective function, X is the feasible region. The first class of RNLPP is defined as follows:

$$\min f(x)$$

subject to $\quad x \in X$

where $\quad X_* \subseteq X \subseteq X^*$

X is the feasible region of the problem. The function $f: X^* \to R$ is a crisp objective function which is continuous on X_* and X^* respectively. The solution sets of the 1st Class of RNLPP are called surely optimal solutions set and possibly optimal solutions set. To solve these problem, firstly solve the following problem:

$$\min f(x)$$

subject to $\quad x \in X^*$

The optimal solutions set of the above problem is

$$C = \{\bar{x} \in X^* | f(\bar{x}) = \min_{x \in X^*} f(x)\}$$

The optimal value of the objective function on upper approximation set is

$$f_* = \min_{x \in X^*} f(x)$$

$$C_1 = C \cap X_* = \begin{cases} \neq \emptyset \\ = \emptyset \end{cases}$$

C_1 is a surely optimal solutions set when $C_1 \neq \emptyset$. $C \sim C_1$ is a possibly optimal solutions set. Otherwise if $C_1 = \emptyset$ then $C \subseteq X_{BN}$. The problem will be resolved on the lower approximation set to get C_2 which is an optimal solutions set on the lower approximation set.

$$C_2 = \left\{\bar{x} \in X_* \middle| f(\bar{x}) = \min_{x \in X_*} f(x)\right\}$$

The problem does not have surely optimal solutions set. The optimal value of the objective function on the lower approximation set is

$$f^* = \min_{x \in X_*} f(x).$$

Then the optimal value of the objective function is between the optimal value of objective function on the lower approximation set X_* and the optimal value of the objective function on the upper approximation set X^*

$$\min_{x \in X^*} f(x) \leq \min_{x \in X} f(x) \leq \min_{x \in X_*} f(x)$$

Definition 1 (The surely optimal solutions set): If $C_1 \neq \emptyset$, then C_1 contains all surely optimal solutions, hence it is called the surely optimal solutions set $C_1 \subseteq X_*$.

$$C_1 = \{\bar{x} \in X_* | f(\bar{x}) = \min_{x \in X^*} f(x)\}$$

Definition 2 (The possibly optimal solutions set): $C \sim C_1$ contains possibly optimal solutions, hence it is called the possibly optimal solutions set because $\{C \sim C_1\} \subseteq X_{BN}$.

$$C \sim C_1 = \{\bar{x} \in X_{BN} | f(\bar{x}) = \min_{x \in X^*} f(x)\}$$

3 The parameters found in the objective function and roughness in the feasible region:

Osman papers [5,6] constitute the basis of theorems and concepts of the parametric convex programming which are used in this paper, i.e. the notions of the set of feasible parameters, the solvability set, the stability set of the first kind and the stability set of the fifth kind are defined and analyzed qualitatively for the problem

$$\min f(x, \lambda) \qquad \text{(II)}$$

subject to $$x \in X$$

where $$X_* \subseteq X \subseteq X^*$$

$$X^* = \{x \in R^n | G_r(x) \leq 0, r = 1,2, \dots, L\}$$

$$X_* = \{x \in X^* | g_i(x) \leq 0 , i = 1,2, \dots, l\}$$

X is the feasible region of the problem. $f(x, \lambda)$ is an objective function, $g_i(x) \leq 0$, $i = 1,2, \dots, l$ and $G_r(x) \leq 0, r = 1,2, \dots, L$ are convex functions possessing continuous first order partial derivatives on R^n and λ are real parameters. The restriction sets X^*, X_* are supposed to be nonempty and fixed.

Definition 3 (The solvability set): The solvability set of problem (II) denoted by B is defined by two sets

$$B_* \subseteq B \subseteq B^* \qquad (1)$$

where $$B^* = \{\lambda \in R^m | \min_{x \in X^*} f(x, \lambda) \text{ exists}\}$$

$$B_* = \{\lambda \in R^m | \min_{x \in X_*} f(x, \lambda) \text{ exists}\}.$$

Definition 4 (The stability set of the first kind): Suppose that $\lambda \in B$ with a corresponding surely optimal solution $\bar{x}$ which are the optimal solution of $f(x, \lambda)$ according to the upper approximation set, then the stability set of the first kind of problem (II) corresponding to $\bar{x}$ denoted by $S^*(\bar{x})$ is defined by

$$S^*(\bar{x}) = \{\lambda \in B^* | f(\bar{x}, \lambda) = \min_{x \in X^*} f(x, \lambda)\}.$$

The stability set of the first kind contains the surely optimal solution for the problem. If $\bar{x}$ is a possibly optimal solution, then the stability of fifth kind can be defined and obtained.

Definition 5 (The stability set of the fifth kind): Suppose that $\lambda \in B$ with optimal values $f(\bar{x}), f(\bar{\bar{x}})$ of the objective function on the upper approximation set and the lower approximation set respectively, then the stability sets of the fifth kind of problem II corresponding to $f(\bar{x}), f(\bar{\bar{x}})$ denoted by

$$T_*(f(\bar{\bar{x}})) = \{(x, \lambda) \in R^n \times R^m | f(x, \lambda) = f(\bar{\bar{x}})\}$$

$$T^*(f(\bar{x})) = \{(x,\lambda) \in R^n \times R^m | f(x,\lambda) = f(\bar{x})\}$$

The stability sets $T^*(f(\bar{x})) \cap T_*(f(\bar{\bar{x}}))(\lambda)$ keep the value of the objective function between the optimal value of objective function on the lower approximation set and the optimal value of the objective function on the upper approximation set however the parameters and variables are change.

The following steps lead us to obtain the stability set of the first kind & fifth kind when the parameters in objective function and roughness in the feasible region:

Step 1: Formulate the problem and include all the parameters that need to be investigated.

Step 2: Start with any certain $\lambda \in B$ and get the optimal solutions set C to the upper approximation set.

Step 3: (a) If there is $\bar{x} \in \{C \cap X_*\}$ then it is a surely optimal solution.

- Formulate the Kuhn- Tucker conditions (KTC) on the upper approximation set at $\bar{x}$.
- Get S^* that $\min_{X^*} f(x,\lambda)$ at $\bar{x}$.

  (b) Otherwise, $\bar{x} \in C$ ($C_1 = \emptyset$) it is a possibly optimal solution, get an optimal solution $\bar{\bar{x}}$ according to the lower approximation set.
- Formulate KTC of the problem on the upper approximation set and the lower approximation set.
- Get T_* that keep $f(x,\lambda)$ at $f(\bar{\bar{x}})$.
- Get T^*that keep $f(x,\lambda)$ at $f(\bar{x})$.
- Then $T^*(f(\bar{x})) \cap T_*(f(\bar{\bar{x}}))(\lambda)$ keep the value of the objective function is between the optimal value of objective function on the upper approximation set and the optimal value of the objective function on the lower approximation set $f(\bar{x}) \leq f(x) \leq f(\bar{\bar{x}})$, however the parameters or variables are changes.

Example: Consider the problem

$$\min x^2 + \lambda_1 x + y^2 + \lambda_2 y$$

subject to $\quad x \in X$

where $\quad X_* \subseteq X \subseteq X^*$

$$X^* = \{x \in R^n | x^2 + y^2 \leq 4, x + y \leq 2\}$$

$$X_* = \{x \in X^* | x^2 + y^2 \leq 1, x + y \leq 1\}$$

The Lagrangian of the problem can be stated according to the upper approximation set and the lower approximation set respectively:

$L_*(x, y, \lambda_1, \lambda_2, \mu_1, \mu_2) = x^2 + \lambda_1 x + y^2 + \lambda_2 y + \mu_1(x^2 + y^2 - 4) + \mu_2(x + y - 2)$

$L^*(x, y, \lambda_1, \lambda_1, \mu_1, \mu_2) = x^2 + \lambda_1 x + y^2 + \lambda_2 y + \mu_1(x^2 + y^2 - 1) + \mu_2(x + y - 1)$

$$KTC_* \longrightarrow \begin{cases} (2+2\mu_1)y + \lambda_2 + \mu_2 = 0 \\ (2+2\mu_1)x + \lambda_1 + \mu_2 = 0 \\ \mu_1(x^2 + y^2 - 4) = 0 \\ \mu_2(x + y - 2) = 0 \\ x^2 + y^2 \leq 4 \\ x + y \leq 2 \end{cases} , \quad KTC^* \longrightarrow \begin{cases} (2+2\mu_1)y + \lambda_2 + \mu_2 = 0 \\ (2+2\mu_1)x + \lambda_1 + \mu_2 = 0 \\ \mu_1(x^2 + y^2 - 1) = 0 \\ \mu_2(x + y - 1) = 0 \\ x^2 + y^2 \leq 1 \\ x + y \leq 1 \end{cases}$$

By using $\lambda_1 = \lambda_2 = 1$, the optimal solution of RNLPP is $(-\frac{1}{2}, \frac{1}{2})$ on the upper approximation set and lower approximation set respectively. The stability sets of the first kind of problem corresponding to the point $(-\frac{1}{2}, \frac{1}{2})$ according to the upper approximation set and lower approximation set are formed as follows:

$$S^* = \{\lambda \in R^2 | \lambda_1^2 + \lambda_2^2 < 16, \lambda_1 + \lambda_2 > -4\}.$$

When $\lambda_1 = 2\lambda_2 = 2$, the optimal solution of RNLPP is $(-1, -\frac{1}{2})$ on the upper approximation set and $(-\frac{2}{\sqrt{5}}, -\frac{1}{\sqrt{5}})$ on the lower approximation set respectively. The optimal value of the objective function is $f\left(-1, -\frac{1}{2}\right) = -\frac{5}{4}$ on the upper approximation set and it is $f\left(-\frac{2}{\sqrt{5}}, -\frac{1}{\sqrt{5}}\right) = 1 - \sqrt{5}$ on the lower approximation set. The stability sets of the fifth kind of problem corresponding to the values $-\frac{5}{4}, 1 - \sqrt{5}$ are formed as follows:

$T^*\left(-\frac{5}{4}\right) =$

$$\left\{\begin{matrix}(-2 \leq x < 0, y = 0) \vee \\ (0 < x \leq 2, y = 0) \vee \\ (x = 0, -2 \leq y < 0) \vee \\ \left(0 < x \leq \sqrt{4 - y^2}, -2 \leq y < 0\right) \vee \\ \left(-\sqrt{4 - y^2} \leq x < 0, -2 \leq y < 0\right) \\ \vee (x = 0, 0 < y \leq 2) \vee \\ (0 < x \leq 2 - y, 0 < y \leq 2) \vee \\ \left(-\sqrt{4 - y^2} \leq x < 0, 0 < y \leq 2\right)\end{matrix}, 4\lambda_2 y = -4x^2 - 4\lambda_1 x - 4y^2 - 5\right\}$$

$$T_*(1 - \sqrt{5}) =$$

$$\left\{\begin{matrix}(x = 0, y = -1) \vee \\ (-1 \leq x < 0, y = 0) \vee \\ (0 < x \leq 1, y = 0) \vee \\ (x = 0, -1 < y < 0) \vee \\ \left(0 < x \leq \sqrt{1 - y^2}, -1 \leq y < 0\right) \vee \\ \left(-\sqrt{1 - y^2} \leq x < 0, -1 \leq y < 0\right) \\ \vee (x = 0, 0 < y < 1) \vee \\ (0 < x \leq 1 - y, 0 < y < 1) \vee \\ \left(-\sqrt{1 - y^2} \leq x < 0, 0 < y < 1\right)\end{matrix}, \lambda_2 y = -x^2 - \lambda_1 x - y^2 + 1 - \sqrt{5}\right\}$$

$$T_*(1 - \sqrt{5}) \cap T^*\left(-\frac{5}{4}\right) = \left\{\begin{matrix}-1 \leq x < 0, \quad y = 0 \\ 0 < x \leq 1, \quad y = 0 \\ -1 < x \leq 0, -\sqrt{1 - x^2} \leq y < 0 \\ -1 < x \leq 0, 0 < y \leq \sqrt{1 - x^2} \\ 0 < x < 1, 0 < y \leq 1 - x \\ 0 < x \leq 1, -\sqrt{1 - x^2} \leq y < 0\end{matrix}\right.$$

4. Conclusion

This paper discusses the parametric study of the rough nonlinear programming problems (RNLPP) when parameters exist on the crisp objective function and roughness in the feasible region. Also, the paper addresses new concepts including the solvability set, the stability set of the first kind and fifth kind on rough environment. The surely- and possibly solutions sets of RNLPP in the 1st Class are characterized. Future studies may focus on the parametric study of the PNLPP when roughness and parameters exist on both the objective function and the feasible region.

References

[1] Baoding, L. (2002). Theory and Practice of Uncertain Programming. (1st ed). Physica-Verlag, Heidelberg.

[2] Bazaraa, M. S., Sherali, H., &Shetty, C. M. (2006). Nonlinear Programming: Theory and Algorithms. (3rd ed). Wiley-Inter science.

[3] Feng, L., Li, T., Ruan, D., and Gou, .S. (2011). A vague-rough set approach for uncertain knowledge acquisition. Knowledge-Based Systems 24. (Pages 837–843).

[4] Osman, M. S., Lashein, E. F., Youness, E.A, and Atteya, T. E. M. (2011). Mathematical Programming in Rough Environment, Optimization: A Journal of Mathematical Programming and Operations Research, Volume 60. Issue 5 (Pages 603 – 611).

[5] Osman. M. S. A.,(1977) Qualitative analysis of basic notions in parametric convex programming I (Parameters in the constraints Applications of Mathematics, Vol. 22 , No. 5, 318—332.

[6] Osman. M. S. A., (1977). Qualitative analysis of basic notions in parametric convex programming. II (Parameters in the objective function) Applications of Mathematics, Vol. 22, No. 5, 333—348.

[7] Pawlak, Z. (1982). Rough sets, International Journal of Computer and Information Sciences (5) (Pages 341–356).

[8] Pawlak, Z. (1991). Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Boston.

[9] Pawlak, Z. (1996). Rough Sets, Rough Relations and Rough Functions. Fundamental Inform. 27(2/3) (Pages 103-108).

[10] Pawlak, Z. (1997). Vagueness: A rough set view. Structures in Logic and Computer Science. Springer (Pages 106-117).

[11] Youness, E.A. (2006). Characterizing solutions of rough programming problems. European Journal of Operational Research 168(3) (Pages 1019-1029).

جامعة القاهرة

معهد الدراسات والبحوث الأحصائية

المؤتمر السنوى السابع والأربعون

للإحصاء

وعلوم الحاسب

وبحوث العمليات

بحـــوث عمليـــات

٢٤-٢٧ ديسمبر ٢٠١٢