New Algorithm to Solve Convex Separable Programming

[1]A.Chikhaoui, [2]B. Djebbar , [3]A. Mokhtari, [3]A.Belabbaci
[1]University of Tiaret Algeria, [2]University of USTO Oran Algeria, [3]University of Laghouat Algeria,
ah_chikhaoui@yahoo.fr, badj2000@yahoo.fr, aek_mokhtari@yahoo.fr, A_Belabbaci@yahoo.fr.

Abstract: Separable programming is very useful for solving problems of nonlinear programming. In this paper we propose a new algorithm for solving problems of nonlinear programming separable. We approximate the nonlinear problem by a polynomial of degree two, we use a quadratic programming algorithm to find the optimal solution.
Keyword: **global optimization, piecewise quadratic function, separable programming.**

Introduction

SEPARABLE PROGRAMMING is a special class of nonlinearly constrained optimization problems whose objective and constraint functions are sums of functions of one variable. Separable programming problems are usually solved by linear programming techniques (Hillier and Lieberman, 2001). A separable programming (SP) problem whose objective and constraint functions are sums of functions of one variable (Gill et al., 1981). The SP problem can be solved efficiently by linear optimization techniques. The flow interaction among wells can play an important role in some rate allocation problems. In such cases, the rate allocation problem is formulated as a general nonlinear constrained optimization problem and solved by a Sequential Quadratic Programming method (Gill et al., 2002). Separable linear programming is a method for solving nonlinear problems by using the simplex algorithm employed in linear programming.

Its use in agricultural economics is illustrated by the Blakley and Kloth study
of plant location and the Holland and Baritelle study of school location. However, a shortcoming of separable linear programming is the risk of not obtaining a global optimum solution. Neither of the above studies reported information on the likelihood of having obtained non-global solutions. While this problem is reasonably well documented in literature on quantitative methods, it is examined and illustrated in the following discussion to help assure the proper use of separable programming in applied research.

Problem statement

Let's consider the general nonlinear programming problem:

$$(P_f)=\begin{cases} Minimize\ f(x) \\ g_i(x)\le b_i \\ i=1,\dots,m \end{cases}$$

with two additional provisions: 1) the objective function and all constraints are separable, and 2) each decision variable x_j is bounded below by 0 and above by a known constant $u_j,\ j=1,\dots,n.$

Recall that a function, $f(x)$, is separable if it can be expressed as the sum of function of the individual decision variables.

$$f(x)=\sum_{j=1}^{n} f_j(x_j).$$

The separable nonlinear programming problem has the following structure.

$$f(x)=\sum_{j=1}^{n} f_j(x_j)$$

$$\text{subject to } \sum_{j=1}^{n} g_{ij}(x_j)\le b_i,\ i=1,\dots,m.$$

$$0\le x_j\le u_{j,}\quad j=1,\dots,n$$

The key advantage of this formulation is that the nonlinearities are mathematically independent. This property in conjunction with the finite bounds on the decision variable permits the development of a piecewise quadratic approximation for each function in the problem.

Consider the general nonlinear function $f_j(x)$ defined on the interval $[a,b]$; and let $a=x_1,\dots,x_n=b$ a subdivision of $[a,b]$ with step $h=x_{i+1}-x_i$, n odd.

On every interval $[x_i,x_{i+2}]$, we replace the function f_j with a polynomial of two degree.

Notations

Let $(x_i)_{i=1,2,\ldots,n_1}$ subdivision of $[a_1, b_1]$, n_1 odd,

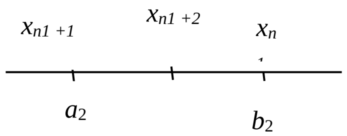
$$x_1 = a_1$$
$$x_2 = a_1 + h_1$$
$$\vdots$$
$$x_{n_1} = a_1 + n_1 h_1 \quad where \quad h_1 = \frac{b_1 - a_1}{n_1}$$

and let $(x_{n_1+i})_{i=1,2,\ldots,n_2}$ subdivision of $[a_2, b_2]$, n_2 odd,

$$x_{n_1+1} = a_2$$
$$x_{n_1+2} = a_1 + h_1$$
$$\vdots$$
$$x_{n_1+n_2} = a_2 + n_2 h_2 \quad where \quad h_2 = \frac{b_2 - a_2}{n_2}$$

For $x_1 \le x \le x_3$, put $x = x_1 + t_1 h_1$, where $t_1 \in [0,2]$, or $t_1 = \frac{x - x_1}{h_1}$.

Generally for $x_{2i-1} \le x \le x_{2i+1}$, $t_i = \frac{x - x_{2i-1}}{h_1}$, $1 \le i \le \frac{n_1 - 1}{2}$.



Interpolation of the function .

Set $f(y_1, y_2) = \varphi_1(y_1) - \varphi_2(y_2)$ with $a_1 \le y_1 \le b_1$, $a_2 \le y_2 \le b_2$

a) Interpolation of the function φ_1.

If $x_1 \le x \le x_3$, the function φ_1 is replaced by the Newton polynomial of degree two noted :

$$P_2(x) = \varphi_1(x_1) + \frac{t_1}{1!}\Delta\varphi_1(x_1) + \frac{t_1(t_1 - 1)}{2!}\Delta^2\varphi_1(x_1),$$

the polynomial can be calculated from the following finite differences table.

x	$\varphi_1(x)$	$\Delta\varphi_1(x)$	$\Delta^2\varphi_1(x)$
x_1			
x_2			
$\vdots$			
x_{n_1}			

With $t_1 = \frac{x - x_1}{h_1}$, set then

$$\psi_1(t_1) = P_2(x) - \varphi_1(x_1) = \alpha_1 t_1 + \beta_1 t_1^2$$

where $\alpha_1 = \Delta\varphi_1(x_1) - \frac{1}{2}\Delta^2\varphi_1(x_1)$ and $\beta_1 = \frac{1}{2}\Delta^2\varphi_1(x_1)$

In general, for $x_{2i-1} \le x \le x_{2i+1}$,

$$P_2(x) = \varphi_1(x_{2i-1}) + \frac{t_i}{1!}\Delta\varphi_1(x_{2i-1}) + \frac{t_i(t_i - 1)}{2!}\Delta^2\varphi_1(x_{2i-1})$$

With $t_i = \frac{x - x_i}{h_1}$, put then

$$\psi_i(t_i) = P_2(x) - \varphi_1(x_{2i-1}) = \alpha_i t_i + \beta_i t_i^2$$

$i = 1,2,\cdots,\frac{n_1 - 1}{2}$.

The study of the optimum of the function ψ defined by

$$\psi\left(t_1, t_2, \cdots, t_{\frac{n_1-1}{2}}\right) = \sum_{i=1}^{\frac{n_1-1}{2}} \psi_i(t_i)$$

replace then the study of the optimum of the function φ_1 on the interval $[a_1, b_1]$. We add the supplementary condition: **one and one only t_i is positive**.

In fact, the linear constraints are written $0 \le t_i \le 2$, furthermore, if $t_{i_0} \in]0,2[$ and $t_i = 0$ for all $i = 1,2,\ldots,\frac{n-1}{2}$, $i \ne i_0$ then

$$\psi\left(t_1, t_2, \ldots, t_{\frac{n_1-1}{2}}\right) = \alpha_{i_0} t_{i_0} + \beta_{i_0} t_{i_0}^2 = \psi_{i_0}(t_{i_0}),$$
$$= P_2(x) - \varphi_1(x_{2i_0-1})$$

Consequently

$$P_2(x) = \psi\left(t_1, t_2, \ldots, t_{\frac{n_1-1}{2}}\right) + \varphi_1(x_{2i_0-1});$$

and we see that the optimum of P_2 is that of ψ

b) **Interpolation of the function φ_2:**

In the same manner as in part a) and for $a_2 \le x \le b_2$, we set $y_2 = x$, , $\Delta^n\varphi_2(x_{n_1+2i-1}) = \Delta(\Delta^{n-1}\varphi_2(x_{n_1+2i-1}))$, ; if

$x_{n_1+1} \le x \le x_{n_1+3}$, the function φ_2 is replaced by the Newton polynomial of degree two noted :

$$P_2(x) = \varphi_2(x_{n_1+1}) + \frac{t_{\frac{n_1-1}{2}+1}}{1!}\Delta\varphi_2(x_{n_1+1}) ,$$
$$+ \frac{\left(t_{\frac{n_1-1}{2}+1}\right)\left(t_{\frac{n_1+1}{2}} - 1\right)}{2!}\Delta^2\varphi_2(x_{n_1+1})$$

the polynomial can be calculated from the finite differences table.

With $t_{\frac{n_1+1}{2}} = \frac{x - x_{n_1+1}}{h_2}$, we set

$$\psi_2\left(t_{\frac{n_1+1}{2}}\right) = P_2(x) - \varphi_2(x_{n_1+1})$$

where

$$\alpha_{\frac{n_1+1}{2}} = \Delta\varphi_2(x_{n_1+1}) - \frac{1}{2}\Delta^2\varphi_2(x_{n_1+1}) \text{ and}$$

$$\beta_{\frac{n_1+1}{2}} = \frac{1}{2}\Delta^2\varphi_2(x_{n_1+1})$$

Generally, for $x_{n_1+2i-1} \le x \le x_{n_1+2i+1}$,

$$P_2(x) = \varphi_2(x_{n_1+2i-1}) + \frac{t_{\frac{n_1-1}{2}+i}}{1!}\Delta\varphi_2(x_{n_1+2i-1})$$
$$+ \frac{t_{\frac{n_1-1}{2}+i}\left(t_{\frac{n_1-1}{2}+i} - 1\right)}{2!}\Delta^2\varphi_2(x_{n_1+2i-1})$$

With $t_{\frac{n_1-1}{2}+i} = \frac{x - x_{n_1+2i-1}}{h_2}$, put then

$$\psi_i\left(t_{\frac{n_1-1}{2}+i}\right) = P_2(x) - \varphi_2(x_{n_1+2i-1})$$
$$= \alpha_{\frac{n_1-1}{2}+i}t_{\frac{n_1-1}{2}+i} + \beta_{\frac{n_1-1}{2}+i}t^2_{\frac{n_1-1}{2}+i}$$

where

$$\alpha_{\frac{n_1-1}{2}+i} = \Delta\varphi_2(x_{n_1+2i-1}) - \frac{1}{2}\Delta^2\varphi_2(x_{n_1+2i-1})$$

, $\beta_{\frac{n_1-1}{2}+i} = \frac{1}{2}\Delta^2\varphi_2(x_{n_1+2i-1})$ and $i = 1,2,\cdots,\frac{n_2-1}{2}$.

The study of the optimum of the function ψ_2 defined by

$$\psi_2\left(t_{\frac{n_1-1}{2}+1}, t_{\frac{n_1-1}{2}+2}, \cdots, t_{\frac{n_1-1}{2}+\frac{n_2-1}{2}}\right) = \sum_{i=1}^{\frac{n_2-1}{2}}\psi_i\left(t_{\frac{n_1-1}{2}+i}\right)$$

replace then the study of the optimum of the function φ_2 on the interval $[a_2, b_2]$. Add the supplementary condition :

one and one only $t_{\frac{n_1-1}{2}+i}$ is positive.

In fact, the linear constraints are written $0 \le t_{\frac{n_1-1}{2}+i} \le 2$. Furthermore, if $t_{i_0} \in]0,2[$ and $t_{\frac{n_1-1}{2}+i} = 0$ for all $i = 1,2,\ldots,\frac{n_2-1}{2},\ i \ne i_0$

then:

$$\psi_2\left(t_{\frac{n_1-1}{2}+1}, t_{\frac{n_1-1}{2}+2}, \cdots, t_{\frac{n_1-1}{2}+\frac{n_2-1}{2}}\right) = \alpha_{i_0}t_{i_0} + \beta_{i_0}t_{i_0}^2 = \psi_{i_0}(t_{i_0}) .$$
$$= P_2(x) - \varphi_2(x_{n_1+2i_0-1})$$

Consequently

$$P_2(x) = \psi_2\left(t_{\frac{n_1-1}{2}+1}, t_{\frac{n_1-1}{2}+2}, \cdots, t_{\frac{n_1-1}{2}+\frac{n_2-1}{2}}\right) + \varphi_2(x_{n_1+2i_0-1}).$$

And we see that the optimum of P_2 is that of ψ_2

When the objective function is not quadratic, replace then problem (P) with the problem (P') deduce from (P) as following :

replace the function φ_1 *by* ψ_1

and replace the function φ_2 *by* ψ_2

i.e.

$$(P')\begin{cases} \psi\left(t_1, t_2, \cdots, t_{n_1}, \cdots, t_{\frac{n_1-1}{2}+\frac{n_2-1}{2}}\right) = \sum_{i=1}^{\frac{n_1+n_1}{2}-1}\alpha_i t_i + \beta_i t_i^2 \\ 0 \le t_i \le 2 \ ;\ 1 \le i < \frac{n_1-1}{2} + \frac{n_2-1}{2} \\ \text{is at most one } t_i \text{ is nonzero in each} \\ \text{of the choosed subdivision.} \end{cases}$$

The calculus of y_i is given by the formulas :

$$y_1 = x_{2i-1} + h_1 t_i \quad \text{if } 1 \le i \le \frac{n_1-1}{2}$$

$$y_2 = x_{n_1+2i-1} + h_2 t_{n_1+i} \quad \text{if } 1 \le i \le \frac{n_2-1}{2}$$

Algorithm:

Solving **separable programming problem** into two parts:
1.expression each function,
2.approximation interval and step,
3.approximat each function by the method of finite differences
4. With this approximation, we construct the quadratic program for each function
5.solve quadratic program associated to each function. We obtain value of component x_j^* and the approximate value of the function $f_j\left(x_j^*\right)$.
6. Go to 1.

Algorithm separable :
Data : number_variable_separable, number_constrainte_separable,
Matrix_A_constraintes, vector_b

```
// input of express functions and bororne inf, borne sup, step.
For j=1: number_variable_separable
  Txt= input(' expression of the jeme function');
   Express_fun(j,1:length(txt))=txt;
  Txt_param = input( 'lower bound,upper bound, step');
  param_fun(j,1:3)=eval(['[' txt_param ']']);
end

// quadratic interpolation of functions
for k=1:n_variables_sep

x=linspace(param_fun(k,1),param_fun(k,2),param_fun(k,3));
   y=eval(express_fun(k,:));
   x_values(k,1:length(x))=x;
   func_values(k,1:(length(y)))= y;
   D1=diff(y) ;    D2=diff(D1) ;
   ndif2=length(D2);     ne=(ndif2+1)/2 ;
   alpha_f(k,1:ne+1)=[ ne D1(1:2:ndif2)-0.5*D2(1:2:ndif2)] ;
   beta_f(k,1:ne+1)=[ne 0.5*D2(1:2:ndif2)] ;
end

// solve quadratic programming problems
 for indice_func  = 1:n_variables_sep
   algorithm_qp(alpha_f, beta_f);
 end
Results = x_optimale_value, f_optimal_value.
```

algorithm_qp($\alpha,\ \beta$)

Begin Algorithm Initialization: vectors $\alpha,\ \beta,\ b$ and matrix $A,\ \delta$ Z Z=Z$_0$, A_positif = true;

While (A_positif = true) do

For all indexes j :

Calculate $\theta_j = \min\limits_i \left\{ \dfrac{b_i}{a_{ij}},\ a_{ij} > 0 \right\}$,

For all indexes j : Calculate $\Delta_j = \alpha_j \theta_j + \beta_j \theta_j$.

Choose $\Delta_{j_0} = \max\limits_j \Delta_j$.

if $\Delta_{j_0} = +\infty$ then STOP : this program don't have optimum.

if $\Delta_{j_0} \le 0$ then STOP : this program is optimal.

Let $z = z + \Delta_{j_0}$, x_{j_0} is entering basic vector.

$\theta_0 = \min\limits_i \left\{ \dfrac{b_i}{a_{ij_0}},\ a_{ij_0} > 0 \right\}$, x_{i_0} is leaving basic vector. $a_{i_0 j_0}$ is the pivot.

For all indexes j : if $j <> j_0$ then

$$\alpha_j^{'} = \alpha_j - \left(\alpha_j + 2\beta_j \theta\right)\frac{a_{i_0 j}}{a_{i_0 j_0}} + \frac{b_{i_0}}{a_{i_0 j_0}}\left(\delta_{j_0 j} + \delta_{jj_0}\right)$$
;

else $\alpha_{j_0}^{'} = 0 ; \beta_{j_0}^{'} = 0$ endif

For all indexes i For all indexes j
if $\left(i <> i_0 \right)$

$$a_{ij}^{'} = a_{ij} - \frac{a_{ij}}{a_{i_0 j_0}} a_{i_0 j} \ \ ; \ b_i^{'} = b_i - \frac{a_{ij_0}}{a_{ij}} b_{i_0} .$$

endif endfor endfor.

A_positif = false;
For all indexes i For all indexes j :
if $a_{ij} > 0$ then A_positif = true
endif endfor endfor
 if A_positif = false
 this program do not have optimum Stop.
 endif
endWhile.

Example Let the function $f_j(x) = x - Log\, x$, defined on $\left[\dfrac{1}{2}, \dfrac{5}{2} \right]$, to maximize. We use the two degree polynomial of Newton with step $h = 0.5$ to interpolate.

x	0.5	1	1.5	2	2.5
$y = f_j\left(x\right)$	1.19	1	1.09	1.31	1.58

Calculate $\psi_1(t_1)$.

x	y	Δy	$\Delta^2 y$
0.5	1.19	-0.19	0.28
1	1	0.09	
1.5	1.09		

To use the polynomial of Newton, we find:
$\alpha_1 = -0.33 \quad \beta_1 = 0.14$ and $\psi_1(t_1) = -0.33t_1 + 0.14t_1^2$.

1. Calculate $\psi_2(t_2)$.

x	y	Δy	$\Delta^2 y$
1.5	1.09	0.22	0.05
2	1.31	0.27	
2.5	1.58		

$\alpha_2 = -0.20 \quad \beta_2 = 0.03$ and $\psi_2(t_2) = 0.20t_2 + 0.03t_2^2$.
Consequently $\psi(t) = -0.33t_1 + 0.20t_2 + 0.14t_1^2 + 0.03t_2^2$ that we maximize.
We use the method describe in [10] to resolve this problem. Recall the expression of θ and Δ (see [9]).

$$\theta_j = \min_{a_{kj}} \left\{ \frac{b_k}{a_{kj}}, \quad for \quad the \quad a_{kj} > 0 \right\} \quad and$$

$$\Delta_j = \alpha_j \theta_j + \beta_j \theta_j^2$$

t_1	t_2	
-0.33	0.20	α
0.14	0.03	β
2	2	θ
-0.10	0.52	Δ
1	0	$t_3 = 2$
0	1	$t_4 = 2$

t_2 is entering variable and it replaces t_4 in the base. More, $t_2 = 2$ and $t_1 = 0$.

The maximum of the function $\psi(t) = -0.33t_1 + 0.20t_2 + 0.14t_1^2 + 0.03t_2^2$ is given by $t_2 = 0 \ and \ t_1 = 0$.
The optimum of this function equal 0.5.
The maximum of the objective function is the in the point $x = x_3 + 2h$ i.e. in the point with abscise $x = 2.5$, this maximum is equal 1.58.

The maximum of $P_2(x)$ is equal $0.52 + f_j(x_3) = 1.61$ who is near the real value of this maximum.

Note that is important to find only the value of t_j for which the function $\psi(t) = -0.33t_1 + 0.20t_2 + 0.14t_1^2 + 0.03t_2^2$ is optimal.

We say that the objective function is optimal in the point $x = x_{2j-1} + t_j h$.
The maximum of the objective function is calculating immediately.

Results and discussion

- It is possible to solve large nonlinear separable problems with the quadratic separable programming,
- We used an approximation of order two which is more accurate than the first order approximation used in the linear approximation to apply simplex procedure.
- It is possible to approximate constraints by similar procedure.
- To get more accurate result, the piecewise quadratic approximation *fi* can be refined.

REFERENCES

[1] Blakley, Leo. V. and Donald W. Kloth, "Price Alignment and Movements of Class 1 Milk Between Markets", *American Journal of Agricultural Economics*, 54(1972): 496-502.

[2] Chikhaoui A, Djebbar B., Bellabacci A, Mokhtari A., 2009. Optimization of a quadratic function under its Canonical form *Asian Journal of Applied Sciences* 2(6):499-510

[3] Chikhaoui A, Djebbar B., and Mekki R., 2010. New Method for Finding an Optimal Solution to Quadratic Programming Problem, *Journal of Applied Sciences* 10(15):1627- 1631-2010. ISSN 1812-5654.

[4] Chikhaoui A, Djebbar B., Bellabacci A, Mokhtari A.,2011. An Algorithm for calculating optimal Solution of Quadratic Programming problems. Not published yet.

[5] Stephen B. and Lieven V., 2004. Convex Optimization, *Cambridge University Press*.

[6] HILLIER, F.S., AND LIEBERMAN, G.J., 2001. *Introduction to Operations Research*, *7th Edition*, McCraw-Hill Companies, Inc., New York, NY,

[7] *Holland, David W. and John L. Baritelle, "School Consolidation in Sparsely Populated Rural Areas: A Separable Programming Approach", American Journal of Agricultural Economics, 57(1975): 567-575.*

[8] *GILL, P.E., MURRAY, W., AND SAUNDERS, M.A., 2002. "SNOPT: An SQP Algorithm for Large-scale Constrained Optimization", SIAM J. Optimization, Vol. 19, No. 4, pp. 979-1006.*

[9] PADBERG, M., 2000. "Approximating Separable Nonlinear Functions Via Mixed Zero-One Programs", *Operations Research Letters*, 27 (200), 1-5. S., 1984, Programmation linéaire. Traduction française. Edition MIR. Moscou.