Decomposition methods in optimization and a novel application in agriculture for zoning management

Víctor M. Albornoz
Departamento de Industrias
Campus Santiago Vitacura
Universidad Técnica Federico Santa María, Chile

Euro Summer Institute 2014
OR in Agriculture and Food Industry
Universidad de Lleida, 29 de Julio de 2014

Agenda

1. Introduction.

2. The column generation method.

3. An optimization model for determining agricultural management zones.

4. Conclusions.

1. Introduction.

There are many problems in economics, engineering, operations management and mathematics that give rise to highly structured large optimization models.

Decomposition methods provide a numerical optimization technique to solve them.

In the case of linear models, the matrix that represent the different equations ($Ax=b$) could be an sparse matrix where nonzero elements follow a given pattern.

For example, A could be a block angular matrix:

$$\begin{bmatrix} A_1 & A_2 & A_3 & \cdots & A_q \\ D_1 & & & & \\ & D_2 & & & \\ & & D_3 & & \\ & & & \ddots & \\ & & & & D_q \end{bmatrix}$$

a dual block angular matrix:

$$\begin{bmatrix} D_1 & & & & & F_1 \\ & D_2 & & & & F_2 \\ & & D_3 & & & F_3 \\ & & & \ddots & & \\ & & & & D_q & F_q \end{bmatrix}$$

or an staircase matrix:

$$\begin{bmatrix} 1 & & & & \\ 1 & 1 & & \mathbf{0} & \\ & 1 & \ddots & & \\ & & \ddots & 1 & \\ \mathbf{0} & & & 1 & 1 \end{bmatrix}$$

The dual block angular form appears in two-stage linear stochastic program with recourse (for a finite number of S scenarios):

$$
\begin{array}{lllllll}
\text{Min} & cx + p_1 q^1 y^1 + p_2 q^2 y^2 + \cdots + p_S q^S y^S & & & & & \\
\text{s.a.} & & & & & & \\
& Ax & & & & & = b \\
& T^1 x & + Wy^1 & & & & = h^1 \\
& T^2 x & & + Wy^2 & & & = h^2 \\
& \dots & & & & & \\
& T^S x & & & & + Wy^S & = h^S \\
& x \geq 0, & y^1 \geq 0, & y^2 \geq 0, & \ldots & y^S \geq 0. &
\end{array}
$$

For example, the capacity expansion of a themal power system solved by Albornoz et al. (2004) gives raise to this kind of model and particular structure.

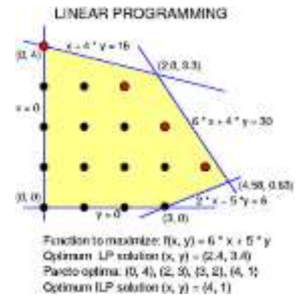
In Albornoz and Canales (2006), we use a *Lagrangean decomposition* algorithm to solve a two-stage stochastic nonlinear program that provides a total allowable catch quota for managing a particular fishery resource.

In Rodriguez et al. (2009) we proposed a two-stage stochastic linear optimization model for determining an optimal purchase and replacement policy in a sow farm.

There exist highly structure large scale optimization problems that contribute to obtain an optimal solution to the decision maker.

To solve these optimization models, the decomposition methods allow to tackle them in an efficient way.

The seminal works in numerical optimization that exploit these type of structures started to appear around fifty years ago.



Dantzig & Wolfe Decomposition (1960) .

Benders Decomposition (1962).

Decomposition method of Rosen (1964).

Kaul’s algorithm (1965).

L-Shaped decomposition method of van Slyke & Wets (1969).

Generalized Benders Decomposition of Geoffrion (1972).

Simplicial decomposition of von Hohembalken (1977).

Cross Decomposition of van Roy (1983).

Horizontal Decomposition of Meijboom (1986).

Lagrangean decomposition of Guignard and Kim (1987).

Local Decomposition of van de Panne (1987).

Nested Decomposition, Robinson (1989).

Stochastic Decomposition, Higle and Sen (1991)

Lagrangean Decomposition, Michelon and Maculan (1991)

Column Generation in IP, Vanderveck and Wolsey (1996)

L-Shaped Method for IPSP, Caroe and Tind (1998),

Branch-and-price, Vanderveck (2000)

Benders and BFC for 0-1 mixed SP, Escudero et al. (2007).

Two-Stage Column Generation, Salani and Vacca (2008).

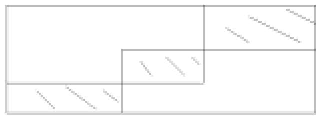
Stochastic Scenario Decomposition (MSSP), Higle et al. (2010)

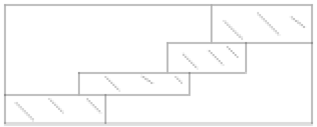
Vertical Decomposition method, Verma et al. (2011).

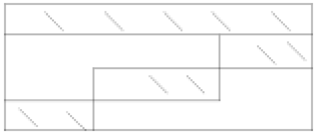
Cluster Benders Decomposition, Aramburu et al. (2012).

To solve a complex problem by a decomposition method, the basic idea is:

decompose or simplify the resolution of the original problem by solving a (reduced) master problem and one subproblem, this last one usually with a set of constraints that have a special structure.







To implement such idea in an algorithmic framework, the strategy consist on sending information, such as dual prices and a feasible solution, between the *reduced master problem* and the *subproblem* until to reach the optimal solution in a finite number of iterations.

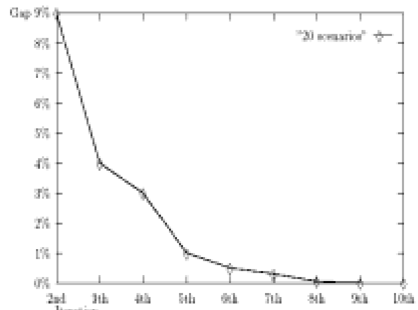


Fig 1. Algorithm behavior.

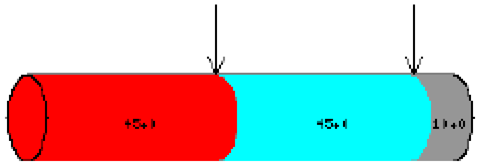
These ideas can also contribute to solve problems in agriculture and food industry as we will see in the application of a column generation method for the management of a zone delineation problem.

1. Introduction.
2. **The column generation method.**
3. An optimization model for determining agricultural management zones.
4. Conclusions.

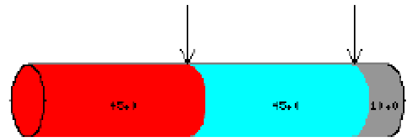
2. The column generation method.

The first application of a decomposition method was introduced by Gilmore y Gomory (1961,1963) to solve a cutting problem, know as:

The Cutting-Stock Problem:
An Application of Integer Linear Programming



The Cutting-Stock Problem:
An Application of Integer Linear Programming



Notation.

L = width of the rolls

m = the number of orders

w_i = width of pieces of order $i=1,\ldots,m$

b_i = total demand of order $i=1,\ldots,m$

a_{ij} = number of rolls of width w_i cut in pattern j, with $i=1,\ldots,m$, $j=1,\ldots,n$

$A=(a_{ij})_{i=1,\ldots,m\ j=1,\ldots,n}$ = matrix with the set of patterns

a^j = j-th column of A

Notice that a pattern j is feasible if and only if:

$$a_{1j}w_1 + a_{2j}w_2 + \cdots + a_{mj}w_m \leq L$$

If $\mathbf{x_j}$ represents the number of roll cuts using pattern $j=1,\ldots,n$, a model that minimizes the number of rolls cut subject to satisfy the demand requirement can be formulated as follows:

$$\begin{array}{ll} \text{Min} & \mathbf{x_1} + \mathbf{x_2} + \cdots + \mathbf{x_n} \\ \text{s.a.} & a_{i1}\mathbf{x_1} + a_{i2}\mathbf{x_2} + \cdots + a_{in}\mathbf{x_n} \geq b_i \qquad i=1,\ldots,m \\ & \mathbf{x_1} \geq 0, \mathbf{x_2} \geq 0, \ldots, \mathbf{x_n} \geq 0, \text{ integer} \end{array}$$

Is possible to solve the model using the column generation method, based on the idea that is not necessary to know all the possible patterns for finding an optimal solution of the LP relaxation of the problem.

The method starts with an initial set of m patterns that provides a basic feasible solution. For example, we can consider a pattern j:

$$a_{ij}=\lfloor L/w_i \rfloor \text{ if } i=j \text{ and } 0 \text{ otherwise, for } j=1,\dots,m$$

Denoting by B the matrix that corresponds to the given patterns, we can know if this basic feasible solution is optimal iff the reduced cost satisfies:

$$1 - c_B^T B^{-1} a^j = 1 - \boldsymbol{\lambda}^T a^j \geq 0$$

for each possible pattern j, which is also equivalent to verify:

$$\min_{j=1,\dots,n} \{1 - \boldsymbol{\lambda}^T a^j\} \geq 0$$

or if and only if $\max_{j=1,\dots,n} \{\lambda^T a^j\} \leq 1$

Fortunately, it is not necessary to know all the patterns to verify the previous condition because by solving the following *Suproblem*:

$$\begin{array}{ll} \text{Max} & \lambda^T a \\ \text{s.a.} & a_1 w_1 + a_2 w_2 + \cdots + a_m w_m \le L \\ & a \ge 0, \text{ integer.} \end{array}$$

If the optimal pattern a satisfies $\lambda^T a \le 1$, then it will be satisfy for all them and we get the optimal solution of the problem. Otherwise, we add a new pattern that will define a new b.f.s.

Modeling languages and optimization systems.

They allow to use common notation and familiar concepts to develop an optimization model and algorithms.

In an easy way you can read parameters, communicate with an appropriate solver, examine solutions and/or write programs for running such algorithms.

A list of available modeling languages:

AIMMS

AMPL, www.ampl.com

GAMS

LINGO

MPL

OPL Studio

PLAM

Implementation in AMPL

www.ampl.com

INDEX TO EXAMPLES

LOOPING AND TESTING 2:
implementing algorithms through AMPL scripts

Script	Uses	Implements
cut1.run	cut1.mod cut.dat	Gilmore-Gomory column generation procedure for the cutting-stock (roll trim) problem
cut2.run	cut2.mod cut.dat	Same as cut1.run, but using an alternative arrangement wherein problems are defined immediately before before their members are declared
cut3.run	cut1.mod cut.dat	Same as cut1.run, but with better formatting of output
multi1.run	multi1.mod multi1.dat	Dantzig-Wolfe decomposition for a multi-commodity transportation problem, using a single subproblem
multi1a.run	multi1.mod multi1.dat	Same as multi1.run, but using the same repeat loop for both phase I (infeasible) and phase II (feasible).
multi2.run	multi2.mod multi2.dat	Same as multi1.run, but using a separate subproblem for each product; subproblems are represented in AMPL by an indexed collection of named problems
multi3.run	multi3.mod multi3.dat	Same as multi2.run, except that the separate subproblems are realized by changing the data to a single AMPL named problem
stoch1.run	stoch1.mod stoch.dat	Benders decomposition for a stochastic programming variant of a multi-period production problem (see Exercise 4-5)
stoch2.run	stoch2.mod stoch.dat	Same as stoch1.run, but using a separate subproblem for each scenario; subproblems are represented in AMPL by an indexed collection of named problems
stoch3.run	stoch3.mod stoch.dat	Same as stoch2.run, except that the separate subproblems are realized by changing the data to a single AMPL named problem
trnloc1.run revised!	trnloc1.mod trnloc.dat	Benders decomposition for a location-transportation problem (original model in trnloc.mod)
trnloc2a.run	trnloc2a.mod trnloc2.dat	Lagrangian relaxation for a location-transportation problem: LP relaxation bound is poor, and subproblem has the integrality property so no improvement can be made
trnloc2b.run	trnloc2b.mod trnloc2.dat	Same as trnloc2a.run, but model has upper limits on the Ship variables: LP relaxation bound is still poor, but subproblem does *not* have the integrality property and considerable improvement is made
trnloc2c.run	trnloc2c.mod trnloc2.dat	Same as trnloc2b.run, but model has 0-1 constraints disaggregated: LP relaxation bound is good, but subproblem has the integrality property and no

NEOS server for optimization

http://www.neos-server.org/neos/

E. D. Dolan, R. Fourer, J. J. Moré, and T.S. Munson. *Optimization on the NEOS Server*. *SIAM News*, Volume 35, Number 6, 2002.

1. Introduction.

2. Column generation method.

3. **An optimization model for determining agricultural management zones.**

4. Conclusions.

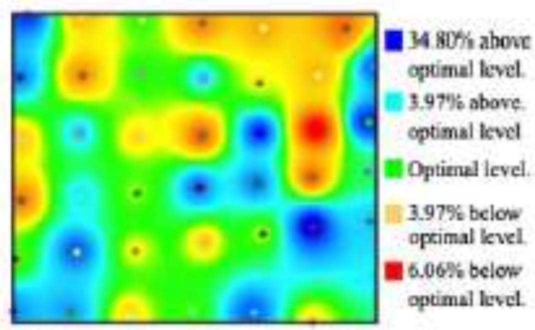
3. An optimization model for determining agricultural management zones.

In agriculture, the spatial variability of soil properties is one of the important aspects that determine productivity and crop quality.

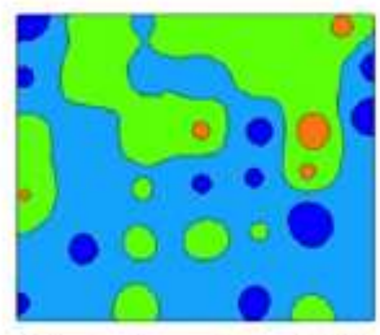
Delineating the field into site-specific management zones allows facing within-field variability to design proper crop management plans.

There are several approaches for properly determine site-specific management zones:

- use of topographic maps
- yield maps from data of several seasons
- clustering methods based on soil samples



Thematic map for the OM property (MapInfo)



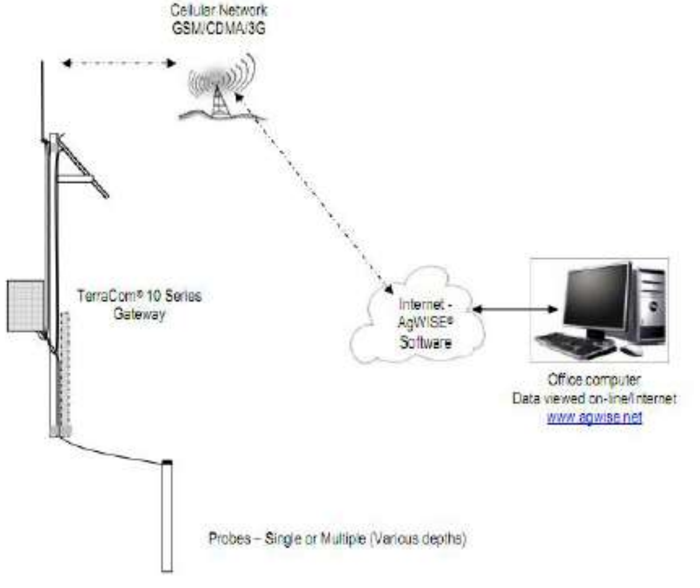
Clustering zoning method (MapInfo) for the OM property

In Cid-García et al. (2013), we present a new zoning method that optimally delineates rectangular homogeneous management zones, using relative variance as a measure of homogeneity (Ortega and Santibañez, 2007).

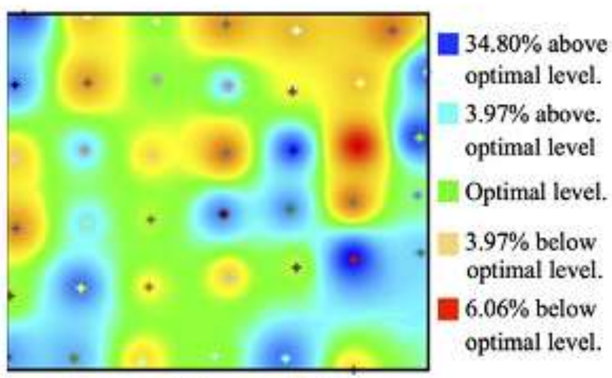
Homogeneous.

Heterogeneous.

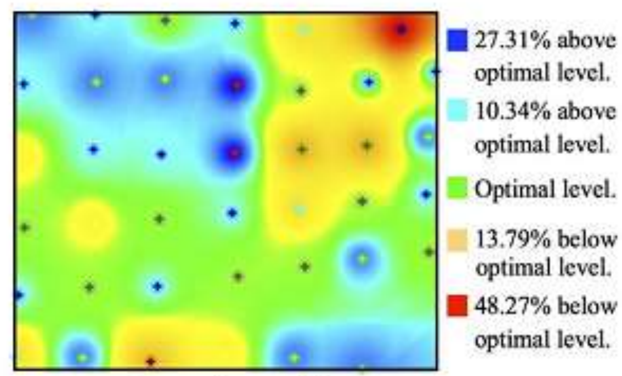
Soil samples are usually generated from a systematic grid sampling with the help of a software (SMS Mobile) and a GPS receiver. Sample positions are usually collected in geographic coordinates using a GIS system.



The visualization of these data, for a given property, is called a thematic map. The next figures show two examples of them.

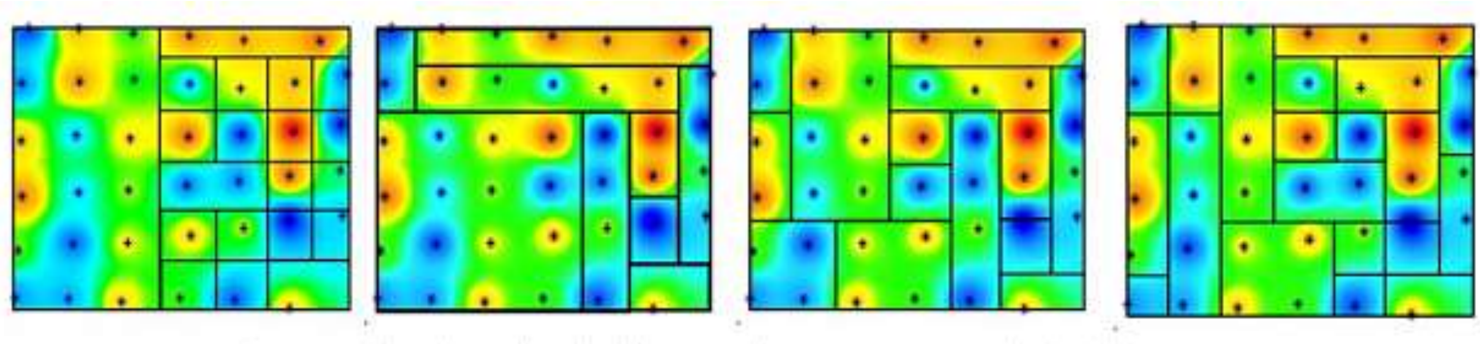


Organic Matter (OM)

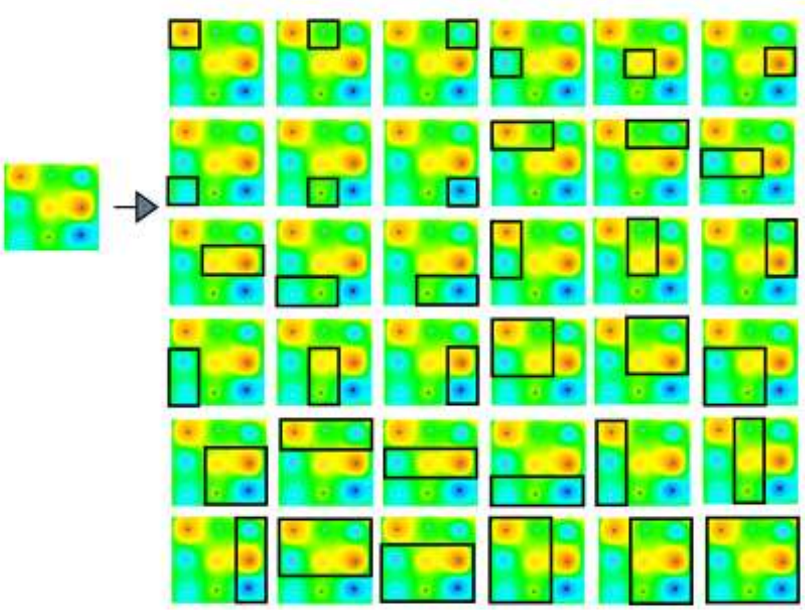


Phosphorus (P)

The proposed solution delineates the most homogeneous rectangular management zones from a field, with respect to one or more soil properties (converted to a soil quality index).

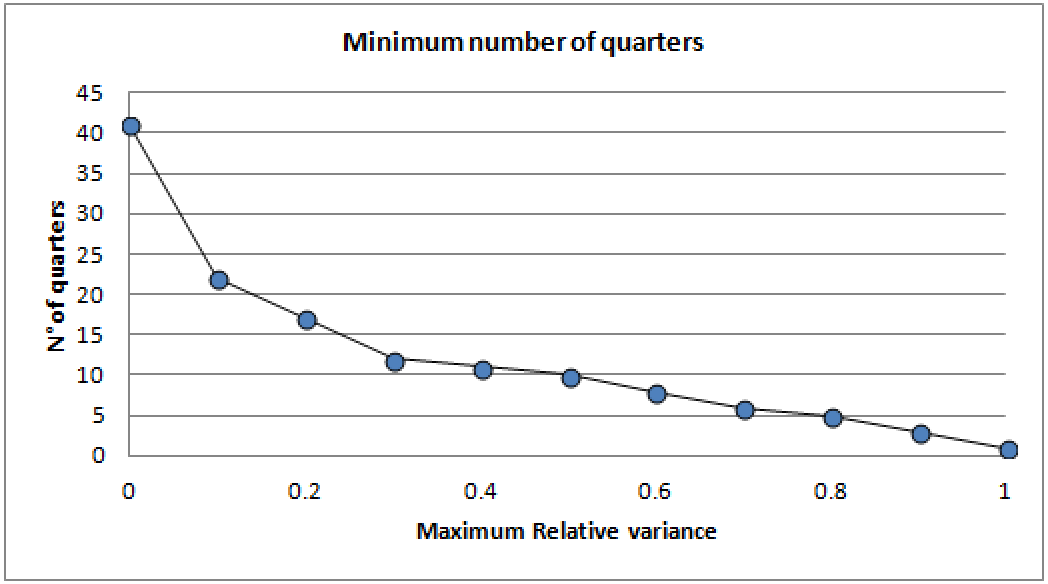


The method delineates rectangular management zones based on an integer programming model that provides a partition of the field, given a set the potential zones (or quarters) within it.

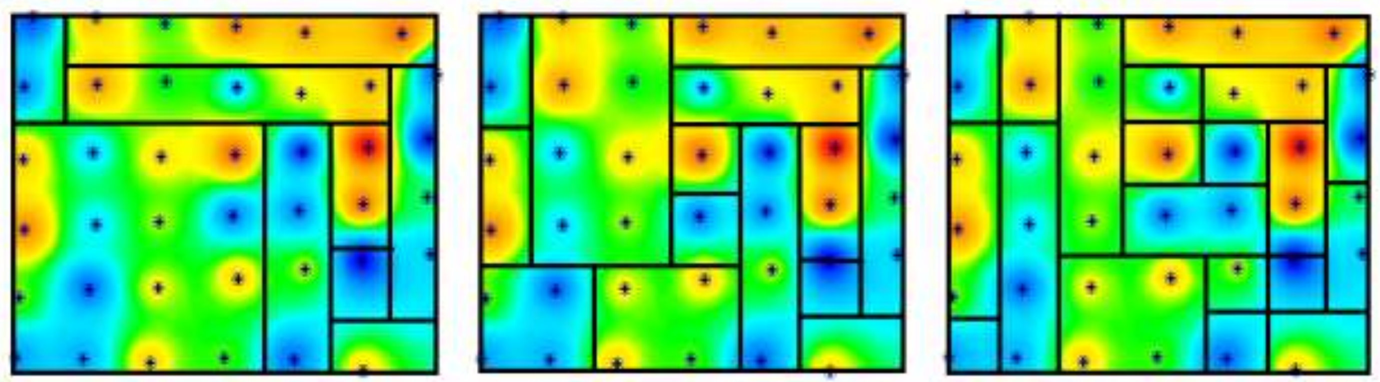


The measurement of the effectiveness of a zoning method is based on the concept or relative variance.

A lower relative variance implies a higher internal homogeneity for the zones in which the field is partitioned.



The relationship between relative variance ($1-\alpha$) and the resulting number of management zones, for the OM soil property:



$\alpha = 0.5$ $\alpha = 0.7$ $\alpha = 0.9$

The following notation was used:

M = the set of potential management zones

S = the set of soil samples in the field

σ_z^2 = the variance of management zone $z \in M$

n_z = the number of samples in the potential zone $z \in M$

$c_{zs}=1$ if zone z includes sample point s, $c_{zs}=0$ otherwise.

LS = a maximum number of zones in the partition

σ_T^2 = the variance of the field

$1-\alpha$ = upper bound for the relative variance

The decision variable of the model is:
q_z = 1 if the potencial management zone $z \in M$ *is part of the partition* and 0 otherwise,

The proposed model is as follows:

$$\min \sum_{z \in Z} q_z \tag{1}$$

$$\sum_{z \in Z} c_{zs} q_z = 1 \quad \forall s \in S \tag{2}$$

$$\sum_{z \in M} q_z \leq LS \tag{3}$$

$$\frac{\sum_{z \in Z} (n_z - 1)\, \sigma_z^2}{\sigma_T^2 \left[N - \sum_{z \in Z} q_z\right]} \leq (1 - \alpha) \tag{4}$$

$$q_z \in \{0, 1\} \quad \forall z \in M$$

The proposed model is an integer linear programming model because the last constraint can be easily put into a linear form.

A complete enumeration of the set of potential zones allows to find the optimal solution of the problem but this alternative is not efficient when we are facing large instances of the problem.

We solve a linear relaxation of model (1)-(4) based on a column generation strategy.

The resulting Subproblem provides a rectangular management zones according to a binary decision variable x_s that take the value 1 if sample point s belong the propose management zone and 0 otherwise.

Parameters

p_s: Value of the dual variable associated to the partitioning constraint (2).

ω: Value of the dual variable associated to constraint (3).

$coef_z$: Value of the dual variable associated to the Relative Variance constraint (4).

d_s: Sample point s value.

Compact Formulation

$$min 1 - \sum_{s \epsilon S} p_s x_s - \omega - coef_z \left[\left(n_z - 1 \right) \sigma_z^2 \left(x \right) + (1 - \alpha) \sigma_T^2 \right] \tag{5}$$

$$x_s \epsilon X \tag{6}$$

$$x_s \epsilon \{0, 1\}$$

We studied four different implementations of the algorithm based on the form in which we solve the subproblem. They can be classified according to their purpose:

Strategies I and II: Avoid solving the pricing problem to optimality. This is used as an initial approach to the optimal solution.

Strategies III and IV: Prove optimality for MP by solving the pricing problem to optimality.

Strategies I and II

We apply this algorithm to the initial RMP. At each iteration of the CG algorithm, we fix a **size** (determined by some number of adjacent rows and columns) for the quarters to be considered, and then price them by enumeration.

If a new potential quarter with negative reduced cost is found, we add it to the column pool, until a predefined maximum number of columns per iterations is reached.

Strategy I: potential quarters are explored in **ascending** size order.

Strategy II: potential quarters are explored in **descending** size order.

When all possible quarter sizes are covered, this algorithm stops.

Strategies III and IV

These strategies are applied once the algorithm associated to Strategy I or II has finished.

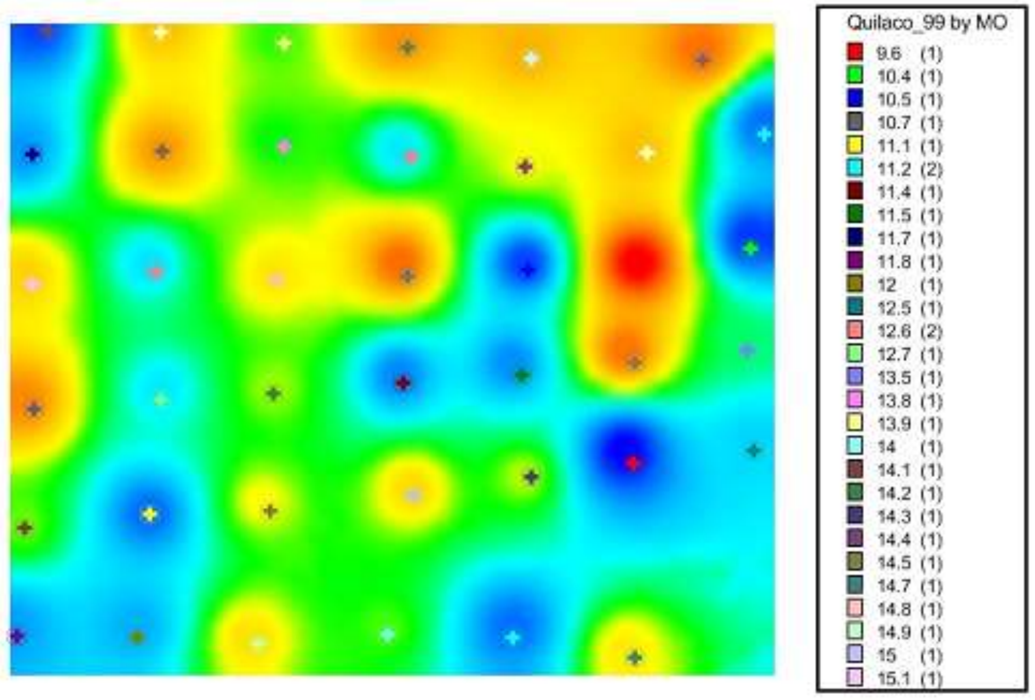
At each iteration of the CG algorithm, the pricing problem is solved to optimality.

Strategy III: the column with the lowest reduced cost is added to RMP.

Strategy IV: a set of columns is added to RMP, which includes the one with the lowest reduced cost.

We included an early termination criterion, which allows us to use the optimal solution of the pricing problem to compute a lower bound on the optimal solution of MP.

Computational experiences include 10 instances. The smaller one consider a vineyard close to Santiago of Chile (around 7.82 ha), that has a total of 42 soil samples.



The algorithm were coded in AMPL. The resulting linear and integer problems were solved with Cplex 12.4. An Intel core i5 of 2.5 GHz with 8 GB of RAM was used for this research.

Instance	N	K	Time[s]	Optimal Value
1	42	588	0.48	10
2	100	3,025	11.52	22
3	150	6,600	106.14	24
4	225	14,400	1,468.62	34
5	300	25,200	8,761.71	47
6	400	44,100	47,654.8	58
7	500	68,250	-	-
8	600	97,650	-	-
9	750	151,125	-	-
10	900	216,225	-	-

Results for Strategy I:

Instance	#col	UB	gap	IS	time[s]
1	148	9.07	0.02	10	0.64
2	401	23.59	0.13	24	2.12
3	609	34.61	0.46	35	5.41
4	1,109	36.88	0.11	37	25.63
5	1,470	56.3	0.23	57	58.67
6	2,101	68.48	0.20	69	152.44
7	2,443	98.37	0.29	99	248.14
8	3,048	109.93	0.30	110	463.67
9	3,829	138.2	0.31	139	891.75
10	4,651	144.99	0.17	146	1,540.15

Results for Strategy II:

Instance	#col	UB	gap	IS	time[s]
1	161	8.93	0.00	10	0.89
2	433	22.38	0.07	23	2.96
3	726	25.12	0.06	26	10
4	1,178	34.99	0.05	36	33
5	1,730	47.09	0.03	48	99.96
6	2,330	59.19	0.03	60	228.17
7	3,115	80.75	0.06	82	482.42
8	3,738	89.94	0.06	91	720.72
9	4,916	112	0.06	113	1,681.48
10	5,198	134.3	0.08	135	2,296.13

Results for Strategy III:

Instance	#col	UB	LB	% gap	IS	ISLB	time [s]
1	172	8.89	8.89	4.50E-14	10	9	1.21
2	479	21.08	20.68	1.9	22	21	8.28
3	810	23.77	23.32	1.9	24	24	33.79
4	1,317	33.24	32.62	1.84	34	33	130.95
5	1,838	46.62	45.82	1.73	47	46	255.14
6	2,579	57.52	56.41	1.94	58	57	934.17
7	3,277	76.51	75.2	1.72	77	76	1,295.44
8	4,172	84.56	82.97	1.87	85	83	4,254.71
9	5,550	106.59	104.58	1.89	107	105	10,766.00
10	6,095	124.33	122.04	1.84	125	123	19,268.80

Results for Strategy IV:

Instance	#col	UB	LB	% gap	IS	ISLB	time [s]
1	188	8.89	8.89	4.50E-14	10	9	1.01
2	636	21.08	20.86	1.05	22	21	5.31
3	981	23.77	23.77	9.71E-14	24	24	17.17
4	1,700	33.23	33.23	1.87E-13	34	34	66.23
5	2,111	46.5	45.71	1.69	47	46	146.47
6	3,229	57.52	57.21	0.53	58	58	429.56
7	3,852	76.51	76.51	1.83E-13	77	77	743.093
8	5,160	84.56	84.56	2.93E-13	85	85	1,484.66
9	7,061	106.59	105.21	1.3	107	106	3,395.15
10	8,794	124.33	124.33	6.00E-13	125	125	6,061.11

1. Introduction.

2. Column generation method.

3. An optimization model for determining agricultural management zones.

4. Conclusions.

4. Conclusions

We present a new zoning method that optimally delineates rectangular homogeneous management zones. Experimental results showed that the proposed methodology enabled a contribution to this problem.

Decomposition methods provide a numerical optimization technique that allows to solve large scale optimization problems.

Thanks for your attention!

Decomposition methods in optimization and
a novel application in agriculture for zoning management

Víctor M. Albornoz
E-mail: victor.albornoz@usm.cl

Euro Summer Institute 2014
OR in Agriculture and Food Industry
Universidad de Lleida, 29 de Julio de 2014