

Wavelength selection with Tabu Search

J. A. Hageman, M. Streppel, R. Wehrens and L. M. C. Buydens*

Laboratory of Analytical Chemistry, University of Nijmegen, Toernooiveld 1, NL-6525 ED Nijmegen, The Netherlands

Received 12 June 2002; Revised 18 October 2002; Accepted 10 December 2002

This paper introduces Tabu Search in analytical chemistry by applying it to wavelength selection. Tabu Search is a deterministic global optimization technique loosely based on concepts from artificial intelligence. Wavelength selection is a method which can be used for improving the quality of calibration models. Tabu Search uses basic, problem-specific operators to explore a search space, and memory to keep track of parts already visited. Several implementational aspects of wavelength selection with Tabu Search will be discussed. Two ways of memorizing the search space are investigated: storing the actual solutions and storing the steps necessary to create them. Parameters associated with Tabu Search are configured with a Plackett–Burman design. In addition, two extension schemes for Tabu Search, intensification and diversification, have been implemented and are applied with good results. Eventually, two implementations of wavelength selection with Tabu Search are tested, one which searches for a solution with a constant number of wavelengths and one with a variable number of wavelengths. Both implementations are compared with results obtained by wavelength selection methods based on simulated annealing (SA) and genetic algorithms (GAs). It is demonstrated with three real-world data sets that Tabu Search performs equally well as and can be a valuable alternative to SA and GAs. The improvements in predictive abilities increased by a factor of 20 for data set 1 and by a factor of 2 for data sets 2 and 3. In addition, when the number of wavelengths in a solution is variable, measurements on the coverage of the search space show that the coverage is usually higher for Tabu Search compared with SA and GAs. Copyright © 2003 John Wiley & Sons, Ltd.

KEYWORDS: wavelength selection; Tabu Search; optimization; multivariate calibration

1. INTRODUCTION

In analytical chemistry, Tabu Search [1–4] is a relatively new technique with only a few published examples, dealing with graph theory [5] and molecular docking [6]. Tabu Search is an iterative deterministic global optimization method. It examines the search space in a highly ordered fashion using memory to keep track of parts already visited. Given a starting solution, it will always come up with the same end solution. In this paper, Tabu Search will be introduced for wavelength selection. Wavelength selection is a much used procedure for improving the quality of calibration models for example. After wavelength selection, predictive abilities are usually higher and the models are simpler and more robust [7–10]. The easiest way of finding the optimal combination of wavelengths would be an exhaustive search. However, an exhaustive search for wavelength selection would require the examination of an astronomical number of combinations. As this is usually not feasible owing to long

computation times, other wavelength selection methods have been designed. Originally, these methods used simple heuristics for locating solutions, but, given the characteristics of the methods, these were likely not the best obtainable solutions. With the recognition that wavelength selection is an optimization problem, and the increasing availability of faster computers, more sophisticated optimization techniques such as simulated annealing (SA) [11–13] and genetic algorithms (GAs) [7,10–12] have frequently been used. SA and GAs are both iterative probabilistic global optimization methods. As a consequence, the two methods do not always end up with the same end solution, given identical starting solutions.

Several implementational aspects of wavelength selection with Tabu Search will be discussed. As memorizing the search space is an important characteristic of Tabu Search, two possible ways of memorization are investigated. In addition, it will be shown that configuring the parameters associated with Tabu Search can be done with an experimental design. To further improve results, two extension schemes for Tabu Search for applying wavelength selection have been implemented: intensification and diversification.

*Correspondence to: L. M. C. Buydens, Laboratory of Analytical Chemistry, University of Nijmegen, Toernooiveld 1, NL-6525 ED Nijmegen, The Netherlands. E-mail: lbuydens@sci.kun.nl

It will be demonstrated that both are valuable assets. Two implementations of Tabu Search have been made, one that searches for solutions with a constant number of wavelengths and one with a variable number of wavelengths. Results of Tabu Search are applied to three real-world data sets and are compared with results obtained by SA- and GA-based methods. Results indicate that Tabu Search works equally well and is a valuable alternative to SA and GAs. In addition, when the number of wavelengths in a solution is variable, measurements on the coverage of the search space show that the coverage is usually higher for Tabu Search compared with SA and GAs.

2. THEORY

Where SA is based on the physical process of cooling down a heated liquid, and GAs are inspired by the process of evolution, Tabu Search is based on concepts from artificial intelligence [4]. It uses basic, problem-specific operators to explore a search space, and memory (which is called the tabu list) to keep track of parts already visited. By guiding the optimization to areas not present in memory, Tabu Search hopes to find the global optimum. The foundations for Tabu

Search were laid out in the late 1970s by Glover, and the principles were described in general terms in 1989 and 1990 also by Glover [1–3]. In recent years, tutorials documenting successes accomplished with Tabu Search have been published [3,4,14].

2.1. Tabu Search

Tabu Search is an optimization technique which tries to optimize a function $G(x)$, where x represents a parameter vector, by iteratively searching the parameter space of x for the optimal solution. The framework of Tabu Search consists of several steps which are described below and depicted in Figure 1.

1. Initialization: a starting solution s is generated by choosing random values for x . This solution is evaluated by the evaluation function, and solution s is stored in the algorithm's memory. This memory is called the tabu list.
2. Neighbourhood exploration: all possible neighbours of solution s are generated and evaluated. Neighbouring solutions are solutions which can be reached from the current solution by a simple, basic transformation

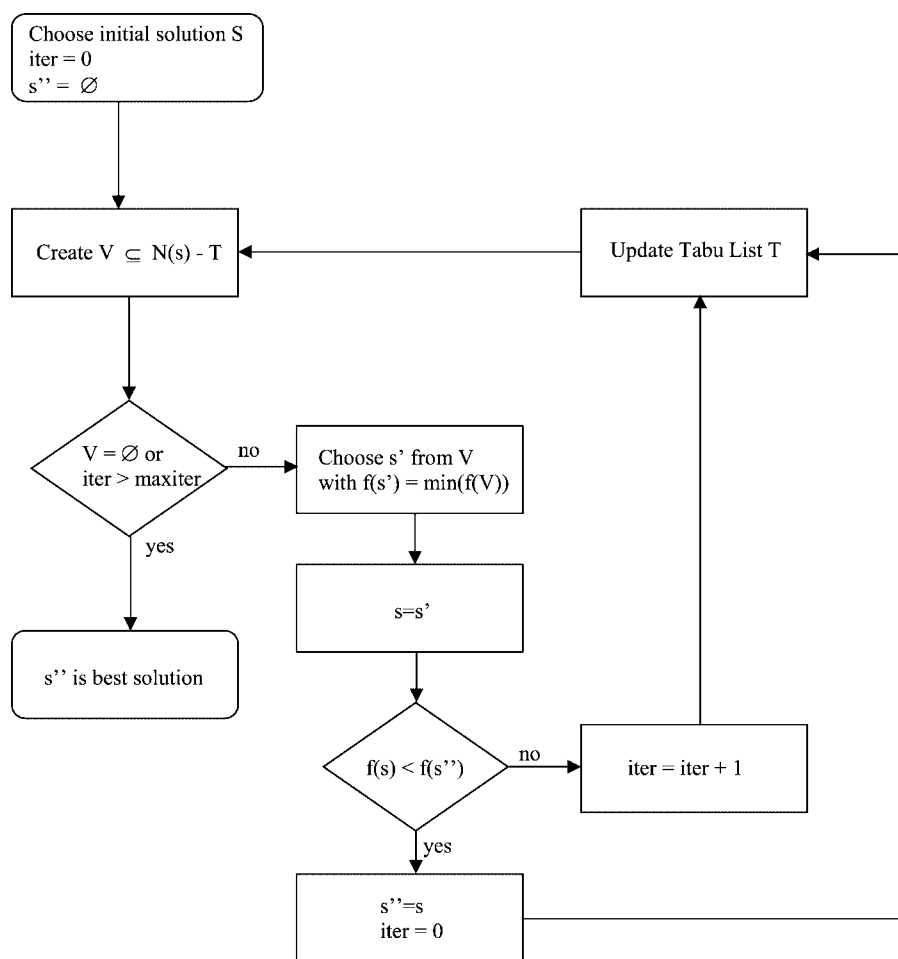


Figure 1. General flow chart of the Tabu Search algorithm: $iter$ keeps track of the number of iterations without an improvement; s is the current solution; s' is the solution with the lowest evaluation value of all neighbours of s ; s'' is the best obtained solution; V denotes all neighbours of solution s ; T is the tabu list; $maxiter$ is the allowable maximum number of iterations without an improvement.

Table I. Pseudo-code of the Tabu Search algorithm

```

choose an initial solution  $s$  in  $x$ 
 $s^* = s$ ;    % best solution so far
 $k = 0$ ;    % iteration number
 $k_{best} = 0$ ; % last improving iteration
 $k_{max} = \dots$ ; % maximum non-improving iteration
 $T = []$ ;    % tabu list
stop = false
while not stop
  generate  $V^* \subseteq N(s) - T$ 
  if (( $k - k_{best} > k_{max}$ ) or ( $V^* == []$ ))
    stop = true;
  else
     $k = k + 1$ ;
    choose best  $s'$  in  $V^*$ 
     $s = s'$ ;
    if ( $f(s) < f(s^*)$ )
       $s = s$ ;
       $k_{best} = k$ ;
    end
    update tabu list
  end
end

```

- of the current solution. Solutions which are present in the tabu list are considered unreachable neighbours.
3. New current solution: a new current solution is chosen from the explored neighbourhood. This solution cannot be in the tabu list and has to have the best evaluation value from all reachable neighbours. The evaluation value can be worse compared with the current solution. In this way the algorithm is able to overcome local minima. The new current solution is added to the tabu list.
 4. Stop: if no more neighbours are present (all are tabu) or a certain evaluation value or a predetermined number of iterations is reached, the algorithm stops, otherwise the algorithm continues with step 2.

In Table I the pseudo-code of the Tabu Search algorithm is given.

2.2. Neighbourhood exploration of wavelength selection

The neighbourhood of a solution is defined by those solutions which can be reached in one step. A solution reachable in one step is called a neighbour. These steps are specific for each optimization problem. Usually, two neighbouring solutions resemble each other closely. In the case of wavelength selection a solution is a combination of a number of wavelengths, and changing this combination can be accomplished by three different types of step.

1. Selecting or adding a number of wavelengths.
2. Deselecting or removing a number of wavelengths.
3. Moving a number of wavelengths.

Selecting and deselecting wavelengths are operators which are used in classical methods for wavelength selection. The move operator can be seen as a combination of the two: it first deselects a wavelength and subsequently selects a different one. The number of possible neighbours of a

solution increases drastically when more than one operator is allowed per step. Equations (1)–(3), give the maximum number of unique neighbouring solutions by using the operators select, deselect and move respectively:

$$\#select = \sum_{i=1}^{m_s} \binom{t-n}{i} \quad (1)$$

$$\#deselect = \sum_{i=1}^{m_d} \binom{n}{i} \quad (2)$$

$$\#move = \sum_{i=1}^{m_m} \binom{t-n}{i} \binom{n}{i} \quad (3)$$

where m_s , m_d and m_m are the maximum numbers of wavelengths considered for selection, deselection and move respectively, n is the number of selected wavelengths in s , and t is the total number of wavelengths in the spectrum. In a data set with 150 wavelengths of which 50 wavelengths are selected, the total number of neighbouring solutions is 6 075 075 when two select, deselect or move steps are allowed. When only one select, deselect or move step is allowed, there are only 5150 neighbouring solutions. Since a NIR spectrum can contain several hundred wavelengths, it is not feasible to allow more than one select, deselect or move step per iteration ($m_s = 1$, $m_d = 1$, $m_m = 1$). The total number of neighbours per iteration is then given by

$$\begin{aligned} & \binom{t-n}{1} + \binom{n}{1} + \binom{t-n}{1} \binom{n}{1} \\ & = t - n + n + (t - n)n \\ & = t + tn - n^2 \end{aligned} \quad (4)$$

The number of neighbours which have to be evaluated is at a maximum when $n = t/2$. This maximum is $t + t^2/4$. To further reduce the number of neighbouring solutions, a restriction is placed on the maximum distance of a move. When a wavelength is moved, it is only allowed to shift a distance d to left or right. This causes Tabu Search to search the solution space in a more structured manner, because the difference between two solutions is smaller when a wavelength is moved over a short distance, since the intensities at wavelengths close to each other tend to be correlated. The total number of neighbours that should be evaluated is now given by $t + 2dn^2$, where d is half the distance allowed for the move.

Another possibility is to only allow the move operator (Equation (3)) and not the select and deselect operators (Equations (1) and (2)). A consequence is that the number of selected wavelengths can be predetermined and will be kept constant during the optimization. When the distance of a move is restricted, the number of possible neighbours which need to be evaluated is $2dn^2$.

2.3. Tabu list

In Tabu Search the tabu list plays an important role. It keeps track of previously explored solutions and prohibits Tabu Search from revisiting them again. In this way, Tabu Search can overcome local minima by forcing the acceptance of solutions worse than the current solution. The tabu list has a finite length l . After l iterations the first tabu restriction is

Table II. Overview of operators on wavelengths p and q and the subsequent tabu restrictions in recency-based Tabu Search

Operator	Tabu list
Select p	Deselect p
Deselect p	Select p
Move q to p	Select q
	Deselect p

removed and this first solution becomes available again for selection. Keeping the tabu list too short may result in visiting the same sequence of solutions over and over again. The algorithm then ends up in a cycle and will not be able to locate better solutions. A list that is too long may lead to unnecessarily long run times but also may prevent the algorithm from reaching an optimal solution. In general, a list with length l will prevent cycles with length l . In the list the actual solution can be stored (called explicit memory). In this case the tabu list contains the actual combination of wavelength indices that are selected. It is also possible to store the steps necessary to generate the new solution (called recency-based memory). In the case when a wavelength is deselected, the selection of this wavelength becomes tabu, to prevent the algorithm from retracting to the original solution. Table II shows the tabu restrictions that take place after the execution of all three operators. If the steps are stored, all operators have their own tabu list. The tabu list is referred to as short-term memory when it contains previously visited solutions. It deals with the most recent history of the search trajectory.

2.4. Intensification and diversification

Two extension schemes are common for Tabu Search: intensification and diversification. Both schemes are referred to as a form of long-term memory, as they use information not available in short-term memory. Intensification focuses on the part of the solution space which seems promising and has often been visited with good results. As an intensification approach the best solutions obtained after several runs with different starting solutions are stored. Wavelengths which contribute to a good model will likely be selected more often in best found solutions. Thus, for a subsequent run, a new starting solution is generated consisting of the wavelengths which were selected in 30% and 60% of the best results of these previous runs. This provides Tabu Search with a promising starting solution.

Diversification is the opposite of intensification. It guides Tabu Search towards unexplored parts of the search space. In this way the solution space will be covered more thoroughly and the chance of missing the optimal solution will be reduced. To be able to guide Tabu Search to an unexplored part of the solution space, it is necessary to keep track of the parts of the solution space which Tabu Search has explored during its search. To accomplish this, each spectrum is divided into several bins, 12 in our case. When one or more wavelengths in a bin are selected, this bin is represented by a one. If no wavelengths in a bin are selected, it will be represented by a zero. All 12 ones and zeros put together form a bitstring. Each bitstring can be seen as a point in the

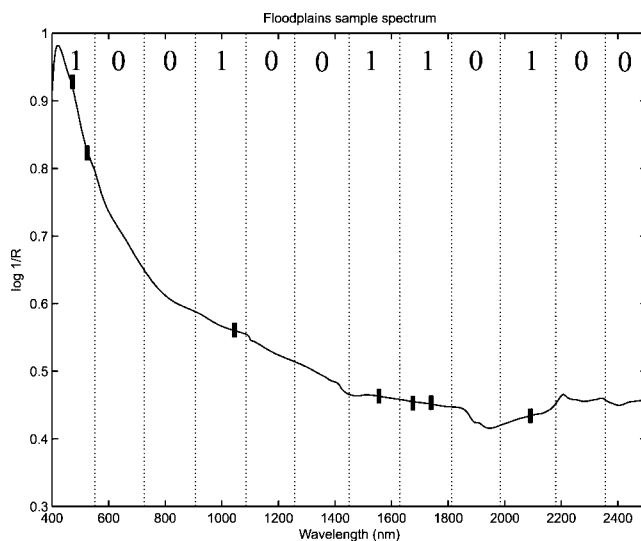


Figure 2. Example of how a combination of selected wavelengths is transformed into a bitstring. The bitstring indicates which region of the solution space has been examined by that solution.

(simplified) solution space. By keeping track of which bitstrings have been visited, the regions which have not been explored can be identified. Figure 2 shows how a combination of selected wavelengths is transformed into a bitstring.

After the end of a run it is checked which regions have not been visited. A solution in the region which is furthest from the visited regions is taken as the initial solution for a new run. By comparing the number of visited regions, it is also possible to measure the coverage of the search space. The higher this number, the better the algorithm has been able to scan different regions of the search space and thus should have been able to locate a good solution.

The distance between regions is calculated using the Hamming distance [15]. The Hamming distance between two regions j and k is given by

$$d_{j,k} = \sum_{i=1}^N \delta_{j,k}(i) \quad (5)$$

where $\delta_{j,k}(i) = 1$ when the i th bins of regions j and k do not contain the same value, $\delta_{j,k}(i) = 0$ when they contain the same value, and N is the number of bins.

2.5. Evaluation function

The goal of wavelength selection is to find a set of wavelengths for the creation of an optimal prediction model. The prediction model used is partial least squares (PLS) regression [16] and in particular SIMPLS [17]. A problem with PLS is that the number of latent variables should be specified. As the information in each set of wavelengths is not the same, this number cannot be kept constant but has to be determined again for each solution. For determining the optimal number of latent variables and to prevent overfitting, data sets are divided into two parts, a training set and a test set.

The correct number of latent variables for each subset is determined by leave- p -out cross-validation on the training set, with p being a number which divides each data set into

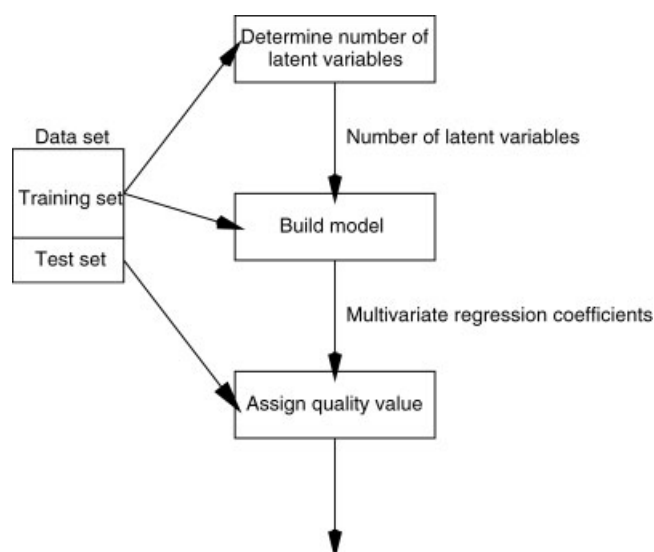


Figure 3. Flow chart of the evaluation function.

roughly 15–20 groups. Outliers should be removed from any used data set, as these will negatively influence the predictive abilities of the obtained models. Cross-validation results in an array of predictive abilities for each number of latent variables. The correct number of latent variables is obtained by comparing two consecutive values. When the next value still increases the predictive ability by more than 10%, the number of latent variables is increased by one. The complete training set in combination with the correct number of latent variables is used to calculate the PLS coefficients. The test set in combination with the PLS coefficients is used to calculate the RMSEP value:

$$\text{RMSEP} = \sqrt{\frac{\sum_{i=1}^n (y_{\text{pred},i} - y_i)^2}{n}} \quad (6)$$

where $y_{\text{pred},i}$ and y_i are the predicted and measured response values respectively and n is the number of samples. This value is used as an evaluation value in Tabu Search. The complete procedure for the evaluation function is shown in Figure 3.

3. EXPERIMENTAL

3.1. Data sets

Three near-infrared spectral data sets are used to test and compare the performance of Tabu Search for wavelength selection. All three data sets are assumed to be free of outliers.

1. Gasoline data set [18]: NIR spectra of gasoline samples with measured octane numbers. Samples are measured from 900 to 1700 nm with a sampling rate of 2 nm. The first 200 nm have been omitted as they show no signal. This results in a total of 300 wavelengths. The training set consists of 40 spectra and the test set of 20 spectra.
2. Wheat data set [18]: NIR spectra of wheat samples. Two responses have been measured, namely the moisture and protein contents. The spectra have been

recorded from 1100 to 2500 nm with a 2 nm interval. Every two wavelengths have been averaged to reduce the number of wavelengths, which resulted in 350 wavelengths. The training set consists of 67 spectra and the test set of 33 spectra.

3. Floodplains data set [19]: 67 NIR spectra of floodplains with four measured response values, namely the Cd, Zn, clay and organic matter contents. The spectra have been measured from 400 to 2500 nm with a sampling rate of 2 nm. To reduce the number of wavelengths, every three wavelengths have been averaged, which resulted in 350 wavelengths. The training set consists of 54 spectra and the test set of 13 spectra.

3.2. Tabu Search configuration

The Tabu Search algorithm for wavelength selection has been implemented as explained in the previous section. To be able to decide whether to store the actual solutions or the steps necessary to create them in the tabu list, both methods are implemented. To achieve optimal performance, the parameters associated with Tabu Search need to be optimized. For both cases these parameters are the locality of the move operator (parameter d), the maximum number of iterations and the number of selected wavelengths in the initial solution. When storing solutions, the length of the tabu list is kept long enough to obtain all encountered solutions. When storing steps, four extra parameters need to be optimized: the length of the tabu list for the *select*, *deselect*, *move from* and *move to* operators. For selecting optimal values for these parameters, a Plackett–Burman experimental design [20] was used in combination with the gasoline data set and leave-one-out cross-validation. The high and low values used in the experimental design are shown in Table III. All experiments in the Plackett–Burman design were executed fivefold, each time with a different starting solution, to cancel out random effects. The response variable in the experimental design was the RMSEP. After configuring Tabu Search while storing steps or storing solutions, it was decided which storing method would be used by running both algorithms with five different starting seeds and comparing the results.

Eventually, two implementations of Tabu Search have been made, one using a variable number of wavelengths and one using a constant number of wavelengths, only allowing the move operator. The configuration of the parameters associated with Tabu Search in combination with a constant number of wavelengths was based on the results of

Table III. Values used for the high and low levels of the variables in the Plackett–Burman design

Variable	High level	Low Level
Locality	45	10
Maximum iterations	65	30
Number initially selected	20	5
Length <i>select</i> tabu list	50	10
Length <i>deselect</i> tabu list	50	10
Length <i>move from</i> tabu list	50	10
Length <i>move to</i> tabu list	50	10

the experimental design, but modified slightly after additional experiments. Both Tabu Search implementations have been used to select optimal combinations of wavelengths of the three different data sets. When the number of wavelengths was constant, solutions consisting of 15, 30 and 45 wavelengths were optimized. All runs were repeated five times, to exclude any random effect of a starting solution.

3.3. Comparison with other wavelength selection methods

The performance of Tabu Search for wavelength selection is compared with the results of four other methods. Two of these are simple heuristic methods: forward selection and backward elimination [21]. The other two methods are SA and GAs. As it is possible for Tabu Search to search with a variable number of wavelengths as well as a constant one, both ways have also been incorporated into SA and GAs. The implementation for the GA approach with a variable number of wavelengths is based on References [7,11,12] with one exception, which is the mutation operator. The mutation operator used in this paper has a 90% chance of selecting a zero and a 10% chance of selecting a one. This ratio ensures that not too many wavelengths will be selected, as this is disadvantageous for a good prediction model. When the number of wavelengths must be kept constant, a different approach is used. Instead of a bitstring representation, an integer array representation is used. This array contains the indices of selected wavelengths. It is made sure that a wavelength index can be selected only once. The implementation of wavelength selection with SA with a constant number of wavelengths is based on References [11,12]. When allowing a variable number of wavelengths, some additions have been made. Besides moving selected wavelengths in the step-generating function, it is also possible to add or remove wavelengths. The step-generating function of SA can apply the same operators as in Tabu Search. The optimal settings for the SA- and GA-based methods were determined by trial and error and are shown in Tables IV and V respectively.

3.4. Software

All software was programmed in ANSI-C. The Tabu Search and SA methods were programmed from scratch. For the GA approach, PGAPack [22] was used as a basis. The SVD routine used in the SIMPLS algorithm was adopted from

Table IV. Settings for the SA method used for all data sets for implementations with a variable and constant number of wavelengths

SA method	Variable number of wavelengths	Constant number of wavelengths
Starting temperature	0.01	0.01
Cooling constant	0.999	0.999
Maximum Markov chain length	300	300
Minimum Markov chain length	150	150
Minimum temperature	0.001	0.001
Minimum number of constant function values	300	300
Chance of being moved ^a	0.1	0.1
Locality of move ^b	10	10
Chance of being added ^a	0.01	—
Chance of being deselected ^a	0.1	—

^aOnly one step of all possible three will be executed for each wavelength, but with different chances.

^bLocality indicates the maximum distance a wavelength can be moved.

Reference [23]. Calculations were performed on a Sun-Ultra 10 running at 440 MHz. Run times were dependent on the size of the data set and on the number of properties that needed to be predicted. For the gasoline data set, run times were of the order of 2 h for the SA, GA and Tabu Search methods for one run.

4. RESULTS AND DISCUSSION

4.1. Tabu Search configuration

The main effects of the Plackett–Burman experimental design are shown in Figure 4 for (A) storing solutions and (B) storing steps in the tabu list. In both cases a larger number of non-improving iterations is beneficial for obtaining a good solution, which is logical, because this prolongs the searching time. The number was set to 65. A high number of initial wavelengths is also beneficial. A high number increases the chance of selecting good contributing wavelengths from the start. However, experiments which were performed to choose the high and low limits for the experimental design demonstrated that, when this number is too high, Tabu Search spends a lot of time getting rid of non-contributing wavelengths, which in the end leads to decreased predictive power. This number was set to 20. The effect of the locality between the two approaches differs. When storing solutions, increasing the locality means that a solution can be refined

Table V. GA settings used for all data sets for implementations with a variable and constant number of wavelengths

GA method	Variable number of wavelengths	Constant number of wavelengths
Number of generations	800	800
Population size	300	300
Elitism	150	150
Crossover type	Uniform crossover	One-point crossover
Crossover probability	0.7	0.7
Mutation type	90%/10% bit flip	Uniform random replacement
Mutation probability	0.05	0.05
Selection type	Proportional selection	Proportional selection
Fitness type	Linear normalization fitness	Linear normalization fitness

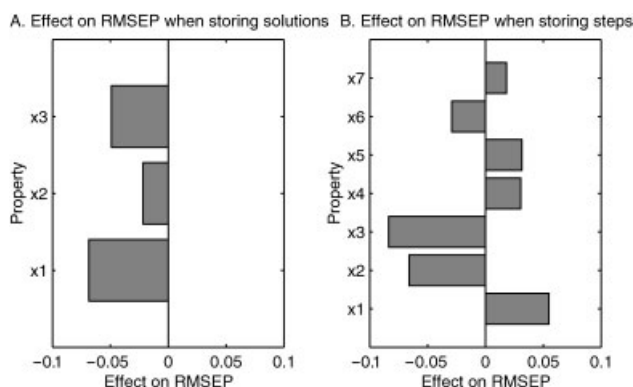


Figure 4. Results of the Plackett–Burman experimental design for determining an optimal configuration of Tabu Search: A, results when storing complete solutions; B, results when only storing steps (x1, locality of the move operator; x2, maximum number of iterations; x3, number of wavelengths in initial solution; x4, length *select* tabu list; x5, length *deselect* tabu list; x6, length *move from* tabu list; x7, length *move to* tabu list).

faster. When steps are being stored, a moved wavelength cannot be removed for a certain number of iterations. In that time, other wavelengths are changed and Tabu Search crosses that (local) optimum without ever locating the exact minimum. This is also reflected in the length of the tabu lists of the different steps. When tabu lists are too long, certain wavelengths are not available anymore and this forces Tabu Search to different areas in the search space. One exception is the *move from* length, this list prohibits wavelengths from being reselected after being moved. When this list is too short, it will lead to cycles. Lengths for *select*, *deselect*, *move from* and *move to* were set to 10, 10, 50 and 10 respectively.

Based on the results of the Plackett–Burman design, optimal configurations were determined for both approaches, as

Table VI. Configurations used by Tabu Search when storing actual solutions or steps leading to solutions. 'WL constant' indicates the setting when using a constant number of wavelengths

Tabu Search approach	Solutions	Steps	WL constant
Locality	45	10	30
Maximum iterations	65	65	65
Number initially selected	20	20	15
Length <i>select</i> tabu list		10	
Length <i>deselect</i> tabu list		10	
Length <i>move from</i> tabu list		50	5
Length <i>move to</i> tabu list		10	20

Table VII. Results of Tabu Search when storing steps in the tabu list and testing different additions. Results are obtained with the gasoline data set and leave-one-out cross-validation

	Run 1	Run 2	Run 3	Run 4	Run 5
Basic Tabu Search	7.88×10^{-2}	1.05×10^{-1}	1.25×10^{-1}	7.00×10^{-2}	7.09×10^{-2}
Intensification ^a		30%: 6.99×10^{-2}		60%: 6.59×10^{-2}	
Diversification ^b	7.69×10^{-2}	4.56×10^{-2}	4.56×10^{-2}	9.28×10^{-2}	6.68×10^{-2}
Both combined	1.30×10^{-1}	8.01×10^{-2}	1.02×10^{-1}	1.27×10^{-1}	1.14×10^{-1}

^aIntensification is performed on the combined results of all previous five replicate runs; 30% and 60% indicate the occurrences of selected wavelengths in the best found solutions.

^bAfter each initial run, five diversification runs have been performed; the best of these results is shown.

shown in Table VI. Analysis of the five replicate runs with different starting solutions with both approaches showed that there was no significant difference between the means of all runs. However, storing steps yielded the best solution with the lowest RMSEP value. A comparison of selected wavelengths in the best solution between all replicate runs also showed that the reproducibility is higher. When steps are being stored, it appears that the algorithm is much more able to select the same wavelengths during different runs. When the selection, deselection or moving of wavelengths is made tabu, the number of neighbours decreases, which is beneficial for running times. Therefore storing steps in the tabu list has more advantages and will be used for the remainder of the work in this paper. The best RMSEP value for each of the five runs is given in Table VII.

Applying intensification resulted in an improvement in RMSEP values. When the new starting solution consisted of wavelengths which were present in 30% of the best found solutions, the new RMSEP value was 6.99×10^{-2} ; when wavelengths were chosen which were present in 60% of the final solutions, the improvement was even greater, 6.59×10^{-2} . It is very likely that wavelengths which are selected more often in best found solutions contribute more to a good prediction model. A starting solution based on these wavelengths enables Tabu Search to come up with an improved prediction model.

Diversification is also able to improve RMSEP values. Results are shown in Table VII. After each replicate run, five diversification rounds were used. In four out of five, diversification yielded RMSEP values which were lower. The best RMSEP value after diversification is 4.5×10^{-2} .

Intensification and diversification have also been combined. Since wavelengths which are important for a good solution are likely to be in every best solution, diversification is performed before intensification. By first applying diversification, a large part of the solution space will be covered. After the initial best solutions have been found, intensification is used to zoom in on interesting wavelengths and perhaps locate a better solution. Results of the combination are given in Table VII. Diversification appears to be highly effective and renders intensification superfluous, because intensification does not lead to an improvement in all five cases. Therefore only diversification is used.

4.2. Comparison with other methods

Table VIII show the results for all runs with all three data sets, including the runs performed with the SA- and GA-based methods. Table IX shows the number of regions each

Table VIII. Results of the different wavelength selection methods. WL indicates the number of wavelengths present in the solution. LV indicates the number of latent variables

Method	Gasoline			Wheat			Floodplains		
	RMSEP	WL	LV	RMSEP	WL	LV	RMSEP	WL	LV
All wavelengths	6.88×10^{-1a}	301	4	7.84×10^{-1}	350	7	2.18	350	5
Stochastic methods									
Backward elimination	2.21×10^{-1}	181	8	6.14×10^{-1}	161	7	1.60	77	3
Forward selection	1.54×10^{-1}	29	7	4.72×10^{-1}	13	10	1.59	42	3
Implementations with a variable number of wavelengths									
GA	4.54×10^{-2}	30	6	3.37×10^{-1}	35	11	1.54	8	3
SA	3.33×10^{-2}	33	7	3.15×10^{-1}	21	12	1.55	6	3
Tabu Search	5.01×10^{-2}	38	5	3.19×10^{-1}	49	11	1.56	39	3
Implementations with a constant number of wavelengths									
GA	7.15×10^{-2}	15	4	3.43×10^{-1}	15	9	1.48	15	4
SA	5.82×10^{-2}	15	5	3.33×10^{-1}	15	10	1.56	15	3
Tabu Search	5.70×10^{-2}	15	6	3.36×10^{-1}	15	12	1.55	15	3
GA	6.00×10^{-2}	30	5	3.33×10^{-1}	30	9	1.49	30	4
SA	4.06×10^{-2}	30	7	3.34×10^{-1}	30	10	1.63	30	3
Tabu Search	6.59×10^{-2}	30	7	3.41×10^{-1}	30	10	1.57	30	3
GA	7.59×10^{-2}	45	8	3.50×10^{-1}	45	9	1.49	45	4
SA	1.11×10^{-1}	45	7	3.47×10^{-1}	45	10	1.63	45	3
Tabu Search	6.03×10^{-2}	45	7	3.42×10^{-1}	45	11	1.58	45	3

^aThe number of latent variables has been chosen visually, as automated selection resulted in the selection of one latent variable.

Table IX. Coverage of the solution space by the different methods

Method	Gasoline regions	Wheat regions	Floodplains regions
Stochastic methods			
Backward elimination	1	1	1
Forward selection	78	63	68
Implementations with a variable number of wavelengths			
GA	202	161	873
SA	226	219	656
Tabu Search	334	362	413
Implementations with a constant number of wavelengths ^a			
GA (15)	702	725	743
SA (15)	301	440	876
Tabu Search (15)	394	279	376
GA (30)	141	207	356
SA (30)	106	125	196
Tabu Search (30)	48	64	142
GA (45)	26	57	248
SA (45)	40	59	71
Tabu Search (45)	12	22	72

^aThe number in parentheses is the number of wavelengths kept constant in the solution.

method has examined. It can be seen that all tested wavelength selection methods are able to increase the predictive abilities of PLS models. It can also be seen that the simple heuristic methods backward elimination and forward selection perform poorly. The resulting RMSEP values are higher and the coverage of the search space is also less.

Implementations of GAs, SA and Tabu Search with a variable number of wavelengths show comparable results with respect to RMSEP values, especially for the wheat and floodplains data sets. The results for the gasoline data set

show some variations. Comparing results between all three methods is somewhat complicated. Not all methods come up with the same number of selected wavelengths and latent variables. Tabu Search selects the most wavelengths but uses a smaller number of latent variables. This might indicate a suboptimal solution. When a smaller number of latent variables is preferable, Tabu Search has a slight advantage; if absolute predictive values are important, SA scores better.

When applying Tabu Search, the coverage of the search space is high for the gasoline and wheat data sets, but the coverage of the floodplains data set is lower than for the SA- and GA-based methods. This is probably due to the large number of selected wavelengths, which makes it difficult to move from one region to another when only changing one wavelength. Coverage for backward elimination is very low, because, with all wavelengths selected in the initial solution, it is difficult to deselect all wavelengths in a bin, which is necessary for changing the bitstring.

The implementations with a constant number of wavelengths show a somewhat different trend. Tabu Search is able to build models which usually have slightly higher predictive abilities, especially for the gasoline and wheat data sets. Again, the largest deviations can be found with the gasoline data set, whereas the results obtained with the wheat and floodplains data sets are more homogeneous. For all three methods, results deteriorate somewhat when solutions are forced to contain more wavelengths. The coverage of the search space shows some trends when keeping the number of wavelengths constant in a solution. GAs usually have the higher coverage, followed by SA and Tabu Search. This reflects the degree of randomness in each method. Where GAs are able to take large steps in the search space, this ability is somewhat reduced in SA, and in Tabu Search

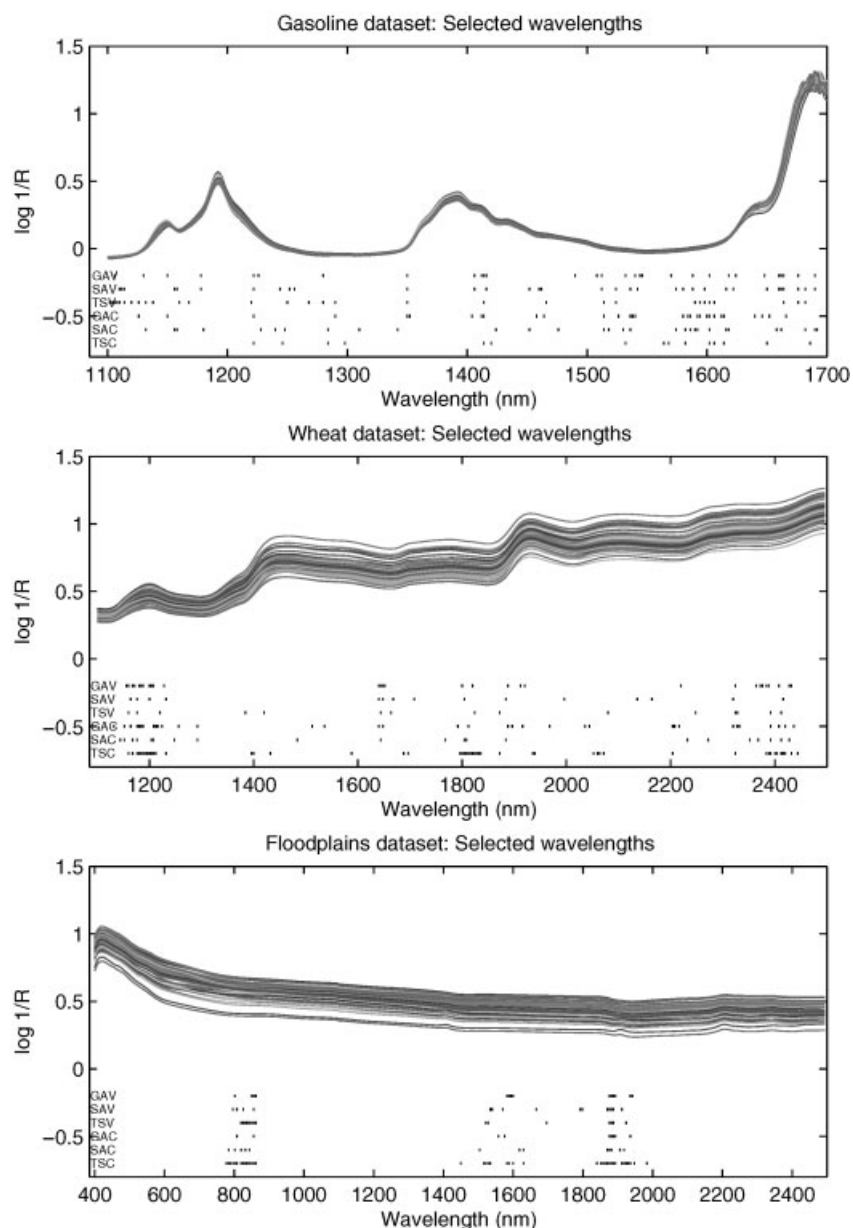


Figure 5. Selected wavelengths in the best solutions obtained with wavelength selection methods based on SA, GAs and Tabu Search. The labels before the solutions indicate which method was used: GAs, SA or TS (Tabu Search); V/C indicates variable/constant number of wavelengths. From the solutions with a different constant number of wavelengths (15, 30 or 45), only the one with the best RMSEP values is shown.

this is highly structured. Nevertheless, solutions obtained with Tabu Search perform equally well as and sometimes better than those obtained with GAs and SA.

Figure 5 shows the selected wavelengths in the best solutions obtained with the SA, GA and Tabu Search methods for each data set. When looking at wavelengths which are selected by the different methods in the best solutions, there is great overlap. Wavelengths selected in the gasoline data set can be found more or less throughout the spectrum, but for the wheat data set, and even more so for the floodplains dataset, specific regions of selected wavelengths can be identified. In these cases, wavelengths from specific regions contain the most valuable information. For the wheat and floodplains data sets, differences in the positions of selected wavelengths in the best solutions between the

replicate runs were small. For the gasoline data set these differences were slightly larger. As wavelengths near each other are often highly correlated, small differences have only small effects on the predictive ability of models. Of all three optimization methods, replicate runs with different starting solutions performed with GAs and Tabu Search have a higher reproducibility. An example of this is shown in Figure 6. In contrast to GAs and Tabu Search, it would appear that SA is more easily caught in a local optimum.

The gasoline and wheat data sets have also originally been used for demonstrating the possibilities of wavelength selection with GAs [10]. When comparing the results from Reference [10] and our findings, selected wavelengths and RMSEP values differ at some points. These differences are likely to be caused by the differences in evaluation functions

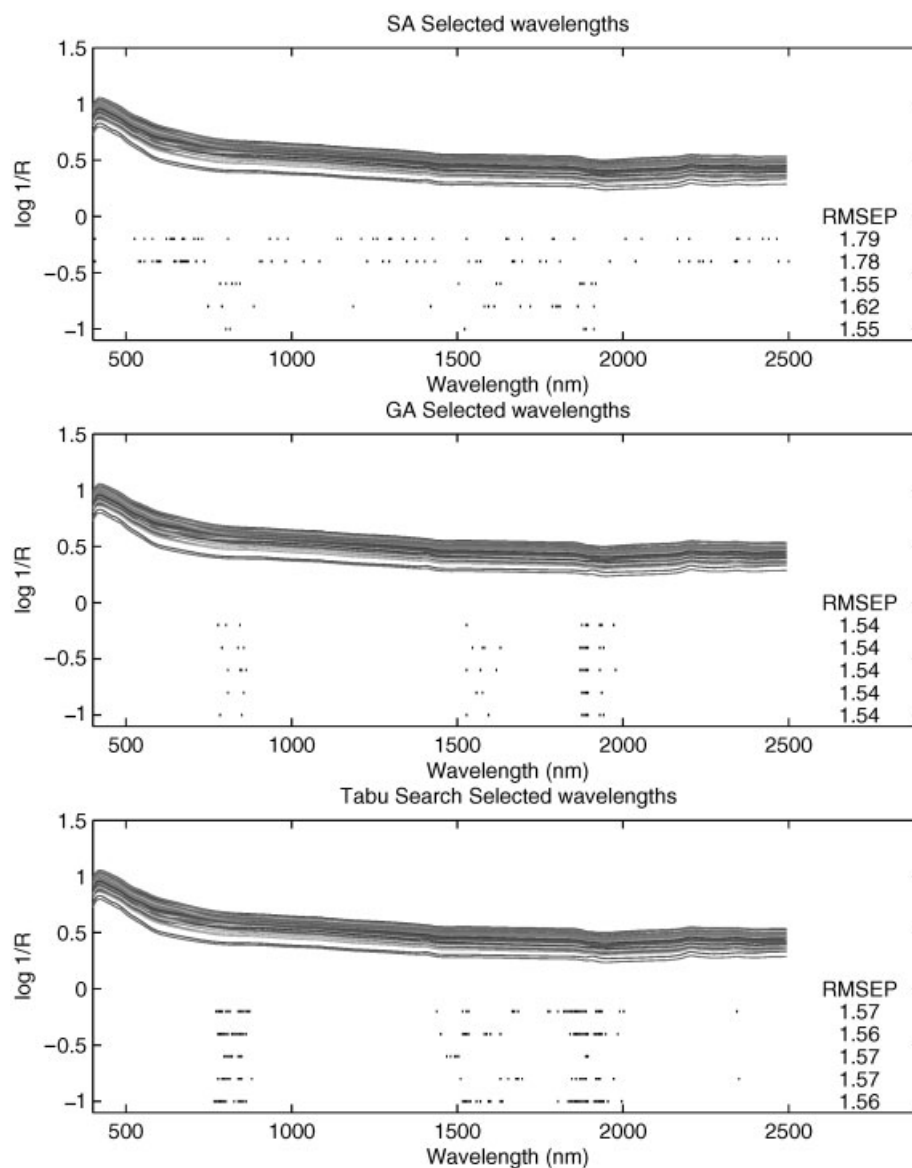


Figure 6. Results of five replicate runs with different starting solutions. Wavelength selection is performed on the floodplains data set with SA, GAs and Tabu Search with a variable number of wavelengths.

and in the number of latent variables which may have been used. In this paper, for each solution, a different number of latent variables is calculated with leave- p -out cross-validation, while Reference [10] used a constant number.

In this paper the data sets have not been preprocessed before wavelength selection and PLS modelling. It is very likely that spectral preprocessing would increase the predictive abilities of the models. It is also very likely that the beneficial effects of preprocessing will be the same for the different types of wavelength selection technique, so this has been left out in this work. To obtain the best possible prediction models, spectral preprocessing can be recommended.

5. CONCLUSION

This paper shows the potential and an implementation of wavelength selection with Tabu Search. Tabu Search is a

relatively new method in analytical chemistry; in contrast to SA and GAs, it is not probabilistic but deterministic. As a consequence, it will always come up with the same best solution if it is provided with the same starting solution.

Wavelength selection is a much used procedure for easily increasing the predictive ability of models. Even simple heuristic methods are able to improve models. However, better improvements are obtained by using more sophisticated methods such as SA, GAs and Tabu Search. It is demonstrated that the implementation described here yields results as good as those obtained by other well-established methods such as SA and GAs. Configuring the parameters of Tabu Search, or meta-optimizing, is no difficult task. It can be done with an experimental design or, if some experience is present, by trial and error. The intensification and diversification approaches applied in this paper are valuable extensions of Tabu Search.

It is possible to use Tabu Search for locating solutions with a variable or constant number of wavelengths. In general, results are better when the number of wavelengths is variable. Using Tabu Search with a variable number of wavelengths, the coverage of the search space is usually better compared with GAs and SA. This can become important when more local minima exist and it becomes harder to avoid getting trapped in them.

Tabu Search is a valuable alternative to SA and GAs, especially in cases where there is a clear definition possible of a neighbourhood of a solution.

REFERENCES

- Glover F. Tabu search—Part I. *ORSA J. Comput.* 1989; **1**: 190.
- Glover F. Tabu search—Part II. *ORSA J. Comput.* 1990; **2**: 4.
- Glover F, Taillard E, de Werra D. A user's guide to tabu search. *Ann. Oper. Res.* 1993; **41**: 3–28.
- Glover F, Laguna M. *Tabu Search*. Kluwer: Dordrecht, 1998.
- Kvasnicka V, Pospichal J. Fast evaluation of chemical distance by tabu search algorithm. *J. Chem. Info. Comput. Sci.* 1994; **34**: 1109–1112.
- Westhead DR, Clark DE, Murray CW. A comparison of heuristic search algorithms for molecular docking. *J. Comput.-Aided Mol. Design* 1997; **11**: 209–228.
- Leardi R, Boggua R, Terrile M. Genetic algorithms as strategy for feature selection. *J. Chemometrics* 1992; **6**: 267–281.
- Centner V, Massart DL, de Noord OE, de Jong S, Vandeginste B, Sterna C. Elimination of uninformative variables for multivariate calibration. *Anal. Chem.* 1996; **68**: 3851–3858.
- Jouan-Rimbaud D, Walczak B, Poppi RJ, de Noord OE, Massart DL. Application of wavelet transform to extract the relevant component from spectral data for multivariate calibration. *Anal. Chem.* 1997; **69**: 4317–4323.
- Leardi R. Application of genetic algorithm–PLS for feature selection in spectral data sets. *J. Chemometrics* 2000; **14**: 643–655.
- Lucasius CB, Beckers MLM, Kateman G. Genetic algorithms in wavelength selection: a comparative study. *Anal. Chim. Acta* 1994; **286**: 135–153.
- Horchner U, Kalivas JH. Further investigation on a comparative study of simulated annealing and genetic algorithm for wavelength selection. *Anal. Chim. Acta* 1995; **311**: 1–13.
- Horchner U, Kalivas JH. Simulated-annealing-based optimization algorithms: fundamentals and wavelength selection applications. *J. Chemometrics* 1995; **9**: 283–308.
- de Werra D, Hertz A. Tabu search techniques, a tutorial and an application to neural networks. *OR Spektrum* 1989; **11**: 131–141.
- Vandeginste BGM, Massart DL, Buydens LMC, de Jong S, Lewi PJ, Smeyers-Verbeke J. *Handbook of Chemometrics and Qualimetrics: Part B*. Elsevier: Amsterdam, 1998.
- Geladi P, Kowalski BR. Partial least-squares regression: a tutorial. *Anal. Chim. Acta* 1986; **185**: 1–17.
- de Jong S. SIMPLS: an alternative approach to partial least squares regression. *Chemometrics Intell. Lab. Syst.* 1993; **18**: 251–263.
- Kalivas JH. Two datasets of near infrared spectra. *Chemometrics Intell. Lab. Syst.* 1997; **37**: 255–259.
- Kooistra L, Wehrens R, Leuven RSEW, Buydens LMC. Possibilities of VNIR spectroscopy for the assessment of soil contamination in river floodplains. *Anal. Chim. Acta* 2001; **446**: 97–105.
- Plackett RL, Burman JP. The design of optimum multifactorial experiments. *Biometrika* 1946; **19**: 305–325.
- Massart DL, Vandeginste BGM, Buydens LMC, de Jong S, Lewi PJ, Smeyers-Verbeke J. *Handbook of Chemometrics and Qualimetrics: Part A*. Elsevier: Amsterdam, 1998.
- Levine D. PGAPack V1.0. [Online]. Available: <ftp://ftp.mcs.anl.gov/pub/pgapack/pgapack.tar.Z>.
- Press WH, Flannery BP, Teukolsky SA, Vetterling WT. *Numerical Recipes in C*. Cambridge University Press: Cambridge, 1988.