Anal. Chem. **2005,** *77,* 290−303

Closed-Loop, Multiobjective Optimization of Analytical Instrumentation: Gas Chromatography/ Time-of-Flight Mass Spectrometry of the Metabolomes of Human Serum and of Yeast Fermentations

Steve O'Hagan, Warwick B. Dunn, Marie Brown, Joshua D. Knowles, and Douglas B. Kell*

School of Chemistry, University of Manchester, Faraday Building, Sackville Street, P.O. Box 88, Manchester M60 1QD, U.K.

The number of instrumental parameters controlling modern analytical apparatus can be substantial, and varying them systematically to optimize a particular chromatographic separation, for example, is out of the question because of the astronomical number of combinations that are possible (i.e., the "search space" is very large). However, heuristic methods, such as those based on evolutionary computing, can be used to explore such search spaces efficiently. We here describe the implementation of an entirely automated (closed-loop) strategy for doing this and apply it to the optimization of gas chromatographic separations of the metabolomes of human serum and of yeast fermentation broths. Without human intervention, the Robot Chromatographer system (i) initializes the settings on the instrument, (ii) controls the analytical run, (iii) extracts the variables defining the analytical performance (specifically the number of peaks, signal/noise ratio, and run time), (iv) chooses (via the PESA-II multiobjective genetic algorithm), and (v) programs the next series of instrumental settings, the whole continuing in an iterative cycle until suitable sets of optimal conditions have been established. Genetic programming was used to remove noise peaks and to establish the basis for the improvements observed. The system showed that the number of peaks observable depended enormously on the conditions used and served to increase them by as much as 3-fold (e.g., to over 950 in human serum) while in many cases maintaining or reducing the run time and preserving excellent signal/noise ratios. The evolutionary closed-loop machine learning strategy we describe is generic to any type of analytical optimization.

Optimization lies at the core of many if not all areas of science, technology, and engineering.[1−12] This is equally true in analytical chemistry method development.

The history of chromatographic optimization goes back more than 20 years, e.g., to the work of Laub and Purnell exploiting "window diagrams" in gas chromatography[13−15] and to that of Glajch, Kirkland, and co-workers on multisolvent LC optimization[16−18]). These pioneering studies showed that comparatively small changes in conditions could have substantial effects on the effectiveness of a particular separation.

Most methods for effecting such optimizations use standard design of experiments strategies to cover the space of possible experiments[6,10,19] followed by simplex kinds of optimization based on their results.[20] Alternatively, they use prediction models based on a more or less linear interpolation of a restricted number of runs to predict the performance of future experiments that might be optimal. This works well when the target metabolites are known, the interactions between the parameters are indeed largely linear and the optimization surface is convex, as is often the case,[21−23] and modeling software for estimating the optimal

* Corresponding author. Phone: 0044 161 200 4492. E-mail: dbk@manchester.ac.uk. Web: http://dbk.ch.umist.ac.uk.

(1) Fletcher, R. *Practical methods of optimization,* 2nd ed.; Wiley: New York, 1987.
(2) Goldberg, D. E. *Genetic algorithms in search, optimization and machine learning*; Addison-Wesley: Reading, MA, 1989.
(3) Walters, F. H.; Parker, L. R.; Morgan, S. L.; Deming, S. N. *Sequential simplex optimization. A technique for improving quality and productivity in research, development and manufacturing*; CRC Press: Boca Raton, FL, 1991.
(4) Myers, R. H.; Montgomery, D. C. *Response surface methodology: process and product optimization using designed experiments*; Wiley: New York, 1995.
(5) Corne, D., Dorigo, M., Glover, F., Eds. *New ideas in optimization*; McGraw-Hill: London, 1999.
(6) Hicks, C. R.; Turner, K. V., Jr. *Fundamental concepts in the design of experiments,* 5th ed.; Oxford University Press: Oxford, 1999.
(7) Zitzler, E. *Evolutionary algorithms for multiobjective optimization: methods and applications*; Shaker Verlag: Aachen, 1999.
(8) Gottlieb, J. *Evolutionary algorithms for constrained optimization problems*; Shaker Verlag: Aachen, 2000.
(9) Deb, K. *Multi-objective optimization using evolutionary algorithms*; Wiley: New York, 2001.
(10) Montgomery, D. C. *Design and analysis of experiments,* 5th ed.; Wiley: Chichester, 2001.
(11) Branke, J. *Evolutionary optimization in dynamic environments*; Kluwer Academic Publishers: Dordrecht, 2002.
(12) Osyczka, A. *Evolutionary algorithms for single and multicriteria design optimization*; Physica-Verlag: Heidelberg, 2002.
(13) Laub, R. J.; Purnell, J. H. *J. Chromatogr.* **1975**, *112*, 71−79.
(14) Laub, R. J.; Purnell, J. H. *Anal. Chem.* **1976**, *48*, 799−802.
(15) Laub, R. J.; Purnell, J. H. *Anal. Chem.* **1976**, *48*, 1720−1725.
(16) Glajch, J. L.; Kirkland, J. J.; Snyder, L. R. *J. Chromatogr.* **1982**, *238*, 269−280.
(17) Kirkland, J. J.; Glajch, J. L. *J. Chromatogr* **1983**, *255*, 27−39.
(18) Glajch, J. L.; Kirkland, J. J.; Minor, J. M. *J. Liq. Chromatogr.* **1987**, *10*, 1727−1747.
(19) Araujo, P. W.; Brereton, R. G. *Trends Anal. Chem.* **1996**, *15*, 63−70.
(20) Morgan, S. L.; Deming, S. N. *J. Chromatogr.* **1975**, *112*, 267−285.

10.1021/ac049146x CCC: $30.25
Published on Web 12/03/2004

separation conditions based on a small number of trials is available (e.g., refs 24 and 25).

However, such strategies cannot work well when the search space is very epistatic (i.e., the optimal value of one parameter depends strongly on the values of other parameters), nor where the number of target components is not a priori known or may be very large in number. This latter is very much the case in the emerging field of metabolomics, where many hundreds of metabolites are likely to be present.[26−28]

Any kind of optimization strategy involves paired sets of (a) the variables that the experimenter controls (the independent variables) and (b) those that are measured in response to them (the response variable(s) or objective function(s)). A common metaphor is then to view the possible combinations of independent variables (properly known as parameters) as the generalized *XY* ("position") coordinates in a "landscape" in which a response variable is encoded as a "height", leading to the concept of a response surface.[4] The search for the optimal combination of parameters is thus a combinatorial optimization problem, whose difficulty increases exponentially with the number of parameters to be varied.[29] Such problems are normally not soluble in polynomial time.[30] For instance, if there are 12 variables, each of which could take just 20 values, the number of combinations is 20^{12} (4.10^{15}) and the lifetime of the Universe is $\sim 10^{17}$ s.[31] Thus, conventional strategies cannot expect to achieve anything like an optimum, and so-called heuristic methods,[32−36] in which a "good" but not provably (globally) optimal solution is sought, are necessary.

A second issue concerns the question of what it is that one is trying to optimize. Most optimizations are single-objective (e.g., improving the single worst pairwise resolution in a chromatogram), although Vanbel used a two-objective optimization[37] in which one objective was the robustness of the separation to small changes in the parameters. In the real world, most optimization problems are multiobjective in nature,[7,9,12,36,38−41] and there is evidence that optimization of even single-objective problems may be improved by recasting them in a multiobjective way.[42] In the case of gas or liquid chromatography, the objective functions might include the number of peaks, the "worst" resolution of any pair of peaks, the median peak separation of adjacent pairs of peaks, the separation time, the number of theoretical plates, and the average signal/noise ratio of the peaks. In the more general context, one might also envisage linearity, limit of detection, and precision as desirable objects for optimization.

Methods in which the choice of which experiment to perform is part of a learning strategy are known as "active learning" methods[43−48] and are appropriate to search and optimization strategies of the present type. To help the experimenter to implement such active learning methods, it would be desirable to effect the complete automation of the optimization cycle, in which the parameters are varied "intelligently" and the response assessed iteratively and automatically in a closed-loop manner. Such approaches have been demonstrated in clinical chemistry,[49,50] electrochemistry,[51] fs-laser control of chemical reactions,[52,53] and functional genomics.[54] It is also of some philosophical interest to recognize that the iterative interplay between the computational choice of the next experiment and the resulting data it provides may be seen as a highly effective, and more or less purely inductive, form of scientific reasoning.[54−61]

(21) Dolan, J. W.; Snyder, L. R.; Djordjevic, N. M.; Hill, D. W.; Waeghe, T. J. *J. Chromatogr., A* **1999**, *857*, 1−20.
(22) Dolan, J. W.; Snyder, L. R.; Djordjevic, N. M.; Hill, D. W.; Waeghe, T. J. *J. Chromatogr., A* **1999**, *857*, 21−39.
(23) Dolan, J. W.; Snyder, L. R.; Wolcott, R. G.; Haber, P.; Baczek, T.; Kaliszan, R.; Sander, L. C. *J. Chromatogr., A* **1999**, *857*, 41−68.
(24) Bautz, D. E.; Dolan, J. W.; Raddatz, W. D.; Snyder, L. R. *Anal. Chem.* **1990**, *62*, 1560−1567.
(25) Molnar, I. *J. Chromatogr., A* **2002**, *965*, 175−194.
(26) Fiehn, O.; Kopka, J.; Dormann, P.; Altmann, T.; Trethewey, R. N.; Willmitzer, L. *Nat. Biotechnol.* **2000**, *18*, 1157−1161.
(27) Goodacre, R.; Vaidyanathan, S.; Dunn, W. B.; Harrigan, G. G.; Kell, D. B. *Trends Biotechnol.* **2004**, *22*, 245−252.
(28) Kell, D. B. *Curr. Opin. Microbiol.* **2004**, *7*, 296−307.
(29) Cook, W. J.; Cunningham, W. H.; Pulleyblank, W. R.; Schrijver, A. *Combinatorial Optimization*; Wiley-Interscience: New York, 1998.
(30) Garey, M.; Johnson, D. *Computers and intractability: a guide to the theory of NP-completeness.*; Freeman: San Francisco, 1979.
(31) Barrow, J. D.; Silk, J. *The left hand of creation: the origin and evolution of the expanding universe*; Penguin: London, 1995.
(32) Reeves, C. R., Ed. *Modern heuristic techniques for combinatorial problems*; McGraw-Hill: London, 1995.
(33) Reeves, C. R. *Ann. Oper. Res.* **1999**, *86*, 473−490.
(34) Michalewicz, Z.; Fogel, D. B. *How to solve it: modern heuristics*; Springer-Verlag: Heidelberg, 2000.
(35) RaywardSmith, V. J., Osman, I. H., Reeves, C. R., Smith, G. D., Eds. *Modern heuristic search methods*; Wiley: Chichester, 1996.
(36) Dasgupta, P.; Chakrabarti, P. P.; DeSarkar, S. C. *Multiobjective heuristic search*; Vieweg: Braunschweig, 1999.
(37) Vanbel, P. F. *J. Pharm. Biomed. Anal.* **1999**, *21*, 603−610.
(38) Ringuest, J. L. *Multiobjective optimization: behavioral and computational considerations*; Kluwer Academic Publishers: Dordrecht, 1992.
(39) Bagchi, T. P. *Multiobjective scheduling by genetic algorithms*; Kluwer Academic Publishers: Dordrecht, 1999.
(40) Coello Coello, C. A.; van Veldhuizen, D. A.; Lamont, G. B. *Evolutionary algorithms for solving multi-objective problems*; Kluwer Academic Publishers: New York, 2002.
(41) Corne, D. W.; Deb, K.; Fleming, P. J.; Knowles, J. D. *IEEE coNNectionS* **2003**, *1*, 9−13.
(42) Knowles, J. D.; Watson, R. A.; Corne, D. W. *Proc. 1st Conf. on Evolutionary multi-criterion optimization (EMO'01)*; Zitzler, E. et al., Eds.; Springer: Berlin, 2001; pp 260−283.
(43) Mackay, D. *Neural Comput.* **1992**, *4*, 590−604.
(44) Raju, G. K.; Cooney, C. L. *AIChE J.* **1998**, *44*, 2199−2211.
(45) Bryant, C. H.; Muggleton, S. H.; Oliver, S. G.; Kell, D. B.; Reiser, P.; King, R. D. *Electron. Trans. Artif. Intell.* **2001**, *5*, 1−36 (http://www.ep.liu.se/ej/etai/2001/2001/).
(46) Cohn, D. A.; Ghabhramani, Z.; Jordan, M. I. *J. Artif. Intell. Res.* **1996**, *4*, 129−145.
(47) Hasenjäger, M.; Ritter, H. *Neural Proc. Lett.* **1998**, *7*, 110−117.
(48) Cohn, D. A.; Atlas, L.; Ladner, R. *Machine Learn.* **1994**, *15*, 201−221.
(49) Olansky, A. S.; Parker, L. R., Jr.; Morgan, S. L.; Deming, S. N. *Anal. Chim. Acta* **1977**, *95*, 107−133.
(50) Olansky, A. S.; Deming, S. N. *Clin. Chem.* **1978**, *24*, 2115−2124.
(51) Zytkow, J. M.; Zhu, J.; Hussam, A. *Proc. 8th Nat. Conf. on Artif. Intell.*; Dietterich, T., Swartout, W., Eds.; AAAI Press: Boston; 1990; pp 889−894.
(52) Judson, R. S.; Rabitz, H. *Phys. Rev. Lett.* **1992**, *68*, 1500−1503.
(53) Daniel, C.; Full, J.; Gonzalez, L.; Lupulescu, C.; Manz, J.; Merli, A.; Vajda, S.; Woste, L. *Science* **2003**, *299*, 536−539.
(54) King, R. D.; Whelan, K. E.; Jones, F. M.; Reiser, P. G. K.; Bryant, C. H.; Muggleton, S. H.; Kell, D. B.; Oliver, S. G. *Nature* **2004**, *427*, 247−252.
(55) Langley, P.; Simon, H. A.; Bradshaw, G. L.; Zytkow, J. M. *Scientific Discovery: computational exploration of the creative processes*; MIT Press: Cambridge, MA, 1987.
(56) Valdés-Pérez, R. E. *Commun. ACM* **1999**, *42*, 37−41.
(57) Langley, P. *Int. J. Hum.-Comput. Stud.* **2000**, *53*, 393−410.
(58) Goldberg, D. E. *The design of innovation: lessons from and for competent genetic algorithms*; Kluwer: Boston, 2002.
(59) Koza, J. R.; Keane, M. A.; Streeter, M. J.; Mydlowec, W.; Yu, J.; Lanza, G. *Genetic programming: routine human-competitive machine intelligence*; Kluwer: New York, 2003.
(60) Vaidyanathan, S.; Broadhurst, D. I.; Kell, D. B.; Goodacre, R. *Anal. Chem.* **2003**, *75*, 6679−6686.
(61) Kell, D. B.; Oliver, S. G. *Bioessays* **2004**, *26*, 99−105.

Our own recent work[60] has used the methods of evolutionary computing to find good solutions to analytical problems in an astronomical search space. These encode the experimental parameters in a tree or a string whose "fitness" is represented by the objective function(s) and which may be evolved by processes akin to mutation, recombination, and Darwinian selection to produce high-quality variants in an effective and efficient manner. (The effective population of the search space by "interesting" examples also assists in the development of rules that allow one to understand the basis for good solutions.[60]) Numerous works attest to the power of such algorithms (e.g., refs 5, 7, 9, 40, and 62−66), which in the experimental bioanalytical arena have also been used in areas such as fermentation optimization.[67−70] However, in all these cases, the work was done with human intervention, with the human being involved in making up media or changing instrumentation settings manually.

A desirable generic goal would therefore be to produce a system that sets up the parameters of an analytical experiment, performs the analysis, evaluates the outcome or quality of the analysis, and on the basis of the outcomes designs the next set of experiments automatically and without human intervention, the whole process continuing iteratively until some appropriately high quality method has been evolved. The purpose of the present paper is to describe our successful implementation of this fully closed loop method of optimization in the field of gas chromatography/mass spectrometry.

METHODS

Instrumentation. All experiments were run on a GC-TOF-MS instrument (Agilent 6890N gas chromatograph and LECO Pegasus III TOF mass spectrometer) using the manufacturer's software (ChromaTof version 2.12). The set of instrument parameters chosen for optimization is given in Table 1. The gas chromatograph was operated in split mode using helium as carrier gas in constant flow mode, with an initial GC temperature of 70 °C and an on-column volume maximum (injection volume/split ratio) set at 0.5 μL. A DB-50 GC column (Supelco, Gillingham, U.K.; 30 m × 0.25 mm × 0.25 μm film thickness) was used. The transfer line and source temperatures used were 250 and 230 °C, respectively. The mass range used was 40−600 Da with a detector voltage of 1700 V. In the ChromaTOF software, the S/N threshold was set at 10, baseline offset at 1.0, data points for averaging at 5, and peak width at 3. The TOF mass spectrometer can collect spectra at up to 500 Hz and uses sophisticated but proprietary deconvolution software to discriminate overlapping peaks on the basis of their mass spectra.

Samples. Two different sample types were used in the optimization. Human serum (Sigma-Aldrich, Gillingham, U.K.; Catalog No. S7023) was prepared by protein precipitation (500 μL of serum rotary mixed with 1500 μL of AR grade methanol for 30 s, centrifuged at 13385*g* for 15 min). The supernatant was dried by evaporation with a HETO VR MAXI vacuum centrifuge attached to a HETO CT/DW 60E cooling trap (Thermo Life Sciences, Basingstoke, U.K.). Yeast supernatant was prepared by inoculation of 15 mL of metabolic footprinting medium[71] with a single yeast colony (strain BY4743, heterozygous diploid, MATa/MATa; his3D1/his3D1; leu2D0/leu2D0) followed by shaking at 30 °C for 24 h (stationary phase, $A_{600} = 0.6$). After 24 h, the supernatant was separated from the cells by centrifugation (same conditions as for serum). The 1.5-mL aliquots of yeast supernatant were dried by evaporation using the methods described above.

Dried samples were derivatized as follows; 100 μL of 20 mg/ml *O*-methylhydroxylamine solution was added and heated at 40 °C for 90 min followed by addition of 100 μL of *N*-acetyl-*N*-(trimethylsilyl)trifluoroacetamide and heating at 40 °C for 90 min. The 20 separate samples were prepared, combined, and then aliquoted into three separate sample vials. To ensure no or minimal degradation at room temperature while sitting in the autosampler, each sample was employed for a maximum of 48 h.

Computational Strategy. Modern, Windows-based analytical instrumentation software normally requires the user to set instrument parameters by typing them into boxes. To automate this process without access to the source code of the manufacturer's software requires the ability to "mimic" this in a manner that may be controlled via a scripting program. The essential strategy was to use software that behaves as a macro recorder and can act as a "shell" to the manufacturer's instrumentation software, with parameter values being obtained by importing them from the output of the genetic algorithm optimization part of the program. The macro recorder used was Eventcorder (http://www.volny.cz/eventcorder/eventcorder.htm). The control software, GCTofControl V1.3, and the genetic algorithm (based on code from JDK), were written by SO′H in (fully compiled) Visual Basic 6.0, the playback functions of Eventcorder being called directly from visual basic via ActiveX (a Windows technology that allows applications to communicate with/control one another).

Table 1. Set of Parameters Used during Optimization, Including Range and Step Sizes[a]

variable	range (unit)	step size	no. of values
sample volume injected	1−5 (μL)	1	5
inlet temperature	200−280 °C	10	9
split ratio	1:2−1:102	1:5	21
helium flow rate	0.8−2.0 (mL·min^{-1})	0.2	7
acquisition rate	1−51 (Hz)	5	11
start temp hold time	4−6 (min)	0.5	5
ramp speed	12−30 (°C·min^{-1})	2	10
final temp	260−300 (°C)	10	5
hold final temp	0−5 (min)	0.5	11

[a] The search space, which is the product of the values in the last column, is some 200,103,750.

(62) Lucasius, C. B.; Kateman, G. *Chemom. Intell. Lab. Syst.* **1993**, *19*, 1−33.
(63) Shaffer, R. E.; Small, G. W. *Anal. Chem.* **1997**, *69*, A236−A242.
(64) Bäck, T., Fogel, D. B., Michalewicz, Z., Eds. *Handbook of evolutionary computation.*; IOPPublishing/Oxford University Press: Oxford, 1997.
(65) Knowles, J. D.; Corne, D. W. *Evol. Comput.* **2000**, *8*, 149−172.
(66) Knowles, J. D.; Corne, D. W. *IEEE Trans. Evol. Comput.* **2003**, *7*, 100−116.
(67) Weuster-Botz, D.; Wandrey, C. *Process Biochem.* **1995**, *30*, 563−571.
(68) Weuster-Botz, D.; Pramatarova, V.; Spassov, G.; Wandrey, C. *J. Chem. Technol. Biotechnol.* **1995**, *64*, 386−392.
(69) Weuster-Botz, D. *J. Biosci. Bioeng.* **2000**, *90*, 473−483.
(70) Davies, Z. S.; Gilbert, R. J.; Merry, R. J.; Kell, D. B.; Theodorou, M. K.; Griffith, G. W. *Appl. Environ. Microbiol.* **2000**, *66*, 1435−1443.
(71) Allen, J. K.; Davey, H. M.; Broadhurst, D.; Heald, J. K.; Rowland, J. J.; Oliver, S. G.; Kell, D. B. *Nat. Biotechnol.* **2003**, *21*, 692−696.

The Eventcorder macro recorder is able to record key stroke/mouse move/click combinations in an editable and programmable form. It differs from other such software in that it tries to address the problems associated with recording events in the Windows graphical user interface (GUI) where the position and hierarchy of the different elements of the GUI are not necessarily fixed from run to run. It does this by recording small regions of the screen at the mouse position during click events and comparing these stored images with those found during playback. To some extent, the software is also able to search for the correct screen position should the playback position on screen become "lost". More importantly, this reduces the likelihood that unwanted or incorrect functions will be triggered during playback.

Despite these features, we found it desirable to minimize mouse use and to plan the recording of the GC/MS operation carefully and in such a way as to avoid "ambiguous" procedures. We nevertheless encountered situations where the software playback of recorded macros lost their screen position and halted, resulting in the need for user intervention; this intervention involved the GC/MS operator resetting the starting screen conditions and reinitializing the genetic algoritm (GA) software. The GA software then searched for the last experiment completed and restarted from the next experiment number. This was performed at the instrument site but could have been performed remotely using remote-operating software. Intervention was typically needed once every day or so.

The GA part of the software was written in C by JDK and translated into Visual Basic by SO′H. While it is of much interest to compare the effectiveness of different flavors of genetic and other algorithms in heuristic search, in the present work, we used a simple multiobjective GA, PESA-II,[72,73] that has already shown strong performance on a suite of well-known and respected test functions (mathematical optimization problems that test for an algorithm's ability to cope with different dimensions of problem difficulty).

In general, genetic algorithms work (partially) on the basis of assigning greater reproductive opportunities to solutions that have higher "fitness". However, in the case of multiobjective optimization, fitness is itself multidimensional, so a further process must be used to convert the raw fitness values associated with a solution into something that can be used by the GA to bias the reproductive opportunities of solutions appropriately. The method used by PESA-II (as for several other multiobjective GAs) is based on the concept of Pareto dominance: given two solutions, one is said to (Pareto) dominate the other if it is better on at least one dimension of fitness, while not being worse on any other dimension; moreover, from any set of solutions, there always exists a subset that is nondominated, that is to say, no other solutions within the set dominate them. These nondominated solutions are highly fit. PESA-II stores such highly fit, nondominated solutions in an "external population", EP, and updates this population, every generation, with new solutions that are generated by reproducing and varying some of them. A block diagram showing the high-level working of PESA-II is given in Figure 1B.

An important aspect of the operation of a multiobjective GA is how diversity in the population is maintained and encouraged. In PESA-II, this is achieved via two closely linked processes. If the external population, EP, becomes too large (larger than a preset size, epsize), then some nondominated solutions must be discarded. To help maintain diversity, PESA-II prefers to discard solutions that are proximal to others in the multiobjective fitness space. In this way, solutions that are more distinctive in their combination of fitnesses are maintained. Similarly, in selecting the "internal population", which actually undergoes reproduction and variation, solutions are selected from the EP in such a way that "unique" or isolated solutions are selected more often than those that populate a densely crowded area of the fitness space. This is achieved by keeping the solutions in "bins" based on their fitness values (i.e., solutions with similar fitness values populate the same bin) and selecting uniformly from among the bins, rather than uniformly from among the solutions. Exact details of these selection and update procedures can be found in ref 73.

GA-Based Parameter Simulation. PESA-II has a number of free parameters that must be set before it can be applied to any specific optimization problem. These include the size of the external and internal population, the type and rate of recombination (crossover), and the mutation rate. While generic settings for these have been derived previously for test functions,[73] in these cases, the number of fitness evaluations available is typically large (of the order of 10 000−100 000) because, for test functions, fitness evaluation of a solution takes a fraction of a second. When applying PESA-II to optimizing "wet" experimental setups, as here, the number of experiments that can be performed is necessarily much more limited. Therefore, we must assume that we are outside the normal operating conditions of the algorithm. To derive appropriate settings, we therefore undertook a large number of in silico simulation runs of the algorithm for different setups, using GA run lengths and numbers and ranges of independent variables (over which to optimize) appropriate to the "real" experiments here. As well as choosing the above parameters of the algorithm, we also investigated the most appropriate number of fitness functions (objectives) to optimize. Our main findings were as follows: that uniform crossover[74] performed better than one-point crossover, that ipsize = 2 should be used, a value much lower than is usual for in silico test functions, probably because this leads to a more aggressive search appropriate to the relatively small number of fitness evaluations available, and that two or three objective functions lead to better results than using four or more objectives. Overall, the parameter settings for PESA-II that we use are as follows: epsize = 200+ (or, as large as the total number of experiments to perform; in the present case, the maxiumum number of nondominated solutions did not exceed this); ipsize = 2; crossover type = uniform; crossover rate = 0.7; mutation rate = $1/L$ (where L is the length of the binary chromosome); number of objectives = 3.

The parameters for each gene were binary-encoded with a number of bits sufficient to cater for the number of possible values each gene could take (Table 1).

(72) Corne, D.; Knowles, J.; Oates, M. *Lecture Notes in Computer Science*; Springer; Paris, France, 2000; pp 869−878.

(73) Corne, D.; Jerram, N. R.; Knowles, J.; Oates, M. *Proc. PPNN VI (LNCS 1917)*; San Francisco, CA, Morgan Kaufmann; 2001; pp 283−290.

(74) Syswerda, G. *Proc. Genet. Evol. Comp. Conf. (GECCO-2001)*; Spector, L., Ed.; Morgan Kaufmann; 1989; pp 2−9.

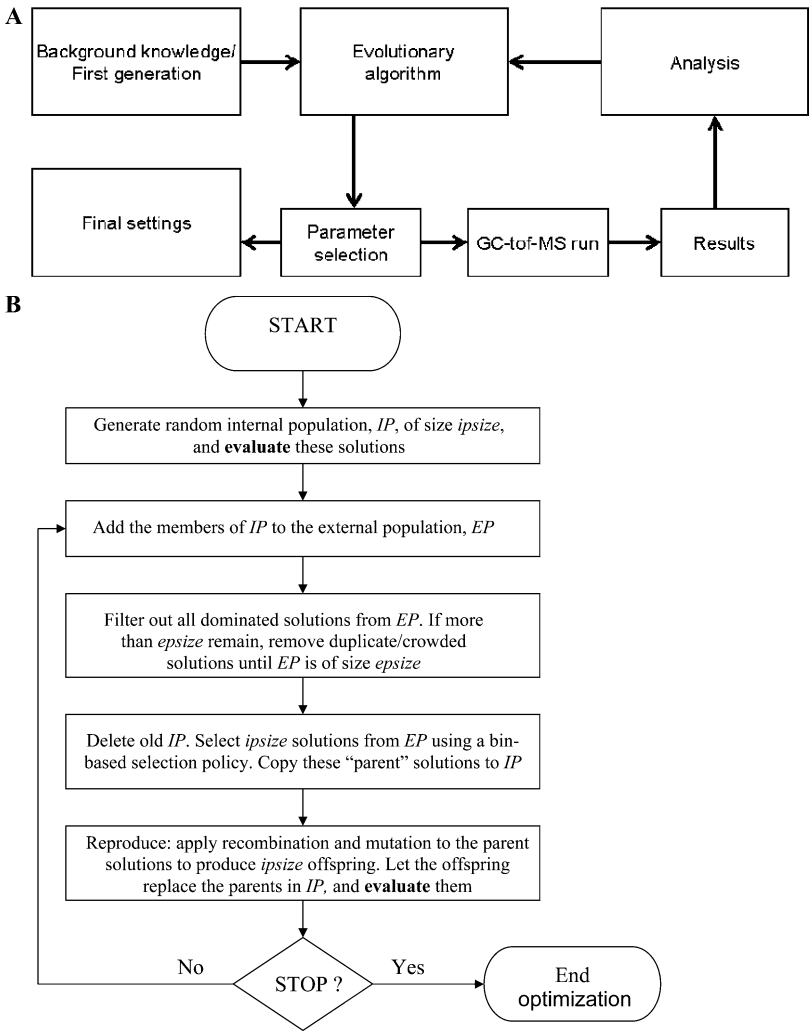

Figure 1. Block diagram of the closed-loop evolutionary optimization process. (A) The overall evolutionary process. Background knowledge is used to parametrize and to "seed" the first generation of the evolutionary algorithm (EA). The system then sets the parameters of the GC-TOF-MS, performs the run, analyses the results, and uses the EA to choose the next set of parameters, run the next experiment, and so on. When a suitable metric or metrics have been produced, the loop exits to provide the experimenter with a choice of "final" parameter settings. (B) The PESA II algorithm uses the concept of Pareto dominance to differentiate between better and worse solutions (here, the quality of GC-TOF conditions, as judged by our three measures). Nondominated solutions are stored in an "external population", which is the memory of the best solutions found so far. Each generation of PESA-II selects, from this external population, a smaller internal population, which will "reproduce" using recombination and mutation. The resulting "offspring" solutions are evaluated (by running the required experiment) and, if they are nondominated, join the evolving external population. This loop continues until a desired level of performance has been reached.

GA-Based Parameter Optimization. One particular feature of the present LECO GC/MS deconvolution software is that it tends to generate artifactual peaks corresponding to noise (as judged by the mass spectrum and TIC chromatogram) and to produce duplicate or multiple peak assignments that (again from the mass spectra) clearly correspond to a single chromatographic peak and chemical entity. Such artifacts can account for 10−20% of the peaks in the chromatogram. With the current version of the LECO software, the operator has little control over the deconvolution parameters to remedy this.

In GC/MS applications in which the analyst is able to scrutinize the chromatogram (e.g., where there are a small number of compounds or a targeted compound method), these artifact peaks present little problem. However, for metabolomics studies, and for this closed-loop optimization in particular, it is desirable to be able to reject such peaks automatically.

In an attempt to obtain a "true" measure of the number of peaks in the chromatogram, we filtered out noise peaks based upon the peak characteristics, which were exported into the GA software. As we had no theoretical basis on which to filter out artifact peaks, we chose to use a simple GA to obtain an empirical filter function. Initially, principal components analysis (PCA) and a genetic programming strategy using the gmax-bio software (see below) were used to model a training set of data for which peaks had been designated as "noise", "multiple", or "true peak" by an experienced GC/MS analyst. Six data sets were employed from the analysis of a single sample with widely different instrument parameters. The gmax-bio models showed that no single peak parameter could account for the duplicate and multiple peaks but that the noise peaks were mainly dependent on the peak full width at half-height, fwhh, and the "purity", *P* (a peak parameter generated by the LECO software, defined as

"a measure of how much the unique mass coelutes with other compounds").

Inspection of the training data set using a plot of P versus fwhh, suggested rather simple criteria, viz:

$$\left.\begin{array}{l} P < r \\ \text{or} \\ \text{fwhh} \times k - P \geq 0 \end{array}\right\} \text{flag as true peak}$$

Here r and k are empirical parameters to be optimized using the genetic algorithm approach. For the GA fitness function, F, we chose to minimize, $F =$ (number of false positives + number of false negatives − number of correctly flagged peaks).

Using a simple single-objective GA running under Microsoft Excel, near optimum values of k and r were found to give, for the yeast data, a correct ID (in terms of noise/not noise) for 87% of peaks in the training set, with 3.5% false negatives, and 9.4% false positives. The serum training data set gave somewhat different values of k and r, but similar performance.

The noise filter function above was incorporated into the closed-loop GA software to give a "truer" estimate of the number of peaks in each experimental run. It was this corrected peak count that was optimized (see below) and that was used, and that appears as such, in all the data on peak numbers.

We were unable to find a filter function to flag duplicate and multiple peaks from the chromatogram using a similar strategy, even when we extended the method to look at the similarity between the mass spectra of adjacent chromatographic peaks. However, while the GA closed-loop optimization does not at present address the deconvolution or integration parameters that contribute to the occurrence of multiple artifact peaks, we do expect that the ratio of multiple peaks to true peaks will remain more or less the same.

Genetic Programming. Genetic programming was carried out using the commercially available software gmax-bio (Predictive Solutions/Aber Genomic Computing http://www.predictivesolutions.co.uk). The inputs were the fields (presented via an Excel sheet) representing the independent variables while the outputs were the objective functions such as peak number, signal/noise ratio, and so on. Default parameters were used.

RESULTS

Figure 1 shows the essential strategy of the closed-loop optimization method. As to the fitnesses, we chose to optimize three separate objectives, which were chosen to reflect distinct features of the separation.

S/N: the mean signal-to-noise ratio for the 15% of peaks with the worst signal-to-noise found in the sample. Signal-to-noise is calculated by the LECO software as the peak height from corrected baseline/noise for the ion of unique mass calculated by the LECO deconvolution software. This unique mass can vary between analyses, and hence, the data are at best an estimate of S/N as different unique ions of different intensities will provide different S/N values. Most peaks, in fact, have very good signal-to-noise virtually irrespective of instrument settings; taking the average of the worst 15% of peaks thus gives a more sensitive fitness function, and the lowest value of S/N permitted by the software was 10. This objective was chosen as larger S/N correlates with both lower detection limits (and also detection of greater numbers of peaks) and greater precision, both important in metabolomic studies.

Peaks: the number of peaks in sample minus the number of peaks flagged as noise peaks. As the noise filter function is not perfect, this is at best an estimate of the true number of peaks. This objective is chosen as in metabolomic studies analyses are not targeted but are nonbiased and so detection of the greatest number of peaks possible is required.

Run time: the time between the injection of sample and the end of data collection. The large number of samples required to be analyzed in metabolomic studies makes short run time desirable. The ability of the LECO software to deconvolute overlapped chromatographic peaks also makes it possible that more compressed chromatograms can be usefully utilized than is perhaps the case in traditional GC/MS software systems.

Figure 2 shows the run from the first generation (using human serum) that had the larger number of peaks. Both the conditions and the outcome are similar to those described previously.[27] While >500 peaks is very respectable, we have no a priori way of knowing the "true" number of realistically observable peaks in a particular metabolome. (There are sampling theories that, given sufficient runs, might allow us to make reasonable estimates[75,76] (and see later)).

Since the closed-loop method is entirely automated (in practice it was restarted three times—see above), we simply allowed the system to evolve improved conditions for 120 generations (i.e., 240 experiments) over a period of some 118 h, with the results shown in Figure 3. Figure 3 encapsulates many of the features of the improvements obtained. Thus, the number of peaks that can in fact be observed in the same sample is more than double the larger number observed in generation 1. However, there is a clear relation between the number of peaks observed and the run time, even when all the other parameters change over a wide range. The Pareto front represents the set of so-called nondominated solution, in which individuals cannot be improved in any dimension of fitness without suffering a deterioration in at least one other dimension. Here, the Pareto front is extremely broad, allowing many tradeoffs between peak number and run time, depending upon the experimenter's preferences. There is a slight tendency for the S/N to improve as the run time decreases, but the numerical values, typically between 10 (the minimum allowed by the software) and 20, suggest that this is not therefore a major issue. Thus, peak number and run time, both of which are important in metabolomics experiments with large data sets, are the chief focus of our interest.

To understand better those parameters which most affected the number of peaks, we ran a genetic program (GP; see for example, refs 59, 60, and 77) to model the process. The purpose of the GP is to evolve a model that best explains the output in terms of a restricted subset of inputs that are allowed to interact in a (potentially highly) nonlinear manner. The results of this

(75) Kallel, L.; Naudts, B.; Reeves, C. In *Theoretical aspects of evolutionary computing*; Kallel, L., Naudts, B., Rogers, A., Eds.; Springer: Berlin, 2001, pp 175−206.
(76) Reeves, C. R. *Genetic algorithms: principles and perspectives: a guide to GA theory*; Kluwer Academic Publishers: Dordrecht, 2002.
(77) Kell, D. B. *Trends Genet.* **2002**, *18*, 555−559.

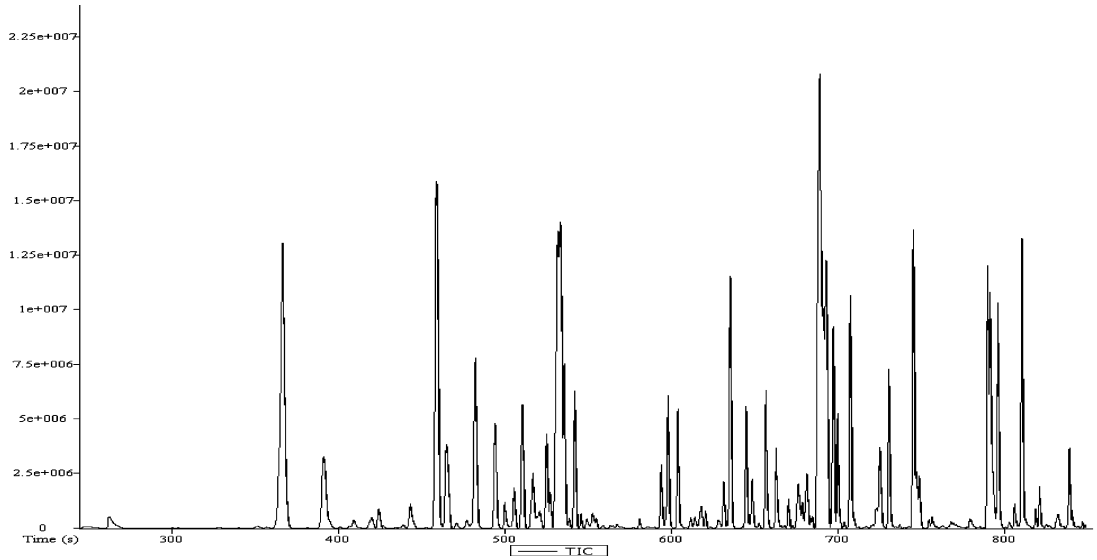

Figure 2. Human serum TIC chromatogram for generation 1, experiment 2. Sample volume 4 μL, Injection temperature 270 °C, split ratio 1:57, flow rate 2.0 mL.min^{-1}, acquisition rate 15 Hz, initial hold time 5 min, GC temperature ramp 24 °C·min^{-1}, final GC temperature 290 °C, and final hold time 0 min.

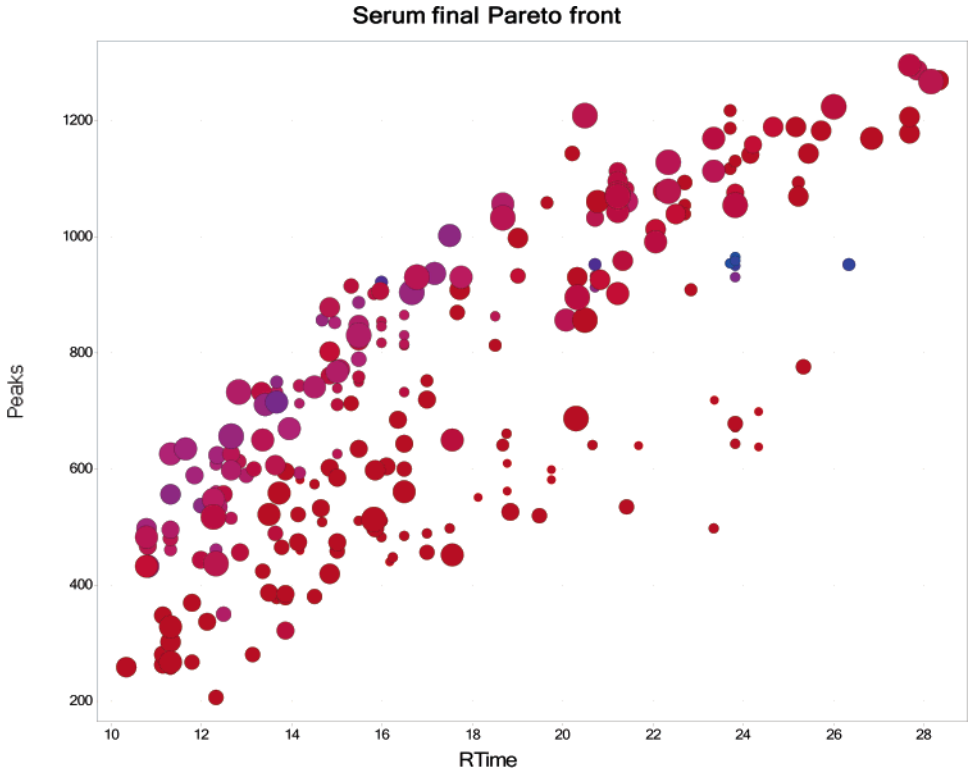

Figure 3. Evolution of GC-TOF conditions for the optimal separation of typical serum metabolites. The diagram shows the 2 main outputs (peak number and run time) for each trial separation in 120 generations (240 examples). The generation number is encoded in the size of the symbol (larger = later) and the signal/noise ratio via the color (bluer = higher). The peak number is the "raw" peak number including duplicates provided by the LECO software after correction for noise peaks.

(Figure 4) indicated that the ramp, the on-column volume, and the final temperature were the chief determinants of this, and that is illustrated in Figure 5.

Other features (data not shown) were the relative independence of good solutions from the initial hold time, flow rate, and sample volume per se, an optimal inlet temperature of 270 °C, a preferred final temperature of 290 or 300 °C, a split ratio below 10 coupled to an on-column volume of 0.5 μL, the lack of need for a final hold time, and an optimal acquisition rate of 15 Hz. The analyses were performed in split mode with a maximum on-column volume of 0.5 μL. Results showed an injection volume of 2.0 μL and split ratio less than 10 were optimal. These are similar to splitless conditions. To assess whether splitless injections would be preferable, three replicate injections were performed at the optimized conditions with the exception of split ratio and injection volume. Instead, an injection volume of 1 μL was used in splitless mode (purge after 60 s). The results (not shown) provided less optimal results in terms of peak number.

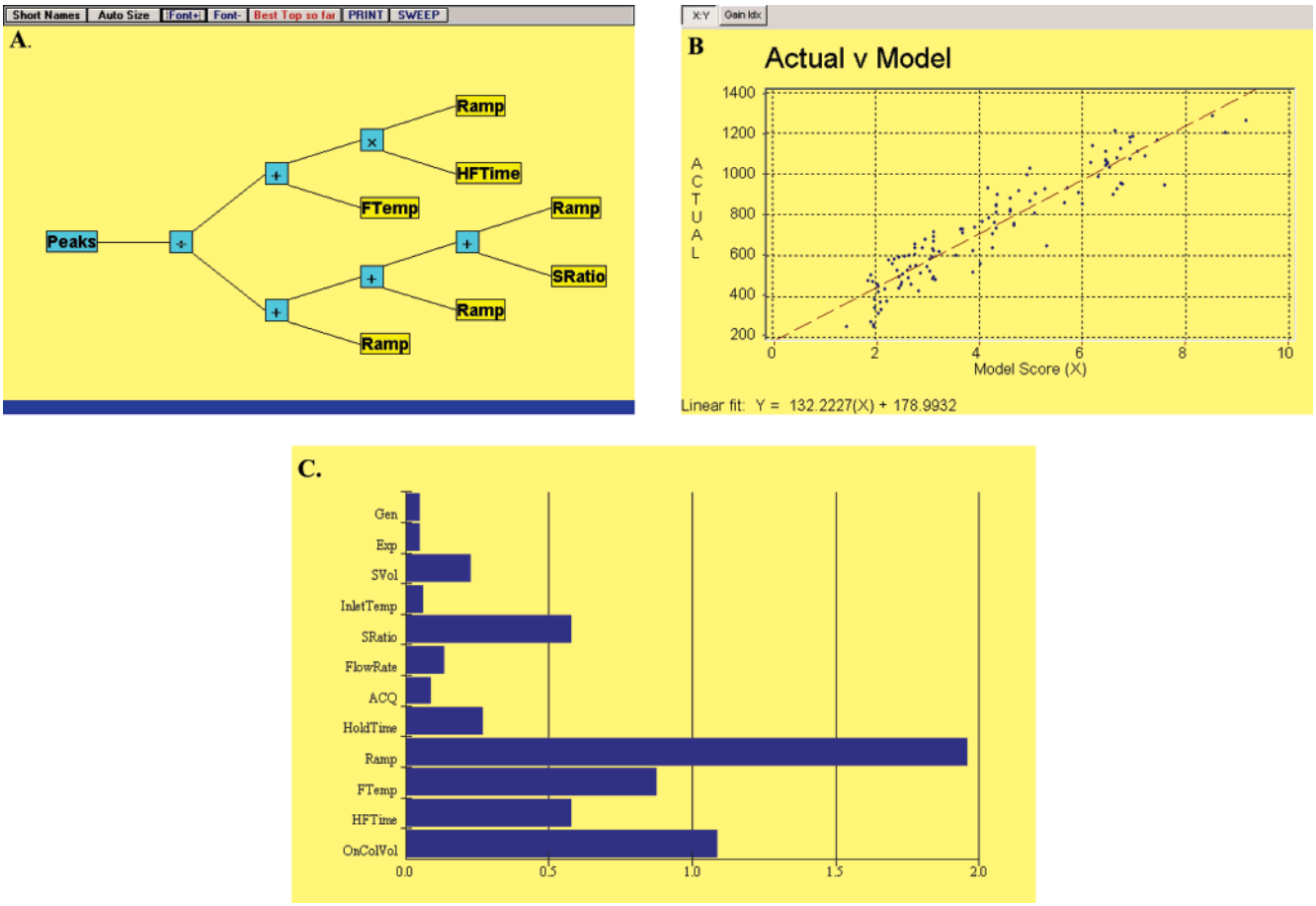

Figure 4. Genetic programming analysis of the GC- optimization process for serum. Data for all 120 generations, including both the independent variables (Table 1) and the outcomes (S/N, peak number, run time) in the form of an Excel sheet were imported into the software gmax-bio and a model evolved, using default settings, that best accounted for the data. A model for the peak number, obtained after 46 generations, is shown. (A) Typical model in the form of a tree. (B) Effectiveness of the model in explaining the number of peaks. (C) Frequency distribution of the use of the different input variables used in the top 300 models following training.

The result of this is that a series of optimal conditions are obtained, from which it was possible to choose that or those which represented, for the experimenter, the best tradeoff between number of peaks and run time. Inspection of Figure 3 suggested that the run with 1208 peaks and a run time of 20.5 min (from experiment 2 in generation 113) represented an excellent choice, and the relevant chromatogram is shown in Figure 6. On further investigation of the analysis results by a GC/MS expert, 228 noise peaks were identified (the software calculated 164 noise peaks) and 193 duplicate peaks were identified, of which 78 were identified as pyridine (solvent peak) at the beginning of the chromatogram. Although a solvent delay was used, this was set at 240 s at the beginning of experimentation and not changed as flow rate was varied. Therefore, 951 true peaks were observed. It should be noted that the method of derivatization can produce more than one derivative for a single metabolite, and therefore, this peak number does not equal the number of metabolites detected. (We note that the application of GC×GC-TOF-MS[78,79] should both increase the number of peaks detected (by improved chromatographic separation and improved S/N in the second-dimension GC column) and reduce the run time (by improved separation conditions).)

Not all biological samples are of course alike, and the yeast supernatants are expected a priori to differ quite markedly from the serum samples, in both the number and nature of the metabolites formed. This is because, for instance, the yeast supernatant metabolome is dominated in terms of mass by some 20 metabolites (Figure 7), which compared to the serum constrains the allowable split ratio much more. Figure 8 shows the number of peaks and run time (in the same manner as did Figure 3) for the evolution of the conditions for the yeast supernatant experiments over 114 generations. In this case, there are noticeably fewer peaks observable, and the structure of the evolving Pareto front is significantly different from that observed for serum. There is again a very wide variation in the peak numbers and run times observed. The algorithm here places especial emphasis on a short run time, although some regions of the search space do also permit a high number of peaks to be observed.

An inspection of the search space showed that high sample volumes were preferred, but with the higher split ratios favored to give on-column volumes near to 0.1 μL, an inlet temperature of 270 °C was probably optimal, that the preferred final temperature was 290 °C, that peak numbers were essentially independent

(78) Blumberg, L. M. *J. Chromatogr., A* **2003**, *985*, 29−38.
(79) Wilson, I. D.; Brinkman, U. A. *J. Chromatogr., A* **2003**, *1000*, 325−356.

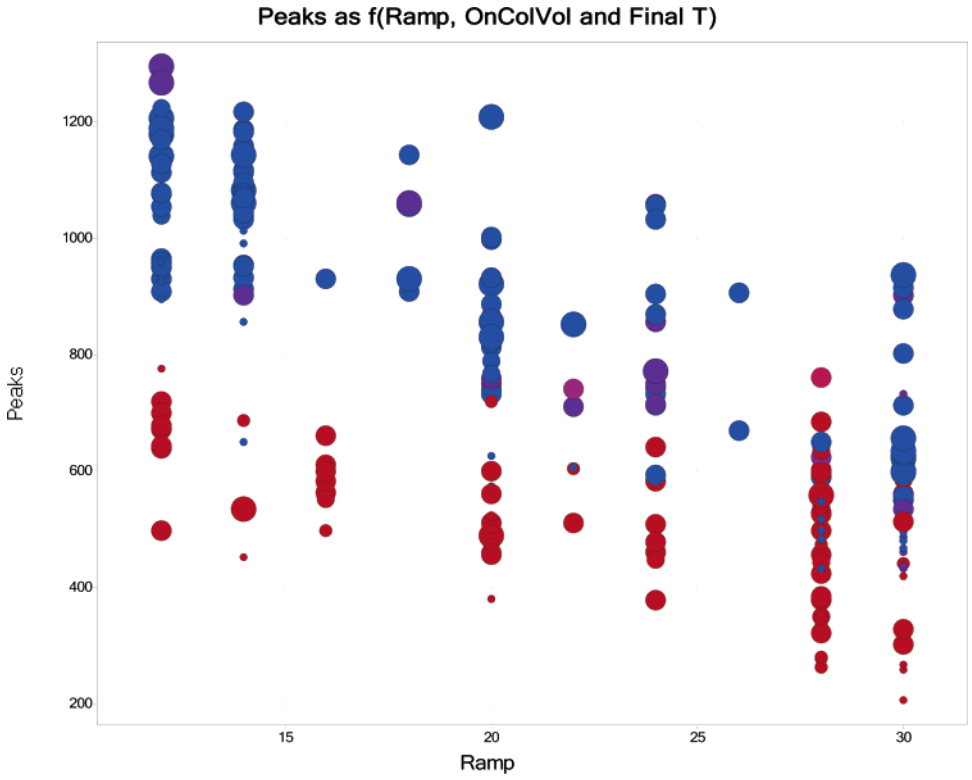

Figure 5. Effect of the size of the temperature ramp on the number of peaks. Additionally, the final temperature is encoded in the size of the symbols and the on-column volume is encoded by color (blue larger).

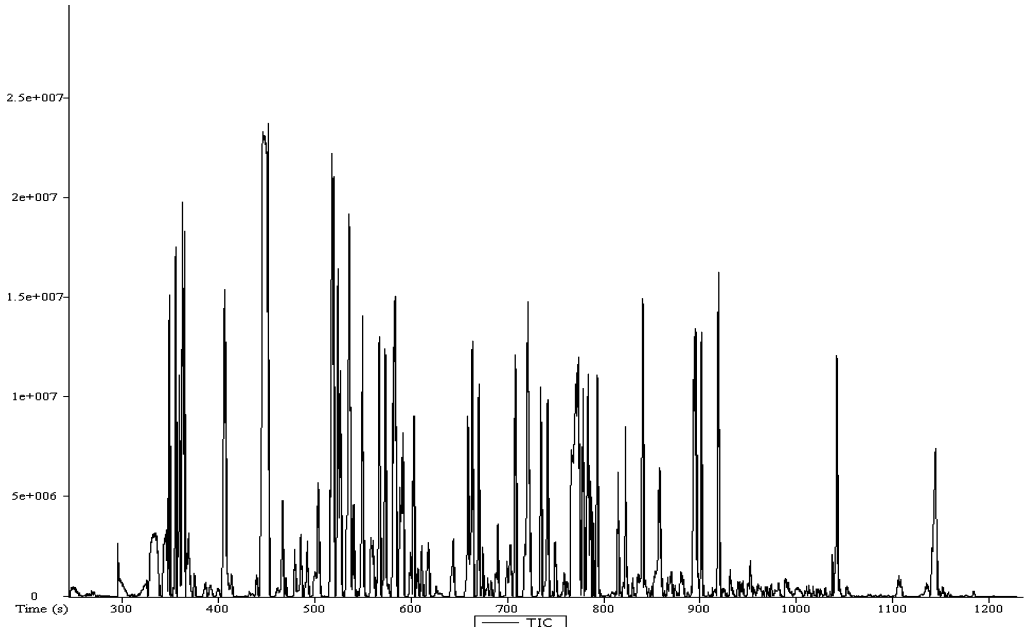

Figure 6. Human serum TIC chromatogram for optimized conditions (generation 113, experiment 2). Sample volume 2 μL, injection temperature 270 °C, split ratio 1:3, flow rate 0.8 mL·min^{-1}, acquisition rate 15 Hz, initial hold time 4.5 min, GC temperature ramp 20 °C·min^{-1}, final GC temperature 300 °C, and final hold time 4.5 min.

of flow rate and hold time, a preferred acquisition rate was 10 Hz, and that—presumably for reasons connected to the run time—fast ramps were preferred. Final hold times of 0 or 1 min were also preferred.

To finalize an optimal set of conditions for our own "metabolic footprinting" experiments in yeast (cf. ref 71), we finally ran a local search around the optimum found above in which just the split ratio (with a 5-μL sample volume) and inlet temperature (230, 250, 270 °C) were varied systematically. This led us to our final set of conditions that are given in Table 2 and the chromatogram shown in Figure 9.

Landscape Analyses. The properties of the "search landscape" of the optimization problem (that is to say, the surface obtained when fitness is plotted in the z dimension, against the independent variables in the $x-y$ plane) are also important and interesting for us because they can give us clues as to the most

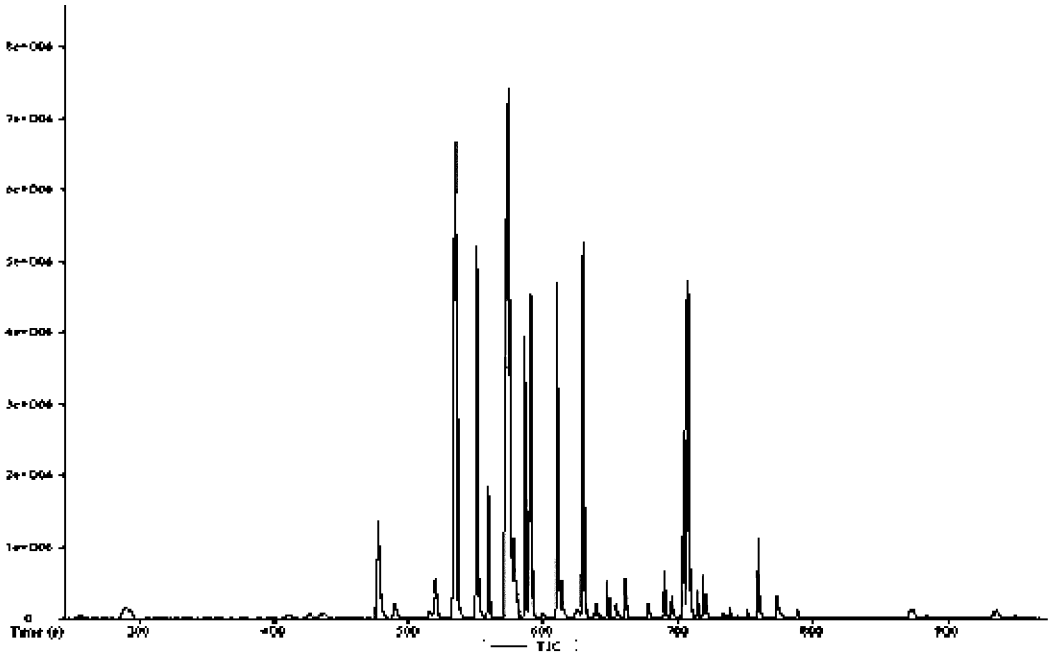

Figure 7. Yeast supernatant TIC chromatogram for generation 1, experiment 2. Sample volume 4 μL, injection temperature 270°C, split ratio 1:57, flow rate 2.0 mL·min^{-1}, acquisition rate 15 Hz, initial hold time 5.0 min, GC temperature ramp 24°C·min^{-1}, final GC temperature 290 °C, and hold time 0.0 min.

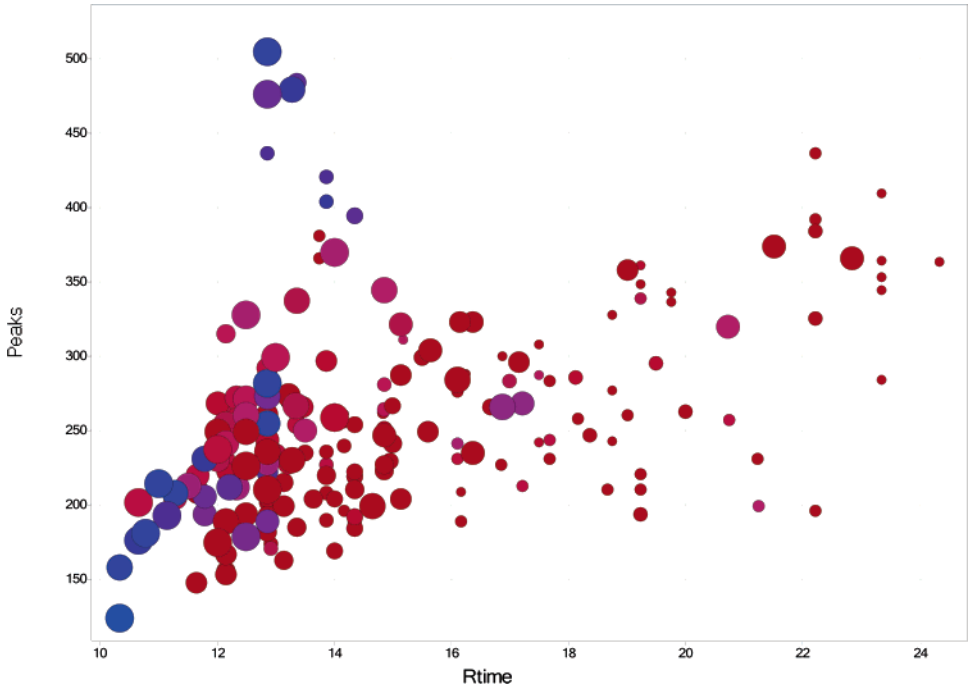

Figure 8. Evolution of GC-TOF conditions for the optimal separation of typical yeast supernatant ("metabolic footprint") metabolites. As in Figure 3, the diagram shows the two main outputs (peak number and run time) for each trial separation in 114 generations (228 examples). The generation number is encoded in the size of the symbol (larger = later) and the signal/noise ratio via the color (bluer = higher). The peak number is the "raw" peak number including duplicates provided by the LECO software after correction as described in the text for noise peaks.

efficient and effective ways to search this and similar problems in future. In simple terms, landscapes that are smooth and contain few local optima can be searched readily by simple hill-climbing methods or even classical fractional factorial design of experiments. More rugged landscapes indicate that fitness is a more complex, nonlinear function of the variables, and in these cases, genetic search may be more appropriate. Ultimately, we would like to be able to parametrize our GA (i.e., choose the mutation rate, crossover type, population size, etc.) as a function of observed landscape properties and using prior experience with the best GA settings as a function of these. At present, the technology for doing this is not available, although several promising methods for measuring landscape properties have been proposed in the relevant literature.[33,75,80−83]

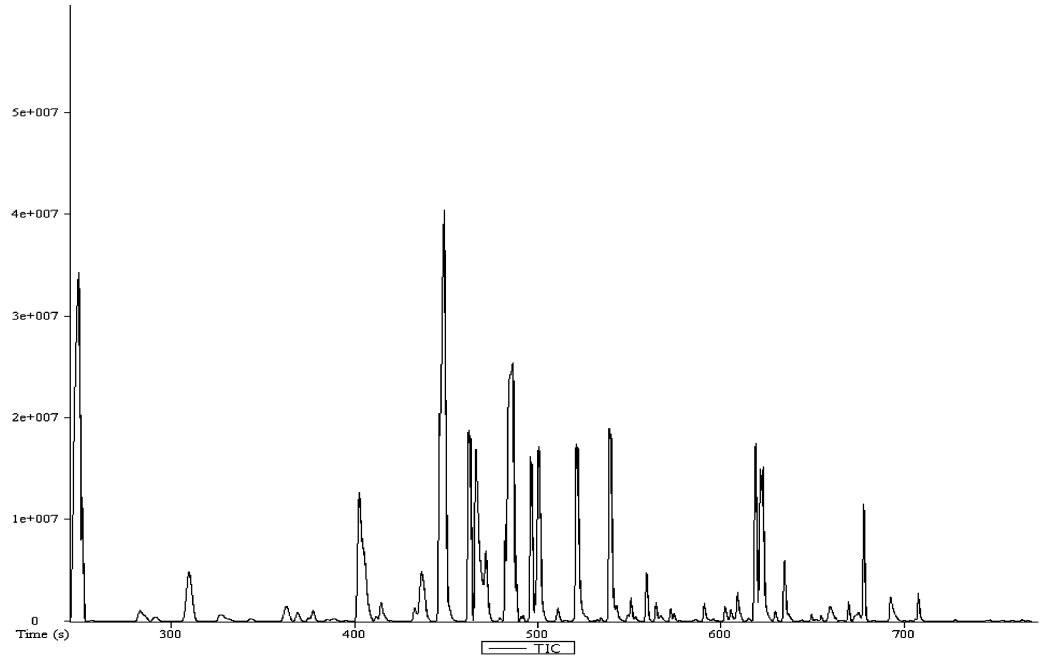

Figure 9. Yeast supernatant TIC chromatogram for optimized conditions (generation 113, experiment 2). Sample volume 5 μL, injection temperature 270 °C, split ratio 1:45, flow rate 1.0 mL·min^{-1}, acquisition rate 10 Hz, initial hold time 4.0 min, GC temperature ramp 28 °C·min^{-1}, final GC temperature 290 °C, and hold time 1.0 min.

Table 2. Conditions Chosen as Optimal for Human Serum and Yeast Supernatant

variable	human serum	yeast supernatant
sample volume injected (μL)	2	5
Inlet temperature (°C)	270	270
split ratio	1:3	1:45
helium flow rate (mL·min^{-1})	0.8	1
acquisition rate (Hz)	15	10
start temp hold time (min)	4.5	4
ramp speed (°C·min^{-1})	20	28
final temp (°C)	300	290
hold final temp (min)	4.5	1
initial GC temperature (°C)	70	70
detector voltage (V)	1700	1700
run time (min)	20.5	12.2

Table 3. Spearman Rank Correlation Coefficients (with Correction for Tied Values) for Each Objective in the Serum Experiments and for the Nondominated Rank vs Each Parameter of the Instrument Setup

	S/N	peaks	run time	nondom rank
sample vol	−0.014	0.066	0.144	0.140
injector temp	0.160	0.130	0.046	−0.068
split ratio	−0.561	−0.542	−0.112	0.670
flow rate	0.027	0.234	0.273	0.017
acquisition rate	0.189	0.263	0.200	0.094

Table 4. Spearman Rank Correlation Coefficients (with Correction for Tied Values) for Each Objective in the Yeast Experiments and for the Nondominated Rank vs Each Parameter of the Instrument Set-up.

	S/N	peaks	run time	nondom rank
sample volume	0.530	0.401	−0.004	−0.266
injector temp	0.106	0.400	0.392	−0.081
split ratio	−0.241	−0.600	−0.440	0.196
flow rate	−0.045	−0.148	0.067	0.137
acquisition rate	−0.499	−0.183	0.224	0.644
init hold time	−0.181	0.089	0.522	0.466
ramp	0.463	0.004	−0.350	−0.126
final temp	0.130	0.428	0.439	0.228
final hold time	0.028	0.217	0.269	0.297

A simple method for gaining an understanding of the relationships between variables and fitness(es), which we can readily apply to our data, is to measure their correlation coefficients. We have chosen to use the Spearman rank correlation because this measures both linear and nonlinear correlations. Tables 3 and 4 give the correlation coefficients for each variable independently, against each of the three fitnesses (first three columns). The fourth column gives the correlation between the variable and the nondominated rank of the solution (which is an overall measure of its fitness —lower ranks being fitter). Several observations can be made: (1) that the two landscapes are significantly different from each other; (2) that all variables make some contribution (positive or negative) to at least one measure of fitness; (3) that a few variables make large contributions. Taken together, these observations indicate that a good setup of the instrument is indeed a rather complicated function of the independent variables and tend to support the choice of a GA to optimize this setup. We can also see from these tables that a low acquisition rate and small hold time are the most important determinants of good solutions in the yeast experiments, whereas a low split ratio seems most

(80) Davidor, Y. *Complex Syst.* **1990**, *4*, 369−383.
(81) Weinberger, E. *Biol. Cybernet* **1990**, *63*, 325−336.
(82) Rosé, H.; Ebeling, W.; Asselmeyer, T. In *Proc. Fourth Conf. Parallel Problem Solving from Nature (PPSN IV)*; Voigt, H.-M., Ebeling, W., Rechenberg, I., Schwefel, H.-P., Eds.; Springer: New York, 1996; pp 208−217.
(83) Merz, P.; Freisleben, B. In *New ideas in optimization*; Corne, D., Dorigo, M., Glover, F., Eds.; McGraw-Hill: London, 1999; pp 245−260.

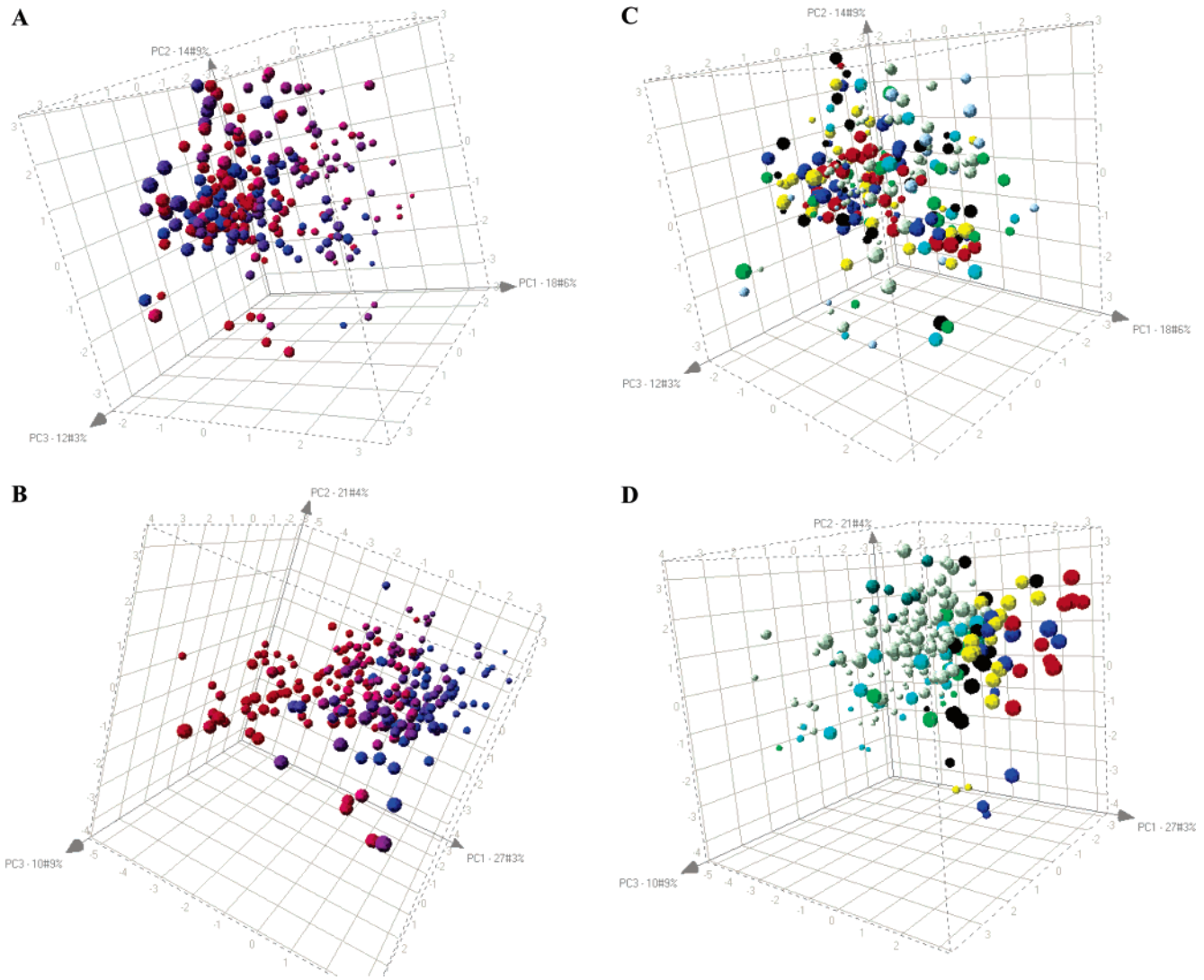

Figure 10. Visualization of the optimization landscapes using PCA. The dimensionality of the independent variables was reduced from 9 to 3 by taking the scores of the principal components of their variance. Such landscapes are shown for both the serum (A) and yeast supernatant (B) optimizations, with the peak numbers encoded by the size of the symbols and the generation by the color (blue = later). To illustrate the multiobjective landscape, the rankings are shown for serum (C) and yeast (D) with the code red (1), blue (2), yellow (3), black (4), green (5), cyan (6), and gray (lower ranks) and with the generation number encoded by size.

important in the serum experiments. However, there were numerous differences, as can be seen from the optimal values recorded in Table 2.

Another way of understanding the nature of the search space is to visualize it by reducing its intrinsic dimensionality to 2 or 3. This is conveniently done using PCA on the independent variables.[60] A typical analysis of this type showing just the data for peak number is shown in Figure 10 for both serum (Figure 10A) and yeast supernatant (Figure 10B). Another way of visualizing the landscape that simultaneously incorporates all three objectives is to indicate the "nondominated rank" of each individual, where an individual of rank 1 is on the Pareto front, an individual of rank 2 is on the Pareto front when all individuals of rank 1 have been removed, and so on. These plots are shown in Figure 10C and D. A number of points emerge: (i) the landscapes are significantly different both in terms of their variance (the first three PCs accounting for 44 and 60% of the variance in the two cases) and in the manner in which the GA explores it, (ii) the GA nevertheless explores the landscape very effectively, and does home in on the optimal regions; (iii) the landscapes are significantly less epistatic than those studied previously for electrospray mass spectrometry.[60] Loadings plots (not shown) also reflect the individual variables that most contribute to the variance that is explored via the guided genetic search.

DISCUSSION

Many problems in analytical science (and in science more generally) can in fact be represented as combinatorial optimization problems, in which even a modest number n of parameters, each of which can take m values, allow m^n combinations. This number (obviously) scales exponentially with n and soon reaches astronomical values in which exhaustive testing is quite impossible. In the case of electrospray mass spectrometry we identified 14 such parameters,[60] while for the GC-TOF separations in the present work we chose to study 9. In the present case, we recognized that there were at least three objectives we wished to optimize, viz. the number of peaks observed, the run time, and the signal/noise ratio of the 15% of peaks with the lowest S/N,

such that this was a multiobjective optimization problem. Classical design of experiment strategies can neither usefully handle these numbers of variables nor are able effectively to cope with genuinely multiple objective functions. By the use of macro recoding software, which acted as a "shell", we developed an entirely automated, closed-loop machine learning strategy for such optimizations that does not require access to the source code of the manufacturer's software and that exploited a modern and effective multiobjective optimization heuristic, the PESA-II algorithm.

Metabolomics in its usual definition (e.g., refs 27 and 84) is a generic strategy in which we seek to measure all the metabolites in a particular sample, which may typically contain hundreds or thousands of individual species. The datacentric philosophy[61,85] behind such metabolome studies means that in addition we often wish or need to run very large numbers of samples. Two important objectives in metabolomics are therefore to maximize the number of peaks observed while minimizing the run time, objectives that are to some extent likely to be orthogonal or even negatively correlated. We applied our strategy successfully to the (separate) optimizations of the GC-TOF-MS analysis of both serum and yeast fermentation broth supernatants.

A number of points are worth discussing here. The first is that "one size does not fit all"; i.e., the nature of the two types of sample means that good instrumental conditions (and indeed the search space for exploring and finding them) are quite different for the two matrixes (serum vs yeast broth). Further variables that could be studied include the derivatization conditions (solution volumes, derivatization temperatures, times) and data processing variables (baseline, peak smoothing, peak width, S/N threshold). Nevertheless, the evolutionary algorithm used was well able to find excellent solutions that provided acceptable tradeoffs between the largely (and demonstrably) incompatible objectives chosen. In each case, in just over 100 experiments, we were able to explore an effective search space of combinations that was more than 2 million times greater.

Although this number is a small fraction of the search space, it is still a reasonably large number by normal standards of instrument optimization, and it is appropriate to check that the algorithm is not simply doing a random search. Indeed, observation of some of the plots of peaks, S/N, and run time, against generation number can give the impression that the GA does not seem obviously or monotonically to evolve better solutions over time. To establish whether this is the case, we can estimate the statistical likelihood of a random search (i.e., where the setup parameters are chosen at random, rather than by evolution) generating the same or a greater progress than the GA. Our method for computing this is as follows. We take the experiments (for either yeast or serum, independently) in the order they were performed and label them 1 to N. We then compute the nondominated rank of each experiment. The sum over the products of the label and the rank is then computed. Clearly, if this sum is small then it means that better solutions (those with lower rank) must have been generated later (are paired with higher labels); whereas for a random search, we would expect that the distribution of the different ranks over the experiments would be random. We computed the likelihood of the sum being less than the observed value, by random chance alone, by computing the sum of the products for 10 000 different random orderings of the experiments, and by comparing this distribution to the observed sum of products, using a Student's t test. The results of this computation are as follows, first for serum: $N = 239$, observed product of ranks and labels = 114 386, mean (10 000 random orderings) = 134 099, SD = 3295.96, t value (observed versus random) = 5.98, p (observed not better than random) $\ll 0.001$. Equivalently for yeast, we have (for the GA experiments only, not the final local search): $N = 226$, observed product of ranks and labels = 142 669, mean (10 000 random orderings) = 166 607, SD = 3914.64, t value (observed versus random) = 6.11, p (observed not better than random) $\ll 0.001$. It is obvious that the success of the GA is not due to chance.

The LECO software deconvolutes coeluting derivatized compounds with peak apices separated generally by more than 0.05 s. However, the software can observe noise as peaks and identify single peaks multiple times. Although this is dependent on operator-set software variables (baseline, peak smoothing, peak width, S/N threshold) there is no one set of variables that will be successful for all peaks in a metabolomics analysis with a wide range of metabolites and metabolite concentrations. Therefore, the peak numbers reported are estimates rather than accurate descriptions. Other parameters could be optimized, such as the initial GC temperature, or we might have used nonlinear GC temperature ramps or operated the GC in splitless mode. However, it was decided to run a linear GC ramp to reduce the search space size (and simplify the GC operation) and to run in split mode to ensure that the column was not overloaded (the maximum on-column volume was set at 0.5 μL) and that the column lifetime was maximized by reducing contamination onto the column. A further investigation showed no benefit in peak number when operating in splitless mode. Also, different GC columns could be investigated in a similar manner, for example, DB-5 columns as used by a number of metabolomics groups. With a certain sample type, although generally similar in nature, the number of peaks observed will vary significantly (as we have observed in the analysis of a wide range of yeast supernatants to be reported elsewhere). Therefore, the conditions chosen are not "globally" optimal but a compromise set based on the analysis of one sample. Thus, it is implicitly assumed that samples within a sample type will not vary excessively.

With the chosen optimized conditions, 10 replicate analyses were performed for serum and yeast supernatant and the precision (coefficient of variation, $n = 10$) for 12 randomly chosen metabolites with a range of area responses was calculated. Precision was 11−30% with no internal standard (peak area determined) and 2−16% when a metabolite peak was used as an internal standard (ratio peak area (internal standard)/peak area (metabolite) determined).

In conclusion, the strategy of using a macro recording "shell" script means that the present approach may be modified and applied to any computer-controlled instrumentation for which the executable code is available, even when it is controlled by software that uses an operating system with a window-based GUI. Clearly such a system might also be used to explore in an entirely automated manner (i.e., without human intervention) the effectiveness of different algorithms for solving any combinatorial optimi-

(84) Fiehn, O. *Comp. Funct. Genomics* **2001**, *2*, 155−168.
(85) Breiman, L. *Stat. Sci.* **2001**, *16*, 199−215.

zation problem in which the objective function requires a "wet" experimental evaluation.

ACKNOWLEDGMENT

We thank the BBSRC, EPSRC, Royal Society of Chemistry, and NERC for financial support. D.B.K. is EPSRC/RSC Research Chair in Bioanalytical Science, and J.D.K. is a BBSRC David Phillips Fellow.

Received for review June 9, 2004. Accepted October 12, 2004.

AC049146X