Chapter 16

APPLICATION ISSUES OF GENETIC PROGRAMMING IN INDUSTRY

Arthur Kordon[1], Flor Castillo[1], Guido Smits[2] and Mark Kotanchek[3]
[1]*The Dow Chemical Company*, Freeport, TX; [2]*Dow Benelux*, Terneuzen, The Netherlands; [3]*Evolved Analytics*, Midland, MI

Abstract This chapter gives a systematic view, based on the experience from The Dow Chemical Company, of the key issues for applying symbolic regression with Genetic Programming (GP) in industrial problems. The competitive advantages of GP are defined and several industrial problems appropriate for GP are recommended and referenced with specific applications in the chemical industry. A systematic method for selecting the key GP parameters, based on statistical design of experiments, is proposed. The most significant technical and non-technical issues for delivering a successful GP industrial application are discussed briefly.

Keywords: Genetic programming, symbolic regression, industrial applications, design of experiments, real world problems, parameter selection

1. Introduction

Recently, Genetic Programming (GP) has demonstrated its growing potential to resolve various industrial problems in modeling, process monitoring and optimization, and new product development (Kotanchek *et al*, 2003). In parallel to the theoretical development in the area of GP, much effort has been spent in developing a robust methodology for practical implementation that is applicable for a broad range of solutions. Unfortunately, the industrial application efforts are not so well published as the theoretical development and are virtually unknown to the research community. The objective of this chapter is to present a systematic view of the key results from exploiting GP in a large global company, such as The Dow Chemical Company.

The chapter is organized in the following manner. Some guidance on finding practical problems which are appropriate to be resolved by GP is given in Section 2. A methodology for selecting robust key GP parameters, based on Design Of Experiments (DOE), is described in Section 3. The key technical and non-technical issues to be resolved for successful GP applications in industry are presented in Section 4.

2. When is Genetic Programming an Appropriate Industrial Solution?

One of the significant factors for success in the current industrial R&D environment is the speed of introducing an emergent technology into practice. Usually a new technology is introduced in two phases: (1) capability exploration and (2) proof-of-concept application. In the first phase, the features of the technology are assessed and matched with the existing specific needs of each industry. An important component is the estimate of the potential effort for adopting the new technology into the existing work processes in research and manufacturing. Critical for business acceptance, however, is the second phase, which includes a convincing demonstration of the benefits in a well-selected case study. Usually it is based on real data and very often illustrates a novel solution to a difficult industrial problem.

The first question that needs to be addressed in any new technology introduction is a clear definition of its competitive advantages relative to other, similar approaches.

Competitive Advantages of Genetic Programming

Computational intelligence is a research area that includes many competitive approaches with different technical nature (fuzzy logic, evolutionary computation, neural networks, swarm intelligence, *etc.*) for solving complex practical problems. On the one hand, this opens new opportunities and broadens the scope of potential applications. On the other hand, however, it requires additional efforts from industrial practitioners to understand the technical features of very diverse technologies and to estimate their potential value. The comparative analysis is not trivial and has to take into account not only the relative technical advantages but also the total cost-of-ownership (potential internal research, software development and maintenance, training, implementation efforts, *etc.*).

From our experience, one generic area where GP has demonstrated a clear competitive advantage is the development of simple empirical models. The specific approach within GP is symbolic regression (Koza, 1992). We have shown in several cases that the models generated by symbolic regression are a low-cost alternative to both high fidelity models (Kordon *et al*, 2003a) and expensive hardware analyzers (Kordon *et al*, 2003b). The specific competitive advantages of symbolic regression generated by GP and related to the generic area of empirical modeling are defined as follows:

- **No *a priori* modeling assumptions** – GP model development does not require assumption space limited by physical considerations (as is in case of first-principle modeling) or by statistical considerations, such as variable independence, multivariate normal distribution and independent errors with zero mean and constant variance.
- **Empirical models with improved robustness** – Using Pareto front GP (Smits and Kotanchek, 2004) allows the simulated evolution and model selection to be directed toward solutions based on an optimal balance between accuracy and expression complexity. The derived symbolic regression models have improved robustness during process changes relative to both conventional GP and neural-network-based models.
- **Easy integration into existing work processes** – Since the derived final solutions, generated by GP are symbolic expressions there is no need for specialized software environment for their run-time implementation. This feature allows for a relatively easy integration of the GP technology into most of the existing model development and deployment work processes.
- **Minimal training of the final user** – The symbolic regression nature of the final solutions generated by GP is universally acceptable by any user with mathematical background at the high school level. This is not the case either with the first-principle models (where specific physical knowledge is required) or with the black-box models (where some training on neural networks is a must). In addition, a very important factor in favor of GP is that process engineers prefer mathematical expressions and very often can find an appropriate physical interpretation. They usually don't hide their distaste toward black boxes.
- **Low total cost of development, deployment, and maintenance** – Contrary to the common opinion, the key disadvantage of GP – the computationally intensive and time-consuming model generation-- does not add significantly to the development cost because it does not occupy the model developer's time. What is required from the model developer is to set the parameters at the beginning of the

simulation and to assess the selected models at the end. With the Pareto front GP method, the derived models have minimal total cost. They are derived and automatically selected at the optimum performance-- complexity Pareto front and as such, have better robustness (*i.e.*, reduced need for model re-tuning during process changes and maintenance cost), are parsimonious (even with potential interpretation by the experts), and with minimal implementation requirements and cost. The alternative approaches require specialized software, expertise on the specific technology, training on the approach and the related software, and significant model validation and support expenses.

The major disadvantages of GP relative to other techniques are (1) the absence of commercial software infrastructure, (2) the computational effort typically required for the model building, and (3) typically lower absolute model accuracy relative to techniques such as neural networks.

Recommendations for Industrial Problems Appropriate for Genetic Programming

With this impressive list of competitive advantages over first-principle, statistical and neural network frameworks for modeling, GP has very broad application potential in industry. Since the mid-90s we've explored the capabilities of GP, developed our internal software toolboxes on MATLAB and *Mathematica*, and gradually introduced the technology to the businesses. Critical for the sustainability of the support of this R&D effort was the continuous series of successful applications that demonstrated the value from our GP development agenda.

Our experience in applying GP to real industrial problems in the chemical industry suggests these suitable targets::

- **Fast development of nonlinear empirical models** – Symbolic-regression problems are very suitable for industrial applications, and are often optimal in terms of both development and maintenance costs. One area with tremendous potential is inferential or soft sensors, *i.e.* empirical models that infer difficult-to-measure process parameters, such as NOx emissions, melt index, interface level, *etc.*, from easy-to-measure process variables such as temperatures, pressures, flows, *etc.* (Kordon *et al*, 2003b). The current solutions in the market, which are based on neural networks, require frequent re-training and specialized run-time software.

  An example of an inferential sensor for propylene prediction based on an ensemble of four different models derived by Genetic

Programming is given in (Jordaan *et al*, 2004). The models were developed from an initial large manufacturing data set of 23 potential input variables and 6900 data points. The size of the data set was reduced by variable selection to 7 significant inputs and the models were generated by five independent GP runs. As a result of the model selection, a list of 12 models on the Pareto front was proposed for further evaluation to process engineers. All selected models have high performance (R^2 of 0.97 – 0.98) and low complexity. After evaluating their extrapolation capabilities with "What-If" scenarios, the diversity of model inputs, and physical considerations, an ensemble of four models was selected for on-line implementation. Two of the models are shown below:

$$GP\_Model1 = A + B\left|\frac{Tray64\_T^4 * Vapor^3}{Rflx\_flow^2}\right|$$

$$GP\_Model2 = C + D\left|\frac{Feed^3\sqrt{Tray46\_T - Tray56\_T}}{Vapor^2 * Rflx\_flow^4}\right|$$

where A, B, C, and D are fitting parameters, and all model inputs in the equations are continuous process measurements.

These models are simple and interpretable by process engineers. The difference in model inputs increases the robustness of the estimation scheme in case of possible input sensor failure. The inferential sensor is in operation since May 2004.

- **Emulation of complex first-principle models** – Symbolic regression models can substitute parts of fundamental models for on-line monitoring and optimization. The execution speed of most complex first-principle models is too slow for real-time operation. One effective solution is to replace a portion of the fundamental model with a simpler symbolic regression called an emulator, which is based only on a subset of variables. The data for the emulator are generated by design of experiments from the first-principle model. Usually the fundamental model is represented with several simple emulators, which are implemented on-line. One interesting benefit of emulators is that they can be used to validate fundamental models as well. The validation of a complex model in conditions where the process is chanting continuously requires tremendous efforts in data collection and numerous model parameter fittings. It is much easier

to validate the simple emulators and to infer the state of the complex model on the basis of the high correlation between them. An example of such an application for optimal handling of by-products is given in (Kordon *et al*, 2003a). The mechanistic model is very complex, and includes over 1500 chemical reactions with more than 200 species. Ten input variables and 12 output variables were suggested by domain experts. A data set based on a four levels design of experiments was generated and used for model development and validation. For 7 of the outputs a linear emulator gave acceptable performance. For the remaining 5 emulators, a nonlinear model was derived by GP. An example of a nonlinear emulator selected by the experts is given below:

$$Y_5 = \frac{6x_3 + x_4 + x_5 + 2x_6 + x_2x_9 - \dfrac{x_2 - 3x_3\sqrt{x_6}}{(x_2^2 + x_7x_1^3)}}{\ln(\sqrt{x_9x_{10}^2})}$$

where Y is the predicted output (used for process optimization), and the x variables are measured process parameters. The emulators have been used for by-product optimization between two chemical plants in The Dow Chemical Company since March 2003.

- **Accelerated first-principle model building** – Beginning first-principle modeling not from scratch but from symbolic regression models and building blocks (transforms) can significantly reduce the hypothesis search space for potential physical/chemical mechanisms. New product development effort can be considerably reduced by eliminating unimportant variables, enabling rapid testing of new physical mechanisms and reducing the number of experiments for model validation. The large potential of this type of application was demonstrated in a case study for structure-property relationships (Kordon *et al*, 2002). The GP-augmented solution was similar to the fundamental model and was delivered with significantly less human effort (10 hours vs. 3 months).
- **Linearized transforms for Design Of Experiments** – GP-generated transforms of the input variables can eliminate significant lack of fit in linear regression models without the need to add expensive experiments to the original design, which can be time-consuming, costly, or maybe technically infeasible because of extreme experimental conditions. An example of such type of application for a chemical process is given in (Castillo *et al*, 2002).

A selected set of GP applications from the above-mentioned industrial problems is given in Table 16-1. For each application the following

information is given: initial size of the data set (including all potential inputs and data points), reduced size of the data set (after variable selection and data condensation), model structure (number of inputs used in the selected final models and the number of models; some of them are used in an ensemble), and a corresponding reference which contains a detailed description of the application, including the GP parameters used. In all the cases the final solutions obtained with the help of GP were parsimonious models with a significantly reduced number of inputs.

Table 16-1. Selected GP applications in Dow chemical

Application	Initial data size	Reduced data size	Model structure	Reference
Inferential sensors				
Interface level prediction	(25 inputs x 6500 data pts)	(2 inputs x 2000 data pts)	3 models 2 inputs	Kordon and Smits, 2001
Interface level prediction	(28 inputs x 2850 data pts)	(5 inputs x 2850 data pts)	One model 3 inputs	Kalos *et al*, 2003
Emissions prediction	(8 inputs x 251 data pts)	(4 inputs x 34 data pts)	Two models 4 inputs	Kordon *et al*, 2003b
Biomass prediction	(10 inputs x 705 data pts)	(10 inputs x 705 data pts)	9 models ens 2-3 inputs	Jordaan *et al* , 2004
Propylene prediction	(23 inputs x 6900 data pts)	(7 inputs x 6900 data pts)	4 models ens 2-3 inputs	Jordaan *et al* , 2004
Emulators				
Chemical reactor	(10 inputs x 320 data pts)	(10 inputs x 320 data pts)	5 models 8 inputs	Kordon *et al*, 2003a
Accelerated modeling				
Structure-property	(5 inputs x 32 data pts)	(5 inputs x 32 data pts)	One model 4 inputs	Kordon *et al*, 2002
Structure-property	(9 inputs x 24 data pts)	(9 inputs x 24 data pts)	7 models 3 -5 inputs	Kordon and Lue, 2004
Linearized transforms				
Chemical reactor model	(4 inputs x 19 data pts)	(4 inputs x 19 data pts)	3 transforms	Castillo *et al*, 2002

3. How to Select the Genetic Programming Parameters

Another important issue in industrial applications of GP is the GP algorithm parameter selection. As a first step, the parameters can be selected according to the rule-of-thumb recommendations of Koza (Koza, 1992). However, a more systematic statistical approach is recommended since the numerous parameters and settings used by GP introduce uncertainty about the way they affect the search algorithm and therefore the solution found. This has significant theoretical implications. Among them is the amount of information the parameters provide and the possible restrictions in the set of right solutions. It is therefore important to understand the effect of the parameters, the effect of the various combinations of them, and how robust they are to different data sets. This is of special importance given that the GP algorithm is used with a variety of data sets with different degrees of complexity.

The optimum set of GP parameters can be determined through statistical experimental design techniques, such as design of experiments (DOE). This section explains how to use an appropriate DOE and the appropriate set of replications to understand the effect of GP parameters.

Statistical Experimental Design: Design of Experiments

Design of Experiments is a statistical approach that provides enhanced knowledge of a system by quantifying the effect of a set of inputs (factors) on an output (response). This is accomplished by systematically running experiments at different combinations of the factor settings (Box *et al*, 1978).

A classical DOE is the 2^k design, in which all factors are investigated at an upper and lower level of a range, resulting in 2^k experiments where k is the number of factors. This design has the advantage that the effects of the individual factors (main effects), as well as all possible interactions (combination of factors), can be estimated. However, the number of experimental runs increases rapidly as the number of factors increases. If the number of experiments is impractical, fractional factorial design can be used. In this case, only a fraction of the full 2^k design is run by assuming that some interactions among factors are not significant. However, this assumption can sometimes confound the main effects and interactions, so they therefore cannot be estimated separately.

Depending on the type of fractional factorial, main effects may be confounded with second-, third-, or fourth-order interactions. The level of confounding is dictated by the design resolution. The higher the design

resolution, the less confounding occurs among factors. For example, a resolution III design confounds main effects with second-order interactions; a resolution IV design confounds second-order interaction with other second-order interactions; and a resolution V design confounds second-order interactions with third-order interactions. Felt and Nordin (2000) investigated the effect of 17 GP parameters on three binary classification problems using highly fractionated designs assuming, in some cases, that even second- and third-order interaction are not significant, *i.e.,* the combined effect of two factors and three factors has no effect on the response. However, these assumptions have not been verified.

Given that the study of GP parameters involves computing experiments as opposed to pilot plant or laboratory experiments, it is desirable to run a full factorial when possible, so that any second and third order interaction which may have statistically significant effects on the response can be quantified.

Pareto Front Genetic Programming DOE

The GP experimental design we would like to describe differs from that of Felt and Nording in three aspects. First, it allows the estimation of interactions. Second, it uses the convergence to the Pareto front as the response variable. Third, the robustness of GP parameters to the different data sets is investigated with industrial data sets with different degrees of complexity based on dimension of input matrix and degree of input correlation.

The need for a more systematic DOE approach is also driven by the significant benefits of the Pareto front-based GP, demonstrated in several industrial applications (Smits and Kotanchek, 2004). In this approach, the optimal models fall on the curve of the non-dominated solutions, called Pareto front, *i.e.*, no other solution is better than the solutions on the Pareto front in both complexity and performance. As discussed above in Section 2.2, parsimonious models with high performance are the greatest importance in industry. These occupy the lower left corner of the Pareto front indicated in the diagram in Figure 16-2. In that context, the goal is to select GP parameters that consistently drive simulated evolution toward the lower left of this diagram. The Pareto front GP parameters (factors) and their ranges are presented in the following table:

Table 16-2. Factors for the Pareto Front GP Doe

Factor	Low level (-1)	High Level (+1)
x1 - Number of cascades	10	50
x2 - Number of generations	10	50
x3 - Population size	100	500
x4 - Probability of function selection	0.4	0.7
x5 - Size of archive	100	500

The response variable proposed is the convergence to the Pareto front (Smits and Kotanchek (2004) which includes the prediction error ($1\text{-}R^2$) as the performance measure and the sum of the number of nodes of all sub-equations as the value of complexity. The factor x1, number of cascades, is the number of independent runs with a freshly generated starting population. The ranges of the factors have been selected based on the experience from various types of practical problems, related to symbolic regression. Since the objective is a consistent Pareto front GP, they differ from the recommendations for the original GP.

Once the factors and ranges are selected the necessary number of replications must be determined. This is of key importance because in the case of GP parameters we do not know for sure if the variability of the response is the same for the different combination of factors. The following figure illustrates this situation for three factors.

To estimate the number of required replications, an initial set of n replications can be run, from which the standard deviation of the response is calculated. In our case, the response is the convergence to the Pareto Front. In this case a fixed level of complexity for the number of nodes is selected.

For this level the corresponding number of models is observed and the standard deviation of the response between these models can be estimated. Figure 16-2 illustrates the concept.

Once the standard deviation is calculated the number of replications can be found applying the half width (HW) confidence interval method (Montgomery, 1999)[1]. The half width can be use to represent the percent error in the point estimate of the mean response. The half width (HW) is defined as:

$$HW = t_{n-1,\,a/2}\,\frac{S}{\sqrt{n}}$$

[1] 100(1-*a*)% *confidence interval* is a range of values in which the true answer is believed to lie with 1- *a* probability. Usually *a* is set at 0.05 so that 95% confidence interval is calculated. Half width, sometimes called accuracy of the confidence interval, is the distance between the estimated mean and the upper or lower range of the confidence interval.

Where $t_{n-1,a/2}$ is the upper $a/2$ percentage point of the t distribution with $n-1$ degrees of freedom, S is the standard deviation and n is the number of runs.

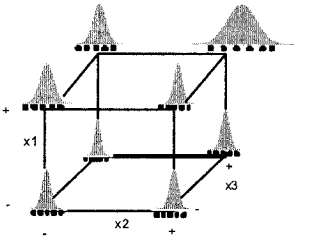

Figure 16-1. Combination of factors in a 2^3 design showing different variances for the different factor combinations.

A plot of the 100(1-a)% HW confidence interval reveals the number of replications above which little improvement in HW is obtained. This is illustrated in Fig. 3.3 with an example with 95% confidence interval in which S=0.08. The graph shows that beyond 10 replications there is little to be gained in terms of half width.

The same procedure can be applied for the different combinations of factors, and the desirable half width can be fixed so that the experimental design can be completed with the required number of replications for the required accuracy. If we knew for certain that the variability of the response is about the same for the different combination of factors (experimental runs), we could find the confidence interval of the *difference in mean response* for any two combinations of factors, and find the number of replications required[2] which in this case will be the same for all combinations of factors. (see, for example, Montgomery, 1999).

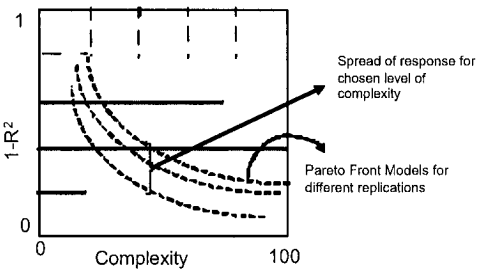

Figure 16-2. Spread of response for a chosen level of complexity.

[2] In this case the HW confidence interval is $t_{an-a,a/2}\ \{2S^2/n\}^{1/2}$. Where a is the number of combination of factors (experimental runs), S is the standard deviation and n is the number of replications

Robustness of Pareto Front GP Parameters to Different Data Sets.

To address the issue of the robustness of GP parameters to the data set, the experimental design previously described needs to be executed for different industrial data sets with various degrees of complexity—for example, low, medium, and high. The complete set of experiments follows an orthogonal array design which is depicted in Figure 16-4 where y_{ij} is the response associated with the *ith* data set and the *jth* combination of GP parameters. If there are n_1 combinations of GP parameters and n_2 data sets, then we need $n_1 * n_2$ runs for the total experimental design and each run of the design will have the required number of replications as indicated by the desired half width. For simplicity, Figure 16-2 only shows one replication per experimental run. The n'_1 n_2 experimental design is an orthogonal design composed of an inner array (GP parameter combinations) and an outer array (the data sets). This type of design allows quantifying the interactions (combined effect of two and three factors). It also reveals information on the combinations of GP parameters that result in a reasonable response even when different data sets are used (combinations of GP parameters that produce correct responses with minimum variation between data sets). Of particular importance in this case are the interactions between the GP parameters and the data sets since these interactions determine the sensitivity of the GP parameters to the type of data set. This is illustrated in the following diagram, Figure 16-5.

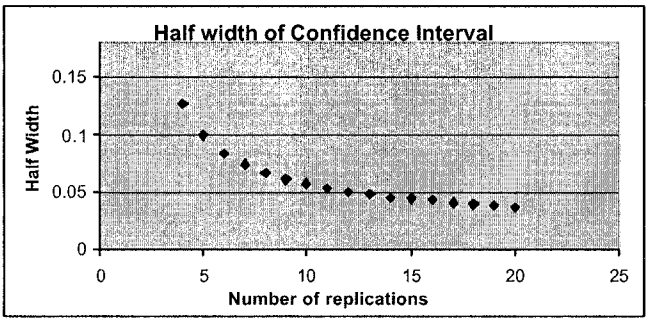

Figure 16-3. 95% Half width confidence interval versus number of replications.

In this case the diagram of the interaction shows a response that is not sensitive to the type of data set if the upper level (+) of parameter x_i is used. Determination of these types of interactions is fundamental to understand the robustness of Pareto front GP parameter combinations.

A proper statistical analysis of the orthogonal design can be valuable; it can provide information on how the response is affected by the Pareto front GP parameter, and how the choice of data can modify that effect. This can be used to determine the best set of parameters for different applications of GP symbolic regression in the chemical industry (and elsewhere).

GP Parameter Variables					Different Types of Data Sets			
					Data 1	Data 2	Data 3	.....
X1	X2	X3	X4	X5				
-1	-1	-1	-1	-1	*y11*	*y21*	*y31*	
-1	-1	-1	1	-1	*y12*	*y22*	*y32*	
1	1	1	-1	-1	*y13*	*y23*	*y33*	
-1	1	-1	1	-1	*y14*	*y24*	*y34*	
-1	1	-1	1	-1	*y15*	*y25*	*y35*	
1	1	1	1	1	*y16*	*y26*	*y36*	
1	-1	1	1	1	*y17*	*y27*	*y37*	
1	-1	1	1	-1	*y18*	*y28*	*y38*	
1	-1	1	1	1	*y19*	*y29*	*y39*	
1	-1	-1	1	-1	*y110*	*y210*	*y310*	
1	1	1	1	-1	*y111*	*y211*	*y311*	
-1	1	-1	-1	-1	*y112*	*y212*	*y312*	
1	1	-1	-1	1	*y113*	*y213*	*y313*	
1	-1	-1	-1	1	*y114*	*y214*	*y314*	
1	-1	-1	-1	1	*y115*	*y215*	*y315*	
1	1	1	-1	-1	*y116*	*y216*	*y316*	
1	-1	1	-1	-1	*y117*	*y217*	*y317*	
-1	1	1	-1	1	*y118*	*y218*	*y318*	
-1	-1	1	-1	1	*y119*	*y219*	*y319*	
1	1	-1	1	1	*y120*	*y220*	*y320*	
-1	-1	1	-1	-1	*y121*	*y221*	*y321*	
-1	-1	-1	1	1	*y122*	*y222*	*y322*	
-1	1	1	-1	-1	*y123*	*y223*	*y323*	
1	1	-1	-1	-1	*y124*	*y224*	*y324*	
-1	1	1	1	1	*y125*	*y225*	*y325*	
-1	-1	1	1	-1	*y126*	*y226*	*y326*	
-1	1	1	1	1	*y127*	*y227*	*y327*	
-1	1	-1	1	1	*y128*	*y228*	*y328*	
1	-1	-1	1	-1	*y129*	*y229*	*y329*	
-1	-1	1	-1	1	*y130*	*y230*	*y330*	
1	1	-1	-1	1	*y131*	*y231*	*y331*	
-1	-1	-1	-1	1	*y132*	*y232*	*y332*	

Figure 16-4. Orthogonal design with 32 runs in three data sets

4. Issues with Genetic Programming Applications

Applying a new technology, such as GP, in industry requires resolving not only many technical issues, but also systematically and patiently

handling problems of a non-technical nature. A short overview of the key technical and non-technical issues is given below.

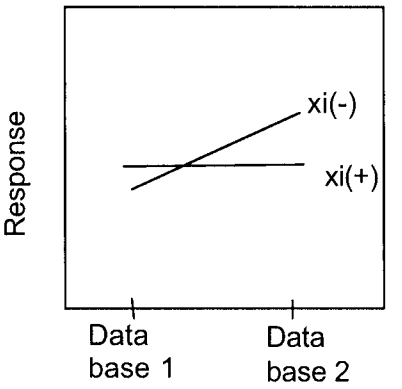

Figure 16-5. Diagram of the interaction of the *ith* GP parameter with the data set type.

Technical Shortcomings

- **Available computer infrastructure** – Even with the help of Moore's Law, GP model development requires significant computational efforts. It is recommended to allocate a proper infrastructure, such as a computer cluster, to accelerate this process. The growing capability of grid computing to handle computationally intensive tasks is another option to improve the GP performance, especially in a big global corporation with thousand of computers. However, development of parallel GP algorithms in user-friendly software is needed.
- **Professional GP software** – The current software options for GP implementation, either external or internally developed, are still used for algorithm development and research purposes. One of the obstacles to mass scale applications of GP is the lack of professional-seeming and user friendly software packages, from well-established vendors, that would also handle continuous product development and product support. Without such a product, the implementation effort is very high and it will be very difficult to convince people to use for GP industrial applications purposes.
- **Symbolic regression is still not accepted as a modeling standard** – One of the difficulties in developing professional GP software is that symbolic regression via GP is still not

included in the recently developed Predictive Model Markup Language (PMML, 2004). Most of the other modeling methods— linear regression, neural networks, rule-based models, support vector machines, *etc.*, are techniques supported by this standard and included in the professional software of well-known empirical modeling vendors like the SAS Institute, SPSS, and StatSoft. The best-case scenario for more widespread industrial applications of symbolic regression with GP is to bundle the technology in the existing popular statistical and data mining tools, such as JMP, STATISTICA, Enterprise Miner, or some other package. If that were done, GP would be introduced to the modeling and statistical communities in a natural way and could be used in combination with the other well-known methods.

- **Special attention to data preparation** – Another requirement of using symbolic regression in an integrated statistical software environment is the need for careful data preparation, including outlier removal, data pre-processing, scaling, normalization, *etc.*, before beginning the simulated evolution. Existing GP software tools do not have built-in capabilities for data preparation. The hidden assumption is that the available data is of high quality, which for industrial data sets is often not the case.
- **Technical limitations of GP** – In spite of the fast theoretical development since the early 90's, and increasing computational speed, GP still has several well-known limitations. Generating solutions in a high-dimensional search space takes significant time. Model selection is not trivial and is still more of an art than a science. Integrating heuristics and prior knowledge is not yet a straightforward process for practical applications. Generating complex dynamic systems by GP is still in its infancy.

Non-technical Issues

- **Critical mass of developers** – It is very important at this early phase of industrial applications of GP to coordinate development efforts. The probability for success based only on individual attempts is very low. The best-case scenario would be the creation of a virtual group that includes not only specialists directly involved in GP development and implementation, but also specialists with similar areas of expertise like machine learning, expert systems, and statistics.
- **GP marketing to business and research communities** – Since GP is virtually unknown not only to business-related users but

also to other research communities as well, it is necessary to promote the approach by significant marketing efforts. Usually an approach to marketing research-grade includes a series of promotion meetings based on two different presentations. One of these presentations is directed toward the research communities focuses on the "technology kitchen," which gives enough technical details to describe GP, demonstrates the differences from other known methods, and clearly illustrates the competitive advantages of GP. The second presentation, for the business-related audience focuses on the "technology dishes," *i.e.*, it demonstrates with specific industrial examples the types of applications that are appropriate for GP, describes the work process to develop, deploy, and support a GP application, and illustrates the potential financial benefits of applying GP.

- **Management support** – Consistent management support for at least several years is critical for introducing any emerging technology, including GP. The best way to win this support is to define the expected research efforts and assess the potential benefits from specific application areas. Of decisive importance, however, is the demonstration of value creation by resolving practical problems as soon as possible.
- **Lack of initial credibility** – As a new and virtually unknown approach, GP has almost no application history for convincing a potential user. Any GP application requires a risk-seeking culture and significant communication efforts. The successful application discussed in this chapter are a good start to gain credibility and increase the potential GP customer base.

5. Summary

Among the emerging technologies in the area of computational intelligence, GP has clear competitive advantages and potential for solving a broad range of industrial problems. Several application areas in the chemical industry—for example, inferential sensors, emulators of complex first-principle models, accelerated development of fundamental models, and generation of linearized transforms for design-of-experiments-model-building—already have demonstrated the power of GP and created value. However, a number of technical and non-technical issues, such as well-defined data preparation, development of well-supported professional software packages, GP marketing to business and research communities,

consistent management support, *etc.*, have to be resolved before we can expect mass-scale applications of GP in industry.

References

Box, G., Hunter, W., and Hunter, J. (1978). *Statistics for Experiments: An Introduction to Design, Data Analysis, and Model Building*, New York, NY: Wiley.

Castillo, F., Marshall, K, Greens, J. and Kordon, A. (2002). Symbolic Regression in Design of Experiments: A Case Study with Linearizing Transformations, In *Proceedings of the Genetic and Evolutionary Computing Conference (GECCO'2002)*, W. Langdon, *et al* (Eds), pp. 1043-1048. New York, NY: Morgan Kaufmann.

Feldt R. and Nordin P. (2000). Using Factorial Experiments to Evaluate the Effects of Genetic Programming parameters. In *Proceedings of EuroGP'2000*, pp. 271-282, Edinburgh, UK

Kalos A., Kordon, A, Smits, G., and Werkmeister, S. (2003) Hybrid Model Development Methodology for Industrial Soft Sensors, *In Proceedings of the American Control Conference (ACC'2003)*, pp. 5417-5422, Denver. CO.

Kordon A. and Smits, G. (2001) Soft Sensor Development Using Genetic Programming, In *Proceedings of the Genetic and Evolutionary Computing Conference (GECCO'2001)*, L. Spector, *et al* (Eds), pp. 1346 – 1351, San Francisco, Morgan Kaufmann.

Kordon A., H. Pham, C. Bosnyak, M. Kotanchek, and G. Smits, (2002). Accelerating Industrial Fundamental Model Building with Symbolic Regression: A Case Study with Structure – Property Relationships, In *Proceedings of the Genetic and Evolutionary Computing Conference (GECCO'2002)*, D. Davis and R. Roy (Eds), Volume Evolutionary Computation in Industry, pp. 111-116. New York, NY: Morgan Kaufmann.

Kordon A., Kalos, A. and Adams, B. (2003a), Empirical Emulators for Process Monitoring and Optimization, *In Proceedings of the IEEE 11th Conference on Control and Automation MED'2003*, pp.111, Rhodes, Greece.

Kordon, A., Smits, G., Kalos, A., and Jordaan, E.(2003b). Robust Soft Sensor Development Using Genetic Programming, In *Nature-Inspired Methods in Chemometrics*, (R. Leardi-Editor), Amsterdam: Elsevier

Kordon A. and Lue, C. (2004) Symbolic Regression Modeling of Blown Film Process Effects, In *Proceedings of the Congress of Evolutionary Computation CEC'2004*, pp. 561-568, Portland, OR.

Kotanchek, M, Smits, G. and Kordon, A. (2003). Industrial Strength Genetic Programming, In *Genetic Programming Theory and Practice, pp 239-258,* R. Riolo and B. Worzel (Eds), Boston, MA:Kluwer.

Koza, J. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, Cambridge, MA: MIT Press.

Jordaan, E., Kordon, A., Smits, G., and Chiang, L. (2004), Robust Inferential Sensors based on Ensemble of predictors generated by Genetic Programming, *In Proceedings of PPSN 2004*, pp. 522-531, Birmingham, UK.

Montgomery, D. (1999) *Design and Analysis of Experiments*, New York, NY: Wiley.

Predictive Modeling Markup Language (PMML V 3.0) Specification, (2004) Data Mining Group, http://www.dmg.org/pmml-v3-0.

Smits, G. and Kotanchek, M. (2004), Pareto -Front Exploitation in Symbolic Regression, *Genetic Programming Theory and Practice, pp 283-300,* U.M. O'Reilly, T. Yu, R. Riolo and B. Worzel (Eds), Boston, MA:Springer.