CIRP International Workshop on
Design Theory and Methodology
June 16-17,1994
Stockholm, Sweden

The FABERCOAT System:
a design tool for plasma-sprayed coatings

Mark Foy, Maurizio Marchese,
Luca Manini and Gianni Jacucci

Università degli studi di Trento
Dipartimento di Informatica e Studi Aziendali
Laboratorio di Ingegneria Informatica
Via Zeni 8, I-38068 Rovereto (Tn), Italy

1. Introduction
2. Brief description of project results
3. FABERCOAT System concept
4. System functionality
5. Deposition model input parameters
6. Selection of objective function
7. Expert system component
8. Computational aspects
9. Conclusions

1. Introduction

In the following we present and discuss the concepts of a prototype knowledge based system named FABERCOAT. The name is taken directly from the name of a currently running European research project on the modelling of the manufacturing process of plasma-sprayed coatings.

The Brite/Euram research project FABERCOAT is aimed to improve the understanding of the physical processes involved in the manufacture of TBCs in order to establish a more reliable correlation between process parameters and coating performance. To this end a combined experimental and theoretical programme had been set out and it comprises:
- the manufacturing of a set of coatings under controlled specifications;
- their analysis and characterization;
- the development of physical models and simulation techniques to model the process as a whole and to correlate the process parameters with the coating characteristics.

The first phase of the project has been focused on sample production and characterization and the set-up of specific tools for modelling and image analysis. The second phase is centred on the transfer of the acquired experimental data to the developed physical model of the deposition model and the mutual integration of all data produced into a knowledge based system.

2. Brief description of the FABERCOAT project's main results

Before describing the FABERCOAT System concept, we recall here the state of the FABERCOAT project summoning some of the main results of the investigation of the concerted research action that are relevant in the definition of the FABERCOAT System.

In regard to the experimental programme, the following main topics and results have been collected so far:

i) experimental data on the plasma spray manufacturing processes at the three different length scales of the process: i) data on the single droplet of ceramic particle (in flight and after impact on the substrate), ii) data on the mechanism for the assembly of droplets, iii) information on the macroscopic properties of the final coatings

ii) development and use of new techniques to measure plasma-particle interactions to provide the necessary data to the model of the deposition process.

iii) development of image analysis tools to characterize coating microstructures (porosity, crack network fractal dimension)

iv) X-ray analysis to measure phase composition and residual stresses

The work of the modelling programme has been focused mainly on:

1) the development of a deposition model (2-d) for the plasma spray manufacture of ceramic coatings;

2) the development of methods and the investigation of models for a description of the mechanical and the thermal material behaviour of ceramic layers;
3) the development of a residual stress model to estimate strain and stresses developed in the coating during fabrication.

From the initial stages of the project, the deposition model has been identified as the central component in order to fulfil the project goals, namely to correlate process parameters and final coating characteristics.

Briefly, the two dimensional deposition model consists of two main part: (i) a numerical model of the impact process of droplets of material on flat and rough substrates and (ii) a knowledge-based program that tests a set of conditions which take into account the various characteristics of the droplet and of the target impact area on the film deposited so far. Based on these conditions, one of multiple simulation scenarios (which correspond to different physical situations) is activated. Thus, one simulation starts with the generation of a particle of given radius, velocity and temperature that travel to the target area; the knowledge-based program chooses which of the possible simulation scenarios to run as each droplet makes contact with the surface of the film. The process continues until the completed film, of the required thickness, is built up.

The main capabilities of the deposition model consist of predicting a number of final coating microstructural characteristics (such as total porosity, porosity profile and shape distributions, and temperature distributions) that can be used to determine, in a second stage, final coating mechanical and thermal properties (among others, elasticity moduli and thermal conductivity). The overall scheme of the deposition model is summarized in Figure 2.1.

In the following we review briefly the input data needed for the deposition model:

(1) correlation between torch settings and plasma and particle properties, i.e.:

- data on plasma gas (mainly temperature) as a function torch settings and gun distance
- data on in-flight particle characteristics such as particle radius, velocity and temperature distributions as a function of torch design, settings and distance.

Such detailed data are complex to measure, but have been obtained in a complete form for at least one specific torch in the course of this project as well as in other reserach work carried out by University of Limoges. Figure 2.2 and 2.3 show typical examples of the kind of experimental data that are required to describe the initial conditions for the deposition process. It is useful to underline that a lot of effort has been put in the last 10 years in the plasma-spray community to model the plasma gas and the transport of particles in it. By now several models have been developed and validated for some plasma torches, and are commercially available. They could be easily integrated in the FABERCOAT System to provide the needed correlation between torch settings and plasma and particle properties. For this prototype version of the deposition model we have limited our simulation to consider only the acquired experimental data for one torch.

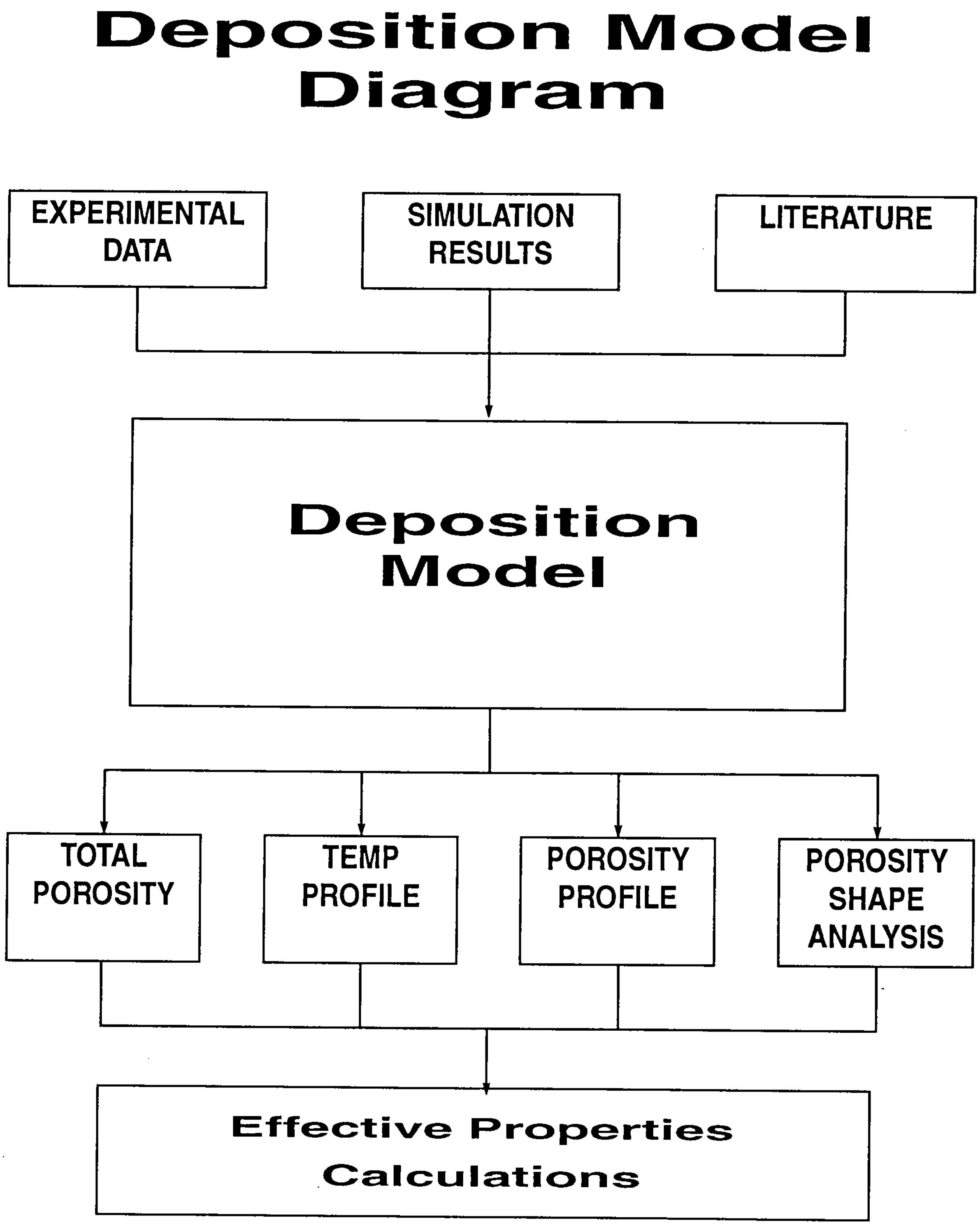

Figure 2.1: Scheme of the deposition model

EXAMPLE OF NEEDED EXPERIMENTAL DATA

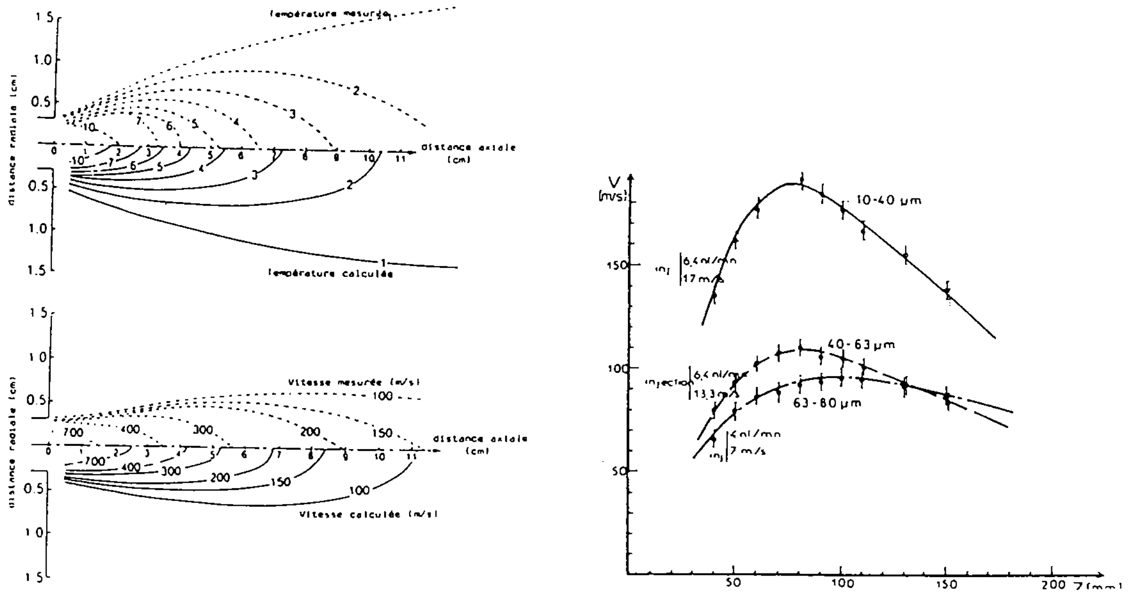

FIG 1 – Profils de vitesse et de température calculés [31] et expérimentaux [12] pour un jet de plasma d'arc Ar-H_2.
FIG 1 – Calculated and experimental contours of plasma gas temperature and velocity

FIG 4 – Evolution axiale de la vitesse des poudres pour différentes granulométries (plasma Ar-H_2 poudre ZrO_2).
FIG 4 – Particulate velocity along the plasma jet axis (plasma gas Ar-H_2 powder ZrO_2).

(from A. Vardelle et al., Rev. Int. Haut. Temp. Ref., 23, 1986, pp. 69–85)

Figure 2.2: (a) Calculated and experimental contours of plasma gas temperature and velocity (plasma gas: Ar-H_2, powder ZrO_2) (b) Particulate velocity along the plasma jet axis (plasma gas: Ar-H_2, powder ZrO_2)

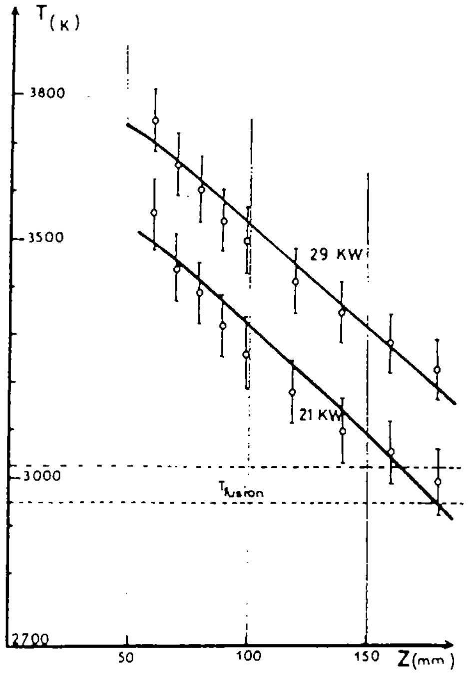

Figure 2.3: Particulate surface temperature along the plasma jet axis (plasma gas: Ar-H_2, powder ZrO_2)

EXAMPLE OF NUMERICAL DATA

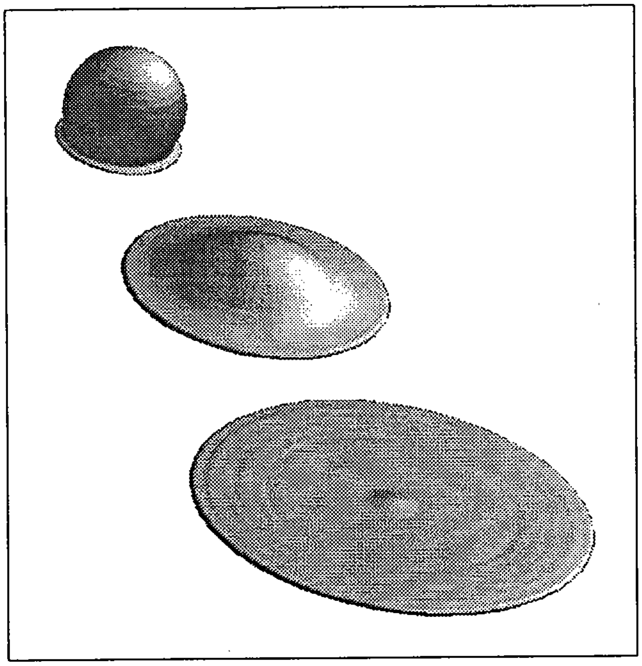

Figure 2.4: Simulated evolution of an impact event

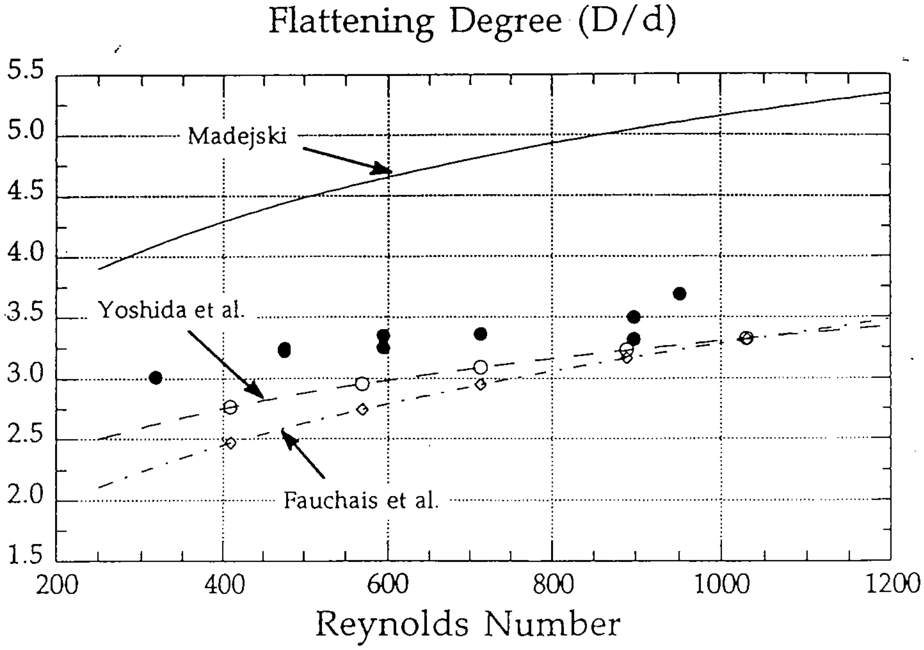

Figure 2.5: Flattening degree as a function of the Reynolds number of the impacting particle: black dots are simulation data and are compared with theoretical [Madejski, 1976], numerical [Yoshida et al, 1991] and experimental data [Fauchais et al, 1989]

(2) Correlation between in-flight and after-impact particle properties, i.e.:

- data on the behaviour of particles after impact as a function of initial conditions

Theoretical and numerical models of the impact process have been developed in order to obtained the information on the particle flattening degree as a function of a particle's initial conditions and material properties. An example of a simulated impact is shown in Figure 2.4, while Figure 2.5 compares the numerical results of the flattening degree with other theoretical [Madejski, 1976], numerical [Yoshida et al, 1991] and experimental data [Fauchais et al, 1989].

Typical output of the deposition model consists of sections of the simulated coatings, such as in Figure 2.6, and graphs of the profile of a number of microstructural characteristics, such as porosity, as a function of process parameters. As an example, Figure 2.7 show the variation of total porosity as a function of substrate temperature (top figure), gun distance and particle radii (bottom figure).

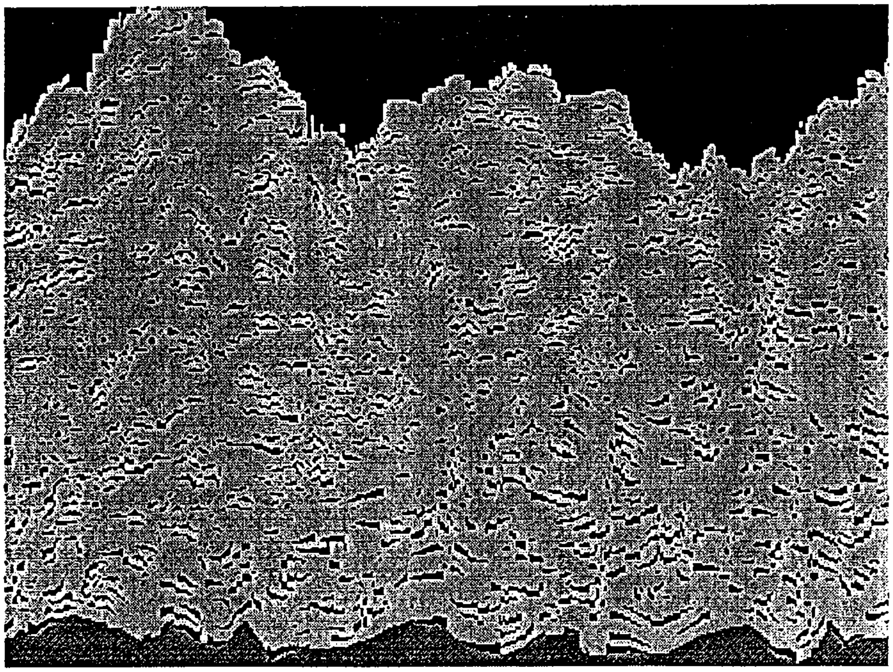

Figure 2.6: Simulated section of coating

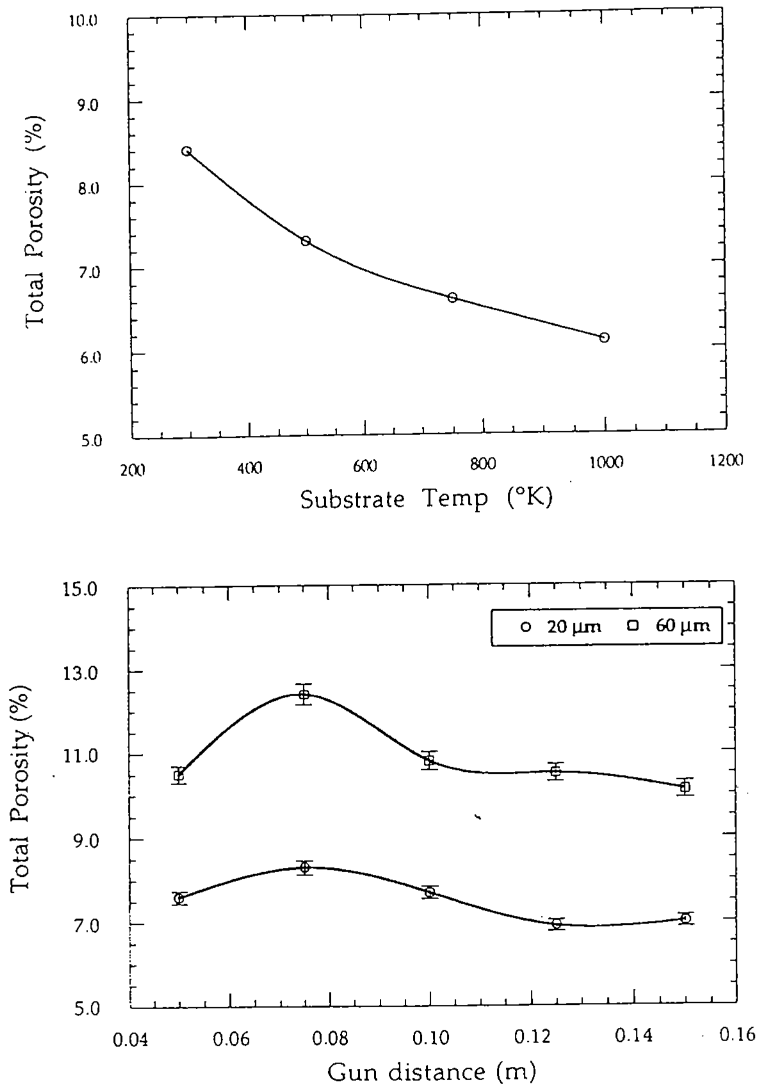

Figure 2.7: Examples of correlation between coating porosity and process parameters

3. FABERCOAT System Concept

The basic idea of the FABERCOAT System is to build a prototype knowledge-base system centred on the deposition model. The knowledge based component is a so called a "front-end system" [O'Keefe, 86]. In fact, in our system, the knowledge-base component acts as a "front-end" for defining a numerical simulation which is subsequently run on its own. The deposition model is been used both (i) to harvest and use the information gained in the extensive production and characterization programme of the project and (ii) as a predictive tool.

Specifically the FABERCOAT System consists of the deposition model together with two new components: (1) a graphic user interface to interface/communicate between operators and the model; (2) an expert system component that coordinates the execution of the model.
The expert system component is used:

- to combine project information and knowledge into one tool

- to allow a user to perform intelligent and automatic searches of the spray parameter space
- to assist final users in performing reliable correlations between initial spray parameters and final coating properties.

The overall scheme of the FABERCOAT System is summarized in Figure 3.1.

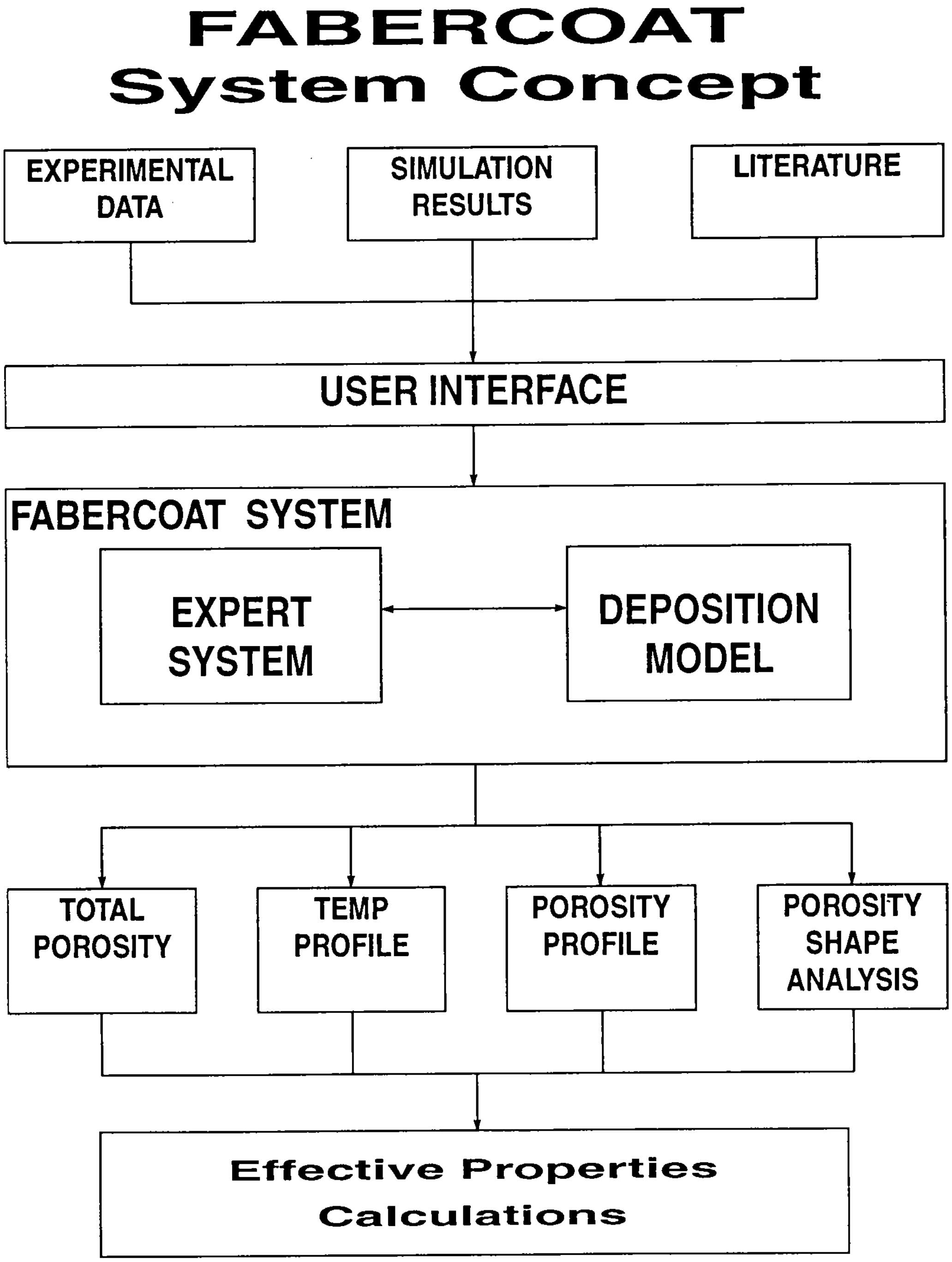

Figure 3.1: FABERCOAT System Concept

4. FABERCOAT System Functionality

The FABERCOAT System should provide multiple levels of functionality so that deposition model simulations can be easily carried out, and results can be examined as to their quality.

The first necessary functionality involves allowing the user to execute multiple simulations automatically. Users often would like to run multiple simulations (possibly varying one spray parameter to see how that effects one or more of the overall characteristics of the final coating), but this task can often be laborious, as the user has to continuously monitor the results, change input files, and then reiterate simulations. The FABERCOAT System should alleviate this difficulty by allowing the user to quickly specify a multiple simulation cycle, and have this cycle automatically performed without user intervention.

Furthermore, beyond this simple iterative simulation functionality, this system will integrate an expert system component, which will intelligently perform more complicated simulation cycles. This expert system component will direct intelligent multiple simulation cycles, specifically, perform intelligent search. Explicitly, the component should execute a search within the space of spray parameters, with the goal to locate a particular set of coating characteristics. For example, a user will be able to specify a certain set of desired outputs (such as a desired total porosity of 15%), and then tell the system to find a set of spray parameters (within certain ranges) which mostly closely matches the set of desired outputs. The system will then respond as an expert would, by giving advice about how to set the spray conditions (gun settings, substrate conditions, etc.) in order to achieve the desired properties in the final coating.

Beyond these iterative search functions, the system should also be able to archive all results from all simulations executed. For example, when performing a search for a set of spray parameters, the system should maintain a "history file" containing a list of all the simulations which have been performed. This is useful for at least three reasons. First, this permits the user to go back and browse the simulation history, so that trends and correlations between model inputs and model outputs (that is, between spray parameters and final coating characteristics) can be discovered. Second, by maintaining a simulation history, if the search were to terminate before finishing, all results would not be lost, and it may be possible to start the search off from where it left off. This is an important issue to some users because the deposition model can take considerable computation time, and therefore, every simulation is valued and they may want to look at the results from all simulations that have been performed. Lastly, a large set of simulations results can be useful to model developers because it can help them detect bugs in the simulation model computer code, so that these can be fixed.

All of these functionalities will be contained within a graphical user interface which will make the execution of these task easier for the user. In particular, the user will be able to specify parameters and preferences within the graphical user interface (using type-in boxes, etc.), and then execute functions with the touch of a button.

Overall, these functionalities should give users useful information by permitting them to structure simulations and searches to assist in the understanding of the deposition model. Therefore users should gain a better understanding of the mechanics of the deposition model, and plasma spray processes, because trends in the model, and correlations between inputs and outputs, will be much easier to observe.

5. The deposition model input parameters.

The deposition program has a long list of input parameters that can be divided in three main groups: "operator" "physical" and "computational".

The "operator" group includes, for example, the gun distance during the spray, the ceramic powder granulometry, and the final thickness of the coating. These parameters are supposed to be directly established by users and users should be able to specify them in standard "engineering" units.

The "physical" group includes, for example, the particle's velocity and temperature at splat time. These values describe the current state of the physical process that is being simulated; they refer to quantities not directly established in real life but that depend on those in the first category by known relations, either analytical, experimental or produced by other simulations.

The "computational" group includes parameters related to the inner workings of the deposition program like grid sizes, random generator seeds and so on. These parameters affect the "precision" and the run time of the program and may depend on those in the "operator" and "physical" classes. For example, the grid size depends on the size of the particles and on the final thickness of the coating. We will not discuss the "tuning" process in this document (for more information see Cirolini et al.,1990).

Originally the second class was the most numerous one because the deposition program was correlating basic physical quantities. The availability of experimental measurements (in-flight velocity and temperature as a function of torch settings) and the results of other simulations (particle splashing and curling for different temperature and velocity combinations) allows for filling the gap between real life parameters and physical parameters thus increasing the size of the "operator" class and reducing the "physical" one.

This change in the input parameter set is very important because the final goal is to make the program usable by operators, not only by physicists or programmers.

6. Selection of the objective function.

The objective function is the physical property of the coating we want to control, i.e. the quantity in the output of the deposition program for which we want to choose a desired value and have the Expert System component search for the unknown "operator" parameters.

The output of the deposition program at present includes: a "picture" of a section of the coating, a temperature profile (through the coating thickness), a porosity profile, and pore aspect ratio distribution.

The "picture" (see Picture 2.6 in the introduction for an example) looks like a microscope image of a section of the coating and allows one to see pores and individual splats. It gives an overall view of the coating, but its information is hard to translate into a quantitative one, and it is better used as a visual check to indicate if something strange (wrong) happened in the simulation.

The temperature profile is available at different times, the last one being especially important for the successive development of residual stresses during cooling. Temperature is usually fairly constant in the substrate, with the exception of a sharp change at interface and quite a big gradient in the coating. Overall, plots should look very similar to the plot shown in Figure 6.1.

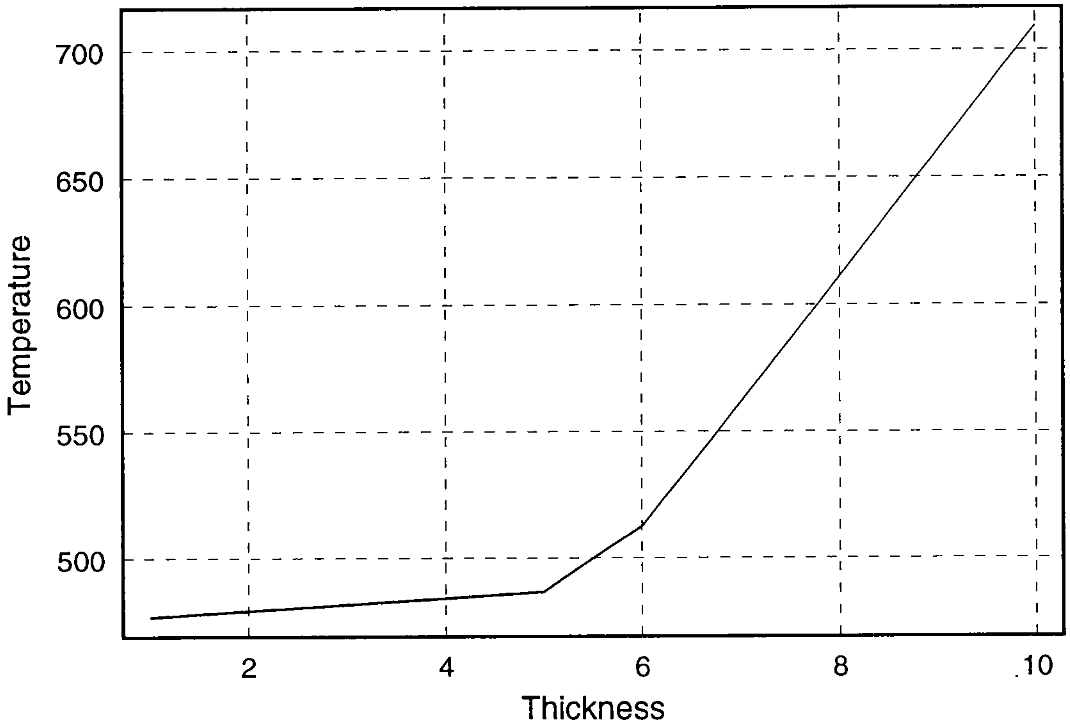

Figure 6.1. – Temperature (° Kelvin) vs. thickness (adimensional units).

The information contained in the temperature profile could be reasonably summarized in one number by taking the gradient in the coating, or measuring the change at the interface.

The porosity profile is probably the most interesting result. Porosity influences the mechanical properties of the coating and can be correlated with measurements done on real coatings. As with the temperature profile there is the problem of compressing its information to just one number.

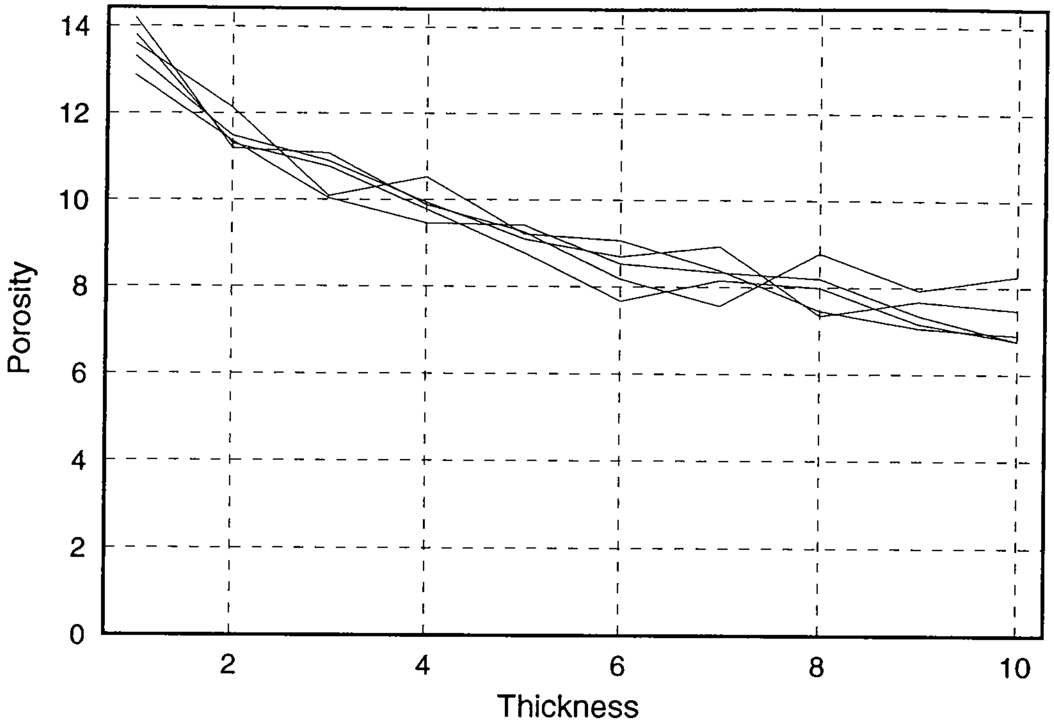

Figure 6.2. Porosity (percentage) vs. thickness (adimensional units) for five different values of the random number generator seed.

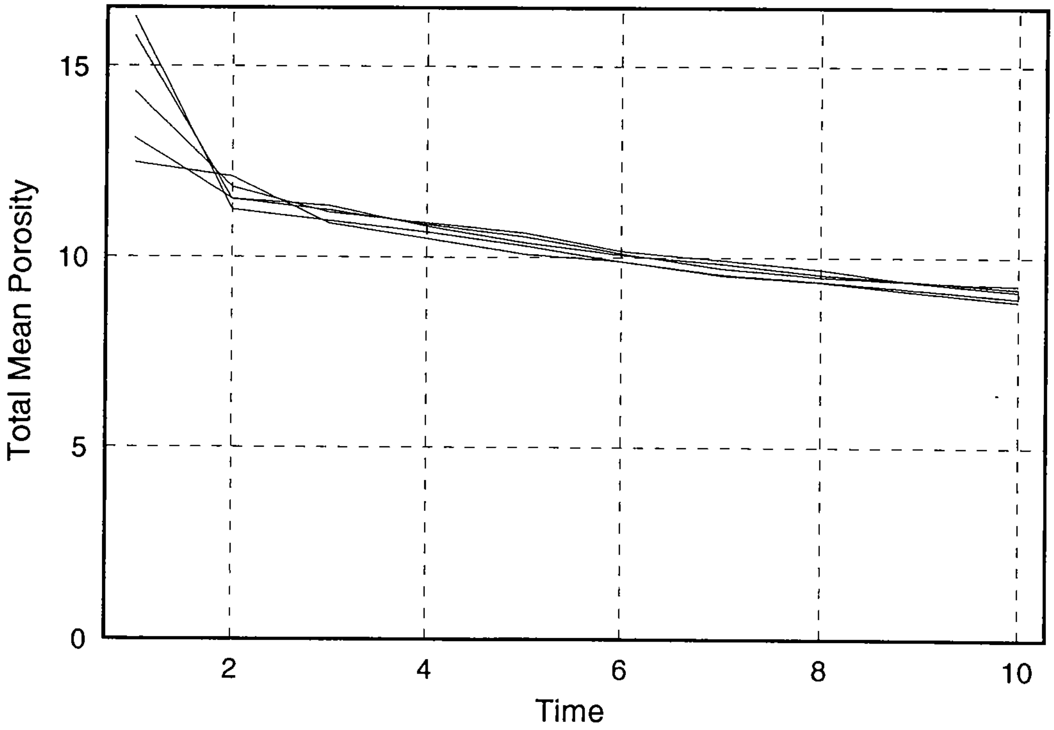

Figure 6.3. Total Mean Porosity (percentage) vs. Time (adimensional units) for five different values of the random number generator seed.

Moreover, local measurements of porosity depend (in real life) on the place in the coating you look at, while in the simulation it has a weak dependence (see Figure 6.2.) on the random number generator seed (that is, it selects a different set of particles, i.e. a different part of the coating). This variability is higher near the substrate and gets smaller moving toward the surface so that it has very little effect on the (final) total mean porosity (see Figure 6.3).

The last output parameter is the distribution of the aspect ratio of the pores. A typical distribution is shown in Figure 6.4. The importance of the aspect ratio of the pores relies in its strong influence on the mechanical and thermal effective properties of the coating. The main feature of Figure 6.4 is the abundance of thin, elongated pores (like cracks) with an aspect ratio less than 0.3 in the simulated coating.

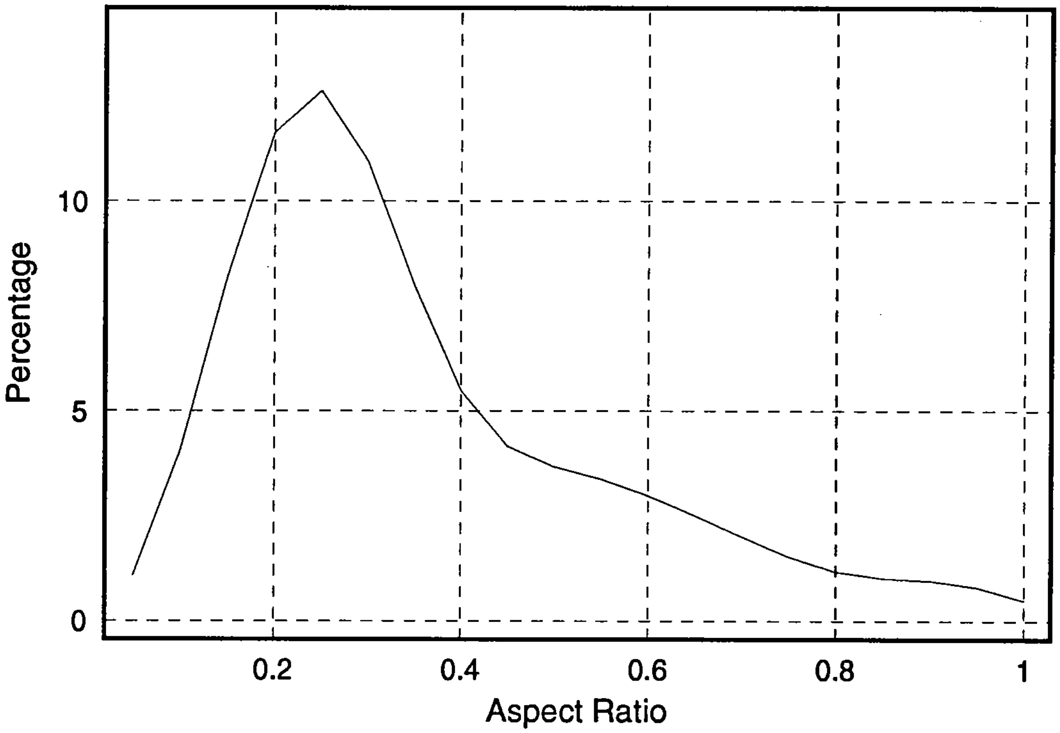

Figure 6.4. Distribution of the aspect ratio of the pores.

In the first development phase of the FABERCOAT System, we decided to use the total mean porosity as object function for its smooth behaviour. The temperature and porosity profiles and the picture of the coating section will be used in this phase as a check of the correctness of the run.

6.1. Selection of the free parameters for the search.

In order to get a desired value for the object function we investigated the input parameter space, changing values within a pre-defined range and looking at the effects on the objective function.

The "operator" parameters, with the corresponding typical values or ranges are listed in the following table.

COATTHIC	0 – 100 μm	coating thickness
GUNDISTA	70 – 150 mm	distance between gun and substrate

PARTRADI	2.5 – 50	radius of particles
PLASINTE	50 KW/m^2	intensity of plasma
SPRAYANG	0 – 20 °	spray angle (respect to normal)
SPRAYRAT	0.1 - 0.5 g/sec	powder feed rate
SUBSROUG	0 – 50 µm	substrate roughness
SUBSTEMP	300 – 800 °K	substrate temperature
NUMPASS	5 - 20	number of torch passes

The search is at present performed on a subset of these parameters that include GUNDISTA, NUMPASS, SUBSROUG, and SUBSTEMP.

7. Expert System Component

As described in a previous section presenting the FABERCOAT System functionalities, this system should have the capability to perform intelligent searches of the spray parameter space (that is, different sets of spray parameters). This should be directed by an expert system component. Due to the requirements of this component, a specialized search method has been selected to fill the roll as the expert system component.

In this particular domain of the processes of ceramic thermal coatings, where knowledge of the domain is hard to codify (that is, 'rules of thumb' are vague and difficult to construct), the selection of a search method for the expert system component is a good choice. This is due to the fact that, in general, search methods do not rely on 'rules of thumb', rather, rules are not required and an intelligent search algorithm can actually facilitate the user in identifying 'rules of thumb'.

The selection of the actual search method was made among the following possible methods: hill-climbing, simulated annealing, and genetic algorithms. In the end, genetic algorithms were selected as the most desirable method because:

(a) they can perform unbiased search,
(b) they make no assumptions about the search space (that is, the search space does not have to be smooth or regular),
(c) they carry out a more effective search of an irregular, multi-dimensional space because they search from a population of points rather than a single point,
(d) their search is not random, but intelligent (they utilize operators which are patterned after natural genetics), and
(e) they have been shown effective at finding optimal or near-optimal solutions to dynamic real-world problems [Goldberg, 1989] [PICGA, 1985] [PICGA, 1987] [PICGA, 1989].

The following sections will define and describe genetic algorithms in more detail.

7.1. Genetic Algorithms (GAs)

Genetic Algorithms (GAs) are defined as artificially intelligent (AI) search algorithms. They search by manipulating populations of structures (that is, binary strings representing data structures that symbolize possible solutions to a problem) into new solution populations using operators patterned after natural genetic operations. These operators may include reproduction, crossover, mutation, and others. The three simple GA operators will be discussed in a later section.

GA components can be split into two parts, application-independent components and application-dependent components, as shown in Figure 7.1.

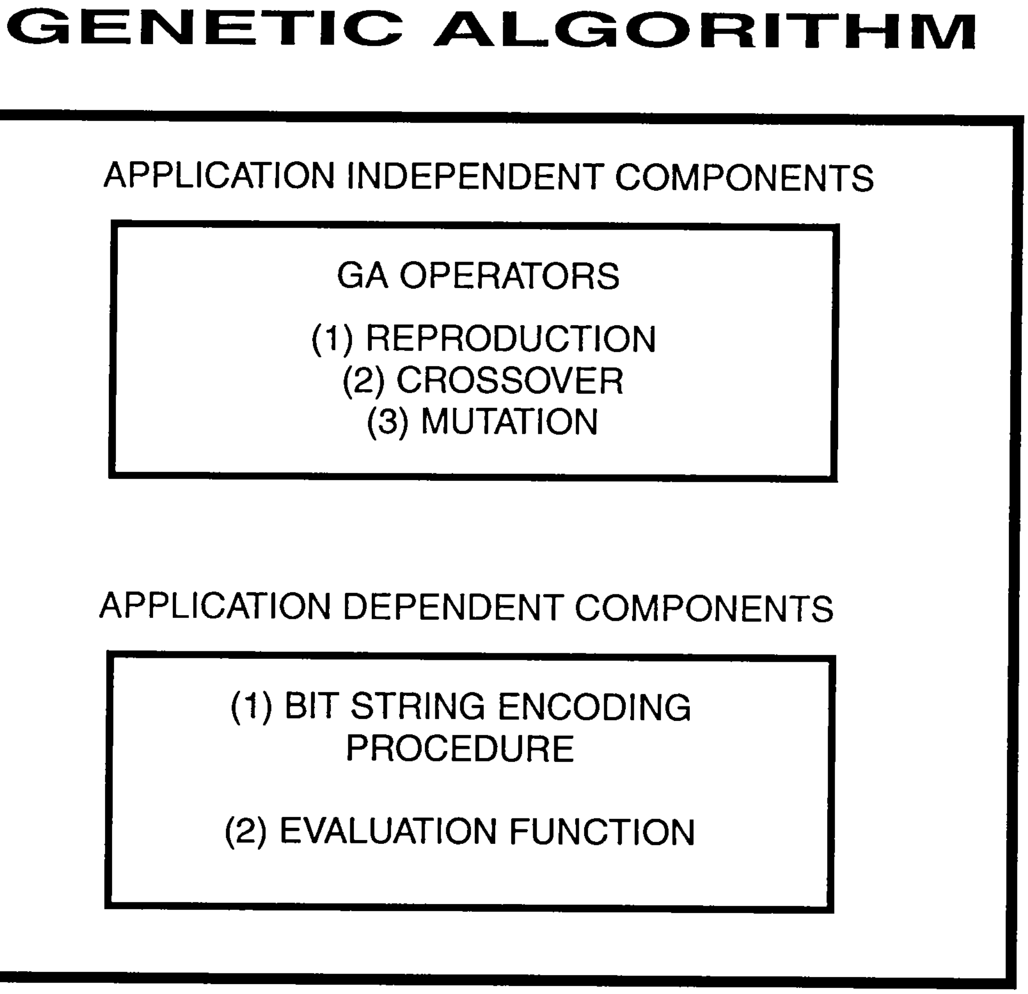

Figure 7.1 - Genetic Algorithm Components

The application-independent components (GA operators) will be discussed in a later section. For the application-dependent components, only two are required:

(1) a procedure to encode bit strings (chromosomes) into solutions to the problem and

(2) an evaluation function that will accept a solution to a problem and evaluate it's fitness or rating (this function is often called a black-box because the GA does not need to know anything specific about this function).

Figure 7.2 illustrate one possible encoding of bit strings into solutions to the problem (that is, in this case, spray parameters). Note that each parameter is mapped into one group of bit strings, and all of these bit strings are linked together to form one large bit string. In this example, each parameter is given 2 bits in the bit string (so that each parameter can take on 4 values), but this can be changed if the user would like to consider more or less values for a given parameter. Additionally, the user has the option to leave certain parameters out of the search, and have these parameters set to fixed values.

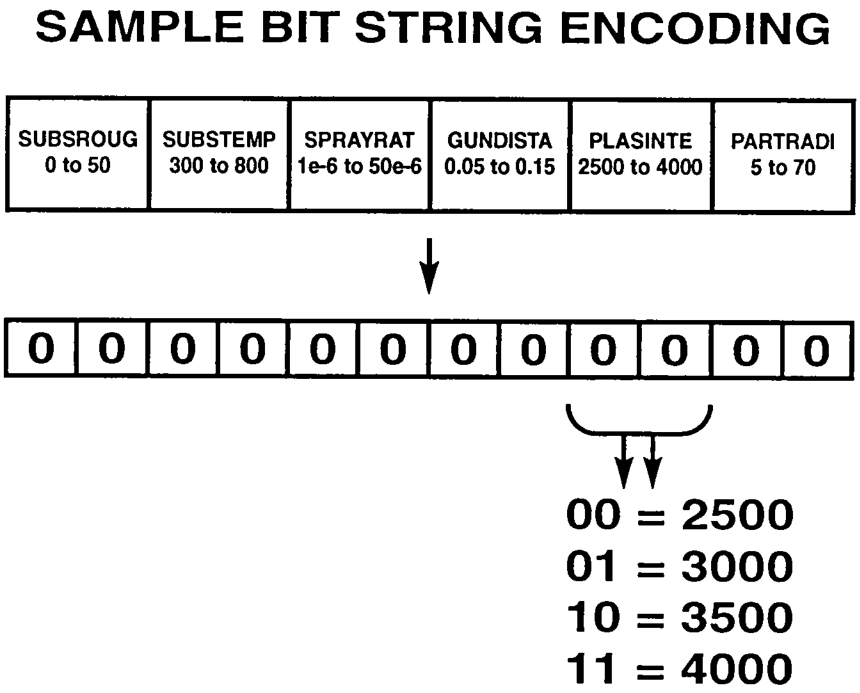

Figure 7.2 - Sample Bit String Encoding

The second application-dependent component is the evaluation function, which is also called the fitness function. This function is very similar to the objective function in traditional search problems. Its purpose is to give the GA a numerical evaluation of a possible solution in the same way that an objective function gives a numerical evaluation of a point in space. The GA uses an evaluation function to locate an optimal solution (traditionally, this is a maximum). In this case, the current fitness function used is (1/(| desired porosity - actual porosity |)), with the Deposition model at the centre of the fitness function. This function should allow the GA to locate a good set of spray parameters which produce the most desired final coating characteristics.

7.2. The Three Simple Genetic Algorithm Operators

The GA used in the project discussed here is a simple genetic algorithm consisting of the three basic GA operators. These are illustrated in Figure 7.3, and discussed below.

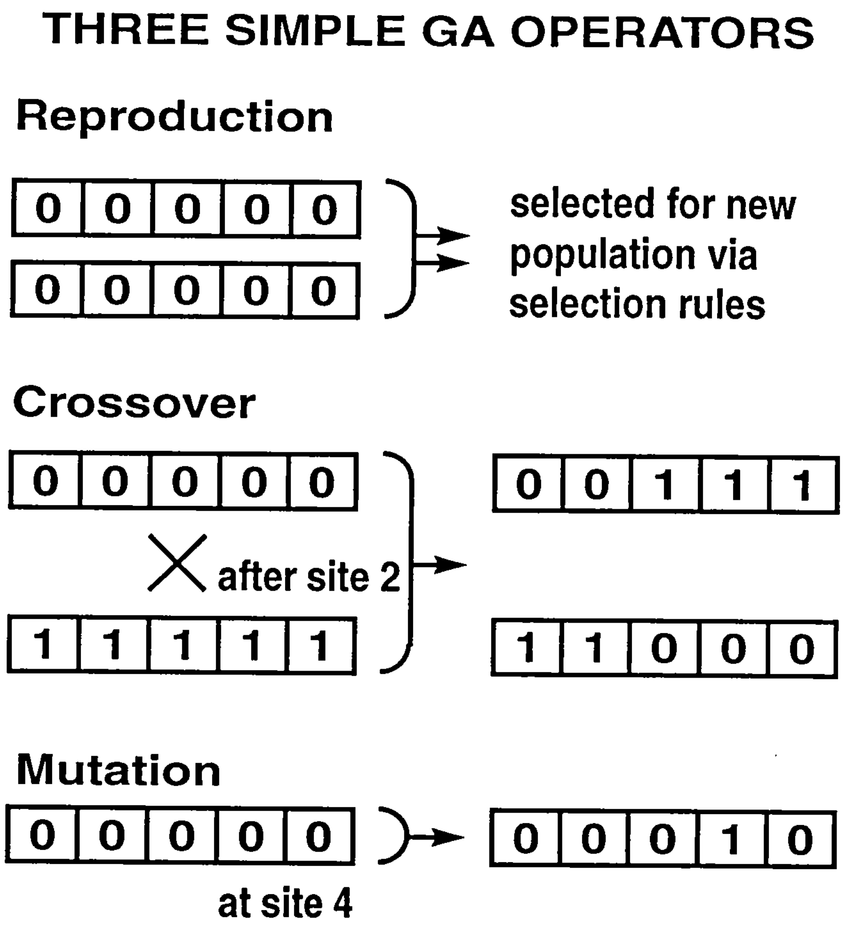

Figure 7.3 - The Three Simple GA Operators

First, reproduction is responsible for choosing the members that will be allowed to reproduce during the current generation. These members are selected based on their fitness values. All reproduction operators are biased to choose higher fitness members over lower fitness members, therefore high fitness characteristic are passed on to future generations. After the required number of population members have been selected for reproduction (some duplicates in this selection probably will exist), the next operator, crossover, can proceed.

The crossover operator randomly selects two members (that is, bit strings) from the new sub-population. Then a location within these two bit strings is selected at random. This location is used as the swapping point for the two strings, that is all bits to the right of this location on the first string are exchanged with all bits to the right of this location on the second string. For example, suppose the two following strings were selected for reproduction, string A=00000 and string B=11111. Then suppose the random bit location was selected as 2 causing the two strings to split after bit 2. This would result in two new strings, string C=00111 and string D=11000. After the new population has been filled with crossed over members, mutation can take place.

The mutation operator is simple: with a small probability, a bit will be selected within a string, and it will be flipped (that is, a 0 would become a 1 and a 1 would become a 0). Then these final members make up the new population, and all old members, from before reproduction, are thrown out. Because we now have a new population with new different members, each member must have its fitness evaluated so this evolutionary process can continue.

These are the three basic GA operators, but many variations on these and other operators exist. For a description of other operators and further details about GAs refer to [Goldberg, 1989] [Holland, 1975] [PICGA, 1985] [PICGA, 1987] and [PICGA, 1989].

7.3. The Genetic Algorithm Evolutionary Process

GAs begin with a population of randomly generated members. The GA then requests that each individual member in the population have its fitness evaluated. This evaluation is done in the fitness function, and the fitness value is returned to the GA. Once a GA has a completely evaluated population, the GA operates on these members to form a new population. This can be thought of as a generation of parents producing a generation of children. Although the new population contains characteristics of the old population, all the new members are different from the members in the last population (unless just by chance, some of the crossovers produced members which were already existing in the last population), so all of its new members must now be evaluated. As this process continues with fitness evaluation and execution of GA operators, new generations of members are created. (Later this process will be discussed with respect to the Deposition Model-GA (DEP-GA)). These new populations are generally more fit (that is, they have higher fitness values) than earlier populations because evolution favours stronger, more fit individuals, due to the nature of the GA operators (reproduction, crossover, and mutation discussed above).

7.4. Linking the Deposition Model and the Genetic Algorithm

To allow the GA to search the space of the Deposition model spray parameters, the Deposition model is linked to the GA, and the GA uses the Deposition model as the evaluation function. Therefore, whenever the GA wants to evaluate a set of spray parameters, the Deposition model is called, and the final outcome is returned to the GA so that a fitness can be computed. This new hybridized-system component is called the Deposition Model-GA or the DEP-GA.

Figure 7.4 illustrates how the Deposition Model (DEP Model) and GA are linked to form the DEP-GA component.

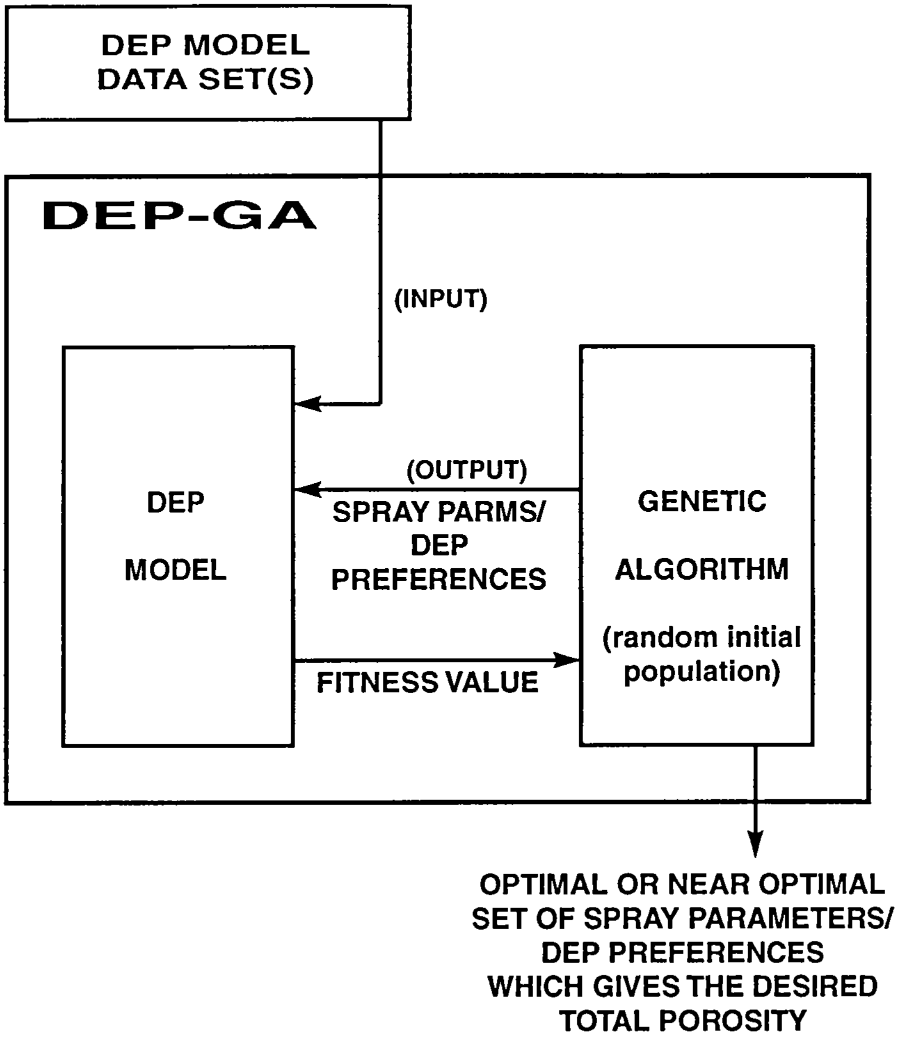

Figure 7.4 - Structure of the DEP-GA

Additionally, Figure 7.5 contains a flow chart of the steps executed by the DEP-GA to find a near-optimal spray parameter set for given desired final coating characteristics.

The three main steps involved in the DEP-GA are illustrated in Figure 7.5. First, the DEP Model and the GA are initialized. The initialization of the DEP Model actually involves nothing. The initialization of the GA involves establishing an initial, completely random population of bit strings. Note that these bit strings symbolize a set of spray parameters.

The second main step in the DEP-GA is the fitness computation. This involves taking each GA population member and executing one or more DEP Model simulations using the spray parameters represented by this member.
This fitness evaluation step is executed many times because new population members are continually being generated by the GA. Fitness evaluation is usually continued until the GA has converged.

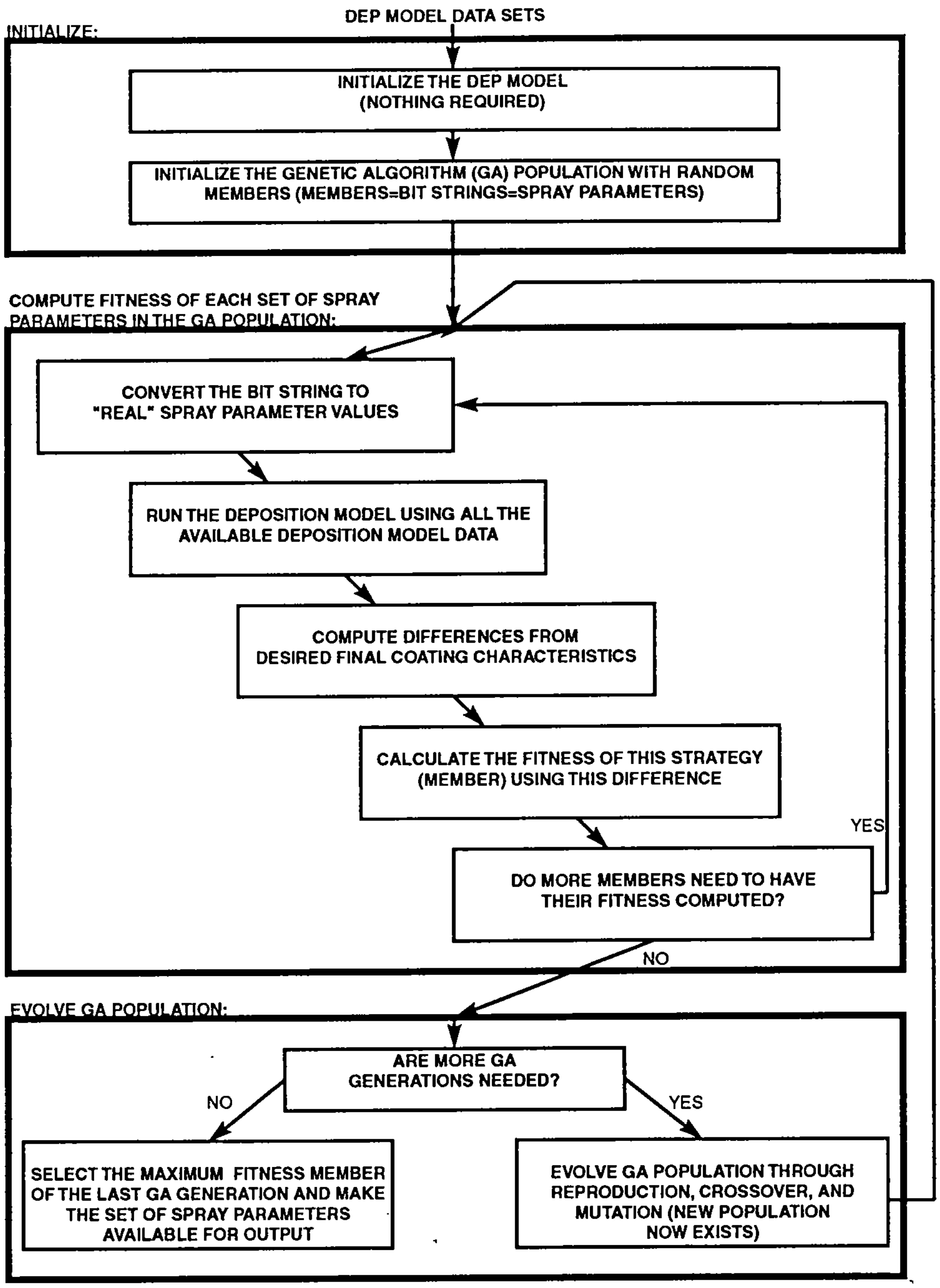

Figure 7.5 - DEP-GA Flow Chart

The last main step is the evolution of the GA population. This involves manipulations on the bit strings (that is, operations on the population members). The three manipulations, or operators, used in the DEP-GA are reproduction, crossover, and mutation, as described above.

7.6. DEP-GA Performance

We expect, when the FABERCOAT System is fully operational, that DEP-GA runs will provide the user with a good set of possible spray parameters which the operator can then use to determine the actual spray parameters to use in a real spraying. This will be provided by the searching capabilities of the GA. We expect the DEP-GA to perform similarly to most GAs, in that the average fitnesses of the populations, over time, will increase. That is, the members in the later populations will converge on maximum members in the space (i.e., on optimal sets of spray parameters). This anticipated performance of the DEP-GA is shown in Figure 7.6.

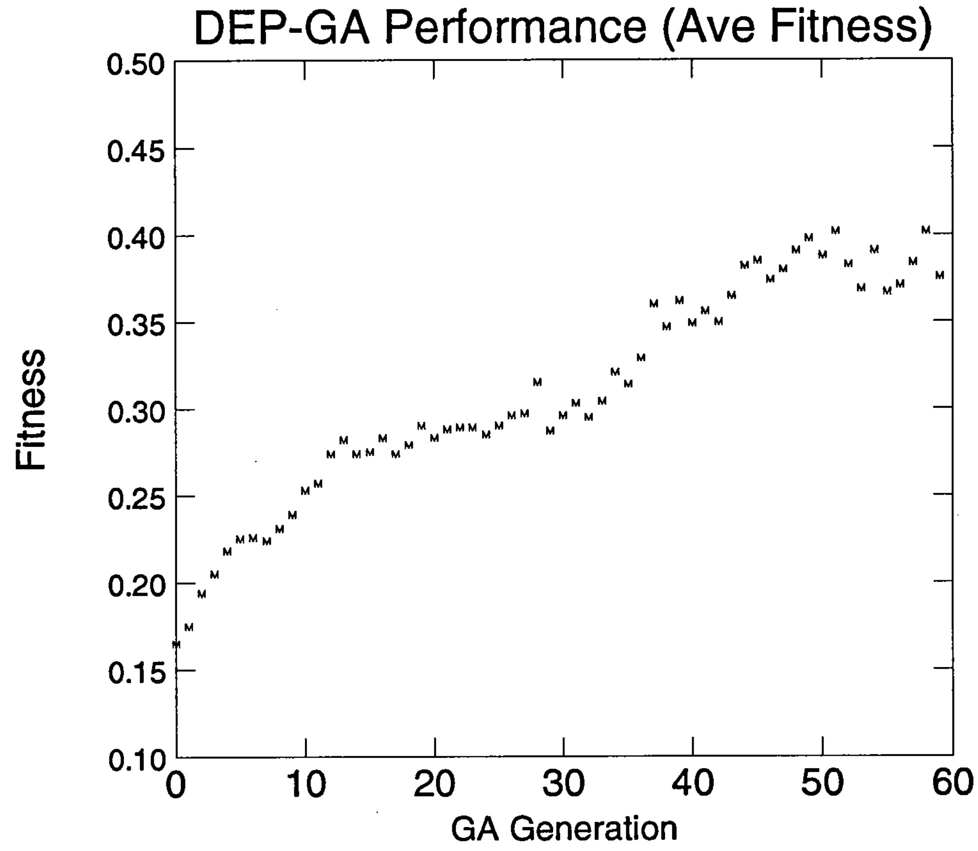

Figure 7.6 - DEP-GA Performance

8. Computer Aspects

The FABERCOAT System integrates a graphical user interface (GUI) around the Deposition model and genetic algorithm, thereby making the system easier to use. This GUI is implemented using a public domain tool kit called SUIT. One of the largest advantages of SUIT is that it makes the FABERCOAT System portable, that is, it will run on 4 different computer platforms: UNIX X-Window machines, IBM-PC DOS machines, IBM-PC MS-Windows machines, and Macintosh machines (as long as there is a Deposition model compiled with the FABERCOAT System). At the present, it is most desirable to run the FABERCOAT System on a fast UNIX X-Window workstation because these machines have the greatest ability to run the Deposition model in a reasonable amount of time (2 to 10 minutes for one simulation; about 10 hours for a full DEP-GA run), but in the future, when all machines become faster, it should even be reasonable to run the DEP-GA component on an IBM-PC type machine.

9. Conclusions

To summarize the FABERCOAT System is a design tool developed as part of a CEC project among production laboratories of plasma-sprayed coatings.

Specifically the FABERCOAT System will utilize a genetic algorithm to find a set of spray settings which can be used to achieve one particular set of desired final (and measurable) properties of the coating. At present, these desired outcomes are based on total porosity requirements. The FABERCOAT System is centred around a model for the plasma spray manufacturing process developed and validated under the CEC project. The model has been used to "harvest" the information gained in the extensive production and characterization programme of the project and has provided an innovative way of feeding back this information to the end-users (mainly technicians at the production laboratories).

Other issues considered in the development of the system concept have been:
portability - it is important to insure the system is portable across many different computer platforms. In particular the members involved in the project have different hardware environments, while it is necessary that all partners be able execute the FABERCOAT System.
standardized program design - program design (related to portability since the system must move between participants) requires modularity and consistency of practice. The approach used will allow users to add new models to the system.
easy of use - information presented in a friendly, graphical, non-redundant, consistent manner, will enable technicians the use the system in a transparent way with regards to the inner workings of the system.

References

[Cirolini, 1991] Cirolini, S., Harding, J.H., and Jacucci, G., 1991, "Computer simulation of plasma--sprayed coatings: I. Coating deposition model", Surface and Coating Technology, Vol. 48, pp 137--145.

[Fauchais et al, 1989] Fauchais, P., Grimaud, A., Vardelle, A., and Vardelle, M., 1989, "La projection par plasma: une revue", Annales de Physique, Vol. 14, pp 261--310.

[Goldberg, 1989] Goldberg, D.E. Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, MA: Addison-Wesley, 1989.

[Holland, 1975] Holland, J.H. Adaptation in Natural and Artificial Systems. Ann Arbor, MI: University of Michigan Press, 1975.

[Madejski, 1976] Madejski, J., 1976, "Solidification of droplets on a cold surface", Journal of Heat Mass Transfer, Vol. 19, pp 1009--1013.

[PICGA, 1985] Proceedings of an International Conference on Genetic Algorithms and Their Applications. Carnegie-Mellon University, July 24-26, 1985. Edited by John J. Grefenstette. U.S. Navy Center for Applied Research in Artificial Intelligence, 1985.

[PICGA, 1987] Proceedings of the Second International Conference on Genetic Algorithms. Massachusetts Instituted of Technology, July 28-31, 1987. Edited by John J. Grefenstette. L. Erlbaum Associates, 1987.

[PICGA, 1989] Proceedings of the Third International Conference on Genetic Algorithms. George Mason University, June 4-7, 1989. Edited by J. David Schaffer. Morgan Kaufmann Publishers, Inc., 1989.

[Yoshida et al., 1991] Yoshida T., Tomohisa, O., Hamatani, H., and Kumoaka, H., 1991, "Integrated fabrication process for solid oxide fuel cells using novel plasma spraying", Plasma Sources Sci. Tech. Vol.1, pp 195 - 201.