# SYBIL DSS : LOCALIZATION OF AGRICULTURAL RISK ASSESSMENT MODELS

Gianni Jacucci, Mark Foy and Carl Uhrik

Laboratorio di Ingegneria Informatica
Dipartimento di Informatica e Studi Aziendali
Università degli Studi di Trento
Via F. Zeni 8
I-38086 Rovereto - TN Italy

email: gianni@lii.unitn.it, mfoy@lii.unitn.it, carl@lii.unitn.it

## ABSTRACT

EU Project SYBIL implements a decision support system (DSS) to help farmers intelligently manage crops (reducing environmental impact and increasing economic returns). Agro-meteorological computer models are used to assess the risk of the crop to pest and fungus damage. In particular, we focus on grape and apple models that help farmers predict when diseases and funguses will attack their plants, so they can make intelligent decisions on preventing these attacks. To achieve this goal in an intelligent manner, a hybrid methodology for localizing these models to specific regions has been designed and implemented. Model localization is frequently necessary because models developed in one region often do not produce valid results when used in a different region. The main component of this intelligent localization methodology is a genetic algorithm (GA), an *artificial intelligence* (AI) search technique. By linking a genetic algorithm to an agricultural risk assessment model, the model become more robust because it is able to *adapt* to the region in which the model is being used. Preliminary testing indicates this localization methodology has the ability to allow regional agricultural models to be effectively utilized in other regions.

## INTRODUCTION

EU Project SYBIL (consisting of five partners from four countries) involves the implementation of a computerized decision support system (DSS) to assist farmers in intelligently governing their crops such that environmental impact is reduced and economic returns are increased. Existing agro-meteorological computer models from multiple sources are integrated into the one portable, user-friendly DSS designed to assess the risk of a crop to pest and fungus damage. By evaluating this risk, the farmer has the option to apply pesticides and fungicides only when needed and avoid using these, often environmentally damaging, chemicals blindly on a regular basis or when the risk of pest and fungus damage is small. This evaluation has the potential to save the farmer both time and money because expensive chemicals will not be applied when they do not benefit the crop.

In particular, this project is interested in *model technology transfer*, that is, transferring models developed and tested in one region to other regions. There are both many advantages and benefits and many disadvantages and problems with this strategy (to be discussed in the next sections), but the main problem we have encountered in utilizing this strategy is the problem of model accuracy in the new location. That is, often when a model developed in one region is used in a different region, the model outputs (such as, recommendations, results, and/or indicators) are inaccurate in the new region. We have categorized this situation into two cases:

(**Case1**) the model is not transportable (i.e., it is too location-specific and not robust enough to allow different conditions to enter into the model) and

(**Case2**) the model is transportable, but the model parameter settings need to be altered to allow the model to give accurate output values for the new region.

**Case 1** is a simple situation (in that the model is just too specific to operate outside one area), but difficult to address and solve if it is strongly desired to transfer the model. This case will not be addressed within this discussion. On the other hand, the dilemmas present in **Case2** (i.e., modifying/adapting/adjusting model parameter settings so that the model functions accurately in a new location) have lead us to develop a hybrid methodology that determines location-specific model parameter settings. This methodology is specifically designed for models that can be placed into **Case2** (i.e., cases where it is beneficial to dynamically adapt the model parameter settings to the existing location). At the heart of this methodology (which will be described later in more detail) is a genetic algorithm (GA) (an *artificial intelligence* search technique) linked with the agricultural risk assessment model engine. This is the combination that makes the methodology hybridized, and the general component created by this methodology is called an 'Agricultural Model-GA' or an AGMOD-GA.

We have used this methodology within our portable-user friendly DSS (called the SYBIL DSS) to find localized model parameter settings for the P.R.O. model, a biological life cycle model that simulates the growth of downy mildew (Plasmopara viticola, also called peronospora) on grape vines. This model is designed to help growers determine when it is necessary to spray against downy mildew. The hybrid component intelligently searches the space of P.R.O. model parameter settings and locates settings that match local conditions. Preliminary testing with the P.R.O. model indicates that this localization methodology has the capacity to allow the parameter values of regional models to be effectively adapted to regions other than where they were developed.

The following sections will elaborate on the issues involved in transporting models between regions, describe the hybrid methodology for localizing model parameter settings (including a detailed discussion of genetic algorithms), and discuss the application of this methodology to the P.R.O. model.

## TRANSPORTING MODELS BETWEEN REGIONS

The strategy of transporting models between regions is appealing because one hopes that the time invested in creating a useful system will be reduced, and the quality of the system will be increased, if existing models are employed. On the other hand, problems with transporting models can appear which can make the opposing strategy of creating new models more appealing. A discussion of the benefits and problems should reveal many of the issues surrounding the transportation of models between regions.

### <u>Advantages</u>

The main potential benefits and advantages of this strategy (i.e., transporting existing models between regions) versus creating new models include:

(a) minimized effort and cost (i.e., generally, less effort and cost is expended transferring and testing a model than designing, developing, implementing, and testing a new model)

(b) minimized risk (i.e., an existing model that has been tested and proven useful to agriculturalists in one region is at less risk of being problematic or rejected than a new model that no one has tested, and it is not sure that agriculturalists will react well to the new model)

(c) increased model robustness (i.e., by applying effort to transport an existing model, it is likely that more will be discovered about the model, thereby refining the model, making it in generally more robust and accurate)

(d) increased model modularity, openness, and independence (i.e., by making a computer model transportable, the model becomes more modular, open, and independent because in transporting models, parameter settings must be easy to adjust; this allows expert agriculturalists to more easily explore, understand and modify the model, even allowing the expert to quickly make changes by hand for testing purposes, often without recompiling the model code)

(e) increased availability, usability, and accessibility (i.e., by taking the step to perform the first transportation of a proven model, the model will become more available, usable, and accessible in the future by a larger set of agriculturalist due to the fact that once a model has been transported to a new region for the first time, work to transport to subsequent regions should be reduced)

(f) increased model technology distribution (i.e., if transportable models are available, regions that do not have resources or expertise to create new models will have access to methods that would never have been available to them without transferable models; this thereby increases the level of technology available in the new region and expands technology distribution).

Disadvantages

The opposing problems and disadvantages of transporting models include:

(1) inability to transfer (i.e., the model does not transfer well; it is too specific, fixed, and specialized, and adapting/changing the model is either very difficult or impossible; this includes the case where the concepts used in the model are truly only usable within certain types of regions, and they can not be generalized)

(2) model rejection (i.e., it is possible, because a model was developed in another region, agriculturalists will reject the model because they feel it simply can not work in their region, even if changes are made)

(3) unexpected costs and efforts (i.e., it may turn out that the resources expended to transport the model end up exceeding the resources that would have been expended to create a new model, due to unforeseen particularities in the model (e.g., features that were tailored to the original area where it was developed))

(4) inadequate data available (i.e., the data available in the new region is not sufficient to run the model, and the model can not be simplified enough to allow different types of data to be used)

(5) unsound model (i.e., the model may have appeared stable when it was used in the original region, but upon further investigation during and after transferring the model, it is found that the logic of the model is not solid, and that it would have been better to start with a new model)

The view presented here takes the stance that the benefits and advantages outweigh the problems and disadvantages. Therefore a hybrid methodology for localizing models is proposed which attempts to fully exploit the potential benefits of transporting models.

## A HYBRID METHODOLOGY FOR LOCALIZING MODEL PARAMETER SETTINGS

The advantages and disadvantages of transporting existing models between regions have motivated the selection of this strategy. To make this, strategy as robust as possible, an intelligent localization methodology is proposed which combines multiple components into one hybrid system to adapt/localize models.

The theory of this methodology is that by utilizing historical data from a particular region, a model's parameter settings can be *adapted* so that the new parameters allow the model to work well in the particular region. This adaptation is done by trying to match the model parameter settings (which should be easily changeable) to the particular region. That is, new model parameter settings are found which allows this model to give accurate output values (such as recommendations, results, and/or indicators) when run in the particular region. To find matching model parameter settings, intelligent search is performed which utilizes historical data as part of the objective function (i.e., the search attempts to *fit* the model parameter settings such that when used in the model, the model gives output near to the given historical data).

For example, suppose a model has been developed in **RegionA** that gives an assessment of the risk that a certain plant will be damaged by a certain fungus. This model uses an instantiation of model parameter setting values (such as values for thresholds and coefficients, etc.) called **ModelParmsA**. The model has proved quite useful in **RegionA**, and now agriculturalists in **RegionB** are interested to use this model. Upon first running this model in **RegionB** (with the original model parameter settings, i.e., **ModelParmsA**), the model gives output values (such as recommendations, results, and/or indicators) which are inaccurate. As an attempt to make this model work accurately in **RegionB**, we suggest that new model parameter settings (which are different from **ModelParmsA**) be derived. By applying the methodology discussed here (which requires historical data and an intelligent search method to be described below), it is possible to locate new model parameter settings, and once this is done, we can call this new instantiation of settings **ModelParmsB**. We propose that after applying this methodology, the model should produce accurate output values (about the same certain plant and the same certain fungus) in **RegionB** when using **ModelParmsB**.

Overall, by following this methodology, a component will be created which can search for good model parameter settings such that when the given model is applied and run at the location in question, the output values given will be consistent with the historical epidemiological data; moreover it is hoped that this will also the model to be generally used in this region, producing accurate output values on data which it has not seen. We can call this component that performs this search/adaptation an expert system component since it modifies and adjusts a model to work in a new location in the same way an expert would modify and adjust a model.

Additionally, it should be emphasized that this methodology is particularly appealing because it is not a strictly empirical or analytical, but both. That is, this methodology does not perform a search to *fit the historical data* from a particular location into an empirical algorithm; rather it performs the search in a larger context, *fitting the model parameter settings* to a particular location. Therefore, the resulting instantiation of the localized agricultural risk assessment model (with the new parameter settings inside) is as good (or as bad) as the original model; consequently, if the model is biologically significant (e.g., if it simulates biological events) then this is not lost by this localization methodology since the model is used in the same form (i.e., the structure of the model is left intact), only the model parameter settings are changed.

In general, this methodology prescribes the utilization of:

(i) historical agricultural model data,
(ii) historical epidemiological data,
(iii) the agricultural risk assessment model, and
(iv) an intelligent search method (in this case, a genetic algorithm, also called a GA, which is an artificial intelligence search technique).

### Historical Agricultural Model Data

Historical agricultural model data is the basic data required by the model in question In the domain of agricultural risk assessment models, this often includes meteorological data since frequently this is an important input to the model. In most cases, the more historical agricultural model data that is available, the better.

### Historical Epidemiological Data

Historical epidemiological data (e.g., data regarding the occurrence of fungus or pest problems in past years) is a particularly notable part of this methodology because it is used to *fit* the model parameter settings to the new region in question. Therefore, when constructing a component using the methodology described here, it must be possible to match model outputs to some event (or events) in the real-world. For example, if the only historical epidemiological data available is a set of historical primary infection dates, then the model should be able to produce a primary infection date (albeit a guess of the primary infection date) as one of it's outputs. In this way, it will be possible for the model to be adapted such that the primary infection date produced by the model nearly matches the actual primary infection date (i.e., the date specified by the historical epidemiological data).

### Agricultural Risk Assessment Model

In this methodology, the agricultural risk assessment model (i.e., the engine or core of this model) is fundamental because it will be used to obtain evaluations of how well particular model parameter settings work in the given region (i.e., with the given data). In particular, the intelligent search method will repeatedly call upon this model engine as it constructs new model parameter settings that need to have their worth evaluated.

### Intelligent search Method

The intelligent search method is an important part of this component because, in this particular domain of agricultural risk assessment models, knowledge of the domain is hard to codify (i.e., ’rules of thumb’ are vague and difficult to construct), and the selection of an intelligent search method can help to alleviate this difficulty. This is due to the fact that intelligent search methods do not rely on ’rules of thumb’, rather, rules are not required and these methods can actually facilitate the user in identifying ’rules of thumb’.

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

(a) they can perform unbiased search,
(b) they make no assumptions about the search space (i.e., 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) (Grefenstette, 1985) (Grefenstette, 1987) (Schaffer, 1989).

The next major section will define and describe genetic algorithms in more detail. This will then be followed by a description of applying this methodology to a particular model (the P.R.O. model), and the results obtained.

## GENETIC ALGORITHMS (GAs)

Genetic Algorithms (GAs) are defined as artificially intelligent (AI) search algorithms. They search by manipulating populations of structures (i.e., 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 1. 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 2 illustrates one possible encoding of bit strings into solutions to the problem (e.g., in this case, a partial set of P.R.O. model parameters, a model that is discussed later). Note that each parameter is mapped into one group of bit strings, and all of these bit

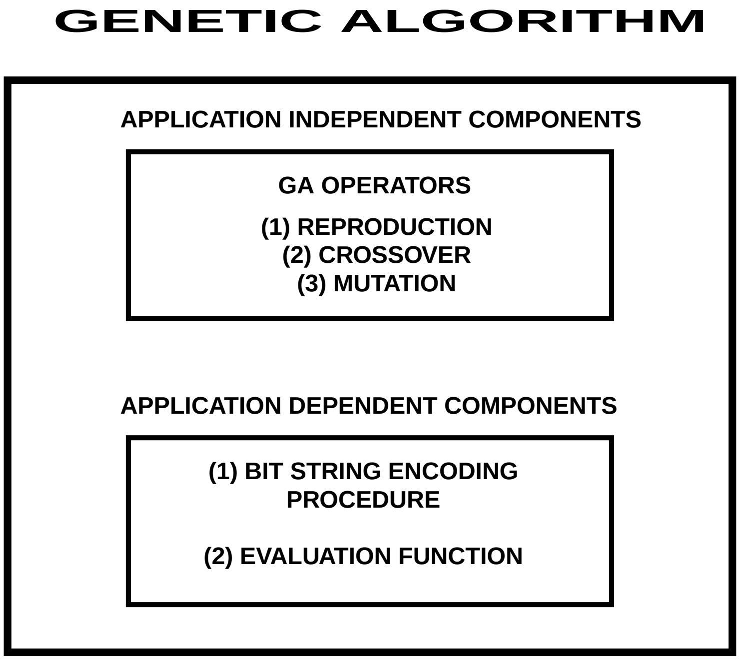

Figure 1. Genetic Algorithm Components

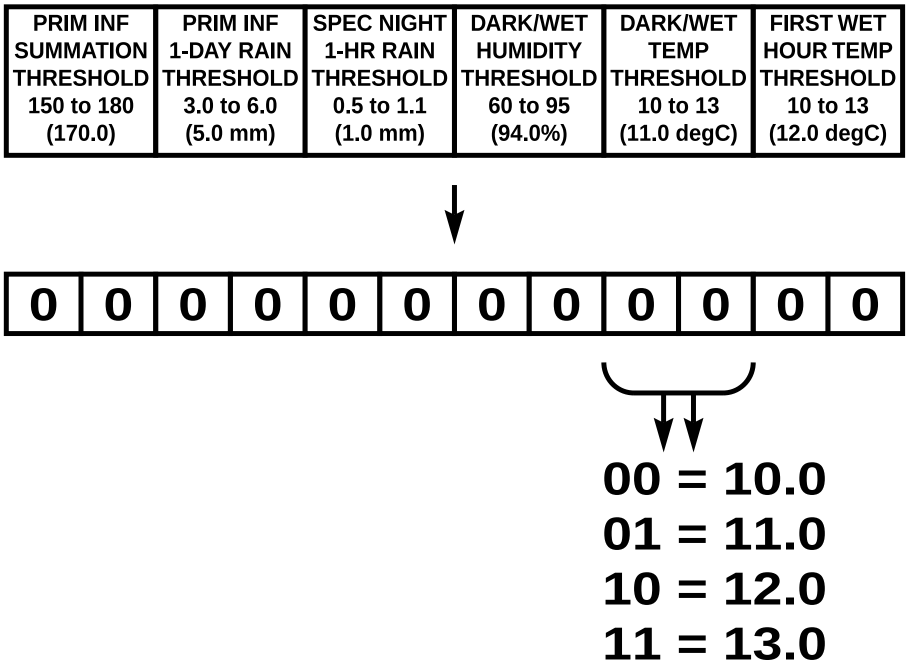

Figure 2. Sample Bit String Encoding

strings are linked to form one larger bit string. In this example, each parameter is given two bits in the bit string (so that each parameter can take on four values), but this can be changed if the user would like to consider more or fewer 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.

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 the case of the P.R.O. model (and probably many agricultural risk assessment models), a valid fitness function can be structured as (*1/(|desired target date - actual target date calculated by the model|)*), with the agricultural risk assessment model (such as the P.R.O. model) at the centre of the fitness function. This function should allow the GA to locate good model parameter settings that allow the model to run accurately in the new region.

<u>The Three Simple Genetic Algorithm Operators</u>

The GA used in the system discussed here is a simple genetic algorithm consisting of the three basic GA operators. These are illustrated in Figure 3.

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 characteristics are passed on to future generations. After the required number of population members are selected for reproduction (some duplicates in this selection probably will exist), the next operator, crossover, can begin.

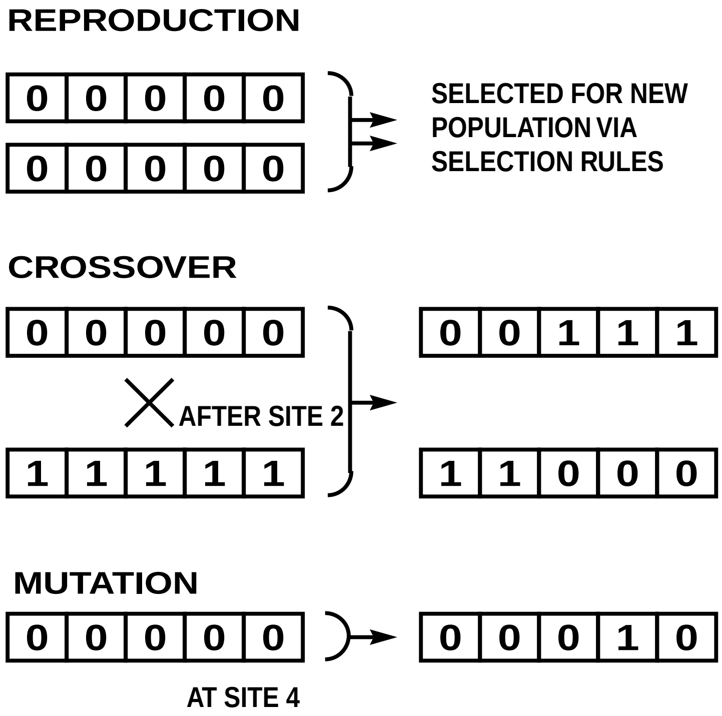

Figure 3. The Three Simple GA Operators

The crossover operator randomly selects two members (i.e., 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 are selected for reproduction, string A=**00000** and string B=**11111**. Then suppose the random bit location is 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 is filled with crossed over members, mutation can take place.

The mutation operator is simple: with a small probability, a bit is selected within a string, and it is flipped (i.e., a **0** becomes a **1** and a **1** becomes a **0**). Then these final members comprise 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) (Grefenstette, 1985) (Grefenstette, 1987) (Holland, 1975) and (Schaffer, 1989).

## 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

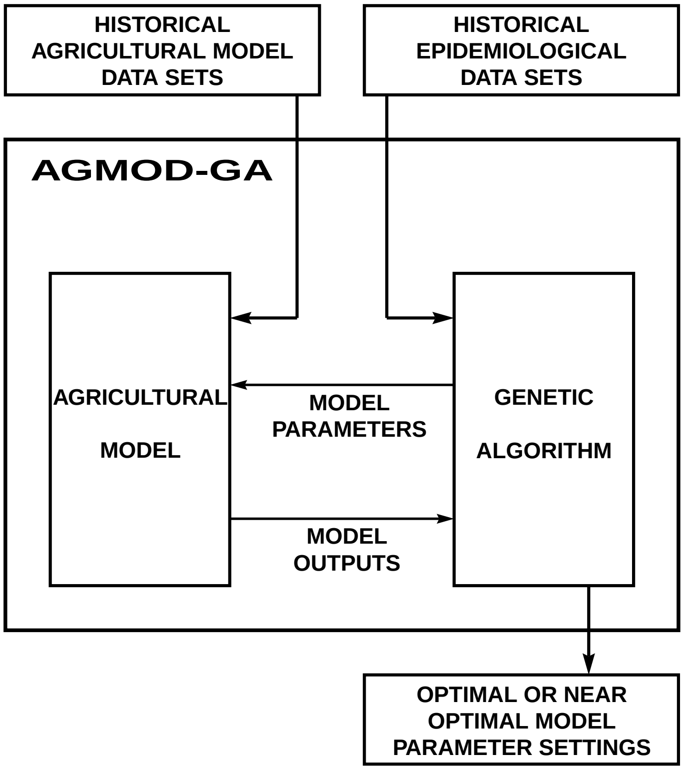

Figure 4. Structure of an AGMOD-GA

the new members are different from the members in the last population (unless just by chance, some of the crossovers produced members that already existed 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 is discussed with respect to a generic agricultural model linked to a GA (which we call an AGMOD-GA).) These new populations are generally more fit (i.e., 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).

## AGRICULTURAL MODEL-GA (AGMOD-GA)

### Linking an Agricultural Risk Assessment Model to a Genetic Algorithm

To allow a GA to search the space of an agricultural risk assessment model’s parameters, the agricultural model is linked to a GA, and the GA uses the model as the evaluation function. Furthermore, the model uses the historical agricultural model data (as this is necessary to run the model in the given historical years), and the GA additionally uses the historical epidemiological data (discussed earlier) in combination with the output of the model. Whenever the GA wants to evaluate one instance of model parameter settings, the agricultural model is called, and the final outcome is returned to the GA so that a fitness can be computed. We call this resulting general component created by employing this hybridized methodology an ’Agricultural Model-GA’ or an AGMOD-GA.

Figure 4 illustrates an agricultural model and a GA linked to form an AGMOD-GA.

Additionally, Figure 5 contains a flow chart of the steps executed by a typical AGMOD-GA to find near-optimal model parameter settings for the given desired behaviour (i.e.,

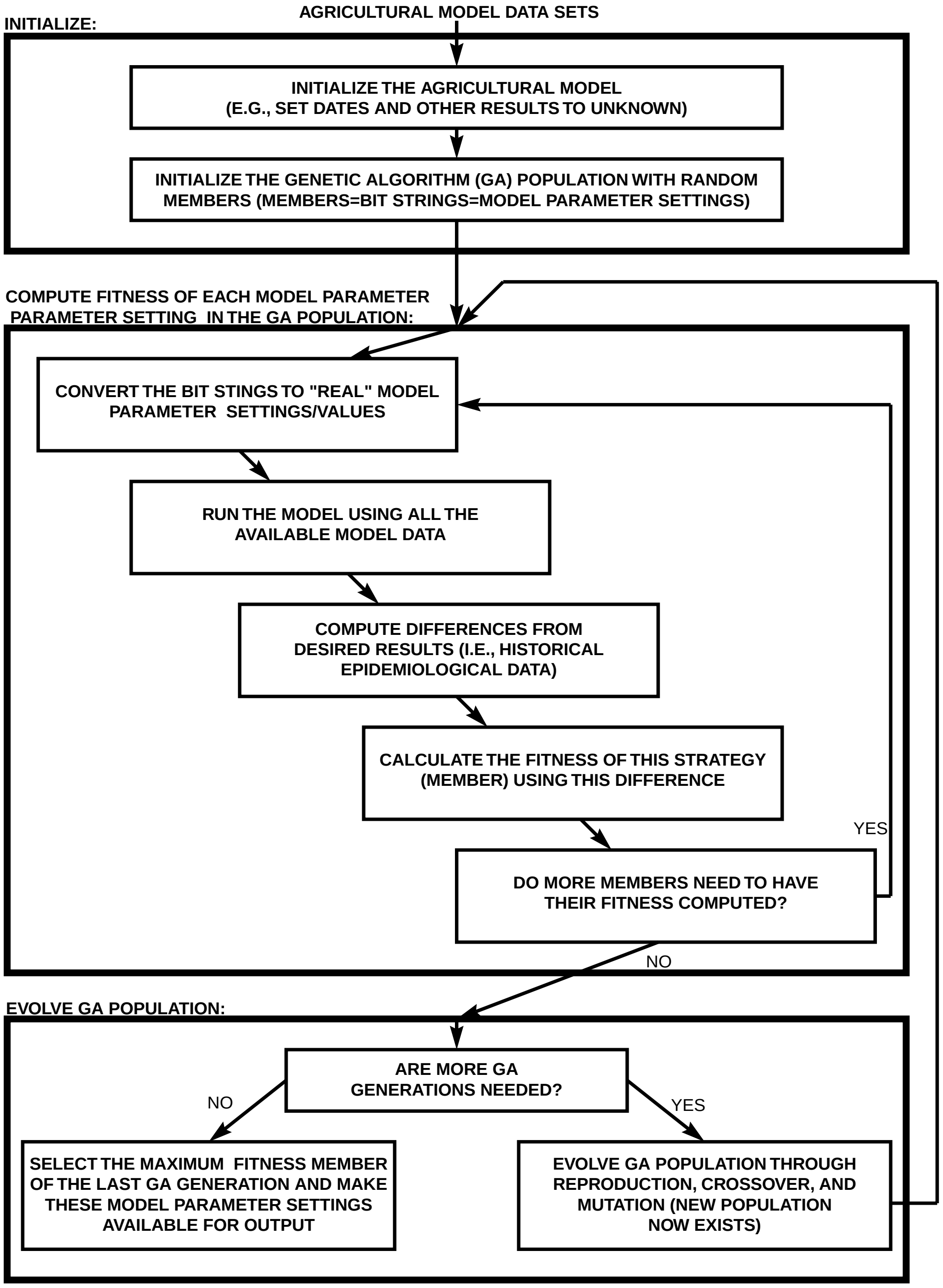

Figure 5. AGMOD-GA Flow Chart

matching the given historical epidemiological data). The three main steps involved in the execution of a typical AGMOD-GA are illustrated. First, the agricultural model and the GA are initialized. The initialization of the model in reality may involve nothing. The initialization of the GA involves establishing an initial, completely random population of bit strings. Note that these bit strings symbolize one instance of model parameter settings.

The second main step in a typical AGMOD-GA is the fitness computation. This involves taking each GA population member and executing one or more model simulations using the model parameter settings 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.

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

### AGMOD-GA Performance

We expect, when particular instantiation of this methodology are employed, that AGMOD-GA runs will provide the user with good model parameter settings that the operator can then use in the original model (e.g., use in the original model to obtain a suggestion of when to spray a crop so no damage is done to the crop, but not before it is necessary). This will be provided by the searching capabilities of the GA. We expect instantiations of the AGMOD-GA to perform similarly to most GAs, in that the average fitness of a generation, over time, will increase. That is, the members of later populations will converge on maximum members of the space (in this case, on optimal sets of model parameter settings). This anticipated performance of a general AGMOD-GA is shown in Figure 6. The two curves ("MAXIMUM FITNESS" and "AVERAGE FITNESS") illustrate the GA converging on high fitness members. This type of performance has in fact been achieved on the PRO-GA constructed using the P.R.O. model and a simple GA. The next section will discuss this instantiation of an AGMOD-GA in more detail.

## AN EXAMPLE - THE P.R.O. MODEL FOR GRAPES

### Description and Origin of the P.R.O. Model

The P.R.O. (**P**lasmopara **R**isikoprognose **O**ppenheim or **P**lasmopara **R**isk **O**ppenheim) model for grapes was the first model selected for the application of the localization methodology discussed above. This analytical model is a biological life cycle model that simulates the infection and growth of downy mildew (Plasmopara viticola, also called peronospora) on grape vines based on meteorological conditions. The model was developed in Rheinhessen, Germany by Dr. Georg K. Hill (Hill 89) (Hill 90a) (Hill 90b)

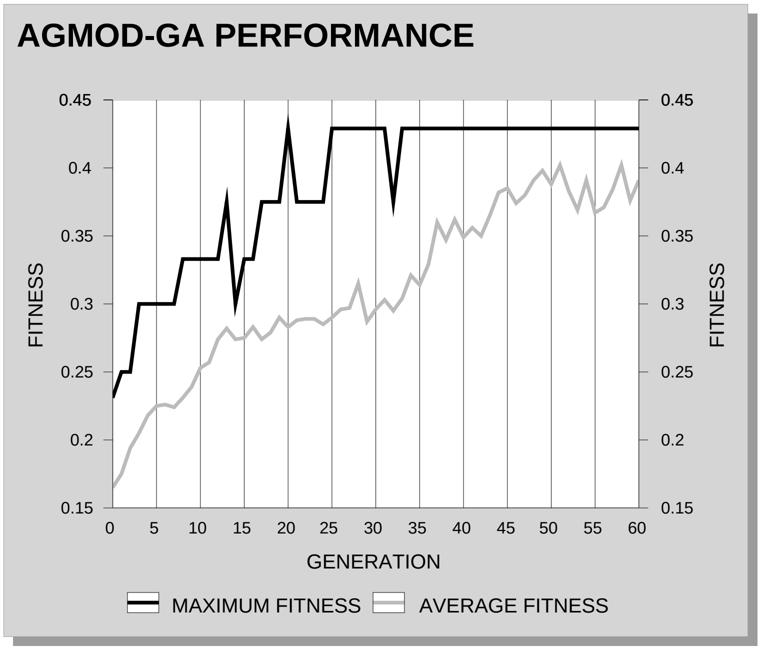

Figure 6. Typical AGMOD-GA Performance
(Note: The values plotted here are actually from one PRO-GA run -
the PRO-GA will be discussed below)

(Hill 90c) (Hill 93a) (Hill 93b). It was designed to help growers determine when it was necessary to spray grape vines against peronospora.

The model had been used by multiple Rheinhessen region farmers with good results; that is, the information provided to the farmer has assisted in making providing intelligent decisions about when to first spray against peronospora. The goal is to overcome the habit (which is not based on temporal information) of performing the first spray early in the season (possibly around May), which is often before it is necessary. This goal is approached by using the P.R.O. model to produce interpreted operational temporal information (i.e., useful up-to-date information about the status of the peronospora growth), then examining this information, and deciding if it necessary to spray at the current moment, or if spraying can be delayed (possibly many weeks) because the grape vines are not currently at risk to being damaged by peronospora. In the common cases where spraying can in fact be delayed beyond when an agriculturalist would normally spray, the overall number of interventions and amounts of chemicals sprayed on the crop are reduced. Agriculturalist using this model in the region around Rheinhessen have been able to save between one and four spraying applications per year, with an average saving of two (Hill 93b).

Results of Transporting the P.R.O. Model

After deciding that the P.R.O. model was a good choice for inclusion into the SYBIL DSS (and therefore a good choice for trying to transfer this model between countries), an instantiation of the model (with only small changes so that the model would accept other types of meteorological data) was programmed into the SYBIL DSS, and test runs were made with various data from regions outside Rheinhessen. Upon running these tests, it was found that the output values (which in the case of the P.R.O. model are: the primary infection data, the end of the incubation period, a list of special night occurrences, and a recommended spray date), were inaccurate in the new regions. That is, the P.R.O. model outputs were rejected by agricultural experts based on their historical epidemiological data and general knowledge of when epidemiological events occur in their regions.

For example, Table 1 displays the results from running the P.R.O. model with data from an area inside the Trentino region of Italy. As this table shows, the dates produced by the original P.R.O. model using original model parameter settings (i.e., model parameter settings selected by Dr. Hill for the Rheinhessen area) (these dates shown in the column titled "PRO results using original PRO model parameter settings") for data coming from Trentino, only approached the dates known to be correct from observations done by agricultural experts in Trentino (these dates shown in the column titled "Actual Dates") for the primary infection dates (rows titled "Prim Inf 19xx"). For the recommended spray dates (rows titled "Rec Spray 19xx"), the model could not even produce estimates of this date (with this data from Trentino) for two out of the three years in which actual dates were available for comparison. Therefore, the model in this state is of little use to agriculturalists in Trentino since it is generally not able to produce an accurate estimate of the recommended spray date. Additionally, the difficulties observed in this case also held true for data taken from other regions, so overall the P.R.O. model was problematic because it did not give accurate output when run in regions external to where it was developed.

In addressing this problem, we proposed that this model corresponded to **Case2** described in the introduction. That is, we assumed that the model parameter settings could be fit to local conditions and that the problems stem from the fact that the model parameter settings were custom tailored to the region where it was developed (in this case,

|  | Actual Dates | PRO results using original PRO model parameter setttings | Difference between results w/ original PRO settings and actual dates |
|---|---|---|---|
| Origin of Model Parameter Settings |  | Dr.Hill's original settings from Rheinhessen |  |
| Prim Inf 1988 |  | 5/3 |  |
| Rec Spray 1988 |  | 12/31 |  |
| Prim Inf 1989 |  | 5/13 |  |
| Rec Spray 1989 |  | 7/3 |  |
| Prim Inf 1990 | 5/24 | 6/5 | 12 |
| Rec Spray 1990 | 6/7 | 12/31 | 207 |
| Prim Inf 1991 | 5/11 | 5/11 | 0 |
| Rec Spray 1991 | 6/7 | 12/31 | 207 |
| Prim Inf 1992 | 5/5 | 5/23 | 18 |
| Rec Spray 1992 | 6/1 | 7/10 | 39 |
| Prim Inf 1993 |  | 4/30 |  |
| Rec Spray 1993 |  | 12/31 |  |
| Avg. Difference |  |  | 80.50 |
| Comments | all dates formated in the form: month/ day | when a date of 12/31 is used, this indicates the model never reached this event | each numerical difference is in absolute days |

Table 1. Results from running the P.R.O. model on data from the Trentino region of Italy with original parameter settings (Note: actual dates in Trentino were only available for three years)

Rheinhessen, Germany). In addition, we wanted to keep the same basic overall structure of the P.R.O. model because it had been proven valid and useful in the past. Therefore, we applied the methodology described above and created an instantiation of an AGMOD-GA for the P.R.O. model, calling the new component the PRO-GA.

By running this PRO-GA component with data from a particular region, new model parameter setting values can be found which should allow the model to give accurate output values (i.e., recommendations, results, and/or indicators) for the region in question. Upon running the PRO-GA with three years (1990, 1991, and 1992) of historical agricultural model data (in the case of the P.R.O. model, meteorological data) and historical epidemiological data (in this case, primary infection dates and recommended spray dates) from an area inside the Trentino region of Italy, new model parameter setting values were derived. Table 2 shows the results of using these new model parameter settings inside the P.R.O. model, again running with data from Trentino.

As expected, this table shows that the model parameter settings, derived by the PRO-GA component, allow the P.R.O. model to function with significantly increased accuracy on the Trentino data. The absolute average difference in days (i.e., the absolute average of how far off the model output is from actual dates) (in the row titled "Avg. Difference") between actual dates (column titled "Actual Dates") and dates produced by the running the P.R.O. model (column titled "PRO results using 1st PRO model parameter settings") drops from 80.50 days to 0.83 days when new parameter settings derived by the PRO-GA

|  | Actual Dates | PRO results using original PRO model parameter setttings | Difference between results w/ original PRO settings and actual dates | PRO results using 1st PRO model parameter settings | Difference between results w/ 1st PRO settings and actual dates |
|---|---|---|---|---|---|
| Origin of Model Parameter Settings |  | Dr.Hill's original settings from Rheinhessen |  | from a PRO-GA run using historical data=90,91, 92 |  |
| Prim Inf 1988 |  | 5/3 |  | 5/3 |  |
| Rec Spray 1988 |  | 12/31 |  | 5/11 |  |
| Rec Spray 1989 |  | 7/3 |  | 5/13 |  |
| Prim Inf 1990 | 5/24 | 6/5 | 12 | *5/24* | *0* |
| Rec Spray 1990 | 6/7 | 12/31 | 207 | *6/6* | *1* |
| Prim Inf 1991 | 5/11 | 5/11 | 0 | *5/12* | *1* |
| Rec Spray 1991 | 6/7 | 12/31 | 207 | *6/7* | *0* |
| Prim Inf 1992 | 5/5 | 5/23 | 18 | *5/4* | *1* |
| Rec Spray 1992 | 6/1 | 7/10 | 39 | *5/30* | *2* |
| Prim Inf 1993 |  | 4/30 |  | 4/30 |  |
| Rec Spray 1993 |  | 12/31 |  | 6/23 |  |
| Avg. Difference |  |  | 80.50 |  | 0.83 |
| Comments | all dates formated in the form: month/ day | when a date of 12/31 is used, this indicates the model never reached this event | each numerical difference is in absolute days | italic results indicate interesting results |  |

Table 2. Results from running the P.R.O. model on data from the Trentino region of Italy with parameter settings found by the PRO-GA using all available historical data

are used instead of the original parameter settings. Generally, an average difference (i.e., an accuracy error) of 0.83 days is not significant or a critical inaccuracy, and this variation is within the tolerable limits.

Because this adaptation is performed with all available historical sets of data, this type of behaviour is generally expected since we are testing the accuracy on the same set of historical model data (in this case, meteorological data) that was used for adapting. This is still an important result because it shows that the model can be fit to the entire set of data, and that it is possible to find parameter settings that will give accurate results over many years. On the other hand, a better verification of this methodology is to adapt the model using one subset of historical model data, and then test the accuracy of the adapted model (i.e., the model with the new parameter settings) by running the model on a different subset of historical data.

Unfortunately, the currently available historical epidemiological data is very limited (three years of data from Trentino); therefore only a few small subsets of historical model data can be formed (in this case, only three interesting subsets: (1991, 1992), (1990, 1992) and (1990, 1991)). Therefore, it is not possible to test newly derived parameter settings as extensively as desired, but we have been able to perform three tests, the results of which are shown in Table 3.

Even with the limited amount of historical data, these results are still interesting and significant. For example, when adapting the model using historical epidemiological data from 1990 and 1991 (shown in the last two columns of the table) the average difference

|  | Actual Dates | PRO results using original PRO model parameter setttings | Difference between results w/ original PRO settings and actual dates | PRO results using 2cd PRO model parameter settings | Difference between results w/ 2cd PRO settings and actual dates | PRO results using 3rd PRO model parameter settings | Difference between results w/ 3rd PRO settings and actual dates | PRO results using 4th PRO model parameter settings | Difference between results w/ 4th PRO settings and actual dates |
|---|---|---|---|---|---|---|---|---|---|
| Origin of Model Parameter Settings |  | Dr.Hill's original settings from Rheinhessen |  | from a PRO-GA run using historical data=91,92 |  | from a PRO-GA run using historical data=90,92 |  | from a PRO-GA run using historical data=90,91 |  |
| Prim Inf 1988 |  | 5/3 |  | 5/3 |  | 5/3 |  | 5/3 |  |
| Rec Spray 1988 |  | 12/31 |  | 5/11 |  | 5/11 |  | 5/11 |  |
| Prim Inf 1989 |  | 5/13 |  | 4/19 |  | 4/19 |  | 4/19 |  |
| Rec Spray 1989 |  | 7/3 |  | 5/13 |  | 5/13 |  | 5/13 |  |
| Prim Inf 1990 | 5/24 | 6/5 | 12 | ***5/24*** | ***0*** | *5/24* | *0* | *5/24* | *0* |
| Rec Spray 1990 | 6/7 | 12/31 | 207 | ***6/4*** | ***3*** | *6/6* | *1* | *6/6* | *1* |
| Prim Inf 1991 | 5/11 | 5/11 | 0 | *5/12* | *1* | ***5/12*** | ***1*** | *5/12* | *1* |
| Rec Spray 1991 | 6/7 | 12/31 | 207 | *6/7* | *0* | ***6/17*** | ***10*** | *6/7* | *0* |
| Prim Inf 1992 | 5/5 | 5/23 | 18 | *5/4* | *1* | *5/4* | *1* | ***5/4*** | ***1*** |
| Rec Spray 1992 | 6/1 | 7/10 | 39 | *5/30* | *2* | *5/30* | *2* | ***5/23*** | ***9*** |
| Prim Inf 1993 |  | 4/30 |  | 4/30 |  | 4/30 |  | 4/30 |  |
| Rec Spray 1993 |  | 12/31 |  | 6/23 |  | 6/23 |  | 6/23 |  |
| Avg. Difference |  |  | 80.50 |  | 1.17 |  | 2.50 |  | 2.00 |
| Comments | all dates formated in the form: month/ day | when a date of 12/31 is used, this indicates the model never reached this event | each numerical difference is in absolute days | bold results indicate results from data which was not used in adaptation |  |  |  |  |  |

Table 3. Results from running the P.R.O. model on data from the Trentino region of Italy with parameter settings found by the PRO-GA using three different subsets of the available historical data

between actual results and P.R.O. model produced results using the new parameter settings is only 2.00. This is a lower accuracy than the average difference shown in Table 2 using an adaptation with all three historical data sets, but it is believed still to be quite adequate.

We believe that if an increased amount of historical epidemiological data was available, and this was used in the adaptation process, that the adaptation should become even more robust, increasing the probability that accurate results are produced in the future when the model is running in real-time, giving output values to the agriculturalist for use in making intelligent crop management decision.

## CONCLUSION

In summary, the goal of EU Project SYBIL is to construct a decision support system (DSS) to provide temporal information to assist agriculturalists in the management of crops with respect to controlling fungus and pests. Because the project is particularly interested transporting model technology between countries (i.e., the moving of functional and useful agricultural risk assessment models that are developed in one location to a new location so they can be used in this new location), and these transfers can be problematic, a methodology for performing this transfer in an intelligent manner has been created. This methodology, which is employed in the SYBIL DSS, utilizes four main components, with a genetic algorithm (GA) (or more generally, an intelligent search method) at the center. By employing this *artificial intelligence* (AI) component in conjunction with the engine of an agricultural risk assessment model and historical data, model parameter settings can be adapted to new locations, allowing the model to give accurate results when run in the new location. Specifically, this hybrid methodology (which combines both a GA and a model) can be applied to *localize* models by deriving new model parameter settings that can be employed in the particular location to give good suggestions/decision support.

Currently, this methodology has been applied to an instantiation of the P.R.O. model that is programmed into the SYBIL DSS. This model (which comes from Rheinhessen, Germany) addresses the infection and growth of downy mildew (Plasmopara viticola, also called peronospora) on grape vines, and has the capability to provide information to a farmer so that decisions regarding when to apply fungicides are made more intelligently. Due to difficulties in transporting this model to run in regions outside Rheinhessen, our hybrid methodology to adapt model parameter settings was employed. The testing of new model parameter settings produced by this *adaptation* showed that this methodology has great potential to localize model parameter settings, and this should assist in achieving the goal of making sound models more widely available.

## ACKNOWLEDGEMENTS

Funding has been provided for the SYBIL project (project number PL 900615) by the EU CAMAR Program (Competitiveness in Agriculture). The authors wish to thank Dr. Val Reilly, SYBIL project supervisor from the European Commission, for his support and encouragement.

Furthermore, the authors would like to thank all EU Project SYBIL partners: Technische Universitaet Muenchen-Weihenstephan (Technical University of Munich) (TUM) (D), Bayerischen Staatsministerium für Ernährung, Landwirtschaft und Forsten (BALIS) (D), Association De Coordination Technique Agricole (ACTA) (F), and Danish Institute of Plant and Soil Science (DIPSS) (DK). In particular, thanks go EU Project SYBIL's coordinator Dr. Johann Bergermeier at BALIS in Munich, Germany and Dr. J.V. Herrmann at Bayerische Landesanstalt für Weinbau und Gartenbau in Würzburg, Germany for their assistance with the P.R.O. model.

## REFERENCES

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

Grefenstette, J.J. 1985, ed. Proceedings of an International Conference on Genetic Algorithms and Their Applications. Carnegie-Mellon University, Pittsburgh, PA, USA, July 24-26, 1985. U.S. Navy Center for Applied Research in Artificial Intelligence, USA.

Grefenstette, J.J. 1987, ed. Proceedings of the Second International Conference on Genetic Algorithms. Massachusetts Instituted of Technology, Cambridge, MA, USA, July 28-31, 1987. L. Erlbaum Associates, USA.

Hill, G.K. 1989. Mit P.R.O. System Spritzungen Sparen. Die Weinwirtschaft, 125, pp. 28-31.

Hill, G.K. 1990a. Das Peronospora-Riskoprognosemodell Oppenheim (P.R.O.) im Praxistest. Der Deutsche Weinbau, 44 (12/27 April 1990), pp. 514-517.

Hill, G.K. 1990b. Studies on Plasmopara Viticola Epidemics in the Vineyard. Proc. Atti del Convengo su Modelli Euristici ed Operativi in Agricoltora, Caserta, Italy, 27-29 Sept. 1990, Societa Italiana de Fitoiatria ed., pp. 266-273.

Hill, G.K. 1990c. Plasmopara Risk Oppenheim - A Deterministic Computer Model for the Viticultural Extension. Proc. Atti del Convengo su Modelli Euristici ed Operativi in Agricoltora, Caserta, Italy, 27-29 Sept. 1990, Societa Italiana de Fitoiatria ed., pp. 182-194.

Hill, G.K. 1993a. Severe Primary Infections of Plasmopara Viticola at bloom-Causal Climatic Conditions and Threshold of Damage on Bunches. Proc. OILB Meeting on Integrated Control in Viticulture, Bordeaux, France, 2-5 March, 1993.

Hill, G.K., Breth, K., Spies, S. 1993b. The Application of the P.R.O. - Simulator for Minimizing of Plasmopara Sprays in the Frame of an Integrated Control Project in Rheinhessen/Germany. Vitic. Enol. Sci., 48, pp. 176-183

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

Schaffer, J.D. 1989, ed. Proceedings of the Third International Conference on Genetic Algorithms. George Mason University, Fairfax, VA, USA, June 4-7, 1989. Morgan Kaufmann Publishers, Inc., USA.