Crop Simulation Model Registrator and Polyvariant Analysis

Sergey Medvedev, Alexander Topaj

Crop simulation model registrator & polyvariant analysis

Sergey Medvedev[1], Alexander Topaj[1]

[1]Agrophysical Research Institute. 14, Grazhdansky prospect
195220 Saint-Petersburg, Russia
{ Sergey.Medvedev, glorguin@yandex.ru, Alexander.Topaj, topaj@hotmail.ru }

Abstract. Typical use cases of the crop simulation models consist of such operations as parametric identification, yield forecast and optimization, analysis of different technologies etc. All these tasks relate to multiple running of the model with several variants of input parameters and can be automated by means of polyvariant crop simulation framework which allows performing the multiple running in the batch mode. Such framework has been developed in our laboratory. It makes it possible to register any crop simulation model in the database and to perform several typical operations with it. As a result we present the wide-functional computer system for planning and automation of multi-factor computer experiment with arbitrary dynamic crop models.

Keywords: Framework, information system, database, crop simulation model, decision support systems.

1 Introduction

Crop simulation models play sufficient role in computer farming as an intellectual core of computer decision support systems (DSS) in agroecology. Programmatically, crop model is a complex dynamic algorithm allowing to get some kind of output information (yield, maturing date etc.) for the concrete set of input data (weather, technology, cultivar, soil etc.). This elementary procedure of isolated model computation can be called the single- or one-variant model run. However, typical use case of model application in DSS needs the multiple running of the model with different variants of input parameter set in order to analyze and to compare the corresponding outputs. One can enumerate the problems requiring model multiple running or, other words, multi-variant analysis (table 1).

Table 1. Sources of input data variability in typical crop simulation model related tasks.

#	Task	Sources of input data variability
1	Parametric identification of the model	Variations of model parameters
2	Operative forecast of crop state during vegetation period	Synthetic scenarios of future weather
3	Optimization of agrotechnology	Variations of technology (rates and termini of human interventions)
4	Investigation of climate change influence into agroecosystem	Synthetic weather scenarios corresponding to future climate
5	Precision agriculture	Crop and soil spatial heterogeneity

It would be sane to perform such multi-variant model running in the batch mode in frames of the special system of polyvariant analysis of crop model. Moreover, taking into account the generic character of corresponding operations (forming input sets, butch running, and statistical analysis of the results) the supplementary challenge for the system is an ability to work with arbitrary crop simulation model. The presented contribution describes the prototype of such a system supporting two basic functionalities — the repository of descriptors of external crop models and generic environment for their polyvariant analysis.

2 Extending relational database meta-information

The system to be developed needs to be versatile i.e. widely customized. It has to permit to registry new models and have functionality for supporting them by all necessary input data in corresponding formats. But neither model execution mode, nor input data format nor output data format has to affect common infrastructure of the polyvariant calculation system. Therefore information model of the system has to consist of two levels of data. The lower one constitutes possible inputs of the registered models and their computed results. All such data sets have a structure and format the model can understand. The higher data level is the information about the models and the data structure required by those models. Such kind of "data about data" is called meta-information. The task of proper data and metadata combined management in the frames of single data storage (relational database, for instance) has a long history and many approaches for possible solution. Fortunately, we don't need to solve the problem in a whole scope, i.e. map into database scheme such abstract concepts as "entity" and "relation". Indeed, any crop model operates with data which can be separated into limited number of information domains. These domains (named "factors" in our system) form the solid ground of the system architecture. We consider the following domains: "location", "weather", "crop", "soil", "initial state" and "technology" for input data and the single domain "results" for output data. The main assumption is that any data for any model can be presented as a data set (with custom columns) linked to the record ("factor level") of one of the root model table, corresponding to one of the predefined factors. This elementary data set may consist of only one record (common cultivar parameters) or many records (soil properties for multiple soil levels, cultivar parameters depending on phenological stage etc.) So, the common problem is to combine domain-specific meta-info and relational data in single data storage.

To solve this problem we appealed to the SQL 92 standard. This standard directs database management systems to provide special views displaying database structure called INFORMATION_SCHEMA. The approach used in the polyvariant analysis system is based on the ability to combine these views with user-defined tables and views of the database in queries. So, We created user-defined views, functions and check constraints based on the views from INFORMATION_SCHEMA. These user-defined views and several tables with such check constraints became database objects of the core of our system. These objects were isolated from the tables to be created by

end user via user interface of the system. Such isolation was reached by the set of special technical restrictions (convention about table name prefixes etc.)

This approach is characterized by the following advantages:

- The system works with the physical database tables without performance losses.
- User-defined views with extended metadata are available for database development tools such as Linq2Sql.
- All required constraints, such as foreign keys, can be automatically included into the tables created by the end user.
- User-defined views with extended metadata are based on the efficient standard mechanism.
- The structure of extended metadata is not limited by the abilities of a database-specific tool.
- The system is portable to each database management system supporting INFORMATION_SCHEMA.

3 Model interaction API

The system provides universal algorithms for arbitrary data formats. Its architecture is like the architecture of the classical gateway with the same fundamental concepts: "schema", "adapter" and "scenario". In our system these concepts have the following meaning:

- Schema is a set of tables and table columns defined by the end user for each factor of the model. Each model can have arbitrary subset of the set of factors predefined in the system. For each table user can select one of the columns as an identifier column. When no such column is selected, only one record of the table can be associated with one level of the factor. When such column is selected, several table rows must have unique values of this column within one level of the factor.
- Adapter is a program component implementing the standard interface declared in the polyvariant analysis system to interact with it. The adapter solves three tasks: transforming input data stored in our database into a model native input data format, launching the model with the prepared input data, and transforming results of the model into intermediate format to be saved in our database.
- Scenario is the data set for a single model running. It contains a single factor level for each factor of the model. Each scenario can be considered as a set of control directives, defining what soil, culture, weather etc. is chosen as input data for the referred variant of model running. After the model running the results are attached to the scenario.

Since we use .NET Framework as the main platform, we have made the intermediate format of data based on ADO.NET dataset [1]. The adapter constitutes an interface defined in our common assembly. The main method of this interface in the current version accepts two parameters: a dictionary with special enumeration FactorType as a key and ADO.NET dataset with data for the corresponding factor

type as a value, and a background worker for the progress notification. This method returns single ADO.NET DataTable with model computing results. The structure of these tables is the same as in the model created by the end user in special designer on model registration. The second method is intended to import data from the model native format. It does not return any value and accepts two parameters: the dictionary of factor types and datasets to fill, and a background worker. For the future we are going to provide the full registrator runtime interaction with a model based on OpenMI standard [2].

The system supports plug-ins for working with adapters. All assemblies in the deployment directory are loaded automatically and analyzed for all implementations of this interface. After that all found implementations are presented in menu of the "adapter panel". Additionally, the system provides a few off-the-shelf bricks to build custom adapters without writing any code. Default implementation of model adapter contains three auxiliary interfaces: converting data into model native format, launching model and converting model calculation results into the intermediate format. The system supports Microsoft Excel and CSV data format and launching model via unmanaged function call. We are going to work on the GUICS adapter to support all GUICS-compliant models [3].

4 Automated computer experiment with crop model.

## 4.1 Operation principles and implementation

We describe below the general (not depending on model-specific data) part of information model of our system. It directly relates to main system functionality – planning and performing of multi-variant computer experiment with crop model. The principal concepts of this part are "Project" and "Scenario". The "Project" is the common description of the single experiment we perform. In the current version, we suppose that the experiment takes place with only one crop model, i.e. we can not use different models in frames of one experiment. Project consists of several scenarios; each of them corresponds to one future model running.

So, the procedure of preparing computer experiment or multi-variant crop model analysis contains two steps: forming all interested "factor levels" and their coupling in the project. Consider them by turn.

There are several ways to receive the data composing the "factor level" in order to store it in the system database. Firstly, one can get the data earlier used for model computation from "native" model sources via adapter. Usually, it would be "real" data, i.e. actual field measurements or parameters of real soils or cultures. Secondly, the system provides different mechanisms for data varying in order to generate data sets for "synthetic" levels of this or that factor. The simplest algorithm here is an examination of the parameter values in selected interval with constant step. As a result we can obtain the row of "factor levels" differ each other only in the value of chosen parameters. Additionally, system has built-in generator of weather realizations

based on the algorithm proposed by Richardson and Wright [4]. It allows forming synthetic data for the factor "weather" that corresponds to the climate conditions of any selected location. At last, data can be imported from SCV-files or entered manually via the user interface.

The final stage of computer experiment planning is a design of computation variants in terms of project (set of scenarios). This procedure is rather trivial. The set of scenarios is a Cartesian product of all levels chosen for every factor registered for the model. Principle of project forming is demonstrated on screenshots presented in Fig.1.
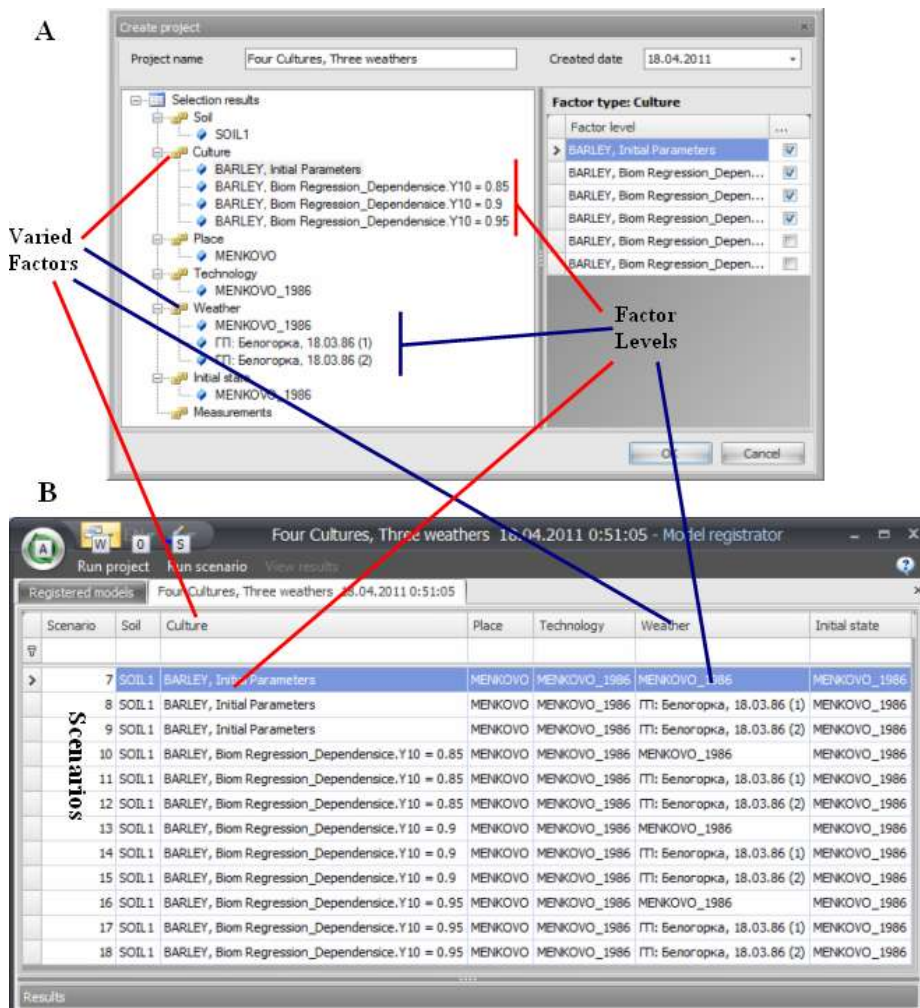
Fig. 1. . Principles of multi-variant computer experiment design in "Crop Model Registrator" (A – Project Creation Dialog, B – Project Viewer).

The project being once created as a set of scenarios or running variants can be "calculated" at any time. It means the serial model running (via adapter) with input data sets determined by current scenario in the project and saving the corresponding results to system databases. After project has been calculated its results can be analyzed by means of one of special tools built in the system. For the moment the following tools for analysis of multi-variant computer experiment has been implemented:

- Two-dimensional graphic visualization of model dynamics. Diagrams can be presented in two regimes – data series are different characteristics of interest for one scenario, or data series are dynamic of one characteristic of interest in different scenarios.
- Analysis of one-factor experiment with many characteristics of interest in tabled form
- Analysis of two-factor experiment with one characteristic of interest in tabled from. Let's note one of the factors for the case can be interpreted as replication.
- Variance analysis for complete factorial in tabled form.

4.1 Examples of usage

We use the system described above for the solution of different tasks in agroecological practice. The main model we use for the moment is dynamic crop model "AGROTOOL" elaborated in the Laboratory of Agroecosystem Simulation, Agrophysical Research Institute, Saint-Petersburg, Russia [5]. For example, the system has been applied to the problem of estimation of the influence of global climate change into agroecosystem stability and productivity in North-West Russian regions. The main idea there was to use the standard weather generator to get the scenarios of "the weather of the future". Indeed, mathematical generation algorithm uses the parameters that have been calibrated from actual weather observations and are the characteristics of the modern climate for the chosen location. So, we can change such parameters according to the forecasts or models of future climate change and interpret the results of generation as weather samples for the future climate state. So, the problem of climate change impact into crop production process adds up to one-factor computer experiment with crop models. Let's note, that weather generator allows having so many tested weather samples as it is necessary for representative statistical conclusions.

Next problem, we solved with a help of the system was the search of economically optimal irrigation strategy for the arid regions in Middle Volga. The task was to determine the most economically efficient parameters of the automatic watering system (i.e. critical soil water content to start watering and the watering rate). Optimization criterion for the case is proceeds from a yield sale minus technological losses (cost of water etc.) The only tool to estimate yield value for different watering strategy under current environment conditions can be dynamic crop model. So, the problem adds up to two-factor computer experiment where first factor would be the

watering strategy and the second one (interpreted as replication) is weather. Statistical analysis of corresponding experiment allowed us to give the sound recommendation to the choice of optimal irrigation politics in regional scale.

5 Perspectives.

The main challenge of further system development seems to be its evolution to parallel computing. Really, project calculation for the moment is independent but serial runs of the model with different sets of input data. The task is to provide the possibility of parallel model running in several computation threads or processes. It is not the efficiency problem only. One can show that parallel run of several instances of the standard one-dimensional dynamic crop model in synchronized regime with temporal interactions can be sufficient emulation of the processes taking place in spatially heterogeneous ecosystems. So, involving the methods of parallel (and, may be, cloud) computing in the system of multi-variant crop model analysis can increase significantly its possible value for the purposes of precision farming, that is the mainstream of applied researches in modern agriculture [6].

References

1. Microsoft Development Network, http://msdn.microsoft.com/en-us/library/e80y5yhx.aspx
2. Moore, R.V and Tindall,I., (2005). An Overview of the Open Modelling Interface and Environment (the OpenMI). Environmental Science & Policy 8, 279-286 (2005)
3. Acock, B., Ya. A. Pachepsky, E.V. Mironenko, F. D. Whisler, and V. R. Reddy. GUICS: A Generic User Interface for On-Farm Crop Simulations. Agronomy Journal, 91:657-665 (1999)
4. Richardson, C. W., Wright, D. A. WGEN: A Model for Generating Daily Weather Variables. In: U. S. Department of Agriculture, Agricultural Research Service, ARS-8, 83 p. (1984)
5. Poluektov R.A., Fintushal S.M., Oparina I.V., Shatskikh D.V., Terleev V.V., Zakharova E. T. AGROTOOL – a system of crop simulation. Archives of Agronomy and Soil Science, 48: 609-635 (2002)
6. Poluektov, R.A., Topaj, A.G., Kobylanski, S.G. System of polyvariant computation of dynamic crop model in precision agriculture applications. Reports of Russian Agricultural Academy, 6, P. 58-62 (2005)