LIDeOGraM : an interactive evolutionary modelling tool

Thomas Chabin, Marc Barnabe, Nadia Boukhelifa, Fernanda Fonseca, Alberto Tonda, Helene Velly, Benjamin Lemaitre, Nathalie Perrot, Evelyne Lutton

LIDeOGraM : An interactive evolutionary modelling tool.

Thomas Chabin, Marc Barnabé, Nadia Boukhelifa, Fernanda Fonseca, Alberto Tonda, Hélène Velly, Benjamin Lemaitre, Nathalie Perrot, Evelyne Lutton

UMR 782 GMPA, Agroparistech, INRA, Université Paris-Saclay
1 Av. Lucien Brétignères, 78850 Thiverval-Grignon, FRANCE

Abstract. Building complex models from available data is a challenge in many domains, and in particular in food science. Numerical data are often not enough structured, or simply not enough to elucidate complex structures : human choices have thus a major impact at various levels (data and model structuration, choice of representative scales, parameter ranges, uncertainty assessment and management, expert knowledge). LIDeOGraM is an interactive modelling framework adapted to cases where numerical data and expert knowledge have to be combined for building an efficient model. Exploiting both stand-alone evolutionary search and visual interaction with the user, the proposed methodology aims at obtaining an accurate global model for the system, balancing expert knowledge with information automatically extracted from available data. The presented framework is tested on a real-world case study from food science : the production and stabilisation of lactic acid bacteria, which has several important practical applications, ranging from assessing the efficacy of new industrial methods, to proposing alternative sustainable systems of food production.

Keywords: Complex systems, Lactic acid bacteria, Interactive modelling, Symbolic regression, Living food system

1 Introduction

Agri-food processes can be regarded as complex systems, as they are characterised by uncertain and intricate interaction effects between physical, chemical, and biological components [10] [11].

In this context, modelling techniques drawn from complexity science prove especially advantageous for dealing with the co-existhing multiscale inter-dependencies, uncertainty, partial knowledge and sparse experimental data.

Expert knowledge yield additional, precious information. [14] [1] Indeed, building a model in these conditions is a complex optimisation: learning from data sets, dealing with sparsity of data, possible overfitting issues, and complexity of the models. In the same time experts knowledge can drastically modify the shape of the search space, the relative impact of some data, or even the optimisation aims.

In this paper, we propose an interactive modelling approach based on a two-level evolutionary optimisation scheme, *local* and *global*. *Local* corresponds to local possible dependencies between variables, while *global* corresponds to a model that represents the system as a whole. Users can interact with the constructed models via a graphical user interface, run various optimisation steps, revisit optimisation results, restart the process, add constraints, and take decisions.

The system and dataset considered in this study concerns the full process of bacteria production and stabilisation, with 49 variables measured at 4 different steps (fermentation, concentration, freeze-drying and storage), at 4 different fermentation conditions (22°C and 30°C, evaluated at the beginning of the stationary growth phase and 6 hours later). The considered variables range from transcriptomic data to fatty acid membrane composition, from acidification activity to viability [15].

The paper is organised as follows: Section 2 provides background on complex systems approaches in food science, on symbolic regression and on the modelled process. *LIDeoGraM* is detailled in section 3. Section 4 describes experimental results from a preliminary user evaluation, comments, conclusions and future developments are given in sections 5 and 6.

2 Background

2.1 Food complex systems

A complex system[1] is a collection of multiple processes, entities, or nested subsystems, where global properties emerge as the result of an imbrication of phenomena occurring at different scales. For these systems, there is a need to appropriate descriptions for the underlying mechanisms with high expressiveness and little uncertainty. Building complex system models is essential, but highly difficult; it is usually necessary to have a robust framework, with strong iterative interaction combining computational intensive methods, formal reasoning and experts from different fields. As shown in the rest of the paper, optimisation plays an important part in this context[8].

The specifics of the food domain (uncertainty and variability, heterogeneity of data, coexistence of qualitative and quantitative information, conjunction of different perspectives) raise the focus on another crucial issue, that can be called the *human factor*. Human expertise and decision making are of major importance for a better understanding of food systems, and should thus be integrated in computing approaches[7].

2.2 Symbolic regression

Symbolic regression, hereafter based on a genetic programming approach, is a technique able to extract free-form equations that expose correlations in a given

[1] Complex Systems Society, see `http://cssociety.org` or `http://www.mathinfo.inra.fr/en/community/complexsystems/presentation` for an introduction to the topic.

experimental dataset. The original idea is presented in [6], and the technique has been applied to a vast array of real-world problems [2,12,5]. Candidate solutions are encoded as trees, with terminal nodes encoding constants and variables of the problem, while intermediate nodes corresponding to mathematical functions such as $\{+, -, *, /, ...\}$. In most implementations, the fitness function is proportional to the absolute or squared error between experimental data, with parsimony corrections to reward simpler solutions. *Eureqa Formulize*[2] is one of the most notable symbolic regression tools. Eureqa deals with the issue of overfitting by returning a Pareto front of candidate solutions, each one presenting a compromise between fitting and complexity [13], leaving the final choice to the user.

2.3 Production and stabilisation process of lactic acid bacteria

Concentrates of Lactic Acid Bacteria (LAB) are widely used in food applications, ranging from yoghurt and cheese to fermented meat, from vegetables to fruit beverages. In industry, these bacterial starters are produced in large quantities by fermentation and must therefore undergo a preservation procedure, called stabilisation. Both production and stabilisation processes aim at protecting the quality of bacterial starters, characterised by their cell viability and their acidification activity. The full process involves numerous control parameters across its different steps (Fermentation, Concentration, Freeze-Drying and Storage). [3] Moreover, the process is a multi-scale system. Indeed, the quality of the starters can be explained by the cellular composition in fatty acid which is in turn explained by the genomic expression in each cell. This latter only depends on the parameters of fermentation and concentration.

3 Proposed approach

Experts in the process of production and stabilisation of lactic acid bacteria have numerous questions about how a given bacteria strain draws its resistance to the process. Different mathematical tools, including mathematical formulas are generally used to help them to answer these questions with more or less success. Finding reliable formulas linking the different variables of such a system is indeed challenging[9]: In biological data, a high level of variability is often encountered for repetitions of a given experimental condition. Moreover, experiments are usually time-consuming and expensive – only a few experiments are thus performed – which makes the task of characterising the existing variability difficult.

LIDeOGraM (***L****ife-based* ***I****nteractive* ***De****velopment* ***O****f* ***Gra****phical* ***M****odels*) tries answering these challenges with an original approach of semi-automatic modelling.

The goal of LIDeOGraM is to help experts build a global model of their complex process by characterising each non-input variable by a mathematical

[2] `http://www.nutonian.com/`

formula that depends on the other variables in the system. Finding the right equation in a context with high variability in the dataset is an ambitious task. Indeed, it is easy to come up with over-fitted equations that perfectly model a dataset including its noise. However over-fitted equations do not generalise well.

In order to rule out over-fitted equations, a solution is to involve experts in the course of the modelling process. The expectation is that they will be able, thanks to their knowledge of the process, to identify over-fitted or under-fitted equations.

Symbolic regression using a Pareto-like approach such as the one implemented in Eureqa, constitues a compelling approach to take advantage of the expert's insight. Indeed, by providing a set of formulas according to different compromises between fitness and complexity, the approach allows the experts to filter out incoherent equations or even designate the most suitable one.

Therefore, as a first optimisation step, LIDeOGraM uses Eureqa runs on each variable, in order to get a set of candidate equations. For automatic learning purposes, the dataset is separated into training and test sets. Moreover, some constraints in the search are defined beforehand by the user, using the interface presented in Figure 2. This tool allows attributing each variable to a given class, and defining authorized links between them. This means that only the variables from a parent class can be used in the equations for determining the variables of the child class. This also means that dependencies will be searched only with variables of other classes and that no intra-class dependencies will be considered. This structure of classes can be used to distinguish between scales and steps in the studied process. Variables measured at a macro-scale, like the viability of the population of bacteria could, for example, be only explained with variables from a micro-scale, such as the composition in fatty acids. Similarly variables measured in a given step could only be explained by variables from previous steps.

A qualitative view of these results is presented to the user in the form of a graphical network (See Figure 1). The goal of this display is to help the user focus on the critical variables, i.e. where expert feedback is most needed. In this prospect, variables are represented as nodes in the graph. The colour of the nodes depends on its attributed class. A link between two variables shows that the parent node is used at least once in the set of equations attributed to the child node. The colour of a link represents the mean value of a given criteria on the equations involving the parent node in the child node. The criteria can be chosen by the user as the fitness of the equation or its complexity.

Additionally, since the displayed graphical network can have a lot of links, making the network hard to read, a slider allows the filtering of links based on their level of importance. The importance of a link is defined by the number of equations in the child node that uses the parent node, divided by the total number of equations in the child node.

By clicking on a node, the equations found by Eureqa are displayed to the user on the top-right side (See Figure 1). Similarly, a click on an equation provides a plot of the experimental measures versus what is predicted by the corresponding

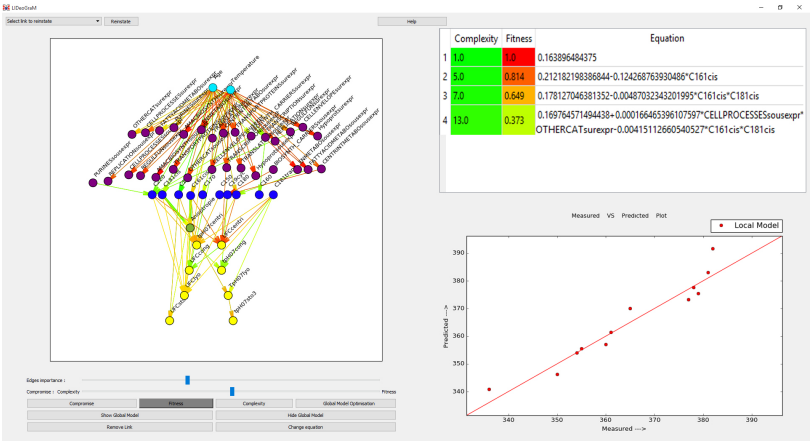

Fig. 1. Screenshot of LIDeOGraM. The left side shows a graphical model representing the mean fitness of the local models obtained by symbolic regression. The top-right part is the list of equations proposed by Eureqa for the selected node, and the bottom-right part shows a plot of the measured versus predicted data associated to the selected equation.

equation. The user can then interact with the system by deleting an equation, deleting a link between a parent node and a child node (i.e. all equations using the parent node in the child node are deleted), or deleting a variable (i.e. all equations using the deleted variable are deleted). After this, few or no equations may remain for some nodes, the user can choose to restart a symbolic regression on any node.

The user can iterate the process for as long as desired: add or suppress constraints, restart symbolic regression on any node. Once the user is satisfied with local models, a global model can be built.

For the global model, one equation only is kept for each node. However, choosing the most reliable ones is a challenging task. Contrarily to the local models, where the experimental measures are used to predict a variable, in a global model the value predicted by an equation depends on the value predicted for the variables used in that equation. For this reason, each choice of equation for a given variable will influence the quality of the prediction of other equations that uses the variable. To tackle this challenge, evolutionary optimisation is used to build a global model.

The evolutionary algorithm has the following features. Its fitness function, to be minimised for the global model, is the mean of the fitness calculated on each non-input nodes. The fitness function of a single node computes a value based on the Pearson correlation coefficient of the measured versus predicted data. Such a fitness function does not take into account the complexity of the equations. The reason behind this choice is that over-fitted equations will be

naturally discarded during the learning of the global model, as they will likely create noise for their child variables.

After the evolutionary optimisation process, remaining incoherence in the choice of equations for the global model can still be edited by the user. For this purpose, a qualitative view of the global model is displayed as a new visualisation (See Figure 4). Contrarily to the graphical network displayed for the local models, here only the variables used in the chosen equation of a child node are considered as parents of the node. A link from a parent node to a child node in this graph represents the fact that the chosen equation for the child node contain the parent variable, its colour depends on the fitness of the child node in the global model. Green represents a good fitting, and red a bad one. The goal of this view is to help the user focus on the nodes with bad predictions. A user change on the selected equation of a node impacts the predicted value of its children nodes. The graphical model is therefore automatically updated, and the update shows the consequence of the change in term of fitness on the other nodes of the graph.

After this step, the user has the possibility to go back to local view and make changes before restarting a new global optimisation. A global model is thus iteratively built via user interaction, local and global optimisation.

4 Experimental results

4.1 The dataset

The case study is based on the work of H. Velly et al. [15][16] about the resistance of *Lactococcus lactis* subsp. *lactis TOMSC161* to freeze-drying. This bacteria is used in the production of *Tomme de Savoie*, a french cheese, for its interesting texturing and acidification properties, but exhibits a high sensitivity to freeze-drying. The resistance of the bacteria is studied for 4 different conditions of fermentation: 22°C and 30°C, evaluated at the beginning of the stationary growth phase and 6 hours later.

The dataset featured 12 data points, with 3 biological repetitions of each experimental condition. The dataset is made of 2 input variables, the temperature of fermentation and the time at which the fermentation is stopped and 49 variables measured at 4 different steps (fermentation, concentration, freeze-drying and storage) for 3 biological scales (Genomic, Cellular and Population).

4.2 Search with Eureqa

The 51 variables described above are first separated into 9 classes of variables: **Inputs**, **Genomic** for **overexpressed** and **underexpressed** genes, **Cellular**, **Anisotropy**, **Population** at the end of the **Concentration** step, **Population** at the end of the **Congelation**, **Population** at the end of the **Drying** step and the **Population** after 3 months of **Storage**. Each class of variables can only be explained by user specified classes. The possible links between classes are shown in Figure 2.

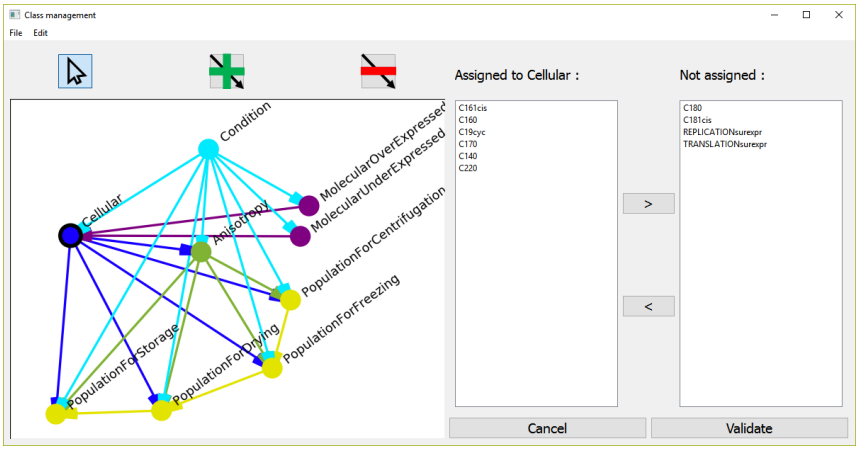

Fig. 2. Screenshot of the interface allowing to choose the authorised links between the defined classes. A link between two classes means that all variables associated to the parent class can be used in the equations for all variables associated to the child class. The displayed graph represents the selected contraints chosed for this experiment.

The dataset is also separated into a training dataset(66%) made of two out of the three repetitions for each experimental condition, and into a test dataset(33%) with the remaining repetition.

The authorised mathematical operators for the Symbolic regression using Eureqa are: Constants, Input variables, Addition, Subtraction, Multiplication, Division, Exponential, and the Natural logarithm. For each non-input variables, 3 minutes of computation were allowed on an Intel(R) Core(TM) i7-4790 CPU. A total of 232 equations were obtained for the variables.

4.3 Optimisation of the Global Model

A $(\mu + \lambda)$-evolutionary algorithm was taken from the Python DEAP package[4] to optimise the global model. The genome of a candidate global model is string of integers, of size equal to the number of variables in the process. Each gene is associated to a variable, and can assume a value between 1 and the number of equations available to describe that variable, thus representing an index for a candidate equation in that node.

The parameters of the evolutionary optimization algorithm used for the global model are reported in Table 1.

The mutation function takes complexity information into account. It has been experimentally proven to be more efficient than a mutation which randomply picks an equation in the list of candidate equations.

The graphical model associated to one of the optimisation runs is shown in Figure 4.

Table 1. Parameters of the evolutionary algorithm used during the optimization process for the global model.

μ	100
λ	80
Number of generations	100
Probability of crossover	0.8
Probability of mutation	0.2
Selection	Tournament of size 2
Crossover function	Uniform
Mutation function	With a probability 0.05 for each gene, change the selected equation to the previous or the next one by order of complexity.

The creation of a global model does not involve only an automatic optimisation, but also requires experts knowledge, obtained via interaction with the software. Feedback on the proposed local models was given by a researcher with 20-years of expertise in the bacteria freeze-drying process. The local models, presented in Figure 1, were explored by the expert during 20 minutes. The expert chose to remove 5 equations. Some equations were removed for using both variables from the Cellular scale with the Anisotropy variable. The reason is that the Anisotropy is an emergent property of the fatty acid composition at the Cellular scale and it is not straightforward to make sense of such an equation. Similarly, an equation using the viability at both the centrifugation step and the drying step was removed. The reason is that the viability at the centrifugation step is used to predict the viability at the drying step, therefore, it is hard to understand the necessity of using both steps since obtained the data values are dependent. The expert also chose to remove 2 nodes, after observing that those nodes were strongly involved in many nodes. Indeed, due to their insignificant measured quantities, they were not expected to be important variables, rather, they were deemed useful for refining some models. Therefore, they were considered as creators of overfitted equations. The deletion of those two variables removed 14 more equations. With such major deletions, some variables were left with only a few equations, therefore, the expert chose to restart a symbolic regression on 3 nodes, obtaining 12 new equations in total. To reveal the contribution of the expert, the global model optimisation was performed 10 times using expertise, and 10 times without. The fitness evolution of these runs are shown in Figure 3. To obtain an accurate comparison of the models, the fitness computed for optimisation without the expertise did not take into account the two removed nodes. The global models obtained using expertise have a median fitness of 0.787 with a standard deviation of 0.010 whereas the global models obtained without expertise have a median fitness of 0.801 with a standard deviation of 0.013. The expert was asked to provide feedback for the last step of the modelling process in which one of the global model obtained was submitted to his expertise. The results were explored during 10 minutes, and the equations

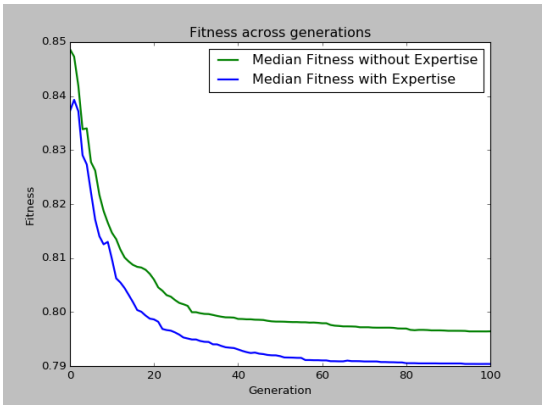

Fig. 3. Comparison of the evolution of the minimum fitness across generations for 10 runs, with and without the expert's contribution.

for three node were changed. Two of the modified equations were indeed overfitted, and the last one was an underfitted. For example, one of the equations selected by LIDeOGraM, at the cellular scale, the variable C18:0 was defined as being equal to the duration of fermentation, which seemed a rather drastic choice. The expert chose to select a more reasonable equation presenting a linear dependency involving the duration of fermentation. The obtained graphical model is presented in Figure 4. The fitness of the final global model was slightly degraded, changing from a fitness of 0.789 to a fitness of 0.801, but the produced model is able to better reflect the underlying reality of the process.

5 Discussion

We proposed a time-saving tool of modelling for the experts, allowing them to design a better global model of their process by a semi-interactive approach. Figure 3 shows that the resulting models are "better", not only according to the expert requirements, but also with respect to the numerical data (faster and better convergence). Above all, this method offers tools for domain experts to design and test different hypothesis, using different datasets and class constraints. The complexity of the modelled process and the scarcity of the dataset is taken into account by allowing the expert to interact with the results all along the optimisation process. The expert who tested the software mentioned it is easy to question her hypothesis, to keep an open-mind posture and to find new mechanisms.

Nevertheless, the approach has some drawbacks. Since the predictions of each node are propagated, only the inputs are indeed used in a global model to determine every other variables. Knowing this, a natural question should be why all variables are not directly linked to the inputs, and why intermediary variables exist. A reason behind this is that the goal is not only to get the best prediction

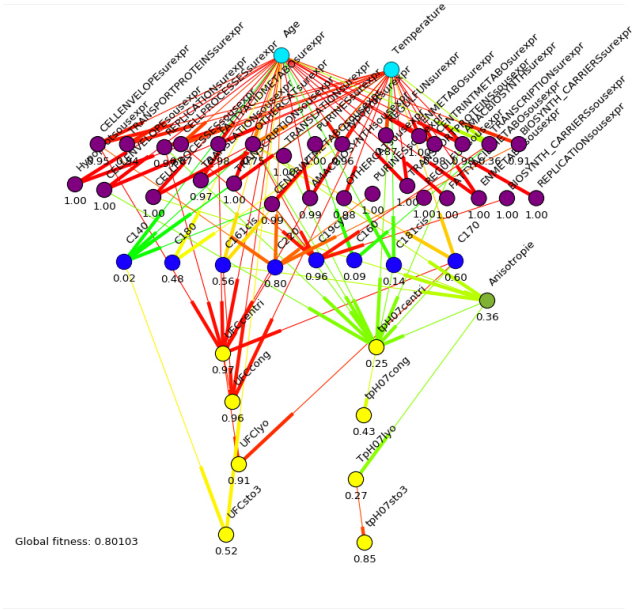

Fig. 4. Graphical model representing a global model. The Global fitness of the model is indicated at the bottom-left. The fitness value of each node is indicated under them.

out of every variable, but also to help the expert understand mathematically the existing dependencies between variables and the multi-scale / multi-step organisation of the process.

Besides this, we should mention that the current results remain not fully satisfying for the genomic scale. The hypothesis made by the expert was that the genomic scale is only explained by the conditions of fermentation (temperature and time at which the fermentation is stopped). This hypothesis needs additional verifications, as the relation with genomic scale might not be so straightforward. Other variables, not measured during the experiments, may be involved. A more refined work on the expressed genes and their classification is necessary. A future study will explore in more details different hypothesis about the possible links between classes of variables.

Finally, some expert-defined variables used in the literature are the sum of some measured variables. For example, the Saturated Fatty Acids variable, is defined as the sum of 6 variables at the cellular scale. New tools could be designed to incorporate this kind of knowledge and allow the user to create "hierarchic" variables. Such variable would allow taking into account different levels of details in the modeling process and would allow to easily test various hypotheses for the computation of variables at the genomic scale.

Finally, creating a global model from local hypothesis means the creation of a global hypothesis. With some local hypothesis being thought of as equally

plausible for a node, a lot of equally plausible global hypotheses could be constructed. Moreover, since the proposed local models of different nodes are not equivalent, a notion of confidence could be associated to each node of a global hypothesis, and each node would be associated with a set of equally probable local models.

6 Conclusions

In this paper, we proposed a new approach to semi-automatic modelling allowing to design complex models for multi-scales and multi-steps processes. Using expert's knowledge integrated during the optimisation process, the proposed approach is able to tackle challenges such as scarcity in a dataset, high dimensionality and high variability. According to experts guidelines, a set of local models are proposed for each variable, using symbolic regression. The local models form a Pareto front of candidate solutions to compromise between model fitness and complexity. These local models are then used to automatically construct a global model where each variable is defined by a given equation from the local models. In a global model, the multi-scales multi-steps process is taken into account by classifying the variables into different classes and by forwarding the predicted value of a variable to equations that use this variable to predict other ones. An expert is able to contribute to the automatic design of a global model in many ways, by acting on the proposed local models and by correcting the global model. The approach was applied to the production and stabilisation process of lactic acid bacteria. The contribution of the expert was shown to be useful to provide a more accurate global model. Future improvements will involve new tools to create and manage hierarchic variables and associate a level of confidence for each variable. These improvements will allow producing a full and efficient study of the production and stabilisation process of lactic acid bacteria.

Aknowledgements

We would like to express our thanks to Jean-Daniel Fekete from Inria Saclay, who provided great advice and insight on the graphical user interface of LIDeOGraM.

References

1. Allais, I., Perrot, N., Curt, C., Trystram, G.: Modelling the operator know-how to control sensory quality in traditional processes. Journal of food engineering 83(2), 156–166 (2007)
2. Babovic, V., Keijzer, M.: An evolutionary approach to knowledge induction: Genetic programming in hydraulic engineering. In: Proceedings of the World Water & Environmental Resources Congress
3. Champagne, C., Gardner, N., Brochu, E., Beaulieu, Y.: freeze-drying of lactic acid bacteria. a review. Canadian Institute of Food Science and Technology journal: Journal de l'Institut canadien de science et technologie alimentaire (1991)

4. Fortin, F.A., De Rainville, F.M., Gardner, M.A., Parizeau, M., Gagné, C.: DEAP: Evolutionary algorithms made easy. Journal of Machine Learning Research 13, 2171–2175 (jul 2012)
5. Gaucel, S., Keijzer, M., Lutton, E., Tonda, A.: Learning dynamical systems using standard symbolic regression. In: European Conference on Genetic Programming. pp. 25–36. Springer (2014)
6. Koza, J.R.: Genetic programming: on the programming of computers by means of natural selection, vol. 1. MIT press (1992)
7. Lutton, E., Perrot, N.: Complex systems in food science: Human factor issues. 6th International Symposium on Delivery of Functionality in Complex Food Systems Physically-Inspired Approaches from the Nanoscale to the Microscale. (2015)
8. Lutton, E., Perrot, N., Tonda, A.: Evolutionary Algorithms for Food Science and Technology. John Wiley & Sons (2016)
9. Passot, S., Fonseca, F., Cenard, S., Douania, I., Trelea, I.C.: Quality degradation of lactic acid bacteria during the freeze drying process: Experimental study and mathematical modelling (2011)
10. Perrot, N., De Vries, H., Lutton, E., Van Mil, H.G., Donner, M., Tonda, A., Martin, S., Alvarez, I., Bourgine, P., Van Der Linden, E., et al.: Some remarks on computational approaches towards sustainable complex agri-food systems. Trends in Food Science & Technology 48, 88–101 (2016)
11. Perrot, N., Trelea, I.C., Baudrit, C., Trystram, G., Bourgine, P.: Modelling and analysis of complex food systems: state of the art and new trends. Trends in Food Science & Technology 22(6), 304–314 (2011)
12. Pickardt, C.W., Hildebrandt, T., Branke, J., Heger, J., Scholz-Reiter, B.: Evolutionary generation of dispatching rule sets for complex dynamic scheduling problems. International Journal of Production Economics 145(1), 67–77 (2013)
13. Schmidt, M., Lipson, H.: Distilling free-form natural laws from experimental data. Science 324(5923), 81–85 (2009)
14. Sicard, M., Baudrit, C., Leclerc-Perlat, M., Wuillemin, P.H., Perrot, N.: Expert knowledge integration to model complex food processes. application on the camembert cheese ripening process. Expert Systems with Applications 38(9), 11804–11812 (2011)
15. Velly, H., Bouix, M., Passot, S., Penicaud, C., Beinsteiner, H., Ghorbal, S., Lieben, P., Fonseca, F.: Cyclopropanation of unsaturated fatty acids and membrane rigidification improve the freeze-drying resistance of lactococcus lactis subsp. lactis tomsc161. Applied microbiology and biotechnology 99(2), 907–918 (2015)
16. Velly, H., Fonseca, F., Passot, S., Delacroix-Buchet, A., Bouix, M.: Cell growth and resistance of lactococcus lactis subsp. lactis tomsc161 following freezing, drying and freeze-dried storage are differentially affected by fermentation conditions. Journal of applied microbiology 117(3), 729–740 (2014)