



“Gheorghe Asachi” Technical University of Iasi, Romania



## ENTOSIM, AN INSECTS LIFE CYCLE SIMULATOR ENCLOSING MULTIPLE MODELS IN A DOCKER CONTAINER

Luca Rossini\*, Stefano Speranza, Angelo Mazzaglia, Silvia Turco

Università degli Studi della Tuscia, Dipartimento di Scienze Agrarie e Forestali (DAFNE),  
Via San Camillo de Lellis snc, 01100 Viterbo, Italy

### Abstract

*EntoSim* is a software aimed to collect different physiologically based models describing the life cycle of ectotherms and, more specifically, of insect pests. It is developed in an open-source framework including the ROOT's software libraries in a C/C++ programming language, with the purpose of an easy diffusion and compatibility. *EntoSim* could be used in model development and validation and in decision making frameworks, to support farmers and technicians to formulate low-impact control strategies. The growing availability, in literature, of physiologically based models and their utilisation in plant protection, led to explore the latest computer technologies that could be applied in entomology. This led to a significant revision of the *EntoSim* philosophy and structure. This work, therefore, aims to describe this conceptual and operational change, and its implications on *EntoSim* in order to: *i*) increase the number of options available to the users, *ii*) simplify the software development and *iii*) increase software dissemination and adoption. These three points led to the addition of two more population density models within *EntoSim*, and to the creation of the first Docker image available in a public repository for entomological purposes, to make its deployment faster. In this regard, the code structure was reorganised in modules corresponding to each *EntoSim* function; the graphical part was developed and implemented using Python language so that results are visualized directly in a web browser page. All the *EntoSim* dependencies were included into a Docker image, using Linux CentOS as base operating system.

**Key words:** decision support systems, descriptive models, growth models, integrated pest management, population modelling

Received: January, 2021; Revised final: April, 2021; Accepted: September, 2021; Published in final edited form: October, 2021

### 1. Introduction

Understanding biological phenomena is surely one of the greatest challenges of the modern times, and among the developments succeeded towards the years, mathematical modelling has always had an important role. In the fields of Agriculture, Forest and Ecological Sciences, for example, mathematical models aroused the interest of researchers since their suitability to describe phenomena, and to be used as Decision Support Systems (DSS) (Capinera, 2001; Rossini et al., 2020e; Tang and Cheke, 2008). More specifically, if a mathematical model faithfully represents a particular phenomenon, such as pests outbreaks, it could provide a prediction of the risks of future infestations, on the basis of the past and present data

(Sinclair and Seligman, 1996). The main reason to invest human and economic resources on the development of mathematical models is their role in the reduction of the inputs (such as pesticides or agrochemicals) in the cultivated fields, with a subsequent increase of the biodiversity and of the environmental health (Bange et al., 2004; Knight and Mumford, 1994).

To protect plants from insect pest infestations, several mathematical models were developed applying different theories and hypotheses (Orlandini et al., 2018), but their common point is the use of computational tools providing numerical solutions and parameters from experimental data (Holst, 2013). Generally speaking, an ideal application of a mathematical model in pest population description is

\* Author to whom all correspondence should be addressed: e-mail: [luca.rossini@unitus.it](mailto:luca.rossini@unitus.it)

composed by three “macro-steps” (Bellocchi et al., 2011; Rossini et al., 2019a, 2019b): *i*) a laboratory phase, where parameters specific for the species are estimated with experimental data; *ii*) a simulation phase, where the parameters of the previous phase and environmental data are inserted into the population density model, subsequently solved numerically; *iii*) a validation phase, where simulations are compared with data collected *ad hoc* in an experimental field.

To date there are several software programs which may support the operations reported so far, but most of them need an adaptation or a license. A recent work by Rossini et al. (2019a) discussed the possibility to use ROOT (Brun and Rademakers, 1997), an open source software developed for high energy physics data analysis, to support the model application in entomology. More specifically, ROOT's libraries are particularly suitable for the fitting operations, which lead to the estimation of the model parameters in the so called “laboratory phase”, and to graphically represent the results. In their initial idea, the description of a series of macros, corresponding to each step of a model application, was provided, showing a great potential in terms of versatility (Rossini et al., 2020d) and reduction of software costs.

However, biological and mathematical modelling are part of a greater field of data analysis which is constantly trying to solve problems of results reproducibility, software and scripts compatibility among different operating systems or software licence requirements. Moreover, this growing field of science needs highly skilled users that not all the research groups are equipped with. This aspect is particularly highlighted by the increasing availability of data, software and scripts provided as supporting information with the scientific articles.

Recently, scientist's attention has been captured by Docker, an open-source project developed with the aim of deliver pre-assembled packages called image containers, that will be run in a virtualization mode (Docker, 2021). Thus, it is possible to build a Docker image container running a basic operating system and already containing all the dependencies required by the software that the developers are going to share. Furthermore, along with the software, the datasets necessary for the analyses could be also provided, so that anyone can use them to reproduce the same results, even without advanced programming skills.

Users can then easily download and use right away this virtual container, without encountering all the problems of incompatibility and reproducibility, as discussed by Di Tommaso et al. (2015) and Boettiger (2015). Furthermore, Docker is even more appealing for scientists because: *i*) it is less memory and hardware consuming than a virtual machine, and *ii*) it is also less time consuming because most of the dependencies and installation operations are already done by the developers.

Given this precondition, the main focus of this work was to improve and share *EntoSim* among model scientists who work with ectotherms, providing the community of users with the latest code and model

structure modifications in a simple deployable computing environment. More specifically, the first major change was the creation of a Docker image, where *EntoSim* has been stored with all its dependencies, which can be easily “pulled” from a public repository and executed as a stand-alone Docker container. In the best of our knowledge, Docker has never been applied before for entomological purposes, and it makes this short communication pioneering such as Rossini et al. (2019a), where the use of ROOT outside its native field of application was discussed for the first time.

The second major change concerned the graphical representation of the simulations, which has been entrusted to a Python script for a faster visualization through the “dash” and “plotly” packages. With this change, necessary for *EntoSim* to work into Docker without any X-server support, the results will be visualised in a smarter way directly on the internet browser, simplifying the use of the software on remote machines, also. The third major change was the addition of two more population density models (Rossini et al., 2020a, 2020c).

Lastly, a complete dataset has been also provided within the *EntoSim* Docker image, so that the users could familiarize with the software and reproduce the same results and graphs, to later move to their own research projects.

## 2. Materials and method

Insects, and ectotherms in general, are strongly dependent by the external environment. Since the first experimentations towards this direction (Harcourt, 1969), temperature was considered the main driving variable for insects-development (Colinet et al., 2015). This led, along the years, several researchers to find mathematical expressions linking the average temperature of the living environment with the insects' development time (Damos and Savopoulou-Soultani, 2012; Ikemoto and Kiritani, 2019). Theoretically, each species has a “thermal spectrum” (Rossini et al., 2019b), namely a range of temperatures bounded by a lower and an upper threshold between which the development of the species is allowed (Shi et al., 2017). Inside this range there is an optimal temperature, which is, by definition, the value where the species closes its life cycle in the shortest time (Quinn, 2017). Hence, the development time is as longer as the environmental temperature is far from its optimal value and near to the thresholds (Mirhosseini et al., 2017).

This typical increasing/decreasing profile is well described by empirical and not empirical expressions, already involved into the initial version of *EntoSim*, as reported in Rossini et al. (2020c, 2019a). More specifically, the expressions included within *EntoSim* are the Brière (Briere et al., 1999), the Logan (Logan et al., 1976), the Sharpe and De Michele (Sharpe and De Michele, 1977), and the linear (Damos and Savopoulou-Soultani, 2012; Severini and Gilioli, 2002) development rate functions.

The major mathematical contribution to discuss in the current version of *EntoSim* is represented by the implementation of new population density models, whose mathematical details and validations were already discussed in the cited literature. The availability of multiple population density models and development rate functions in a single software allows a faster analysis and comparison: in this way the user has a wider scenario about the best combination of phenological and population density models for her/his specific use. The selection criteria of the most suitable development rate function and the most suitable population density model were discussed by Rossini et al. (2020a), and can be also extended to all the physiologically based models.

Besides the Manetsch's Distributed Delay Model (Manetsch, 1976; Vansickle, 1977a) and the Generalised Von Foerster's equation (Rossini et al., 2020d, 2020f, 2019b; Vansickle, 1977b; Von Foerster, 1959) two additional population density models were added in this version of *EntoSim*. The first one is a revised form of the Manetsch's Distributed Delay Model (Rossini et al., 2020a), which expresses, as solution, the distribution of the individuals within the life stages.

The insertion of this revised version of the Distributed Delay Model in *EntoSim* would be beneficial to the wide set of users which are still confident with the Distributed Delay Model theory. The classical version (Manetsch, 1976; Vansickle, 1977a), in fact, describes the flow of individuals exiting from each stage, while its revised version has, as solution, the number of individuals which are present in each life stage.

However, the difference of behavior of these two versions of the model has not been widely explored enough. *EntoSim* can promote the comparison between the two model versions and future validations in other case studies of entomological relevance.

One of the main concerns in applying physiologically based models in varying temperature environments is the absence of an expression which designates (exactly) the daily temperature variations (Rossini et al., 2020f). Accordingly, this value is usually provided by direct measurements with meteorological stations and inserted into the development rate functions, requiring numerical solutions to solve the population density models.

The second population density model introduced with this version of *EntoSim* is a generalised form of the Von Foerster's equation with a reproduction term (Rossini et al., 2020c).

The third block of *EntoSim*, instead, concerns the validation or, more in general, the comparison between two or more data series. In the initial version of the software only the  $\chi^2$  function was used as overlapping index (Rossini et al., 2020d). In this revised version, instead,  $\chi^2$  was supported by the coefficient of determination  $R^2$ , in order to make applicable the protocol analysis proposed by Shi et al.

(2017), Rossini et al. (2020a) and Bellocchi et al. (2011).

On the side of the programming language, the choice of C/C++ and the ROOT's libraries was maintained: they still manage all the calculations and fitting operations, respectively, but in this revised version, Python language was chosen for the graphical part. The reasons behind this choice are manifold: firstly, the capability to visualise results directly on a web browser page results into a faster option if compared to ROOT's *TApplication*. This become a key factor when *EntoSim* has run in remote machines, since the visualization using an internet browser is faster than using the Linux or MacOS X server. Secondly, using Python instead of ROOT's *TApplication* gives to the end-users the possibility to reverse back directly to the *EntoSim* menu, instead of restarting the executable code each time.

As already stated in the Introduction, the most important novelty on the programming and deployment side is the use, for the first time in entomological contexts of 'containers' technology based on Docker runtime (Docker, 2021). In fact, as the complexity of *EntoSim* increased, the runtime dependencies become complex as well as risking to miss some library objects at build time.

A Docker image can be considered as an executable software package including the essential to run the specific program, from the source code to the system tools, libraries, and other settings. In simple words, a Docker image container is similar to a virtual machine but with the great advantage to be more efficient, since it virtualizes the operating system instead of the hardware. Hence, once a Docker container image is developed, the user needs only to install the Docker engine on its computer, which is available for all the operating systems and makes the runtime of the containers the same in every machine.

The *EntoSim* Docker image was built through a Dockerfile, "a text document that contains all the commands a user could call on the command line to assemble an image" (DD, 2020) and made it available on GitHub page of *EntoSim* (ESGHP, 2021).

Proceeding by order, the first dependence provided into the Dockerfile was a base version of Linux CentOS 7 (LC, 2020). Since this version has only the essentials packages, the following commands inserted into the Dockerfile concerned the update of the existing CentOS packages and the installation of the dependencies required by ROOT. Subsequently, the Dockerfile provided for the installation of Python v3.6 and its packages *plotly*, *pandas*, *dash* and *numpy*.

At this point, the environment of the Docker image was suitable for the *EntoSim* installation that, to simplify the operation, has been entrusted to the related Makefile. The Makefile compiles the source code and builds three libraries containing the functions which supports the three blocks of the software, namely "fittasso", "simulator" and "validator".

The *EntoSim* Docker image is stored on the dedicated repository (ESDP, 2021) and it is available to the users through the Docker pull service (DD,

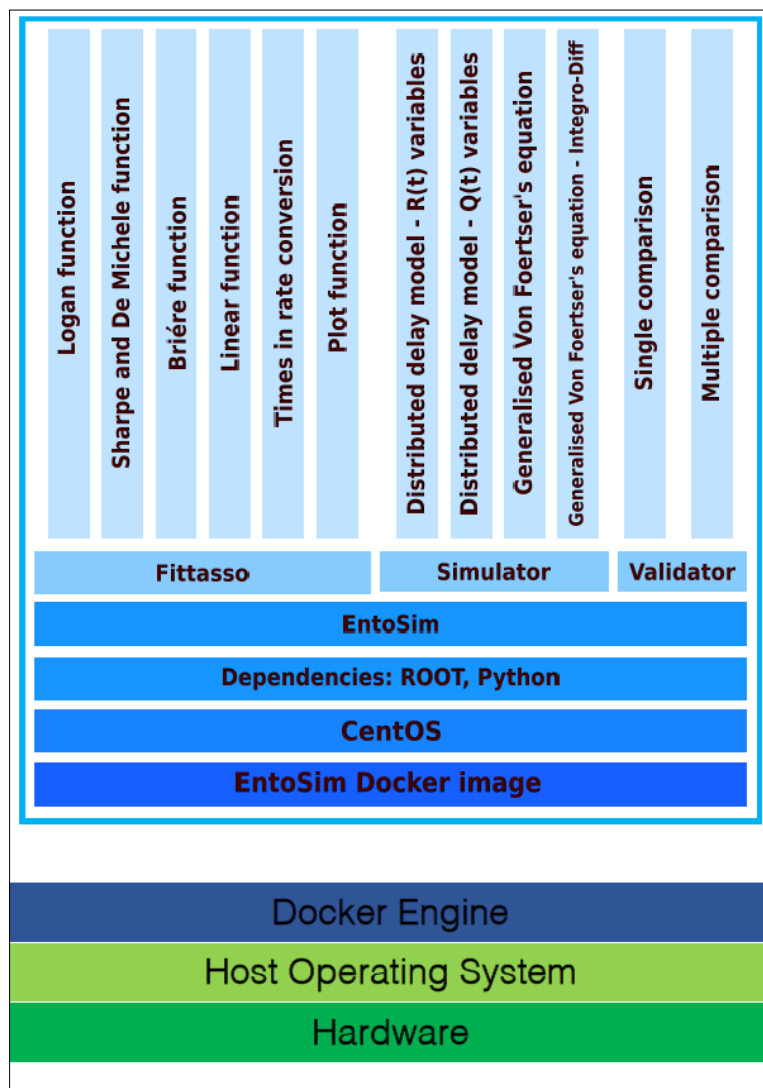
2020). In this way users have just to pull the image from the Docker repository to use *EntoSim*; moreover, it is possible to remove and pull it again every time there is the need to make the space available on the hard drives or when an update is made available by the developers. It is worth to mention that the Docker image solves all the problems of installation and compatibility, but the *EntoSim* code has been made available on GitHub for the users who wants to: *i*) compile and install it from the source code (solving all the dependencies by themselves) and *ii*) contribute to *EntoSim* with novel functionalities.

A schematic representation of *EntoSim* within the Docker image is shown in Fig. 1. *EntoSim* is composed by a main menu, printed on the computer shell, which recall the three main “macro-steps” of model application: *fittasso*, *simulator* and *validator*.

In the next section, a brief application of the software has been reported as result, considering literature data about the case study of *Tuta absoluta* (Meyrick) from Özgökçe et al. (2016). This case study is contained into the Docker image as working example for the users.

### 3. Results

Once the user has downloaded and run the *EntoSim* Docker image, following the instructions provided, she/he is ready to use the software for its own purposes. To allow the transfer of input/output files, a dedicated folder, *EntoSim-data*, has been created into the Docker container so the user may mount its data directory (typically the user’s local homedir) at container start time. Hence, through this folder, the user can insert the input files in the specific *EntoSim*’s input folders and store the results in the local machine copying them from the *EntoSim*’s output folders. The first block of *EntoSim* is “*Fittasso*” composed, in turn, by six functions. The first function converts the experimental life tables data series from temperature-mean development time format in temperature-development rate format. As shown in Fig. 1, four *Fittasso* functions manage the non-linear fitting operations with the development rate functions, providing the results both in a graphical form on a web browser page and stored into specific ASCII text output files.



**Fig. 1.** Schematic representation of *EntoSim* docker image and how it is related to the computer hardware and operating system

For a visual inspection, fit results are displayed on the computer shell: in such way the user can quickly see if the results are satisfying or not. The best fit parameters of the development rate function are stored, also, in a hidden folder, so that “*Simulator*” can directly take these values as input. The last function involved in *Fittasso* allows the user to show the resulting best fit functions, provided by the previous operations, overlapped in an interactive single plot with life tables data.

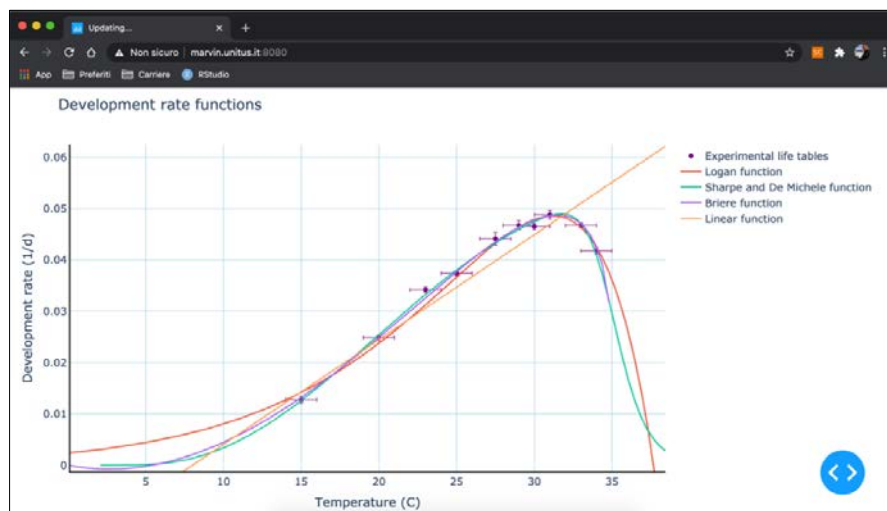
A working example of this first block of the *EntoSim* is provided in (Fig. 2) in the case of *T. absoluta*. Once the life tables data from Özgökçe et al. (2016) are inserted, and once the parameters of all the development rate functions are estimated, a graphical resume is important to select which is the most representing function for the dataset and for the species under study. Even though the  $\chi^2$  and  $R^2$  values are a valuable indication of the best fit function, it is worth to keep in mind that the final choice is weighted also by the biological meaning of the functions’ parameters and by the experience of the users in respect of the species simulated.

The second block of *EntoSim* is “*Simulator*” (Fig. 1) composed, in turn, by four functions: each for each population dynamics model involved, that is the Manetsch’s DDM, its revised form, the generalised Von Foerster’s equation and the generalised Von Foerster’s integro-differential equation. Once the user selects a model, the input files are automatically absorbed from specific folders. More specifically, the number of inputs changes with the model selected, while the outputs are provided all in the same way. The latter are stored in specific .txt files, to be utilised in further analyses, and plotted in a web browser page in three tabs. The first tab hosts a 3D representation of the population density with stage and time as independent variables (Fig. 3); the second tab hosts a plot where the population densities over time provided

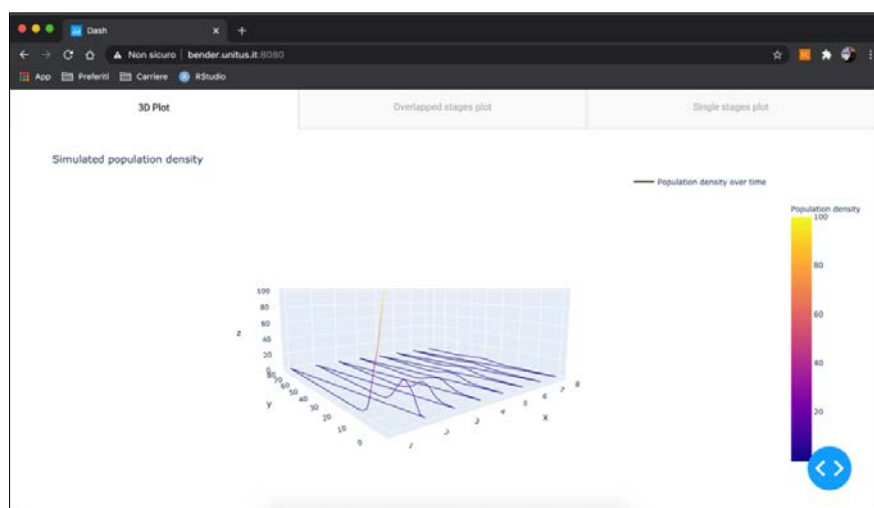
by simulations are overlapped (Fig. 4) and the third tab hosts multiple plots, each of those reports the population density, over time, of the simulated life stages.

The working example provided in Fig. 3 and Fig. 4 is referred to simulations in the case of *T. absoluta* using the generalized Von Foerster’s equation as population dynamics model, and the Sharpe and De Michele development rate function as input. Temperature data, instead, are not referred to a particular case study, and were generated randomly (considering an optimum for the species of 32 °C) only for demonstration purposes.

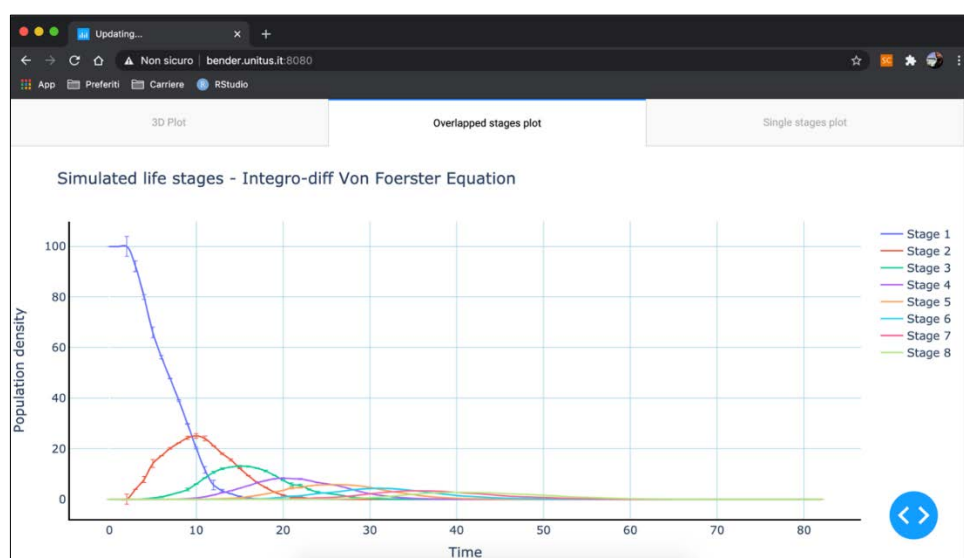
The third and last block of *EntoSim* is “*Validator*”. This tool was initially thought to estimate the  $h$  value needed by the Manetsch’s DDM (Rossini et al., 2019a), but in the course of time its usefulness has grown. It is composed by two functions (Fig. 1) with similar tasks. Proceeding by order, the first function of *Validator* calculates the  $\chi^2$  and  $R^2$  values comparing only two series of data; results are then provided both in numerical (a .txt file) and graphical form (a QQ-plot). The second function of *Validator*, instead, allows users to compare a data series (which may be validation data or theoretical values) with all the simulations provided by *Simulator*. Then the results are stored in a .txt file and ordered from the highest to the lowest overlapped on the basis of the  $\chi^2$ -value. The aim of *Validator* is to provide the user a tool to quantify the overlap between data series and beyond. In fact, it is possible to compare: (i) validation data with simulations; (ii) different outputs provided by different models, or (iii) outputs by the same model with different development rate functions in input, supporting the *a priori* and *a posteriori* analysis introduced by Rossini et al. (2020a). The *EntoSim* code was organised in separated modules, so that just minimal modifications of the existing code are typically required to implement new features.



**Fig. 2.** Visual result of *Fittasso*. This output provides an overlap of the best fit functions, calculated by *Fittasso*, and of the experimental life tables. Plot are shown in the internet browser in an interactive mode, so that the user can zoom specific part of the plot or visualize/remove plotted functions. As example, the life tables of the tomato leaf miner *Tuta absoluta* (Meyrick) from Özgökçe et al. (2016) were used, as showed by Rossini et al. (2019b).



**Fig. 3.** Visual result of *Simulator*. A triad of plots is shown in an internet browser webpage. In this specific picture a 3D plot of the simulated population density developing over time and through the life stages is reported



**Fig. 4.** Visual result of *Simulator*. A triad of plots is shown in an internet browser webpage. In this specific picture an overlap of the simulated life stages over time is reported

This can foster future software developments: for instance, if a new function is required by the user, it is possible to develop only the dedicated module. Subsequently, the latter can be inserted into *EntoSim* only modifying the menu and the code of the block of destination.

#### 4. Conclusions

Since its first introduction, *EntoSim* aimed to support a faster application of physiologically based models for decision making in agriculture, forest and ecological sciences. The inclusion of *EntoSim* into a Docker image introduced with the present work endorse the *EntoSim* shareability and development, providing a powerful tool to the insiders.

The development and validation of new models describing the same natural phenomenon, in fact, increases the need of an easier model application and

comparison of their outputs. The use of Docker complies with this aim, and with a better reproducibility of the results in entomological modelling.

Thus, a modular software enabling multiple physiologically based models, as well as *EntoSim*, is a great resource for scientists, technicians, and farmers, since they could find the better combination of phenological and population density models for their specific use. Furthermore, an open-source code software, with a high shareability, strongly reduces the costs of development and distribution, in favour of the users with limited economic resources. This is the main reason of why the *EntoSim* code was made available on two different repositories: the Docker image makes the use of *EntoSim* easier, but at the same time the GitHub source code allows future development and contribution from the scientific community.



In the future the purpose is to enrich even more *EntoSim* with new functions, as well as to simplify the operations. In this regard, the development of a web application to substitute for the actual menu would be beneficial to the less specialised users.

## Acknowledgements

The authors are grateful to Prof. Nico Sanna for the help and advice given during the development of the present study, and to the two anonymous Reviewers for their careful read of this manuscript, and for the helpful comments and suggestions.

The research was carried out in the frame of the Italian MIUR (Ministry for Education, University and Research) initiative "Department of Excellence" (Law 232/2016).

## References

- Bange M., Deutscher S., Larsen D., Linsley D., Whiteside S., (2004), A handheld decision support system to facilitate improved insect pest management in Australian cotton systems, *Computers and Electronics in Agriculture*, **43**, 131-147.
- Bellocchi G., Rivington M., Donatelli M., Matthews K., (2011), *Validation of Biophysical Models: Issues and Methodologies*, In: *Sustainable Agriculture*, vol 2. Springer Netherlands, Dordrecht, 577-603.
- Boettiger C., (2015), An introduction to Docker for reproducible research, *ACM SIGOPS Operating Systems Review*, **49**, 71-79.
- Briere J.-F., Pracros P., Le Roux A.-Y., Pierre J.-S., (1999), A novel rate model of temperature-dependent development for arthropods, *Environmental Entomology*, **28**, 22-29.
- Brun R., Rademakers F., (1997), ROOT - An object oriented data analysis framework, *Nuclear Instruments & Methods in Physics Research Section A*, **389**, 81-86.
- Capinera J.L., (2001), *Handbook of Vegetable Pests*, Academic Press, Elsevier, London, UK.
- Colinet H., Sinclair B.J., Vernon P., Renault D., (2015), Insects in fluctuating thermal environments, *Annual Review of Entomology*, **60**, 123-140.
- Damos P., Savopoulou-Soultani M., (2012), Temperature-driven models for insect development and vital thermal requirements, *Psyche: A Journal of Entomology*, 123405, <https://doi.org/10.1155/2012/123405>
- Di Tommaso P., Palumbo E., Chatzou M., Prieto P., Heuer M.L., Notredame C., (2015), The impact of Docker containers on the performance of genomic pipelines, *PeerJ*, **9**, 1-10.
- Docker, (2021), Developer Preview Program, On line at: <https://www.docker.com/>
- DD, (2021), Docker docs, On line at: <https://docs.docker.com/>
- ESDP, (2021), "EntoSim" Docker page, On line at: <https://github.com/lucaros1190/EntoSim-2.1>
- ESGHP, (2021), "EntoSim" GitHub page, On line at: <https://hub.docker.com/r/lucaros1190/entosim/>
- Harcourt D.G., (1969), Development and use of life tables in study of natural insect populations, *Annual Review of Entomology*, **14**, 175-196.
- Holst N., (2013), A universal simulator for ecological models, *Ecological Informatics*, **13**, 70-76.
- Ikemoto T., Kiritani K., (2019), Novel method of specifying low and high threshold temperatures using thermodynamic SSI model of insect development, *Environmental Entomology*, **48**, 479-488.
- Knight J.D., Mumford J.D., (1994), Decision support systems in crop protection, *Outlook on Agriculture*, **23**, 281-285.
- LC, (2020), Linux CentOS, On line at: <https://www.centos.org/>
- Logan J.A., Wollkind D.J., Hoyt S.C., Tanigoshi L.K., (1976), An analytic model for description of temperature dependent rate phenomena in arthropods, *Environmental Entomology*, **5**, 1133-1140.
- Manetsch T.J., (1976), Time-varying distributed delays and their use in aggregative models of large systems, *IEEE Transactions on Systems Man and Cybernetics*, **SMC-6**, 547-553.
- Mirhosseini M.A., Fathipour Y., Reddy, G.V.P., (2017), Arthropod development's response to temperature: a review and new software for modeling, *Annals of Entomological Society of America*, **110**, 507-520.
- Orlandini S., Magarey R.D., Park E.W., Sporleder M., Kroschel J., (2018), *Methods of Agroclimatology: Modeling Approaches for Pests and Diseases*, In: *Agroclimatology: Linking Agriculture to Climate*, Hatfield J.L., Sivakumar M.V., Prueger J.H. (Eds.), Wiley, Indianapolis, USA, 453-488.
- Özgökçe M.S., Bayındır A., Karaca İ., (2016), Temperature-dependent development of the tomato leaf miner, *Tuta absoluta* (Meyrick) (Lepidoptera: Gelechiidae) on tomato plant *Lycopersicon esculentum* Mill. (Solanaceae), *Turkish Journal of Entomology*, **40**, 51-59.
- Quinn B.K., (2017), A critical review of the use and performance of different function types for modeling temperature-dependent development of arthropod larvae, *Journal of Thermal Biology*, **63**, 65-77.
- Rossini L., Severini M., Contarini M., Speranza S., (2019a), Use of ROOT to build a software optimized for parameter estimation and simulations with Distributed Delay Model, *Ecological Informatics*, **50**, 184-190.
- Rossini L., Severini M., Contarini M., Speranza S., (2019b), A novel modelling approach to describe an insect life cycle vis-à-vis plant protection: description and application in the case study of *Tuta absoluta*, *Ecological Modelling*, **409**, 108778, <https://doi.org/10.1016/j.ecolmodel.2019.108778>
- Rossini L., Contarini M., Severini M., Speranza S., (2020a), Reformulation of the distributed delay model to describe insect pest populations using count variables, *Ecological Modelling*, **436**, 109286, <https://doi.org/10.1016/j.ecolmodel.2020.109286>
- Rossini L., Contarini M., Severini M., Talano D., Speranza S., (2020b), A modelling approach to describe the *Anthonomus eugenii* (Coleoptera: Curculionidae) life cycle in plant protection: A priori and a posteriori analysis, *Florida Entomologist*, **103**, 259-263.
- Rossini L., Contarini M., Speranza S., (2020c), A novel version of the Von Foerster equation to describe poikilothermic organisms including physiological age and reproduction rate, *Ricerche di Matematica*, <https://doi.org/10.1007/s11587-020-00489-6>.
- Rossini L., Severini M., Contarini M., Speranza S., (2020d), *EntoSim*, a ROOT-based simulator to forecast insects' life cycle: Description and application in the case of *Lobesia botrana*, *Crop Protection*, **129**, 105024, <https://doi.org/10.1016/j.cropro.2019.105024>.
- Rossini L., Contarini M., Giarruzzo F., Assennato M., Speranza S., (2020e), Modelling *Drosophila suzukii* adult male populations: A physiologically based approach with validation, *Insects*, **11**, 751, <https://doi.org/10.3390/insects11110751>
- Rossini L., Speranza S., Contarini M., (2020f), Distributed Delay Model and Von Foerster's equation: Different

- points of view to describe insects' life cycles with chronological age and physiological time, *Ecological Informatics*, **59**, 101117, <http://doi.org/10.1016/j.ecoinf.2020.101117>
- Severini M., Gilioli G., (2002) Storia e filosofia dei modelli di simulazione nella difesa delle colture agrarie (in Italian), *Notiziario sulla Protezione delle piante*, **15**, 9-29.
- Sharpe P.J.H., DeMichele D.W., (1977), Reaction kinetics of poikilotherm development, *Journal of Theoretical Biology*, **64**, 649-670.
- Shi P.-J., Reddy G.V.P., Chen L., Ge F., (2017), Comparison of thermal performance equations in describing temperature-dependent developmental rates of insects: (I) Empirical models, *Annals of the Entomological Society of America*, **110**, 113-120.
- Sinclair T.R., Seligman N.G., (1996), Crop modeling: from infancy to maturity, *Agronomical Journal*, **88**, 698.
- Tang S., Cheke R.A., (2008), Models for integrated pest control and their biological implications, *Mathematical Biosciences*, **215**, 115-125.
- Vansickle J., (1977a), Attrition in distributed delay models, *IEEE Transactions on Systems Man and Cybernetics*, **7**, 635-638.
- Vansickle J., (1977b) Analysis of a distributed-parameter population model based on physiological age, *Journal of Theoretical Biology*, **64**, 571-586.
- Von Foerster H., (1959), *Some Remarks on Changing Population*, In: *The Kinetics of Cellular Proliferation*, Stohlman J.F. (Ed), Grune and Stratton, New York, 382-407.