Simulation and Visualization of Nectarine Branching and Fruiting Responses to Pruning Using PrungingSim Software

Ning Xia[*], Aishuang Li and Weijun Lin

Institute of Agricultural economy, Guangdong Agricultural Academy of Sciences, 510640 China

Abstract: In this paper, we present a software tool named PruningSim dedicated to dynamic simulation and visualization of plants. The kernel framework provides various classes for event manager to schedule the handling of different types of events. The simulation in PruningSim platform is based on Markov model corresponding to branching processes in plants. Topological and geometrical parameters can be assigned automatically or modified manually in the simple graphical user interface. Visualization and image setting option is also available. Furthermore, we present one case study for simulation nectarine branches' responses to pruning intensities using PruningSim.

Keywords: Visualization, Growth simulation, Plant, Model.

INTRODUCTION

Some significant progresses have been obtained over the past 30 years in plants growth modeling and simulation. Some simulation system are designed mainly based on the stochastic process, *e.g*., AmapPara for simulation of structure (Barczi *et al*. 2008) and function of plants (Barthelemy and Caraglio, 2007; Godin and Sinoquet, 2005; de Reffye *et al*., 1995), and others focus on the formal grammar, for example, L-system and particle systems with emphasis on visual effects (Prusinkiewicz and Lindenmayer, 1990; Kurth, 1994).

These simulation system have been used in a wide range of agronomy application (Xia *et al*., 2012; Leroy 2005; Leroy *et al*. 2009), such as plantation density optimization, crop ideotype design, prioritization fertilizing scheme and forestry management (Godin and Sinoquet, 2005; Turnbull, 2005).

The general purpose of the study is to represent how the PruningSim plant architecture simulator deal with the correlation between structure and function as it includes both botanical branching patterns and growth dynamics, based on topological and geometrical rules, and plug-in functions that enable interfacing with other applications.

The first section of this paper outlines software architecture of the kernel module of growth generator.The second section gives a description of some parent branches' response to different pruning intensities that are dynamically simulated. Emphasis is the data management, *i.e*. topological parameters, geometrical parameters, simulation process, and data output. The discussion and conclusion in the third section has served to emphasize the important role the PruningSim software plays in the instruction for dynamic crop growth simulation and its application for precision agronomy and horticulture.

KERNEL MODULE OF GROWTH GENERATOR

Programs designated by the name of PruningSim allow the user to simulate a plant according to various parameter values through what is known as a growth generator. Growth generator runs under Linux operating platform. Kernel module of growth generator can simulate various probability distributions of branches, numbers of internodes, numbers of leaves, numbers of flowers and fruits along the parent branches.
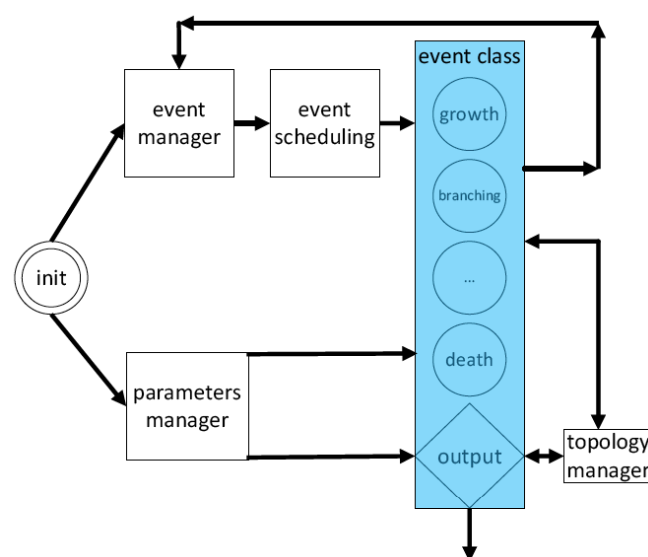

Figure 1: Schematic diagram of kernel module of growth generator

[*]Address correspondence to this author at the Institute of Agricultural economy, Guangdong Agricultural Academy of Sciences, 510640 China; Tel: 0862038319940; Fax: 0862038319940; E-mail: nxia01@sina.com

The software provides various classes for event manager to schedule the handling of different types of events. When an event is attached to a bud, the corresponding Effect Event Process function is called in the event manager. Effect Event Process function is run each time, an event occurs with index. Effect New Entn Data function creates the appropriate data structure to be attached to each internode of the plant structure. Load Event function loads an event stack from a file. This stack had to be saved using the Save Event function. Effect Parameter Mod if function is called each time a parameter in the manager is accessed. Each parameter can be accessed either through its decimal index or through its name. Save Event function saves an event stack to a file for further use. This method can reduce the computational and memory requirement of algorithm as the intricacy of the branching structure increases. Topology manager can easily create various types of plant topological structures which corresponds to a set of components either succession or branching relations at different scales. For example, Effect Load Bud Data function is called when a node (bud) is loaded from a temporary file or when processing a hot start. Effect New Bud function is run each time a new bud is created, it is called just after the internal initialization are done for the new branch. Effect New Bud Data function create the appropriate data structure to be attached to each node (bud) of the plant structure. It is called after each node (bud) creation. Effect New Entn Data function create the appropriate data structure to be attached to each internode of the plant structure. Geometry manager consists in a collection of functions which gives a precise description of a set of steps for computing the geometric characteristic, *e.g.* Compute Diam function computes the current diameter of an organ according to its position within the current branch. There are two supported interpolation modes, *i.e.* nearest-neighbor interpolation and linear interpolation, available for various geometry transformations in the geometry manager. The former can be a favorable choice for its efficiency and ease-of-use, and the latter is simple and fast, but it is not enough precise (Figure **1**).

PruningSim offers dedicate functions which create the appropriate data structure to be attached to each node (bud, internode) of the plant structure during a simulation, for instance, Effect Begin Geometry function is called before any geometrical output from the kernel that creates a line tree set of files. If Effect Geometry Function is set to 1, the default geometry computing provided by the kernel will be overwritten by the corresponding effect function. If Effect Geometry Function. Diameter is set to 1, this function will be able to overwrite the default diameter kernel computing for the current organ. If Effect Geometry Function. Length is set to 1, this function will be able to overwrite the default length kernel computing for the current organ. If Effect Geometry Function. Orientation is set to 1, this function will be able to modify (or overwrite) the default orientation computing. The input values are the one that are computed according to the geometrical model of the kernel.

Plant structure is reconstructed relying on the production of virtual bud population with probability of occurrence such as growth, branching or death. The essential step for growth simulation is the output of internodes, which means that each of virtual buds systematically calls its own model in the identical kernel for formation of an internode. A simulation run time yield a parallel simulation architecture, which can be used to animate the resulting 3d structure generated. When execution of a loop starts, the virtual bud verifies its function kept alive. If bud does not die, it will generate an internode and a new terminal bud that build the basic structure by a set of rules. Next the branching tool will execute which can refresh the terminal buds datasets using new lateral buds, subsequently state tests will iteratively start adding internodes and buds to direct the growth forward (Figure **2**).
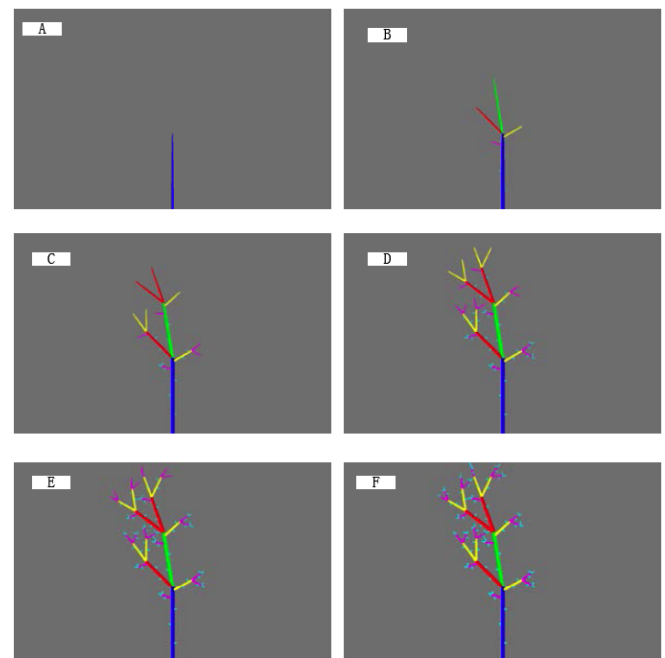

Figure 2: Schematic diagram of the basic steps in the simulation process. A-F represents a step, respectively.

As an example, nested if statement, whose basic syntax shown in the pseudo-code, is demonstrated as follows

```
 for each run time do
   for each bud which is still alive do
     (order, dimension, position, etc. are custom attributes of the bud)
       if bud doesn't die then
         if bud doesn't make a pause then
           create internode (with topological and geometric parameters)
             create apical bud
               for each possible bud do
                 if branching then create lateral buds
               endfor
         endif
       endif
     endfor
   endfor
```

When compared with other software applications such as L-studio, Plant GL, and ALEA, available for plant simulation, PruningSim can reconstruct 3D plant structure from field data at the internode and shoot scale, which has many potential applications in agriculture (Xia *et al.*, 2012).

CASE STUDY

We use nectarine as one case study on parent shoots' branching and fruiting responses to pruning. This example demonstrates the application of PruningSim software and the validity of the simulation results.

TOPOLOGICAL PARAMETERS

A nectarine parent branch had been produced at the end of the previous growing season and its axillary shoots were formed in the current one. In such case, a parent branch can be decomposed into different branching zones. Each branching zone contains single or mix shoot types, such as short shoots, long shoots, and sylleptic shoots, which can be defined by variations in the internode length and number. In PruningSim, the smallest unit that can be generated is the internode without buds and leaves. The notion of internode here is therefore different from the morphological one which

represents the portion of stem between any two successive nodes (Figure **3**).

To describe the topological organization, binomial laws will be used to simulate internodes distribution along a parent branch and its axillary shoots, which mean the engine, construct n internodes with probability p during a computation cycle.

```
Maxi test count       (inside UT)      1   10        1/ 5 c
Overadded test count  2nd cycle        1    0        1/ 1 c
Overadded test count  3rd cycle        1    0        1/ 1 c
Overadded test count  4th cycle        1    0        1/ 1 c
Overadded test count  5th cycle        1    0        1/ 1 c
```

Figure 3: The internode number of the pruned parent branch.

The distribution of axillary shoots along the pruned parent branches was computed according to a Markov process with initial probability and transition probability for different axillary shoot types (Figure **4**).
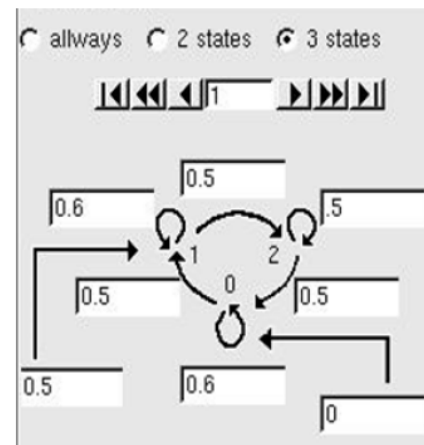

Figure 4: Example of branching Markov process.

All leaves, fruits and axillary shoots along the pruned parent branch were constructed at the same computation cycle.

The internode number of long shoots follows binomial law with parameters n=34 and p=0.33. In the same manner as long shoot, corresponding parameters of short shoots are n=15 and p=0.48).

The distribution of fruit number per pruned parent branch can also be computed according to first-order Markov model.

GEOMETRICAL PARAMETER

If the user does not provide a value for the general geometrical parameter, for instance, phyllotaxy, insertion angle, insertion type, elastic modulus and straightening, the default value will automatically assign

to software kernel. If the user input a value for the specific geometrical parameter, such as length, diameter, position, orientation, allometric coefficient and random ratio, the user-supplied value will overwrite the default value.

```
UT global geometry

Insertion type                     1   1            1/ 2 c

Initial sympode angle              1   0.000000     1/ 1 c

Young module                       1   100000.000000 1/ 1 c
Force location ratio               1   0.000000     1/ 1 c
Conicity ratio                     1   0.000000     1/ 1 c
Straightening angle                1   0.000000     1/ 1 c
Straightening area                 1   0            1/ 1 c

Organ initial bending angle        1   0.000000     1/ 1 c

Random ratio                       1   0.000000     1/ 1 c

geometry counting direction        1   0            1/ 1 c
       (0 basal / 1 apical)
```

Figure 5: The geometrical parameter of the pruned parent branch.

In the most common case, the user can apply the built-in default geometrical parameters (Figure **5**) to compute the geometrical shapes and variation of the organs. Nevertheless, modification of the parameter values of parent branches, axillary shoots and leaves over the time is fairly simple. Indeed, the user may edit his or her own parameter file to change the parameter for different organs, *e.g.*, parent branches, axillary shoots, leaves, flowers, and fruits.

SIMULATION PROCESS

The plant simulator can be manually invoked through a command-line or a graphical user interface named New Prg that will make simulation easier (Figure **6**).
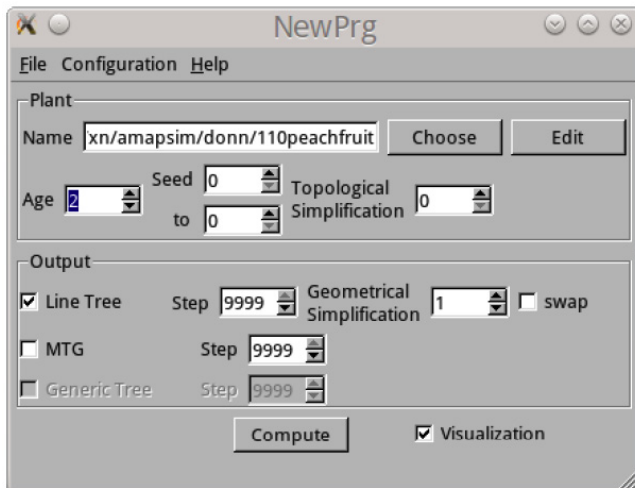
Figure 6: Graphical user interface of simulator.

If initialization function exists, the shared library is linked to the kernel and this function is called before processing the first event of the simulation and after any internal initialization. If it does not exist, the kernel will throw an exception. Then, the parameters and the simulation times are loaded. Next, the initial event for initial computing time step is created. Last, the growth engine will compute organ position and dimensions using plant topological and geometrical parameter until the subsequent time step (Figure **7**).

The event stack is an array of structure to list the event to apply to nectarine. Every event has an instant and a priority attached to it, the kernel sorts the event stack so that the current event is always the one with smaller instant (the older one) and the higher priority, for events with same instant and priority, the kernel makes a random choice. When an event is applied to kernel, the corresponding function is called. The events that are attached to the plants buds have priority equal to their branching order. The branches stack is an array of structure to describe each of the simulation
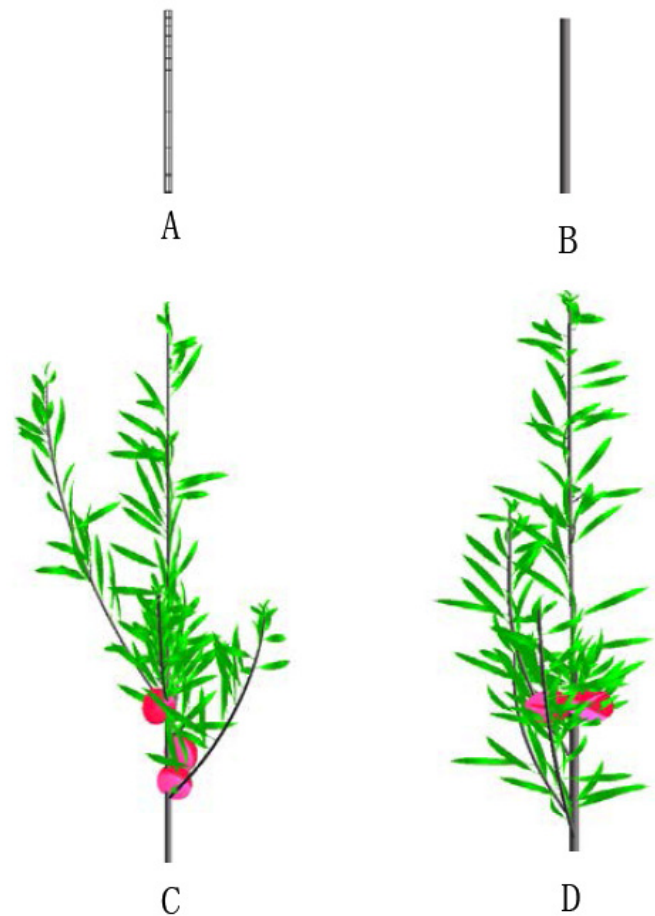

Figure 7: Examples of nectarine branching and fruiting responses to pruning.
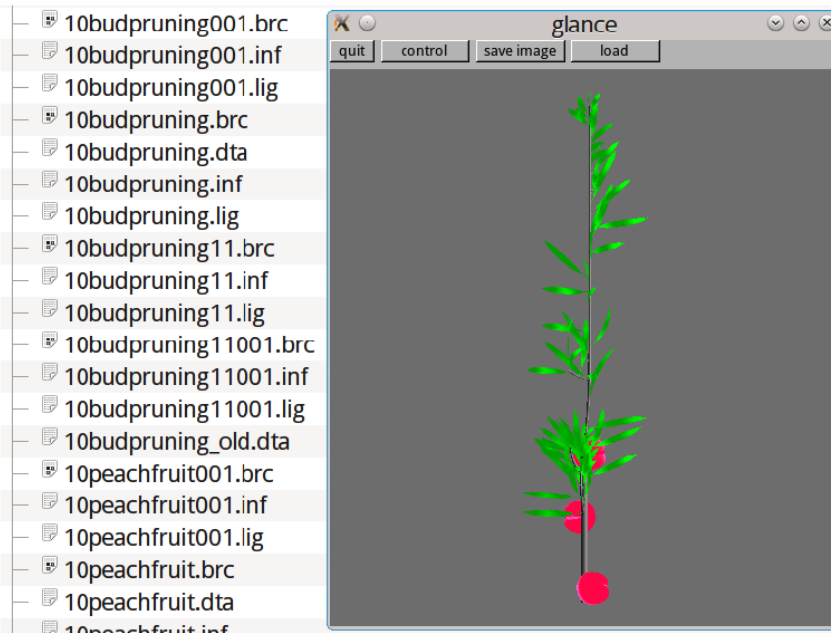
Figure 8: Examples of the output file formats.

branches. This structure consists in variables linked to all of the branches. The simulation branches are decomposed into internode. The hierarchic stack is an array of structure to describe the dependency between branches. Each structure gives the index of the parent branches and the axillary shoots in the hierarchic stack. This array provides the topological arrangement of all of the branches.

A indicates wireframe mode of pruned parent branch; B indicates polygon fill mode of pruned parent branch; C and D indicates polygon fill mode of pruned parent branch, axillary shoots and fruits

OUTPUT

There are three major groups of file formats for the storage particular types of simulation output data. The topology file (.brc extension), for example, contains the total number of axillary productions and the topological position of an axillary shoot according to its topological neighborhood along the parent branch. The geometry file with .dta extension contains symbols to represent organs' shapes and color to be used to draw the plant. The line tree file (.lig extension) consists of geometrical symbols to be associated to axillary productions (leaves, axillary shoots, flowers and fruits), numbers and positions of axillary productions along the parent branch. For line trees whose name ends with numbers, a single generic data file can be created. For instance, peachfruit001.lig, peachfruit002.lig, and peachfruit003.-

lig can use the same plant. These line trees can easily then be visualized using relevant software program, *e.g.*, glance, to evaluate the consistency between observed and simulated results.

DISCUSSION AND CONCLUSION

We propose a simulation tool, which is based on stochastic process, to reconstruct 3D architecture of plant species. The simulation results indicate that it can be faithful to real plant with detailed topological and geometrical characteristic at the different scale (Xia *et al*., 2014, Heuret *et al*., 2006). For topological attributes that are usually available in topology manager, various event classes are used to describe different states of plant components, such as sprout, branching, flowering, death. Geometrical feature of plant component are simulated through built-in functions with default parameter value, which makes it easy for new users to generate some plant shapes what they want, without introducing new functions for each of them (Xia *et al*., 2004; Prusinkiewicz and Lindenmayer, 1990).

Some parameter values in the model are observed values derived from field experiments and lab experiments, and others are theoretical values based on the sample data (Godin and Caraglio, 1998). These two types of value are results that a plant response to all environmental factors including abiotic and biotic, thus a single environmental factor that influences a plant is difficult to simulate. Nevertheless, the present

software architecture for simulation of plant growth and development provides an object-oriented manager based framework for generator implementations on a computer. The software further provides external interface which allows integrating external functions with simulation kernel and easy extensibility of software application as an interactive visualization tool in agricultural research fields (Zhang *et al*., 2006), including crop breeding (Dupuy *et al*., 2005), agrophysics (Chenu *et al*., 2005, Picard *et al* 2004, Lanwert, 1998), bioinformatics, and precision farming (Fourcaud *et al*., 2003, Castel *et al*., 2001).

Depending on user's purpose, simulation outputs data may be either JPG, GIF, PNG, or BMP image format.

While runtime errors seldom happen, the severity of those errors that do occur, makes it particularly important to minimize the effects of mistakes or provide adequate guidance on troubleshooting and error recovery in simulation processes.

This paper describes important features of PruningSim software and implementation details for plant growth simulation. Unlike most other plant simulation systems based on production rules (Perttunen and Sievanen, 2005; Karwowski and Prusinkiewicz, 2004), PruningSim offers great adaptability to various user types and requirements due to its built-in 23 architectural models found in the plant kingdom (Hallé and Oldeman, 1970). With no function model built into the PruningSim system as with DigiPlant, but PruningSim can integratewith function model through external modules and then simulate plant physiological responses to changes at the tissue level.

Further works certainly involve creating a more effective framework, in which a model is separate from function sets and methodologies, a novel model could therefore focus on plant growth simulation and also stochastic simulation algorithms could replace the set of functions. In addition, the present event class ought to be designed to be more effective so that a single event easily communicates and combines with other events such as growth, branching, death, occur whether synchronously or asynchronously. For topology manager, the hierarchical representation of plant components is necessary to represent plant topological structures, in which the topological decomposition can be easily applied to create different plant models with flexibility and adaptability.

Despite aforementioned improvements, usability, compatibility still need to be adequately addressed.

This paper presents a visualization platform for plant growth and development with detailed topological and geometrical characteristics. We solve for stochastic structure and motion of each plant component, using Markov chain algorithm. Our results demonstrate that it is possible to reconstruct 3D structures of plant from real plant's response to different pruning intensities with no user input at all.

One major advantage of the simulation method described here in that it allows a user to make wise pruning choice, as well as to make mistakes. Practices on a virtual fruit tree create an immersive training environment to train pruning skills of user, in which a user can not only observe flowering and fruiting responses to various pruning in a real time without a delay, but also practice repetitively tree pruning skills without sacrificing yield and tree health.

The simulation result of nectarine demonstrates the use of this software interface, showing how parent branches responses to different pruning intensities can be visualized by PruningSim system, thus learn to effectively apply pruning skills and use a variety of problem solving strategies.

ACKNOWLEDGEMENTS

REFERENCE

[1]   Barthelemy D, Caraglio Y. Plant architecture: a dynamic, multilevel and comprehensive approach of plant form, structure and ontogeny. Annals of Botany 2007; 99: 375-407
        http://dx.doi.org/10.1093/aob/mcl260

[2]   Castel T, Caraglio Y, Beaudoin A, Borne F. Using SIR-C SAR data and the AMAP model for forest attributes retrieval and 3-D stand simulation. Remote Sensing of Environment 2001; 75: 279-90.
        http://dx.doi.org/10.1016/S0034-4257(00)00173-5

[3]   Chenu K, Franck N, Dauzat J, Barczi JF, Rey H, Lecoeur J. Integrated responses of rosette organogenesis, morphogenesis and architecture to reduced incident light in Arabidopsis thaliana results in higher efficiency of light interception. Functional Plant Biology 2005; 32: 1123-34.
        http://dx.doi.org/10.1071/FP05091

[4]   Dupuy L, Fourcaud T, Stokes A. A numerical investigation into the influence of soil type and root architecture on tree anchorage. Plant and Soil 2005; 278: 119-34.
        http://dx.doi.org/10.1007/s11104-005-7577-2

[5]   Fourcaud T, Dupuy L, Sellier D, Ancelin P, Lac P. Application

of plant architectural models to biomechanics. In: Hu BG, Jaeger M, eds. Plant growth modeling and applications. Beijing: Tsinghua University Press, 2003; 462-79.

[6]   Godin C, Caraglio Y. A multiscale model of plant topological structures. Journal of Theoretical Biology 1998; 191: 1-46.
http://dx.doi.org/10.1006/jtbi.1997.0561

[7]   Godin C, Sinoquet H. Functional-structural plant modelling. New Phytologist 2005; 166: 705-8.
http://dx.doi.org/10.1111/j.1469-8137.2005.01445.x

[8]   Gorte B, Pfeifer N. Structuring laser scanned trees using 3d mathematical morphology. International Archives of Photogrammetry and Remote Sensing 2004; 35: 929 -33.

[9]   Halle F, Oldeman RAA. Essai sur l'architecture et la dynamique de croissance des arbres tropicaux. Paris: Masson1970.

[10]   Heuret P, Meredieu C, Coudurier T, Courdier F, Barthelemy D. Ontogenetic trends in the morphological features of main stem annual shoots of Pinus pinaster (Pinaceae). American Journal of Botany 2006; 93: 1577-87.
http://dx.doi.org/10.3732/ajb.93.11.1577

[11]   Karwowski R, Prusinkiewicz P.The L-system-based plant-modeling environment L-studion 4.0. In Proceedings of the 4th International Workshop on Functional-Structural Plant Models. 2004: p. 403 -5.

[12]   Kurth W. Morphological models of plant growth: possibilities and ecological relevance. Ecological Modelling1994; 75-76: 299-308.
http://dx.doi.org/10.1016/0304-3800(94)90027-2

[13]   Lanwert D, Dauzat J, Friih T. Water use of forest trees: a possibility of combining structures and functional models. Bayreuther Forum Okologie 1998; 52: 117-28.

[14]   Perttunen J, Sievanen R. Incorporating Lindenmayer systems for architectural development in a functional-structural tree model. Ecological Modelling 2005; 181: 479-91.
http://dx.doi.org/10.1016/j.ecolmodel.2004.06.034

[15]   Picard G, Le Toan T, Castel T, Caraglio Y. Radiative transfer modelling of cross-polarized backscatter from a pine forest using the discrete ordinate and eigenvalue method. IEEE Transactions on Geoscience and Remote Sensing 2004; 42: 1720-30.
http://dx.doi.org/10.1109/TGRS.2004.831229

[16]   Prusinkiewicz P, Lindenmayer A. The algorithmic beauty of plants. New York: Springer Verlag1990.
http://dx.doi.org/10.1007/978-1-4613-8476-2

[17]   de Reffye P, Houllier F, Blaise F, Barthelemy D, Dauzat J, Auclair D. A model simulating above and below ground tree architecture with agroforestry applications. Agroforestry System1995; 30: 175-97.
http://dx.doi.org/10.1007/BF00708920

[18]   Turnbull C. Plant architecture and its manipulation. Annual Plant Review, Vol. 17. Oxford: Blackwell 2005.

[19]   Xia N, Guo Y, Deng XM, Li BG. Modeling the branching patterns of peach tree branches (Prunus persica (L) Batsch) after being pruned. Acta Botanica Sinica 2004; 46: 793-802.

[20]   Xia N, Li AS, Lin WJ. Simulation and Quantitative Analysis of Branching Patterns of the Plum Tree. Journal of Computer Science Technology Updates 2014; 1: 9-18.
http://dx.doi.org/10.15379/2410-2938.2014.01.01.02

[21]   Xia N, Lin FH, and Li AS. Modeling and Visualization of Fruit Trees in Horticulture. In: Abramovich S. editor. Computers in Education. New York: Nova Science Publishers, Inc. 2012; p.135-58.