Original article

Towards universality of growth grammars: Models of Bell, Pagès, and Takenaka revisited

Winfried Kurth*

Institute of Forest Biometry and Informatics, University of Göttingen, Büsgenweg 4, D-37077 Göttingen, Germany

(Received 1 February 1999; accepted 17 February 2000)

Abstract – "Growth grammars" are extended parametric Lindenmayer systems, enriched by some novel features (expand operator, global sensitivity, interpretive rules, arithmetical-structural operators). They can serve as a formal basis for describing functional-structural plant models from the literature. This is demonstrated on three well-known models having in common that they were originally developed without using any formal grammars: an early, but quite general structural plant simulator by Bell, a root model by Pagès and Kervella, and an above-ground tree model (involving competition for light) by Takenaka. The study considers the special extensions of L-systems necessary to rebuild some characteristic features of each of these models. The obtained degree of universality and the current limitations of the growth-grammar approach with respect to functional-structural tree models are discussed.

L-system / grammar / formalization / model specification / plant architecture

Résumé – Vers l'universalité des grammaires de croissance : une nouvelle rencontre entre les modèles de Bell, Pagès et Takenaka. Les « grammaires de croissance » sont une version étendue de systèmes de Lindenmayer paramétrisés, élargi par quelques propriétés nouvelles (opérateur d'expansion, sensitivité globale, règles interprétatives, opérateurs arithmétiques-structuraux). Ils peuvent servir comme une base formelle pour la déscription de modèles fonctionnels-structuraux déjà publiés. Cette possibilité est démontrée par trois modèles bien connu qui ont en commun la qualité d'être développé sans utilisation de quelque grammaire formelle : un simulateur structural des plantes de Bell, un modèle de racines de Pagès et Kervella, et un modèle d'arbre sur-sol (inclus la concurrence pour la lumière) de Takenaka. Cette étude considère les extensions spéciales de L-systèmes qui sont nécessaire pour reconstruire quelques propriétés caractéristiques de chacun de ces modèles. Le degré d'universalité obtenu et les limitations actuelles de l'approche de grammaires de croissance en relation avec les modèles fonctionnels-structuraux d'arbre sont discutées.

L-système / grammaire / formalisation / spécification de modèle / architecture des plantes

1. INTRODUCTION

Lindenmayer systems (L-systems) are essentially sets of rules, each rule stating "replace symbol *a* by string *s*", where *s* consists of symbols again. In most cases, the symbols stand for botanical entities like internodes, leaves, root segments etc., and for topological and geometrical specifications about how these entities are interconnected. Some symbols can additionally have numerical parameters associated with them, and deterministic conditions or stochastic laws can control the applicability of rules. By applying the rules to the symbols of a given start string in parallel and by repeating this rewriting process, a development of branching structures in

* Correspondence and reprints
Tel. +49-551-399715; Fax. +49-551-393465; e-mail: wkurth@ufobi4.uni-forst.gwdg.de

discrete time steps is modelled. A precise mathematical definition of L-systems and of rule application and string interpretation can be found elsewhere [21, 39, 42].

L-systems and extensions of the formalism, described under the name “growth grammars” by the author, have had numerous applications in simulations of flowering plants [39], tree architecture [25], shoot development [41], crop plants [8, 12, 16], reaction to pruning [40], root growth [31], competition for light [28, 31], and carbon allocation [26] – to mention only some examples. For a more complete overview and discussion of the applications, the reader is again referred to the literature [36, 42].

Given the situation that a large variety of problems in plant modelling have already been dealt with in the formal framework of growth grammars, the mathematician working on this formalism could be satisfied and lean back. However, the crucial test for the universality (in practical terms) of a formal tool comes when models are to be reconstructed (using the formalism) which were designed by people who did *not* have the considered formalism in mind. In fact, in the above-mentioned examples, the preoccupation of the model designers with the grammar machinery could have biased the choice of problems and methods, and then would have feigned a genericness which doesn’t exist in practice.

To dispel this suspicion, the author is currently undertaking efforts to rebuild several plant models from the literature, old and new, using growth grammars and a generic software tool named GROGRA (Growth Grammar interpreter [21]) for their execution [24]. “Rebuilding a model” here means not just a reproduction of the output, but of the way that the model works. For the examples presented here, this was done with the available literature about the chosen models as the sole basis (with the exception of the models by Adrian Bell, who offered the author the opportunity to get insight in some unpublished material). The restriction to the literature was an interesting exercise: as an early experience it became obvious that *no verbal model description in the literature is complete*. Although there were considerable differences in the precision of the papers, it was never possible to reproduce the models accurately to a degree of 100%. This fact demonstrates already a main advantage of using a formalism like growth grammars for model specification – such formalisms are much more precise and shorter than verbal descriptions partially merged with some equations.

In this short study, four of the examples from [24] are presented. (Other examples of models from the literature rebuilt with grammars can be found in [31, 42].) There will not be enough space to explain the used grammars for each model in full detail, neither can an exhaustive general introduction into the syntax and semantics of growth grammars be given here (see [21, 24] for such material). Instead, the focus will be on some essential feature of each model which required specific extensions of the usual L-system formalism: expand operator, global sensitivity, interpretive rules, and arithmetical-structural operators, together resulting in that what we will call “growth grammars”. For the rules in our examples, we will use the notations from [21, 24]. A quick guide to the notations is also given in [27]. Only those parts of the formalism which are relevant for the discussed extensions will be explained here. — Finally, we will discuss the advantages of using grammars as well as some of the current problems and limitations of the approach.

We will not address the issue of universality in the sense of recursive function theory here. It is well known that parametric L-systems (and hence growth grammars, too) have the full computational power of a universal programming language (see e.g. [37]; an overview of results on L-systems from the viewpoint of formal language theory is given in [20]). However, such a theoretical statement does not say much about the usefulness of the formalism for practical purposes: Nobody will use a Turing machine for programming, although it is a classical result that any calculation can be done on a Turing machine.

2. EXAMPLE 1: BELL, ROBERTS & SMITH (1979)

The modelling system by Bell et al. [4] was among the first published dynamical, morphological plant growth simulators with a certain generality and *not* using L-systems. Their software, called RHIZOM, was able to read formatted tables containing morphological specifications of the growth behaviour of meristems, and to produce graphical results. It was restricted to two dimensions and was used primarily for rhizomatous plants growing in a planar layer below ground: *Alpinia speciosa* (*Zingiberaceae*), *Medeola virginiana* (*Trilliaceae*), *Maranta spp.* (*Marantaceae*), *Carex arenaria* and others. A further application was the seed dispersal from trees with exploding fruits. Later on, first steps towards modelling tree crowns with the software system were undertaken [18]. In all these cases, a table in the input data file determined the production and growth directions of daughter shoots from their respective mother shoots, and this had to be given for each type of shoot occurring in the plant. Normally, the translation of such a table into a set of L-system rules is straightforward. The only complication arises from one special feature of the specification system of RHIZOM: the potential daughter shoots of one shoot are given in a fixed

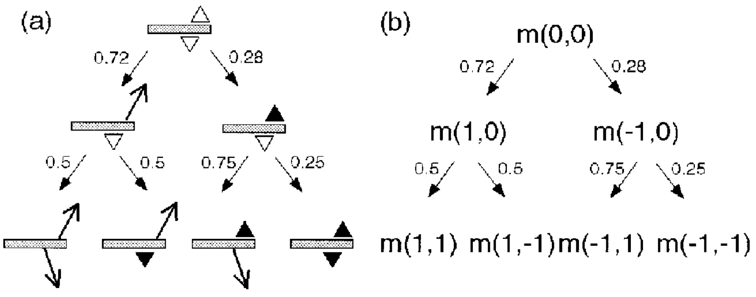

Figure 1. (a) Interaction of daughter meristems of a shoot in Bell's RHIZOM model. Long rectangle: mother shoot (segment of a rhizome system), open triangles: potentially active meristems, filled triangles: inactive meristems, bold arrows: new shoots. The probabilities attached to the possible developmental paths correspond to the values given in [4] for *Alpinia speciosa.* (b) Translation of the process into a symbolic notation usable in a growth grammar. ***m*** stands for the whole set of daughter meristems, the numerical parameters encode the status: 0 = undecided, 1 = growing, –1 = inactive.

order, and the success or failure of a daughter meristem to produce a shoot can have influence on the fate of the later sisters (cf. *figure 1a*). Such interactions between potentially-growing morphological units in the same vegetative developmental step do certainly exist in nature, but they are not a normal part of the lineage-controlled rewriting process of classical L-systems, where the states of the daughter meristems could only be influenced by the state of the mother shoot, not by the sisters. Nor are classical context-sensitive rules appropriate here since they do not take care of the given order among the sisters.

However, it is easy to formalize the mechanism proposed by Bell et al. in a Markov chain-like structure (*figure 1b*), using a single parametric symbol for the whole set of potentially-growing daughters. (The same scheme easily extends to more than two daughter meristems by using a larger list of parameters.) It would even be easy to go beyond the linear scheme of interaction proposed in [4], e.g. by letting "later" sister shoots compete with earlier ones, leading to a reduction of growth. The only remaining problem is that the complete process sketched in *figure 1* takes a variable number of steps, depending on the number of daughter meristems, and has to be considered at a time scale different to that of the growth of the whole plant. To "compress" a sequence of rule applications into a single, superior rule, an expand operator "***E***" was introduced in [22]. It triggers a variable number of applications of further rules, called decomposition rules, which are in our case of a stochastic character and control the development of ***m***(0,0). The single rule encompassing the whole interaction can now be stated as

$$\mathbf{bud} \rightarrow \mathbf{shoot}\ \boldsymbol{E}(3)\ <\boldsymbol{m}(0,0)>,$$

where ***E***(3) enforces three successive rule applications to ***m***(0,0), two of them according to *figure 1* and the third transforming ***m***(*i, j*) into the appropriate number of "**bud**" symbols. (We could also specify a variable number of rule applications here by an arithmetical expression containing stochastic or external variables.) The same effect of performing several rule applications during one step of vegetative development could also be obtained using the instrument of "timed L-systems" introduced in [39], but we feel that this would evoke an inadequate complexity in a case where the focus is not on modelling the underlying biological processes in a more detailed way. – Decomposition rules were introduced independently from [22] also in [30].

In the same manner, it was possible to rebuild the other RHIZOM examples from [4] with growth grammars as well (cf. [24]). Since the RHIZOM software and its specification files today no longer exist in digital form (D. Roberts, oral commun.), we abstained from implementing an automatic translation interface between the softwares, though this would have been easy.

However, this example is of more than historical interest: Markov-chain constructions play an important role also in advanced morphological simulators like *AMAPsim* [1], cf. also [17]. The usage of (possibly nested) expand operators enables an integration of such approaches in the grammar framework.

3. EXAMPLE 2: BELL (1984)

A later version of the RHIZOM system, called SEED3 [2], involved the modelling of spatial interactions between plants by allowing the growth of a shoot to depend on the occurrence or non-occurrence of plant parts in a neighbourhood of the meristem under consideration. Bell used this feature in [2] to study the colonizing habit of two prototypic rhizomatous plant forms, which he called “clumps” and “runners” and which were assumed to exhibit contrasting growth strategies. The important property of both plant types which is interesting in our context is their tendency to avoid sites which are already occupied. A smaller “substitute” shoot into another direction is formed if the place where a shoot normally grows is occupied by some other shoot. Such a dependence on the geometrical context can be represented in a growth grammar by using globally-sensitive functions, as defined in the GROGRA system [21]; in our case, calling a simple distance function (*figure 2*) in a condition of a certain rule suffices. Such a sensitive function gives back some value from the 3-dimensional context of the considered element which is to be transformed, taking the already-created structure into account. Several predefined functions of this type are available in GROGRA [24]. A similar extension of the L-system concept was provided by Prusinkiewicz et al. [40] under the name “environmentally sensitive L-systems”, and later under the generalized notion of “open L-systems” [31], which contain communication symbols for the interaction of the grammar-controlled rewriting process with other models representing the environment.

The central rule switching the state of a meristem from normal growth to “grow a substitute shoot avoiding an occupied site” can with the help of a predefined function “*search*” simply be expressed as

$$(\textit{search} <= \textit{rad})\ \boldsymbol{m} \rightarrow \boldsymbol{msub},$$

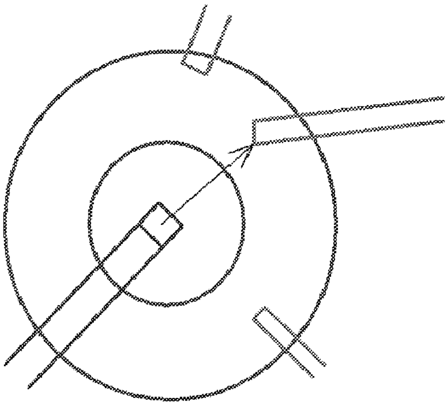

Figure 2. The sensitive function “*search*” gives back the distance to the nearest neighbour shoot (with the exception of the mother shoot itself). If this value falls below a certain threshold, the rewriting process can switch to another rule, e.g. inhibiting further growth, or initiating the growth of a substitute shoot elsewhere.

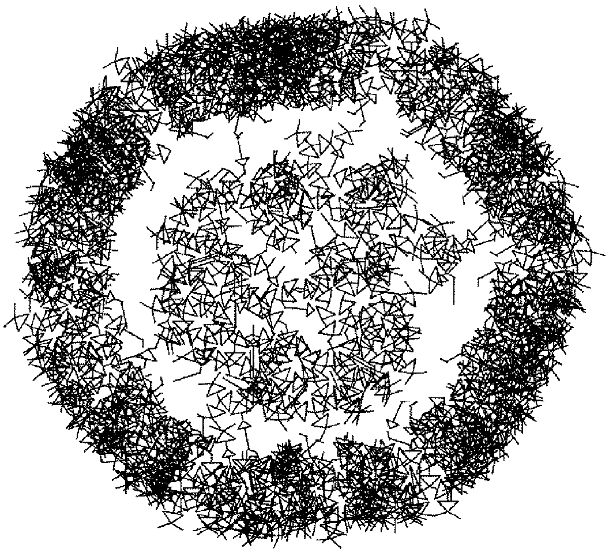

Figure 3. Result of the application of the sensitive growth grammar “clump” (*table I*), obtained with the GROGRA software and resembling Bell’s [2] original simulation. Sites of living plants are symbolized by triangles, rhizome segments by lines.

where *rad* is the threshold radius (cf. *figure 2*), ***m*** the meristem in its normal state and ***msub*** the meristem producing the substitute shoot in the next step.

Although the model description in [2] was rather incomplete compared with that of the previous model in [4], it was possible with some additional information from the author of the model to reconstruct the colonizing behaviour of both plant types discussed in [2] with growth grammars. *Figure 3* shows a colony of “clump” plants, grown from one individual after 20 steps and exhibiting an emergent “witches’ ring” structure, which resembles very much the picture in [2].

Although the mechanism of avoidance realized in this simple model is not “biological” in its functioning, similar mechanisms can serve as phenomenological rules for stand development and competition, also in tree stands [24, 28]. Just to give an impression of the amount of information needed for Bell’s “clump” model, we give the complete growth grammar (readily interpretable by the GROGRA software) in *table I.* Detailed explanations of the used symbolic notations can be found in [21] or [24].

The most refined example which was implemented by Bell with the help of his SEED3 system was a model of growth and carbon transport in clover (*Trifolium repens*) [3]. Here, casting of shadows and translocation of “packages” of assimilates in the plant were simulated.

Table I. The sensitive growth grammar “clump”, simulating the spread of a rhizomatous model plant according to [2].

```
/* declarations: */
\const alpha1 72,               /* branching angles */
\const alpha2 60,
\const beta 60,
\var sw distribution 0.5 0 0.5, /* value 0 or 2 with equal prob */
\var search function 2 2,       /* function checking occupation */
\const rad 3,                   /* radius for search */
\const rot_age 5,               /* age when rhizome begins to rot */
\const co 14,                   /* colour specification */

/* developmental rules: */
* → P(co) RL90 m,    /* start rule */
m ⇒ Pl1 F(8),        /* fictitious shoot for sensing occupation */
msub(p) ⇒ Pl1 F(5) f(-1) RU(p*beta) Pl1 F(16),
(search(0, co) <= rad) m → msub(sw-1),        /* m = meristem */
m → p(0) [ RU(alpha1) m ] [ RH180 RU(alpha2) m ] ?0.5,
m → p(0) [ RU(alpha2) m ] [ RH180 RU(alpha1) m ] ?0.5,
(search(0, co) <= rad) msub(p) → ,
msub(p) → psub(0, p) m,  /* msub = substitution state of meristem */
p(t) → p(t+1),           /* p = plant */
psub(t, p) → psub(t+1, p),  /* psub = substitute plant */

/* graphical rules: */
(t < rot_age)   p(t) ⇒ F(5) [ RU90 F(6) S(1) ] [ RU-90 F(6) S(2) ]
                       f(6) S(3) [ C(1,3) C(2,3) ], /* triangle */
(t >= rot_age) p(t) ⇒ f(11),  /* rotted (invisible) shoot */
(t < rot_age)   psub(t,p) ⇒   F(5) f(-1) RU(p*beta) F(13)
                              [ RU90  F(4) S(1) ] [ RU-90 F(4) S(2) ]
                              f(4) S(3)[ C(1,3) C(2,3) ],
                              /* small triangle */
(t >= rot_age) psub(t,p) ⇒    f(4) RU(p*beta) f(17)
                              /* rotted shoot */
```

However, rather artificial constructions and several hundreds of lines in the original data format of SEED3 were necessary to realize such mechanisms with nothing more than the simple types of interaction and specification discussed so far. We will see in example 4 that more efficient formal tools are at hand now to specify such a kind of model (cf. also the L-system examples in [31]).

4. EXAMPLE 3: PAGES & KERVELLA (1990)

The root growth model of Pagès and Kervella [34], based on morphometric data, was used to simulate a variety of diverse root architectures (cf. also [33]). A similar model was later exploited for numerical calculations of water uptake from the soil [9, 10]. The model contains stochastic components, geotropism, differentiation according to branching order, and a determination of branching directions by the number of vascular bundles in the mother axis. Here we concentrate again on one feature which makes the representation of the model structure by a growth grammar a non-trivial issue: Among the model parameters, which are to be specified for each branching order, are the length *laz* of the *apical non-branching zone* (which is a morphological characteristic of root tips), the distance *dpr* between consecutive lateral roots, and the length of the growth increment in each step, determining the speed of growth of the respective axis (*figure 4*).

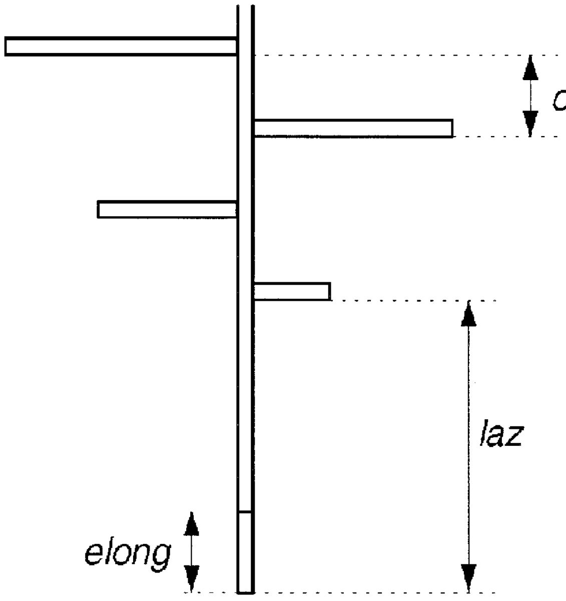

Figure 4. Basic morphometric parameters of Pagès’ & Kervella’s root growth model. *laz*: length of apical non-branching zone, *dpr*: distance between consecutive lateral roots, *elong*: elongation rate (after [34]). These numerical parameters are not necessarily multiples of each other.

The crucial point is that *laz* and *dpr* can (theoretically) have arbitrary values which are not necessarily multiples of the elongation rate *elong*. Hence a grammar based on symbols standing for growth units of length *elong* cannot easily reproduce the model structure. However, an approximate solution is quickly obtained when the whole root system is discretized into small units of fixed length *dl*, all length parameters of the model being assumed as multiples of this smallest length unit. At first sight, this seems to introduce a merely technical complication into the model, making it less transparent. But nevertheless, the model structure can be reproduced almost directly if we introduce symbols ***az*** (for the apical non-branching zone) and ***bseg*** (for a branched segment), and if we rewrite these symbols into a multiplicity of elementary units using a repetition operator, denoted by “&”,

followed by the repetition factor in parentheses and the part to be iterated in < ... >; e.g.:

$$\boldsymbol{az}(l) \Rightarrow \&(l/dl) < \boldsymbol{F}(dl) > \qquad (*)$$

(here, $\boldsymbol{F}(dl)$ generates a unit of the standard length dl). Stated in this simple way, the right-hand side of (*) is equivalent to $\boldsymbol{F}(l)$, but when we additionally insert a rotation operator in the iterated part, an (approximately) smooth curvature of the root can be obtained.

However, this rule must not be applied in the rewriting process leading to the description of the next developmental step of the root, since ***az*** should survive and will be used in the next step again – a rule rewriting ***az*** for the next step could e.g. be $\boldsymbol{az}(l) \rightarrow \boldsymbol{az}(l + elong)$, which is simple to understand and reflects the basic structure of the growth process in the original model. The rule (*) above is therefore only to be applied for *interpretation* of ***az*** (which precedes the generation of the 3D structure corresponding to the string), not for the generation of the next string. Rules of this kind are called *interpretive rules* and are distinguished in our notation by a double arrow. *Figure 5* summarizes the application of generative (normal) rules and interpretive rules of a growth grammar in a general scheme. A variant of interpretive rules has already been used in [15] for non-biological applications. In [30] and [12] they were called “homomorphisms”, but this terminology should be dropped because the generative rules can also be considered as homomorphisms in the mathematical sense of the term.

Using this technique, it is possible to reproduce growth patterns from the original paper [34] quite easily (*figure 6*). Only some parameters in the declaration part

$$\begin{tikzcd}
\alpha \arrow[r] & \sigma_1 \arrow[r] \arrow[d, Rightarrow] & \sigma_2 \arrow[r] \arrow[d, Rightarrow] & \sigma_3 \arrow[r] \arrow[d, Rightarrow] & \cdots \\
 & \sigma'_1 \arrow[d] & \sigma'_2 \arrow[d] & \sigma'_3 \arrow[d] & \cdots \\
 & S_1 & S_2 & S_3 & \cdots
\end{tikzcd}$$

Figure 5 (from [21]). Way of functioning of a growth grammar with two classes of rules, generative rules (application: horizontal arrows) and interpretive rules (vertical double arrows). The interpretive rules have no influence on the development of the structure in time. α: start word, σ_i : string giving a symbolic description of the structure S_i at step i of its development in time, σ_i': string giving a more explicit description of S_i.

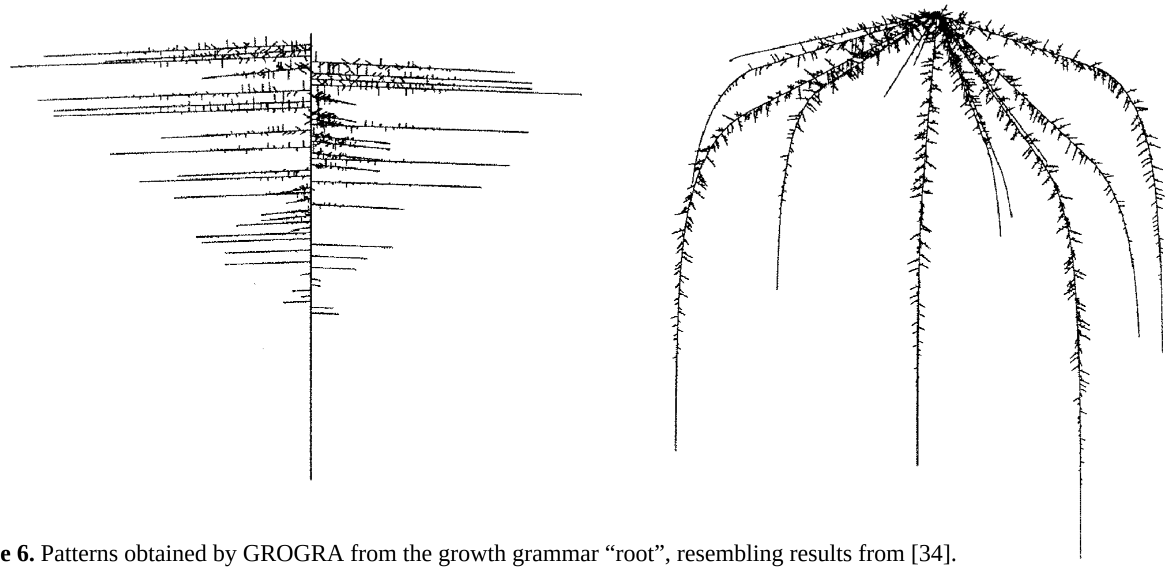

Figure 6. Patterns obtained by GROGRA from the growth grammar “root”, resembling results from [34].

of the growth grammar need to be changed to switch from the left pattern to the right one.

We show again the complete grammar (with the declarations for the example pattern shown in the left-hand part of *figure 6*, alternative parameter values for the right-hand part are indicated in the comments), just to give an impression of the amount of necessary information (*table II*).

Table II. The nonsensitive growth grammar “root”, simulating root growth approximately according to [34].

```
\const idiam 1,       /* initial diameter of all segments, | alt.: 2 */
\const ddiam 0,       /* diameter increment of shoot | 0.6     */
\const c 0.2,         /* delimits root-emerging proportion of shoot  */
\var ia0 uniform 30 80,/* insertion angle of main roots at shoot     */
\const rhy 1000,      /* rhythm of emergence of main roots | 1  */
\const dl 2,          /* discrete length unit of all axes          */
\const omax 3,        /* maximal branching order     */
\const mainax 1,      /* existence of central axis | 0  */
\var amax table 200 200 200 6,
              /* max. meristem age (per order) | 200 200 6 3           */
\var elong table 15 60 80 4,
              /* constant length increment | 15 60 6 2                 */
\var laz table 400 400 120,
              /* length of apical unbranched zone | 160 160 10 2       */
\var dpr table 20 16 20,  /* distance between branches | 20 14 6     */
\var nxp table 100 12 6,  /* number of xylem poles | 100 5 3         */
\var ia table 80 88,      /* insertion angle | 70 70 70 90           */
\var vt table 0,          /* vertical tendency | 0 0.01              */
\var lstiff table 100000, /* length of undeformed zone|600 600 0 */
\var rnd uniform 0 360,   /* random rotation */
\var i index,
\axiom * 1-20,            /* start word | 1-32 */

* → D(idiam) dd(0) L(elong(0)) RL180 Pl4 F mainrts(0,0)
      [ RH(rnd) tip(if(mainax>0,1,omax+1)) ],
dd(t) → dd(t+1),
(r=rhy) mainrts(t,r) → [ @(5*(1-c)/(6+t)+c) RH(45+t*137.5)
                          RL(ia0) tip(1) ] mainrts(t+1,0),
mainrts(t,r) → mainrts(t,r+1),
(o<=omax) tip(o) → az(0,o,elong(o), floor(lstiff(o)/dl)),
(l <  laz(o)+dpr(o) && t < amax(o))   az(t,o,l,s) →
                                      az(t+1,o,l+elong(o),s),
(l >= laz(o)+dpr(o) && t < amax(o))  az(t,o,l,s) →
            &((l-laz(o))/dpr(o))
            < bseg(0,o,dpr(o),s-i*floor(dpr(o)/dl))
              [ RH(floor(rnd*nxp(o)/360)*(360/nxp(o)))
                RL(ia(o)) E(1) < tip(o+1) > ] >
            az(t+1, o, l+elong(o)-floor((l-laz(o))/dpr(o))*dpr(o),
               s-floor((l-laz(o))/dpr(o))*floor(dpr(o)/dl)),
(l <  laz(o)+dpr(o)   && t >= amax(o)) az(t,o,l,s) → az(t+1,o,l,s),
(l >= laz(o)+dpr(o)  && t >= amax(o)) az(t,o,l,s) →
            &((l-laz(o))/dpr(o))
            < bseg(0,o,dpr(o),s-i*floor(dpr(o)/dl)) >
            az(t+1, o, l-floor((l-laz(o))/dpr(o))*dpr(o),
            s-floor((l-laz(o))/dpr(o))*floor(dpr(o)/dl)),

/* interpretive rules: */
dd(t) ⇒ Dl+(t*ddiam),          /* diameter increment of aerial part  */
az(t,o,l,s) ⇒ P14 &(l/dl) < RV(if(i<s,0,vt(o))) F(dl) >,
                                                          /* apical zone */
bseg(t,o,l,s) ⇒ P(3+o) &(l/dl) < RV(if(i<s,0,vt(o))) F(dl) >
                                                     /* branched segment */
```

5. EXAMPLE 4: TAKENAKA (1994)

The tree growth model by Akio Takenaka [45] represents a tree as a collection of woody axial elements and spherical leaf clusters. The gross photosynthetic production of a leaf cluster is assumed to be proportional to the intercepted amount of photosynthetically-active radiation (PAR). After subtracting a cost for maintenance, the resulting net gain of assimilates is utilized in the next growth step for the construction of new axial elements and leaf clusters. Following the assumption of branch autonomy, this surplus is used only for shoots emerging from the shoot where it was obtained. No secondary growth or consumption by roots is considered in the model. – The architectural pattern of the modelled monopodial tree follows a simple ternary/quaternary branching scheme taken from an earlier model [11], modified only by a dependence of number and length of new shoots upon the amount of locally available photosynthates, and the grammar translation of this pattern is straightforward. The only difficulty for the grammar formalism lies in the treatment of PAR transmittance and interception. In [45], the PAR is assumed to emerge from a discretized hemisphere consisting of 20 sectors with brightnesses according to the model of the “standard overcast sky” [32], and from each of the sectors, a ray is followed ending in the potentially-absorbing leaf cluster under consideration and possibly crossing other leaf clusters, reducing its intensity (*figure 7*).

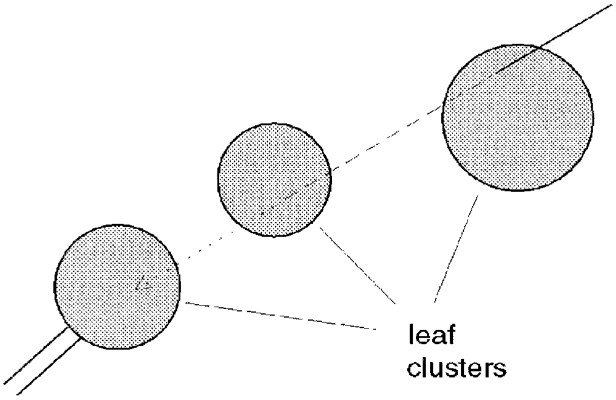

Figure 7. Light absorption by a leaf cluster in Takenaka’s model [45]. Competing leaf clusters can reduce the amount of available light from the sky sector under consideration.

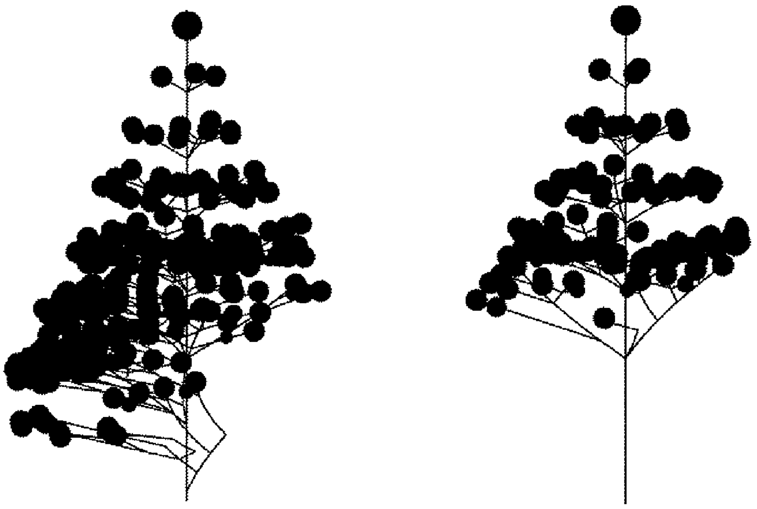

Figure 8. Trees resulting from the GROGRA realization of Takenaka's model [45] after 8 annual steps, taken from a stand of 5 × 5 trees planted in a square pattern. Left: Tree from the edge of the stand, right: central tree. The filled circles represent the leaf clusters.

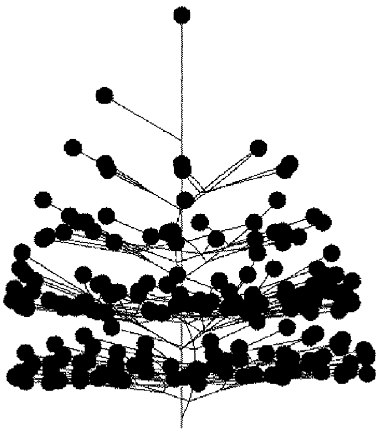

Figure 9. Tree generated by GROGRA according to the "equally sharing" version of Takenaka's model [45]. Like in *figure 8* (right side), the tree is taken from the center of a stand of 25 trees. No visible local competition effect on shape occurs, in contrast to the branch-autonomy version of the model.

This mechanism is represented in the grammar in the form of a sensitive function *npp* which calculates the amount of intercepted PAR and gives back, after subtracting a respiration term, the net photosynthetic production of the leaf cluster ***lc***. This leaf cluster is "loaded" by applying the rule

$$\boldsymbol{lc}(x) \to \boldsymbol{lc}(npp).$$

The only change made in comparison with the original model [45], which was done for technical reasons, was the replacement of the 20-sector sky by a 46-sector discretization of the hemisphere which was already used in other models [5, 7]. – In later steps of the grammar application, the obtained assimilates are distributed iteratively to new shoots, each one first satisfying its own demand and then leaving the rest to the next one, until the assimilates are exhausted or the maximal number of new shoots is reached. This mechanism was realized in the grammar again by using an expand operator like in Example 1. – The light absorption algorithm, together with a pruning rule for branches with insufficient production, leads to the emergence of competition effects when tree stands are modelled, as was already observed in [45] (*figure 8*).

In [45], Takenaka introduced also an alternative variant of his model which used an equal sharing of assimilates among all potentially-growing new shoots of the whole tree, replacing the branch-autonomy principle. For the grammar form of this model version, it was necessary to use *local register variables* (one for each tree, representing the carbon pool of the whole plant) and an *arithmetical-structural operator* which sums up all obtained assimilates in the global pool of the tree in each step. These formal concepts, which are again extensions of the classical L-system formalism, were introduced in [26] and are explained in full detail in [23, 24]. (The "local variables" in [37] have a different, more technical meaning.) The resulting trees – *figure 9* shows again the central one out of a stand of 25 trees – show a quite different habit and lack the adaptation of their leaf mass distribution to the local light environment.

To give again an impression of the size of the necessary formal description in our grammar language, we show in *table III* the growth grammar specification of the "equally sharing" version of the model. (A listing of the other version can be found in [24].) Note that the calculation of PAR transmittance and absorption is “hidden” in the sensitive function named “*npp*”. — Mech and Prusinkiewicz [31] did also present an L-system based on ideas from Takenaka’s model, but their model deviated considerably from the original one.

Table III. The sensitive growth grammar “takenaka_sharing”, simulating the version of the model from [45] which distributes the assimilates evenly.

```
\const Nmax 3,       /* maximal number of daughter shoots  */
\const Nmax0 4,      /* same for the stem axis (order 0)   */
\const lwpbuw 0.5,   /* leaf weight per branch unit weight */
\const stlpstw 200,  /* stem length per stem weight */
\const larpw 10000,  /* leaf area per weight */
\const sconst 1.5,   /* parameter for leaf density in cluster */
\const smin 0.3,     /* min. amount of assim. for shoot growth */
\const smax 1.0,     /* max. amount of assimilates for one shoot */
\const smax0 2.0,    /* same for order 0 */
\const leafcolor 10,
\var de depth,
\var an table 5 -50 60,    /* branching angle */
\var ro uniform 0 360,     /* random rotation */
\var npp function 20 1,    /* sensitive function
                              for PAR capture and npp  */
\const start 0.3,    /* start reserve of assimilates */
\var lwgh n_value,   /* leaf weight */
\var carb carbon,    /* locally available amount of assimilates */
\var col color,
\var pool local 0,   /* assimilate pool of plant */
\var nb local 1,     /* number of leaf clusters  */
* → Pl0 F(0) K(pool) K(nb) z0 s(0,start,0) lc(0,start,0,0),
z0 → A(pool, sum(carb)) A(nb, sum(col>9 && col<13, 1)),
                                                  /* assignments */
/* assimilative step: */
(de=0 && j<2) lc(t,p,a,j) → lc(t,p,npp(leafcolor),j+2),
(de=0 && j=0 && sum(lwgh)<=0) s(t,p,j) → %,
(de=0 && j=0) s(t,p,j) → s(t,p,1),
/* growth step: */
(de=0 && j=1) s(t,p,j) → s(t+1,p,0),
(de=0 && j=2) lc(t,p,a,j) → E(Nmax0) < rrt m0(pool/nb,0) >,
                                                       /* stem */
(de=0 && j=3) lc(t,p,a,j) → E(Nmax) < lrt $ m(pool/nb,0) >,
                                                   /* laterals */
(de=0) m0(p,t) →,
(de=0) ml(p,t) →,
(de=0) m(p,t) →,
(p < smax0  && p>=smin) m0(p,t) → [ s(0,p,0) lc(0,p,0,0) ],
(p >=smax0)             m0(p,t) → [ s(0,smax0,0) lc(0,smax0,0,0) ]
                                    ml(p-smax0, t+1),
(p < smax   && p>=smin) ml(p,t) → [ RU60 s(0,p,0) lc(0,p,0,1) ],
(p >=smax)              ml(p,t) → [ RU60 s(0,smax,0)
                                      lc(0,smax,0,1) ]
                                    vrt ml(p-smax, t+1),
(p < smax   && p>=smin) m(p,t)  → [ RU(an(t)) s(0,p,0)
                                      lc(0,p,0,1) ],
(p >=smax)              m(p,t)  → [ RU(an(t)) s(0,smax,0)
                                      lc(0,smax,0,1) ]
                                    m(p-smax, t+1),
rrt → RH(ro),
vrt → RH135,
lrt → RH180 ?0.5,
lrt → ?0.5,
/* interpretive rules: */
s(t,p,j)    ⇒ P(if(t=0,14,15)) L(stlpstw*(1.-lwpbuw)*p) C(0) F,
lc(t,p,a,j) ⇒ P(if(j<2,leafcolor,12)) N(lwpbuw*p)
                 Dl(sconst*sqrt(larpw*lwpbuw*p/3.14))
                 Cl(if(j<2,p,a)) F(0)
```

6. DISCUSSION

The models which were taken here from the literature and translated into growth grammars belong certainly to the simpler ones in the realm of functional-structural plant models. The functional part was only weakly developed, or missing completely. Only Takenaka’s model contained a carbon budget, but important processes like secondary growth are still omitted.

However, it was felt that it makes sense to start the process of rebuilding models in the grammar language with simple examples, and then to proceed to more advanced ones. In this way, experiences gained from the first examples can be used for the more complicated models which are attacked later. E.g., we were able to reuse the tool of the “expand operator”, introduced to manage the first example from Bell, for handling assimilate distribution in the grammar version of Takenaka’s model.

Advanced and complex models like LIGNUM [35], TRAGIC++ [19], INCA [29], AMAPsim [1] and AMAPpara [44] are not only characterized by a greater degree of genericness (ability to simulate a multitude of species under a variety of different conditions etc.), but they have normally also undergone a kind of “historical development” which makes the reconstruction nontrivial. E.g., some features were tailored to computer platforms or systems which are now outdated; several software developers have left some remnants of their sometimes differing philosophies of programming. However, the possibility to express at least the earlier, more structurally-oriented versions of the AMAP software [43] in terms of grammars was already noted rather early [13]. For the LIGNUM model [35], a “partial” realization by grammars was described in [24]. A rather complex root growth model proposed in [6] has been rebuilt with an L-system in [31], but in a simplified way. — These and other examples from the literature (see e.g. [42]) support the conclusion derived from our work that grammars are a possible general tool to specify plant models of quite different characteristics.

The advantages of a formalization of models are: a greater precision of the model specification, enhanced reproducibility and comparability of models, a better separation of the essential causalities and laws of the model from unimportant boundary conditions, and possibly the applicability of general theorems (in the case of grammars: from formal language theory, see [38] for an example). The language of growth grammars provides a markedly versatile formalism which is capable to represent stochastic as well as causal models, in contrast to more restricted mathematical tools like e.g. linear regressions. However, the formal framework can certainly *not* ensure a consistency of the models with physical or physiological principles. Due to their generality, growth grammars give a great degree of freedom, and it lies in the responsibility of the modeller to use it the right way.

Some limitations of the present growth grammars can also be stated. First of all, some parts of the syntax of the grammars can certainly be improved further to enhance transparency. A combination with interactive, graphical tools to specify the grammar rules should also be considered in the future. The whole formal framework as well as the software GROGRA [21] and other softwares working with L-systems (cf. [31]) are still being extended, and user-friendly interfacing is an important issue in this development.

A more substantial question concerns the handling of interactions of the modelled plant with the environment, and (more generally) the interfacing with process-oriented models. Is it the right way to exclude the treatment of the processes and interactions from the grammars, like we did with our “sensitive functions” in the examples 2 and 4? Potentially, with sensitive functions, or with “open L-systems” [31], all thinkable process-oriented models can be linked to a grammar. The rationale behind this concept is to separate plant growth (modelled by the grammar) from processes in the environment (taken care of in external functions or software modules). However, it may be asked whether such a borderline can always be drawn clearly. Furthermore, we give up a certain part of the advantage of transparency of the model specification brought by the grammar formalism if we exclude essential parts of the model from the grammar, making it unavoidable to refer to verbal descriptions or to lengthy computer code again if one wants to have access to the whole, precise model specification. This is the situation in the grammar-based realization of Takenaka’s model, where the important part describing PAR transmittance and absorption had to be “externalized” in a sensitive function. It should therefore be considered whether some extensions of the grammar formalism could be capable to represent more explicitely some of the calculations which some of the sensitive functions do in the present version. — On the other hand, for a number of process-oriented models, particularly those based on large systems of differential equations, a realization with classical modelling techniques will probably remain superior to the grammar-based approach, at least in terms of calculation efficiency.

A further useful extension of L-systems concerns the handling of multiscale structures, which often occur in plants (cf. [14]). The introduction of a “hierarchy counter” in growth grammars [24], i.e. an integer indicating the entrance into an entity on a higher scale, is a first step into this direction.

The development of models and of abstract formalisms for model description are interacting processes. A formalization of more advanced functional-structural plant models will lead to new improvements of the grammar machinery, and this will in turn enable an easier construction of models which will be more transparent and better to document and to compare with each other than today.

Acknowledgements: The author wishes to express his thanks to Adrian D. Bell (UCSW, Bangor) for giving insight in unpublished work, and to Gerhard Buck-Sorlin (formerly UCSW, now IPK Gatersleben) for his hospitality. I am also grateful to Branislav Sloboda at the University of Göttingen for his encouragement and support, and to Christophe Godin, Loïc Pagès, Przemyslaw Prusinkiewicz, Risto Sievänen, Akio Takenaka, and to the organizers and other participants of the “Second International Workshop on Functional-Structural Tree Models” in Clermont-Ferrand for the fruitful discussions on this workshop. Last not least I am grateful to Michael Hauhs (BITÖK, Bayreuth) who enabled the financement of a part of this work.

REFERENCES

[1] Barczi J.-F., Reffye P. de, Caraglio Y., Essai sur l’identification et la mise en œuvre des paramètres nécessaires à la simulation d’une architecture végétale. Le logiciel AMAPsim, in: Bouchon J., Reffye P. de, Barthélémy D. (Eds.), Modélisation et Simulation de l’Architecture des Végétaux, INRA, Paris, 1997, pp. 205-254.

[2] Bell A.D., Dynamic morphology: A contribution to plant population ecology, in: Dirzo R., Sarukhán J. (Eds.), Perspectives on Plant Population Ecology, Sinauer, Sunderland, 1984, pp. 48-65.

[3] Bell A.D., The simulation of branching patterns in modular organisms, Phil. Trans. Royal Soc. London B 313 (1986) 143-159.

[4] Bell A.D., Roberts D., Smith A., Branching patterns: The simulation of plant architecture, J. Theor. Biol. 81 (1979) 351-375.

[5] Blaise F., Simulation du parallélisme dans la croissance des plantes et applications, Thèse, Université Louis Pasteur, Strasbourg, 1991.

[6] Clausnitzer V., Hopmans J.W., Simultaneous modeling of transient three-dimensional root growth and soil water flow, Plant Soil 164 (1994) 299-314.

[7] den Dulk J.A., The interpretation of remote sensing: a feasibility study, Dissertation, University of Wageningen, 1989.

[8] Diaz-Ambrona C.H., Tarquis A.M., Minguez M.I., Faba bean canopy modelling with a parametric open L-system: a comparison with the Monsi and Saeki model, Field Crops Res. 58 (1998) 1-13.

[9] Doussan C., Pagès L., Vercambre G., Modelling of the hydraulic architecture of root systems: An integrated approach to water absorption – Model description, Ann. Bot. 81 (1998) 213-223.

[10] Doussan C., Vercambre G., Pagès L., Modelling of the hydraulic architecture of root systems: an integrated approach – Distribution of axial and radial conductances in maize, Ann. Bot. 81 (1998) 225-232.

[11] Fisher J.B., Honda H., Computer simulation of branching pattern and geometry in Terminalia (Combretaceae), a tropical tree, Bot. Gaz. 138 (1977) 377-384.

[12] Fournier C., Andrieu B., A 3D architectural and process-based model of maize development, Ann. Bot. 81 (1998) 233-250.

[13] Françon J., Sur la modélisation informatique de l'architecture et du développement des végétaux, in: Edelin C. (Ed.), L'Arbre, Biologie et Développement, Naturalia Monspeliensia h.s., Montpellier, 1991, pp. 231-249.

[14] Godin C., Caraglio Y., A multiscale model of plant topological structures, J. Theor. Biol. 191 (1998) 1-46.

[15] Goel N.S., Rozehnal I., Some non-biological applications of L-systems, Int. J. Gen. Syst. 18 (1991) 321-405 + color plates.

[16] Goel N.S., Knox L.B., Norman J.M., From artificial life to real life: Computer simulation of plant growth, Int. J. Gen. Syst. 18 (1991) 291-319.

[17] Guédon Y., Modélisation de séquences d'événements décrivant la mise en place d'éléments botaniques, in: Bouchon J., Reffye P. de, Barthélémy D. (Eds.), Modélisation et Simulation de l'Architecture des Végétaux, INRA, Paris, 1997, pp. 187-202.

[18] Harper J., Bell A.D., The population dynamics of growth form in organisms with modular construction, in: Anderson R.M., Turner B.D., Taylor L.R. (Eds.), Population Dynamics, Blackwell, Oxford, 1979, pp. 29-52.

[19] Hauhs M., Kastner-Maresch A., Rost-Siebert K., A model relating forest growth to ecosystem-scale budgets of energy and nutrients, Ecolog. Modell. 83 (1995) 229-243.

[20] Kari L., Rozenberg G., Salomaa A., L systems, in: Rozenberg G., Salomaa A. (Eds.), Handbook of Formal Languages, Vol. 1: Word, Language, Grammar, Springer, Berlin, 1997, pp. 253-328.

[21] Kurth W., Growth grammar interpreter GROGRA 2.4 – A software tool for the 3-dimensional interpretation of stochastic, sensitive growth grammars in the context of plant modelling. Introduction and Reference Manual, Berichte des Forschungszentrums Waldökosysteme Göttingen B 38 (1994), available at http://www.uni-forst.gwdg.de/~wkurth/public.html

[22] Kurth W., Elemente einer Regelsprache zur dreidimensionalen Modellierung des Triebwachstums von Laubbäumen, in: Hempel G. (Ed.), 8. Tagung der Sektion Forstliche Biometrie und Informatik, Deutscher Verband Forstlicher Forschungsanstalten, Biotechnical Faculty, Ljubljana, 1996, pp. 174-187.

[23] Kurth W., Some new formalisms for modelling the interactions between plant architecture, competition and carbon allocation, Bayreuther Forum Ökologie 52 (1998) 53-98.

[24] Kurth W., Die Simulation der Baumarchitektur mit Wachstumsgrammatiken, Habilitation Thesis, University of Göttingen, 1998, Wissenschaftlicher Verlag Berlin, 1999.

[25] Kurth W., Lanwert D., Biometrische Grundlagen für ein dynamisches Architekturmodell der Fichte (Picea abies (L.) Karst.), Allgemeine Forst- und Jagdzeitung 166 (1995) 177-184.

[26] Kurth W., Sloboda B., Growth grammars simulating trees – an extension of L-systems incorporating local variables and sensitivity, Silva Fenn. 31 (1997) 285-295.

[27] Kurth W., Sloboda B., Tree and stand architecture and growth described by formal grammars. I. Nonsensitive trees, J. For. Sci. 45 (1999) 16-30.

[28] Kurth W., Sloboda B., Tree and stand architecture and growth described by formal grammars. II. Sensitive trees and competition, J. For. Sci. 45 (1999) 53-63.

[29] Le Dizès S., Cruiziat P., Lacointe A., Sinoquet H., Le Roux X., Balandier P., Jacquet P., A model for simulating structure-function relationships in walnut tree growth processes, Silva Fenn. 31 (1997) 313-328.

[30] Mech R., Modelling and simulation of the interaction of plants with the environment using L-systems and their extensions, Ph.D. Thesis, University of Calgary, 1997.

[31] Mech R., Prusinkiewicz P., Visual models of plants interacting with their environment, Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH 96 (1996) 397-410.

[32] Moon P., Spencer D.E., Illumination from a non-uniform sky, Illumin. Eng. 37 (1942) 707-726.

[33] Pagès L., Ariès F., SARAH : Modèle de simulation de la croissance, du développement et de l'architecture des systèmes racinaires, Agronomie 8 (1988) 889-896.

[34] Pagès L., Kervella J., Growth and development of root systems: Geometrical and structural aspects, Acta Biotheor. 38 (1990) 289-302.

[35] Perttunen J., Sievänen R., Nikinmaa E., Salminen H., Saarenmaa H., Väkevä J., LIGNUM: A tree model based on simple structural units, Ann. Bot. 77 (1996) 87-98.

[36] Prusinkiewicz P., A look at the visual modeling of plants using L-systems, in: Hofestädt R., Lengauer T., Löffler M., Schomburg D. (Eds.), Bioinformatics – Proceedings GCB'96, Springer, Berlin, Lect. Notes Comp. Sci. 1278 (1997) 11-29.

[37] Prusinkiewicz P., Hanan J., L-systems: From formalism to programming languages, in: Rozenberg G., Salomaa A. (Eds.), Lindenmayer Systems, Springer, Berlin, pp. 193-211.

[38] Prusinkiewicz P., Kari L., Subapical bracketed L-systems, L. N. Comp. Sci. 1073 (1996) 550-564.

[39] Prusinkiewicz P., Lindenmayer A., The Algorithmic Beauty of Plants, Springer, New York, 1990.

[40] Prusinkiewicz P., James M., Mech R., Synthetic topiary, Computer Graphics Proceedings, Ann. Conf. Ser., SIGGRAPH 94 (1994) 351-358.

[41] Prusinkiewicz P., Remphrey W.R., Davidson C.G., Hammel M.S., Modeling the architecture of expanding Fraxinus pennsylvanica shoots using L-systems, Can. J. Bot. 72 (1994) 701-714.

[42] Prusinkiewicz P., Hammel M., Hanan J., Mech R., Visual models of plant development, in: Rozenberg G., Salomaa A. (Eds.), Handbook of Formal Languages, Vol. 3: Beyond Words, Springer, Berlin, 1997, pp. 535-597.

[43] Reffye P. de, Edelin C., Françon J., Jaeger M., Puech C., Plant models faithful to botanical structure and development, Comp. Graph. ACM/SIGGRAPH 22 (1988) 151-158.

[44] Reffye P. de, Fourcaud T., Blaise F., Barthélémy D., Houllier F., A functional model of tree growth and tree architecture, Silva Fenn. 31 (1997) 297-311.

[45] Takenaka A., A simulation model of tree architecture development based on growth response to local light environment, J. Plant Res. 107 (1994) 321-330.