Rule-based integration of LIGNUM into GroIMP

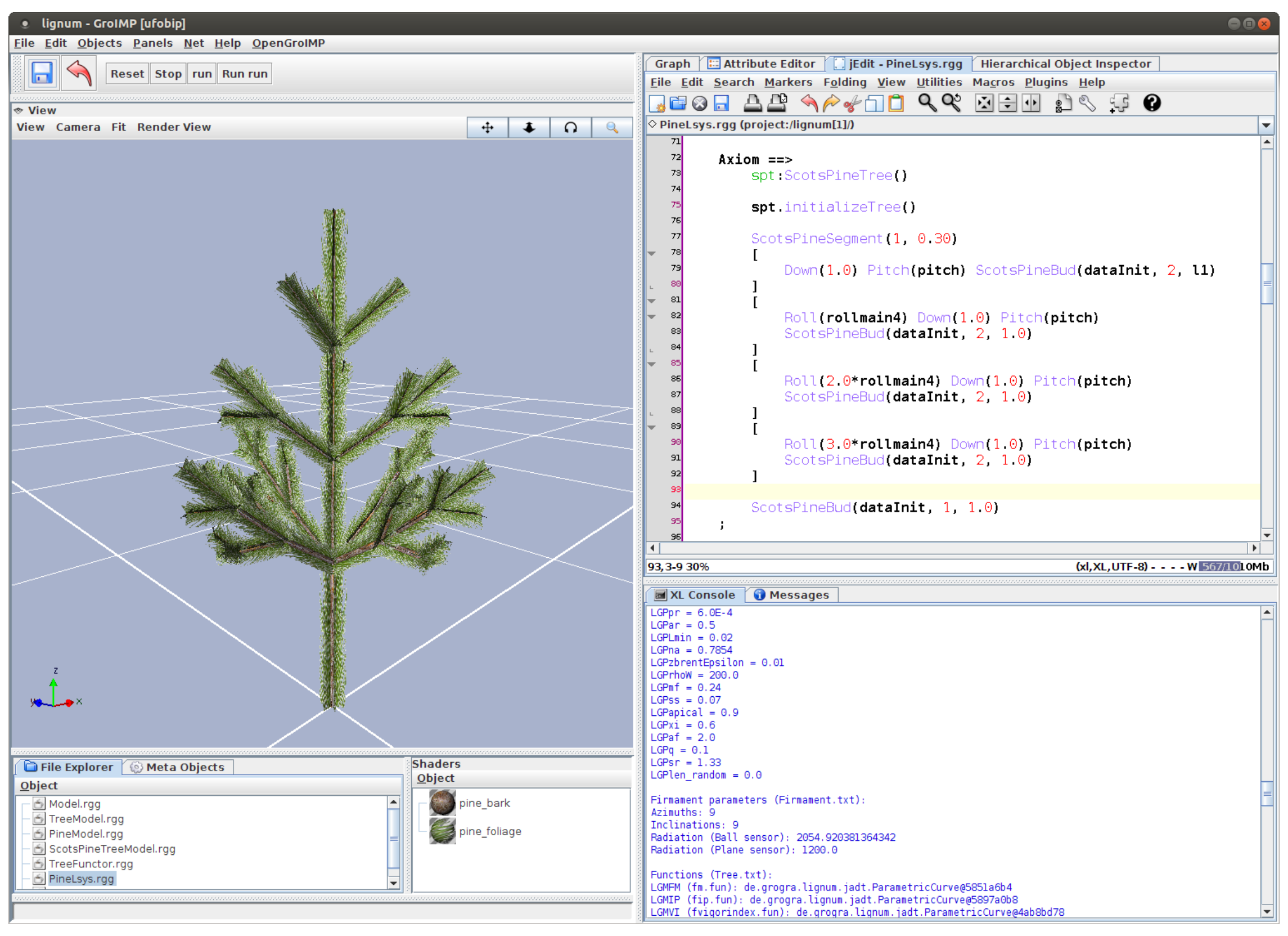
Katarína Smoleňová, Michael Henke, Yongzhi Ong, Winfried Kurth
Department Ecoinformatics, Biometrics and Forest Growth, University of Göttingen, Germany

Introduction

LIGNUM is a generic functional-structural plant model (FSPM) designed and parameterized for modelling coniferous as well as deciduous tree species. It combines the use of L-systems for structural development and the programming language C++ for modelling metabolic processes and radiation regimes. We show how both the structural and functional part of LIGNUM, applied to Scots pine (*Pinus sylvestris* L.) (Perttunen et al. 1998, Perttunen and Sievänen 2005), can be translated into the rule-based language XL (Hemmerling et al., 2008), thus offering new possibilities for model reuse and comparison.

Modelling tools

- *LIGNUM*
  - Generic FSPM, with focus on trees
  - Implemented in C++
  - Uses L-systems for architectural development, specified with the language L
  - Data structure: list
- *XL*
  - The modelling language of GroIMP
  - Extends Java with rule-based constructs
  - Data structure: graph
- *GroIMP*
  - **Gro**wth Grammar-related **I**nteractive **M**odelling **P**latform (Fig. 1), in Java



```
Axiom ==>
    spt:ScotsPineTree()

    spt.initializeTree()

    ScotsPineSegment(1, 0.30)
    [
        Down(1.0) Pitch(pitch) ScotsPineBud(dataInit, 2, l1)
    ]
    [
        Roll(rollmain4) Down(1.0) Pitch(pitch)
        ScotsPineBud(dataInit, 2, 1.0)
    ]
    [
        Roll(2.0*rollmain4) Down(1.0) Pitch(pitch)
        ScotsPineBud(dataInit, 2, 1.0)
    ]
    [
        Roll(3.0*rollmain4) Down(1.0) Pitch(pitch)
        ScotsPineBud(dataInit, 2, 1.0)
    ]

    ScotsPineBud(dataInit, 1, 1.0)
;
```

```
LGPpr = 6.0E-4
LGPar = 0.5
LGPLmin = 0.02
LGPna = 0.7854
LGPzbrentEpsilon = 0.01
LGPrhoW = 200.0
LGPmf = 0.24
LGPss = 0.07
LGPapical = 0.9
LGPxi = 0.6
LGPaf = 2.0
LGPq = 0.1
LGPsr = 1.33
LGPlen_random = 0.0

Firmament parameters (Firmament.txt):
Azimuths: 9
Inclinations: 9
Radiation (Ball sensor): 2054.920381364342
Radiation (Plane sensor): 1200.0

Functions (Tree.txt):
LGMFM (fm.fun): de.grogra.lignum.jadt.ParametricCurve@5851a6b4
LGMIP (fip.fun): de.grogra.lignum.jadt.ParametricCurve@5897a0b8
LGMVI (fvigorindex.fun): de.grogra.lignum.jadt.ParametricCurve@4ab8bd78
```

Graph
Attribute Editor
jEdit - PineLsys.rgg
Hierarchical Object Inspector
File Edit Search Markers Folding View Utilities Macros Plugins Help
PineLsys.rgg (project:/lignum[1]/)
93,3-9 30%
(xl,XL,UTF-8) - - - - W 567/1010Mb
XL Console
Messages

Fig. 1 Screenshot of the GroIMP software with the translated LIGNUM model for Scots pine.

Architecture

The following structural components of LIGNUM (implemented as C++ classes) were specified as XL modules (analogous to L-system symbols) with associated attributes and methods:

- Main structural units: *TreeSegment*, *Bud* (*BranchingPoint* and *Axis*, inevitable in LIGNUM, were inherently expressed in XL)
- Modules for trees in general: *Tree*, *TreeCompartment*
- Modules specific for conifers, pine and Scots pine: *CfTreeSegment*, *PineSegment*, *PineBud*, *ScotsPineTree*, *ScotsPineSegment*, *ScotsPineBud*

By applying L-system rules, new buds and segments are created at each time step (one year). The rules were translated in a straightforward way (Tab. 1).

Tab. 1 Comparison of the L (LIGNUM) and XL (GroIMP) syntax.

LIGNUM	GroIMP	Description
Start : { produce A(); }	*Axiom ==> A;*	Start symbol
A() : { produce B() A(); }	*A ==> B A;*	L-system rule
SB()	*[*	Branch start
EB()	*]*	Branch end

Instead of using the L symbols *F* and *B* interpreted in LIGNUM as a tree segment and a bud, respectively, we used directly the modules *ScotsPineSegment* and *ScotsPineBud*.

Additional XL modules were specified, corresponding to L symbols, to control branch orientation: *Pitch*, *Roll*, *Turn*, *Down*, *HDir*.

Radiation regime

Two LIGNUM light models (segment and voxel space based) as well as the sky model were translated directly into GroIMP framework code, therefore hidden from the user. They can be started by simple method calls from XL code.

Metabolism

Four generic algorithms were identified in LIGNUM, used to traverse the modelled tree and to perform functional computations:

- *ForEach* - applies a defined functor to each tree component
- *Accumulate* - collects information from the tree
- *AccumulateDown* - traverses the tree from the branch tips to the tree base (and collects/passes information)
- *PropagateUp* - traverses the tree from the base to the tips

These were translated using selected features of the language XL (Tab. 2).

Tab. 2 Alternatives to LIGNUM's generic algorithms in XL.

LIGNUM	GroIMP	Example
ForEach	execution rule (::>)	*sps:ScotsPineSegment ::>* *sps.photosynthesis();*

Accumulate	aggregators (*sum*, *count*, *max*, ...), with queries ((* *))	*sum((* ScotsPineSegment *)*
		.getP())
		// sum up photosynthates

AccumulateDown
PropagateUp
	path patterns with directed relations (-*r*->, <-*r*-), with *r* = *minDescendants*, *descendants*, *ancestor*, shortcuts:
	-->, <-- = any edge,
	>, < = successor,
	+>, <+ = branch
		sps:ScotsPineSegment ::>
		if (sps.getAge() == 0) {
		sps.setQin(
		(sps -ancestor->*
		*ScotsPineSegment *)*
		.getQin()
		);
		}
		// propagate up incoming radiant flux
		// to new segments

Results

We could demonstrate that the advanced features of XL allow to translate even complex models like LIGNUM, applied to Scots pine (Fig. 2), into rule-based models in an elegant and transparent way. Further research will address a comparison of the LIGNUM and the GreenLab FSPM on the same platform, as both models have been translated into XL.

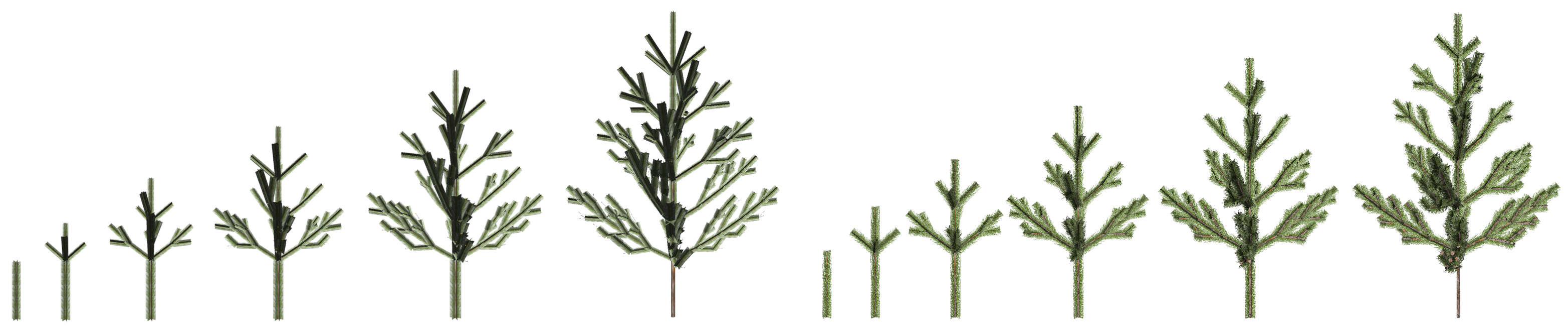


Fig. 2 Comparison of Scots pine simulations: original LIGNUM model, visualized in LignumWB (left) and its translated XL version in GroIMP (right).

Literature

- **Hemmerling R, Kniemeyer O, Lanwert D, Kurth W, Buck-Sorlin G. 2008.** The rule-based language XL and the modelling environment GroIMP illustrated with simulated tree competition. *Functional Plant Biology* **35**: 739-750.
- **Perttunen J, Sievänen R, Nikinmaa E. 1998.** LIGNUM: a model combining the structure and the functioning of trees. *Ecological Modelling* **108**: 189–198.
- **Perttunen J, Sievänen R. 2005.** Incorporating Lindenmayer systems for architectural development in a functional-structural tree model. *Ecological Modelling* **181**: 479–491.

Acknowledgement & Contact

We thank J. Perttunen and R. Sievänen for their support. A. Brinkmann translated LIGNUM's light models into GroIMP. This research was funded by the German Research Foundation (DFG) under project identifier KU 847/8-1. All support is gratefully acknowledged.

- E-Mail:
  `ksmolen@gwdg.de`
- GroIMP 1.4.2 (free, open source software):
  `http://www.grogra.de`
- Department Ecoinformatics:
  `http://www.uni-goettingen.de/en/67072.html`

FSPM 2013
7th International Conference on
FUNCTIONAL–STRUCTURAL PLANT MODELS
9–14th June 2013, Saariselkä, Finland