Electronic Notes in Theoretical Computer Science 110 (2004) 173–175 

[www.elsevier.com/locate/entcs](http://www.elsevier.com/locate/entcs)

The GTB and PAT tools

# Adrian Johnstone[1](#_bookmark0) Elizabeth Scott[2](#_bookmark0) Giorgios Economopoulos

*Department of Computer Science Royal Holloway, University of London Egham, Surrey, United Kingdom*

Abstract

The Grammar Tool Box is a toolset for manipulating Context Free Grammars and objects asso- ciated with them such as parsers, languages and derivations. GTB has three main rˆoles: as a pedagogic tool; as an experimental platform for novel algorithms and representations; and as a production tool for translator front end generation. In this tool demonstration we give an overview of GTB and its companion Java-based animator tool PAT.

*Keywords:* Parsing, software animation

GTB is an environment for maipulating grammars and producing au- tomata and parsers from them. Most of our experimental work with GTB has focussed on the development of novel GLR parsing algorithms [[4](#_bookmark4),[3](#_bookmark5)] and their comparison with existing algorithms, in particular that of Farshi [[1](#_bookmark2)].

We use the following grammar to highlight performance aspects of the algorithms:

SS::=S.

S ::= T ’b’.

T ::= A ’b’ | T T T.

A ::= T ’b’ A A A | T T ’b’ | #.

GTB can output intermediate structures in a form suitable for visualisa- tion with the VCG tool [[2](#_bookmark3)], but for many applications a dynamic animation is desirable. PAT is a parser animator which takes tables exported from GTB; constructs parse-time objects such as GSS’s and SPPF’s and then displays the

1 Email: [a.johnstone@rhul.ac.uk](mailto:a.johnstone@rhul.ac.uk)

2 Email: [e.scott@rhul.ac.uk](mailto:e.scott@rhul.ac.uk)

1571-0661 © 2004 Elsevier B.V. Open access under [CC BY-NC-ND license.](http://creativecommons.org/licenses/by-nc-nd/3.0/)

doi:10.1016/j.entcs.2004.06.003

174 *A. Johnstone et al. / Electronic Notes in Theoretical Computer Science 110 (2004) 173–175*

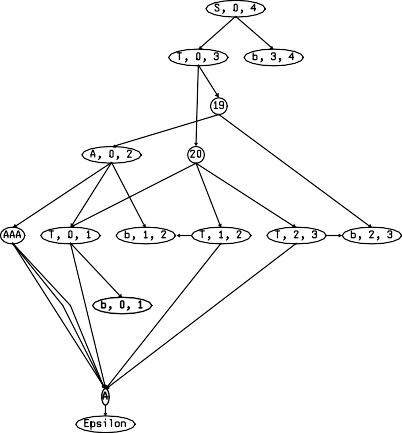
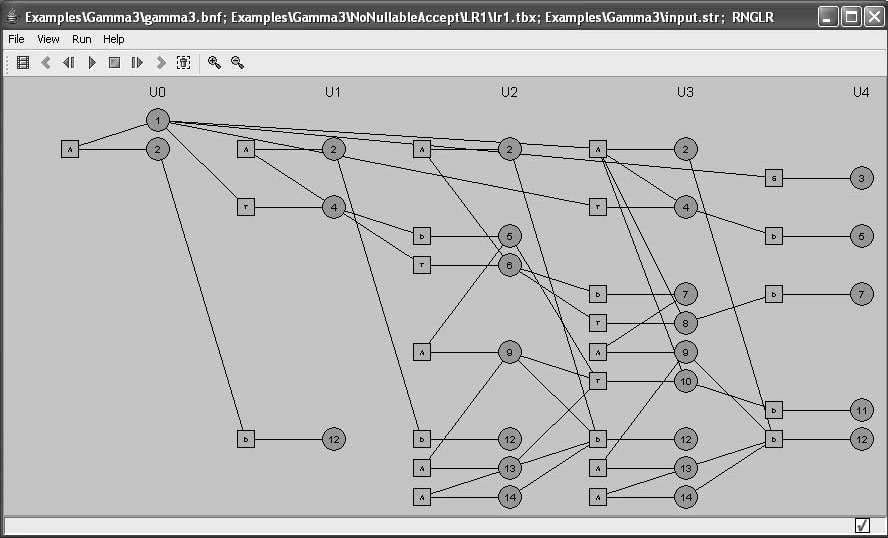


Fig. 1. GSS animation in PAT and equivalent VCG rendition

results. The user can interactively go backwards and forwards in time to see the way in which these structures are built up. We shall demonstrate anima- tions of Earley, Farshi, LR(1) RNGLR and BRNGLR algorithms: Figure [1](#_bookmark1) shows the output from an RNGLR parse of the above grammar.

# References

1. Rahman Nozohoor-Farshi. GLR parsing for *G*-grammars. In Masaru Tomita, editor, *Generalized LR parsing*, pages 60–75. Kluwer Academic Publishers, Netherlands, 1991.
2. Georg Sander. *VCG Visualisation of Compiler Graphs*. Universit¨at des Saarlandes, 66041
   1. *Johnstone et al. / Electronic Notes in Theoretical Computer Science 110 (2004) 173–175* 175

Saarbru¨cken, Germany, February 1995.

1. E.A. Scott, A.I.C. Johnstone, and G.R. Economopoulos. BRN-table based GLR parsers. Technical Report TR-03-06, Royal Holloway, University of London, Computer Science Department, 2003.
2. Elizabeth Scott, Adrian Johnstone, and Shamsa Sadaf Hussain. Tomita-style generalised LR parsers. Technical Report TR-00-12, Royal Holloway, University of London, Computer Science Department, December 2000.