

Hansl for Dynamic Macroeconomics

Federico Giri *

February 27, 2015

Extended abstract

The aim of this paper is to evaluate the performance of Hansl computing language as instrument to solve dynamic macroeconomics problems. In order to assess Hansl's potential performance to solve this class of problems, we replicate the exercise proposed by Aruoba and Fernández-Villaverde (2014). They choose to solve the standard neoclassical growth model using several programming languages in order to rank their speed and numerical accuracy¹. Instead of the usual Lagrangian method, they decide to solve the model using the value function iteration technique which is well known for its computational burden. The general classification for the Aruoba and Fernández-Villaverde (2014) exercise is the following: *C++* and *Fortran* lead the group, the surprising novelty of *Julia* reaches the third place overtaking the most famous *Java*, an excellent *Matlab* performance, *R* and *Mathematica* sadly at the bottom of the competition.

*Address: Università Politecnica delle Marche, Piazzale Martelli n 8 Ancona, Italy.
E-mail: f.giri@univpm.it

¹The languages used are : *C++*, *Fortran*, *Julia*, *Python*, *Matlab*, *Java*, *R* and *Mathematica*

Where is Hansl placed in this imaginary horse race? Since Hansl has several appealing features to analyse the outcome of several kind of macro exercises, assessing its performance in terms of speed, especially with respect to the other open source software, is the necessary starting point to be a credible alternative to such computing languages.

We replicate the exercise on a Window machine for a subsample of the previous computing languages ² focusing on the open source softwares and using *Matlab* as a benchmark.

The results are encouraging and presented in table 1.

Table 1: Time performances

| Programm | Time |
|----------|-------|
| Matlab | 9,95s |
| Python | 117s |
| Gretl | 646s |
| R | 700s |

Our exercise proof that *Hansl* is competitive with respect to *R* in terms of speed while for the moment *Python* remains one step ahead.

²*Matlab, Julia, Python 2.7* and *R 3.1.2*.

References

S. Borağan Aruoba and Jesús Fernández-Villaverde. A Comparison of Programming Languages in Economics. NBER Working Papers 20263, National Bureau of Economic Research, Inc, June 2014. URL <http://ideas.repec.org/p/nbr/nberwo/20263.html>.