SC/69A/ASI/16

Sub-committees/working group name: ASI

Implementation of the Transim Marine Mammal Survey Simulation Suite on a Windows System

Daniel W Smith

Implementation of the Transim Marine Mammal Survey Simulation Suite on a Windows System

Daniel W. Smith

P.O. Box 493

Woods Hole MA 02543 USA

The Transim suite of marine mammal survey simulation programs[1] was originally developed on Unix systems (first Sun and later Linux). To implement this suite efficiently on Windows systems, one needs the ability to install a gnu C++ compiler version chosen for compatibility with existing code and access to a command-line interface that will enable running the scripts which control and iterate the compiled C code.

Options to provide the needed command-line interface include Cygwin and MinGW-w64, but both seem to constrain which versions of the gnu compilers g++ and gcc can be easily chosen to accompany them. Thus, the most practical way to implement Transim under Windows within a reasonable time appears to be to capitalize on the Windows System for Linux (WSL), available under Windows versions later than Windows 10 v. 1709. While not enabled by default, this environment for running Linux can be unveiled by first enabling Developer Mode in Windows, then checking the box for WSL under "turn Windows features on or off" and restarting.[2] In the present case, WSL was activated on a Dell Latitude E6430 laptop running Windows 10 Enterprise v. 21H2, but the process should work equally well on any machine running a sufficiently recent Windows version. Ubuntu Linux 22.04.2 LTS was then set up within WSL. (On the latest versions of Windows [Windows 10 v. 2004 and higher; Windows 11], running the command "wsl –install" within the PowerShell should both enable WSL and install Ubuntu Linux.) Linux distributions (distros) other than Ubuntu are also available for WSL, e.g. from the Microsoft Store, but the distro (CentOS) that was on the machines at the Northeast Fisheries Science Center (NEFSC) during the most recent Linux update of the Transim code was evidently not available, hence the choice of Ubuntu in this case.[3]

Other items installed to enable compiling and running the simulations, since these were apparently not included in the Ubuntu distro by default, included g++ and gcc compilers, versions 4.8;[4] make (compilation command); libgsl (Gnu Scientific Library, here used for random number operations); csh, tcsh (additional command shells to enable cycling scripts to run); and bc (basic calculator). These could generally be installed straightforwardly using "sudo apt-get install [name]". The GSL installation presumably defaulted to version 2.7; though the version installed on NEFSC Linux machines in 2020 and used in subsequent Linux-only development of these simulations was most likely 2.6, this difference does not appear to have affected random-number behavior in the primary stage of output generation.

Output from the main stage of the simulation (mkroster, locatereadship_timefile and whalesim programs, invoked from cycling scripts) replicated past Linux-only output exactly when initialized with the same random number seeds, as did output from the second and final post-processing pass, though this second pass involved no random numbers.

Results from the first post-processing pass did not exactly replicate data generated in 2007, but the resulting final output, when checked against a sample past scenario (#42 from 2007 and 2020), nonetheless appeared to be substantively the same: using 77 replications of final sightings files from the old and new data, a Kolmogorov-Smirnov test comparing total sightings strongly rejected the hypothesis that they were drawn from different distributions ($P = 0.968$).[5]

Instructions for running the simulation suite under WSL on Windows remain the same as for the most recent Linux-only versions; see e.g. Appendices 1 and 2 in reference (1). Some minor alterations to the code of cycling scripts were made for the WSL case, so the updated versions of these scripts should be used.

NOTES

1. See e.g. D.W.Smith and D. Palka, 2021, "Update on modernization of visual survey simulation programs," report to Scientific Committee of the International Whaling Commission.

2. The full recipe from https://www.notebookcheck.net/How-to-run-Linux-binaries-natively-in-Windows-10.282676.0.html : "Firstly, enable **Developer Mode** by going to Settings > Update and Security > or Developers. If this is the first time Developer Mode is being turned on, you might have to wait for some time for the feature to get activated. Then, in the Cortana search box, search for **Turn Windows Features on or off** (you might require Admin [privileges] for this), scroll down and check the box against **Windows Subsystem for Linux**. You might be prompted to restart the computer for the changes to take effect."

3. It may be worth noting that even among native Linux machines, unlike most current Windows or Mac computers, executable programs may not be binary compatible between different operating system distributions (e.g. CentOS, Ubuntu, etc.).

4. In order to retrieve this earlier version of g++, it was necessary to add xenial to the end of the sources.list file per instructions found here:

https://askubuntu.com/questions/1036108/install-gcc-4-9-at-ubuntu-18-04

A "public key is not available" error resulted at first when trying to add the xenial repository as above, but this was able to be addressed by adding a key as described here:

https://chrisjean.com/fix-apt-get-update-the-following-signatures-couldnt-be-verified-because-the-public-key-is-not-available/

The keys missing seemed to be same ones cited at the link above. Adding the one starting with 4097 didn't help, but adding the one starting with 3B4FE did, and it was then possible to run "apt update" successfully.

5. The Kolmogorov-Smirnov test is recommended to assess whether two samples are likely to have been drawn from the same distribution when tests that require specific assumptions about those samples (e.g. normality) cannot be used. A. Solow, Woods Hole Oceanographic Institution, pers. comm., 2015 and after.