Parallel processing in WPS services for geological mapping

M. Alvioli,

I. Marchesini, M. Rossi, M. Santangelo, M. Cardinali,

P. Reichenbach, F. Ardizzone, F. Fiorucci,

V. Balducci, A. C. Mondini, F. Guzzetti

Research institute for Hydro-Geological Hazards

of

National Research Council

(Perugia, Italy)

PROTEZIONE CIVILE

Regione Umbria

CONTENTS

1. Purposes of Web Processing Services - WPS

- WPS is an Open Geospatial Consortium (OGC) standard for publishing geospatial processes, ***providing***:

⟶ dissemination of processing services and algorithms

⟶ flexibility of use with Python (PyWPS), GRASS GIS, bash scripting, R

⟶ easy maintainance of the code

- the main ***purposes*** for which we used WPS services are:

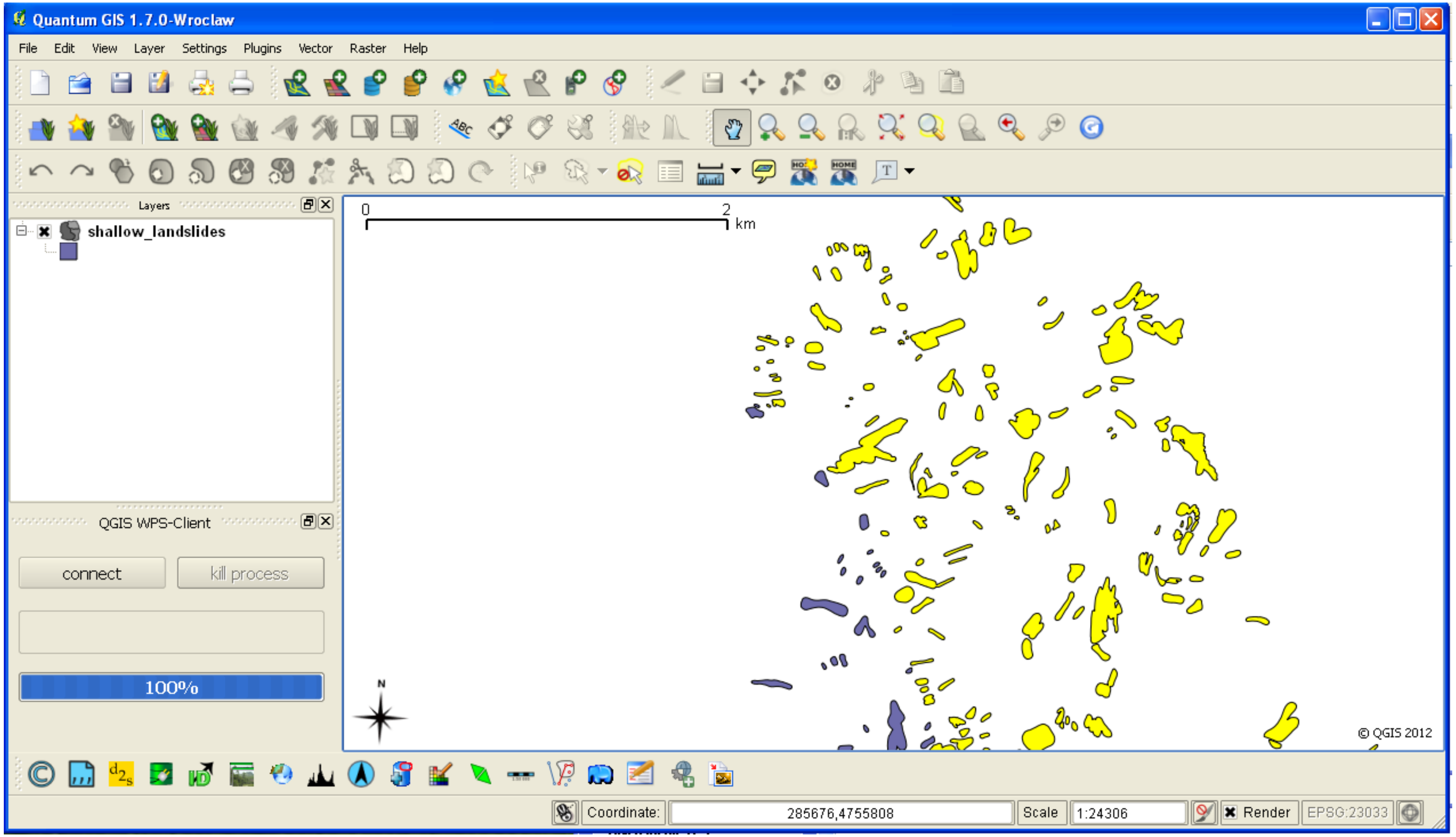
⟶ publication of a procedure on the web – without disclosing the source code

⟶ implementation of algorithms for interactive use within a collaboration

- WPS clients are found in open source projects (*i.e.* QGIS) as well as in commercial software

- computing intensive procedures can be executed on *parallel or multicore machines*

1.1 The Web Processing Services interface (in QGIS)

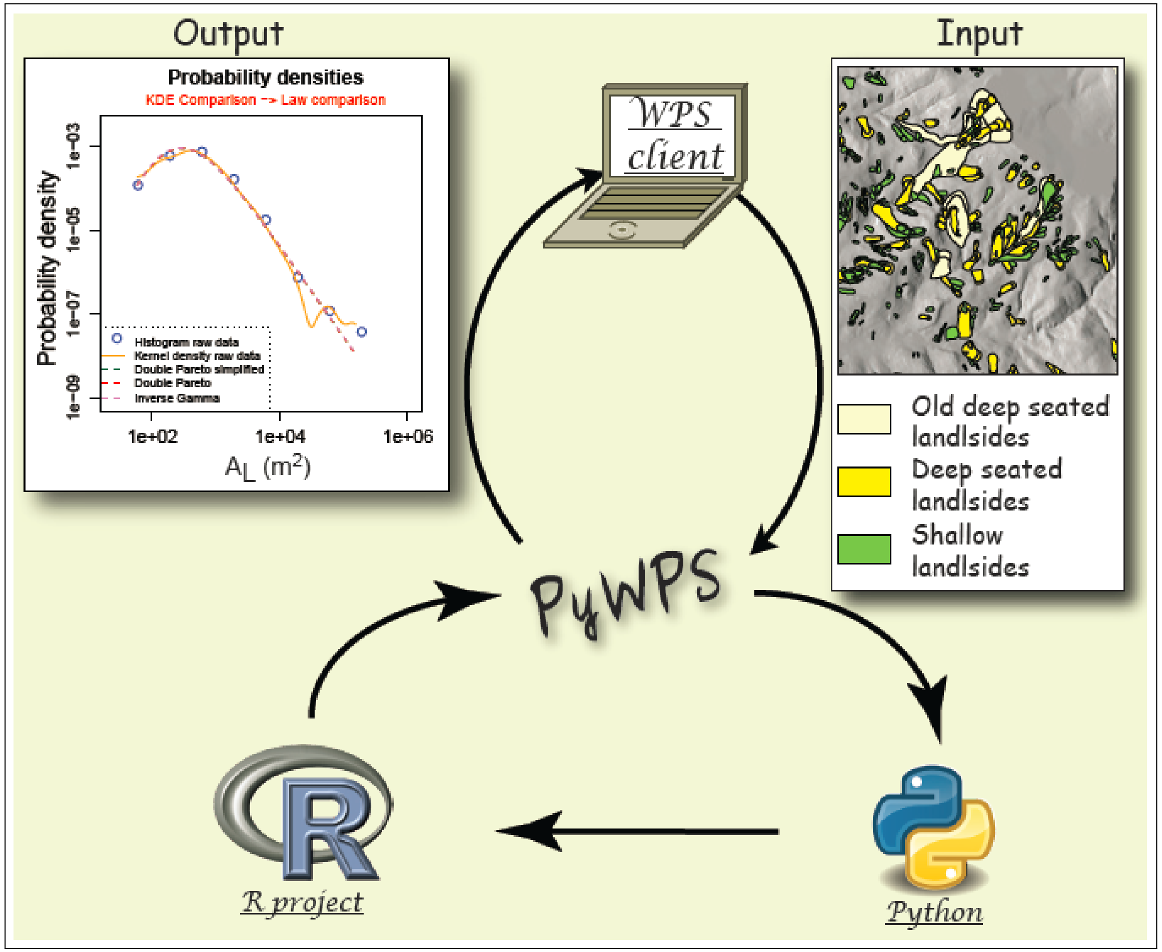


The interface to upload *maps*, specify *parameters* and *run the service*

2.1 Our algorithms implemented as WPS services

Landslide area distribution

- a tool for estimating the statistics of landslide areas, A_L
- takes as an input a landslide inventory map
- estimates are based on Histogram Density, Maximum Likelihood and Kernel Density Estimation
- the output is the frequency density of input landslides areas



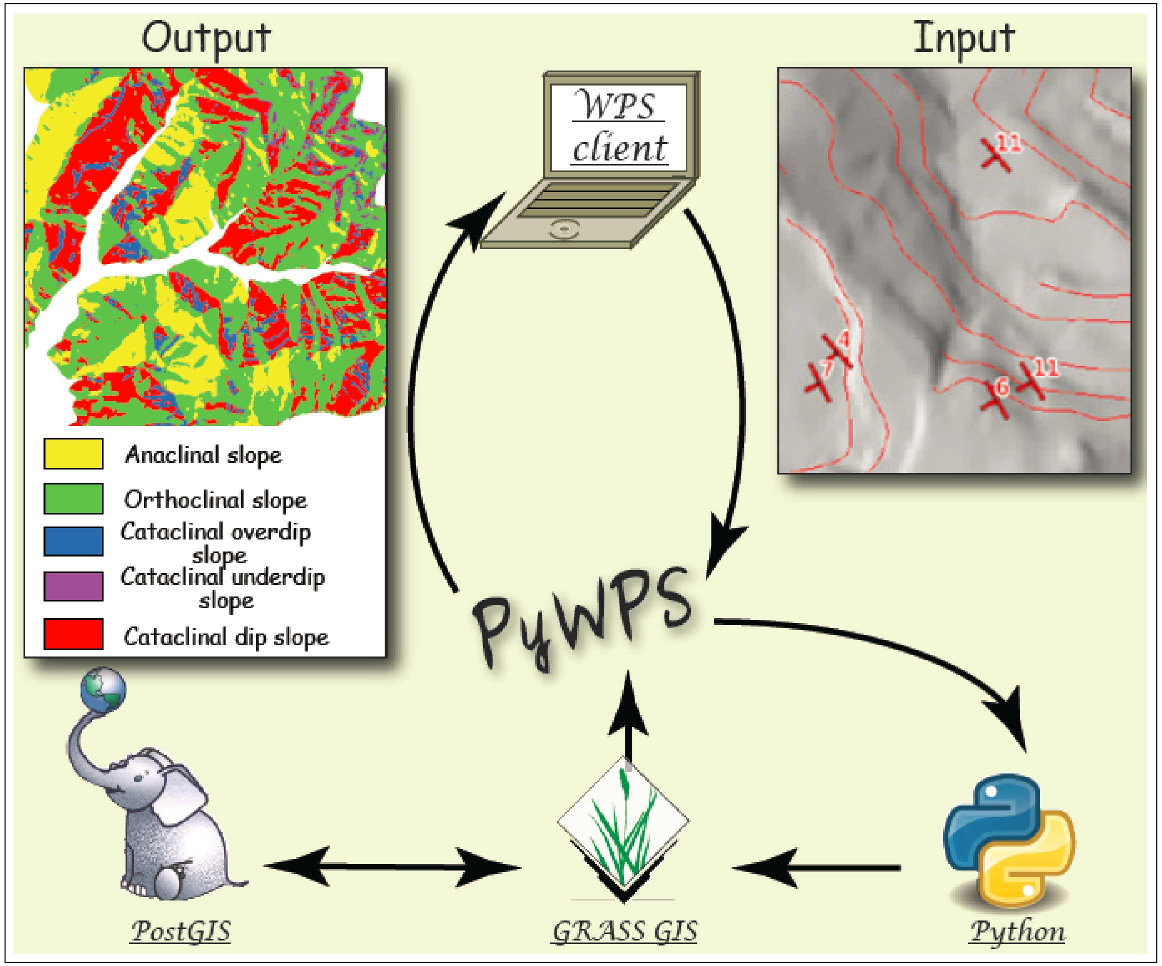
M. Rossi *et al.*, Geophys. Res. Abs. Vol. 14, EGU2012-9438-1 (2012)

see also: M. Alvioli *et al.*, `http://arxiv.org/abs/1306.1529` [physics.geo-ph]

2.2 Our algorithms implemented as WPS services

Morpho-structural domains

- interpolation of bedding planes to establish a geometrical relationship between bedding attitude and slope
- takes as an input a map of bedding attitudes
- the output is raster layer classified on five morpho-structural domains: i) anaclinal, ii) orthoclinal, iii) cataclinal over-dip iv) cataclinal under-dip v) pure cataclinal

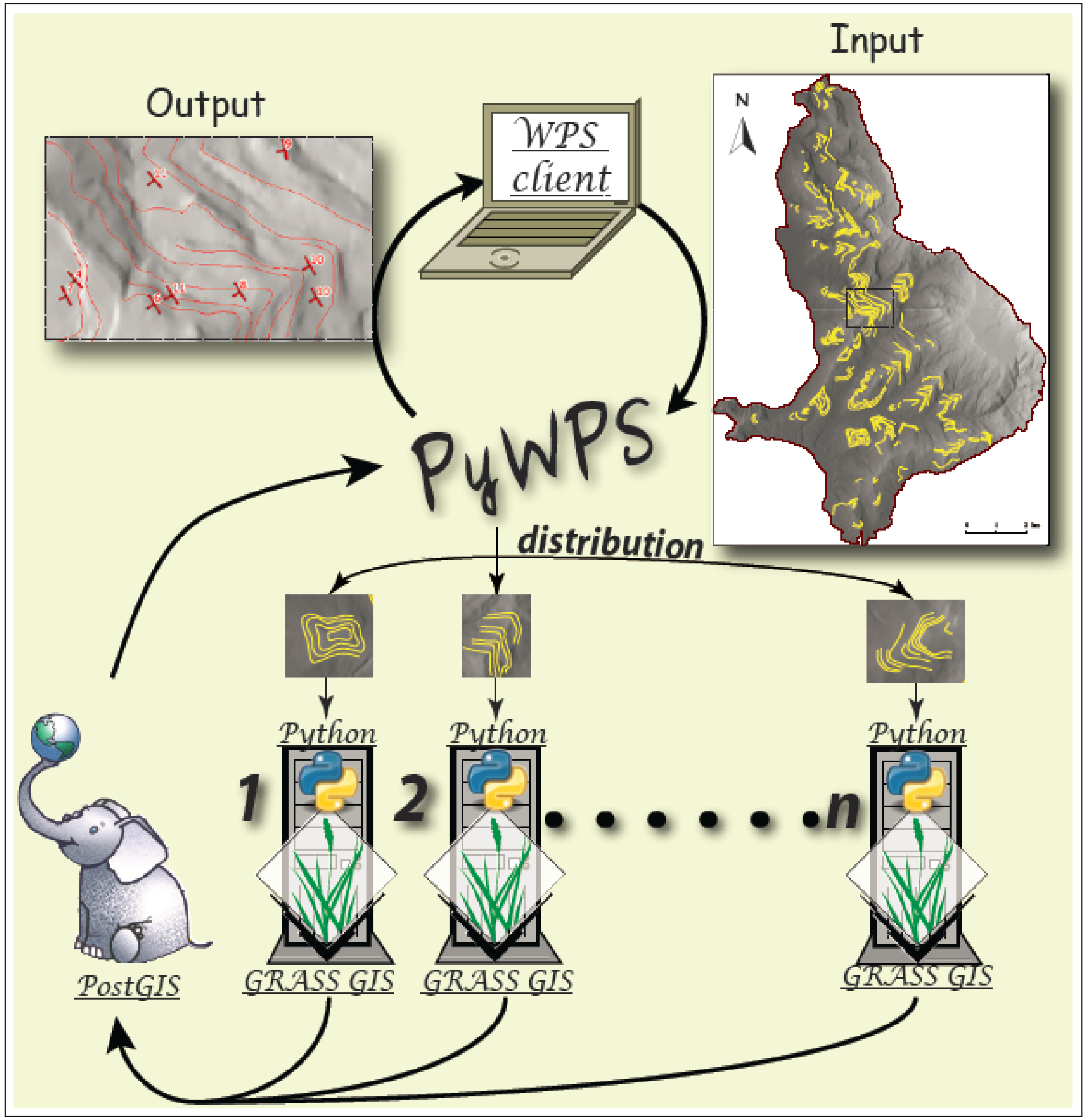


M. Santangelo *et al.*, Geophys. Res. Abs. Vol. 14, EGU2012-12457 (2012)

2.3 Our algorithms implemented as WPS services

Bedding attitude

- a tool for estimating the dip direction and angle (*bedding traces*) of bedding planes
- takes as an input a DEM and a map with intersections of bedding planes and terrain
- *processed in **parallel*** with GRASS GIS; a database collects partial (vector) maps
- the output is a vector layer with *bedding traces*

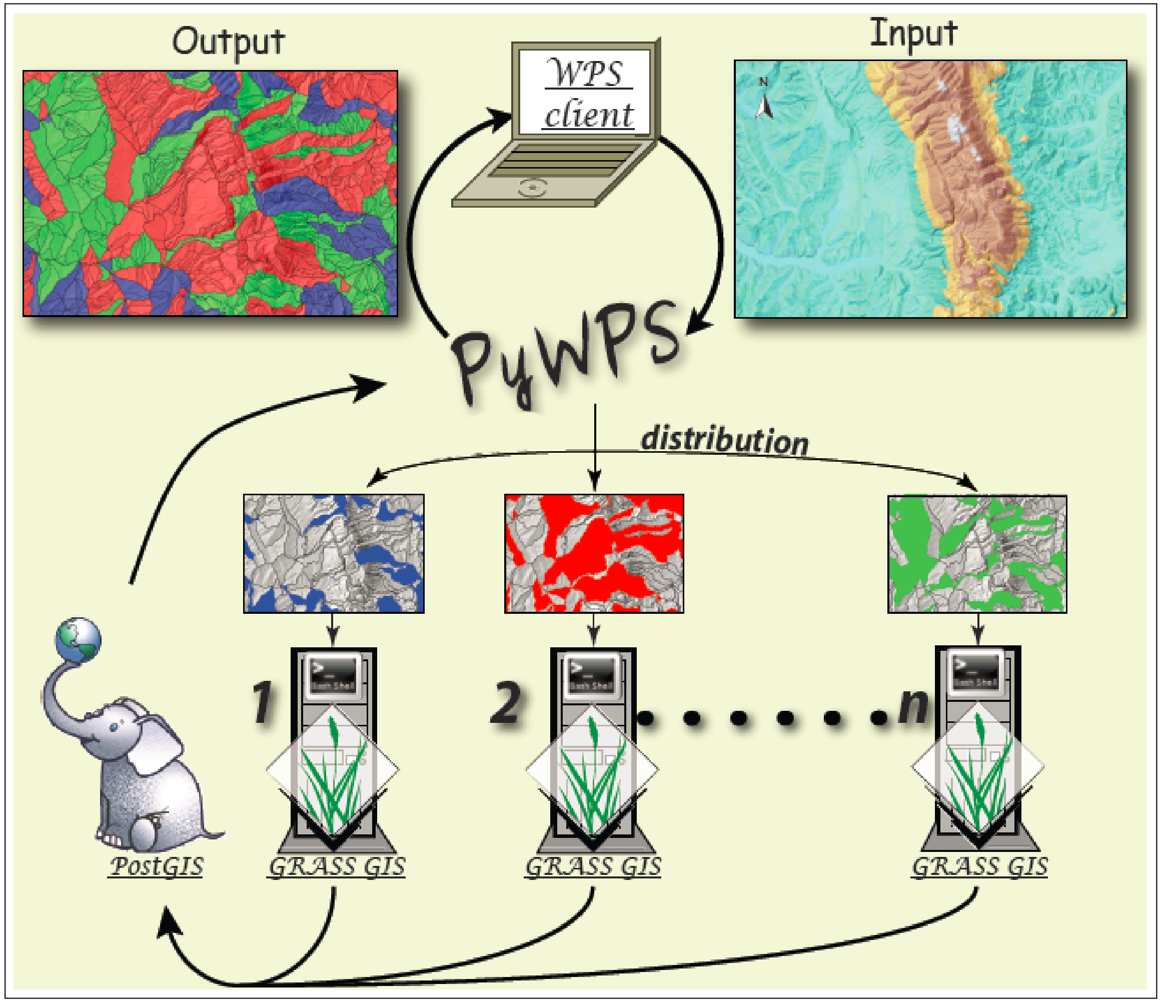


I. Marchesini *et al.*, Proceedings of 2nd World Landslide Forum, Rome (2011)

2.4 Our algorithms implemented as WPS services

Slope units delineation

- *slope units* are hydrological partitions of an area, bounded by drainage and divide lines
- takes as an input a DEM and model parameters
- *processed in* ***parallel*** with GRASS GIS and a database
- the output is a raster layer with *slope units*



A. Carrara *et al.*, Earth Proc. Surf. Land. 16, 427 (1991)

F. Guzzetti *et al.*, Geomorphology 72, 272 (2005)

3.1 Exploiting parallel processing

- parallel processing needed when we want to achieve:

$\longrightarrow$ ***reduce computing time*** of a problem of given size

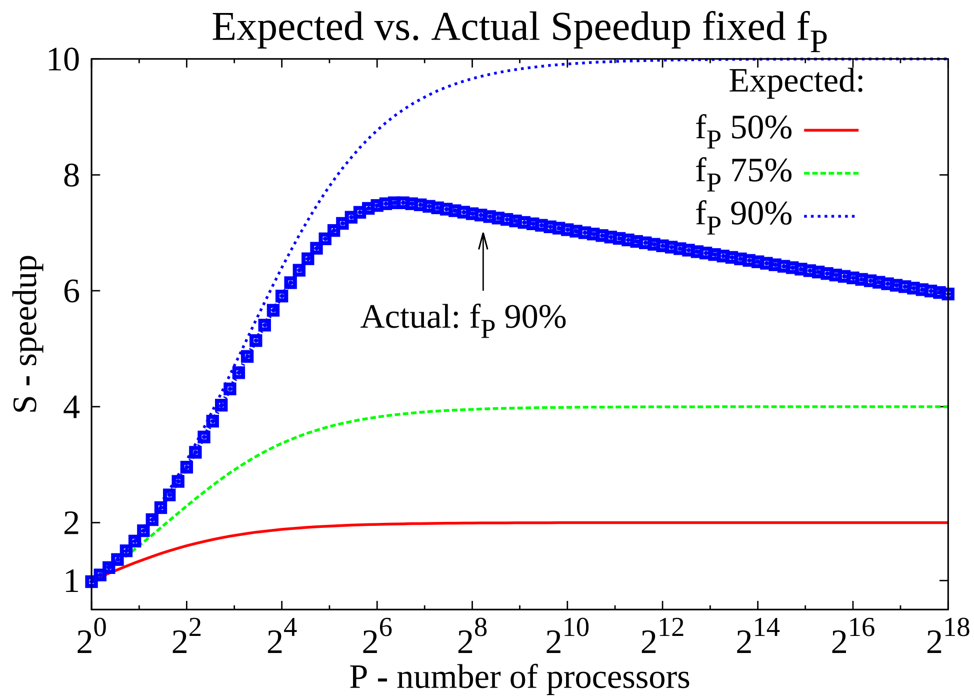
$\longrightarrow$ ***increase the size*** of the largest solvable problem

- most of the calculations quoted in this contribution *unfeasible* without parallel processing
- we exploit parallel processing for WPS services as well as for standalone calculations with *mixed techniques*, mostly using *GRASS GIS* with (*Bash* and *Python* scripting) and a "central" *PostGIS* database; *OpenMP* used for selected applications
- we implemented a parallel version of `rotstab` stability model for rotational slides $\longrightarrow$ see poster by I. Marchesini *et al.* at this conference
- we implemented a parallel version of `TRIGRS` stability model for shallow landslides $\longrightarrow$ M. Alvioli, R. Baum *et al.*, to be published; see also S. Raia, M. Alvioli *et al.*, Geosci. Mod. Dev. Discuss. **6** 1367 (2013)

3.2 Performance of parallelism for geo-spatial processes

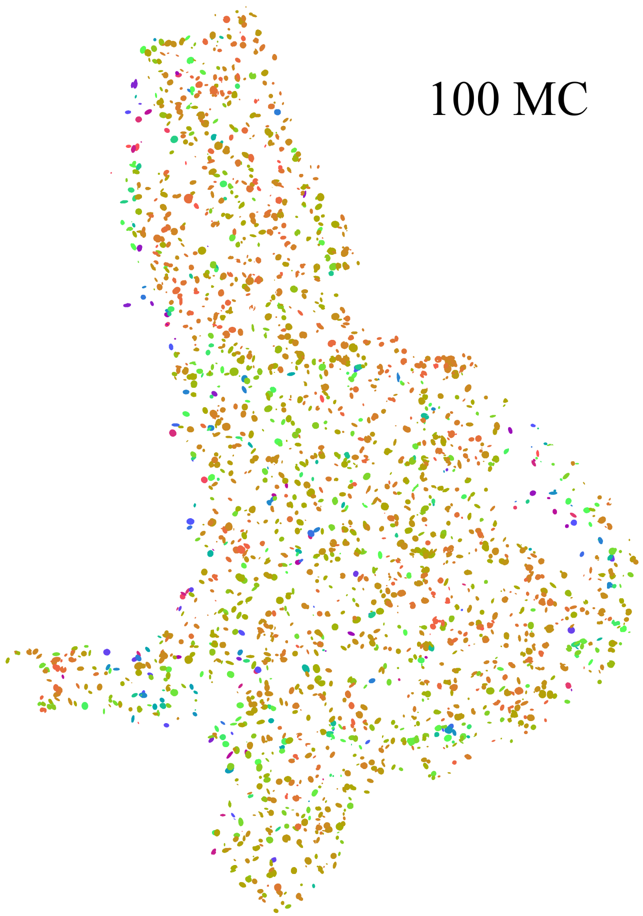
- performance of geo-spatial processing severely limited by "monolithic" softwares $\longrightarrow$ ubiquitous *multi-core capabilities ignored*
- chaining mixed techniques and softwares introduces significant ***overhead***; not much beyond partitioning a *raster map* in ***tiles*** can be done; use of ***databases*** to manage *vector maps* significantly limits performance
- we measure the ***speedup*** S achieved by using a number P of processing units (computing cores on one or more CPUs and physical machines)

$$S = \frac{T_1}{T_P} = \frac{1}{f_S + \frac{f_P}{P}}$$



- the ***serial fraction*** $f_S = 1 - f_P$ of a code can never be reduced to zero

3.3 A detailed example: `r.rotstab` GRASS GIS module



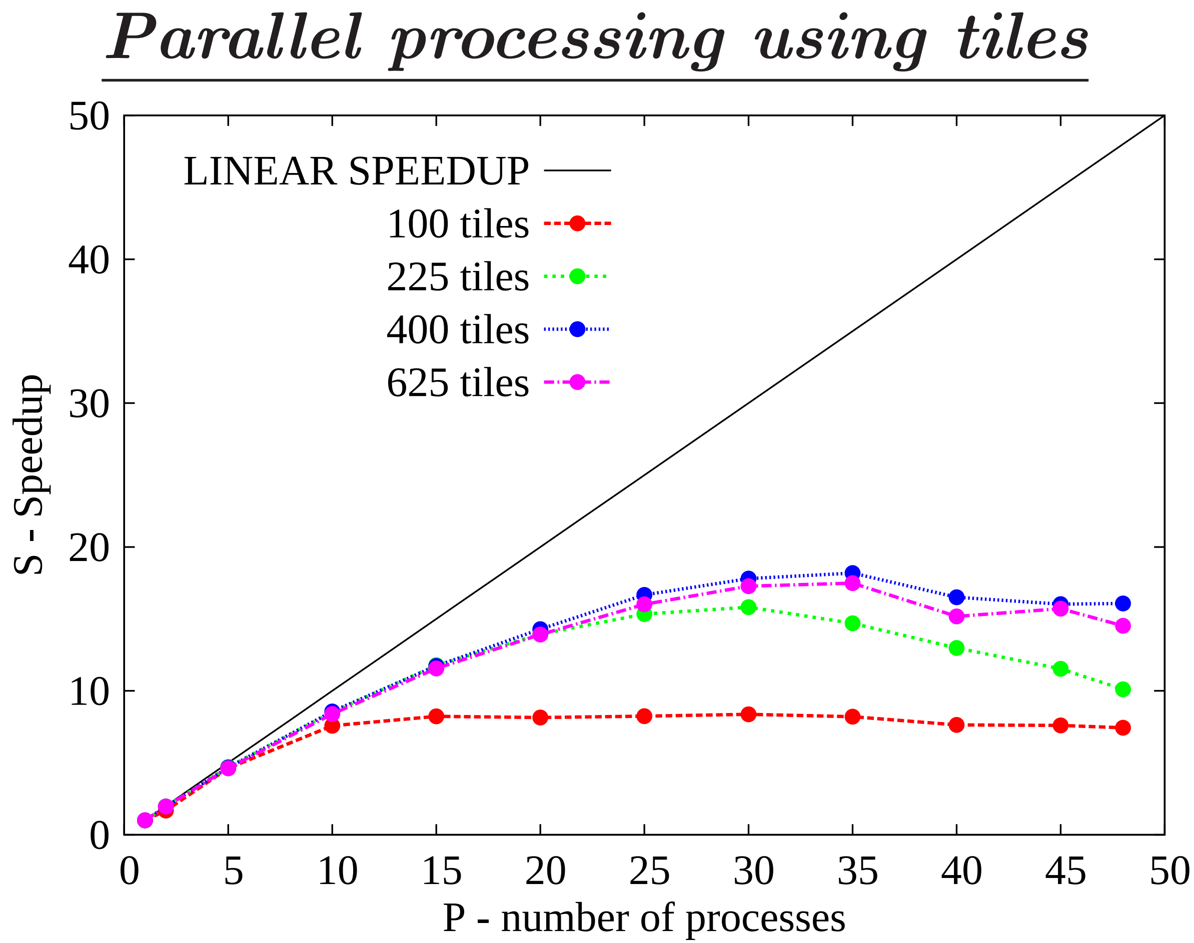
convergence of Monte Carlo sampling

3D slope stability model

- the model applies to deep-seated slides and it is implemented as a GRASS GIS module
- input of the code are a DEM and model parameters
- output of the code is a *factor of safety* raster map, after convergence of the model
- see poster by I. Marchesini *et al.* at this conference for model details
- soon to be implemented as a WPS service at

`http://alpha.irpi.cnr.it/cgi-bin/pywps.cgi`

3.4 A detailed example: `r.rotstab` GRASS GIS module



cartoon of the partition in *tiles*

we see the expected trend on speedup; overhead takes over after a few processes are used; optimal number P can be chosen

3.5 A lower level parallel implementation: TRIGRS model

- **TRIGRS**: Transient Rainfall Infiltration and Grid-Based Regional Slope-Stability Analysis - R. Baum, R., W. Savage, J. Godt - U.S.G.S. Of Report, 1159, 75 (2008)
- our probabilistic extension **TRIGRS-P**: S. Raia, M. Alvioli *et al.*, Geosci. Mod. Dev. Discuss. **6** 1367 (2013)
- running the (FORTRAN) code may be very demanding on large maps
- we are testing an **OpenMP** parallel version running on shared-memory machines
- the parallel version shows significant reduction of running time M. Alvioli, R. Baum *et al.*, to be published
- *relevant fact*: OpenMP provides ***minimal overhead*** and it is *best suited for parallelizing existing code*, where applicable - i.e. when *source code is available*

4. Conclusions

- we have implemented a number of **W**eb **P**rocessing **S**ervices

$\longrightarrow$ *publicly available* using a WPS client (*i.e. QGIS*)

$\longrightarrow$ part of the services exploit *parallel processing* on CNR IRPI (Perugia branch) computing infrastructure

$\longrightarrow$ developement and testing of a number of *upcoming WPS services* is under way

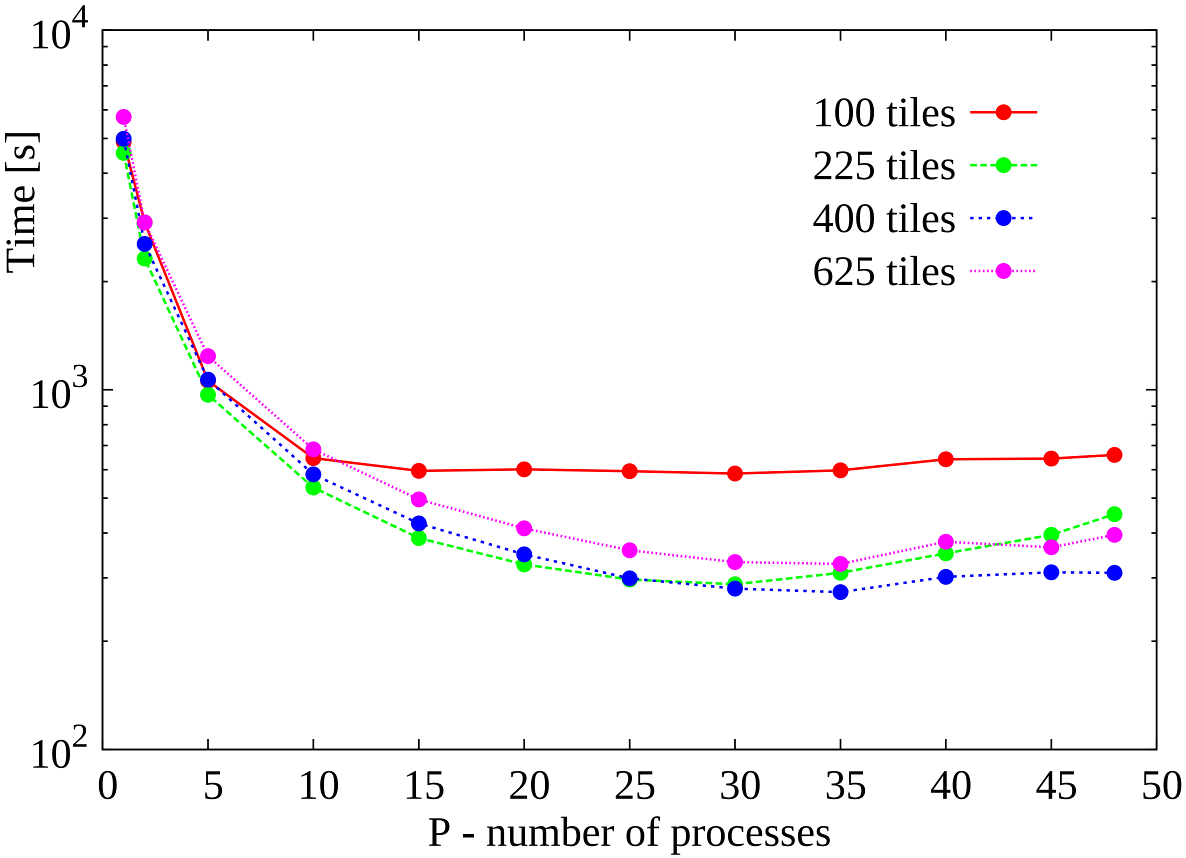
- we show that *low-level parallelization* (FORTRAN, C) of geo-spatial applications is possible and very performing - TRIGRS is an example
- updated versions of GIS and related softwares fully exploiting computing power of modern machines are highly desirable!
- our **final message**: use our WP Services available at

`http://alpha.irpi.cnr.it/cgi-bin/pywps.cgi`

Additional Slides

A detailed example: `r.rotstab` GRASS GIS module

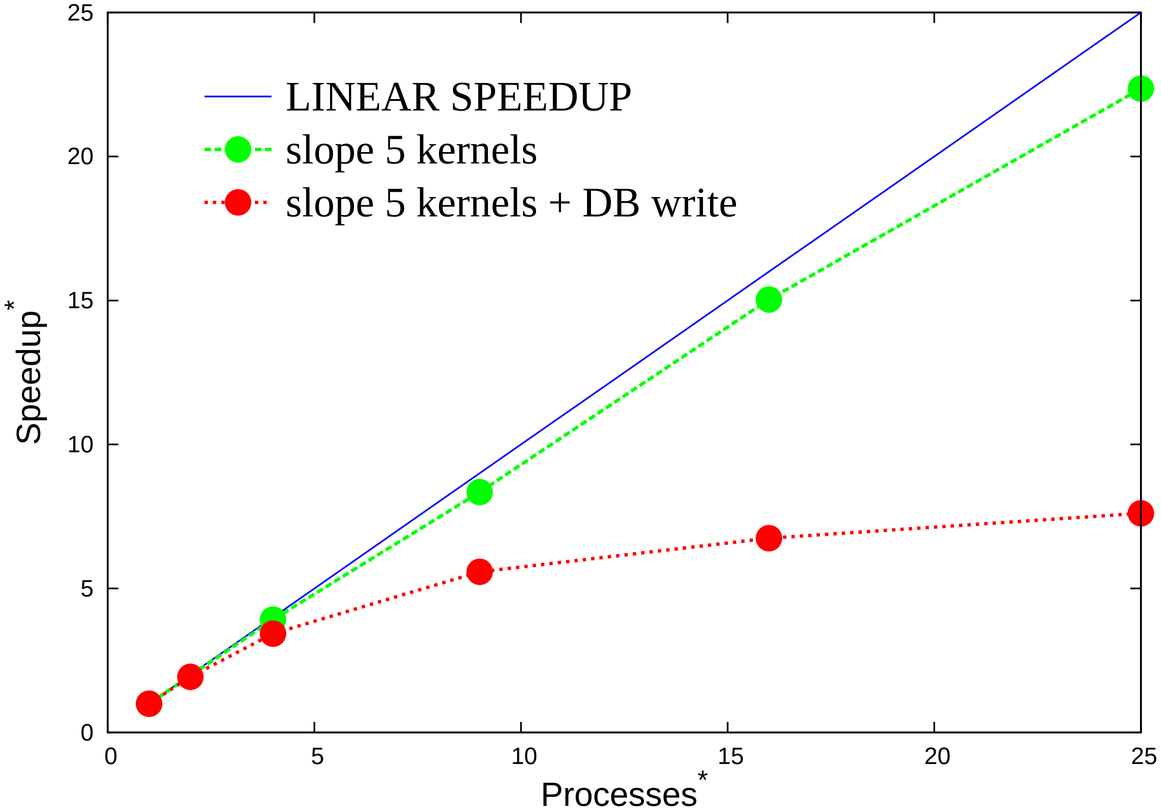
Parallel processing using tiles



Reduction of computing time

importance of I/O operations

Parallel processing using tiles



Speedup with and without the I/O operations