Geoderma xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Geoderma

journal homepage: www.elsevier.com/locate/geoderma

CyberSoLIM: A cyber platform for digital soil mapping

Jingchao Jiang [a,b], A-Xing Zhu [a,c,d,e,*], Cheng-Zhi Qin [a,e], Tongxin Zhu [f], Junzhi Liu [c,e], Fei Du [d], Jing Liu [d], Guiming Zhang [d], Yiming An [a,b]

[a] State Key Laboratory of Resources and Environmental Information System, Institute of Geographic Sciences and Natural Resources Research, CAS, Beijing 100101, China
[b] Graduate University of Chinese Academy of Sciences, CAS, Beijing 100049, China
[c] School of Geography, Nanjing Normal University, Nanjing, Jiangsu 210023, China
[d] Department of Geography, University of Wisconsin-Madison, Madison, WI 53706, USA
[e] Jiangsu Center for Collaborative Innovation in Geographical Information Resource Development and Application, Nanjing, Jiangsu 210023, China
[f] Department of Geography, University of Minnesota-Duluth, Duluth, MN 55812, USA

ARTICLE INFO

Article history:
Received 16 December 2014
Received in revised form 15 April 2015
Accepted 16 April 2015
Available online xxxx

Keywords:
Digital soil mapping (DSM)
Heuristic model building
Visualized modeling
Parallel computing
Online mapping system

ABSTRACT

In recent years, new demands and trends have emerged in the digital soil mapping (DSM) field: the range of applications as well as the range of users has become much diverse. Users of DSM include not only experts in soil science community but also those from relevant domains (e.g., hydrology, ecology). In addition, the rapid expansion of areas for DSM and the ever increasing spatial resolution of covariates call for an accelerated level of computation. These new trends have raised the bar for DSM software platforms. This paper presents CyberSoLIM, a prototype system to illustrate an idea of easy-of-use and high performance enabled cyber environment for DSM. CyberSoLIM is implemented to have five main features: (1) heuristic modeling, which allows digital soil mappers to construct DSM workflow easily; (2) visualized modeling with which the conceptual workflow of DSM is expressed by graphic icons; (3) workflow reuse, which increases the efficiency of DSM deployment; (4) online execution and high-performance computing, which can use the advantage of cyber infrastructure and high-performance computing; and (5) web service enabled, which provides effective and easy means to share and integrate models and algorithms. As an illustration of such environment, CyberSoLIM was used to infer the silt content in topsoil (0–20 cm) in China's Anhui Province. The case study confirms that software platform as illustrated by CyberSoLIM is easy to use and efficient in terms of mapping productivity.

1. Introduction

The spatial variation of soil types and soil properties is an important basic information in the research of soil erosion (Putman et al., 1988), hydrological modeling (Mukundan et al., 2010), and ecological modeling (Tan et al., 2005). Digital soil mapping (DSM) provides an effective way to obtain high-precision and high-resolution soil distribution information (McBratney et al., 2003). Because it requires complicated soil inference models (McBratney et al., 2000) and tedious data preparation processes (e.g., RS and GIS data [Ryan and McKenize, 1999; Dobos et al., 2000; Mulder et al., 2011]), DSM typically requires software tools.

The software tools for DSM can be divided into three types: (a) general-purpose geostatistics tools such as Gstat (Pebesma, 2004), ArcGIS Geostatistical Analyst (ESRI, 2014) and GS plus (Gamma Design Software, 2007); (b) statistical packages such as R and Matlab; and (c) special soil mapping tools such as FuzME (Minasny and McBratney, 2002), SoLIM Solutions (Zhu et al., 1997; Smith et al., 2006; Zhu, 2006) and TAL (Teh, 2002; Sung, 2014). Emergent demands and trends in the DSM field pose new challenges for these software tools. The range of applications and user groups has widened as more users from relevant domains (e.g., hydrology, ecology) who are interested in soil mapping are using DSM. At the same time study areas are expanding in size with data increasing in spatial resolution, both of which require greater amounts of computation. This raises the requirements for software tools of DSM. Existing tools cannot meet these new demands due to the following constraints: (1) the functions of DSM are provided with a traditional desktop metaphor style (e.g., menu, button and command line) or a programming style (e.g., R and Matlab), which keeps users from obtaining a visual and holistic understanding of modeling processes; (2) the modeling workflow (including data preparation flow) is too heavily tied to user knowledge and its use is heavily dependent on the level of knowledge the user possesses of the specifics about the programs and the methods required for data preparation; (3) the softwares are usually packaged in stand-alone versions, which often require tiresome download, installation and configuration process, and are also difficult to be shared and integrated; (4) the software programs can hardly take the advantage of recent developments in high-performance computing and cyber infrastructure.

* Corresponding author at: School of Geography, Nanjing Normal University, Nanjing, Jiangsu 210023, China.
E-mail address: azhu@wisc.edu (A.-X. Zhu).

http://dx.doi.org/10.1016/j.geoderma.2015.04.018
0016-7061/© 2015 Elsevier B.V. All rights reserved.

To address these challenges in DSM, we present a new form of computing environment (or platform) for DSM which is illustrated through a prototype system named CyberSoLIM. The basic idea and overall design of the platform are presented in Section 2. Section 3 provides the details on the main implementations of CyberSoLIM for DSM. We demonstrate an application of CyberSoLIM through the inference of the silt content in Anhui Province in Section 4. Section 5 provides a discussion of possible future developments.

2. Basic idea and overall design

2.1. Basic idea for a new DSM platform

A typical process of a predictive DSM would involve the following tasks once the research area and soil property of interest are defined: 1) determine which soil inference model to use; 2) define the set of environmental covariates for the prediction; 3) select the algorithms needed to prepare spatial data on the environmental covariates; 4) organize the model and algorithms into a conceptual workflow that includes information on the input–output relationship and execution order among them; 5) execute data preparation algorithms and the soil inference model in accordance with the conceptual workflow; and 6) visualize and evaluate the results of the selected soil properties.

The problems as stated in the Introduction section are associated with Steps 3) through 5). Our idea to address these problems is to develop a new DSM platform which will reduce the dependence of DSM application on the knowledge of the user about data processing and algorithm selection and free the users from the burdens of constructing conceptual workflow, the burdens of setting up the computing environment and the efforts in generating and managing spatial data on the environmental covariates. At the same time it allows users to revise the conceptual workflow if needed in an intuitive way and DSM method developers to share their predictive models and methods effectively and easily.

With this idea, the new DSM platform should possess the following properties: first, the platform must have sufficient intelligence to assist the construction of conceptual workflow of DSM once the mapping objectives (such as the area and the property to be mapped) are set. A conceptual workflow for DSM includes the selection of soil prediction model and the selection of soil environmental covariates and the process of generating spatial data on these covariates. The latter often involves a great deal of geographic information processing often referred to as geographic information system (GIS) and or remote sensing (RS) analysis. To facilitate the construction of the workflow, the platform must provide means for the selection of the prediction models and for the selection of the environmental covariates, even for users to add user-specific (user-defined) covariates. Once the environmental covariates are set, the platform needs to be able to automatically construct the process of generating spatial data on the environmental covariates by linking GIS/remote sensing analytical techniques. This would dramatically reduce the GIS/RS processing knowledge requirement for DSM users.

The second property is that the platform must be able to make the conceptual workflow visible in the form that users not only can visually see the workflow but also can interact with it to change model selection and make algorithm selection choices. In this way, the processing of DSM is not a black-box.

The third property is that the platform must have the ability and intelligence to manage the data generated by the workflow and intelligently use/reuse these data and the conceptual workflow developed. This will significantly reduce the overhead on DSM or rerun of the existing DSM workflow thus increase the efficiency of DSM deployment.

The fourth property is that the platform should be able to execute the workflow in the cyberspace, meaning that the execution of the workflow does not necessarily tie to the computing environment of the user. In addition, the platform must be able to utilize the high performance computing capacity if available to increase the execution of DSM.

The fifth property is that the platform must provide an effective and easy means for practitioners to share or contribute soil prediction models and/or methods for generating spatial data on environmental covariates, without requiring them to develop extensive interface to their models/methods. This would dramatically increase the distribution and adoption for new ideas and new methods developed by the DSM community.

The first, second and third properties ensure ease of use. The fourth property ensures high-performance in DSM and the Fifth property facilitates sharing in DSM.

2.2. Overall design of the DSM environment

Based on the idea stated in Section 2.1, we prototyped CyberSoLIM, a Java Web environment, which can be accessed directly through web browsers online, consisting of a knowledge base, a web service base, a spatial database and five modules (Fig. 1).

Knowledge base is used to store the formalized knowledge of DSM. At present, it contains the workflow building knowledge which documents the input and output data of the soil inference model and data preparation algorithms. The knowledge is used by heuristic model

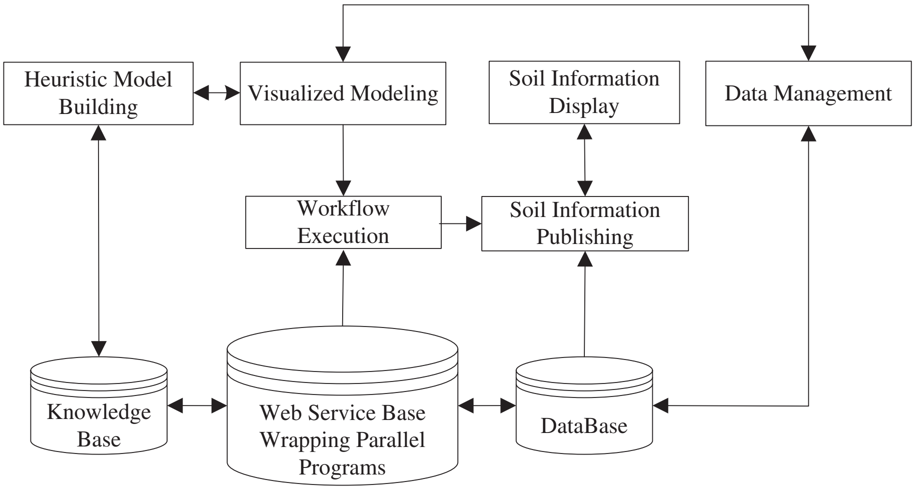


Fig. 1. Overall design of CyberSoLIM.

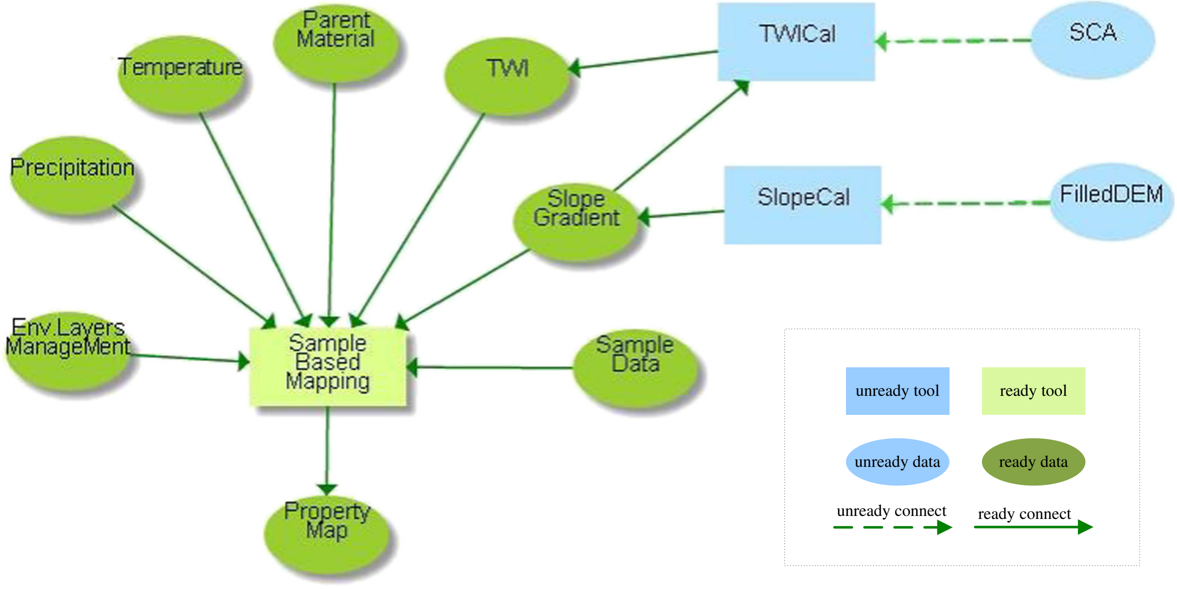


Fig. 2. An example of visual conceptual workflow of digital soil mapping. In this example, the visual workflow shows the relationships between the soil property map to be produced and the environmental covariates through a soil inference model (the sample-based mapping model). It also illustrates how the two environmental covariates (topographic wetness index [TWI] and Slope) can be prepared using a TWI algorithm and a slope algorithm, each of which leads to another data element.

building module to determine how certain input data could be calculated from other input data.

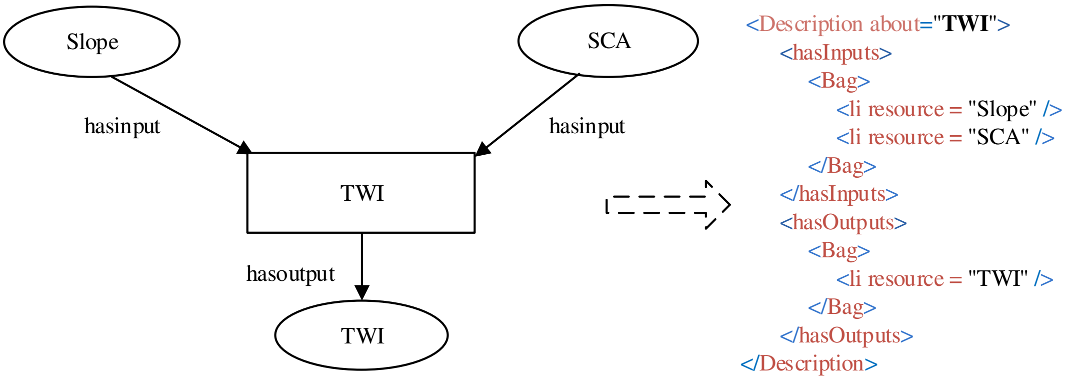
Web service base is used to infer soil property map and prepare input data for soil inference models. Web service database could be extended flexibly. At present, the web service database contains web services of a soil inference model based on samples, part of the Soil Land Inference Model (SoLIM, Zhu et al., 2001), referred to as individual predictive soil mapping (iPSM, Zhu et al., 2015) and data preparation algorithms (mostly focused on terrain variables). iPSM is based on the assumption that similar environmental conditions have similar soils and uses the soil–environment relationship at each individual soil sample location to predict soil properties at environmentally similar but unvisited locations and estimate prediction uncertainty (Zhu et al., 2015). The preparation algorithms include pit-remove algorithm, single flow direction algorithm, multiple flow direction algorithm, slope gradient algorithm, topographic wetness index (TWI) algorithm, specific catchment area (SCA) algorithm, profile curvature algorithm, and plan curvature algorithm. The model and algorithm programs are parallel and wrapped as web services. Practitioners in DSM can add their individual contributions to this service by wrapping their own respective functions (either soil prediction models or data preparation methods) as web services and share them out.

The spatial database is used to store the input data on the environmental covariates, intermediate data and results. The database contains the environmental raster covariates, soil samples, and other raster data such as digital elevation model (DEM).

The five modules are for visualized modeling, heuristic model building, workflow execution, data management, and soil information publishing/soil information display.

The visualized modeling module provides a canvas to display the DSM workflow in an intuitive flow diagram of graphic icons. This module also provides functions to add, delete, and move icons of model/algorithm and input/output data on the canvas.

The heuristic model building module uses the formalized knowledge in the knowledge base to select and organize the soil inference model and data preparation algorithms together to build a DSM workflow. Meanwhile, it invokes functions provided by the visualized modeling module to display the icons of the model/algorithm and input/output data on the canvas. After the conceptual workflow is built, it will be sent to the workflow execution module.



```
<Description about="TWI">
   <hasInputs>
      <Bag>
         <li resource = "Slope" />
         <li resource = "SCA" />
      </Bag>
   </hasInputs>
   <hasOutputs>
      <Bag>
         <li resource = "TWI" />
      </Bag>
   </hasOutputs>
</Description>
```

Fig. 3. An example of encoding input data and output data of TWI algorithm in RDF: the input data element of the algorithm is slope and SCA; the output data element is TWI.

The workflow execution module converts the conceptual workflow into a web service chain and invokes the web services described in the web service chain, which are then executed automatically.

The data management module is provided to users with the ability to manage their own data. Users can upload input data and download the soil property map and uncertainty map to their own computers. A validation function is provided to calculate the root-mean-square error (RMSE) between the soil property map and an independent sample dataset. One of the other functions of this data management module is to organize and manage the data generated from the workflow execution.

After the execution of the web services in the workflow, the soil information publishing/display module publishes the results in the format of Web Mapping Service (WMS) and displays the results of soil mapping on a webpage in the format of WMS.

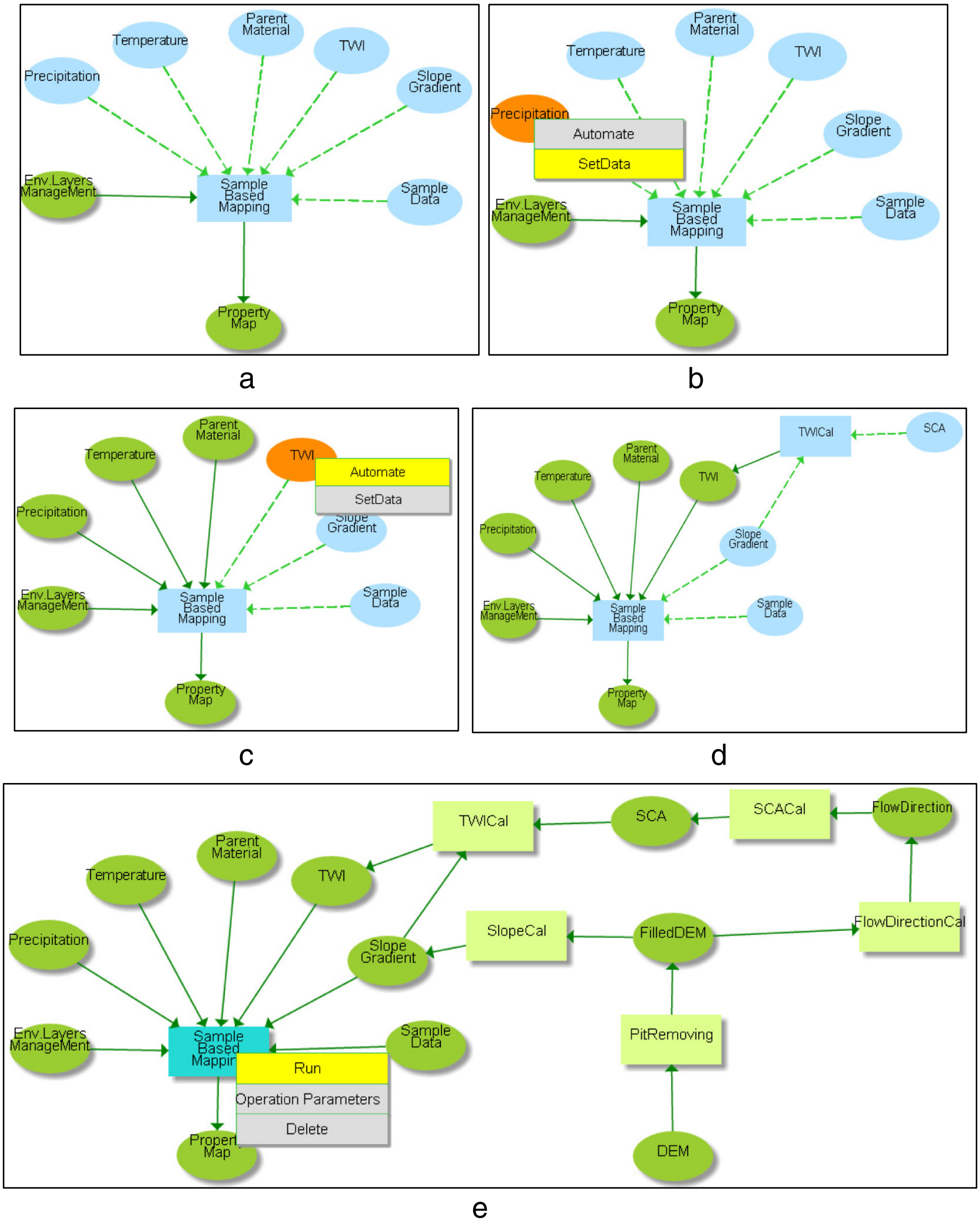


Fig. 4. Operation process of heuristic model building. a) The user selects the property map as target output data. The heuristic-driven method finds that the output of the sample-based mapping model is property map, so the icons of soil mapping model and its input/output data are added to the canvas; b) the user clicks each of the input data icon of soil mapping model and a context menu pops up, which contains "Automate" and "SetData" items. If the input data of soil mapping model is available, the user clicks the 'SetData' item to prepare the input data. Input data for precipitation, temperature, and parent material are set in this means; c–d) if input data (for example, TWI in this case) of soil mapping model is not available, the user clicks the "Automate" item, the heuristic-driven method finds the output of the TWI algorithm is TWI, so the TWI algorithm and its input data (i.e., slope gradient and SCA) are added into the canvas in the form of graphic icons. Slope gradient is also generated by this means; e) the user clicks the input data icon of the preparation algorithm, the context menu pops up again; the user chooses the 'SetData' or 'Automate' item according to the data availability. Repeat the above operations until all the input data of algorithms is available or can be generated by other algorithms.

Please cite this article as: Jiang, J., et al., CyberSoLIM: A cyber platform for digital soil mapping, Geoderma (2015), http://dx.doi.org/10.1016/j.geoderma.2015.04.018

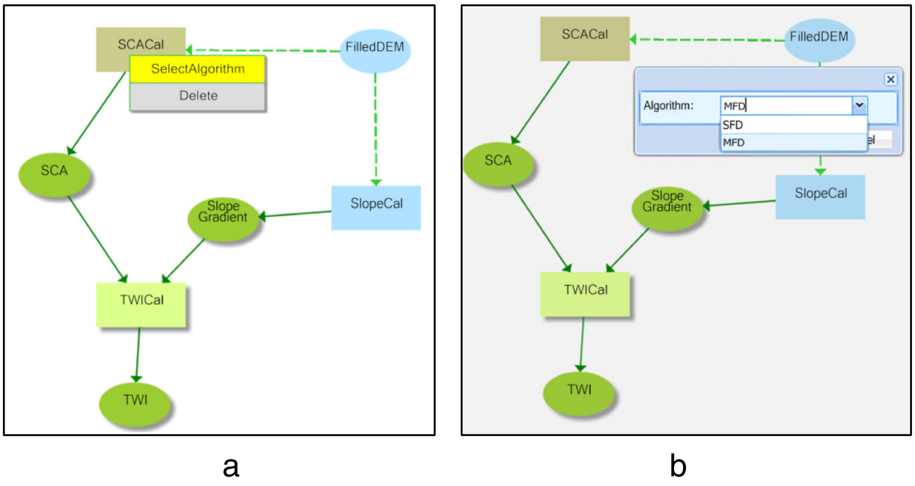


Fig. 5. An example of users assigning a specific algorithm for the conceptual workflow in CyberSoLIM. a) A pop-up box for selecting an algorithm appears when the user right-clicks the "SCACalc" rectangle representing the SCA-calculating tool. b) The user can select the MFD or SFD algorithm through the pop-up box.

3. Preliminary implementation

3.1. Implementation of visualized modeling

A conceptual workflow for DSM consists of one soil inference model and data preparation algorithms needed for the environmental covariates selected by the user. In CyberSoLIM, the workflow is visualized in an intuitive flow diagram of graphic icons.

The <canvas> tag of HTML5 is used to draw graphics on a web page via JavaScript. HTML5 is selected because all mainstream browsers (e.g., Internet Explorer 9 +, Firefox, Opera, Chrome, and Safari) support the <canvas> element without any plug-in installation.

There are three basic types of visualized elements in CyberSoLIM: tools (soil inference models and data preparation algorithms), data, and connectors (Fig. 2). A tool element is expressed as a rectangle, a data element is expressed as an oval, and the input or output relationship between a tool element and a data element is expressed as a unidirectional arrow line. The colors or styles of these icons will change when the corresponding parameter values or input data have been supplied so that the user can know which part of the workflow is ready. In CyberSoLIM implementation, the tool icons change colors from blue to yellow; the data icons from blue to green, and the connector icons from dotted line to full line when they are set. A finished workflow can be saved and reused.

3.2. Implementation of heuristic modeling

Although visualized modeling is intuitive and reusable, it is still difficult for soil mappers to construct a workflow (Qin et al., 2013). Construction of a proper conceptual workflow for DSM involves several types of knowledge: 1) input data required for the soil mapping inference model; 2) algorithms capable of generating the input data; and 3) the organization and execution orders for the model and algorithms. In fact, 1) and 2) can be expressed as one knowledge format called "declarative knowledge" (Anderson, 1983). Declarative knowledge documents the input and output data of a model (and algorithm). The third type of knowledge can then be derived by the input–output relationship among models and algorithms. Thus, it is automatically to construct the conceptual workflow once the soil inference model and the environmental covariates are defined by the user.

3.2.1. Encoding modeling knowledge

Declarative knowledge can in fact be formalized as an "object-attribute-value" triple. This triple can be encoded using the Resource Description Framework (RDF), a W3C standard for describing identifiable resources (Cyganiak et al., 2014). For example, Fig. 3 shows the RDF of the input data and output data of TWI algorithm. The declarative knowledge about the soil inference model and data preparation algorithms is stored in the knowledge base of CyberSoLIM in the RDF format.

3.2.2. Heuristic-driven method

The basic inference steps of the heuristic-driven method are as follows:

Step 1 Users determine the soil property map (the output data T) to be inferred;

Step 2 Query the RDF of model/algorithm A_i, whose output data is T, from the knowledge base. Then A_i is added into temporary array L. The icons of A_i and its input/output data are added on the canvas;

Step 3 For each input I_j ($j = 1, 2, \ldots, m$; m is the input data count of A_i), users are alerted to check if there is an available data for I_j. If so and the user decides to select this data I_j, the heuristic inference for I_j will be finished. Otherwise, assign I_j to T and return to Step 2. I_j could be the input data for the soil mapping model, or input data for data preparation algorithm.

The steps are repeated until the input data for the model and algorithms in L are supplied by the data base (user defined) or by the output data of other algorithms in the workflow. L records the algorithm execution order, in which L_n will execute first and L_1 (i.e., soil inference model) last.

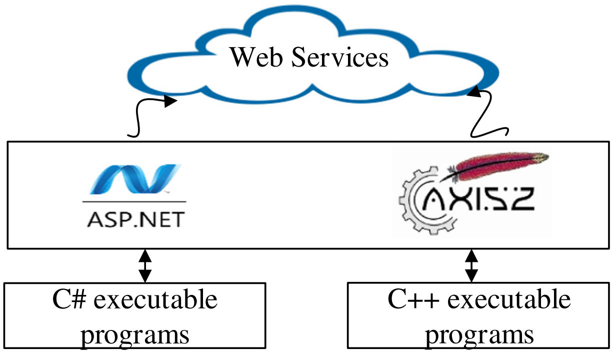


Fig. 6. Wrapping executable programs as web services.

Fig. 4 provides an example of heuristic model building for soil mapping. In this example, a user is attempting to infer a particular soil property. There are multiple input data for the soil inference model (e.g., precipitation, temperature, parent material, TWI, slope gradient and soil sample data). Among these inputs, precipitation, temperature, parent material and soil sample data can be set up using the available data, while TWI and slope gradient would be derived in an interactively expanded way until the DEM data are available in the data base.

Users can select a specific algorithm to compute the data to customize the workflow. For example, the user may want a different algorithm to compute slope gradient other than the one defaulted in CyberSoLIM. This can be achieved by right-clicking the calculation tool, represented by the rectangle. In Fig. 5, for example, there are two alternative algorithms, multiple-flow direction (MFD) and single-flow direction (SFD), for SCA data. By right-clicking the rectangle representing the SCA-calculating tool, the soil mapper can assign the specific algorithm through a pop-up box.

3.3. Implementation of workflow execution

3.3.1. Implementation of high-performance computing

The computations needed for DSM are on the server side which is typically more powerful in terms of computational ability and storage capacity. To speed up the computation of DSM, the soil inference model and data preparation algorithms in CyberSoLIM are implemented as parallel computing capable programs.

3.3.2. Wrapping models and algorithms as web services

To facilitate the sharing and reuse of soil inference models and data preparation algorithms (e.g., digital terrain analysis algorithms), each model and algorithm is wrapped as a web service. At present, two types of executable programs for models and algorithms are available on the server side: C++ and C# programs. The C++ executable programs (i.e., sample-based inference model and pit-remove algorithm) are wrapped as web services employing Apache Axis2; the C# executable programs (other data preparation algorithms) are wrapped as web services that employ ASP.NET (Fig. 6). For a more detailed description of wrapping executable programs as web services, see Li et al. (2010). All the above web services are stored in the web service base.

By employing web services, CyberSoLIM can easily incorporate soil inference models and data preparation algorithms developed by other researchers. An example of this is shown in Fig. 7. A developer can easily submit the Web Services Description Language (WSDL) of the web service wrapping of the developer's methods that will be used in the CyberSoLIM platform. At the time of submission, users are also required to provide the meta-information of the algorithm, including the algorithm introductions, references, and developer's basic information. At this time, CyberSoLIM does not yet have the ability to enforce license agreement and to protect intellectual property, which should be addressed in later stages. As a result at present, only the person who shares the algorithms can use the methods. However, if he or she would like to share the algorithms with the public, he or she can contact with the administrators to make it so. Developers of the new methods which are shared in CyberSoLIM are acknowledged for their work through the information they provide at the time of submission. This information is presented to users at the time when the methods are selected during the stage of constructing the conceptual framework. The prototype system will generate a RDF file for each shared algorithm in accordance with the interface of Fig. 7. Users don't need to create the RDF file manually.

The shared algorithms need examination and verification of the system administrators, which means that the administrator of CyberSoLIM will test the security and performance of the submitted algorithms. If the algorithms can pass the testing, they will be integrated into CyberSoLIM. This allows other developers to easily contribute and share methods, which also makes the functionalities of CyberSoLIM open-ended and expandable while maintaining the stability and security of the system.

3.3.3. Orchestrating and executing web services based on a conceptual workflow

The conceptual workflow generated by visualized modeling and heuristic model building contains the models and algorithms, with the input–output relationship among them, and the operation orders. According to the conceptual workflow, web services wrapped as described in Section 3.3.2 can be orchestrated as a web service chain in the Web environment (Fig. 8). Business Process Execution Language (BPEL) is a popular XML-based language for defining and orchestrating a web service chain that is supported by most workflow engines, including, for example, Oracle BPEL, Active BPEL, Apache ODE and BPELPower (Yu et al., 2012). In this paper, Apache ODE is used as the workflow engine to execute the web service chain defined by BPEL (Zhao et al., 2012). The intermediate data and the results generated by the web service chain are stored in the spatial database.

3.4. Implementation of data management

The -database is organized and managed in the form of metadata. Each data in the database has a metadata. A metadata contains some information about data such as the users' name, spatial extent, semantic filed, and data access path. The users' name information ensures that the data is available for specified users. In default, users can only access their own data. If users would like to share their data, they could contact with the administrator of CyberSoLIM; the spatial extent information ensures that the data can cover the study area; the semantic field information is used to explain data type (e.g., soil sample data, DEM, and precipitation). Based on the three above information, a data in the database can be quickly identified and accessed by CyberSoLIM. When users upload their data into the database, the metadata of the data is also stored in the databse.

3.5. Implementation of soil information publishing/display

The results of DSM are normally raster data, which cannot be displayed directly on webpages, but must first be converted into image formats (e.g., png, gif, and jpeg) that webpages support. MapServer is employed to publish raster data as Web Map Services (WMS), which is a standard protocol for servicing georeferenced map images over the Internet. To publish a WMS, a map file (Gkatzoflias et al., 2013), which records basic information such as certain layers, classes and styles and raster data locations, must be provided to MapServer.

OpenLayers is used to display WMS map data published by MapServer in most modern web browsers (Fig. 9). OpenLayers is a pure JavaScript library, with no server-side dependencies. Users do not need to install on-premises GIS software to display the soil mapping results.

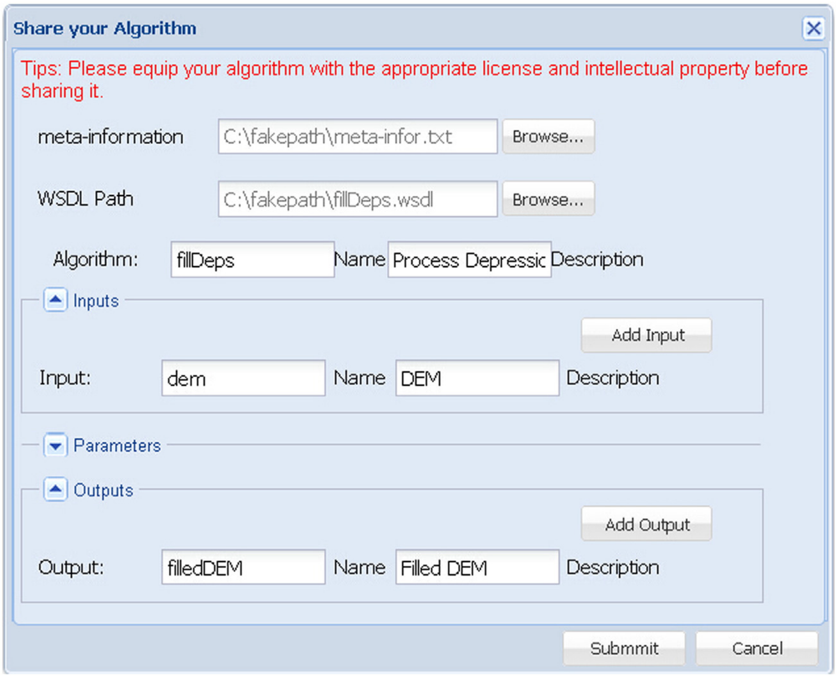
4. Application

To evaluate its ease of use and high performance, CyberSoLIM was used to infer the silt contents in topsoil (0–20 cm) in the southern region of Anhui Province, China. URL of CyberSoLIM is 159.226.110.183:8080/egc/. An online and detailed user manual is available at http://solim.geography.wisc.edu/software/CyberSoLIM/CyberSoLIM_handout.pdf.

4.1. Study area and data

The study area is located between 48°53′24″–48°59′24″N and 125°8′24″–125°16′12″E and includes Xuanzhou county, Langxi county and Guangde county in southern Anhui Province. It has an area of about 5900 km^2 and a maximum altitude of 1039 m.

Please cite this article as: Jiang, J., et al., CyberSoLIM: A cyber platform for digital soil mapping, Geoderma (2015), http://dx.doi.org/10.1016/j.geoderma.2015.04.018



Share your Algorithm

Tips: Please equip your algorithm with the appropriate license and intellectual property before sharing it.

meta-information: C:\fakepath\meta-infor.txt Browse...

WSDL Path: C:\fakepath\fillDeps.wsdl Browse...

Algorithm: fillDeps Name: Process Depressic Description:

Inputs

Add Input

Input: dem Name: DEM Description:

Parameters

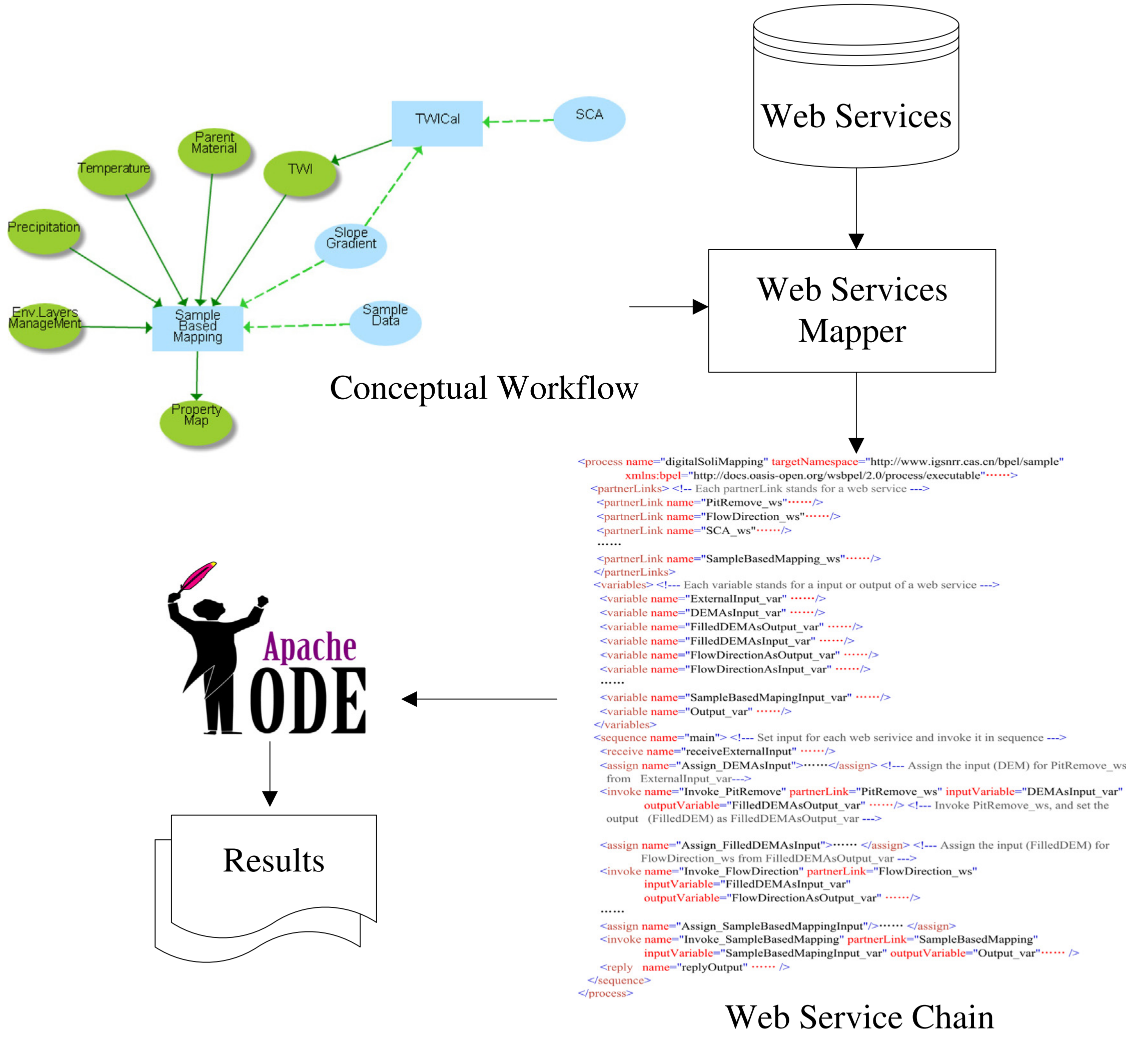
Outputs

Add Output

Output: filledDEM Name: Filled DEM Description:

Submit Cancel

Fig. 7. Interface for developers to share their own respective algorithms to be used in CyberSoLIM.



```
<process name="digitalSoliMapping" targetNamespace="http://www.igsnrr.cas.cn/bpel/sample"
         xmlns:bpel="http://docs.oasis-open.org/wsbpel/2.0/process/executable"······>
  <partnerLinks> <!-- Each partnerLink stands for a web service --->
   <partnerLink name="PitRemove_ws"······/>
   <partnerLink name="FlowDirection_ws"······/>
   <partnerLink name="SCA_ws"······/>
   ······
   <partnerLink name="SampleBasedMapping_ws"······/>
  </partnerLinks>
  <variables> <!--- Each variable stands for a input or output of a web service --->
   <variable name="ExternalInput_var" ······/>
   <variable name="DEMAsInput_var" ······/>
   <variable name="FilledDEMAsOutput_var" ······/>
   <variable name="FilledDEMAsInput_var" ······/>
   <variable name="FlowDirectionAsOutput_var" ······/>
   <variable name="FlowDirectionAsInput_var" ······/>
   ······
   <variable name="SampleBasedMapingInput_var" ······/>
   <variable name="Output_var" ······/>
  </variables>
  <sequence name="main"> <!--- Set input for each web serivice and invoke it in sequence --->
   <receive name="receiveExternalInput" ······/>
   <assign name="Assign_DEMAsInput">······</assign> <!--- Assign the input (DEM) for PitRemove_ws
    from   ExternalInput_var--->
   <invoke name="Invoke_PitRemove" partnerLink="PitRemove_ws" inputVariable="DEMAsInput_var"
           outputVariable="FilledDEMAsOutput_var" ······/> <!--- Invoke PitRemove_ws, and set the
    output   (FilledDEM) as FilledDEMAsOutput_var --->

   <assign name="Assign_FilledDEMAsInput">······ </assign> <!--- Assign the input (FilledDEM) for
          FlowDirection_ws from FilledDEMAsOutput_var --->
   <invoke name="Invoke_FlowDirection" partnerLink="FlowDirection_ws"
           inputVariable="FilledDEMAsInput_var"
           outputVariable="FlowDirectionAsOutput_var" ······/>
   ······
   <assign name="Assign_SampleBasedMappingInput"/>······ </assign>
   <invoke name="Invoke_SampleBasedMapping" partnerLink="SampleBasedMapping"
           inputVariable="SampleBasedMapingInput_var" outputVariable="Output_var"······ />
   <reply   name="replyOutput" ······ />
  </sequence>
</process>
```

Fig. 8. Orchestration and execution of web services based on conceptual workflow.

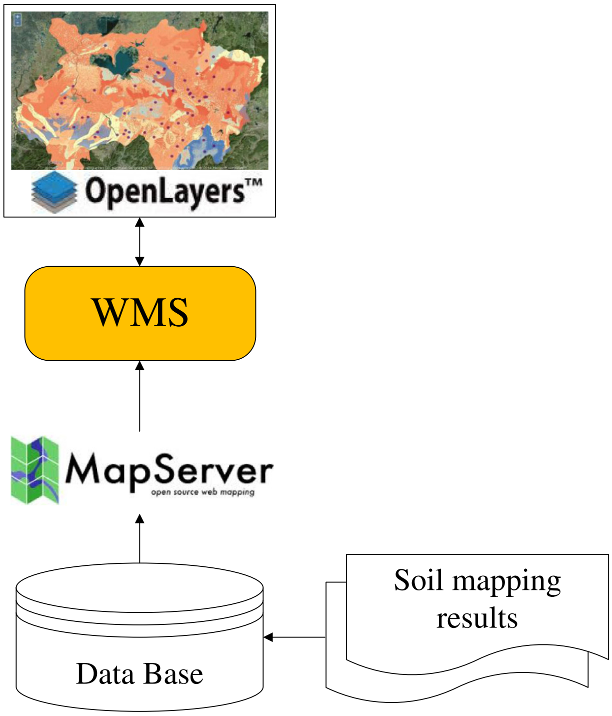


Fig. 9. The publishing and display of soil mapping results using MapServer and OpenLayers.

Existing environmental data include precipitation, air temperature and parent material. Precipitation and air temperature come from the Meteorological Data Sharing Service System of China, and parent material comes from the Data Sharing Infrastructure of Earth System Science of China. Other environmental data (e.g., TWI, slope gradient, profile curvature and plan curvature) can be obtained through heuristic model building using DEM. The DEM data comes from SRTM (90 m). The format of the data is ASCII with 1242 ∗ 928 pixels at 90 m resolution. Sixty-four soil samples are in CSV format. All the data are stored in the database of CyberSoLIM.

4.2. CyberSoLIM processes

CyberSoLIM was used online to infer the silt content in topsoil (0–20 cm). The basic steps are shown in Fig. 10 and are described as follows.

Step 1 Select the study area of interest from the geographical map webpage of CyberSoLIM, then right-click on the area to bring up a pop-window with various instructions, and select the "digital soil mapping" (Fig. 10a).

Step 2 CyberSoLIM shows the visualized modeling interface (Fig. 10b). There are two kinds of input data (environment variables and sample point data) for the sample-based mapping model (the soil inference model presented by Zhu et al., 2015). Click on the "Env. Layers Management" icon to show a pop-up window and select the environment variables needed. In this application, precipitation, air temperature, parent material, TWI, slope gradient, profile curvature and plan curvature are selected (Fig. 10c–d). Precipitation, air temperature and parent material data are available in the database. Other data layers are generated by other input data. A new workflow is built with the aid of the heuristic model-building module (Fig. 10e) (see Fig. 4 for process details). Once all environmental data layers are defined, sample point data are set by right-clicking the "sample data" icon. The sample data file is selected and then the soil property of interest "silt" is chosen (Fig. 10f). The parameters for the sample-based model are checked (Fig. 10g). Users can set the sample-similarity measurement method and the uncertainty threshold at their option. Otherwise, a set of defaults that normally work well for the selected method will be used. Then run the workflow.

Step 3 When the workflow is finished, the soil property and uncertainty map are displayed on the webpages (Fig. 10h). Users can download the soil-property map data to their computers by clicking the download link. Users can validate the output maps by click the "validate" button to gain RMSE which certainly requires a validation dataset.

This application example shows that CyberSoLIM provides an easy fashion of DSM. By employing CyberSoLIM, users can implement soil mapping visually and heuristically through the browser.

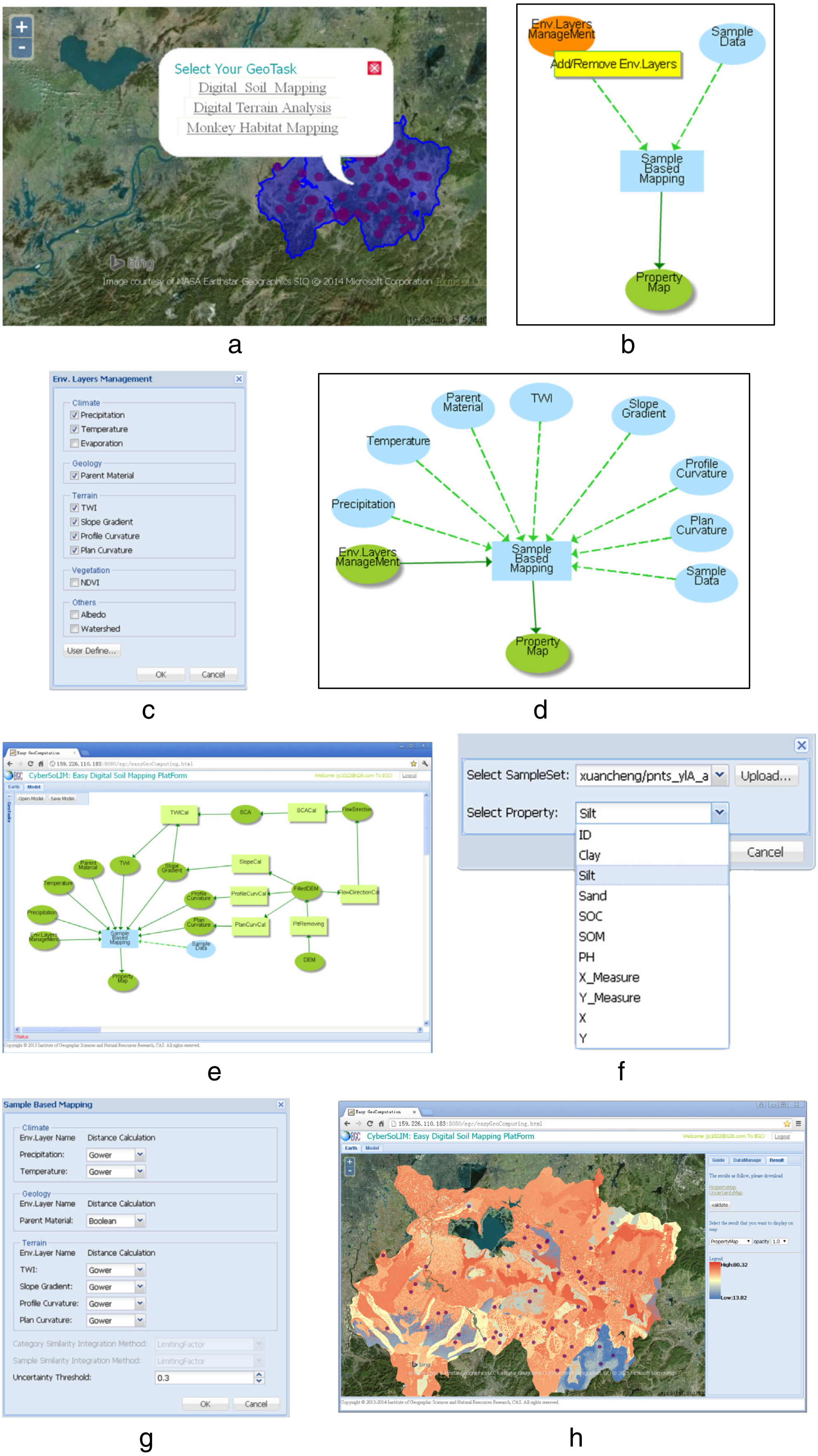
Parallel computing also allows users to obtain the soil mapping results relatively quickly. Table 1 provides the run time of the soil inference model for the study area with 90 m and 10 m resolution, respectively. As shown in Table 1, the run time of the soil inference model at both 90 m and 10 m resolutions decreases as the number of cores increases. The finer the resolution is (which means a greater amount of computing), the more significant the advantage of parallel computing is. With a resolution of 90 m, the wait of 3.7 s fro 32 cores is not much shorter than is one of 36.1 s for 1 core (used to approximate the time for a sequential implementation). However, with a resolution of 10 m, the run time of the soil inference model was shortened from about 48 min at 1 core to about 3 min at 32 cores.

5. Conclusions and future work

In this paper we presented CyberSoLIM, a new platform for DSM. This study shows that it is possible to put the modeling knowledge into software tool. This allows users to easily construct the workflow of digital soil mapping in a visual and intelligent manner without advanced soil inference and data preparation knowledge. Through an online deployment, users can complete the soil mapping task on webpages without software installation or plug-in. With this new architecture of digital soil mapping tool users can make use of the increasingly-available computational resources (e.g., parallel computing, web services), not only shortening the computing time for soil mapping, particularly over large areas with high spatial resolution timeframe but also gaining the convenience to disseminate their methods or employ the latest development in DSM. The new DSM architecture greatly reduces the difficulty and dramatically improves the efficiency of conducting DSM.

It should be noted that CyberSoLIM serve only to make the mapping process simpler and easier, but it does not mean that the person who performs this does not require the knowledge of soil and its relationships to the environment. It is not suggested here that knowledge of soil science is unimportant. Knowledge of soil–environmental relationships, for example, is needed to specify what environmental covariates are needed for mapping a given soil property. In fact, this knowledge is required for any kind of soil mapping. Mapping platforms of this kind can free soil mappers from tedious operations needed to prepare the data (including knowledge of the procedure and algorithm used to derive the data) so that they can focus on the substantive, scientific side of soil mapping. This type of mapping platform allows the mapper more time for the study of, for example, the relationship between spatial variation in soil properties and environmental covariates, rather than spending time on how to compute some common environmental covariates, such as the topographic wetness index and other well-documented environmental covariates that are common and simple to compute for GIS and remote sensing specialists but time consuming for soil scientists.

In addition, platforms like CyberSoLIM do allow an experienced or technically-capable soil mapper to alter the methods/algorithms for



c:

Climate
- ☑ Precipitation
- ☑ Temperature
- □ Evaporation

Geology
- ☑ Parent Material

Terrain
- ☑ TWI
- ☑ Slope Gradient
- ☑ Profile Curvature
- ☑ Plan Curvature

Vegetation
- □ NDVI

Others
- □ Albedo
- □ Watershed

f:

Select SampleSet: xuancheng/pnts_ylA_a

Select Property: Silt

g:

Env.Layer Name	Distance Calculation
Climate	
Precipitation:	Gower
Temperature:	Gower
Geology	
Parent Material:	Boolean
Terrain	
TWI:	Gower
Slope Gradient:	Gower
Profile Curvature:	Gower
Plan Curvature:	Gower

Category Similarity Integration Method: LimitingFactor

Sample Similarity Integration Method: LimitingFactor

Uncertainty Threshold: 0.3

Fig. 10. CyberSoLIM steps. a) Select the study area and choose the digital soil mapping instruction; b) enter into the visualized modeling interface; c–d) select the environment variables; e) build a visual workflow employing the heuristic method; f) select the sample point data and soil property “silt”; g) set parameters for the sample-based model; h) display silt content distribution map.

Please cite this article as: Jiang, J., et al., CyberSoLIM: A cyber platform for digital soil mapping, Geoderma (2015), http://dx.doi.org/10.1016/j.geoderma.2015.04.018

Table 1
Run time (s) of the soil inference model with dataset of 90 m and 10 m resolutions in different numbers of cores.

Number of cores	1	2	4	8	16	32
Run time (s) with a 90 m resolution	36.1	18.9	11.2	6.4	4.7	3.7
Run time (s) with a 10 m resolution	2864.7	1489.9	872.7	477.9	281.6	178.1

A linux cluster with 4 computing nodes was used as the hardware platform. Each computing node has 32 GB memory and 2 CPUs. The CPU type is Intel Xeon E5645 2.4 GHz; each CPU has 12 logical cores. MPICH2 version 1.3.1 was used as the parallel programming library.

soil mapping and computing. The methods/parameters can be altered through the properties of the tool element in the workflow. What's more, digital soil mapping technical developers can easily share their tools through the web services function provided in CyberSoLIM.

However, it should also be noted that CyberSoLIM is a prototype system and there will surely be a need for improvements in the future. One example is that at present, the data and algorithms on the server side are limited. Automatically finding and using the data and algorithms shared on the internet will be the subject of future work. Other frequently used digital soil inference models and algorithms (e.g., kriging, multiple regression, random forest, and geographically weighted regression) could be integrated into CyberSoLIM to provide more options for users. And the operations to collate, format, re-sample, transfer and document the data are tedious for users, these operations also need to be automated. The computing power and storage capacity of a single high-performance server is likely to be unable to support digital soil mapping for a large number of concurrent users. Cloud computing could be employed to solve this problem.

Acknowledgments

This study was funded by the National Natural Science Foundation of China (41431177), the Natural Science Research Program of Jiangsu (14KJA170001), the National Key Technology R&D Program (No. 2013BAC08B03-4), and the PAPD program of Jiangsu Higher Education Institutions (164320H116). The support received by A-Xing Zhu through the Vilas Associate Award, the Hammel Faculty Fellow, and the Manasse Chair Professorship from the University of Wisconsin-Madison is greatly appreciated. Cheng-Zhi Qin thanks the National Natural Science Foundation of China (41422109) for its support.

References

Anderson, J., 1983. The Architecture of Cognition. Harvard University, Cambridge, MA.
Cyganiak, R., Wood, D., Lanthaler, M., 2014. RDF 1.1 Concepts and Abstract Syntax. W3C Recommendation. 25 February 2014. URL:. http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/.
Dobos, E., Micheli, E., Baumgardner, M.F., Biehl, L., Helt, T., 2000. Use of combined digital elevation model and satellite radiometric data for regional soil mapping. Geoderma 97 (3–4), 367–391.
ESRI, 2014. ArcGIS geostatistical analyst. http://www.esri.com/software/arcgis/extensions/geostatistical (10-06).
Gamma Design Software, 2007. Geostatistics for the Environmental Sciences, GS+ 7.0. Gamma Design Software LLC, Plainwell, Michigan, USA.
Gkatzoflias, D., Mollios, G., Samaras, Z., 2013. Development of a web GIS application for emissions inventory spatial allocation based on open source software tools. Comput. Geosci. 52, 21–33.
Li, X.Y., Di, L.P., Han, W.G., Zhao, P.S., Dadi, U., 2010. Sharing geoscience algorithms in a Web service-oriented environment (GRASS GIS example). Comput. Geosci. 36 (8), 1060–1068.
McBratney, A.B., Odeh, I.O.A., Bishop, T.F.A., Dunbar, M.S., Shatar, T.M., 2000. An overview of pedometric techniques for use in soil survey. Geoderma 97 (3–4), 293–327.
McBratney, A.B., Mendoca, S.M.L., Minasny, B., 2003. On digital soil mapping. Geoderma 117 (1–2), 3–52.
Minasny, B., McBratney, A.B., 2002. FuzME Version 3.0, Australian Centre for Precision Agriculture. The University of Sydney, Sidney, Australia.
Mukundan, R., Radcliffe, D.E., Risse, L.M., 2010. Spatial resolution of soil data and channel erosion effects on SWAT model predictions of flow and sediment. J. Soil Water Conserv. 65 (2), 92–104.
Mulder, V.L., Bruin, S.D., Schaepman, M.E., Mayr, T.R., 2011. The use of remote sensing in soil and terrain mapping—a review. Geoderma 162 (1–2), 1–19.
Pebesma, E.J., 2004. Multivariable geostatistics in S: the Gstat package. Comput. Geosci. 30 (7), 683–691.
Putman, J., Williams, J.R., Sawyer, D., 1988. Using the erosion productivity impact calculator (EPIC) model to estimate the impact of soil-erosion for the 1985 RCA Appraisal. J. Soil Water Conserv. 43 (4), 321–326.
Qin, C.Z., Jiang, J.C., Zhan, L.J., Lu, Y.J., Zhu, A.X., 2013. A Browser/Server-based Prototype of Heuristic Modelling Environment for Digital Terrain Analysis. Geomorphometry'2013, Nanjing Normal University, October 15–20, 2013.
Ryan, P.J., McKenize, N.J., 1999. Spatial prediction of soil properties using environmental correlation. Geoderma 89 (1–2), 67–94.
Smith, M.P., Zhu, A.X., Burt, J.E., Stiles, C., 2006. The effect of DEM resolution and neighborhood size on digital soil survey. Geoderma 137 (1–2), 58–69.
Sung, C.T.B., 2014. Texture autoLookup (TAL). http://www.christopherteh.com/tal/index.html (10-06).
Tan, Z., Liu, S., Johnston, C.A., Loveland, T.R., Tieszen, L.L., Liu, J., Kurtz, R., 2005. Soil organic carbon dynamics as related to land use history in the northwestern Great Plains. Glob. Biogeochem. Cycles 19 (3), GB3011.
Teh, C.B.S., 2002. A computer program to determine the soil texture class for any classification scheme. Agro-Search 9, 19–21.
Yu, G.N., Zhao, P.S., Di, L.P., Chen, A.J., Deng, M.X., Bai, Y.Q., 2012. BPELPower—a BPEL execution engine for geospatial web services. Comput. Geosci. 47, 87–101.
Zhao, P.S., Di, L.P., Yu, G.N., 2012. Building asynchronous geospatial processing workflows with web services. Comput. Geosci. 39, 34–41.
Zhu, A.X. SoLIM solutions. http://solim.geography.wisc.edu/software/index.htm, 2006-04-10 /2014-10-06.
Zhu, A.X., Band, L., Vertessy, R., Dutton, B., 1997. Derivation of soil properties using a Soil Land Inference Model (SoLIM). Soil Sci. Soc. Am. J. 61 (2), 523–533.
Zhu, A.X., Hudson, B., Burt, J.E., Lubich, K., Simonson, D., 2001. Soil mapping using GIS, expert knowledge, and fuzzy logic. Soil Sci. Soc. Am. J. 65 (5), 1463–1472.
Zhu, A.X., Liu, J., Du, F., Zhang, S.J., Qin, C.Z., Burt, J., Behrens, T., Scholten, T., 2015. Predictive soil mapping with limited sample data. Eur. J. Soil Sci. http://dx.doi.org/10.1111/ejss.12244.

Please cite this article as: Jiang, J., et al., CyberSoLIM: A cyber platform for digital soil mapping, Geoderma (2015), http://dx.doi.org/10.1016/j.geoderma.2015.04.018