Composing Geoinformatics Workflows with User Preferences

David Chiu Sagar Deshpande‡ Gagan Agrawal Rongxing Li‡
Department of Computer Science and Engineering
‡ Department of Civil and Environmental Engineering and Geodetic Science
The Ohio State University
Columbus, OH 43210

ABSTRACT

With the advent of the data grid came a novel distributed scientific computing paradigm known as service-oriented science. Among the plethora of systems included under this framework are scientific workflow management systems, which enable large-scale process scheduling and execution. To ensure quality of service, these systems typically seek to minimize workflow execution time as well as costs for slices of data grid access. The geospatial domain, among other sciences, involves yet another optimization factor, the accuracy of results. The relationship between execution time and workflow accuracy can often be exploited to offer more flexibility in handling user preferences. We present a system which meets user constraints through a dynamic adjustment of the accuracy of workflow results.

Categories and Subject Descriptors

H.3.4 [**Systems and Software**]: Question-answering systems—*Automatic Geospatial Workflow Composition*

Keywords

Geospatial workflows, QoS, accuracy awareness

1. INTRODUCTION

The burgeoning advancements of web and grid technologies have helped launch a call for pushing the availability and distribution of data and resources in various scientific domains. Specifically within the geosciences, such authorities as the Federal Geographic Data Committee (FGDC) and the Open Geospatial Consortium (OGC) have pushed this initiative through the standardization of geo-semantics and geospatial services. With distributed hetergeneous datasets and processes comes the challenge for scientists to manage geoinformation. For instance, obtaining certain information involves execution of several processes with disparate data sources in a particular sequence. Ultimately, the hope for enabling these sequences of process executions (traditionally known as geospatial service chains or workflows [2]) is to automate their composition while simultaneously hiding low-level implementation details from the user. Thus, several efforts in geospatial workflow management systems [10, 9, 13, 7] have been initiated to address these tasks.

Often, there exists multiple ways of answering a given query using different combinations of data sources and services. Some combinations are likely to result in higher cost, but better accuracy. Others might lead to faster results and lower accuracy. This could be due to the fact that some data collection methods involve higher resolution than others, or that some datasets are available at servers with lower-access latencies than others. A main challenge for workflow management systems is the need for supporting user preferences on these circumstances.

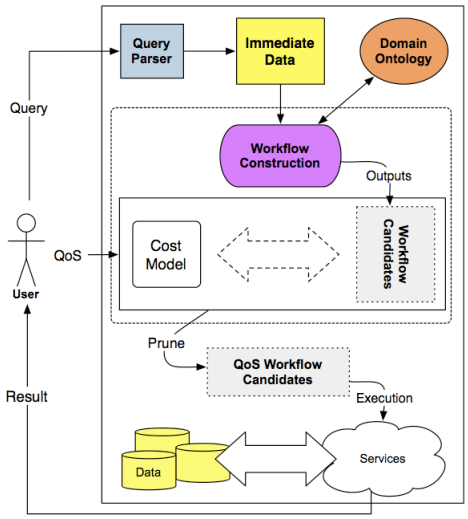

Figure 1: System Overview

While most efforts focus directly on minimizing execution times [12, 14, 1] through scheduling heuristics, it would be highly desirable if we could enable preferences in both accuracy and time. In other words, we seek to alleviate users from the need of understanding the cost-accuracy tradeoffs associated among datasets and services. We present such a framework for geospatial workflow composition which uses a novel approach for dynamically supporting user preferences on time and accuracy.

ACM GIS '08 , November 5-7, 2008. Irvine, CA, USA

To automate time/accuracy tradeoff in workflow management, we allow developers to expose an accuracy parameter, e.g., sampling rate. Our system also takes as input arbitrary models for predicting process completion time and error/accuracy propagation of the applications. We study and present the effects of data sampling on the physical accuracy of the output from a specific geoinformatics application: land elevation change. A detailed discussion, including algorithms and extended examples, for materials presented herein can be found in [4].

2. SYSTEM OVERVIEW

A conceptual view of our system is shown in Figure 1. Here we briefly summarize the major system components while details can be found in [3, 5]. Most autonomous systems for information retrieval, including ours, require semantic descriptions of datasets and services. For geospatial datasets, CSDGM (Content Standard for Digital Geospatial Metadata) [8] is recommended by domain authorities. CSDGM annotates geospatial data with such descriptions as area coverage, date of creation, coordinate system, etc.

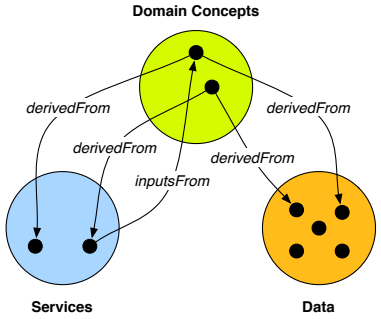

Figure 2: Ontology for Domain Description

Additional semantic annotations are required by our system to enable autonomous determination of service and data suitability when composing workflows. This includes a classification of datasets and services, and a description of their relationships with geospatial concepts. For this purpose, we define an ontotology, shown in Figure 2. Its structure is general, and perhaps even naïve, compared to other such efforts in the geospatial domain [11], but it serves the specific purpose of workflow composition for our system. It is conceivable that its generality carries the possibility for porting our system into other domains simply by swapping in an equally defined ontology. Although, admittedly, we have not yet exploited this potential.

2.1 Ontology Mapping

A key goal is to handle automatic workflow composition against high-level keyword queries. An natural language parser breaks queries down to concepts within the geospatial domain. This is done through mapping keywords onto geospatial concepts defined in the ontology and, when available, assigns user-given values to these concepts. Figure 3 shows a abstraction of this process. Throughout this paper, we focus on processing the following running example query:

```
DEM Query = return land surface change at
(482593, 4628522) between 07/08/2000 and 07/08/2005
```

From the top, the NLP decomposes the string and identifies the query's desired target. Here, land surface and change both map to distinct domain concepts and are merged into a meta-superconcept: land surface change. Their respective edges are intersected and the rest removed. Next, *(482593, 4628522)* is mapped to the *coordinates* concept and both *7/08/2000* and *7/08/2005* to the *date* concept via simple pattern matching.

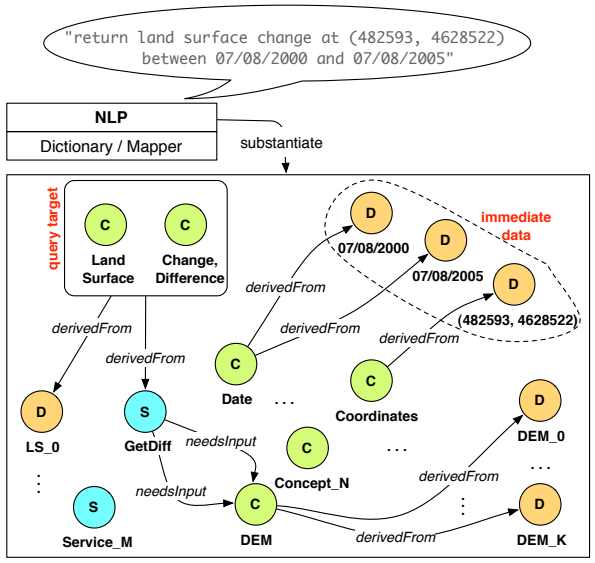

Figure 3: Query Mapping to Ontology

It is evident that the target concept, *land surface change* is derivable from multiple sources: via datasets $LS_0 \ldots$ and the service execution of *GetDiff*. This particular service further requires two inputs of concept DEM, which are in turn derived from datasets $DEM_0, \ldots, DEM_K$. Naturally, processes may expect preconditions to be met before carrying out execution. *GetDiff*$(file_1, file_2)$ requires $(file_1.date \prec file_2.date \wedge file_1.location = file_2.location)$. Mechanisms for supporting such preconditions are discussed further in [3].

2.2 Workflow Composition

After query-to-concept decomposition, appropriate services and datasets are selected for use via consultation with the ontology. Starting with the target concept, our algorithm follows *derivedFrom* edges in the ontology to identify appropriate services and/or datasets[§]. When a service is visited, its input might again refer to concepts within the domain, and the algorithm is called recursively on the sub-concept until all paths end on data nodes. A workflow is thus built in this manner: on visiting each relevant service or data node, it is recorded before invoking Depth-First Search on the same node. This process allows workflows to be defined in the recursive structure,

$$w = \begin{cases} \epsilon \\ d \\ (s, P_s) \end{cases}$$

[§]CSDGM annotated datasets are indexed using a spatio-temporal indexing structure, e.g., a variation of R-Trees.

such that terminals ϵ and d denote a null workflow and a data instance respectively. The nonterminal case involves service description (s, P_s) where s denotes a service with parameter list $P_s = (p_1, \ldots, p_k)$ and each p_i is itself a (sub)workflow.

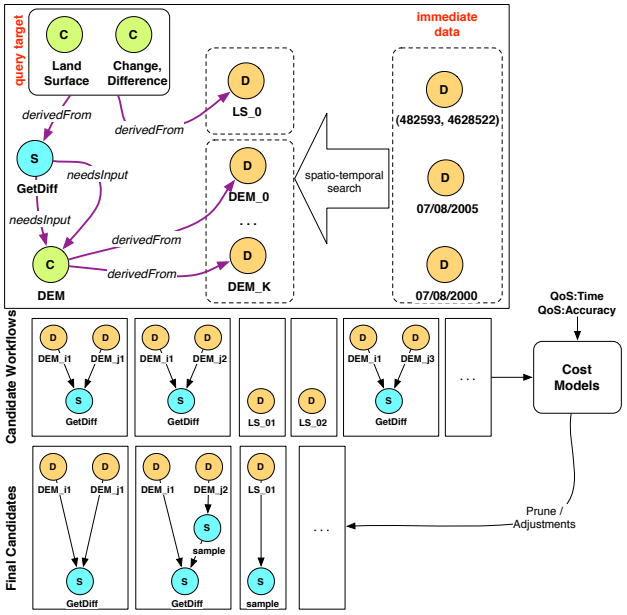

Figure 4: Workflow Composition

The process of combining different datasets with various services can conceivably output many workflows (candidates) capable of answering the same query. The set of candidates is then reduced to those meeting user constraints by aggregating the cost of each. Figure 4 depicts the aforementioned workflow composition phase. A candidate's time cost is predicted as the summation of the time taken from its service executions and data movement.nService execution times are predicted with curve fitting models that have been trained on various-sized inputs, and transfer times are calculated as a function of the links' bandwidths and size of data. Thus, it is also desirable to model a service's expected output size. The candidate's accuracy cost is defined as a function of physical errors propagated by the specific application. Discussions on this topic can be found in [4], but we briefly present a case study in the context of our running example.

3. CASE STUDY: DEM QUERY

The DEM Query involves two digital elevation model files DEM_1 and DEM_2. A DEM is an $m \times n$ grid with elevation, e.g., Z-values, at each point. The queried information can then be obtained simply by taking the difference between the matrices.

3.1 Modeling Error Propagation

The impetus behind modeling physical errors is two-fold. For obvious reasons, an error prediction model is desirable, if not necessary, for providing accuracy-aware user preferences. Another is for supporting workflow accuracy adjustment in attempt to meeting a user's time constraints. For instance, if the user allows for some margin of error, but requires the information in a very short time, the datasets involved in the query could be sampled to reduce transfer and processing time. It is the relationship between sampling and the physical errors propagated as a result that must be managed.

In our example, suppose that DEM_2 is sampled by only considering every kth point. This variation between resolutions, as shown in the left-hand side of Figure 5, presents difficulties to the otherwise trivial computation of elevation difference, $Z_{DIFF} = Z_{DEM_1} - Z_{DEM_2}$. Interpolation, which undoubtedly contributes errors to the calculation, is necessary to compensate the unknown Z-values of the newly reduced DEM_2. It is worth noting that even without sampling, interpolation may still be compulsory for normalizing disparate grid sizes between DEMs measured by heterogeneous instruments.

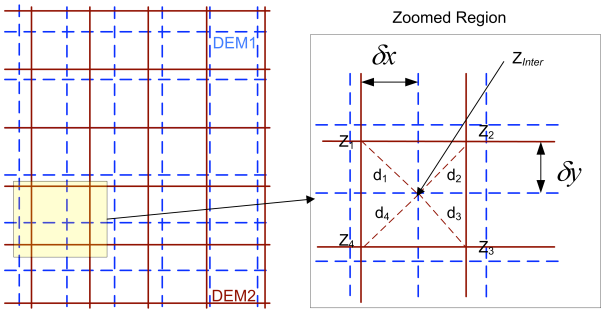

Figure 5: The Overlay of DEM_1 and DEM_2

The Kriging method [6] is implemented for interpolating the value Z_{inter} at an unobserved location corresponding to the grid location on DEM_1 from observations of nearby points on DEM_2. This technique computes the optimal linear unbiased estimator of Z_{inter} based on a stochastic model of the spatial dependence quantified either by the variogram $\gamma(x, y)$ or by the expectation $\mu(x) = E[Z(x)]$, and the covariance function $c(x, y)$ of the random field.

In our case, the Simple Kriging method applies,

$$\hat{Z}(r_0) = \sum_{i=1}^{N} \lambda_i Z(r_i)$$

which assumes the expectation of the random field to be known and relies on a covariance function. It is the weighted sum of the data set and the weights which depend on the semivariogram (e.g., the distance to the prediction location, and spatial relationship among the measured location around the prediction value).

$$\begin{bmatrix} \gamma_{1,1} & \cdots & \gamma_{1,n} \\ \vdots & \ddots & \vdots \\ \gamma_{n,1} & \cdots & \gamma_{n,n} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} \gamma_{1,0} \\ \vdots \\ \gamma_{N,0} \end{bmatrix}$$

The above matrix contains the modeled semivariogram values between all pairs of sampled locations. The λ vector contains the weights and the γ values contain the modeled semivariogram values between the measured and the prediction location. The n equations are solved to obtain the weights, after which the interpolated value is determined. Assuming again that DEM_2 is the coarser-grained grid, we

find DEM_2's interpolated variance at each point with the following equation,

$$\tilde{\sigma}^2_{Z_{DEM_2}} = \sigma^2_Z - \sum_{i=1}^{N} \lambda_i \gamma_{i,0}$$

Finally, the error at each point for the difference calculation is summarized as,

$$\sigma_{Z_{DIFF}} = \sqrt{\sigma^2_{Z_{DEM_1}} + \tilde{\sigma}^2_{Z_{DEM_2}}}$$

Thus, through the above method, we can effectively predict physical errors associated with land elevation change using various sampling rates on DEM datasets. For instance, if the user seeks to optimize execution time for this type of query under a supplied accuracy constraint, the most suitable sampling rate can be deduced by consulting this error model.

3.2 Experimental Evaluation

We continue our discussion with an evaluation of the workflow accuracy adjustment technique. For these experiments, the sampling rate is the exposed accuracy parameter, and the error model described in the previous section has been defined as a function of sampling rate.

Table 1: Errors Incurred by Suggested Sampling Rates

Ideal		Suggested	
Acc %	Error (meters)	Acc %	Error (meters)
10	8.052	11.81	8.052001
20	7.946	21.15	7.945999
30	7.911	28.61	7.911001
40	7.893	34.96	7.892999
50	7.868	50.52	7.867996
60	7.859	60.16	7.858989
70	7.852	70.65	7.851992
80	7.847	80.71	7.847001
90	7.8437	89.07	7.843682
100	7.8402	99.90	7.840197

Table 1 shows the ideal and actual (system provided) error targets. On the left half of the table, the ideal accuracy rate is the user provided accuracy constraint, and ideal error is the physical error value that is to be expected given its respective accuracy preference. The right half of Table 1 shows the actual accuracy and errors given by the system through dynamic manipulation on the DEM sampling rate. Although the error model appears to be extremely sensitive to insignificant amounts of correction, our system's suggestions do not deviate more than 1.246% on average and 5.04% in the worst case.

Next, the DEM query was executed with user given accuracy preferences of 10%, 20%, ..., 100%. DEM files of size 125mb and 250mb were utilized to show the consistency of the algorithm. As shown in Figure 6, the sampling rates, along with the workflow's corresponding execution times at each accuracy preference, expectedly increase as the user's accuracy preference increases.

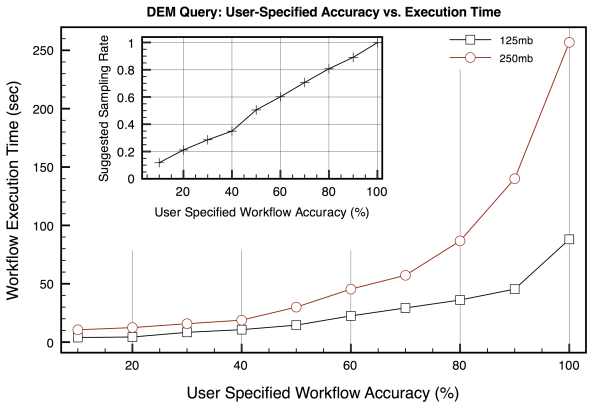

Figure 6: Workflow Accuracy and Corresponding Execution Times

4. CONCLUDING REMARKS

In modern heterogenous computing paradigms, execution time and accuracy of geospatial workflows can vary due to the possibility of having to move and analyze massive datasets. We report a novel approach towards enabling user preferences on time and accuracy in geospaital workflow composition. Through modeling application-specific error and predicting service costs, our system supports bi-criteria optimization by automatically adjusting workflow accuracy.

Acknowledgments

This work was supported by NSF grants 0541058 and 0619041. The equipment used for the experiments reported here was purchased under the grant 0403342.

5. REFERENCES

[1] A. Afzal, J. Darlington, and A. S. McGough. Qos-constrained stochastic work'ow scheduling in enterprise and scienti↓c grids. In *GRID*, pages 1–8, 2006.
[2] N. Alameh. Chaining geographic information web services. *IEEE Internet Computing*, 07(5):22–29, 2003.
[3] D. Chiu and G. Agrawal. Enabling ad hoc queries over low-level geospatial datasets. Technical Report OSU-CISRC-01/08–TR01, The Ohio State University, 2008.
[4] D. Chiu, S. Deshpande, G. Agrawal, and R. Li. Composing geoinformatics work'ows with user preferences. Technical Report OSU-CISRC-08/08–TR44, The Ohio State University, 2008.
[5] D. Chiu, S. Deshpande, G. Agrawal, and R. Li. Cost and accuracy sensitive dynamic work'ow composition over grid environments. In *Proceedings of the 9th IEEE/ACM International Conference on Grid Computing (Grid'08)*, 2008.
[6] N. A. C. Cressie. *Statistics for Spatial Data*. Wiley Series in Probability and Statistics, 1993.
[7] L. Di, P. Yue, W. Yang, G. Yu, P. Zhao, and Y. Wei. Ontology-supported automatic service chaining for geospatial knowledge discovery. In *Proceedings of American Society of Photogrammetry and Remote Sensing*, 2007.
[8] Metadata ad hoc working group. content standard for digital geospatial metadata, 1998.
[9] G. Hobona, D. Fairbairn, and P. James. Semantically-assisted geospatial work'ow design. In *GIS '07: Proceedings of the 15th annual ACM international symposium on Advances in geographic information systems*, pages 1–8, New York, NY, USA, 2007. ACM.
[10] R. Lemmens, A. Wytzisk, R. de By, C. Granell, M. Gould, and P. van Oosterom. Integrating semantic and syntactic descriptions to chain geographic services. *IEEE Internet Computing*, 10(5):42–52, 2006.
[11] R. Raskin and M. Pan. Knowledge representation in the semantic web for earth and environmental terminology (sweet). *Computer and Geosciences*, 31(9):1119–1125, 2005.
[12] M. Wieczorek, R. Prodan, and T. Fahringer. Scheduling of scienti↓c work'ows in the askalon grid environment. *SIGMOD Rec.*, 34(3):56–62, 2005.
[13] P. Yue, L. Di, W. Yang, G. Yu, and P. Zhao. Semantics-based automatic composition of geospatial web service chains. *Comput. Geosci.*, 33(5):649–665, 2007.
[14] L. Zeng, B. Benatallah, A. H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang. Qos-aware middleware for web services composition. *IEEE Transactions on Software Engineering*, 30(5):311–327, 2004.