A New Paradigm for Parameterizations in Atmospheric Models

April 12, 2005

Roger A. Pielke Sr., Toshihisa Matsui, Giovanni Leoncini, and Timothy Nobis*
*Atmospheric Science Department, Colorado State University, Fort Collins, CO 80523
*US Air Force, Air Force Weather Agency, Offutt AFB, NE

Udaysankar S. Nair
NSSTC, University of Alabama at Huntsville
320 Sparkman Dr, Huntsville, AL 35805

Er Lu
Department of Atmospheric Science, University of Arizona, Tucson, AZ 85721

Joe Eastman, Sujay Kumar, Christa Peters-Lidard and Yudong Tian
GEST/UMBC, Hydrological Services Branch, GSFC/NASA Code 614
Greenbelt, MD 20771

Robert L. Walko
Department of Civil and Environmental Engineering
Duke University, Box 90287, Durham, NC 27708-0287

ABSTRACT

The use of look-up-tables (LUTs) to represent parameterizations within atmospheric models is presented. We discuss several approaches as to how the use of LUTs can be optimized in order to retain the physical representation of the parameterization, yet be much more computationally efficient than the parent parameterization from which they are derived.

1. Introduction

Atmospheric models are composed of a dynamical core, which represents advection, the pressure gradient force, gravitational acceleration, and the Coriolis effect, and of a set of parameterizations which represent all other physical processes in the model. Only the dynamical core is based on fundamental physical concepts. For example, the pressure gradient force does not involve tunable coefficients. In contrast, parameterizations, although often based on fundamental concepts of physics, involve tunable coefficients and functions. In atmospheric models, parameterizations are constructed, for example, for deep cumulus convection, stratiform cloud and precipitation processes, subgrid-scale mixing, short- and longwave radiative fluxes, and land-surface interactions (Pielke 2002).

The computational costs of the parameterizations, however, are becoming much greater than for the dynamical core of a model, as parameterizations introduce greater complexity. Matsui et al (2004) reports that parameterizations occupy up to 90% of the wall clock time in a simulation. To reduce this cost, parameterizations, such as for radiative transfer and moist convection, are often called within the atmospheric model only after multiple time steps. In this paper, we outline a procedure to very significantly reduce this computational cost. A discussion of one possible approach is available from Matsui et al. (2004).

2. Methodology

The goal of a parameterization is to mimic the physical process that it is designed to represent without requiring a detailed comprehensive high spatial and temporal resolution model. Since the parameterization itself is an engineering module (i.e., it consists of empirical equations with tunable coefficients derived from observations and/or from a higher resolution model), the goal is to accurately represent the physics it is designed to simulate at a minimum of computational cost. The parameterization concept can be written as:

$$Output\,(\mathbf{x}) = \mathrm{T}[Input(\mathbf{x}),\mathbf{y}]$$

where the dependent variables that need to be computed (the *Output* **x**), are obtained from the *Input* values **x** and the prescribed constants, **y** of the parameterization, through the transfer function, T, which is the parameterization. The constants **y** are obtained from observations and/or a higher resolution model, when the parameterization was created (such as through a fit to the observed data as a function of observed values of **x**). T can provide an instantaneous change (i.e., over a time step) or be inserted over a period of time, such as performed with the Fritsch-Chappell (1980) deep cumulus parameterization. This approach is also common in remote-sensing algorithms (e.g., see the algorithms used by King et al. 1992 and Platnick et al. 2003, as given in Jin et al. 2005).

The current paradigm is to exercise the parameterization, T, *within* the atmospheric model for each gridpoint during the period of model integration.

However, there is another approach that can significantly reduce the cost. The concept is to integrate the parameterization *offline* for the universe of **x**, where the number of values of **x** that is needed depends on the graining that is chosen. This approach can be described as a look-up table (LUT). The LUT, expressed as a multi-dimensional array or fitting function, provides the needed value of T.

There has been an impediment to the use of the LUT technique. The universe of permutations of **x** that are needed produces an enormous number of values. Such large data arrays cannot be accommodated within the available CPU memory of any existing computer. The choice of the modeling community, therefore, has been to include the parameterizations within the atmospheric models and exercise them as the model integration proceeds.

As an alternate approach, we propose the solution of equation (1) offline in order to construct a LUT (or its functional interpolation). The LUT is then applied in lieu of actually running the parent parameterization as the atmospheric model is integrated in time and space. There are several items that permit the feasibility of this approach:

1) *Existing parameterizations are exercised in 1-D vertical columns with the input values of* ***x*** *obtained just from one x-y gridpoint.* This simplifies significantly the number of calculations that must be performed in creating the LUT.

2) *Existing parameterizations include mathematical complexity which is not justified by the skill that it has in defining T.* In other words the dimensionality (i.e. as represented by its degrees of freedom) of the parameterization is much greater than warranted. This means the number of separate values of T can be much less than provided by the parent parameterization. The term *graining* can be used to describe the number of separate values.

3) *Techniques to efficiently access very large data bases have been achieved by the private sector, and these can be applied to quickly assess data in the LUTs.* For example, when we perform a search on an internet search engine (e.g., http://labs.google.com/papers/gfs-sosp2003.pdf), information is very rapidly

obtained. The software BitTorrent (www.bittorrent.com) provides another example of an efficient algorithm to quickly access information from very large databases. A similar approach can be applied here to access data from LUTs.

3. Discussion

To use the LUT-based approach to reproduce essentially all of its values requires the organization and search for the correct LUT from perhaps billions of the available LUT values. To address the limitation of existing computer memory, the minimum size (one case of input and output values) of binary LUT can be stored in the input-oriented hierarchical director with files on the hard disk. To efficiently search the LUT for the required value of T for each situation, programming is required to convert a set of the input variables into the directory and file names, and then let the machine operational system (e.g., UNIX) search the binary LUT instantaneously with the given director and file name. This is the type of procedure used by the business community to access specific values within vast data sets.

There are many ways to store the LUT in order to enable fast retrievals. One such scheme is a hashing technique, enabling the mapping of a unique key to the LUT entry. The hashing techniques are known to be fast lookup techniques compared to other common approaches such as binary, or tertiary tree structures. Improvements to the hash-table implementation of the LUT can be achieved by the use of a relational database. For example, in a specific simulation, if certain entries of the LUT are accessed repeatedly, this information can be used to weigh the LUT lookup, enabling faster turnaround times.

Conceptually this is similar to how the web search engines weighs and caches frequently accessed pages.

When the storage space required for the LUT becomes too large to be handled on a single processor, the use of distributed I/O storage or distributed databases can be employed. A distributed I/O system with large, scalable storage space can be created by taking advantage of easily available and inexpensive commodity resources instead of using large, expensive, centralized storage systems. The large storage space available on a distributed I/O system can be used to create a fault-tolerant, fail-safe LUT storage by the use of multiple data servers and data replication. Parallel, asynchronous LUT retrievals can also be used to improve the performance of the LUT approach.

A hard-disk input-output approach, for example, enables the delta-four-stream Fu-Liou radiation code (Fu and Liou 1992) (30 vertical layer, 140 input, and 33 output) to run 443-time faster than the original code in the Sun-Blade-1000 workstation (Dual CPU: 900 MHz frequency and 8 mb cash size) (Matsui et al. 2004). With this magnitude of speedup, the computational cost of the parameterization becomes negligible in comparison with that of the dynamic core.

This illustration demonstrates that the use of data-based access algorithms provides an efficient procedure to access data from large LUTs.

However, we do not require billions of values to reproduce a parameterization with the accuracy needed for a model. To illustrate the hyperspace space of a transfer function T and how slices through it can be applied to establish the needed resolution of a parameterization, the Louis surface flux parameterization (Louis 1979) is discussed here. The Louis surface flux scheme, although a simple parameterization, still requires considerable storage if used as an LUT. The surface heat flux, as calculated from the Louis surface flux parameterization is a function of the wind (u) and the potential temperature (theta) at a height (z), the surface potential temperature (theta), and the roughness length (z_0). Figure 1 shows one slice through hyperspace where the surface heat flux varies with u and theta while the other variables are fixed (z = 1.0 m, theta = 300 K, z_0 = 0.1 m). The domain of u is set from 0.05 to 2.05 m/s with an interval of 0.02 m/s, and the domain of theta is from 290 to 310 K with an interval of 0.2 K. This graining of the parameterization (with 100 by 100 data points) indicates that this resolution is sufficient to capture the physically important variations that are represented by the parameterization.

In the context of a general parameterization, we do not need billions of data points in an LUT, in order to realistically parameterize a process for use in an atmospheric model.

The dimensionality of the input space of the T operator can be further reduced from the number obtained by simply combining the number of variables with the number of discretization intervals. Such a large number of combinations results in a large number of physically meaningless inputs that result from the mathematical formulation used to construct a parameterization, rather than based on the data used to construct the parameterization. *No parameterization can justify a dimensionality in the billions.*

We are applying the technique of empirical orthogonal functions to the parameterizations as one method to reduce the dimensionality to a physically justified level. The values for T are obtained by combining the output of the individual EOFs (Leoncini and Pielke 2005). A second technique that could reduce the dimensionality is cluster analysis, since it can group input variables that provide outputs within the error range of the parameterization. Thus when a set of input variables is determined to belong to a particular cluster, the output associated with the cluster itself can be provided to the parent model without further computations.

The LUT approach described up to this point can be thought of as the complement of carrying out all parameterization computations during model timesteps. It reduces model runtime computations to an absolute minimum and relies instead on efficient access of pre-computed values from a very large database. The LUT approach also sacrifices some accuracy from the parent parameterization because it must approximate the parameter space with a finite number of data values and interpolation methods between these data values does not capture the full complexity of the parameterization (which may or may not have physical realism).

However, there are levels of compromise between these two extremes that may provide an optimal combination of accuracy and efficiency between the full LUT and the full parameterization method.

One form of compromise is possible for parameterizations of low dimensionality, such as the Louis surface layer parameterization, where parameter space can be adequately covered with relatively few data values (e.g., less than 1 million). Such a small LUT may be computed at model initialization time and stored in model arrays where access of table values is faster than from a disk.

A more important compromise that is often possible is a hybrid approach where LUTs are constructed for subsets of a full parameterization, particularly those that consume the most time. For example, LUTs have been used for years to store pre-computed rates of hydrometeor collisions, melting, and nucleation in the RAMS microphysics parameterization (Walko et al. 1995), while the overall parameterization is computed in the conventional way. Schultz (1995) developed an explicit cloud physics parameterization for use in operational models which encompasses the hybrid LUT concept. These LUTs have only 2 or 3 dimensions and are thus easy to fill at high density for good accuracy. The speed of the overall scheme was increased several-fold to the point where it consumes much less time than the model dynamics. While this speed does not match what might be obtained by constructing an LUT of the full microphysics parameterization, the accuracy is improved and the complexity of the LUT is reduced to the point that the hybrid approach is probably the most attractive.

The hybrid LUT approach may be particularly attractive for a parameterization of very high dimensionality, such as a radiative transfer model representing, say, 50 vertical levels. For example, it is probably an impossible task to pre-compute all possible combinations of moist and dry model levels that may occur, and thus the full LUT approach will be prone to incorrect heating and cooling rates at some model levels for a subset of situations if the LUT does not have fine enough graining of the range of combinations. A hybrid LUT approach could be designed to replace only certain time-consuming calculations in the parameterization while keeping the computations involved in the specific vertical atmospheric profile within the realm of the parameterization.

There is an additional approach that can be applied once either a hybrid or complete LUT is constructed. Since the LUT is a parameterization itself, if new observations (or higher resolution model simulations) are obtained that would warrant the updating of the parent parameterization, that parameterization might be bypassed and the

LUT itself adjusted. This will be a particularly straightforward approach to use when a functional interpolation is applied to represent the LUT.

4. Relevance to Superparameterizations

It has been proposed (Randall et al. 2003) to embed a cloud-resolving model within a larger-scale model in order to improve the accuracy of simulating cloud interactions with the larger-scale model. However, there is an enormous computational cost associated with this approach.

The LUT offers an alternate, much more efficient approach. The 2-D (or 3-D) cloud-resolving model is run off-line in the same manner as applied to create T for the vertical column models. The embedding of a 2-D (or 3-D) cloud-resolving model within a GCM grid, as the GCM is integrated forward in time, can be closely mimicked by the LUT approach, since both are driven by the GCM grid-resolved variables from one grid area. Pielke (1984; pages 263-265) proposed this approach to parameterize the response of cumulus clouds to the larger-scale environment.

An advantage of the superparameterization approach, in contrast with the column parameterizations, is that it can dynamically more directly interact with the parent model at each time step. However, there is an alternate method. Once T is selected from the suite of available off-line cloud-resolving simulations, its values can be fed into the vertical profiles at the GCM gridpoint as they are produced (i.e., after each time step) for the lifetime of the cloud system for that particular value of T. This lifetime is determined for each specific set of GCM input variables from the lifetime that comes out of running the off-line cloud field model that is used to construct T.

This approach of inserting the cumulus cloud effect over time is adopted from the procedure used by Fritsch and Chappell (1980). Comparisons of the much more computationally efficient LUT approach with the use of the superparameterization methodology should be made. With the LUT approach, it should be computational possible to utilize higher-resolution 3-D cloud resolving models, instead of relying on coarser-resolution 2-D cloud resolving models, with a resultant possible improvement in realism of the parameterization. A key aspect of realism enabled by the LUT approach is the ability to represent the full spatial heterogeneity of the land surface, which is known to significantly impact the initiation, growth and maintenance of convective clouds (e.g., Avissar and Liu, 1996).

Acknowledgements

We would like to acknowledge funding provided by NASA CEAS Fellowship No. NAG5-12105, NASA Grant Nos. NNG04GB87G and NNG04GL61G, and DoD Grant DAAD19-02-2-0005 (through a cooperative agreement with the Army Research Lab). The views expressed in this article are those of the author (T. Nobis) and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

References

Avissar, R. and Y. Liu (1996), Three-dimensional numerical study of shallow convective clouds and precipitation induced by land surface forcing. *J. Geophys. Res.* ***101***, 7499-7518.

Fritsch, J.M., and C.F. Chappell (1980), Numerical prediction of convectively driven mesoscale pressure systems, Part I. Convective parameterization, *J. Atmos Sci., 37,* 1722-1733.

Fu, Q., and K.N. Liou (1992). On the correlated k-distribution method for radiative transfer nonhomogeneous atmospheres. *J. Atmos. Sci.*, ***49,*** 2139-2156.

Jin, M., J.M. Shepherd, and M.D. King (2005), Urban aerosols and their variations with clouds and rainfall: A case study for New York and Houston. *J. Geophys. Res.*, submitted.

King, M.D., Y. J. Kaufman, W.P. Menzel, and D. Tanré (1992), Remote sensing of cloud, aerosol, and water vapor properties from the Moderate Resolution Imaging Spectrometer (MODIS), *IEEE Trans. Geosci. Remote Sens.,* ***30***, 2-27.

Leoncini, G. and R.A. Pielke (2005), The use of EOFs to reduce the dimensionality of a radiation LookUp Table, in preparation.

Louis, J.F. (1979), A parametric model of vertical eddy fluxes in the atmosphere. *Bound.-Layer Meteor., 17,* 187-202.

Lu, E. (2004), Louis surface flux scheme and look-up table approach for parameterization, (http://blue.atmos.colostate.edu/courses/ErLu_2.pdf)

Matsui, T., G. Leoncini, R.A. Pielke Sr., and U.S. Nair (2004), A new paradigm for parameterization in atmospheric models: Application to the new Fu-Liou radiation code, Atmospheric Science Paper No. 747, Colorado State University, Fort Collins, CO 80523, 32 pp.

Pielke, R.A. (1984), *Mesoscale meteorological modeling,* 1st Edition, Academic Press, New York, N.Y., 612 pp.

Pielke, R.A., Sr. (2002), *Mesoscale meteorological modeling*, 2nd Edition, Academic Press, San Diego, CA, 676 pp.

Platnick, S., M.D. King, S.A. Ackerman, W.P. Menzel, B.A. Baum, J.C. Riédi and R.A. Frey (2003), The MODIS cloud products: algorithms and examples from Terra, *IEEE Trans. Geosci. Remote Sens., 41,* 459-473.

Randall, D. A., M. Khairoutdinov, A. Arakawa, and W. Grabowski (2003), Breaking the cloud-parameterization deadlock. *Bull. Amer. Meteor. Soc.,* ***84****,* 1547-1564.

Schultz, P. (1995), An explicit cloud physics parameterization for operational numerical weather prediction. *Mon. Wea. Rev.* ***123****,* 3331-3343.

Walko, R.L., C.J. Tremback, R.A. Pielke, and W.R. Cotton (1995), An interactive nesting algorithm for stretched grids and variable nesting ratios. *J. Appl. Meteor., 34*, 994-999.

List of Figures

Figure 1: Surface heat flux calculated from wind (u, m/s) and potential temperature (theta, K) at z = 1.0 m. The u is in domain of 0.05 to 2.05 m/s with an interval of 0.02. The domain of theta is 290-310 K and the interval is 0.2 K. The surface potential temperature theta = 300 K and the roughness length z_0 = 0.1 m (from Lu 2004).

Figure 1

MAY 02 2005

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
28.Apr.05

3. REPORT TYPE AND DATES COVERED
MAJOR REPORT

4. TITLE AND SUBTITLE
A NEW PARADIGM FOR PARAMETERIZATIONS IN ATMOSPHERIC MODELS

5. FUNDING NUMBERS

6. AUTHOR(S)
MAJ NOBIS TIMOTHY E

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
COLORADO STATE UNIVERSITY

8. PERFORMING ORGANIZATION REPORT NUMBER
CI04-1055

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
THE DEPARTMENT OF THE AIR FORCE
AFIT/CIA, BLDG 125
2950 P STREET
WPAFB OH 45433

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Unlimited distribution
In Accordance With AFI 35-205/AFIT Sup 1

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

14. SUBJECT TERMS

15. NUMBER OF PAGES
14

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT

18. SECURITY CLASSIFICATION OF THIS PAGE

19. SECURITY CLASSIFICATION OF ABSTRACT

20. LIMITATION OF ABSTRACT

Standard Form 298 (Rev. 2-89) (EG)
Prescribed by ANSI Std. 239.18
Designed using Perform Pro, WHS/DIOR, Oct 94