An economical strategy for storage of chemical kinetics: Fitting in situ adaptive tabulation with artificial neural networks

Article *in* Proceedings of the Combustion Institute · December 2000

DOI: 10.1016/S0082-0784(00)80202-7

4 authors, including:

Javier Blasco
University of Zaragoza
7 PUBLICATIONS **405** CITATIONS

SEE PROFILE

Norberto Fueyo
University of Zaragoza
86 PUBLICATIONS **2,364** CITATIONS

SEE PROFILE

Cesar Dopazo
University of Zaragoza
118 PUBLICATIONS **3,716** CITATIONS

SEE PROFILE

Proceedings of the Combustion Institute, Volume 28, 2000/pp. 115–121

AN ECONOMICAL STRATEGY FOR STORAGE OF CHEMICAL KINETICS: FITTING *IN SITU* ADAPTIVE TABULATION WITH ARTIFICIAL NEURAL NETWORKS

J.-Y. CHEN,[1] J. A. BLASCO,[2] N. FUEYO[2] AND C. DOPAZO[2]
[1]*Department of Mechanical Engineering*
University of California at Berkeley
Berkeley, CA 94720, USA
[2]*Fluid Mechanics Group, CPS/LITEC*
Maria de Luna 3, 50.015 Zaragoza, Spain

Reducing the computational time of chemical kinetics is essential for implementation of realistic chemistry into large-scale numerical simulations. Among the storage-based techniques, the *in situ* adaptive tabulation (ISAT) method features storing and retrieving data during the simulation; therefore, only the needed data are stored. As ISAT is based on linear approximation, the required storage can grow rapidly when a wide range of chemical states is involved, such as occurs in turbulent flames. An economical strategy for storing chemical kinetics data is proposed here by fitting results obtained from ISAT with artificial neural networks (ANN). This concept is explored in this study using a partially stirred reactor (PaSR) with two reduced chemical mechanisms of 9 and 17 reactive scalars. The performance of the ANN fitting is assessed on the basis of accuracy, memory, and CPU time. Test results based on PaSR demonstrate that a significant saving in memory can be realized with the ANN. Both the accuracy and CPU time with the ANN are found comparable with those of ISAT, suggesting great promise for use of ANN in large-scale computations.

Introduction

Modeling of reacting flows with realistic finite rate chemistry would normally require integration of stiff equations for chemical kinetics. However, the required CPU time can be so large that simulations of reacting flows are possible only with simple chemistry. This situation is especially severe for turbulent flows in which a wide spectrum of chemical states can exist. Consequently, development of efficient numerical schemes for fast chemical kinetics calculations is an important prerequisite for the implementation of realistic chemistry into large-scale simulations. Many techniques have been developed in the past, such as the look-up table approach (LUT) [1], the intrinsic low dimensional manifold method (ILDM) [2], the repro-modeling (RP) [3], the *in situ* adaptive tabulation (ISAT) [4], the use of artificial neural networks (ANN) [e.g., 5–7], and the piecewise reusable implementation of solution mapping (PRISM) with quadratic polynomials [8]. Each of these methods aims at reducing the CPU time needed for computing chemical kinetics. Most of these methods store the computed kinetics of reactive species for a large number of chemical states and retrieve the results on demand. The size of storage is dictated by the accuracy as well as by the number of possible chemical states. As such, the required computer memory can be quite large and may become a limiting factor in modeling of practical combustion. When subsequent simulations of similar conditions are performed (e.g., to explore variation of combustor designs or boundary and initial conditions), the above storage-base strategies can be very efficient as many chemical states have already been computed and stored.

However, due to the vast possibilities of chemical kinetics states in turbulent flames, it is difficult to define the entire chemical kinetics space unless the total number of species is very small. Among the existing storage-based approaches, ISAT offers the unique feature of storing and retrieving the chemical kinetics during the simulations. As ISAT uses linear approximation for evaluating retrieved values of chemical kinetics, the number of records can grow significantly if the simulation involves a vast range of chemical states. Other approaches require samplings of chemical kinetic states within the anticipated combustion regimes prior to the simulation and then fit the sampled chemical states by various means, such as ANNs or polynomials. As ANNs are compact nonlinear models, past experience has demonstrated that ANNs can fit chemical kinetics with sufficient accuracy. However, finding the proper sampling domain can be a problem for ANNs since the needed chemical states are often unknown prior to simulations. The above consideration led us to believe that

a combination of ISAT and ANNs may offer a promising strategy to speed up similar calculations with a small memory requirement. First, a simulation with ISAT is run to provide the chemical state samples that will be fit with ANN. Once the ISAT is fit by ANN, repeated simulations with small variations in boundary conditions or inlet conditions will be run mainly using ANN with ISAT to handle new chemical states should they occur. To explore this concept, we examine the merits of fitting ISAT with ANN to save memory. Brief descriptions of ISAT and ANN are given along with details on the fitting strategy. Test runs with a partially stirred reactor (PaSR) are conducted, and the results are analyzed on the basis of accuracy, memory, and CPU time.

In Situ Adaptive Tabulation

The ISAT developed here follows closely the principles proposed by Pope [4], and we have added two new features that enhance the ISAT performance. One of the new features allows variable time interval which is treated as one of the input. The second new feature facilitates an automated trimming of infrequently accessed ISAT when the allocated memory is used up. The essential idea of ISAT can be viewed as an inquiry process with the initial reactive scalars and time interval Ψ^0 as the input and the changes of reactive scalars due to chemical reactions $\Delta\phi$, as the output. An unstructured (binary tree) table is built to store the changes of scalar properties $\Delta\phi(\Psi^0)$. When an inquiry Ψ^q arrives, one locates the nearest point in the table, say, Ψ^p. If Ψ^q lies within an accuracy domain associated with Ψ^p (called the ellipsoid of accuracy, EOA), the following linear approximation is used to calculate the changes of reactive scalars

$$\Delta\phi\ (\Psi^q) = \Delta\phi(\Psi^p) + \partial[\Delta\phi(\Psi^p)]/\partial\Psi\cdot(\Psi^q - \Psi^p) \quad (1)$$

The EOA is the region around Ψ^p such that $\varepsilon = \mathrm{Max}\{|\Delta\phi_1(\Psi^p) - \Delta\phi_1(\Psi^q)|/\phi_{1,\mathrm{nom}}\} < \varepsilon_{\mathrm{tol}}$, where ε is the maximum relative error, $\phi_{i,\mathrm{nom}}$ is the value used for normalization, and $\varepsilon_{\mathrm{tol}}$ is a present value for tolerance. If the point falls outside the EOA, direct integration of chemical kinetics is performed and the outcome is used to evaluate ε. If ε is smaller than the preset value, the EOA is expanded to cover Ψ^q. If not, a new record is generated for Ψ^q. As the acceptable retrieval depends on $\varepsilon_{\mathrm{tol}}$, the EOA decreases with decreasing value of $\varepsilon_{\mathrm{tol}}$.

Artificial Neural Networks

Applications of ANN for modeling combustion chemistry have been reported in many previous works (e.g., Refs. [5–7]) with the intent of building ANN for the entire regime of turbulent combustion. Fig. 1 depicts the basic concept of ANN in fitting combustion chemistry. The inquiry Ψ^q, which may be specified by mixture fraction, reactive scalars, temperature, and time interval, is used to compute the changes of reactive scalars using a stiff solver [9]. A number of inquiries along with their changes are used to train the ANN. The outcomes are the weight values which are used for computing the output of each neuron according to

$$O_i^L = F\left(\sum_{j=1}^{n_{n,L}} w_{ij}^L O_j^{L-1} + \beta_i^L\right) \quad \text{for } i = 1, \ldots, n_{n,L} \quad (2)$$

where O_i^L is the output of the ith neuron of the Lth layer, w_{ij}^L represents the weight value for connecting the jth neuron of the $(L-1)$-th layer and the ith neuron of the Lth layer, β_i^L is the bias value, $n_{n,L}$ is the total number of neurons in the Lth layer, and F is the transfer function. In this study, a hyperbolic-tangent function is used as the transfer function. As ANN are known robust for modeling nonlinear systems, ANN can be fit to ISAT over a region much larger than EOA. A schematic of this concept is shown in Fig. 2 in which we use boxes to denote the extreme values covered by EOA. We will explore the use of ANN for fitting ISAT over many EOAs enclosed by the heavy shaded box to save memory. After the ISAT is built, we first sort the ISAT records into domains according to mixture fraction and temperature (f, T) similar to the approach reported by Blasco et al. [7]. The mixture fraction is divided into 50 bins with fine divisions clustered near the stoichiometric value. The temperature is divided into 30 bins of uniform size. We combine the EOAs of ISAT records falling into the same domain to form an extended region as illustrated by the shaded region in

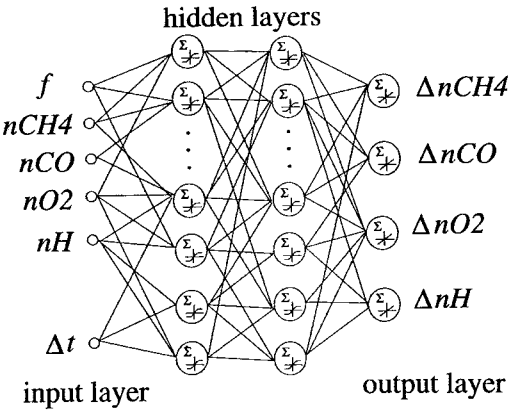

FIG. 1. Schematic of a typical ANN for fitting chemical kinetics.

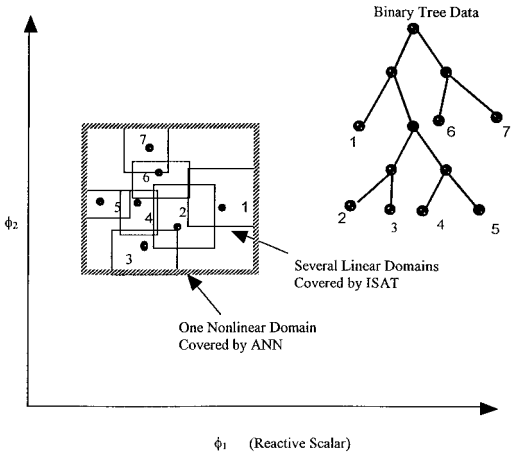

FIG. 2. Schematic of fitting chemical kinetics results obtained from ISAT over several linear domains by ANN with one nonlinear domain.

Fig. 2. Second, within this extended region, random samplings are conducted to generate inquiry points and their chemical changes are computed. Note that most of the bins are empty, and we take only the top 150 sets for ANN fitting. These data are used for training and testing ANN as detailed in Ref. [7]. In the current applications, 8000 and 4000 data are used for training and testing ANN, respectively.

Results and Discussions

A PaSR [10] is used for assessing the merits of fitting ISAT with ANN in order to save memory. Methane and air with stoichiometric proportion are introduced into PaSR with an average residence time of 10 ms. The modified Curl's mixing model is used for modeling the molecular mixing with a mixing frequency of 3 kHz. This leads to a unmixedness level about 0.09. The time step is chosen randomly between 3.3×10^{-5} s and 1.6×10^{-4} s and the initial state of PaSR is assumed to be fully burned stoichiometric mixture. Stochastic simulations with 200 Monte Carlo particles are conducted over a period of 10 residence times to reach the statistically stationary state. Once this state is reached, ISAT is built by running additional 3,000 steps. The maximum number of records of ISAT is set at 20,000. In the present study, the values of $\phi_{i,nom}$ are based on the peak values from a laminar opposed jet flame calculation, and ε_{tol} is set equal to 0.01. Simulations are conducted with a 4-step [11] and a 12-step methane/air reduced mechanisms [12] containing 9 and 17 reactive scalars (species and temperature), respectively. With these two reduced mechanisms, we will explore the scaling relations of memory and CPU time. In this study, we fit ISAT by ANN with

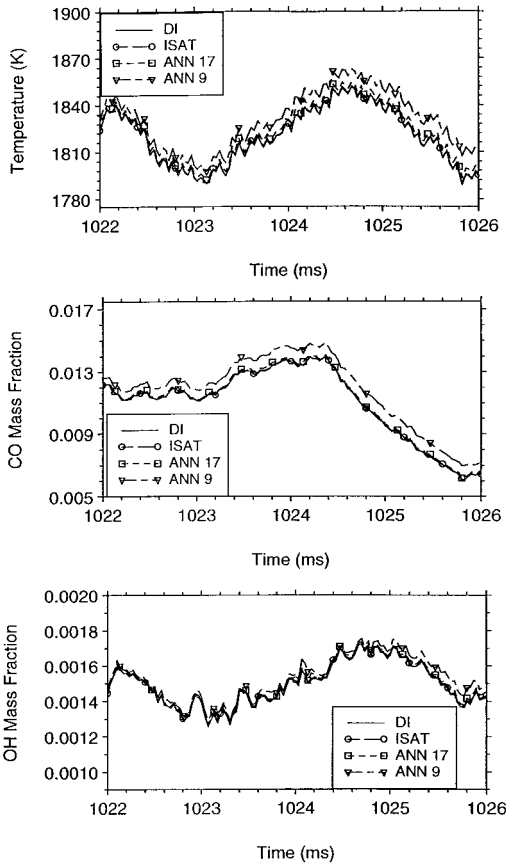

FIG. 3. Comparisons of ensemble averages from a PaSR of stoichiometric methane/air mixture with different methods in computing chemical kinetics, including direct integration (DI), ISAT, and ANN with 9 and 17 neurons.

only one hidden layer, and the total number of neurons in the hidden layer is set equal to the number of reactive scalars. From explorative runs, these choices seem to give a good balance between the CPU time used in building ANN and the resulting accuracy.

PaSR runs with ANN are repeated to reach the statistically stationary state. To explore whether further saving is possible, additional ANN were constructed with the total number of neurons reduced by half so that the memory per ANN is reduced by half. In addition, simulations of PaSR were conducted with direct integration with the 12-step and the 4-step reduced mechanisms to establish the baseline for comparison. Fig. 3 shows comparisons of ensemble averages of temperature, CO, and OH versus time over a short period of 4 ms with the 12-step mechanism for clarity. The computed results for the rest 163 ms show similar trends as the simulations have been carried out over a period of 167 ms.

Overall, the ISAT results and the ANN results with 17 neurons follow closely with those obtained by the direct integration. The ANN results with 9 neurons are seen as less accurate. A similar conclusion can be drawn from the comparisons of the corresponding 4-step results (not shown). Further comparisons of averages conditioned on mixture fraction are shown in Fig. 4. The results from ISAT and ANN with 17 neurons are seen in very good agreement with those from direct integration. Consistent with the time serial comparisons, the results obtained with ANN and 9 neurons are less accurate.

The accuracy may be quantified by the average error over 5000 steps through a period of 167 ms as

$$\varepsilon_{i,\text{method}} = \frac{1}{N}\sum_{i=1}^{N}\frac{|\langle\phi_i\rangle_{\text{DI}} - \langle\phi_i\rangle_{\text{method}}|}{\phi_{i,\text{nom}}} \qquad (3)$$

where $\langle\ \rangle$ denotes the ensemble average over the 200 stochastic particles, the subscript DI means direct integration, and N is the total number of steps. A total of 1 million inquiries were conducted, and the results with the 12-step chemistry are presented in Fig. 5. As seen in the figure, the maximum errors of ISAT and ANN with 17 neurons are less than 0.5%. When the number of neurons is reduced to 9, the maximum error of ANN increases to 3.5%. Similar trends are seen in the results obtained with the 4-step mechanism (not shown).

Table 1 lists the memory and total CPU time per retrieval based on the PaSR simulations. Estimates of the corresponding memory and CPU time are included in Table 2. In the estimates, it is assumed that basic arithmetical operation, such as addition, subtraction, multiplication, and division, needs a single floating point operation. The evaluation of hyperbolic-tangent function is assumed to require 4 floating point operations. As seen in Table 1, the required memory per record for ISAT is about half of that for ANN with 17 neurons, but it is comparable with that for ANN with 9 neurons. As estimated in Table 2, the memory per record for both ISAT and ANN scales as n_s^2 if the total number of hidden neurons is proportional to n_s, where n_s is the total number of scalars. However, the total memory used by ISAT is large due to the large number of records. This is understandable as ISAT uses linear approximation which limits the conditions covered by the EOA. Consequently, the number of records can be quite large in order to cover a wide range of conditions. In the present simulation, a total of 20,000 records are used, and the total memory use is 67 MB and 25 MB for the 12-step and 4-step mechanisms, respectively.

Since one ANN set can cover a much larger domain than the EOA of ISAT, the total number of records (ANN sets) needed is much fewer in comparison with ISAT. In the present study, a total of 150 ANN sets are found to be sufficient to cover the entire domain of ISAT. Hence, the total memory used by ANN with 17 neurons is only about 1.4% of that used by ISAT. When the total neurons is cut in half, the corresponding memory is only 0.7% of that used by ISAT. However, the average errors increase to 1% with the 4-step mechanism and to 3.5% with the 12-step mechanism. These results suggest that the total number of neurons needs to scale with the total number of scalars in order to keep the accuracy comparable with that of ISAT. In view of future applications with large reaction mechanisms, ANN are very promising in cutting down the memory demands.

The CPU time is another important criterion in assessing the methods. In comparison with direct integration, the CPU speed-up factor by using ISAT is

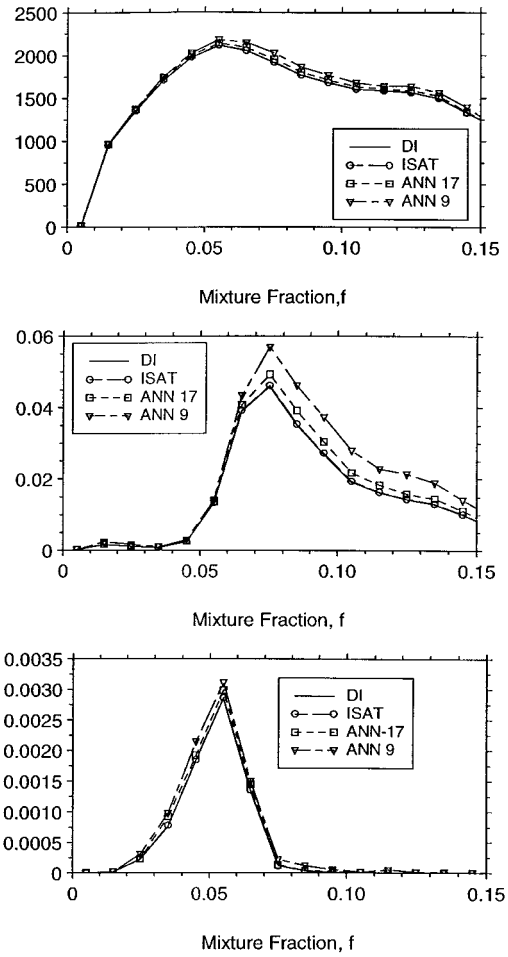

FIG. 4. Comparisons of averages conditioned on mixture fraction from a PaSR of stoichiometric methane/air mixture with different methods in computing chemical kinetics, including direct integration (DI), ISAT, and ANN with 9 and 17 neurons.

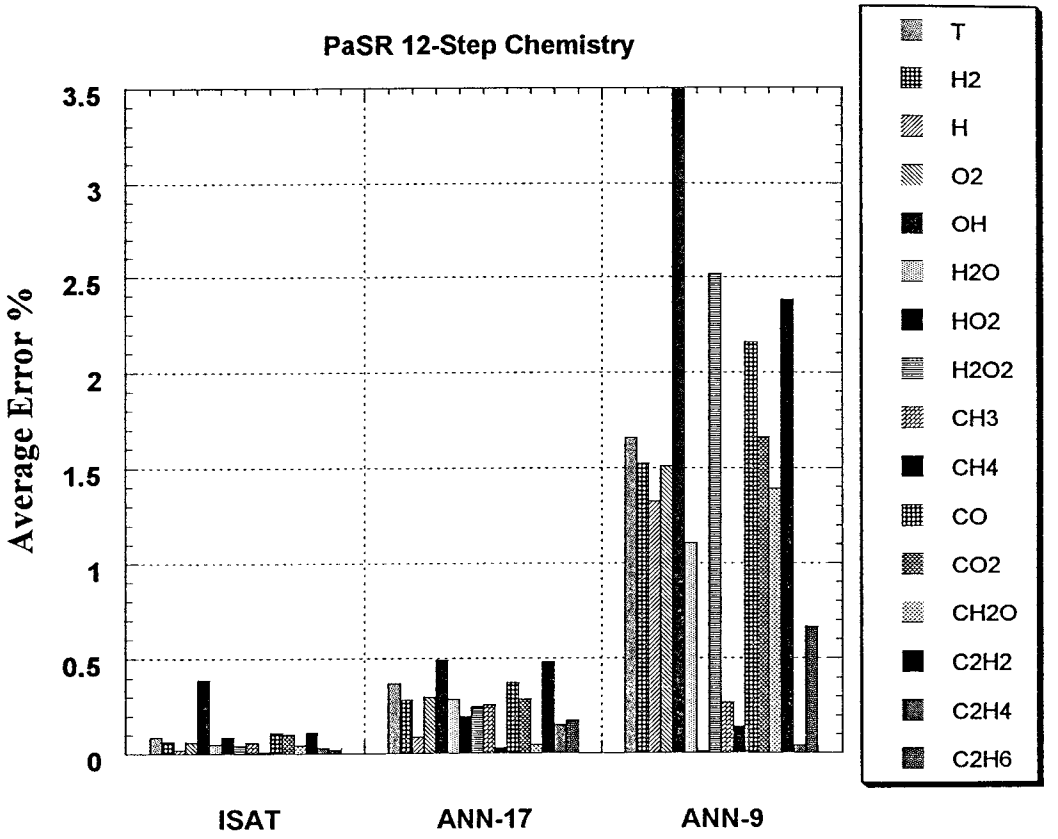

FIG. 5. Average errors over 5000 steps from a PaSR of stoichiometric methane/air mixture using ISAT and ANN with 9 and 17 neurons.

TABLE 1
Memory and CPU-based on PaSR stimulations

		ISAT	ANN ($L = 1$, $n_{n,1} = n_s$)	ANN ($L = 1$, $n_{n,1} = n_s/2$)
12-step ($n_s = 17$)	Total memory	67.523 MB (20,000 records)	0.901 MB (150 sets)	0.465 MB (150 sets)
	Memory/record	3.38 kB	6.01 kB	3.10kB
	CPU time/retrieval (Pentium II 450 MHz, Linux 6.0)	8.712×10^{-5} s	9.171×10^{-5} s	5.357×10^{-5} s
4-step ($n = 9$)	Total memory	25.281 MB (20,000 records)	0.339 MB (150 sets)	0.191 MB (150 sets)
	Memory/record	1.26 kB	2.26 kB	1.27 kB
	CPU time/retrieval (Pentium II 450 MHz, Linux 6.0)	3.312×10^{-5} s	3.362×10^{-5} s	2.101×10^{-5} s

Note: n_s = total number of reactive scalars = total number of species + 1 (temperature); L = total number of ANN hidden layers excluding input and output layer; $n_{n,j}$ = total number of neurons at the jth ANN hidden layer.

TABLE 2
Estimates of memory size and CPU time

	ISAT	ANN (with $L \geq 2$)	ANN (with $L = 1$)
Memory per record	$n_s^2 + 7n_s + 11 \rightarrow O(n_s^2)$	$(n_s + 1)n_{n,1} + (n_{n,L} + 1)n_s + \sum_{i=1}^{L-1} (n_{n,i} + 1)n_{n,(i+1)}$	$(n_s + 1)n_{n,1} + (n_{n,1} + 1)n_s + 4n_{n,1} \rightarrow O(n_s^2)$ if $n_{n,1} \propto n_s$
Evaluation (FLOP)	$2n_s^2 + 3n_s \rightarrow O(n_s^2)$	$2n_{n,1} \cdot (n_s + 1) + 2n_s \cdot (n_{n,L} + 1) + \sum_{i=1}^{L-1} 2(n_{n,i} + 1) \cdot n_{n,i+1} + 4 \sum_{i=1}^{L} n_{n,i}$	$2n_{n,1} \cdot (n_s + 1) + 2n_s \cdot (n_{n,1} + 1) + 4n_{n,1} \rightarrow O(n_s^2)$ if $n_{n,1} \propto n_s$

Note: n_s = total number of reactive scalars = total number of species + 1 (temperature); L = total number of ANN hidden layers excluding input and output layer; $n_{n,j}$ = total number of neurons at the jth ANN hidden layer; n_{record} = total number of records; FLOP = floating point operation.

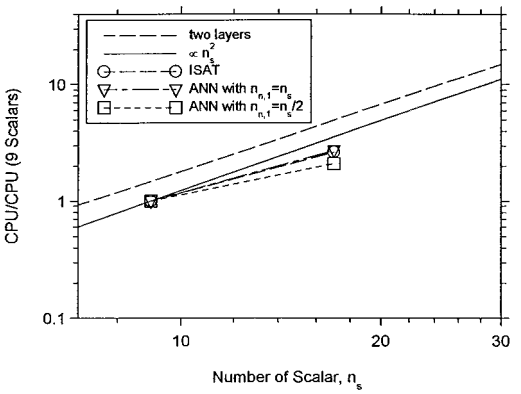

FIG. 6. Scaling of CPU times for ISAT and ANN with the total number of scalars in comparison with the n_s^2 relation.

about 100. As revealed in Table 2, the CPU times per retrieval (including searching and evaluation) for ISAT and ANN with 17 neurons are about the same. As listed in Table 2, the CPU time per evaluation for ISAT scales with n_s^2. For ANN with one hidden layer, the corresponding CPU time is proportional to $n_s n_{n,1}$, where $n_{n,1}$ is the total number of neurons in the first hidden layer. Therefore, the CPU time seen in Table 1 is cut almost in half when the total number of neurons is reduced to half. The CPU time used in searching through the records is roughly proportional to $n_s \cdot \log_2(N_{\text{record}})$ for a perfectly balanced binary tree, where N_{record} is the total number of records. For the current binary tree used, we found the maximum depth of ISAT is about $2 \log_2(N_{\text{record}})$ and the average depth during the retrieval is about $\log_2(N_{\text{record}})$. Therefore, the ratio of searching time over evaluation time scales with $\log_2(N_{\text{record}})/n_s$. These estimates indicate that given a fixed number of records, the fraction of CPU time spent in searching decreases with the total number of scalars. For instance, using the estimates in Table 2 with 20,000 records of ISAT, the fractions are about are 0.24 and 0.15 for the 4-step and the 12-step mechanisms, respectively. Fig. 6 compares the CPU times clocked from the PaSR simulations listed in Table 1 against the scaling relation of n_s^2. The estimates are seen higher than the clocked times as these include searching times which scales as n_s. Also shown are the estimated CPU times for ANN with two hidden layers and the total number of neurons in each layer being equal to n_s. These estimates indicate that ANN with two hidden layers will need 50% more CPU time than one layer ANN; thus, ANN with more than one hidden layer are less attractive in terms of CPU time.

Conclusions

Exploration of fitting ISAT with ANN to save memory usage has been conducted and tested using PaSR. The merits of the ANN fitting are judged on the basis of accuracy, memory requirement, and CPU time. Both ISAT and ANN with a proper number of hidden neurons are shown to give accurate retrieved values. The memory per record for both ISAT and ANN is estimated to scale with n_s^2 if the total number of neurons is equal n_s. As ANN can cover a wider region than the EOA of ISAT, the total memory used by ANN with accurate retrieval is only about 1.4% of that used by ISAT. The CPU times clocked from the PaSR simulations show that the CPU times used by ISAT and ANN are comparable.

The present results demonstrate the potential saving in memory by fitting ISAT with ANN. Future research is warranted to further improve the accuracy and memory requirement of ANN.

Acknowledgments

Research at U. C. Berkeley was supported by NASA grant no. NAG3-2103.

REFERENCES

1. Chen, J.-Y., Kollmann, W., and Dibble, R. W., *Combust. Sci. Technol.* 64:315–346 (1989).
2. Maas, U., and Pope, S. B., *Combust. Flame* 88:239–264 (1992).
3. Turanyi, T., *Comput. Chem. Eng.* 18:45–54 (1994).
4. Pope, S. B., *Combust. Theor. Model.* 1:41–63 (1997).
5. Christo, F. C., Masri, A. R., and Nebot, E. M., *Combust. Flame* 106:406–427 (1996).
6. Blasco, J. A., Fueyo, N., Dopazo, C., and Ballester, J., *Combust. Flame* 113:38–52 (1998).
7. Blasco, J. A., Fueyo, N., Larroya, J. C., Dopazo, C., and Chen, J.-Y., *Comput. Chem. Eng.* 23:1127–1133 (1999).
8. Tonse, S. R., Moriarty, N. W., Brown, N. J., and Frenklach, M., *Isr. J. Chem.* 39:97–106 (1999).
9. Hindmarsh, A. C., *LSODE: Solver for Ordinary Differential Equations,* Lawrence Livermore Laboratory, Livermore, CA, 1980.
10. Chen, J.-Y., *Combust. Sci. Technol.* 122:63–94 (1997).
11. Chang, W.-C., "Modeling of NO_x Formation in Turbulent Flames: Development of Reduced Mechanisms and Mixing Models," Ph.D. thesis, Department of Mechanical Engineering, U.C. Berkeley, Berkeley, CA, 1996.
12. Sung, C. J., Law, C. K., and Chen, J.-Y., *Proc. Combust. Inst.* 27:195–304 (1998).

COMMENTS

Stephen B. Pope, Cornell University, USA. In ISAT, the tabulation error is controlled to be less than a specified error tolerance. In ISAT-ANN, is the error guaranteed to be less than a tolerance?

Author's Reply. Both the ISAT and ISAT-ANN have guaranteed error controls. During the fitting of ISAT by ANN, testing was performed to evaluate the accuracy of ANN. When a specific ANN fails to achieve the desired accuracy, it is not included in the final ANN data bank.

•

A. R. Masri, The University of Sydney, Australia. The accuracy of the ANN calculations depends very much on the training set and how representative it is of the problem that you are trying to solve. Have you checked how good the extrapolation capability of ANN is? If not this is a must. It could be done by generating ANN for a case, applying the same ANN to a different case (with different time scales), and then checking it against ISAT calculations.

Author's Reply. It is difficult to control the error of ANN for extrapolations and is not the intent of this paper. Fitting ISAT by ANN within specific domains and restricting the ANN to these domains ensures accuracy. As explained in the paper, ISAT can be used for points outside the existing domains.