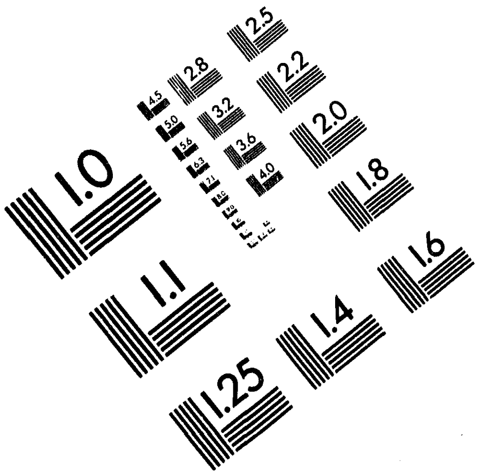
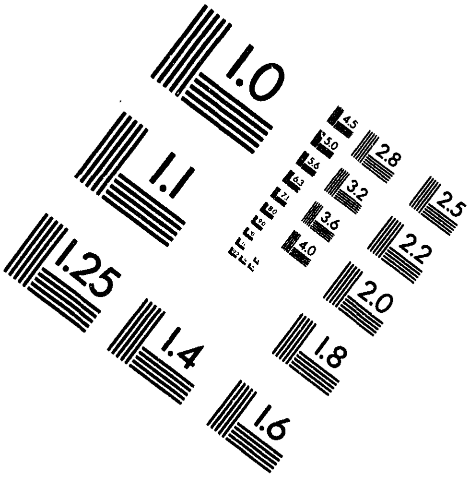
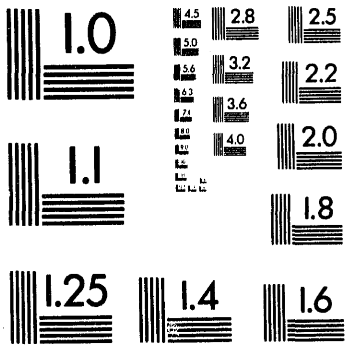
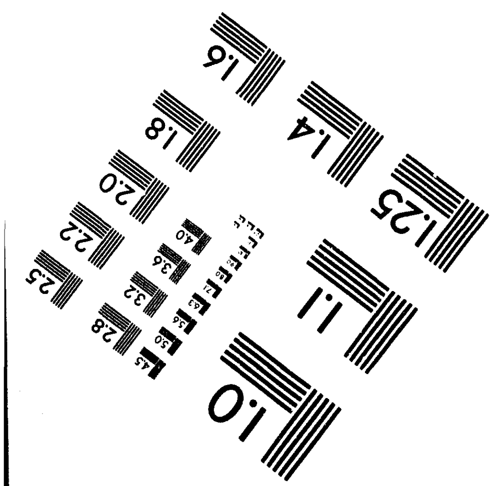


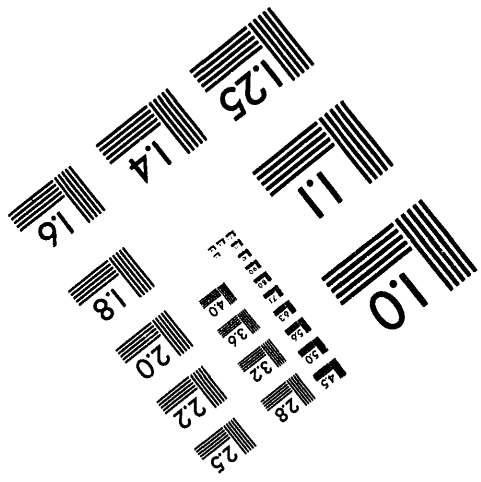


AIIM

Association for Information and Image Management
1100 Wayne Avenue, Suite 1100
Silver Spring, Maryland 20910
301/587-8202







MANUFACTURED TO AIIM STANDARDS
BY APPLIED IMAGE, INC.

1 of 1

Conf-941102--15

LA-UR 94-2523

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36.

TITLE: A PRACTICAL GUIDE TO USING MCNP WITH PVM

AUTHOR(S): G. W. McKinney

SUBMITTED TO: ANS 1994 Winter Meeting, Nov. 13-17, 1994, Washington, DC

Los Alamos

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

MASTER

FORM NO 836 R4
ST NO 2629 5/81

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

A PRACTICAL GUIDE TO USING MCNP WITH PVM

Gregg W. McKinney
Los Alamos National Laboratory
X-6, MS B226
Los Alamos, NM 87545
(505)665-8367

ABSTRACT

The introduction of large workstation clusters and massively parallel computing platforms has lead to an increased interest in the distributed-memory multiprocessing feature of MCNP.™ Such systems offer the potential of hundredfold increases in speedup. This paper provides step-by-step instructions on the creation and use of MCNP on such systems.

I. INTRODUCTION

Distributed-memory multiprocessing has emerged as the leading candidate for achieving TFLOPS (trillion floating point operations per second) performance. Current GFLOPS systems are comprised of workstation clusters or massively parallel processors interconnected by high-speed networks. Such systems employ RISC (reduced instruction set computer) based microprocessors that approach 100 MFLOPS and network speeds that range from several Mbits per second (Ethernet) to hundreds of Mbytes per second (FDDI, HiPPI, ATM, and Fibre Channel). Sequential application programs, such as Monte Carlo radiation transport codes, executed on these virtual machines have attained order-of-magnitude speedup that scales with the number of processors.[1,2] In the near future, this performance will likely increase by yet another order of magnitude, enabling computations once thought unattainable.

Version 4A of the Monte Carlo neutron, photon, and electron transport code MCNP[3], developed by LANL (Los Alamos National Laboratory, Radiation Transport Group), became available through RSIC (Radiation Shielding Information Center) early this year. This version of MCNP supports distributed-memory multiprocessing through the implementation of a communications software package called PVM[4] (Parallel Virtual Machine, version 3.1.4). Using PVM for inter-processor communication, MCNP can simultaneously execute a single problem on a cluster of UNIX-based workstations (e.g., HP, Sun, IBM RS/6000, etc.). With moderate modifications, this capability has been extended to massively parallel systems (e.g., CRAY T3D, CM-5, Meiko, etc.).

The following sections provide a guide on how to create a PVM version of MCNP, how to execute MCNP on a workstation cluster, and the status of MCNP execution on massively parallel systems.

II. CREATING A PVM VERSION OF MCNP

PVM is a network communications package developed by ORNL (Oak Ridge National Laboratory). This public-domain software package supports a variety of networks and numerous UNIX-based systems. To obtain PVM version 3.1.4, simply send electronic mail to **netlib@ornl.gov** with the message "**send index from pvm3**" or contact the Mathematical Sciences Section of ORNL (Al Geist). The User's Guide provides a detailed description of PVM and the installation procedure. Once installed, a **pvm3** subdirectory is created that contains the PVM libraries and interactive command monitor (or console). The libraries can be found in **pvm3/lib/ARCH**, where ARCH represents the architecture name given in the PVM User's Guide (e.g., SUN4, HPPA, RS6K, etc.).

A single MCNP source contains all MCNP installation options for every supported computer system. Desired options are extracted with the PRPR preprocessor prior to compilation. Recently, this process has been automated with release of the MCNP 4A install package. This package has unified the installation procedure on all supported systems (i.e., workstations, mainframes, and personal computers) and includes a README file that walks the user through the process. Figure 1 shows the main menu of the setup program, listing five categories (i.e., Computer System, General, etc.) and a total of ten options. The **Computer System** option should be set first, since the availability of other options depends on this option. The PVM version of MCNP is created by setting the **Multiprocessing** option to **Distributed Memory**. In doing so, the setup program prompts the user for the PVM library path (default is **/usr/lib**) and names (default is **libfpvm3.a** and **libpvm3.a**). Entering the **Process** command at the main menu prompt will complete the setup procedure and initiate the MAKEMCNP script. This script will create the PRPR executable, preprocess and compile the C and FORTRAN source files, and produce the MCNP executable. The install package completes the installation procedure by running the MCNP 25 problem test set.

III. MCNP ON WORKSTATION CLUSTERS

To execute MCNP in a multiprocessing mode on a workstation cluster requires the initialization of the PVM daemon on all nodes identified for use. The PVM console provides an interactive, user-friendly means to accomplish this. Once this is completed, the MCNP executable can be initiated with the desired number of tasks. The PVM daemon handles all message passing between the identified nodes. The MCNP master task, identified with node from which MCNP is initiated, initializes the problem, spawns the subtasks, collects the results, and writes the standard output files. Shortly after spawning the subtasks, the master task sends all the MCNP common block data to the subtask nodes. These initial messages can range from a few Kbytes to several Mbytes in length depending on the application. Subsequently, the master task determines the frequency of required processor rendezvous and assigns an equal workload for each subtask. The master task and subtasks transport the assigned particles using a random number sequence consistent with that of a sequential execution. At processor rendezvous, the subtasks send transport results and await their next assignment. Again, this communication can range from a few Kbytes to several Mbytes. The master task collects the subtask results and determines the workload for the next rendezvous.

The following steps are suggested for executing MCNP with PVM:

(1) Rename the MCNP executable as **mcnp.pvm** and place in **pvm3/bin/ARCH** or provide a link in **pvm3/bin/ARCH** to its actual location. One may also want to provide a link to the PVM console (**pvm**) found in **pvm3/lib/ARCH**.

(2) Create a **.rhosts** file containing all available hosts:

host1 userid
host2 userid
.
.
.

(3) Use the **rup** utility to list the load average of each host and help determine prospective hosts:

rup host1 host2 host3 ...

(4) Start the PVM console and add the desired hosts:

pvm
pvm> add host1 host2 ...
pvm> conf
pvm> quit

The **conf** command shows the virtual machine configuration. Note the last host listed in the configuration for the following step. The **quit** command closes the console but leaves the PVM daemons running in the background.

(5) Log into the last host listed in the PVM configuration [see (e) below] and execute MCNP with the desired number of tasks (this number must be less than or equal to the number of hosts listed in the PVM configuration):

rlogin host4
mcnp inp=input outp=output tasks 4 ...

(6) Terminate the PVM daemons:

pvm
pvm> halt

It is recommended that you terminate the PVM daemons between MCNP runs, otherwise multiple tasks may end up being executed on the same host [see (e) below].

As with the sequential version, the PVM version of MCNP produces one OUTP and RUNTPE file consisting of the combined results from all the tasks. A summary statement is provided in the OUTP file as to the number of particle histories transported by each task (i.e., just prior to print table 126). The following suggestions and warnings are appropriate:

(a) PVM must be installed on each host of the cluster in order for the console to initiate the daemon. Also, the MCNP executable must exist in **pvm3/bin/ARCH** of each host of the cluster. This requirement is simplified if all hosts share a common disk space.

(b) A user must be able to **rlogin**, without a password, to every host for which PVM needs access. Errors encountered when adding PVM hosts are often evidence of this problem. Contact your system administrator if you have created a **.rhosts** file and are still unable to **rlogin** without a password.

(c) During MCNP execution, a warning message "**could only initialize 1 of 4 requested tasks.**" may be a result of: not enough memory to execute MCNP, PVM could not find **mcnp.pvm** in **pvm3/bin/ARCH**, or not enough hosts were available in the PVM configuration. If a subtask host is lost during a calculation, the message "**one or more subtasks have prematurely terminated.**" is printed, and results subsequent to the previous rendezvous are lost.

(d) When using an inhomogeneous cluster (i.e., nodes of differing speeds), the master task should be executed on the fastest host, since, in addition to transporting an equal number of particles, it must also collect results from every subtask.

(e) The requirement of executing MCNP on the last host of the PVM configuration is an artifact of how PVM chooses the subtask hosts. PVM starts with the first host in the configuration and steps down through the list with each spawned subtask. If the master task is started on the first host of the configuration, then the first subtask will also be executed on the first host. With the introduction of PVM version 3.2, the PVMFCONFIG() routine provides host information, allowing for the removal of this nuisance. For this reason, a small MCNP upgrade was developed, in the form of an **install.fix** file, that supports PVM 3.2. This upgrade can be obtained via electronic mail to **gwm@lanl.gov.**

IV. MCNP ON MASSIVELY PARALLEL SYSTEMS

While the execution of MCNP on massively parallel systems is in its infancy, the following statements generally reflect the current state of such computing environments:

(1) The PVM interface is typically integrated into the operating system., replacing user concerns related to PVM initialization with that of system configuration. Interprocessor communication is usually accomplished via an efficient native message-passing software package.

(2) Vendors typically modify PVM to facilitate its use with their native communication package. These PVM derivatives often require MCNP modifications related to task initialization (not to mention other peculiarities like 64 bit integers, etc.). Currently, MCNP upgrades are being pursued for the CRAY T3D, Thinking Machine CM-5, and Meiko.

(3) As the number of processing elements (PE) increases, (e.g., proposed 2048 for CRAY T3D), the current MCNP task configuration will likely prove inefficient. Other approaches include: a tree structure for results collection, a master task that simply assigns workloads and collects results, and multiple "master" tasks.

In summary, the execution of MCNP on these systems tends to be much more straight forward than that on workstation clusters. For example, on the CRAY T3D one simply establishes the configuration (via an environment variable) and executes MCNP without the "tasks" parameter. In this case, the master task is identified with PE 0, and the number of subtasks is provided to MCNP via a PVM call for the system configuration.

REFERENCES

1. G. W. McKinney, "Parallel Processing Monte Carlo Radiation Transport Codes," Proceedings of the 8th ICRS, Arlington, Texas, April 24-28 (1994).

2. G. W. McKinney et al., "Multiprocessing MCNP on an IBM RS/6000 Cluster," Trans. Am. Nucl. Soc., 68, 212 (1993).

3. "MCNP - A General Monte Carlo N-Particle Transport Code, Version 4A," J. F. Briesmeister, Editor, LA-12625-M, Los Alamos National Laboratory (1993).

4. A. Geist et al., "PVM 3 User's Guide and Reference Manual," ORNL/TM-12187, Oak Ridge National Laboratory (1993).

DATE FILMED

10/5/94

END