GRID-ENABLED TREATMENT PLANNING FOR PROTON THERAPY USING MONTE CARLO SIMULATIONS

Ravi Vadapalli[a,*], **Pablo Yepes**[b], **Wayne Newhauser**[c], and **Roger Lichti**[d]

[a]Texas Tech University, High Performance Computing Center, Box 41167 Lubbock, Texas 79409-1167

[b]Rice University, Department of Physics and Astronomy, MS 315, 6100 Main Street Houston, Texas 77005

[c]The University of Texas MD Anderson Cancer Center, Department of Radiation Physics Unit 1202, 1515 Holcombe Boulevard, Houston, Texas 77030

[d]Texas Tech University, Department of Physics, Box 41051, Lubbock, Texas 79409-1051

Abstract

Grid computing is an emerging technology that enables computational tasks to be accomplished in a collaborative approach by using a distributed network of computers. The grid approach is especially important for computationally intensive problems that are not tractable with a single computer or even with a small cluster of computers, e.g., radiation transport calculations for cancer therapy. The objective of this work was to extend a Monte Carlo (MC) transport code used for proton radiotherapy to utilize grid computing techniques and demonstrate its promise in reducing runtime from days to minutes. As proof of concept we created the Medical Grid between Texas Tech University and Rice University. Preliminary computational experiments were carried out in the GEANT4 simulation environment for transport of 25×10^6 200 MeV protons in a prostate cancer treatment plan. The simulation speedup was approximately linear; deviations were attributed to the spectrum of parallel runtimes and communication overhead due to Medical Grid computing. The results indicate that $\sim 3 \times 10^5$ to 5×10^5 proton events with processor core would result in 65 to 83% efficiency. Extrapolation of our results indicates that about 10^3 processor cores of the class used here would reduce the MC simulation runtime from 18.3 days to ~1 h.

Keywords

I. INTRODUCTION

External-beam radiotherapy treatment planning (RTP) is a process in which radiation fields are custom designed to maximize dose to the tumor while minimizing dose to surrounding healthy tissues. Dose calculations for proton RTP are usually performed with analytical

*Ravi.Vadapalli@ttu.edu.

methods[1] because of these methods' high computational speed and adequate accuracy. However, some cases call for dose predictions using the Monte Carlo (MC) method,[2] which provides superior accuracy[3] and additional capabilities, such as absolute dosimetry[4] and predictions of doses from stray and leakage radiation.[5] Several studies suggested that the MC technique will become possible for routine RTP (Refs. 6 through 10) once the obstacle of long computation times is overcome. The MC approach is particularly powerful when the tumor is surrounded by healthy critical organs that must be spared from radiation exposure and traditional analytical approaches are not sufficiently accurate.

In the MC-based RTP process, complex microscopic interactions between protons and the tumor medium are randomly sampled. Many proton trajectories are simulated in order to obtain small random statistical variations in radiation dose predictions in the patient. Ideally, for clinical implementation, the MC simulations are to be accomplished in minutes or seconds. Currently, the major obstacle is the limited availability of computing power and capacity in a single hospital. Multihospital collaborations are a promising avenue for increasing computing capacity. However, secure and fast data movement technologies would be necessary for collaborations among organizations. Grid computing[11,12] is an emerging collaborative computing environment that securely connects computing, storage, visualization, and database resources across several organizational and administrative domains. In addition, the grid security supports proprietary standard and privilege-based resource and data access and their seamless integration, especially when multihospital collaborations are the goal. Multihospital collaborative efforts are also gaining substantial importance in efforts to address challenges in cancer biomedicine such as Campbell and Dellinger[13] and Kanup et al.[14] The Cancer Bioinformatics Grid[15] (caBIG) and Biomedical Informatics Research Network[16] (BIRN) efforts of the National Institute of Health substantiate this vision.

Processing for MC simulations is embarrassingly parallel, and this feature is expected to exhibit linear speedup with an increase in the number of cores. Since there is no interprocess communication in MC simulations, grid computing environments should be ideal for overcoming computational challenges in these types of computations. Downes et al.[17] found that was the case in their preliminary studies on grid computing for intensity-modulated radiotherapy using Radio Therapy Grid[18] (RTGrid).

Despite the promise of grid computing, there is not sufficient evidence that it can mitigate the currently insurmountable clinical challenge of reducing MC simulation runtimes from days to minutes or seconds. Therefore, our goal in this analysis was to demonstrate the scope and power of grid computing for MC simulations in RTP. For this proof-of-concept study, we set up the Medical Grid by connecting three Linux clusters, two from Texas Tech University (TTU) and one from Rice University.

II. METHODS

II.A. Database and Materials Information

We have used a proton field phase-space that was used to treat a prostate cancer patient who was treated at MD Anderson Cancer Center (MDACC), Houston. The treatment technique

was described previously by Fontenot et al.[19] A simulation of the passive-scattering proton therapy treatment unit was implemented previously.[20] The final beam-shaping components of the treatment unit, the collimating aperture, and the range compensator were included. The proton phase-space was generated using the MCNPX code,[21] version 2.7a. The phase-space was input to the GEANT4 (version 4.8.3)toolkit,[22,23] which simulated the transport of protons through the final elements of the treatment unit and into the patient phantom, as described previously.[10]

In this proof-of-concept study, we chose a well-characterized prostate treatment plan that was anonymized previously. About 25×10^6 200-MeVprotons with realistic phase-space distributions were used to simulate the RTP. The number of proton events in the MC simulation was chosen such that the resulting uncertainty in the dose matrix would be <2%. The number of simulation events was chosen so that (a) the serial runtime would be large enough to demonstrate parallel speedup and (b) the least amount of parallel runtime due to heterogeneity in the Medical Grid, when maximum processor cores are employed for the simulation, would be sufficiently large. By virtue of criterion (b), the runtime overhead due to scheduling policies for cluster-specific batch schedulers in the Medical Grid could be ignored without significant loss of accuracy in the total job runtime. To account for processor heterogeneity and communication overhead for the Medical Grid, we defined an aggregate speedup S_{Agg} and efficiency ε_{Agg}.

II.B. Aggregate Speedup and Efficiency

The total events in the simulation were divided equally across the available cores in the Medical Grid. The amount of work assigned to each cluster in the Medical Grid depends on the cluster's size (number of cores). Therefore, the serial runtimes for the clusters may differ from one another.

Let there be N Linux clusters (L_1, L_2, etc.) in the Medical Grid. Let τ_1, τ_2, etc., be the respective serial runtimes for L_1, L_2, etc. Then, the total serial runtime for the entire work is given by $\tau = \sum_{i=1}^{N} \tau_i$. Further, the parallel runtimes may differ among processor cores due to processor heterogeneity and application-specific input conditions, such as phase-space in this problem. Let $(T_P)_i$ be the maximum parallel runtimes for each L_i. Then, the parallel runtime across the grid $T_P = \max[(T_p)_i]$. There-fore, grid-wide speedup S_{Agg} and efficiency ε_{Agg} are given by $S_{Agg} = \tau/(T_P + \delta)$ and $\varepsilon_{Agg} = S_{Agg}/P$, where P is the number of cores in the Medical Grid and δ is the overhead due to grid security (see Sec. II.C).

II.C. Medical Grid Simulation Environment

The Medical Grid represented a typical grid computing environment with strong heterogeneity. The clusters (shown in Table I) in the Medical Grid ran the Virtual Data Toolkit[24]–based grid services through the Globus toolkit[25] (GTK) 4.2.x. The Grid Security Infrastructure[26] (S) and Web Services (WS) Grid Resource Allocation and Management[27] (GRAM) are the main components in the GTK. Pertinent details of other components can be found in Ref. 11. The GSI provides implementations for both the message-level[28] and transport-level[26,28] securities via X.509-based resource and user certificates. In the present work, these certificates were generated by using Simple Authority[29] Certificate Authority

issued by TTU. These certificate settings were internally modified to conform to the Open Grid Forum profile standard.[30] At TTU, we used the Grid User Management System[31] to map user accounts dynamically to a pool of local accounts created for grid operations. At Rice University, the grid users were mapped to local accounts through the grid mapfile.

The grid job management was accomplished through WS-GRAM in GTK 4.2.x. This feature allows writing job submission scripts in extensible markup language (XML). The GTK WS translates the XML scripts into target cluster-specific batch schedulers such as Sun Grid Engine (SGE), Load Sharing Facility (LSF), and Condor[32] in the Medical Grid. High-bandwidth file transfer protocols, such as gridftp and uberftp, were used for staging files in and out of the target machine. The gsissh service, for grid-enabled remote login, was used to access and debug programs on the target machine.

The input database was made available in all clusters in the Medical Grid. This step eliminated the communication overhead due to staging input database files. The simulations were conducted using up to 250 cores across the TTU clusters in the Medical Grid. Postprocessing was carried out using the osg-gate cluster at Rice University. We used uberftp to securely stage-out all outputs as a postprocessing step. This approach substantially reduced the impact of communication latency on the speedup, reported by Downes et al.,[17] and our estimates for S_{Agg} and ε_{Agg} are reasonably accurate.

II.D. Runtime Studies

Several input files covering the spectrum of proton energies that might be used in a typical prostate cancer treatment plan were used in the present work. A customized shell script was used to generate input-specific XML scripts for WS-GRAM, their scheduling, and their management in the Medical Grid. To compute S_{Agg} and ε_{Agg}, the 25×10^6 proton events in the simulation were equally divided into 50 to 250 subjobs per job set. Each subjob was allocated to a core. To investigate the dependence of parallel runtimes on input conditions, a profile run was conducted using 200 subjobs in a job set. We have used a power function of the form $f(T_p) = cP^{-b}$ to fit the parallel runtime in the present work.

III. RESULTS AND DISCUSSION

The serial runtime for the entire simulation was $\tau = 434.9$ h (18.3 days). Figure 1 shows that the parallel runtimes for a job set with 200 subjobs varied in the interval [1.21, 3.26] hours. In addition to the processor heterogeneity, significant spread in parallel runtimes due to the phase-space conditions in input files could have affected S_{Agg} and ε_{Agg}. Table II shows the computed S_{Agg} and ε_{Agg} for several job sets with 50 to 250 subjobs per job set, and cluster site specific runtime efficiencies for computations in the present work. The results indicate that S_{Agg} was approximately linear (Table II). As shown in Fig. 2, extrapolation of the results for S_{Agg} indicate that the simulation runtime can be reduced to ~1 h when $P \approx 10^3$ with ε_{Agg} of ~50%. Figure 3 shows the proton therapy treatment plan constructed from the present work.

The ε_{Agg} was ~83% for ~5×10^5 proton simulation events per core and decreased to ~53% for 1×10^5 proton simulation events per core. Since there is no interprocess communication

in MC simulations, we hypothesize that the loss in efficiency was largely due to the distribution of parallel runtimes. In addition, communications over the Internet in grid computing have low bandwidth and high communication latency; therefore, the overhead due to the message-level and transport-level securities in GSI could also have been a factor in the computed ε_{Agg}. Our preliminary studies also indicate that the number of simulation events of $\sim 3 \times 10^5$ to 5×10^5 that result in ε_{Agg} of 65 to 83% may be optimal for current cluster configurations in the Medical Grid. It would be interesting to investigate load-balancing strategies for improving S_{Agg} and ε_{Agg}.

In the present work, we have employed two cluster sites to best represent a minimal configuration required for the grid computing environment. Therefore, the present work is a true test for grid computing. Extension of this work to involve more cluster sites, which are well controlled and secure, would be a relatively straightforward approach. In the MC method, there is no communication between the subjobs (or cores). Therefore, the impact of the type of the processor interconnect fabric—Infiniband or Gigabit Ethernet shown in Table I—is minimal, if any.

IV. CONCLUSIONS

Our simulations showed that S_{Agg} exhibits an approximately linear trend as the number of cores increases, and by extrapolation of S_{Agg}, we conclude that it is feasible to reduce total simulation runtimes from several days to 1 h when using about 1000 cores. Although this increase in speedup was accompanied by a loss in processing efficiency, we attributed that loss mainly to the spectrum of parallel runtimes due to the input conditions and processor heterogeneity; the communication overhead due to GSI may also have contributed.

Accurate simulations require $\sim 50 \times 10^6$ to 100×10^6 proton events per treatment plan, and several such treatment plans may be necessary for accurate prediction of dose to the tumor geometry. Clusters with about 256 to 512 cores are now common in many hospital environments. By grid-enabling the clusters across several hospitals with related interests in RTP, clinically infeasible MC simulations can be undertaken. In addition, grid computing provides an excellent platform for knowledge sharing through multihospital collaborations.

V. FUTURE WORK

In the present work, we have used user-controlled shell scripts for job management across the Medical Grid. In the future, we plan to deploy a metascheduler that supports dynamic resource management and strategies for load balancing across the Medical Grid. The current effort has helped us better understand the dependence of grid speedup and efficiency on cluster heterogeneity, GSI, and input conditions due to phase-space and will assist us and others in identifying strategies for improvements. We will employ load-balancing strategies to reduce the spread in parallel runtime and improve grid speedup and efficiency. It would be interesting to extend the simulations using 1000 or more cores and validate the observations in the present work.

ACKNOWLEDGMENTS

REFERENCES

1. Schaffner B. Proton Dose Calculation Based on In-Air Fluence Measurements. Phys. Med. Biol. 2008; 53:1545. [PubMed: 18367787]

2. Newhauser W, Fontenot J, Zheng Y, Polf J, Titt U, Koch N, Zhang X, Mohan R. Monte Carlo Simulations for Configuring and Testing an Analytical Proton Dose-Calculation Algorithm. Phys. Med. Biol. 2007; 52:4569. [PubMed: 17634651]

3. Titt U, Sahoo N, Ding X, Zheng Y, Newhauser W, Zhu X, Polf J, Gillin M, Mohan R. Assessment of the Accuracy of an MCNPX-Based Monte Carlo Simulation Model for Predicting Three-Dimensional Absorbed Dose Distributions. Phys. Med. Biol. 2008; 53:4455. [PubMed: 18670050]

4. Koch NC, Newhauser W. Development and Verification of an Analytical Algorithm to Predict Absorbed Dose Distributions in Ocular Proton Therapy Using Monte Carlo Simulations. Phys. Med. Biol. 2010; 55:833. [PubMed: 20071765]

5. Newhauser W, Fontenot J, Mahajan A, Komguth D, Stovall M, Zheng Y, Taddei P, Mirkovic D, Mohan R, Cox J, Woo S. The Risk of Developing a Second Cancer After Receiving Craniospinal Proton Irradiation. Phys. Med. Biol. 2009; 54:2277. [PubMed: 19305036]

6. Koch N, Newhauser W. Virtual Commissioning of an Ocular Treatment Planning System. Radiat. Prot. Dosim. 2005; 115:195.

7. Fippel MA. A Monte Carlo Dose Calculation Algorithm for Proton Therapy. Med. Phys. 2004; 31:2263. [PubMed: 15377093]

8. Paganetti H, Jiang H, Trofimov A. 4D Monte Carlo Simulation of Proton Beam Scanning: Modeling of Variations in Time and Space to Study the Interplay Between Scanning Pattern and Time-Dependent Patient Geometry. Phys Med. Biol. 2005; 50:983. [PubMed: 15798270]

9. Tourovsky A, Lomax AJ, Schneider U, Pedroni E. Monte Carlo Dose Calculations for Spot Scanned Proton Therapy. Phys. Med. Biol. 2005; 50:971. [PubMed: 15798269]

10. Yepes P, Randeniya S, Taddei P, Newhauser W. A Track-Repeating Algorithm for Fast Monte Carlo Dose Calculations of Proton Radiotherapy. Nucl. Technol. 2008; 168:736. [PubMed: 20865140]

11. Foster, I.; Kesselman, C., editors. The GRID 2: Blueprint for a New Computing Infrastructure. San Francisco, California: Morgan Kaufmann Publishers; 2004.

12. Vadapalli, R.; Luo, P.; Kim, T.; Kumar, A.; Siddiqui, S. Demonstration of Grid-Enabled Ensemble Kalman Filter Data Assimilation Methodology for Reservoir Characterization. Proc. 15th ACM Mardi Gras Conf., Workshop Session, Papers from Workshop on Grid-Enabling Applications; January 31–February 2, 2008; Baton Rouge, Louisiana. 2008. ACM; http://portal.acm.org/citation.cfm?id=1341811.1341852

13. Campbell DA Jr, Dellinger EP. Multi-hospital Collaborations for Surgical Quality Improvement. J Am. Med. Assoc. 2009; 302:1584.

14. Kanup P, Garde S, Merzweiler A, Graf N, Schilling F, Weber R, Haux R. Towards Shared Patient Records: An Architecture for Using Routine Data for Nationwide Research. Int. J. Med. Informatics. 2006; 75:191.

15. Cancer Bioinformatics Grid. http://cabig.nci.nih.org.

16. Biomedical Informatics Research Network. http://www.birncommunity.org.

17. Downes P, Yaikhom G, Giddy J, Walker D, Spezi E, Lewis DG. High-Performance Computing for Monte Carlo Radiotherapy Calculations. Philos. T. Roy. Soc. A. 2009; 367:2607.

18. Yaikhom, G.; Giddy, J.; Walker, D.; Spezi, E.; Lewis, D.; Downes, P. A Distributed Simulation Framework for Conformal Radiotherapy. Proc. 22nd IEEE Int. Parallel and Distributed Processing

Symposium (IPDPS); April 14–18, 2008; Miami, Florida. Institute of Electrical and Electronics Engineers; 2008.

19. Fontenot D, Lee A, Newhauser W. Risk of Secondary Malignant Neoplasms from Proton Therapy and Intensity-Modulated X-Ray Therapy for Early-Stage Prostate Cancer. Int. J. Radiat. Oncol. Biol. Phys. 2009; 74:616. [PubMed: 19427561]

20. Zheng Y, Newhauser W, Klein E, Low D. Monte Carlo Simulation of the Neutron Spectral Fluence and Dose Equivalent for Use in Shielding a Proton Therapy Vault. Phys. Med. Biol. 2009; 54:6943. [PubMed: 19887713]

21. Hendricks, J.; Mckinney, G.; Durkee, J.; Finch, P.; Fensin, M.; James, M.; Johns, R.; Pelwitz, D.; Waters, L.; Gallmeier, F. MCNPX, Version 2.6.c. Los Alamos National Laboratory; 2006.

22. Agostinelli S. GEANT4—A Simulation Tool Kit. Nucl. Instrum. Methods A. 2003; 506:250.

23. Allison J. GEANT4 Developments and Applications. IEEE Trans. Nucl. Sci. 2006; 53:270.

24. Virtual Data Toolkit. http://vdt.cs.wisc.edu.

25. Globus Alliance. http://www.globus.org/toolkit/downloads/4.2.1/.

26. Welch V. Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective. 2005 Sep. www.globus.org/toolkit/docs/4.0/security/GT4-GSI-Overview.pdf.

27. WS-GRAM. http://www.globus.org/toolkit/docs/3.2/gram/ws/.

28. Dierks T, Allen C. The TLS Protocol Version 1.0. http://www.ietf.org/rfc2246.txt.

29. SimpleAuthority Certificates. http://simpleauthority.com.

30. Open Grid Forum GFD.125 standard. https://forge.gridforum.org/sf/projects/caops-wg.

31. [current as of May 9, 2010] The Grid User Management System (GUMS). https://www.racf.bnl.gov/Facility/GUMS/index.html
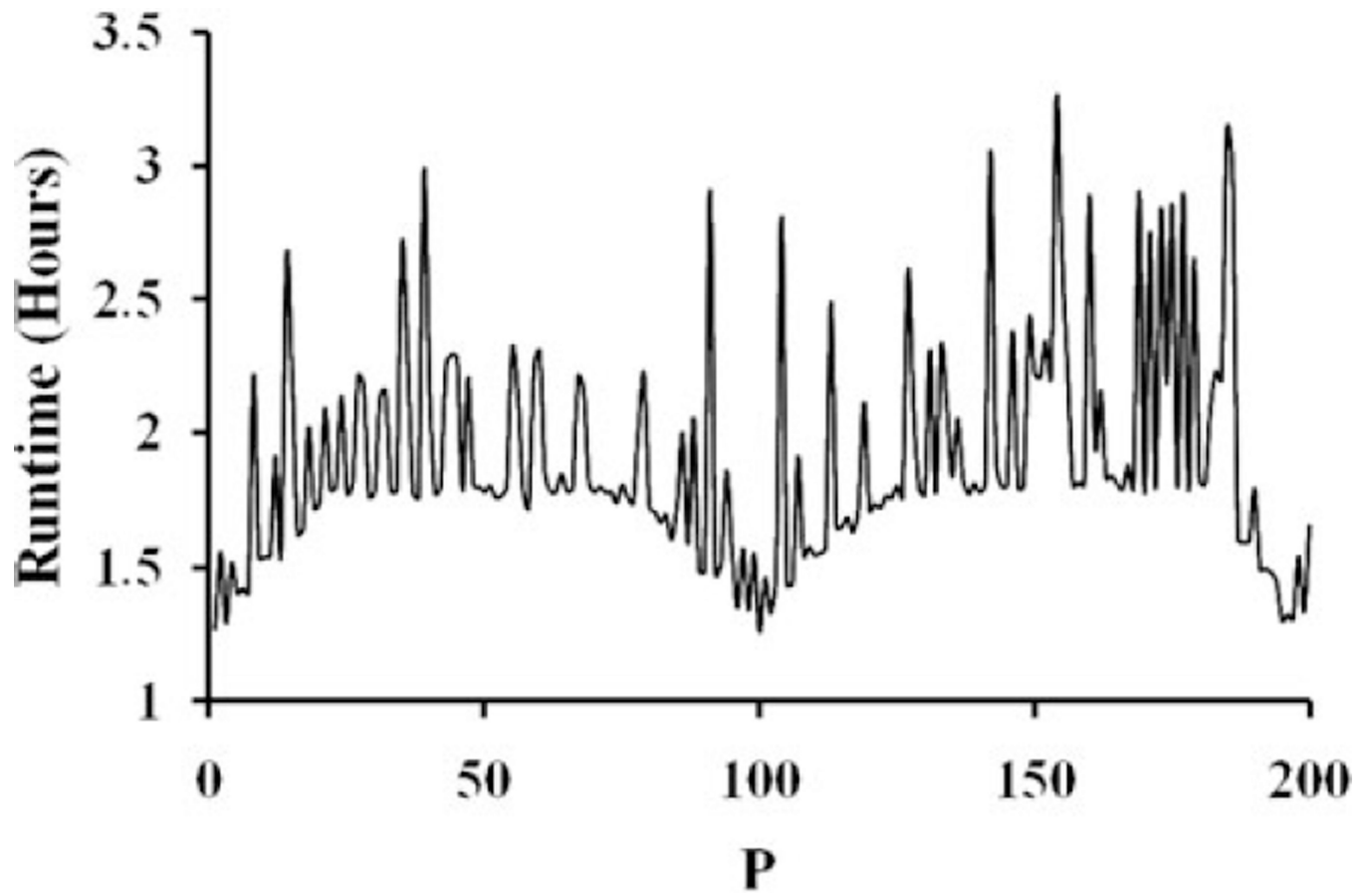
32. Condor. http://www.cs.wisc.edu/condor.

Fig. 1.
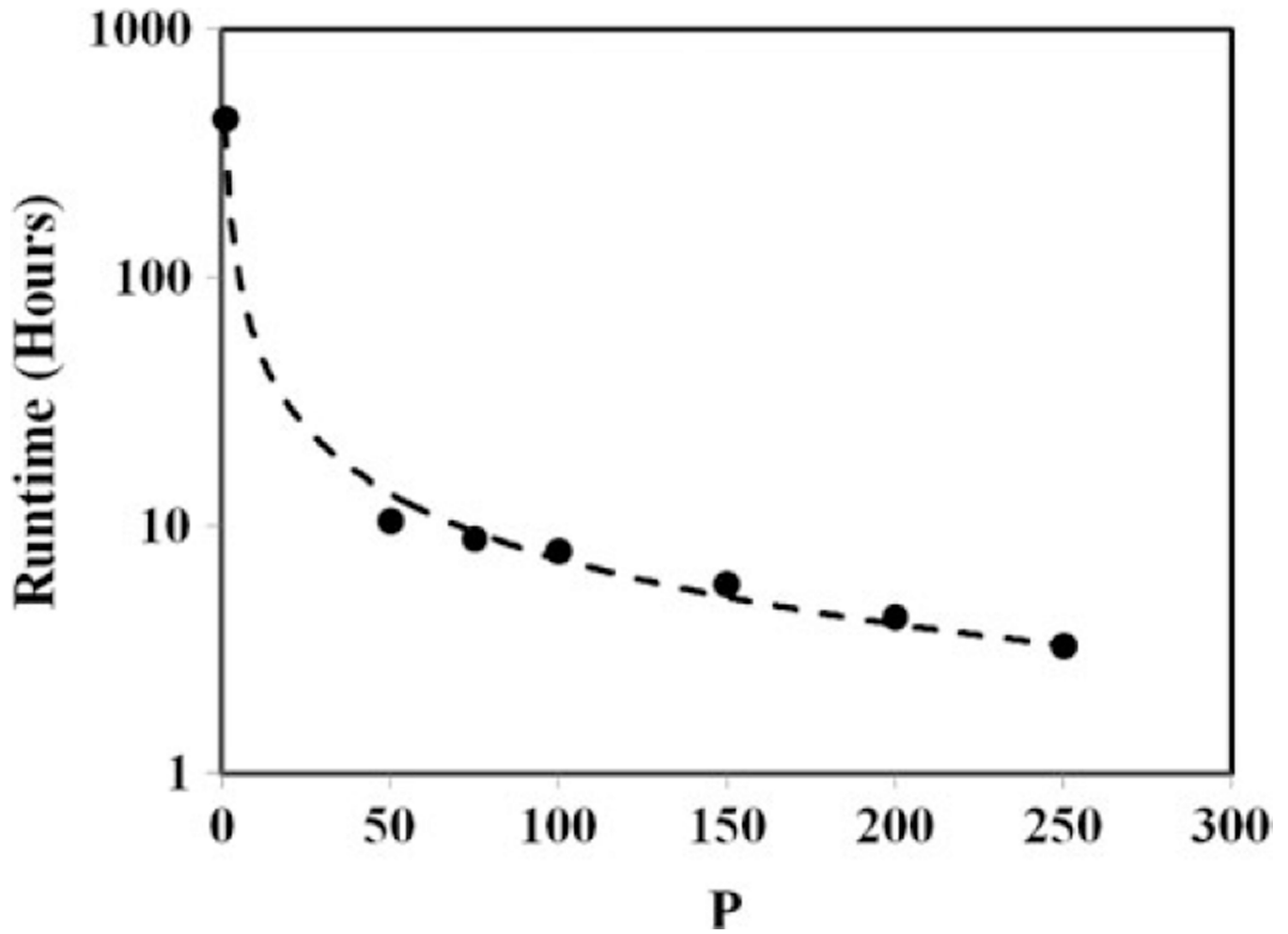Typical distribution of runtimes across the processor cores.

Fig. 2.
Medical Grid–wide speedup. Extrapolation of the results indicated that the parallel runtime T_P would be ~1 h if the number of cores $P \approx 10^3$. (The serial runtime is included in the plot.)
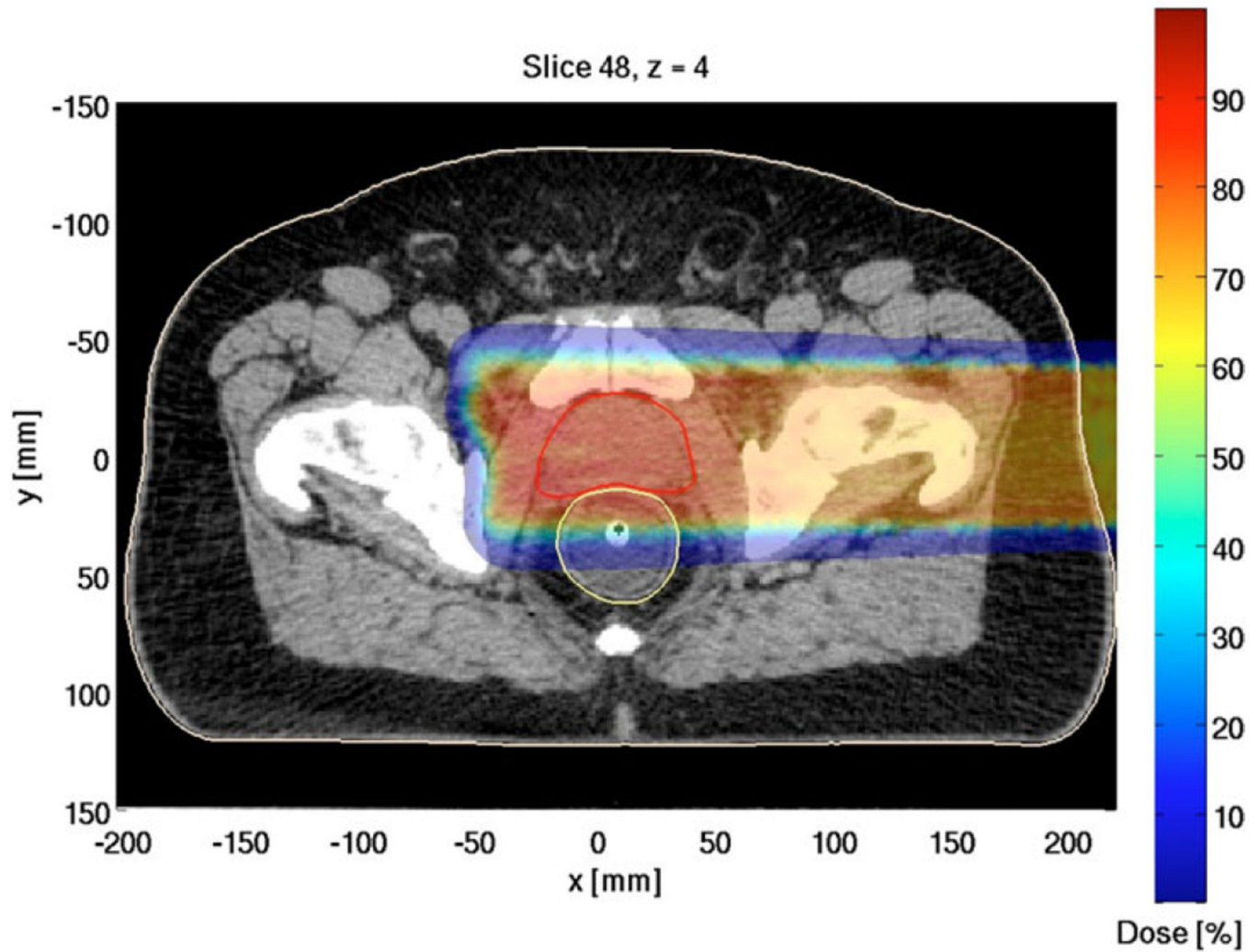
Fig. 3.
Proton therapy treatment plan for prostate cancer resulting from the simulation runs in the current study (with appropriate adjustments).

TABLE I

Technical Specifications of the Clusters in the Medical Grid*

Institution	Cluster Name	Number of Cores	Architecture	Physical Memory(GB/core)	Storage (TB)	Batch Scheduler	Interconnect
TTU	Antaeus	200	Xeon 3.2 and 2.4 GHz Dual-quad; dual-dual core	2	46	SGE	Gigabit Ethernet
TTU	Weland	100	Nehalem 3.0 GHz Dual-quad	2	46	LSF	Infiniband
Rice University	Osg-Gate	8	Intel 2.66 GHz Core duo	2	10	Condor	Gigabit Ethernet

*
*TTU clusters were connected through the Lustre file system.

TABLE II

Aggregated Speedup S_{Agg} and Efficiency ε_{Agg} in GEANT4-Based Monte Carlo Simulations Using the Medical Grid for 200-MeV Proton Distributions for Prostate Cancer[*]

Job Set Size (Number of Cores)	Subjob Size (protons/core)	τ(h) 25×10^6 Protons	T_P (h)	S_{Agg}	ε_{Agg}	ε_A	ε_W
50	500 000	434.9	10.47	41.54	0.83	0.79	0.93
75	333 333	434.9	8.86	49.09	0.65	0.74	0.83
100	250 000	434.9	7.90	55.05	0.55	0.67	0.79
150	166 667	434.9	5.79	75.12	0.50	0.63	0.69
200	125 000	434.9	4.28	101.61	0.54	0.59	0.55
250	100 000	434.9	3.26	133.4	0.53	0.58	0.59

[*] Respectively, ε_A and ε_W are efficiencies of the clusters Antaeus and Weland shown in Table I.