Fast optimization and dose calculation in scanned ion beam therapy
S. Hild, C. Graeff, J. Trautmann, M. Kraemer, K. Zink, M. Durante, and C. Bert

Citation: Medical Physics 41, 071703 (2014); doi: 10.1118/1.4881522
View online: http://dx.doi.org/10.1118/1.4881522
View Table of Contents: http://scitation.aip.org/content/aapm/journal/medphys/41/7?ver=pdfcov
Published by the American Association of Physicists in Medicine

Articles you may be interested in
WEG213CD01: 4D Optimization for Scanned Ion Beam Tracking Therapy for Moving Tumors
Med. Phys. 39, 3970 (2012); 10.1118/1.4736202

Comparative simulations of neutron dose in soft tissue and phantom materials for proton and carbon ion therapy with actively scanned beams
Med. Phys. 38, 3149 (2011); 10.1118/1.3584196

Heuristic optimization of the scanning path of particle therapy beams
Med. Phys. 36, 2043 (2009); 10.1118/1.3121506

Optimization for fast-scanning irradiation in particle therapy
Med. Phys. 34, 3302 (2007); 10.1118/1.2754058

Design of a fast multileaf collimator for radiobiological optimized IMRT with scanned beams of photons, electrons, and light ions
Med. Phys. 34, 877 (2007); 10.1118/1.2431082
Fast optimization and dose calculation in scanned ion beam therapy

S. Hild
Department of Biophysics, GSI Helmholtzzentrum für Schwerionenforschung GmbH, 64291 Darmstadt, Germany; Department of Radiation Oncology, University Clinic Erlangen and Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 91054 Erlangen, Germany; and Institute for Medical Physics and Radiation Protection, University of Applied Sciences, 35390 Giessen, Germany

C. Graeff, J. Trautmann, and M. Kraemer
Department of Biophysics, GSI Helmholtzzentrum für Schwerionenforschung GmbH, 64289 Darmstadt, Germany

K. Zink
Institute for Medical Physics and Radiation Protection, University of Applied Sciences, 35390 Giessen, Germany and Department of Radiotherapy and Radiooncology, University Hospital Giessen-Marburg, 35043 Marburg, Germany

M. Durante
Department of Biophysics, GSI Helmholtzzentrum für Schwerionenforschung GmbH, 64289 Darmstadt, Germany and Faculty of Physics, Technische Universität Darmstadt, 64289 Darmstadt, Germany

C. Bert
Department of Biophysics, GSI Helmholtzzentrum für Schwerionenforschung GmbH, 64289 Darmstadt, Germany and Department of Radiation Oncology, University Clinic Erlangen and Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 91054 Erlangen, Germany

(Received 1 August 2013; revised 21 May 2014; accepted for publication 22 May 2014; published 13 June 2014)

Purpose: Particle therapy (PT) has advantages over photon irradiation on static tumors. An increased biological effectiveness and active target conformal dose shaping are strong arguments for PT. However, the sensitivity to changes of internal geometry complicates the use of PT for moving organs. In case of interfractionally moving objects adaptive radiotherapy (ART) concepts known from intensity modulated radiotherapy (IMRT) can be adopted for PT treatments. One ART strategy is to optimize a new treatment plan based on daily image data directly before a radiation fraction is delivered [treatment replanning (TRP)]. Optimizing treatment plans for PT using a scanned beam is a time consuming problem especially for particles other than protons where the biological effective dose has to be calculated. For the purpose of TRP, fast optimization and fast dose calculation have been implemented into the GSI in-house treatment planning system (TPS) TRiP98.

Methods: This work reports about the outcome of a code analysis that resulted in optimization of the calculation processes as well as implementation of routines supporting parallel execution of the code. To benchmark the new features, the calculation time for therapy treatment planning has been studied.

Results: Compared to the original version of the TPS, calculation times for treatment planning (optimization and dose calculation) have been improved by a factor of 10 with code optimization. The parallelization of the TPS resulted in a speedup factor of 12 and 5.5 for the original version and the code optimized version, respectively. Hence the total speedup of the new implementation of the authors’ TPS yielded speedup factors up to 55.

Conclusions: The improved TPS is capable of completing treatment planning for ion beam therapy of a prostate irradiation considering organs at risk in this has been overseen in the review process. Also see below 6 min. © 2014 American Association of Physicists in Medicine. [http://dx.doi.org/10.1118/1.4881522]

Key words: particle therapy, prostate cancer, adaptive treatment planning, fast dose calculation

1. INTRODUCTION

A world-wide increasing number of particle therapy centers promotes the technical development in the field.\(^1\)\(^2\) Especially the scanned particle beam technique provides target conformal dose distribution.\(^3\) The strong dependency of the beam range on the water equivalent path length (WEPL) of the traversed material however makes this therapy approach especially sensitive to changes of the internal patient geometry. Large changes in patient geometry, if left unconsidered, can lead to insufficient clinical target volume (CTV) coverage and unacceptable organ at risk (OAR) exposure.\(^4\)

The total dose to a voxel in scanned ion beam therapy is a superposition of numerous elementary beams and therefore the homogeneity of the total distribution depends on the precise positioning of each beam.\(^5\) An adaptation of a
treatment plan for scanned ion beam therapy without reoptimization would cause a deformation of the beam spot grid contradicting homogeneous dose coverage. Additional challenges are the proximity of OARs, large range changes caused by moving air cavities, or tissue displacements relative to bony anatomy.

A method to adapt the treatment plan to the daily patient geometry is treatment replanning (TRP). It starts directly after patient positioning and daily CT acquisition, and consists of (a) target and OAR delineation, (b) plan optimization, and (c) forward dose calculation and quality assurance. Long calculation times could reduce the potential benefit originating from this approach as the probability of tumor displacement from its initial position increases with time. An ideal workflow would foresee the patient staying immobilized during the entire process until the daily radiation fraction is delivered to minimize positioning uncertainties.

Treatment planning for scanned beam applications requires complex optimization and dose calculation models in particular for calculating the relative biological effect (RBE) of the respective ions used. Currently, this task takes hours to complete.

Ahunbay et al. report on a reasonable time frame for their adaptation procedure in photon beam therapy of 5–10 min. Additionally, they accomplish contour generation with their setup in less than 5 min. In this work we aim to realize treatment plan generation within these constraints. We improved and investigated the calculation speed of our in-house treatment planning system (TPS) TRiP98 (Refs. 5 and 9) by re-arranging internal code sequences and porting the code to a multicore environment. The manuscript will focus on the computing techniques to achieve this goal and benchmark the improved TPS with data of five prostate cancer patients originally treated during the GSI therapy pilot project.

2. MATERIALS AND METHODS

The objective of reducing the calculation time needed for treatment planning from around 1 h to below 10 min required substantial improvements. The TPS was analyzed by means of a profiling program to identify code sections using the most calculation time and promising the greatest overall impact. Subsequently these sections were first checked for superfluous calls as well as code changes that would increase the calculation speed. Second, the sections were checked for parallelization possibilities.

Modern particle radiotherapy centers often rely on active scanning technology. The target volume is scanned intensity controlled line-by-line in the lateral direction, each line comprising individual beam spots, referred to as beamlets, thus forming a regular grid of beam positions that is often called “raster” (with grid spacing s). Coverage in depth is achieved by changing the ion beam energy in predefined steps, so that the regular grids are further arranged in M so-called isoenergy slices (IES). One of the main tasks of treatment planning is the optimization of particle fluences [i.e., particle number N per area s² (in mm²)] for each grid position to form a homogeneous dose in the designated target volume.

2.A. Treatment planning for particle therapy

During the optimization part particle fluences for 10³–10⁵ grid positions R have to be determined. The resulting total absorbed dose $D_{phys}^i$ to each voxel of the dose matrix is thereby calculated as $D_{phys}^i = \sum_{j=1}^{R} c_{ij} \cdot N_j$ where $N_j$ is the fluence at the jth grid position and $c_{ij}$ are the elements in the dose correlation matrix (DCM) which quantify the dose a beamlet stopping at grid position j contributes to the ith voxel.

Due to nonlinearity and complexity the modulation of the RBE is the most time consuming part of treatment planning for particle therapy. Our TPS incorporates the local effect model (LEM) (Ref. 15) to calculate the RBE for each voxel individually.

2.B. Profiling and code accelerations

An executable of the TPS was recompiled for the IBM operating system AIX 6.1 for which code profiling was enabled via compiler flags. We separately profiled treatment plan optimization and dose calculation using the AIX profiler gprof. A single treatment plan (Patient 5, see below) was used to generate the profiler output (e.g., number of calls per function and total time spend in each function). The top ranked functions by means of time consumption were screened for possibilities to improve their performance and for dispensable calls. Finally, three functional parts were improved:

- interpolation of the particle energy spectra (>50% of total time),
- calculation of the water equivalent path length (>15% of total time), and
- interpolation of the depth–dose distribution (>5% of total time).

Sections 2.B.1–2.B.3 summarize the changes made and we refer to the improved version of the TPS that contains the accelerations as TPS\textsubscript{acc}.

2.B.1. Interpolation of particle energy spectra

As high-energy ions traverse the patient’s tissue, they produce a complex radiation field due to deceleration and nuclear fragmentation, so that each voxel is exposed to a mixture of ions of different charges and kinetic energies. An appropriate physical description is mandatory for the calculation of RBE-weighted dose. The TPS uses precalculated and experimentally verified spectra tables which are part of its base dataset. Spectra for beam energies which are not included in these tables are interpolated. Using a low dose approximation, an interpolation of the complete spectra is not longer needed. Only the closest primary energy to each energy slice, $E_{near}$, has to be interpolated in the spectra table and its Bragg peak position $P_{Bragg}$ has to be calculated. However, due to the exponential relation between particle range and beam energy the TPS uses logarithmic interpolation to find $E_{near}$ causing slow performance of this code section.

The original TPS implementation interpolates $E_{near}$ and $P_{Bragg}$ for every CT voxel that enters in the calculation. In the new implementation, the closest energy and the Bragg peak
position for all energies contained in the individual treatment plan are precalculated once and stored for faster access. The time expensive logarithmic interpolations are hence reduced to a minimum on expense of additional memory consumption, which is, however, negligible compared to the overall memory requirements (e.g., additional 10 byte per IES vs several 100 MB). Improvements in this code section will effect the calculation time of the RBE-weighted dose.

2.B.2. Calculation of the water equivalent path length

Like other treatment planning systems for particle therapy, our TPS also considers anatomic inhomogeneity with the concept of WEPL. As described by Krämer et al., a Siddon like raytracing algorithm that follows basic vector mathematics determines the intersections of the ray with all three orthogonal planes of the CT. Each intersection is assigned a WEPL using an experimentally determined Hounsfield-range look-up table. During dose calculation, the WEPL of typically $10^5$–$10^6$ voxels has to be determined. The number of intersections was reduced by making this routine stop at the geometrical depth of the voxel center rather than calculating a ray through the entire CT cube.

2.B.3. Interpolation of the depth dose distributions

As described in previous publications, the dose accumulated in a voxel $i$ at position $\vec{x}$ can be described by

$$D_i(\vec{x}) = \sum_{j=1}^{R} c_{ij}N_j = c \sum_{k=1}^{M} \sum_{j=1}^{R_k} d(E_k, z_k) \times \frac{N_{j,k}}{2\pi \sigma(z_k)^2} \exp \left( -\frac{r_{ij}^2}{2\sigma(z_k)^2} \right),$$

(1)

where $c$ is a constant, $M$ is the total number of IESs, $r_{ij}$ is the radial distance of voxel position $i$ from the center of beamlet $j$, $\sigma(z_k)$ is the actual width of the Gaussian beam profile of the IES with WEPL $z_k$, $N_{j,k}$ is the total number of particles delivered by beamlet $j$ of isoenergy slice $k$ containing $R_k$ beamlet positions each. The depth dose distribution (DDD) $d(E_k, z_k)$ describes the dose deposition of a beamlet with initial energy $E_k$ at depth $z_k$. They are precalculated, experimentally verified, and stored in normalized DDD tables as part of the base dataset for a sparse subset of potential beam energies. For each beamlet position that contributes to the dose of a voxel, $d(E_k, z_k)$, $\sigma(z_k)$, and additional weighting factors have to be interpolated linearly between the neighboring datasets of the DDD tables. All these quantities depend on the initial energy $E_k$ but not on the lateral beamlet position $j$. Thus, the analysis described in Eq. (1) can be rearranged to

$$D_i(\vec{x}) = c \sum_{k=1}^{M} \frac{d(E_k, z_k)}{2\pi \sigma(z_k)^2} \sum_{j=1}^{R_k} N_{j,k} \exp \left( -\frac{r_{ij}^2}{2\sigma(z_k)^2} \right).$$

(2)

Hence, by equivalent code rearrangements we were able to reduce the total number of calls to the interpolation routine for $d(E_k, z_k)$ to a minimum, i.e., once per IES $k$.

2.C. Multithreading

In addition three larger parts of the treatment planning code were identified to be responsible for approximately 90% of the total computation time. These code sections have been modified to support parallel operations:

- forward calculation of the RBE weighted dose,
- setup of the DCM, and
- optimization of the RBE weighted dose.

In the following, setting up the DCM is considered to be a part of plan optimization and is not analyzed separately. For parallelization, the OpenMP library (http://www.openmp.org) has been used. OpenMP provides a programming interface that allows easy handling of parallel code sections and runtime measurements.

2.C.1. Forward calculation of the RBE weighted dose

The TPS uses a voxel based dose calculation method. Three nested loops drive the calculation routine through the dose cube voxel by voxel. As the dose to each voxel is calculated separately, i.e., is independent from all other calculations, all iterations could in principle be distributed to multiple threads. For a reasonable tradeoff between initializing new threads and fully utilizing the available CPU cores, only the outer loop counting up the transversal slices has been used for parallelization. In the original TPS a beamlet is considered to contribute dose to voxels up to a certain radial distance from its central axes, depending on the beams lateral profile and the WEPL of the voxel center. In addition to this constraint, the TPS only calculates the dose in voxels which are enclosed in the volume directly traversed by any beamlet of all fields plus an adjustable margin.

2.C.2. Calculation of the dose correlation matrix

Each entry in the DCM contains a list of the considered grid positions and their corresponding contribution factor $c_{ij}$ [Eq. (1)]. The calculation of all quantifiers $c_{ij}$ represents a large computational task of independent calculations that can be parallelized extremely well as there are no dependencies.

2.C.3. Optimization of RBE weighted dose

The optimizer needs to work serially in the iterative optimization process, prohibiting multithreading. However, between the single iterations the actual dose as well as the RBE and the dose gradients for each entry in the DCM have to be calculated. These calculations have no dependencies and can therefore be calculated within multiple threads. The voxel group arrangement already used for setting up the DCM is reused for parallelization of the optimization.

2.D. Benchmark tests

We timed the treatment planning process for five prostate cancer patients (Table I) that were treated with an 18.0 Gy (RBE) carbon boost during GST’s pilot project. Each
treatment plan was optimized for a fraction dose of 3.0 Gy (RBE) with a single field uniform dose (SFUD) and an intensity modulated particle therapy (IMPT) (Ref. 7) calculation protocol. For the SFUD approach, each field was optimized separately. For the IMPT approach both fields were optimized simultaneously. Thereafter, the total RBE weighted dose resulting from both fields has been calculated. Both protocols were calculated with \( n_{\text{threads}} = \{1, 4, 8, 12, 16, 24, 32\} \) using our TPS with and without code optimization. The simulation results are compared to the performance of the original TPS implementation. Part of the comparison addressed the accuracy of the optimized treatment plan parameters, i.e., machine data and dose distributions which were performed as part of the internal TPS quality assurance for the studied patients and artificial geometries.

All planning CTs used in this work had a matrix of 256 \( \times \) 256 voxels with a size of 1.953 \( \times \) 1.953 \( \times \) 3.000 mm\(^3\) as they are used in clinical setups. The target volume, as well as bladder and rectum were delineated on the transversal slices. A treatment plan with two lateral opposing fields was setup and optimized on the PTV used for patient treatment slices. A treatment plan with two lateral opposing fields was used in the pilot project. 10 The fields were optimized with a setup and optimized on the PTV used for patient treatment. The target volume, as well as bladder and rectum were delineated on the transversal slices. A treatment plan with two lateral opposing fields was optimized with the OpenMP 3.0 support. Part of the comparison addressed the accuracy of the optimized treatment plan parameters, i.e., machine data and dose distributions which were performed as part of the internal TPS quality assurance for the studied patients and artificial geometries.

### Table I. Treatment plan characteristic of the five prostate patient cases. With \( V_{\text{CTV}} \) representing the volume of the clinical target volume in cm\(^3\). Size DCM being the size of the dose correlation matrix in mega byte, and \( N \) representing the number of elements [voxels/grid positions (RST)] contained in the respective unit.

<table>
<thead>
<tr>
<th>Patient</th>
<th>( V_{\text{CTV}} ) (cm(^3))</th>
<th>( N_{\text{TARGET}} )</th>
<th>( N_{\text{RST}} )</th>
<th>SFUD</th>
<th>IMPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field 1</td>
<td>Field 2</td>
<td>Field 1</td>
<td>Field 2</td>
<td>Size DCM (MB)</td>
<td>Size DCM (MB)</td>
</tr>
<tr>
<td>Patient 1</td>
<td>55.5</td>
<td>4900</td>
<td>4000</td>
<td>4000</td>
<td>180</td>
</tr>
<tr>
<td>Patient 2</td>
<td>53.5</td>
<td>4700</td>
<td>3800</td>
<td>3900</td>
<td>159</td>
</tr>
<tr>
<td>Patient 3</td>
<td>66.4</td>
<td>5800</td>
<td>8400</td>
<td>8100</td>
<td>370</td>
</tr>
<tr>
<td>Patient 4</td>
<td>56.7</td>
<td>4900</td>
<td>6700</td>
<td>6700</td>
<td>260</td>
</tr>
<tr>
<td>Patient 5</td>
<td>72.8</td>
<td>6400</td>
<td>8700</td>
<td>8800</td>
<td>370</td>
</tr>
</tbody>
</table>

Time measurements using OpenMP’s timing routines span the entire optimization and dose recalculation. The time used for reading and writing data via the network is included in the time measurements permitting an estimation of the total duration for the entire treatment planning process.

### 3. RESULTS

The treatment protocols needed an average of 4 h to complete using the original implementation. Enabling profiling output roughly doubles the calculation time. Based on the profiler output, shown in Table II, the code sections for optimization and multithreading were identified. All final benchmark tests, however, are done without profiling and hence are not influenced by this additional time consumption. Treatment plan optimization resulted in identical machine parameter files for the original TPS and the new implementation TPS\(_{\text{acc.}}\). Regarding the dose calculation, the restriction of calculating dose deposition only for voxels enclosed in the volume traversed by any beamlet of all fields plus an adjustable margin led to dose deviations outside the calculated volume and hence outside the target/OAR volume of 5% at most. The margins were automatically determined as 1.8 times the maximum beam width contained in the treatment plan (min/max margin: 9.9/16.2 mm). The results for all dose values within the calculated volume however were identical for TPS and TPS\(_{\text{acc.}}\).

#### 3.A. Profiling and code acceleration

**3.A.1. Interpolation of particle energy spectra**

We were able to drastically reduce the time spent for interpolation of the particle spectra by predetermination and storing \( E_\text{meas} \) and \( P_\text{bragg} \) for each IES contained in a treatment plan. The effect is reflected in a reduced time spend for calculating the RBE effect. Benefits are found to be larger for plan optimization than for dose calculation as the number of calls to this routine during optimization is larger (see Table II).

**3.A.2. Calculation of the water equivalent path length**

A reduction of the total number of elements in the intersection vector saves time when sorting for increasing geometrical depth. The improvements are largest for dose calculation as
TABLE II. Comparison of the most time consuming routines prior and post code optimization. Absolute and relative times are stated for optimization and dose calculation separately. Patient 5 data were used for all measurements.

<table>
<thead>
<tr>
<th>Routine</th>
<th>Original</th>
<th>Optimized</th>
<th>Speed up</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time (s)</td>
<td>Fraction of total time</td>
<td>Time (s)</td>
</tr>
<tr>
<td>SFUD optimization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total time</td>
<td>8135</td>
<td>100.0%</td>
<td>932</td>
</tr>
<tr>
<td>Calculation of RBE</td>
<td>6642</td>
<td>81.7%</td>
<td>368</td>
</tr>
<tr>
<td>IMPT optimization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total time</td>
<td>17210</td>
<td>100.0%</td>
<td>1980</td>
</tr>
<tr>
<td>Calculation of RBE</td>
<td>13838</td>
<td>80.4%</td>
<td>833</td>
</tr>
<tr>
<td>SFUD dose calculation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total time</td>
<td>12368</td>
<td>100.0%</td>
<td>1033</td>
</tr>
<tr>
<td>Calculation of Dphys</td>
<td>2261</td>
<td>18.3%</td>
<td>632</td>
</tr>
<tr>
<td>Calculate WEPL</td>
<td>3607</td>
<td>29.2%</td>
<td>114</td>
</tr>
<tr>
<td>Interpolation DDD</td>
<td>1522</td>
<td>12.3%</td>
<td>106</td>
</tr>
<tr>
<td>Calculation of RBE</td>
<td>3513</td>
<td>28.4%</td>
<td>137</td>
</tr>
<tr>
<td>IMPT dose calculation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total time</td>
<td>11069</td>
<td>100.0%</td>
<td>936</td>
</tr>
<tr>
<td>Calculation of Dphys</td>
<td>1967</td>
<td>17.8%</td>
<td>567</td>
</tr>
<tr>
<td>Calculation of WEPL</td>
<td>3565</td>
<td>32.2%</td>
<td>116</td>
</tr>
<tr>
<td>Interpolation DDD</td>
<td>1264</td>
<td>11.4%</td>
<td>91</td>
</tr>
<tr>
<td>Calculation of RBE</td>
<td>2924</td>
<td>26.4%</td>
<td>117</td>
</tr>
</tbody>
</table>

most of the function calls originate from there (see Table II). On average the modification reduced the vector elements by about 50%.

3.3.3. Interpolation of the depth dose distributions

The total function calls to the routine interpolating the DDDs could be reduced to ≤5.0 % for dose calculation. The speedup is shown in Table II. For optimization the speed up coming from this code improvement is negligible.

Considering all code accelerations, we were able to speedup the calculation time of TRP by factors 10.4 and 9.7 for SFUD and IMPT plans, respectively. A detailed list with a profiling summary and resulting speedup factors is presented in Table II. Performance gain from code acceleration in the following multithreading benchmark analysis was timed using the OpenMP functionality to avoid additional time consumption by profiled code. The results show comparable speedup factors of ≤10 [Fig. 1(c); one thread].

3.B. Multithreading

3.B.1. Optimization

The original TPS reached a maximum mean speedup factor of 10.4 for SFUD and 8.2 for IMPT optimization when using 32 threads [Fig. 1(a)]. Relative speedup factors for TPSacc (normalized to TPSacc with 1 thread) of 3.9 and 4.0 for SFUD (16 threads) and IMPT (8 threads), respectively, are somewhat smaller. The time measurements for plan optimization using the TPSacc show a maximum speedup at eight threads for IMPT and a slow decrease as more threads are being used. Within the limits of the variation of the data points SFUD optimization levels off at eight threads as well, but the speedup does not deteriorate with increasing number of threads. The combination of accelerated code and parallel execution yields speedup factors of 22.9 and 24.3 when calculating with 16 threads and 8 threads for SFUD and IMPT, respectively. Figure 1(a) shows that for IMPT (i.e., large DCMs) using a number of threads larger than eight is somewhat counterproductive with the TPSacc version. This might be due to the fact that in this case the working set is considerably larger and thus the probability of cache misses increases. Another possibility would be access conflicts since the nominal four threads per core do not run completely independent but partially compete for calculation units. However, we did not investigate this issue further within the scope of this paper.

3.B.2. Dose calculation

The speedup factors for dose calculation exceed the ones found for optimization. They range from 16.2 for the original TPS to 189.0 for the combination of accelerated code with the maximum number of threads [Fig. 1(b)]. The relative speedup factors for TPSacc, 13.0 for SFUD and 13.1 for IMPT, are again smaller compared to the ones found for the original TPS using parallel execution. The peak performance in all calculations is reached with the maximum number of threads (32) provided by the system.

3.B.3. Total performance

SFUD calculations spend about 70% of their total time for dose calculation. For IMPT, optimization and dose calculation need about the same fraction of the total time. The speedup for
a complete replanning procedure [Fig. 1(c)] has been calculated based on the total runtime. As dose calculation reached a higher speedup compared to plan optimization SFUD shows a higher speedup in the total performance. The evaluation was extended by one series combining the best results from optimization [8 (IMPT)/16 (SFUD) threads] and the best results from dose calculation (32 threads), labeled with "***" in Fig. 1(c). In terms of absolute time, plan optimization and dose calculation needed an average of 3.5 and 4.5 h to complete with the original implementation, for SFUD and IMPT, respectively. The TPSacc accomplished the same calculation in less than 4 min (SFUD) and about 6 min (IMPT) (Fig. 2).

4. DISCUSSION

With code optimization and multithreading we reduced the calculation time for treatment planning to about 6 min. Reducing the calculation time further might be achievable using a yet larger number of threads, provided concurrent access of shared resources (RAM, Cache, Network) can be kept at a minimum. Additional speedup might be gained by optimization of the memory access patterns (avoiding cache misses, etc.), but since our speedup factors are already remarkable, we have decided to consider this a second order effect requiring a detailed analysis of the memory usage beyond the scope of the current paper.

In general code optimization should be considered first as they reduce the total calculation workload and are independent of hardware. As a second step, parallel execution can be utilized in addition. This however is hardware dependent as the workload is delegated to different cores. Apart from quicker daily treatment planning especially applications requiring multiple dose calculations will benefit from the proposed implementation. Examples are robust treatment plan optimization20 or beam angle optimization. If in addition precise contouring and fast quality assurance methods are provided, same day treatment planning21 or even adaptive treatments like described in principle seem to be possible in the future.

Porting the calculation algorithms to graphical processor units (GPUs) (Ref. 22) is another promising possibility for further speed up. This approach requires major code adaptation as the respective routines have to be split into their basic components to exploit the massive parallelization possible with GPUs.
5. CONCLUSION

We have implemented fast treatment planning into our in-house treatment planning system TRiP98. Code optimization and parallelization improved the treatment planning time for scanned carbon beam therapy by more than a factor of 50 for SFUD and more than a factor 40 for IMPT. The calculation times of 6 min for plan optimization and calculation of the RBE weighted dose are only slightly above the constraints. The TPS$_{acc}$ has reached the boundary where TRP starts to be feasible.

ACKNOWLEDGMENT

The present work was performed in (partial) fulfillment of the requirements for obtaining the degree “Dr. rer. biol. hum.”

---


