Prism Dose Computation Server Communication Protocol*

Jonathan Unger, Ira Kalet, and Mark Martin†

Technical Report 93-08-03

Radiation Oncology Department RC-08
University of Washington
Seattle, Washington 98195

April 11, 1994

Abstract

Dose computation in the Prism radiation therapy treatment planning system is not part of the Prism user interface. The dose computations are handled by separate server programs that run as subprocesses or as network servers. Each dose computation program reads its input data, specifying what is to be computed, from an ASCII stream, and writes its results to an ASCII stream as well. These streams are the standard input and standard output that most programming languages support. The programs are written in standard languages such as ISO Level 0 compliant Pascal, for portability. The protocol allows one to run the programs as servers in interactive environments, or as "batch" jobs (by providing an input file, and simply redirecting the standard input and output streams in the usual way, dependent on the particular operating system). The protocol consists of a format for data and an order in which data are transmitted and received.

*This work was supported in part by NIH grant LM04174 from the National Library of Medicine and NIH contract CM97566 from the National Cancer Institute, and a research grant from General Electric Medical Systems, Milwaukee, Wisconsin.

†It is a pleasure to acknowledge many helpful discussions with and other contributions from Jonathan Jacky.

©1992,1993 by Jonathan M. Unger, Ira J. Kalet, and Mark A. Martin. This work may not be copied or reproduced in whole or part for any commercial purpose. Permission to photocopy in whole or part without payment of fee is granted for nonprofit educational and research purposes provided that all such copies include the following notice: a notice that such copying is by permission of the authors; an acknowledgement of the authors of the work; and all applicable portions of this copyright notice. All rights reserved.

1 Introduction

We have implemented the Prism dose computation as a kind of server, that may be run in a network environment either locally or on a remote host from Prism's user interface. Any dose computation program to be used in the Prism system in this way must comply with the protocol described in this report. A program compliant with the protocol reads from `standard input` several categories of information, then computes a dose distribution based upon this information, and writes the results to standard output. These categories include a description of patient anatomical and related information, a specification of either a dose computation grid geometry or a set of points at which to compute the dose, and a description of a single radiation source (external beam, line source or seed).

This report describes only the protocol for specifying a dose computation and reading the results. For information about dose computation methods, see [1]. For information about the beam data file format, see [2]. For information about the Prism graphical user interface, see [3].

Each time a radiation source description is read, a dose computation program compliant with the protocol computes the dose distribution for that source. The protocol supports two styles of interaction with a program — a mode where results are written back after each radiation source is supplied, and a mode where results are accumulated and then returned after a number of sources have been read. In this way, several radiation sources can be provided sequentially and the results collected to build a dose distribution consisting of multiple types of sources, and/or multiple instances of a given source type. Alternatively, the same radiation source may be sent interactively, providing a feedback loop for the optimization of plans, based on the sequence of returned dose distributions. The program keeps the most recently read instance of each category of information for reference, and effectively updates a category when another instance of that category is read in. This allows for the modification of the anatomical information and the dose grid/point specification, so that a dose computation can be performed for a slightly modified or entirely different case.

The dose computation server program typically will be run as a subprocess of the Lisp environment in which Prism is running, with the standard input and output of the subprocess redirected into Lisp. When run in this configuration, it is started when Prism is started, runs for the duration of the Prism session, and terminates when Prism is exited. The dose computation program can also run as a stand alone process in "batch mode", reading input from one file and writing output to another. As a third possibility, communication with a client process may be handled with TCP/IP sockets through an intermediary program. Though the protocol supports the management of multiple radiation source computations and the combination of the dose matrices by the dose computation program to compute total dose, these functions are usually handled in Prism to provide more flexibility in planning. Other clients of the dose computation server may choose to rely on a dose computation program that sums the results from individual radiation sources together.

The protocol currently supports external beam dose computation, including photon, neutron, and electron beams. The implementation of a program conforming to this protocol is discussed in [4]. Another part of the protocol (and another program) will cover the specification and computation

of dose for line sources and seeds.

2 External beam protocol

The input protocol for the external beam dose computation program consists of seven different categories of information. A program compliant with the protocol can read a succession of instances of these categories in any reasonable order (there is some restriction on the ordering of input to ensure that there is sufficient data to perform a dose computation). Each category is identified by a code, which is an integer:

0. Quit program.
1. Treatment couch position.
2. Anatomical information.
3. Dose grid specification.
4. Dose point list specification.
5. External beam description.
6. Results mode directive.

A protocol compliant dose computation program repeatedly reads in an input category code and performs one or more actions, based on the code read. When it reads in the "quit program" code, it writes out a termination status (see below) and exits. When it reads any of the other six codes, it then reads additional information for that category, and optionally performs additional actions, including computing dose information. Clients of the dose computation program must send information an entire category at a time; the results of sending only a portion of the information in a given category are unpredictable.

The dose computation program keeps the most recently read instance of each category in memory, and each time a category is reread, the information for that category overwrites any previous instance. The program will compute the dose for either a dose grid specification or a dose point list specification, but not both at the same time. Each time the dose computation program reads a dose grid specification, it overwrites any existing dose point list specification in memory. Similarly, when a dose point list specification is read, any existing dose grid specification in memory is overwritten.

When a beam description is read, the dose computation program checks for several types of possible input error, and if none is found, it then performs a dose computation. The results mode determines whether the dose computation program returns normalized results immediately (referred to as *immediate mode*), or whether it accumulates results in memory into a summary structure, to be returned to the client later (referred to as *summary mode*). In either case, all results are returned to standard output. If no results mode is specified in the input stream, the results mode defaults

to immediate mode. The choice of results mode is specified in category 6 of the input stream. If the program is in summary mode, the directive to return the summary results is also specified in category 6 of the input stream.

A detailed description of the order and content of information in each of the seven input categories follows. Note that the content of the external beam description depends upon whether the supplied beam name is for a photon, neutron, or an electron beam. Photon and neutron beams may be collimated with either a rectangular or multileaf collimator. The particular kind of collimator associated with a given beam is determined from the beam name, supplied as part of the protocol described below. For the purposes of dose computation, all collimators with rectangular jaws (i.e., symmetric, combination, and variable) are considered to be of the rectangular collimator type. If the collimator type is rectangular, then a list of zero or more shielding blocks is also specified. Electron beams do not contain collimators or blocks, but do include a description of a cutout contour plane.

The quantities in the input stream that describe machine settings, including the gantry, collimator, and couch angles, the couch lateral, longitudinal, and vertical displacements, the rectangular collimator apertures, the monitor units, the wedge identification and orientation, and block transmission coefficient, are all defined in the same vendor-independent format as is used internally by Prism, described in [4], and not in any vendor-specific or machine-specific coordinate systems.

Multileaf collimator outlines are defined in Prism's gantry space, and are assumed to lie in the plane through the treatment machine isocenter and orthogonal to the beam's central axis.

Block outlines are defined in Prism's collimator space, and are also assumed to lie in the plane through the treatment machine isocenter and orthogonal to the beam's central axis.

Electron cutout contour outlines are also defined in Prism's collimator space, but are assumed to lie in the collimator plane (not the isocenter plane. They are again orthogonal to the beam's central axis.

Anatomical contours and the table position are defined in Prism's patient coordinate system. Only Prism organs, and not Prism tumors and targets may be sent to the input stream, since the latter two objects do not possess density attributes.

All contours supplied to the dose computation program are interpreted as closed loops, but the first point of each contour should not be duplicated in the input stream. Any contour whose first point and last point are identical is incorrectly specified. This currently applies to the organ contours, block outlines, multileaf collimator outline, and the electron cutout contour.

The input is assumed to be an ASCII stream. The format of integers, floating point numbers, and strings in the stream is modeled after the format recognized by standard programming languages, such as ISO Level 0 Pascal. Each integer is specified as a contiguous string of characters between '0' and '9' inclusive, preceeded by an optional plus or minus sign. No integer should contain a decimal point, and none should be greater than 32767 or less than –32768. Each real number is specified in decimal floating point notation or in scientific notation. In decimal floating point notation, each

number is a contiguous string of characters of the form *mmm.nnn*, preceeded by an optional plus or minus sign. Each *m* and *n* are between '0' and '9' inclusive. The *mmm* part represents the whole part of the number, and *nnn* part the decimal fraction. There should be no more than six digits before the decimal point and no more than six digits after it. Each real number *must* contain at least one character before the decimal point, the decimal point itself, and at least one character after the decimal point. Thus, inputs like 123, 456., and .789 all illegal real numbers — they should be specified as 123.0, 456.0, and 0.789 respectively. In scientific notation, the number is a contiguous string of characters of the form *mmm.nnnEpp*. The *mmm.nnn* component is specified in decimal floating point notation and represents the mantissa of the number. The *Epp* component consists of either a capital or lower case 'E', followed by an optional plus or minus sign, followed by a sequence of one or two digits, representing the exponent. The *pp* component must be between -30 and +30 inclusive. Any string in the input stream is delimited by double quotes and may contain any alphanumeric character, including white spaces.

The input stream must contain at least one white space or newline, but no comma or anything else, between each integer, real number, and string. The input stream should contain at least one newline between every 128 characters at the most. This will ensure that the input buffer of the dose computation program does not overflow, and also serves to make the input stream human readable.

The dose computation program ignores all additional white space (tab or space characters) and newlines in the stream.

The text in italics, including a data type for each piece of information, is provided for documentation purposes. The details of each input category are as follows.

Category 0:

- The number 0 (*code: terminate program*)

Category 1:

- The number 1 *(code: couch position follows)*
- X coordinate of table position (*real number*)
- Y coordinate of table position (*real number*)
- Z coordinate of table position (*real number*)

Category 2:

- The number 2 *(code: anatomical information follows)*
- Number of organs (*integer*)
- *For each organ:*
  - Organ density (*real number*)
  - Number of contour planes in organ (*integer*)
  - *For each contour plane:*

        * Z displacement of contour plane (*real number*)
        * Number of contour points in plane (*integer*)
        * *For each contour point:*
            * X coordinate of point (*real number*)
            * Y coordinate of point (*real number*)

Category 3:

- The number 3 *(code: dose grid specification follows)*
- X coordinate of dose grid origin (*real number*)
- Y coordinate of dose grid origin (*real number*)
- Z coordinate of dose grid origin (*real number*)
- Dose grid size in X direction (*real number*)
- Dose grid size in Y direction (*real number*)
- Dose grid size in Z direction (*real number*)
- Number of points in X direction of grid (*integer*)
- Number of points in Y direction of grid (*integer*)
- Number of points in Z direction of grid (*integer*)

Category 4:

- The number 4 *(code: dose point list specification follows)*
- Number of points (*integer*)
- *For each point:*
    - X coordinate of point (*real number*)
    - Y coordinate of point (*real number*)
    - Z coordinate of point (*real number*)

Category 5:

- The number 5 (*code: external beam description follows*)
- Beam name (*text string delimited by double quotes*)
- Gantry angle (*real number*)
- Arc size (*real number*)
- Couch lateral position (*real number*)
- Couch longitudinal position (*real number*)
- Couch vertical position (*real number*)
- Couch angle (*real number*)
- Collimator angle (*real number*)
- Attenuation factor (*real number*)

- Monitor units (*real number*)
- *Beam particle type specific information*
  - *For photon and neutron beams:*
    - *Wedge specific information*
      - Wedge identification number (0 if no wedge) (*integer*)
      - *If the wedge id* $\neq$ *0*
        - Wedge orientation (*real number*)
    - *Collimator type specific information*
      - *For rectangular collimators*
        - X-plus displacement of collimator (*real number*)
        - Y-plus displacement of collimator (*real number*)
        - X-minus displacement of collimator (*real number*)
        - Y-minus displacement of collimator (*real number*)
        - Number of blocks (*integer*)
        - *For each block:*
          - Transmission coefficient (*real number*)
          - Number of outline points (*integer*)
          - *For each outline point:*
            - X coordinate (*real number*)
            - Y coordinate (*real number*)
      - *For multileaf collimators*
        - Number of outline points (*integer*)
        - *For each outline point:*
          - X coordinate (*real number*)
          - Y coordinate (*real number*)
  - *For electron beams*
    - Number of cutout contour points (*integer*)
    - *For each cutout contour point:*
      - X coordinate (*real number*)
      - Y coordinate (*real number*)

Category 6:

- The number 6 (*code: results mode directive follows*)
- Mode directive (*the number 1 (enter immediate mode), 2 (enter summary mode), or 3 (return summary results)*)

The rectangular collimator displacements are labeled *x-plus*, *y-plus*, *x-minus*, and *y-minus* in the beam description above, and represent collimator jaw displacements from the beam's central axis. For example, the x-plus displacement is the distance of the right hand collimator jaw to the right of the beam's central axis. The y-minus displacement is the distance of the lower collimator jaw

below the beam's central axis. The x-minus and y-plus displacements are similarly defined, with respect to the left hand and upper collimator jaws respectively. In all cases, a positive number indicates that the respective collimator jaw is a positive distance away from the central axis. A negative number indicates that a particular jaw is overcentered across the central axis.

The attenuation factor in the beam description is a real number between 0.0 (total opacity) and 1.0 (total transmittance). The raw dose for each output point is multiplied by the attenuation factor and the result is the dose that is reported to the output stream. Users of the protocol can supply an attenuation factor less than 1.0 to compensate for the presence of other objects, such as an alternate tray or headholder, in the beam.

The input protocol supports arc therapy. The arc size is a non-negative real number, which specifies the number of degrees the gantry moves during arc therapy for this field. The gantry angle specifies the beginning position of the gantry in this case. Specifying an arc size of 0.0 indicates that the beam is fixed during treatment.

The dose computation program may be placed into immediate mode or summary mode by specifying category 6 and then the number 1 or 2, respectively. If category 6 is specified and then the number 3 is supplied, the summary results accumulated by the dose computation program are returned (see below for details). The summary results in the program's memory are then set to zero, so a new set of results will be accumulated when more beams are read. When summary mode is initially specified, either a dose grid or a list of dose points (but not both) will be summarized, depending on which of the two has most recently been specified. If another dose grid or points list specification is received by the dose computation program while it is in summary results mode, the current summary grid or list of points is discarded. When the summary results are requested by the client, the current summary grid or points list is returned.

The number of monitor units is specified in the beam description. If the dose computation program has been put in summary mode, the results for each beam are weighted by the number of monitor units before being added to the total. If the dose computation program has been put into immediate mode, the normalized results are returned immediately and the number of monitor units is ignored.

The dose computation program may also read ancilliary information from a dosimetric database. The structure of such a dosimetric database and the manner in which the dose computation reads the information contained therein is not part of the protocol described in this document.

After a beam description is read, the dose computation program attempts to perform a dose computation. It first makes a number of checks to ensure that the data supplied to it are valid (see below). If the data are invalid, the dose computation program terminates. If the data are valid, a dose computation is performed, based upon the current treatment couch position, anatomical information, and whichever of the dose grid or point list specifications is valid at the time. It then writes a success code to standard output, optionally followed by the dose computation results if the program is in immediate result mode. In either case, the program then returns to a state where it is ready to read another input category code. When the dose computation program terminates (in response to a "quit program" input category code or upon encountering an error), it writes a termination code to standard output.

The output codes that the dose computation program writes follow. The success codes are positive integers.

0. Program quitting.
1. Successfully computed dose grid.
2. Successfully computed dose point list.

If the program is instructed to quit (by reading the appropriate input code), or if it encounters any of the errors described below, it writes the quit code to standard output and terminates. Note that not all possible types of errors are considered here, and the behavior of the dose computation program when encountering such an error is unspecified.

The input errors that the dose computation program recognizes are:

Insufficient data. This error occurs when the program is unable to perform a dose computation because it has yet to read a treatment couch position, or a set of anatomical information, or a dose grid or point specification since the time the dose computation program was invoked.

Beam name not in database. This error occurs when the beam name supplied in the field description does not match any beam known to the dose computation program.

Wedge not found. This error occurs when the wedge ID specified in the field description does not match any of the known wedge ID's for the specified treatment machine.

Summary mode unsupported. This error occurs if the dose computation program does not support the summary results mode and it is either instructed to compute the dose from a source in summary results mode or is requested to return summary results.

Immediate mode unsupported. This error occurs if the dose computation program does not support the immediate results mode and it is instructed to compute the dose from a source in immediate results mode.

After computing the dose for a grid or list of points, the dose computation program writes the appropriate success code (1 or 2), optionally followed by several more types of information. The additional information is returned only when the program is in immediate results mode. In summary results mode, the resulting information is added to the summary results in the program's memory.

In immediate mode, the set of information written out for a successful dose computation can be summarized as follows:

- Success code
- Particle type

- List of preliminary results
- Main results

The particle type determines the number of elements on the list of preliminary results. The main results are a grid or list of dose values with some accompanying information derived from the grid or point list specification. The dose grid values are sent z-plane by z-plane from the back of the grid to the front. Within each z-plane, the values are sent y-row by y-row, from the left end of the row to the right. The (0,0,0) index of the grid is in the lower left back corner, and the dose grid values are returned in order of increasing index in each dimension. Though not explicitly part of the protocol, order of indices for the dose grid returned by the dose computation program shall be assumed to be (x, y, z) — so the x coordinate occupies the first index, and the z coordinate the last — for the sake of describing the ordering of the output below. The dose point list absorption values are sent to the output in the same sequence that the points' coordinates were supplied in the input.

If the program is in summary mode and is instructed to return its summary results, then it will return a single instance of:

- Success code
- Main results

for either a summary dose grid or a summary list of dose points, depending on which one was most recently specified. The format of the summary grid and summary points list (main results) are the same as the format of the individual grid and points list results, described above.

All output is to an ASCII stream. The same conventions regarding lengths, coordinate systems, and the composition of strings, integers, and real numbers that apply to the input stream apply here as well. Items in italics are provided for documentation purposes. Newlines are sent where specified below, indicated with <NL>. The output protocol is as follows:

- Output code <NL> (*an integer between 0 and 2 inclusive*)
- *Mode & output code specific information*
    - *When in immediate mode and the output code is 1 or 2*
        - Beam particle type <NL> (*the number 1 (photon), 2 (neutron), or 3 (electron)*)
        - *Beam particle type specific information*
            - *Prelim results for photon and neutron beams*
                - Source to skin distance <NL> *(real number)*
                - Equivalent square <NL> *(real number)*
                - Output factor <NL> *(real number)*
                - TPR at depth of isocenter <NL> *(real number)*

            - ⋆ *Prelim results for electron beams*
                - ◇ Source to skin distance <NL> *(real number)*
    - ◦ *When in either mode and the output code is 1*
        - ∗ Number of points in X direction of grid (*integer*)
        - ∗ Number of points in Y direction of grid (*integer*)
        - ∗ Number of points in Z direction of grid <NL> (*integer*)
        - ∗ *For each Z-plane of the grid, from lowest to highest Z index:*
            For each Y-row of the plane, from lowest to highest Y index:
            For each point of the row, from lowest to highest X index:
            · Dose grid value at location (x, y, z) (*real number*)
    - ◦ *When in either mode and the output code is 2*
        - ∗ Number of points <NL> (*integer*)
        - ∗ *For each point:*
            - ⋆ Dose value at that point <NL> (*real number*)

Any client that reads the dose computation program's output protocol should also have access to the input stream that the dose computation program used to generate the output. The output stream alone is not intended to provide sufficient information for a client to make sense of the resulting data. Furthermore, such a client of the dose computation program may also need access to any ancilliary databases used by the dose computation program itself, such as a database of dosimetric information.

The success codes (1 for dose grid, 2 for dose point list) are the same whether the dose computation program is returning the results from individual fields, or summary results. The format of the data is different, though, since a particle type and preliminary results are only returned in immediate results mode. The client is responsible for interpreting the data, based upon the results mode it requested.

More information about the source to skin distance, the equivalent square, the output factor, and the tissue phantom ratio (TPR) at depth of isocenter can be found in [1]. A negative number will be reported for the source to skin distance to indicate the case that the beam's central axis does not intersect the patient volume.

The units of the returned dose grid and dose point list values are rads per monitor unit when results are returned in immediate mode, and rads when results are returned in summary mode. If arc therapy for a beam was indicated (by specifying a non-zero arc size in the input protocol), then each dose grid or point list value represents the total dose absorbed at that point for the entire arc, and not the dose absorbed per arc degree.

When arc therapy in indicated in immediate results mode, the four preliminary results that are returned (SSD, equivalent square, output factor, and TPR) correspond to the starting point of the arc.

Newlines also occur after each new plane and row of the dose grid is written out. As with the input stream, there will be at least one newline in any given run of 128 characters, so that the output buffer of the dose computation program does not overflow. Thus, more newlines may occur than those explicitly mentioned above, where needed (such as in the dose grid output). Additional newlines may also appear at other arbitrary places in the output, to make the output stream more human readable.

3 Lines and seed sources

The protocol for line sources and seeds is not yet defined.

Appendix: Samples of dosecomp input and output

This is an example of an input file for the dose computation program. The input file consists of the following sequence: a dose point list specification, a set of anatomical information, a treatment couch position, a beam description for an electron beam with a cutout, a dose grid specification, and a beam description for a photon beam with a rectangular collimator, a wedge, and a block.

```
4                                          input category code
6                                          # dose points
0.0  0.0  0.0                              dose point coords.
0.0 -0.5  0.0
0.0 -1.0  0.0
0.0 -1.5  0.0
0.0 -2.0  0.0
0.0 -2.5  0.0

2                                          input category code
1                                          # organs
1.0                                        density of organ
3                                          # contours
-10.0                                      z pos. of contour 1
4                                          # points
 10.0   0.0                                coords. of points
-10.0   0.0
-10.0 -20.0
 10.0 -20.0
0.0                                        z pos. of contour 2
4                                                 ⋮
 10.0   0.0
-10.0   0.0
-10.0 -20.0
 10.0 -20.0
10.0
4
 10.0   0.0
-10.0   0.0
-10.0 -20.0
 10.0 -20.0

1                                          input category code
0.0 0.0 0.0                                couch position
```

```
5                                   input category code
"ELECTRON TESTER "                  beam name
0.0                                 gantry angle
0.0                                 arc size
0.0                                 couch lat
0.0                                 couch long
0.0                                 couch vert
0.0                                 couch turn
0.0                                 collimator angle
1.0                                 attenuation factor
1.0                                 monitor units
5                                   # pts in cutout contour
 4.5  0.0                           cutout contour pt coords
 2.0  2.0
-2.0  1.0
-1.0 -2.0
 0.0 -1.0

3                                   input category code
-5.0 -5.0 -5.0                      origin of dose cube
10.0 10.0 10.0                      sizes of dose cube
4 4 4                               # dose pts on x,y,z axes

5                                   input category code
"CLINAC20-15MV   "                  beam name
0.0                                 gantry angle
0.0                                 arc size
0.0                                 couch lat
0.0                                 couch long
0.0                                 couch vert
0.0                                 couch turn
0.0                                 collimator angle
1.0                                 attenuation factor
1.0                                 monitor units
1                                   wedge number
90.0                                wedge angle
6.0 5.0 6.0 5.0                     collimator settings
1                                   number of blocks
0.05                                block transmissivity
4                                   number of block pts
1.177  -5.88                        coords of block pts
5.977  -3.96
5.977  -6.12
1.257  -6.20
0                                   input category code
```

This is the corresponding output for the above input file, which includes the dose point list for the electron beam and the dose grid for the photon beam.

```
2                                                    output category code
3                                                    electron beam code
 100.00000                                           ssd
6                                                    # dose points

                                                     doses to points
   0.85341
   0.85558
   0.83186
   0.74994
   0.63360
   0.51781

1                                                    output category code
1                                                    photon beam code
         4          4          4                     # dose pts on x,y,z axes
 100.00000                                           ssd
  10.90909                                           side of equiv square
   1.00636                                           output factor
   0.00000                                           tpr at isocenter

                                                     doses
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
   0.79188   0.78343   0.75442   0.71156
   0.81478   0.81351   0.78232   0.72962
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
   0.79187   0.78341   0.75427   0.71115
   0.81477   0.81326   0.78215   0.72923
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
   0.00000   0.00000   0.00000   0.00000
0                                                    end of data code
```

References

[1] Ira J. Kalet, Jonathan Jacky, Paul Cho, and Mark Martin. Prism dose computation methods. Technical Report 93-09-01, Radiation Oncology Department, University of Washington, Seattle, Washington, 1993.

[2] Mark Martin. Prism beam data file description. Technical Report 93-08-01, Radiation Oncology Department, University of Washington, Seattle, Washington, 1993.

[3] Ira J. Kalet, Jonathan Unger, Jonathan Jacky, and Mark Phillips. Prism system capabilities and user interface specification. Technical Report 93-08-02, Radiation Oncology Department, University of Washington, Seattle, Washington, 1993.

[4] Ira J. Kalet and Jonathan Unger. Prism system implementation. Technical Report 93-09-02, Radiation Oncology Department, University of Washington, Seattle, Washington, 1993.