

# **PULSE-HEIGHT TALLY VARIANCE REDUCTION IN MCNP**

by

**Thomas E. Booth**

## **ABSTRACT**

This report describes the variance reduction methods implemented for the pulse-height tally (f8) in MCNP<sup>TM</sup> version 5. The method currently has been coded for photon-only problems without bremsstrahlung; the thick-target bremsstrahlung sampling is not microscopically correct. Note that coupled photon/electron pulse-height tally variance reduction is planned for the future.

---

## **I. INTRODUCTION**

Two different approaches for applying variance reduction with pulse height tallies (f8) in MCNP were developed in the early 1990's.<sup>1</sup> The “deconvolution approach” was selected for implementation in MCNP. Reference 1 describes only integer splitting, but the implementation in MCNP also includes:

- 1) noninteger splitting (IMP card)
- 2) implicit capture and weight cutoff (CUT card)
- 3) weight window (WWN card)
- 4) forced collisions (FCL card)
- 5) exponential transform (EXT card)
- 6) DXTRAN (DXT card)

---

MCNP is a trademark of the Regents of the University of California, Los Alamos National Laboratory.

This report assumes the reader has understood pages 1-42 of Ref. 1, and this report modifies the concepts therein so that they can be applied to items 1-6 above.

Instead of presenting the material in the most concise and general form, the report proceeds in a tutorial manner starting with the easiest concepts. The report first discusses the biasing of a single physical tree with no variance reduction branching (e.g., source biasing and exponential transform biasing). Second, the report discusses variance reduction splits for which the sum of the split branch weights is equal to the weight of the presplit particle (e.g.,  $n:1$  integer splitting, forced collision, and implicit capture). Third, the report discusses variance reduction splits for which the sum of the split branch weights is not equal to the weight of the presplit particle (e.g., noninteger splitting and DXTRAN).

Note that DXTRAN refers to the variance reduction method whereas dxtran and nondxtran are adjectives used to label particles, spheres, and branches associated with the DXTRAN method.

## II. BIASING A SINGLE PHYSICAL TREE—I

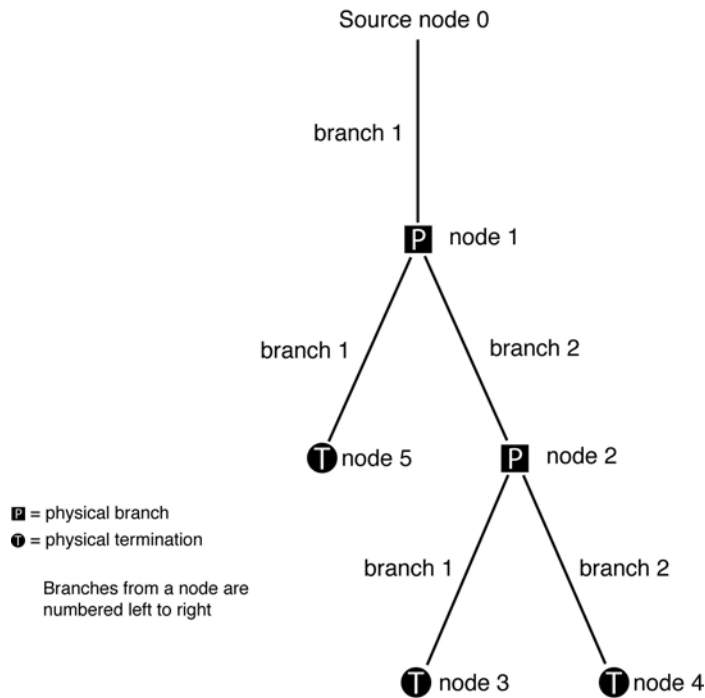
This section ignores variance reduction branching and considers the effects of biasing on a single physical tree. For example, if the “distance to collision” is sampled using the exponential transform in MCNP, then the probability of sampling any particular physical tree has been altered from the analog sampling and the tally score must be weighted by the ratio of the true probability that the tree occurs to the biased probability that the tree occurs. Consider the tree of Fig. 1. Note that there are three types of nodes, the source node, physical branch nodes, and physical termination nodes. In general there can be many surface crossings and many collisions between nodes. For instance, suppose that branch 2 under node 1 (abbreviated b12) had three distance-to-collision samplings on the branch. That is, between node 1 and node 2 there were three biased distance to collision samplings. Let  $p_i$  and  $q_i$  be the true pdf (probability density function) and the biased pdf for the  $i^{th}$  distance to collision sampling. (Note that in MCNP if the distance to collision is greater than the distance to boundary, then the particle is put on the boundary without collision. That is, one possible outcome of the “distance to collision” sampling is that no collision occurs.) Because of the biasing, branch b12 occurs  $(q_1q_2q_3)/(p_1p_2p_3)$  times as often as it does in an analog walk. For this reason, branch b12 is assigned the “branch weight multiplier”  $w = (p_1p_2p_3)/(q_1q_2q_3)$ . The entire physical tree has been made

$$\frac{1}{w_{01}w_{11}w_{12}w_{21}w_{22}} \quad (1)$$

times as likely to occur as in analog Monte Carlo. Thus, the collective weight of the tree is the reciprocal of the probability change in Eq. 1. That is, the collective weight of the tree is the product of the branch weight multipliers

$$w_c = w_{01}w_{11}w_{12}w_{21}w_{22} \cdot \quad (2)$$

Note that in the other (non f8) standard MCNP tallies, the tracks carry weights that multiply the tally when the track tallies. When a track's walk is made 5 times as likely to occur as an analog walk, the track will carry a weight of 1/5 so that the expected track tally will be preserved. The difference here is that for the f8 tally it is the tree that tallies so that the weight is assigned to the tree. If the entire tree has been made 5 times as likely to occur as the analog tree, then the tree will carry a weight of 1/5 so that the expected tree tally will be preserved. The tree has a “collective weight” associated with all physical branches of the tree.



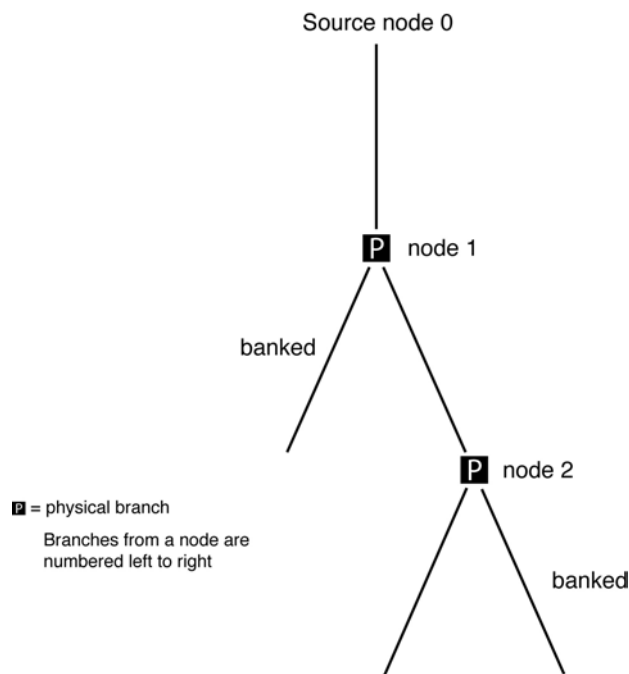
**Fig. 1. Typical Physical Tree.**

Note that both the track weights and the collective tree weight are being adjusted, in different ways, by the same weight factors. As the calculation proceeds, any track's weight is the product of all the branch weight multipliers in a direct path back to the source. For example, the track weight upon entering node 2 in Fig. 1 is  $w_{01}w_{12}$ .

### III. BIASING A SINGLE PHYSICAL TREE—II

In the previous section, branch weights were presented as a way to account for samplings from nonanalog pdfs. In standard variance reduction the track weights are modified for a variety of purposes, not just to account for samplings from nonanalog pdfs. The situation is similar for branch weight multipliers. Some examples follow.

Consider the physical tree under construction in Fig. 2. The current track being followed is b21, with b22 and b11 in the bank. Suppose that a roulette game is played on b21. That is, with probability  $s$  the track survives the roulette and with probability  $1 - s$  the track does not survive. If the track survives, the branch weight multiplier is multiplied by  $1/s$ . If the track does not survive, the branch weight multiplier is multiplied by 0. This terminates the track because the



**Fig. 2. Physical Tree Under Construction.**

track is the product of all branch weight multipliers traced back to the source. Note that this track may already have tallied to non-f8 tallies and these tallies are not affected by the track's termination. Furthermore, branches b22 and b11 are not affected for non-f8 tallies.

The situation is very different for the pulse height tally. First, the pulse height tally cannot be made until the physical tree is complete, so there have not been any tallies yet made for a physical tree under construction. Second, once b21 has been rouletted, the tree is no longer a physically generated tree. That is, nature does not play roulette; to be physically relevant the pulse height tally energy bin selection must include the energy deposited by branch b21 (and all branches emanating from branch b21). Because the tree is no longer physically relevant for pulse height tallies, the tree needs to have a tree weight of zero. Because the tree weight is the product of the branch weight multipliers (e.g., see Eq. 2), assigning a branch weight multiplier  $w_{21} = 0$  will make the tree weight  $w_c^{new} = 0$ . Note that although assigning a branch weight multiplier makes the tree weight zero, the track weights not associated with branch b21 are unaffected so that the particle tracks still contribute to the non f8 tallies. If only f8 tallies were desired, the tracking could be stopped at this point. (Currently the tracking is not stopped because I have presumed that users will have a mixture of f8 and non f8 tallies. If the future shows that there are a significant number of f8 tallies without non f8 tallies, some extra logic could be inserted to stop the tracking as soon as possible.)

If branch b21 survives (with probability  $s$ ) the roulette game, then the new branch weight multiplier becomes  $w_{12}^{new} = w_{12}/s$ . Because the tree weight is the product of the branch weight multipliers, the tree weight has also been multiplied by  $1/s$ ; that is,  $w_c^{new} = w_c/s$ . Note that the expected tree weight for the tree is preserved. That is, with probability  $(1 - s)$  the tree weight is  $0 \times w_c$  and with probability  $s$  the tree weight is  $(1/s)w_c$ . The expected tree weight is thus:

$$\langle w_c^{new} \rangle = (1 - s)(0 \times w_c) + s((1/s)w_c) = w_c \quad (3)$$

and thus the roulette game is fair.

The roulette game above is a special case of a more general game in which the branch weight multiplier is sampled between two possibilities. Let the current branch weight multiplier be  $w$  and let the two possible post-game branch weight multipliers be  $a$  and  $b$  with

$$a < w < b \quad . \quad (4)$$

Let the probabilities of picking  $a$  and  $b$  be

$$p_1 = \frac{b-w}{b-a} \quad (5)$$

$$p_2 = \frac{w-a}{b-a} \quad . \quad (6)$$

The expected branch weight multiplier is

$$\langle w^{new} \rangle = p_1 a + p_2 b = \frac{b-w}{b-a} a + \frac{w-a}{b-a} b = w \quad . \quad (7)$$

Because the tree weight is proportional to the branch weight multiplier, the expected tree weight will also be preserved. With probabilities  $p_1$  and  $p_2$  the tree weight will be

$$w_{c1} = w_c(a/w) \quad (8)$$

$$w_{c2} = w_c(b/w) \quad . \quad (9)$$

The expected tree weight is

$$\langle w_c^{new} \rangle = p_1 w_{1c} + p_2 w_{2c} = \frac{w_c}{w} \langle w^{new} \rangle = w_c \quad . \quad (10)$$

#### IV. VARIANCE REDUCTION SPLITS I

Reference 1 showed that simple integer splitting could be correctly treated by randomly sampling for one branch of each split. The arguments in this section are very similar to those in Ref. 1. It is the author's intention that this report mostly be self-contained, but far more details are given in Ref. 1. Thus, if the reader has difficulty with this section, it may be useful to read Ref. 1.

This section considers track splitting techniques that have the same total track weight before and after the split. For example, an  $n:1$  integer split of a track of weight  $w$  produces  $n$  split tracks of weight  $w/n$  so the total weight both before and after the split is  $w$ . Other

examples in MCNP are the forced collision split into collided and uncollided track weights and the implicit capture split into captured and surviving track weights. (Sections VI and VII will treat noninteger splitting and DXTRAN which do not preserve the total post-splitting track weight.)

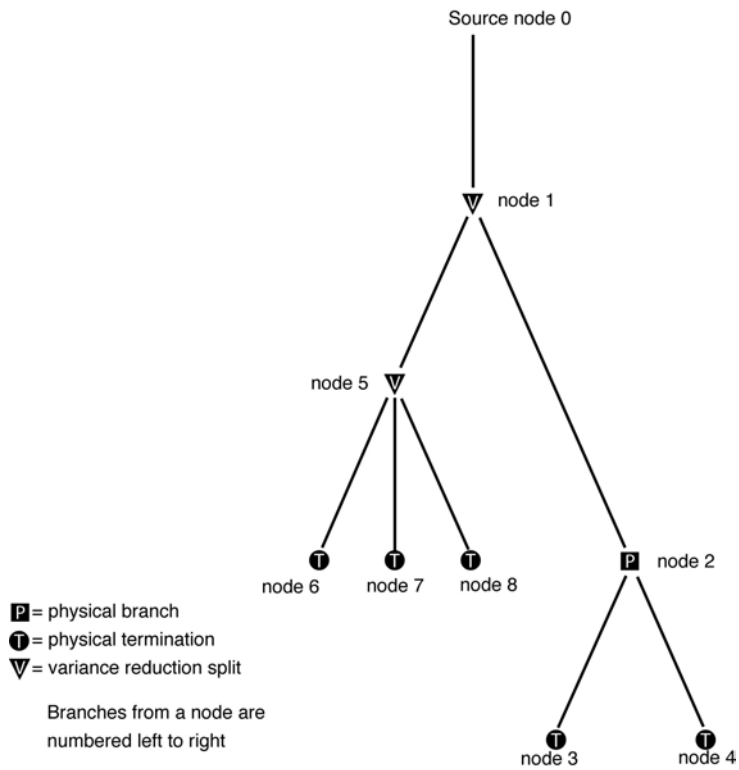
Consider Fig. 3. The key to deciphering what tally the tree of Fig. 3 should post is to note that nature chooses exactly one branch under each variance reduction node. In Fig. 3, there are four possible choices of trees that physically could have occurred. These four trees are shown in Figs. 3.1-3.4. Note that any of the variance reduction splits can be “undone” by reassigning the total weight to one of the split branches. For example, suppose branch b11 is the uncollided part of a forced collision and branch b12 is the collided part. If  $w_1$  is the track weight at node 1 and  $\lambda$  is the number of free paths to the cell boundary, then the branch weight multipliers assigned to the uncollided and collided branches are

$$w_{11} = \exp(-\lambda) = p_{11} \quad (11)$$

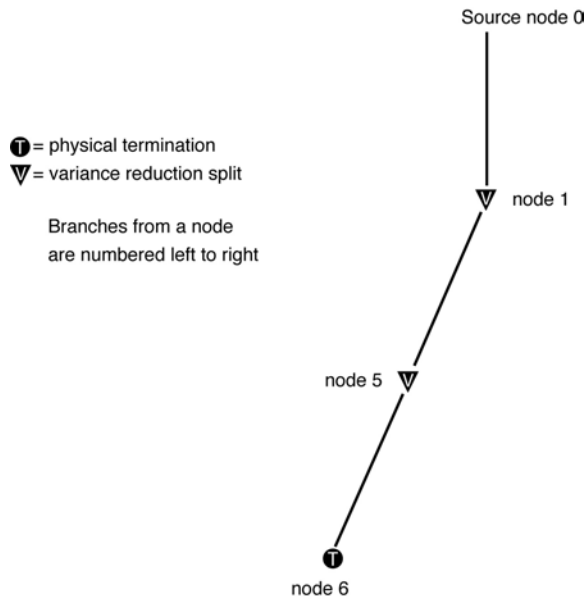
$$w_{12} = 1 - \exp(-\lambda) = p_{12} \quad (12)$$

where  $p_{ij}$  is the probability that branch  $j$  under variance reduction node  $i$  is sampled in an analog sampling. Now, sample one of the branches according to  $p_{1j}$ , and assign the total weight  $w_1$  to the sampled branch. Note that this is just a complicated way of doing analog sampling; that is, some time is wasted, but in the end weight  $w_1$  reaches the boundary with probability  $\exp(-\lambda)$  and weight  $w_1$  collides with probability  $1 - \exp(-\lambda)$ . Returning to a general tree, if all the variance reduction branches are sampled according to  $p_{ij}$ , then it is just the same as never having done the variance reduction splits. The physical tree of Fig. 3.1 is obtained by sampling the branches b11 and b51 with probabilities  $p_{11}$  and  $p_{51}$ . That is, Fig. 3.1 occurs with probability  $p_{11}p_{51}$ . Similarly, Figs. 3.2 and 3.3 occur with probabilities  $p_{11}p_{52}$  and  $p_{11}p_{53}$ . If branch b12 is sampled, then the split at node 5 never occurs, and Fig. 3.4 occurs with probability  $p_{12}$ . The sum of the probabilities for Figs. 3.1-3.4 is

$$p_{11}p_{51} + p_{11}p_{52} + p_{11}p_{53} + p_{12} = p_{11}(p_{51} + p_{52} + p_{53}) + p_{12} = p_{11} + p_{12} = 1 \quad (13)$$

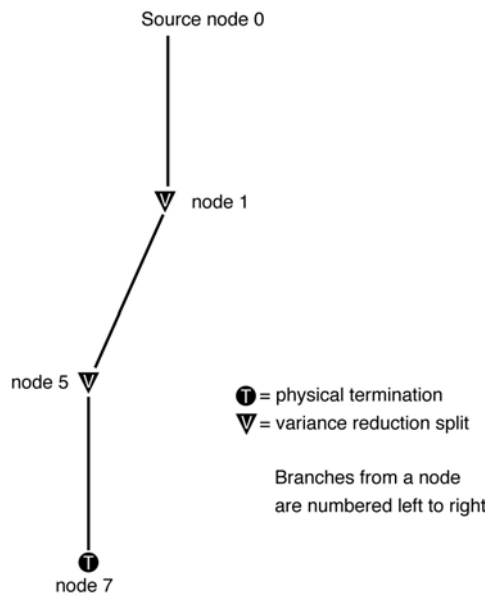


**Fig. 3. Tree with Variance Reduction Branches.**

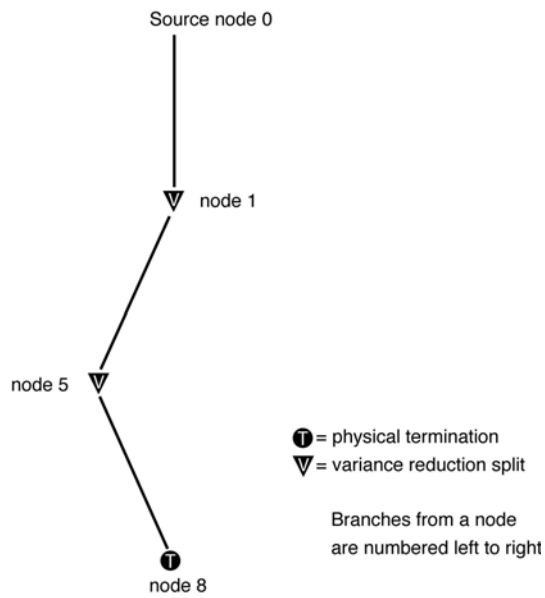


**Fig. 3.1. First Possible Physical Occurrence.**

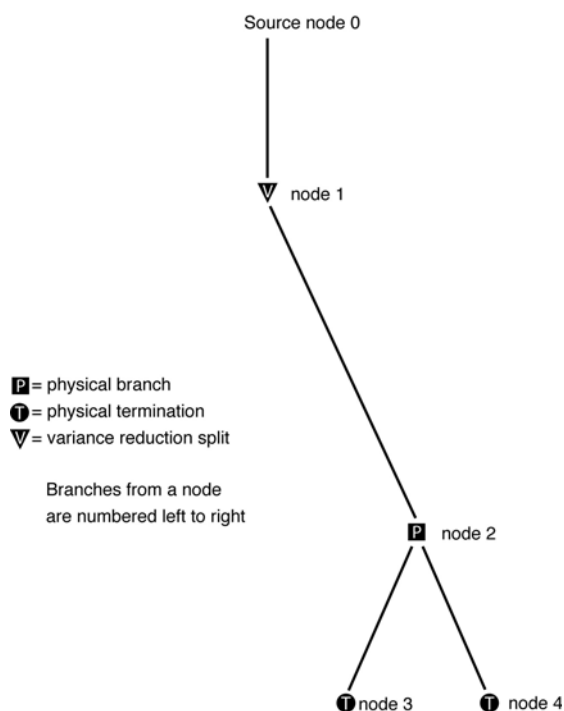




**Fig. 3.2. Second Possible Physical Occurrence.**



**Fig. 3.3. Third Possible Physical Occurrence.**



**Fig. 3.4. Fourth Possible Physical Occurrence.**

Let  $E_{ij}$  be the energy deposited on branch  $b_{ij}$  between the two nodes of the branch. If there were no physical biasings in the problem, then all the branch weight multipliers would be one and, as in Ref. 1, a “count” of unit weight is tallied in the bin determined by the sum of the deposited energies. That is, referring to Figs. 3.1-3.4, see Table I.

**TABLE I**  
**POSSIBLE PULSE HEIGHT ENERGIES DEPOSITED**

energy deposited	probability	tally (tree weight)
$E_{01} + E_{11} + E_{51}$	$p_{11}p_{51}$	1
$E_{01} + E_{11} + E_{52}$	$p_{11}p_{52}$	1
$E_{01} + E_{11} + E_{53}$	$p_{11}p_{53}$	1
$E_{01} + E_{12} + E_{21} + E_{22}$	$p_{12}$	1

Rather than sampling and tallying a unit hit to the appropriate energy bin, the expected tally can be used. That is, a tally is made for each of the trees of Figs. 3.1-3.4; see Table II.

**TABLE II**  
**EXPECTED PULSE HEIGHT ENERGIES DEPOSITED**

energy deposited	probability	tally (tree weight)
$E_{01} + E_{11} + E_{51}$	1	$p_{11}p_{51}$
$E_{01} + E_{11} + E_{52}$	1	$p_{11}p_{52}$
$E_{01} + E_{11} + E_{53}$	1	$p_{11}p_{53}$
$E_{01} + E_{12} + E_{21} + E_{22}$	1	$p_{12}$

## V. VARIANCE REDUCTION SPLITS AND BRANCH BIASING

In Sections II and III the branch weight multipliers were used on a single physical tree to determine the tree's tally. In Section IV the branch weight multipliers (initially) were used as probabilities to determine which tree occurred. The two branch weight multiplier uses coexist in the deconvolution patch. That is, the branch weight multipliers contain both types of biasing information.

Reconsider Fig. 3. Suppose that in addition to the variance reduction splits, there is a physical branch weight multiplier as in Sections II and III. In fact, let  $w_{ij}^s$  be the splitting branch weight multiplier (as in Section IV) and let  $w_{ij}^b$  be the physical branch weight multiplier (as in Sections II and III). If the variance reduction splits are “undone” as in Section IV, the trees of Figs. 3.1-3.4 still occur with the same probabilities as in Table I, but now they have physical branch weight multipliers  $w_{ij}^b$ . Applying the theory in Sections II and III to each of the four trees gives Table III.

Replacing the sampling in Table III with expected values in the same procedure as in Section IV yields Table IV.

**TABLE III**  
**POSSIBLE PULSE HEIGHT ENERGIES DEPOSITED WITH**  
**PHYSICAL BRANCH WEIGHT MULTIPLIERS**

energy deposited	probability	tally (tree weight)
$E_{01} + E_{11} + E_{51}$	$p_{11}p_{51}$	$w_{01}^b w_{11}^b w_{51}^b$
$E_{01} + E_{11} + E_{52}$	$p_{11}p_{52}$	$w_{01}^b w_{11}^b w_{52}^b$
$E_{01} + E_{11} + E_{53}$	$p_{11}p_{53}$	$w_{01}^b w_{11}^b w_{53}^b$
$E_{01} + E_{12} + E_{21} + E_{22}$	$p_{12}$	$w_{01}^b w_{12}^b w_{21}^b w_{22}^b$

**TABLE IV**  
**EXPECTED PULSE HEIGHT ENERGIES DEPOSITED WITH**  
**PHYSICAL BRANCH WEIGHT MULTIPLIERS**

energy deposited	probability	tally (tree weight)
$E_{01} + E_{11} + E_{51}$	1	$p_{11}p_{51} w_{01}^b w_{11}^b w_{51}^b$
$E_{01} + E_{11} + E_{52}$	1	$p_{11}p_{52} w_{01}^b w_{11}^b w_{52}^b$
$E_{01} + E_{11} + E_{53}$	1	$p_{11}p_{53} w_{01}^b w_{11}^b w_{53}^b$
$E_{01} + E_{12} + E_{21} + E_{22}$	1	$p_{12} w_{01}^b w_{12}^b w_{21}^b w_{22}^b$

Because the split probabilities are the same as the split branch weight multipliers, i.e.,

$$w_{ij}^s = p_{ij} \cdot \tag{14}$$

Table IV can be rewritten as Table V.

If one assigns a “split” branch weight multiplier of 1 to any branch that has not been split, then one can define a total branch weight multiplier as the product of the physical branch weight multiplier and the split branch weight multiplier. That is,

$$w_{ij} = w_{ij}^s w_{ij}^b . \quad (15)$$

In Fig. 3,

$$w_{01}^s = w_{21}^s = w_{22}^s = 1 \quad (16)$$

because these branches are not below a variance reduction split node. Using Eqs. 15 and 16, Table V becomes Table VI.

**TABLE V  
MODIFICATION OF TABLE IV**

<b>energy deposited</b>	<b>probability</b>	<b>tally (tree weight)</b>
$E_{01} + E_{11} + E_{51}$	1	$w_{11}^s w_{51}^s w_{01}^b w_{11}^b w_{51}^b$
$E_{01} + E_{11} + E_{52}$	1	$w_{11}^s w_{52}^s w_{01}^b w_{11}^b w_{52}^b$
$E_{01} + E_{11} + E_{53}$	1	$w_{11}^s w_{53}^s w_{01}^b w_{11}^b w_{53}^b$
$E_{01} + E_{12} + E_{21} + E_{22}$	1	$w_{12}^s w_{01}^b w_{12}^b w_{21}^b w_{22}^b$

**TABLE VI  
PULSE HEIGHT TALLIES FOR FIGURE 3**

<b>energy deposited</b>	<b>probability</b>	<b>tally (tree weight)</b>
$E_{01} + E_{11} + E_{51}$	1	$w_{01} w_{11} w_{51}$
$E_{01} + E_{11} + E_{52}$	1	$w_{01} w_{11} w_{52}$
$E_{01} + E_{11} + E_{53}$	1	$w_{01} w_{11} w_{53}$
$E_{01} + E_{12} + E_{21} + E_{22}$	1	$w_{01} w_{12} w_{21} w_{22}$

Table VI is the basis for the deconvolution approach in MCNP. The variance reduction nodes separate the random walk into physical trees. An individual tree tally is the product of its (total) branch weight multipliers and the tally is posted in the energy bin containing the sum of all energies deposited by the tree.

Note that there are many possible ways to view the deconvolution process. For instance, instead of sampling the variance reduction trees of Figs. 3.1-3.4 according to the actual split probabilities  $p_{ij}$ , one could have used arbitrary probabilities  $q_{ij}$  and adjusted the tree weight by the ratio  $p_{ij}/q_{ij}$  to account for this. When the expected value of each of the four choices in Table VII is used, Table IV results as before with the unbiased  $p_{ij}$  sampling.

Note that the coding in MCNP does not save the  $p_{ij}$ , the  $w_{ij}^s$  nor the  $w_{ij}^b$ . Only the product  $w_{ij} = w_{ij}^s w_{ij}^b$  is saved as per Eq. 15. Thus, at the time the tree is being deconvoluted, one actually cannot sample the variance reduction splits with the  $p_{ij}$ . This sampling is irrelevant because MCNP uses the expected value. It is worth pointing out because there will be many cases for which the sum of the branch weight multipliers under a variance reduction node will not be unity. If the reader has not understood the argument in this paragraph, the next paragraph is essentially a redundant argument with different wording.

**TABLE VII**  
**VARIANT OF TABLE III**

energy deposited	probability	tally (tree weight)
$E_{01} + E_{11} + E_{51}$	$q_{11}q_{51}$	$w_{01}^b w_{11}^b w_{51}^b (p_{11}p_{51}) / (q_{11}q_{51})$
$E_{01} + E_{11} + E_{52}$	$q_{11}q_{52}$	$w_{01}^b w_{11}^b w_{52}^b (p_{11}p_{52}) / (q_{11}q_{52})$
$E_{01} + E_{11} + E_{53}$	$q_{11}q_{53}$	$w_{01}^b w_{11}^b w_{53}^b (p_{11}p_{53}) / (q_{11}q_{53})$
$E_{01} + E_{12} + E_{21} + E_{22}$	$q_{12}$	$w_{01}^b w_{12}^b w_{21}^b w_{22}^b (p_{12}/q_{12})$

Because the  $p_{ij}$  are unknown, reconsider the previous sampling of the arbitrary density  $q_{ij}$ . The details are given in Table VII. Note from Eqs. 14 and 15 that

$$w_{ij}^b p_{ij} = w_{ij} . \quad (17)$$

Using Eqs. 15–17 in Table VII yields Table VIII. Taking the expected value of Table VIII results in Table VI.

## VI. NONINTEGER SPLITTING

Noninteger splitting may be thought of as the weight modification game of Eqs. 5–10 followed by integer splitting. The first step is to adjust the branch weight multiplier before the split. Consider a noninteger  $v:1$  split with  $n < v < n + 1$ . Let the  $a$  and  $b$  of Eq. 4 be

$$a = w \frac{n}{v} \quad (18)$$

$$b = w \frac{n+1}{v} . \quad (19)$$

**TABLE VIII  
DERIVED FROM TABLE VII**

<b>energy deposited</b>	<b>probability</b>	<b>tally (tree weight)</b>
$E_{01} + E_{11} + E_{51}$	$q_{11}q_{51}$	$w_{01}w_{11}w_{51}/(q_{11}q_{51})$
$E_{01} + E_{11} + E_{52}$	$q_{11}q_{52}$	$w_{01}w_{11}w_{52}/(q_{11}q_{52})$
$E_{01} + E_{11} + E_{53}$	$q_{11}q_{53}$	$w_{01}w_{11}w_{53}/(q_{11}q_{53})$
$E_{01} + E_{12} + E_{21} + E_{22}$	$q_{12}$	$w_{01}w_{12}w_{21}w_{22}/q_{12}$

Then the probabilities of Eqs. 7 and 8 are

$$p_1 = \frac{w \frac{n+1}{v} - w}{w \frac{n+1}{v} - w \frac{n}{v}} = n + 1 - v \quad (20)$$

$$p_2 = \frac{w - w \frac{n}{v}}{w \frac{n+1}{v} - w \frac{n}{v}} = v - n \quad (21)$$

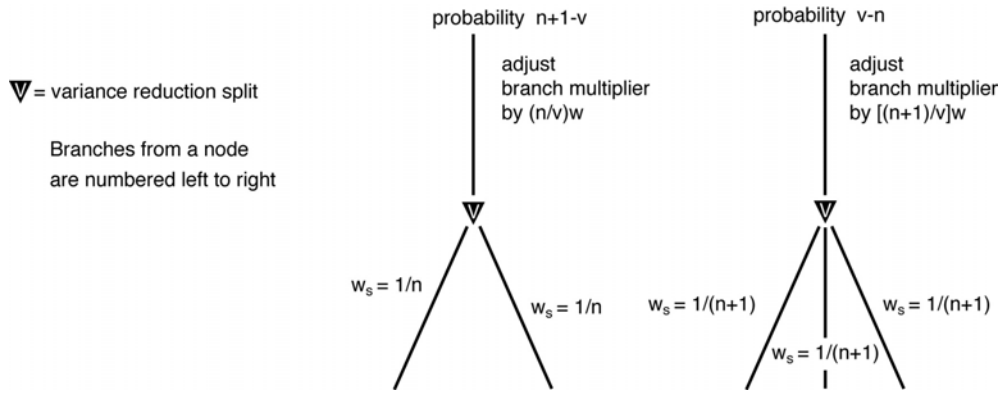
The second step is to do the splitting conditional on the outcome of the branch weight multiplier sampling. If the physical branch weight multiplier sampled in the game above is  $n/v$  then split  $n:1$  and assign each of the branches a split branch weight multiplier of  $1/n$ . If the physical branch weight multiplier sampled in the game above is  $(n+1)/v$  then split  $n+1:1$  and assign each of the branches a split branch weight multiplier of  $1/(n+1)$ .

The third step is a matter of convenience to make the branch weight modification similar to the usual track weight modification for noninteger splitting. Figure 4 shows the branch weight multiplications as discussed above. That is, there is a branch weight multiplication for the weight game played above the node and there are branch weight multiplications below the node for the splitting game. Because the tree weight is the only weight multiplying the f8 tally, the branch weight factor above the node can be eliminated (set to 1.0) and incorporated into all the branches below the node as in Fig. 5. That is, in Fig. 5 the products of weight branch multipliers are the same as in Fig. 4. The interpretation of Fig. 5 is used in MCNP because the noninteger splitting track weight multiplication is done is a single multiplication. This makes the f8 splitting branch weight multiplication the same as the splitting track weight multiplication.

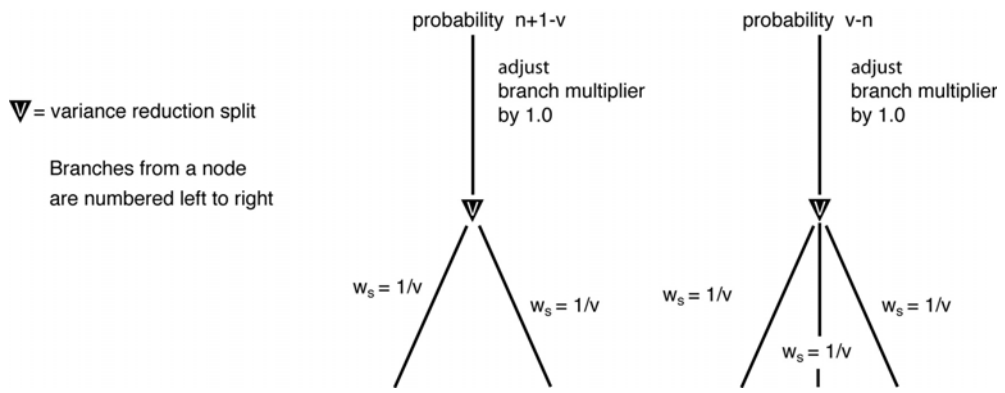
## VII. DXTRAN

DXTRAN can be treated as a special kind of variance reduction split in which the sum of the weights of the split particles does not equal the pre-DXTRAN weight. In some aspects, this is similar to the noninteger splitting of the previous section; there are some additional aspects, however.





**Fig. 4. Possible  $v:1$ : Noninteger Splitting Branch Weight Modification.**  
 $n < v < n + 1$



**Fig. 5. Equivalent  $v:1$  Noninteger Splitting Branch Weight Modification.**  
 $n < v < n + 1$

Rather than explain DXTRAN in one section, it will be useful to treat the cases in order of increasing complexity. (The DXTRAN explanations in Refs. 2 and 3 may be useful to review if the explanation herein needs further background for the reader.)

### VII.A. DXTRAN with a Single Particle Exiting

Let  $\Omega_s$  be the set of all scattering directions pointed at the dxtran sphere. Start by thinking of DXTRAN as a variance reduction split into the fraction of weight that reaches the dxtran sphere without collision,

$$D = \int_{\Omega_s} e^{-\lambda(\Omega)} p(\Omega) d\Omega \quad (22)$$

and the fraction of weight that does not (i.e., the particle either collides or escapes the system before reaching the dxtran sphere)

$$C = 1 - D \quad (23)$$

Figure 6 shows a picture of the DXTRAN variance reduction split at this stage.

For now, concentrate on the dxtran branch. The branch weight fraction reaching the sphere is  $D$  and the conditional probability (given the particle reaches the sphere) of reaching the sphere at  $\Omega$  is

$$p(\Omega|sphere) = \frac{e^{-\lambda(\Omega)} p(\Omega)}{\int_{\Omega_s} e^{-\lambda(\Omega)} p(\Omega) d\Omega} = \frac{e^{-\lambda(\Omega)} p(\Omega)}{D} \quad (24)$$

The integral in Eq. 24 (i.e.,  $D$ ) is expensive to evaluate and so MCNP samples from a biased pdf,  $b(\Omega)$ . The dxtran branch weight multiplier is adjusted for this biased sampling as per Section II by multiplying by

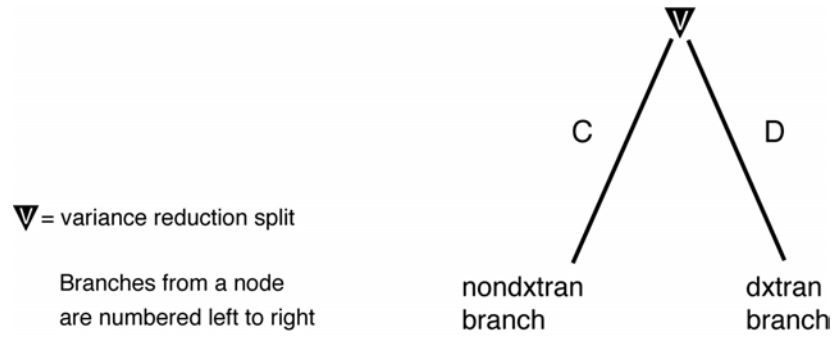
$$\frac{p(\Omega|sphere)}{b(\Omega)} \quad (25)$$

That is, the branch weight is no longer  $D$  but instead is

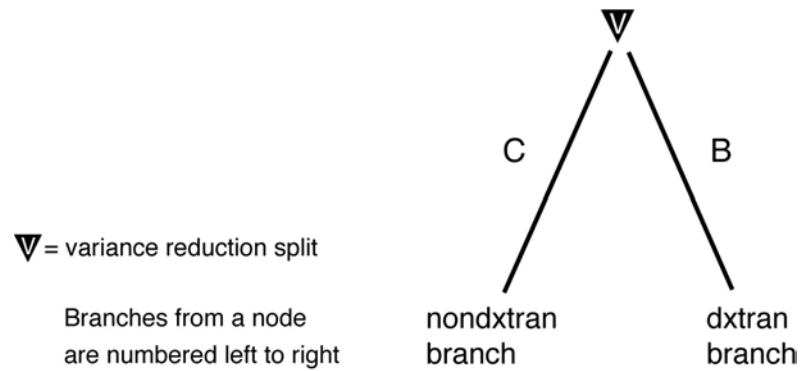
$$B = \frac{p(\Omega|sphere)}{b(\Omega)} D = e^{-\lambda(\Omega)} \frac{p(\Omega)}{b(\Omega)} \quad (26)$$

With this adjustment for biased pdf sampling, Fig. 6 changes to Fig. 6.1. (Note that the ratio  $p(\Omega)/b(\Omega)$  is calculated in MCNP subroutine CALCPS.) This completes discussion of the dxtran branch.

Now consider the nondxtran branch. Let  $P$  be the next event the track undergoes. That is,  $P$  is either the next collision, the next escape, or the next cross of the dxtran sphere. The



**Fig. 6. First Stage: DXTRAN Split with Branch Weight Multipliers.**



**Fig. 6.1. Second Stage: DXTRAN Split with Branch Weight Multipliers.**

nondxtran track should be sampled from the conditional probability (conditional on not reaching the dxtran sphere as the next event)

$$\begin{aligned}
 p(P|not\ sphere) &= \frac{p(P)}{C} & P \notin sphere \\
 &= 0 & P \in sphere
 \end{aligned}
 \tag{27}$$

Note that the weight fraction  $C$  involves evaluating  $D$ , i.e., the integral in Eq. 22. As with the dxtran track, this evaluation is avoided by sampling from a biased pdf. Instead of the correct pdf of Eq. 27, sample from

$$p(P) \tag{28}$$

and multiply the branch weight by

$$\begin{aligned} \frac{p(P|not\ sphere)}{p(P)} &= \frac{1}{C} & P \notin sphere \\ &= 0 & P \in sphere \end{aligned} \quad (29)$$

Multiplying the current branch weight multiplier  $C$  by the above factor, the new branch weight multiplier is

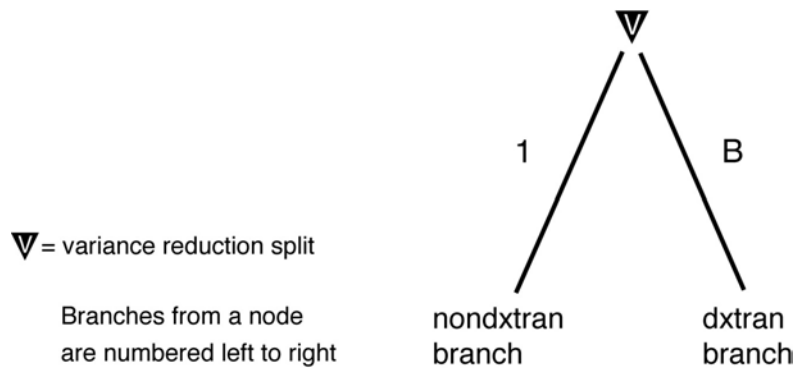
$$\begin{aligned} R &= 1 & P \notin sphere \\ R &= 0 & P \in sphere \end{aligned} \quad (30)$$

With this multiplication Fig. 6.1 becomes Fig. 6.2.a if the dxtran sphere is not crossed as the next event and Fig. 6.2.b if the dxtran sphere is crossed as the next event. Generically, using the random variable  $R$  in Eq. 30, Figs. 6.2.a and 6.2.b may be collapsed into Fig. 6.2.c.

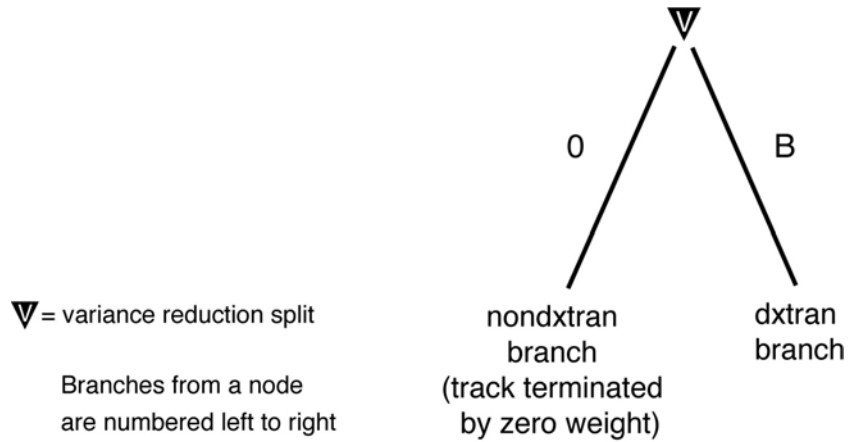
### VII.B. DXTRAN with Double Fluorescence

For DXTRAN and the double fluorescent event there are four possibilities:

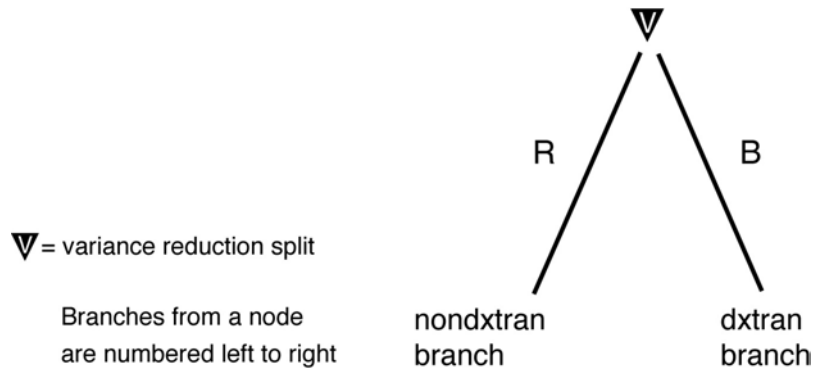
- 1) neither photon crosses the dxtran sphere as its next event
- 2) the first fluorescent photon crosses and the second does not
- 3) the second fluorescent photon crosses and the first does not
- 4) both the first and the second fluorescent photon cross



**Fig. 6.2.a. Third Stage: DXTRAN Split without Sphere Crossing.**



**Fig. 6.2.b. Third Stage: DXTRAN Split with Sphere Crossing.**



**Fig. 6.2.c. Random Branch Weight R from Eq. 30.**

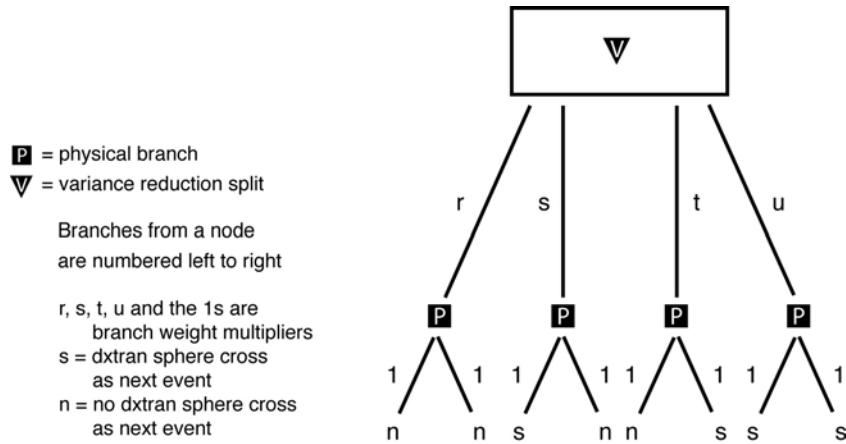
Figure 7 shows the diagram analogous to Fig. 6 for the single particle exiting. Note in Fig. 7 that  $r, s, t,$  and  $u$  are the probabilities in items 1 to 4 above. At this point  $r + s + t + u = 1$  just as  $C + D = 1$  in Fig. 6.

Now note that the double fluorescence photons are sampled independently so that their joint pdf is the product of the individual pdf's:

$$p_{12}(P_1, P_1) = p_1(P_1)p_2(P_2) . \quad (31)$$

Before evaluating  $r, s, t,$  and  $u,$  a few definitions are required.

If the first photon's scattering is tagged with a 1 and the second particle with a 2, then the development can proceed in a similar fashion to Section VII.A.



**Fig. 7. DXTRAN with Four Possibilities for Double Fluorescence.**

$$D_i = \int_{\Omega_s} e^{-\lambda_i(\Omega)} p_i(\Omega) d\Omega \quad (32)$$

$$C_i = 1 - D_i \quad (33)$$

$$p_i(\Omega|sphere) = \frac{e^{-\lambda_i(\Omega)} p_i(\Omega)}{\int_{\Omega_s} e^{-\lambda_i(\Omega)} p_i(\Omega) d\Omega} = \frac{e^{-\lambda_i(\Omega)} p_i(\Omega)}{D_i} \quad (34)$$

$$B_i = \frac{p_i(\Omega|sphere)}{b_i(\Omega)} D_i = e^{-\lambda_i(\Omega)} \frac{p_i(\Omega)}{b_i(\Omega)} \quad (35)$$

Using Eqs. 31–33 the branch weight multipliers are

$$r = C_1 C_2 \quad (36)$$

$$s = D_1 C_2 \quad (37)$$

$$t = C_1 D_2 \quad (38)$$

$$u = D_1 D_2 \quad (39)$$

Because the tree weight is the product of all branch weight multipliers on the same variance reduction branch, Fig. 7.1 will have the same tree weights as Fig. 7.2.

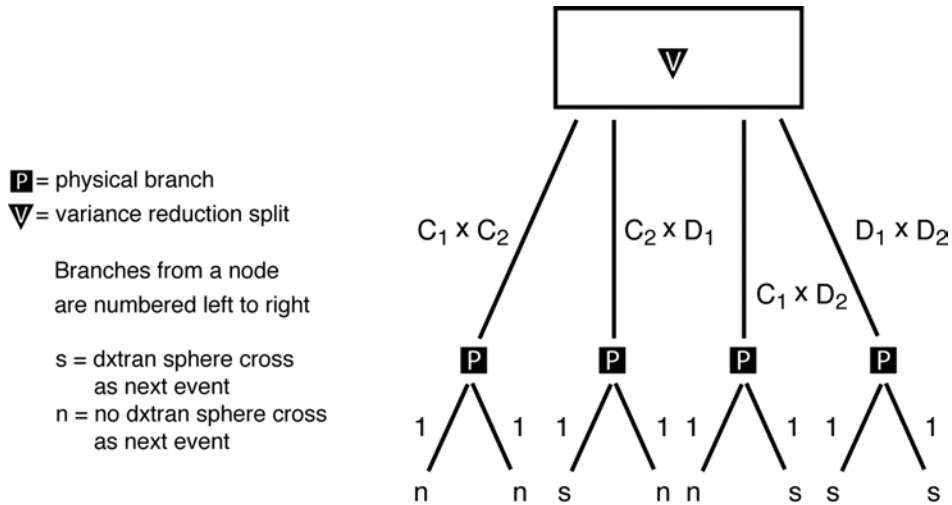
Apply the games played on the  $D$  and  $C$  branches in Section VII.A to the  $D_1$  and  $D_2$  branches and the  $C_1$  and  $C_2$  branches in Fig. 7.2. In analogy with Eq. 30, define

$$\begin{aligned} R_i &= 1 & P_i &\notin \text{sphere} \\ R_i &= 0 & P_i &\in \text{sphere} \end{aligned} \quad (40)$$

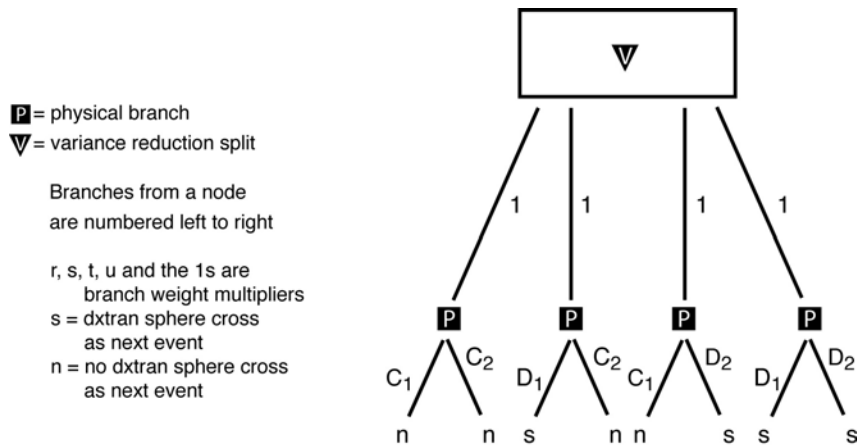
These games applied to Fig. 7.2 result in Fig. 7.3. Note that in Fig. 7.2 the “n” at the lowermost branches indicated that the track could not be sampled crossing the dxtran sphere. In Fig. 7.3 the notation “non” is used instead because the new sampling allows the nondxtran track to cross the dxtran sphere, provided the branch weight is zero when this happens. Furthermore, note that the  $R_i$  and  $B_i$  are random values depending in the first case on whether the dxtran sphere is crossed and in the second case on the  $\Omega$  sampled from the arbitrary density  $b_i(\Omega)$  in Eq. 35. If the four branches under the V node of Fig. 7.3 are sampled independently, then the outcomes of sampling the same pdf may be different: that is, the prime and unprimed values.

Noting that exactly one of the the variance reduction branches occurs for a physical tree, it is irrelevant whether a primed value or an unprimed value is used because both have exactly the same distribution. The tree of Fig. 7.3 is then replaced by the tree of Fig. 7.4. The branch numbers below the diagram in Fig. 7.4 refer to branches having the same photon (i.e., 1 or 2) and the same character (i.e., dxtran or nondxtran). (The branches start at the P nodes.) For example, branch 3 is the dxtran particle from the first fluorescent photon. All photons with the same branch number can be correlated to have identical (subsequent to the node V in Fig. 7.4) random walks. (In MCNP parlance, the branches can be made to “track.”) Another way of understanding this concept is to note that if

- 1)  $F_i(E)$  = the distribution of energy deposited by branch  $i$ , and
- 2)  $F_{ij}(E)$  = the distribution of total energy deposited by branches  $i$  and  $j$  together



**Fig. 7.1. Using the Independence of the Double Fluorescent Photons.**



**Fig. 7.2. Alternative Branch Weighting for Double Fluorescent Photons.**



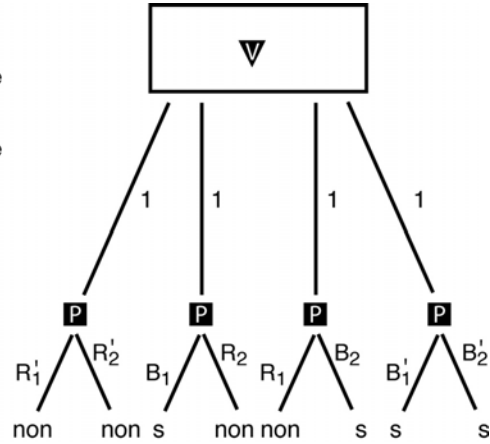
$R_i = 0$  if  $i$ 's next event is a cross of the dxtran sphere  
 $R_i = 1$  if  $i$ 's next event is not a cross of the dxtran sphere

$R_i = 0$  if  $i$ 's next event is a cross of the dxtran sphere  
 $R_i = 1$  if  $i$ 's next event is not a cross of the dxtran sphere

**P** = physical branch  
**V** = variance reduction split

Branches from a node  
 are numbered left to right

s = dxtran sphere cross as next event  
 n = no dxtran sphere cross as next event



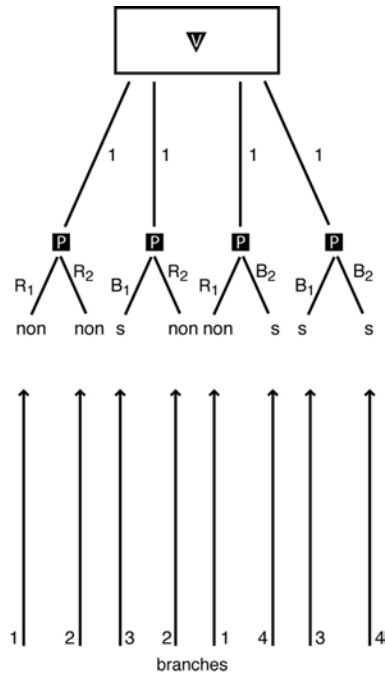
**Fig. 7.3. Avoiding the Evaluation of the Integral in Eq. 31 by Appropriate Sampling.**

$R_i = 0$  if  $i$ 's next event is a cross of the dxtran sphere  
 $R_i = 1$  if  $i$ 's next event is not a cross of the dxtran sphere

**P** = physical branch  
**V** = variance reduction split

Branches from a node  
 are numbered left to right

s = dxtran sphere cross as next event  
 n = no dxtran sphere cross as next event



**Fig. 7.4. Correlating the Walks on Different Variance Reduction Branches.**

then (using the Dirac  $\delta$  function to require  $E_i + E_j = E$ )

$$F_{ij}(E) = \int \int F_i(E_i)F_j(E_j)\delta(E - E_i - E_j) dE_i dE_j . \quad (41)$$

Note that  $F_{ij}(E)$  for any pair under a P node in Fig. 7.3 is exactly the same as for the corresponding pair in Fig. 7.4. Thus, Fig. 7.4 will have the same mean f8 tallies (though the higher moments will be different) as Fig. 7.3.

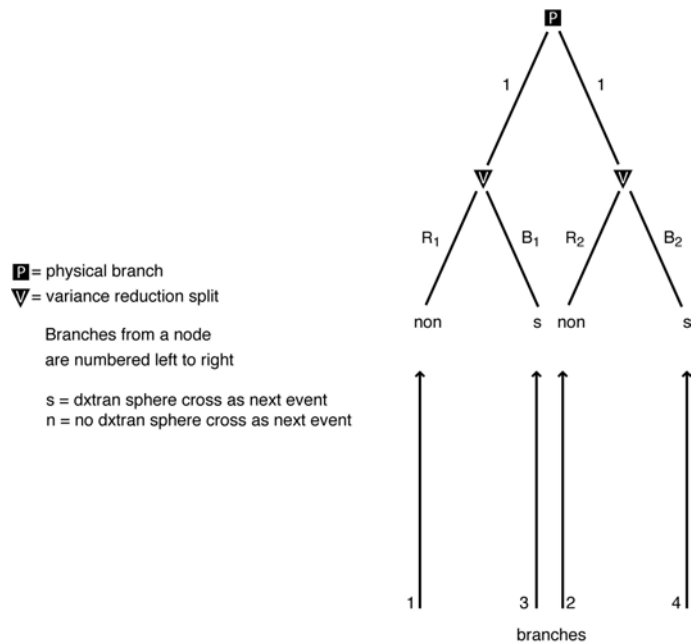
The final trick for DXTRAN with double fluorescence is to note that the tree of Fig. 7.5 is equivalent to the tree of Fig. 7.4 in the deconvolution process. (See Fig. 7.6 for the four physical trees deconvoluted from Fig. 7.5.) That is, if  $E_i$  is the energy deposited on the  $i^{th}$  branch, then Table IX shows the possible pulse height energies deposited under the top nodes in Figs. 7.4 and 7.5.

Two final notes are worthwhile.

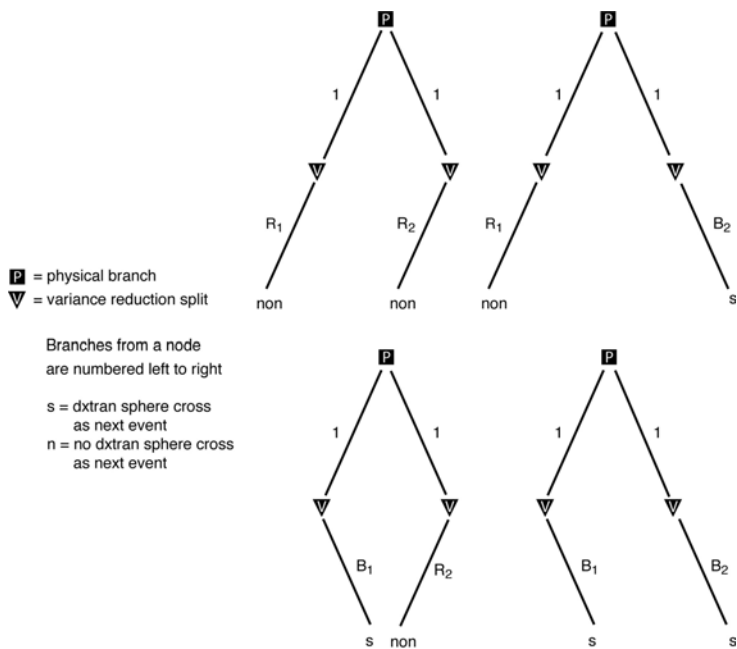
- 1) The production of the dxtran particles from double fluorescence photons does not track MCNP4C3 because MCNP4C3's DXTRAN treatment was not microscopically correct. The possibility that both double fluorescence photons reached the dxtran sphere was precluded in MCNP4C3. For this reason, problems with double fluorescence and DXTRAN will not track MCNP4C3.
- 2) The development herein requires that the double fluorescence photons are sampled independently, as is currently the case in MCNP. If new physical data and/or models introduce a correlation between the double fluorescence photons, the double fluorescence deconvolution treatment will need to be revised.

### **VII.C. DXTRAN with Pair Production**

Unlike the uncorrelated double fluorescence photons in Section VII.B, the annihilation photons resulting from an electron-positron pair production event are correlated because they always have exactly opposite directions  $\Omega$  and  $-\Omega$ . Because DXTRAN is applied only outside the dxtran sphere, it is impossible for both annihilation photons to arrive at the dxtran sphere without collision. The analysis for the annihilations photons thus parallels closely the development in Section VII.A for the single particle exiting a collision.



**Fig. 7.5. Equivalent Branch Scheme for Double Fluorescent Photons.**



**Fig. 7.6. The Four Trees from the Deconvolution of Fig. 7.5.**

**TABLE IX**  
**POSSIBLE PULSE HEIGHT ENERGIES DEPOSITED**  
**FOR FIGS. 7.4 AND 7.5**

energy deposited	probability	(tree weight)
$E_1 + E_2$	1	$R_1 R_2$
$E_1 + E_4$	1	$R_1 B_2$
$E_3 + E_2$	1	$B_1 R_2$
$E_3 + E_4$	1	$B_1 B_2$

Let  $p(\Omega)$  be the density of annihilation photons emitted at  $\Omega$ . Note that because there are two annihilation photons

$$\int p(\Omega) d\Omega = 2 \quad . \quad (42)$$

The probability that one of the annihilation photons reaches the dxtran sphere without collision is

$$D = \int_{\Omega_s} e^{-\lambda(\Omega)} p(\Omega) d\Omega \quad . \quad (43)$$

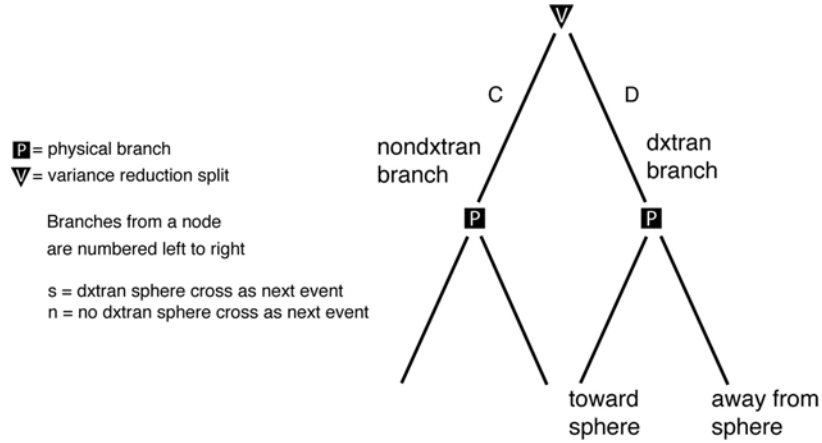
The probability that neither photon reaches the dxtran sphere without collision is

$$C = 1 - D \quad . \quad (44)$$

Figure 8 shows a picture of the dxtran variance reduction split at this stage.

For now, concentrate on the dxtran branch. The probability of reaching the sphere is  $D$  and the conditional probability (given the particle reaches the sphere) of reaching the sphere at  $\Omega$  is

$$p(\Omega|sphere) = \frac{e^{-\lambda(\Omega)} p(\Omega)}{\int_{\Omega_s} e^{-\lambda(\Omega)} p(\Omega) d\Omega} = \frac{e^{-\lambda(\Omega)} p(\Omega)}{D} \quad . \quad (45)$$



**Fig. 8. First Stage: DXTRAN Split with Branch Weight Multipliers.**

The integral in Eq. 43 (i.e.,  $D$ ) is expensive to evaluate and so MCNP samples from a biased pdf,  $b(\Omega)$ . The dxtran branch weight multiplier is adjusted for this biased sampling as per Section II by multiplying by

$$\frac{p(\Omega|sphere)}{b(\Omega)} . \quad (46)$$

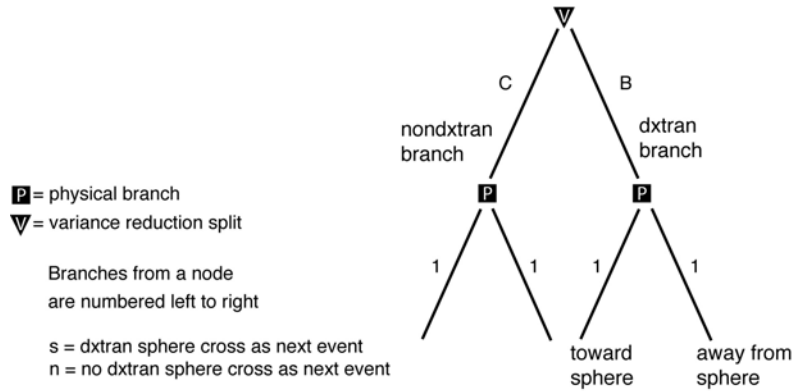
That is, the branch weight is no longer  $D$  but instead is

$$B = \frac{p(\Omega|sphere)}{b(\Omega)} D = e^{-\lambda(\Omega)} \frac{p(\Omega)}{b(\Omega)} . \quad (47)$$

With this adjustment for biased pdf sampling, Fig. 8 changes to Fig. 8.1. This completes discussion of the dxtran branch.

Now consider the nondxtran branch. Let  $P_i$  be the next event track  $i$  undergoes. That is  $P_i$  is either the next collision, the next escape, or the next cross of the dxtran sphere. The nondxtran annihilation pair should be sampled from the conditional probability (conditional on neither of the pair reaching the dxtran sphere as the next event)

$$\begin{aligned} p(P_1, P_2|neither\ sphere) &= \frac{p(P_1, P_2)}{C} & P_i \notin sphere\ for\ i = 1\ and\ 2 \\ &= 0 & P_i \in sphere\ for\ i = 1\ or\ 2 \end{aligned} \quad (48)$$



**Fig. 8.1. Second Stage: DXTRAN Split with Branch Weight Multipliers.**

Note that the weight fraction  $C$  involves evaluating  $D$ , i.e., the integral in Eq. 43. As with the dxtran pair, this evaluation is avoided by sampling from a biased pdf. Instead of the correct pdf of Eq. 48, sample from

$$p(P_1, P_2) \quad (49)$$

and multiply the branch weight by

$$\frac{p(P_1, P_2 | \text{neither sphere})}{p(P_1, P_2)} = \frac{1}{C} \quad P_i \notin \text{sphere for } i = 1 \text{ and } 2$$

$$= 0 \quad P_i \in \text{sphere for } i = 1 \text{ and } 2 \quad (50)$$

Multiplying the current branch weight multiplier  $C$  by the above factor, the new branch weight multiplier is

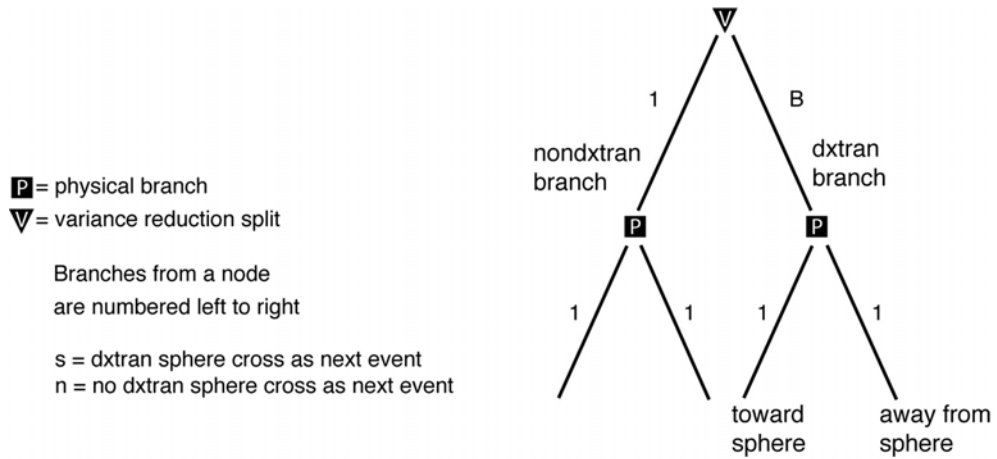
$$R = 1 \quad P_i \notin \text{sphere for } i = 1 \text{ and } 2$$

$$R = 0 \quad P_i \in \text{sphere for } i = 1 \text{ and } 2 \quad (51)$$

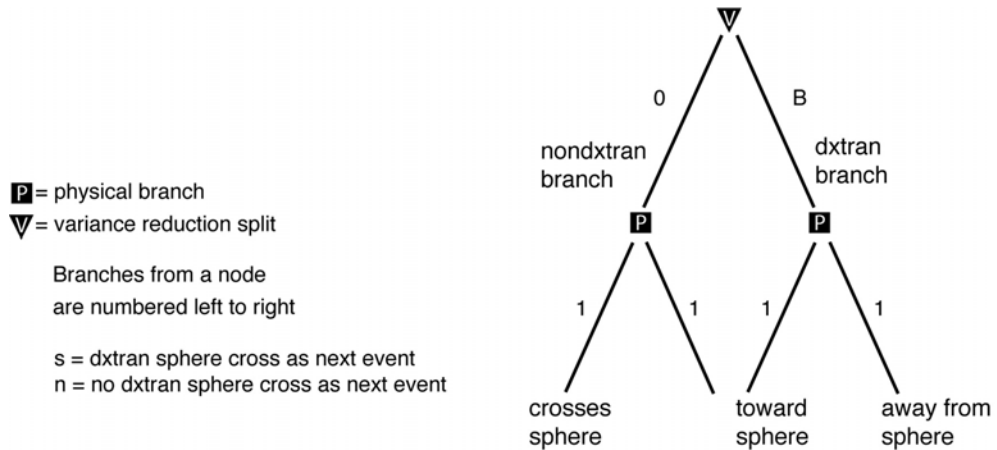
With this multiplication, Fig. 8.1 becomes Fig. 8.2.a if the dxtran sphere is not crossed as the next event and Fig. 8.2.b if the dxtran sphere is crossed as the next event. Generically, using the

random variable  $R$  in Eq. 51, Figs. 8.2.a and 8.2.b may be collapsed into Fig. 8.2.c. Because the tree weight is the product of all its branch weights, the tree of Fig. 8.2.d with

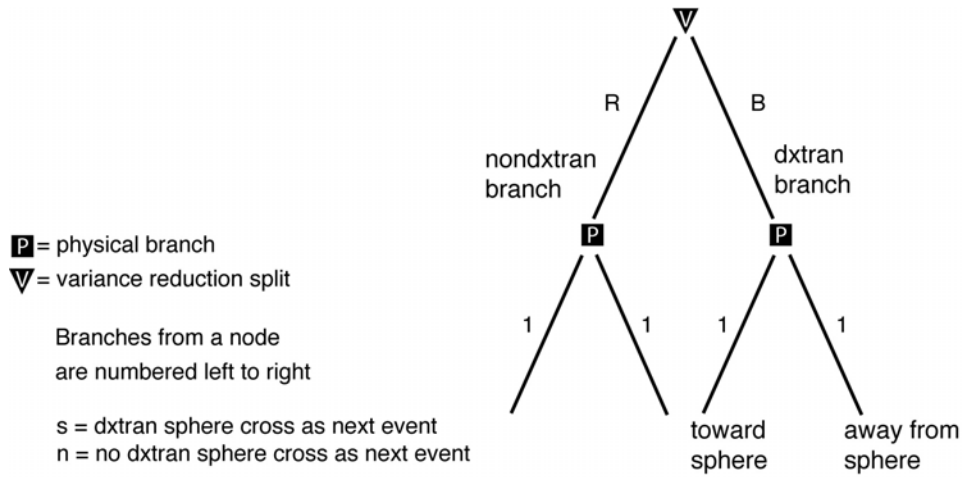
$$\begin{aligned} R_i &= 1 & P_i &\notin \text{ sphere} \\ R_i &= 0 & P_i &\in \text{ sphere} \end{aligned} \tag{52}$$



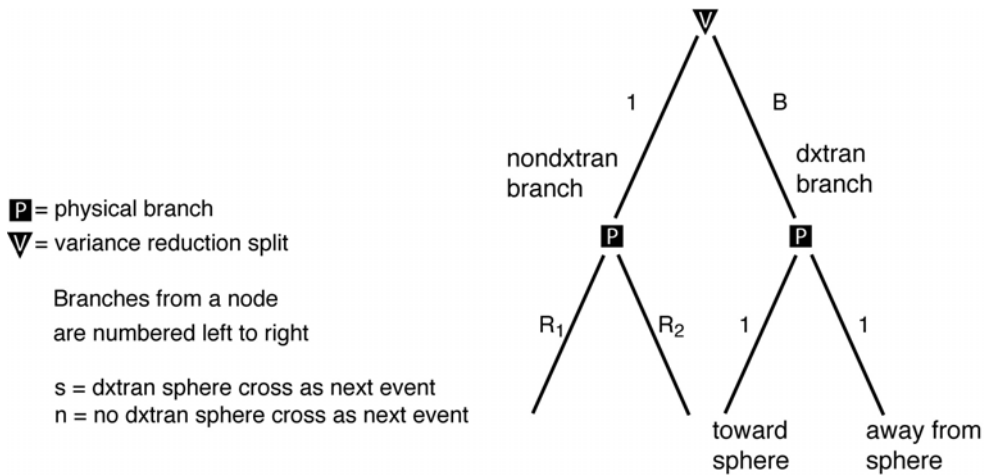
**Fig. 8.2.a. Third Stage: DXTRAN Split Without Sphere Crossing.**



**Fig. 8.2.b. Third Stage: DXTRAN Split with Sphere Crossing.**



**Fig. 8.2.c. Random Branch Weight R from Eq. 51.**



**Fig. 8.2.d. Alternative Random Branch Weighting Used in MCNP.**

will be equivalent to the tree of Fig. 8.2.c because

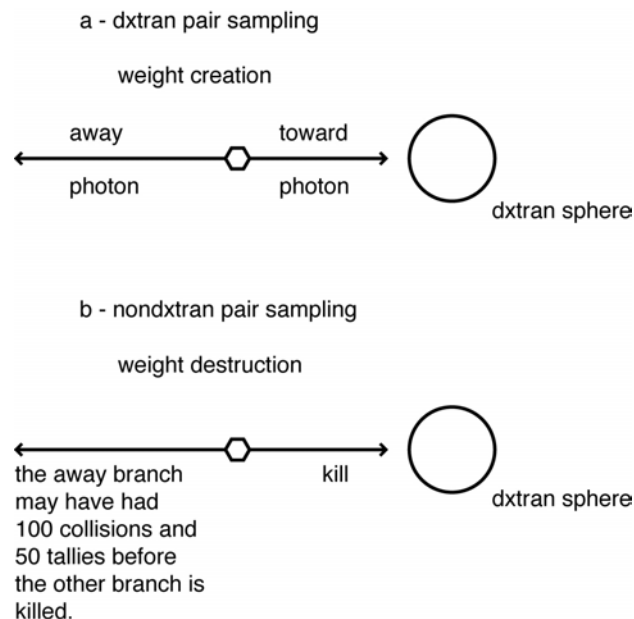
$$R = R_1 R_2 \quad (53)$$

The scheme of Eq. 53 is preferred because the track weight multipliers are identical to the branch weight multipliers, allowing the f8 variance reduction to coexist more gracefully with the non f8 variance reduction.



The attempt at a graceful coexistence of the f8 coding and the non f8 coding brings up a sneakier legerdemain. Note that the f8 DXTRAN creates a pair of photons and the extra weight created by DXTRAN must somehow, on average, be destroyed by DXTRAN. The creation and destruction of track weight is illustrated in Fig. 9. The top diagram indicates how weight is created by the f8 DXTRAN and the bottom diagram indicates how weight ought to be destroyed by the f8 DXTRAN. The trouble is that the left photon in the bottom diagram may have had a very complicated random walk by the time that the right photon is killed by the dxtran sphere. For instance, there might be 100 particles in the bank that were related to the left branch of the bottom diagram. These particles would need to be killed, and any non f8 tallies that were made would have to be undone, else the left branch would have accounted for (and tallied) twice.

One possible solution is to check if one of the particles in the bottom diagram is moving toward the dxtran sphere. If so, the “toward” particle could be tracked first and the “away” particle killed if the toward particle reached the dxtran sphere without collision. In this manner the “away” particle could be killed before it had a chance to post any tallies. In addition to requiring all future code tracking to be “ordered” in this manner, there is another complication. Suppose



**Fig. 9. Pair Sampling.**

the “toward” particle is split along the path to the dxtran sphere so that now there are 100 particles heading toward the sphere; some of them make it to the sphere and some of them do not. This can be handled, but it starts to become a nightmare as one imagines all the variance reduction that is applied to the “toward” branch having to be accounted for on the “away” branch.

There are other possible solutions to the problem exemplified in Fig. 9, but most of the other solutions seem to be nightmares similar to the possible solution in the preceding paragraph. The other possible solutions that were investigated will not be described here, except for the solution chosen for MCNP.

The solution in MCNP was to finesse the problem by creating the “away” dxtran photon only for the f8 tallies and not for the other tallies. That is, the “away” branch in the top diagram is assigned a zero track weight for non f8 tallies, so that the “away” branch in the top diagram is not tallied for non f8 tallies. Because the top “away” photon is not created for non f8 tallies, the bottom “away” photon need not be killed for non f8 tallies when the bottom “toward” photon crosses the dxtran sphere. The f8 branch weight on the “away” dxtran photon is not set to zero and therefore exists for the f8 tally. Note that the f8 tally mechanics sets the tree weight to zero if one of the nondxtran annihilation photons reaches the dxtran sphere. In some sense, DXTRAN simultaneously is creating and destroying one annihilation photon for the non f8 tallies while DXTRAN is creating and destroying an annihilation pair for the f8 tallies. This solution has one unfortunate drawback in that the f8 branch weighting scheme is now not always the same as the track weighting scheme.

One cautionary note is worthwhile here. MCNP currently assumes that the annihilation pair photons have exactly opposite directions. This means that both annihilation pair photons cannot be pointed toward the dxtran sphere. It is the author’s understanding that when the photons are produced in the presence of a nucleus, some of the momentum may be transferred to the nucleus so that the annihilation pair photons need not have exactly opposite directions. If the physics treatment in MCNP is updated so that the annihilation pair photons need not have exactly opposite directions, then the DXTRAN treatment for the annihilation pair photons will need to be modified.

## VIII. THE DECONVOLUTION MECHANICS IN MCNP

Reference 1, Section II, has a detailed deconvolution of a somewhat complicated tree. The only thing different in the present work is that the variance reduction branches arise from processes other than the simple  $n:1$  splits treated in Ref. 1. The mechanics are still the same except that the branch weights are multiplied together instead of the branch probabilities in Ref. 1. Nonetheless, a little redundancy will not hurt, so this section will show how MCNP deconvolutes the tree shown in Fig. 3 herein.

Consider the tree of Fig. 3. Let  $E_{ij}$  be the energy deposited on branch  $j$  under node  $i$ .  $E_{ij}$  only includes the energy deposited until the next node on branch  $j$ . That is,  $E_{ij}$  is the energy deposited between nodes. The deconvolution is started at the highest numbered node (e.g., 8 here) and proceeds backwards (e.g., 8,7,6, ... , 1,0 here) until node 0 is reached. Because nodes 8, 7, and 6 are termination nodes, no energy deposit is associated with them. The possible energy deposited choices under node 5 are shown in Table X. Nodes 3 and 4 are termination nodes, so no energy is deposited. The possible energy deposited choices under node 2 are shown in Table XI. The possible energies deposited under node 1 are shown in Table XII. The possible energies deposited under node 0 are shown in Table XIII. The entries in Table XIII are, in order, associated with the physical trees of Figs. 3.1-3.4.

**TABLE X**  
**POSSIBLE PULSE HEIGHT ENERGIES DEPOSITED**  
**FOR UNDER NODE 5**

energy deposited	(tree weight)
$E_{51}$	$w_{51}$
$E_{52}$	$w_{52}$
$E_{53}$	$w_{53}$

**TABLE XI**  
**POSSIBLE PULSE HEIGHT ENERGIES DEPOSITED**  
**FOR UNDER NODE 2**

energy deposited	(tree weight)
$E_{21} + E_{22}$	$w_{21}w_{22}$

**TABLE XII**  
**POSSIBLE PULSE HEIGHT ENERGIES DEPOSITED**  
**FOR UNDER NODE 1**

energy deposited	(tree weight)
$E_{51} + E_{11}$	$w_{51}w_{11}$
$E_{52} + E_{11}$	$w_{52}w_{11}$
$E_{53} + E_{11}$	$w_{53}w_{11}$
$E_{21} + E_{22} + E_{12}$	$w_{21}w_{22}w_{12}$

**TABLE XIII**  
**POSSIBLE PULSE HEIGHT ENERGIES DEPOSITED**  
**FOR UNDER NODE 0**

energy deposited	(tree weight)
$E_{51} + E_{11} + E_{01}$	$w_{51}w_{11}w_{01}$
$E_{52} + E_{11} + E_{01}$	$w_{52}w_{11}w_{01}$
$E_{53} + E_{11} + E_{01}$	$w_{53}w_{11}w_{01}$
$E_{21} + E_{22} + E_{12} + E_{01}$	$w_{21}w_{22}w_{12}w_{01}$

## **IX. SUMMARY**

This report describes the pulse height tally variance reduction in MCNP version 5 for photon problems. The author has tried to give enough detail so that any future variance reduction techniques required for pulse height tallies can be easily understood and implemented in the deconvolution framework.

## **REFERENCES**

1. Thomas E. Booth, "Monte Carlo Variance Reduction Approaches for Non-Boltzmann Tallies," Los Alamos National Laboratory report LA-12433 (December 1992).
2. J. F. Briesmeister, ed., "MCNP<sup>TM</sup>—A General Monte Carlo N-Particle Transport Code," Los Alamos National Laboratory report LA-13709-M, Version 4C (March 2000).
3. Thomas E. Booth, "A Sample Problem for Variance Reduction in MCNP," Los Alamos National Laboratory report LA-10363-MS (1985).