Appendix F Residence time extraction algorithm

F.1 Algorithm Idea

The idea of the algorithm is to collapse a feliciton to a point with the following properties:

<table>
<thead>
<tr>
<th>Coordinates</th>
<th>x, y, z</th>
</tr>
</thead>
<tbody>
<tr>
<td>residence time</td>
<td>t, i</td>
</tr>
<tr>
<td>transition rates</td>
<td>r, i, j</td>
</tr>
<tr>
<td>transition times</td>
<td>t, i, j</td>
</tr>
</tbody>
</table>

F.2 Algorithm Details

The following algorithm elaborates how to extract this data from the Monte Carlo diffusion simulation:

Inputs: voids description, transition probabilities.

Output: residency time and transition rates and times for each void.

1. Simulate penetrant dynamics according to the MC diffusion model. Start in void 1.

2. Record all "leave" and "entry" events and their times.

3. Each time the following sequence of events happens:
   A. Leaving void i
   B. Entering void j
   C. Leaving void j
   D. Entering void k

Perform the following:

Add \( \text{time}(C) - \text{time}(B) \) to residency \( CT_j \) time of void j
Increment counter of \( CR_{jk} \) and total counter \( C_j \)
Add \( \text{time}(D) - \text{time}(C) \) to transition time \( CRT_{jk} \) and increment counter \( CT_{jk} \)

Time(A) means the step at which event A happened.

When this is done many times (\( C_j >> 1 \)), compute

Residency times: \( T_j = CT_j / C_j \)
Transition rates: \( R_{jk} = CR_{jk} / C_j \)

Transition times: \( TR_{jk} = CRT_{jk} / TC_{jk} \)

**F.3 Assumptions**

Leaving a void is independent of which channel the particle entered from. This assumes enough friction in the void so the particle forgets where it entered from.

We expect \( R_{jk} \) to be symmetric and the residency time distribution to be Poisson.