Whirling Tours and Hitting Times on Trees

Andrew Hannigan

Februrary 25, 2013

Overview

In Dumitriu et al. [1], the authors define a whirling tour on a finite tree as follows:

> Let T be any tree, possibly with loops. Fix a target vertex t, and let v be any other vertex. Order the edges (including loops) incident to each $u \neq t$ arbitrarily subject to the edge on the path from u to t being last. Now walk from v by choosing each exiting edge in round-robin fashion, in accordance with the edge-order at the current vertex, until t is reached. For example, if the edges incident to some degree-3 vertex u are ordered e_1, e_2, e_3, then the first time u is reached it is exited via e1, the second time by e2, the fourth time by e1 again, etc. We call such a walk a whirling tour.

The authors claim that this tour possesses the following property: *In any finite tree (possibly with some loops) the length of any whirling tour from v to t is exactly the expected hitting time from v to t.* Dumitriu et al. leave the proof of this claim to the amusement of the reader. The following is one approach towards proving this claim.

Notation

We define the following notation. We will label vertices slightly differently than in the paper: y will indicate the starting vertex and x will indicate the target vertex. Additionally:

$T_y =$ The tree consisting of all nodes reachable from x *only* via y.
$e_y =$ number of non-loop edges in T_y
$l_y =$ number of loops in T_y
$d_y =$ degree of y in T_y
$whirl(y, x) =$ length of a whirling tour from y to x
$y \sim x$ indicates y is adjacent to x, and $x \sim x$ indicates that there is a loop on x.

For some node z, the first d_z edges in the edge list of z are called child edges. The final edge in the list (the edge on the path from z to x) is called a parent edge. Lastly, henceforth the term "edge" will only indicate a non-loop.

We begin by looking at the length of a whirling tour between two adjacent nodes (Thm 1). We will then show that the expected hitting time between two adjacent nodes is equivalent the length of the whirling tour (Thm 2). Lastly, we will prove that this equivalence generalizes to non-adjacent nodes as well (Thm 3).

Theorem 1: If $y \sim x$, the length of a whirling tour from y to x is $2e_y + l_y + 1$

In order to prove this claim, we will need three lemmas. These lemmas only apply to whirling tours between adjacent vertices $y \sim x$.

Lemma 1: If we reach some arbitrary non-leaf v, every child edge is visited exactly twice and every loop is visited exactly once before taking the parent edge.

- Loops are traversed once and will not be taken again before the parent edge. Recall that a loop contributes only once to the degree of a vertex z. Thus, it will appear only once in the edge list of v.
- If an edge leads to a leaf, it is simply traversed immediately back to v. Thus child edges leading to leaves are traversed exactly twice.
- If an edge leads to a non-leaf, w, we must guarantee that it will return to v. This is simple. Because we are walking on a tree, the only way to return to w is on the edge (w, v). By contradiction, if we never returned on (w, v), then we took some child edge (w, u) and never returned to w. If all child edges from w *had* been taken and returned on, we would have surely returned to z already. The exact same logic can be applied to u. There must have been some child edge of u that we took and never returned on. This logic continues infinitely, which is a contradiction since our tree is finite. Thus, it must be that we always return z.

Lemma 2: All vertices in T_y are reached

If some vertex v was not reached, then by Lemma 1 it's parent node wasn't reached. Continuing this logic inductively, we would conclude that y was never reached. This is a contradiction because we start at y. Thus, all nodes T_y must have been reached by the whirling tour from y to x.

Lemma 3: After leaving some node v by taking its parent edge, we never visit v again on the whirling tour.

When we take the parent edge of v, we traverse back to the parent vertex of v, $p(v)$. We visit the remaining child edges of $p(v)$, and then we move to the parent of $p(v)$, $p(p(v))$. Following this logic, it is evident that we will return to y.

Conclusion

By Lemmas 1, 2, and 3, and the fact that every edge in T_y belongs uniquely to some non-leaf node in T_y, $whirl(y,x) = 2e_y + l_y + 1$ for $y \sim x$.

Theorem 2: $E_y\tau_x = 2e_y + l_y + 1$ in the case where $y \sim x$

Observe the following additional notation.

λ_y = expected return time to y for random walk on T_y.
$E_y\tau_x$ = expected hitting time from from y to x.
$\mathbb{1}_{expr} = 1$ if $expr$ is true, and 0 if $expr$ is false.

If we start a random walk at y, we have a $\frac{1}{d_y+1}$ probability that we will step to x on the first try, and a $\frac{d_y}{d_y+1}$ probability that we will not. If we do not, we must return to y and begin the random walk from y to x once again. Thus:

$$\begin{aligned} E_y\tau_x &= \frac{1}{d_y+1}\cdot 1 + \frac{d_y}{d_y+1}\cdot(\lambda_y + E_y\tau_x) \\ &= 1 + d_y\lambda_y \end{aligned} \tag{1}$$

Additionally, to compute λ_y we can consider two possibilities. If we take a loop away from y, the return time equals 1. Otherwise, the return time equals the initial step we took away from y plus the time to return to y. We consider the average of these possibilities.

$$\lambda_y = \frac{\sum_{z\sim y}[\mathbb{1}_{z\neq y}(1 + E_z\tau_y) + \mathbb{1}_{z=y}]}{d_y} \tag{2}$$

This leads to the following recursion

$$\begin{aligned} d_y\lambda_y &= \sum_{z\sim y}[\mathbb{1}_{z\neq y}(1 + E_z\tau_y) + \mathbb{1}_{z=y}] \\ &= \sum_{z\sim y}[\mathbb{1}_{z\neq y}(2 + d_z\lambda_z) + \mathbb{1}_{z=y}] \\ &= \sum_{z\sim y}[2\cdot\mathbb{1}_{z\neq y} + \mathbb{1}_{z=y}] + \sum_{z\sim y}\mathbb{1}_{z\neq y}(d_z\lambda_z) \end{aligned} \tag{3}$$

This recursion shows that every edge in T_y will be counted twice and every loop will be counted once. Thus, $d_y\lambda_y = 2e_y + l_y$. Note that the $\mathbb{1}_{z\neq y}$ term in the left summation prevents infinite recursion from occurring. Plugging this result back into (1), we have our final result: $E_y\tau_x = 2e_y + l_y + 1$. (Parts of this proof are borrowed from J.W. Moon [2]).

Now we know that the authors' claim holds true if y and x are adjacent. Next, we'll show that this equivalence generalizes to non-adjacent nodes.

Theorem 3: If $x_1, x_2, ..., x_n$ is the unique path from x_1 to x_n, $whirl(x_1, x_n) = whirl(x_1, x_2) + whirl(x_2, x_3) + ... + whirl(x_{n-1}, x_n)$.

Because T is a tree, the whirling tour from x_1 to x_n must visit all vertices on the unique path connecting x_1 and x_n. Consider this whirling tour at the point it first reaches x_2. At this point, we have effectively performed a whirling tour between adjacent nodes x_1 and x_2. By Lemma 3, the last edge taken from every node in T_{x_1} was the parent edge, so the edge list for every node in the graph is set to its first edge. Thus, the next sequence of steps behaves exactly like a whirling tour from x_2 to x_3. This pattern continues until we reach x_n.

Conclusion

Note that, due to linearity of expectation and the uniqueness of the path from x_1 to x_n, $E_{x_1}\tau_{x_n}$ is equal to the expected hitting times between adjacent nodes on the path from x_1 to x_n. This gives us our final result:

$$\begin{aligned} E_{x_1}\tau_{x_n} &= E_{x_1}\tau_{x_2} + ... + E_{x_{n-1}}\tau_{x_n} \\ &= whirl(x_1, x_2) + ... + whirl(x_{n-1}, x_n) \\ &= whirl(x_1, x_n) \end{aligned}$$

Works Cited

- [1] I. Dumitriu, P. Tetali, and P. Winkler. On Playing Golf With Two Balls. SIAM J Discrete Math. Vol 16, No 4. 604-615. 2003.
- [2] J. W. Moon. Random walks on random trees. Journal of the Australian Mathematical Society. Vol 15, No 1. 42?-53. 1973.