The Towers of Hanoi Revisited: Number of Moves and Distribution of Disks

Ernst L. Leiss
[S1]
Department of Computer Science, University of Houston
Houston Texas 77204-3010, USA
coscel@cs.uh.edu

Abstract. The “Towers of Hanoi” is a problem that has been well studied and frequently generalized. We are interested in the generalization to arbitrary directed graphs. We examine two different questions, namely how many moves suffice to move n disks from the starting peg to the destination peg, and under which conditions we can move specific disks using only legal moves to specific pegs. We show that the minimal number of moves to move n disks can be substantially less than 2^n. Moreover, under very mild conditions we can distribute the n disks in completely arbitrary ways over all the available pegs.

Keywords: Towers of Hanoi, number of moves, distribution of disks.

1 Background

A quarter of a century ago, I developed an interest in the Towers of Hanoi game. This is a problem that is frequently (ab)used in data structures and algorithms classes to illustrate the power of recursion. I subsequently generalized the game to be played on graphs; specifically, I assume a finite directed graph G=(V,E) with two distinguished nodes S and D, there are n disks of different sizes on node S such that no larger disk may lie on top of a smaller disk, and the objective is to move the n disks from S to D subject to the following rules:

1. Only one disk may be moved at a time and only along an edge in G.
2. A disk is always placed on top of all the disks on the node where it is moved and no larger disk may ever be placed on top of a smaller disk.

If the problem can be solved for a given graph G for all $n \geq 1$, I call this Hanoi problem solvable. It turned out that there is a rather elegant characterization of all those graphs with solvable Hanoi problems [3]. If for a given graph the associated Hanoi problem is not solvable, I call it a finite Hanoi problem.

Here is the characterization of all graphs G=(V,E) that permit solvable Hanoi problems [3]:

Theorem 0: A Hanoi problem with graph G=(V,E) is solvable if and only if there exist three different nodes u, v, w in V such that:

1. There exists a path in G from S to one of the three nodes u, v, w.
2. There exists a path in G from one of the three nodes u, v, w to D.
3. There exist paths from u to v, from v to w, and from w to u in G.

I subsequently raised the question how many disks can be moved in finite Hanoi problems and derived somewhat surprisingly that one can accommodate at least super-polynomially many disks [5]. Much later it was shown in [1] that there are graphs where sub-exponentially many disks can be moved. More specifically, there exists a constant C such that the number of disks that can be moved in a finite Hanoi game is at most $C \cdot m^{0.5 \cdot \log_2(m)}$. Moreover, for each $\varepsilon>0$, there exists a constant $C_\varepsilon>0$ such that the number of disks that can be moved in some finite Hanoi game is at least $C_\varepsilon \cdot m^{(0.5-\varepsilon) \cdot \log_2(m)}$.

A related question is how many moves are required. While the original Hanoi problem (the complete graph on three nodes) requires 2^n-1 moves, it turns out that significantly fewer moves may be required for some graphs with solvable Hanoi problems. We note that for very few graphs, the number of moves required is known; in [2] an infinite family of graphs is given for which the minimum number of moves is derived.

In this paper, I raise the question what might be the minimum number of moves for any graph to move n disks. For the global minimum, we can of course assume that the graph is the complete graph, as any graph can be embedded into it. More specifically, I assume the complete graph $G=(V,E)$ on the following set of nodes $V=\{S,D,A_1,\ldots,A_m\}$ where S stands for the starting peg, D for the destination peg, and the A_i are auxiliary pegs, for $m\geq 1$. Obviously, for $m=1$, we have the original problem.

Another problem of interest relates to the distribution of the disks over the pegs. More specifically, given a directed graph G on nodes $\{N_1,\ldots,N_\ell\}$ (not necessarily the complete graph), we ask how the n disks may be accumulated on the ℓ pegs using only legal moves. In other words, given n disks, under what conditions can we split them up into ℓ groups G_i, $i=1,\ldots,\ell$, such that $G_1\cup\ldots\cup G_\ell = \{1,\ldots,n\}$ and carry out legal moves in G such that node N_i contains exactly all disks in group G_i, for all $i=1,\ldots,\ell$.

2. The Number of Moves

Generally, more pegs will result in fewer moves. This is quite intuitive since there is more storage for the disks. For example, consider the following graph $G_0=(\{S,D,A\}, \{(S,A),(A,D),(D,S)\})$. One can see that the **only** way (except for redundant, that is, repetitive moves) to move n disks from S to D is as follows:

Two-step move, from S to D:

 Move the smallest n-1 disks from S to D, a two-step move.
 Move the largest disk n from S to A.
 Move the smallest n-1 disks from D to S, a one-step move.
 Move the largest disk n from A to D.
 Move the smallest n-1 disks from S to D, a two-step move.

Of course, now we have to define how to do a one-step move, since the above is a one-step move (from D to S, or in general, from any node to any directly adjacent node). Here is how to achieve this:

One-step move, from S to A:

 Move the smallest n-1 disks from S to D, a two-step move.

Move the largest disk from S to A.

Move the smallest n-1 disks from D to A, a two-step move

First we can verify that all the moves are forced since the largest disk's movements drive the movement of the smaller disks. As a result, we can write the minimum number of required moves in the following recurrence relations: Let $T_1(n)$ be the minimal number of moves required for a one-step move of n disks, and let $T_2(n)$ be the minimum number of moves required for a two-step move of n disks. Then we have:

$T_1(n) = 2T_2(n-1) + 1$ and $T_1(1) = 1$

$T_2(n) = 2T_2(n-1) + T_1(n-1) + 2$ and $T_2(1) = 2$.

Now consider the graph obtained by replacing the edge (A,D) in the graph G_0 by a chain of k edges, for $k\geq 2$; specifically, $G_1 = (\{S,D,A,X_1,\ldots,X_{k-1}\}, \{(S,A), (A,X_1),(X_1,X_2),\ldots(X_{k-1},D),(D,A),(A,S)\})$. It can now be seen that we can "park" edges on the additional pegs so that the number of moves is reduced. Specifically,

(1) Move the n-k+1 smallest disks from S to D;

(2) Move the k-1 largest disks to the pegs $X_{k-1}, \ldots, X_1$ such that disk n is on X_1 through disk n-k+2 on X_{k-1};

(3) Move the n-k+1 smallest disks from D to S;

(4) Move disks n through n-k+2 to D,

(5) Move the n-k+1 smallest disks from S to D.

Clearly, if $M_k(n)$ is the number of moves required by this algorithm to move n disks from S to D, then Steps (1), (3), and (5) require each $M_k(n-k+1)$ moves, Step (2) requires $k(k+1)/2 - 1$ moves, and Step (4) can be implemented as follows: Since A and D are empty, each of the disks n through n-k+2 can be circulated along the ring $A,X_1,\ldots,X_{k-1}$, D by moving each disk one peg further, starting with disk n-k+2 from X_{k-1} to D, then disk n-k+3 from X_{k-2} to X_{k-1}, and so on until disk n is on peg D where it remains. Then the process is repeated for the remaining k-2 disks, until all of them are on D. One complete loop requires k+1 moves, and disk n makes no complete move, disk n-1 makes one, etc., until disk n-k+2 which makes k-2 complete loops. In addition, we have to account for the number of moves required to get from X_i to D, for all i=1,…,k-1. This requires a total of $(k^3 - k^2 - 2k + 2)/2$ moves. Therefore we obtain

$M_k(n) = 3M_k(n-k+1) + (k^3 - k)/2$ for $n\geq k$, and $M_k(n) = (n^2+n)(k+1)/2$ for $1\leq n\leq k-1$.

Now it is not difficult to verify that $M_k(n)$ is smaller than $T_2(n)$ for sufficiently large n, demonstrating that more pegs result in fewer moves.

We formulate the following

Question 1: For every value $m\geq 1$, find the largest number of moves MAX(m) required to move n disks, for any graph G with m+2 nodes and with solvable Hanoi problem.

Clearly, the answer will be a function of n. We know that MAX(3) = 3^n-1. Note that we compare the two functions MAX(m) and MAX(m') of n asymptotically.

Generally, if one considers the complete graph, there is the fairly obvious expectation that with m auxiliary disks, one should achieve a number of moves that is roughly proportional to $2^{n/m}$. This can in fact be achieved, as can be seen as follows. Recall that for the original problem, with one auxiliary peg, 2^n-1 moves for n disks are minimal. This observation is used in the following derivation. Note that in order to simplify the notation, we assume that the number of disks n to be moved is a multiple of the number of auxiliary pegs m.

Assume the complete directed graph on the nodes $\{S,D,A_1,\ldots,A_m\}$ for $m\geq 1$.

Divide the n disks into m groups G_i defined as follows: G_1 contains the smallest n/m disks, G_2 contains the next n/m smallest disks, until G_m contains the largest n/m disks.

Move the disks in G_1 from S to A_1, involving only the three pegs S, D, and A_1. This takes $2^{n/m}$-1 moves.

Move the disks in G_2 from S to A_2, involving only the three pegs S, D, and A_2. This takes $2^{n/m}$-1 moves.

In general, for all i=1,…,m-1, move the disks in G_i from S to A_i, involving only the three pegs S, D, and A_i. This takes $2^{n/m}$-1 moves, for each i=1,…,m-1.

Move the disks in G_m from S to D, involving only the pegs S, D, and A_m. This takes $2^{n/m}$-1 moves.

For all j=m-1,m-2,…,2,1, move the disks in G_j from A_j to D, involving only the three pegs S, D, and A_j. This takes $2^{n/m}$-1 moves for each j.

Adding all the moves up yields $(2m-1)(2^{n/m}-1)$ moves. Since of all the edges in the complete graph, we really need only the 2m edges between S and the A_i and the 2m edges between the A_i and D, plus the 2 edges between S and D, we can do this process on a planar graph with m+2 nodes and 4m+2 edges. Consequently we can state:

Lemma 1: For any $m\geq 2$, there exists a planar graph with m+2 nodes and 4m+2 edges where one needs no more than $(2m-1)\bullet(2^{n/m}-1)$ moves to move n disks from S to D.

In other words, we need at most order of $m\bullet 2^{n/m}$ moves.

Just as in the case of finite Hanoi problems where it turns out that far more disks can be accommodated than I expected, a similar situation exists with the number of moves. We will show below that we can in fact move n disks in a graph with m+2 nodes with a number of moves proportional to $m^2\bullet 2^{n/O(m\bullet m)}$.

For this result, we need the complete graph on the nodes $V=\{S,D,A_1,\ldots,A_m\}$. Again, we define m groups of disks together with their movements:

(1_i) For each i=1,2,…,m-2,m-1, the group G_i consist of the n_i largest disks, excluding all disks in $G_m\cup G_{m-1}\cup\ldots\cup G_{i+1}$. The disks in G_i are moved from S to A_i using the m-i+3 pegs S, D, A_m,…, A_i.

(2) The group G_m consists of the largest n_m disks. The disks in G_m are moved from S to D, using only the 3 pegs S, D, and A_m.

(3_j) Finally, for each j=m-1,m-2,…,2,1, the disks in group G_j are moved from A_j to D, using the m-j+3 pegs S, D, A_m,…, A_j.

Clearly, the number of moves required to achieve (1) is exactly the number of moves required to achieve (3): Simply execute the moves backwards and in reverse order, replacing every occurrence of S by D.

Let us now compute the number of moves that suffice to carry out these steps. We use Lemma 1 to see how many moves are sufficient. Thus, for the first group, G_1 with n_1 disks, to be moved from S to A_1, using pegs S, D, $A_m,\ldots,A_1$, Lemma 1 gives the number of moves as $(2m-1)\bullet(2^{n_1/m}-1)$. In general, for G_i with n_i disks using pegs S, D, $A_m,\ldots, A_i$, Lemma 1 gives the number of moves as $(2(m-i+1)-1)\bullet(2^{n_i/(m-i+1)}-1)$. Let us now choose the following values for the n_i; note that until now they had been unspecified:

n_m:=p for some integer p; n_{m-1}:=2p; n_{m-2}:=3p; in general n_{m-i}:=(i+1)p for i=1,…,m-1.

Then we obtain as the total number of moves for the step (1):

$$(2m-1)\bullet(2^{n_1/m} - 1) + \ldots + (2(m-i+1)-1)\bullet(2^{n_i/(m-i+1)} - 1) + \ldots + 3\bullet(2^{n_{m-1}/2} - 1) =$$
$$(2m-1)\bullet(2^p - 1) + \ldots + (2(m-i+1)-1)\bullet(2^p - 1)+ \ldots + 3\bullet(2^p - 1) =$$
$$(m^2-1)\bullet(2^p - 1).$$

Adding the number of moves for step (2) and those for step (3) (which is the same as for step (1)), we get $(2m^2-1)\bullet(2^p - 1)$. Since $n = n_1+\ldots+ n_m=p\bullet m\bullet(m+1)/2$, we have $(2m^2-1)\bullet(2^p - 1)$ moves for $n = p\bullet m\bullet(m+1)/2$ disks, or on the order of $m^2\bullet 2^{n/O(m\bullet m)}$ moves for n disks on m+2 pegs. Thus, we can summarize:

Proposition 2: Moving n disks from S to D in a complete graph on nodes $\{S,D,A_1,\ldots,A_m\}$ can be achieved in $O(m^2\bullet 2^{n/O(m\bullet m)})$ moves.

Proposition 2 establishes an upper bound on the minimal number of moves. There is no assurance that this is the best result. Thus, we formulate the following

Question 2: For every value m≥1, find the smallest number of moves MIN(m) required to move n disks in the complete graph with m+2 nodes, for m≥3.

Again, MIN(m) is a function of n. We know that MIN(3)=2^n-1. Here I would expect that for larger m, MIN(m) will be strictly smaller.

3. The Distribution of Disks

Given a directed graph G, a natural question is the following: What groups of disks can be accumulated on the nodes of G? I will focus exclusively on solvable Hanoi problems here; it is probably quite difficult to come up with a reasonable characterization in the case of finite Hanoi problems.

In view of Theorem 0, it is necessary that the conditions of this theorem hold, namely there are three different nodes, u, v, and w, such that there is a path from S to one of them, there is a path from one of them to D, and there exist paths from u to v, from v to w, and from w to u. If we now stipulate that there exists a path from one of u, v, or w to each of the nodes $\{N_1,\ldots,N_\ell\}$ of G, then we can choose our groups G_i for i=1,…,ℓ completely arbitrarily, subject only to the stipulation that $G_1\cup\ldots\cup G_\ell = \{1,\ldots,n\}$, and carry out legal moves in G such that node N_i contains exactly all disks in group G_i, for all i=1,…,ℓ. The proof consists of the observation that each disk d with 1≤d≤n can circulate from u to v to w and back to u because the smaller disks 1,…,d-1 can also be circulated. Thus, disk d can be moved to peg N_k, for any 1≤k≤ℓ, since there exists a path from one of the three nodes to N_k. Note however that this condition is sufficient but not necessary. This is because for certain choices of the groups G_i, it might be possible to leave some on S even though there is no path from the distinguished nodes u, v, and w to S. Moreover, in view of the result in [1], there can be sub-exponentially many disks that can be split out onto a subgraph giving rise to a finite Hanoi problem without having a path from any distinguished node to any node in that subgraph. However, it can be seen that in each of these exceptions, the number of disks that can be involved is bounded from above by a function of the number of nodes.

A similar characterization holds if we want to start with the nodes initialized with groups of disks, node N_i holding group G_i, for $i=1,\ldots,\ell$, and ask whether the total number n of disks can be moved, using legal moves, to the destination peg D (which is of course one of the nodes in $\{N_1,\ldots,N_\ell\}$). Again, we can state that if in addition to the conditions of Theorem 0, each node N_i has a path to one of the distinguished nodes u, v, or w, then the problem can always be solved. Once again, there are exceptions, namely if there is a path from a node N_k to D that does not involve any of the distinguished nodes. However, once again, this will be only possible for a limited number of disks, limited by a function of the number ℓ of nodes in G.

4. Conclusion

Although the Tower of Hanoi problem was first published in 1893, it still retains its ability of raise new questions. We have exposed two of these questions. Specifically, we asked what is the minimum number of moves required to move n disks in a given graph with m auxiliary pegs. We have shown that this number is no more than $O(m^2 \bullet 2^{n/(m \bullet m)})$; however, we believe that this can be reduced. Recall that the original problem stated that the world would come to an end once the 64 disks have been successfully moved from S to D. Since with three pegs, this requires 2^{64}-1 moves, which is greater than $18 \bullet 10^{18}$; even assuming just 10 nanosecond per move, this would take over 5700 years. However, if just two more pegs were acquired, using Proposition 2 the number of moves could be reduced to below 34,000, and with three more pegs to below 4000, with obvious significant implications for the end of the world! We have also posed the opposite problem, namely determining what graphs required the largest number of moves.

Then we turned to the problem of distributing, via legal moves, the n disks over the ℓ nodes of a general directed graph. We have shown that except for a bounded number of disks, there is a simple characterization of the graphs for which this is possible for arbitrary n. This bound depends only on ℓ. In a similar vein, moving the n disks to the destination peg starting with the disks arbitrarily distributed over the various nodes in G is also always possible, except for a bounded number of disks, provided a very simple property holds for the graph. In [4], a number of restrictions in terms of the nodes and edges are discussed (specifically, disks have different “colors” and certain nodes and edges will accommodates only certain colors), and it is expected that the characterizations of graphs that permit arbitrary distribution of disks over their nodes will be somewhat more challenging.

Bibliography

1. D. Azriel and D. Berend: On a Question of Leiss Regarding the Hanoi Tower Problem, Theoretical Computer Science 369, 1-3, 377-383 (2006).
2. D. Azriel, N. Solomon, and S. Solomon: On an Infinite Family of Solvable Hanoi Graphs, ACM Transactions on Algorithms, Vol. 5, No. 1, Article 13, Nov. (2008).
3. E. L. Leiss: Solving the 'Towers of Hanoi' on graphs, Journal of Combinatorics, Information and System Sciences, Vol. 8, No. 1, 81-89, March (1983).
4. E. L. Leiss: On restricted Hanoi problems, Journal of Combinatorics, Information and System Sciences Vol. 8, No. 4, 277-285, December (1983).
5. E. L. Leiss: Finite Hanoi problems: How many discs can be handled? Fifteenth Southeastern Conference on Combinatorics, Graph Theory, and Computing, Baton Rouge, Louisiana, March 5-8, 1984; Congressus Numerantium Vol. 44, 221-229, December (1984).