

Relazione tra torre di Hanoi a p pioli e funzioni binomiali generalizzate

Lo scopo di questo documento è indagare la relazione tra il numero minimo di mosse richieste per risolvere una variante della torre di Hanoi con p pioli e le equazioni binomiali generalizzate. Inoltre vorrei proporre un metodo per la costruzione di serie numeriche, capaci di calcolare con complessità $O(n)$ (o $O(1)$ se portate in forma chiusa) il numero minimo di mosse necessarie per risolvere torri di Hanoi con un numero di pioli maggiore di 3 in funzione al numero di dischi k. Formulando la seguente congettura:

Congettura di Marabottini:

Esiste una soluzione s dell'equazione

$$\prod_{i=0}^{p-2} \frac{n+i}{i+1} = k$$

$$(\prod_{i=0}^{p-2} \frac{n+i}{i+1} = k)$$

tale che

$$\sum_{i=0}^{k-1} 2^{\lfloor s(i) \rfloor}$$

$$(\sum_{i=0}^{k-1} 2^{\lfloor s(i) \rfloor})$$

corrisponde al numero minimo di mosse necessarie in una torre di Hanoi a m piloni con k dischi

E, da questa congettura, estrapolare alcune formule (su 4, 5 e 6 piloni) e confrontare i risultati con i dati noti dell'algoritmo di Frame-Stewart allo scopo di validarla empiricamente

Indice generale

1. Introduzione.....	3
1.1 Cosa è la torre di Hanoi ?.....	3
1.2 Varianti.....	3
1.3 Perché ciò è importante.....	3
2. Bibliografia.....	4
3. Metodologia.....	5
3.1 Raccolta dati.....	5
3.2 Osservazione dei pattern.....	6
3.3 Ricavo della formula per Hanoi a 4 piloni.....	6
3.4 Altre varianti sulla torre di Hanoi.....	9
3.5 Generalizzazione.....	13
4. Evidenze empiriche.....	14
4.1 Test a 4 piloni.....	14
4.2 Test a 5 piloni.....	16
3.4 Test a 6 piloni.....	21

1. Introduzione

1.1 Cosa è la torre di Hanoi ?

La torre di Hanoi è un rompicapo matematico inventato da Edward Lucas nel 1883 composto da 3 piloni e un numero definito di anelli posti in ordine decrescente (il più grande in fondo).

Lo scopo del gioco è portare ogni disco in uno dei due piloni, usandone un terzo come appoggio, spostando i dischi uno alla volta e facendo in modo che l'anello più grande rimanga sempre sotto il più piccolo.

Questo gioco ha interessanti proprietà in quanto può essere usato come metafora della ricorsione e il numero di mosse necessarie cresce in modo esponenziale in base al numero di anelli (il numero di mosse necessarie per la torre di Hanoi a 3 piloni può essere calcolato con la formula $2^{\{n\}}-1$).

1.2 Varianti

Negli anni sono state inventate varie versioni, molte delle quali variano solo per numero di piloni. La presenza del più piloni, aggiunge piani di appoggio al gioco e rende necessarie meno mosse man mano che ne cresce il numero.

Nonostante ciò, calcolare il numero minimo di mosse necessarie diventa molto più complicato, in quanto aumentano le possibilità ed esistono più strade che possano risolvere il problema applicando il numeri di passi minimi possibili.

Al momento, da quanto mi risulta mi risulta, l'unico modo per sapere quante sono le mosse minime che un giocatore debba fare è usando la congettura di frame-stewart, che risolve il problema ad un costo sub-esponenziale.

1.3 Perché ciò è importante

La mia congettura, invece, da la possibilità di risolvere il problema con un costo lineare, basandosi su serie numeriche, ma che può diventare costante qualora queste serie venissero trasformate in formula chiusa.

2. Bibliografia

Se si volessero approfondire i vari ambiti è possibile leggere le pagine di wikipedia della torre di Hanoi

Torre di Hanoi wikipedia → https://it.wikipedia.org/wiki/Torre_di_Hanoi

Spiegazione del problema di reeves → <https://demonstrations.wolfram.com/TheRevesPuzzle/>

Di seguito riporto dei tabulati trovati online su varie versioni della torre di hanoi

hanoi tower 4 pegs → <https://oeis.org/A007664>

hanoi tower 5 pegs → <https://oeis.org/A007665>

hanoi tower 6 pegs → <https://oeis.org/A182058>

Un jsfiddle per testare il numero di mosse sulla torre di Hanoi a 4 pioli

test → <https://jsfiddle.net/83kwLdfj/>

3. Metodologia

3.1 Raccolta dati

I dati di riferimento sono stati raccolti in parte tramite il sito oeis.org, ed in parte interpellando l'LLM ChatGpt.

Inizialmente chiedendogli di calcolare, tramite l'algoritmo di Frame-Stewart, il numero minimo di passaggi casistiche da 1 a 30 dischi su un caso a 4 piloni.

Dischi	Mosse	Incremento	Numero Gauss
1	1		1
2	3	2	
3	5	2	2+1
4	9	4	
5	13	4	
6	17	4	3+2+1
7	25	8	
8	33	8	
9	41	8	
10	49	8	4+3+2+1
11	65	16	
12	81	16	
13	97	16	
14	113	16	
15	129	16	5+4+3+2+1
16	161	32	
17	193	32	
18	225	32	
19	257	32	
20	289	32	
21	321	32	6+5+4+3+2+1
22	385	64	
23	449	64	
24	513	64	

25	577	64	
26	641	64	
27	705	64	
28	769	64	7+6+5+4+3+2+1
29	897	128	
30	1025	128	

3.2 Osservazione dei pattern

Osservando la differenza di mosse minime tra una torre a 4 piloni con un certo numero di dischi e lo stesso valore aggiungendone uno, ho notato che questo valore aumentava sempre in potenze di 2.

Ho inoltre identificato un pattern in cui la potenza di 2 di questo incremento aumenta con una periodicità correlata ai numeri di Gauss (ricavabile con la formula $n(n+1)/2$).

3.3 Ricavo della formula per Hanoi a 4 piloni

Da queste informazioni ho intuito che il pattern doveva essere una sommatoria di potenze di 2 che avesse a che fare con l'arrotondamento e l'inversione della formula di gauss.

Per prima cosa ho eguagliato la formula di gauss con il numero di diski (k)

$$n(n+1)/2 = k.$$

La soluzione della seguente equazione mi ha portato ai seguenti risultati, come anche calcolato su [wolfram alpha](#),

$$1/2(\sqrt{8k+1}-1)$$

$$1/2(-\sqrt{8k+1}-1)$$

Dato che il risultato deve essere sempre positivo, ho scelto la soluzione reale e positiva per ogni $k > 0$, escludendo quindi la seconda soluzione in favore della prima.

Essendo il cambio di potenze di 2 a scaglioni, ho pensato che i valori di questa soluzione andassero arrotondati, probabilmente per difetto provando a vedere cosa succedesse applicando il floor.

A quel punto ho confrontato le differenze con le potenze di 2 ricavate dalla seguente formula

$$\lfloor \frac{\sqrt{1+8i} - 1}{2} \rfloor$$

Dischi	Incremento	f(k)	2 ^{f(k)}
1		1	2

2	2	1	2
3	2	2	4
4	4	2	4
5	4	2	4
6	4	3	8
7	8	3	8
8	8	3	8
9	8	3	8
10	8	4	16
11	16	4	16
12	16	4	16
13	16	4	16
14	16	4	16
15	16	5	32
16	32	5	32
17	32	5	32
18	32	5	32
19	32	5	32
20	32	5	32
21	32	6	64
22	64	6	64
23	64	6	64
24	64	6	64
25	64	6	64
26	64	6	64
27	64	6	64
28	64	7	128
29	128	7	128
30	128	7	128

Il valore risulta simile, ma solo disallineato.

Il numero di anelli deve essere -1

Dischi	Incremento	f(k-1)	2 ^{f(k-1)}
1		0	1
2	2	1	2
3	2	1	2
4	4	2	4
5	4	2	4
6	4	2	4
7	8	3	8
8	8	3	8
9	8	3	8
10	8	3	8
11	16	4	16
12	16	4	16
13	16	4	16
14	16	4	16
15	16	4	16
16	32	5	32
17	32	5	32
18	32	5	32
19	32	5	32
20	32	5	32
21	32	5	32
22	64	6	64
23	64	6	64
24	64	6	64
25	64	6	64
26	64	6	64
27	64	6	64
28	64	6	64
29	128	7	128
30	128	7	128

Da li ho ricavato la formula finale: $\sum_{i=0}^{k-1} 2^{\lfloor \frac{\sqrt{1+8i} - 1}{2} \rfloor}$

e verificato che riuscisse a prevedere i risultati fino al puzzle di reeves con 121 anelli.

3.4 Altre varianti sulla torre di Hanoi

Successivamente ho provato ad applicare lo stesso modus operandi anche per la torre di Hanoi a 5 pioli e quella a 6.

Nello specifico ho notato che il numero minimo di soluzioni per la torre di hanoi a 5 pioli seguiva un pattern molto simile, con la differenza che i break point non erano più calcolabili tramite la formula $n(n+1)/2$, ma tramite $n(n+1)(n+2)/6$ mentre in quella a 6 tramite $n(n+1)(n+2)(n+3)/24$, portandomi alle formule già citate in precedenza e che riciterò nel capitolo sottostante, quando andremo alla dimostrazione empirica dei risultati.

Inoltre, la formula della torre di hanoi classica $2^n - 1$ può essere riscritta come $\sum_{i=0}^{k-1} 2^i$

Di seguito la tabella a 5 piloni e la configurazione della soluzione-reale-positiva

1	1	
2	3	2
3	5	2
4	7	2
5	11	4
6	15	4
7	19	4
8	23	4
9	27	4
10	31	4
11	39	8
12	47	8
13	55	8
14	63	8
15	71	8
16	79	8
17	87	8
18	95	8
19	103	8
20	111	8

21	127	16
22	143	16
23	159	16
24	175	16
25	191	16
26	207	16
27	223	16
28	239	16
29	255	16
30	271	16
31	287	16
32	303	16
33	319	16
34	335	16
35	351	16
36	383	32
37	415	32
38	447	32
39	479	32
40	511	32
41	543	32
42	575	32
43	607	32
44	639	32
45	671	32
46	703	32
47	735	32
48	767	32
49	799	32
50	831	32
51	863	32
52	895	32
53	927	32
54	959	32

55	991	32
56	1023	32
57	1087	64
58	1151	64
59	1215	64
60	1279	64
61	1343	64
62	1407	64
63	1471	64
64	1535	64
65	1599	64
66	1663	64
67	1727	64
68	1791	64
69	1855	64
70	1919	64
71	1983	64
72	2047	64
73	2111	64
74	2175	64
75	2239	64
76	2303	64
77	2367	64
78	2431	64
79	2495	64
80	2559	64
81	2623	64
82	2687	64
83	2751	64
84	2815	64
85	2943	128
86	3071	128
87	3199	128
88	3327	128

89	3455	128
90	3583	128
91	3711	128
92	3839	128
93	3967	128
94	4095	128
95	4223	128
96	4351	128
97	4479	128
98	4607	128
99	4735	128
100	4863	128
101	4991	128
102	5119	128
103	5247	128
104	5375	128
105	5503	128
106	5631	128
107	5759	128
108	5887	128
109	6015	128
110	6143	128
111	6271	128
112	6399	128
113	6527	128
114	6655	128
115	6783	128
116	6911	128
117	7039	128
118	7167	128
119	7295	128
120	7423	128
121	7679	256

--	--	--

Ecco le distanze

Val cambi	Differenza	Differenza doppia
1		
4	3	
10	6	3
20	10	4
35	15	5
56	21	6
84	28	7
120	36	8

3.5 Generalizzazione

Da qui mi è parso lampante che l'esponenziale fosse sempre il floor di una soluzione dell'equazione

$\prod_{i=0}^{n-2} \frac{n+i}{i+1} = k$, ed ho potuto arrivare all'assunto che

Esiste una soluzione s dell'equazione $\prod_{i=0}^{n-2} \frac{n+i}{i+1} = k$ tale che $\sum_{j=0}^{k-1} s(j)$ corrisponde al numero minimo di mosse a n piloni

4. Evidenze empiriche

4.1 Test a 4 piloni

Analisi empirica Mosse con frame stewart rispetto alla formula trovata

$\sum_{i=0}^{n-1} 2^{\lfloor \frac{\sqrt{1+8i}-1}{2} \rfloor}$

la formula base deve andare da 0 a n-1

wolfram alpha formula:

sum $2^{\text{floor}((\sqrt{1+8*i}) - 1) / 2}$ for i = 0 to n-1

esempio 80 => sum $2^{\text{floor}((\sqrt{1+8*i}) - 1) / 2}$ for i = 0 to 79

funzione riscritta in javascript

$$\sum_{i=0}^{k-1} 2^{\lfloor \frac{\sqrt{1+8i}-1}{2} \rfloor}$$

formula chiusa

$\text{Bigl}(n + K - 1 - \frac{K(K+1)}{2}\text{Bigr)\,}2^K + 1, \quad K = \left\lfloor \frac{\sqrt{1+8(n-1)}-1}{2} \right\rfloor.$

$$\left(n + K - 1 - \frac{K(K+1)}{2}\right) 2^K + 1, \quad K = \left\lfloor \frac{\sqrt{1+8(n-1)}-1}{2} \right\rfloor.$$

Dischi	Mosse frame-stewart	funzione
1	1	1
2	3	3
3	5	5
4	9	9
5	13	13
6	17	17
7	25	25
8	33	33
9	41	41
10	49	49
11	65	65

Relazione tra torre di Hanoi p pioli e funzioni binomiali generalizzate – David Marabottini

12	81	81
13	97	97
14	113	113
15	129	129
16	161	161
17	193	193
18	225	225
19	257	257
20	289	289
21	321	321
22	385	385
23	449	449
24	513	513
25	577	577
26	641	641
27	705	705
28	769	769
29	897	897
30	1025	1025
50	6657	6657
80	53249	53249

4.2 Test a 5 piloni

```

\sum_{i=0}^{k-1} 2^{\left\lfloor \frac{\sqrt[3]{\sqrt{3}\sqrt{243i^2-1}+27i}}{3} + \frac{1}{3\sqrt[3]{\sqrt{3}\sqrt{243i^2-1}+27i}} - 1 \right\rfloor}
// da rivedere in quanto perde di chiarezza
// la formula base deve andare da 0 a n-1
wolfram alpha fomula (ho aggiunto 10^-10 per risolvere un problema di arrotondamento)
sum 2^floor( ( (3^(1/2) * (243*i^2 - 1)^(1/2) + 27*i)^(1/3) ) / (3^(2/3)) + 1 / ( 3^(1/3) * ( 3^(1/2) *
(243*i^2 - 1)^(1/2) + 27*i)^(1/3) ) ) - 1 + 10^-10) for i= 0 to n-1
esempio 121 => sum 2^floor( ( (3^(1/2) * (243*i^2 - 1)^(1/2) + 27*i)^(1/3) ) / (3^(2/3)) + 1 / ( 3^(1/3) *
( 3^(1/2) * (243*i^2 - 1)^(1/2) + 27*i)^(1/3) ) ) - 1 + 10^-10) for i= 0 to 120

```

$$\sum_{i=0}^{k-1} 2^{\left\lfloor \frac{\frac{\sqrt[3]{\sqrt{3}\sqrt{243i^2-1}+27i}}{3}}{\frac{2}{3}} + \frac{1}{3\sqrt[3]{\sqrt{3}\sqrt{243i^2-1}+27i}} - 1 \right\rfloor}$$

X	Frame-stewart	Mia funzione
1	1	1
2	3	3
3	5	5
4	7	7
5	11	11
6	15	15
7	19	19
8	23	23
9	27	27
10	31	31
11	39	39
12	47	47
13	55	55
14	63	63
15	71	71
16	79	79

17	87	87
18	95	95
19	103	103
20	111	111
21	127	127
22	143	143
23	159	159
24	175	175
25	191	191
26	207	207
27	223	223
28	239	239
29	255	255
30	271	271
31	287	287
32	303	303
33	319	319
34	335	335
35	351	351
36	383	383
37	415	415
38	447	447
39	479	479
40	511	511
41	543	543
42	575	575
43	607	607
44	639	639
45	671	671
46	703	703
47	735	735
48	767	767
49	799	799
50	831	831

51	863	863
52	895	895
53	927	927
54	959	959
55	991	991
56	1023	1023
57	1087	1087
58	1151	1151
59	1215	1215
60	1279	1279
61	1343	1343
62	1407	1407
63	1471	1471
64	1535	1535
65	1599	1599
66	1663	1663
67	1727	1727
68	1791	1791
69	1855	1855
70	1919	1919
71	1983	1983
72	2047	2047
73	2111	2111
74	2175	2175
75	2239	2239
76	2303	2303
77	2367	2367
78	2431	2431
79	2495	2495
80	2559	2559
81	2623	2623
82	2687	2687
83	2751	2751
84	2815	2815

85	2943	2943
86	3071	3071
87	3199	3199
88	3327	3327
89	3455	3455
90	3583	3583
91	3711	3711
92	3839	3839
93	3967	3967
94	4095	4095
95	4223	4223
96	4351	4351
97	4479	4479
98	4607	4607
99	4735	4735
100	4863	4863
101	4991	4991
102	5119	5119
103	5247	5247
104	5375	5375
105	5503	5503
106	5631	5631
107	5759	5759
108	5887	5887
109	6015	6015
110	6143	6143
111	6271	6271
112	6399	6399
113	6527	6527
114	6655	6655
115	6783	6783
116	6911	6911
117	7039	7039
118	7167	7167

119	7295	7295
120	7423	7423
121	7679	7679

Ho verificato che la mia formula funziona almeno fino al 121 disco

3.4 Test a 6 piloni

$$\sum_{i=0}^{k-1} 2^{\left\lfloor \frac{1}{2} (\sqrt{4\sqrt{24n+1}+5}-3) \right\rfloor}$$

$$\sum_{i=0}^{k-1} 2^{\left\lfloor \frac{1}{2} (\sqrt{4\sqrt{24n+1}+5}-3) \right\rfloor}$$

formula wolfram alpha

sum 2^floor((1/2)*(sqrt(4*sqrt(24*i+1)+5) - 3)) for i=0 to k-1

esempio 40 => sum 2^floor((1/2)*(sqrt(4*sqrt(24*i+1)+5) - 3)) for i=0 to 39

la formula base deve andare da 0 a n-1

n	frame-stewart	f(n)
1	1	1
2	3	3
3	5	5
4	7	7
5	9	9
6	13	13
7	17	17
8	21	21
9	25	25
10	29	29
11	33	33
12	37	37
13	41	41

14	45	45
15	49	49
16	57	57
17	65	65
18	73	73
19	81	81
20	89	89
21	97	97
22	105	105
23	113	113
24	121	121
25	129	129
26	137	137
27	145	145
28	153	153
29	161	161
30	169	169
31	177	177
32	185	185
33	193	193
34	201	201
35	209	209
36	225	225
37	241	241
38	257	257
39	273	273
40	289	289