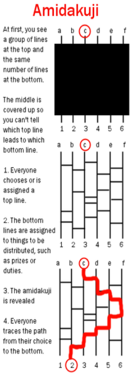
Amidakuji (ghost leg): From a simple game to research in different areas

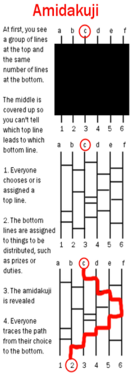
Chi-Kwong Li
The College of William and Mary

Amidakuji/Ghost Leg Drawing



Amidakuji/Ghost Leg Drawing

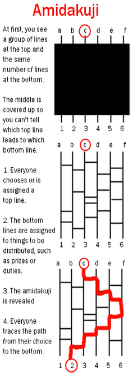
It is a scheme for assigning n people $P_1, \ldots, P_n$ to n jobs $J_1, \ldots, J_n$ "randomly".



Amidakuji/Ghost Leg Drawing

It is a scheme for assigning n people $P_1, \ldots, P_n$ to n jobs $J_1, \ldots, J_n$ "randomly".

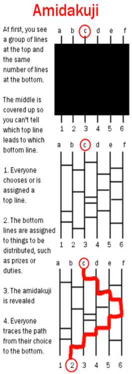
- Draw vertical lines from P_i to J_i from $i = 1, \ldots, n$.



Amidakuji/Ghost Leg Drawing

It is a scheme for assigning n people $P_1, \ldots, P_n$ to n jobs $J_1, \ldots, J_n$ "randomly".

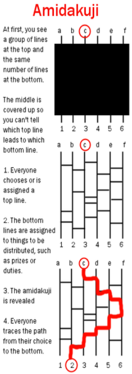
- Draw vertical lines from P_i to J_i from $i = 1, \ldots, n$.
- Draw some horizontal line segments randomly between any two vertical lines that are next to each other so that no horizontal lines meet.



Amidakuji/Ghost Leg Drawing

It is a scheme for assigning n people $P_1, \ldots, P_n$ to n jobs $J_1, \ldots, J_n$ "randomly".

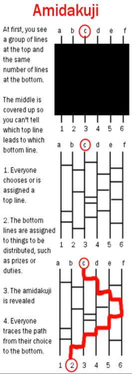
- Draw vertical lines from P_i to J_i from $i = 1, \ldots, n$.
- Draw some horizontal line segments randomly between any two vertical lines that are next to each other so that no horizontal lines meet.
- To assign a job for P_i, start from the top of the i-th line to the bottom, and make a turn whenever a horizontal segment is encountered.



Amidakuji/Ghost Leg Drawing

It is a scheme for assigning n people $P_1, \ldots, P_n$ to n jobs $J_1, \ldots, J_n$ “randomly”.

- Draw vertical lines from P_i to J_i from $i = 1, \ldots, n$.
- Draw some horizontal line segments randomly between any two vertical lines that are next to each other so that no horizontal lines meet.
- To assign a job for P_i, start from the top of the i-th line to the bottom, and make a turn whenever a horizontal segment is encountered.



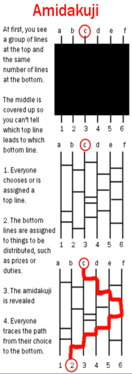
Questions

- Why do we always get an one-one correspondence (bijection)?

Amidakuji/Ghost Leg Drawing

It is a scheme for assigning n people $P_1, \ldots, P_n$ to n jobs $J_1, \ldots, J_n$ "randomly".

- Draw vertical lines from P_i to J_i from $i = 1, \ldots, n$.
- Draw some horizontal line segments randomly between any two vertical lines that are next to each other so that no horizontal lines meet.
- To assign a job for P_i, start from the top of the i-th line to the bottom, and make a turn whenever a horizontal segment is encountered.



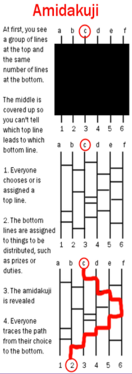
Questions

- Why do we always get an one-one correspondence (bijection)?
- Can we get all possible job assignments?

Amidakuji/Ghost Leg Drawing

It is a scheme for assigning n people $P_1, \ldots, P_n$ to n jobs $J_1, \ldots, J_n$ "randomly".

- Draw vertical lines from P_i to J_i from $i = 1, \ldots, n$.
- Draw some horizontal line segments randomly between any two vertical lines that are next to each other so that no horizontal lines meet.
- To assign a job for P_i, start from the top of the i-th line to the bottom, and make a turn whenever a horizontal segment is encountered.



Questions

- Why do we always get an one-one correspondence (bijection)?
- Can we get all possible job assignments?
- What is the minimum number of horizontal segments needed for a given job assignment?

Answer of Question 1

Answer of Question 1

- Polya principle:
  If one cannot solve a problem, one can try to solve an easier problem first.

Answer of Question 1

- Polya principle:
  If one cannot solve a problem, one can try to solve an easier problem first.
- What if there is no horizontal line segment?

Answer of Question 1

- Polya principle:
  If one cannot solve a problem, one can try to solve an easier problem first.
- What if there is no horizontal line segment?
- What if there is one horizontal line segment?

Answer of Question 1

- Polya principle:
  If one cannot solve a problem, one can try to solve an easier problem first.
- What if there is no horizontal line segment?
- What if there is one horizontal line segment?
- An easy induction argument!

Bubble sort

- Regard the job assignment as a permutation (a seat reassignment)

$$\sigma = [i_1, \ldots, i_n] = \begin{pmatrix} 1 & 2 & \cdots & n \\ i_1 & i_2 & \cdots & i_n \end{pmatrix}.$$

Bubble sort

- Regard the job assignment as a permutation (a seat reassignment)

$$\sigma = [i_1, \ldots, i_n] = \begin{pmatrix} 1 & 2 & \cdots & n \\ i_1 & i_2 & \cdots & i_n \end{pmatrix}.$$

- Use Coxeter transpositions $(i, i+1)$ for $i = 1, \ldots, n-1$.

Bubble sort

- Regard the job assignment as a permutation (a seat reassignment)

$$\sigma = [i_1, \ldots, i_n] = \begin{pmatrix} 1 & 2 & \cdots & n \\ i_1 & i_2 & \cdots & i_n \end{pmatrix}.$$

- Use Coxeter transpositions $(i, i+1)$ for $i = 1, \ldots, n-1$.
- For any σ, we can determine its number of inversions, which will be the minimum number of Coxeter transpositions needed to generate σ.

Bubble sort

- Regard the job assignment as a permutation (a seat reassignment)

$$\sigma = [i_1, \ldots, i_n] = \begin{pmatrix} 1 & 2 & \cdots & n \\ i_1 & i_2 & \cdots & i_n \end{pmatrix}.$$

- Use Coxeter transpositions $(i, i+1)$ for $i = 1, \ldots, n-1$.
- For any σ, we can determine its number of inversions, which will be the minimum number of Coxeter transpositions needed to generate σ.

Example For $\sigma = [5, 3, 1, 2, 4]$, total number of inversions is: $4 + 0 + 2 = 6$, and

$$\sigma \to [3, 5, 1, 2, 4] \to [3, 1, 5, 2, 4] \to [3, 1, 2, 5, 4]$$

$$\to [3, 1, 2, 4, 5] \to [1, 3, 2, 4, 5] \to [1, 2, 3, 4, 5],$$

Bubble sort

- Regard the job assignment as a permutation (a seat reassignment)

$$\sigma = [i_1, \ldots, i_n] = \begin{pmatrix} 1 & 2 & \cdots & n \\ i_1 & i_2 & \cdots & i_n \end{pmatrix}.$$

- Use Coxeter transpositions $(i, i+1)$ for $i = 1, \ldots, n-1$.
- For any σ, we can determine its number of inversions, which will be the minimum number of Coxeter transpositions needed to generate σ.

Example For $\sigma = [5, 3, 1, 2, 4]$, total number of inversions is: $4 + 0 + 2 = 6$, and

$$\sigma \to [3, 5, 1, 2, 4] \to [3, 1, 5, 2, 4] \to [3, 1, 2, 5, 4]$$

$$\to [3, 1, 2, 4, 5] \to [1, 3, 2, 4, 5] \to [1, 2, 3, 4, 5],$$

So $\sigma = (1, 2)(2, 3)(3, 4)(4, 5)(1, 2)(2, 3)$.

Bubble sort

- Regard the job assignment as a permutation (a seat reassignment)

$$\sigma = [i_1, \ldots, i_n] = \begin{pmatrix} 1 & 2 & \cdots & n \\ i_1 & i_2 & \cdots & i_n \end{pmatrix}.$$

- Use Coxeter transpositions $(i, i+1)$ for $i = 1, \ldots, n-1$.
- For any σ, we can determine its number of inversions, which will be the minimum number of Coxeter transpositions needed to generate σ.

Example For $\sigma = [5, 3, 1, 2, 4]$, total number of inversions is: $4 + 0 + 2 = 6$, and

$$\sigma \to [3, 5, 1, 2, 4] \to [3, 1, 5, 2, 4] \to [3, 1, 2, 5, 4]$$

$$\to [3, 1, 2, 4, 5] \to [1, 3, 2, 4, 5] \to [1, 2, 3, 4, 5],$$

So $\sigma = (1, 2)(2, 3)(3, 4)(4, 5)(1, 2)(2, 3)$.

Answers of Questions 2 and 3

- We can always convert a permutation σ to $[1, \ldots, n]$ using m steps, where is the number of inversions of σ.

Bubble sort

- Regard the job assignment as a permutation (a seat reassignment)

$$\sigma = [i_1, \ldots, i_n] = \begin{pmatrix} 1 & 2 & \cdots & n \\ i_1 & i_2 & \cdots & i_n \end{pmatrix}.$$

- Use Coxeter transpositions $(i, i+1)$ for $i = 1, \ldots, n-1$.
- For any σ, we can determine its number of inversions, which will be the minimum number of Coxeter transpositions needed to generate σ.

Example For $\sigma = [5, 3, 1, 2, 4]$, total number of inversions is: $4 + 0 + 2 = 6$, and

$$\sigma \to [3, 5, 1, 2, 4] \to [3, 1, 5, 2, 4] \to [3, 1, 2, 5, 4]$$

$$\to [3, 1, 2, 4, 5] \to [1, 3, 2, 4, 5] \to [1, 2, 3, 4, 5],$$

So $\sigma = (1, 2)(2, 3)(3, 4)(4, 5)(1, 2)(2, 3)$.

Answers of Questions 2 and 3

- We can always convert a permutation σ to $[1, \ldots, n]$ using m steps, where is the number of inversions of σ.
- **Worst case** occurs at $[n, n-1, \ldots, 1]$; we need

Bubble sort

- Regard the job assignment as a permutation (a seat reassignment)

$$\sigma = [i_1, \ldots, i_n] = \begin{pmatrix} 1 & 2 & \cdots & n \\ i_1 & i_2 & \cdots & i_n \end{pmatrix}.$$

- Use Coxeter transpositions $(i, i+1)$ for $i = 1, \ldots, n-1$.
- For any σ, we can determine its number of inversions, which will be the minimum number of Coxeter transpositions needed to generate σ.

Example For $\sigma = [5, 3, 1, 2, 4]$, total number of inversions is: $4 + 0 + 2 = 6$, and

$$\sigma \to [3, 5, 1, 2, 4] \to [3, 1, 5, 2, 4] \to [3, 1, 2, 5, 4]$$

$$\to [3, 1, 2, 4, 5] \to [1, 3, 2, 4, 5] \to [1, 2, 3, 4, 5],$$

So $\sigma = (1, 2)(2, 3)(3, 4)(4, 5)(1, 2)(2, 3)$.

Answers of Questions 2 and 3

- We can always convert a permutation σ to $[1, \ldots, n]$ using m steps, where is the number of inversions of σ.
- **Worst case** occurs at $[n, n-1, \ldots, 1]$; we need

$$(n-1) + \cdots + 1 = n(n-1)/2 \text{ steps.}$$

A variation

- A general principle in study and research:
  If you have solved a problem, extend the techniques to related problems.

A variation

- A general principle in study and research:
  If you have solved a problem, extend the techniques to related problems.
- What if we consider transpositions of the forms $(i, i+1)$ and $(i, i+2)$?

A variation

- A general principle in study and research:
  If you have solved a problem, extend the techniques to related problems.
- What if we consider transpositions of the forms $(i, i+1)$ and $(i, i+2)$?
- How about using transpositions $(i, i+1), (i, i+2), (i, i+3)$, etc.?

A variation

- A general principle in study and research:
  If you have solved a problem, extend the techniques to related problems.
- What if we consider transpositions of the forms $(i, i+1)$ and $(i, i+2)$?
- How about using transpositions $(i, i+1), (i, i+2), (i, i+3)$, etc.?

An extreme case: Using all (i, j) with $1 \leq j \leq n$

Decompose σ as product of k disjoint cycles (including fixed points).

A variation

- A general principle in study and research:
  If you have solved a problem, extend the techniques to related problems.
- What if we consider transpositions of the forms $(i, i+1)$ and $(i, i+2)$?
- How about using transpositions $(i, i+1), (i, i+2), (i, i+3)$, etc.?

An extreme case: Using all (i, j) with $1 \leq j \leq n$

Decompose σ as product of k disjoint cycles (including fixed points).

Then σ is a product of $n - k$ transpositions.

A variation

- A general principle in study and research:
  If you have solved a problem, extend the techniques to related problems.
- What if we consider transpositions of the forms $(i, i+1)$ and $(i, i+2)$?
- How about using transpositions $(i, i+1), (i, i+2), (i, i+3)$, etc.?

An extreme case: Using all (i, j) with $1 \leq j \leq n$

Decompose σ as product of k disjoint cycles (including fixed points).

Then σ is a product of $n - k$ transpositions.

So, the worst case requires $n - 1$ steps.

A variation

- A general principle in study and research:
  If you have solved a problem, extend the techniques to related problems.
- What if we consider transpositions of the forms $(i, i+1)$ and $(i, i+2)$?
- How about using transpositions $(i, i+1), (i, i+2), (i, i+3)$, etc.?

An extreme case: Using all (i, j) with $1 \leq j \leq n$

Decompose σ as product of k disjoint cycles (including fixed points).

Then σ is a product of $n - k$ transpositions.

So, the worst case requires $n - 1$ steps.

Example. $\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 3 & 4 & 5 & 9 & 6 & 7 & 1 & 8 & 2 \end{pmatrix} = (1,3,5,6,7)(2,4,9)(8).$

A variation

- A general principle in study and research:
  If you have solved a problem, extend the techniques to related problems.
- What if we consider transpositions of the forms $(i, i+1)$ and $(i, i+2)$?
- How about using transpositions $(i, i+1), (i, i+2), (i, i+3)$, etc.?

An extreme case: Using all (i, j) with $1 \leq j \leq n$

Decompose σ as product of k disjoint cycles (including fixed points).

Then σ is a product of $n-k$ transpositions.

So, the worst case requires $n-1$ steps.

Example. $\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 3 & 4 & 5 & 9 & 6 & 7 & 1 & 8 & 2 \end{pmatrix} = (1,3,5,6,7)(2,4,9)(8)$.

Then $\sigma = (1,7)(1,6)(1,5)(1,3)(2,9)(2,4)$.

Open problems

Let $1 \leq m < n$, and let G_m be the set of transpositions of the form $(i, i+\ell)$ with $1 \leq \ell \leq m$.

Open problems

Let $1 \leq m < n$, and let G_m be the set of transpositions of the form $(i, i + \ell)$ with $1 \leq \ell \leq m$.

- For a given $\sigma \in S_n$, find the smallest r such that σ is the product of r transpositions in G_m.

Open problems

Let $1 \leq m < n$, and let G_m be the set of transpositions of the form $(i, i+\ell)$ with $1 \leq \ell \leq m$.

- For a given $\sigma \in S_n$, find the smallest r such that σ is the product of r transpositions in G_m.
- Determine the optimal (smallest) $r^* = r^*(n, m)$ so that every $\sigma \in S_n$ is a product at most r^* transpositions in G_m.

Partial results of the general problem

We have the following list for $r^*(n,m)$ for S_n and $(i, i+\ell)$ with $\ell \le m$,

$n \backslash m$	1	2	3	4	5	6	7	8	9	10
2	1									
3	3	2								
4	6	4	3							
5	10	5	5	4						
6	15	[7]	6	6	5					
7	21	[10]	8	7	7	6				
8	28	[14]	[10]	9	8	8	7			
9	36	[16]	[11]	10	10	9	9	8		
10	45	[19]	[14]	[12]	11	11	10	10	9	
11	55	24?	18?	15?	13	12	12	11	11	10

where the entries marked by brackets are obtained by computer programming.

Cayley graphs

Cayley graphs

- The Cayley graph of the the set S_n of permutations using special generators.

Cayley graphs

- The Cayley graph of the the set S_n of permutations using special generators.
- It is known that S_n can be generated by $L = (1, 2, \ldots, n)$ and $S = (1, 2)$. Factorize $\sigma \in S_n$ into the product of L and S with the smallest number of terms.

Cayley graphs

- The Cayley graph of the the set S_n of permutations using special generators.
- It is known that S_n can be generated by $L = (1, 2, \ldots, n)$ and $S = (1, 2)$. Factorize $\sigma \in S_n$ into the product of L and S with the smallest number of terms.
- Diameter of the Cayley graph of S_n using L and S:

S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}
1	2	6	11	18	25	35	45	58	71	???

Cayley graphs

- The Cayley graph of the the set S_n of permutations using special generators.
- It is known that S_n can be generated by $L = (1, 2, \ldots, n)$ and $S = (1, 2)$. Factorize $\sigma \in S_n$ into the product of L and S with the smallest number of terms.
- Diameter of the Cayley graph of S_n using L and S:

S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}
1	2	6	11	18	25	35	45	58	71	???

- Diameter of the Cayley graph of S_n using L, S, and $R = L^{-1} = (n, n-1, \ldots, 1)$:

S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
1	2	6	10	15	21	28	36	45	???

Computing

- Every orthogonal (unitary) matrix P can be written as the product of orthogonal matrices of the form

$$\begin{pmatrix} I_{j-1} & & \\ & Q & \\ & & I_{n-j-1} \end{pmatrix}, \quad \text{with } Q \in M_2, \qquad j = 1, \ldots, n-1.$$

Computing

- Every orthogonal (unitary) matrix P can be written as the product of orthogonal matrices of the form

$$\begin{pmatrix} I_{j-1} & & \\ & Q & \\ & & I_{n-j-1} \end{pmatrix}, \quad \text{with } Q \in M_2, \qquad j = 1, \ldots, n-1.$$

For $n = 4$, we need

$$\begin{pmatrix} * & * & 0 & 0 \\ * & * & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & * & * & 0 \\ 0 & * & * & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & * & * \\ 0 & 0 & * & * \end{pmatrix}.$$

Other problems

- How about the problem associated with the Rubik's cube?

Other problems

- How about the problem associated with the Rubik's cube? Find simple moves to restore the Rubik's cube.

Other problems

- How about the problem associated with the Rubik's cube? Find simple moves to restore the Rubik's cube.
- The study of genomics and mutations,

Other problems

- How about the problem associated with the Rubik's cube?
  Find simple moves to restore the Rubik's cube.
- The study of genomics and mutations, i.e.,
  the change of genetic sequences $x_1x_2x_3\cdots$, with $x_i \in \{A, U, G, C\}$.

Other problems

- How about the problem associated with the Rubik's cube?
  Find simple moves to restore the Rubik's cube.
- The study of genomics and mutations, i.e.,
  the change of genetic sequences $x_1x_2x_3\cdots$, with $x_i \in \{A, U, G, C\}$.
- More on Quantum computing.
  It is of interest to decompose certain quantum gates into simpler quantum gates (CNOT gates).

We can do some more research on the topic!

We can do some more research on the topic!

Thank you for your attention!