효율적인 라틴 스퀘어 동위성 결정 알고리즘 설계

TOPIC : Combinatorics, Algorithms

강원과학고등학교 김준혁

TABLE OF CONTENTS

INTRODUCTION

INTRODUCTION

? Background

1. 두 라틴 스퀘어 L, L'에 대하여 서로 동위적인지 결정하는 문제
2. $O(N^{\log_2 N})$ 알고리즘은 Miller, Gray L. (1977)에 의해서 제시됨.
3. $O(N^3)$ 알고리즘은 Grošek, Otokar (2010)에 의해서 제시됨.

INTRODUCTION

? Goal

본 연구에서는 이와 같은 배경에 이어
더욱 효율적이고 간단한 알고리즘 제시를
목표로 함.

INTRODUCTION

1 Latin square

1 N차 라틴 스퀘어는 알파벳 집합 Σ의 원소로 구성된 $N \times N$ 배열이며, 다음 조건을 만족한다.

- 임의의 행에 대하여 그 행에는 집합 Σ의 원소가 모두 포함되어있다.
- 임의의 열에 대하여 그 열에는 집합 Σ의 원소가 모두 포함되어있다.

A	B	C
C	A	B
B	C	A

INTRODUCTION

1 Latin square

2 라틴 스퀘어에서 정의된 연산.

행 바꾸기 연산 ($\boldsymbol{R_{ij}}$) – 주어진 행 i를 임의의 행 j와 바꾼다.

열 바꾸기 연산 ($\boldsymbol{C_{ij}}$) – 주어진 열 i를 임의의 열 j와 바꾼다.

기호 치환 연산 ($\boldsymbol{S_{s_i s_j}}$) – 주어진 기호 $s_i \in \Sigma$를 $s_j \in \Sigma$로 치환한다.

A	B	C
C	A	B
B	C	A

INTRODUCTION

1 Latin square

3 동위성

이와 같은 라틴 스퀘어의 연산들을 0번 이상 사용하여
다른 임의의 라틴 스퀘어와 같게 만들 수 있는가?

A	B	C
C	A	B
B	C	A

INTRODUCTION

2 Quasigroup

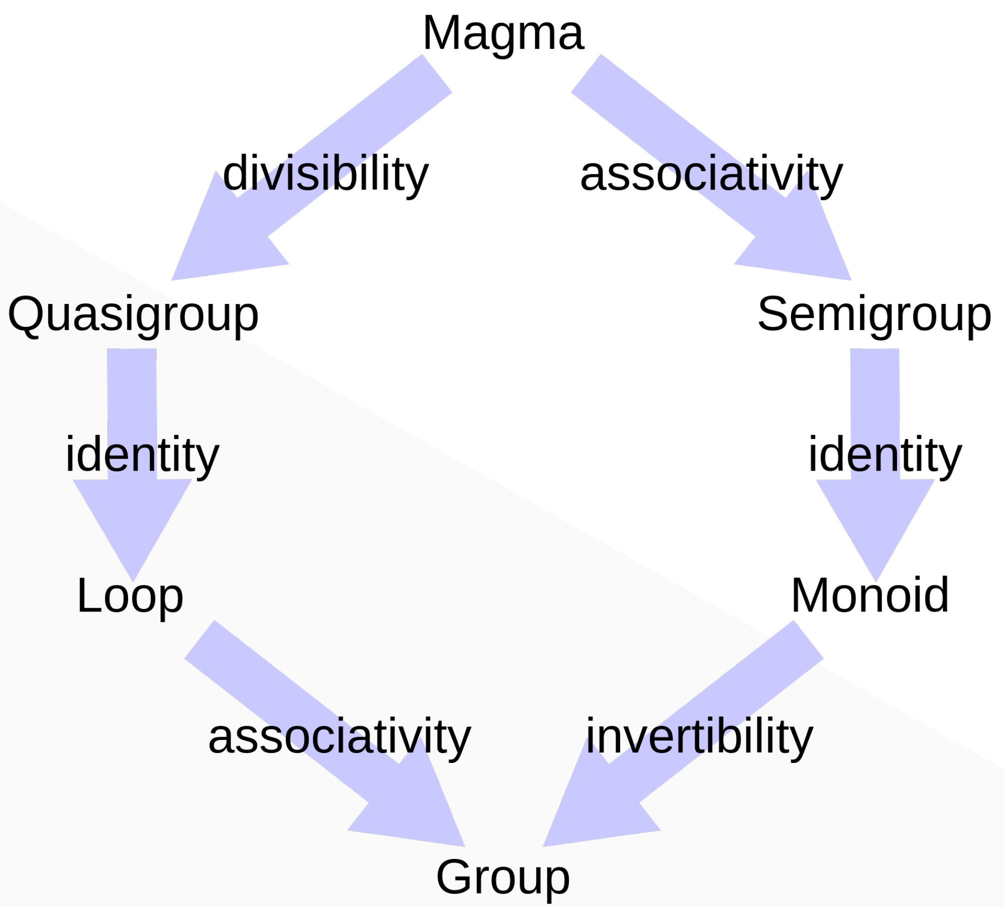
1. 유사군(Quasigroup)은 집합 Σ에 대하여 2-Tuple로 나타내며 임의의 유사군을 Q라고 하고 $*: \Sigma \times \Sigma \rightarrow \Sigma$에 대하여 $\mathrm{Q} = (\Sigma, *)$ 로 표기하며 다음 조건을 만족하는 대수구조이다.

- $\forall a, b \in \Sigma\ \exists!\, x \in \Sigma(\mathrm{a} * \mathrm{b} = \mathrm{x})$
- $\forall a, b \in \Sigma\ \exists!\, x \in \Sigma(a * x = b)$
- $\forall a, b \in \Sigma\ \exists!\, x \in \Sigma(x * a = b)$

INTRODUCTION

2 Quasigroup

$$GROUP \subset QUASIGROUP$$



INTRODUCTION

2 Quasigroup

*	a	b	c
a	a	b	c
b	b	c	a
c	c	a	b

2 Quasigroup의 연산표 예시

$$Q = (\Sigma, *), \Sigma = \{a, b, c\}$$

IDEA: Latin square!

INTRODUCTION

3 Latin square – quasigroup representation

1. N차 라틴 스퀘어는 알파벳 집합 Σ의 원소로 구성된 $N \times N$ 행렬 M, 그리고 유사군 $Q = (\Sigma, *)$에 대하여 $M_{ij} = i * j$ (단, $i, j \in \{1, \dots, N\}$)와 같이 정의되며 3-Tuple (Σ, M, Q)로 표현할 수 있다.

이전에 본 유사군의 연산 표를 상기하자.

INTRODUCTION

4 Property of quasigroup

1 유사군의 동위성

- 유사군 Q, Q'이 서로 동위적(Isotopic)이라는 것은 $Q = (\Sigma, *)$, $Q' = (\Sigma, *')$에 대하여 다음 조건을 만족하는 동위사상(Isotopism) $\theta, \phi, \psi \in Sym(\Sigma)$가 존재하여 $\forall i, j \in \Sigma$에 대하여 각각 $\theta(\phi(i) * \psi(j)) = i *' j$가 성립하는 것이다.

- 유사군 Q, Q'이 서로 동위적(Isotopic)이면 $Q \simeq_t Q'$이라고 표기한다.

INTRODUCTION

4 Property of quasigroup

2 Left translation

- 유사군 $Q = (S,*)$에 대해서 $a \in S$에 대한 left translation은 $L_a : S \to S$ (단, $a \in S$)로 표기되며, $L_a(x) = a * x$ (단, $x \in S$)로 정의된다.
- 유사군 $Q = (S,*)$에 대한 left translation은 L_Q로 표기되며, $L_Q = \{L_a | a \in S\}$로 정의된다.

INTRODUCTION

4 Property of quasigroup

3 Quasicoset

- 군 $G = (S,*)$가 주어졌을때, $H \subset S$에 대한 $a \in S$의 좌유사잉여류(left quasicoset)은 $aH = \{a * h|h \in H\}$로 정의됨.
- 마찬가지로 우유사잉여류(right quasicoset)은 $Ha = \{h * a|h \in H\}$로 정의된다.

INTRODUCTION

5 Operations on the Latin square

1 Background: 이전에 정의한 연산의 정의는 모호함.

행 바꾸기 연산 ($\boldsymbol{R_{ij}}$) – 이 연산을 적용한 결과를 $L' = R_{ij}(L) = (\Sigma, M', Q')$이라고 하자. 이때 다음과 같이 정의할 수 있다. (단, $Q' = (\Sigma, *')$이다)

$$a *' b = \begin{cases} i * b \ (a = j) \\ j * b \ (a = i) \\ a * b \ (Otherwise) \end{cases}$$

5 Operations on the Latin square

열 바꾸기 연산 ($\boldsymbol{C_{ij}}$) – 이 연산을 적용한 결과를 $L' = C_{ij}(L) = (\Sigma, M', Q')$이라고 하자. 이때 다음과 같이 정의할 수 있다. (단, $Q' = (\Sigma, *')$이다)

$$a *' b = \begin{cases} a * i \ (b = j) \\ a * j \ (b = i) \\ a * b \ (Otherwise) \end{cases}$$

5 Operations on the Latin square

기호 바꾸기 연산 ($\boldsymbol{S_{s_i s_j}}$) – 이 연산을 적용한 결과를 $L' = S_{s_i s_j}(L) = (\Sigma, M', Q')$이라고 하자.
이때 다음과 같이 정의할 수 있다. (단, $Q' = (\Sigma, *')$이다)

$$a *' b = \begin{cases} s_i \ (a * b = s_j) \\ s_j \ (a * b = s_i) \\ a * b \ (Otherwise) \end{cases}$$

ALGORITHMS & CORRECTNESS

1 라틴 스퀘어에서 연산의 가환성

정리 2.2.1. 라틴 스퀘어에서 정의된 연산은 가환적이다.

Proof. *생략*

의의: 알고리즘화 할때 중요함

2 유사군 동위성의 동치 조건

정리 2.2.2. 알파벳 집합 Σ에 대하여 유사군 $Q_1 = (\Sigma, *_1)$, $Q_2 = (\Sigma, *_2)$를 고려하자.
여기에서 $Q_1 \simeq_t Q_2$이기 위한 필요충분조건은
$\exists p \in L_{Q_1} \exists q \in L_{Q_2} \exists \theta^{-1} \in (L_{Q_1} p^{-1})(\theta L_{Q_1} p^{-1} = L_{Q_2} q^{-1})$인 것이다.

Proof. 생략

매우 핵심적인 부분

ALGORITHMS & CORRECTNESS

3 알고리즘

Time complexity: $O(N^4), \Omega(N^2)$

INPUT: Latin squares $L_1 = (\Sigma, M_1, Q_1),\ L_2 = (\Sigma, M_2, Q_2)$

OUTPUT: Decision YES – NO

Function ISOTOPY:

Choose an arbitrary $p \in L_{Q_1}$

Evaluate $L_{Q_1} p^{-1}$

For $q \in L_{Q_2}, \theta \in L_{Q_1} p^{-1}$**:**

If $\theta^{-1} L_{Q_1} p^{-1} = L_{Q_2} q^{-1}$**:**

Return YES

Return NO

4 대칭군 연산 자료구조

1. Background: 이전에 제시한 알고리즘에서의 대칭군 연산을 효과적으로 처리하기 위한 자료구조 제시 필요.

- 유한 집합 S에 관한 대칭군의 원소는 함수 $f: S \to S$로 표현할 수 있다.
- 여기에서 이 집합에 단순 순서관계를 부여하면 $f: \{1, \dots, N\} \to \{1, \dots, N\}$로도 표현할 수 있다. 따라서 이 함수의 정의역을 크기가 $|S|$인 배열 A의 인덱스(index)로 가정한다면, $A[n] = f(n)$ (단, $1 \le n \le |S|$)로 대표할 수 있다.

4 대칭군 연산 자료구조

집합 S에 관한 대칭군의 원소 $\sigma \in Sym(S)$를 대표하는 배열을 A_σ라고 하면,
다음과 같은 연산을 정의할 수 있음.

1. $SWAP(i, j)$: A_σ의 두 원소의 위치를 바꾼다. (단, $1 \le i, j \le |S|$), 시간 복잡도: $O(1)$
2. $MULTIEVAL(\tau)$: $A_{\sigma\circ\tau}$를 계산한다. (단, $\tau \in Sym(S)$), 시간 복잡도: $O(|S|)$
3. $EVAL(i)$: $f(i)$를 계산한다. (단, $1 \le i \le |S|$)

CONCLUSION

CONCLUSION

1 연구 요약 및 의의

1. 이 연구에서 라틴 스퀘어의 동위성을 결정하는 최선의 경우 $\Omega(N^2)$, 최악의 경우 $O(N^4)$의 시간 복잡도를 가지는 더욱 간단하고 효과적인 알고리즘을 제시함.
2. 대칭군 연산을 위한 자료구조 모델을 제시함.

CONCLUSION

2 개선 방향

1. 결국 알고리즘에서 문제가 되는 부분은 유사잉여류 연산임.
2. 여기에서 최선의 경우를 개선시키면 평균적인 시간 복잡도를 개선시킬 수 있음.
3. 이미 연산을 한 결과를 Trie 자료구조를 통해서 재활용 가능 (본문 참고)

THANKS.

References

1. https://en.wikipedia.org/wiki/Quasigroup#/media/File:Magma_to_group2.svg
2. D. Guichard, "An Introduction to Combinatorics and Graph Theory," 30 1 2020. [Online]. Available: https://www.whitman.edu/mathematics/cgt_online/cgt.pdf. [Accessed 31 8 2020].
3. G. L. Miller, "On the $n^{\log_2{n}}$ Isomorphism Technique," *UR Research,* pp. 1-2, 1977.
4. O. Grošek and M. Sy´s, "Isotopy of latin squares in cryptography," *Tatra Mountains Mathematical Publications,* vol. 45, no. 1, pp. 27-36, 2010.

본 발표 자료 (PT)는 다음 링크에서 확인하실 수 있습니다.

https://github.com/ANEP-Research/Isotopy_Latin_Square