情報保護學會論文誌
第16卷 第2號, 2006. 4

선형계를 위한 실용적인 프라이버시 보존형 다자간 계산 프로토콜

강 주 성[1†‡], 이 옥 연[1], 홍 도 원[2]
[1]국민대학교 수학과, [2]한국전자통신연구원

A Practical Privacy-Preserving Multi-Party Computation Protocol for Solving Linear Systems

Ju-Sung Kang[1†‡], Ok-Yeon Yi[1], Dowon Hong[2]
[1]Department of Mathematics, Kookmin University
[2]Information Security Research Division, ETRI

요 약

여러 개체가 각자의 정보를 제공하여 이를 바탕으로 정보 제공자의 프라이버시를 보존하면서 공통의 유익한 정보를 얻고자 하는 다자간 협력 계산 프로토콜에 대해서 논한다. 금융, 제조업, 통신 분야 등에서 널리 응용되는 선형계(linear system)의 일반해(general solution)와 최소제곱해(least-square solution)를 구하는 문제에서 프라이버시를 보존하는 실용적인 다자간(multi-party) 협력 계산 프로토콜을 제안한다. 본 논문에 제안된 프로토콜은 기존의 양자간(two-party) 협력 계산 방식을 확장한 새로운 것으로 효율성 측면에서 우수한 실용적인 다자간 계산 프로토콜이다.

ABSTRACT

We consider a privacy-preserving cooperative computation protocol evaluating a beneficial function of all participants' secret inputs, such that each party finally holds a share of the function output. We propose a practical privacy-preserving cooperative computation protocol for solving the linear system of equations problem and the linear least-squares problem. Solutions to these problems are widely used in many areas such as banking, manufacturing, and telecommunications. Our multi-party protocol is an efficiently extended version of the previous two-party model.

Keywords : *Secure multi-party computation, Linear system, Privacy*

Ⅰ. 서 론

안전성을 가진 다자간 계산(secure multi-party computation, SMC) 문제는 기본적으로 여러 개체가 각자의 프라이버시 노출 없이 서로 협력하여 어떤 계산을 수행하기 원하는 것이다. 계산에 참여하는 개체는 계산 수행에 필요한 개인 정보를 입력(private input)으로 제공해야 하고, 이 개인 정보는 다른 개체들 또는 제 삼자에게 노출되지 않아야 한다. SMC 문제에서 달성하고자 하는 이와 같은 기능은 제 삼의 신뢰 기관(TTP)을 가정하거나 서비스 제공자를 전적으로 신뢰함으로써 간단히 해결 가능하다. 그러나 현재의 인터넷을 비롯한 컴퓨터 네트워크 환경은 대단히 역동적이어서 악의적 환경을 배제할 수 없기 때문에 TTP를 가정하지 않는 협력 계산 프로토콜이

접수일: 2005년 9월 1일; 채택일: 2006년 3월 22일
† 주저자, ‡ 교신저자 : jskang@kookmin.ac.kr

필요하다. SMC 문제는 TTP를 가정하지 않고 프라이버시를 보호하는 다자간 협력 계산 방식을 개발하기 위해서 정의된 것이라 할 수 있다.

SMC 문제는 Yao[1]에 의해서 처음 소개된 이래로 Goldreich-Micali-Wigderson[2]을 비롯한 여러 학자들에 의해서 연구되었으며, Goldreich[3]는 SMC 문제에 대한 최근까지의 이론적 결과들을 종합적으로 정리하여 기술go 놓았다. 지금까지의 이론적 연구 결과에 의하면, 비트 회로 계산(circuit evaluation) 관점에서 SMC 문제는 모두 해결 가능하다. 즉, 일방향 트랩도어 순열(trapdoor permutation)이 존재하면, OT(oblivious transfer) 프로토콜의 구축이 가능하고, 이로부터 안전한 양자간(2-party) 계산 방식을 만들 수 있으며, 비트별 연산의 특성 때문에 양자간 계산 방식을 다자간(multi-party) 계산 방식으로 확장 가능하다는 것이다. 모든 계산 문제들은 비트 회로의 조합으로 표현 가능하기 때문에 이론적으로는 일반적인 의미의 SMC 문제가 모두 해결 가능하다는 의미이다. 그러나 SMC 문제를 여러 응용 시스템에 적용하기 위해서 장애 요인이 되는 것은 계산의 효율성(efficiency) 문제이다. 실제로 [3]에서 Goldreich는 이 점을 언급하면서 효율성 문제 때문에 특정한 경우마다 그 응용에 적합한 효율적인 SMC 방식의 개발이 필요함을 지적하였다.

SMC 관련 연구가 주로 이론적인 면에 치우친 경향을 보였지만, 2000년대에 들어서 그 필요성이 대두되어 SMC 응용문제와 관련된 연구 결과들이 발표되기 시작하였다. SMC 응용 관련 연구 결과로는 프라이버시를 보존하는 협력적 과학 계산[4], 프라이버시 보존형 데이터베이스 질의(database query)[5], 프라이버시 보존형 데이터 마이닝(data mining)[6], 프라이버시 보존형 통계적 분석(statistical analysis)[7] 등을 들 수 있다. 하지만 최근에 발표된 결과인 [7]을 제외하고는 모두 양자간(2-party) 계산 모델이다. 이는 실제 응용문제에서 사용하는 연산이 비트별 연산과는 상이한 특성을 보이므로 [3]에 제시된 것과 같은 다자간(multi-party) 계산으로의 일반적인 확장 방법이 효율적인 관점에서 자연스럽게 이루어지지 못하기 때문이다.

본 논문에서는 [4]에서 양자간 협력 계산 모델로 제시된 선형계(linear system)의 프라이버시를 보존하는 일반해(general solution)와 최소제곱해(least-square solution)의 계산 문제에 대한 다자간(m-party) 협력 계산 버전을 제안한다. [4]에서 제시된 양자간 계산 방식은 반복적인 적용에 의해서 다자간 계산 방식으로 확장할 경우 대단히 비효율적임을 지적하며, [7]에서 제안된 다자간 계산 방식을 행렬 연산에 적용한 프로토콜을 기반으로 하여 선형계의 일반해와 최소제곱해를 계산하는 새로운 효율성 높은 다자간 계산 프로토콜을 제안하고 그 안전성을 논한다. 한편, Cramer와 Damgard[8]는 상수 라운드(constant-round)의 검증가능 비밀 공유(verifiable secret sharing, VSS) 방식의 존재성과 곱셈 및 덧셈 연산을 계산하는 상수 라운드 프로토콜을 이용할 수 있다는 가정 하에서 여러 가지 선형대수 관련 문제를 안전하게 계산할 수 있는 상수 라운드 계산 프로토콜을 제안하였다. 여기에 제안된 프로토콜은 이러한 가정들이 필요 없는 직접적이고 실질적으로 응용 가능한 것이라 할 수 있다.

II. SMC 문제 관련 연구 및 활용 분야

최근까지 SMC 관련 연구는 제한적인 영역의 연산에 대해서만 연구되어 왔으나, 다양한 분야의 협력 계산이 요구됨으로써 이에 따른 여러 계산 영역에서의 SMC 문제들이 논의되기 시작하였다. Du와 Atallah[9]는 이러한 다양한 영역의 SMC 문제들을 정의하고 이 문제들을 개발할 수 있는 프레임워크와 활용 분야를 제시하였다. 이 절에서는 이들의 연구 결과를 간략히 소개하고, 학계의 SMC 관련 최신 기술 동향을 살펴보기로 한다.

보통의 계산 문제를 SMC 문제로 전환시키는 프레임워크는 다음과 같다. 먼저 구별되는 입력의 개수에 따라서 다중 입력 계산(multi-input computation) 모델과 단일 입력 계산(single-input computation) 모델로 분류한다. 다중 입력 모델은 일반적으로 두 가지 구별되는 입력들을 가진다. 클라이언트/서버 계산이 그 예가 된다. 단일 입력 계산 모델은 하나의 입력 또는 하나의 입력 집합을 가진다. 데이터 마이닝과 통계적 분석에서 모든 입력은 여러 개의 데이터 항목들로 구성된다고 할지라도 일반적으로는 하나의 데이터 집합으로부터 나온다.

두 가지 모델에서 SMC 모델로 전환시키는 방법의 기본적인 요소는 참여하는 개체들의 입력이 비밀로 간주되어 다른 어떤 개체들에게도 개인정보가 노출되지 않도록 하는 것이다. 특별한 경우에는 계산의 결과도 비밀로 간주될 수 있으며, 몇몇 개체들은 계

산 결과를 얻지 못할 수도 있다. 다중 입력 계산 모델에서 SMC 모델로 전환시키는 방법은 매우 자연스런 것이다. 이 모델에서는 입력에 참여하는 개체가 적어도 둘 이상이기 때문에 각 개체들의 프라이버시가 유지되기만 하면 SMC 모델로 전환되는 것이다.

단일 입력 계산 모델에서는 하나의 입력이 존재하기 때문에 다중 입력 계산 모델과 같이 자연스런 전환이 가능하지 않다. 그러므로 단일 입력 계산 모델을 다중 입력 계산 모델로 간주해서 볼 수 있는 방법이 요구된다. 단일 입력 집합을 D라 하고, D가 서로 교집합이 없는(disjoint) D_1과 D_2로 분리된다면 다중 입력 계산 모델로 볼 수 있는 것이다. D를 분리하는 방법에는 여러 가지가 있을 수 있으며, 각 분리 방법에 따라서 다른 SMC 문제가 유도된다. [9]에서 저자들은 D가 분리되는 방법에 따라서 동종(homogeneous) 전환(transformation)과 이종(heterogeneous) 전환의 두 가지로 분류하였다. 이렇게 분리된 후에 원래의 계산 문제는 D_1과 D_2의 노출이 없이 두 집합의 합집합인 D위에서 계산하고자 하는 SMC 모델로 전환될 수 있다.

이와 같은 프레임워크 하에서 Du와 Atallah 등은 다양한 SMC 문제를 개발하고 활용 분야를 개척하였다. 대표적인 것이 프라이버시를 보존하는 협력적 과학 계산[4], 프라이버시 보존형 데이터베이스 질의(database query)[5], 프라이버시 보존형 데이터 마이닝(data mining)[6], 프라이버시 보존형 통계적 분석(statistical analysis)[7] 등이다.

한편, SMC 분야의 이론적인 측면에서도 최근까지 꾸준히 수준 높은 연구 결과들이 발표되고 있다. 최신 연구 결과 중에서 가장 주목할만한 성과는 2005년 크립토 학회에서 발표된 Kissner와 Song[10]의 논문이다. 이들은 프라이버시 보존형 다중집합 연산을 효율적으로 수행할 수 있는 획기적인 기술을 제안하였다. 2004년도에 발표된 Freedman 등[11]의 연구 결과를 확장한 것으로 다항식의 성질을 충분히 이용하여 다중집합 연산에서 프라이버시를 효율적으로 보호할 수 있는 프로토콜을 발표했던 것이다. 그러나 이들 결과는 비트 회로 관점의 계산을 사용하는 일반적인 SMC에 비해서 효율성이 개선되었다는 것이지 현실적으로 사용가능한 방법으로 볼 수는 없다. Kissner와 Song이 제시한 프로토콜이 성립하기 위해서는 실제로 현재 효율성 문제 때문에 사용이 권장되지 않고 있는 덧셈적으로 동형인(additively homomorphic) 공개키 암호 시스템의 이용이 필수적이다. 그러므로 효율적인 SMC 프로토콜이 반드시 실용적인 프로토콜이라고 볼 수는 없다.

지금까지의 SMC 관련 연구 동향으로 비추어볼 때, 안전성을 면밀히 분석하는 이론적인 영역의 연구와 효율성 측면을 강조한 실용적인 영역의 연구가 상호 보완적으로 이루어지고 있다. 즉, 안전성을 강조한 영역의 이론적 연구는 효율성을 개선시키는 방향으로, 그리고 효율성을 강조한 실용적 연구는 안전성 분석을 합리적으로 실시하고자 하는 방향으로 전개되고 있는 것이다. 본 논문에서 제안하는 프로토콜들은 후자에 속한다고 할 수 있다. 제안하는 실용적 방식에서는 이론적인 결과들에서 흔히 사용되는 공개키 암호 시스템이나 OT 프로토콜 등의 사용을 배제하고, 단순한 랜덤화(randomization) 기법에 의해서 프라이버시를 보호하고자 하였다. 그러므로 안전성 분석도 이론적인 프로토콜들에서와 같은 완벽한 증명 방법은 적용할 수 없으며, 공격 성공 확률과 계산량 관점의 분석 방법을 적용하였다.

Ⅲ. 선형계의 일반해와 최소제곱해에 대한 협력 계산 모델

여러 개체들이 자신들만의 중요 정보인 개인정보(private information)를 노출시키지 않은 상태에서 서로 공통으로 원하는 정보를 얻고자 하는 협력 계산 프로토콜을 고려하자. 프로토콜에 참여하는 각 개체의 입력을 바탕으로 모두에게 유익한 정보를 출력하고자 하는 경우에 각 입력의 형태는 각자의 요구조건을 정보로 표현한 것이다. 이때, 이러한 요구조건이 몇 개의 일차방정식들로 이루어진 경우의 협력 계산 방법을 생각해 보자. 프로토콜에 참여하는 m명의 참여자(player)들을 $P_1, P_2, \cdots, P_m$이라 놓는다. 일차방정식에 나타나는 n개의 미지수는 $x_1, \cdots, x_n$으로 놓고, 각 참여자 $P_j (1 \le j \le m)$는 k_j개의 방정식들을 요구조건으로 가지고 있다고 한다. 문제를 단순화하기 위해서 $k_1 + \cdots + k_m = n$임을 가정하고 방정식들이 일차종속(linearly dependent)인 경우도 포함한다고 하자.

$P_1, P_2, \cdots, P_m$은 방정식의 개수와 미지수의 개수가 각각 n개인 선형계(linear system)의 해(solution)를 얻고자 한다. 각 참여자들이 자신의 요구조건을 표현한 일차방정식들을 모두 공개적으

로 제공한다면 가우스 소거법(Gaussian elimination) 등과 같은 해법에 의해서 간단히 선형계의 해를 구할 수 있다. 그러나 여기에서 우리가 고려하는 것은 각자의 요구조건들은 개인정보로 간주하여 다른 참여자들에게 노출되지 않도록 하는 협력 계산 방법이다.

3.1. 선형계의 협력 계산 모델

A를 $n\times n$ 행렬이라 하고, b를 n차원 열벡터라고 할 때, 방정식과 미지수의 개수가 각각 n개인 선형계는 다음과 같이 표현된다.

$$A\,x=b,\quad x=(x_1,\cdots,x_n)$$

이 선형계는 $P_j(1\le j\le m)$가 k_j개의 방정식들을 제공하여 얻어진 것이기 때문에 일반적으로 다음과 같이 표현할 수 있다.

$$(A_1+\cdots+A_m)x=(b_1+\cdots+b_m)$$

여기에서 $A_j,b_j(1\le j\le m)$는 P_j의 요구조건을 나타내는 $n\times n$ 행렬과 n차원 열벡터로 자신들의 개인정보인 k_j개 일차방정식들에 해당하는 부분을 제외하고는 성분이 모두 0이다. 프라이버시를 보존하는 협력 계산 프로토콜은 각 $P_j(1\le j\le m)$가 A_j와 b_j를 노출시키지 않은 상태에서 선형계 $Ax=b$ 의 유일한 해(unique solution) 또는 일반해(general solution)을 구하기 위한 것이다.

3.2. 최소제곱해를 위한 협력 계산 모델

여러 가지 응용에서 선형계 $Ax=b$ 는 이론적으로는 해를 가져야 하지만 A 또는 b 성분의 측정 오차로 인해서 해가 존재하지 않는 경우가 있다. 이러한 경우는 방정식의 개수가 미지수의 개수 보다 많은 경우로 생각할 수 있다. 즉, A가 $l\times n$ 행렬이고 x 는 n 차원 벡터, b는 l차원 벡터일 때, $l>n$ 인 경우이며, 이런 경우에 일반적인 풀이 과정은 $\|b-Ax\|$를 최소화함으로써 해에 가능한 한 가까운 벡터들을 찾는 것이다. 여기에서 l차원 벡터 a에 대하여

$$\|a\|=\sqrt{\sum_{j=1}^{l}a_j^2}$$

이므로 $\|b-Ax\|$를 최소로 하는 벡터 x를 최소제곱해(least-square solution)라고 부른다. 최소제곱해를 구하는 일반적인 방법은 $Ax=b$를 정방행렬이 포함된 선형계인 정규계(normal system)로 표현하여 정규계의 해를 구하는 방법을 적용하는 것인데 이에 대응하는 정규계는 다음과 같다[12].

$$A^TAx=A^Tb$$

여기에서 A^TA는 $n\times n$ 행렬이 되고, A^Tb는 n 차원 벡터가 되기 때문에 정방행렬이 포함된 선형계가 되는 것이다. 이 정규계의 협력 계산 모델은 $A=A_1+\cdots+A_m$ 일 때, 다음과 같이 표현된다.

$$\left(\sum_{j=1}^{m}\sum_{i=1}^{m}A_j^TA_i\right)x=\left(\sum_{j=1}^{m}\sum_{i=1}^{m}A_j^Tb_i\right)$$

이 때, $A_j,b_j(1\le j\le m)$는 P_j의 요구조건을 나타내는 $l\times n$ 행렬과 l차원 열벡터이므로 노출되지 않아야 한다.

위와 같은 선형계로 모델링 되는 문제들은 산업공학 또는 경영학 등의 분야에서 흔히 볼 수 있는 것으로 금융, 운송, 에너지, 통신 등과 관련된 많은 응용분야가 있다. 최소제곱해를 구하는 문제는 수학적 모델링이나 통계학의 회귀분석(regression) 등에 많이 응용된다. 예를 들면, 몇몇 금융 기관들이 서로의 이익을 위한 어떤 프로젝트를 공동으로 수행하고자 계획하는 경우를 생각해볼 수 있다. 금융 기관들은 내부 사정에 따른 각자 자신들만의 요구조건(requirements)을 가지고 있을 것이다. 이러한 요구조건들에는 특정 상품의 가격, 이자율과 인플레이션 비율, 경제적 통계량, 고객의 포트폴리오 등이 포함될 것이며, 이와 같은 중요 정보들은 어떤 금융 기관이든지 제 삼의 신뢰기관에게 조차도 노출을 원하지 않을 것이다. 각 금융 기관들의 중요 정보 노출 없이 공동 프로젝트 수행에 필요한 공동의 정보를 얻기 위해서 협력할 수 있는 방안을 제공해 줄 수 있는 방법이 바로 SMC 기술인 것이다.

Ⅳ. 프라이버시의 이론적 정의

본 논문에서 고려하는 안전성 모델은 준정직한(semi-honest) 참여자들을 가정하는 일반적인 다자간 계산 방식이다. Goldreich[3]에 의하면 좀 더

안전성 높은 모델인 악의적(malicious) 참여자 모델은 매크로(macro)라는 개념을 도입하여 준정직한 참여자 모델로 강제시킬 수 있다. 이러한 이유 때문에 대부분의 SMC 관련 문제는 준정직 모델(semi-honest model) 하에서 안전성을 분석하는 것이 일반적이다. 준정직 모델 또는 "정직하지만 의심스러운(honest but curious)" 모델로 불리는 환경 하에서 프라이버시의 정의는 다음과 같다.

정의 1. (준정직 모델에서의 프라이버시[3]) 프로토콜에 참여하는 m명의 참여자(party)들을 $P_1, P_2, \cdots, P_m$이라 하고, m차 범함수(functionality)를 $f:(0,1^*)^m \rightarrow (0,1^*)^m$이라 하며, $f_i(x_1,\cdots,x_m)$를 $f(x_1,\cdots,x_m)$의 i번째 성분이라 하자. $I=i_1,\cdots,i_t \subset 1,\cdots,m$에 대하여, $f_I(x_1,\cdots,x_m)$는 부분수열 $f_{i_1},\cdots,f_{i_t}$로 놓는다. Π는 범함수 f를 계산하는 m자간(m-party) 프로토콜이라 하고, Π가 $x=(x_1,\cdots,x_m)$에 대해서 실행되는 동안 $P_{i_1},\cdots,P_{i_t}$의 관찰(view)은 $VIEW_I^{\Pi}(x)$로 표시한다. 임의의 I에 대해서 다음을 만족하는 다항식 시간 알고리즘 S가 존재할 때, Π는 범함수 f의 **프라이버시 보존형 계산 프로토콜**이라 불린다.

$$S(I,(x_{i_1},\cdots,x_{i_t}),f_I(x))_{x\in(0,1^*)^m} \cong VIEW_I^{\Pi}(x)_{x\in(0,1^*)^m}$$

여기에서 기호 "$\cong$"는 두 알고리즘이 계산량적으로 동치임을 의미한다.

직관적으로 살펴보면, 준정직 모델에서는 모든 참여자들이 프로토콜을 준수하지만, 프로토콜 수행 중에 얻은 데이터를 기반으로 독자적으로 또는 다른 참여자들과 연합함으로써 더 많은 정보를 얻으려고 노력할 수 있다. 한편, Atallah 등[7]은 다자간 협력 계산의 안전성을 논하기 위해서 악의적 행동 양식(malicious behavior)을 보이는 프로토콜 참여자들을 다음과 같은 세 가지 유형으로 구분하였다.

1. 준정직 참여자(semi-honest players) : 프로토콜을 준수하지만 프로토콜 수행 중에 얻은 데이터를 기반으로 더 많은 정보를 발견하기 위한 시도를 하는 참여자들을 말한다.
2. 공모하는 참여자(colluding players) : 준정직 참여자의 행동 양식을 보이지만, 다른 참여자들의 개인 정보를 얻기 위해서 서로 공모하는 참여자들을 일컫는다.
3. 악의적인 참여자(malicious players) : 다양한 형태의 악의적 행동 양식을 보이는 참여자를 말한다. 서로 공모하거나 프로토콜 중간에 개입하여 데이터를 수정하거나 틀린 정보를 삽입할 수 있으며, 프로토콜 중간에 탈퇴하거나 옳지 못한 계산을 의도적으로 수행할 수도 있다.

여기에서 고려하는 다자간 협력 계산 프로토콜은 참여자들 모두에게 유익한 정보를 공동으로 얻기 위한 것이 일반적인 경우이므로 악의적인 참여자는 가정하지 않기로 한다. 이는 이론적으로 악의적 모델을 준정직 모델로 강제시킬 수 있다는 사실을 상기할 때, 합리적인 가정으로 생각할 수 있다. 그리고 참여자들이 사업상 경쟁 관계에 있는 경우에도 적용 가능한 협력 계산 프로토콜을 고려할 것이므로 공모하는 참여자까지는 가정하는 것이 타당하다고 볼 수 있다.

V. 선형계를 위한 프라이버시 보존형 계산 프로토콜

5.1. 선형계를 위한 양자간 계산 프로토콜

Du와 Atallah[4]는 선형계의 일반해와 최소제곱해를 구하는 문제에 대한 프라이버시 보존형 양자간 협력 계산 프로토콜을 제안하였다. 이들이 제안한 방법은 SMC의 양자간 계산 모델의 기초를 형성하고 있는 OT_1^N(1-out-of-N Oblivious Transfer)[13] 프로토콜을 적용한 것으로 일차연립방정식

$$(A_1+A_2)x=(b_1+b_2)$$

의 해(solution)와 정칙 행렬 T와 S를 이용한 다음 일차연립방정식

$$T(A_1+A_2)SS^{-1}x=T(b_1+b_2)$$

의 해가 동일하다는 기본적 사실을 이용하고 있다.

OT_1^N 프로토콜은 송신자(sender)와 수신자(re-

ceiver) 간의 비대칭 양자간 프로토콜이다. 프로토콜을 시작할 때, 송신자는 N개 항목의 입력 값을 가지고 있으며, 프로토콜 종료 시에 수신자는 N개 중에서 자신이 원하는 항목의 값을 얻게 된다. OT 프로토콜의 안전성에서 고려하는 바는 송신자는 수신자가 어떤 항목을 선택했는지를 몰라야 하고, 수신자는 자신이 얻은 값 이외의 입력 값에 대해서 아는 정보가 없어야 한다는 것이다.

첫 번째 개체 P_1이 행렬 $T(A_1+A_2)S$와 열벡터 $T(b_1+b_2)$를 알고 있으면, 이것으로 구성된 선형계의 일반해 $\hat{x}$을 구할 수 있고, $x=Q\hat{x}$의 관계식으로부터 원하는 해를 구할 수 있다. P_1은 두 번째 개체 P_2의 개인 정보인 A_2와 b_2를 알지 못해야 하는데, P_2가 정칙 행렬 T와 S를 비밀리에 선택하고, P_1을 수신자(receiver), P_2를 송신자(sender)로 간주하여 세부 프로토콜에 적절한 OT_1^N 프로토콜을 수행함으로써 이러한 프라이버시는 유지될 수 있다. Du와 Atallah는 선형계의 일반해를 구하는 프라이버시 보존형 양자간 협력 계산 프로토콜을 이용하여 선형계의 최소제곱해를 구하는 양자간 계산 방식도 제시하였다[4]. 이들은 최소제곱해 구하는 문제를 선형계의 일반해 구하는 정규방정식으로 전환한 다음, 행렬곱(matrix product) 프로토콜을 핵심 요소로 사용하여 최소제곱해를 구할 수 있는 프라이버시 보존형 양자간 프로토콜을 제안한 것이다.

한편, 이 양자간 계산 프로토콜을 반복적으로 적절히 수행함으로써 다자간 계산 방식으로 자연스런 확장이 가능하다. 임의의 개체 $P_j(1 \le j \le m)$는 자신 이외의 다른 $m-1$명의 개체들과 위의 양자간 계산 프로토콜을 수행함으로써 각각의 연립방정식에 대응하는 $m-1$개의 해 공간(solution space)을 얻을 수 있고, 이러한 해 공간들의 공통부분(intersection)이 구하고자 하는 해가 된다. 그런데 이러한 확장 방법은 상당히 비효율적이다. 프라이버시 보존을 위해서 양자간 계산 프로토콜을 수행할 때마다 계산량이 많은 OT_1^N 프로토콜이 수행되어야 한다. 한 번의 다자간 계산이 완성되기 위해서 전체적으로 $m(m-1)/2$번의 양자간 계산이 필요하며, 각 개체들마다 연립방정식을 풀어야 하는 계산적 부담을 갖는다. 본 논문에서는 OT_1^N 프로토콜을 사용하지 않고, 프라이버시를 보존할 수 있는 효율적이고 실용적인 다자간 계산 방식을 제안한다. 여기에서 제안하는 프로토콜은 통신복잡도(communication complexity)와 각 개체의 계산 복잡도(computational complexity) 측면에서 양자간 프로토콜을 사용한 확장 방법보다 매우 효율적인 것이지만 안전성이 정보이론적 관점에서 완벽하다고 말할 수는 없는 것이다. 즉, 안전성과 효율성의 적절한 조화를 고려하고자 할 때, 본 논문의 결과는 효율성 측면이 강조된 방식으로 볼 수 있는 것이다.

5.2. 선형계를 위한 다자간 계산 프로토콜

본 소절에서는 양자간 계산 방식의 단순한 확장으로 구성된 다자간 계산 방식이 아닌 새로운 프로토콜을 제안한다. Du와 Atallah의 양자간 계산에서 프라이버시 보존을 위한 핵심 기술로 사용한 OT_1^N을 사용하지 않는 효율적인 다자간 계산 방식을 기술한다. 각 개체의 계산 복잡도를 최소화하기 위해서 신뢰성 없는 제 삼의 개체(untrusted third party)를 가정한다. 이 제 삼의 개체는 [7]에서 저자들이 어슐라(Ursula)라고 명명한 개체의 역할과 유사한 것으로 신뢰성은 없고, 단순히 필요한 계산만을 대행해 주는 서버로 간주되는 것이다.

프라이버시를 보존하는 선형계의 다자간 협력 계산 프로토콜을 도출해 내기 위하여, 먼저 고려하고자 하는 문제를 엄밀히 정의하기로 하자. 먼저 우리가 여기에서 다루는 두 가지 문제를 다음과 같이 명명하기로 한다. PCMC-LSE(Privacy-preserving Cooperative Multi-party Computation for Linear System of Equations)는 선형계의 일반해를 구하기 위한 것이고, PCMC-LSS(Least Square Solution)는 최소제곱해를 구하기 위한 것이다.

문제 1. (PCMC-LSE) 프로토콜에 참여하는 m명의 참여자(party)들을 $P_1, P_2, \cdots, P_m$이라 하고, 각 참여자 $P_j(1 \le j \le m)$는 $n \times n$ 행렬 A_j와 n차원 열벡터 b_j를 개인 정보(private information)로 소유한다고 하자. 각 개체의 개인 정보를 다른 개체들에게 노출시키지 않은 상태에서 다음의 선형계 $(A_1+\cdots+A_m)x=(b_1+\cdots+b_m)$의 해(solution)를 모든 참여자들이 공유하고자 하는 것이 PCMC-LSE 문제이다.

문제 2. (PCMC-LSS) 프로토콜에 참여하는 m명의 참여자(party)들을 $P_1, P_2, \cdots, P_m$이라 하고, 각 참여자 $P_j (1 \le j \le m)$는 $l \times n$ 행렬($l > n$) A_j와 l 차원 열벡터 b_j를 개인 정보(private information)로 소유한다고 하자. 각 개체의 개인 정보를 다른 개체들에게 노출시키지 않은 상태에서 다음의 선형계 $(A_1 + \cdots + A_m)x = (b_1 + \cdots + b_m)$ $\Leftrightarrow$ $Ax = b$의 최소제곱해(least-square solution)를 모든 참여자들이 공유하고자 하는 것이 PCMC-LSS 문제이다. 여기에서 최소제곱해는 $\|b - Ax\|$를 최소로 하는 벡터 x를 의미한다.

5.2.1. 안전한 분리 프로토콜

다음에 기술하는 프로토콜은 [7]에 나타나 있는 안전한 분리 프로토콜(secure split protocol)을 행렬 연산에 적용한 것으로 각 참여자의 개인 정보를 위장(blinding)시키는 과정이다.

프로토콜 1. 안전한 분리(split) 프로토콜

입력 : 각 참여자 $P_j (1 \le j \le m)$는 개인 정보인 $n \times n$ 행렬 A_j를 입력한다.

출력 : 각 참여자 $P_j (1 \le j \le m)$는 다음 성질을 만족하는 $n \times n$ 행렬 B_j를 얻는다.

$$\sum_{j=1}^{m} A_j = \sum_{j=1}^{m} B_j \ .$$

프로토콜 수행 과정 :

1. 모든 참여자들은 $1 \le t \le m-1$인 t를 공통의 안전성 파라미터(security parameter)로 정한다.
2. 각 참여자 $P_j (1 \le j \le m)$는 각 성분들이 같은 부호이고, 충분히 큰 난수로 구성된 t개의 랜덤한 행렬들 $R_1^{(j)}, \cdots, R_t^{(j)}$를 생성하여, 자신을 제외한 $m-1$명의 참여자들 중 t명을 랜덤하게 선택하여 하나씩 분배한다. P_j는 A_j의 지분(share)으로 $R_0^{(j)} = A_j - \sum_{i=1}^{t} R_i^{(j)}$라고 놓는다.
3. 각 P_j가 t'명의 다른 참가자들로부터 랜덤한 행렬들을 분배받았다고 할 때, $B_j = R_0^{(j)} + \sum_{i \ne j} R_{l_j}^{(i)}$를 계산한다. 여기에서 $R_{l_j}^{(i)}$는 P_i가 P_j에게 분배한 랜덤 행렬을 의미하고, $\sum_{i \ne j}$의 항 수는 t'개 이다.

위 프로토콜 수행 과정 3단계에서 각 참여자가 다른 참여자들로부터 분배받는 랜덤 행렬의 평균 개수는 t개 이다. 이는 다른 $m-1$명의 참여자들 각각이 $t/(m-1)$의 확률로 자신을 선택할 것이므로 기대값(expectation)은 $t(m-1) \cdot t/(m-1) = t$ 로 계산되기 때문에 자명하다. 그리고 $\sum_{j=1}^{m} A_j = \sum_{j=1}^{m} B_j$가 성립하는 이유는 $\sum_{j=1}^{m}\sum_{i=1}^{t} R_i^{(j)} = \sum_{j=1}^{m}\sum_{i \ne j} R_{l_j}^{(i)}$이기 때문이다.

프로토콜 1에서 데이터를 숨기는 방법은 정보이론(information theory) 관점에서는 안전하지 않지만, 실질적인 의미로는 안전하다고 할 수 있다. 정보이론 관점에서 각 개인 정보는 $t+1$명의 참여자들에게 분배되어 있으므로 프로토콜 1이 다른 프로토콜의 부분으로 사용될 경우 개인 정보의 부분적 노출이 발생할 수 있다. 그러나 개인 정보 A_j를 알아내기 위해서는 수행 단계 2에서 랜덤 행렬을 분배한 t명의 참여자와 단계 3에서 P_j에게 랜덤 행렬을 분배한 t'명의 참여자 모두가 공모해야만 한다. 이러한 이유 때문에 A_j를 얻기 위해서 공모하는 참여자가 t명 미만일 경우, A_j가 노출될 확률은 랜덤 행렬 하나를 올바르게 추측하는 확률보다 작아서 거의 0에 가깝다. 실제로는 공모하는 참여자가 t명 이상인 경우에도 A_j의 노출 확률이 t가 증가함에 따라 지수적으로(exponentially) 감소한다는 사실을 알 수 있다.

프로토콜 1은 각 참여자가 t명의 다른 사용자에게 행렬을 분배하는 것으로 종료되기 때문에 1라운드에 수행된다. 그러므로 전체 통신량(total communication)은 $O(tm)$이 되고, 각 참여자의 계산 복잡도(computational complexity)는 행렬의 연산 단위로 셈할 때, $O(t)$가 됨을 알 수 있다. 한편, 프로토콜 1은 $n \times n$ 행렬 A_j 대신에 n차원 열벡터 b_j를 적용하여 이로부터 $\sum_{j=1}^{m} b_j = \sum_{j=1}^{m} c_j$를 만족하는 n차원 벡터 c_j를 얻는 프로토콜로도 활용 가능함을 알 수 있다.

5.2.2. PCMC-LSE 프로토콜

안전한 분리 프로토콜을 적용하여 일차연립방정식

의 해를 구하기 위한 프라이버시 보존형 다자간 계산 프로토콜을 다음과 같이 구성할 수 있다.

프로토콜 2. PCMC-LSE 프로토콜

입력 : 각 참여자 $P_j(1 \le j \le m)$는 개인 정보인 $n \times n$ 행렬 A_j 와 n차원 열벡터 b_j 를 입력한다.

출력 : 각 참여자 $P_j(1 \le j \le m)$는 일차연립방정식 $(A_1 + \cdots + A_m)x = (b_1 + \cdots + b_m)$의 해 $x = (x_1, \cdots, x_n)$을 얻는다.

프로토콜 수행 과정 :

1. 각 참여자 $P_j(1 \le j \le m)$는 프로토콜 1을 0 행렬과 0 벡터를 개인 정보로 간주하여 수행한 다음, 수행 결과인 $Q^{(j)}$, $q^{(j)}$를 저장한다. 즉,

$$\sum_{j=1}^{m} Q^{(j)} = 0, \quad \sum_{j=1}^{m} q^{(j)} = 0$$

가 성립하고, $Q^{(j)}, q^{(j)}$의 안전성은 프로토콜 1의 안전성 파라미터 t에 의해서 보장된다.

2. P_1 부터 P_m까지의 참여자들은 공동의 랜덤한 $n \times n$ 정칙행렬 T와 S를 선택하고, 각 $P_j(1 \le j \le m)$는 $\hat{A}_j = T \cdot (A_j + Q^{(j)}) \cdot S$, $\hat{b}_j = T \cdot (b_j + q^{(j)})$를 계산하여 제 삼의 계산 의뢰 서버인 $\mathbb{U}$에게 보낸다. 여기에서 정칙 행렬 T와 S는 제 삼의 서버 $\mathbb{U}$가 알지 못하는 정보이다.

3. 제 삼의 계산 의뢰 서버 $\mathbb{U}$는 각 개체들로부터 $\hat{A}_j$와 $\hat{b}_j$을 받아서

$$\tilde{A} = \sum_{j=1}^{m} \hat{A}_j, \quad \tilde{b} = \sum_{j=1}^{m} \hat{b}_j$$

를 계산하여, 일차연립방정식 $\tilde{A}x = \tilde{b}$ 의 해 $\tilde{x}$를 구한 다음, $\tilde{x}$를 P_1 부터 P_m까지의 모든 개체들에게 전송한다.

4. 각 $P_j(1 \le j \le m)$는 $x = S\tilde{x}$를 계산하여 저장한다.

프로토콜 2는 P_1 부터 P_m까지의 참여자들 중 제 삼의 서버 $\mathbb{U}$와 공모하는 개체가 없다는 전제 하에 안전성이 보장된다. $m-1$명의 개체들이 공모한다고 할지라도 나머지 한 개체의 프라이버시는 보장된다. 서버 $\mathbb{U}$와 프로토콜 참여자 중의 한 명이 공모한다면, $\mathbb{U}$가 참여자들만이 공통으로 공유하고 있는 해 x를 알 수 있는 것과 같은 정보 누출이 발생할 수 있다. 그러나 참여자 개인의 프라이버시인 A_j와 b_j를 $\mathbb{U}$ 또는 다른 참여자들이 알아내기 위해서는 프로토콜 1의 안전성 파라미터인 t와 연관된 충분한 수의 참여자들과 공모해야만 한다.

프로토콜 2는 각 개체 $P_j(1 \le j \le m)$ 사이에 안전한 분리 프로토콜 1라운드가 수행되고, 참여자들과 서버 $\mathbb{U}$사이에 2라운드가 수행되어 총 3라운드로 구성된다. 전체 통신량은 $O(tm)$ 이고, 각 개체 P_j $(1 \le j \le m)$의 계산 복잡도는 분리 프로토콜 수행이 주가 되어 여전히 $O(t)$가 된다. 서버 $\mathbb{U}$는 선형계의 해를 구하기 위한 가우스 소거법(Gauss elimination)을 사용할 경우 $O(n^3)$의 계산 복잡도를 갖는다.

5.2.3. PCMC-LLS 프로토콜

최소제곱해를 구하기 위한 PCMC-LLS 문제를 해결하는 프로토콜은 선형계 $Ax = b$를 정방행렬이 포함된 정규계로 변환하는 작업으로 출발한다. 주어진 선형계에 대응하는 정규계는

$$A^T Ax = A^T b$$

이다. 여기에서 $A^T A$는 $n \times n$ 행렬이 되고, $A^T b$는 n 차원 벡터가 된다. 이 정규계의 협력 계산 모델은 $A = A_1 + \cdots + A_m$ 일 때,

$$\left(\sum_{j=1}^{m} \sum_{i=1}^{m} A_j^T A_i \right) x = \left(\sum_{j=1}^{m} \sum_{i=1}^{m} A_j^T b_i \right)$$

으로 표현된다. 이 때, $A_j, b_j (1 \le j \le m)$는 P_j의 프라이버시를 나타내는 $l \times n$ 행렬과 l차원 열벡터이므로 노출되지 않아야 한다. 각자의 프라이버시를 보호하면서 $A_j^T A_i$ 와 $A_j^T b_i$의 정보를 담고 있는 행렬과 벡터를 P_j와 P_i가 도출해 내기 위한 방법으로는 참고문헌 [4]에서 행렬곱(matrix product) 프로토콜로 명명된 방식을 이용한다. 행렬곱 프로토콜은 다음과 같다.

프로토콜 3. 행렬곱 프로토콜[4]

입력 : 개체 P_j는 개인정보인 $n\times n$ 행렬 B_j를 입력하고, 개체 P_i는 개인정보인 $n\times n$ 행렬 B_i를 입력한다.

출력 : P_j는 랜덤 행렬 R_j를 얻고, P_i는 랜덤 행렬 R_i를 얻는다. 여기에서 $R_j+R_i=B_jB_i$를 만족한다.

프로토콜 수행 과정 :

1. 개체 P_j와 P_i는 u^v번의 덧셈 연산이 계산량적으로 불가능한(computationally infeasible) 정수 u, v를 공동으로 정한다.
2. 개체 P_j는 v개의 랜덤 행렬 $X_1,\cdots,X_v$를 생성하여 $B_j=X_1+\cdots+X_v$가 되도록 한다.
3. 모든 $r=1,\cdots,v$에 대해서 P_j와 P_i는 다음의 부분 단계를 수행한다.
   (1) P_j는 P_i에게 $(H_1,\cdots,H_u)$를 전송한다. 여기에서 P_j의 비밀 정보 $k(1\le k\le u)$에 대해서만 $H_k=X_r$이 되고, 나머지 성분들은 모두 랜덤한 행렬들이다. 이렇게 하면 P_i는 X_r의 위치를 모르게 되는 것이다.
   (2) P_j는 $l=1,\cdots,u$에 대해서 $H_lB_j-R_r$을 계산한다. 여기에서 R_r은 랜덤한 행렬이다.
   (3) OT_1^N 프로토콜을 사용하여 P_j는 $H_kB_i-R_r=X_rB_i-R_r$을 얻는다.
4. 개체 P_j는

$$R_j=\sum_{r=1}^{v}(X_rB_i-R_r)=B_jB_i-\sum_{r=1}^{v}R_r$$

을 얻고, P_i는 $R_i=\sum_{r=1}^{v}R_r$을 얻는다.

행렬곱 프로토콜의 단계 4에서 두 개체가 각각 얻은 랜덤 행렬 R_j와 R_i 사이에는 $R_j+R_i=B_jB_i$라는 관계식이 성립함을 쉽게 알 수 있다. 또한, P_j가 행렬을 개인 정보로 가지고 있고, P_i가 벡터 b_i를 개인 정보로 가지고 있는 경우 $r_j+r_i=B_jb_i$가 성립하는 랜덤 벡터 r_j와 r_i를 서로 나눠 가질 수 있다는 사실도 알 수 있다. 이러한 프로토콜을 행렬-벡터곱 프로토콜이라 부른다. 이 두 가지 프로토콜을 사용하여 PPMC-LLS 문제를 다음 프로토콜과 같이 해결할 수 있다.

프로토콜 4. PCMC-LLS 프로토콜

입력 : 각 참여자 $P_j(1\le j\le m)$는 개인 정보인 $l\times n$ 행렬 A_j와 l차원 열벡터 b_j를 입력한다.

출력 : 각 참여자 $P_j(1\le j\le m)$는 선형계 $(A_1+\cdots+A_m)x=(b_1+\cdots+b_m)$의 최소제곱해 $x=(x_1,\cdots,x_n)$을 얻는다.

프로토콜 수행 과정 :

1. 각 참여자 $P_j(1\le j\le m)$는 프로토콜 3의 행렬곱 프로토콜과 행렬-벡터곱 프로토콜을 자신을 제외한 $(m-1)$명의 개체 $P_i(i\ne j)$와 수행하여 $R_j^{(i)}+R_i^{(i)}=A_j^TA_i$, $r_j^{(i)}+r_i^{(i)}=A_j^Tb_i$가 성립하는 $(R_j^{(i)}, r_j^{(i)})$와 $(R_i^{(i)}, r_i^{(i)})$를 각각 나눠 갖는다.
2. 각 참여자 $P_j(1\le j\le m)$는 $M_j=A_j^TA_j+\sum_{i\ne j}R_j^{(i)}$, $c_j=A_j^Tb_j+\sum_{i\ne j}r_j^{(i)}$를 계산한다.
3. 각 참여자 $P_j(1\le j\le m)$는 $(M_1+\cdots+M_m)x=(c_1+\cdots+c_m)$을 풀기 위해서 프로토콜 2의 PCMC-LSE 프로토콜을 수행한다.

위 프로토콜 4는 최소제곱해를 직접 구하는 방식이 아니고, 프라이버시를 보존하는 상태에서 최소제곱해 구하는 문제를 정규계의 일반해 구하는 문제로 변환시키는 프로토콜이다. 대응하는 정규계인

$$\left(\sum_{j=1}^{m}\sum_{i=1}^{m}A_j^TA_i\right)x=\left(\sum_{j=1}^{m}\sum_{i=1}^{m}A_j^Tb_i\right)$$

가 올바르게 도출되는 이유는

$$\sum_{j=1}^{m}M_j=\sum_{j=1}^{m}A_j^TA_j+\sum_{j=1}^{m}\sum_{i\ne j}^{m}R_j^{(i)}=\sum_{j=1}^{m}\sum_{i=1}^{m}A_j^TA_i$$

가 성립하기 때문이다. $\sum_{j=1}^{m}c_j=\sum_{j=1}^{m}\sum_{i=1}^{m}A_j^Tb_i$가 성립하는 사실도 유사하게 알 수 있다.

선형계의 최소제곱해를 구하기 위한 프라이버시 보존형 다자간 협력 계산 프로토콜인 PCMC-LLS 프로토콜은 부분 단계로 행렬곱 프로토콜과 행렬-벡터 프로토콜, 그리고 PCMC-LSE 프로토콜을 이용

한다. 행렬곱과 행렬-벡터 프로토콜은 프라이버시 보존 기술로 OT_1^N을 이용하기 때문에 PCMC-LSE 프로토콜에 비해서 여전히 비효율적이라 볼 수 있다. 그러므로 향후 행렬곱 프로토콜을 사용하지 않는 효율적인 프로토콜의 개발이 요구된다고 하겠다.

Ⅵ. 제안 프로토콜의 효율성

실용적인 관점에서 각 프로토콜의 효율성을 정량적으로 정확히 비교 측정하는 것은 쉽지 않은 일이다. 프로토콜에 따라서 사용하는 연산이 다르고 기준으로 삼는 파라미터가 상이하기 때문에 단순 비교에 의하여 프로토콜의 효율성을 논한다는 것이 큰 의미가 없을 수도 있다. 그러나 기호 $O(\cdot)$를 사용한 계산 복잡도의 측정은 이론적인 관점에서 프로토콜의 효율성을 제약적으로나마 비교해 볼 수 있는 기준을 제시해 준다. SMC 관련 프로토콜의 효율성에서 가장 중요한 항목은 통신비용(communication cost)이다. 그러므로 통신 복잡도 측면에서 본 논문에 제안된 프로토콜과 기존의 다른 방식들을 비교해 보고자 한다.

PCMC-LLS 문제는 PCMC-LSE 문제로 전환되기 때문에 PCMC-LSE 프로토콜을 중심으로 효율성을 비교하는 것이 타당하다. 프로토콜에 참여하는 개체들의 수는 m이라 하자. OT 프로토콜로 효율성 높은 Naor-Pinkas[13]의 방식이 사용될 경우, OT 프로토콜의 통신 복잡도는 $O(l)$이다. 여기에서 l은 공개키 암호 연산에서의 안전성 파라미터(security parameter)로 소인수 분해가 불가능한 정수의 비트 길이를 의미한다. 현재 상황에서 $l=1024$ 또는 $l=2048$ 정도를 고려하면 된다.

계산되는 행렬이 $n\times n$이라 하고, 행렬 원소들의 최대 길이가 d-비트라고 하자. 그러면 비트 회로 계산(circuit evaluation) 관점의 일반적 SMC 프로토콜을 사용하여 PCMC-LSE 문제를 해결할 경우, OT 프로토콜을 사용해야 하고, $m(m-1)/2$번의 양자간 계산이 반복되어야 한다. 가우스 소거법은 $O(n^3)$ 만큼의 곱셈 연산을 필요로 하고, 한 번의 곱셈을 위한 안전한 회로의 크기는 $O(d^2)$이므로 통신 복잡도는 $O(m^2\cdot l\cdot n^3\cdot d^2)$이 된다.

한편, 5절에서 언급한 바와 같이 OT 프로토콜을 기반 요소로 사용하고 있는 [4]의 양자간 계산 프로토콜을 반복 적용함으로써 다자간 계산 프로토콜을 구축할 수 있다. 양자간 프로토콜의 통신 복잡도는 OT 연산을 포함할 경우 $O(l\cdot\mu)$가 된다. 여기에서 μ는 [4]에서 사용된 안전성 파라미터로 $\mu=1024$ 정도를 적절한 수치로 삼고 있다. 그러므로 양자간 계산 방식의 반복 적용으로 다자간 계산 방식을 구축할 경우 통신 복잡도는 $O(m^2\cdot l\cdot\mu)$가 된다.

본 논문에 제안된 PCMC-LSE 프로토콜(프로토콜 2)의 통신 복잡도는 OT 프로토콜을 사용하지 않으므로 상대적으로 높은 효율성을 보인다. 제안 프로토콜 2의 통신 복잡도는 $O(t\cdot m)$이다. 여기에서 t는 공모자의 최대 수를 나타내는 안전성 파라미터이다. 각 방식의 효율성을 종합적으로 정리한 것이 [표 1]에 나타나 있다.

Ⅶ. 결 론

이론적으로는 모든 SMC 문제가 해결 가능하지만 이는 비트 회로 계산 관점의 결과이기 때문에 실제 응용 환경에서는 효율성이 큰 문제가 된다. 이러한 문제 때문에 특정 응용 환경에 따른 효율적인 다자간 계산 프로토콜의 개발이 요구된다. 최근 이러한 요구에 부응하여 다양한 SMC 응용 프로토콜이 발표되고 있지만 대부분이 양자간 계산 방식에 머무르고 있는 실정이다.

본 논문에서는 선형계의 일반해를 구하는 문제에 있어서 OT_1^N 프로토콜을 사용하지 않고, 프라이버시를 보존하는 효율적인 다자간 협력 계산 방식인 PCMC-LSE 프로토콜을 제안하였다. 여기에서 제안한 프로토콜은 통신복잡도(communication complexity)와 각 개체의 계산 복잡도(computational complexity)

표 1. 프로토콜의 효율성 비교

SMC 프로토콜	통신 복잡도	비 고
비트 회로 계산 프로토콜(Generic SMC)	$O(m^2\cdot l\cdot n^3\cdot d^2)$	- l: 공개키 암호 관련 안전성 파라미터
양자간 계산 프로토콜의 확장	$O(m^2\cdot l\cdot\mu)$	- μ: 양자간 계산 프로토콜의 안전성 파라미터
PCMC-LSE 프로토콜(프로토콜 2)	$O(t\cdot m)$	- t: 공모자의 최대 수

측면에서 양자간 프로토콜을 사용한 확장법 보다 매우 효율적이다. 또한, 선형계의 최소제곱해를 구하는 문제에 있어서는 정규계로 변환하여 효율적인 PCMC-LSE 프로토콜을 이용할 수 있는 PCMC-LLS 프로토콜을 제안하였다. 그러나 PCMC-LLS 프로토콜은 정규계로 변환하는 과정에서 기존의 양자간 계산 모델에서 사용한 방법을 차용하였기 때문에 효율성 측면에서 개선의 여지가 있을 것으로 사료된다.

참 고 문 헌

[1] A.C. Yao, "Protocols for secure computations", *Proceedings of the 23th Annual IEEE Symposium on Foundations of Computer Science*, 1982.

[2] O. Goldreich, S. Micali, A. Wigderson, "How to play any mental game", *Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, pp. 218-229, 1987.

[3] O. Goldreich, "Secure Multi-Party Computation (Final Draft, Version 1.4)", http://www.wisdom.weizmann.ac.il/home/oded/public_html/foc.html, 2002.

[4] W. Du, M.Atallah, "Privacy-preserving cooperative scientific computations", *14th IEEE Computer Security Foundations Workshop*, pp. 273-282, 2001.

[5] W. Du, M. Atallah, "Protocols for secure remote database access with approximate matching", *ACMCCS2000*, 2000.

[6] Y. Lindell, B. Pinkas, "Privacy preserving data mining", *CRYPTO 2000, LNCS 1880*, 2000.

[7] M. Atallah, M. Bykova, J. Li, K. Frikken, M. Topkara, "Private collaborative forecasting and benchmarking", *WEPS2004*, 2004.

[8] R. Cramer, I. Damgard, "Secure distributed linear algebra in a constant number of rounds", CRYPTO 2001, LNCS 2139, pp. 119-136, 2001.

[9] W. Du, M. Atallah, "Secure multi-party computation problems and their applications: A review and open problems", Proceedings of New Security Paradigms Workshop, pp. 11-20, 2001.

[10] L. Kissner, D. Song, "Privacy-Preserving Set Operation", Advances in Cryptology - CRYPTO 2005, LNCS 3621, Springer-Verlag, pp. 241-257, 2005.

[11] M. Freedman, K. Nissim, B. Pinkas, "Efficient private matching and set intersection", Advances in Cryptology -Eurocrypt 2004, LNCS 3027, Springer-Verlag, pp. 1-19, 2004.

[12] H. Anton, R. C. Busby, *Contemporary Linear Algebra*, John Wiley & Sons, 2003.

[13] M. Naor, B. Pinkas, "Oblivious transfer and polynomial evaluation", *Proceedings of the 31th ACMSTC*, pp. 245-254, 1999.

〈著者紹介〉

강 주 성 (Ju-Sung Kang) 정회원
1989년 고려대학교 수학과(학사)
1991년 고려대학교 일반대학원 수학과 (이학석사)
1996년 고려대학교 일반대학원 수학과 (이학박사)
1996년~1997년 과학재단 박사후연구원
1997년~2004년 한국전자통신연구원 선임연구원, 팀장
2001년~2002년 벨기에 루벤대학 COSIC 방문연구원
2004년~현재 국민대학교 수학과 부교수
〈관심분야〉 암호 알고리즘, 정보보호 프로토콜
e-mail : jskang@kookmin.ac.kr

이 옥 연(Ok-Yeon Yi) 정회원
1988년 고려대학교 수학과 졸업
1990년 고려대학교 일반대학원 수학과 (이학석사)
1996년 University of Kentucky 수학과 (이학박사)
1999년~2001년 한국전자통신연구원 선임연구원, 팀장
2001년~현재 국민대학교 수학과 조교수
〈관심분야〉 정보보호, 이동통신, 암호론
e-mail : oyyi@kookmin.ac.kr

홍 도 원(Dowon Hong) 정회원
1994년 고려대학교 수학과(학사)
1996년 고려대학교 일반대학원 수학과 (이학석사)
2000년 고려대학교 일반대학원 수학과 (이학박사)
2000년~현재 한국전자통신연구원 선임연구원, 팀장
〈관심분야〉 암호 이론, 정보보호 이론, 이동통신 정보보호
e-mail : dwhong@etri.re.kr