부호론

양자 컴퓨터 시대에도 깰 수 없는 암호 체계와 부호론

2018년 10월 12일

김종락

신용카드번호, 주민등록번호, 국제 표준 도서 번호ISBN의 공통점은 무엇일까? 그건 바로 확인 숫자, 즉 체크 디지트를 사용한다는 것이다. 이 일련의 숫자들의 마지막 자릿수는 미리 정해진 의미가 있는 것이 아니라 앞에 있는 수들이 맞는지 확인해주는 역할을 한다.

예를 들어 주민등록번호를 살펴보자. 우리나라의 주민등록번호는 총 13자리로 이루어져 있으며 ABCDEF-GHIJKLM의 형태를 띠고 있다. 처음 여섯 자리는 생년월일을 나타낸다. 일곱 번째 자리 G는 성별을 나타내거나 외국인을 나타낸다. 그다음 HI는 처음 등록한 지역의 고유번호를 의미한다. 예를 들어 서울이었다면 00에서 08번을, 부산이면 09에서 12번을, 인천이면 13에서 15번과 같은 식이다. 그다음 JK는 등록을 한 읍면동 주민센터의 고유 번호이다. 그 다음 L은 같은 주민등록번호를 가지는 사람이 생기지 않도록 순차적으로 부여되는 번호이므로 보통 1, 2, 3 정도가 될 것이다 (만일 우연히 10명을 넘기게 되면 JK 부분을 조정한다고 한다). 그렇다면 마지막 자릿수 M은 무엇을 의미할까? 다음과 같은 간단한 식으로 M을 구할 수 있다.

$(9A + 8B + 7C + 6D + 5E + 4F + 3G + 2H + 9I + 8J + 7K + 6L)$을 11로 나눈 나머지의 끝 자리수

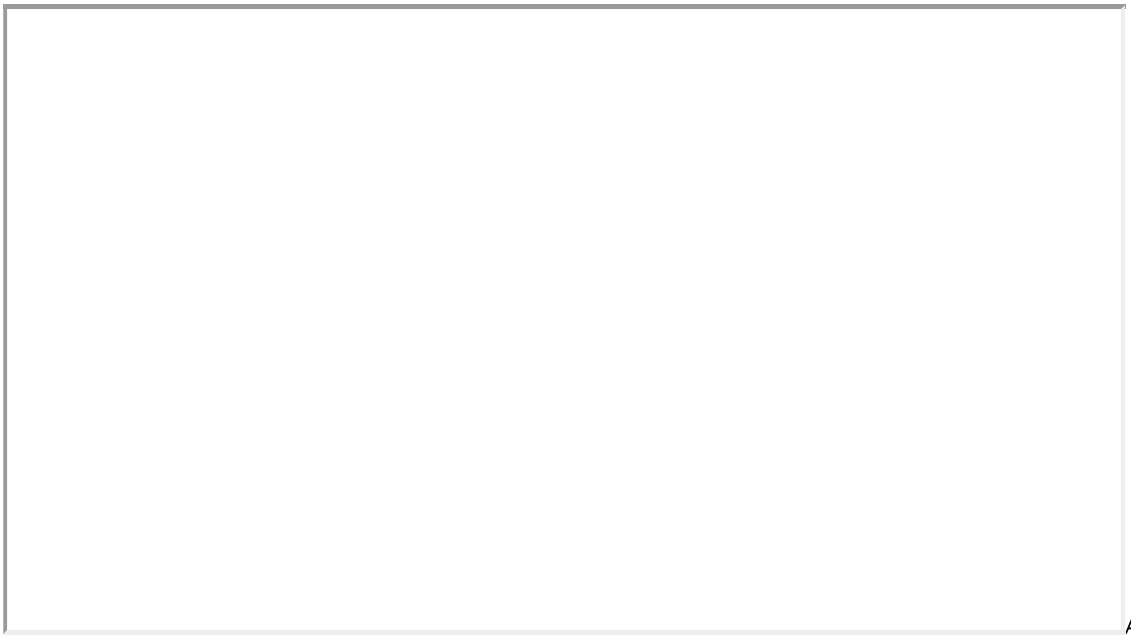
예를 들어 주민번호가 801115-123456X 인 경우 맨 마지막 X는 다음과 같이 계산하면 된다. 우선 9×8+8×0+7×1+6×1+5×1+4×5+3×1+2×2+9×3+8×4+7×5+6×6=247을 11로 나눈 나머지는 5이므로 X=5가 된다. 만일 주민등록번호를 쓰면서 어느 한 자리를 실수로 잘못 쓰면 위의 공식에 의해 마지막 수가 달라져서 오류가 있음을 바

로 확인할 수 있다.

하지만 이러한 방식으로는 오류가 있는지 확인만 할 수 있다. 오류가 있을 때 이를 바로잡는 일까지 할 수 있다면 더 좋을 것이다. 사실 이러한 기술이 없었다면 현재의 통신 수단 대부분은 노이즈 때문에 내용을 정확하게 전달할 수 없었을 것이다. 통신뿐만 아니라 CD와 HD TV에도 오류를 찾고 정정할 수 있게 하는 부호 이론이 사용된다. 이 글에서는 부호론 역사를 먼저 살펴본 후, 이것이 양자 컴퓨터 시대를 대비하여 개발 중인 새로운 암호 체계에 어떻게 활용되는지 살펴보고자 한다.

부호론의 역사와 기초 이론

클로드 섀넌Claude Shannon, 1916-2001은 1948년 "통신의 수학적 이론A Mathematical Theory of Communication"이라는 논문을 써서 '정보이론의 아버지'라고 불리게 된다. 이 기념비적인 논문에 영향을 끼친 두 사람이 있다.



AT&T Tech Channel에서 제작한 섀넌에 대한 다큐멘터리 <Tech Icons: Claude Shannon>

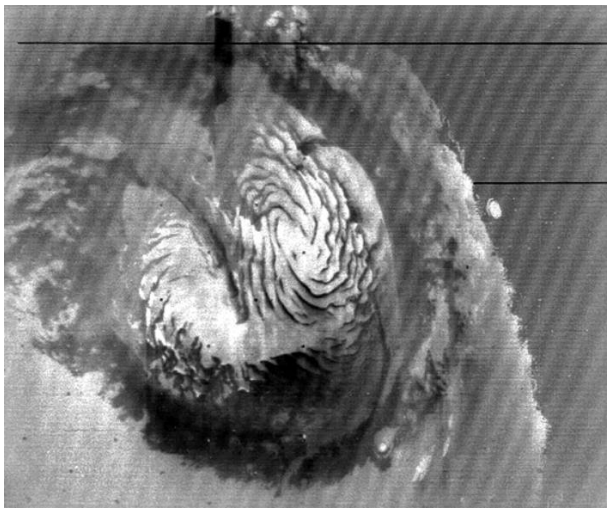
모튼 틸덤, <이미테이션 게임 *The Imitation Game*>(2014). 튜링이 영화의 주인공이다.

첫 번째 사람은 그 유명한 앨런 튜링Alan M. Turing, 1912-1954이다. 이차 대전 중 영국 정부는 독일군의 암호를 해독하기 위해 다양한 방법을 시도하였다. 영국 정부를 위해 일하던 튜링은 자신의 팀이 시도한 방법을 미 해군과 공유하기 위하여 1943년에 워싱턴 DC를 방문하였다. 일정을 마친 튜링은 통신 암호화에 관심을 갖게 되어 그 당시 최고의 통신사 연구소인 벨 연구소Bell Labs를 방문하게 된다. 이때 섀논과 튜링은 매일 티타임마다 구내 식당에서 만나 연구 이야기를 나누었는데, 이때 튜링이 자신이 1936년 논문에서 고안한 범용 튜링 기계Universal Turing Machine라는 개념을 섀넌에게 소개하였다고 한다. 범용 튜링 기계는 프로그램을 입력할 수 있어서 어떠한 튜링 기계도 모두 시뮬레이션이 가능한 튜링 기계이다. 이 개념은 그 후 새년에게 큰 영향을 주었다.

두 번째로 섀넌에게 영향을 준 사람은 벨 연구소에서 함께 근무하던 수학자 리차드 해밍Richard Hamming, 1915-1998이다. 해밍은 대공황 시기에 대학교에 지원하였는데, 공대가 없던 시카고 대학에서만 장학금 제의를 받아 어쩔 수 없이 원래 희망하던 공학 대신 수학을 전공하게 되었다. 그는 나중에 수학을 전공하길 잘했다면서 얼마나 다행인지 모른다고 했다고 한다. 그 후 네브라스카 주립대에서 석사 학위를 받은 뒤, 일리노이 대학교 어바나 샴페인Univ. of Illinois at Urbana-Champaign에서 미분방정식에 관한 학위 논문으로 1942년 박사학위를 받았다. 이후 해밍은 잠시 루이빌대학에서 조교수를 하다가 1945년 4월부터 원자폭탄을 제작하는 프로젝트였던 맨하탄 프로젝트에 참여하여 계산 관련 일을 하였으며, 1946년부터 벨 연구소에서 섀넌과 함께 일하게 되었다. 직급이 낮았던 그는 주말에 한 번 대형 컴퓨터를 사용할 수 있었는데, 초반부터 컴퓨터에 오류가 생겨 월요일에 출근해보면 결국은 아무것도 얻지 못하기 일쑤였다. 이때부터 그는 오류를 정정할 수 있는 부호를 만들기로 결심했다. 얼마 지나지 않아 해밍은 한 자리 정도는 오류가 생겨 틀리더라도 자동으로 정정할 수 있는 '해밍부호'를 발명하였다.

해밍부호는 $n = 2^r - 1$ 꼴이고 $k = n - r$일 때, k개의 0, 1로 구성된 문자열을 n개의 0, 1로 구성된 문자열로 적절하게 변형한다. 이런 식으로 얻어지는 길이 n인 0, 1로 구성된 문자열을 해밍부호어라고 한다. 길이가 n이고 0, 1로 구성된 문자열 y가 아무렇게나 주어져도 그것과 한 글자 이하로만 다른 해밍부호어 x가 정확히 하나 밖에 없다는 것이 해밍 부호의 특징이다. 즉 오류가 많아야 한 자리에서만 생겼다면 그렇게 해서 받은 문자열 y로부터 원래 받아야 했던 해밍부호어 x가 하나밖에 없어서 오류를 정정할 수 있게 된다. 해밍은 이 사실을 섀넌에게 알려주었다. 여기에서 영감을 받은 섀넌은 임의의 오류를 정정하면서도 정보의 양은 일정하게 유지할 수 있는 이른바 '좋은 부호'의 존재성을 보였다. 이후부터 부호론 학자들의 중요 연구 주제는 수학적인 방법, 특히 대체로 대수적 혹은 조합론적인 방법으로 좋은 부호를 구체적으로 찾아내는 것이었다.

섀넌의 통신이론의 핵심 아이디어는 '덧붙임redundancy '이다. 다시 말해서 보내고자 하는 정보, 예를 들어 길이 k인 0, 1로 구성된 문자열을 그대로 보내는 대신에 r개의 $0, 1$이 덧붙여진 길이가 n인 문자열을 보내는 것이다. 보내고 싶은 내용, 즉 원문이 길이 k인 문자열일 때, 보내는 사람은 그것을 길이 n인 문자열 x로 바꾸어 통신 채널을 통해 받는 사람에게 보낸다. 채널에는 노이즈가 있기 때문에 오류가 생길 가능성이 있어서 실제 받는 사람은 원래 보내려던 x가 아닌, 오류 벡터 e가 더해진 $y = x + e$를 받게 된다. 받는 사람은 사용된 부호의 특징을 이용하여 효과적으로 오류 벡터 e를 찾아낼 수 있고 따라서 x가 무엇인지 알 수 있으며 이를 토대로 원문을 복원할 수 있다.



나사의 화성탐사선 매리너 9호가 1972년 10월 촬영한 화성의 북극 사진. 리드 뮬러 부호를 사용하여 지구로 전송되었다.

NASA

부호론의 기본적인 문제는 크게 두 가지로 요약된다. 하나는 어떻게 원문을 부호화encoding 할 것인가라는 질문이다. 이것은 앞서 언급한 좋은 코드를 생성하는 문제와 같다. 또 다른 하나는 주어진 부호를 어떻게 효율적으로 해독하느냐는 질문이다. 실생활에 적용하기 좋으려면 오류를 많이 정정할 수 있는 좋은 코드이면서 동시에 효율적으로 해독하는 알고리듬이 있는 부호가 좋다. 부호론이 초기에 핵심적으로 사용된 곳은 어디일까? 바로 미국 항공우주국 나사NASA였다. 그 당시 부호론을 현실적으로 구현해낼 수 있는 분야는 자본이 집약된 우주 탐사선이었기 때문이다. 1971년 5월

30일 발사된 화성탐사선 매리너 9호Mariner 9는 화성의 사진을 찍어서 지구로 보낼 때 리드 뮬러 부호Reed-Muller code를 사용하였다. 이때 사용된 리드 뮬러 부호는 전문적으로 말하면 길이가 32이고 차원이 6인 이진 부호인데 이것을 이용하면 32개의 0, 1이 화성에서 지구까지 전송되는 동안 무려 7개의 오류가 발생하여도 정정할 수 있다.

부호론 기반 공개키 암호 체계

부호론에 관심이 있는 분이라면 지금까지 이야기는 많이 알고 있을 듯하다. 하지만 부호론으로 공개키 암호 체계가 가능하다는 이야기는 많이 알려지지 않았다.

먼저 공개키 암호 체계가 어떤 것인지 잠깐 살펴보자. 1970년대 이전까지는 암호화 할 때 쓰는 키(암호)와 복호화(암호화된 것을 푸는 과정)를 할 때 쓰는 키가 같은 대칭키 암호 체계만 알려져 있었다. 즉 암호화를 할 때 사용한 키를 알아야 그것을 이용하여 암호화된 문서를 복호화할 수 있었다. 그런데 이렇게 하면 사람 수가 많은 경우 두 사람의 쌍마다 일일이 다른 키를 써서 암호문을 주고 받아 키를 매우 많이 보관하여야 했다. 예를 들어 사람이 n명이라면 키를 $n(n-1)/2$개의 키를 만들어야 한다. 게다가 모르는 사람과 처음으로 통신을 하려면 함께 쓸 키를 정하고 아무도 모르게 키를 안전하게 주고받는게 어려웠다.

1970년대 새롭게 등장한 공개키 암호 체계는 이러한 불편을 크게 개선하였다. 모든 사람이 각자 공개키와 비밀키를 하나씩 가지고 있고, 공개키는 누구에게나 알려줄 수 있으며, A라는 사람에게 암호문을 보내고 싶은 사람은 A의 공개키로 암호화하여 내용을 보내면 되었다. A는 자기만 알고 있는 비밀키로 그 암호문을 복호화하여 읽어볼 수 있었다. 이렇게 되면 사람수가 n명이면 전체적으로 공개키 n개와 비밀키 n개만 만들면 된다. 모르는 사람과 통신하더라도 공개키는 이미 다 알려져 있는 것이므로 그것을 이용하여 암호화를 할 수 있다. 현재 인터넷에서 사용되는 공인인증서와 같은 시스템은 바로 1970년대 만들어진 공개키 암호 체계를 사용하고 있다. 이 중 가장 유명하고 지금도 전자상거래 등에서 널리 쓰이는 것이 바로 MIT에서 근무하던 세 수학자 라이베스트Ron Rivest, 샤미르Adi Shamir, 애들먼Leonard Adleman의 이름을 딴 RSA라는 암호 체계이다. 이 암호체계는 합성수를 소인수분해하는 것이 어렵다는 사실에 기반한 정수론의 아이디어를 써서 만들어졌다.

부호론의 대가인 로버트 맥엘리스Robert J. McEliece, 1942-는 RSA를 목격하고 얼마 안 있어서 부호론을 이용하여도 공개키 암호 체계를 만들 수 있음을 보였다. 그가 만든 방식은 아래와 같다.

먼저 이 암호체계에서는 0,1으로만 구성된 선형 독립이고 길이 n인 벡터 k개를 이용하여 부호어를 만든다(이런 부호를 선형 부호linear code라고 부른다. 여기서 $1+1=0$으로 가정한다). 각 행이 이 부호어로 이루어진 $k \times n$ 행렬 G를 이 부호의 생성 행렬generator matrix이라고 한다. 먼저 이렇게 만들어진 부호가 t개까지의 오류를 고칠 수 있다고 가정하자. 즉 길이 n인 $0, 1$ 벡터 어느 것을 보더라도 t개 이하의 자리를 바꾸어서 얻을 수 있는 부호어는 많아야 하나뿐이다.

이때 암호문을 받을 사람 A는 0과 1로만 구성된 $n \times n$ 행렬 중에서 각 행에 정확히 1이 하나씩 있고 각 열에도 정확히 1이 하나씩 있는 행렬 P와 역행렬이 존재하는 $k \times k$ 행렬 S를 준비한다. 이때 P와 S가 이 사람이 기억하여야 할 비밀키가 된다. 그리고 A의 공개키는 세 행렬 S, G, P를 순서대로 곱하여 얻은 $G' = S \cdot G \cdot P$이다.

이제 A에게 암호문을 보낼 사람 B가 있다고 하자. 이때 B는 길이가 k인 0, 1로 구성된 벡터 m을 암호화하여 보내고자 한다. 먼저 B는 1의 개수가 t개 이하인 길이 n인 0, 1로 구성된 벡터 하나를 아무거나 뽑아서 e라고 한다. 그리고 나서 $c = mG' + e$를 계산하여 벡터 c를 암호문으로 A에게 전송한다.

암호문을 받은 사람 A는 아래와 같이 원래 보내려던 내용 m을 구할 수 있다. 먼저 $c' = cP^{-1}$을 구한다. 그 후 C의 부호 체계를 이용하여 c'에 가장 가까운 부호어 m'을 찾는다. 그 후 $m'S^{-1}$을 구하면 그것이 m과 같아진다. 이때 A는 P와 S를 자신만 알고 있기 때문에 이런 계산이 가능하다.

이 암호문을 가로챈 사람이 있었다면 원래 보내려던 내용 m을 알 수 있을까? 이 공격자는 $c = mG' + e$ 값을 알지만 S나 P를 모르기 때문에 m을 알기가 어렵다. 이런 m을 찾아내는 문제를 신드롬 디코딩 문제Syndrome Decoding problem라고들 하는데 이 문제가 NP-complete라는 것이 증명되어 있어서, P와 NP가 다른 이상 효율적인 알고리듬을 기대하기 어렵다. 특히 이 문제를 해결하는 양자 알고리듬 또한 아직까지 알려져 있지 않다. 즉, 양자 컴퓨터 시대가 열린다고 하여도 이 문제가 어렵다는 것에 기반하여 만들어진 공개키 암호 체계는 여전히 안전할 가능성이 있다고 하겠다.

양자 컴퓨터 시대를 대비하는 공개키 암호 체계

양자 컴퓨터는 양자 효과를 이용하여 계산을 할 수 있는 새로운 방식의 컴퓨터로, 기존에 0과 1의 비트bit 단위로만 계산하던 튜링 기계 방식의 컴퓨터를 뛰어넘어 복소수에 대한 2차원 벡터 공간의 길이 1인 점으로 표현되는 큐비트qubit라는 단위로 계산이 이루어진다.

기존의 컴퓨터, 즉 튜링 기계에서는 아직까지 소인수분해를 효율적으로, 즉 자릿수 n에 대한 다항식 시간 이내에 해내는 알고리듬은 알려져 있지 않다. 하지만 1994년 MIT 수학과 교수 피터 쇼어Peter Shor, 1959-는 양자 컴퓨터를 이용하면 n자리 자연수를 소인수분해 하는 것을 $n^2 \log n \log \log n$에 비례하는 시간 이내에 할 수 있는 효율적인 양자 알고리듬을 만들었다. 따라서 양자 컴퓨터가 실용화된다면 소인수분해는 쉬운 문제가 된다.

기존에 널리 사용되는 RSA 암호 체계는 바로 소인수분해가 어렵다는 것에 기반한 것이라서 양자 컴퓨터 시대가 온다면 계속 쓸 수가 없다. 물론 당장은 걱정할 필요가 없다. 양자 컴퓨터는 아직 실용화까지 먼 길이 남아 있기 때문이다. 실험실에서 만들 수 있는 양자 컴퓨터는 아직까지 오류에 민감하여 큐비트 수를 많이 늘릴 수 없는 형편이다. 현재까지 만들어진 양자 컴퓨터로는 2001년에 겨우 15를 3×5로 소인수분해하는데 성공하였고, 2012년에 21을 7×3으로 소인수분해 하는 기록을 세웠다고 하니, 아직까지 RSA 암호 체계에 실제 쓰이는 매우 큰 소수까지 소인수분해하려면 매우 오래 걸릴 것이다.

하지만 벌써 미국 국립표준연구소NIST에서는 양자 컴퓨터 시대를 대비하여 새로운 암호 체계를 준비하고 있다. 공개적으로 NIST 포스트 퀀텀 암호 표준화를 진행하면서 제안서를 받았다. 그리하여 2017년 11월 말까지 25개 나라에서 제출한 82개 제안서 중 64개가 1라운드를 통과하였다. 이것들을 유형별로 나누면, 격자 기반 암호, 부호 기반 암호, 다항식 기반 암호, 해쉬 기반 암호, supersingular 타원곡선기반 암호 등이 있었다. 그중 가장 많은 것은 격자 기반 암호로서 총 26개의 제안서가 제출되었다. 총 20개의 제안서가 부호 기반 암호였는데 대부분이 McEliece가 만든 암호 체계

와 Niederreiter가 만든 암호 체계를 변형한 것들이었다. 필자의 서강대 연구팀 또한 NIST에 McEliece와 Niederreiter의 암호 체계를 결합된 McNie라는 것을 제안하였는데 1라운드를 통과하였고, 2라운드를 통과하기 위해 노력하고 있다.

필자의 연구팀 외에도 국내에서 4개의 팀이 1라운드를 통과하였다. 서울대의 천정희 교수 연구팀이 제안한 Lizard, 서울대의 노종선 교수 연구팀이 제안한 pqsigRM, 국가수리과학연구소의 심경아 박사팀이 제안한 HiMQ-3, 고려대 정보보호 대학원의 이동훈 교수 연구팀이 제안한 EMBLEM과 R.EMBLEM이 있다. 이처럼 앞으로 양자 컴퓨터 시대가 오더라도 안전한 암호 체계를 만들기 위한 연구 및 투자는 이미 국내외에서 치열하게 진행 중이고 부호론에서 그 실마리가 나올 수 있다고 생각한다.