

압축센싱 소개*

이홍노 · 박상준 · 박순철 (광주과학기술원)

I. 요약

최근에는 Compressive Sensing (CS)이라는 새로운 연구분야가 학계에서 많은 관심을 받고 있다. 이 분야는 2006년 이후 신호처리 및 정보이론 학회를 중심으로 매우 빠르게 성장해 왔는데, 현재는 정보통신, 이미징, 센서 및 인스트루멘테이션 등 유관 분야로 영향력을 넓혀 가고 있다. 우리는 이러한 흐름의 원인을 다음 두 가지로 본다. 그 하나는, CS가 장기간에 걸쳐 연구되어온 탄탄한 이론적 토대 위에 그 근거를 두고 있어서 많은 학자들의 관심을 끌고 있는 점이다. 두 번째는, CS가 제시하는 방향이 현재 대학 및 산업계에서 가르치고 또 행하고 있는 디지털 신호처리 방식을 근본적으로 바꿀 수 있게 할 정도로 커다란 그림을 그리고 있다는 점이다. 이러한 때에, 본 논문은 CS가 과연 무엇인지와 그것이 제시하고 있는 궁극적 목표를 소개하고자 한다. 또 CS를 이해하는데 필수적인 몇 가지 기술적 요소를 설명하고자 한다.

II. Introduction

기존의 정보/통신 시스템은 Shannon과 Nyquist에 의한 Sampling Theorem에 입각하여 설계된 디지털 시스템 위주로 발전되어 왔다. 우리가 말하는 디지털 시스템은 자연적인 아날로그 신호를 디지털 신호로 바꾸는 데서부터 출발한다. 일단 사진이나 음성 같은 아날로그 신호를 디지털 신호로 바꾸고 난 후에는, 신호를 실수체계가 아닌 정수체

계로 표현할 수 있으므로, 이진법으로 계산하고 기억하는 컴퓨터를 통해 신호를 저장할 수도 있고, 복제할 수도 있고, 또는 디지털 통신네트워크를 통해 오류 없이 전달할 수도 있게 되는 것이다.

이렇게 자연적인 신호를 디지털 신호로 만드는 첫 과정은 ADC (Analog-to-Digital Converter)라는 장치에 의해 진행된다. 그러므로 ADC 기술은 오늘날 디지털 시스템이 쓰이는 모든 곳에 필수불가결한 존재이다. 이러한 ADC 장치는 바로 Shannon-Nyquist sampling 이론에 근거해서 만들어져 왔는데 보다 작은 transistor 기술이 등장함에 따라 점점 더 고단위로 집적화 되어왔다. 이는 ADC 장치가 보다 더 빠르게 보다 더 적은 파워로 동작하는 쪽으로 진화를 거듭해 왔음을 의미한다. 그런데 ADC의 sampling rate은 표현할 수 있는 정보의 양에 비례한다. 보다 정확하게 말하면, 신호의 최고 주파수의 2배 이상으로 sampling을 하면 그 신호를 정확하게 다시 아날로그 신호로 복원할 수 있다. 이것이 바로 Shannon-Nyquist sampling 이론인데, 오늘날까지 이 이론은 디지털 시스템을 구축하는 기초이론으로 충실하게 이용되어 왔다.

이렇게 60여년전에 처음 소개된 이래로 그 권위를 자랑하며 사용되어 왔던 Shannon-Nyquist Sampling 이론이 최근 몇 년 사이에 학계를 중심으로 재조명되고 있다. 본 논문은 이러한 최신동향을 살펴 보고자 한다.

2006년에 IEEE Trans. on Information Theory에 압축센싱^[1,3,4] (Compressed Sensing; CS)이라는 불리우는 이론이 Donoho (Stanford 통계학과 교수), Candes (Caltech 수학과 교수), Tao (UCLA 수학과 교수, 2006 Fields Medalist) 등에 의해 소개되었다. 본 논문에서 의미하는 CS 이론이라 함은 이들 초기 연구자들에 의해 만들어진 것에 국한하고자 한다.

* 이 논문은 2010년도 정부 (교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (중견연구자 도약연구사업, NO. 2010-0017944)



이 이론이 제시하는 가장 흥미로운 점은 신호를 Nyquist rate 이상으로 sampling 하지 않아도, 완전하게 신호를 재생할 수 있는 경우를 다루고 있다는 것이다. 이 이론은 우리가 통상적으로 다루는 신호들은 대부분 어떤 특정한 신호 space로 변형(transform)시켜 보았을 때 대부분의 값이 0인 소위 “sparse” 신호라는 것이다. Sparse 신호는 말 그대로 $x-y$ graph 상에 신호를 그려 보았을 때 대부분의 x 값에서 y 는 0(zero) 이고, 상대적으로 아주 적은 수의 x 값에서 non-zero y 값을 가지는 신호를 말한다. CS 이론에 의하면 이런 sparse 신호는 아주 적은 수의 선형 측정(linear measurements)만으로도 원래의 신호를 거의 완벽하게 복원시킬 수 있다고 말하는 것이 그 중심적 이론이다. 뒤에서 보다 자세히 언급할 내용이지만, CS 이론에서 이러한 선형 측정 샘플의 갯수가 약 $O(K \log(N/K))$ 쯤 되면 신호 복구를 거의 완벽하게 할 수 있다는 정리가 있다. 여기서 K 는 원래 신호에서 0이 아닌 수들의 갯수이고, sparsity라 부른다. N 은 원래 신호의 길이이다.

CS 이론의 효과를 간단히 설명하기 위해서 다음과 같이 디지털 카메라의 예를 들 수 있다. 해상도가 1mega pixel인 카메라를 가지고 얼굴사진을 찍는다면, 약 1 백만 개의 pixel 값을 얻게 된다. 이 pixel 값을 일렬로 나열하여 벡터에 저장하면 우리는 N 개의 값을 가지는 측정신호 벡터를 얻게 된다. 그런데 이 신호를 Wavelet transform 해서 그 값을 계산해 보면 0이 아닌 값이 300개 또는 그 이하뿐이라는 것을 알게 된다. N 개의 pixel 값을 저장하기 보다는 300개 정도의 0이 아닌 transform 값들을 저장하는 것이 훨씬 쉽고 간편하다는 것은 상식상 진리이므로 우리는 300개의 Wavelet transform 값들만 저장한다. 원 영상 복원은 이들을 inverse Wavelet transform 함으로써 얻는다.

그런데 이 영상압축 및 복원 과정을 살펴보면 300개의 transform 값들을 얻기 위해 많은 수의 (1백만개) 센서가 센싱 하고, 그 결과를 센서 별로 ADC 하고, 또 그 값들을 측정 신호 벡터에 모아서 Wavelet transform 해야 하는 등 많은 신호 처리 과정과 계산을 거쳐야 한다는 것을 알 수 있다. 중국에는 300개를 뺀 나머지 값들은 (1000000-300=999700개) 모두 0에 가까우므로 쓸모가 없어 버리고 말 것임을 생각해 보면 계산을 얼마나 낭비적으로 하는지 알 수 있다.

CS 이론은 이러한 기존의 불필요한 계산과정을 훌쩍 건너 뛰어 버릴 수 있는 방법을 제시한다. 대부분 영상이 300개 또는 그 보다 적은 수의 0이 아닌 값들을 가지는 sparse 신호라는 사전 정보만 가지고 있으면 CS 이론은 곧바로 $1400 \approx O(300 * \log(10^7/300))$ 개 또는 그보다 약간 많은 정도의 선형 측정 샘플만 얻게 한다. 그래서 원하면 언제든지

원 영상을 복원할 수 있게끔 해준다. 이러한 최저 샘플갯수는 CS 이론이 제시하는 신호 복원을 위한 충분조건이다. 금방 알 수 있듯이 $M \approx 1400$ 개의 선형 측정 샘플 갯수는 샘플링 이론의 충분조건 $N = 1000000$ 보다 훨씬 적다. 그렇지만 이렇게 훨씬 적은 수의 샘플로 CS 이론은 본래 사람 얼굴을 거의 같은 해상도로 복원해 낼 수 있는 방법을 제시한다. 이 때 사용되는 복원 방법은 linear programming 이나 greedy algorithm에 의해 수행 된다. 이런 신호복구에 대한 이야기는 앞으로 (III 장 절 2와 V 장) 에서 좀더 설명 하겠다.

앞의 예를 통해 본 것처럼, CS 이론은 모든 디지털 시스템에 필수 불가결한 신호 취득 과정을 새롭고 보다 효과적으로 할 수 있는 방법을 제시하기 때문에, 그 파급 효과가 매우 크리라 예상되고, 그로 인해 많은 연구자로부터 관심을 끌고 있다. 신호 측정 방식이 바뀌면, 그 위에 올려진 변복조, 필터링, 신호 검파 및 복원 등 많은 요소 기술의 변화가 필요하게 되고 이러한 기운은 새로운 형태의 정보통신 산업 발전을 촉발할 수 있기 때문이다.

이렇게 수학적인 CS 이론을 엔지니어링과 보다 밀접하게 만들어주는 연구자 중 빼놓을 수 없는 학자가 RICE University의 Richard Baraniuk 교수다. 그는 2007년부터 NSF (National Science Foundation, U.S.A.)와 DARPA (Defense Advanced Research Project Agency, U.S.A.) 등으로부터 지원을 받아 CS를 연구하고 있으며, 여러가지 빠른 신호 복원 방법을 개발 하였으며, 또 single-pixel compressive sensing camera를 만들어서 CS가 단지 이론만이 아닌 실제 구현 가능한 것임을 보여 많은 관심을 끌고 있다. 또 자신이 속한 Electrical and Computer Engineering 학과에 논문을 관리하는 web-site (<http://dsp.rice.edu/cs>) 도 운영하고 있는데 현재 여기에는 수백 편의 CS 논문과 튜토리얼 및 프리젠테이션 등이 분야별로 일목 요연 하게 정리되어 있다.

이제 III 장에서는 CS 이론의 배경을 설명하면서, CS 이론에서 자주 거론되는 기본적인 해법들을 논하겠다. 제 IV 장에서는 또한 측정하고 있는 신호가 예측했던 것보다 sparse 하지 않았을 때에도 CS는 원활한 신호복원 (Stable Recovery)을 할 수 있다는 점을 이야기 하고자 한다. 제 V 장에서는 sparse 한 해를 하나씩 찾아 나가는 greedy 해법 중 하나인 Homotopy 알고리즘을 설명하고자 한다. 제 VI 장에서는 이 논문의 결론을 제시한다.

III. The CS Theory

앞에서 CS 이론이 가지는 의의에 대하여 소개하였다. 이

번 섹션에서는 CS 이론에서 사용하는 기본적인 선형 측정 방법과 신호 복원 방식 등에 대해서 소개 하고자 한다.

CS 이론의 핵심은 다음 등식의 해를 찾는 것으로 요약 할 수 있다:

$$\mathbf{y} = \mathbf{A}\mathbf{x}. \quad (1)$$

위와 같은 선형 측정 식이 있을 때, 원래의 신호 \mathbf{x} 로부터 어떤 행렬을 곱해서 얻은 \mathbf{y} 를 선형 측정된 신호라고 정의 한다. 이 때 \mathbf{A} 는 $[M \times N]$ 행렬이고 일반적으로 M 은 N 에 비해서 작은 값을 가진다. 이때 M/N 은 신호 압축률로써 쓰인다. 압축률을 높이려면 주어진 N 을 두고, M 을 최대한 줄여야 한다. 따라서, 위 선형측정 식은 under-determined system을 푸는 것이다. 한편, 원 신호에서의 0이 아닌 값들의 수를 sparsity라고 정의 하고, sparsity를 K 라는 변수로 표현 하면, M, N, K 의 관계는 $K < M \ll N$ 이 된다. CS에서 말하는 선형 측정이란 신호 \mathbf{x} 를 선형측정 행렬 \mathbf{A} 를 곱하여 측정벡터 \mathbf{y} 를 얻는것을 의미한다.

앞에서 CS 이론은 우리가 통상적으로 다루는 신호들은 대부분 어떤 특정한 신호 space로 변형(transform)시켜 보았을 때 대 부분의 값이 0인 소위 “sparse” 신호라고 제시 한다고 말하였다. 이때에는, CS 기본식 (1)을 다음과 같이 표현 하면 된다:

$$\begin{aligned} \mathbf{y} &= \mathbf{A}\mathbf{x} \\ &= \mathbf{A}\mathbf{T}\mathbf{s} \\ &= \mathbf{F}\mathbf{s}. \end{aligned}$$

즉 신호 \mathbf{x} 가 직접적으로 K -sparse신호가 아닐 때에는 어떤 선형 변형 과정을 거치면 K -sparse신호로 표현될 수 있는 형식으로 모델을 세운다. 여기서, \mathbf{T} 는 역이 존재하는 $[N \times N]$ 선형 변형 행렬이고 \mathbf{s} 는 K -sparse신호가 된다. 이때에는 선형 측정 행렬을 $\mathbf{F} = \mathbf{A}\mathbf{T}$, K -sparse신호를 \mathbf{s} 라 놓고 진행하면 다시 기본식 (1)의 형태가 된다. 그러므로, 본 논문에서는 일반화를 잃지 않으면서도, 간결한 CS 기본식 (1)에 초점을 맞추고 이야기를 계속 진행하고자 한다.

1. 행렬 \mathbf{A} 와 RIP 조건

이 섹션에서는 “어떻게 \mathbf{A} 를 만들 것인가 또는 찾을 것인가?”라는 질문을 던지고 답하고자 한다. Candès, Romberg와 Tao^[3,4]는 선형측정 행렬 \mathbf{A} 를 Gaussian 이나 Bernoulli 분포로 생성하거나, Fourier 선형 변형 행렬을 사용

하였다. 그때에 선형 측정 값의 갯수 M 이 $O(K \log(N/K))$ 정도면 아주 높은 확률로 정확한 신호 복구가 가능하다는 결과를 발표하였다. 이러한 행렬군 \mathbf{A} 는 Restricted Isometry Property(RIP)라는 조건을 만족 시킬 확률이 크고, 그때에 CS는 K -sparse신호를 높은 확률로 복구할 수 있다는 결론을 낸다. RIP는 식으로 표현 하면 다음과 같다:

$$(1 - \delta_{2K}) \|\mathbf{x}\|_2 \leq \|\mathbf{A}\mathbf{x}\|_2 \leq (1 + \delta_{2K}) \|\mathbf{x}\|_2. \quad (2)$$

여기서, $0 < \delta_{2K} < 1$ 는 RIP Constant라고 불리는 상수다. 이 상수가 0에 가까울 때 RIP 조건이 의미를 가진다.

이 RIP조건을 간략하게 요약하면, 행렬 \mathbf{A} 가 모든 sparse한 신호 \mathbf{x} 를 균일한 에너지로 투사해야 한다는 의미다. 다시 말하면, 측정된 벡터 \mathbf{y} 의 길이가 \mathbf{x} 의 길이와 거의 같아야 한다. 중요한 것은 이것이 임의의 \mathbf{x} 를 사용하더라도 일정하게 같아야 한다는 것이다. 만일 \mathbf{A} 가 어떤 특정 지역(subspace)에 속한 \mathbf{x} 를 더 잘 투사하는 특성이 있다면, 그것은 곧 그 행렬 \mathbf{A} 가 잘 투사하지 못하는 비 선호지역도 존재하게 된다는 논리다. 그러면 큰 측정값을 얻게 해주는 선호 지역에 속한 신호가 \mathbf{x} 일 때에는, 측정값이 대체로 크므로 그를 이용한 신호 복구가 잘 이루어지겠지만, 비 선호지역에 속한 신호를 복구 하려고 하면, 복구 에러가 크게 발생하게 될 것이기 때문이다. 따라서, 주어진 측정 에너지를 가지고 모든 지역의 신호를 균일하게 커버 하려면, 어떤 임의의 벡터 \mathbf{x} 도 균일한 에너지로 투사하는 \mathbf{A} 가 가장 좋은 측정 행렬이라는 것이다.

하지만 임의의 sparse한 신호에 대해서 (2)가 성립되는 것을 확인 하는 것은 NP-hard문제 즉 모든 가능한 \mathbf{x} 에 대하여 조사해 봐야 하는 문제이기 때문에, 어떤 특정 행렬 \mathbf{A} 가 좋은 행렬인지 판단하는 척도로 (2)를 사용하는것은, 그다지 좋은 방법은 아니다. 절 3에서는 보다 실용적인 확인 가능한 구분방법을 제시한다.

이제 CS 기본식 (1)을 풀 수 있는 sparse 신호 복원 방식에 대해서 설명하겠다.

2. CS Recovery Criteria

앞에서 말한 것과 같이 (1)은 under-determined system이다. 따라서 (1)의 해는 무수히 많다. 이는 \mathbf{A} 의 null space를 이용하여 다음과 같이 설명할 수 있다. 원래 신호 \mathbf{x} 와 \mathbf{A} 의 null space에 있는 모든 벡터 \mathbf{u} 를 더한 $\mathbf{x} + \mathbf{u}$ 또한 (1)의 해가 된다. 즉 $\mathbf{y} = \mathbf{A}(\mathbf{x} + \mathbf{u})$ 이다. 하지만 우리는 sparse한 해를 찾고자 원하기 때문에, $\mathbf{x} + \mathbf{u}$ 중에서 가장



sparse한 해를 찾으면 된다.

아래에서는 (1)을 풀 수 있는 3가지 방식을 제시 하고, 각각의 방식이 어떤 특징을 가지고 있는지 간략하게 설명하고자 한다.

- L2 최소화를 이용한 복원: 가장 보편적으로 사용되어 온 방식인 L2 최소화 문제를 풀어 해를 구한다:

$$\bar{x} = \arg \min \|x\|_2 \text{ such that } Ax = y. \quad (3)$$

이런 최적화 문제의 해는 $\bar{x} = A^T(AA^T)^{-1}y$ 로 잘 알려져있다. 하지만 이런 방식은 K -sparse 신호에는 적합하지 않다. 왜냐하면, L2 최소화는 0이 아닌 수가 많은 non-sparse 해를 찾아주기 때문에, 정답인 sparse 해와 매우 큰 차이가 나기 때문이다.

- L0 최소화를 이용한 복원: 0이 아닌 요소의 개수를 나타내는 L0 norm $\|x\|_0$ 을 최적화에 이용하면 신호가 sparse하다는 특성을 최대한 활용하는 해를 구할 수 있다. 즉, L0 최적화 문제는 다음과 같이 표현 된다:

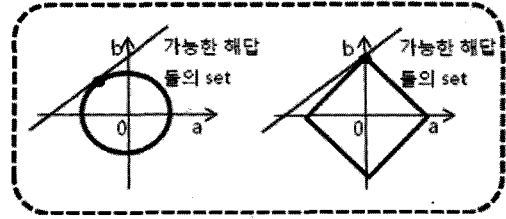
$$\bar{x} = \arg \min \|x\|_0 \text{ such that } Ax = y. \quad (4)$$

이때 측정 샘플의 갯수가 $M \geq 2K$ 정도로 충분하면, K -sparse 신호를 얻을 수 있다. 하지만 (4)는 $\binom{N}{K}$ 개의 K -sparse 벡터를 모두 점검해야 하는 NP-hard 문제가 되어 실용성이 떨어지는 단점이 있다.

- L1 최소화를 이용한 복원: L1 크기를 최소화하는 문제는 다음과 같이 정의 된다:

$$\bar{x} = \arg \min \|x\|_1 \text{ such that } Ax = y. \quad (5)$$

벡터 x 의 L1 norm은 모든 원소의 절대값의 합으로 정의된다. 즉, $x = (1 \ 2 \ -1)$ 일 경우 4 이다. 반면에, L2 norm은 모든 원소의 제곱의 합의 제곱근인 $\sqrt{6}$ 가 된다. <그림 1>의 예에서 보듯이 L1 norm이 일정한 벡터들의 집합은 마름모 모양의 L1 구의 표면에 위치한 벡터들이 되고, L2 norm의 경우는 동근 구의 표면에 위치한 벡터들이 된다. L1 최소화를 이용하면 단지 $M \geq cK \log(N/K)$ 개 정



<그림 1> 행렬A에 의해 가능한 해의 집합은 직선으로 표현된다. 왼쪽그림에서는 이 직선이 최소 L2 크기의 구(ball)와 만나는 지점이 해가 되는 것을 나타내고 있고, 오른쪽에서는 최소 크기 L1 구와 만나는 곳이 해가 되는 것을 보이고 있다

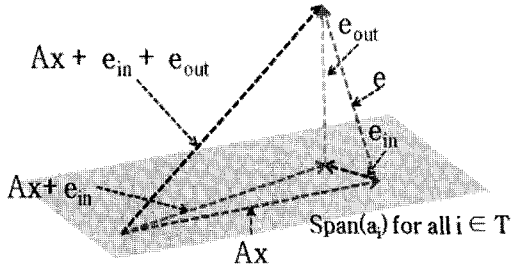
도의 측정값이 있으면 K -sparse 신호를 높은 확률로 복원할 수 있다^[3,4]. 이 최적화 문제는 Basis Pursuit^[11]이라 불리는 선형 문제(linear program)로 치환하여 풀수 있는데 이의 계산 복잡도는 대략 $O(N^3)$ 이다.

<그림 1>은 L2 최소화와 L1 최소화의 결과를 간단한 그림으로 예를 든 것이다. 그림에 나타난 바와 같이 행렬A에 의해 가능한 해의 집합은 직선으로 표현 된다. 왼쪽 그림에서는 이 직선이 최소 L2 크기의 구 (ball)와 만나는 지점이 해가 되는 것을 나타내고 있고, 오른쪽에서는 최소 크기 L1 구와 만나는 곳이 해가 되는 것을 보이고 있다. 특이한 점은 L1 구는 보여진 대로 정마름모 꼴이라는 점이다.

<그림 1>은 하나의 선형측정값을 가지고 sparse 신호 x 를 찾는 문제이다. $N = 2$ 이고 $M = 1$ 인 이 예는 다음과 같이 표현될 수 있다. $y = a^*v_1 + b^*v_2$. 그러므로 x 는 $[2 \times 1]$ 벡터 이고 그 첫째, 둘째 원소는 각각 a와 b다. 즉 $x = (a \ b)^T$ 이다. v_1 과 v_2 는 선형 측정 행렬 $[1 \times 2]A$ 의 첫째와 둘째 원소들이고 y 는 측정된 값이다. 두 개의 미지수 a 와 b 를 1개의 측정값 y 를 이용하여 찾으려는 시스템이다. 그러므로 가능한 해답의 갯수는 무한하며, 직선의 방정식 $y = a^*v_1 + b^*v_2$ 를 만족시키는 모든 (a, b)가 해답이 된다. 즉 가능한 해답들의 집합인 직선은 행렬 A로부터 결정이 되는 것이다.

이 예로부터 우리는 복구 하는 방식과 행렬 A는 밀접한 관련이 있다는 것을 볼 수 있다.

이제 L2 최소화의 해와 L1 최소화의 해를 비교해 보도록 하자. <그림 1>에서 표현 된 것과 같이, L2 norm과 직선이 만나는 부분은 x 축, y 축이 아닌 제2사분면이다. 즉 직선과, L2 norm의 교집합은 a, b의 값이 0이 아닌 것을 알 수 있다. 하지만 L1 norm과 직선의 교집합은 y 축 상에서 만나는 것을 알 수 있다. 이는 L1 최소화 해가 sparse하다는 것을 말해준다. 따라서, L2 최소화 방법은 정답을 찾지 못 하였지만 L1 최소화 방법은 정답을 찾는다라는 것을 알 수 있다.



〈그림 2〉 L1 최소화 사용되는 신호는 잡음이 부분적으로 제거가 된 $Ax + e_{in}$ 이 사용이 된다

〈그림 2〉는 L1 과 L2방법의 차이점을 보다 구체적인 방법으로 다시한번 확인한 결과를 보여준다. 다섯개의 non-zero값을 가지는 신호 \mathbf{x} 를 선형 측정하고 L1 최소화 와 L2 최소화의 해답을 비교해본 것이다. 보여진 대로 L2 최소화의 해답은 신호 전반적으로 대부분의 값이 non-zero값이 나오지만 L1 최소화는 5개의 non-zero값과 위치를 정확히 알아내 sparse한 답을 찾았다. L1 해답은 완벽하지만 L2는 완전히 틀린 답을 내놓고 있다.

결론적으로, 〈그림 1〉을 설명 할 때 잠시 언급한 바와 같이 행렬 A 는 신호의 정확한 복구에 직접적으로 영향을 준다.

따라서 우리는 이제 행렬 A 와 sparse 신호와의 관계에 대해서 알아 보고자 한다.

3. Conditions for Unique Solutions

앞에서 설명한 L1 최소화 해는 sparse한 해를 찾고 그러므로 정답을 찾을 가능성이 높다는 것을 그림과 예로써 설명하였다. 여기에서는 L1 최소화 해가 유일한 해를 가질 조건과, 또 그 유일한 해가 과연 L0 최소화 해와 일치할 조건은 무엇인가를 알아보고자 한다.

이러한 조건은 앞에서 언급한 바와 같이 선형 측정 행렬 A 와 밀접한 관계가 있다. 즉, A 는 절 1 에서 설명한 바와 같이 RIP 조건을 만족해야 한다. 그러나, 이 조건을 확인하는 것은 거기서 언급한 바와 같이 NP-hard 문제이므로, 여기서는 다른 확인 가능한 방법을 제시하고자 한다. 즉, 행렬 A 의 열간의 상호연관관계 (Coherence of A)를 조사 함으로써 L0 와 L1최소화 조건을 제시 할 수 있다.

A 의 최대 상호상관값: 행렬 A 의 열을 \mathbf{a}_i , $i = 1, 2, \dots, N$ 이라 하고, 이들이 각각 정규화(L2 norm이 1이 되도록) 되어 있을 때, 최대 상호상관 값은 다음과 같이 정의 된다:

$$\mu(\mathbf{A}) := \max_{i \neq j} \left| \langle \mathbf{a}_i, \mathbf{a}_j \rangle \right|. \quad (6)$$

제 3장에서 언급하였듯이, 선형측정 행렬 A 는 복구된 해와 밀접한 관련이 있다. 그렇기 때문에 최대 상호상관 값 역시 복구된 해와 밀접하게 관계한다.

우리는 CS 기본식 (1)을 만족하는 해는 무수히 많이 존재하는 것을 안다. 하지만 그 무수히 많은 \mathbf{x} 중에서 가장 sparse한 신호를 유일하게 얻어야 한다. 이러한 유일한 해를 얻을 필요조건을 Donoho 와 Huo는 [10]에서 최대 상호상관 값(6)을 이용하여 표현 하였다.

아래에서는 “L0 최소화 해가 유일할 조건”과, “L1 최소화의 해가 그 유일한 L0 최소화 해와 일치할 조건” 등에 대하여 소개하고자 한다.

L0 최소화 해가 유일할 조건: 신호 \mathbf{x} 가 K -sparse신호이고, L0 최소화를 통해 \mathbf{x} 를 유일하게 복구하려면, 다음 조건

$$\|\mathbf{x}\|_0 < \frac{1}{\mu(\mathbf{A})} \quad (7)$$

이 만족되어야 한다^[10]. 이 결과가 의미하는 바는 동일한 크기의 두 $[M \times N]$ 선형 측정 행렬을 비교할 때, 최대 상호상관 값 $\mu(\mathbf{A})$ 가 작은 행렬이 보다 큰 sparsity를 가지는 신호를 복구 할 수 있다는 것이다. 즉 최대 상호상관 값이 작으면 작을 수록 더 많은 정보를 복구 할 수 있기 때문에 더 좋은 선형 측정 행렬이라고 간주 할 수 있다.

L0 최소화 해와 L1 최소화 해가 일치할 조건: (7)이 성립하는 A 를 사용하면, L0 최소화 해(4) 는 유일하다. 하지만 L1 최소화를 사용하면, (7)의 조건은 L1 최소화 해가 L0최소화 해와 일치한다는 보장을 주지 않는다. 이것을 보장하는 보다 더 엄밀한 조건은 다음과 같다:

$$\|\mathbf{x}\|_0 \leq \frac{1}{2} \left(1 + \frac{1}{\mu(\mathbf{A})} \right). \quad (8)$$

식 (8)을 만족 하면, L0 최소화 해와 L1 최소화 해는 일치한다^[10]. 다시말하면, (8)의 오른쪽 값이 (7)의 오른쪽 값보다 작은 숫자이다.

한편 (8)은 2절 에서 Homotopy 알고리즘이 LASSO 알고리즘과 같아지기 위한 필요조건으로 사용된다. 이에 대한 것은 그 부분에서 다루기로 한다.

이제 최대 상호상관 값의 lower bound에 대해서 생각



해보자. 즉 선형 측정 행렬이 $[M \times N]$ 일 때 최대 상호상관 값의 lower bound를 M 과 N 에 대한 식으로 표현 하자는 것이다. 이렇게 최대 상호상관 값을 M 과 N 에 대해서 표현을 할 수 있다면, sparsity $\|\mathbf{x}\|_0$ 와 M, N 과의 관계를 대략적으로 알 수 있다. 즉, M, N 이 주어지면, L1 최소화 방식으로 정확한 해를 찾을 수 있는 최대 sparsity 레벨을 바로 알 수 있게 된다.

최대 상호상관 값의 lower bound: $[M \times N]$ 인 선형 측정 행렬 A 가 주어졌을 때, $\mu(A)$ 의 lower bound는 다음과 같이 표현된다:

$$\mu(A) \geq \sqrt{\frac{N-M}{M(N-1)}}. \quad (9)$$

즉 M 이 증가 하면 할 수록 $\mu(A)$ 의 값이 0에 가까워진다. M 과 N 이 같게 되면 $\mu(A)$ 는 0이 되고, L0 최소화의 해가 유일할 조건 (7)의 sparsity $\|\mathbf{x}\|_0$ 가 항상 무한대 보다 작다는 무의미한 결론이 나온다. 실제로 M 이 N 과 같고 또 A 의 역행렬이 존재하면 항상 유일한 해를 구할 수 있게 된다.

한편 절 2 에서 우리는 L1 최소화를 통하여 신호를 복구 할 때 M 의 개수가 대략 $O(K \log(N/K))$ 이면 높은 확률로 복구가 성공 한다는 결과를 말하였다. 이와 비교하여 볼 때, (8)과 (9)를 이용하여, M 에 대한 lower bound를 구할 수 있다. 이를 통하여, 우리는 선형 측정 샘플의 개수 M 이 대략 $O(K^2)$ 정도 필요 하다는 결론을 내릴 수 있다. 한가지 주목할 내용은, 그런 선형 측정 행렬 A 를 사용하면, 신호 \mathbf{x} 를 100% 확률로 복구 할 수 있다는 점이다.

IV. Stable Recovery

이제, 좀 더 실용적인 상황일 때의 L1 최소화의 성능에 대해서 이야기 해보자. 즉, 실제 시스템에서는 선형 측정을 하게 되면 잡음이 반드시 포함되게 된다. 또한 우리가 다루고 있는 신호들이 항상 K -sparse 신호가 아닐 때도 고려 해야 한다. 먼저 신호에 잡음이 있을 때 L1 최소화의 성능에 대해서 이야기 해보자.

첫 번째로, 신호에 잡음이 있다면, 신호를 복구하기 위한 L1 최소화 문제 (5)가 다음 식으로 바뀐다:

$$\bar{\mathbf{x}} = \arg \min \|\mathbf{x}\|_1, \text{ such that } \|\mathbf{Ax} - \mathbf{y}\|_2 \leq \varepsilon. \quad (10)$$

이 때, $\mathbf{y} = \mathbf{Ax} + \mathbf{e}$ 이고 $\|\mathbf{e}\|_2 \leq \varepsilon$ 이다.

위의 식이 뜻하는 것은 선형 측정시 잡음 \mathbf{e} 가 섞여들어오지만, 그 에너지가 특정 수준 보다 작은 한정된 것일 경우이다. 이런 상황일 때, Candès 등은 [9]에서 최적화 문제 (10)으로 해를 구하면, 복구 에러의 절대값 제곱이 잡음 에너지의 상수배보다 작거나 같다는 것을 보였다.

즉, K -sparse 신호에 대해서, $\delta_{3K} + 3\delta_{4K} < 2$ 가 성립이 되고, 원 신호 \mathbf{x} 의 support set, non-zero 원소들의 index의 집합, 의 크기가 K 보다 작거나 같고, 잡음 벡터 \mathbf{e} 의 에너지가 ε 보다 작을 때, (10)의 해답을 \mathbf{x}^* 라고 하면 다음 아래의 식을 만족 한다:

$$\|\mathbf{x}^* - \mathbf{x}_0\|_2 \leq C_K \cdot \varepsilon.$$

여기서 C_K 는 δ_{4K} 에 의해서만 결정이 된다. 이는 신호에 잡음이 포함되더라도, L1 최소화를 통해 신호를 원만하게 복구 할 수 있다는 것을 말해 준다.

이 과정을 보다 쉽게 이해하기 위하여, <그림 2>를 살펴보자.

<그림 2>가 보여주는 것처럼, $\mathbf{Ax} + \mathbf{e}_{in} + \mathbf{e}_{out}$ 을 support set 에 대응하는 열 벡터들로 span되는 space로 투사를 시켜서 얻은 $\mathbf{Ax} + \mathbf{e}_{in}$ 은 잡음이 포함되지 않은 \mathbf{Ax} 와 가까운 것을 알 수 있다. 즉 잡음의 에너지가 작으면 L1 최소화는 원하는 해를 근접하게 찾을 수 있다.

다음으로, 신호 \mathbf{x} 가 K -sparse가 아닐 때 L1 최소화의 성능에 대해서 이야기 하겠다. 기본적으로 L1 최소화는 신호 \mathbf{x} 를 K -sparse 신호라고 가정하고 수행한다. 즉 K -sparse 신호가 아니면 K 개의 큰 값을 제외한 나머지 것들을 0이라는 가정하에서 L1 최소화가 이루어 지는 것이다. 그러므로 복구에러에는 K 개의 큰 값들을 제외하고 남은 신호 에너지만 존재한다.

복구 에러를 좀 더 자세히 설명하기 위하여, K -compressible 신호를 정의하자. 신호 x_k 는 신호 x 의 성분들 중에 크기가 큰 K 개의 성분을 유지하고, 나머지 성분들을 0으로 만든 신호이다. 관련된 결과는 다음과 같다. K -compressible 신호란 어떤 non-sparse 신호 \mathbf{x} 가 한정된 $\|\mathbf{x} - \mathbf{x}_K\|_1$ L1 norm 을 가질 경우를 말한다.

K -compressible 신호 \mathbf{x} 에 대해서, $\delta_{3K} + 3\delta_{4K} < 2$ 가 성립이 되고, K -sparse 신호를 신호 \mathbf{x}_K 이고, 잡음 벡터 \mathbf{e} 의 에너지가 ε 보다 작을 때, (10)의 해답을 \mathbf{x}^* 라고 하면

다음 식을 만족 한다:

$$\|\mathbf{x}^* - \mathbf{x}\|_2 \leq C_{1,K} \cdot \varepsilon + C_{2,K} \cdot \frac{\|\mathbf{x} - \mathbf{x}_K\|_1}{\sqrt{K}}.$$

여기서 $C_{1,K}$ 와 $C_{2,K}$ 는 δ_{4K} 에 의해서만 결정이 된다.

위 결과가 말해주는 흥미로운 점은, 앞에서 말한 것과 같이, 복구오차의 크기는 모델링 오차 $\|\mathbf{x} - \mathbf{x}_K\|_1$ 보다 크게 커지지 않는다 라는 결과이다.

지금까지 우리는 III, IV장을 통하여 CS와 관련된 기초 이론들에 대해서 살펴보았다. 또 우리는 최대 상호상관 값에 대해서 정의하였다. 상호상관 값은 임의의 두 열 벡터들 사이의 유사성을 말하는 것으로서, 상호상관 값이 작으면 작을 수록 두 열 벡터들간의 독립성이 강해진다고 말할 수 있다. 그 이후에 L0 최소화의 해가 유일하게 될 조건을 최대 상호상관 값을 이용하여서 표현 하였고 L1 최소화 해가 L0 최소화 해와 같아질 조건 또한 최대 상호상관 값을 이용하여서 표현 하였다. 그리고 나아가서 최대 상호상관 값의 lower bound를 얻었고, 그 것으로부터 필요한 M 의 갯수를 알아보았다. 마지막으로, 신호에 잡음이 끼었을 때와, 신호가 K -sparse하지 않을 때, 그것들이 복구에 어떤 영향을 미치는지에 대하여 이야기 하였다.

이제 다음 장에서는, L1 최소화 해를 얻기 위해 개발된 빠른 알고리즘들을 Homotopy 알고리즘을 중심으로 하여 간략하게 살펴 보겠다.

V. Fast Recovery Algorithms

우리는 III장에서, L0 최소화, L1 최소화, L2 최소화에 대해서 이야기 하였다. 즉, 우리가 선택해야 할 방법이 L1 최소화인 것을 알았다. 여기에서는 구체적으로 어떤 알고리즘으로 L1 최소화 해를 빠르게 찾을 수 있는지 연구해 보기로 한다.

기본적으로 L1 최소화를 풀기 위한 방법은 interior point, exterior point, simplex 등의 일반적으로 잘 알려진 최적화 방법으로 풀 수 있다. 하지만 일반적으로 선형문제 해결법은 N 개의 연립 방정식을 풀어야 함으로 $O(N^3)$ 의 복잡도를 가진다. 즉 신호의 길이가 길다면 L1 최소화의 방법은 많은 시간을 소비해야 하는 문제점이 있다. 따라서, 많은 연구자들이 L1 최소화 문제를 복잡도를 줄이고 더 효율적으로 풀기 위하여 여러 가지 greedy 알고리즘을 제안하였

다. 그 중에는 Basis Pursuit^[11], Homotopy 알고리즘^[6], Least Absolute Shrinkage and Selection Operator [LASSO]^[8], Least Angle Regression (LARS)^[7], Orthogonal Matching Pursuit (OMP) 등이 있다.

이번 장에서는 L1 최소화 문제를 풀기 위한 알고리즘중의 하나인 Homotopy 알고리즘을 중점적으로 소개하고자 한다. 그 이유는 Homotopy를 알면 LASSO, LARS 나 OMP 등의 다른 알고리즘들을 쉽게 이해할 수 있기 때문이기도 하지만 Homotopy 알고리즘이 K -sparse 신호를 K 단계만에 정확하게 찾아낸다고 하는 흥미로운 결과물을 가지고 있기 때문이기도 하다.

1. Homotopy 알고리즘의 의미

Homotopy란 공통 끝점을 가진 두 경로에서 한 경로가 영역 밖으로 나가지 않고 끝점까지 포함해서 다른 한 경로로 연속하여 변형되는 것을 의미한다. 잘 알려진 예로 손잡이가 있는 커피잔과 도넛은 homotopic하다. 다음 두 문제의 해의 경로들은 서로 homotopic하다.

LASSO^[8]: Tibshirani가 제안한 LASSO문제는 다음과 같다:

$$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}\|_1 \leq q. \quad (11)$$

Homotopy^[6]: L1최소화 문제 (5)로부터 다음의 제약조건이 없는 최적화 문제를 생각해 보자:

$$\min_{\mathbf{x}} \frac{\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2}{2} + \lambda \|\mathbf{x}\|_1. \quad (12)$$

즉, L1 penalized least squares의 형태이다. 문제 (11)과 (12)는 서로 homotopic한 문제로 여겨 질 수 있다. (12)에서 $\lambda \in [0, \infty)$ 는 각 단계에서 특정 값을 갖고 \mathbf{x}_λ 는 그에 상응하는 해라 하자. 그러면 λ 에 관한 해집합 $\{\mathbf{x}_\lambda : \lambda \in [0, \infty)\}$ 은 λ 가 아주 클 때는 $\mathbf{x}_\lambda = \mathbf{0}$ 이고 $\lambda \rightarrow 0$ 이면 \mathbf{x}_λ 는 L1 최소화 (5)의 해로 근접하는 해의 변화를 보여준다. 같은 방법으로 (11)에서 $q \in [0, \infty)$ 이고 \mathbf{x}_q 는 해라 하자. 그러면 q 에 관한 해집합 $\{\mathbf{x}_q : q \in [0, \infty)\}$ 은 $q=0$ 이면 $\mathbf{x}_q = \mathbf{0}$ 이고 $q \rightarrow \infty$ 이면 \mathbf{x}_q 가 (5)의 해로 근접하는 해의 변화를 보여준다. 따라서 $q(\lambda) = \|\mathbf{x}_\lambda\|_1$ 라 하면, (11)과 (12)의 해



들은 (5)의 해로 근접하는 변화를 보인다. 여기서 (12)의 해의 변화가 다각형의 형태가 된다는 것이 Osborne, Presnell, 그리고 Turlach^[5]에 의해 밝혀졌다. 해의 변화를 따라가면 0이 아닌 요소의 집합은 λ 에 대하여 구분적으로 (piecewise) 상수 함수이며 λ 의 임계치 (critical value)에서만 변화한다. 이런 0이 아닌 요소의 집합을 active set 이라 한다.

2. 빠른 Homotopy 알고리즘:

이 절에서는, Homotopy를 빠르게 작동시키기 위하여 필요한 조건을 생각해 보고자 한다.

Homotopy는 반복 알고리즘이며, 빠른 Homotopy란 K 단계 반복으로 정답을 찾는 것을 말한다. Donoho는 주어진 신호가 K -sparse 신호이고, 선형 측정 행렬 A 가 incoherent 한 성질을 가지면, Homotopy 알고리즘이 K -단계안에 끝난다는 것을 증명하였다^[6]. 이것은 알고리즘이 정답을 정해진 횟수의 반복만으로 찾아낼 수 있다는 것을 의미하므로 매우 흥미로운 결과이다.

이러한 좋은 성질을 가진 incoherent 행렬 A 는 A 의 열 벡터의 상호상관값을 특정 수치 미만으로 작게함으로써 얻을 수 있다. 벡터간 상호상관값은 쉽게 확인할 수 있으므로 좋은 성질을 가지는 행렬 A 는 어렵지 않게 찾을 수 있다. 그런 선형 측정행렬 A 의 최대 상호상관 값 $\mu(A)$ 와 sparsity K 는 L_1 최소화 해가 L_0 최소화 해와 같은 조건 (8)을 만족해야 한다.

그런 A 를 사용하면, Homotopy 알고리즘은 K 단계만에 끝나고 그 해를 찾는다. 이 성질을 만족하는 행렬 A 를 사용한 Homotopy 알고리즘을 Dohono가 빠른 Homotopy 알고리즘이라 정의 하였다.

이제 빠른 Homotopy 알고리즘을 살펴보자. 식 (12)를 보면 λ 가 아주 크면 최소값은 \mathbf{x} 에 크게 의존하게 되어 해는 $\mathbf{x} = \mathbf{0}$ 이 된다. 하지만 λ 가 점점 작아져서 0에 가까워지면 해는 $\mathbf{y} - \mathbf{Ax}$ 에 크게 영향을 받고 \mathbf{x} 는 0에서 자유로워진다. 이 때 해의 변화 과정에서 (12)의 목적함수 값을 최소화 하며 새로운 해를 찾아간다.

최소화 문제 (12)의 목적함수를 $f_\lambda(\mathbf{x})$ 라고 하자. 특정 λ 에서 $f_\lambda(\mathbf{x})$ 가 \mathbf{x}_λ 에서 최소값을 가지는 필요조건은 우리가 잘 알고있는 것처럼 \mathbf{x}_λ 에 대한 미분이 0이 되는 것이다. 즉, 특정 λ 에서 다음의 미분이 0이 되는 \mathbf{x}_λ 를 찾는 것이다:

$$\partial_{\mathbf{x}_\lambda} f_\lambda(\mathbf{x}_\lambda) = -\mathbf{A}^T(\mathbf{y} - \mathbf{Ax}_\lambda) + \lambda \partial_{\mathbf{x}_\lambda} \|\mathbf{x}_\lambda\|_1. \quad (13)$$

이제 $\mathbf{c} = \mathbf{A}^T(\mathbf{y} - \mathbf{Ax}_\lambda)$ 를 나머지 상관 (residual correlation) 이라 정의하고, \mathbf{x}_λ 의 support를 active set, $I = \{i : \mathbf{x}_\lambda(i) \neq 0\}$, 이라 정의 하자. 그러면 $\|\mathbf{x}_\lambda\|_1$ 은 0에서 미분 가능하지 않으므로 subdifferential을 이용한다. 그러면, $\partial_{\mathbf{x}_\lambda} f_\lambda(\mathbf{x}_\lambda) = \mathbf{0}$ 는 다음의 두 조건으로 나타내어진다:

$$\mathbf{c}(I) = \lambda \cdot \text{sgn}(\mathbf{x}_\lambda(I)), \quad (14)$$

$$|\mathbf{c}(I^c)| \leq \lambda. \quad (15)$$

다시 말하면, active set에 있는 원소의 나머지 상관 값의 크기는 반드시 λ 이어야 하고 그 부호는 대응 되는 \mathbf{x}_λ 의 부호와 같아야 하지만 active set의 원소가 아니면 나머지 상관 값의 크기는 λ 보다 작거나 같아야 한다.

그 이유를 살펴보면 다음과 같다. 어떤 원소의 나머지 상관값이 크다는 것은 그 원소에 대응되는 열 벡터가 \mathbf{y} 를 구성 하는데 큰 역할을 하였다는 것을 의미 한다. 따라서 원래 신호에서 그 원소의 값이 0이 아닐 가능성이 그만큼 크다는 것을 뜻한다.

Homotopy 알고리즘은 최적의 해를 찾아가는 동안 위의 두 조건 (14)과 (15)을 계속해서 만족 하도록 유지한다. 그러면 λ 가 0으로 간다면 나머지 상관값은 0으로 가게된다. 즉, $\mathbf{y} = \mathbf{Ax}_\lambda$ 를 만족하는 해를 찾게 되는 것이다.

알고리즘의 흐름을 대략적으로 살펴보면 다음과 같다. 빠른 Homotopy 알고리즘은 매 단계에서 다음의 과정을 수행한다. (a) 나머지 상관을 구한다. (b) 조건 (14)를 유지하는 방향 벡터를 결정한다. (c) 두 조건 (14)과 (15)을 만족하는 보폭을 결정한다. (d) active set I 와 해 \mathbf{x}_λ 를 업데이트 한다.

이 알고리즘은 초기값을 $\mathbf{x}_0 = \mathbf{0}$ 으로 하고 반복적으로 \mathbf{x}_λ 를 구한다. 이 과정에서 나머지 상관값이 가장 큰 원소를 active set에 넣고 이 active set에 속한 원소에 대응하는 열 벡터들로 구성된 부분행렬의 상호상관값을 이용하여 새로운 방향 벡터를 결정하고 그 방향으로 새로운 해를 찾아 간다. 좀 더 자세히 논하면 다음과 같다. 매 단계에서 방향 벡터 \mathbf{d}_λ 는 다음의 식에서 구한다.

$$\mathbf{A}_I^T \mathbf{A}_I \mathbf{d}_\lambda(I) = \text{sgn}(\mathbf{c}_\lambda(I)). \quad (16)$$

여기서 I 에 있지 않은 요소의 \mathbf{d}_λ 값은 그대로 0으로 둔다. 이렇게 방향 벡터를 정하여 active set에 있는 원소의 나머지 상관값의 크기가 모두 같이 줄어들게 조절 한다. 그리하여 식 (14)의 조건이 자연스럽게 유지된다.

이제 active set의 변화가 있을 다음 해를 결정할 보폭 (step-size)을 결정 하여야 한다. 보폭을 결정 하는 것은 반복 알고리즘에서 가장 중요하면서 어려운 문제 중 하나이다. K 번의 반복으로 해를 찾고 끝내기 위해서는 active set에 있는 원소들이 active set 에서 빠져 나오는 것을 방지 하여야 한다. 또한, 조건 (15)를 어기고 잘못된 해로 수렴 하는 것을 막기 위해 한번에 둘 이상의 원소가 active set 에 들어가게 되는 것도 허용하지 말아야 한다. 그래서 보폭 결정을 위해서는 다음 두 가지로 경우로 나누어 생각하여야 한다.

첫째, active set에 있는 원소는 보폭이 너무 클 경우에는 그 원소에 대응하는 해가 다시 0이 될 수 있다. 그래서 이런 경우가 발생하지 않도록 다음과 같이 보폭을 조절한다:

$$\alpha_\lambda^- = \min_{i \in I} \left\{ \frac{-\mathbf{x}_\lambda(i)}{\mathbf{d}_\lambda(i)} \right\} \quad (17)$$

둘째, active set에 있지않은 원소는 각각 자신이 active set에 들어갈 수 있도록 보폭을 결정한다. 그러면 이들의 보폭 중에서 다음과 같이 가장 작은 값을 택하여 한 번에 하나의 원소만 active set에 들어가게 조절한다.

$$\alpha_\lambda^+ = \min_{i \in I^c} \left\{ \frac{\lambda - \mathbf{c}_\lambda(i)}{1 - \mathbf{a}_i^T \mathbf{A}_I \mathbf{d}_\lambda(I)}, \frac{\lambda + \mathbf{c}_\lambda(i)}{1 + \mathbf{a}_i^T \mathbf{A}_I \mathbf{d}_\lambda(I)} \right\} \quad (18)$$

이 두 개의 보폭 $\alpha_\lambda^-, \alpha_\lambda^+$ 중에 작은 것을 현 단계의 보폭으로 택한다. 이때, active set에 속한 원소의 보폭이 작으면 가장 작은 보폭을 가진 원소는 active set에서 빠져나오게 되고, 그렇지 않으면 active set에 속하지 않고 가장 작은 보폭을 가진 원소가 active set에 속하게 된다. 물론, 보폭은 항상 양수 이어야 한다.

보폭 결정을 요약해 보면 다음과 같다. 매 단계에서 active set에 있는 원소가 빠져 나오지 않고 active set에 있지 않은 원소 중에 하나만 active set에 들어 가도록 보폭을 결정한다.

이렇게 보폭 결정을 함으로서 active set에 들어갈 원소도 결정되었고 이제 해 \mathbf{x}_λ 와 active set I , 그리고 나머지 상관 벡터 \mathbf{c} 를 업데이트 하고 active set에 있는 원소들의 나머지 상관값, 즉 최대 나머지 상관값을 λ 로 하여 다시 반복한다. 이때, 최대 나머지 상관값이 0, $\lambda = 0$, 이면 현재의 해 \mathbf{x}_λ 를 최종 해, 즉, 찾고자 하는 원래의 신호로 여기고 알고리즘을 끝낸다.

3. Homotopy와 다른 알고리즘 비교

이 절에서는 Homotopy로부터 파생될 수 있는 LARS와 OMP 알고리즘을 정리해 보고자 한다.

Least Angle Regression (LARS): III장에서 언급한 LARS 알고리즘은 Homotopy와 아주 유사하다. 다만, LARS는 각 단계에서 새로운 근사해로 이동 할 때 새로운 원소를 active set 에 추가는 하지만, 제거 하지는 않는다. LARS 알고리즘은 Homotopy 알고리즘의 라인 (14) 즉 현재의 해와 나머지 상관의 부호가 일치 해야 한다는 조건을 제거하여 간소화 한 것이다. 하지만 Homotopy 알고리즘이 K -단계 이내에는 해를 찾는다면, 조건 (14)를 제거 한 것이 커다란 간소화 효과를 주지 않는다.

Orthogonal Matching Pursuit (OMP): OMP 는 또한 LARS 알고리즘과 매우 유사하다. 이 두 알고리즘 모두 매 단계에서 새로운 원소를 active set 에 더한다.

이 때 후보 원소군에서 새롭게 active set에 더해질 원소를 결정하는 것은 다르다. 둘 다 새로운 후보 원소와 현재의 residual (오차)의 내적을 줄여주는 sparse한 근사해를 찾는 과정을 반복한다. 즉, A 의 몇 개의 열 들을 잘 선택하여 그 선형조합이 압축된 신호 벡터 \mathbf{y} 로 근접해 가도록 하는 greedy한 방법이다. 이때 선택되어 선형조합에 사용된 열 들이 바로 Homotopy 알고리즘의 active set 원소들에 해당하는 열 들이 된다. 매 반복 단계 마다 한 개의 새로운 열 이 active 집합에 더해지는데 이 열은 그 단계에서 오차 $\mathbf{y} - \mathbf{A}\mathbf{x}$ 와 가장 상관이 잘 맞는 열이다. 그러므로 OMP는 \mathbf{y} 를 구성하는 K 개의 A 의 열 벡터들을 높은 확률로 찾아 낼 수 있다. 하지만 L1 최소화 방법으로는 sparsest 해를 찾을 수 있지만 OMP 방법으로는 sparsest 해를 찾는데 실패하는 경우도 있다.

결론적으로 LARS 알고리즘은 매 단계에서 새로운 근사해로 이동 할 때 새로운 원소를 active 집합에 추가는 하

지만 제거하지는 않는 반면 Homotopy는 매 근사해 단계에서 새로운 원소를 더하거나 빼면서 active set을 갱신한다. 때문에 더해지거나 빠지는 원소가 아주 많으면 Homotopy가 LARS에 비하여 아주 많은 단계를 거쳐야 할 수도 있다. 그러나 조건 (19)을 만족하는 K -sparse 신호 복원 문제에서는 이렇게 더해졌다가 빠지는 순환 현상이 없다. 그러므로 Homotopy는 LARS나 OMP처럼 빠른 알고리즘이 된다.

4. Homotopy 알고리즘의 복잡도

지금부터 Homotopy의 계산 복잡도에 대하여 알아보자. 매 단계에서 방향 벡터를 결정할 때 연립 방정식 (16)을 풀어야 하는데 이를 효율적으로 하기 위해 $A^T A$ 의 Cholesky 분해를 이용하고 active set에 새로운 요소를 첨가하거나 제거하여 이를 업데이트 한다. 이렇게 하면 보통 $O(M^3)$ 의 연산이 필요한 연립방정식 (16)의 해를 $O(M^2)$ 의 연산만으로 구한다. 신호가 sparse하다는 조건을 무시하고 K 단계의 Homotopy 알고리즘에 필요한 계산량은 기껏해야 $4KM^2/3 + KMN + O(KN)$ 이다. 하지만 K 단계 만에 신호를 복원 할 수 있다면 훨씬 더 작은 계산량이 필요하게 된다.

여기서 유의할 점은 신호가 K 단계 신호 복원 성질을 가지면 Homotopy는 K 단계에 sparse 신호를 복원 한다. K 단계 신호 복원 성질을 충족 시키지 못 할 보다 일반적인 경우에는 K 보다 더 많은 단계를 거쳐야 한다.

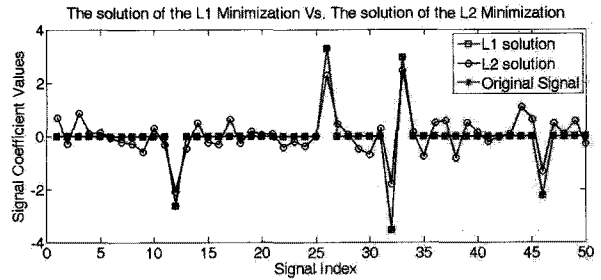
소개한 알고리즘의 계산 복잡도를 표로 만들어 비교해 보면 다음과 같다. <표 1>에서 M, N 은 센싱 행렬의 크기를 나타내고, R 은 두 행렬 Ax 와 $A^T y$ 의 계산량을 나타낸다.

<표 1> Sparse 신호 복구 알고리즘 계산 복잡도 비교

알고리즘	LARS	OMP	빠른 Homotopy
매단계의 계산량	$R + O(K^2)$	$R + O(MN)$	R
최대 반복횟수	제한 없음	K	K

VI. Conclusion

본 논문에서는 현재 학계를 중심으로 주목받고 있는 신호를 동시에 압축하고 취득하는 Compressive Sensing 이론을 살펴 보았다. CS를 크게는 신호를 어떻게 측정해야



<그림 3> L2 최소화의 해는 non-sparse지만, L1 최소화의 해는 sparse인 것을 알 수 있고 원래의 신호와 거의 똑같은 것을 알 수 있다

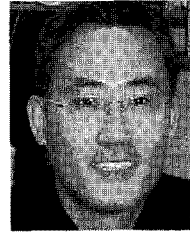
하는 지의 선형 측정 행렬을 디자인을 하는 문제와 빠르고 정확하게 신호를 복원하는 신호복원 문제등 두 가지로 구분하여 보았다. Donoho, Candès, Tao 등이 제안했던 선형 측정 행렬의 좋고 나쁨을 구별하는 방법들과, greedy 알고리즘의 하나인 Homotopy 신호 복원 방법을 통해 여러가지 빠른 신호 복원 알고리즘들을 살펴 보았다.

논문 전체에 걸쳐서 설명하고 있지만, Compressive Sensing은 간단히 말하면 압축가능한 신호를 취득단계에서부터 압축하여 샘플링하는 새로운 신호 취득 방법 및 복원 방법을 말하는 것이라 할 수 있다. Donoho는 [2]에서 "the sampling theorem is wrong; not literally wrong, but psychologically wrong." 이라고 표현할 정도로 CS의 우수성을 강조하고 있으며, 자신의 연구결과에 강한 자신감을 표출하고 있다. 샘플링 이론이 정확한 신호복원을 위해 많은 수의 샘플이 필요할 것이라는 잘못된 부담감을 준다고 주장하고 있다. 그의 이러한 자신감은 오늘에 이르기까지 오랜 세월을 이 분야 발전을 위해 꾸준히 연구해온 학자의 것이어서 예사롭게 보고 넘길수 없다. 본 논문에서는 그러한 이유로, Donoho, Candès, Tao^[1,3,4] 등의 논문에 근거하여 그들이 주창하고 있는 CS 이론의 근거를 살펴보고자 하였다.

참고 문헌

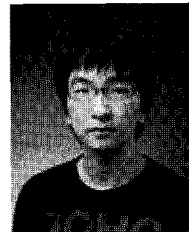
- [1] David Donoho, "Compressed sensing," *IEEE Trans. on Information Theory*, 52(4), pp.1289-1306, April, 2006.
- [2] Donoho, D.L. and Tanner, J., "Precise Undersampling Theorems", *Proceedings of the IEEE*, Vol.98, pp.913-924, May, 2010.
- [3] Emmanuel Candès, Justin Romberg, and Terence Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. on Information Theory*, 52(2) pp. 489-509, February, 2006.

- [4] Emmanuel Candès and Terence Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Trans. on Information Theory*, 52(12), pp.5406-5425 begin_of_the_skype_highlightingend_of_the_skype_highlighting, December, 2006.
- [5] M. R. Osborne, B. Presnell, and B. A. Turlach, "A new approach to variable selection in least squares problems," *IMA J. Numer. Anal.*, Vol.20, pp.389-403, 2000.
- [6] D. L. Dohono and Yaakov Tsaig, "Fast solution of L1-Norm Minimization Problems When the Solution May Be Sparse," *IEEE Trans on Information Theory*, Vol.54, No.11, pp.4789-4812, 2008.
- [7] B. Efron, T. Hastie, I. M. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist.*, Vol.32, No.2, pp.407-499, 2004.
- [8] Rober Tibshirani, "Regression Shrinkage and Selection via the LASSO," *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol.58, Issue 1, pp.267-288, 1996.
- [9] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, Vol.59, No.8, pp.1207-1223, Aug., 2006.
- [10] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. Inform. Theory*, Vol.47, pp.2845-2862, Nov., 2001.
- [11] S.S. Chen, D.L. Donoho, M.A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, Vol.20, No.1, pp.33-61, 1998.



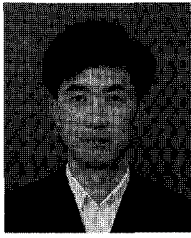
이 흥 노

2009년 1월~Present Associate Professor in School of Information and Communications, GIST, Korea.
 2002년 1월~2008년 12월 Assistant Professor in Department of Electrical and Computer Engineering, the University of Pittsburgh.
 1999년 3월~2001년 11월 Member of Technical Staff of Hughes Research Lab (HRL Laboratory, L.I.C.) in Malibu, California.
 1994년 11월~1999년 11월 Ph.D. in Department of Electrical Engineering, University of California, Los Angeles (UCLA).
 1993년 9월~1994년 11월 M.S. in Department of Electrical Engineering, University of California, Los Angeles (UCLA).
 1988년 9월~1993년 5월 B.S. in Department of Electrical Engineering, University of California, Los Angeles (UCLA).



박 상 준

2005년 2월~2009년 8월 충남대학교 컴퓨터과 학사.
 2009년 9월~현재 광주과학기술원 정보통신공학과 석박 통합과정.



박 순 철

2010년~현재 Research Associate Professor, GIST, Korea.
2006년~2010년 Research Associate Professor, Kyungpook National University, Korea.
2003년~2004년 Research Assistant Professor, University of Florida.
2001년~2003년 Assistant Professor in the Department of Mathematics, the West Texas University, Canyon, Texas.
1999년~2001년 Post-doc. in the Department of Mathematics, University of Florida.
1993년~1999년 Ph.D. in Department of mathematics, University of Florida, Gainesville, USA.
1985년~1992년 B.S. Department of Mathematics, Yonsei University, Korea.