

# 프로그램 합성, 컴퓨터 사용의 패러다임을 바꾸다

김태은

2022.03.22

## Abstract

현재 컴퓨터 사용의 패러다임은 오랜 시간 제자리에 머물고 있다. 하지만 프로그램 합성은 이렇게 정체되어있는 컴퓨터 사용의 패러다임을 바꾸어 줄 것이다. 최근 프로그램 합성 기술은 눈부신 성장을 이루고 있는데, 특히 구문이 이끄는 합성(syntax-guided synthesis), 일명 SyGuS에 있어 놀라운 성과가 있었다. 물론 아직 복잡한 문법의 프로그램은 합성이 어렵지만 우리는 이제 다양하고 간단한 프로그램을 합성할 수 있게 되었다. SyGuS에서 이루어낸 성과와 그 성공 비결을 잘 활용한다면 곧 이 기술은 모두의 것이 될 것이다.

현재 컴퓨터 사용의 패러다임은 약 40년간 정체되어 있다. 1940년대에 에니악과 콜로레스 등 최초의 컴퓨터들이 처음 등장하였을 때, 컴퓨터는 비쌌고, 다루기 어려웠다. 이 당시 컴퓨터는 전문가, 즉 필요에 의해 프로그램을 만들어 쓸 수 있는 사람들의 것이었으며, 그런 사람들은 극히 일부였다. 일반인은 그 혜택을 받지도, 필요를 느끼지도 못했다. 하지만 약 40년이 지난 1980년대에는 컴퓨터가 각 가정에 보급되었고, 컴퓨터의 사용자는 전문가에서 일반인으로 확장되었다. 이러한 일이 가능했던 것은 일반적인 필요를 충족하는 프로그램들이 만들어져 함께 보급되었기 때문이다. 다만, 대부분의 사용자는 컴퓨터 근본적인 기능, 즉, 보편 만능 기계로서의 기능을 온전히 활용하지 못하고, 미리 정해진 기능을 소비하는 수준에 그쳤다. 그리고 다시 40여년이 흘러 2020년대가 되었지만 컴퓨터 사용의 패러다임은 아직 40년 전에 머물러 있다.

프로그램 합성은 이렇게 정체되어있는 컴퓨터 사용의 패러다임을 바꿀 수 있는 열쇠가 될 것이다. 프로그램 합성 기술이 보편화 된다면, 모든 사용자는 이미 만들어진 프로그램을 사용할 뿐 아니라 자신의 필요에 의해 프로그램을 만들어서 사용할 수 있기 때문이다. 즉, 1940년대에 전문가들이 갖추었던 역량을 (적어도 부분적으로) 모두가 갖추게 되어 진정한 의미로 모두를 위한 컴퓨터의 시대가 열리게 되는 것이다.

아직 프로그램 합성 기술이 우리가 원하는 모든 프로그램을 만들어주지는 못하지만, 최근 눈부신 성장을 이루고 있는 것 또한 사실이다. 프로그램 합성은 1940년대에 앨런 튜링이 예견한 이후로 꾸준히 연구되어 온 문제이다. 탐색 기반 프로그램 합성(search-based program synthesis)은 이 문제에 대한 비교적 새로운 접근 방법으로, 논리적으로 표현된 요구 사항과 형태적 제약 사항을 함께 활용하여 수많은 후보 프로그램 중 정답 프로그램을 탐색하는 방법을 통칭한다.

탐색 기반 프로그램 합성 기법 중 유념해서 볼만한 더 구체적인 접근은 구문이 이끄는 합성(syntax-guided synthesis), 일명 SyGuS이다. SyGuS 문제는 다음과 같이 좁은 의미의 프로그램 합성 문제로 정의된다. 미지의 함수  $f$ 를 표현할 수 있는 문맥 무관 문법(context free grammar), 그 문법으로 표현할 수 있는 것들의 집합  $Exp$ , 그리고  $f$ 의 기능을 논리적으로 표현한  $Spec$ 이 주어졌을 때,  $Spec$ 에서 사용되는  $f$ 를 대체할 수 있는 표현  $e$ 를  $Exp$ 에서 찾는 것. 좀 더 간단히 말하자면, 의미적 조건(semantic)과 형태적 조건(syntactic)을 만족시키는 함수  $f$ 를 찾는 것이다.

SyGuS 문제로 접근할 수 있는 프로그램 합성 예시들에는 다양한 것들이 있다. 먼저 논리적으로 표현된 요구사항(logical specification)으로부터 프로그램을 합성하는 문제이다. 여기서 의미적 조건은 논리적으로 표현된 요구사항이며, 형태적 조건은 별도로 정의되는 문법이 된다. 두 번째로는 조금 더 우리 일상에 가까운 문제인 예제 입출력으로부터 프로그램을 합성하는 문제가 있다. 여기서는 예제 입출력이 바로 의미적 조건으로 사용된다. 또한 프로그램 최적화 문제도 SyGuS 문제로 볼 수 있는데, 이 때는 기존 프로그램 자체가 의미적 조건이 된다. 마지막으로 프로그램 검증에 위해 필요한 프로그램의 불변 속성(invariant)을 찾는 것도 SyGuS 문제로 변환하여 풀 수 있는 문제가 된다.

이렇게 간단하지만 다양한 프로그램을 합성하는 것은 중요한 성취이고 큰 의미가 있다. 물론 아직 복잡한 문법의 프로그램은 합성이 어려운 것은 사실이다. 대부분의 범용 언어(general purpose language)에 존재하는 요소인 반복문이나 재귀문은 프로그램 합성이 넘어야 할 큰 산이기 때문이다. 따라서 지금까지의 프로그램 합성 연구는 주로 도메인 특화 언어(domain-specific language)에 적용되어 이루어져왔다. 하지만 도메인 특화 언어로

프로그램을 합성한다고 해서 그 실용성이 없어지는 것은 아니다. 왜냐하면 도메인 특화 언어가 한번 정의되고 나면 사용자는 의미적 조건(주로 입출력 예제)만을 새로이 입력함으로써 새로운 프로그램을 합성할 수 있기 때문이다. 또한 SyGuS 문제는 다양한 프로그램 합성 문제들을 포함한다. 즉, 다양한 도메인에 프로그램 합성을 적용함으로써 프로그램 합성의 유용성을 알리고 더 많은 발전을 꾀할 수 있는 것이다.

SyGuS의 유독 놀라운 발전에서 유념할 또 다른 점은 바로 탄탄한 연구 커뮤니티가 형성되어서 시너지를 발휘했다는 것이다. SyGuS로 정의되는 프로그램 합성 문제는 입력의 형식이 일찍이 표준화 되었고, 수많은 벤치마크가 다양한 영역에서 수집되었으며 매년 대회가 열렸다. 이렇게 판이 깔리자 많은 연구자들이 해당 분야에 손쉽게 진입할 수 있었다. 그리고 그 결과 뛰어난 성능과 실용성을 갖춘 합성기들이 등장하였다. 2004년, DARPA 그랜드 챌린지 대회가 개최된 이후로 전세계의 연구 팀들이 뛰어들어 자동차 자율 주행 기술의 눈부신 발전이 이루어 졌고, 이미지넷(ImageNet)이라는 어마어마한 벤치마크를 통해 인공지능 모델의 성능이 비약적으로 발전했듯이, 세계적인 규모의 대회와 벤치마크는 기술 발전에 있어 필수적이다.

프로그램 합성은 어렵고 전문가의 영역으로만 느껴지지만, 언젠가 이 기술은 모두의 것이 될 것이다. 왜냐하면 프로그램 합성은 모두가 컴퓨터를 온전하게 사용할 수 있도록 하는 열쇠이기 때문이다. 프로그램 합성 기술이 더욱 발전하기 위해서는 먼저 다양한 분야에 프로그램 합성 기술을 도입하여 그 유용성을 알려야 한다. 또한 현재보다 더 큰 규모의 표준 벤치마크가 제작되고 세계적인 규모의 대회가 개최된다면 그 발전은 더욱 눈부실 것이다. 그렇게 된다면 머지않아 프로그램 합성이 우리가 컴퓨터를 사용하는 방식을 바꿔놓게 될 것이다.