

# 「数理科学」は語る

30年前から現代へのメッセージ

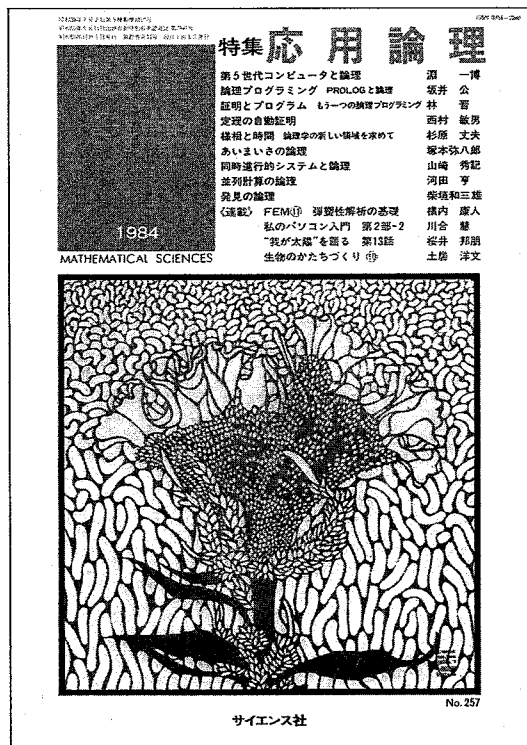
坂井 公

1984年11月号

30年前の本誌の特集は「応用論理」だった。考えてみれば、奇妙な言葉かもしれない。応用××と言う場合、何らかの形でそれが実用的な用途に役に立つというイメージが喚起される。ところが応用論理という場合、それがどういう風な実用性を持ち得るかを想像することは、たやすくならない気がする。もちろん、論理が役に立たないということではない。もろもろの考え方を整理したり、多くの学問で推論を進めたりする上で、論理的であることは不可欠である。しかし、それは、学問などに従事する人は論理的でなければならないということであって、論理学を学んだからといって人は論理的になれるわけではない。

ところが、このときの応用論理は、確かにある実用イメージをもって論理学が語られていたような気がする。おそらく、その背景に当時の日本では第5世代コンピュータ計画というものが進行中であって、その中核技術として論理プログラミングを採用したということがあっただろう。いまさら説明するまでもないが、論理プログラミングとは、論理式の変換を核にして動作するプログラミング原理で、当時はまだ目新しく、確かに論理学を応用している側面があった。当時の筆者の記事を改めて読み返してみると、若かったせいでもういふんと気負いが感じられ、いささか恥ずかしくなる部分もあるが、それまで、中心的だった手続き型や関数型のプログラミングに対峙させるものとしては大変魅力的だったとはいえる。

30年の時間の流れは、予測不可能だったものをたくさん生み出した。情報技術でもそうで、例えば、携帯電話とかインターネットがこんなに普及しようとは全く想像できなかった。しかし、もちろん従来型のコンピュータもこれらを取り込むようにして盛んに使われている。プログラミングの世界でも同じで、論理型が手続き型を席卷するというようなことは起こらず、Cはいまだ健在である。だが、もちろん旧態依然ということでもなく、オブジェクト指向など新しい技術を取



り入れながら進化し続けている。一方、論理プログラミング言語 Prolog もまだ使われているし、その特長であったパターンマッチという機構は、Haskell などの関数型を含めた多くの後継言語の中で重要な技術となっている。

生物の進化もそうだが、このようにして、古いシステムの良いところを残しながら、新しい技術によって次々と小さな変貌を繰り返して行くことで、やがて全く新しいものが生まれてくるのかもしれない。

(さかい・こう、筑波大学数理物質系)