

C. A. R. Hoare : Communicating Sequential Processes

Communications of the ACM 21(8), pp.666-677 (Aug. 1978)

これは並行プログラミングの混沌の世界に Hoare が投じた一石である。頭文字をとって CSP と称されるようになったプログラミング言語の提案であるが、「入出力こそがプログラミングの基本プリミティブであり...」というアブストラクトの書き出しを大学院生のときに読んで、目から鱗が落ちた気分になったのを思い出す。

並行プログラミングは逐次プログラミングより格段に難しいものと信じられている。教科書を開くと、並行プロセス、プロセス間通信、同期などの概念が通過儀礼のように次々登場して、学ぶ者を身構えさせる(実際、E. W. Dijkstra によるこの分野のパイオニア論文“Co-operating Sequential Processes”(1968)はまさに初学者を身構えさせるための論文であった)。これに対して「入出力機能を備えた逐次プロセスを構成要素として並行プログラムが組める」という本論文のスローガンは何とも優しい。単に表現が優しいだけでなく、共有メモリによる情報交換からメッセージ通信による情報交換へのパラダイムシフトを、誰にでも分かる具体的なかたちで明快に示している。多くのプログラム例を用いて CSP の単純さと表現力を紹介すると同時に、プログラミング言語設計に対する氏の哲学を、構文設計の細部に至るまで存分に披露している。最近のプログラミング言語の論文ではあまり見るのできなくなってしまったアプローチと構成である。

CSP は正確にはプログラミング言語ではなくて言語モデルと呼ぶべきかもしれないが、CSP をベースに実用並列言語に仕立てあげられたのが Occam である。オッカムの剃刀に由来するその名称は、CSP の設計哲学の強烈な対外アピールともなっている。Occam のプラットフォームとして作られた並列処理マイクロプロセッサ Transputer のアーキテクチャもまた、CSP の思想の1つの体现である。

CSP は静的な資源割付けを可能にするために、再帰呼び出しや動的データ構造、非同期通信などの動的要素を排除した質素な造りとなっている。世の中のもの後の研究は、通信路の再構成やプロセスの移動が可能な言語に

興味移っていったが、25年たった今の技術をもってすれば、再帰呼び出しも動的データ構造も非同期通信も備え(て計算機科学者の欲求を満たし)た上で、広い条件の下で静的な資源割付けもできる並列言語が作れるような気がする。最近のハードウェア環境の多様化に伴って、このようなかたちでの CSP への回帰は挑戦する価値の大きなテーマかもしれない。

“Communicating Sequential Processes”と題する著作物には、標題の論文のほかに1985年刊行の同名の著書(Prentice-Hall)がある。Hoare の最初の著書でもあるこの本では、CACM の論文とはうって変わって並行計算の基礎をなす数学的理論が展開されている。並行性という難しい事象にきちんと立ち向かうには数学という道具が必須となることは論を待たないが、あまりに難しい数学になったら誰も近づかない。しかし Hoare は、決定性プロセスの理論ならば大学2年生に教えることもできるとまがきで述べている。本書は現在では無料でダウンロードできるので読んでみるのもよいであろう。ちなみに CiteSeer によれば、この本と CACM の論文とを合わせると“3rd most cited computer science reference”となるそうである。

著書の方で展開された数学的理論としての CSP (Theoretical CSP) とよく並び称されるのが、Robin Milner の CCS (Calculus of Communicating Systems) とその発展である π 計算である。理論計算機科学の分野では Milner の枠組みの方がはるかによく研究されているが、ここには両者の立場の違いが見て取れる。Milner はいわば、並行計算の数学モデルを検討するための「枠組み」を作って、他の研究者がその中でさまざまな検討を行って論文を書くことができるようにした。Hoare は、並行計算の数学モデルはどうあるべきかを洞察し、完成された「解」を1985年の著書で呈示した。企業のソフトウェア技術者という経歴を持つ Hoare のさまざまな仕事には、実務経験を持つエンジニアの観点と問題意識がにじみ出ているように思える。

(平成16年11月21日受付)

上田和紀 / 早稲田大学理工学部コンピュータ・ネットワーク工学科
ueda@ueda.info.waseda.ac.jp