A Wait-Free Classification of Loop Agreement Tasks

Sergio Rajsbaum
IMATE–UNAM

(joint work with Maurice Herlihy)

A *task* is a distributed coordination problem in which each process starts with a private input value taken from a finite set, communicates with the other processes by applying operations to shared objects, and eventually halts with a private output value, also taken from a finite set. Examples of important, previously studied tasks include *consensus*, *renaming*, and *set agreement*.

A *protocol* is a distributed program (i.e., a set of programs, one for each process) that solves a task. A protocol is *wait–free* if it tolerates failures or delays by n out of $n+1$ processes. One task *implements* another if one can transform any protocol for one into a protocol for the other.

Loop agreement is a family of wait–free tasks that includes set agreement and approximate agreement tasks. This paper presents a complete classification of loop agreement tasks. Each loop agreement task can be assigned an *algebraic signature* consisting of a finitely-presented group G and a distinguished element g in G.

This signature completely characterizes the task's computational power. If two loop agreement tasks T_1, T_2 have respective signatures $\langle G, g\rangle$ and $\langle H, h\rangle$, then T_1 implements T_2 if and only if there exists a group homomorphism $\phi : G \to H$ carrying g to h.