Abstract: Alchemists wanted to create gold, Hilbert wanted an algorithm to solve Diophantine equations, researchers want to make deep learning robust in AI, MATLAB wants (but fails) to detect when it provides wrong solutions to linear programs, etc. Why do we fail in so many of these fundamental cases? The reason is typically methodological barriers. The history of science is full of methodological barriers – reasons for why we never succeed in reaching certain goals. In many cases, this is due to barriers in the foundations of mathematics. This talk introduces new such barriers from foundations: the phenomenon of generalised hardness of approximation (GHA). GHA grows out of our solution to the extended 9th problem from Smale's list of mathematical problems for the 21st century. This phenomenon is not a rare issue – but happens on a daily basis in numerical analysis – and causes modern software such as MATLAB to fail on basic problems, and even certify nonsensical solutions as correct.

GHA is close in spirit to hardness of approximation (HA) in computer science. Assuming P/= NP, HA is the phenomenon that one can easily compute an epsilon- approximation to the solution of a discrete computational problem for $\varepsilon > \varepsilon_0 > 0$, but for $\varepsilon < \varepsilon_0$ it suddenly becomes intractable. HA was discovered decades ago and has been transformative being the subject of several Goedel, Nevanlinna and ACM prizes. GHA is a similar but distinct mathematical phenomenon that requires a new set of mathematical tools in numerical analysis rather than computer science (NB: GHA is independent of P vs. NP). The GHA phenomenon has so far been detected in optimi- sation, inverse problems, deep learning and AI, as well as computer-assisted proofs. It is essential in the Solvability Complexity Index (SCI) hierarchy and for understanding "why things don't work", as well as a key tool to understand "why things sometimes work".