Proximal envelopes for nonconvex splitting algorithms:
Block coordinate and Newton-type variants

Professor Panagiotis (Panos) Patrinos

Department of Electrical Engineering, KU Leuven, Belgium

Monday 3 February 2020
2.00pm
MR4, IEB

Abstract:

The classical understanding of splitting algorithms hinges on monotone operator theory, and thus heavily hinges on convexity. Alternatively, when applying these schemes to nonconvex problems convergence can be shown once a suitable merit function is identified; in other words, a function that decreases along the generated iterates.
The challenge of convergence analysis of splitting algorithms in the nonconvex setting is thus the identification of a suitable merit function, and so far there doesn't seem to be a clear way as to how to construct one, let alone for randomized and/or block-coordinate variants. As a result, for many algorithms it is still unclear whether or not their application to nonconvex problems is feasible.
In this talk we show how proximal envelopes provide a positive answer to this challenge, as they serve as suitable merit functions for many splitting algorithms.
Moreover, thanks to their regularity properties they enable the possibility to robustify splitting algorithms by means of second-order-type information, stemming for instance from quasi-Newton schemes, without affecting global convergence.
Block-coordinate (BC) variants of forward-backward splitting are also investigated for the minimization of the sum of a separable smooth function and a (nonseparable) nonsmooth function, both of which allowed to be nonconvex. Differently from classical study cases where it is the nonsmooth function to be separable, the cost cannot serve as merit function and as a result this setting is only little known. Once again we show how the "forward-backward envelope" serves as the suitable merit function, providing new convergence result for a large class of BC-type algorithms for nonsmooth and nonconvex problems with rather general sampling strategies, and that include the popular Finito/MISO algorithm as a special case.