Six Thinking Hats Heuristic Methodology: Towards a Fuzzy-Logic-Based Explanation

Karen Villaverde  
Department of Computer Science  
New Mexico State University  
kvillave@nmsu.edu

Vladik Kreinovich  
Department of Computer Science  
University of Texas at El Paso  
vladik@utep.edu

In his widely used Six Thinking Hats book, Eduardo de Bono proposed heuristic for enhancing reasoning and decision making by separating different reasoning modes ("hats"). In this talk, we show that his heuristics can be (at least partly) explained in terms of fuzzy logic, a technique in which we explicitly associate a degree of confidence (usually, a number from the interval [0,1]) with each fact and, more generally, each statement from our knowledge base.

Specifically, ideally, it is desirable to come up with a solution by only using facts that statements about which we are absolutely confident; this corresponds to white hat. If such absolutely confident knowledge is not sufficient, we need to also take into account less confident knowledge -- this corresponds to red hat. In this reasoning, we can use known solutions s to similar problems -- by assuming that the desired solution to the actual problem is close to s.

Sometimes, the red hat reasoning leads to success, but sometimes, by incorporating less confident statements, we get confused since now, for different proposed solutions, we have both arguments for and against the proposal to be an actual solution. In this case, instead of trying to combine all the facts and statements, it may make sense to separately consider statements and fact which are "for" and the ones which are "against" -- this corresponds to yellow and black hats -- and try to reconcile the results.

Sometimes, this still does not work. In this case, instead of looking for solutions which are close to the known solution s of a similar problem, we can try to looks for solutions which are explicitly required to be different from s. We can explicitly add a requirement that the solution is different from s, or we can randomly perturb s and require that the actual solution is close to the result s' of this perturbation. This corresponds to the green hat.

Finally, we need to decide when we give up on one "hat" and move to the next one -- and/or when we go back to the original "hat". The corresponding meta-strategy is what is called blue hat.

Reference: