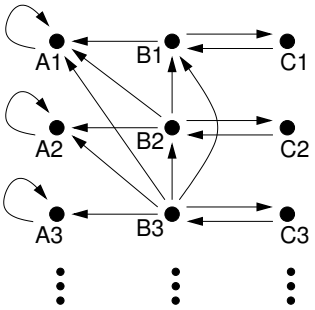
SemBuster: a benchmark example for semi-stable semantics

Martin Caminada[1] and Bart Verheij[2]

[1]Prifysgol Caerdydd
[2]Rijksuniversiteit Groningen



The *sembuster* example in its current form (as illustrated on the right) has been taken from [1], although it originates from work that was done in the context of DefLog [2, 3]. The idea is that there is a set of arguments $\{A_1, \ldots, A_n\} \cup \{B_1, \ldots, B_n\} \cup \{C_1, \ldots, C_n\}$ where n is some positive integer. Each A_i attacks itself. Each C_i attacks B_i, and each B_i attacks A_i and C_i as well as each A_j and B_j with $j < i$.

When considered from the perspective of argument labellings under complete semantics, it becomes clear that at most one B-argument can be labelled `in`. Hence, the sembuster example has $n + 1$ complete labellings. The first labelling labels B_1 `in` and all the other B-argument `out` (this implies that C_1 is labelled `out` and all other C-arguments are labelled `in`, and that A_1 is labelled `out` and all the other A-arguments are labelled `undec`). The second labelling labels B_2 `in` and all the other B-arguments `out` (this implies that C_2 is labelled `out` and all other C-arguments are labelled `in`, and that A_1 and A_2 are labelled `out` and all the other A-arguments are labelled `undec`), etc. In this way, n complete labellings can be constructed. An additional complete labelling can be constructed by labelling each B-argument `out` (this implies that each C-argument is labelled `in` and each A-argument is labelled `undec`). This brings the total number of complete labellings to $n + 1$.

In the sembuster example, each complete labelling is also a preferred labelling. However, only one complete labelling (the one that labels B_n `in`) is a semi-stable labelling.[1] When n is relatively big, this means that there are many preferred labellings, but still only one semi-stable labelling. This shows that finding a semi-stable labelling by first trying to find all preferred labellings might not be an efficient approach. Solvers that try to do so can be expected to show a really bad performance on the sembuster example.

Relevant solver questions: (1) give one semi-stable labelling, (2) give all semi-stable labellings, (3) is argument B_n labelled `in` by one (or all) semi-stable labellings (answer: yes), and (4) is argument B_j ($j < n$) labelled `in` by one (or all) semi-stable labellings (answer: no).

References

[1] M.W.A. Caminada and B. Verheij. On the existence of semi-stable extensions. In G. Danoy, M. Seredynski, R. Booth, B. Gateau, I. Jars, and D. Khadraoui, editors, *Proceedings of the 22nd Benelux Conference on Artificial Intelligence*, 2010.

[2] B. Verheij. DefLog - a logic of dialectical justification and defeat. Technical report, Department of Metajuridica, Universiteit Maastricht, 2000.

[3] B. Verheij. DefLog: on the logical interpretation of prima facie justified assumptions. *Journal of Logic and Computation*, 13:319–346, 2003.

[1]In [1] $n = \infty$, which would lead to an infinite argumentation framework without semi-stable labellings.