**BLEVE loading prediction via multi-task learning**

# Problem definition

Boiling Liquid and Expanding Vapour Explosion (BLEVE) is an extreme blast event which can cause catastrophic consequences, imposing enormous threats to surrounding structures and personnel. Prediction of BLEVE blast loading is not feasible using simple tools. Current practice are often based on computational fluid dynamics (CFD), which requires profound expertise and are computational expensive to run, e.g., the simulation of a single BLEVE case can take days and weeks, if realistic environmental geometry is considered. Machine learning provides effective and efficient alternatives to mitigate this gap. The goal of this study is to develop machine learning approaches for BLEVE loading prediction.

# Data

All data are provided in the excel file “BLEVE\_Open\_500Butan\_500Propane\_5-40m\_8outputs.xlsx”, which contains 8 spreadsheets each of which is data for one particular task. The sheet name should be self-explained, e.g., “positive\_peak\_time” means the data for the time instance when positive pressure reaches its peak. Each spreadsheet is like a tabular dataset of 1001 rows (1000 simulation cases plus header) and 47 columns (11 features including id and 36 outputs). The 36 outputs are the readings of the same quantity, e.g., positive\_peak\_time, from the same simulation case but at different standoff distances (5 meters to 40 meters). So you can verify data by the fact that the positive peak time should increase along with the standoff distances. There are different ways to deal with these 36 readings and what I did is to separate them as 36 data points. Except the 36 output columns, the feature columns should be the same for all 8 sheets (tasks).

# Resources

It’s a good idea to find an existing Github repository as the starting point of coding. For MTL, the best I found so far is <https://github.com/median-research-group/LibMTL>. Note that it is designed for computer vision and therefore you need to rewrite at least the data loader part.

I also have some code for manipulating the excel, such as breaking each simulation to 36 data points, and pass it to you if needed. I have also used other libraries such as <https://github.com/Yura52/tabular-dl-revisiting-models> to develop single-task learning models for tabular data.

You can easily find many research papers about MTL and I suggest you start with some review papers such as:

*Zhang, Yu, and Qiang Yang. "A survey on multi-task learning." IEEE Transactions on Knowledge and Data Engineering 34.12 (2021): 5586-5609.*

*Ruder, Sebastian. "An overview of multi-task learning in deep neural networks." arXiv preprint arXiv:1706.05098 (2017).*