Optimisation in industrial applications: disaster management and signal processing

Nadia Sukhorukova, Behrooz Bodaghi, Julien Ugon, Zahra Roshan Zamir and Aiden Fontes

Swinburne University of Technology and Federation University Australia

Resource allocation: disaster management and staff shift patterns

Disaster management

Any disaster response situation with scarce resources has to be examined in order to coordinate several teams dealing with expandable and non-expendable resources. There is a given processing time for each incident once the relief operation for particular point starts. The processing time varies for each incident and depends on each team: each team has a unique processing and transportation time to respond to each incident.

The objective function is the total weighted completion times overall incidents. The weighted factor depends on the severity level of damage and the total number of casualties that require relief on each incident. Hence, synchronization of the teams with expandable and non-expendable resources during the disaster response is required to lessen the incident's completion time and delay on the relief required on each incident.

The Figure represents a map where 17 public hospitals (incident points, each point has a specified demand) have to be supplied from 2 airports (processing centres, each centre has a specified capacity). For each connection (processing centre-incident point) we assign a cost (total processing and transportation time). The goal is to minimise the total transportation time subject to the capacity constraints of each processing centre. We are currently running experiments on larger size problems.

One way to deal with this problem is to reformulate it as a mixed-integer linear problem. In general, it is not very easy to work with integer and mixed-integer problems when the number of incident points and/or processing centres is increasing. One way to reduce the size of the problem is to implement various clustering techniques and develop approximation models that avoid integer variables.

We have noticed that the problem of allocating expandable resources is equivalent to a well-known type of linear programming problems called Transportation Problems. It is enough to think about incident points as "Factories" (each factory capacity corresponds to the corresponding incident point demand), while the processing centres are "Markets" (each market demand corresponds to the processing centre capacity). The transportation costs are "total processing and transportation time".

There are a number of efficient methods for solving Transportation Problems. One approach is based on formulation of the problem as a linear relaxations (that is, we do not require the solution to be integer). In practical situation, the solution has to be integer, since we can not transfer a fractional number of doctors, nurses, patients, etc. It is well-known, however, that the Simplex Method applied to a Transportation Problem terminates at an integer optimal solution (there may be several optimal solutions). Therefore, we can reduce a mixed-integer linear programming problem to a linear programming problem and also found a way to obtain an optimal integer solution.

In the case of non-expandable resources, the problem can also be formulated as an integer programming problem, where some of the summations from a classical transportation problem constraints are replaced with maximisation. This problem is not a transportation problem, but it can be demonstrated that the applications of the Simplex Method also leads to an integer optimal solution. It can also be shown that this approach leads to other types of problems whose optimal solutions reached by the Simplex Method are guaranteed to be integer.

THEOREM If in a feasible linear programming problem with equality constraints the right-hand-side of the constraints are integers and all the components of the constraint matrix are from {0,1,-1} then the Simplex Method applied to this problems leads to an integer optimal solution.

There are other types of practical problems where the conditions of the above theorem hold. These problems are not limited to disaster management problems.

Shift patterns optimisation

This problem can be seen as a further refinement of disaster management problem, resource allocation problem or as an independent problem. Supposed that a specified workload has to be covered by permanent staff members (lower rates, but have to be paid even when there is no job to be done) and casual (higher rates, but can be employed only for short periods). Permanent contracts do not have to be Monday to Friday, 8 hours every day, but they have to satisfy certain constraints: safety requirements, union agreement, etc.

Given the workload (one year in advance) and possible shift patters, find the number of employees for each contract type that minimises the expenses.

Several scenarios have been suggested. In all these scenarios the problem was formulated as a convex optimisation problem and solved using convex optimisation tools. The results have been published in:

***Sukhorukova*, N., *Ugon*, J., & *Yearwood*, J. (2009). *Workload* coverage through nonsmooth *optimization*. *Optimization Methods & Software*, 24, 285-298.**

Example: six types of contract
•Type 1 : 5 days a week, 8 hours a day;
•Type 2 : 3 days a week (Tue-Thu) 8 hours a day (part time at 60%);
•Type 3 : 3 days a week (Tue-Thu) 13 hours a day Wed-Thu, 14 hours on Tue;
•Type 4 : 4 days a week (Mon-Thu) 10 hours a day;
•Type 5 : 4 days a week (Tue-Fri) 10 hours a day;
•Type 6, which is "Weekend workers".

Figure 3 represents a pattern of one week duration:
•Black dots represent the total workload on a particular day;
•Blue rectangle represents Type 1 employment contract: 5 days a week, 8 hours a day;
•Light purple rectangle represents Type 2 employment contract: 3 days a week (Tue-Thu) 8 hours a day;
•Red rectangle represents Type 3 employment contract: 3 days a week (Tue-Thu) 13 hours a day Wed-Thu, 14 hours on Tue;
•Green rectangle represents Type 4 employment contract: 4 days a week (Mon-Thu) 10 hours a day;
•Dark purple rectangles represent Type 5 employment contract: 4 days a week (Tue-Fri) 10 hours a day;
•Yellow rectangles represent Type 6 employment contract or "Weekend workers".

On the current example (Figure 1) Monday and Friday are the days when additional resources are needed for workload covering. For these days additional subcontractor will have to be employed.
On Tuesday, Wednesday and Thursday the situation is different: the actual workload is lower than the resources of the permanent staff.
The dark curve represents an approximation of the workload which can be done by permanent staff (all types of employment).

Figure 2 represents modified levels, after changing the size of the permanent staff (different change for different types of employment):
•Type 1 employment contract: decreased;
•Type 2 employment contract: decreased;
•Type 3 employment contract: unchanged;
•Type 4 employment contract: increased;
•Type 5 employment contract: unchanged;
•Type 6 employment contract: unchanged.

Manually changing the size of the permanent staff, it is possible to choose a better size for the permanent staff level.

The salary savings are around 10%.

FIGURE 1 Original levels of workload (week pattern)

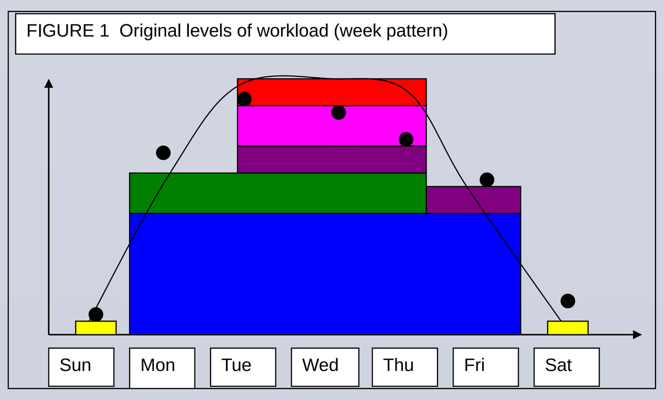
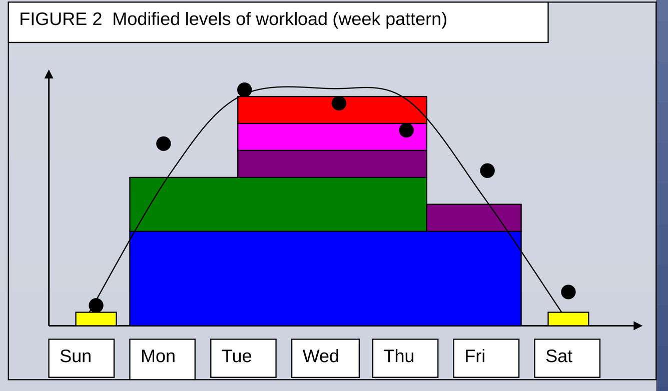


FIGURE 2 Modified levels of workload (week pattern)



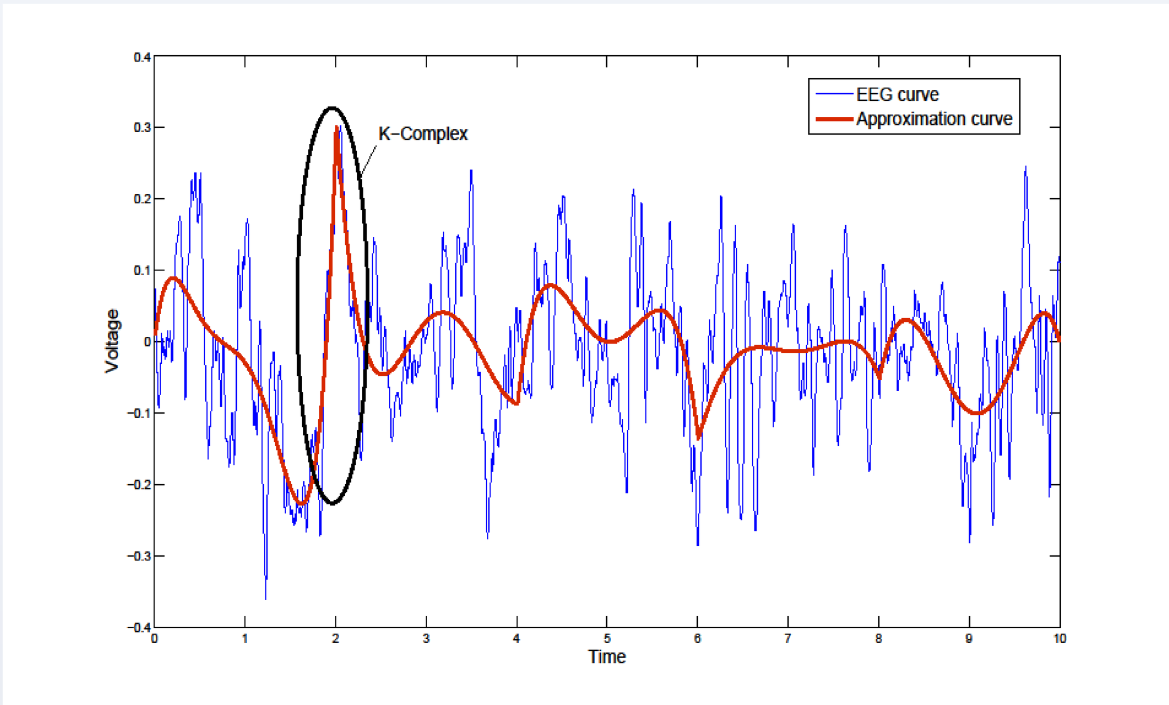
Signal processing: classification and clustering

Signal classification

K-complex is a special type of electroencephalogram (EEG, brain activity) waveform that is used in sleep stage scoring. An automated detection of K-complexes is a desirable component of sleep stage monitoring.

This automation is difficult due to the ambiguity of the scoring rules, complexity and extreme size of data. We develop several convex optimisation models that extract key features of EEG signals. These features are crucial for detecting K-complexes. Essentially, our models are based on approximation of the original signals by sine functions with piecewise polynomial amplitudes. Then, we apply standard classification tools (from Weka) to the corresponding approximations rather than raw data to test the presence of K-complexes.

The proposed approach significantly reduces the dimension of the classification problem (by extracting essential features) and the computational time while the classification accuracy is improved. Numerical results show that these models are efficient for detecting K-complexes.



Approximations
W1 = Sm(x1;t) sin($\omega t + \varphi$)
W2 = Sm(x1;t) sin($\omega t + \varphi$) + Sm(x2;t),
where Sm(x1;t) and Sm(x2;t) are piecewise polynomial functions of degree m, whose parameters x1 and x2 are subject to optimisation.
W1 and W2 approximations were used in Least Squares models (LLSOM1 and LLSOM2) and uniform approximation models (UOM1 and UOM2)

Algorithm.
1: Specify the initial and final values for the frequency ($\omega 0$ and ωN) and shift ($\varphi 0$ and φN)
2: for $\omega = \omega 0 : \omega N$ do
3: for $\varphi = \varphi 0 : \varphi N$ do
4: Solve the corresponding optimisation problem with fixed ω and φ; and record the minimal value of the objective function.
5: end for
6: end for

Test set accuracy (K-complex detection) for raw and preprocessed signals

Accuracy on raw signals	Accuracy on preprocessed signals				Classifiers
	LLSOM1	LLSOM2	UOM1	UOM2	
47%	47%	47%	47%	47%	LibSVM
N/A	74%	68%	52%	63%	Logistic
74%	63%	74%	68%	53%	RBF
47%	63%	84%	47%	53%	SMO
53%	63%	74%	53%	79%	LazyIB1
53%	74%	79%	47%	69%	LazyIB5
47%	74%	74%	47%	69%	KStar
47%	74%	74%	47%	79%	LWL
37%	47%	47%	47%	47%	OneR
47%	74%	79%	47%	74%	J48
47%	74%	79%	47%	74%	J48graft
42%	74%	74%	53%	79%	LMT

Comments
1. N/A means Weka failed to produce classification results due to memory problems.
2. From the accuracy table one can see that the application of classifiers to approximations improved the classification accuracy, but also these approximation were universally "helpful", that is, improved the accuracy of almost ALL the classifiers,

References

1. Optimization-based features extraction for K-complex detectionZR Zamir, N Sukhorukova, H Amiel, A Ugon, C Philippe, ANZIAM Journal 55, 384-398, 2014
2. Convex optimisation-based methods for K-complex detection ZR Zamir, N Sukhorukova, H Amiel, A Ugon, C Philippe Applied Mathematics and Computation 268, 947-956, 2015
3. Linear least squares problems involving fixed knots polynomial splines and their singularity study ZR Zamir, N Sukhorukova, Applied Mathematics and Computation 282, 204-215
4. Detection of epileptic seizure in EEG signals using linear least squares preprocessing ZR Zamir, Computer methods and programs in biomedicine 133, 95-109

Signal clustering

In signal processing, there is a need for constructing signal prototypes. Signal prototypes are summary curves that may replace the whole group of signal segments, where the signals are believed to be similar to each other. Signal prototypes may be used for characterising the structure of the signal segments and also for reducing the amount of information to be stored. Any signal group prototype should be an accurate approximation for each member of the group. On the top of this, it is desirable that the process of recomputing group prototypes, when new group members are available, is not computationally expensive.

We suggest a k-means and least square approximation based model. This is a convex optimisation problem. There are several advantages of this model. First of all, it provides an accurate approximation to the group of signals. Second, this problem can be obtained as a solution to a linear system and can be solved efficiently. Finally, the proposed approach allows one to compute prototype updates without recomputing from scratch.

Assume that there is a group of l signals $S_1(t), \dots, S_l(t)$, whose values are measured at discrete time moments $t_1, \dots t_N, t_i \in [a,b], i = 1, \dots, N.$
We suggest to construct the prototype as a polynomial of degree n, whose least squares deviation from each member of the group on [a;b] is minimal. That is, one has to solve the following optimisation problem:

minimise $F(x) = \|Y - BX\|$
where
$X \in R^{n+1}$ is the vector of decision variables (that is, polynomial parameters), Y is the vector of signals and B is a constant matrix, that contains l identical blocks.

We propose a fast and efficient algorithm for K-means signal clustering and cluster prototype recalculation when several signals move from one cluster to another.

Reference:
N Sukhorukova and J. Ugon, Schur functions for approximation problems, submitted to MATRIX2018. Available on Arxiv as well.