Kaban: a tool for analysis of railway capacity

J. Ekman
SICS AB, Sweden

Abstract

The capacity analysis tool Kaban aims at being efficient for examining if planned infrastructure meets the expected need for railway capacity. Infrastructure, safety rules, signalling and traffic are all modelled in great detail in Kaban and hence the tool is a useful support for signalling design. The tool is also useful for finding out which routing and what train order suits existing or planned track layout. The idea of Kaban is that traffic patterns can be modelled as discrete event systems, with the minimum cycle time as a capacity measure. This measure answers the question if a certain timetable is possible at a station and tells how much buffer time there is. Kaban also presents results on what is critical for the capacity, aiming at explaining how to adjust to increase capacity. The GUI of Kaban displays the infrastructure and train paths and takes care of the user interaction. The development of Kaban is supported by the Swedish Transport Administration (Trafikverket).

1 Introduction

Kaban aims at being a user-friendly capacity analysis tool, efficient to work with and producing relevant and dependable results. Kaban aims in particular at supporting signalling design, but the tool is useful also for other phases of building and utilising the railway. The computations of capacity in Kaban are analytic. The tool is based on the methods presented in [1] and [2].

1.1 The goal of capacity analysis

In this paper the point of view is that analysis of railway capacity is conducted with the aim of supporting decisions on how to most efficiently build and utilise

railway infrastructure. The basis for the development of the capacity analysis tool Kaban is that

- capacity analysis results of high quality make a substantial difference for the efficiency in building and utilising the railway; and
- there is a lot to gain by more efficiently building and utilising the railway.

To build railway infrastructure that efficiently meets the present and future traffic demands and to efficiently utilise the existing railway network are difficult tasks. Nevertheless, those are important tasks and the efficiency of the railway industry relies on how well those tasks are performed. In the railway building phase it is important with analysis results feasible for:

- suggesting how to adjust the infrastructure to increase performance
- making clear advantages and drawbacks of infrastructure proposals
- making a decision on an infrastructure investment.

1.2 Capacity analysis efficiency

Reducing the time and work to answer specific analysis questions is an important step to obtain capacity analysis of higher overall quality. It enables for the analysis to cover a larger number of cases of infrastructure and traffic. Thereby favouring a more complete understanding of the infrastructure utilisation possibilities. The result is that advantages and drawbacks of different proposals are better understood. Possibly the risk of mistakes or unforeseen consequences are also reduced. In this context a capacity analysis tool needs to be supportive for entering data. Since a lot of data is needed for analysing capacity it is often the case that entering all this data is time consuming. This is especially the case if infrastructure and train conflicts are considered at a detailed level.

Transparency of the capacity analysis is helpful for understanding and trusting the results. Such a transparency is provided by presenting how the results are derived. If computations are presented as divided in parts, where each part is conceived as reasonable, then the result can be trusted. A benefit of capacity analysis of high quality in the building phase is the potential to use those results for track utilisation at stations and for planning traffic.

2 Capacity analysis in Kaban

2.1 Basic ideas

Kaban is based on analytic methods for estimating capacity, as presented in [2]. That is, Kaban is not a simulation tool. The models of railway operation of Kaban are static and not probabilistic. Kaban is designed for analysing capacity at complex stations with a lot of conflicting movements. These are cases where it is crucial how safety rules are interpreted and implemented in the interlocking system.

Kaban implements the Swedish safety rules in detail and offers the possibility to specify certain kinds of deviations from the rules.

The static and analytic character of Kaban makes it in a sense model traffic in an ideal world. Some results can be thought of as theoretical maximum capacity, to be used as measures for comparing different solutions. Other results, to become realistic, needs to be calibrated by asserting buffer times. There are also results, though, that has a static nature of their own. Static are some requirements on vehicle positions in the moment a train route is locked. Such requirements are capacity results in their own, they are significant as they are the basis for other capacity analysis results. The advantages with Kaban's models of railway operation are that:

- the assumptions are precisely defined and we know exactly what the results mean
- fast computations are made possible and this allows for considering large number of infrastructure proposals and modifications within available time
- precise measures of capacity are provided for measuring the benefits of infrastructure modifications, comparing infrastructure proposals and for deciding on which proposal to choose.

It is possible to further develop Kabans, still being analytic, to become a tool based on probabilistic models. For instance using probabilistic models for running times. But such a development may be hard to enable without the drawback of slowing down the computations. A development more likely to be advantageous is the search for the minimum cycle time for all traffic pattern satisfying some specification.

2.2 The concepts waiting point and cleared path

The essential concepts on which the method of Kaban relies are *waiting point* and *cleared path*. Detailed description and exemplification are given also in [2]. A waiting point is a point at which a trains stops to wait for prior trains to pass. Waiting points of trains are input to the capacity analysis. A waiting points is associated with the end of a train route. A train path is divided into parts called *cleared paths* by the waiting points. A train path here means the path on which the train runs through the analysed railway section. Cleared paths shall not to be confused with train routes and waiting points may not be confused with stops in general.

2.3 Conflicts and cycle time computation

The model of train operation of Kaban is a discrete event system (see [3]). As events we chose starting times of movements on cleared paths. In the continuation we will just use movements to refer to those movements on cleared paths. Considering two events separately from all other events the following principles

will assure that the minimum duration in between two events is uniquely defined

- a train moves unhindered by other trains on each of its cleared paths
- stop durations are fixed, but waiting for other trains at waiting points is not included in those durations
- it is uniquely determined where and how two trains are in conflict.

Notice that the first two points uniquely determine the time from the start of a movement to any point along the way to the end. The last point gives what in addition is needed to know for uniquely determining the minimum duration between events. Another way to formulate the first point is to say that a movement may start no earlier than that it will be unhindered by prior movements. This means that the speed of trains is not at all affected by the occurrence of other trains. The interaction between trains is manifested as conditions on time for trains to enter the analysed railway section and time to wait at waiting points. The first point does not mean that all of the train routes of a cleared path must be locked before the movement on the cleared path starts.

As is shown in [2] max-plus algebra provides notation for defining and characterising the cycle time of a traffic pattern in Kaban. The cycle time of a traffic pattern in Kaban is expressed as the *minimum cycle mean* for a discrete event system. This suits the use of Howard's algorithm [4], a fast algorithm for computing the minimum cycle mean. The approach to model train operation as a discrete event system and to use max-plus algebra to express cycle time is well studied, see for instance [5]. Although such approaches are basically the same, they may differ in what the analysis aims at and precisely how the modelling is done.

2.4 An example of a cycle time computation in Kaban

The figures 1 and 2 show the drawing area of the Kaban GUI in which a fictive railway section is pictured. The two figures also show a traffic pattern with just
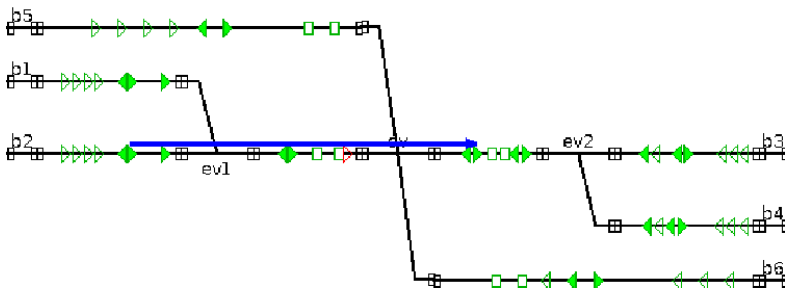

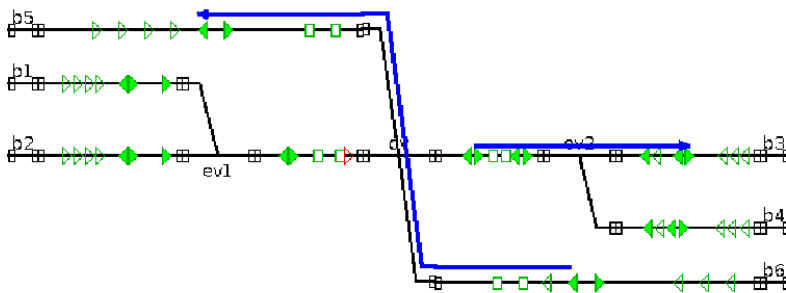Figure 1: Movement *R1* entering the railway section.

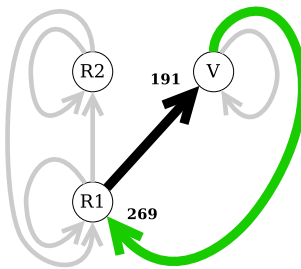Figure 2: Movement *R2* leaving and movement *V* passing the railway section.

Figure 3: The conflict graph for the traffic pattern *R1-V-R2*.

two trains. One of the trains, say train *TR*, is divided in two movements, *R1* and *R2*, by a waiting point. The other train, say train *T*V, has no waiting point and consists of just one movement *V*. Figure 1 shows us the movement *R1* and figure 2 shows us the two movements *V* and *R2*. Assume that, for this example all data needed for the analysis capacity has been given to the system. Among the given data is the traffic pattern given by ordering the movements, here given as *R1-V-R2*.

The first step in the capacity analysis is to make clear which the relevant conflicts between movements are. In Kaban, the conflicts of a traffic pattern is represented by a *conflict graph*. Figure 3 shows the conflict graph for the traffic pattern *R1-V-R2*. The conflict graph has events, i.e. movement starts, as nodes and conflicts between movements as *arcs*. Subsequently we will refer to conflicts between movements and minimum durations between two events as *arcs* and *arc weights*, respectively. Since the analysis is concerned with a repeated traffic pattern there are two kinds of arcs, those of conflicts of movements in the same period in the traffic pattern and those of movements in successive periods. The first kind is depicted and referred to as *straight arcs* and the second kind is depicted and referred to as *bowed arcs*.

Straight arcs:

R1→R2	262.2
R2→R1	191.1

Bowed arcs:

R1→R1	270.0
R2→R1	82.8
R2→R2	240.0
V→R1	269.4
V→V	270.0

Critical Graph:

V→R1	269
R1→V	191

Cycle Time: 460

Figure 4: Some Kaban results from the cycle time computation.

Let us now study the traffic pattern *R1-V-R2* as given by the conflict graph. *R2* starts from a waiting point and there it waits for other trains until *R2* can go on unhindered. But the graph tells us that movement *R2* and *V* are not in conflict, so what does *R2* wait for? The arcs ending in *R2* tell us that. There are two such arcs: *R1→R2* and *R2→R2*. The arc *R1→R2* concerns one and the same vehicle, the vehicle of train *TR* and therefore the time of the start of *R2* depends on the time of the start of *R1*. The other arc, *R2→R2*, says that the time of the start of *R2* also depends on the movement *R2* of the previous period. Hence, train *TR* in one period waits at the waiting point for the train *TR* of the previous period to move ahead far enough not to interfere with movement *R2* of the waiting train. Therefore, in this example, the waiting point makes a difference.

The *critical graph* is the subgraph of the conflict graph that determines the cycle time. In figure 3 the critical graph for the traffic pattern *R1-V-R2* is given by the bold arcs. We have here chosen to show the arc weights only in the critical graph. The weight of the arc *R1→V* means that *V* can start earliest 191 seconds after the start of *R1* and the weight of the arc *V→R1* means that *R1* can start earliest 269 seconds after the start of *V*. Hence, the minimum time for one period, i.e. the cycle time, is $191 + 269 = 460$ seconds. Figure 4 show us how Kaban presents the results concerning conflict graph, critical graph and cycle time for the studied example.

2.5 Analysis results for separate conflicts

As is shown in the previous section Kaban estimates capacity for repeated traffic patterns on a railway section and gives as main results, cycle time, critical sequences of movements and conflict graph. The cycle time estimates the feasibility of timetable options. For instance by answering the questions if it possible to

run a given timetable on a railway section and how much buffer time there is. The conflict graph tells us which the relevant conflicts are and can be used to answer questions about slack time for a chosen sequence of movements and with respect to the cycle time or a given timetable. The critical sequences of movements are of interest for robustness sensitivity analysis and for finding out how to reduce the cycle time further.

Since the cycle time can be reduced only by reducing the arc weight of arcs in the critical graph, we need to study the computation of those arc weights for understanding what the options are for reducing the cycle time. For this reason Kaban describes, for each arc $A{\rightarrow}B$ of the analysed traffic

- how is the arc weight computed
- the critical train route *RB* of the succeeding movement *B*
- the position that vehicle *A* must pass before *RB* is locked and the position that *B* must not pass before *RB* is locked.

Concerning the first point Kaban considers capacity limits for the surroundings of the analysed area and hence the weight of an arc is either a consequence of that capacity limit or an internal limitation. Moreover vehicles has a brake reaction time and the signalling system also has some reaction time which makes the final arc weight become a sum of some terms. That is what the first point refers to.

Regarding the second and third points let us once again consider the traffic pattern $R1{-}V{-}R2$ described by figures 1 and 2. The arc weight of $V{\rightarrow}R1$ is the minimum duration from the start of *V* till the start of *R1*. *R1* consists of more than one train routes, each one locked in one piece. It is not only the locking of the first train route of *R1* that is critical for the arc weight of $V{\rightarrow}R1$, it may be any of the train routes of *R1*. The reason for this is the assumption that *R1* is an unhindered movement all the way to the waiting point. Hence, we need to know which is the critical train route of *R1*, w.r.t. $V{\rightarrow}R1$ to reduce the arc weight of $V{\rightarrow}R1$. Given that the capacity limitation is internal and given the critical route, then the arc weight depends on requirements of the vehicle positions at the moment in time the critical train route is locked. The safety regulation says which position *V* must pass before locking the critical train of *R1*. The assumption that *R1* is unhindered does not allow the vehicle of *R1* to adjust its speed to the fact that some train route ahead is not yet locked. Hence, this assumption will determine the position the vehicle of *R1* must not pass before locking the critical train route of *R1*.

2.6 Support for entering data

An extensive amount of data is required for careful analysis of capacity and a tool with a supportive user interface will save a lot of time for the users. Capacity analysis in Kaban is based on data on

- infrastructure
- vehicles
- train routes and protection of train routes
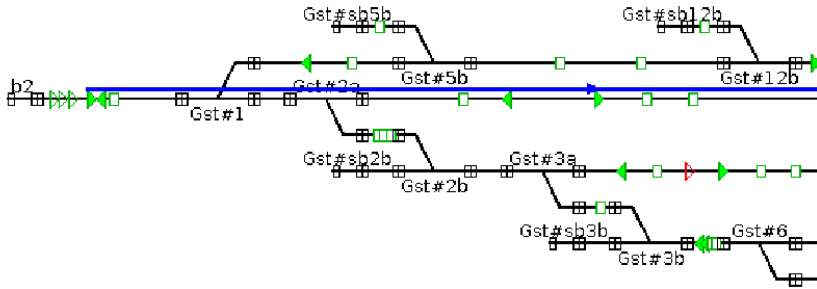- trains and traffic.

Figure 5: A station and a train. The station, Gamlestad, is based on BIS-data.

All needed data can be entered via the Kaban GUI. Some of the features of Kaban are

- visualisation of infrastructure, train routes and movements
- import of infrastructure data
- construction and modification of infrastructure data
- automatic generation of train routes including protection.

Infrastructure data includes, points, track circuits, speed limits, various signals and buffer stops. It is possible to import infrastructure data on a special XML-format generated from BIS, the infrastructure database of Banverket (the Swedish rail administration). Kaban does not automatically draw a visually nice picture of the track layout, but the Kaban GUI has got drawing modification support for concisely picturing the real track layout. Figure 5 shows an example on a Kaban drawing obtained by modifying BIS-imported data.

The vehicle data includes length, brake reaction time, acceleration and deceleration. The computations in Kaban is based on constant acceleration and deceleration. Train routes are automatically generated for all pairs of successive main signals. Protection for train routes are generated according to the Swedish safety regulation, where the first valid object is chosen in all directions needing protection. For any train using any train route the user may adjust the chosen protection. Those adjustments appear as allowance of non-valid objects to protect the train route and by disallowance of otherwise valid protection. There is several reasons for enabling such adjustments:

- to modify the train route protection is to adopt to older safety regulation, which may still be in use at some sites
- to give the user the possibility to override the safety regulation for some *almost* acceptable protective objects
- to increase capacity by cancelling conflicts which in turn is done by forbidding some protective objects for certain train routes used by certain trains.

2.7 Comparison with other tools for capacity analysis

The research in the area of capacity analysis is extensive and so are the number of tools. For surveys on methods and tools see for instance [6–8] and [8]. In contrast to Kaban, many of the tools simulate railway traffic and many of the methods aim at line capacity. Another tool that is based on the same kind of cycle time computations as Kaban, is presented in [9]. This tool used for evaluating timetable performance. The user-friendliness of capacity analysis tools ought to be measured by the time needed to obtain results of sufficient quality to support decisions on how to build railway or how to utilise existing railway.

3 Conclusions

This paper views *capacity analysis* as a means to support efficient construction and utilisation of railway infrastructure. Different methods and tools may contribute in different ways to help getting a solid understanding of which economic investments will make the most for satisfying the growing railway transportation needs. In this context time efficiency is crucial. Saving time and work is a decisive factor for increasing the quality of analysis results, balancing operational aspects and further developing capacity analysis, by equipping it with new powerful and relevant results. In this context analytic, static methods of analysing capacity has the potential to contribute with valuable results such as

- fast computations that enable a lot of cases for infrastructure and traffic to be part of the analysis
- analysis of trains behaving like we would like them to behave
- providing measures of capacity for comparing different solutions and enabling automatic search for the best solution
- offering results with a precise and clear meaning.

The paper presents Kaban aiming at being a user-friendly capacity analysis tool. Kaban is based on analytic, non-probabilistic methods and the results of Kaban aim at helping users to get a good understanding of the capacity advantages and drawbacks of proposals infrastructure and traffic. A main result is the cycle time of a repeated traffic pattern, useful for estimating feasibility of time table options. Other results aim at explaining how to adjust the infrastructure to increase capacity. The Swedish safety regulation is the basis for Kaban's careful model of train conflicts. Hence, Kaban suits capacity analysis in the signalling design phase of railway construction.

References

[1] Forsgren, M., Computation of capacity on railway networks. SICS Technical Report T2003:12, Swedish Institute of Computer Science, 2003. FoU-rapport, Banverket.

[2] Ekman, J., Capacity estimation of new infrastructure based on a discrete event model of train paths. *Computers in Railways*, WIT Press, www.witpress.com, volume IX, pp. 539–548, 2004.

[3] Baccelli, F., Cohen, G., Olsder, G. & Quadrat, J.P., *Synchronization and Linearity: An Algebra for Discrete Event Systems*. Wiley: Chicester, 1992.

[4] Cochet-Terrasson, J., Cohen, G., Gaubert, S., Mc Gettrick, M. & Quadrat, J.P., Numerical computation of spectral elements in max-plus algebra. *IFAC Conference on System Structure and Control*, Nantes, France, 1998.

[5] Goverde, R. & Soto y Koelemeijer, G., *Performance Evaluation of Periodic Timetables: Theory and Algorithms*. Number S2000/2 in TRAIL Studies in Transportation Science, Delft University Press: Delft, The Netherlands, 2000.

[6] Barber, F., Abril, M., Salido, M., Ingolotti, L., Tormos, P. & Lova, A., Survey of automated systems for railway management. Technical Report DSIC-II/01/07, Department of Computer Systems and Computation, Technical University of Valencia, 2007.

[7] Abril, M., Barber, F., Ingolotti, L., Salido, M.A., Tormos, P. & Lova, A., An assessment of railway capacity. *Transportation Research*, **Part E(44)**, 2008.

[8] Kontaxi, E. & Ricci, S., Techniques and methodologies for railway capacity analysis: comparative studies and integration perspectives. *Rail Zurich 2009 3rd International Seminar on Railway Operations Modelling and Analysis*, 2009.

[9] Goverde, R. & Odijik, M., Performance evaluation of network timetables using peter. *Computers in Railways*, WIT Press: Southampton, volume VIII, 2002.