ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

Dynamically Reschedule the Trains in a Railway Traffic Network

[1]Bala Ramesh Vanukuru, [2]G. B. V. Padmanadh
[1,2]Dept. of CS, Kakinada Institute of Engg. & Tech (KIET), Korangi, Kakinada, AP, India

Abstract
Railway is a very important mode of transportation for passengers and freight, due to its peculiar characteristics. Few other transportation modes combine dedicated infrastructure connecting point to point cities and places of interests, high operational speed, high reliability, cost effective operations, high energy efficiency and very high safety rate. The aim of railway traffic control is to ensure safety, regularity, reliability of service and punctuality of train operations. Railway business strongly needs to improve the quality of service and to accommodate growth while reducing the costs. The punctuality analysis represents an important measure of rail operation performance and is often used as standard performance indicator. The development of new strategies for railway traffic control experienced an increasing interest due to the expected growth of traffic demand and to the limited possibilities of enhancing the infrastructure, which increase the needs for efficient use of resources and the pressure on traffic controllers. Improving the efficiency requires advanced decision support tools that accurately monitor the current train positions and dynamics, and other operating conditions, predict the potential conflicts and reschedule trains in real-time such that consecutive delays are minimized. We design a model predictive controller based on measurements of the actual train positions. The core of the model predictive control approach is the railway traffic model, for which a switching max-plus linear system is proposed. If the model is affine in the controls, the optimization problem can be recast as a mixed-integer linear programming problem. To this end we present a permutation-based algorithm to model the rescheduling of trains running on the same track

Keywords
Switching Max-Plus Linear Models, Model Predictive Control, Mixed-Integer Linear Programming, Railway Traffic Management

I. Introduction
Railway is a very important mode of transportation for passengers and freight, due to its peculiar characteristics. Few other transportation modes combine dedicated infrastructure connecting point to point cities and places of interests, high operational speed, high reliability, cost effective operations, high energy efficiency and very high safety rate.

Key areas of improvement include providing a better service, decreasing or keeping low fares, increasing reliability and connectivity for both railway traffic and inter-modal connections [1]. In the Netherlands, the Government and the Ministry of Transport have the long-term goal of making train services quicker and more frequent, more comfortable for passengers and more reliable [5]. In fact, the quality of service of the Dutch railway system was relatively low around the year 2000, with less than 80% of trains arriving at major stations within 3 minutes of delay.

Railway traffic management is mainly directed towards the implementation of an existing plan of operations (off-line timetabling) and its adjustment due to disruptive events as quickly as possible (real-time dispatching). The timetable is characterized by departure and arrival times of each train at station platforms and/or at relevant merging and crossing points. The assignment of routes, platforms and passing times may require months, during which several variants are analyzed in depth under economical and operational constraints. In real-time, unforeseen events may disrupt the timetable and thus the resolution of route conflicts and other infeasibilities is required.

The real-time dispatching process is to determine feasible (conflict-free and deadlock free) train movements minimizing timetable deviations. An accurate prediction of the effects of delays and other disturbances requires modeling the evolution of train traffic in sufficient detail and considering the actual state of the network, both the dynamic behavior of circulating trains and the dispatching measures used to control traffic. Hence, the precise delay propagation cannot be predicted by dispatchers, especially in case of complicated station areas, high density traffic and severe disturbances. Furthermore, railway managers are looking for decision support systems that enable their operators to determine implementable control actions as quick as possible. For these reasons, there is a need for developing more sophisticated and efficient decision support tools to forecast the network delay propagation for individual dispatching measures.

A railway network with rigid connection constraints and a fixed routing schedule can be modeled using max plus linear (MPL) models (Heidergott and de Vries, 2001). An MPL model is linear in the max-plus algebra (Baccelli et al., 1992), which has maximization and addition as its basic operations. Max-plus-linear systems can be characterized as discrete event systems in which only synchronization and no concurrency or choice occurs (Baccelli et al., 1992). In the railway context, synchronisation means that some trains should give predefined connections to other trains, and a fixed routing schedule means that the order of departure is fixed. In this paper we model a controlled railway system using the switching max-plus-linear system Research funded by the Dutch Technology Foundation STW project 'Model-Predictive Railway Traffic Management' (11025). description of van den Boom and De Schutter (2006). In this description we use a number of MPL models, each model corresponding to a specific mode, describing the network by a different set of connection and order constraints. We control the system by switching between different modes, allowing us to change the order of trains to minimize the delays of all trains in the network while considering the cost of the control actions.

II. Related Works
Existing models for solving routing and scheduling problems can be classified according to two levels of approximation: macroscopic models and microscopic models. Offline timetabling usually relies on macroscopic models, while microscopic approaches are mandatory when dispatching train traffic in real-time.

To keep complexity low at the planning stage, macroscopic approaches model a railway network as a simplified series of links connecting stations. A fixed running time is required to travel between two stations, and a fixed headway time is imposed between consecutive trains on the same link or platform at stations.

The time variables are normally bounded to full minutes. Several works on timetabling use a formulation based on the periodic event scheduling problem by Serafini and Ukovich [4], which assumes infinite capacity at stations and a rough model of headway times and safety system.

For the Dutch railways, DONS is the macroscopic tool adopted to design timetables. A network scheduler module, called CADANS (see e.g. [3]), determines a feasible cyclic network timetable, while having fixed running and minimum headway times, and m neglecting capacity at stations. A second level module, called STATIONS (see e.g. [2]), manages the routing of trains in complex station areas. The model takes into account incompatibility between routes according to predefined safety constraints and builds a graph of incompatibilities in which a maximum weight node packing corresponds to a feasible routing solution, while neglecting the impact of signalling and train length on blocking times.

Carey and Crawford [5] present a sophisticated model and a novel heuristic procedure to assess the benefits of an existing draft timetable for a network of busy stations. The precise track layout of stations and incompatibility of conflicting routes and platform tracks occupations are taken into account in the model, while train separation is formulated as a fixed minimum headway distance. The heuristic solves the route conflicts along the tracks by selecting the least immediate delay cost.

Caimi et al. [4] solve ordering and routing problems in station areas simultaneously, by building a large conflict graph that takes into account multiple scheduling possibilities for each train. A fixed point iteration algorithm has been implemented to compute a feasible solution in a reasonable computation time. The procedure assumes that trains have fixed running and headway times in interlocking areas, while acceleration, braking, dwell time extensions, as well as variations of train length, are not discussed.

A greater level of detail is needed to properly control railway traffic during operations. Accurate train positions, speeds, and acceleration and braking time losses have to be computed for a reliable prediction of the trains trajectories including blocking times. The speed profile of trains has to be computed according to the actual speed limits and the corresponding acceleration and deceleration rates (see e.g. [7]). The signaling system with actual signal aspects needs to be modeled along corridors and in station areas, while a precise layout of interlocking areas is required to take into account incompatibilities between routes.

Rodriguez [2] studies rerouting and reordering possibilities for a small network with up to 12 trains. A job shop scheduling model with additional state resources constraints is proposed to detect and solve route conflicts. Synchronization constraints are used to keep train running with sufficient headway distances, even in case of yellow or red signal aspects.

However, variability of train speed profiles is not considered. Dispatching solutions are computed by constraint-based programming in a short computation time.

III. Proposed System

1. Consider a periodic railway operations system that follows a schedule with period T.
2. In nominal operation mode, we assume that all the trains follow a pre-scheduled route, with a fixed train order and predefined connections.
3. If for any of the reasons mentioned before delays are introduced in the network, it might be advantageous to change the train order so as to minimize delays.
4. In this case we will operate in a perturbed mode with an associated new schedule.
5. First we discuss about normal operation mode. Consider an unperturbed railway operations system where each train running on each track of the railway network has a number assigned to it.
6. We will say 'train i' to denote the (physical) train on (virtual) track i, and 'station i to denote the (virtual) station at the beginning of track i (cf. fig. 1). Let n be the number of 'virtual' tracks in the network. We say virtual to denote that some of the virtual tracks or stations may actually be the same physical track or station (corresponding to different trains using the same track or station). This means that the actual number of tracks is usually smaller than n.

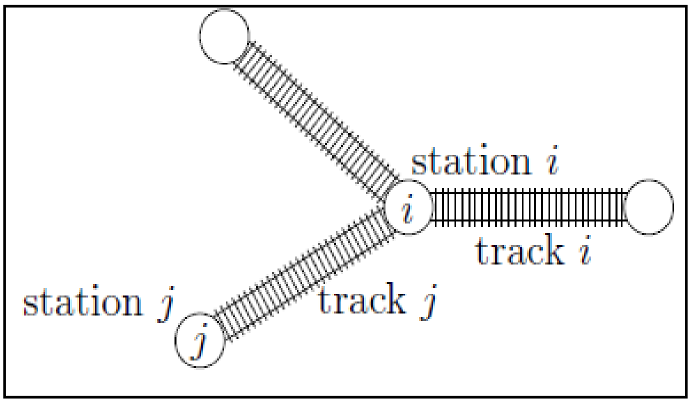

Fig. 1 : A part of a Railway Network.

- Let Pi be the predecessor track of train i, i.e., the track that ended at station i, and let C_i (k) be the set of trains that give a connection to train i in the kth cycle.
- Let F_i(k) be the set of trains that move over the same track as train i, in the same direction as train i, and that are scheduled before train i in the kth cycle.
- Let W_i(k) be the set of trains that move over the same track as train i, in the opposite direction of train i, and are scheduled before train i in the kth cycle.
- Let ti(k) be the running time of train i in the kth cycle.
- Define a minimum connection time c_{ij}(k) for passengers to get from train j to train I for each train j C_i(k) in the kth cycle, and define a minimum dwell time s_j(k) of train j at station j in the kth cycle to allow passengers to board or alight the train.
- Finally, define a minimum headway time h_{ij}(k) between two different trains i and j moving over the same track and in the same direction in the kth cycle, and a minimum headway time w_{ij}(k) between two different trains moving over the same track and in the opposite direction in the kth cycle.
- The departures and arrivals of train i are subject to the following constraints:

Time schedule constraint:

$$d_i(k) \geqslant r_i(k),$$
$$a_i(k) \geqslant r_{i+n}(k),$$

where ri(k) = r_i(0)+kT with r_i(0) the initial scheduled departure time.

- Running time constraint:

$$a_i(k) \geqslant d_i(k-\delta_{ii}) + t_i(k),$$

where δ_{ii} = 0 if train i is scheduled to arrive at its destination in the same cycle as its departure, and δ_{ii} = m if it arrives m cycles after its departure.

- Continuity constraints

$$d_i(k) \geqslant a_{p_i}(k-\delta_{ip_i}) + s_{p_i}(k),$$

where δ_{ij} = m if the (k-m)th train j gives a connection to the kth train i.

- Headway constraints

$$d_i(k) \geqslant d_j(k-\delta_{ij}) + h_{ij}(k), \ \forall j \in \mathcal{F}_i(k),$$
$$a_i(k) \geqslant a_j(k-\delta_{ij}) + h_{ij}(k), \ \forall j \in \mathcal{F}_i(k),$$

- Meeting constraints:

$$d_i(k) \geqslant a_j(k-\delta_{ij}) + w_{ij}(k), \ \forall j \in \mathcal{W}_i(k),$$

Since a train is allowed to depart as soon as all constraints are satisfied, we have

$$\begin{aligned} d_i(k) = \ &\max \Big\{ r_i(k), a_{p_i}(k-\delta_{ip_i}) + s_{p_i}(k), \\ &\max_{j \in \mathcal{C}_i(k)} (a_j(k-\delta_{ij}) + c_{ij}(k)), \\ &\max_{j \in \mathcal{F}_i(k)} (d_j(k-\delta_{ij}) + h_{ij}(k)), \\ &\max_{j \in \mathcal{W}_i(k)} (a_j(k-\delta_{ij}) + w_{ij}(k)) \Big\}, \\ a_i(k) = \ &\max \Big\{ r_{i+n}(k), d_i(k-\delta_{ii}) + t_i(k), \\ &\max_{j \in \mathcal{F}_i(k)} (a_j(k-\delta_{ij}) + h_{ij}(k)) \Big\}. \end{aligned}$$

A. Perturbed Operation

- Some trains should give predefined connections to other trains, and that the order of trains on the same track is fixed.
- if one of the preceding trains has a too large delay then it is sometimes better — from a global performance viewpoint — to let a connecting train depart anyway or to change the departure order on a specific track.
- This is done in order to prevent an accumulation of delays in the network. In this paper we consider the switching between different operation modes, where each mode corresponds to a specific order of train departures and arrivals.

$$x(k) = \bigoplus_{m=m_1}^{m_2} A_m(\ell(k)) \otimes x(k-m) \oplus r(k),$$

B. Permutation Method

- Let us assume that T = {1, 2, ..., nt} is the set containing all physical tracks in the network with nt the total number of physical tracks.
- The method computes for each physical track all the allowed permutations of the currently scheduled train sequence.
- In the following a superscript τ denotes the association to the τ th physical track with τ ∈ T . Each permutation of trains on each physical track τ ∈ T is described with a binary control vector

$$u^{(\tau)}(k) = [u_1^{(\tau)}(k), \ldots, u_{\sigma_\tau(k)}^{(\tau)}(k)]^{\mathrm{T}},$$

with $u_v^{(\tau)}(k) \in \{0,1\}$, $v = 1, 2, \ldots, \sigma_\tau(k)$ and

$$\sigma_\tau(k) = \binom{n_\tau(k)}{2} = \frac{n_\tau(k)(n_\tau(k)-1)}{2}$$

- The number of control entries for the τ th physical track and nr (k) the number of trains scheduled to run on the τ th physical track in cycle k.
- The permutation of the order of two specific trains scheduled to run on the same physical track τ with respect to the currently scheduled sequence of trains. This means that if in a given permutation the order of two trains is swapped, then the associated control will be equal to one. On the other hand, if the order of the two trains does not change with respect to the currently scheduled sequence, then the associated control will be equal to zero.
- To define the mode matrices Am,v(k) two cases have to be accounted for: the case that a control is associated to changing the order of two trains that are scheduled in the same direction, and the case that a control is associated to changing the order of two trains that are scheduled in opposite directions.
- In either case the mode matrices will implement the removal of an existing precedence constraint between two trains and the addition of a new one.
- Let us assume that a control u(T) v (k) is associated to the allowed permutation of the pth and qth train on track τ in cycle k, with q < p, i.e., the qth train is originally scheduled before the pth train in a cycle. Then the corresponding mode matrix A(T)m,v(k) can be

Written as

$$[\hat{A}_{m,v}^{(\tau)}(k)]_{ij} = \begin{cases} \beta - h_{pq}(k) & \text{if } (i,j) = (p,q), \\ h_{qp}(k) - \beta & \text{if } (i,j) = (q,p), \\ \beta - h_{pq}(k) & \text{if } (i,j) = (n+p, n+q), \\ h_{qp}(k) - \beta & \text{if } (i,j) = (n+q, n+p), \\ 0 & \text{otherwise}. \end{cases}$$

- In the case that train q is scheduled before and in opposite direction as train p, then matrix A(T)m,v(k) can be Written as

$$[\hat{A}_{m,v}^{(\tau)}(k)]_{ij} = \begin{cases} \beta - w_{pq}(k) & \text{if } (i,j) = (p, n+q), \\ w_{qp}(k) - \beta & \text{if } (i,j) = (q, n+p), \\ 0 & \text{otherwise}. \end{cases}$$

- Where affinity with respect of the controls is still preserved. The total number of controls nu is then calculated as

$$n_u = \sum_{\tau=1}^{n_t} \sigma_\tau(k) = \sum_{\tau=1}^{n_t} \frac{n_\tau(k)(n_\tau(k)-1)}{2},$$

Reformulation as a Mixed-Integer Linear Programming Problem

- The model predictive control problem with Am (ℓ(k)) can be recast into an MILP problem.
- Assuming that in general m1 = 0 and m2 ≥ m1, we outline now the main ideas behind this transformation. For the sake of simplicity of notation we drop the notation ˆ· and |t for a prediction from now on. Define the vectors

$$\begin{aligned} \tilde{x}(k) &= [x^{\mathrm{T}}(k), \ldots, x^{\mathrm{T}}(k+N_{\mathrm{p}})]^{\mathrm{T}}, \\ \tilde{u}(k) &= [u^{\mathrm{T}}(k), \ldots, u^{\mathrm{T}}(k+N_{\mathrm{p}})]^{\mathrm{T}}, \\ \tilde{z}(k) &= [x^{\mathrm{T}}(k-1), \ldots, x^{\mathrm{T}}(k-m_2)]^{\mathrm{T}}, \\ \tilde{\ell}(k) &= [\ell(k), \ldots, \ell(k+N_{\mathrm{p}})]^{\mathrm{T}}, \\ \tilde{r}(k) &= [r^{\mathrm{T}}(k), \ldots, r^{\mathrm{T}}(k+N_{\mathrm{p}})]^{\mathrm{T}}, \end{aligned}$$

where ˜x(k) represents the partially known or completely unknown states and ˜z(k) represents the completely known states at cycle k .

$\tilde{A}(\tilde{\ell}(k)) =$

$$\begin{bmatrix} A_0(\ell(k)) & A_{-1}(\ell(k)) & \cdots & A_{-N_p}(\ell(k)) \\ A_1(\ell(k+1)) & A_0(\ell(k+1)) & \cdots & A_{1-N_p}(\ell(k+1)) \\ \vdots & \vdots & \ddots & \vdots \\ A_{N_p}(\ell(k+N_p)) & A_{N_p-1}(\ell(k+N_p)) & \cdots & A_0(\ell(k+N_p)) \end{bmatrix}$$

where $A_m(k+j) = \mathcal{E}$ for $m < m_1$ and for $m > m_2$, and
$\tilde{B}(\tilde{\ell}(k)) =$

$$\begin{bmatrix} A_1(\ell(k)) & \cdots & A_{m_2-1}(\ell(k)) & A_{m_2}(\ell(k)) \\ A_2(\ell(k+1)) & \cdots & A_{m_2}(\ell(k+1)) & \mathcal{E} \\ \iddots & & \iddots & \vdots \\ A_{m_2}(\ell(k+m_2-1)) & \cdots & \mathcal{E} & \mathcal{E} \\ \mathcal{E} & \cdots & \mathcal{E} & \mathcal{E} \\ \vdots & \ddots & \vdots & \vdots \\ \mathcal{E} & \cdots & \mathcal{E} & \mathcal{E} \end{bmatrix},$$

- we can write

$$\tilde{x}(k) = \tilde{A}(\tilde{\ell}(k)) \otimes \tilde{x}(k) \oplus \tilde{B}(\tilde{\ell}(k)) \otimes \tilde{z}(k) \oplus \tilde{r}(k).$$

- This is not a major problem if the matrix A˜(ℓ˜(k)) has a strictly lower triangular structure, which can always be achieved by a renumbering of the departures and arrivals. the matrices A˜(ℓ˜(k)) and ˜B (ℓ˜(k)) will be affine in u˜(k), and there exist matrices A˜v and ˜Bv such that:bbb

$$\tilde{A}(\tilde{\ell}(k)) = \tilde{A}_0 + \sum_{v=1}^{n_u} \tilde{A}_v\, \tilde{u}_v(k),$$

$$\tilde{B}(\tilde{\ell}(k)) = \tilde{B}_0 + \sum_{v=1}^{n_u} \tilde{B}_v\, \tilde{u}_v(k).$$

- Note that ˜ℓ(k) is a function of ˜u(k), which can be expressed as ˜ℓ(k) = ˜L(˜u(k)). The objective function J(k) is linear in ˜u(k) and ˜x(k), and can be written as:

$$J(k) = c_e^{\mathrm{T}}\, \tilde{x}(k) + c_u^{\mathrm{T}}\, \tilde{u}(k),$$

- Where the constant term −cTe˜r(k) has been omitted since it does not affect the optimization. It can be written as

$$\tilde{x}_i(k) = \max(\tilde{r}_i(k), \max_j(\tilde{x}_j(k) + [\tilde{A}(\tilde{\ell}(k))]_{ij}), \\ \max_j(\tilde{z}_j(k) + [\tilde{B}(\tilde{\ell}(k))]_{ij})),$$

- which can be transformed into

$$\begin{cases} \tilde{x}_i(k) & \geq \tilde{r}_i(k), \\ \tilde{x}_i(k) & \geq \tilde{x}_j(k) + [\tilde{A}_0]_{ij} + \sum_{v=1}^{n_u} [\tilde{A}_v]_{ij}\, \tilde{u}_v(k) \quad \forall j, \\ \tilde{x}_i(k) & \geq \tilde{z}_l(k) + [\tilde{B}_0]_{il} + \sum_{v=1}^{n_u} [\tilde{B}_v]_{il}\, \tilde{u}_v(k) \quad \forall l. \end{cases}$$

- It is clear that all these constraints are linear in ˜x(k) and ˜u(k), and we end up with the linear inequality constraint:

$$A_c \begin{bmatrix} \tilde{x}(k) \\ \tilde{u}(k) \end{bmatrix} \leq b_c(k),$$

Where bc(k) contains all known elements of (20) at cycle k

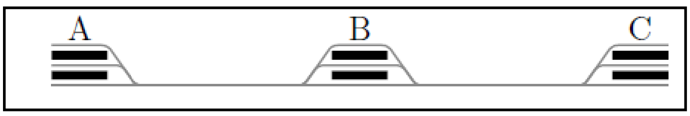

Fig. 2: A Simple Railway Network

The network has three stations denoted as A, B, and C.
In this example we are only interested in trains in the direction A-C and so we consider only a single track. At all three stations there are multiple platforms which means there is sufficient capacity for take over operations. The period of the timetable is T = 60 [min]. During every period there are four local trains and four intercity trains running from station A to C. Both types of train make a stop at station B, the local train also stops at some intermediate stations. We assume that these intermediate stations have no overtaking possibility, and are therefore omitted in the analysis. The corresponding timetable is given in Table 1.

Table 1: Timetable (d=Departure, a=arrival)

		Line 1: Local train				Line 2: Intercity			
Train number		101	102	103	104	201	202	203	204
Station A	d	00	15	30	45	09	24	39	54
Station B	a	15	30	45	00	18	33	48	03
	d	23	38	53	08	20	35	50	05
Station C	a	35	50	05	20	27	42	57	12

Two delay scenarios are simulated

- In the first scenario intercity train 201 with scheduled departure time 8:09 at station A has a delay less than 6 minutes. In this scenario changing the departure order with the proposed approach results in no delay reduction.
- In the second scenario intercity train 201 at station A has a delay of 12 minutes. By not changing the order of the trains, the sum of delays becomes 51 [min]. Using the proposed approach intercity train 201 is rescheduled behind local train 102 on track A-B and behind local train 101' on track B-C. The total delay now reduces to30 [min] resulting in a delay reduction of 21 [min]. The corresponding time-distance diagram is given in fig. 3.

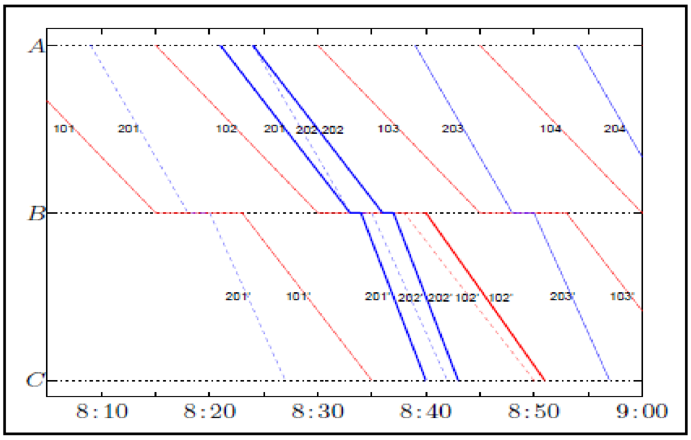

Fig. 3: Time-distance Diagram for Scenario 2. The Bold Line Represent Rescheduled Trains, Where as the Dashed Line Represent the Original Train Paths

IV. Conclusion

Current practice in the operational-level management of railway traffic networks is mostly based on predefined rules and on the ability of traffic controllers and train dispatchers to detect and avoid conflicting situations. Delays caused by technical failures, fluctuation of passenger volumes, and/or weather conditions can be partly absorbed by a stable and robust timetable (Goverde, 2007 [2]). In the case of large delays, network managers might be forced to re-route or to change the order of trains, break connections,

ISSN : 0976-8491 (Online) | ISSN : 2229-4333 (Print)

or even cancel a scheduled service to prevent the accumulation of delays in the network. In this paper we design a predictive feedback controller that computes the most effective actions, based on measurements of the actual train positions. The control measures are restricted to changing the order of trains running on the same track. We apply the algorithm to a simple railway traffic network simulation model and show a significant reduction of delays compared to the uncontrolled case.

References

[1] Baccelli, F., Cohen, G., Olsder, G., Quadrat, J. "Synchronization and Linearity: An Algebra for DiscreteEvent Systems", Wiley, 1992.
[2] Goverde, R.M.P.,"Railway timetable stability analysis using max-plus system theory", Transportation Research Part B, 41(2), pp. 179–201, 2007.
[3] Goverde, R.,"A delay propagation algorithm for large-scale railway traffic networks", Transportation Research Part C: Emerging Technologies, 18(3), pp. 269–287, 2010.
[4] Heidergott, B., de Vries, R.,"Towards a (max,+) control theory for public transportation networks", Discrete Event Systems: Theory and Applications, 11(4), pp. 371–398, 2001.
[5] van den Boom, T.,"De Schutter, B.,"Modelling and control of discrete event systems using switching max-plus-linear systems", Control Engineering Practice, 14(10), pp. 1199–1211, 2006.
[6] Van Den Boom, T., De Schutter, B.,"On a model predictive control algorithm for dynamic railwaynetwork management", In I. Hansen, A. Radtke, J. Pachl, E. Wendler (eds.), Proceedings of the 2nd International Seminar on Railway Operations Modelling andAnalysis (Rail Hannover 2007). Hannover, Germany, 2007.

Mr. Bala Ramesh Vanukuru is a final year student of KIET-Kakinada Institute of Engg. and Tech., Korangi. Presently He is pursuing his M.Tech (C.S.E) from this college and he received his Bachelor of Technology (B.Tech) from JNTUK University, His area of interest includes Computer Networks and Object oriented Programming languages in Computer Applications.

Mr. G.B.V.Padmanadh, an Efficient teacher, received M.Tech in the stream of ComputerScience&Technology from Githam university, Visakhapatnam, and is been persuing Ph.D from JNTU Kakinada and He is working as an Associate Professor in Department of C.S,Kakinada Institute of Engg. and Tech, Korangi. He has 8 years of teaching experience. He has published many papers in both National & International Journals. His area of Interest includes Data Communications & Networks, Database Management Systems, C Programming and other advances in Computer Applications.