EXDAFS - An Expert System For Dynamic Allocation Of Facilities At Stations


Centre for Railway Information Systems, Chanakyapuri, New Delhi-100 021.

Abstract: EXDAFS is a rule-based system designed to facilitate allocation of facilities such as platforms, washing lines, stabling lines etc. to trains arriving and/or departing from a Railway Station. EXDAFS develops a static allocation for all trains, referred to as the Master Berthing Chart (MBC), based on the train schedules and their berthing requirements, station characteristics, various constraints and their interplay.

However, the trains get delayed due to unavoidable circumstances causing deviations from the normal schedule, thereby necessitating a dynamic re-assignment of the facilities. A technique based on transformational synthesis [1] to control the computational complexity of the re-assignment process has been devised and implemented. This requires the problem to be decomposed into sub-tasks which can be solved independently and then synthesized into a final solution.

The system takes decisions based on inherent knowledge which concern diverse areas, such as, safety, efficient allocation of scarce resources, and transportation efficiency, globally in a real-time environment.

EXDAFS has an explanation capability which provides a better understanding of its reasoning directions. This along with a visual capability and interaction through a user friendly interface results in an increased user acceptance of the system.

AI Topic: Expert System, search, transformational synthesis, belief maintenance, planning system.

Domain Area: Dynamic scheduling, Transportation.

Languages Used: Combination of Prolog and 'C'.

Status: The system is under trial and would be deployed at New Delhi Railway Station.

Impact: EXDAFS obviates the possibility of mishaps due to human error by relieving the controller of stress, particularly, during peak hours by presenting alternative scenarios for station facility assignment to trains in complex situations.

1 Problem Description

The utilization of facilities by trains is derived from the railway time table. The allocation of platforms and facilities (P&F), such as washing lines and stabling lines, are made taking into account all constraints, such as length of platform, track layout etc. as well as movement needs.

This allocation of facilities is plotted against time on a chart indicating their occupancy by all trains arriving & departing from the station. This chart is referred to as the Master Berthing Chart (MBC) (Figure 1), and forms the working basis for the re-allocations.

Indian Railway revise their-time table twice a year. The MBC for a station is adjusted to meet the requirements of the new train schedules. The MBC also requires changes when new trains are introduced or existing schedules are altered by the Railway from time to time.

In case of a disruption, P&F assignment becomes a task of re-allocating trains to facilities while not violating resource constraints or creating new conflicts. Every effort is made to keep changes to preassigned slots on MBC to a bare minimum. A controller faces such facility allocation problems daily and decisions taken in a complex scenario may not always be optimal and may vary from controller to controller under similar circumstances.

It is seen that the number of parameters affecting the decision making process is large and often it is not possible to mathematically equate their behavior to the final solution. If the task is modelled as an integer linear programming model, the number of problem states will be too large. It is most likely that no polynomial algorithm exists for solving the general constraint satisfaction problem [2]. Also a controller's expertise cannot be captured by any single algorithm as it is intuitive and is coded as thumb rules. It is much better modelled in the form of a production system using If Then type of rules. This also helps understand the expert system's reasoning process and thus forms the basis for an effective explanation system.

2 The Solution Process

As stated earlier, the system performs two major tasks i.e. the generation of the Master Berthing Chart due to the periodic changes in the train schedules and secondly the on-line re-allocation of trains due to the disruption of the regular schedule by unforeseen circumstances.

Static allocation requires the train schedules and specifications, and the station characteristics as input. EXDAFS models the domain specific procedural knowledge [3] of an expert in the form of production rules, which are activated by an inferencing mechanism to arrive at practically acceptable solutions.

The system incorporates the concept of certainty factors (CF) for belief maintenance, as developed by Buchanan and Shortliffe in the MYCIN expert system [4], to arrive at an acceptable allocation considering the available evidence. Certainty factors provide a relatively informal mechanism as compared to probability theory etc. [5], for quantifying the degree to which, based on the
presence of a given set of evidence, we believe (or disbelieve) a given hypothesis. They are well suited to our domain because it uses incrementally acquired evidence to make decisions. The certainty factors associated with the rules provide great flexibility in tuning the system.

The second task of re-allocation of trains requires information about deviations from the regular schedule and the current state of the berthing chart. Dynamic scheduling using only a rule-based approach [6] is not viable because of complexity of the problem due to large number of trains and limited resources. Therefore a technique based on transformational synthesis is employed. The system tries to make re-assignments with minimal changes to the existing schedule so as to resolve all conflicts which may have been introduced.

2.1 Inputs to EXDAFS.

EXDAFS requires information on station and train characteristics for scheduling decisions, such as:-

* Railway station characteristics:
  - Layout, Class and Signalling at the station.
  - Number of platforms, washing lines, stabling lines, and their specifications.
  - Master berthing chart.
  - Special features and restrictions if any.

* Train characteristics:
  - Composition of the train.
  - Services required during the halt, i.e.
    i. primary or
    ii. secondary.
  - Engine characteristics and requirements.
  - Line & direction of arrival and departure.
  - Class/Priority of the train.

It also requires following information in a real time environment:-
  - Expected time of arrival and departure of trains.
  - Rake links.
  - Actual time of arrival and departure.
  - Movements for shunting trains.
  - Any temporary restrictions at the station.

3 Rules and Constraints

To perform allocations, EXDAFS has to satisfy a number of constraints [7]. To arrive at assignments without violating these constraints EXDAFS uses production rules which can broadly be divided into two categories i.e. conflict rules and preference rules, depending upon the type of constraint to which they pertain.
We shall first develop a notation for representing the facilities and their characteristics, to model various constraints and rules.

The facilities comprise of the following characteristics -

i. Platforms \( P_i \): length \( L \), direction \( D \) [up, down], raised \( R[\text{true}, \text{false}] \), ...
ii. Washing Lines \( WL_j \): length \( L \), type \( T[\text{pit, nonpit}] \), complex \( C \), ...
iii. Stabling Lines \( SL_k \): length \( L \), complex \( C \), Subscripts \( i, j \) and \( k \) denote instances of the facilities.

Trains are denoted by \( T_m \) with characteristics of -

name \( N \), coaches \( C \), power \( P[\text{electric, diesel}] \), direction \( D \), ...

The schedule of train \( T_m \) is denoted by \( SCH_{T_m} \), and comprises of - arrival time \( A \), departure time \( D \), scheduled platform \( PF \).

The characteristic of a facility is referred to as facility.characteristic.

Platforms with special characteristics are grouped as -

- set of platforms with overhead equipment (OHE) - E.
- set of dead end platforms - D.

3.1 Conflict Rules.

The constraints modelled by these rules cannot be violated under any circumstances otherwise improper functioning, and a lack of safety will result.

Physical Constraints: These occur due to certain physical specifications and requirements, i.e., to say some P&F options may have some physical constraints which prohibit them from servicing certain train types. See Figure 2.

Note: If the expressions shown in figure 2. 3 & 4 are satisfied, a conflict arises and such an allocation can not be made.

- If the train has an electric locomotive then it should be allocated to a platform with OHE. This can be expressed as follows:

  \[ \neg \text{element}(P_i, E) \land \text{equal}(T_m.P, E) \]

- The train cannot be assigned to the platform if it does not fit within the fouling marks in accordance with the line capacity restrictions, i.e.

  \[ \neg \text{lessthan}(P_i.L, T_m.C) \]

Figure 2. Examples of physical constraints.

Movement Constraints: Another set of constraints arise when the movements of the trains in the yard have to be considered. For this task information about the track layouts and the permissible direction of movements at the junction points is required. The feasible simultaneous movements can be computed based on the above data. In a real time situation the state of the signals also has to be considered to avoid improper routing which may result in train collisions. In the present implementation signalling is not taken into account, however, to avoid conflicts during train movements during the same time-frame, information is kept about the combinations of lines which are feasible for simultaneous operation as derived from the station working rules. See Figure 4.

- The direction of the line should conform to the train route, i.e., up or down, i.e.

  \[ \text{equal}(T_m.D, P_i.D) \]

- Don't allocate a dead end platform to a passing train, i.e.

  \[ \text{element}(P_i, D) \land \neg \text{equal}(SCH_{T_m}.A, \text{origin}) \land \neg \text{equal}(SCH_{T_m}.D, \text{terminate}) \]

Figure 3. A typical example of a dependency constraint.

3.2 Preference Rules

Broadly speaking preference rules provide guide-lines for arriving at viable assignments. The class of constraints represented by these rules can be violated if it is not possible to make an allocation otherwise. They take into consideration such aspects as passenger convenience, optimality of allocation as well as certain other factors like priority and delays. While conflict rules work to limit the number of practically possible alternatives, the preference rules help choose the best option within them. The constraints modelled by these rules are discussed below:

Convenience Constraints: These constraints help to minimize passenger inconveniences. All passenger trains should be halted only at raised platforms but in the case of non-availability of such a platform a train can also be routed to a level platform. Sometimes due to disruption of the normal schedule, passenger trains can not be brought into their usual platforms. If such changes are announced at the last moment they may result in chaos due to movement of the large number of passengers. Therefore, last moment changes should be avoided, although if necessary, a preference is given to the adjacent platform.
In the case of terminating trains which also require services by other facilities like washing lines and engine stabling lines, shunting has to be done between the various lines. In such situations, the number of movements should be kept to a minimum. For example, if a train requires a primary service at the washing line which is presently occupied it can be kept at the platform until the washing line is free.

**Optimality Constraints:** A main line or a sub-line may connect to a number of platforms. One should avoid arrival and departure from platforms served by the same sub-line within a short span of time to maintain a smooth flow.

### 4 Platform and Facilities Assignment

As discussed earlier, EXDAFS maintains two types of rules—rules with constraints and preference rules.

The static assignment strategy is to first use the conflict rules to eliminate the options which are not practically feasible thus pruning the search space, and then apply the set of preference rules to the (train, platform) combination for the feasible platforms to arrive at a good option.

We use the concept of certainty factors to choose between several competing hypotheses. A measure of belief is associated with each hypothesis, and the degree of belief attributed to each. After all the evidence has been analysed the system concludes from the hypothesis with a higher measure of belief. The use of CFs also provides a mechanism for tuning the system. Higher preference can be given to some rules by increasing the degree of belief associated with them while reducing those of others. By doing this we arrive at totally different solutions using the same set of rules and evidence.

EXDAFS prioritizes trains depending upon their category, i.e. express train, mail train etc. It first performs an allocation for the highest priority trains. Also platforms with greater constraints are filled up first.

In the case of terminating trains requiring other facilities such as washing lines and stabling lines, their availability and connection to the chosen platform has to be verified.

In case EXDAFS is unable to allocate all the trains as specified by the schedule, it tries to minimally change some previous assignments to accommodate the unallocated trains. This is done by invoking a re-assignment strategy based on transformational synthesis as discussed in the next section.

If certain assignments are unacceptable to the user, a manual system of assignment is also provided in the menu options with which he can assign trains to P&F, unassign trains from P&F, or specify that trains can be assigned only to a specific P&F. In this mode the system performs a consistency check at each step to obviate the chances of any invalid assignment by invoking the rules with constraints.

### 5 Dynamic Re-assignment

During the normal course of operation there are many deviations from the standard schedule due to the delays in train arrival and departures. Whenever such information is received by the controller in the power cabin, it is entered into the system. If the disruption is a minor one, and does not create conflicts with other train allocations, no action is required by the system. However, a suitable re-allocation strategy needs to be evolved to resolve a conflicting situation.

It is not possible to formulate the allocation plan for all the facilities, i.e. platforms, washing lines and engine stabling lines, as a single search problem. This necessitates the decomposition of the entire problem into a set of tractable sub-tasks. Therefore transformational synthesis paradigm, which allows integration of various planning techniques so that the most suitable one can be applied to solve each sub-task independently, is useful.

The problem is represented in the knowledge-base initially in the form of incomplete partial-plans corresponding to each sub-task. The solution of each sub-task, using a set of transforms, results into conflict-free partial-plans. These transforms, in fact, act to elaborate the partial plans incrementally to generate a set of alternative solutions.

The transforms operate independently of each other, and their invocation is driven by the present status of the knowledge base. The alternative combinations of the partial plans generated are synthesised and evaluated to help arrive at viable solutions.

#### 5.1 Re-allocation Strategy

Re-allocation is required when a delayed train(s) results in conflicts to one or more facility allocation plans. This implies that the partial-plans corresponding to these facilities become incomplete partial plans.

Suitable transformations are invoked in a data-driven manner, depending upon the current status of information such as train requirements and its schedules stored in the knowledge-base, to arrive at independent complete partial-plans for each facility. For example, transform for making the allocation plan of washing lines is invoked only if the train terminates at the station and requires servicing. Also, if the delay of a train is small and does not create a conflict at the platform, the corresponding transform will not be invoked. However, if a conflict is caused at washing line complex, its plan will have to be modified.

After the partial plans have been generated, the synthesizing transform works to analyze and integrate alternatives to arrive at a final allocation without violating any of the dependency constraints mentioned before (Figure 3).

#### 5.2 Re-allocation Guide-lines

There are some constraints which which have to be adhered to during the re-assignment process. These are enumerated below:

1. During re-allocation process, a train should not be moved more than once.
2. A limit is kept on the number of permissible changes made to resolve the conflicts because performing too many re-allocations disrupts other associated activities.
3. The re-allocation process is not invoked until the actual arrival time is within the re-allocation interval specified with respect to the current time. This is done to avoid unnecessary changes since a train which has been delayed may further get delayed changing the entire plan again.
4. A time limit is used to generate an exit condition to resolve situations when a solution does not exist within the given constraints. The system then switches over to manual mode.
STATALLOC(CSN, IPPN) Evolves incomplete partial plan IPPN by reducing conflict set CSN by allocating trains by:
   i. direct fit on some platform.
   ii. regulating their arrival time.

Transforms used -
REduce(T, CSN) Remove train T from the conflict set CSN.

Operators used -
MOVE(T, PFs, PFD) Move train T from PFs to PFD.
REGULATE(T, PFs) Resolve conflict by regulating arrival time of T on same platform PFs.
PLACE(T, PFs, PFD, IPPN) Place train movement in partial plan IPPN.

The STATALLOC transformation -
STATALLOC(CSN, IPPN) =>
   ∀T ∀PFD
   element(T,CSN) A element(PFD, PFset) A (MOVE(T,PFs,PFD) A FITS(T, PFD)) =>
   PLACE(T, PFs, PFD, IPPN))
   V ((REGULATE(T, PFs) A FITS(T, PFs)) =>
   PLACE(T, PFs, PFs, IPPN)))
   => REduce(T, CSN)

Figure 5 The STATALLOC transform.

v. If during the re-assignment a train falling outside the reallocation time-frame has to be moved, the movement is placed in the deferred assignment list.

5.3 Transformations.
The transform for generating the partial-plan (IPPN) for platform allocation is described in detail. Other transforms have been evolved on similar lines.

The status of the facility allocation at any time is represented by the berthing chart. During the planning process the knowledge-base also maintains a representation of:
   i. intermediate states of incomplete partial-plans i.e. (IPPN).
   ii. the set of conflicts (CSN) which have to be resolved to arrive at the complete partial-plan.

Initially the conflict set (CSN) consists of the train to be re-allocated. As a first step a transform is applied to reduce the CS by trying to assign the train to another platform which may be vacant at that time. If required the arrival of the delayed train may be regulated, within specified limits, until a platform becomes vacant to accommodate it. If the STATALLOC transform is able to resolve all conflicts, then the final solution is reached. Figure 5 shows the STATALLOC transform along with the other transforms and operators invoked by it.

If STATALLOC is unable to resolve the conflicts to arrive at a partial-plan, then the current incomplete partial-plan IPPN is expanded to yield all possible combinations of the next level of partial plans. This is achieved by the application of the EXPAND transform. The existing plans have to be evaluated to be able to select the most promising plan to be explored further. The EVALUATE operator uses heuristics to select plans which converge to a solution with minimal changes to the original state (Figure 6.).

EXPAND(CSN, IPPN) Generate new partial plans by moving a train from CSN to IPPN+1 and introducing the new conflict train into CSN+1. A weight is also computed for the node based on a set of rules.

Operators used -
EVALUATE(CSN, IPPN, W) Evaluate partial plan N yielding weight W based on a set of rules
   i. Prefer platforms with least no. of conflicting trains.
   ii. Prefer conflicting trains with lesser utilization of facilities.

Figure 6. The EXPAND transform.

Before selecting the next intermediate partial plan to work on, the allocation status has to be reset to the original status (S0). The best partial plan is selected according to weights computed and the status of the knowledge-base is set to conform to it (SN). These transforms are shown in Figure 7.
Figure 7. SELECT & STATUS transforms.

The following procedure is used to convert the incomplete partial-plan for platform allocation into a complete partial-plan.

Initialize (CS0 = Train to be re-allocated, IPP0 = 0)

LOOP: SELECT(IPPPN)
SET STATE(IPPN, S0, SN)
STATALLOC(CSN, IPPN)
IF (CSN = Null) /* All conflicts resolved */
Then
exit /* Solution Reached (IPPN) */
EXPAND(CSN, IPPN)
RESET STATE(IPPN, S0, SN)

6 Front End

A graphics interface is provided with a system of pull down menus to facilitate user interaction. After the allocation process, the user can interact with the explanation system to understand the system's reasoning in arriving at a particular solution. Some typical questions which can be answered are:

- Why an allocation was made?
- Why a particular allocation was not made?
- What movements were required to arrive at the new allocation?
- What would be the implications of a particular re-allocation?

The user can interactively change various parameters which result in generating alternate solutions to the same problem. EXDAFS also provides a graphic display showing the present state of berthing. The duration of occupancy of the facilities namely platforms, washing and stabling lines is depicted on the diagram. After a dynamic re-allocation, the changes are highlighted so their effect can be studied.

7 Implementation and Results

The system has been implemented using Prolog and 'C' on an IBM compatible PC/AT-386. Prolog was used to form the rule-base and the inferencing mechanism. Being a data driven language, Prolog provides a more conducive environment for logic programming. Prolog's power lies in its ability to infer new facts from existing ones, its emphasis on symbolic rather than numerical processing, and also its inherent capability of list processing and recursion. However, the implementation of any procedural programming in Prolog leads to a crudely structured code. For such areas, we have interfaced Prolog with 'C' to provide the necessary procedural language support[8].

EXDAFS is under trial at New Delhi Railway Station, one of the largest stations in India, which comprises of 11 platforms, 8 washing lines grouped into 4 complexes and 5 engine stabling lines. A database containing the required characteristics and specifications of these facilities is maintained. There are over 150 trains which arrive and depart from this station. The schedules and characteristics of these trains are also maintained in the static database.

The Master Berthing chart arrived at by EXDAFS is close to the original as prepared by the domain experts. It also performs the dynamic allocation within 3 to 10 seconds depending upon the complexity of the situation. Heuristic knowledge has been incorporated to accelerate this process by limiting the search to more promising paths.

At present the system helps to ease the decision making and planning task of the controller by providing explanations as well as allowing him to perform 'what if' analysis to project the results of alternate allocations he may want to try. One can also validate the feasibility of introducing a new train on that station given the timings and its characteristics.

We have tried to make a flexible system by taking into consideration most of the constraints which may arise in the context of Indian Railways. The system can be made operational for any other station by providing the station characteristics and train schedules. However, since a few stations have their own peculiarities, some changes in the rules and limits of parameters may also be required.

Example. A representative illustration of the platform re-allocation process is given in Figure 8. It shows the movements required to re-allocate train no. 9019.

Case: Delayed train = 9019 scheduled on platform 4.
New arrival = 7:30
New departure = 8:50
Results: No. of intermediate partial-plans created = 42,
No. of intermediate partial-plans analysed = 7,
Time taken = 2 seconds.
Movements: 2553 from platform 4 to 7.
1H1NK from platform 7 to 2.

8 Summary

EXDAFS performs static planning using a rule-based approach and re-assignment based on transformational synthesis. Modelling the re-assignment task as independent transforms eases solution development and system extension.

When allocations have to be made in real-time in an environment where schedules are continuously being disrupted and the allocatable facilities are relatively scarce or the constraints are numerous, the re-allocation process becomes complex and more time consuming. In such a situation the system can ease the decision making task of the controller by giving a solution in only a few seconds and also generating alternative scenarios. The system enables him to do perspective planning which will help him in taking timely decisions.

EXDAFS can easily be modified to act as a trainer. The use of AI techniques illustrated for planning strategic actions may also be used to enhance the level of expertise of the user by presenting plans under simulated conditions and also summarizing the reasoning through a
more elaborate explanation system.

We propose to enhance our system by interfacing it to the control panel in the power cabin, so as to capture status inputs directly, thereby reducing manual intervention. This will also enable it to implement decisions about the routing of trains in the interlocked yard.

Acknowledgements

The authors would like to thank Mr. N. C. Gupta, Executive Director (CRIS), for providing not only encouragement and motivation but also interacting with us, and thereby contributing in a meaningful way in carrying out this work.

References


