A BI-LEVEL DSS FOR ASSIGNMENT PLANNING: THE CASE OF PLATFORM ALLOCATION IN BUS STATIONS

RAQUEL BLANCO, JAVIER TUYA, B. ADENSO-DÍAZ

Engineering School at Gijón, Universidad de Oviedo, Spain

rblanco@uniovi.es, tuya@uniovi.es, adenso@uniovi.es

1 Introducción

Nowadays, Decision Making Support Systems are being used in many environments. The difficulty of efficient decision making under quick response requirements forces the use of computer-based systems in industry and services. Decision Support Systems (DSS) may be considered to be a new generation of Information Systems, the goal of which is to try to discover what would happen if a series of decisions are taken and/or to automatically provide these decisions so that they may be applied by the person managing them.

These systems, which began to appear in the early 1970's, may be defined (Sprague, 1980) as: (1) Computer-based systems (2) that help in decision making (3) in poorly structured problems (4) via direct interaction (5) with data and models of analysis. The generic character of the term DSS has given rise to specific systems aimed at solving concrete types of problems, such as for instance Enterprise Information Systems (EIS), Group Decision Support Systems (GDSS), Management Support Systems (MSS) or Executive Support Systems (ESS), all of which have the five aforementioned points in common.

The typical architecture of a DSS is based on the concept known as DDM (Data, Dialog and Model), a term coined by (Sprague and Watson, 1996). Data is the information used as the starting point for the decision-making process, Dialog is the information exactly as it is presented to and interactively managed by the user, and finally Model is an abstract representation of reality from which the DSS will generate the information for decision making.

A potentially interesting field of application of these systems is the management of the transportation sector. Many DSS have been reported in the air and railway transportation industry, including yield management, intelligent routes planning, rescheduling, or gate assignment in airports. However, not so many applications have been reported in the bus transportation sector, in spite of the huge number of passengers moved daily all around the world.

Among all the tasks related to bus station management, one of the important decisions, and one that greatly affects the quality of the service, is the allocation of platforms. A bad assignment procedure may be uncomfortable for frequent passengers, who often suffer changes in departure gates, or even worst, who may be affected by the lack of platforms. The assignment is complicated as a result of the frequent incidents that cause delays or advances in services (with the corresponding changes in allocation), and due to an increase in services that converts platforms into a limited resource.

The system proposed here was designed at the request of the management of the new bus station of the city of Oviedo, the capital of the Principality of Asturias, located in the North of Spain. The new station, which was inaugurated in 2003, handles a service density of 1100 bus per day, with 44 platforms available to cover these services. Not all the platforms may be used for any type of bus, nor do all the bus services have the same departure and arrival requirements. For instance, departure of a national service requires more previous time for platform allotment, since passengers usually carry a lot of heavy baggage. Thus, part of the system consists in deciding the arrival and departure platforms for each service offered at the station, in such a way that they satisfy a series of constraints.

In order to manage the station properly, the needs and preferences of the different companies that operate there as regards platforms need to be known. In the light of this data, an allocation must be made that is operative for a particular day and passengers must be informed via information panels both of initial allocations as well as the changes produced throughout the day.

In this chapter, we shall discuss the work carried out, an intelligent DSS system for the daily management of bus station platforms employing rule-based models, and the experiences obtained with the implementation of the system developed for this real environment. We shall start off by presenting the state of the art of the models defined for this allocation problem, to then go on to present the conceptual description of the two-level system, to describe in detail the on-line level and the discussion of the results.

2 Theoretical background

Concerning decision making in the transport industry, many problems have been identified for which models have been defined attempting to give a satisfactory response to the interests of the person taking the decisions. Several of these problems are related to allocation, whether of schedules (Adenso-Díaz et al, 1999) or resources such as engine drivers to trains, crews to airplanes, etc.

Of the problems associated with resource allocation, one of special interest is that of gate allocation. The majority of these types of problems are related to gate allocation in airports. Efficient allocation will basically attempt to oblige passengers in transit to move around as little as possible. However, some authors (Yan and Huo, 2001) incorporate multicriteria decisions, such as the minimization of passenger waiting time, that may be extended to other criteria, such as different companies departing from the same areas, avoiding congestion in certain areas of the airport, having the arrivals of flights with greater movement of passengers close to the baggage pick-up area, etc.

Since the mid 1980's, in which one of the first automatic systems of gate allocation was published (Hamzawi, 1986), we may identify two main types of models in the literature: those based on expert systems (perhaps in conjunction with results obtained by means of simulation) and those that attempt to find exact solutions to the problem.

Regarding the former, (Gosling, 1990) published one of the first studies that served as the basis for many other subsequent studies, such as that of (Su and Srihari, 1993). (Srihari and Muthukrishnan, 1991) carried out a review of the literature on the use of ES in the gate allocation problem. Subsequently (Yan et al., 2002) solved the case of Taipei airport by means of the use of simulation, introducing the possibility that specific flights suffer certain delays.

Other authors address the problem via the use of exact methods or metaheuristics, such as the use of LP (Haghani and Chen, 1998; Gill et al 1997), QAP-type models with temporal constraints (Bolat, 2000; Ding et al, 2004), or network models (Yan and Chang, 1998) applied to Taiwan airport. (Bolat, 2001) identified the conditions for which exact solutions may be obtained in polynomial time, and defined a genetic algorithm to obtain alternative solutions. Among other alternative approximate-type models, we may highlight the one proposed by (Cheng, 1997), who defined a hybrid rule-based system embedded with a heuristic assignment procedure for Singapore airport.

In contrast, outside the sphere of airports we are not aware of similar work having been published for other types of transport, such as railways and buses. However, although the criteria of interest may be different, the problem is of major importance in both these areas, above all if the station registers a substantial number of services. In this case, the distance that passengers have to move around is perhaps not so important, given the more reduced size of these stations. However, other possible interests appear that condition efficient allocation: not all the platforms are equally valid depending on the size of the buses; services that cover certain geographical areas usually have assigned platforms near to one another so that regular customers may associate platforms with destinations; competing companies may want to have their platforms far from one other; regional services may wish to be close to the main doors, as they will be the ones that transport the most passengers; it is not convenient to put two long-distance services (usually with passengers carrying a lot of luggage) next to each other due to the bothersome accumulation of people and baggage; delays are usually more frequent than in other means of transport due to traffic jams in cities, etc.

Numerous constraints can be found, which change with time, associated with platform allocation in bus stations. If the number of services is high (hundreds per day), trying to fulfill them all with low response times so as to give an immediate solution to the necessary assignment when events arise that demand it turns out to be a very difficult task unless some kind of automatic tool is used that helps in the decision-making process. Such a tool should permit the definition of the type of assignment rules mentioned above on the part of the decision maker.

From this point of view, the tool presented here is original, since it is the first to be developed for this means of transport. It cannot be compared to other systems developed for the airline industry, since both the constraints and the limitations are very different. However, the complexity as regards the size of the solution space to explore might be considered similar. On the other hand, exact solutions in this kind of environment, in which the assignment rules may even be contradictory, do not seem to be the most appropriate, since on occasions the objective is simply to minimize the unsatisfied rules. For example, imagine that the aim is for the departures of two competing companies offering international services to be far from one another, but that at the same time the goal is for the departures of all the international services to occupy a specific area of the station. At the same time, one characteristic that any system that aims at solving this problem must have is that of being able to respond rapidly, since decisions must be taken rapidly when delays occur in departures or any other cause that provokes re-assignments is produced. .

In the following sections, we show how different levels can be identified in the planning process, depending on whether the decisions to be made are long or short term. Different tools should be used in each case: for long-term planning, a rule-based system prepares off-line plans for different periods and characteristics; for the short term, on-line systems should be developed that try to amend the selected off-line plan for the specific day when unplanned events occurs that force changes in the assignment. In this case, a rule-based approach seems to be both sufficiently rapid as well as efficient for the reassignment.

Although another alternative approach could be a one-level system taking all the decisions in just one step, in this case a bi-level DSS seems to be more appropriate. The carriers operating in the bus station desire to keep a stable allotment for the services (passengers always should take the same buses leaving from the same platforms), which only changes when an incident occurs. In addition, the decision algorithms are simplified, since the off-line module prepares plans according to the rules given by the manager, and the on-line module modifies the plans regarding the incidents.

Similar situations with two levels and valid similar approaches could appear in other environments where an on-line response is needed to modify pre-selected plans. In this chapter we shall present the logic behind the design of this bi-level model, the details of the algorithms of the on-line level underlying this design, and some examples of implementation.

3 Conceptual description

The development of this type of system differs substantially from that of conventional information systems. From the viewpoint of the decision-making process, different phases are usually considered (Turban and Aronson, 1998): Intelligence phase (identification of the problem), Design phase (formulation and validation of a model), Choice phase (formulation of the solution proposed to the model), and finally Implementation phase (achieve the objective of solving the problem that gave rise to the DSS).

We now go on to give a brief description of the first two phases of the development of the system within the context of the general process of decision making. Choice and Implementation will be dealt with in subsequent sections.

3.1 Intelligence phase

The need to have a system to help in the effective assignment of a platform to the buses arose due to the opening of a new station that was to be used by different carriers who had their destinations, points of origin or traffic in the city. This meant a change in the ownership of the problem, the responsibility of which passed to an organization that managed the station offering said service to the different carriers and users. The previous situation was one in which each of these carriers managed their own traffic, which took place in different locations in the city. This means that the interests and preferences of different companies that are competing one with another must be managed in the new station, thus requiring the establishment of the most adequate allocations. On the other hand, from the end user’s perspective, the concentration of services means that it is necessary to adapt the distribution of these services to the user, regardless of the company that operates said service.

The structure of the system is defined as a function of inputs, outputs and processes. Inputs will be defined by:

- The different needs of the carriers (daily routes and schedules)
- The availability and arrangement of platforms in each period of time.
- A set of rules defined by the station manager (who is the decision maker and the user of the system). These rules may be customized (added, eliminated or modified) by said user, thus altering the outputs produced by the system. These rules will be called Dynamic Knowledge because they represent a set of configurable and changing policies, in contrast to the heuristics included in the DSS, which are not modified by the user and are part of the process.

Outputs will be made up of the plan for assigning platforms to buses at each moment.

3.2 Design phase

The type of problem being addressed here may be considered a Programmed problem in which two main goals exist when carrying out the valid assignments for a particular day. Firstly, the theoretical assignment that must initially be built trying to satisfy the responses of the transport companies; and secondly, the difficulty derived from the daily alterations to the theoretical plan due to unforeseen delays and events that mean that reassignments must be made with respect to the theoretical plan.

Because of these two difficulties, the designed system has been structured at two different levels (Fig. 1), which allows the different inputs/outputs mentioned above to be detailed.

- An off-line module: this module generates the general platform allocation plans for the bus station, which are valid for different periods with specific common schedules (Summer, Easter, etc.). The output from this module provides valid assignments generated using different rules prepared to be chosen for a specific set of days (Assignment Plan). All these assignments make up a database with different options (one for each set of chosen rules) that are valid for different types of days (service schedule).

- An on-line module: This module runs during the daily operations at the bus station, taking as input an assignment plan corresponding to the day that has been generated as output from the off-line module. However, it is quite common for unforeseen events to occur (delays, an increase in the number of buses on a service, etc.) that require a change in one or other of the current assignments. Thus, incidents (e.g. delays in schedules) and changes (e.g. an increase in the number of buses on a service, or the creation/suppression of services) are also added to the inputs. This module carries out the reprogramming, taking into account the defined constraints and the current configuration of the station, indicating to the user the changes that he must carry out in the initially programmed assignment (Daily Re-Assignments).

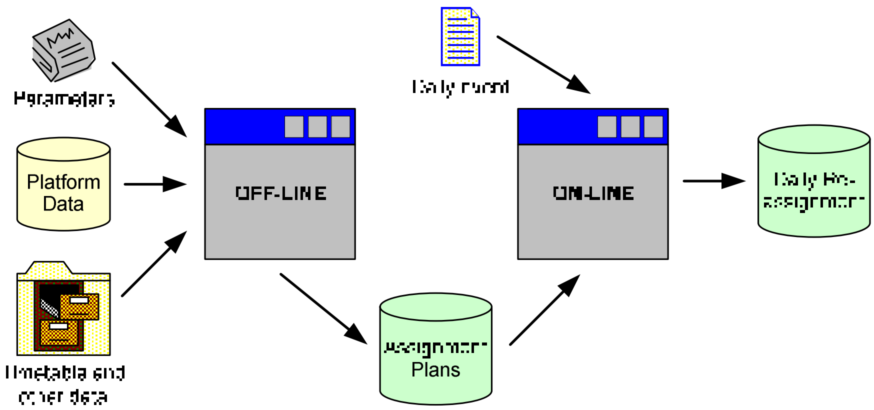

Fig. 1. Architecture of the system

The system model is of the heuristic type, based on the experience accumulated by the decision makers. The process thus uses the heuristics embedded in the model, together with the requirements that define the Dynamic Knowledge and which are parameterized by the user as a set of rules. The goal is to find a good enough assignment by satisfying the rules.

The variables that are managed are:

- Result variables (the measure of cost/goodness). This is measured by the fulfillment (0 to 100%) in the solution of the rules defined by the user

- Decision variables (the output of the algorithm). In the case of the off-line model, the Assignment Plan for each day, and in the on-line model, the changes to carry out with respect to the initial plan.

To predict and measure the outcome of each alternative, the system evaluates, the fulfillment of the rules introduced and presents the one that obtains the best score, as well as a set of other alternatives with good scores, in case of the best one obtained not being to the liking of the decision maker for whatever reason not previously contemplated (Fig. 2).

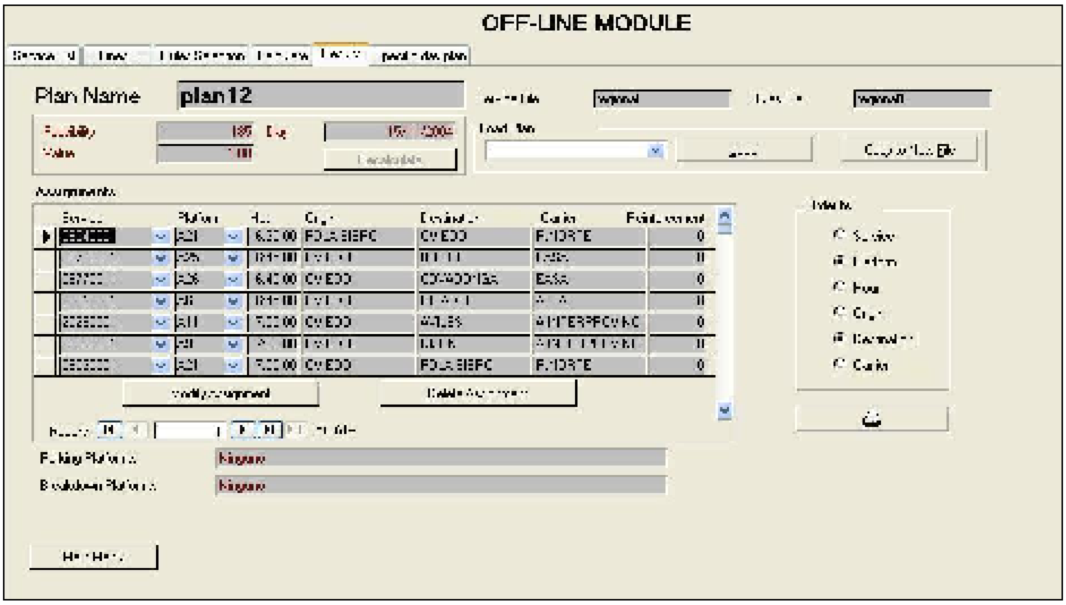

Fig. 2. Output from the off-line module: platform allocations (central part of the screen) and evaluation of their quality (upper left)

4 System description

Within the above hierarchization, the on-line module is in charge of carrying out the daily management of the platforms, taking into account all the incidents that are produced throughout a particular day. The on-line module proposes new platform allocations with the aim of resolving the incidents that have been produced.

The basis of the on-line plan corresponding to a particular day (Fig. 3) is a specific off-line plan (generated by the off-line module) that contains the general allocations of platforms. When incidents arise, the on-line module implements new allocations, which modify the on-line plan without producing changes in the original off-line plan.

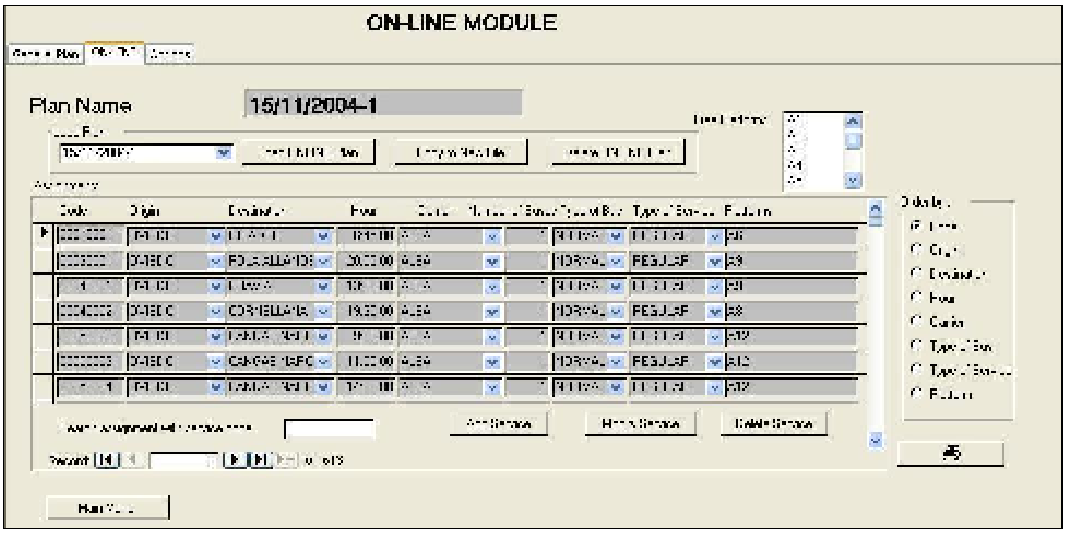

Fig. 3. On-line module work screen

Incidents may arise owing to three causes:

- Incidents that alter the arrival and/or departure of buses, giving rise to delays that may mean that two services demand the same platform at the same time. When faced with this event, the need arises of finding a new allocation for one of the services.
- Changes in the number of buses on a particular service, since the decision might be taken as a result of passenger demand to reinforce a service with more buses, for which new platforms need to be found.
- The creation of new services that were not contemplated in the general off-line plan on which the on-line plan is based. In this case, a totally new assignment will have to be made for the service.

The re-allocation of platforms is carried out taking into account a series of predefined fuzzy rules, which the operator may select on an options screen (Fig. 4). Moreover, there exist two general rules that may affect any incident (see bottom of figure 4). The first of these determines that platforms already assigned to another company will never be proposed again, even though this means that the re-assignment cannot be carried out (hard constraint). The second rule will attempt to fulfill, as far as possible, the re-assignment of platforms that have free spaces on both sides or only on one side. However, since this is a soft constraint, if this criterion can not be fulfilled by any of the candidate platforms, the re-assignment to the best of them, according to the rest of the rules, will equally take place.

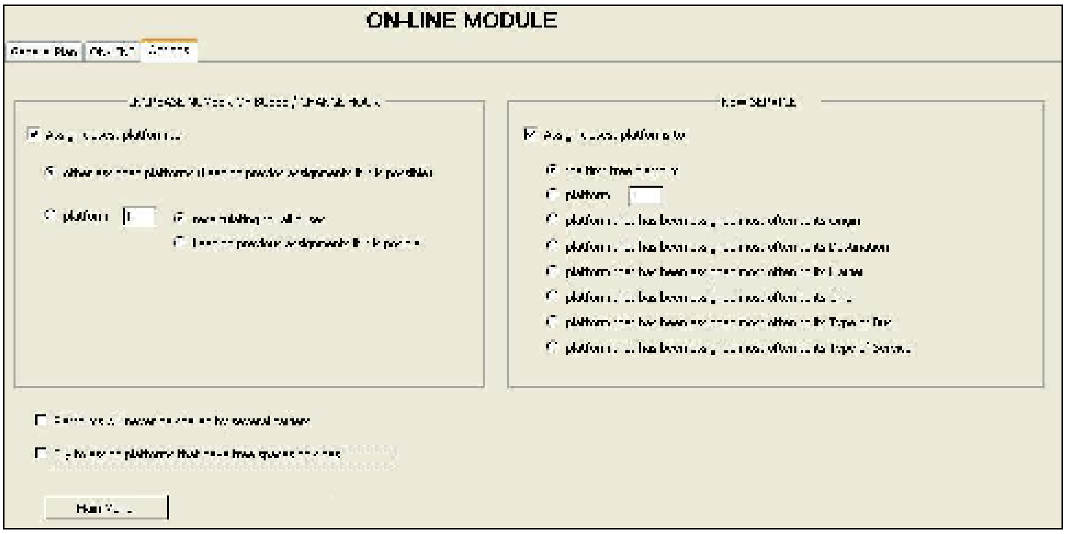

Fig. 4. Options for re-assigning platforms, depending on the type of incident (a new service, change in the arrival and/or departure times, more buses on a service).

When a change in the arrival and/or departure times of a service is carried out, the rules that are applied during the re-assignment of platforms are of the type "assign the closest platform to the pre-assigned one", "assign the closest platform to a particular one" or "assign the first free platform" (left side of figure 4). The rules defined in this way are of approximate assignment and are based on the measure of closeness existing between two platforms, with the goal of finding the one that most approximates the one taken as a reference. This measure is obtained by the function *closeness(d,k)*, which takes values in the interval [0, 1] and indicates the degree of acceptance of closeness between any two platforms d and k, and is defined as:

$$\text{closeness}(d,k) = \begin{cases} 1 & \text{if} \quad |d-k| \le 1 \\ 1 + \dfrac{1-|d-k|}{threshold - 1} & \text{if} \quad 1 < |d-k| < threshold \\ 0 & \text{if} \quad |d-k| \ge threshold \end{cases} \qquad (1)$$

Moreover, these rules are used to determine which is the leading platform D in the search process, i.e. the platform that is taken as a reference in the calculation of the measure of closeness.

On the other hand, the general rule that determines whether platforms assigned to other carriers can or cannot be used affects the creation of a table listing the platforms that are free in the period of time needed for the altered service. This table is called *FreePlatforms* and is the basis of the search performed by the platform re-assignment algorithm.

The specific rule that indicates whether it is preferable to assign platforms with free spaces on each side has an effect on the search process. First, the platform to re-assign must be searched for among the D_i platforms present in the *FreePlatforms* table, whose adjacent platforms D_{i+1}, D_{i-1} are also present in said table. If no platform exists with these characteristics, the search is carried out among those that have one of their adjacent platforms in the *FreePlatforms* table. Last of all, the search is carried out over all the platforms in *FreePlatforms*.

Summarizing, the algorithm that carries out the re-allocation of platforms after the modification of the arrival and/or departure time of a service is the following:

1. Create table FreePlatforms
2. Determine leading platform D:
    - IF option = closest platform to platform x THEN
        D = x
    - ELSE IF option = closest platform to pre-assigned platform THEN
        D = pre-assigned platform to the first bus of that service
    - ELSE
        D = 1
3. Iterate to perform the assignments:
    - $\forall$ bus $b_i \in$ Service S_j
    - IF option of spaces between assignments was chosen:
        - IF $\exists$ platforms $\in$ FreePlatforms with spaces at both sides THEN
            Determine platform D_i / *closeness*(D, D_i) = max *closeness*(D, D_k), $D_k \in$ FreePlatforms $\wedge$ $D_{k-1} \in$ FreePlatforms $\wedge$ $D_{k+1} \in$ FreePlatforms
        - ELSE IF $\exists$ platforms $\in$ FreePlatforms with free space on any side
            Determine platform D_i / *closeness*(D, D_i) = max *closeness*(D, D_k), $D_k \in$ FreePlatforms $\wedge$ ($D_{k-1} \in$ FreePlatforms $\vee$ $D_{k+1} \in$ FreePlatforms)
        - ELSE

Determine platform D_i / *closeness*(D, D_i) = max *closeness*(D, D_k), $D_k \in$ FreePlatforms

- IF option of spaces between assignments was not chosen:

Determine platform D_i / *closeness*(D, D_i) = max *closeness*(D, D_k), $D_k \in$ FreePlatforms

- Store assignment of platform D_i for bus b_i
- Delete platform D_i from FreePlatforms

5. Output the assignments made

Another incident that modifies the on-line plan of a specific day is the change in the number of buses on a service. This may consist in the reduction in the number of buses on the service (a decrease in the number of passengers with respect to estimates), in which case the platforms that were reserved for these buses will be freed, or in the increase in said number, which leads to the application of the approximate assignment rules as described for the above incident. The actions that the platform-assignment algorithm has to carry out are similar to those used in the above incident. The only difference lies in the possibility of recalculating the platforms for all the buses when the approximate assignment rule is of the type "assign the closest platform to a particular one". In this case, the platforms occupied by the buses on the service are freed to form part of the *FreePlatforms* table, to then proceed with the allocation of platforms for all of them, taking as a reference the leading platform and taking into account the aforementioned general rules.

The third type of incident that modifies the on-line plan of a specific day is the creation of a new service (right side of figure 4). The platform re-allocation algorithm is similar to the one described for the incident of the change in the arrival and/or departure time of a service. The difference resides in the obtaining of the leading platform, since the criteria of closeness that the system provides are different. This is due to the fact that, as no previous allocations exist, the criteria of prior incidents in which the pre-assigned platforms are taken into account cannot be employed. In this case, the rules carry out the search among the free platforms closest to a particular one or may follow assignment frequency criteria (i.e. assign a service to the free platforms that have been assigned most often in relation to destination, origin, carrier, etc.). To do so, it is necessary to analyze the information stored in the system's database.

As already mentioned, prior to the use of the on-line module, it is necessary to generate general plans from among which a plan for the specific day that is being managed is chosen. To do so, the off-line module uses a series of rules defined by the station manager (Dynamic Knowledge). These rules may be of two kinds: direct allocation rules (of the type “services with Madrid as their destination will depart from platforms 10 to 15”) and approximate allocation rules (of the type “services to Barcelona will depart close to those that go to Madrid”). The former indicate some type of strong constraint that must be complied with by the solution, while the latter are of the fuzzy kind and are built with close/far relations. Each one of the defined rules has an associated weighting $w_r \in \{1..9\}$ that indicates the importance assigned by the person in charge. The system orders the services taking into account the number of alternative platforms available, considering the direct allocation rules and incompatibilities in scheduling. The service with the least number of alternatives will be the first to be assigned a platform, determined by the approximate allocation rules. Details of the off-line module calculation model appear in (Adenso-Díaz, 2003).

5 Discussion of implementation and experiences

Regardless of the technological aspects of developing a decision-making software program, the implementation of a DSS consists in the effective introduction of the change in the organization with the acceptance of this system on the part of its users.

Apart from the precision of the DSS, the most important risk is that of not gaining its acceptance on the part of its users, which is especially critical if use of the system is not mandatory, as is the case in point. This problem has been called the “problem of implementation” by (McCown, 2002), who discusses these issues in depth, identifying four different functions in which a DSS can be successful. Specifically, two of these are applicable to the system described here:

- Decision making for highly structured tasks: in this case, the exploration of different platform-allocation alternatives taking into account the rules and preferences provided by the actual user.

- The need to meet external regulatory demands: in this case, the need is more pragmatic than regulatory, due to the change in station that necessitates the carrying out of allocations that satisfy the different needs of different carriers, which may at times be conflicting.

In line with the above ideas, the developed DSS may be considered to be aimed in the adequate direction.

Many other factors affect the successful implementation of Information Systems in general and DSS in particular. The two most important of these that facilitate the integration of these systems in business processes are their usefulness and ease of use, factors that have been identified in the Technology Acceptance Model (Davis, 1989) and whose results are consistent, as has been demonstrated by means of multiple empirical studies and via meta-analysis (Legris et al., 2003). In the case of this DSS, ease of use is attained due to the fact that the system guides the user in his tasks and the amount of screens and different information that he must introduce is not large. Moreover, the style of interface used is known by the user, who also participated in its design. Once the rules have been introduced, the daily operating of the system requires no more input than the selection of a few basic criteria on a screen in order to immediately obtain outputs. The calculation time in this system is around five minutes for the off-line module, and does not reach 15 seconds in the on-line module. Although ease of use is not the most significant factor in the use of the system, previous studies show its positive influence on the perceived usefulness of the system, by having a positive influence on the simplification of the tasks carried out by the user.

Participation of the user in the design of the DSS is another of the major factors that increase usefulness and use (Lawrence et al., 2002), an influence that has also been confirmed by means of meta-analysis (Mahmood et al., 2000), who establish a ranking of nine factors that positively affect user satisfaction. Among these, user involvement obtains the highest ranking. In this case, user involvement was very significant throughout the entire process of development of the system, since the decision maker himself actively collaborated from its specification until its final development. What is more, the project was carried out on his initiative. The remaining factors in the ranking also score high in this case, especially the perceived usefulness of the system, as in TAM models, user experience, organizational support and user attitude toward information systems

6 Conclusions

This chapter presents the experience of developing a DSS to manage platforms in a bus station in which the number of services leads to saturation of available platforms and in which there are numerous and conflicting assignment interests. The problem was solved in two phases: off-line elaboration of diverse seasonal plans (from among which the initial one to be effective each day is chosen) and on-line modification of the initial plan when faced with unforeseen events that make its modification necessary.

The system is found to be user-friendly for a decision maker (the station manager) who has no experience in computing, and supplies solutions in an acceptable response time that cover to a very high percentage of the majority of constraints established by means of the chosen rules. Due to the complexity of the exploration of the solution space, a manual solution would always fall far below the level of fulfillment achieved by the designed DSS.

References

Adenso-Díaz B., González M.O., González P. (1999) On-line timetable re-scheduling in regional train services. Transportation Research-B, 33 (6): 387-398

Adenso-Díaz B. (2003) Rule-based system for platform assignment in bus stations. 1st Multidisciplinary International Conference on Scheduling, MISTA-2003, Nottingham (UK)-August 13-16

Bolat A. (2000) Procedures for providing robust gate assignments for arriving aircrafts. European Journal of Operational Research, 120: 63-80

Bolat A. (2001) Models and genetic algorithm for static aircraft-gate assignment problem. Journal of the Operational Research Society, 52 (10): 1107-1120.

Cheng Yu (1997) A knowledge-based airport gate assignment system integrated with mathematical programming. Computers & Industrial Engineering, 32 (4): 837-852

Davis F.D. (1989) Perceived usefulness, perceived ease of use, and user acceptance of information technologies. MIS Quarterly, 13: 319-340

Ding H., Lim A., Rodrigues B., Zhu Y. (2004) New heuristics for over-constrained flight to gate assignments. Journal of the Operational Research Society, 55 (7): 760-768

Gill C.O., Badoni M., Yu C. (1997) A knowledge-based airportgate assignment system integrated with mathematical programming. Computers & Industrial Engineering, 32 (4): 837-852

Gosling G.D. (1990) Design of an expert system for aircraft gate assignment. Transportation Research-A, 24(1): 59-69

Haghani A., Chen M-Ch (1998) Optimizing gate assignments at airport terminals. Transportation Research-A, 32 (6): 437-454

Hamzawi S.G. (1986) Management and planning of airport gate capacity: a microcomputer-based gate assignment simulation model. Transportation Planning & Technology, 11: 189-202

Lawrence M., Goodwin P., Fildes R. (2002) Influence of user participation on DSS use and decision accuracy. Omega, the International Journal of Management Science, 30: 381-392

Legris P., Ingham J., Collerette P. (2003) Why do people use information technology? A critical review of the Technology Acceptance Model. Information & Management, 40: 191-204

Mahmood M.A., Burn J. M., Gemoets L. A., Jacquez C. (2000) Variables affecting information technology end-user satisfaction: a meta-analysis of the empirical literature. Int. J. Human-Computer Studies, 52: 751-771

McCown R. L. (2002) Changing systems for supporting farmers' decisions: problems, paradigms and prospects. Agricultural Systems, 74: 179-220

Sprague R.H. (1980) A Framework For the Development of Decision Support Systems. Management Information Systems Quarterly, 4: 1-26

Sprague R.H., Watson H.J. (1996) Decision Support for Management. Prentice Hall, New Jersey

Srihari K., Muthukrishnan R. (1991) An expert system methodology for aircraft-gate assignment. Computers & Industrial Engineering, 21 (1): 101-105

Su Y.Y., Srihari K. (1993) A knowledge based aircraft-gate assignment advisor. Computers & Industrial Engineering, 25 (1): 123-126

Turban E., Aronson R. (1998) Decision Support Systems and Expert Systems (Management Support Systems). Englewood Cliffs, NJ: Prentice-Hall, Chapter 2

Yan S., Chang C-M (1998) A network model for gate assignment. Journal of Advanced Transportation, 32(2): 176-189

Yan S., Huo C-M (2001) Optimization of multiple objective gate assignments. Transportation Research-A, 35: 413-432

Yan S., Shieh C-Y, Chen M. (2002) A simulation framework for evaluating airport gate assignments. Transportation Research-A, 36: 885-898