Seat allocation for massive events based on region growing techniques

Abstract

Massive events, as sportive events, involves a huge amount of spectators. Citizens that wish to attend buy a ticket that allows them to seat in a given zone of the stadium with several features, but usually they do not buy the physical seat at the sport ground. Then, one of the duties of the organizing committees is to allocate the different localities of the enclosure to the persons that have bought tickets. In the seat allocation process, the ticket category, groups, ranks, distribution rules and many other factors should be taken into account. In this paper we present a method in order to support the seat allocation process based on a region growing technique inherited from the Computer Vision field, with which a first candidate to the solution is obtained. Then, with a local search method, the solution is improved. The experimental results have been tested with the data of the 2003 Grand Prix Racing (F1).

1 Introduction

Massive events, as sportive events, involve a huge amount of spectators. This enormous flow of people requires a good organization and control so that no problems arise in its development. In this kind of events, as for example the Olympic Games, the Grand Prix racings (Formula One, F1), and soccer world cups or eurocups, citizens that wish to attend usually buy a ticket that allows them to enjoy the competition in some kind of seats of the stadium with several features, but they do not buy the physical seat at the sport ground. Then, one of the duties of the organizing committees is to distribute the different available localities of the enclosure to the persons that have bought tickets.

Distributing persons is a complex process when myriads of tickets should be assigned to multiple stadium zones. An additional difficulty is the fact that often, tickets are not sold to a single person but in group, so a team of people come together. This is the case of sponsors who, taking advantage of the sportive events to make publicity, they receive from the organization a given amount of entrances to share among the different members of the company. There are also other collectives, as for example, safety and security staff, first aid and lifesaving people, selected members of the organizing committee, and very important persons (VIP) that occupy some of the seats. In figure 1, a typology of the different entities involved in the F1 championship is shown.

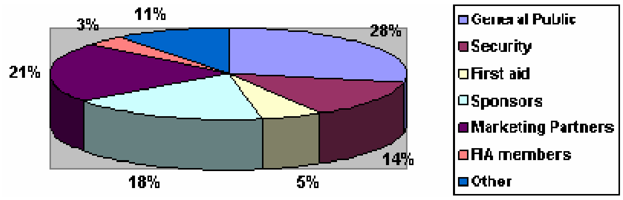


Figure 1. Entities typology for the F1 Grand Prix racing.

Nowadays, the distribution of tickets to the different physical seats of the sport ground has been performed manually. This task takes a lot of time, is repetitive and tedious, and what is more important, it is quite complex due to the constraints imposed by the organization committee. Such constraints can involve the fact of distributing the spectators widespread so that, in case that not all the tickets had been sold, the stadium looks like full.

Therefore, a system that supports the assignment of tickets to seats, taking into account categories, groups, distribution constraints, etc., can help a lot to the people involved in the allocation process. Our work is concerned with the development of a tool that supports such allocation task. Particularly, we have applied search techniques combined with region growing techniques from the Computer Vision field, obtaining significant results. The developed techniques have been applied in the data provided by the FIA (*Federation Internationale de l'Automobile*) regarding the 2003 Formula 1 championship, one of the most important sportive events regarding the number of participants.

This paper is organized as follows. In section 2, we describe the optimization problem we are dealing with. We continue in section 3 by providing the cost function used in the optimization method. We proceed in section 4 by describing our method. In section 5 we analyse the results obtained and in section 6 we relate our research to some previous works. We conclude with some conclusions and discussion in section 7.

2 Seat allocation for massive events

Seat allocation in massive events is characterized by three main components: ticket groups (TG), seats and distribution rules established by the organization. In this section we first provide the description of all this features, and then we formulate the allocation problem.

Ticket groups (TG)

In the sportive events scenario, costumers are provided by a set of tickets that are split in different ticket groups (TG). A ticket group is composed by the following attributes: Request, TOG, customer id, amount of tickets, category, type, price type, dispersion and rank (see table 1).

Request	TOG	Client Id	Number of Tickets	Category	Type	Price Type	Dispersion	Rank
1	1	1	50	1	Purchasable	Regular	False	1
1	2	1	250	1	Purchasable	Regular	True	6
2	1	2	10	1	Complimentary	Regular	False	1
2	2	2	30	1	Purchasable	Regular	False	6
2	3	2	150	1	Purchasable	Regular	True	11
3	1	8	40	1	Purchasable	Regular	False	10
3	2	8	150	1	Purchasable	10%	True	135

Table 1. Examples of ticket groups for the F1.

Zone	Row	Col	Categ	Sector	Type	Status	Price Type	Rank	Assign.
1	1	1	VIP		Purchasable	Std	Regular	1	
1	1	2	VIP		Purchasable	Std	Regular	2	
1	10	10	VIP		Purchasable	Std	Regular	10	
10	1	1	1		Purchasable	Obstr	Regular	20	
5	1	1	1	Confort	Purchasable	Std	Regular	40	
6	5	5	1	SkyBox	Purchasable	Std	Regular	100	
6	6	1	1		Complimentary	Std	Regular	101	
6	6	2	1		Purchasable	Killed	Regular	102	
11	7	7	1		Purchasable	Std	10% Discount	103	RedBull
30	12	1	2		Purchasable	Std	Regular	500	RedBull

Table 2. Examples of seat configuration for the F1.

Category relates the kind of seat to which the ticket should be allocated: Tribune VIP, Comfortable, 1, 2, 3 … Type can be either Complimentary (has been provided as a gift) or Purchasable. Price type informs whether the ticket has been purchased at the normal price (regular) or some kind of discount has been applied. In this latter case, type price is set to the discount applied (for example, 10%). The dispersion attribute is a flag that is activated (on) when the groups are submitted to a dispersion criteria (see distribution rules). Finally, the rank attribute is associated to the priority of the group in the assignment process. The range of the rank attribute is from 1 (the higher) to infinite (the lower).

Seats

The amount of seats available for each event depends on the stadium that often is divided into different categories and zones due to its huge dimension. For example, in figure 2, the scenario of one F1 Grand Prix is represented. Each seat is characterized by the following attributes: zone, row, column, category, sector, type, status, price type, rank and reservation. Table 2 shows and example of seats for the circuit of figure 2.

The zone, row and column are related to the real place in which the seat is physically located. The category, type, price type and rank, means the same as for the TG. Regarding the latter, seats of a given zone usually have the same category, but they can have different ranks. The sector is related to the services provided in the zone, as cushions (*comfort*), catering (*hospitality*) or standard services. The status attribute is related to the visibility of the seat. The usual value of the status attribute is standard, but it can also be obstructed or killed. Finally, the reserved attribute indicates if the seat is already booked by either a given sponsor of for some security or safety reasons. This attribute is also useful for giving the possibility of allocate a semi-occupied stadium, or allocating the entire stadium in different phases if required.

Figure 2. F1 racing scenario. Zones corresponding to the physical seats are marked by letters, from A to N.

Regarding the zones, their seats are distributed in several different forms. They can be represented as a nxm table, in which each cell keeps information about a seat. The table can contain blocked cells when, for example, some of the seats do not exists due to the particular structure of the zone (i.e. non squared zones).

Distribution rules

Each organizing committee can establish a particular set of rules in order to perform the assignments of group tickets to seats. However, two main kinds of rules should be distinguished: mandatory and optional rules. The former should always be satisfied, while the latter give the hints to produce optimal solutions.

The mandatory rules are the following:

- RM1. Each TG should be assigned to seats of the same category. For example, a Tribune VIP TG should be assigned to a Tribune VIP seat.
- RM2. Each TG should be assigned to seats of the same status. For example, a standard TG cannot be assigned to an obstructed seat.

These rules imply that a preliminary filtering should check if the category and status of the tickets sold are less or equal to the category and status of the seats available, and to detect possible overbooking situations.

Regarding the optional distribution rules, they are particular to each competition. For example, the FIA rules for the F1 championship are the following:

- RO1: TG and seats should agree regarding the sector, type and price type that the TG has.
- RO2: Big TGs are divided into subgroups (TS) according to a given sub-group size, allowing some margin deviation and a remainder. These

parameters (group size, deviation and remainder) are provided in each event. Each subgroup inherits the attributes of the group (category, rank, etc.).

RO3: The ranks of the tickets of the TG should agree as much as possible with the ranks of the seats assigned.

RO4: A maximum and a minimum amount of tickets of one group (having more tickets than the minimum) are allowed in the same row.

RO5: Never leave one ticket alone (of a group having more then one ticket) (see figure 3 a)

RO6: If some tickets of a TS do not fit in a single zone, the TS can also be split while maintaining a minimum number of tickets in each part.

RO7: Two TS of the same TG having the dispersion flag activated, either cannot be assigned to the same zone, or can be assigned to the same zone if there is some distance between them (measured in number of seats).

R08: Avoid leaving empty seats at the edge of rows (see figure 3 b).

R09: When not all the tickets have been sold, there should be an uniform distribution (sparsity) of the assigned seats in a given zone and in the overall scenario, in order to give the appearance that the zone and the entire stadium is fuller than it really is (see figure 3 c).

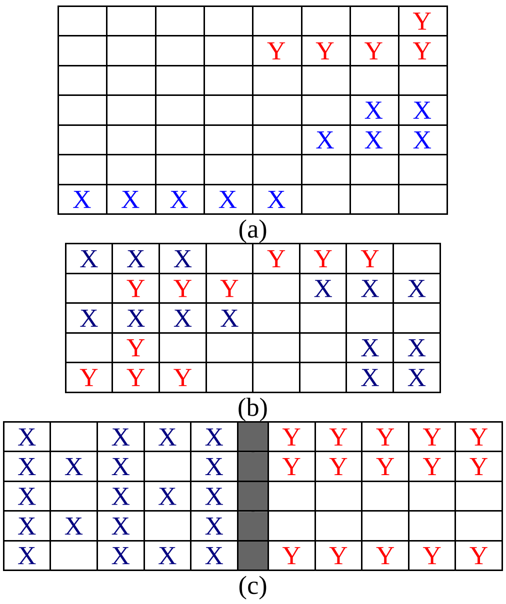


Figure 3. Examples of good (X) and bad (Y) seat allocations. In (a) some tickets are leaving alone in a row; in (b) there are some empty seats at the edge of rows; in (c-right) the distribution of allocated seats is not uniform.

The problem

Once the different components of our problem have been defined, namely, the tickets groups (split in several sub-groups), the seats (and zones) and the distribution rules, the seat allocation problem can be defined. It consists on finding seats for each ticket of a group, so that the mandatory and the selected optional rule distributions are satisfied and the optimal ones chosen.

3 The fitness function

In order to operationalize the optimization process based on the optimization rules, we have defined the fitness function of a candidate solution, GF. This function tries to measure the distribution degree and fitness of the different groups in the allocation, penalising the fact of leaving some tickets unassigned. It has been defined as follows:

$$GF = \frac{p_{GTOS} \cdot GTOS + p_{GD} \cdot GD}{p_{GTOS} + p_{GD} + nu^{p_u}} \qquad 0 \le GF \le 100 \tag{1}$$

Where GTOS is the fitness of the groups, p_{GTOS} is the weight of the GTOS, GD is the sparsity of the groups, p_{GD} is the weight of the GD (being zero when R09 is not applied), nu is the number of unassigned tickets (individuals), and p_u is the weight regarding the number of unassigned tickets. All the weighs used in equation (1) have been provided by experimentation as well as most of the weights used in our method (see table 3).

In the remaining of this section, the different components of the functions are explained. See also (Authors, 2005) for further details.

CONSTANT	DESCRIPTION	DEFAULT VALUE
Y	constant indicating the distance between two overlapped seats	2.5
X	constant indicating the distance between two consecutive seats	1
p_{TOSR}	TOS Ranking Fitness weight	1
p_{TOSJ}	TOS Joint Fitness weight	1
p_{BTOS}	Block TOS Fitness weight	1
p_{BD}	Block Dispersion Fitness weight	0 (no sparsity) 1 (sparsity)
p_{GBD}	Global Block Dispersion weight	1
p_{GDesv}	Global Desviation weight	2
p_{GTOS}	Global TOS Fitness weight	1
p_{GD}	Global Dispersion Fitness weight	0 (no sparsity) 1 (sparsity)
p_u	Penalization for unassigned individual tickets weight	0.5

Table 3. Weights and constants used in our methodology.

GD: measuring the allocation sparsity

In order to compute the sparsity of the allocated seats in the stadium, we propose the following expression:

$$GD = \frac{p_{GBD} \cdot GBD + p_{GDesv} \cdot GDesv}{p_{GBD} + p_{GDesv}} \qquad 0 \le GD \le 100 \tag{2}$$

Where: GDB is the in-zone sparsity degree, p_{GBD} is the weight of GDB, GDesv indicates the overall zones sparsity degree, and p_{GDesv} is the weight of GDesv.

First, the zone sparsity GBD can be calculated as an average measure regarding the different sparsities of tickets in all the zones as follows:

$$GBD = \frac{\sum_{k=1}^{nb} BD_k \cdot nsb_k}{\sum_{k=1}^{nb} nsb_k} \qquad 0 \le GBD \le 100 \tag{3}$$

Where nsb_k is the number of available seats in the k zone, nb the number of zones in the stadium, and BD_k is the sparsity degree of the k zone.

BD can be computed based on the idea of dividing a zone into 9 parts[1], and combining the standard deviation of the percentage of seat assignments in each part. Formally:

$$BD_k = 100 - \frac{2}{\sqrt{2}} \sqrt{\frac{\sum_{t=1}^{9} \left((P_t - Mean(Pt))^2 \cdot nst_t \right)}{\sum_{t=1}^{9} nst_t}} \qquad 0 \le BD_k \le 100 \tag{4}$$

Where P_t is the occupation percentage of the t zone part, and nst_t is the total number of available seats in the t zone part.

And second, the global deviation GDesv measures the spreading out of the tickets assigned to each zone. For this purpose, we compute it based on the standard deviation of the percentages of seats assigned in each zone according to the following expression:

$$GDesv = 100 - \frac{2}{\sqrt{2}} \sqrt{\frac{\sum_{k=1}^{nb} \left((P_k - Mean(Pk))^2 \cdot nsb_k \right)}{\sum_{k=1}^{nb} nsb_k}} \qquad 0 \le GDesv \le 100 \tag{5}$$

Where p_k is the percentage of seats assigned in the k zone regarding the total amount of available seats in the zone, Mean(Pk) is the mean of the percentages, Nsb_k is the amount of seats available in the k zone, and Nb is the number of zones in the complete stadium.

[1] The division into 9 zones has been tested experimentally.

GTOS: measuring fitness

The GTOS component of the fitness function tries to capture the degree of fitness of the whole allocated groups, based on the fitness of the sub-groups, since groups are split in several sub-groups (see rule RO2). Then, the GTOS is defined as follows:

$$GTOS = \frac{\sum_{o=1}^{no} TOSF_o \cdot w_o}{\sum_{o=1}^{no} w_o} \qquad 0 \le GTOS \le 100 \tag{6}$$

Where, no is the total number of allocated subgroups, w_o is the weight of each subgroup, and TOSF is the individual fitness of the o subgroup.

In this particular case, the weight w_o is automatically computed as the difference between the highest rank of all the groups and the rank of a subgroup. That is:

$$w_o = Max(rank) - rank_o \tag{7}$$

Regarding the individual fitness of each subgroup, TOSF, it is based on the average of the fitness of the tickets regarding their rank ($TOSR_0$), their joint fitness ($TOSJ_o$) regarding the distribution of the seats, and the relevance of the different optimization rules unsatisfied. The following formula expresses such average:

$$TOSF_o = \frac{p_{TOSR} \cdot TOSR_o + p_{TOSJ} \cdot TOSJ_o}{(p_{TOSR} + p_{TOSJ}) \cdot \left(1 + \sum_{r=1}^{nr_o} q_r\right)} \qquad 0 \le TOSF_o \le 100 \tag{8}$$

Where, p_{TOSR} is the weight corresponding to the fitness of the tickets regarding their rank, p_{TOSJ} is the weight of the joint fitness TOSJ0, nr_o is the number of unsatisfied optimal rules, and q_r is the relevance of the unsatisfied rule r.

The $TOSR_o$ fitness is defined as an average of the individual fitness of each ticket belonging to the TS. It is defined as follows:

$$TOSR_o = \frac{\sum_{i=1}^{nto_o} ITF_i}{nto_o} \qquad 0 \le TOSR_o \le 100 \tag{9}$$

Where, nto_o is the total amount of tickets in the TS, and ITF_i the individual fitness of the i ticket.

Regarding the individual fitness of a ticket, ITF, it is computed according to the relation between the ticket rank and the seat rank, as shown following:

$$ITF_i = 100 - \left| \frac{TR_i * 100}{MaxGR} - \frac{SR_j * 100}{MaxGR} \right| \qquad 0 \le ITF_i \le 100 \tag{10}$$

Where TR_i is the rank of the i ticket, SR_j is the rank of the j seat, and MaxGR is the maximum between the highest seat rank and the highest ticket rank.

Finally, the joint fitness of a TS, $TOSJ_o$, is computed according to the rate of the distribution of the TS and the ideal distribution. Formally:

$$TOSJ_o = 100 \cdot \frac{N'_o \cdot Y + M'_o \cdot X}{N_o \cdot Y + M_o \cdot X} \qquad 0 \le TOSJ_o \le 100 \tag{11}$$

Where N_o and M_0 are the amount of rows and columns in which the TS has been allocated, N'_o and M'_0 are the ideal number of rows and columns correspondingly, and X and Y are constants which express the ratio of the desired rectangle size that forms the space of the allocated seats. See table 3 for the X and Y values, and (Authors, 2005) for details about the estimation of N'_o and M'_0 .

4 Methodology

The goal of our methodology is to provide an allocation of physical seats to all the tickets of the TGs according to the fitness function defined in the previous section. Since we are dealing with a large scale problem, our goal has been to develop a method that assures to find a solution as soon as possible, and then to improve the solution as time passes, so we have developed an anytime method.

Then, regarding our problem, the first of the mandatory rules tells us that seat and ticket categories must agree. This constraint helps us in dividing the problem in as much sub-problems as categories we have. Then, several allocation processes, one per category, can concurrently be run (see figure 4). Each process deals exclusively with the data corresponding to its category, and so, with a lower complexity that the global problem. The final solution is the joint of the results obtained in each category.

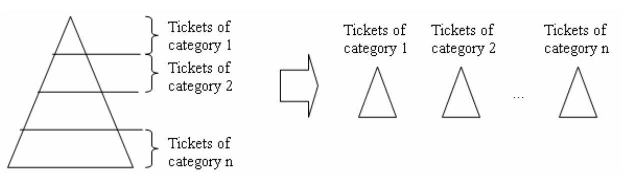


Figure 4. Division of the search space in n parallel processes according to ticket categories.

The allocation of TG to seats in a given category is based on a search algorithm that tries to obtain a first candidate solution as soon as possible. The inputs of the algorithm are the set of TS, ts_1, …, ts_m, sorted according to their priority, that is, the category and rank. Since the category is the same for all the members of the same assignment process (see figure 4), the key attribute is then the rank. In each level of the search tree, a TS is being assigned to seats corresponding to one zone or more zones (depending on the optional rules) according to an adapted version of the region growing method (Zucker, 1976). If some TSs cannot be assigned anywhere, they are temporarily forgotten, and the algorithm continues with the rest of the TS. The forgotten group is treated at the end, when some optional rules can be relaxed in order to allocate them. Once a first solution to the problem is achieved, a local search method is used in order to improve it.

Region growing for seat assignment

In order to allocate seats to a TS, a method has been defined based on a common technique used in Computer Vision for image region segmentation: the *region growing* algorithm (Zucker, 1976). This algorithm roughly consists on sowing a seed in an image, so as the seed grows, it occupies all the pixels of a given region. We though that such method can be applied to our allocation problem, if the TS can be mapped as the regions, and the seat zones as the image. From this point of view, for each TS it is necessary to select a seed that is a seat in a given zone, and then, to grow up the seed until all tickets of the TS take up the seats.

Consistently, the method that we propose is based on three steps:

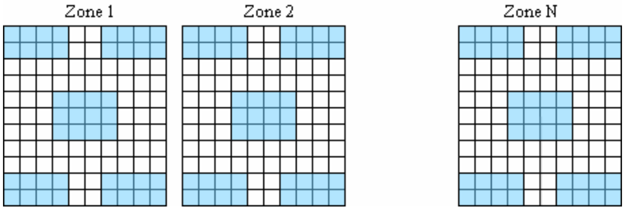
1. Select the zone with the ranking most according to the TS rank that has enough seats to allocate it
2. Select a seed.
2. Grow up the seed until all tickets of the TS has been allocated.

These steps are iterated until the TS has been entirely assigned. In the remaining of this section, all the steps are detailed.

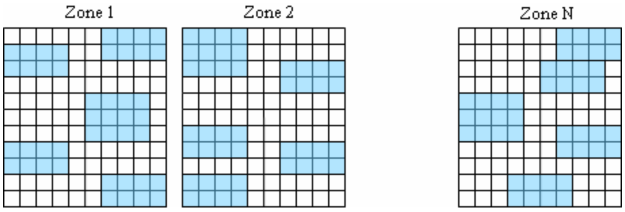
Seed selection

Seed selection depends on the dispersion value of the TG and the sparsity rule (RO9). The easiest case is when the dispersion attribute is off and the sparsity rule is not activated. Then the process consists of selecting as seed the empty seat of the zone that has the highest rank.

When the sparsity rule is on, the TS assigned to a zone should be distributed widespread in the zone that is, trying to not collide with any other TS (see figure 3c). One way to achieve such distribution is to compute a distance from the assigned TSs in the zone. However, this strategy has a deterministic behaviour that makes all the zones have similar distributions (see figure 5a). Conversely, a random seed selection provides a uniform and widely better distribution in each zone (see figure 5b). So, we use a random method.



(a)



(b)

Figure 5. Distance-based (a) and random (b) sparsity

Seed growing up

Once a seed has been selected, that is, a ticket of the TS has been assigned to a seat of a zone, the remaining seats of the TS should be allocated around it. This process is iterative: in each iteration one ticket is assigned to a seat. The seat is selected according to a neighbourhood policy. At the beginning, the seat is selected among the neighbours of the seed; in the second interaction, the seat is selected among all the neighbours of the previous allocated seats (the seed, and the second seat), and so on until all tickets have been assigned. So, at each iteration, the seed growing up algorithm keeps a list of seat candidates (neighbours) among which the best seat is selected for a ticket (see figure 6). The selection method is based on the fact that all the seats of the group should be together (grouping factor) and the seat category as the distribution rules point out.

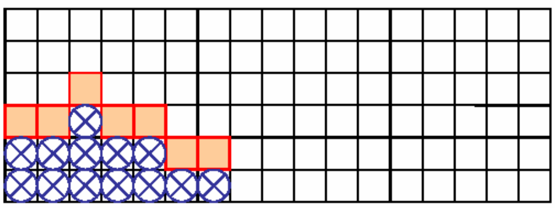


Figure 6. Example of seed growing up. Cross circles are allocated seats, while grey cells are candidates (neighbours) for new tickets.

An important problem arises when the selected seed cannot grow enough to appropriately allocate all the tickets of the TS due to, for example, some of the rules are not satisfied (see figure 7). That means that another seed should be selected for the group. This seed, however, can be valid for another TS. Since the growing process is costly, one interesting thing to know is whether the seed is a bad choice for all the TS or not. Then, the seed is checked for all the TS of the same category. If it does not work for any of them, then the seed is labelled as no-good and no other TS will test it again.

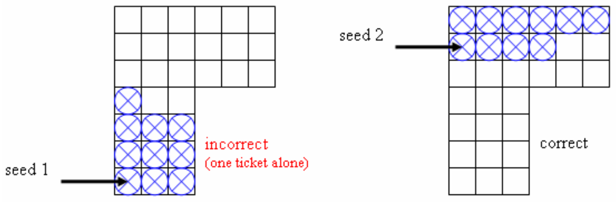


Figure 7.Different results changing the seed of a TS.

Unassigned group treatment

Leaving TS unassigned is a good option trying to find a first approximation to the solution to the allocation problem. However, if there are some unassigned groups, the fitness of the solution hardly descends, since it penalizes a lot unassigned groups. Then, an additional treatment is required trying to assign as much as TS as possible.

If a TS has been skipped, that means that there is no zone with enough seats to allocate it. Then, the only way to have room enough in a zone is by undoing the allocation of some of the TS and checking a new combination that diminishes the resulting number of unassigned groups.

The strategy we propose is to undo assignments of TS close to free seats (undo-TS). Then, the resulting free zone is bigger and eventually, unassigned zones can be placed there. The undoned TS remaining can be tested in other zones. This strategy is based on the fact that the allocation process is based on the TS priority, instead of their size.

Then, in each iteration of this step, we expect to decrement the number of unassigned TS, while completing more seat zones.

Local search

The method described above regarding region growing provides a first candidate solution to the problem, one for each category. Then, for each candidate, a local search algorithm is started in order to iteratively move to a better neighbour solution. This local search is based on changing the assignments of the different TS allocated to a zone, in order to improve the fitness of the overall allocation. Among the different trials in a zone, the best allocation is finally selected.

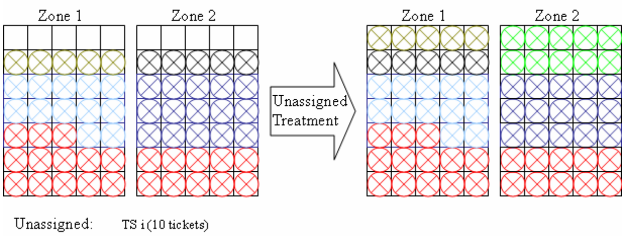


Figure 8. Example of finding space for unassigned TS.

Note, that the improvement process can be applied in parallel for any zone of each category.

Complexity analysis

Given *n* tickets and *p* seats, the following expression measures the cost of searching the space starting from the first ticket to the last one, and testing all available seats in each level (*q* at the first level, *q-1* at the second level, … until *q-n* at the n level):

$$C = \frac{p!}{(p-n-1)!} \quad (12)$$

Regarding to our methodology, we think that we dramatically reduce the cost, as analysed in this section.

The cost of our methodology is the sum of the cost of the different steps: C'=C1+C2+C3, where C1 is the cost of finding the first candidate solution, C2 is the cost of treating unassigned groups, and C3 is the cost of the local search.

First of all, the search space is split in sub-spaces, one per each category. Given *k* categories, we have *p' = p/k* seats in this new sub-space.

Second, the assignment is performed by groups instead of tickets. Given *m* groups, the number of groups per category is *m'=m/k*, being *m' << n.*

Assume that the average number of zones in each category is *z*, and that the cost of finding a seed in a zone is *r* (*r* can be seen as the seat average of each zone). Then, the cost of finding a seed in a category search space is *r*z*.

Since there are *m'* groups per each category, the cost of finding a first candidate solution to the problem is

$$C1 = m' * r * z \quad (13)$$

Regarding C2, assume that *u* is the amount of unassigned groups, *z'* the amount of zones with unassigned seats (**z'** < *z*), μ the average of undo-groups (undone allocations), and *k* the average of trials. Then, the cost of treating unassigned groups is:

$$C2 = u * z' * \mu * k \quad (14)$$

Note that in the worst case μ is *r*, and *k=u*z'*, resulting a cost:

$$C2 = u^2 * z'^2 * \mu \quad (15)$$

Finally, C3 depends on the number of trials we whish to perform, *t*. Given *m''* groups assigned to each zone (so *m''<<m'*), C3 can be estimated as:

$$C3 = z * m'' * t \quad (16)$$

In our experiments we have set *t=10*, resulting in the following cost:

$$C3 = 10 z m'' \quad (17)$$

Looking at the different components of C', we can see how the complexity of the problem is reduced.

5 Experimental Results

In order to experimentally prove our methodology, we have chosen the following configuration:

- Stadiums which seat ranges oscillates from 5,000 to 50,000 seats
- Parameters of the distribution rules: 40 tickets maximum in a subgroup, 10 tickets of the same TS maximum in a same seat row, and the dispersion flag is not activated.
- 5 categories
- 5,000 ticket groups, ranging from 1 to 40 tickets per group.

All the experiments have been carried out in a Pentium IV 3GHz, 1 GB of RAM

In Figure 9 there is an example of two zones with the seats assigned according to the first candidate solution step and in figure 10 the same zone after improving the results with the treatment of unassigned groups method. In the first solution, 4960 tickets have been assigned, while 40 remains unassigned. The fitness value of the first solution is 12.28. In the improved solution, all the tickets have been assigned, and the fitness achieved is 87.08.

In general, figure 11 shows the fitness behaviour related to the number of tickets to be distributed (from 5000 to

50000) for both, the first candidate solution and the improved solution. It is possible to see how the improvement step influences the results. However, such results are not achieved for grant: they consume much more time than the first solution.

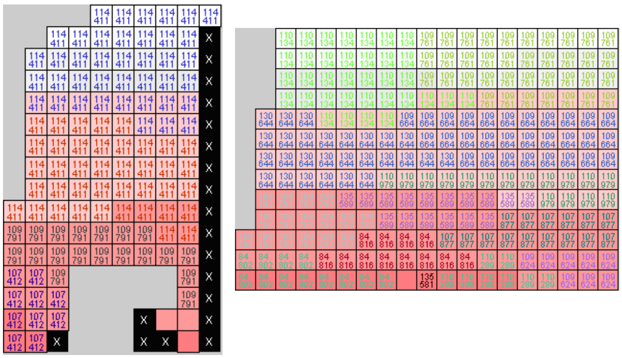


Figure 9. Results obtained as a first candidate solution.

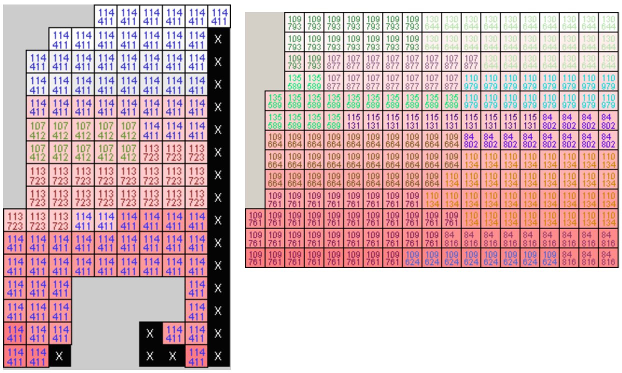


Figure 10. Results obtaining after improving the allocation thanks to the treatment of unassigned tickets step.

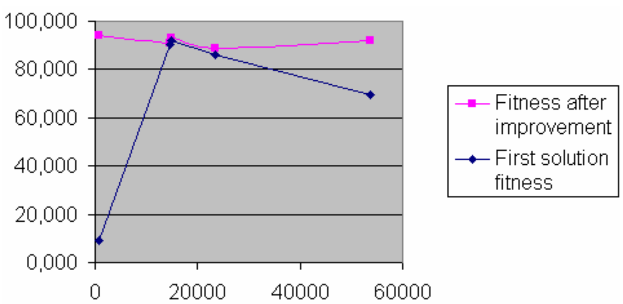


Figure 11. Fitness behaviour. X axis are the number of tickets to assign and the Y axis is the fitness value.

Regarding execution time, there is a significant difference between the time required for the first candidate solution and the one for the improvements. The former are much lower: for 50,000 tickets only 6 minutes (aprox.) have been required (see figure 12). Regarding the local search, times are expensive, it takes some hours: 3:38:20 for 14864 tickets, 6:51:44 for 23335 tickets and 11:29:09 for 53673 tickets.

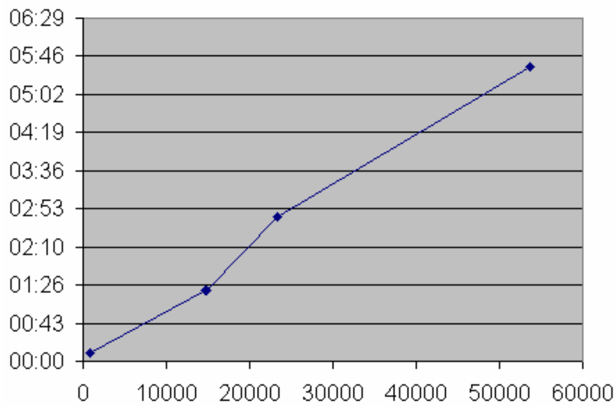


Figure 12. Time required for finding a first candidate solution. X axis are the number of tickets to assign and the Y axis are the time measured in minutes:seconds.

The time required for the local search is high, but it has the advantage that can be stopped at any moment, providing the best solution found up so far. So in this sense, the algorithm exhibits an anytime behaviour.

6. Related work

There are some related works regarding seat allocation, mainly in the airline domain. For example, (Freisleben and Gleichmann, 1993) propose the use of a neural network mechanism to control airline seat allocations in order to predict overbooking situations. Overbooking problems are related to capacity allocation, which goal is to optimize the sells, while in our problem, seat allocation, the tickets have been already sold and the key issue is to distribute them in a given scenario.

Another interesting work related to pricing and fare optimization in the airline industry is (Coté et al., 2003), which provides a joint solution of the capacity allocation and pricing problems of such kind of companies. They use a special case of hierarchical mathematical optimization modelling technique (what they called *bileval mathematical programming paradigm*) in order to maximize companies revenues. (Bertsimas and Popescu, 2003) deals also with the problem of intelligent allocate the limited inventory of transport companies (airlines linked to hotel and car rentals) to demand from different market segments, so that their revenues are maximized. Their approach is also related to the process of selling goods, dealing with cancellations and other unexpected events, and avoiding the payment of overbooking penalties. The authors design a decision support tool, based on stochastic and dynamic optimisation technique that at each point in time accepts or rejects a reservation request, in order to deal with different fares so that the maximum revenue is obtained.

Again, a revenue approach is related to fares and class ticket limits. In our problem, the number of tickets of a class is already predefined, so the approach to be followed is different. That is, we believe that one important different from revenues approach to ours is the kind of the data to be treated. Revenues problems as fare optimization and pricing are subjected to some kind of data as tracking information of competitor's fares, demand forecast, historical sales patterns, etc. that require from particular data bases in order to be treated properly. Such amount of data is necessary to solve the problem. In our seat allocation problem, we have less amount of information, mainly, the distribution rules. However, what it is important in our problem is the amount of seats to be allocated. While in a flight, the capacity is 500 maximum, in our seat allocation problem we are dealing with huge scenarios regarding thousands of seats (up to 100.000). So the kind of techniques required to solve the problems should be different.

Regarding the region growing technique, it has been applied to site allocation problems, as for example in (Brookes, 97). Site allocation is related to the fact of finding some kind of regions in Geographical Information Systems (GIS), as for example, the spatial arrangement of wildlife reserves. The detection of the regions depends not only on land cover and other local attributes of images but on the size and shape of patches and their spatial relations to each other. Site allocation is also related to resource allocation in which the optimal allocation of multiple sites of different land uses to an area is trying to be optimized. For example, recent works of Aerts and his colleagues (Aerts, 2003; Aerts et al., 2002; Aerts et al. 2002 bis) are investigating integer programming techniques for integrating spatial decisions and resource allocation. We think that our work is in line with this kind of research.

7 Conclusions

In this paper we have presented a methodology to deal with the seat allocation problem for massive events. In such kind of problems groups of tickets with very different attributes (categories, ranks, status) should be assigned to stadium seat zones, characterized also by categories, ranks, size, etc. In addition, the organization committee imposes some distribution rules that should be satisfied in some cases and optimized in other ones.

The methodology we propose is based on a region growing technique which provides a first candidate solution for the allocation process. Then, in a subsequently step, a local search method is applied in order to optimize the solution. The experimental results obtained have shown that our methodology works well, so in the case of dealing with a huge amount of tickets (about 50.000), we obtain a realistic response time.

As a future work, we are thinking to add to our system new functionalities, as for instance, to extend the allocation process to tickets that have been sold for more than one competition. This is a common situation that happens for example in the Grand Prix racings: the fans of either a given constructor (Ferrari, McLaren, Williams, etc.) or driver (Fernando Alonso, Christijan Albers, etc.) wish to attend to all the events of the corresponding team. So, in addition of groups of tickets, bundles of tickets regarding different competitions should be also considered in the allocation process.

Acknowledgements

This work has been supported by the Spanish MEC Project TIN2004-06354-C02-02.

References

Aerts, J.C.J.H., Eisinger, E., Heuvelink, G.B.M., Stewart, T.J.: "Using Linear Integer Programming for Multi-Site Land-Use Allocation". Geographical Analysis, Vol. 35, No. 2 (April 2003)

Aerts, J.C.J.H., Heuvelink, G.B.M.: "Using simulated annealing for resource allocation". Int. Jorunal of Geographical Information Science, 2002, vol. 16, no. 6, 571-587

Aerts, Jeroen C. J. H.: "Using Linear Integer Programming for Multi-Site Land-Use Allocation". Geographical Analysis - Volume 35, Number 2, April 2003, pp. 148-169

Bertsimas D., Popescu, I.: "Revenue Management in a Dynamic Network Environmen". Transportation Science, 37(3):257-277, 2003.

Brookes, C.J.: "A parameterized region-growing programme for site allocation on raster suitability maps". Int. Journal of Geographical Information Science, 1997, vol. 11, no. 4, 375- 396

Côté, J-P., Marcotte P., Savard G.: "A bilevel modeling approach to pricing and fare optimization in the airline industry". Journal of Revenue and Pricing Management 2 (2003) 23-36

Freisleben, B.; Gleichmann, G.: "Controlling airline seat allocations with neural networks". Proceeding of the Twenty-Sixth Hawaii International Conference on System Sciences, 1993, Volume iv, 5-8 Jan. 1993 Page(s):635 - 642 vol.4

Authors 2005. Not cited due to blind revision.

Zucker, S.W: Region Growing: Childhood And Adolescence. Journal CGIP, p 382-399, 1976.