

A Software Application to Optimize the Visits of Sales/Marketing Agents to their Customers in a Brewing Company

Marcos Colebrook, Ana González-Larsson and Antonio Sedeño-Noda

Abstract We present a software application developed to optimize the annual planning of visits of a brewing company's sales/marketing staff to their customers. Each of these customers must be annually visited a provided number of times. Thus, each salesperson is assigned to a set of customers that must be visited each week. The application will assign all the visits of a salesperson to each customer so that all weeks should have more or less the same number of visits. By virtue of this approach, the brewing company diminished their marketing operating costs, as well as improved their customer relationships.

1 Introduction

The biggest brewing company (the name is undisclosed due to confidentiality reasons) in the Canary Islands (Spain) is among the leading local business, generating wealth and development for the regional economy, with nearly 11,700 customers, as well as fostering nearly 900 direct jobs and 13,000 indirect jobs. The estimated revenue (in 2009) was €137 million (= US \$188 million).

For such a big company in the brewing business, the relationship with its customer is vital. Salespersons are required to visit a set of assigned customers each week for marketing purposes. This information can be obtained from a database table which is consulted through a PDA (Personal Digital Assistant). Every time a salesperson visits a customer, the date and a description of the visit is written down in the PDA and transferred to the database. Each of the 53 salespersons has to accomplish approximately 2,000 visits a year.

In this sense, the company wanted to improve the visit planning application because it was not very efficient, and it did not consider the history of past visits. The problem also was justified by:

- The need to optimize company's resources.
- To balance the numbers of visits of each salesperson.
- The requirement that each customer should be visited regularly within a fixed schedule.

M. Colebrook (✉)
Departamento de Ingeniería Informática
Universidad de La Laguna, Tenerife, Spain
e-mail: mcolesan@ull.edu.es

A. González-Larsson
NOVASOFT TIC, Tenerife, Spain
e-mail: anaglezlarsson@gmail.com

A. Sedeño-Noda
Departamento. Matemáticas, Estadística e Investigación Operativa
Universidad de La Laguna, Tenerife, Spain
e-mail: asedeno@ull.edu.es

Unfortunately, there are few references in the literature regarding this problem. Ferrer et al. (2009) presented a procedure to design the routes for the salespersons of a multinational entertainment company. The company's selling network had around 450 customers covered by 8 salespersons, whereas the rental network held 400 customers managed by 6 salespersons. Each salesperson was assigned most of the customers in the zone, and had to visit their customers an established number of times (1 to 5) within a month. The objective function was to minimize the weighted sum of the travel time plus overtime penalties.

In another related paper, Polacek et al. (2007) dealt with determining daily routes for a traveling salesperson of a large and global food wholesaler, who had to visit up to 335 customers. Besides, each customer had to be visited at least once a year, with some customers requiring one visit per month. The primary objective was to minimize the total travel time of the salesperson. Another goal addressed was to minimize the number of days needed by the salesperson to visit all customers in a given month. They also enhanced the objective function by considering periodicity requirements for customer visits.

These papers are more concerned with routing than with planning. Besides, both the objective and the dimensions of our problem are quite different, mainly due to the requirements of the company.

The remainder of the chapter is organized as follows. In Sect. 2, the problem is described in detail. Section 3 presents the initial goals and the final scope of the requirements that was developed as a software application. The software application that the company had before tackling this project along with other software solutions are described in Sect. 4. In Sect. 5, we present and discuss the mathematical models for both the Static and the Dynamic Planning, whereas the solution algorithms are developed in Section 6. Finally, the computing results are provided in Sect. 7 before the conclusions of the chapter in Sect. 8.

2 Problem description

The company has 53 salespersons (SP) that must accomplish a certain number of visits per year to their 11,700 customers. Each salesperson belongs to a fixed delegation and it is assigned to a set of customers. In turn, each customer belongs to a presales zone (PSZ) and has associated a category to indicate the number of visits (24, 18, 12, 4, 3 and 2) he/she should receive during one year time.

The main question for each salesperson is: *Which is the best planning for visiting my set of customers?*

Clearly, it would not be very logical that all the visits should be performed in the first month. Therefore, the optimal planning should try to space the visits in time as best as possible.

On the other hand, it is not suitable for a salesperson to have a set of weeks with lots of visits and another set with few or none, since this would mean that there is

no regularity in their work. This can be achieved by evenly spreading all the visits of a salesperson to his/her customers, and taking into account holidays on each week.

Each customer belongs to a presale zone, so customers belonging to the same zone are geographically near. The company was also interested in bounding the number of presale zone that a salesperson must visit to a certain number (say for example 5), so each salesperson would not go on long trips every day, which in turn can save fuel (money) and time.

The optimal solution (planning) should consider the equitable distribution of visits all over the year (divided in working weeks), and that each salesperson should not pass through more than 5 PSZ each week.

Finally, the visits already performed in the past should also be considered, in order to plan the remaining visits in the future (dynamic planning). For instance, inserting promotional campaigns, new customers, etc.

3 Initial goals and final scope

Given the above description of the problem, at a first stage, we planned the following goals according to the company's requirements:

Static Planning

For each salesperson, perform a Static Planning (12 months ahead, from January 1st to December 31st) of all visits to its customers, given the following hints:

- Visits should be adequately spaced on time over the final planning.
- For each salesperson, the visits should be equitably distributed over all the year taking into account the weeks with holidays.

Dynamic Planning

For each salesperson, accomplish a Dynamic Planning, with regards to the following:

- Take into account visits already made in the past months to carry out a new optimal planning: this requirement was the most innovative for the company.
- Make a balanced distribution of visits.
- Visits should be also adequately spaced on time (regular frequency of visits).

Promotional Campaigns

Given a planning, insert Promotional Campaigns, which are considered as special visits made to a selected set of customers in a determined range of dates.

Besides, the final scope involved all these initial goals plus the following requirements:

- Each salesperson should visit at most 5 presale zones each week.
- Chance to choose which salesperson planning could be re-scheduled at any time.

- Friendly user interface to build the planning.
- User interface to insert Promotional Campaigns in the database.

4 Software applications

By the time the project was discussed, there were several applications that could be used for planning in large companies. Many of them had different types of interesting features, such as salespersons route computation, work planning, vehicle localization in real time, etc.

OPTI-TIME

Opti-Time SA is a company offering independent and complementary solutions to plan and optimize sales visits. One of the products they offer is *TourSolver* (OPTI-TIME 2014). It can be used to optimally arrange the order of visits to the customers to increase efficiency. It also improves the visibility of the sales teams' visits and facilitate their orientation in the route.

LOGISPLAN

Logisplan Professional Appointments (LOGISPLAN 2014) is a leading optimization software oriented to transportation and logistics that calculates routes and optimal loads for each vehicle in the fleet, providing the roadmap for each of them.

Previous company's application

To perform each salesperson's static planning, the company had a previous software application, which was really a SQL Server Stored Procedure inside the data base. This method ran over all the customers' information to insert in the database the corresponding visit trying to distribute them in time as far as possible in order to be non-consecutive.

It was actually executed once a year, and calculated the static planning of the entire year (i.e., from January 1st to December 31st). This caused several problems:

- This planning did not take into account visits made in the past.
- The visits were not reasonably distributed: some weeks were overloaded, and some other very underused.
- It did not insert all the visits planned for a certain year, and the process was not fully automated.
- They could not insert Promotional Campaigns.
- It did not take into account the 5 Pre-Sale Zone per week limit.

In conclusion, every software application we have briefly described can be used for a specific planning task, with regard mainly to vehicle routing and not so explicitly to visit planning. Therefore, none of them fit exactly to the company's needs. This was mainly the reason of the new application.

The model presented in the following section along with the subsequent algorithm development completely solved the company's problem. Besides, being developed as a custom tailored application, the company avoided the cost of licensing other external products.

5 Mathematical model

The next table describes both the parameters and variables of the problem.

Table 12.1. Parameters of the problem. The term VpW stands for "Visits per Week".

Parameters	Description
NC	Number of customers, $c = 1, \dots, NC$.
NW	Number of weeks, $w = 1, \dots, NW$. Depending on the year, $NW = 52$ or 53 .
$V[c]$	Number of visits to a certain customer c in a certain period of time (usually a year).
$MinVpW[w]$	Minimal number of visits that a SP could make in a week w .
$MaxVpW[w]$	Maximal number of visits that a SP could make in a week w .
$F[c]$	Frequency of visits (in weeks) to a customer c .

Table 12.2. Decision variables of the model. The term VpW stands for "Visits per Week".

Variables	Description
$X[c,w]$	Binary variable equal to 1 if customer c is visited in week w , and 0 otherwise.
$VpW[w]$	Number of visits for a certain SP in week w .
$NVpW$	Maximal number of visits per week.

As an initial approach, and for each SP, the problem was modeled using a bipartite graph (see Fig. 12.1), where the set of NC customers are represented as the left nodes, and the set of NW weeks are denoted as the right nodes. Besides, nodes A and B represent the source (in this case, the salesperson) and the sink (dummy node), respectively.

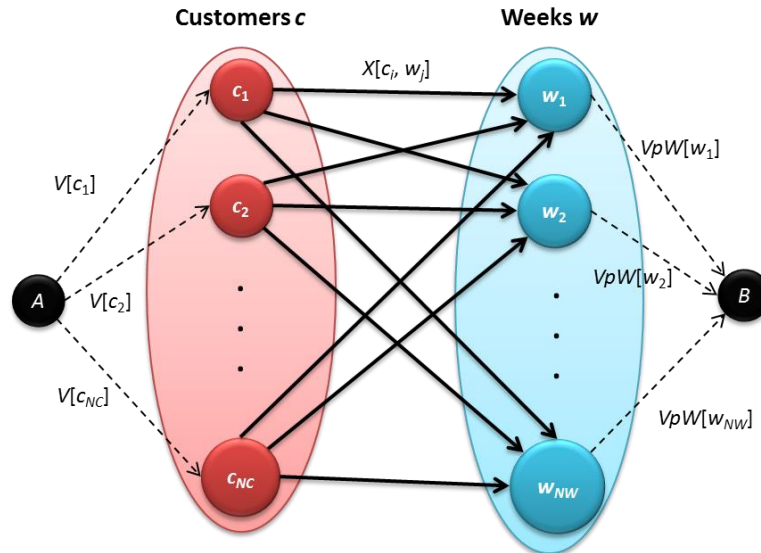


Fig. 12.1. Network model of the problem as a bipartite graph.

Then, given the input parameters of Table 12.1, the problem can be expressed as finding the right assignment between the customers and the weeks, so that all the requirements are fulfilled.

After this initial graph, and depending on the type of planning (static or dynamic), we devised the following mathematical models.

5.1 Static Planning

The model for the Static Planning is as follows:

minimize $NVpW$

$$\begin{aligned}
 (1) \quad & \sum_{c=1}^{NC} X[c, w] = VpW[w], & w = 1, \dots, NW \\
 (2) \quad & \sum_{w=1}^{NW} X[c, w] = V[c], & c = 1, \dots, NC \\
 (3) \quad & MinVpW[w] \leq VpW[w] \leq \min(NVpW, MaxVpW[w]) & w = 1, \dots, NW \\
 (4) \quad & \left\{ \begin{array}{l} X[c, 1] + X[c, 2] + \dots + X[c, F[c]] \leq 1 \\ X[c, F[c] + 1] + X[c, F[c] + 2] + \dots + X[c, 2 \cdot F[c]] \leq 1 \\ X[c, 2 \cdot F[c] + 1] + X[c, 2 \cdot F[c] + 2] + \dots + X[c, 3 \cdot F[c]] \leq 1 \\ \vdots \\ X[c, NW - F[c] + 1] + X[c, NW - F[c] + 2] + \dots + X[c, NW] \leq 1 \end{array} \right. & c = 1, \dots, NC \\
 (5) \quad & 0 \leq X[c, w] \leq 1 & \left\{ \begin{array}{l} c = 1, \dots, NC \\ w = 1, \dots, NW \end{array} \right. \\
 (6) \quad & VpW[w] \geq 0 & w = 1, \dots, NW \\
 (7) \quad & NVpW \in \mathbf{Z}^+
 \end{aligned}$$

Objective function

We wish to minimize the maximal value of visits in a week $NVpW$. This allows to distribute equitably the weekly task of each SP.

$$\text{minimize } NVpW$$

Constraint (1)

For each week $w = 1, \dots, NW$, the total number of visits in such week must be equal to the sum of visits to all customers.

$$\sum_{c=1}^{NC} X[c, w] = VpW[w] \quad (1)$$

Constraint (2)

For each customer $c = 1, \dots, NC$, the total number of visits to that customer must be equal to the total sum of visits in all weeks to that customer.

$$\sum_{w=1}^{NW} X[c, w] = V[c] \quad (2)$$

Before describing constraint (3), we must explain how the values of $MinVpW[\cdot]$ and $MaxVpW[\cdot]$ are computed. For each week $w=1, \dots, NW$, we define a $WeekWeight[w] = 1$. Then, for each non-working day (i.e. holidays), we subtract 0.2 to the $WeekWeight[w]$. For example, in a 5-day working week w with two holidays, we get $WeekWeight[w] = 1 - 0.2 - 0.2 = 0.6$. Thus, the values of $MinVpW[w]$ and $MaxVpW[w]$ are the following:

$$z[w] = \frac{\sum_{c=1}^{NC} V[c]}{\sum_{w=1}^{NW} WeekWeight[w]} \cdot WeekWeight[w]$$

$$MinVpW[w] = \lfloor z[w] \rfloor, \quad MaxVpW[w] = \lceil z[w] \rceil$$

This parameters are fine-tuned later in the algorithm to get a feasible solution.

Constraint (3)

For each week $w = 1, \dots, NW$, the total number of visits cannot exceed the maximal number of visits $\min(NVpW, MaxVpW[w])$ that a SP can make in a certain week, and also cannot be less than the minimal number of visits $MinVpW[w]$.

$$MinVpW[w] \leq VpW[w] \leq \min(NVpW, MaxVpW[w]) \quad (3)$$

In the right hand side expression, $NVpW$ is usually the smaller value, except for the weeks with $WeekWeight[\cdot] < 1$.

Constraints (4)

For each customer $c = 1, \dots, NC$, the frequency of visits $F[c]$ depends on the number of visits $V[c]$ per time unit (week):

$$F[c] = \left\lfloor \frac{NW}{V[c]} \right\rfloor$$

The frequency of visits should be adequately spaced and it should not be accumulated in adjacent weeks:

$$\begin{aligned} X[c,1] + X[c,2] + \dots + X[c,F[c]] &\leq 1 \\ X[c,F[c]+1] + X[c,F[c]+2] + \dots + X[c,2 \cdot F[c]] &\leq 1 \\ X[c,2 \cdot F[c]+1] + X[c,2 \cdot F[c]+2] + \dots + X[c,3 \cdot F[c]] &\leq 1 \\ &\vdots \\ X[c,NW - F[c]+1] + X[c,NW - F[c]+2] + \dots + X[c,NW] &\leq 1 \end{aligned} \quad (4)$$

In most cases the frequency $F[c]$ is not integer, so we have to add the remainder weeks to the last group. For example, given a customer c who is going to be visited 3 times in a year with 53 weeks, the ideal frequency of visits is:

$$F[c] = \left\lfloor \frac{53}{3} \right\rfloor = 17 \text{ weeks}$$

The three groups of weeks where the 3 visits should be accomplished are the following:

- 1 visit in the first group of weeks: 1, 2, 3, ..., 15, 16, 17.
- 1 visit in the second group of weeks 18, 19, 20, ..., 32, 33, 34.
- 1 visit in the third group of weeks: 35, 36, 37, ..., 50, 51, **[52, 53]**.

Since there are two weeks remaining (52 and 53), we assign them to the last group. This might imply that the weeks on this last group will not be as balanced as the rest of weeks.

Besides, once the model is solved, the result might yield that some visits must be performed in consecutive weeks. For example, in week $F[c]$ (first group of constraints) and the next one in week $F[c] + 1$ (second group). This is solved using a method discussed in Sect. 6.2.1.

To avoid an excessive load of constraints to the mathematical solver, we decided to include only these simple constraints, instead of the complete set of frequency constraints:

$$\sum_{j=k}^{k+F[c]} X[c, j+1] \leq 1, \quad k = 0, \dots, NW - F[c]$$

This lack of constraints truly speeds up the first basic and feasible solution of the problem.

Constraints (5)-(7)

Finally, for each customer $c = 1, \dots, NC$ and for each week $w = 1, \dots, NW$, the decision variables range over the following values:

$$0 \leq X[c, w] \leq 1 \tag{5}$$

$$\forall p W[w] \geq 0 \tag{6}$$

$$NVpW \in \mathbf{Z}^+ \tag{7}$$

To make the problem more tractable, we relaxed variables $X[c, w]$ to be real numbers between 0 and 1. In case any variable $X[c, w]$ gets a real value, we just round up to 1 the first $X[c, w] > 0$ for the weeks of the same group of constraints (4). This

allows obtaining an initial basic solution much faster, which is later enhanced with specific algorithms (this will be discussed in Sect. 6).

It is worth mentioning that constraint (6) is superfluous due to (1) and (5), and hence, it can be removed from the model.

Finally, the model without the constraints in (4) corresponds to a parametric maximum flow problem in a bipartite graph, and it can be solved in the same time complexity than the maximum flow problem (see Ahuja et al. 1993).

5.2 *Dynamic Planning*

Before presenting the model for the Dynamic Planning, we must introduce some additional parameters which are described in Table 12.3.

Table 12.3. Additional parameters for the Dynamic Planning model.

Parameters	Description
PW	Number of weeks already planned (past weeks).
$V[c]$	Number of visits to a certain customer c that remain to be planned in the future, considering the visits already performed in the past.
$R[c,w]$	Records the past visits. Variable equal to 1 if customer c was visited in week w , and 0 otherwise.

Taking into account the previous additional parameters, we present the mathematical model for the Dynamic Planning, which resembles the Static Planning model, but it has some slight modifications:

minimize $NVpW$

$$(8) \quad \sum_{c=1}^{NC} X[c, w] = VpW[w], \quad w = PW + 1, \dots, NW$$

$$(9) \quad \sum_{w=PW+1}^{NW} X[c, w] = V[c], \quad c = 1, \dots, NC$$

$$(10) \quad MinVpW[w] \leq VpW[w] \leq \min(NVpW, MaxVpW[w]) \quad w = PW + 1, \dots, NW$$

$$(11) \quad X[c, w] = R[c, w] \quad \begin{cases} c = 1, \dots, NC \\ w = 1, \dots, PW \end{cases}$$

$$(12) \quad \left\{ \begin{array}{l} X[c, PW + 1] + X[c, PW + 2] + \dots + X[c, F[c]] = 1 \\ X[c, F[c] + 1] + X[c, F[c] + 2] + \dots + X[c, 2 \cdot F[c]] = 1 \\ X[c, 2 \cdot F[c] + 1] + X[c, 2 \cdot F[c] + 2] + \dots + X[c, 3 \cdot F[c]] = 1 \\ \vdots \\ X[c, NW - F[c] + 1] + X[c, NW - F[c] + 2] + \dots + X[c, NW] = 1 \end{array} \right. \quad c = 1, \dots, NC$$

$$(13) \quad 0 \leq X[c, w] \leq 1 \quad \begin{cases} c = 1, \dots, NC \\ w = PW + 1, \dots, NW \end{cases}$$

$$(14) \quad VpW[w] \geq 0 \quad w = PW + 1, \dots, NW$$

$$(15) \quad NVpW \in \mathbf{Z}^+$$

Since the main goal of the model has not changed, the objective function is the same as the one presented in the Static Planning.

Constraint (8) and (9)

These constraints are the same as constraints (1) and (2), but the week index w starts from $PW + 1$ to avoid the past weeks.

Constraint (10)

This constraint is very similar to constraint (3) of the Static Planning model. However, the weeks in constraint (10) range in $w = PW + 1, \dots, NW$ to avoid past weeks, and the values of $MinVpW[w]$ and $MaxVpW[w]$ are recomputed accordingly with the updated number of visits of each customer in the remaining weeks.

$$MinVpW[w] \leq VpW[w] \leq \min(NVpW, MaxVpW[w]) \quad (10)$$

Constraints (11)

This is a new constraint that is used only in the Dynamic Planning model. For each customer $c = 1, \dots, NC$, and for each week $w = 1, \dots, PW$, parameter $R[c, w]$ records the visits already performed in the past weeks w to customer c .

$$X[c, w] = R[c, w] \quad (11)$$

Constraint (12)

This constraint resemble constraint (4) of the Static Planning model, though the weeks start at $PW + 1$.

$$\begin{aligned}
 X[c, PW + 1] + X[c, PW + 2] + \dots + X[c, F[c]] &= 1 \\
 X[c, F[c] + 1] + X[c, F[c] + 2] + \dots + X[c, 2 \cdot F[c]] &= 1 \\
 X[c, 2 \cdot F[c] + 1] + X[c, 2 \cdot F[c] + 2] + \dots + X[c, 3 \cdot F[c]] &= 1 \\
 &\vdots \\
 X[c, NW - F[c] + 1] + X[c, NW - F[c] + 2] + \dots + X[c, NW] &= 1
 \end{aligned} \tag{12}$$

Constraints (13)-(15)

Same as constraints (5)-(7) but the weeks start in $PW + 1$.

6 Algorithm development

Once the problem has been modeled in both its static and dynamic versions, we proceed to describe the algorithms that have solved these models. First, we introduce the software tools used to devise the solution, and subsequently, we describe the two algorithms developed.

6.1 Software tools

For the development of this solution, we used the following tools:

Integrated Development Environment (IDE) and programming language

We used Microsoft Visual Studio 2008 (version 9.0.30729.1 SP) as the IDE, specifically, Microsoft Visual C# (MVS 2014). However, any other IDE supporting C# as the programming language (as open sourced SharpDevelop) should work fine.

Database management

To get the data from the company's database, we used Microsoft SQL Server 2008.

Solver

Microsoft Solver Foundation (MSF) is a set of tools for mathematical modeling and optimization problems using OML (Optimization Modeling Language), and that integrates directly with Visual C#. Due to the large number of variables and constraints, we had to use MSF Enterprise Edition (MSF 2014).

6.2 Solution algorithms

Before describing the specific methods to solve the Static and Dynamic Planning, Fig. 12.2 shows a flow diagram with the different steps of the process to get the solutions.



Fig. 12.2. Flow diagram of the process to get the optimal planning.

6.2.1 Static Planning

The solution to the Static Planning returns a planning of visits from January 1st to December 31st of the current year. This version ignores the past visits, and it can be run at any time.

(1) Basic feasible planning

Initially, a basic planning is performed, which consists of the planning for all SPs and their associated customers considering only those with more than 2 visits per year. The reason to postpone the planning of this type of customers lies in the fact that when customers with a high number of visits are inserted first, the frequency of visits of the final solution is highly improved. Besides, we also want to avoid the basic planning with 2 visits too close between them.

The solution provided by this stage of the process is show in Table 12.4.

Table 12.4. Data table returned by the first stage if the process.

Name	Description
CUSTOMER	Code of the customer.
WEEK	Week number.
STARTING DATE	Date of the first day (Monday).
ENDING DATE	Date of the last day (Sunday).
SALESPERSON	Salesperson code.
PRESALE ZONE	Presales Zone code.

(2) Remove visits in adjacent weeks

After obtaining the basic planning, we apply a swap method to remove visits in two adjacent weeks. As we mentioned in constraint (4), a visit might be inserted, for example, in week $F[c]$ and another in $F[c] + 1$.

Therefore, this method swaps visits with another different customer in previous or subsequent weeks. In the case of a customer with a high number of visits (for instance, 24), it is almost inevitable that two visits won't occur in consecutive weeks.

For example, customer A has two visits in a row in weeks 12 and 13. We look for customer B who has a visit in week 11. Besides, there is no problem for customer B to have a visit in week 12. Then, we swap weeks 12 and 11 between customers A and B , as shown in Fig. 12.3.

Customer A	Customer B	Customer A	Customer B
5	3	5	3
12	11	11	12
13	16	13	16
18	18	18	18
...

Fig. 12.3. An example of the swap method to solve the visits in adjacent weeks.

(3) Insert customers with 2 visits

With this new planning, and for each SP, we go through all the customers with 2 visits and we try to insert 1 visit in the emptiest week w belonging to the first part of the year, and always guaranteeing that $VpW[w] \leq MaxVpW[w]$.

Then, the second visit is inserted the emptiest week within the range $[w + f - 2, w + f + 2]$, where $f = 26$ is the ideal frequency for customers with 2 visits.

Once this method is completed, the planning covers all the visits to all customers to be accomplished by all SPs.

(4) Improve the frequency for customers with 3, 4 and 24 visits

First, we try to improve the frequency for customers with 3 and 4 visits by checking whether the distance between one visit and the next one is less than a half the ideal frequency. Then, we insert the visit in the emptiest week within the biggest gap of weeks.

For example, customer A will receive 4 visits this year (frequency = 13). As it is shown in Fig. 12.4, it has two visits in weeks 13 and 16, respectively, that are very near in time since the gap between them is less than $13/2 = 6$ weeks. Then, we remove week 16 from the planning, and we replace it for the best week in the range of possible solutions that, in this case, is week 21.

Customer A	Customer A
13	13
16	21
30	30
...	...

Fig. 12.4. Improving the frequency for a customer with 4 visits.

The method also tries to improve the frequency for customers with 24 visits that have two visits in consecutive weeks, and the swap method previously described in step 2 failed to improve this planning. To solve this problem, we look for the position in the planning with a gap of at least 3 weeks, and then we insert one of the two consecutive visits in that position.

For example, in Fig. 12.5 we can see that the visit planning of customer *A* has two consecutive visits in weeks 18 and 19. Fortunately, there is a gap of size 3 (weeks 11, 12 and 13) between weeks 10 and 14. Therefore, we shift the visit of week 18 to week 12.

Customer A	Customer A
1	1
3	3
6	6
8	8
10	10
14	12
16	14
18	16
19	19
21	21
...	...

Fig. 12.5. Improving the frequency of a customer with 24 visits.

By virtue of this improvement we have solve two problems. On the one hand, there are no visits in successive weeks. On the other hand, the new planning is more balanced and hence, the weeks are not too separated from each other.

(5) Reduce PSZ to at most 5 in a week

In this final step, a procedure will try to keep each SP's planning below 5 PSZ in the same week. To do this, the method proceeds on each week with more than 5 different PSZ and we try to swap other visits placed in previous or subsequent

weeks. Actually, not only a single visit is swapped, but an entire group of visits from PSZ X to another group of visits in PSZ Y .

The result of this method dramatically improves the overall solution of the planning, reducing the PSZ to 5 or below on a large number of weeks that exceeded this limit.

At this stage, we have obtained the final optimized planning.

(6) Insert Promotional Campaigns

As we mentioned in the initial goals, the Promotional Campaigns are special visits made to a selected set of customers in a particular range of dates. If a visit has been already set within that range of dates, the planning remains unchanged. Otherwise, we must insert a new visit in the emptiest week of that range of dates.

6.2.1 Dynamic Planning

The steps to obtain the Dynamic Planning are almost the same ones than the Static Planning, with the slight difference that only the $(53 \text{ or } 52) - PW$ future weeks will be considered, since the past PW weeks are already planned.

The planning that was already accomplished for the past weeks PW is obtained from a table in the database, and the resulting data is inserted in the model using parameter $R[c,w]$, for all the customers $c = 1, \dots, NC$ and the corresponding weeks $w = 1, \dots, PW$.

For each customer c , the frequency of visits $F[c]$ must be recomputed according to the remaining number of visits. Besides, we can choose which SP will be re-planned, leaving the rest of them unchanged.

The main similarities and differences with the Static Planning described in Sect. 6.2.1 are the following:

- The swap method is almost the same, but it will only check the planning for those weeks greater than PW .
- The method to insert the customers with 2 visits is different, since it has to verify if such visits were performed completely (2), partially (just 1) or none of them.
- The procedures to improve the frequency for customers with 3 and 4 visits, as well as the method to reduce the number of PSZ in a week, will only consider weeks from $PW + 1$ onwards.
- The step to improve the frequency of customers with 24 visits is not applied.

At this point, it is worth mentioning that, as the project progressed, the company realized and concluded that **re-planning every few months considering the past visits** would yield better results than re-planning once a year. Therefore, they decided to **run the Dynamic Planning every 3 months**, and every time there is a change in the data from the customers, presales zones or salespersons.

7 Computing results

As we stated in the introduction, the company has 11,700 customers approximately and 53 salespersons. Some salespersons have few customers (< 100) assigned, and some others have a lot of them (> 400). Anyway, each salesperson must achieve 2,000 visits a year, approximately.

For each salesperson's planning model, we get around 13,000 variables and 26,000 constraints. Accordingly, we then run all 53 models, one for each salesperson. The running times on a regular laptop computer were 75 minutes for the Static Planning and 90 minutes for the Dynamic Planning.

The results obtained were very promising as a result of:

- For all customers and salespersons, all visits were fulfilled.
- The number of planned visits in a week remains below the maximum number of visits, and over the minimum number of visits.
- All visits are relatively spaced in time, although not perfectly space, since otherwise the equitable distribution of weeks would not hold.
- It was intended in the planning to leave no consecutive weeks, but sometimes this was impossible for customers with high number of visits (24).
- It was also intended for each salesperson to keep the number of presale zones in a week below 5, but sometimes this was not possible for salespersons with a large number of customers who are located in different presales zones.

Next, we comment out some of the results achieved for both the Static and the Dynamic Planning.

7.1 Static Planning results

For the Static Planning, we obtained in the worst case that a salesperson with the greatest number of visits (3938) had the following results:

- Number of Visits per Week ($NVPW$): 79 visits.
- Number of weeks with more than 5 PSZ: only 2 weeks with 6 PSZ, and the rest of them with 5 PSZ or less.
- Maximal number of visits per week ($MaxVPW[\cdot]$): 82 visits.
- Minimal number of visits per week ($MinVPW[\cdot]$): 49 visits.

In the case of a salesperson with an average number of visits (1946), we obtained:

- Number of Visits per Week ($NVPW$): 36 visits.
- Number of weeks with more than 5 PSZ: no week has more than 5 PSZ.
- Maximal number of visits per week ($MaxVPW[\cdot]$): 40 visits.
- Minimal number of visits per week ($MinVPW[\cdot]$): 24 visits.

7.1 *Dynamic Planning results*

The results of the Dynamic Planning considering 3 months in the past (12 weeks) for a salesperson with the higher number of visits (3938) are the following:

- Number of Visits per Week ($NVpW$): 90 visits.
- Number of weeks with more than 5 PSZ: there is only one week with 7 different PSZ, and 10 weeks with 6 PSZ. The rest of them are less than or equal to 5 PSZ.
- Maximal number of visits per week ($MaxVpW[\cdot]$): 91 visits.
- Minimal number of visits per week ($MinVpW[\cdot]$): 54 visits.

We remark that, for this particular salesperson, the visits accomplished in the past were not available due to a synchronization problem with his PDA. Consequently, the planning considered a full year starting from the following week.

On the other hand, the results for a salesperson with an average number of visits (1946) are as follows:

- Number of Visits per Week ($NVpW$): 47 visits.
- Number of weeks with more than 5 PSZ: there is only one week with 6 different PSZ, and the rest of them are less than or equal to 5 PSZ.
- Maximal number of visits per week ($MaxVpW[\cdot]$): 50 visits.
- Minimal number of visits per week ($MinVpW[\cdot]$): 30 visits.

Finally, Fig. 12.6 shows an example of a real planning for two customers with 12 visits each. The columns in the table (from left to right) denote the customer code, the planned week for the visit, starting date, ending date, salesperson code and presales zone code.

	Customer	Week	Starting date	Ending date	SP code	PSZ code
	Cliente	Semana	FechaIni	FechaFin	Comercial	ZonaPV
1	PV200018	2	2010-12-13 00:00:00.000	2010-12-19 00:00:00.000	00001292	ZPR201
2	PV200018	5	2011-01-03 00:00:00.000	2011-01-09 00:00:00.000	00001292	ZPR201
3	PV200018	9	2011-01-31 00:00:00.000	2011-02-06 00:00:00.000	00001292	ZPR201
4	PV200018	14	2011-03-07 00:00:00.000	2011-03-13 00:00:00.000	00001292	ZPR201
5	PV200018	19	2011-04-11 00:00:00.000	2011-04-17 00:00:00.000	00001292	ZPR201
6	PV200018	21	2011-04-25 00:00:00.000	2011-05-01 00:00:00.000	00001292	ZPR201
7	PV200018	26	2011-05-30 00:00:00.000	2011-06-05 00:00:00.000	00001292	ZPR201
8	PV200018	29	2011-06-20 00:00:00.000	2011-06-26 00:00:00.000	00001292	ZPR201
9	PV200018	33	2011-07-18 00:00:00.000	2011-07-24 00:00:00.000	00001292	ZPR201
10	PV200018	39	2011-08-29 00:00:00.000	2011-09-04 00:00:00.000	00001292	ZPR201
11	PV200018	42	2011-09-19 00:00:00.000	2011-09-25 00:00:00.000	00001292	ZPR201
12	PV200018	51	2011-11-21 00:00:00.000	2011-11-27 00:00:00.000	00001292	ZPR201
13	PV200084	4	2010-12-27 00:00:00.000	2011-01-02 00:00:00.000	00001292	ZCAT02
14	PV200084	8	2011-01-24 00:00:00.000	2011-01-30 00:00:00.000	00001292	ZCAT02
15	PV200084	11	2011-02-14 00:00:00.000	2011-02-20 00:00:00.000	00001292	ZCAT02
16	PV200084	13	2011-02-28 00:00:00.000	2011-03-06 00:00:00.000	00001292	ZCAT02
17	PV200084	17	2011-03-28 00:00:00.000	2011-04-03 00:00:00.000	00001292	ZCAT02
18	PV200084	21	2011-04-25 00:00:00.000	2011-05-01 00:00:00.000	00001292	ZCAT02
19	PV200084	28	2011-06-13 00:00:00.000	2011-06-19 00:00:00.000	00001292	ZCAT02
20	PV200084	32	2011-07-11 00:00:00.000	2011-07-17 00:00:00.000	00001292	ZCAT02
21	PV200084	36	2011-08-08 00:00:00.000	2011-08-14 00:00:00.000	00001292	ZCAT02
22	PV200084	39	2011-08-29 00:00:00.000	2011-09-04 00:00:00.000	00001292	ZCAT02
23	PV200084	42	2011-09-19 00:00:00.000	2011-09-25 00:00:00.000	00001292	ZCAT02
24	PV200084	50	2011-11-14 00:00:00.000	2011-11-20 00:00:00.000	00001292	ZCAT02

Fig. 12.6. An example of a real planning for two customers (PV200018 and PV200084) with 12 visits each.

8 Conclusions

The initial goal of this project was to improve the planning of visits for the 53 salespersons of a brewing company to their 11,700 customers, in order to optimize the company's resources and improve efficiency.

In this sense, several goals were accomplished from the view of each salesperson:

- We obtained a planning with regular visits and equitably spaced in time.
- Balanced working weeks.
- No more than 5 Pre-Sales Zones was achieved in a remarkable number of cases.
- The Static Planning plans the annual visits for all salespersons to all the customers from January 1st to December 31st, regardless of the past visits.

- The Dynamic Planning takes into account the visits already made in the past.
- Promotional Campaigns can be inserted in the current planning.
- Besides, the user interface allows selecting which salespersons will be optimized.

All these requirements were fully completed by generating a basic feasible planning with a mathematical solver, and then running specialized methods to improve the initial solution.

There is no doubt that, although the planning has successfully enhanced the previous method, it could be further improved to better fit the frequency of customers with 12, 18 and 24 visits, particularly in the last case that happens quite often. Since our main goal is to hold the number of visits per week initially computed by the solver, some visits remain adjacent in consecutive weeks and there is no way to perform any swap in order to solve it.

Furthermore, the running time could be significantly reduced by changing the mathematical solver for a parametric maximum flow algorithm with additional constraints.

Finally, just to mention that the application has been running every three months since January 2011.

Acknowledgments This work is partly supported by the former Spanish Ministry of Science and Innovation through the Research Project with reference MTM2010-18591 and by the new Spanish Ministry of Economy and Competitiveness through the Research Project with reference MTM2013-43396-P.

References

- Ahuja RK, Magnati TL, Orlin JB (1993) *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Upper Saddle River, NJ.
- Ferrer L, Pastor R, García-Villoria A (2009) Designing salespeople's routes with multiple visits of customers: A case study. *Int. J. Production Economics* 119: 46–54.
- LOGISPLAN (2014) Logisplan Professional Appointments. <http://planificacionderutas.logisplan.com/en/description-logisplan-appointments.html>. Accessed 10 Jun 2014.
- MSF (2014) MS Solver Foundation. <http://msdn.microsoft.com/es-es/devlabs/hh145003.aspx>. Accessed 10 Jun 2014.
- MVS (2014) MS Visual Studio. <http://msdn.microsoft.com/en-us/vstudio/aa718325>. Accessed 10 Jun 2014.
- OPTI-TIME (2014) TourSolver. <http://www.opti-time.com/en/route-optimization-software>. Accessed 10 Jun 2014.
- Polacek M, Doerner KF, Hartl RF et al. (2007) Scheduling periodic customer visits for a traveling salesperson. *European Journal of Operational Research* 179: 823–837.