[[1]](#footnote-1)

Retail Cafe

David Supančić

# Problem Statement

Retail Cafe problem is a complex optimization problem in which the goal is to optimize the allocation of units across 4 stores and efficiently route 4 waiters throughout the store to maximize the total profit.

# Description and Hypotheses

This optimization modeling project’s goal was to solve a complex optimization problem for a retail café with four branches within a mall. The primary focus was on how to efficiently allocate products and staff. To achieve this and find the optimum, it was needed to give special care to daily reception and sales as well as profit margins while taking into consideration waiters’ work schedules. For each of those either a binary or integer variable was used. Finally, necessary constraints were added. Some constraints needed relaxation to achieve flexibility. The final goal was to create a mixed integer programming model, which was ultimately achieved.

In this project, it was assumed that a server can work only in 1 store per day and in only 1 person can work in 1 store per day.

# Mathematical Formulation of the Optimization Problem

## Sets

day of the week {“Monday”, “Tuesday”, “Wednesday”, “Thursday”, “Friday”, “Saturday”, “Sunday”}

person working {“Emily”, “John”, “Sarah”, “Mike”}

store {“Bucks”, “Negro”, “Taste”, “Flower”}

## Parameters

extra cost of person [€/day]

extra profit of person [%]

maximum number of days person can work [number of days]

maximum number of units a store can receive [unit]

maximum number of days a person can work in store *s* [number of days]

minimum number of days person has to work [number of days]

minimum number of units stores can have [unit]

equals 1 if person is not in their base store, otherwise 0

profit of store s per unit [€/unit]

total reception on day [unit]

total sales on day [unit]

wage of person [€/day]

positive big number, equal to 1000000

negative big number, equal to -1000000

## Variables

table of amount of units left on day d in store s [unit]

binary variable if person *p* worked in store *s* on day {0,1}

number of units received on day in store [unit]

number of units sold on day in store [unit]

## Equations

, minimum number of days person *p* must work (1)

, maximum number of days person *p* can work (2)

, maximum one person *p* per store *s* per day *d (3)*

maximum one store *s* on day *d* per person *p (4)*

*(5)*

*(6)*

*(7)*

*(8)*

*(9)*

(10)

*(11)*

*(12)*

*(13)*

*(14)*

*ADDED CONSTRAINTS:*

(15)

(16)

## Objective Function

# Code

$TITLE *Retail Cafe*

**SETS**

d day of the week /Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday/

s store /Bucks, Negro, Taste, Flower/

p person (waiter) /Emily, John, Sarah, Mike/

;

**ALIAS** (d2, d);

**ALIAS** (p2, p);

**ALIAS** (s2, s);

**PARAMETERS**

Profit(s) profit per unit in store s /Bucks 0.6, Negro 0.8, Taste 0.9, Flower 0.7/

Sales(d) number of total units sold per day /Monday 250, Tuesday 250,

Wednesday 300, Thursday 450,Friday 550, Saturday 700, Sunday 650/

Reception(d) number of total units received per day /Monday 550, Tuesday 500,

Wednesday 500, Thursday 500,Friday 550, Saturday 550, Sunday 0/

MaxRec max number of units a store can receive per day /150/

MinStock min number of units in a store /100/

Wage(p) wage per day of person p /Emily 100, John 150, Sarah 120, Mike 110/

ExtraP(p) extra profit earned by person p /Emily 0.18, John 0.25, Sarah 0.22, Mike 0.20/

ExtraCost(p) extra cost per day from transferring person to another store /Emily 30,

John 45, Sarah 36, Mike 33/

MaxSameStore max num of days per week a person can work in the same store /3/

MinDays min days of work a person must work /4/

MaxDays max days of work a person can work /6/

BigNumber M /1000000/

NegBigNumber m /-1000000/

;

**TABLE** NotBaseStore(p,s) equals 1 if not person's base store - otherwise 0

**Bucks Negro Taste Flower**

**Emily** 0 1 1 1

**John** 1 0 1 1

**Sarah** 1 1 0 1

**Mike** 1 1 1 0

;

**VARIABLES**

X(d,s,p) bin var if p worked in s on day d

Y1(d,s) units recepted in s on day d

Y2(d,s) units sold in s on day d

UnitsLeft(d,s) amount of units left at store s at the end of day d

OFV obj func value

;

**BINARY VARIABLES** X;

**POSITIVE VARIABLES** Y1, Y2;

**EQUATIONS**

OF obj func

MAXUNITS(d,s) each store can receive max 150 units

UNITSPERSTORE(d) sum of units per store = total units

MAXUNITSSOLD(d) max total units sold

MAXDAYSSAMESTORE(s,p) max num of days a waiter can work in the same store

MINDAYSPERSON(p) min days person must work per week

MAXDAYSPERSON(p) max days person can work per week

ONESTOREONLY(s,d) only 1 person can work in a store on day d

ONEPERSONONLY(p,d) 1 person can only work at 1 store on day d

UNITSLEFTGREATER(d,s,d2) units left at the end of every day must be greater than MinStock

UNITSLEFTGREATER\_M(d,s) monday

UNITSLEFTCALC(d,s,d2) calculation of UnitsLeft variable

UNITSLEFTCALC\_M(d,s) monday

Y2\_1(d,s) Y2=0 if noone is working in s on day d

Y2\_2(d,s) Y2=0 if noone is working in s on day d

Y\_MAX(d,s)

ADDED\_1(d,p,p2) John and Mike must work on the same days

ADDED\_2(d,s,p,s2,p2) Emily stays at Bucks -> Sarah stays at Taste

ADDED\_3(d,p) Emily and John can't work at the weekend

ADDED\_4(d,s,p) all the waiters want to work outside of their base store

;

OF .. OFV =E= **SUM**((d,s,p), Y2(d,s) / **CARD**(p) \*(1+ExtraP(p))\*Profit(s)- X(d,s,p)\*Wage(p) - X(d,s,p)\*NotBaseStore(p,s)\*ExtraCost(p));

MAXUNITS(d,s) .. Y1(d,s) =L= MaxRec;

UNITSPERSTORE(d) .. **SUM**(s, Y1(d,s)) =E= Reception(d);

MAXUNITSSOLD(d) .. **SUM**(s, Y2(d,s)) =L= Sales(d);

MAXDAYSSAMESTORE(s,p) .. **SUM**(d, X(d,s,p)) =L= MaxSameStore;

MINDAYSPERSON(p) .. **SUM**((d,s), X(d,s,p)) =G= MinDays;

MAXDAYSPERSON(p) .. **SUM**((d,s), X(d,s,p)) =L= MaxDays;

ONESTOREONLY(s,d) .. **SUM**(p, X(d,s,p)) =L= 1;

ONEPERSONONLY(p,d) .. **SUM**(s, X(d,s,p)) =L= 1;

UNITSLEFTGREATER(d,s,d2)$(**ord**(d)=**ord**(d2)+1) .. UnitsLeft(d2, s) + Y1(d,s) - Y2(d,s) =G= MinStock;

UNITSLEFTGREATER\_M(d,s)$(**ord**(d)=1) .. MinStock + Y1(d,s) - Y2(d,s) =G= MinStock;

UNITSLEFTCALC(d,s,d2)$(**ord**(d)=**ord**(d2)+1) .. UnitsLeft(d,s) =E= UnitsLeft(d2, s) + Y1(d,s) - Y2(d,s);

UNITSLEFTCALC\_M(d,s)$(**ord**(d)=1) .. UnitsLeft(d,s) =E= MinStock + Y1(d,s) - Y2(d,s);

Y2\_1(d,s) .. Y2(d,s) =L= **SUM**(p, X(d,s,p)) \* BigNumber;

Y2\_2(d,s) .. Y2(d,s) =G= **SUM**(p, X(d,s,p)) \* NegBigNumber;

ADDED\_1(d,p,p2)$(**ord**(p)=2 **and** **ord**(p2)=4) .. **SUM**(s, X(d,s,p)) =E= **SUM**(s, X(d,s,p2));

ADDED\_2(d,s,p,s2,p2)$(**ord**(s)=1 **and** **ord**(p)=1 **and** **ord**(s2)=3 **and** **ord**(p2)=3) .. X(d,s,p) =L= X(d,s2,p2);

ADDED\_3(d,p)$((**ord**(d)=6 **or** **ord**(d)=7) **and** (**ord**(p)=1 **or** **ord**(p)=2)) .. **SUM**(s, X(d,s,p)) =E= 0;

ADDED\_4(d,s,p)$(**ord**(p) = **ord**(s)) .. X(d,s,p) =E= 0;

$ontext

*MODEL RETAILCAFE0 /OF, MAXUNITS, MAXUNITSSOLD, UNITSPERSTORE, MAXDAYSSAMESTORE, MINDAYSPERSON,*

*MAXDAYSPERSON, ONESTOREONLY, ONEPERSONONLY, UNITSLEFTGREATER, UNITSLEFTGREATER\_M,*

*UNITSLEFTCALC, UNITSLEFTCALC\_M, Y2\_1, Y2\_2/;*

*SOLVE RETAILCAFE0 MAXIMIZING OFV USING MIP;*

$offtext

$ontext

*MODEL RETAILCAFEA /OF, MAXUNITS, MAXUNITSSOLD, UNITSPERSTORE, MAXDAYSSAMESTORE, MINDAYSPERSON,*

*MAXDAYSPERSON, ONESTOREONLY, ONEPERSONONLY, UNITSLEFTGREATER, UNITSLEFTGREATER\_M,*

*UNITSLEFTCALC, UNITSLEFTCALC\_M, Y2\_1, Y2\_2,*

*ADDED\_1/;*

*SOLVE RETAILCAFEA MAXIMIZING OFV USING MIP;*

$offtext

$ontext

*MODEL RETAILCAFEB /OF, MAXUNITS, MAXUNITSSOLD, UNITSPERSTORE, MAXDAYSSAMESTORE, MINDAYSPERSON,*

*MAXDAYSPERSON, ONESTOREONLY, ONEPERSONONLY, UNITSLEFTGREATER, UNITSLEFTGREATER\_M,*

*UNITSLEFTCALC, UNITSLEFTCALC\_M, Y2\_1, Y2\_2,*

*ADDED\_2/;*

*SOLVE RETAILCAFEB MAXIMIZING OFV USING MIP;*

$offtext

*\*$ontext*

**MODEL** RETAILCAFEAB /OF, MAXUNITS, MAXUNITSSOLD, UNITSPERSTORE, MAXDAYSSAMESTORE, MINDAYSPERSON,

MAXDAYSPERSON, ONESTOREONLY, ONEPERSONONLY, UNITSLEFTGREATER, UNITSLEFTGREATER\_M,

UNITSLEFTCALC, UNITSLEFTCALC\_M, Y2\_1, Y2\_2,

ADDED\_1, ADDED\_2, ADDED\_4/;

**SOLVE** RETAILCAFEAB **MAXIMIZING** OFV **USING** **MIP**;

*\*$offtext*

# Results

The maximum total profit equals 882.38€. The optimal waiter routing and units allocation is shown in the following tables:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Bucks | Negro | Taste | Flower |
| Monday | - | John | - | Mike |
| Tuesday | - | John | - | Mike |
| Wednesday | Emily | John | Sarah | Mike |
| Thursday | Emily | - | Sarah | - |
| Friday | John | Mike | - | - |
| Saturday | - | Emily | - | Sarah |
| Sunday | Emily | - | Sarah | - |

**Table 1. Which person p works on day d in store s, variable X.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Bucks | Negro | Taste | Flower |
| Monday | 100 | 150 | 150 | 150 |
| Tuesday | 50 | 150 | 150 | 150 |
| Wednesday | 50 | 150 | 150 | 150 |
| Thursday | 50 | 150 | 150 | 150 |
| Friday | 100 | 150 | 150 | 150 |
| Saturday | 100 | 150 | 150 | 150 |
| Sunday | - | - | - | - |

**Table 2. How many units are received in store s on day d, variable Y1.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Bucks | Negro | Taste | Flower |
| Monday | - | 100 | - | 150 |
| Tuesday | - | 200 | - | 50 |
| Wednesday | - | 50 | - | 250 |
| Thursday | 100 | - | 350 | - |
| Friday | 250 | 300 | - | - |
| Saturday | - | 250 | - | 450 |
| Sunday | 100 | - | 550 | - |

**Table 3. How many units are sold in store s on day d, variable Y2.**

# Extension of the Case Study

This section describes the extensions to the original statement of the problem that the working group has considered reasonable to analyze. Own initiative would be evaluated as you, the group, have complete freedom to propose any extension. The study and quality of the extension would also be assessed.

The extensions can be, for instance, parametric sensitivity analysis, creative changes in the problem statement, and referenced comments of possible real applications that briefly adhere to the problem statement.

1. If the waiter’s salary is increased by 10€ and their profit by 10%, the new value of maximum total profit is 965.38€.
2. If Emily and John can’t work at the weekend then the maximum total profit is 652.31€.

This is done by adding the following constraint:

1. In case that this week all the waiters want to work only outside of their base store, the maximum total profit for the week is 450.38€.

This is done by adding the following constraint:

# Conclusions

To conclude, 2 added constraints made by the human resources department do not change the schedule nor the total profit which equals to 882.38€.

1. Put the submission date in this footnote: November 5, 2023. [↑](#footnote-ref-1)