# A Pure Binary LP Model to the Facility Layout Problem

Christos Papahristodoulou[1]

## *Abstract*

In facility layout problems, a major concern is the optimal design or remodeling of the facilities of an organization. The decision maker's objective is to arrange the facility in an optimal way, so that the interaction among functions (i.e. machines, inventories, persons) and places (i.e. offices, work locations, depots) is efficient. A simple pure-binary LP model is developed and solved for a small hospital, where five functions are to be redesigned to five different locations. The model is rather flexible and can be used, with small modifications, for larger facility layouts.

JEL classification: C60, C61, C63.

Keywords: Facility Layout, Binary, Linear Programming, Optimization, Efficiency.

[1] Department of Economics, Uppsala University, Box 513, SE-751 20 Uppsala
email: Christos.Papahristodoulou@nek.uu.se

## Introduction

It is well known that, in facility layout problems one investigates where each function, in a given floor space, will be placed, when all functions interact with each other. Such locations will influence material handling or distance costs and consequently the efficiency of facility. Typical facility layout problems arise in the design or renovation of factories, distribution centers, hospitals, banks, department stores, military supply depots, universities etc., so that functions with high (low) rate of interaction will be placed close (away) to (from) each other. Thus, the distance or time cost of items or persons will be minimized and the efficiency will increase.

Nahmias [1997], refering to some studies, argues that the US spent more than $ 500 billion annualy on construction and modification of facilities. Effective facilities planning could reduce costs by 10 to 30 percent per year. He also believes that intelligent layout is a key factor to the Japanese production efficiency.

Finding a facility's optimal layout has been studied mainly by industrial engineers and researchers in operations research. Among the first who studied this problem are Armour and Buffa [1963]. Little seems to have been published in the 1950s. Francis and White [1974], were the first who collected and updated the early research on this area. Later research has been updated by two recent studies, the first by Domschke and Drexl [1985] and the other by Francis, McGinnis and White [1992].

In general, two approaches have been applied to solve facility layout problems with many functions and locations. The first approach is based on greedy pairwise exchange heuristic. It starts with an initial layout and then seeks an improved one by exchanging the locations of a pair of functions. These approaches are ”greedy” in a sense that they often exchange the pair of functions with the largest net reduction in total travel time from the locations. The pairwise exchanges[2] are repeated as long as improvements are possible. Since these heuristics consider only two-way exchanges, they do not guarantee that the optimal layout will be found, if for instance all functions need to be exchanged. The second approach is based on a binary integer quadratic objective function. Particular software packages such as CRAFT (Computerized Relative Allocation of Facilities Technique), by Armour and Buffa [1963], SDPIM (Steepest Descent Pairwise Interchange Method) developed by STORM Software, or GRASP (Greedy Randomized Adaptive

[2] If there is an exchange in locations between functions *k* and *l*, the number of function pairs whose travel time change is 2(n-2), where n is the number of locations or functions. This is because there are 2 ways to choose the member of the pair that must be either *k* or *l*, and (n-2) ways to choose the member of the pair that is different from *k* or *l*.

Search Procedure) by Resende, Li and Pardalos [1996], claim that these algorithms are efficient and solve large problems (with more than 15 functions and locations). These algorithms are however, rather complex. For instance, the GRASP algorithm is almost fifteen pages long!

In this paper a simple pure-binary (integer) linear programming (PBLP) model has been developed to solve a hypothetical layout problem for a small hospital. The model is rather flexible and could be applied to more functions and locations with simple modifications. As in binary integer quadratic models, the number of iterations increases dramatically and it may take a long time to find the optimal layout of more functions and locations.

## A hypothetical example

Limited resources in the public sector necessitate additional measures to increase efficiency. Hospital services is a typical example where doctors' and nurses' time is insufficient to meet the demand by patients. According to nurses' observations and experience, from a small hospital, some, or all five functions located on the same floor, are placed in wrong positions with regard to the daily interactions between each pair of functions. That leads to unnecessary long trips by both the staff and patients. Considerable travel time savings might be derived if these particular functions are reallocated to rooms.

The following figure depicts the actual allocation of these five functions to the respective location (room). Since there are 5! = 120 different layouts, the problem is to find out the best one.

**A**
Examination Room

**C**
Hematology Lab

**B**
X-ray Room

**D**
Waiting Room

**E**
Medical Records

**Entrance**

**Figure 1: Hospital's actual layout**

The symmetric figure 2 displays two data sources. The first entry shows the time (in seconds) it takes for the hospital staff (and the patients as well) to travel from room to room. For instance, it takes almost half a minute to travel from A (Examination Room) to C (Hematology Lab), or vice versa. The second entry shows the daily frequency of interaction between each pair of functions (which might be the average of an observed period). For instance, a nurse will make 180 single trips following all patients who must go from the Waiting Room (D) to the Hematology Lab (C), and another 150 single trips from A to B. Contrary to the distance observations which are easy to collect, the trips are of course not stable and change very often.

In addition, there are some other idiosyncrasies with a hospital's layout. For instance, different functions might need different space and must be placed in certain locations. Even safety or aesthetic reasons should be taken into consideration when facilities are to be located. All that makes the problem dynamic or stochastic. Moreover, it would be extremely difficult to formulate, solve and above all implement all statistically significant layouts. Even a deterministic facility layout problem is complicated enough. We therefore, disregard all these problems and consider it as static or deterministic. Obviously, the importance of idiosyncrasies, of fixed costs and the large fluctuations in the average number of trips, must be taken into account before the optimal layout is implemented.

| | (A) **Examin. Room** | (B) **X-ray** | (C) **Hematology** | (E) **Med. Records** |
|---|---|---|---|---|
| (D) **Waiting Room** | (10, 230) | (15, 0) | (35, 180) | (28, 50) |
| (A) **Examin. Room** | | (18, 150) | (28, 130) | (35, 70) |
| (B) **X-ray Room** | | | (18, 200) | (15, 60) |
| (C) **Hematology Lab** | | | | (10, 100) |

**Figure 2: Time and Patient-trips between each pair of functions**

It is meaningless to know the number of patients. Some of them will have to go through all the functions while others might need only an X-ray or a Hematology eaxamination. Some of them might go straight to get their Medical Records from previous examinations, while the majority of them will have to wait at the Waiting room. Even if all of them must wait at the Waiting room before they go to the specific investigation rooms, we cannot count the number of patient-services.[3] In addition, since the examination sequence is not

[3] The maximum amount of patients (not patient-services) is all those who start from the Waiting room, i.e. 460, if they go straight to these different functions and never return there. A possible solution consistent with 460 patients is given in the following matrix (minus indicates inflow of patients from the respective room):

taken into account, the number of patient-trips shown in figure is the maximum amount from both directions. For instance, 200 patient-trips between B and C should be regarded as any combination of 200 patients going between B and C, such as 120 from B to C and 80 from C to B.

If we multiply the travel time with the number of trips made daily, we compute the total travel time. According to this layout, it takes 24,290 seconds (almost 6 hours and three quarters).

In principle, one has to enumerate 120 different total travel times and choose the minimum. Obviously, if we had to assign ten functions to ten locations we would have to choose one out of 10! = 3,628,800 possible layouts, a very cumbersome, if not erratic work.

**A simple PBLP model**

Although the enumeration of layouts is not avoided, a PBLP model has been developed to find the best layout.

Define first 100 binary (0,1) variables $X_k = L_i * F_j$ , where k = 1, 2,.....100, $L_i$ = pair of locations according to the capital letters (rooms), with i = 1,2, ... 10, and $F_j$ = pair of functions according to their locations, with j = 1,2,....10. This definition of variables allows us to linearize the problem which in fact is non-linear (multiplicative). Thus, $X_1$ = (AB)*(Waiting Room/Examination Room), $X_2$ = (AC)*(Waiting Room/Examination Room), ...,$X_9$ = (CE)*(Waiting Room/Examination Room), $X_{10}$ = (DE)*(Waiting Room/Examination Room). In accordance with that, we define all $X_k$. For instance,

$X_{11}$ = (AB)*(Waiting Room/X-Ray Room), ...,
$X_{21}$ = (AB)*(Waiting Room/Hematology Lab), ...,
$X_{100}$= (DE)*(Hematology Lab/Medical Records).

| | | To | | | | |
|---|---|---|---|---|---|---|
| | | X-ray | Hematol. | Examin. | Med. Rec. | Out |
| From | Waiting | 0 | 180 | 230 | 50 | 0 |
| | X-ray | 0 | -200 | 150 | -60 | -110 |
| | Hematol. | 200 | 0 | -130 | 100 | 170 |
| | Examin. | -150 | 130 | 0 | 70 | 50 |
| | Med. Rec. | 60 | -100 | -70 | 0 | -110 |
| | Out | 110 | 10 | 180 | 160 | 460 |

All variables are shown in the function-location pair matrix (Table 1). The actual layout is also marked in italics. For instance, $\boldsymbol{X_3}$ implies that the Examination Room is placed in A and the Waiting Room in D, $\boldsymbol{X_{75}}$ implies that the X-ray room is placed in B and Hematology in C and so on.

Different objective functions can be chosen. One for instance might care of the largest number of patients and want to minimize the total travel time for them. However, since the aim is to increase the hospital efficiency, the hospital's objective function is:

**min** $4{,}140X_1 + 6{,}440X_2 + \ldots.. + 1{,}080\, X_{91} + \ldots.. + 2{,}800\, X_{100}$

Regarding the structural constraints, we proceed as follows.

Let us consider the (Examination Room/Waiting Room) pair. That pair can be placed in one of 10 possible pair of locations [ (AB), (AC), (AD), (AE), (BC), (BD), (BE), (CD), (CE) and (DE)]. Therefore we require that:

$$X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7 + X_8 + X_9 + X_{10} = 1 \quad (1)$$

| | **AB 18** | **AC 28** | **AD 10** | **AE 35** | **BC 18** | **BD 15** | **BE 15** | **CD 35** | **CE 10** | **DE 28** |
|---|---|---|---|---|---|---|---|---|---|---|
| **ER-WR 230** | $X_1$ | $X_2$ | $\boldsymbol{X_3}$ | $X_4$ | $X_5$ | $X_6$ | $X_7$ | $X_8$ | $X_9$ | $X_{10}$ |
| **XR-WR 0** | $X_{11}$ | $X_{12}$ | $X_{13}$ | $X_{14}$ | $X_{15}$ | $\boldsymbol{X_{16}}$ | $X_{17}$ | $X_{18}$ | $X_{19}$ | $X_{20}$ |
| **He-WR 180** | $X_{21}$ | $X_{22}$ | $X_{23}$ | $X_{24}$ | $X_{25}$ | $X_{26}$ | $X_{27}$ | $\boldsymbol{X_{28}}$ | $X_{29}$ | $X_{30}$ |
| **WR-MR 50** | $X_{31}$ | $X_{32}$ | $X_{33}$ | $X_{34}$ | $X_{35}$ | $X_{36}$ | $X_{37}$ | $X_{38}$ | $X_{39}$ | $\boldsymbol{X_{40}}$ |
| **ER-XR 150** | $\boldsymbol{X_{41}}$ | $X_{42}$ | $X_{43}$ | $X_{44}$ | $X_{45}$ | $X_{46}$ | $X_{47}$ | $X_{48}$ | $X_{49}$ | $X_{50}$ |
| **ER-He 130** | $X_{51}$ | $\boldsymbol{X_{52}}$ | $X_{53}$ | $X_{54}$ | $X_{55}$ | $X_{56}$ | $X_{57}$ | $X_{58}$ | $X_{59}$ | $X_{60}$ |
| **ER-MR 70** | $X_{61}$ | $X_{62}$ | $X_{63}$ | $\boldsymbol{X_{64}}$ | $X_{65}$ | $X_{66}$ | $X_{67}$ | $X_{68}$ | $X_{69}$ | $X_{70}$ |
| **XR-He 200** | $X_{71}$ | $X_{72}$ | $X_{73}$ | $X_{74}$ | $\boldsymbol{X_{75}}$ | $X_{76}$ | $X_{77}$ | $X_{78}$ | $X_{79}$ | $X_{80}$ |
| **XR-MR 60** | $X_{81}$ | $X_{82}$ | $X_{83}$ | $X_{84}$ | $X_{85}$ | $X_{86}$ | $\boldsymbol{X_{87}}$ | $X_{88}$ | $X_{89}$ | $X_{90}$ |
| **He-MR 100** | $X_{91}$ | $X_{92}$ | $X_{93}$ | $X_{94}$ | $X_{95}$ | $X_{96}$ | $X_{97}$ | $X_{98}$ | $\boldsymbol{X_{99}}$ | $X_{100}$ |

**Table 1: The function-location pair matrix with 5 functions and 5 locations**

In a similar way we proceed with all other pairs.

$X_{11} + .... + X_{20} = 1$ (2)

.......

$X_{91} + .... + X_{100} = 1$ (10)

In addition, to avoid the possibility of placing more than one function to the same location, we require that each pair of location should receive only one pair of functions, i.e. for the location pair (AB) the appropriate constraint is:

$$X_1 + X_{11} + X_{21} + X_{31} + X_{41} + X_{51} + X_{61} + X_{71} + X_{81} + X_{91} = 1 \quad (11)$$

We continue with the remaining location pairs (i e. all the columns).

$X_2 + .... + X_{92} = 1$ (12)

......

$X_{10} + .... + X_{100} = 1$ (20)

To avoid illogical pair allocations, i.e. the possibility of multiplying correct (incorrect) trips with incorrect (correct) time, we introduce 25 = (5$*$5) binary variables $Y_t$ = (0,1), where t = 1, 2, .....,25, in order to catch up every ”1” of location pairs and combine it with every ”1” of function pairs, in such a way so that wrong multiplication of trips by time will be excluded.

To understand the function of this binary variable, let us look at the tableau above. Consider the first four rows and the first four columns marked (i.e. all Waiting Room and A pairs). It is obvious that either <u>one</u> or <u>four</u> of these 16 variables will take the value 1. For instance, $X_3 = 1$ is possible, as in the actual layout. The first and the thirteenth constraints are then satisfied, implying that either the Waiting Room or the Examination Room will be placed either in A or in D. We are not certain yet which one will be placed where. That will depend upon which of the remaining $X_k$ will take the value 1. The other possibility is having $X_3 = X_{14} = X_{21} = X_{32} = 1$, satisfying constraints (1) to (4) and (11) to (14), if all other $X_k$:s below these four columns and to the right of these four rows are zero. But, it is not possible to have only two or three of $X_{k:s}$ equal to 1! If for instance $X_3 = X_{14} = 1$, we are certain that the Waiting Room will be placed in A, the Examination Room in D and the X-ray Room in E. Given the remaining Waiting Room pairs with the Hematology Lab and Medical Records, and the remaining A pairs with B and C, two more variables must be equal to 1, either ($X_{21}$, $X_{32}$), or ($X_{22}$, $X_{31}$). Now, it is easy to formulate that constraint.

$$X_1 + X_2 + X_3 + X_4 + X_{11} + X_{12} + X_{13} + X_{14} + X_{21} + X_{22} + X_{23} + X_{24} + X_{31} + X_{32} + X_{33} + X_{34} - 3\,Y_1 = 1 \qquad (21)$$

Let us check how this binary constraint works. If $Y_1 = 0$, then only one of these four function pairs (Waiting Room with all others) is placed correctly and all others are placed wrong, i.e., only one of these 16 variables will take the value 1. If $Y_1 = 1$, then <u>all</u> four function pairs are placed correctly, like the above mentioned ($X_3 = X_{14} = X_{21} = X_{32} = 1$), implying that the Waiting Room will be placed in A, the Examination Room be placed in D, the X-ray Room in E, the Hematology Lab in B and the Medical Records in C.

We continue with the same four columns and the Examination Room pairs (i.e. rows 1, 5, 6 and 7). The same argument applies. Either <u>one</u> or <u>four</u> of these 16 variables will take the value 1. That constraint is formulated as:

$$X_1 + X_2 + X_3 + X_4 + X_{41} + X_{42} + X_{43} + X_{44} + X_{51} + X_{52} + X_{53} + X_{54} + X_{61} + X_{62} + X_{63} + X_{64} - 3\,Y_2 = 1 \qquad (22)$$

Obviously, it is possible for both $Y_1$ and $Y_2$ to be zero, but not both to be equal to 1. If for instance, $Y_1 = 1$, then four variables of the first block (such as those mentioned above) will be equal to 1, implying that the functions will be placed as above. The 22$^{st}$ constraint must be consistent with that too, that is, $Y_2 = 0$. That is easily proved. From the first four column constraints, (11 to 14), all variables below that block (i.e. $X_{41}$, $X_{42}$...., $X_{64}$), will take the value zero. Since, from the first constraint, only one out of the first four $X_{k:s}$ equals to 1 (in our case $X_3$), constraint (22) implies that $Y_2 = 0$. Therefore, this new binary has not changed the layout. On the other hand, the possibility of $Y_1$ being equal to zero, does not necessarily imply that $Y_2 = 1$. If only $X_3 = 1$, does not necessarily imply that three more variables in constraint (22) will be equal to 1 too. There are three more functions (i.e. three more $Y_t$:s) left which can be paired with A. Thus, only one of these first five $Y_{t:}$s is equal to 1. Moreover, such a constraint is superfluous and is taken care of by the row and column constraints above.

With proceed similarly with the X-ray Room, the Hematology Lab and the Medical Records constraints:

$$X_{11} + X_{12} + X_{13} + X_{14} + X_{41} + X_{42} + X_{43} + X_{44} + X_{71} + X_{72} + X_{73} + X_{74} + X_{81} + X_{82} + X_{83} + X_{84} - 3\,Y_3 = 1 \qquad (23)$$

$$X_{21} + X_{22} + X_{23} + X_{24} + X_{51} + X_{52} + X_{53} + X_{54} + X_{71} + X_{72} + X_{73} + X_{74} + X_{91} + X_{92} + X_{93} + X_{94} - 3\,Y_4 = 1 \qquad (24)$$

$$X_{31} + X_{32} + X_{33} + X_{34} + X_{61} + X_{62} + X_{63} + X_{64} + X_{81} + X_{82} + X_{83} + X_{84} + X_{91} + X_{92} + X_{93} + X_{94} - 3\,Y_5 = 1 \qquad (25)$$

We have now finished with the location A and repeat it with the remaining locations (B, C, D and E). There are 20 constraints left (5 for each location).

For instance, the last constraint (location E) is:

$$X_{34} + X_{37} + X_{39} + X_{40} + X_{64} + X_{67} + X_{69} + X_{70} + X_{84} + X_{87} + X_{89} + X_{90} + X_{94} + X_{97} + X_{99} + X_{100} - 3\,Y_{25} = 1 \qquad (45)$$

The formulation is now complete. We used alltogheter 125 binary variables and 45 constraints. Mathematica provided the following optimal solution (in almost twenty minutes of computing time[4]:

$$X_3 = X_{14} = X_{21} = X_{32} = X_{50} = X_{56} = X_{68} = X_{77} = X_{89} = X_{95} = 1$$

$$Y_1 = Y_9 = Y_{15} = Y_{17} = Y_{23} = 1$$

Minimum objective function = 20,940 seconds (i.e. almost 5 hours and 49 minutes), that is an efficiency gain of 55 minutes per day (almost 14 %), compared with the actual layout (see table below). All five functions were placed wrong! Observe that although $X_3 = 1$, as in the initial layout, the Waiting and Examination Rooms have now changed place, so that the number of trips and the time remains unchanged for that pair. In addition, the X-ray shifted from B to E, the Medical Records from E to C and the Hematology Lab from C to B. The solution is shown in Table 2. The initial layout variables are marked in italics.

As was mentioned earlier, if the rooms are not equal in size, this layout might not be optimal. For instance, if the X-ray is moved from, say, the large room B into smaller room E, the hospital's X-ray capacity will decline, unless the space of room B is expanded. That of course will increase the capital costs and disturb the hospital's services during the works.

[4] QSB+ provided the same solution in 6 minutes (after 59 iterations).

| | AB | AC | AD | AE | BC | BD | BE | CD | CE | DE | time |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **WR-ER** | | | $\mathbf{X_{3}}$ $x_{3}$ | | | | | | | | 2,300 |
| **WR-XR** | | | | $\mathbf{X_{14}}$ | | $x_{16}$ | | | | | 0 |
| **WR-He** | $\mathbf{X_{21}}$ | | | | | | | $x_{28}$ | | | 3,240 |
| **WR-MR** | | $\mathbf{X_{32}}$ | | | | | | | | $x_{40}$ | 1,400 |
| **ER-XR** | $x_{41}$ | | | | | | | | | $\mathbf{X_{50}}$ | 4,200 |
| **He-ER** | | $x_{52}$ | | | | $\mathbf{X_{56}}$ | | | | | 1,950 |
| **MR-ER** | | | | $x_{64}$ | | | | $\mathbf{X_{68}}$ | | | 2,450 |
| **He-XR** | | | | | $x_{75}$ | | $\mathbf{X_{77}}$ | | | | 3,000 |
| **MR-XR** | | | | | | | $x_{87}$ | | $\mathbf{X_{89}}$ | | 600 |
| **He-MR** | | | | | $\mathbf{X_{95}}$ | | | | $x_{99}$ | | 1,800 |

**Table 2: The optimal and the initial solution**

## Modifications

Various modifications to the formulation above are possible. For instance, the rows and columns constraints [(1) to (20)] are not necessary, if instead, the following five constraints are introduced:

$$Y_1 + Y_2 + Y_3 + Y_4 + Y_5 = 1,\ Y_6 + Y_7 + Y_8 + Y_9 + Y_{10} = 1,$$
$$Y_{11} + Y_{12} + Y_{13} + Y_{14} + Y_{15} = 1,\ Y_{16} + Y_{17} + Y_{18} + Y_{19} + Y_{20} = 1,$$
$$Y_{21} + Y_{22} + Y_{23} + Y_{24} + Y_{25} = 1$$

These constraints exclude the possibility of having more than one $X_k$ at the same row or the same column (see proof in Appendix 1).

The PBLP formulation is rather flexible and can be applied to larger layouts of rectangular and no-rectangular structure. Other structural constraints, such as a particular function must be placed at that specific place, are of course easy to formulate and save considerable computing time. Table 3 summarizes some key points of larger rectangular layouts.

For instance, when the number of Rooms and Functions increases to 6 and 6, the function-location matrix dimension is 15x15 and includes 225 pair variables. There are 36 sub-matrices of 5x5 dimension and therefore 36 binary variables and 36 constraints where each binary will operate. In addition, there are 30 structural constraints, one for every row and every column.[5] Therefore,

---

[5] The number of columns or rows is equal to the number of ways to choose 2 of the n-functions, for an exchange in location, i.e. n!/(n-2)!2!

| Rooms & Functions | Number of Layouts N! | Number of X Variables $[n!/2!(n-2)!]^2$ | Binary Y Variables n*n | Number of Constraints | Computing Time (min) |
|---|---|---|---|---|---|
| 4,4 | 24 | 36 | 16 | 2*6 + 16 | <1 |
| 5,5 | 120 | 100 | 25 | 2*10 + 25 | 20 |
| 6,6 | 720 | 225 | 36 | 2*15 + 36 | 436 |
| 7,7 | 5040 | 441 | 49 | 2*21 + 49 | ? |
| 8,8 | 40320 | 784 | 64 | 2*28 + 64 | ? |
| 9,9 | 362880 | 1296 | 81 | 2*36 + 81 | ? |
| 10,10 | 3628800 | 2025 | 100 | 2*45 + 100 | ? |

**Table 3: Key characteristics of rectangular layouts**

the new 31st constraint where the first binary ($Y_1$) operates will be formulated as:

$$X_1 + X_2 + X_3 + X_4 + X_5 + X_{16} + X_{17} + X_{18} + X_{19} + X_{20} + X_{31} + X_{32} + X_{33} + X_{34} + X_{35} + X_{46} + X_{47} + X_{48} + X_{49} + X_{50} + X_{61} + X_{62} + X_{63} + X_{64} + X_{65} - 4\,Y_1 = 1$$

The same argument applies. If $Y_1 = 0$, then only one of these five function pairs is placed correctly. If $Y_1 = 1$, then all five pairs are placed correctly. We do not need to examine the possibility of having two, three or four pairs placed correctly, because in these cases all five pairs are placed correctly too. If more than one variables take the value one, such as $X_1 = X_{17} = X_{33} = 1$, we are certain that the first function will be placed in A, the second in B, the third in C and the fourth in D. Given the remaining function pairs (first with fifth and sixth), and the remaining room pairs (A with E and F), two more variables must be equal to 1, either ($X_{49}$, $X_{65}$), or ($X_{50}$, $X_{64}$) to satisfy the equality constraint.

In general, the coefficient of $Y_t$ is equal to (n-2), where n is the number of functions or locations.

It took almost seven and a half hours to solve that problem in Mathematica, i.e. the computing time increased by a factor of 21.8 times. Considerable time can be saved though, if the following simple steps are considered.

Consider the binary $Y_t$ Table 4. Start first with location A (the first binary row) and select which function will be placed there (based on the lowest value in objective function). Take then all columns (one at a time), to check if the same function will be placed in A or somewhere else. There are 1 + 6 subproblems to solve. The minimum objective function (31,850) is obtained when $Y_1 = 1$.

| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| A | $Y_1$ | $Y_2$ | $Y_3$ | $Y_4$ | $Y_5$ | $Y_6$ |
| B | $Y_7$ | $Y_8$ | $Y_9$ | $Y_{10}$ | $Y_{11}$ | $Y_{12}$ |
| C | $Y_{13}$ | $Y_{14}$ | $Y_{15}$ | $Y_{16}$ | $Y_{17}$ | $Y_{18}$ |
| D | $Y_{19}$ | $Y_{20}$ | $Y_{21}$ | $Y_{22}$ | $Y_{23}$ | $Y_{24}$ |
| E | $Y_{25}$ | $Y_{26}$ | $Y_{27}$ | $Y_{28}$ | $Y_{29}$ | $Y_{30}$ |
| F | $Y_{31}$ | $Y_{32}$ | $Y_{33}$ | $Y_{34}$ | $Y_{35}$ | $Y_{36}$ |

**Table 4: The binary $Y_t$ table for a rectangular 6x6 layout**

Second, set all other binaries at the same row and column equal to zero, and continue with location B and all remaining (five) columns (one at a time) as before. Apply the same criterion, when the binaries are selected. There are 1 + 5 subproblems to solve. The minimum objective function (32,325) is obtained when $Y_8 = 1$.

Third, set all other binaries at the same row and column equal to zero, and continue with location C and all remaining (four) columns (one at a time) as before. Apply the same criterion, when the binaries are selected. There are 1 + 4 subproblems to solve. The minimum objective function (32,845) is obtained when $Y_{15} = 1$.

Repeat with all remaining functions and locations. The minimum objective function (36,195) is obtained when $Y_{22} = 1$, $Y_{30} = 1$ and $Y_{35} = 1$.

To check if that solution is optimal, there are at least two options. The first one is to use that value as an upper bound and solve the entire problem (with all its 36 binaries). Mathematica found the optimal solution (35,650) in almost five hours (shown in Appendix 2). The second option is to check, after each step, if an optimal solution (i. e. if all functions are placed correctly) has been found already. In the third step for instance, the minimum objective function selected (32,845), increased to 36,195 when the remaining three functions were placed correctly. That sub-solution did not care about what it was going to happen later on. On the other hand, the third step (trying to pair functions with location C), provided a higher value (third higher) in objective function (35,650), and placed all other functions correctly too! Therefore, since that value is lower than 36,195, it might be an optimal one. To check it, we set that value as an upper bound and solve the entire problem. Mathematica solved (and found) it, in almost four hours.

The model has not been formulated yet for layouts with more than 6 Rooms and 6 Functions. I expect that there will be an optimal solution though, at least in two to three days of calculations, if Mathematica does not run out of memory! My first priority though is to collect real patient data, take into account the size of rooms, for the 5x5 layout, to check the hospital's efficiency.

## References

Armour, J.M., and Buffa, E.S. 1963, A Heuristic Algorithm and Simulation Approach to Relative Allocation of Facilities, *Management Science 9, pp. 294-309.*

Chang, Y-L., and Sullivan, R.S. 1989, *QSB+, Quantitative Systems for Business Plus*, Prentice Hall, Englewood Cliffs, NJ.

Domschke, W., and Drexl, A. 1985, *Location and Layout Planning: An International Bibliography*, Springer Verlag, Berlin.

Emmons, H et al. 1992, *STORM, Quantitative Modeling for Decision Support*, Prentice Hall, Englewood Cliffs, NJ.

Francis, R.L., and White J.A.1974, *Facility Layout and Location: An Analytical Approach*, Prentice-Hall, Englewood Cliffs, NJ.

Francis, R.L., McGinnis L.F. and White J.A. 1992, *Facility Layout and Location: An Analytical Approach*, Prentice-Hall, Englewood Cliffs, NJ.

Nahmias, S. 1997, *Production and Operations Analysis*, 3rd ed. Irwin/McGraw-Hill, Singapore.

Nemhauser, G.L., and Wolsey L.A. 1989, "Integer Programming," in *Optimization, Handbooks in Operations Research and Management Science*, eds. G.L. Nemhauser et al., Elsevier Science, B.V.

Resende, M.G.C., Li Y. and Pardalos P.M. 1996, " A Greedy Randomized Adaptive Search Procedure (GRASP) for the Quadratic Assignement Problem (QAP)," *Transactions on Mathematical Software*, Vol. 22, No. 1, pp. 104-118.

Shogan, A.W.1988, *Heuristics* (suppl. to *Management Science*), Prentice-Hall, Englewood Cliffs, NJ.

## Appendix 1

To prove that these new binary constraints exclude the possibility of having more than one $X_k$ at the same row or the same column, let us assume that the first four variables of the second row ( $X_{11} = X_{12} = X_{13} = X_{14}$ ) equal to one. Since these variables are part of constraints (21) and (23) too, it would imply that $Y_1 = Y_3 = 1$, which contradicts the first, new constraint. The same argument applies to any four $X_{k:s}$ at the same row, because they always appear together with two $Y_{t:}$s of the same constraint, like the above.

Assume instead that the first four variables of the first column ( $X_1 = X_{11} = X_{21} = X_{31}$ ) equal to one. Since these variables are part of constraints (21) and (26)[6] too, it would imply that $Y_1 = 1$, $Y_6 = 1$ and $Y_2 = Y_3 = Y_4 = Y_5 = 0$, and $Y_7 = Y_8 = Y_9 = Y_{10} = 0$. Therefore, the function-location matrix would be transformed to the following.

| | AB | AC | AD | AE | BC | BD | BE | CD | CE | DE |
|---|---|---|---|---|---|---|---|---|---|---|
| 1,2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | $X_8$ | $X_9$ | $X_{10}$ |
| 1,3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{18}$ | $X_{19}$ | $X_{20}$ |
| 1,4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{28}$ | $X_{29}$ | $X_{30}$ |
| 1,5 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{38}$ | $X_{39}$ | $X_{40}$ |
| 2,3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{48}$ | $X_{49}$ | $X_{50}$ |
| 2,4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{58}$ | $X_{59}$ | $X_{60}$ |
| 2,5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{68}$ | $X_{69}$ | $X_{70}$ |
| 3,4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{78}$ | $X_{79}$ | $X_{80}$ |
| 3,5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{88}$ | $X_{89}$ | $X_{90}$ |
| 4,5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $X_{98}$ | $X_{99}$ | $X_{100}$ |

**Table 1a: The transformed function-location matrix when $X_1 = X_{11} = X_{21} = X_{31} = 1$**

Constraints (31) to (35), (not shown in the text) together with $Y_{11} + Y_{12} + Y_{13} + Y_{14} + Y_{15} = 1$, imply that four variables from columns 8 and 9 will take the value 1 and all others the value 0. Let $X_{39} = X_{69} = X_{89} = X_{99} = 1$. Constraints (36) to (40), (not shown in the text) together with $Y_{16} + Y_{17} + Y_{18} + Y_{19} + Y_{20} = 1$, imply that four variables from the last column will take the value 1 and all others the value 0. Let $X_{30} = X_{60} = X_{80} = X_{100} = 1$. That is not possible because constraint (45) is violated since $X_{39}$ , $X_{69}$ , $X_{89}$ , $X_{99}$ and $X_{100}$ are part of that constraint too. By the same token, any other combination of four variables in the last column will contradict some constraint from (41) to (44). A similar argument applies for all other combinations of variables from columns 8 and 9. Thus, if the first twenty

[6] Not shown in the text; it is formulated as constraint (21), with the B columns instead of A.

constraints are substituted by these new five constraints, it is guaranteed that no more than one variable can lie on the same row or the same column.

It took more than four hours to solve this modified formulation. A slightly better formulation included five more binary constraints, to explicitly take care of the columns, i.e.:

$Y_1 + Y_6 + Y_{11} + Y_{16} + Y_{21} = 1$, $Y_2 + Y_7 + Y_{12} + Y_{17} + Y_{22} = 1$,

$Y_3 + Y_8 + Y_{13} + Y_{18} + Y_{23} = 1$, $Y_4 + Y_9 + Y_{14} + Y_{19} + Y_{24} = 1$,

$Y_5 + Y_{10} + Y_{15} + Y_{20} + Y_{25} = 1$

That formulation speeded up the execution time and solve it within two hours. Thus, it might be concluded that, sometimes, the best formulation of an integer LP might contain more constraints, if the execution time is to be limited.

## Appendix 3

The **optimal** and the *initial* solution for a rectangular 6x6 layout

| | AB | AC | AD | AE | AF | BC | BD | BE | BF | CD | CE | CF | DE | DF | EF |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1,2** | ***$x_1$*** | | | | | | | | | | | | | | |
| **1,3** | | | | **$X_{19}$** | | *$x_{21}$* | | | | | | | | | |
| **1,4** | | | | | **$X_{35}$** | | *$x_{37}$* | | | | | | | | |
| **1,5** | | | **$X_{48}$** | | | | | | *$x_{54}$* | | | | | | |
| **1,6** | | **$X_{62}$** | | | | | | *$x_{68}$* | | | | | | | |
| **2,3** | | *$x_{77}$* | | | | | | **$X_{83}$** | | | | | | | |
| **2,4** | | | *$x_{93}$* | | | | | | **$X_{99}$** | | | | | | |
| **2,5** | | | | | *$x_{110}$* | | **$X_{112}$** | | | | | | | | |
| **2,6** | | | | *$x_{124}$* | | **$X_{126}$** | | | | | | | | | |
| **3,4** | | | | | | | | | | *$x_{145}$* | | | | | **$X_{150}$** |
| **3,5** | | | | | | | | | | | | *$x_{162}$* | **$X_{163}$** | | |
| **3,6** | | | | | | | | | | | ***$x_{176}$*** | | | | |
| **4,5** | | | | | | | | | | | | | | ***$x_{194}$*** | |
| **4,6** | | | | | | | | | | | | **$X_{207}$** | *$x_{208}$* | | |
| **5,6** | | | | | | | | | | **$X_{220}$** | | | | | *$x_{225}$* |

*Initial layout min: 39840;  A:2, B:1, C:3, D:4, E:6, F:5,*

*i e.* $Y_2 = Y_7 = Y_{15} = Y_{22} = Y_{30} = Y_{35} = 1$

Optimal layout min: 35650; A:1, B:2, C:6, D:5, E:3, F:4,

i e. $Y_1 = Y_8 = Y_{18} = Y_{23} = Y_{27} = Y_{34} = 1$