

Electronic version of an article published as [Omega, 2008, vol. 36, No. 6, p. 1126-1132] [DOI: <http://dx.doi.org/10.1016/j.omega.2006.03.003>] © [copyright Elsevier]

Balancing assembly line with skilled and unskilled workers

Albert Corominas^a, Rafael Pastor and Joan Plans^a

^aIOC Research Institute—Universidad Politécnica de Cataluña, Av. Diagonal 647 (edif. ETSEIB), p.11, 08028
Barcelona, Spain

Abstract

In this paper, we present the process of rebalancing the line at motorcycle-assembly plant. The company found it necessary to rebalance its line, since it needs to increase production in the spring and summer months. The main characteristics of the problem are as follows: (i) the company hires temporary staff, who need more time to carry out their tasks than permanent workers; (ii) there must always be at least one skilled employee working alongside an unskilled one; and (iii) different task groups are incompatible with each other (clean-hands tasks and dirty-hands tasks). The goal is to minimise the number of temporary workers required, given a cycle time and the team of workers on staff. The problem is modelled as a binary linear program (BLP) and solved optimally by means of the ILOG CPLEX 9.0 optimiser. The solution provided, namely 12 permanent workers (skilled) and two temporary workers (unskilled), is an improvement on the solution implemented by the business, which involved 12 permanent workers and four temporary workers.

Keywords: Line balancing; Application; Manufacturing; Production

1. Introduction

Assembly lines are components of many production systems, such as those used in the automotive and household appliance industries. The problem of design and balancing of assembly lines is very difficult to solve optimally, given its combinatorial nature—it is NP-hard (see e.g. Wee and Magazine [1])—and the large number of tasks and restrictions found in industrial problems.

The balancing process consists of assigning the indivisible tasks to be carried out at workstations in such a way as to optimise an objective function (e.g. number of workstations, cycle time or unit cost of the product). In addition, assignment can be subject to various types of restrictions, including total work time at each station not being greater than a given upper bound (which is called cycle time), precedence and incompatibility relations between tasks.

The company is a business located in Spain that assembles small-engine motorcycles. It produces various models of motorcycles and the most complex model (in terms of the number of assembly operations) involves 138 tasks (the number of tasks to assign is reduced to 103 after applying a pre-processing phase, as explained in Section 3.1). The bulk of its production is for the Spanish market, where the yearly demand for motorcycles is highly seasonal: demand peaks are highly concentrated in the last months of spring and throughout the summer months. This heterogeneous demand means that production must be increased during those months up to one motorcycle every 217.37 s, which requires the rebalancing of the assembly line.

The most important characteristic of the line is that in order to increase production (reduce the cycle time to 217.37 s), temporary workers are hired. These temporary workers need more time than the “standard” (the time needed by a permanent worker) to complete their tasks; consequently, the task time depends on the type of worker performing the task. More specifically, a task requires the “standard” time if it is assigned to a skilled permanent worker, or the “standard” time multiplied by a factor greater than 1 if the task is assigned to an unskilled temporary worker; in the company this factor is equal to 2 and a temporary worker needs twice as much time as a permanent one to perform the same task.

There are two other characteristics that must also be taken into account in balancing the line. First, there is the need for each unskilled (temporary) worker to work alongside at least one skilled worker (who will be able to assist the temporary worker if necessary). In addition to this, there are incompatibility relations between groups of tasks; specifically, there are two dirty-hands and 18 clean-hands tasks: tasks number 1 and number 62 that handle oil and grease; and the other 18 tasks that handle the handlebars, the optical materials, the fairing, the footrest and the saddle.

The reduced precedence graph includes 103 tasks and 136 immediate precedence relations, and its order strength is equal to 0.67. The times of the tasks are not homogeneous: the average processing

time is equal to 26.99, but the maximum and the minimum processing times are equal to 169.18 and 2.04 s, respectively (see Table 1 for further details).

Table 1.

Data of the problem

Task	Processing time	Clean/dirty	Immediate precedence
01	31.02	Dirty	
02	12.08		01
03	4.44		02
04	6.28		03
05	28.58		04
06	7.12		
07	5.81		05
08	2.87		
09	2.04		08
10	3.34		09
11	2.68		10
12	9.28		11
13	6.32		12

Task	Processing time	Clean/dirty	Immediate precedence
14	10.28		13
15	60.15		14
16	4.29		15
17	12.99		16
18	23.29		17
19	8.63		18
20	17.87		19
21	8.21		20
22	2.28		21
23	70.25		22
24	40.90		17
25	12.98		33
26	15.02		05,23
27	5.54		26
28	27.76		27

Task	Processing time	Clean/dirty	Immediate precedence
29	41.07		28
30	26.81		29
31	14.59		05,06,07,23,24
32	4.46		31
33	10.81		32
34	4.52		38
35	20.99		33
36	9.55		33
37	8.98		36
38	78.17		33
39	40.21	Clean	33
40	38.12		39
41	21.89		25,40
42	2.46		40
43	13.00		34,40

Task	Processing time	Clean/dirty	Immediate precedence
44	5.34		35,40
45	23.47		36,40
46	11.41		37,40
47	11.68		40,60
48	18.50		40,51
49	4.09		40,52
50	4.70		33
51	57.06		30,33
52	61.94		59
53	21.18		52
54	24.63		41
55	19.27		41
56	5.32		55
57	28.49		43,44,45,46,47,48,49
58	19.18		40

Task	Processing time	Clean/dirty	Immediate precedence
59	57.03		51
60	115.35	Clean	53
61	14.13		57
62	22.09	Dirty	53
63	14.76		62
64	36.00		61
65	24.55		33
66	61.25		07
67	29.69		66
68	13.89		67
69	7.12		65
70	18.14	Clean	64
71	18.29	Clean	70
72	5.04	Clean	70
73	31.09	Clean	54,70

Task	Processing time	Clean/dirty	Immediate precedence
74	28.13	Clean	54
75	16.83	Clean	71
76	92.41	Clean	50,73,74
77	8.22	Clean	72
78	34.76		70
79	66.73	Clean	76
80	66.77	Clean	79
81	26.87		63
82	33.60		80
83	34.55		82
84	6.66		81
85	169.18		84
86	5.85		100
87	7.09		86
88	107.78	Clean	68,69,76,77,78

Task	Processing time	Clean/dirty	Immediate precedence
89	32.60	Clean	88
90	63.97	Clean	89
91	16.51		58,90
92	42.42		91
93	14.16		92
94	36.74		93
95	12.72		89
96	12.30		95
97	22.05		96
98	14.54		97
99	8.65		98
100	61.74	Clean	85
101	44.65	Clean	85
102	50.93	Clean	80
103	78.31		42,56,60,75,87,94,99,101,102

All workstations are equally equipped and when the number of workers (skilled and unskilled) is established, each worker, together with the tasks that have been assigned him or her and the necessary tools, is placed in a station. The goal is to minimise the number of temporary workers needed, given the production rate that is to be attained (or, equivalently, the cycle time that is to be maintained) and the number of permanent workers on staff (in the company, 217.37 s and 12 skilled workers, respectively).

The balancing carried out by the company for the required production rate calls for the 12 permanent (skilled) workers on staff and four (unskilled) temporary workers. The company calculated that it could also attain the desired level of productivity with 14 skilled workers.

2. Literature review

The problem of design and balancing of assembly lines has been examined extensively in the literature and a number of synthesis studies have been published, including Rekiek et al. [2], Becker and Scholl [3] and Scholl and Becker [4]. Both exact and heuristic procedures—and more recently, metaheuristic procedures—have been developed to solve this problem. Most papers assume that resources are homogeneous; hence, the duration of tasks does not depend on the stations to which they are assigned and any task can be carried out at any station. Working with heterogeneous resources, in terms of times and/or costs, involves solving a two-fold assignment problem: resources must be assigned to stations, while tasks are assigned simultaneously to those same stations. This case is often referred to as the assembly line design problem (ALDP).

Graves and Whitney [5] were among the first researchers to address the problem of selecting non-identical teams to minimise the sum of fixed and variable costs; however, as in Graves' subsequent works of 1983 and 1988, the precedence relationships of the tasks are sequential (i.e. they use a fixed assembly sequence). Faaland et al. [6] consider the assignment of alternative resources and tasks to stations; their proposed solution includes an exact procedure and two heuristic algorithms. Falkenauer [7] proposes a genetic algorithm that assigns tasks to stations, followed by a branch and bound (B&B) algorithm that selects the optimum resource for each station. Pinnoi and Wilhelm [8], who also deal with the ALDP, propose a branch and cut (B&C) procedure. McMullen and Frazier [9] present a simulated annealing for the multi-product problem, with stochastic task performance times, parallel stations and multiple objectives. Nicosia et al. [10] propose a dynamic programming procedure with several fathoming rules. Rekiek et al. [11] consider a multi-objective ALDP and set out a genetic algorithm, hybridised with a branch and cut algorithm and the multi-criteria decision-aid method PROMETHEE II. Bukchin and Rubinovitz [12] present the problem with the possibility of parallel stations; this work is an extension of Bukchin and Tzur [13], and the solution uses an

adaptation of the B&B procedure presented in that work. Levitin et al. [14] solve the problem on robotised assembly lines using a genetic algorithm that they compare with a truncated B&B.

As set out in Becker and School [3], the equipment selection problem is equivalent to a problem of selecting workers whose task performance speeds are different. In any event, in these works—see, for example, Akagi et al. [15], Wilson [16] and Lutz et al. [17]—it is normally assumed that the performance speed of all persons who are manufacturing the same task is equal and that the time necessary to finish a task depends on the number of workers assigned to a station (on a linear basis with the number of workers in some papers and on a non-linear basis in others); in other cases, it is a question of deciding which workers should work on a given shift and which should not.

Hopp et al. [18] set out a case in which workers can vary in speed and are benchmarked by defining the speed factor of each worker relative to a “standard worker”; moreover, they assume that a worker's speed factor applies uniformly across all tasks (similar to the industrial case examined here). However, it is based on a line that is already designed and balanced, and the goal is to minimise the number of changes of workers from stations with surplus capacity to stations with a heavier workload, to help out temporarily.

In real industrial circumstances, we find lines where some tasks cannot be performed at all the stations and where performance times depend on the worker performing the task. This situation occurs on the aforementioned motorcycle assembly line and on the assembly line of a factory producing plotters and printers [19].

3. Solving the problem

To solve the industrial problem described above, the data were processed to reduce the dimensions of the problem (Section 3.1); then, the problem was modelled as a binary linear program (BLP) to attain the optimal solution (Section 3.2). In Section 3.3 we present the results obtained.

3.1. Pre-processing

We applied pre-processing to reduce the dimensions of the problem. Specifically, there were sets of tasks that had to be performed at the same station: these tasks were grouped into a single aggregate task with a process time equal to the sum of the performance time for the tasks included in the set. In this way, we reduced the number of tasks from 138 to 103.

In addition, we applied the well-known concept of earliest and latest station, E_i and L_i , respectively, to which a task i can be assigned (before assigning a task we must assign the total time of the preceding tasks and, likewise, after assigning the task we need to assign the total time of the tasks following it). As a result, we obtain the range of stations $[E_i, L_i]$ to which we can assign task i .

3.2. Mixed linear programming model

In most published papers, the exact procedures for the assembly line balancing problem have mainly been dealt with from an academic perspective and the mathematical models are only used to formalise the problem. On the other hand, most real-life problems have been addressed by means of heuristic procedures. Nevertheless, during recent years, formidable progress has been made in terms of both computational power and computational technology, so we know that it would be possible to explore the potential of exact procedures to solve the problem.

To solve the problem of balancing lines with different categories of workers, we formulated and solved the BLP set out below. Its main characteristic is the fact that the task times depend on the type of worker performing the task.

In the case examined, some of the permanent workers W can perform all of the tasks, but there are not enough permanent workers to attain the production capacity needed in the summer months. Consequently, temporary workers are hired during these periods: the number of stations on the line is increased and production is therefore increased (and the cycle time is reduced).

The goal is to minimise the cost. The cost that needs to be addressed is exclusively the cost of the temporary workers, since the cost of the permanent workers is fixed. Given that the cost of the temporary workers is the same in each case, the goal is equivalent to minimising the number of temporary workers hired.

The permanent workers can perform all the tasks. However, the temporary (and unskilled) workers can only perform a subset of tasks, and they will take longer to perform these tasks than the permanent workers. The task performance time for temporary workers can be expressed as the product of the task performance time for a permanent worker and a coefficient $\beta (> 1)$, which is approximately equal for all the tasks that the temporary workers can perform. In addition, we must satisfy the restriction that alongside the station assigned to a temporary worker there must be at least one station assigned to a permanent (and skilled) worker.

Lastly, the incompatibility relations between clean-hands tasks and dirty-hands tasks must also be observed.

The resulting model is set out below.

Data:

i index of tasks.

j index of stations.

- W number of staff workers (permanent and skilled).
- T set of indivisible assembly tasks ($i=1, \dots, T$) (coincides with the tasks that can be performed by permanent workers).
- \hat{T} set of tasks that cannot be performed by temporary workers.
- d_i task time i when performed by a permanent worker ($i=1, \dots, T$).
- β coefficient (>1) by which d_i is multiplied to determine the task time i when performed by a temporary worker ($\forall i \in T \setminus \hat{T}$).
- P set of pairs of tasks (i, k) such that there is an immediate precedence relation between them.
- I set of pairs of tasks (i, k) such that there is specific incompatibility between them for assignment to the same station.
- C upper bound of cycle time.
- n, N lower and upper bounds, respectively, of the number of workstations ($j=1, \dots, N$).
- m upper bound of the number of tasks that can be assigned to a workstation.
- E_i, L_i earliest and latest stations, respectively, to which a task i ($i = 1, \dots, N$) may be assigned.
- \bar{T}_j set of tasks that can be performed at station j ($j = 1, \dots, N$).

Variables:

- $x_{ij} \in \{0, 1\} \quad \forall i, \forall j \in [E_i, L_i]$ where $x_{ij} = 1$
iff task i is assigned to station j .
- $y_j \in \{0, 1\} \quad j = 1, \dots, N$ where $y_j = 1$
iff station j is assigned to a temporary worker.
- $v_j \in \{0, 1\} \quad j = n + 1, \dots, N$ where $v_j = 1$
iff station j is vacant (has no tasks assigned and
is not assigned to any type of worker).

Model:

$$\min Z = \sum_{j=1}^N y_j, \quad (1)$$

$$\sum_{j=E_i}^{L_i} x_{ij} = 1 \quad \forall i \in T, \quad (2)$$

$$\sum_{i \in \tilde{T}_j} d_i \cdot x_{ij} \leq C - \frac{\beta - 1}{\beta} \cdot C \cdot y_j, \quad j = 1, \dots, n, \quad (3)$$

$$\sum_{i \in \tilde{T}_j} d_i \cdot x_{ij} \leq C \cdot (1 - v_j) - \frac{\beta - 1}{\beta} \cdot C \cdot y_j, \quad j = n + 1, \dots, N, \quad (3')$$

$$\sum_{i \in (\tilde{T} \cap \tilde{T}_j)} x_{ij} \leq m \cdot (1 - y_j), \quad j = 1, \dots, N, \quad (4)$$

$$y_j + v_j \leq 1, \quad j = n + 1, \dots, N, \quad (5)$$

$$v_j \leq v_{j+1}, \quad j = n + 1, \dots, N - 1, \quad (6)$$

$$\sum_{j=E_i}^{L_i} j \cdot x_{ij} \leq \sum_{j=E_k}^{L_k} j \cdot x_{kj} \quad \forall (i, k) \in P, \quad (7)$$

$$x_{ij} + x_{kj} \leq 1 \quad \forall (i, k) \in I; \quad \forall j \in [E_i, L_i] \cap [E_k, L_k], \quad (8)$$

$$y_1 + y_2 \leq 1, \quad (9)$$

$$y_j + y_{j+1} + y_{j+2} \leq 2, \quad j = 2, \dots, N - 3, \quad (9')$$

$$y_{N-1} + y_N \leq 1, \quad (9'')$$

$$W + \sum_{j=1}^N y_j + \sum_{n+1}^N v_j = N, \quad (10)$$

$$x_{ij}, y_j, v_j \in \{0, 1\}. \quad (11)$$

The objective function (1) minimises the number of temporary workers; (2) imposes that each task must be assigned to a single station; (3) and (3') express that the total working time at every workstation must not be greater than the upper bound of the cycle time C (if a skilled worker is assigned to the station) or than $\frac{1}{\beta} \cdot C$ (if an unskilled worker is assigned to the station); (4)

prevents tasks that can only be performed by permanent workers from being assigned to a station

assigned to a temporary worker; (5) does not allow a temporary worker to be assigned to an empty station; (6) ensures that if a station is empty, the following station is also empty, so non-empty stations are consecutive and it is assured that the used stations are numbered consecutively; (7) and (8) impose, respectively the conditions of precedence and incompatibility between pairs of tasks; (9), (9') and (9'') make it necessary for each temporary worker to be located alongside at least one permanent worker; (10) ensures that the number of permanent workers plus the number of temporary workers plus the number of empty stations is equal to the potential number of stations in the model; and (11) expresses the binary nature of the variables.

3.3. Results

The BLP formulated was solved using ILOG CPLEX 9.0 on a 1.8 MHz Pentium IV PC with 512 Mb RAM. For this task, we used the optimiser's default options, including an initial pre-processing stage. After pre-processing, the size of the problem to be solved was as follows: 365 restrictions, 666 variables and 2927 non-zero elements.

As previously stated, the balance implemented by the company for a given cycle time of 217.37 s requires 12 skilled workers (permanent staff) and the hiring of four temporary (and unskilled) workers. The optimiser obtains and guarantees the optimal solution in 78.36 s of calculation: 12 permanent workers and two temporary workers. The optimal solution attained is a clear improvement on the solution implemented by the business: the number of temporary workers to be hired is reduced by 2.

Fig. 1 shows the workload assigned to the 14 workstations of the solution; these are 187.75,106.36,107.24,217.14,217.35,217.15,217.34,215.25,216.87,213.35,216.54,214.93,216.19 and 216.87 s, respectively. Workstations 2 and 3 are assigned to temporary workers, with a real workload of 212.72 and 214.48 s, respectively; it can be seen that one permanent (and skilled) worker is working alongside each unskilled one. The obtained cycle time is equal to 217.35 s, so the initial cycle time is slightly improved and consequently the production rate is increased. Straightaway an optimal solution including only 13 skilled workers can be obtained by adding the load of workstations 2 and 3 (this optimal number of workers can be also obtained with the mathematical program, as is discussed subsequently). The workload is fairly assigned among the workers, which prevents industrial conflict. Finally, dirty-hands tasks 1 and 62 are assigned to workstations 1 and 6, whereas clean-hands tasks are assigned to stations 4,7,9,10,11,12 and 13.

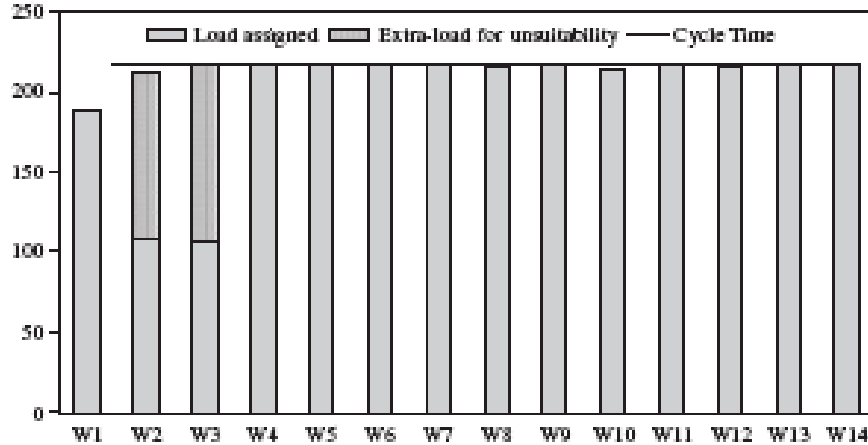


Fig. 1. Workload assigned to the 14 workstations.

In addition to obtaining an optimal solution for the problem, we also performed other tests. The company stated that, taking into account only skilled workers, it had a balance consisting of 14 workers. The mathematical programming model was executed with 13 skilled workers (it was already known that with 12 skilled workers, two temporary workers were needed), and an optimal solution with no temporary workers was found: the best solution found by the business was reduced by one skilled worker. We therefore demonstrated the potential usefulness of the model for solving cases in which there is only one type of worker.

Lastly, we tested other CPLEX search options in addition to the standard strategy that maintains a balance between the feasibility and optimality of the solution sought, with the following results: the optimum solution was obtained in 2.75 s when the focus of the search was the feasibility of the solution being sought, and in 73.17 s when the focus of the search was its optimality.

4. Conclusions

In this paper, we present a motorcycle assembly factory that has to rebalance its assembly line, since it needs to increase production in the summer months due to the seasonal nature of the demand for its products.

The main characteristic of the line is that production is increased through the hiring of temporary workers who take longer to perform tasks than permanent workers; consequently, the task time depends on the type of worker performing the task. Another characteristic of the process is the fact that a temporary (and unskilled) worker must always work alongside at least one permanent (and skilled) worker. Furthermore, there are incompatibility relations between clean-hands tasks and

dirty-hands tasks. The goal is to minimise the number of temporary workers required, given a cycle time and the team of workers on staff.

The problem is modelled and solved optimally by means of binary linear programming, and the solution provided—namely 12 permanent workers and two temporary workers—is a clear improvement on the solution implemented by the business, which involved 12 permanent workers and four temporary workers.

Acknowledgement

This research has been supported by the Spanish Ministry of Education and Science, project DPI2004-03472 co-financed by (FEDER).

References

- [1] T.S. Wee and M.J. Magazine, Assembly line balancing as generalized bin packing, *Operations Research Letters* 1 (1982), pp. 56–58.
- [2] B. Rekiek, A. Dolgui, A. Delchambre and A. Bratcu, State of art of optimization methods for assembly line design, *Annual Reviews in Control* 26 (2002), pp. 163–174.
- [3] C. Becker and A. Scholl, A survey on problems and methods in generalized assembly line balancing, *European Journal of Operational Research* 168 (2006), pp. 694–715.
- [4] A. Scholl and C. Becker, State-of-the-art exact and heuristic solution procedures for simple assembly line balancing, *European Journal of Operational Research* 168 (2006), pp. 666–693.
- [5] Graves SC, Withney DE. A mathematical programming procedure for equipment selection and system evaluation in programmable assembly. Proceedings of the IEEE decision and control, 1979. p. 531–6.
- [6] B.H. Faaland, T.D. Klastorin, T.G. Schmitt and A. Shtub, Assembly line balancing with resource dependent task times, *Decision Sciences* 23 (1992), pp. 343–364.
- [7] Falkenauer E. A grouping genetic algorithm for line balancing with resource dependent task times. Proceedings of the fourth international conference on neural information processing, 1997. p. 464–8.
- [8] A. Pinnoi and W.E. Wilhelm, Assembly system design: a branch and cut approach, *Management Science* 44 (1998), pp. 103–118.
- [9] P.R. McMullen and G.V. Frazier, Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel stations, *International Journal of Production Research* 36 (1998), pp. 2717–2741.
- [10] G. Nicosia, D. Pacciarelli and A. Pacifici, Optimally balancing assembly lines with different workstations, *Discrete Applied Mathematics* 118 (2002), pp. 99–113.
- [11] B. Rekiek, P. de Lit and A. Delchambre, Hybrid assembly line design and user's preferences, *International Journal of Production Research* 40 (2002), pp. 1095–1111.
- [12] J. Bukchin and J. Rubinovitz, A weighted approach for assembly line design with station paralleling and equipment selection, *IIE Transactions* 35 (2003), pp. 73–85.
- [13] J. Bukchin and M. Tzur, Design of flexible assembly line to minimize equipment cost, *IIE Transactions* 32 (2000), pp. 585–598.
- [14] Levitin G, Rubinovitz J, Shnits B. A genetic algorithm for robotic assembly line balancing. *European Journal of Operational Research* 2004, 2006; 168: 811–250.

- [15] F. Akagi, H. Osaki and S. Kikuchi, A method for assembly line balancing with more than one worker in each station, *International Journal of Production Research* 21 (1983), pp. 755–770.
- [16] J.M. Wilson, Formulation of a problem involving assembly lines with multiple manning of work stations, *International Journal of Production Research* 24 (1986), pp. 59–63.
- [17] C.M. Lutz, K.R. Davis and C.F. Turner, Development of operator assignment schedules: a DSS approach, *Omega* 22 (1994), pp. 57–67.
- [18] W.J. Hopp, E. Tekin and M.P. Van Oyen, Benefits of skill chaining in serial production lines with cross- trained workers, *Management Science* 50 (2004), pp. 83–98.
- [19] Pomes J. Equilibrat de línies de muntatge en una empresa del sector informàtic. Proyecto Final de Carrera, UPC, Barcelona, 2001.