Management and Production Engineering Review
Volume 12 • Number 4 • December 2021 • pp. 38–44
DOI: 10.24425/mper.2021.139993

Optimization of Aggregate Production Planning Problems with and without Productivity Loss using Python Pulp Package

Hakeem Ur REHMAN[1], Ayyaz AHMAD[1], Zarak ALI[1], Sajjad Ahmad BAIG*[2], Umair MANZOOR[2]

[1] *Institute of Quality & Technology Management, University of the Punjab, Lahore, Pakistan*
[2] *Faisalabad Business School, National Textile University, Faisalabad, Pakistan*

Received: 10 October 2020
Accepted: 26 October 2021

Abstract
Traditionally the aggregate production plan helps in determining the inventory, production, and work-force, based on the demand forecasts without considering the productivity loss at a tactical level in supply chain planning. In this paper, we include the productivity loss into traditional aggregate production plan and the prescriptive analytics technique, linear programming, is used to solve this problem of practical interest in the domain of multifarious businesses and industries. In this study, we discussed two model variations of the aggregate production planning problem with and without productivity loss, i) fixed work-force, and ii) variable Work Force. The mathematical models were designated to be solved by using an open-source python pulp package in order to evaluate the impacts of the productivity loss on both the models. PuLP is an open-source modeling framework provided by the COIN-OR Foundation (Computational Infrastructure for Operations Research) for linear and integer Programing problems written in Python. The computational results indicate that the productivity loss has direct impact on the workforce hiring and firing.

Keywords
Aggregate Production Planning, Productivity, Python PuLP, Optimization.

Background and Introduction

Aggregate Production Planning (APP) is a mid-term capacity planning problem that is used to determine inventory, production, and work-force levels to meet the changing requirements for a planning period, comprising approximately six months to twelve months (Cheraghalikhani et al., 2019; Demirel et al., 2018). Usually, the planning period incorporates the next demand at the peak of the season. The entire planning period is further divided into a small period. For example, a one-year planning period can be divided into periods of one-half year and two quarter year. Commonly, physical resources of the company are supposed to be fixed during the whole planning period of interest, fulfilling the external requirements of the planning; the efforts are directed toward the best utilization of these resources (Shi & Peng, 2001; Souza, 2014). It is very difficult to assess every phase of the production process while maintaining a long planning period, it is important to aggregate all the information which has to be processed. The APP technique is implemented to determine a unit of aggregate production, like the average item, or in terms of production time, volume, weight, or dollar value (Jayakumar, 2017). Plans are dependent on aggregate demand for one or more aggregate of items. Once the aggregate production plan is made, the constraints are applied to the production scheduling process thoroughly to determine the specific quantity of each item which should be produced accordingly (Shi & Peng, 2001). Another fundamental phase is to evaluate how much the plan is profitable or not to give maximum yield and if there is any productivity loss how we would be able to tackle this problem by reviewing every aspect of the implemented plan (Piper & Vachon, 2001). This evaluative phase is very significant but the most people are not interested to resolve this issue of loss of productivity (Filho et. al., 2010).

It is very difficult to discuss all research papers, relevant with APP Problems but we focused on to sum up the reviews of the research papers which are published in the last decade (Cheraghalikhani et al., 2019). Silva Filho et. al., (2010) introduced a managerial decision support system to deal with

***Corresponding author**: Sajjad Ahmad Baig – National Textile University, Faisalabad, Pakistan, phone: (+92) 300 478 1314, e-mail: Sajjad.baig@hotmail.com*

APP, based on an Excel spread sheet. Chaturvedi & Bandyopadhyay (2015) launched an automated targeted model to develop the resource conservation network that is used in APP and energy supply chain. Gholamian et al. (2015) developed a model that is used to highlight the significant aspects of supply chain production planning, approved by industries. Jamalnia and Soukhakian (2009) reported a complex multi-objective non-linear programming model that was developed for achievement with different priorities by resolving APP problem in a complex environment (Piper & Vachon, 2001). Some researchers e.g. (Gilgeous, 1989) criticized the limitation of the existing methods for solving the APP problem. Stockton and Quinn (1995) who analyze these limitations by conducting a literature review. Mathematical models are the abstraction of real-world problems which should express the decision maker's needs; produce readable and understandable results (VoB et. al., 2006). The most frequently used important techniques for APP are i) Trial and error methods ii) Graphical techniques iii) Parametric production planning iv) Production switching heuristic v) Linear programming vi) Goal programming vii) Mixed-integer programming viii) Transportation method ix) Simulation models. The mathematical programming theoretician take a great interest in these problems due to the simplicity of the problem statement and complicated solutions (see e.g., (Cheraghalikhani et al., 2019). The APP problem with and without productivity loss indicates the basic problems in the planning as if there is α% productivity loss, pointing out the erroneous planning which brings about α% loss for the company. There is a need to evaluate all these problems to find out the basic reasons of the productivity loss of α% and it is very advantageous to identify the faulty line of processes which can be improved to reduce the productivity loss to a minimum level (Biazzi, 2018). This study covers a variety of APP model' characteristics that include modeling structure, solving approaches and important issues, as compared to other literature reviews in this filed which had focused on APP methodologies but mainly, we focused on the productivity loss, caused by APP problem with fixed and variable workforce (Khoshnevis et al,. 1982). Based on the previous works, we presented the model for APP problem, related to productivity loss with the help of linear programming and mixed-integer programming as compared to the model of productivity loss with the help of non-linear programming which was considered a complicated model (Jayakumar et. al., 2017). In this research paper, we have set two main objectives, firstly, to incorporate the productivity loss in APP problem and make the mathematical models and secondly, resolve this problem by using open-source python pulp package. In order to achieve these objectives, both models of fixed work force and variable workforce for Aggregate Production Planning problem with and without productivity loss are discussed in this research paper. In the fixed workforce model, hiring and firing of the labor are disallowed and production rates can fluctuate only by using over time; whereas, in the variable workforce model, hiring and firing of the labor are allowed for using overtime from the regular workforce, moreover, backorders are allowed.

Problem Description and Mathematical model

The growing competition in the global market force the manufacturing companies to manage their processes effectively and efficiently. One of the critical success factors for achieving this is the better

Aggregate Production Plan that may help the companies to compete in the market successfully. The optimal Aggregate Production Plan for the company to minimize the cost of inventory, labour (regular time and overtime) and produce the desired quantity of products to gain higher ratio of profitability with and without productivity loss by handling with all kinds of expected problems during the production processes problems.

The two variations of the problem studied are as follows:

i. Assume hiring and firing are not allowed (i.e., fixed workforce problem). Backorders are not allowed.
ii. Assume hiring and firing are allowed (i.e., variable workforce problem). Backorders are allowed.

Both the problem variations are dealt with and without productivity loss to check its impact on the cost.

The following notations are used for the describing both the linear programming models.

Notations:

c_t – Unit production cost in period t (exclude labor costs)

h_t – Inventory carrying cost per unit held in stock for the period t

π_t – Backorder cost per unit for the period t

r_t – Cost per manhour in period t of the regular labor

o_t – Cost per man-hour in period t of overtime labor

d_t – Forecasted demand in period t
h_t – Cost of hiring one manhour in period t
f_t – Cost of firing one manhour in period t
m – Required manhours to produce a product
$\bar{R}_t$ – Available total manhours in period t of regular labour
$\bar{O}_t$ – Available total manhours in period t of overtime labor
I_0 – Initial Inventory level of the product
P – Fraction of regular hours allowed as overtime
T – Time horizon in periods
α – Productivity Loss

Model A: Fixed work force model

Decision Variables:
X_t – Production quantity in period t
I_t – Inventory in period t
R_t – Manhours of regular labor used during period t
O_t – Manhours of overtime labour used during period t

The objective (1) is to minimize the production, inventory holding and labour (i.e., regular and overtime man-hours) costs.

$$\sum_{t=1}^{T} \left[c_t X_t + h_t I_t + r_t R_t + o_t O_t\right]. \quad (1)$$

The well known Inventory-Balancing constraint (2) make sure that period t demand must be fullfil. In otherwords, sum of period $(t-1)$ inventory and production quantity must be equal to leftover inventory and demand of period t.

$$X_t + I_{t-1} - I_t = d_t \quad \forall t. \quad (2)$$

The constraint (3) make sure that total production time of the product must be equal to the available regular and over time in periods t (i.e., capacity of the period t).

$$mX_t - (R_t + O_t) = 0 \quad \forall t. \quad (3)$$

Using the constraints (4) and (5), upper bounds on regular and over production time is applied.

$$0 \le R_t \le \bar{R}_t \quad \forall t, \quad (4)$$
$$0 \le O_t \le \bar{O}_t \quad \forall t. \quad (5)$$

Model B: Variable work force model

Decision Variables:
X_t – Production quantity in period t
I_t^+ – Inventory in period t
I_t^- – Back ordered quantity in period t
I_t – Inventory in period t
R_t – Manhours of regular labor used during period t
O_t – Manhours of overtime labor used during period t
H_t – Manhours of regular workforce hired in period t
F_t – Manhours of regular work force fired in period t

The objective (6) is to minimize the total production, inventory holding, backorder, labour (i.e., regular and overtime man-hours), hiring and firing costs.

$$\sum_{t=1}^{T} \Big[c_t X_t + h_t I_t^+ + \pi_t I_t^- + r_t R_t + \\ + o_t O_t + h_t H_t + f_t F_t\Big]. \quad (6)$$

The Inventory-Balancing with backorders constraint (7) make sure that period t demand must be fullfil.

$$X_{it} + I_{i^+,t-1} - I_{it^+} - I_{i^-,t-1} + I_{it^-} = d_{it} \quad \forall i, t. \quad (7)$$

Constraint (8) is same as constraint (3) for periods t.

$$mX_t - (R_t + O_t) = 0 \quad \forall t. \quad (8)$$

Constraint (9) makes sure the availability of the required regular time enough to produce the product. Whereas constraint (10) applies the upper bound on the overtime production.

$$R_t - R_{t-1} - H_t + F_t = 0 \quad \forall t, \quad (9)$$
$$O_t - pR_t \le 0 \quad \forall t. \quad (10)$$

Model A and B with productivity loss

The APP problem with productivity loss is similar to the problems which are mentioned above but these problems are also different from the above-mentioned problems which are presented without productivity loss while these are demonstrated with productivity loss. In both model's problem the one big difference is in the 2^{nd} constraint (Time Required to produce products), if the productivity loss is 10% then in the 2^{nd} constraint $(mX_t - (1-\alpha)(R_t + O_t)) = 0)$ the value of $(1-\alpha)$ is 0.90. It is because we consider the 10% regular time and overtime labor hour is giving loss to the company. Similarly, constraint changes for 20% to 50% with the α value 0.80 to 0.50.

Computational Study

We use the numerical data as shown below (Table 1) in the table to conduct the computational experiment by using python pulp package (see Appendices A and B). We perform the computational work on HP EliteBook Workstation on Windows 10 Using anaconda navigator: python 3.0 with 8 Gb RAM and 2.70 GHz processor.

Table 1
Data set

	Jan	Feb	Mar	Apr	May	Jun
Demand	110	110	120	210	160	90
Unit production cost	8	7	7	9	6	9
Unit holding cost	2	5	5	3	4	3
Unit regular labor cost	18	17	19	17	15	17
Unit overtime labor cost	23.5	24.5	28	28	23.5	23.5
Available man hour R labour	130	140	150	160	110	110
Available man hour O labour	40	50	40	40	40	40
Unit backorder cost	15	20	25	30	25	15
Hiring cost	22	22	22	22	22	22
Firing cost	20	20	20	20	20	15

Results and discussion: Model A & B with and without productivity loss

Table 2 shows the total minimized cost without and with 10% to 50% productivity loss for both the models. The results indicate that the loss of productivity has negative impact on the total cost. In other words, as the productivity loss increases the total cost of the aggregate production plan also increases. For example, the total aggregate production plan cost without productivity loss is 20456.0 for model-A, that is less than the value of production with 50% productivity loss that is 41651.0. Similar results produced by the model-B; the total aggregate production plan cost without productivity loss is 24382.0 that is less than the value of production with 50% productivity loss that is 41573.

Table 2
Model A & B Total Cost with and without productivity loss

Without Productivity Loss	With Productivity Loss				
	10%	20%	30%	40%	50%
Model – A: Fixed work force model					
20486.0	23572.4	25749.5	29661.1	34667.6	41651.0
Model – B: Variable work force model					
24382.0	26345.1	28799.0	31900.0	35945	41573

The comparative analysis of both models with and without productivity loss are discussed here with the help of figures which are displayed below on the basis of cost. As mentioned earlier that the key decisions in aggregate production plan are production quantity, inventory and workforce. The Figure 1 shows that there is no significant impact on inventory cost due to productivity loss in model-B. Because model-B considered variable workforce but model-A not, therefore the impact of the productivity loss can be observed on inventory cost while considering model-A. So, the impact of productivity loss can be handled while having large inventory in fixed workforce condition (model-A) compare to variable workforce (model-B) which results high inventory cost.

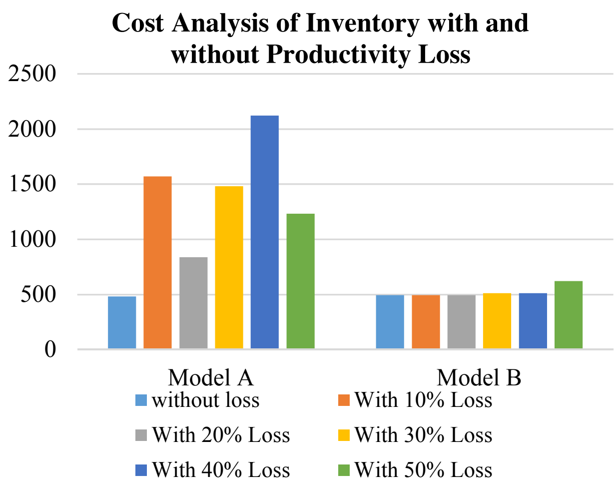

Fig. 1. Cost analysis of inventory with and without productivity loss

Whereas Figure 2 shows that the cost of hiring start increasing as the productivity loss increases. Because productivity loss has direct impact on the production capacity which can be handled either by improving the quality of the production process or through hiring more workforce in order to meet the customer demand. In other words, the loss of productivity has direct impact on quantity produced by the manufacturer.

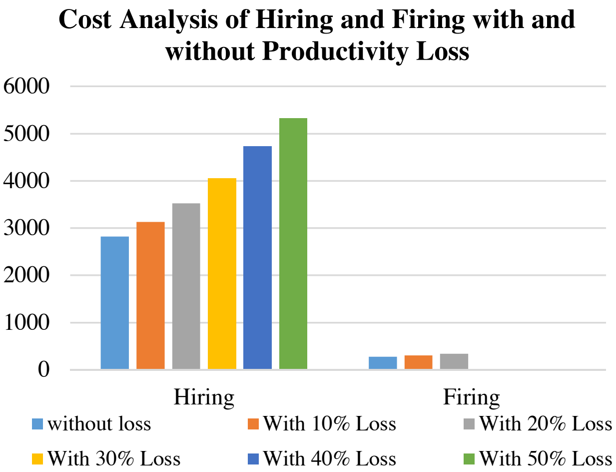

Fig. 2. Cost Analysis of Hiring and Firing with and without productivity loss

Conclusions

On the basis of the results of the computational experiments and logical discussions in this research-oriented study that the evaluation of the productivity is indispensable for demonstrating the aggregate production plan which helps the production units of the firms in indicating the particular reasons or fault which is causing the high APP total cost. The APP models with and without loss of productivity are the significant tools to identify such problems and reduce the impact of the loss of productivity at the place of the production unit. In this way, the APP models would be helpful to minimize the cost of production which is increased with the loss of productivity.

Appendices

(A)

Python pulp package: Codes

```
# Import Library
    from pulp import *
# Lsit (TimePeriods from january to june)
    t = [0,1,2,3,4,5,6]
# Assigning the variable
    Xt = LpVariable.dicts("Quantity Produced", t, 0)
    It = LpVariable.dicts("Inventory", t, 0)
    Rt = LpVariable.dicts("R_Labor Used", t, 0)
    Ot = LpVariable.dicts("O_Labor Used", t, 0)
# Parameters and Data
    demand = {1:110, 2:110, 3:120, 4:210, 5:160, 6:90}
    # Demand data
    UPC = {1:8, 2:7, 3:7, 4:9, 5:6, 6:9}
    # Unit Production Cost (Excluding Labor)
    UHC = {1:2, 2:5, 3:5, 4:3, 5:4, 6:3}
    # Unit Holding Cost
    URLC = {1:18, 2:17, 3:19, 4:17, 5:15, 6:17}
    # Unit Regular Labor Cost
    UOLC = {1:23.5, 2:24.5, 3:28, 4:28, 5:23.5, 6:23.5}
    # Unit Overtime Labor Cost
    R_MH = {1:130, 2:140, 3:150, 4:160, 5:110, 6:110}
    # Available Man-hours R
    (Regular time) Labor
    O_MH = {1:40, 2:50, 3:50, 4:40, 5:40, 6:40}
    # Available Man-hours O (Overtime) Labor
# Setting the Problem
    prob = LpProblem("Aggregate Production
    Planning: Fixed Work Force Model", LpMinimize)
# Create the LP object, set up as a
MINIMIZATION problem
    prob + = lpSum(UPC[i]*Xt[i] for i in t[1:])
    +lpSum(UHC[i]*It[i] for i in t[1:])
    +lpSum(URLC[i]*Rt[i] for i in t[1:])
    +lpSum(UOLC[i]*Ot[i] for i in t[1:])
# Constraints
    It[0] = 4 #Inventory in Dec
for i in t[1:]:
    prob + = (Xt[i] + It[i−1] − It[i]) = demand[i]
    # Inventory-Balancing Constraints
for i in t[1:]:
    prob + = Xt[i] − 1*(Rt[i] + Ot[i]) = 0
    # Time Required to produce products
for i in t[1:]:
    prob + = Rt[i] <= R_MH[i]
    # Regular Time Required
for i in t[1:]:
    prob + = Ot[i] <= O_MH[i]
    # Over Time Required
# View the model
    print(prob)
# Solve the model:
    prob.solve()
    print("Solution Status =", LpStatus[prob.status])
# Print the solution of the Decision Variables
for v in prob.variables():
    if v.varValue>0:
    print(v.name, "=", v.varValue)
# Print Optimal value
    print("Total Production Plan Cost = ",
    value(prob.objective))
```

(B)

```
# Import Library
    from pulp import *
# Lsit (TimePeriods)
    t = [0,1,2,3,4,5,6]
# Parameters and Data
    demand = {1:110, 2:110, 3:120, 4:210,
             5:160, 6:110}
    # Demand data
    UPC = {1:8, 2:7, 3:7, 4:9, 5:6, 6:9}
    # Unit Production Cost (Excluding Labor)
    UHC = {1:2, 2:5, 3:5, 4:3, 5:4, 6:3}
    # Unit Holding Cost
    UBC = {1:15, 2:20, 3:25, 4:30, 5:25, 6:15}
    # Unit Back order cost
    URLC = {1:18, 2:17, 3:19, 4:17, 5:15, 6:17}
    # Unit Regular Labor Cost
    UOLC = {1:23.5, 2:24.5, 3:28, 4:28, 5:23.5, 6:23.5}
    # Unit Overtime Labor Cost
    HC = {1:22, 2:22, 3:22, 4:22, 5:22, 6:22}
    # hiring cost
    FC = {1:20, 2:20, 3:20, 4:20, 5:20, 6:15}
    #firing cost
# Setting the Problem
    prob = LpProblem("Aggregate Production
    Planning: Variable Work Force Model",
    LpMinimize)
# Decision Variables
    Xt = LpVariable.dicts("Quantity Produced",
    t, 0, None, LpInteger)
    It = LpVariable.dicts("Inventory", t, 0)
    Bt = LpVariable.dicts("Backorder", t, 0)
    Rt = LpVariable.dicts("R_Labor Used", t, 0)
    Ot = LpVariable.dicts("O_Labor Used", t, 0)
    Ht = LpVariable.dicts("Labours Hired", t, 0)
    Ft = LpVariable.dicts("Labours Fired", t, 0)
# Objective Function
    prob += lpSum(UPC[i]*Xt[i]
    for i in t[1:]) + lpSum(UHC[i]*It[i]
    for i in t[1:]) + lpSum(UBC[i]*Bt[i]
    for i in t[1:]) + lpSum(URLC[i]*Rt[i]
    for i in t[1:]) + lpSum(UOLC[i]*Ot[i]
    for i in t[1:]) + lpSum(HC[i]*Ht[i]
    for i in t[1:]) + lpSum(FC[i]*Ft[i]
    for i in t[1:])
# Constraints
    It[0] = 4
    Rt[0] = 0
    Bt[0] = 0
for i in t[1:]:
    prob += (Xt[i] + It[i−1] − It[i] −
    Bt[i−1] + Bt[i]) = demand[i]
    # Inventory-Balancing Constraints
for i in t[1:]:
    prob += Xt[i] − 1*(Rt[i] + Ot[i])
    <= 0
    # Time Required to produce products
for i in t[1:]:
    prob += Rt[i] − Rt[i−1] − Ht[i] + Ft[i] = 0
    # Regular Time Required
    for i in t[1:]:prob += (Ot[i] − 0.25*Rt[i]) <= 0
    # Regular Time Required
# Solving the problem
    prob.solve()
    print("Solution Status =", LpStatus[prob.status])
# Print the solution of the Decision Variables
    for v in prob.variables():
    if v.varValue>0:
    print(v.name, "=", v.varValue)
# Print Optimal solution
    print("Total Production Plan Cost = ",
    value(prob.objective))
```

References

Biazzi J.L. (2018), Aggregate planning for probabilistic demand with internal and external storage, *Journal of Operations and Supply Chain*, vol. 11, no. 1, pp. 37–52.

Chaturvedi N.D. and Bandyopadhyay S. (2015), Targeting Aggregate Production Planning for an Energy Supply Chain, Industrial & Engineering Chemistry Research, vol. 54, no. 27, pp. 6941–6949. DOI: 10.1021/acs.iecr.5b00587.

Cheraghalikhani A., Khoshalhan F. and Mokhtari H. (2019), Aggregate production planning: A literature review and future research directions, *International Journal of Industrial Engineering Computations*, vol. 10, no. 2, pp. 309–330.

Demirel E., Özelkan E.C. and Lim C. (2018), Aggregate planning with Flexibility Requirements Profile, *International Journal of Production Economics*, vol. 202, no. May, pp. 45–58. DOI: 10.1016/j.ijpe.2018.05.001.

Filho O.S.S., Cezarino W. and Ratto J. (2010), Aggregate production planning: Modeling and solution via Excel spreadsheet and solver, IFAC Proceedings Volumes, vol. 43, no. 17.

Gholamian N., Mahdavi I., Tavakkoli-Moghaddam R. and Mahdavi-Amiri N. (2015), Comprehensive fuzzy multi-objective multi-product multi-site aggregate production planning decisions in a supply chain under uncertainty, *Applied Soft Computing*, vol. 37, pp. 585–607. DOI: 10.1016/j.asoc.2015.08.041.

Gilgeous V., Modelling realism in aggregate planning: A goal-search approach, *International Journal of Production Research*, vol. 27, no. 7, pp. 1179–1193, 1989. DOI: 10.1080/00207548908942616.

Jamalnia A. and Soukhakian M.A. (2009), A hybrid fuzzy goal programming approach with different goal priorities to aggregate production planning, Computers & Industrial Engineering, vol. 56, no. 4, pp. 1474–1486. DOI: 10.1016/j.cie.2008.09.010.

Jayakumar A. (2017), Optimization of Fixed Charge Problem in Python using PuLP Package, *International Journal of Control Theory and Applications*, vol. 10, no. 02, pp. 443–447.

Jayakumar A., Krishnaraj C. and Nachimuthu A.K. (2017), Aggregate production planning: Mixed strategy, *Pakistan Journal Biotechnology*, vol. 14, no. 3, pp. 487–490.

Khoshnevis B., Wolfe P.M. and Terrell M.P. (1982), Aggregate planning models incorporating productivity – an overview, *International Journal of Production Research*, vol. 20, no. 5, pp. 555–564.

Piper C.J. and Vachon S. (2001), Accounting for productivity losses in aggregate planning, *International Journal of Production Research*, vol. 39, no. 17, pp. 4001–4012.

Shi Y. and Peng Y. (2001), *Multiple Criteria and Multiple Constraint Levels Linear Programming: Concepts, Techniques and Applications*. World Scientific.

Souza G.C. (2014), Supply chain analytics, *Business Horizons*, vol. 57, no. 5, pp. 595–605. DOI: 10.1016/j.bushor.2014.06.004.

Stockton D.J. and Quinn L. (1995), Aggregate production planning using genetic algorithms, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 209, no. 3, pp. 201–209. DOI: 10.1243/PIME_PROC_1995_209_074_02.

VoB S. and Woodruff D.L. (2006), *Introduction to Computational Optimization Models for Production Planning in a Supply Chain*, 2nd ed. Springer-Verlag Berlin Heidelberg.