Journal of Cybernetics and Informatics

published by

Slovak Society for
Cybernetics and Informatics

Volume 6, 2006

http://www.sski.sk/casopis/index.php (home page)

ISSN: 1336-4774

ANT COLONY OPTIMIZATION FOR SERIES-PARALLEL CONTINUOUS PRODUCTION SYSTEMS WITH BUFFERS UNDER RELIABILITY CONSTRAINTS

A. Allali, A.Zeblah Y. Massim, A. Ghoraf and M. Rahli

Oran University of Technology, Faculty of Electrical Engineering
El M'naoura, 15050 Oran, Algeria
e-mail:alalia@yahoo.fr

Abstract

This paper uses an ant system (AS) meta-heuristic optimization method to solve the problem of structure optimization of series-parallel production systems. In the considered problem, redundant machines (elements) and buffers in process are included in order to attain a desirable level of reliability. A procedure which determines the minimal cost system configuration is proposed. In this procedure, multiple choices of producing machines and buffers are allowed from a list of product available in the market. The elements of the system are characterized by their cost, estimated average up and down times, productivity rates and buffers capacities. The reliability is defined as the ability to satisfy the consumer demand which is represented as a piecewise cumulative load curve. The proposed meta-heuristic is used as an optimization technique to seek for the optimal design configuration. The advantage of the proposed AS approach is that allows machines and buffers with different parameters to be allocated.

Keywords: Manufacturing system, Buffers, Meta-heuristic Optimization, Ant algorithm, Reliability evaluation, Optimal design

1 INTRODUCTION

The design of manufacturing systems is of great economic importance, because of the need to face requirements for increasing productivity and reducing cost. One of the main important problems in the design of manufacturing systems is structure optimization, and the natural objective function of this problem is cost minimization subject to requirement of meeting the demand with the desired system reliability level.

In manufacturing process, production chain is prone to productivity fluctuation. As the mean to smooth out this variation and to increase system reliability, a temporary storage between the stages of manufacturing process is often introduced. In this case the determination of the buffers size and structure of producing machines affect principally the total cost of system and its availability.

Indeed, a great part of analytical solutions for the structure optimization problem have been addressed in many studies some of which can be found in [1-3]. It is interesting to note that some investigations found solutions of optimal size of the buffers storage and the optimal stock to minimize the total inventory cost. In the example of a queuing system an approximate solution of a buffers design problem is presented [3] to determine the smallest buffer capacity. This case does not consider a reliability and structure upstream and downstream components of the system. On the contrary, in [4-5] the reliability constraints are considered and the design of zero-buffer production system is examined. The objective is to minimize the total cost of machines while providing the required level of reliability. Algorithms presented in these works are of great help to designers of production systems using machines with different productivity, reliability and cost.

In this paper, an optimal production system design problem is considered, and an AS algorithm is proposed. In our case the system studied includes process intermediate buffers and has series-parallel configuration. The AS is used to find the optimal design system by choosing the appropriate technology from a list of available products in market. Several technologies belong to a variety of available products. Each technology is characterized by its cost, productivity and estimated average up and down times. Our objective is to select the optimal combination of machines used in parallel and intermediate buffers for all components. This has to correspond to the minimal total cost with regard to the selected level of the system reliability. The AS algorithm is inspired from nature like other meta-heuristic ones, e.g.: simulated annealing, genetic algorithm, evolutionary strategy, and tabu search. The AS allows each component and intermediate buffer to contain elements with different technologies. In this work, to evaluate the reliability for production system design, times to failures and repairs are considered as exponentially distributed. The following assumptions are considered in this study:

1. We consider the probability of simultaneous unavailability of machines is negligible.

2. We assume that the convoying time between machines is negligible compared with the other characteristic time scales.

3. All buffers are full when they have to be used to compensate the system productivity deficit.

Organization of the remaining part of this paper is as follows. Section 2 of the paper consists of a general description of the model used and a formulation of the problem. In section 3, we describe the computation method of system unsatisfied demand probability. Section 4 describes the basic AS approach and its adaptation to the problem. In section 5, an illustrative example is represented. Conclusions are drawn in section 6.

2 DESCRIPTION OF SYSTEM MODEL AND PROBLEM FORMULATION

A system design considered in our work, contain N components connected in series arrangement as shown in Figure 1.

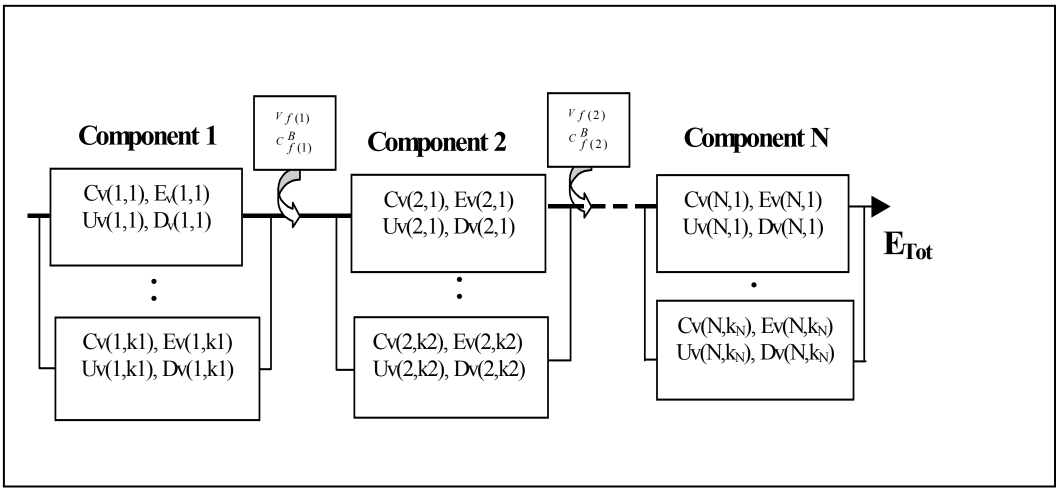

Figure 1: Series-Parallel Manufacturing Systems with Buffers

Each component of type $i = 1, ..., N$ contains a number of producing machines belonging to different technologies put in parallel. For some technical constraints, each component of type i contains no more than $K_{\max}$ machines connected in parallel. A multi-choice of machines and technologies will be adopted for each given system component. Each technology available in market has different costs, productivities and estimated average up and down times according to their technology. A vector of parameters C_{v_i}, E_{v_i}, U_{v_i}, D_{v_i} can be specified for each technology $v(i, j)$ of the component of type i. The structure of system component i is defined by the version numbers $v(i, j)$ of parallel elements for each component. These numbers can vary in the range $0 \le v(i, j) \le V_i$ where V_i is the total number of technologies available for element of type i and the number of parallel machines is k_i $1 \le k_i \le K_{\max}$, where $K_{\max}$ is the maximum allowed number of parallel machines of type i.

Each component m, $1 \le m \le N-1$ can also contain a buffer chosen from the list of available buffers. Note that the buffers of technology f differ by their capacity V_f and cost C_f^B. A vector $f = \{f(m)\}$, where $0 \le f(m) \le F_m$ defines technologies of buffers chosen for each component. Here F_m is the total number of different buffers technologies available for m^{th} component. $f(m) = 0$ means that no buffer is installed at component m. For a given set of vectors $f, v_1, v_2, \ldots, v_n$ the total cost of the system can be calculated as:

$$C = \sum_{m=1}^{N-1} C^B_{f(m)} + \sum_{i=1}^{N} \sum_{j=1}^{K_i} C_{v(i,j)} \tag{1}$$

With the above assumption 1, the entire system reliability estimation can be based on consideration of unavailability of single machines. Therefore, the different system states that contribute to the reduction of the total productivity of a system are equal to the total number of machines in system. All the possible states can be defined as:

$$S = \sum_{i=1}^{N} \sum_{j=1}^{K_i} \alpha_{ij} \tag{2}$$

where

$$\alpha_{ij} = \begin{cases} 1, & v(i,j) \ne 0 \\ 0, & v(i,j) = 0 \end{cases}$$

Each state corresponds to the unavailability of machine j of component i and characterized by the total system productivity $E_{tot}(i, j)$ and by the probability $p_{ij}(d)$ that demand d will not be satisfied, due to the unavailability of this machine.

In order to describe the production system facing to a variable demand, we consider the time period of demand is a set of M intervals, with duration T_i $(i = 1, ..., M)$. Each demand level d_i has duration T_i. The probability q_i of demand d_i can be computed as:

$$q_i = \frac{T_i}{\sum_{j=1}^{M} T_j} \tag{3}$$

The demand is represented by the vector $d = \{d_m\}$ and the corresponding probability vector by $q = \{q_m\}$ $(1 \prec m \prec M)$. Both vectors define the cumulative demand curve, which is usually known for every system.

To improve the accuracy of the results, the boundary effects of elements being unavailable between two adjacent demand intervals are considered. For each element $v(i, j)$ the downtime is divided into L equal intervals with duration $\theta = \frac{D_{(i,j)}}{L}$. Consequently, a new cumulative demand curve is obtained for each element $v(i, j)$. This curve, as shown in Fig.(2), is defined by the new demand and corresponding probability of demand vectors $\boldsymbol{d} = \{d_{(m,s)}\}$ and $\boldsymbol{q} = \{q_{(m,s)}\}$, $(1 \le m \le M)$ and $(1 \le s \le L)$.

The elements of the ***d*** and ***q*** vectors are calculated as

$$\begin{cases} q_{(m,1))} = \left(q_m - D_{(i,j)}\right) \big/ q_m \\ q_{(m,s)} = \theta \big/ q_m \ , \ (2 \le s \le L) \end{cases} \tag{4}$$

$$\begin{cases} d_{(m,1)} = d_m \\ d_{(m,s)} = \dfrac{d_m \times (L - s + 1) + (s - 1) \times d_{\mathrm{mod}_M (m+1)}}{L}, \\ (2 \le s \le L) \end{cases} \tag{5}$$

Note, that in production systems with variable demand we assume that the overall probability that the demand will not be met is used as a measure of system unreliability; like in electric power system, Loss of Load Probability (LOLP) is often estimated [6-7]. To compute this index, first the total unsatisfied demand probability should be calculated as

$$\begin{aligned} P_{ud} &= \sum_{i=1}^{N} \sum_{j=1}^{K_i} \sum_{m=1}^{M} \sum_{s=1}^{L} \Pr\left(E_{tot}(i, j) \prec d_{(m,s)}\right) \\ &= \sum_{i=1}^{N} \sum_{j=1}^{K_i} \sum_{m=1}^{M} \sum_{s=1}^{L} q_{(m,s)} p_{ij} (d_{(m,s)}) \end{aligned} \tag{6}$$

The measure of reliability of entire system is defined by R index, given by the expression $R = 1 - P_{ud}$. This index will be compared and must not be less than some preliminarily specified level R_0.

The problem of optimal production system design can be formulated as follow:

find the system configuration $f, v_1, v_2, ..., v_n$ that provides the minimum total cost under reliability constraint. This problem can be stated as:

Minimise

$$C = \sum_{m=1}^{N-1} C^B_{f(m)} + \sum_{i=1}^{N} \sum_{j=1}^{K_i} C_{v(i,j)} \tag{7}$$

Subject to

$$R(f, v_1, v_2, ..., v_n, d, q) \geq R_0$$

3 DETERMINATION OF SYSTEM UNSATISFIED DEMAND PROBABILITY

Let consider the production manufacturing system with its given structure ($f, v_1, v_2, ..., v_n$). Each machine j of component i, the technology $v(i, j)$ is chosen, and for each component i the technology of buffer $f(i)$ is chosen.

Under the assumption 1, we consider each component separately. Therefore, if one of elements of component i is unavailable at any moment, the whole capacity of the rest of components have their maximal productivity. In this case only the component with an unavailable machine can become the bottleneck of the system and, indeed, only the influence of this component on the entire system productivity should be estimated. The latter should be calculated as flows, let the i th component has total productivity:

$$e_i = \sum_{v=1}^{K_i} E_v(i, j) \tag{8}$$

Suppose that element n of this component is unavailable, the potential productivity of the entire system is reduced to $e_i - E_v(i,n)$. The productivity deficit in component of type i, due to the unavailability of machine n, depends on the system demand d and is defined as follows:

$$\varepsilon_{in}(d) = d - e_i + E_v(i,n) \tag{9}$$

In the case when $\varepsilon_{in}(d) \leq 0$, the machine unavailability does not cause the entire system productivity reduction and the corresponding probability $p_{in}(d) = 0$. On the contrary, if the productivity deficit exists ($\varepsilon_{in}(d) > 0$), this situation can be compensated by unloading downstream located buffers. If the downtime of the unavailable machine is $D_{v(i,n)}$, the time of buffers unloading can be calculated as:

$$t_{in}(d) = \begin{cases} D_{v(i,n)}, & if \quad \dfrac{\sum_{j=i}^{N-1} V_{f(j)}}{\varepsilon_{in}(d)} \geq D_{v(i,n)} \\ \dfrac{\sum_{j=i}^{N-1} V_{f(j)}}{\varepsilon_{in}(d)}, & if \quad \dfrac{\sum_{j=i}^{N-1} V_{f(j)}}{\varepsilon_{in}(d)} < D_{v(i,n)} \end{cases} \tag{10}$$

where $\sum_{j=i}^{N-1} V_{f(j)}$ represents the total capacity of downstream located buffers.

Due to the unavailability of the machine n, the probability that demand d will not be satisfied can be calculated as follows:

$$p_{in}(d) = \begin{cases} 0, & if \quad \varepsilon_{in}(d) \leq 0 \\ \dfrac{D_{v(i,n)} - t_{in}(d)}{U_{v(i,n)} + D_{v(i,n)}}, & if \quad \varepsilon_{in}(d) > 0 \end{cases} \tag{11}$$

In the end, we apply this equation for all the demand levels and all system machines, after using the expression (6) we can obtain P_{ud} for entire system.

4 THE ANT COLONY OPTIMIZATION (ACO) APPROACH

Ants lay down an aromatic substance, known as *pheromone,* in some quantity in their way to food. The pheromone quantity depends on the length of the path and the quality of the discovered food source. An ant chooses a specific path in correlation with the intensity of the pheromone. The pheromone trail evaporates over time if no more pheromone is laid down. Other ants can observe the pheromone trail and are attracted to follow it. Thus, the path will be marked again and will therefore attract more ants. The pheromone trail on paths leading to rich food sources close to the nest will be more frequented and will therefore grow faster. In this way, the best solution has more intensive pheromone and higher probability to be chosen. The described behaviour of real ant colonies can be used to solve combinatorial optimization problems by simulation: artificial ants searching the solution space simulate real ants searching their environment. The objective values correspond to the quality of the food sources. The ACO approach associates pheromone trails to features of the solutions of a combinatorial optimization problem, which can be seen as a kind of adaptive memory of the previous solutions. In addition, the artificial ants are equipped with a local heuristic function to guide their search through the set of feasible solutions. Solutions are iteratively constructed in a randomized heuristic fashion biased by the pheromone trails left by the previous ants. The pheromone trails are updated after the construction of a solution, enforcing that the best features will have a more intensive pheromone.

4.1 Applying the ACO Meta-Heuristics to Structure Optimisation Problem (SOP)

To apply the ACO meta-heuristics to a combinatorial optimization problem, it is convenient to represent the problem by a graph G = (ς, Λ), where ς are the nodes and Λ is the set of edges. To represent the SOP as such a graph, the set of nodes ς is given by components and elements, and edges connecting each component to its available elements. Some nodes are added to represent positions where additional component was not used. As in [8-10], these nodes are called blank nodes and have attributes of zero. The obtained graph is partially connected. Ants cooperate by using indirect form of communication mediated by pheromone they deposit on the edges of the graph G while building solutions.

Informally, the algorithm works as follows: y ants are initially positioned on node representing a component. Each ant represents one possible structure of the entire system. This structure is represented by K_i elements in parallel for N different components. The K_i elements can be chosen in any combination from the V_i available type of elements. Each ant builds a feasible solution (called a tour) to the SOP by repeatedly applying a stochastic greedy rule, i.e., the *state transition rule.* While constructing its solution, an ant also modifies the amount of pheromone on the visited edges by applying the *local updating rule.* Once all ants have terminated their tour, the amount of pheromone on edges is modified again (by applying the *global updating rule).* Ants are guided, in building their tours, by both heuristic information (they prefer to choose "less expensive" edges), and by pheromone information. Naturally, an edge with a high amount of pheromone is a very desirable choice. The pheromone updating rules are designed so that they tend to give more pheromone to edges which should be visited by ants.

4.2 State Transition Rule

In the above algorithm, at each step of the construction process, ants use problem-specific heuristic information (denoted by η_{ij}) and pheromone trails (denoted by τ_{ij}) to select K_i elements for each sub-system. An ant positioned on node i (representing a system component i) chooses the element $v(i,j)$ by applying the rule given by:

$$V(i,j) = \begin{cases} \max\limits_{m \in [0,\ldots,V_i]} ([\tau_{im}]^{\alpha}[\eta_{im}]^{\beta}) & if \quad q \le q_o \\ V(i,j) & otherwise \end{cases} \tag{12}$$

and V(i,j) is a random element selected according to the probability distribution given by:

$$p_{v(i,j)} = \frac{\left[\tau_{v(i,j)}\right]^{\alpha}\left[\eta_{v(i,j)}\right]^{\beta}}{\sum\limits_{m=0}^{V_i}\left[\tau_{im}\right]^{\alpha}\left[\eta_{im}\right]^{\beta}} \tag{13}$$

α and β are parameters that control the relative weight of the pheromone (τ_{ij}) and the local heuristics (η_{ij}), respectively; AC_i is the set of available element choices for the system component i, q is a random number uniformly distributed in [0,1]; and q_o is a parameter ($0 \le q_o \le 1$). The parameter q_o determines the relative importance of exploitation versus exploration: every time an ant in node i has to choose an element j, it samples a random number $0 \le q \le 1$. If $q \le qo$ then the best edge, according to equation (12), is chosen (exploitation), otherwise an edge is chosen according to equation (13) (biased exploration). The state transition rule resulting from theses equations is a *pseudo-random proportional* *rule*. The heuristic information used is $\eta_{ij} = 1/(1+c_{ij})$ where c_{ij} represents the associated cost of element j for component i. In equation (13) we multiply the pheromone on edges by the corresponding heuristic value. In this way we favour the choice of edges which are weighted with smaller costs and which have a greater amount of pheromone. That is, elements with smaller cost have greater probability to be selected.

4.3 Local Updating Rule

While building a solution of the problem of structure optimisation, ants choose elements by visiting edges of the graph *G,* and change their pheromone level by applying the following local updating rule:

$$\tau_{v(i,j)}^{new} \rightarrow (1-\rho)\tau_{v(i,j)}^{old} + \rho\tau_o \quad (14)$$

where ρ is a coefficient such that $(1-\rho)$ represents the evaporation of trail; and τ_o is the initial value of trail intensities.

The application of the local updating rule, while edges are visited by ants, has the effect of lowering the pheromone on visited edges. The pheromone reduction is small but sufficient to lower the attractiveness (or desirability) of precedent edge. This favours the exploration of edges not yet visited, since the visited edges will be chosen with a lower probability by the other ants in the remaining steps of an iteration of the algorithm. Thus, by discouraging the next ant from choosing the same element during the same cycle, ants do not converge to a common solution and premature convergence is avoided.

4.4 Global Updating Rule

Once all ants have built a complete system, pheromone trails are updated. Only the globally best ant (i.e., the ant which constructed the best design solution during a complete cycle) is allowed to deposit pheromone. A quantity of pheromone $\Delta\tau$ is deposited on each edge that the best ant has used. The quantity $\Delta\tau$ is given by $(1/TC_{best})$,where TC_{best} is the total cost of the designed feasible solution constructed by the best ant. Therefore, the global updating rule is:

$$\tau_{v(i,j)}^{new} = (1-\rho)\tau_{v(i,j)} + \Delta\tau_{v(i,j)} \quad (15)$$

where $0 \prec \rho \prec 1$ is the pheromone decay parameter representing the evaporation of trail. Global updating is intended to allocate a greater amount of pheromone to less expensive design solution. Equation (15) dictates that only those edges belonging to the globally best solution will receive reinforcement. The meta-heuristics described above has been applied successfully to a power station coal transportation system which supplies a boiler. This example was taken from reference [8] where a genetic algorithm is used to solve the SOP.

4.5 Overview of The Ant Algorithm

STEP 1. Initialisation

Set: NC=0 /*NC : Cycle counter*/

For every combination (i,j) /*i: Sub-System index j: element index*/

Set an initial value $\tau_{v(i,j)}(0) = \tau_0$ and $\Delta\tau=0$

End

For every combination (i,j) of available buffers

Set an initial value $\tau_{ij}^{B}(0)=\tau_0^{B}$ and $\Delta\tau^{B}=0$

End

STEP 2. System selection and system cost computation

For k:=1 to y /*y: number of ants*/

For i:=1 to N /*N: number of sub-systems*/

For j:=1 to k_{max} /*k_{max}: maximum number of parallel components allowed in sub-systems */

Choose a component *V(i,j)* with transition probability given by equations (12), (13) /* This selection can return blanks : No component selected*/

Set $C_e:=C_e+C_{V(i,j)}$ /* C_e : cost of selected elements*/

Local pheromone updating of elements according to equation (14) applied to elements selection

End

Set: $k_i := k_{max}$-number of blanks /*k_i: number of elements in sub-system i*/

End

For i:=1 to N -1

Choose a buffer $V_{f(i)}$ with transition probability given by equations (12), (13) (application for buffers) /* This selection can return blanks : No buffer installed at subsystem *i* */

Set $C_b:=C_b+C_{Vf(i)}$ /*C_b: cost of buffers; $C_{Vf(i)}$:cost of selected buffer of version f */

Local pheromone updating of selected buffers according to equation (14) applied to buffers

End

Compute the system cost for each ant

Set $C_{sys}^{K}=C_e+C_b$ /* C_{sys}^{K} : total cost of system selected by kth ant*/

STEP 3. Availability computation

Set P_u:= 0 /*P_u: total unsatisfied demand*/

For i:=1 to n

For j:=1 to k_i

Compute demand level curve for each element using equations (4) and (5)

For m:=1 to M

For s:=1 to L

Compute unsatisfied demand $P_{uV(i,j)}$ for elements v(i,j) for all demand levels (m,s) according to equations (10), (11) and (6)

End, End

Set: $P_u = P_u + P_{u\,V(i,j)}$

End, End

Compute the system availability for each ant

$R_k = 1 - P_u$

If $R_k < R_0$

Then Set $W := W_p$ /* W_p: large penalty cost*/

Else Set $W := 0$

Set $C_{sys}^{K} = C_{sys}^{K} + W$

End

Set: $\Delta\tau = G \big/ \left(C_{sys}^{K}\right)_{\min}$ /* G: constant; $\left(C_{sys}^{K}\right)_{\min}$: minimum cost in cycle (best ant)*/

Set $\begin{cases} \Delta\tau_{v(i,j)} = \Delta\tau, \ldots \textit{if element } V(i,j) \in \textit{best solution} \\ 0, \ldots \textit{otherwise} \end{cases}$

Update the best solution according to equation (15)

/* this global updating includes elements and buffers*/

STEP 4. Set: Nc:=Nc+1

For each combination (i, j), set $\Delta\tau = 0$

STEP 5. If $\{NC < NC_{max}\}$ and { not stagnation behaviour}

Then Go To Step **2**.

Else Print and save the best feasible solution.

End

Stop.

5 ILLUSTRATIVE EXAMPLE

5.1 Description Of The System To Be Optimized

The power station coal transportation system which supplies the boilers is designed with five basic components as depicted in Figure 3. Figure 2 shows the detailed process of the power station coal transportation.

The process of coal transportation is as follows: The coal is loaded from the bin to the primary conveyor (Conveyor.1) by the primary feeder (Feeder.1). Then the coal is transported through the conveyor 1 to the Stacker-reclaimer, where it is lifted up to the burner level. The secondary feeder (Feeder.2) loads the secondary conveyor (Conveyor.2) which supplies the burner feeding system of the boiler. Each element of the system is considered as unit with total failures.

To provide a desired total availability, the system should be constructed by the choice among several products available on the market. The characteristics for each type of components are presented in Table 1. The latter shows for each component (*Component i*) the corresponding available versions, their capacities Σ, up and down times and their costs C. Without loss of generality both the equipment capacity and the demand levels can be measured as a percentage of the maximum boiler capacity (Demand) at each interval as shown in Table 2. Interval duration of load can be measured as a fraction (percentages %) of the total operation time (T).

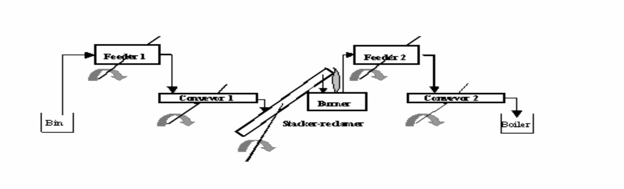

Figure 2: Detailed Power Coal Station System

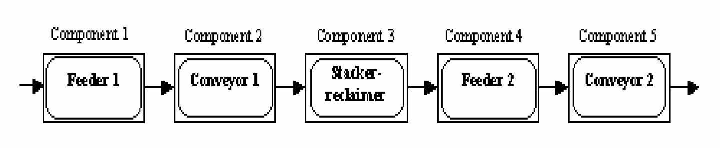

Figure 3: Structure Synoptic of a Power System

5.2 Discussion of Obtained Result

Table 4 shows the optimal or near optimal solutions obtained by the suggested Ant colony algorithm for different reliability constraints specified by the index R_0 in the range varying from 0.92 to 0.99. Table 4 also shows the calculated reliability index R, the cost of the system C_{sys} and its structure represented by a vector of element version numbers and a vector of buffer numbers for each component. In order to estimate the effects of introducing buffers on the system cost and reliability, optimal solutions of structures without buffers under similar reliability constraints are included in the same table. A comparison of these results shows that with buffers the same reliability level is achieved with lower cost.

The results in Table 4 enable the decision maker to evaluate the reliability-cost trade-off, e.g. less than 5% of the investment is needed to improve the reliability level from 92% to 96% while 25% of investment is required to increase this level from 96% to 99%. This will obviously make the decision making process much easier.

The parameters considered here are those that affect directly the computation of the formulas used in the algorithm (α, β and ρ). We tested several values for each parameter, all the others being constant. The values tested were: $\alpha \in \{0, 0.5, 1, 2\}$, $\beta \in \{0.5, 1, 2, 5, 10\}$ and $\rho \in \{0.3, 0.5, 0.7, 0.9\}$. The best values for these parameters that converge rapidly to optimal solutions were: $\alpha = 1$, $\beta = 2$ and $\rho \in (0.5, 0.7)$.

The program was run on a 2.4 Ghz processor with an ant colony of 50 agents and a maximum of 100 cycles. The running time required to obtain the optimal or near optimal solution did not exceed 95s and 83 cycles was the maximum number of cycles to reach the best solution.

Table 1: Data Example

Comp#	**Vers#**	**Capacity** Ξ(%)	**Uptime** $\boldsymbol{U}$(h)	**Downtime** $\boldsymbol{D}$(h)	**Cost** $\boldsymbol{C}$
	1	120	755.0	16.0	0.590
	2	100	841.0	19.0	0.535
	3	85	710.0	20.0	0.470
1	4	85	705.0	15.0	0.420
	5	48	720.0	12.0	0.400
	6	31	380.0	18.0	0.180
	7	26	677.0	12.0	0.220
	1	100	1200.0	11.0	0.205
	2	92	1080.0	8.0	0.189
2	3	53	880.0	9.0	0.091
	4	28	1140.0	5.0	0.056
	5	21	940.0	10.0	0.042
	1	100	70.0	2.0	7.525
	2	60	73.0	2.0	4.720
3	3	40	66.0	2.0	3.590
	4	20	70.5	1.5	2.420
	1	115	780.0	17.0	0.180
	2	100	895.0	15.0	0.160
	3	91	643.0	12.0	0.150
	4	72	956.0	16.0	0.121
4	5	72	845.0	19.0	0.102
	6	72	776.0	14.0	0.096
	7	55	739.0	12.0	0.071
	8	25	658.0	17.0	0.049
	9	25	713.0	19.0	0.044
	1	100	530.0	13.0	7.525
	2	60	435.0	8.0	4.720
5	3	40	485.0	12.0	3.590
	4	20	385.0	10.0	2.420

Table 2: Data of Cumulative Load-Demand Curve

d_m (%)	100	80	40	70
q_m (h)	0.31	0.19	0.22	0.28

Table 4: Parameters of Optimal Solutions for Different Reliability Requirements

R_0	With buffers					NO buffers			
	R	C_{sys}	Component	Elements	Buffers	R	C_{sys}	component	Elements
0.92	0.921	16.442	1	2	---	0.922	19.802	1	2
			2	3,3	---			2	3,3
			3	1	---			3	1,3
			4	2	3			4	2
			5	2,3				5	2,3
0.94	0.942	16.964	1	4,6	---	0.941	20.395	1	4,6
			2	3,4,5	---			2	3,4,5
			3	1	---			3	1,3
			4	1	3			4	1,7
			5	2,2				5	2,2
0.96	0.961	17.228	1	5,,6	---	0.962	21.628	1	5,6,6
			2	3,4,5	---			2	3,4,5
			3	1	---			3	2,2
			4	3,9	2			4	3,9
			5	2,2				5	2,2
0.98	0.980	19.545	1	4,6,6	---	0.981	22.726	1	4,6,6
			2	3,4,4	---			2	3,4,4
			3	2,3	3			3	2,2
			4	6,6	1			4	6,6,7
			5	2,2				5	1,2
0.99	0.991	21.512	1	4,6,7	---	0.990	27.573	1	4,4,6
			2	3,4,4,4	---			2	3,4,4,4
			3	2,2	2			3	2,2,2
			4	7,7,7	1			4	7,7,7
			5	2,2				5	2,2,3

6 CONCLUSION

In this paper, we solve the interesting cost reliability design problem which is often encountered when designing industrial production systems. An ACO based algorithm is used for the resolution of this problem. The developed algorithm, which minimizes total investment cost subject to reliability constraints, is proposed for choosing an optimal buffer incorporating series-parallel production system configuration. This algorithm seeks and selects system elements among a list of available products according to their cost and compute the reliability coupled to the demand load of the selected systems. So as to eliminate less expensive systems that do not satisfy the reliability requirements a large penalty cost is inflicted on such systems.

The proposed method provides a practical way to solve wide scope of production systems reliability optimization problems without limitation on the diversity of available versions of elements and buffers.

REFERENCES

[1] Levitin G., Meizin L.: Structure optimization for continuous production systems with buffers under reliability constraints. International Journal of Production Economics 70, 2001, 77-87.

[2] Meizin L. K.: Choosing the optimal buffer stock between two section of a continuous process. Automation and Remote control 45 (8), 1984, 1086-1089.

[3] Meizin L. K.: Optimizing contingency reserve. Automation and Remote control 48 (2), 1987, 268-271.

[4] Kimura T.: Optimal buffer design of an M/G/s queue with finite capacity. Stochastic Models 12 (1), 1996, 165-180.

[5] Levitin G., Lisniaski A., Elmakis D.: Structure optimization of power system with different redundant elements. Electric Power System Research 43, 1997, 19-27.

[6] Levitin G., Lisniaski A., Ben-haim H., Elmakis D.: Redundancy optimization for series-parallel multistate system. IEEE Transactions reliability 47 (2), 1998, 165-172.

[7] Billiton R., Allan R.: Reliability Evaluation of power systems. Pitman, London, 1984.

[8] Liang Y.C., Smith A. E.: An Ant Colony Approach to Redundancy Allocation. 2001

[9] Di Caro G., Dorigo M.: Mobile Agents for Adaptive Routing. *Proceedings for the 31st Hawaii International Conference On System Sciences*, Big Island of Hawaii, 1998, 74-83.

[10] Costa D., Hertz A.: Ants Can Colour Graphs. Journal of the Operational Research Society, Vol. 48, 1997, 295-305.