8

Hybrid Petri Nets and Metaheuristic Approach to Farm Work Scheduling

Senlin Guan[1], Morikazu Nakamura[1] and Takeshi Shikanai[2]
[1]Faculty of Engineering,
[2]Faculty of Agriculture,
University of the Ryukyus, 1 Senbaru Nishihara,
Okinawa
Japan

1. Introduction

Scheduling problems for general cases are characterized as NP hard, and the computation time required to obtain the optimal schedule will grow exponentially with the problem size. The scheduling problems that consider the limited or shared resources, alterable constraints or environmental changes become very complex in both formulation and solution. Since the solution for these problems has great serviceability and reliability against environmental changes, much research has been devoted in optimization strategy in the presence of a wide range of uncertainties (Li & Ierapetritou, 2008). Such research with application is applicable to not only the manufacturing in industry, but also production in agriculture. Modeling and scheduling in the agricultural domain may be more promising because of the requirement of new approaches to handling the uncertainties in the nature environment.

In agriculture, a system that aims to produce maximum amount of profit from available land by high inputs of capital, labour, or efficient usage of machinery, is defined as intensive farming (or intensive agriculture). Like common businesses, many intensive farming units are operating their businesses by the ways to improving profits in farming while reducing costs. In Japan, there are over 190,000 intensive farming units such as farmers' cooperatives/ agricultural corporations that aim at efficient and large-scale farm management (The Ministry of Agriculture, Forestry and Fisheries of Japan, 2006). These corporations lease and consolidate agricultural lands in vicinal regions, manage large-scale farmland with full mechanization, and carry out farm works entrusted by vicinal farmers. The farmlands managed by these corporations sometimes number over thousands and are scattered within a wide area. In order to gain substantial economic increase and further development, these corporations need to improve the daily work management, extend the contracts of leasing farmland, lease more farmlands, and carry out more extra works. As a consequence, they considerably require wise management decisions such as timeliness in all operations, equipment adjustments, crop rotations, land rent, taxes and so on. The best decision certainly conduces to the increase of yield, profitability, and work efficiency.

Solving the farm work scheduling problem requires appropriate approaches to modeling and optimization. There are plenty of mathematical models and approaches have addressed optimization for scheduling for different demands (Bassett et al., 1997; Balasubramanian & Grossmann, 2003; Janak & Floudas, 2006; Lin et al., 2004; Till et al., 2007; Wang, 2004; Santiago- Mozos et al., 2005; Suliman, 2000). In agriculture, the existing researches on scheduling in the cropping system involve such as the farming and planning systems for paddy rice production (Nanseki, 1998; Nanseki et al., 2003; Daikoku, 2005), a stochastic farm work scheduling algorithm based on short-range weather variation (Astika et al., 1999), several models simulating a single operation (Arjona et al., 2001; Higgins & Davies, 2005) and operating with one or more crops (Chen & McClendon, 1985; Tsai et al., 1987; Lal et al., 1991; Haffar & Khoury, 1992), and so on. However, none of these studies paid attention to the scheduling in the intensive farming system by using a more promising tool - Petri nets model. A Petri net is a very applicable to model distributed, concurrent, nondeterministic and/or stochastic events, and considerably accommodates nondeterministic events in the farming system such as machine breakdown and labor absence, or concurrent activities such as cooperative works.

In this study, we proposed a hybrid Petri nets and metaheuristic approach to the farm work scheduling in the cropping system. We used sugarcane farming as an example to demonstrate the approache to constructing an efficient farm work plan. The farm work scheduling comprises a model for modeling the farm works and a scheduling system for optimizing the farm work schedule. The model, which is a hybrid Petri nets first introduced into agricultural production (Guan et al., 2008), graphically formulates the farm work flow and simulates the overall status of the progress of farm work and the state of resources. The scheduling system performs the resource assignment and the computation for a long-term schedule (Guan et al., 2009). A part of contents and experiment data in this chapter are originally published in these two articles.

2. Modeling and formulating the farm work scheduling

In this section, we neglect the basic definition of discrete Petri net and continuous Petri net and briefly review the key concepts of hybrid Petri nets.

2.1 Hybrid Petri nets

As defined in Murata (1989), a Petri net is a graphical and mathematical modeling tool for describing and simulating the distributed systems. A hybrid Petri nets informally contains a discrete part and a continuous part (Fig. 2). The discrete part of hybrid Petri nets usually models the state of resource, and the continuous part simulates the process over time.

A hybrid Petri net system is defined as $\mathcal{N} = \langle \boldsymbol{P}, \boldsymbol{T}, \boldsymbol{Pre}, \boldsymbol{Post}, \boldsymbol{m}_0, \boldsymbol{h} \rangle$, where $\boldsymbol{P}$ is a set of places; $\boldsymbol{T}$, a set of transitions; $\boldsymbol{Pre}$ ($\boldsymbol{Post}$), the pre- (post-) incidence function representing the input (output) arcs; $\boldsymbol{m}_0$, a function representing the initial number of tokens in each place; $\boldsymbol{h}$, a hybrid function that indicates a discrete or continuous node. In a hybrid Petri net, all discrete input places must also be output places with arcs of the same weight, and vice-versa.

Figure 1 illustrates a hybrid Petri nets. The double circles, boxes and blue arcs are the continuous part of Petri nets. At time $t = 0$, continuous transition T_{11} starts firing because of the existence of a token in discrete place P_1. Discrete transition T_1 fires when the token in continuous place P_{11} decreases to 2100 at time $t = 3$, and the system is in the break state. After T_2 fires at $t = 3.25$, the system switches to the working state again. Likewise, a continuous transition can be flexibly broken and well controlled by handling with the tokens in the discrete places.

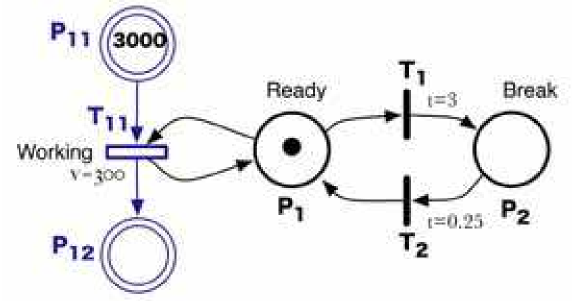


Fig. 1. A hybrid Petri nets system

In a hybrid Petri net that time is associated either with the places or with the transitions, the marking *m* at time *t* of hybrid Petri nets can be written as:

$$m(t) = m(0) + A\cdot\left(n(t) + \int_0^t v(\tau)\cdot d\tau\right) \tag{1}$$

where *A* is the incidence matrix, and *n*(*t*) denotes the number of firings of the discrete transitions from the initial time to time *t*. $v(\tau)$ is the firing speed of the continuous transitions at an arbitrary time τ.

2.2 Hybrid Petri nets modeling for farm work flow

In order to model the variable farming process due to environmental changes, we applied the hybrid Petri net to model the discrete and continuous farming activities. For example, the major works for sugarcane production involve tilling, planting, irrigating, weeding, fertilizing and harvesting. Each work starts when satisfying the conditions such as timeliness of operation, availability of farmland, machinery, labor and so on. After completion of the work, the farmland shifts into the next state, and the resources such as the machinery and the labor are released and ready for the other works. In our proposed model, the farm work is defined as the transition; the condition, or state of a farmland or a resource as the place, and resources like labor or machinery as the tokens. The transitions corresponding to the farm work and the places corresponding to the farmland are the continuous part of the model; otherwise, the places are discrete. A farming process is a continuous transition which the working speed is determined by the capability of the labor, the efficiency of the machinery or a combination of both of them. Uncertainties such as machinery breakdown and breaks, and the state of resources, farmland and machine are considered as discrete objects. The work flow of producing sugarcane by certain resources and the simple model for modeling the work flow in one farmland are illustrated in Fig. 2.

In the figure, continuous places P_{ij} and transitions T_{ij} represent the status of the farmland and execution of the farm work, respectively. The discrete places correspond the status of the resources such as the tractor, rotary tiller, labor, and so on. P_{ij} is a timed continuous place associated with time window for timeliness of operation. The real number in places P_{ij} means the amount the farm work. Note that the resources are assigned to only two works of the tilling and harvesting in this figure. At the initial state, the existence of the token in P_{11} indicates that the farmland is ready for the work of tilling. Along with the execution of the subsequent work with the available resources of R_1,R_2,R_3, the value in P_{11} decreases while

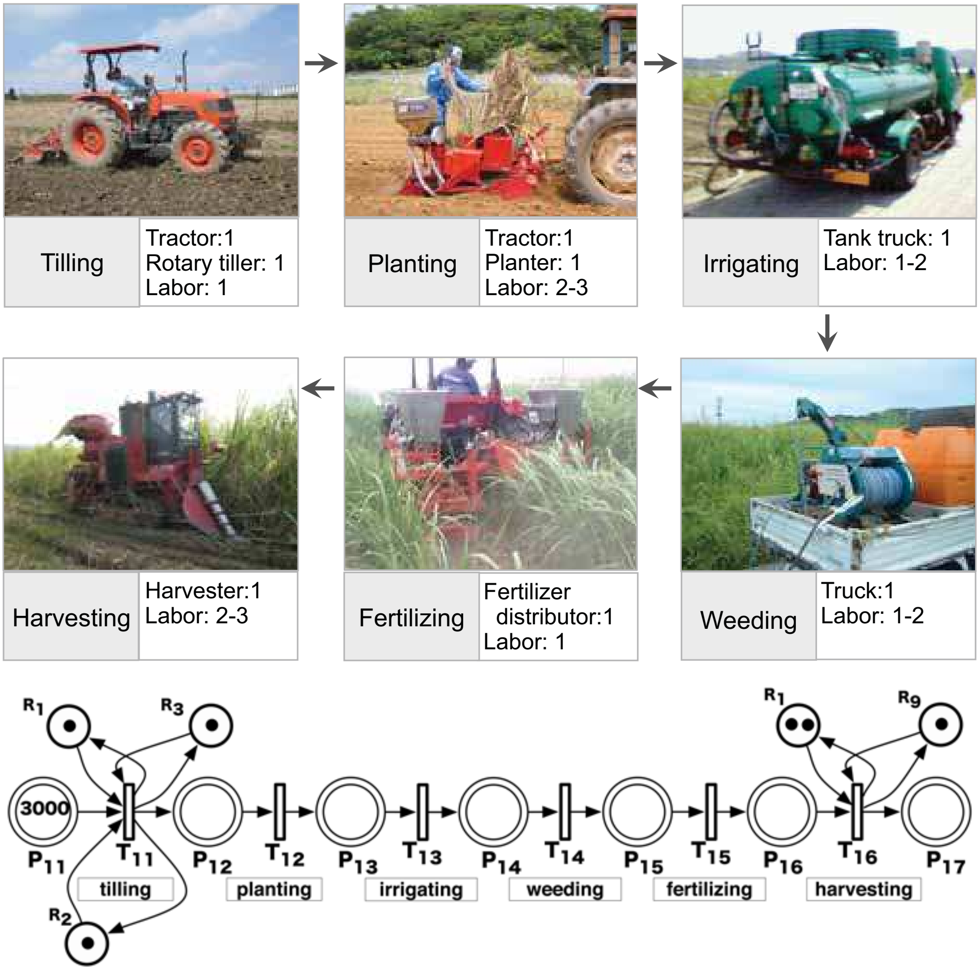


Fig. 2. Hybrid Petri nets modeling for farm work flow in sugarcane production

that in P_{12} increases. When the work of tilling is completed, the token in P_{11} and P_{12} reach 0 and 3,000, respectively, and the resources are released and ready for the works in the other farmlands. The model for modeling the farm works in multiple farmlands is based on this elementary model.

2.3 Formulating the farm work scheduling on hybrid Petri nets

A farm work schedule is to plan the farm works in the farmlands with the necessary resources over time. The farm works in a farmland range from the tilling to harvesting in a crop growth cycle. Since more than one of machinery and labor are available for any work, the cooperative work sometimes takes place for early completion of the work.

Figure 3 simulates the scheduled farm work on the hybrid Petri nets model. A continuous transition denotes the execution of task w_{ijk}. Not only cooperative work but also breaks are modeled in the figure. From the initial state, the first work in farmland F_1 is started cooperatively by resource R_1 and R_2. At the same time, the first work in F_2 is also started by only one resource R_3. The break time includes the normal break time and the time that may be consumed by uncertainties such as machinery breakdown, poor weather, and so on. For each resource R_k, the break and resumption for task w_{ijk} are modeled by the discrete part of Petri net connected to a continuous transition, which consists two discrete places and two timed discrete transitions.

The model acts as modeling the farm work process as well as simulating farm work schedule. Since the marking of hybrid Petri net implies the farming progress and the state of farmlands and resources, we can monitor the entire state of the system by the marking migration according to Equation (1).

Generating the hybrid Petri nets in Fig. 3 requires assigning resources to the discrete places and designating the firing sequence of the transitions in advance. We stipulate the rules for resource assignment and firing operation of hybrid Petri nets as follows:

1. The number of assigned resources for the cooperative work is limited from at least one to the total number of available resources for this work. In the dynamically generated hybrid Petri nets, the number of continuous transitions is equal to the number of assigned resources. If the resource assignment is determined, the hybrid Petri nets model including the continuous places, continuous transitions, and discrete places except the arcs from the continuous transition to the discrete place can be generated. The cooperative work may cause a deadlock that a resource is scheduled to an already completed work. For this case in the computing process, this resource will be rescheduled to the next task.
2. The firing operation of the hybrid Petri nets stops when all tasks are completed.
3. A resource cannot be assigned to two works at the same time.
4. The timed continuous places and transitions are enabled during the time window and over the waiting time, respectively.
5. The firing operation suffers from the *precedence constrained* relationship. For example, tasks w_{124} and w_{125} cannot be started if the token in P_{12} is less than 2880.
6. The moving time of the resource between farmlands is associated to the discrete transition.
7. Arbitrary breaks are possible during the farm work.

2.4 Formulating the farm work scheduling in mathematical method

In this subsection, we redefine the farm work scheduling problem in mathematical method and compare it with the one by hybrid Petri nets. The variables for the farm work scheduling and their descriptions are listed in Table 1. In the table, resource R_k is not an individual resource but rather a set of the minimum machinery and labor required for the work. m_{ij} represents the amount of scheduled work W_j in farmland F_i. W_j can be carried out only if $m_{ij} > 0$ and I_{ij} =1. Waiting time O_{ij} and time window U_j are used to define an appropriate cultivation time. The execution time of task w_{ijk} is subject to the completion time of the pervious work, waiting time O_{ij} and the period of $[U_j(s), U_j(e)]$. The relations between these variables are shown in Fig. 4.

In Fig. 4, w_{bjk} may be performed in cooperation with other resources. Such cooperative work is defined as a process where multiple machineries perform the same work, and the entry time of a resource to perform cooperative farming work is arbitrary. The execution of work

W_j requires at least one resource, and the total number of assigned resources is less than or equal to the number of resources available to perform W_k ($\Sigma_k S_{jk}$). The number of assigned resources for the execution of work W_j, $\Sigma_k S'_{jk}$, , corresponds to the following condition:

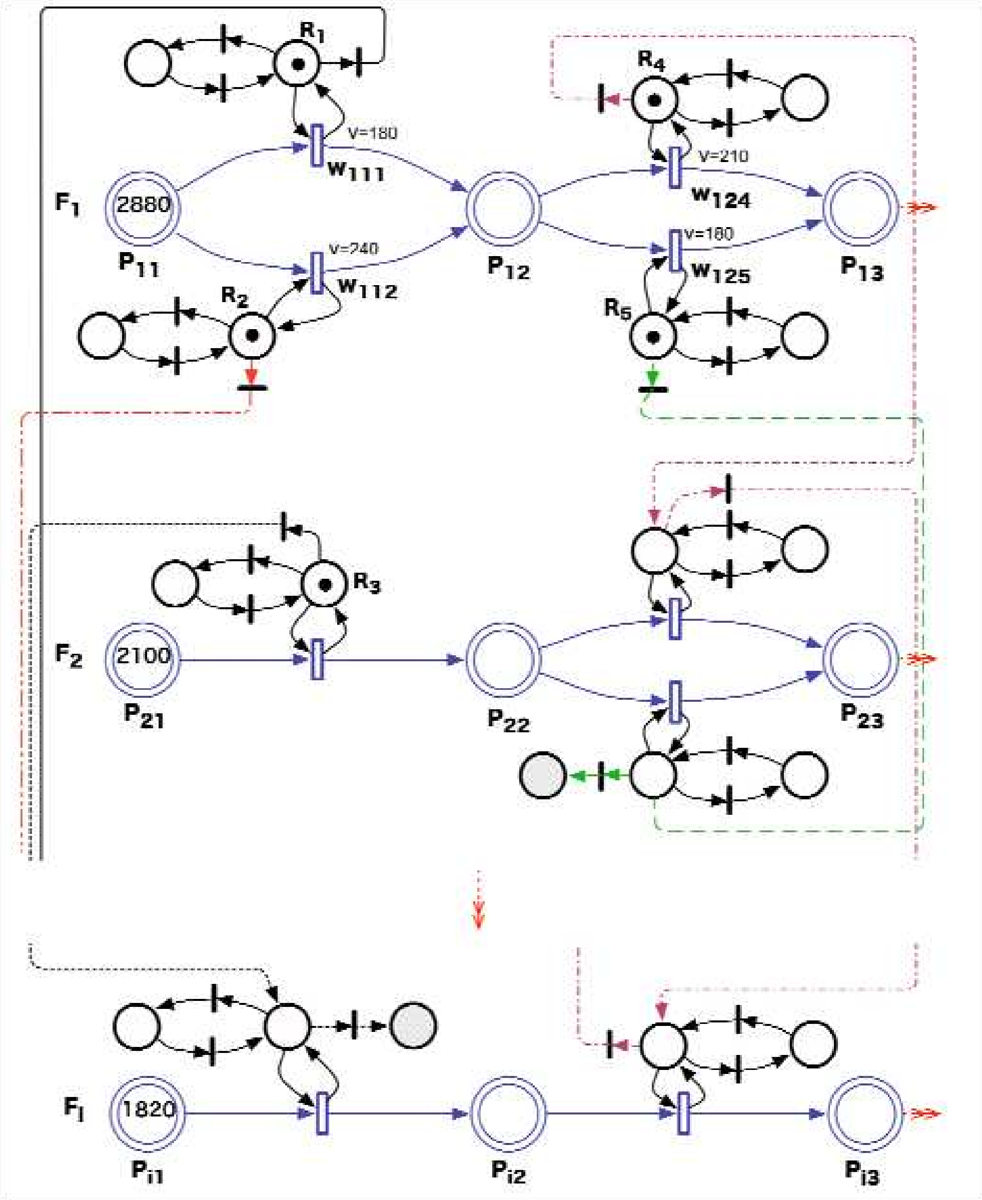


Fig. 3. Hybrid Petri nets model simulates the scheduled farm work

Notation	Definition
N_F	Total number of farmlands
N_W	Total number of works in a crop growth cycle
N_R	Total number of available resources for a work
F_i	Farmland i, $i \in \{1,\ldots, N_F\}$
W_j	Work j, $j \in \{1,\ldots, N_W\}$
R_k	Resource k, $k \in \{1,\ldots, N_R\}$
A_i	Area of F_i
I_{ij}	$I_{ij} \in \{0,1\}$, 1: W_j should be performed in F_i; otherwise, 0
m_{ij}	Amount of scheduled work W_j in F_i, $m_{ij} \in [0, A_i]$
S_{jk}	$S_{jk} \in \{0,1\}$, 1: R_k is available to perform W_j; otherwise, 0
S'_{jk}	$S'_{jk} \in \{0,1\}$, 1: R_k is scheduled to perform W_j; otherwise, 0
v_k	Average working speed of R_k
O_{ij}	Waiting time between end time of W_{j-1} and start time of W_j in F_i
U_j	Time window [start time $U_j(s)$, end time $U_j(e)$] for W_j
w_{ijk}	Task in F_j by R_k, for W_j
t_{ijk}	Working time [Start time $t_{ijk}(s)$, end time $t_{ijk}(e)$] of W_j in F_i by R_k
v'_k	Moving speed of R_k
D_{ab}	Distance between F_a and F_b, a, $b \in \{1,\ldots, N_F\}$

Table 1. Notations for farm work scheduling

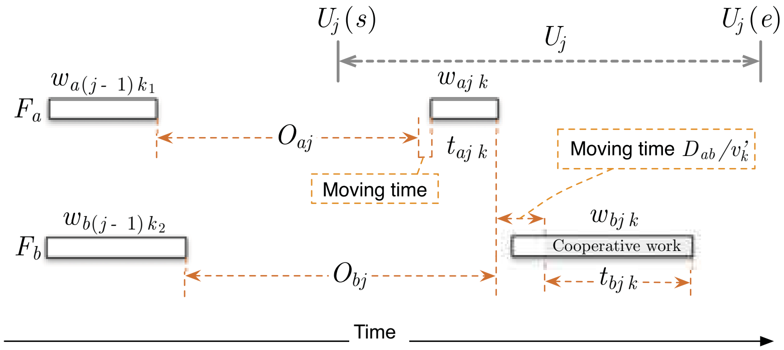


Fig. 4. Defined variables on time axis

$$1 \le \sum\nolimits_k S'_{jk} \le \sum\nolimits_k S_{jk} \tag{2}$$

The amount of scheduled work m_{ij} must be completed by certain resources R_k during t_{ijk} at working speed v_k. As a consequence, the following equation exists:

$$\begin{bmatrix} t_{ij1} & t_{ij2} & \cdots & t_{ijk} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_k \end{bmatrix} = m_{ij} \tag{3}$$

In order to avoid the superposition of work time by a same resource, we have the following equation:

$$\begin{aligned} \forall i,j,p,q,k, \quad & i \neq p, j \neq q, p \in \{1,\ldots,N_F\}, q \in \{1,\ldots,N_W\} \\ & t_{ijk}(e) < t_{pqk}(s) \cdots \text{if } t_{ijk}(s) < t_{pqk}(s) \end{aligned} \tag{4}$$

For the timeliness of the work, the start working time $t_{ijk}(s)$ and the end working time $t_{ijk}(e)$ are subject to the additional conditions stated in Equation (5).

$$\begin{aligned} \forall i,j,k,k', \quad & k' \in \{1,\ldots,N_R\} \\ & t_{ijk}(s) \geq max(U_j(s), t_{i(j-1)k'}(e) + O_{ij}) \\ & t_{ijk}(e) \leq U_j(e) \end{aligned} \tag{5}$$

In a farmland, work W_j can only start after the completion of the former one W_{j-1}, which is defined in the form

$$\forall i,j,k,k', \qquad t_{ijk}(s) > t_{i(j-1)k}(e) \tag{6}$$

Considering the moving time between farmlands, we have

$$\begin{aligned} \forall a,b,j,k, \quad & a \neq b \\ & t_{ajk}(s) \geq t_{bjk}(e) + D_{ab} / v'_k \cdots \text{if } t_{ajk}(e) < t_{bjk}(s) \end{aligned} \tag{7}$$

2.5 Comparison between the two formulation methods

Although the objectives of the farm work scheduling can take many forms such as minimizing the make-span, maximizing plant throughput, maximizing profit or minimizing production costs, we only consider minimizing the idle time between works for both formulation methods in this paper. For the formulation by mathematical definition, the objective function is written as

$$\min \left(\sum\nolimits_{a,b,j,k} [t_{bjk}(s) - t_{ajk}(e)]\right) \tag{8}$$

where task w_{bjk} is a latter task of w_{ajk} ($t_{bjk}(s) \geq t_{ajk}(e)$). It is apparent that the problem defined by Equations (2) - (8) corresponds to a mixed integer nonlinear programming (MINLP). In contrast with mathematical definition, the scheduling objective for the formulation by hybrid Petri nets is to find a firing sequence in that the idle time is minimum.

The firing rules or natural characteristics of Petri nets completely accommodate the constraints defined in the above equations. For example, Rule 1, 2, 3, 4, 5 and 6 accord with Equation (2), (3), (4), (5), (6) and (7), respectively. Rule 7 is reserved for the real-time scheduling and has not defined in mathematical method. The corresponding relation indicates that the formulation by the hybrid Petri nets substantially reduce the complexity of problem, which is also agued in Ghaeli et al. (2006); Sadrieh et al. (2007). Furthermore, the scheduling by the hybrid Petri nets is more representable and comprehensible for the farm work corporations than that in the form of mathematical equations.

3. Metaheuristic approach for optimization

Generating a farm work schedule on the hybrid Petri nets includes the two phases for assigning resources and arranging the work sequence. In the first phase, a scheme of resource assignment to each task is supposed and optimized by a simulated annealing (SA) algorithm; and in the second phase, the seeking for the best work sequence based on the resource assignment obtained in the first phase is executed by the procedure of a genetic algorithm (GA). The work sequence here is designated as a priority list in which works are arranged according to a specific priority. The second phase is in charge of inheriting the present best task sequence, dynamically creating Petri nets, simulating the activities on Petri nets, and evaluating the schedule. At the end of the GA procedure, the present best resource assignment scheme is obtained, and it will be inherited in the continuing first phase.

3.1 SA for optimizing resource assignment

The cooperative work during the farm work process has to be taken into account. The number of assignable resources is defined in Rule 1 or Equation (2). An independent variable *x* in the SA procedure is set to a resource assignment scheme. *x*', that is, another independent variable in the neighboring region of *x*, represents an alterable resource assignment scheme. The pseudo code of the SA in the first phase is described as:

```
begin
  initialize temperature $\Gamma$, neighboring space $n$;
  initialize resource assignment $x$, and minimum fitness $min$;
  evaluate fitness $f_x$ (= $gaPls(x)$) in $2nd$ phase;
  while (not termination-condition) do
    for $i = 1$ to $n$
      generate another resource assignment $x'$;
      evaluate fitness $f_{x'}$ (= $gaPls(x')$) in $2nd$ phase;
      if ($f_{x'} < f_x$) then
        replace $x$ with $x'$;
      else
        if ($random(0,1) < exp(f_x - f_{x'})/\Gamma$) then
          replace $x$ with $x'$;
        end if
      end if
      if ($f_{x'} < min$) then
        update $min$ with $f_{x'}$, and memorize $x'$;
      end if
    end for
    replace $\Gamma$ with ($\Gamma - \Gamma * \alpha$); //$0 < \alpha < 1$
  end while
end
```

At the end of the first phase, a resource assignment is obtained. And then, the length of chromosomes in the GA in the next phase can be designated, and the places and transitions of the hybrid Petri nets can be constructed before the iteration computation by the GA.

3.2 GA for scheduling

Based on the assigned resources, the second metaheuristic GA seeks priority lists to generate the present optimal schedules. The priority list is encoded into a chromosome, in which the tasks (genes) are grouped by the same works W_j. Be similar to operations in traditional GAs, the one-point order crossover, one-bit reverse mutation, roulette selection and an elite selection are incorporated in the GA procedure. The crossover and mutation operations are restricted to those between the tasks in the same works W_j. The fitness function is to evaluate the sum of the moving time and the idle time between the tasks. This objective is achieved by firing the hybrid Petri nets until no continuous transition can fire. When the firing operation stops, the generated schedule will be memorized along with the priority list if it has the current best fitness. The pseudo code of the GA is briefly described in procedure gaPls(x), followed by the procedure of evaluating the fitness.

```
procedure gaPls(x)
begin
   initialize population c with the chromosomes inherited
                 from the present best priority list;
   construct continuous part of hybrid Petri nets;
   evaluation(c);
   while not-termination-condition do
      selection;
      crossover;
      mutation;
      evaluation(c);
   end while
end
```

```
procedure evaluation(c)
begin
   for r = 1 to popSize
      construct the discrete part of the hybrid Petri nets;
      initial time interval δ; current time s = 0;
      while tasks-are-not-completed do
         if (firing-conditions-are-satisfied) then
            firing and update the amount of tokens in the corresponding places;
         end if
         update s with s + δ;
         update the sum of moving time and idle time;
      end while
      if (best-fitness-found) then
         update current best fitness, priority list, and schedule;
      end if
   end for
end
```

3.3 Deadlock removal

A deadlock in farm work scheduling is a situation where two or more competing works await the release of resources and neither obtains the necessary resources. In general, conflicts on resource use have to be examined for deadlock removal in a conventional optimization. For example, assigning a resource to a work in a conventional optimization have to check whether or not the resource is already being used for another work simultaneously; if no resource is available for the work, the computing will shift into a waiting state until some resource is released. Since the computation time in the GA iteration is the product of the size of the population, generation, and evaluation, a long evaluation time that is wasted in resolving the deadlock of resource use results in an inefficient search. Furthermore, some randomly generated individuals in the iterations may be infeasible solutions if the work is scheduled across the time window for cultivation.

In contrast, assigning resources in the first phase before the GA iterations may remarkably prevent deadlocks caused by resource conflict. The assigned resources are independent each other, and the inheriting operation in the second phase avoids resuming a search from an unknown origin; therefore, the searching efficiency is improved.

4. Computational results

We conducted a simulation experiment on the farm work scheduling, and the experiment data was mainly obtained from a sugarcane-producing agricultural corporation. The major farm works of cultivating spring-growth sugarcane in 76 farmlands, defined as W_j, involved from the work of the tilling to the harvesting within a predefined time window. The number of available resources required for these works W_1, W_2, ...W_6 was assumed to 2, 1, 1, 1, 1, and 3, respectively. The program was written in the C language, and a Mac Pro with Quad-Core Intel Xeon and 4GB RAM running Mac OS X 10.5 was used as the computing platform. The computation time relied on the parameters of the SA, GA, and time increment in the hybrid Petri nets. The terminate condition was set as $\Gamma < 0.1$ and total computation time ≤ 2 h when $n = 200$, $\alpha = 0.02$ in the SA; population size = 20 and the number of generations = 200 in the GA; and time increment = 10 min in the hybrid Petri nets.

4.1 Optimizing resource assignment and priority list

The impact on evolutionary solution by the particular emphasis on resource assignment was examined in the experiment. Figure 5 shows the contrastive effect on optimizing resource assignment and priority list by the different generation sizes of the GA. The curves are plotted with the current best solution over the computation time. Curve ″gen-100″ represents the evolution process for the high frequency of optimizing resource assignment but a short computation time for the GA iterations. Compared with curve ″gen-100″, curve ″gen-1000″ emphasizes optimizing the priority list in the GA but results in a reduction in the frequency for optimizing resource assignment in the SA at the same computation time. As shown in the figure, not only a fast evolution but also a good solution quality appears in curve ″gen-100″ in the evolution. This reveals that increasing the frequency of optimizing resource assignment is conductive to a fast evolution and convergence, and is more efficient than the optimization computation on the priority list.

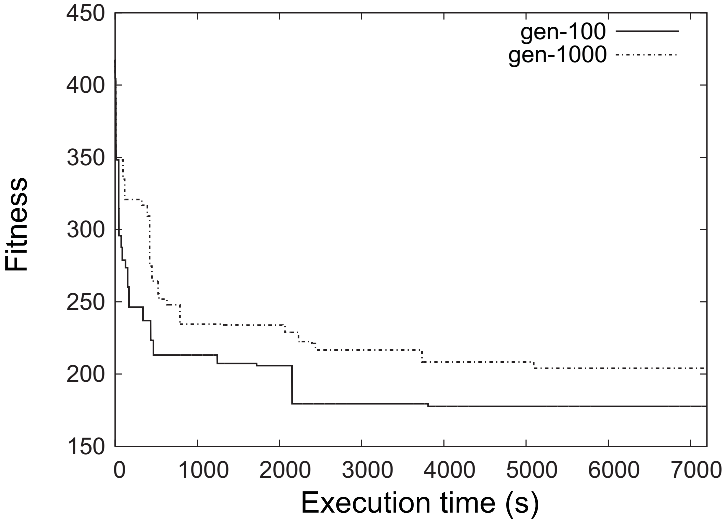


Fig. 5. Evolution based on optimizing resource assignment and priority list

4.2 Inheriting operation

Inheriting the present best work sequence starts with initializing the population for the second scheme of resource assignment. Before the inheriting operation, the procedure gaPls(x) is completed, and the present best work sequence for each resource is ascertained. In general, an inheriting operation can reserve and further improve the solution quality. Accordingly, the inheriting rate for the present best work sequence may impact the evolutionary computation. In order to clarify this, we have investigated the effect of inheriting operation at different inheriting rates and show the comparison of the obtained results in Fig. 6.

In the figure, curve ″cpr-0%″ indicates that the inheriting rate is zero, and the chromosomes in the initial population are entirely randomly generated. Similarly, curve ″cpr-10%″ implies that 10% of the chromosomes are inherited from the best priority list from the previous scheme of resource assignment, and the remaining chromosomes are randomly generated. Although several curves intersect at the beginning of the evolution, the best fitness is finally arranged in the descending order of the inheriting rate. The comparison result demonstrates that both the better solution and evolution speed is obtained by the higher inheriting rate. Conventionally, the inheriting operation for all chromosomes in the initial population may be disadvantageous because of a lack of variety in the chromosomes. Nevertheless, in our experiment, the inherited chromosomes continue to exhibit varieties because the resource assignment is renewed and the partial genes in the chromosomes are generated randomly after the inheriting operation.

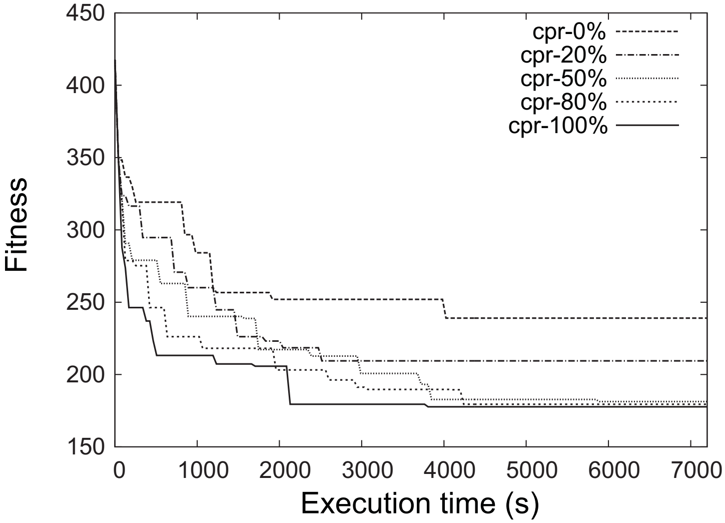


Fig. 6. Effect of inheriting the present best priority list

4.3 Scheduling result

The information on the schedule with the best fitness is listed in Table 2. Resources $R_1 \rightarrow R_2$, $R_7 \rightarrow R_9$ are available to perform W_1 and W_6 in cooperation, respectively. Such cooperative works are performed eight times. The idle time caused by waiting time O_{ij} is very short; and the average rate of utilization for each resource reaches 94.0%, which does not involve the moving time. The schedule length, which is the time period between the start of the first task and the completion of the last task, is applicable to the farm works in a growth cycle because a sugarcane-producing agricultural corporation usually requires time to carry out extra farm works. In order to reserve the time for these extra farm works and the risks such as rain and other uncertainties, we calculate the unscheduled time for each resource. This is very valuable to make an estimate of how much extra works the agricultural corporation can carry out.

Resource	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9
Moving time (h)	14.7	19.7	15.8	25.8	24.7	26.5	14.2	15.8	10.0
Idle time (h)	1.0	0.3	0.0	0.0	0.0	1.8	0.0	0.0	0.0
Number of tasks	47	56	47	76	71	74	47	39	41
Work duration (h)	292.5	289.2	310.8	432.5	405.2	464.2	209.5	207	209.7
Rate of utilization	95.0	93.2	94.9	94.0	93.9	94.3	93.2	92.4	95.2
Unscheduled time	91.2	95.2	79.3	0.0	0.0	8.3	94.0	108.3	101.2
Times of performing work cooperatively					8				
Total area of farmland (hectare)					9.36				
Total amount of work (hectare)					48.9				
Schedule length (h)					2128.8				

Table 2. Information on generated schedule

5. Conclusions and future works

In this study, a hybrid Petri nets model was developed for modeling and formulating the farm work scheduling, and a metaheuristic approach for optimizing the schedule. A comparison on the formulating method between by hybrid Petri nets and mathematical definition was clarified according to their corresponding relations. In the experiment, the computational result revealed that a fast evolution and good solution quality were obtained by emphasizing the resource assignment optimization, and initializing the priority lists inherited from the present best task sequence in the previous resource assignment. Assigning resources first in the two-phase optimization for deadlocks removal considerably improved searching efficiency. Finally, the generated schedule had a high ratio of resource utilization, and it was applicable for devising a practical farm work plan in the agricultural corporation.

We put emphasis on the methodology of formulating and solving the farm work scheduling problem. The generated schedule was for the long-term schedule in a crop growth cycling, but not for the real-time schedule in which the schedule should be calculated in a short time. The proposed model has adequate compatibility and expansibility for modeling the discrete, continuous, concurrent, static, and dynamic events in farming processes. The stochastic event such as the data of weather can be also formulated on the Petri nets model by associating a time vector with a probability distribution to transitions. Although such environmental changes or breaks were ignored in the experiment, they will be considered in the real-time scheduling on a mobile device in our continuing work.

With respect to the computation time, the maximum time was required for the GA iterations and the simulation computation for the firing of the hybrid Petri nets. We are considering some approaches to reduction of computation time such as improved the crossover of the GA operations, parallel computing and an alterable strategy for simulating the firing operations of the hybrid Petri nets.

6. References

Arjona, E., Bueno, G. & Salazar, L. (2001). An activity simulation model for the analysis of the harvesting and transportation systems of a sugarcane plantation, *Computers and Electronics in Agriculture* 32: 247–264.

Astika, I. W., Sasao, A., Sakai, K. & Shibusawa, S. (1999). Stochastic farm work scheduling algorithm based on short range weather variation, *Journal of the Japanese Society of Agricultural Machinery* 61: (2)157–164, (3)83–94, (4)141–150.

Balasubramanian, J. & Grossmann, I. E. (2003). Scheduling optimization under uncertainty - an alternative approach, *Computers and Chemical Engineering* 27(4): 469 – 490.

Bassett, M. H., Pekny, J. F. & Reklaitis, G. V. (1997). Using detailed scheduling to obtain realistic operating policies for a batch processing facility, *Industrial Engineering and Chemical Research* 36: 1717–1726.

Chen, L. H. & McClendon, R. W. (1985). Soybean and wheat double cropping simulation model, *Transaction of the ASAE* 28(1): 65–69.

Daikoku, M. (2005). Development of computer-assisted system for planning work schedule of paddy and transplanting in distributed fields, *Japanese Journal of Farm Work Research* 40(4): 210–214. (In Japanese).

Ghaeli, M., Bahri, P. A. & Lee, P. L. (2006). Timed arc hybrid Petri nets based scheduling of mixed batch/continuous plants, *Proceedings of the 17th IMACS World Congress*.

Guan, S., Nakamura, M., Shikanai, T. & Okazaki, T. (2008). Hybrid petri nets modeling for farm work flow, *Computers and Electronics in Agriculture* 62(2): 149–158.

Guan, S., Nakamura, M., Shikanai, T. & Okazaki, T. (2009). Resource assignment and scheduling based on a two-phase metaheuristic for cropping system, *Computers and Electronics in Agriculture* 66(2): 181–190.

Haffar, I. & Khoury, R. (1992). A computer model for field machinery selection under multiple cropping, *Computers and Electronics in Agriculture* 7: 219–229.

Higgins, A. & Davies, I. (2005). A simulation model for capacity planning in sugarcane transport, *Computers and Electronics in Agriculture* 47: 85–102.

Janak, S. L. & Floudas, C. A. (2006). Production scheduling of a large-scale industrial batch plant. II. Reactive scheduling, *Industrial and Engineering Chemistry Research* 45(25): 8253–8269.

Lal, H., Peart, R. M., Jones, J. W. & Shoup, W. D. (1991). An object-oriented field operation simulator in PROLOG, *Transaction of the ASAE* 34(3): 1031–1039.

Li, Z. & Ierapetritou, G. (2008). Process scheduling under uncertainty: Review and challenges, *Computers and Chemical Engineering* 32: 715–727.

Lin, X., Janak, S. & Floudas, C. (2004). A new robust optimization approach for scheduling under uncertainty: I. Bounded uncertainty, *Computers and Chemical Engineering* 28: 1069–1085.

Murata, T. (1989). Petri nets: Properties, analysis and applications, *Proceedings of the IEEE* 77(4): 541–580.

Nanseki, T. (1998). Operations of FAPS97: A decision support system for evaluating agricultural technology and farm planning, *Miscellaneous Publication Tohoku National Agricultural Experiment Station* 21: 1–119. (In Japanese).

Nanseki, T., Matsushita, S. & Ikeda, M. (2003). A farming-systems database for farm planning, *Agricultural Information Research* 12: 133–152. (In Japanese).

Sadrieh, S., Ghaeli, M., Bahri, P. & Lee, P. (2007). An integrated Petri nets and GA based approach for scheduling of hybrid plants, *Computers in Industry* 58: 519–530.

Santiago-Mozos, R., Salcedo-Sanz, S., DePrado-Cumpli, M. & Bousono-Calzon, C. (2005). A two-phase heuristic evolutionary algorithm for personalizing course timetables: A case study in a Spanish university, *Computers and Operations Research* 32(7): 1761–1776.

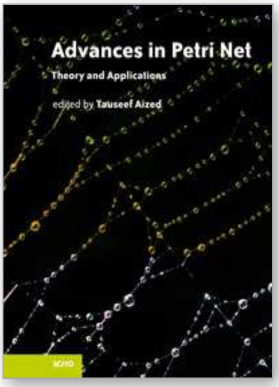
Suliman, S. (2000). A two-phase heuristic approach to the permutation flow-shop scheduling problem, *International Journal of Production Economics* 64: 143–152.

The Ministry of Agriculture, Forestry and Fisheries of Japan (2006). Japan's post-war agricultural land reform and subsequent agricultural land system, *National Report for the International Conference on Agrarian Reform and Rural Development* .

Till, J., Sand, G., Urselmann, M. & Engell, S. (2007). A hybrid evolutionary algorithm for solving two-stage stochastic integer programs in chemical batch scheduling, *Computers and Chemical Engineering* 31: 630–647.

Tsai, Y. J., Jones, J. & Mishoe, J. (1987). Optimizing multiple cropping systems: A systems approach, *Transaction of the ASAE* 30(6): 1554–1561.

Wang, J. (2004). A fuzzy robust scheduling approach for product development projects, *European Journal of Operational Research* 152: 180–194.



Advances in Petri Net Theory and Applications
Edited by Tauseef Aized

ISBN 978-953-307-108-4
Hard cover, 220 pages
Publisher Sciyo
Published online 27, September, 2010
Published in print edition September, 2010

The world is full of events which cause, end or affect other events. The study of these events, from a system point of view, is very important. Such systems are called discrete event dynamic systems and are of a subject of immense interest in a variety of disciplines, which range from telecommunication systems and transport systems to manufacturing systems and beyond. There has always been an intense need to formulate methods for modelling and analysis of discrete event dynamic systems. Petri net is a method which is based on a well-founded mathematical theory and has a wide application. This book is a collection of recent advances in theoretical and practical applications of the Petri net method and can be useful for both academia and industry related practitioners.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Senlin Guan, Morikazu Nakamura and Takeshi Shikanai (2010). Hybrid Petri Nets and Metaheuristic Approach to Farm Work Scheduling, Advances in Petri Net Theory and Applications, Tauseef Aized (Ed.), ISBN: 978-953-307-108-4, InTech, Available from: http://www.intechopen.com/books/advances-in-petri-net-theory-and-applications/hybrid-petri-nets-and-metaheuristic-approach-to-farm-work-scheduling

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.