J. Soc. Korea Ind. Syst. Eng Vol. 40, No. 1 : 79-86, March 2017
https://doi.org/10.11627/jkise.2017.40.1.079

ISSN : 2005-0461(print)
ISSN : 2287-7975(online)

Optimizing Work-In-Process Parameter using Genetic Algorithm

Jungseop Kim* · Jiyong Jeong** · Jonghwan Lee***†

*Graduate School of Consulting, Kumoh National Institute of Technology
**KMAC
***School of Industrial Engineering, Kumoh National Institute of Technology

유전 알고리즘을 이용한 Work-In-Process 수준 최적화

김정섭* · 정지용** · 이종환***†

*금오공과대학교 컨설팅대학원
**KMAC
***금오공과대학교 산업공학부

This research focused on deciding optimal manufacturing WIP (Work-In-Process) limit for a small production system. Reducing WIP leads to stable capacity, better manufacturing flow and decrease inventory. WIP is the one of the important issue, since it can affect manufacturing area, like productivity and line efficiency and bottlenecks in manufacturing process. Several approaches implemented in this research. First, two strategies applied to decide WIP limit. One is roulette wheel selection and the other one is elite strategy. Second, for each strategy, JIT (Just In Time), CONWIP (Constant WIP), Gated Max WIP System and CWIPL (Critical WIP Loops) system applied to find a best material flow mechanism. Therefore, pull control system is preferred to control production line efficiently. In the production line, the WIP limit has been decided based on mathematical models or expert's decision. However, due to the complexity of the process or increase of the variables, it is difficult to obtain optimal WIP limit. To obtain an optimal WIP limit, GA applied in each material control system. When evaluating the performance of the result, fitness function is used by reflecting WIP parameter. Elite strategy showed better performance than roulette wheel selection when evaluating fitness value. Elite strategy reach to the optimal WIP limit faster than roulette wheel selection and generation time is short. For this reason, this study proposes a fast and reliable method for determining the WIP level by applying genetic algorithm to pull system based production process. This research showed that this method could be applied to a more complex production system.

Keywords : Genetic Algorithm, Work-In-Process Limit, Optimization

1. 연구배경 및 방법

Pull System의 대표적인 예로는 JIT System이 있다. 도요다자동차는 전쟁으로 인해 파괴된 산업시설 대신에 새로운 생산방식을 도입할 필요가 있었다. 이 당시 Henry Ford의 컨테이너 방식은 표준화된 모델을 대량으로 생산하는 데 적합한 생산방식이었다. 하지만 컨테이너 방식은 수많은 단점들을 보유하고 있었다. 많은 재고와 재공품들, 제조라인의 불균형, 과잉생산 등 여러 문제가 있었기 때문에 도요다자동차는 "생산은 소비가 있을 때에만 이

Received 25 November 2016; Finally Revised 9 March 2017; Accepted 10 March 2017
† Corresponding Author : shirjei@kumoh.ac.kr

루어져야 한다"라는 아이디어를 내어 이를 생산현장에 도입했던 것이다. 간단히 JIT는 "필요한 시기에, 필요한 양만큼, 필요한 제품을 정확히 생산하자"라는 기존 제조업의 관리에 대한 관념을 깨뜨린 새로운 개념의 관리철학이라고 할 수 있다. 따라서 과잉생산, 대기시간, 재공품/재고 등을 획기적으로 줄임으로써 고객이 원하는 제품을 짧은 생산리드타임에 생산, 판매하는 것을 가능케 한다. 하지만 JIT에서는 재공품 또는 완제품을 이동시키는 신호의 역할일 하는 칸반(Kanban)의 수가 많아져서 관리에 힘든 점이 있다. 이것을 보완하기 위해 CONWIP과 Gated MaxWIP, Critical WIP loops 등 여러 가지 Pull System 기반의 생산 시스템들이 제시되었다. 이 같은 모델들은 수리적 모델이나 전문가의 견해에 의해 공정의 WIP 수준을 결정하게 된다. 하지만 수리적 모델은 공정의 수가 많아지거나 고려해야 할 변수들이 많아진다면 그 값을 계산하기에 어려움이 따른다. 따라서, 본 연구에서는 Pull System 기반의 생산 시스템들에 대해서 최적의 WIP 수준을 손쉽게 구할 수 있는 Generic Algorithm의 적용 방법을 제시한다. Matlab을 이용하여 생산공정의 Model을 구현하고 Genetic Algorithm을 실시간 연동이 가능하도록 하여 최적 해를 손쉽게 추출해 낼 수 있다는 것을 제안한다.

2. 생산 시스템

2.1 JIT System

Pull System 전략의 가장 대표적인 생산방식은 일본 도요다자동차의 JIT System이다. JIT System의 기본적 취지는 생산시스템의 모든 형태의 낭비를 최소화 하고, 그러한 활동을 지속하여, 모든 불합리 요소를 제거하여 기업 내 가용 자원들의 운영 효율을 향상하고 최적화 하는 것을 목적으로 한다. JIT System은 크게 두 가지 특징으로 요약 할 수 있다. 필요한 만큼, 필요한 시간에 생산하는 '적시생산'과 전 공정 후 공정 사이에 최소 버퍼를 운영하며, 이 버퍼로부터 고객공정인 후 공정이 사용하여 소비된 양만큼 만을 다시 전 공정에서 생산하는 '적량생산'을 원칙으로 한다.

일반적으로 Pull System 방식으로 생산라인 운영전략을 적용하는 경우에 Push System 방식에 비해 생산량의 감소현상 발생의 위험성이 증가한다. 예상하지 못한 돌발적인 생산의 변동 발생에는 대응하기 어렵다는 문제점을 갖고 있기 때문이다[1].

또한, 공급자가 부품을 제대로 공급하지 못할 때 전체 시스템이 중단될 우려가 있으며 각 공정마다 각기 다른 칸반 개수를 사용하여 많은 양의 매개변수로 관리하기가 어렵고, 작업자에게는 많은 스트레스를 준다.

2.2 CONWIP System

CONWIP은 Constant WIP을 뜻하고 있다. 따라서 공정 내 WIP을 일정하게 유지하면서 생산하는 방식이다. 마지막 공정에서 생산물이 하나 완성되면 완제품에 붙어있던 칸반이 첫 공정의 작업지시서가 되어 새로운 제품을 생산하게 되는 공정흐름 운영방법이다.

마치, 고객이 시스템 내를 떠날 수 없는 폐쇄된 대기행렬과 같이, 공정이 종료된 제품이 다시 처음 공정으로 돌아와서 순환하는 것과 같은 형식으로 운영된다[8].

초기 CONWIP System에 대한 연구는 Spearman et al. [8]의 연구를 들 수 있다. 이 중에서도 Spearman은 처음 Kanban System이 생산공정에서 성공적으로 적용될 수 있었던 이유가 수요가 발생할 때마다 선행 공정에서 견인하는 Pull System 방식에서 기인되는 것이 아니라 재공품재고를 일정한 수준 이하로 유지, 통제하는 방식에서 기인된 것임을 제시하였다[3]. 이러한 개념은 한층 발전되어 모든 공정에서 견인방식을 사용하지 않더라도 Kanban System의 장점을 살리고, 단점을 보완하는 CONWIP System을 제안하기에 이르렀다.

따라서, CONWIP System은 JIT System의 단점을 보완하면서 파생된 생산공정의 운영기법이다. 또한, 하나의 공정을 복수의 CONWIP System으로 나누어 제어를 하는 경우를 Multi-shop CONWIP이라고 한다. 이것을 세분화하여 하나의 공정마다 CONWIP System으로 WIP 수준을 제어한다면 JIT System과 동일한 흐름관리 방법이 된다[3].

2.3 Gated MaxWIP System

Gated MaxWIP System은 CONWIP System의 공정 시작지점에 Pace machine 또는 Gate Machine이라 하는 새로운 공정이 추가된 시스템이다. Pace Machine을 통과한 원자재나 부품은 CONWIP system의 공정 진행과 동일한 방법으로 작업이 진행된다.

Gated MaxWIP을 공정에 적용할 때 Pace Machine의 선택사항에 따라 WIP의 상태가 변하게 되는데, 그것은 <Table 1>과 같다.

<Table 1> WIP State According to the Selection of Pace Machine[5]

Options	Contents
1	Always has one WIP in the pace machine.
2	No WIP in the pace machine.
3	Only one WIP in the pace machine when production line is in the state of blocked.

2.4 Critical WIP Loops System

CWIPL System도 기본적으로 CONWIP과 같은 방식으로 원재료나 부품을 공정에 투입하게 된다. 하지만 CWIPL은 산출율에 지대한 영향을 미치는 마지막 공정의 버퍼(buffer)를 관리하여 산출율을 높이고 있다. 산출율을 구하는 공식은

$$TP = u_n(1-P_n)$$

이다. u_n은 마지막 공정의 생산능력, P_n은 마지막 공정의 총 유휴시간의 비율을 의미한다. CWIPL System을 적용하는 공정이 균형라인으로써 생산능력이 변하지 않는다고 한다면 P_n을 줄이는 것이 산출율을 높일 수 있는 유일한 방법이라 할 수 있다[6].

CWIPL System에서는 Dispatching Rule의 <Rule 1>과 <Rule 2>를 동시에 만족 했을 경우에만 공정에 원재료가 투입될 수 있다. Rule을 통해 첫 번째 공정의 WIP을 관리할 수 있고, 따라서 사이클타임이 감소하는 효과를 가져올 수 있다.

<Table 2> Dispatching Rule of WIP in CWIPL[4]

Rules	Contents
Rule 1	When WIP in the last process is less than WIP parameter and the first process is idle state.
Rule 2	WIP level in the entire production line is less than assigned WIP level and the first process is idle state.

3. 유전 알고리즘

3.1 유전 알고리즘의 개념

유전 알고리즘은 자연계에 있어서 생물진화과정 현상에 기반을 둔 최적화 기법으로 자연선택의 원리와 다윈의 적자생존의 이론을 기본개념으로 하여 컴퓨터 시뮬레이션으로 모방한 확률적 탐색, 최적화 학습에 대한 하나의 기법이다. 이른바 자연도태와 유전 현상의 메커니즘을 단순화한 meta heuristics의 하나이며, 개체라 불리는 염색체의 집합이 외부환경에 적응하듯이 다음에 나타내는 규칙에 근거하는 집단의 구성을 세대마다 생성시키는 것이다.

유전 알고리즘을 조합한 최적화 문제의 한가지 해법으로 <Rule 1>을 확률적 탐색법, <Rule 2>을 경험적 탐색 법이라고 볼 수 있으며 유전 알고리즘은 두 개의 측면을 모두 갖는 Meta heuristics 기법의 하나이다[7, 9].

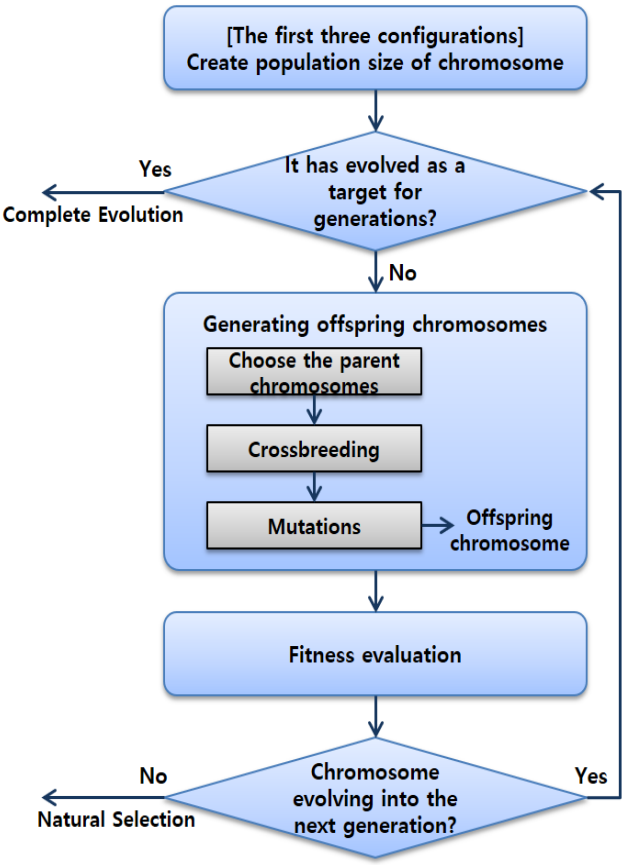
<Table 3> Rules of Genetic Algorithm

Rules	Contents
Rule 1	The more likely to survive when the objects have high suitability(natural selection).
Rule 2	Use the objects of previous generations to produce a new object of the next generation by genetic manipulation (genetic phenomenon).

3.2 유전자 진화과정

유전 알고리즘의 최적 해를 찾기 위한 탐색의 원리는 자연계의 진화를 지배하는 적자생존의 원리와 유전자의 교환에 의한 세대 교배 및 진화의 과정을 통해 이루어지며 적자생존을 구현하기 위하여 최적화 목표 함수 값을 해석하고 우수한 부모 해를 보전 시킨다. 또한 우수한 부모들 간의 재조합과 돌연변이를 이용하여 자손 해를 생성한다. 이처럼 세대교체가 이루어질 때 적자생존의 원리는 확률적 이론에 입각한 차등적인 참여 기회를 부여한다.

따라서, 유전 알고리즘은 확률적 탐색이나 학습 및 최적화를 위한 기법의 한 가지라고 할 수 있다.



<Figure 1> Genetic Evolution

<Figure 1>에서 보는 것과 같이 교배와 돌연변이를 통해 얻어진 자손들은 선택과 보다 나은 개체로의 편향 등으로 다음 세대에서의 생존을 위해 노력한다. 세대의 교체가 반복되면서 염색체가 지닌 결과값들이 수렴 할 때, 염색체는 최적의 값을 지닌다고 기대할 수 있다[2].

4. Simulation Model 개발

4.1 프로세스 정의

본 연구는 최적 WIP 수준을 찾기 위해 사용될 프로세스는 다음과 같이 정의하였다.

<Table 4> Defining the Process

Outline	Contents
Common-ness	• A single line and single product. • Balanced product lines. • Supply and demand are always existed. • No failure process. • Working parts are considered as a WIP.
Gated MaxWIP	• When the production line is in the blocked state there is only one WIP in the pace machine.

모든 공정은 10개로 구성되며 균형라인으로 이루어져 있고, Process Time은 정규분포를 사용하고 있다. Gated MaxWIP System은 <Table 1>의 <Option 3>을 따르고 있는데, 선행연구 “Decentralized WIP-oriented manufacturing control(DWIP)”[5]에서도 <Option 1>과 <Option 2>의 절충안이 <Option 3>이라고 밝히고 있기에 본 연구에서도 <Option 3>을 적용하고 있다.

4.2 프로세스 구현

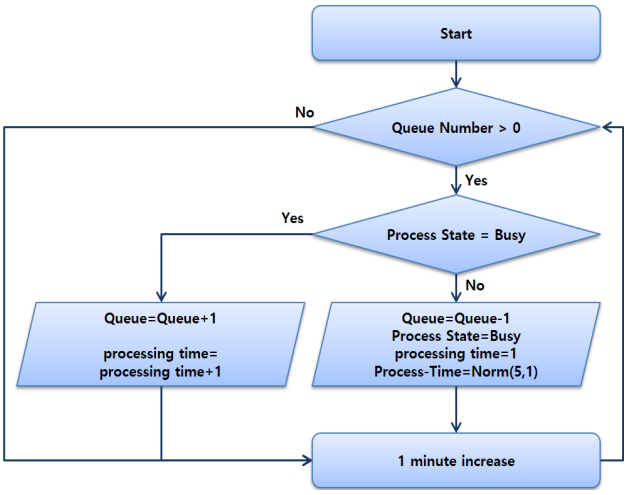
유전알고리즘이 적용 된 프로그램 내부에는 공정의 상태 값들을 저장하는 결과데이터 Table이 존재하며, Table과 유전알고리즘이 유기적으로 데이터를 주고받으며 최적의 WIP 수준을 도출할 수 있도록 프로그램을 구현하였다.

생산공정의 결과 데이터는 <Figure 2>와 같이 도출하였다. 총 공정은 10개이나 <Figure 2>에는 1에서 6공정까지의 일부 raw data만 나타내었다. Time의 단위는 분(min)을 사용하고, 각 프로세스 당 대기행렬(Waiting Number : w)과 작업유무(Busy : b)의 상태를 나타내는 열 값이 존재한다. 대기행렬에는 각 프로세스의 대기열에 대기하고 있는 작업물의 수가 표시되고, 작업유무를 나타내는 행렬에는 작업 시 ‘1’, 유휴 시 ‘0’의 값이 표시되도록 한다.

Time	Process											
	w1	b1	w2	b2	w3	b3	w4	b4	w5	b5	w6	b6
1	1	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0	0	0
3	2	1	0	0	0	0	0	0	0	0	0	0
3	2	1	0	0	0	0	0	0	0	0	0	0
4	2	1	0	0	0	0	0	0	0	0	0	0
4	2	1	0	0	0	0	0	0	0	0	0	0
5	3	0	1	0	0	0	0	0	0	0	0	0
5	2	1	0	1	0	0	0	0	0	0	0	0
6	2	1	0	1	0	0	0	0	0	0	0	0
6	2	1	0	1	0	0	0	0	0	0	0	0
7	3	0	1	0	1	0	0	0	0	0	0	0
7	2	1	0	1	0	1	0	0	0	0	0	0
8	2	1	0	1	0	1	0	0	0	0	0	0
8	2	1	0	1	0	1	0	0	0	0	0	0
9	2	1	0	1	0	1	0	0	0	0	0	0
9	2	1	0	1	0	1	0	0	0	0	0	0
10	3	0	1	1	0	0	1	0	0	0	0	0
10	2	1	1	1	0	0	0	1	0	0	0	0

<Figure 2> Total Process Data

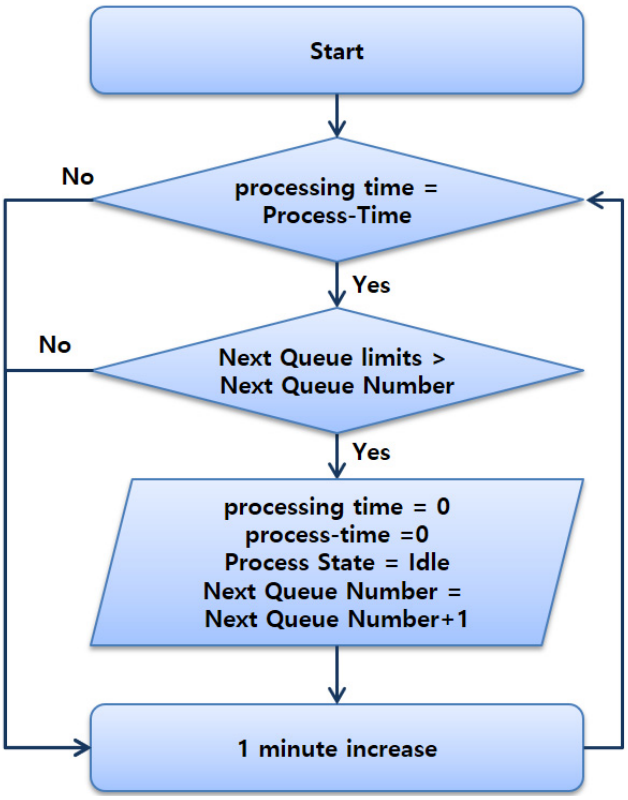
Time은 가상의 “준비시간”과 “실제 작업시간”으로 이루어져 있는데, 시간에 따라 생산공정 결과표를 작성하는 방식이 다르게 적용된다.



<Figure 3> Preparation time Flow-Chart

<Figure 3>의 흐름도에서 보는 것과 같이 “준비시간”에서는 “실제 작업시간”이 일어나기 전, 프로세스의 대기 행렬과 작업유무 행렬에 작업들의 이동 과정을 나타내고, 이를 다음 “실제 작업시간”에 반영할 수 있도록 상태 값들을 변경한다.

Total Time Data에는 작업물이 공정에서 작업 시간(In Progress)과 공정에 할당된 작업시간(Process Time)을 기록하고 있다.



<Figure 4> Actual time Flow-Chart

Time	Time Data											
	i1	p1	i2	p2	i3	p3	i4	p4	i5	p5	i6	p6
1	0	0	0	0	0	0	0	0	0	0	0	0
1	1	4	0	0	0	0	0	0	0	0	0	0
2	1	4	0	0	0	0	0	0	0	0	0	0
2	2	4	0	0	0	0	0	0	0	0	0	0
3	2	4	0	0	0	0	0	0	0	0	0	0
3	3	4	0	0	0	0	0	0	0	0	0	0
4	3	4	0	0	0	0	0	0	0	0	0	0
4	4	4	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
5	1	2	1	2	0	0	0	0	0	0	0	0
6	1	2	1	2	0	0	0	0	0	0	0	0
6	2	2	2	2	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
7	1	3	1	2	1	3	0	0	0	0	0	0
8	1	3	1	2	1	3	0	0	0	0	0	0
8	2	3	2	2	2	3	0	0	0	0	0	0
9	2	3	2	2	2	3	0	0	0	0	0	0
9	3	3	3	2	3	3	0	0	0	0	0	0

<Figure 5> Total Time Data

총 공정은 10개이나 <Figure 5>에는 1에서 6 공정까지의 일부 raw data만 나타내었다.

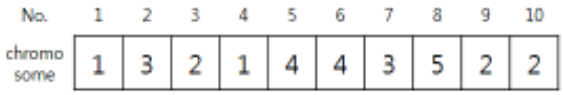
Total Process Data에서 각 프로세스에 작업들이 투입되어 프로세스가 Busy 상태로 변경되면, Total Time Data 결과표의 각 프로세스에는 재공품을 가공한 시간(In Progress : i)이 Time 변화에 따라 누적되며, 각 프로세스의 가공시간(Process Time : p)만큼 가공된 후(i = p) 다음 프로세스의 대기열로 이동된다.

재공품이 마지막 프로세스를 거쳐 최종 생산품이 될 때, 각 생산품의 생산 주기를 기록하여 Fitness Value를 구할 때 사용한다.

4.3 유전 알고리즘의 적용

4.3.1 유전자 표현

Simulation Model에 유전 알고리즘을 적용하기 위해 <Figure 6>과 같은 Gene Type의 염색체 집단을 생성한다. 염색체 내의 각각의 유전자는 프로그램 상에서 각 공정의 WIP 수준을 의미하도록 설계하였다.



<Figure 6> Examples of Chromosome

하나의 염색체 값은 1~5사이의 무작위 값을 가지는 1×10행렬로 표현된다. 본 연구에서의 집단 크기는 10으로 정하고 있기 때문에 초기 염색체 집단의 행렬은 10×10행렬이 된다.

<Table 5> The Role of Chromosome

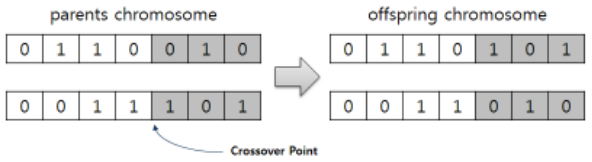
Product System	The role of chromosomal
JIT System	WIP level of each chromosome per process.
CONWIP System	The sum of the chromosome is the total process WIP level.
Gated MaxWIP System	
Critical WIP loops System	• the sum of 1~9 chromosome is the WIP level in the total process. • 10^{th} chromosome is the size of WIP parameter.

염색체는 <Table 5>에서 정리된 것처럼 생산 시스템에 따라 역할이 나뉘게 된다. JIT System에서는 염색체 하나당 각 프로세스의 WIP 수준을 의미한다. CONWIP과 Gated MaxWIP System에서는 모든 염색체 합이 생산 시스템의 전체 WIP 수준을 나타내며, CWIPL System에서는 1에서 9번의 염색체의 합이 전체 생산 시스템의 WIP 수준을 나타내고 마지막 10번 염색체는 WIP parameter를 의미하게 된다.

이로서, 전문가의 견해에 의해 결정되었던 WIP 수준과 WIP parameter값을 유전자 알고리즘을 통해 구할 수 있도록 표현을 하게 된다.

4.3.2 유전자 조작

본 연구에서의 염색체 값들은 하나의 유전자 내에서 중복된 값을 가질 수 있기 때문에, 유전 알고리즘에서 가장 보편적으로 사용되는 일점 교배와 단순돌연변이 방법으로 돌연변이를 일으켜 준다.



<Figure 7> Example of Crossbreeding

일점 교배는 교차점을 기준으로 두 개의 부모 염색체의 값을 교환하는 방식으로 교배를 일으킨다.

<Figure 8> An Example of a Simple Mutation

돌연변이는 교배와 다르게 하나의 부모 염색체에서 임의의 유전자를 선택하여 값을 변경해 주는 방법으로 새로운 자식 염색체를 생성하게 된다.

<Table 6> Minimum Value of Chromosomes

System	division	Minimum
JIT	Each chromosome	1
CONWIP Gated MaxWIP Critical WIP loops	Each chromosome	0
	The sum of chromosomes	1
CWIPL WIP parameter	10th Chromosome	1

본 연구의 유전자 조작과정에서 <Table 6>의 조건을 만족할 수 있도록 값을 보정해 줄 필요가 있다. JIT System의 경우 각 염색체의 값이 각 공정의 WIP 수준을 나타내고 있다. 따라서 염색체의 최소값은 1의 값을 가져야 한다. 그 외의 시스템들은 각 염색체의 최소값은 0의 값을 가질 수 있으나, 전체의 합은 1이 최소가 될 수 있도록 보정을 해 주어야 한다. 마지막으로 CWIPL의 WIP parameter 값이 최소 1이 되어야 하므로 CWIPL의 10번 염색체의 최소값이 1이 될 수 있도록 보정해 주어야 한다.

4.3.3 평가함수

Simulation 과정을 거친 후 전체 WIP 수준의 합, WIP 수준과 실제 WIP의 차이, 제품의 평균 생산시간이 결과값으로 도출된다. 이 세 가지의 결과값을 이용하여 평가함수를 다음과 같이 정의하였다.

$$F(p) = \frac{100{,}000}{((10(wipN)+3(wipD)+2(averT))}$$

$F(p)$ = 평가함수 값
$wipN$ = WIP수준의 합
$wipD$ = WIP수준과 실제 WIP의 차
$averT$ = 제품의 평균 생산시간

각각의 Parameter에 곱해진 상수는 문제를 해결함에 있어서 Parameter가 가지고 있는 가중치를 표현한 것이다. 분자의 상수 값은 각각의 Parameter 값이 작아질수록 평가함수의 값이 커지도록 하는 효과를 가지는데, 이 때 결과치를 좀 더 명확하게 구분하여 확인하기 위해 100,000이라는 상수값을 적용하였다. 위의 식은 CWIPL System을 제외한 시스템에서만 적용되었다.

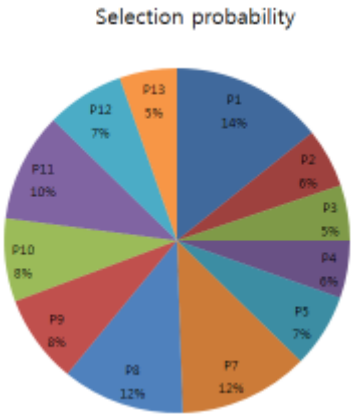
CWIPL System에서는 WIP parameter의 수준도 고려해야 하기 때문에 다음과 같은 식으로 평가함수를 정의하였다.

$$F(p) = \frac{100{,}000}{((10(wipN)+3(wipD)+2(averT)+WIPparameter)}$$

평가 함수에 WIP parameter의 값을 반영함으로써 유전자 알고리즘을 통한 적정 수준의 WIP parameter를 구할 수 있다.

4.3.4 선택

가장 흔히 쓰이는 염색체 선택 기법인 룰렛 휠 선택을 사용한다. 평가 함수를 통해 구해진 값이 클수록 룰렛 휠 선택에서 선택될 확률은 높아진다. 따라서 집단에서 발달이 덜 된 유전자(평가 함수 값이 낮은 유전자)가 생태계에서 자연도태 되는 현상을 프로그램으로 구현할 수 있다.



<Figure 9> An Example of a Roulette Wheel Selection

또한, 우수한 유전자를 항상 다음세대로 진화 시킬 수 있는 엘리트 전략(Elite Strategy) 기법을 적용해 보았다. 룰렛 휠 선택 기법에 엘리트 전략 기법의 적용 유/무에 따라 최적 WIP 수준을 찾아내는 과정을 비교해 볼 것이다.

5. 실험 및 결과분석

본 연구에서 유전자 알고리즘을 이용한 생산 시스템의 최적 WIP 수준을 결정하는 결정 모형을 구축하고, 결정 모형에 의한 결과치인 WIP Level, 최저 WIP 수준이 처음 발생한 시점, Fitness Value를 도출하였다.

모든 생산 시스템의 WIP 수준 결정 모형에 룰렛 휠 선택과 엘리트 전략 두 가지 방법을 적용하여 실험하였다. 염색체의 집단 크기는 10개, 최대 세대수는 500개로 설정하였다.

<Table 7> The Whole System Results

Method	Fitness Value	Generation	WIP Level	WIP Parameter
JIT Roulette	698.02	320	10	
JIT Elite	701.16	171	10	
CONWIP Roulette	1255.12	455	4	
CONWIP Elite	1324.10	166	3	
Gated Roulette	1081.07	319	19	
Gated Elite	1207.01	230	3	
CWIPL Roulette	745.42	399	9	5
CWIPL Elite	1299.41	214	3	1

<Table 7>은 전체 데이터를 정리해 놓은 표이다. <Table 7>의 데이터를 볼 때 같은 생산시스템의 결과치에서 엘리트 전략을 사용하였을 경우가 Fitness Value가 우수한 것을 확인할 수 있다.

결과적으로 WIP의 수준도 낮은 것을 확인 할 수 있다. 또한, 엘리트 전략은 최적의 WIP 수준이 발생한 시점(Generation)도 룰렛 휠 선택에 비해 빠르다는 것을 알 수 있다.

기존에는 Pull System의 최적 WIP 수준을 계산하기 위해 수리적 모델을 사용하거나 전문가의 견해를 통해 값을 도출 하였는데, 본 논문에서 제시한 바와 같이 유전자 알고리즘을 사용한다면 빠른 시간에 최적 값이라고 볼 수 있는 결과를 도출 할 수 있다.

6. 결론 및 향후 과제

지금까지 생산 시스템의 WIP 수준과 WIP parameter는 수리적 모형이 아닌 주로 생산 시스템 전문가의 견해에 의해 결정되었다. 따라서, 생산 시스템을 다룬 연구 논문들은 시스템의 최적 WIP 수준을 결정할 수 없기에 임의의 값들로 WIP 수준을 결정하여 생산 시스템간의 성능만을 비교 분석하고 있다.

본 연구는 이와 다르게 전문가의 견해에만 의존해야 하는 WIP 결정 문제를 유전 알고리즘을 적용한 생산 시스템 WIP 수준의 결정 모델을 이용하여 최적 WIP 수준을 결정해 보았다. JIT, CONWIP, Gated MaxWIP, CWIPL System 모두 유전알고리즘을 통해 WIP 수준을 결정할 수 있었다.

본 연구에서 Matlab Software로 작성된 유전자 알고리즘은 사용자가 손쉽게 생산 시스템 모델과 평가함수를 수정할 수 있다. 따라서, 사용자가 목표로 하는 결과값을 도출 할 수 있도록 여러 방면으로 수정하여 사용이 가능하다.

본 연구에서는 WIP 수준의 크기가 평가 함수에 가장 큰 영향을 미쳤지만, 예를 들어, 납기일을 중요시하는 모델에 적용 한다면 총 생산 시간에 큰 가중치를 주거나, 특정 생산 시간을 초과할 경우에는 페널티 함수(Penalty Function)를 적용하여 평가함수값에 페널티를 주어 유전자가 자연도태 되도록 하면 된다. 또한 유전자 알고리즘의 선택 기법에서 룰렛 휠 선택 기법과 엘리트 전략을 각각 적용하여 결과를 도출하고 비교해 보았는데, 엘리트 전략을 적용하였을 경우가 평가함수값이 가장 높은 결과들을 도출 하였고, 최적 WIP 수준도 빠른 세대에서 도출 할 수 있다는 것을 확인할 수 있었다. 따라서, 엘리트 전략이 유전자 알고리즘에 좀 더 나은 결과를 줄 수 있다는 이론을 입증 해볼 수 있었다.

결론적으로 본 연구는 다양한 생산 시스템과 공정의 상황을 고려한 시뮬레이션 모델에 유전자 알고리즘을 적용하여 최적 WIP 수준을 결정하는 것이 가능함을 보이고 있다. 또한 사용자가 만드는 평가 함수에 따라 다양한 결과값을 도출 할 수 도 있다는 것을 제시한다. 하지만 평가 함수를 작성하는 수학적인 계산식이 없고 사용자 임의의 설정에 결과값이 크게 변동하는 부분에 대해서는 더욱 다양한 연구가 지속되어야 할 것이다.

Acknowledgement

This paper was supported by Kumoh National Institute of Technology.

References

[1] Bayanat, B. and Dallery, Y., A Unified View of Product-from Approximation Techniques for General Closed Queueing Networks, *Performance Evaluation*, 1993, Vol. 18, No. 3, pp. 205-224.

[2] Han, S.-M., Scheduling for Satellite Mission Using Genetic Algorithm, Graduate School of Pusan National University, [Master's thesis], 2009, pp. 15-64.

[3] Kim, J.M., A study on the decision of the optimized production control policy using the simulation, Ulsan University, [Master's Thesis], 2008, pp. 1-4.

[4] Kim, W., Comparison and Analysis of performance values for CONWIP Systems and Kanban Systems, Graduate School of Kyung Hee University, [Master's thesis], 2007, pp. 1-6

[5] Lodding et al., Decentralized WIP-oriented manufacturing control(DWIP), *Production Planning & Control*, 2003, Vol. 14 No. 1, pp. 42-54.

[6] Park, H.J., Comparison of Control Policies for a Flow Line Manufacturing System with Bottleneck Stations, Graduate School of Kyung Hee University, [Master's thesis], 2011, pp. 1-5.

[7] Park, S.G. and Ha, C.H., Performance Comparison between Material Flow Control Mechanisms Using Simulation, *Journal of Society of Korea Industrial and Systems Engineering*, 2012, Vol. 35, No. 1, pp. 115-223.

[8] Spearman, M., Woodruff, D., and Hopp, W., CONWIP : a pull alternative to kanban, *International Journal of Production Research*, 1990, Vol. 28 No. 5, pp. 879-894.

[9] Yoon et al., A Genetic Algorithm and Discrete-Event Simulation Approach to the Dynamic Scheduling, *Journal of Society of Korea Industrial and Systems Engineering*, 2013, Vol. 36, No. 4, pp. 116-122.

ORCID

Jungseop Kim | http://orcid.org/0000-0003-0385-4522
Jiyong Jeong | http://orcid.org/0000-0002-0974-6808
Jonghwan Lee | http://orcid.org/0000-0001-9630-5236