**PUSH & PULL**

**< Experiment result & analysis >**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. Report on simulation result for each model.   |  |  |  |  | | --- | --- | --- | --- | |  | Order waiting time | product CT | product WIP | | PUSH | 2.1796 | 84.6592 | 17.3957 | | PULL | 46.6447 | 214.72 | 3.9912 |   PUSH        PULL        Pull 케이스에서는 Order waiting time 과 product CT가 상당히 늘어났지만, 작업 중인 제품 수인 product WIP 는 작게 나타났다. 이는 Pull 케이스에서는 WIP을 제한하기 위해 Hold를 사용했기 때문이다. WIP 상한선이 설정되면, 라인이 혼란스러워져도 WIP은 상한선을 초과하지 않는다. 그러나 Push 케이스에서는 기계 고장, 품질 문제로 인한 중단 등으로 인해 라인이 혼란스러워질 경우 계획에 따라 진행되어야 하므로 중단된 기계 앞의 WIP은 크게 증가할 것이다.  2. A total of 4 Hold modules are used to implement the Pull system. Explain **in detail** what the operating mechanic is.  Pull system에서, 4개의 hold 모듈을 사용했다. Pull system은 신호가 있을 경우에만 buffer에 있는 entity를 이동시키기 때문에 buffer가 필요하다. hold 모듈이 각 station에서 바로 그 buffer의 역할을 수행하는 것이다.    설정한 condition을 살펴보면, 이후의 상태를 확인한 후 entity를 넘긴다.  구체적으로는, Raw 1은 다음 buffer의 대기 숫자가 0이면서 process 1의 WIP이 0일 때 entity를 넘겨준다. Raw 2 역시 마찬가지이다. Raw 3은 process 3의 WIP이 0이면서 다음 buffer인 finished goods의 number in queue가 5 이하일 때 entity를 넘겨준다. finished goods은 completed entity가 기다리는 곳으로, matching order with product라는 batch 모듈의 number in queue가 1 이상이면 entity를 넘겨준다.  이처럼 hold module은 WIP의 개수를 제한함으로써 공정상 문제가 발생하더라도 라인의 안정성을 유지할 수 있게 해준다. |

**< Discussion & conclusion >**

|  |
| --- |
| (1) List as many the ways as possible that a company can set the WIP upper bound.  WIP level이 높을 경우 우선순위나 일정계획 변경이 어려워지고, 부품을 사전에 투입해야 한다. 이로 인해 Time horizon이 길어지면 고객 주문이 불확실해지고 수요 예측에 의존해야 하며, 예측치의 부정확성으로 인해 performance가 악화될 수 있다.  Pull system은 WIP 상한을 설정하여 시스템의 유연성과 안정성을 높이는데, 다양한 방식으로 WIP 상한을 정할 수 있는데, 그 방법론을 소개해보겠다.  Capacity 분석: 생산 공정의 생산 용량, 주기 시간 및 처리량 비율을 기반으로 최적의 WIP 수준을 결정할 수 있다.  과거 데이터 분석: 과거 생산 데이터를 분석하여 WIP 수준의 추세와 패턴을 식별하고 해당 정보를 사용하여 과거 성능 및 효율성을 고려하여 상한선을 설정할 수도 있다.  린 제조 원칙: JIT(Just-in-Time) 및 Kanban 시스템과 같은 린 제조 원칙을 적용하여 고객 요구에 따라 WIP 제한을 설정하고, 낭비를 최소화하며, 자재 및 생산의 원활한 흐름을 유지할 수 있다.  대기 이론(Queue Theory): 대기 이론 원리를 적용하여 생산 프로세스를 모델링하고 분석하며, 도착률, 서비스 비율, WIP 한도를 설정하기 위한 원하는 성능 지표와 같은 요소를 고려할 수 있다.  시뮬레이션 모델링: 수업에서 다룬 Arena와 같은 시뮬레이션 모델링을 통해 다양한 시나리오를 시뮬레이션하고 다양한 WIP 수준이 시스템 성능에 미치는 영향을 평가하여 최적의 상한을 식별할 수 있다.  업계 모범 사례: 회사 부문별 표준 및 지침을 고려하여 업계 동료에 대한 벤치마킹 및 WIP 관리 및 제어를 위한 모범 사례를 채택하는 것 역시 좋은 방식이다.  투입량 조절: 다소 작위적인 방식이긴 하지만, Capacity보다 투입량을 작게 함으로써 WIP 한도를 지정할 수도 있다. 다만 이는 TH의 감소로 이어질 수 있다.  위와 같은 방식으로 Pull system의 도입과 함께 적절한 WIP 상한을 설정하여 WIP을 관리할 수 있다.  (2) Why is a Pull system more robust than a Push system? What practical results does this have on the manufacturing floor?  강의안을 참고하면, system의 robustness는 profit function of the form으로 확인할 수 있으며, Pull system이 WIP 오차에 강건한 정도는 Push system이 TH 오차에 강건한 정도보다 크다.  Profit = p ∗ TH − h ∗ WIP (p : 작업당 한계 이익, h : WIP 한 단위당 비용)  Profit function은 위와 같은데, 다음 상황에서의 Profit function을 비교함으로써 답을 도출할 수 있다. 5 machine이 serial하게 있으며, 모든 machine은 effective process time이 1 hour/job이고, process time은 exponential distribution인 상황을 살펴보자.  CONWIP system의 TH(w) = =  Push system의 WIP(TH) =  CONWIP의 TH를 대입해보면 WIP(= = 이므로, 동일한 TH 수준에서 WIP level이 CONWIP system에서 Pull system보다 항상 25% 높다는 것을 확인할 수 있다.  돌아가서 Profit function을 보면,  CONWIP system의 Profit(w) =  Push system의 Profit(TH) =    Profit을 그래프로 그리면 위 사진과 같다. 그래프를 살펴보면, Optimum에서 벗어날수록 Push system은 linear한 감소를 보이지만 CONWIP 은 보다 완만하게 감소한다. 즉, 더 robust하다.  추가로, WIP은 직관적으로 관찰이 가능하지만 TH는 비교적 관찰하기 어렵다. 이 점에서, Pull system 은 WIP을 효율적으로 제한해서 Optimum에 도달하기 보다 쉬울 뿐 아니라 앞서 살펴본 강건성으로부터 비용과 안정성 면에서 이점이 있다는 practical result를 도출할 수 있다. |