

# シミュレータを用いた搬送システム設計に関する研究

池田 佳 則  
(技術研究所)

村越 修 平  
(技術研究所)

## § 1. はじめに

工場の自動化技術の高度化は、コンピュータ技術の進展と相まって機械単体の自動化からセルレベルの自動化へ、さらには工場全体の自動化へとその範囲を広げている。これに伴い、原料・製品の運搬やハンドリングは人手から無人搬送車やコンベヤライン、ロボットに置き換わり、各種の自動機械が停滞なく、かつバランスよく稼働するために、緻密なスケジュールを効率良くこなす役割を担うようになってきた。

一方、生産システムの設計活動は自動化レベルが高くなる、あるいは自動化システムの範囲が広がるに従って互いに作用し合う要因が複雑に絡み合う上に、運用ルールを含めた高度なシステム計画が要求されるため、従来のIE的分析手法に基づいた机上の計算だけでは対応が困難となる傾向にある。

こうした背景のもと、さまざまな仕様の生産システムについて事前に実際の運用をシミュレートし、その評価を行なうことがシステム設計の重要な方策となってきた。

搬送シミュレータはその代表的なものの一つで、プラントレイアウト問題、倉庫仕様決定問題等と絡んで多くのパラメータがダイナミックに関与するため、運用ルールを踏まえた生産システム設計を行なう上で有効なツールとして位置付けられている。

本研究では、パソコンで扱える搬送シミュレータを開発し、設定したモデルについて種々の要因が搬送効率に与える影響について考察している。

## § 2. 搬送シミュレーションシステムの開発

### 2.1 システムの概要

本システムは、任意の閉ループ型搬送路上に出庫用倉庫・入庫用倉庫・ワークステーションを配置し、各ワ

ークステーションの出庫指令と入庫指令に従って搬送車が原材料(ワーク)の供給と加工品の回収を行なうものである。本システムの基本動作は以下のようになっている。

- ワークステーションからの出庫要請は、各ワークステーションのバッファが空になった時点で出す。
- ワークは出庫用倉庫から供給され、加工処理後、入庫用倉庫に回収される。
- 各ワークステーションが要求するワークはワークステーション固有のもので、出庫用倉庫は要求のあったワークステーションに対応するワークを出庫する。
- 出庫用倉庫は出庫要請の都度出庫作業を行ない、出庫バッファにワークを並べる。
- 搬送車の速度は一定で、発進時・停止時・カーブ時の加減速について特に考慮はしない。

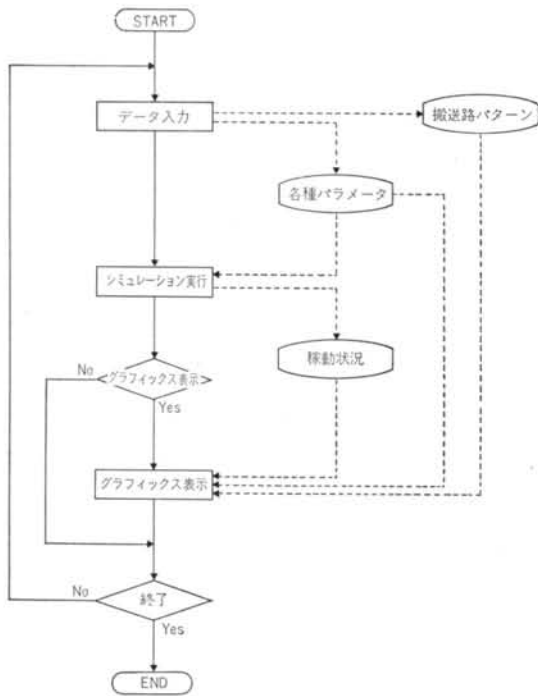
本システムは、汎用性の高いシステムを目指し、各種のパラメータにより様々な環境を設定し、シミュレーションを実行することができる。

### 2.2 基本設計

本シミュレーションシステムの基本設計に当たって、以下の事項に留意した。

- 搬送路構築に汎用性を持たせ、容易に設計・設計変更が可能であること。
  - 実際の搬送車の制御ロジックにとらわれることなく対応が可能であること。
  - 搬送車・ワークステーションの稼働状況が目に見えるようにすること(グラフィック表示)。
  - 各種の機能付加に対して対応が可能であること。
  - 処理速度が高速であること。
  - 使いやすさの面から大型の計算機は使用せずに、パソコン程度で手軽に使えるようにすること。
- これらの問題に対して、次のように対処した。

搬送路構築の汎用性についてはグラフ理論を応用し、有向グラフによって搬送路系をモデル化することで問題解決を図った。さらに、搬送路のパターンを特に定めず



図一 1 システムの全体フロー

主搬送路の長さや分岐路の始点・終点・長さをパラメータとすることで、任意の搬送路を構築可能とした。

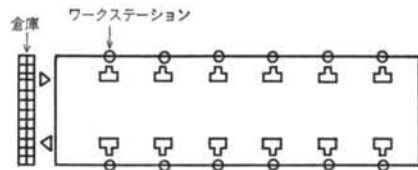
搬送車の制御をどのようにするかは、大きな問題である。実際の制御ロジックには様々なものがあり、それら一つ一つのためにシステム設計するのではなく、シミュレータとしてはむしろ制御ロジックへの提言を行なえることが最適だと考える。本研究報告においては、制御ロジックへの提言はひかえたが、搬送車の動作を簡単なロジックにより理にかなったものとして実現し、様々な制御ロジックに対してある程度まで対応できるよう配慮した。今回採用した制御ロジックは、

- 通常走行時の衝突回避のための前方サーチ
- 分岐点における目的地へのルート判断
- 各合流点における衝突回避のためのフラグの設定

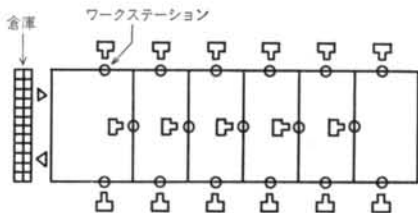
という最小限のものである。

搬送車・ワークステーションの稼働状況については、統計データとして出力するのももちろんのこと、視覚に訴えることで数値からだけでは分からない状況、例えば搬送車の干渉状況等をとらえられるようにした。

各種の機能付加に対する対応策としては、構造化プログラミング手法を用いることでプログラムの構造を明確にとらえられるようにするとともに、保守性を高め、追加・修正が楽になるよう配慮した。また、ワークステーション・倉庫・搬送車の一連の動作を独立な行動サイク



図一 2 分岐のない搬送路



図一 3 分岐のある搬送路

ルとして確立することにより、プログラムとしての見通しがきくようになり、機能の追加・修正はさらに楽になったといえる。

処理速度については、グラフィックとも関連して次のような方針とした。すなわち、本システムを、

- (1) データ入力部
- (2) シミュレーション実行部
- (3) グラフィック表示部

の3つに分け、各サブシステムの最適化を図り、処理速度の高速化をねらった。これにより、グラフィック表示を必要に応じて使用できるようになり、従来の一体化したシステムよりも使いやすくなったといえよう。各サブシステム間のデータの授受は、データファイルを設定し行なうこととした。

最後に開発環境についてであるが、このようなシミュレータを大型の計算機上で実現するには、グラフィック端末等も必要となり、コストもかかる。本システムは何よりも使いやすさ・手軽さを考慮し、さらには設計段階において、大型の計算機を必要とするほど大きなシステムにはならないであろうとの見通しから、パソコンによる開発を進めた。

## 2.3 開発経過

本システムの開発は、以下のようなステップに従って行なった。

Step 1: 分岐のない閉ループ型、一方向搬送路

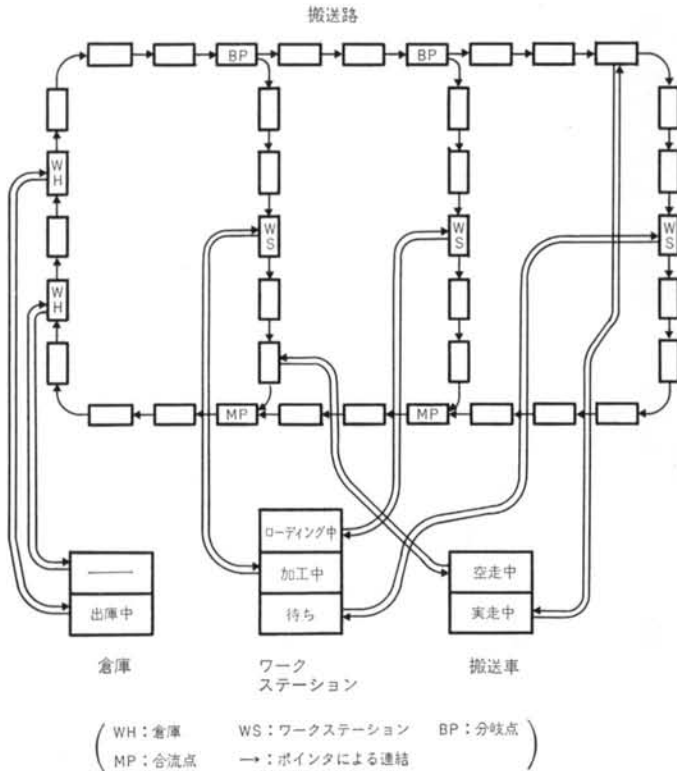
Step 2: 分岐のある閉ループ型、一方向搬送路

Phase 1: 各ワークステーションに対する供給専用

Phase 2: 各ワークステーションにバッファを設定

Phase 3: 加工品の回収を行なう

上位の Step, Phase は、下位のものの機能を全て含



図一四 搬送路系のデータ構造  
んだものになっている。

## 2.4 データ構造

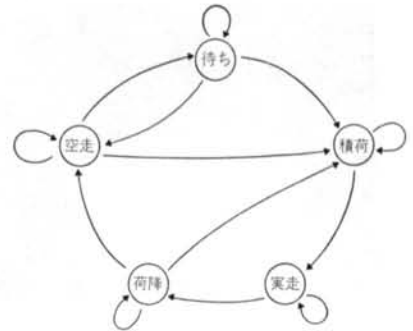
本システムは、手続き型言語 Pascal を用いて記述されている。Pascal にはデータタイプとしてレコード型・ポインタ型等があり、本システムの開発を非常に容易にしたといえる。

搬送路はポインタ型変数によるリストとして表現し、必要なデータをレコード型変数として保持させた。

搬送車・ワークステーション・倉庫は各々レコード型変数により表現し、搬送路上にポインタを設定することで位置を表現した。

## 2.5 行動サイクル

行動サイクルとは、ワークステーション・倉庫・搬送車の各々の状態推移を有限オートマトンとして表現したものである。各々の状態は、ある一定の条件によって推移する。また、ワークステーション・倉庫・搬送車の間は、ある種の通信により情報の伝達を行なっている。搬送車は、主にこの情報（この場合、出庫要請等）をもとにワークの搬送を行なっている。



図一五 搬送車の行動サイクル

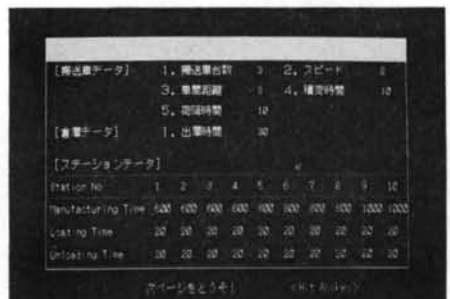
## 2.6 入出力データ

本システムに必要な入力データは大別すると、

- 搬送路に関するもの（主搬送路の長さ、分岐路の数・位置・長さ等）
- ワークステーションに関するもの（数・位置・加工時間・ローディング時間等）
- 倉庫に関するもの（位置・出庫時間等）
- 搬送車に関するもの（数・速度・車間距離・積降時間・搬送ルート等）
- 運用上の機能等に関するもの（回収の有無・シミュレーション時間等）



写真一 入力画面（メニュー）



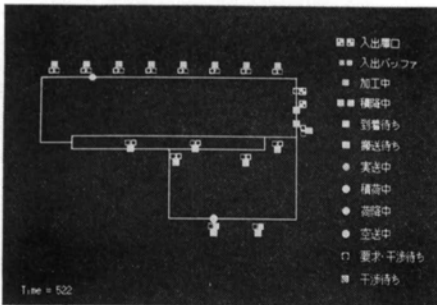
写真二 入力画面（データ一覧）

NO.	稼働率	平均待ち時間	到着ワーク数
1	0.382	430.000	4
2	0.432	657.500	4
3	0.432	657.500	4
4	0.432	657.500	4
5	0.681	350.750	4
6	0.656	370.750	4
7	0.636	400.000	4
8	0.611	428.000	4
9	0.666	147.500	3
10	0.666	156.567	3
11	0.218	2440.000	0
12	0.666	196.667	3
13	1.000	0.000	3
14	1.000	0.000	3
15	0.666	20.000	3
16	0.662	56.000	3
av	0.665	453.333	**

NO.	稼働率	干渉回数	平均干渉時間
1	0.010	1	4291.000

写真一 3 出力画面



写真一 4 グラフィック表示

となる。一方、出力データは、

- ワークステーションの稼働率，平均待ち時間，到着ワーク数
  - 搬送車の稼働率，干渉回数，平均干渉時間
- などの統計データと、
- グラフィック表示による稼働状況
- である。

ここで、干渉について述べておく。一般的には複数のものが互いに影響を及ぼし合うことをいうが、本研究報告においては干渉という語を搬送車の状態の一つとして定義し、扱っている。本研究報告における定義は、ワークを積んだ状態で走行（実走）中の搬送車が、なんらかの前方障害に対して停止することである。ワークを積んでいない状態での走行（空走）中の停止は、干渉として扱っていない。

### § 3. シミュレータ利用による実験と考察

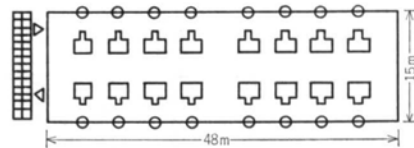
#### 3.1 モデルの設定

搬送システムは、扱う品目・生産形態・生産量等によって様々で一般化が難しい。ここでは、機械加工を行なう無人搬送システムを想定して表一1，図一6に示すようなモデルを設定した。

搬送路長	126m
ステーション数	16
加工時間	*10~40分, 30~60分, 30~120分
ローディング時間	1分
アンローディング時間	1分
ステーションバッファ数	0, *1, 2, 3
搬送車台数	1~7台
搬送車速度	30m/分
積荷・荷降時間	1分
出庫時間	*60~100秒
シミュレーション実行時間	8時間 (14,400 clock)

注) \*は他の項目をパラメータとするとときに固定した値

表一 1 設定したモデル



図一 6 設定したモデル

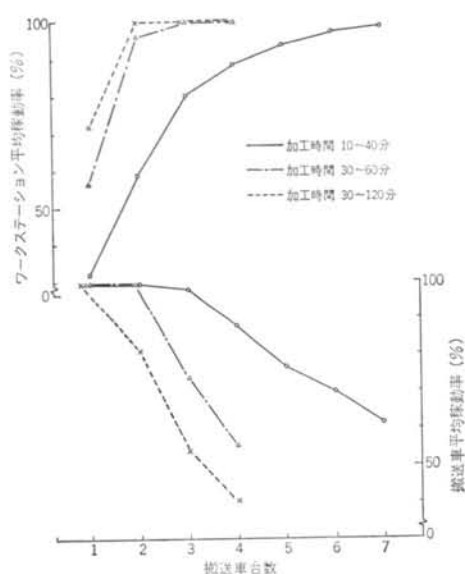
#### 3.2 システムパラメータの評価

シミュレータの利用に当たっては、各パラメータの値を変え、繰り返し実行することによって最少の設備投資で最も効率の良い搬送システムを得ることになるが、パラメータの特性を予め知っておくことはシミュレータを用いた搬送システム設計を効率よく行なう上で有効である。本節では、設定したモデルについて次の項目をパラメータとして実験・評価を試みた結果について述べる。

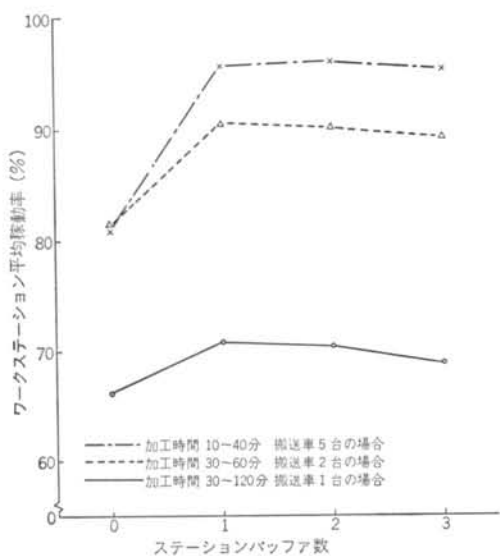
##### 3.2.1 搬送車台数

搬送車台数は、搬送システムの能力を設定する上で大きく影響を与えるパラメータである。一般には、ワークステーション（機械）の能力を十分に発揮できるだけの台数を用意することになるが、安全側を見るあまり過剰な設備投資となっている例も少なくない。

実験では、搬送車台数を増やすほどワークステーションの平均稼働率は向上し、搬送車1台当たりの平均稼働率はワークステーションの稼働率がある点に達したところから徐々に低下することが確認できた(図一7)。システム設計に当たってはワークステーション、搬送車双方の平均稼働率が共に100%に近くなる搬送車台数が（メンテナンス等を考慮したときの余裕はないが）適切な台数設定となる。しかし、多くの搬送車を必要とするケース（例えば、加工時間が短いケース）においては搬送車間の干渉が原因して搬送車の能力を十分に生かせないため、ワークステーションの平均稼働率が100%に至る点では搬送車1台当たり66%の稼働率しか得られないことが分かる。このようなケースでは、後で述べる他のパラ



図一七 ワークステーション、搬送車の平均稼働率の変化



図一八 ステーションバッファ設置による効果  
メータの与え方によって搬送車台数を低減する余地があるものと考えられる。

### 3.2.2 ステーションバッファ数

出庫要求間隔のばらつきを吸収し、搬送待ちのロスによるワークステーションの稼働率の低下を防ぐ意味で、ステーションバッファは重要な役割をする。しかし、バッファを多く持つことは、スペースを食うばかりか工場内在庫を増大させることになり、好ましくない。

実験では、ワークステーションの加工時間のばらつき

が大きい場合 (30~120分)、そうでない場合 (30~60分、10~40分) についてバッファ数を変化させてワークステーションの平均稼働率の変化をみた (図一八)。いずれの場合もバッファを持つことは有効であるが、複数個持った場合の稼働率の向上はみられなかった。また、加工時間が短いワークステーションに多くのバッファを持たせたケースでは若干の稼働率の向上が得られたが、期待したほどの大きなものではなかった。グラフィック表示により各々の場合について搬送車の動きを観察したところ、定常状態に入ってから搬送車の動きは、バッファを各ワークステーションが1個以上持つ場合には変わらないことが明らかとなった。したがって、バッファを持たない場合を除いては、バッファ数によるワークステーションの平均稼働率の変化は、初期状態において各ワークステーションが一度に出した出庫要求をこなす時間に起こるロスの大小であると考えられる。今回開発したシミュレーションシステムは、バッファに空きが出た時点で出庫要求を倉庫に送り、先入先出(FIFO)ルールに従って出庫・搬送するロジックを採用している。バッファを多く持つ場合に初期状態でロスが発生するのは、このロジックが作用して全てのワークの加工を終了し、待ち状態になったワークステーションからの出庫要求を後回しにするからである。FIFOルールに代わってワークステーションの状態を常に監視して、リアルタイムに出庫順の入れ換えを行なう制御ロジックを開発・採用すれば、バッファの有効性は変わるものと考えられる。

### 3.2.3 ワークステーションへの搬送優先順位

各ワークステーションへの搬送の優先順位を変えることにより、干渉回数を減少させワークステーションの稼働率を向上させ得るケースがある。グラフィック表示画面の観察により、干渉の起こるワークステーション群の搬送優先順位を倉庫から遠い順に入れ換えた例では、ワークステーションの平均稼働率が最大5.5%高くなった (図一九)。

### 3.2.4 出庫時間

ワークステーションからの出庫要求間隔が短い場合には、倉庫での所要時間 (最大値 (出庫時間、積荷時間)) がボトルネックになるケースがある。ワークステーションがフル稼働すると仮定したとき、平均出庫要求間隔 (OI) は次式で与えられる。

$$OI = 1 / \sum_{i=1}^n \{1 / (PT_i + LT_i + UT_i)\} \quad \dots\dots(1)$$

$PT_i$ : 加工時間

$UT_i$ : アンローディング時間

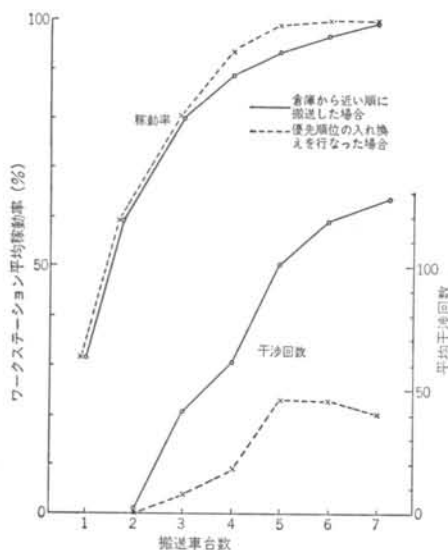


図-9 搬送優先順位の入替えによる効果

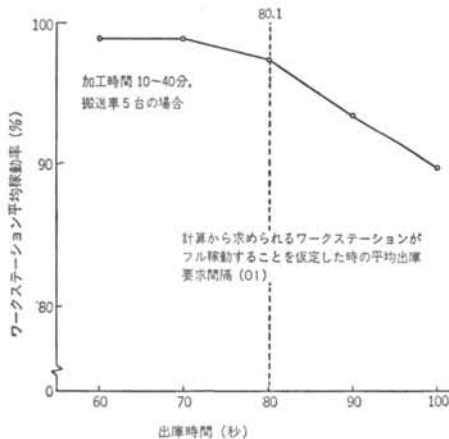


図-10 出庫時間を変化させたときのワークステーション稼働率への影響

$LT_i$ : ローディング時間

$n$ : ワークステーション数

倉庫をボトルネックとしないためには、倉庫での処理サイクル(最大値<出庫時間, 積荷時間>)が出庫要求間隔(OI)より十分に小さいことが必要である。

実験では、搬送車台数を十分に与え、出庫時間を変化させて式(1)で得た値(OI)との比較を試みた。OIより出庫時間が長い場合には、ワークステーションの平均稼働率が落ちていることが分かる(図-10)。

本システムでは、出庫時間をパラメータとして与えているが、1回の出庫に要する時間は一定として扱っている。自動倉庫を用いた場合の出庫時間は、倉庫規模、ス

タッカークレーンの上下・左右の速度、フォークの動作時間、ロード(ワーク)の格納位置等を与えることによって計算から求めることができる。今後、これらをシミュレートする自動倉庫シミュレータとの連結を図り、倉庫仕様を含めたシステムの検討を行なうことが課題として残される。

### 3.2.5 その他のパラメータ

今回の実験では搬送車速度、積荷時間、荷降時間は定数として扱っている。これは、安全性を考慮した設備設計等の理由から、パラメータとして扱う余地のない項目と考えたためである。もちろん、本システムではパラメータとして扱うことが可能なので、実用上の利用に際しては検討を加えていきたい。

この他、シミュレータ開発過程で、空走中の搬送車が完成したワークの搬送待ち状態にあるワークステーションを通過する際に、そのワークステーションに立ち寄って回収を行なう“キャッチルール”を考案し、ルール適用の有無を比較検討したが、適用による効果はみられなかった。

## 3.3 レイアウト設計と搬送システム

限られたスペースの中に生産ラインを設置するには、搬送効率を最優先としたレイアウトが選択されるとは限らない。本節ではレイアウトスペースと形状を固定し、考えられる幾つかのパターンについて搬送システムの評価を試みる。

### 3.3.1 外通路・内通路

機械群を通路を挟んで中央に置き、その周りを搬送路とする外通路(図-11(a))と通路の両側に機械群を配置する内通路(図-11(b))では、搬送路長が異なることから搬送能力は内通路の方が高くなると考えられる。搬送車が干渉なくフル稼働することを仮定し、搬送サイクルタイム( $CY_i$ )を式(2)から求め、搬送車1台の単位時間当たりの搬送量(搬送能力)を比較する。

$$CY_i = R_i + (WL_i + WU_i) + (SL_i + SU_i) \quad \dots\dots(2)$$

$R_i = RL/SP$ : 走行時間=搬送路長/スピード

$WL_i, WU_i$ : 倉庫での積荷時間, 荷降時間

$SL_i, SU_i$ : ワークステーションでの積荷時間, 荷降時間

●外通路の搬送能力  $CY_{t_1} = 7.32$

●内通路の搬送能力  $CY_{t_2} = 7.97$

約9%、内通路の方が搬送能力が高いことになる。

実験では、ワークステーションの平均稼働率に3~5%の向上がみられたが、倉庫での搬送車待機場所がないために干渉回数は多くなっている(図-12)。また、倉

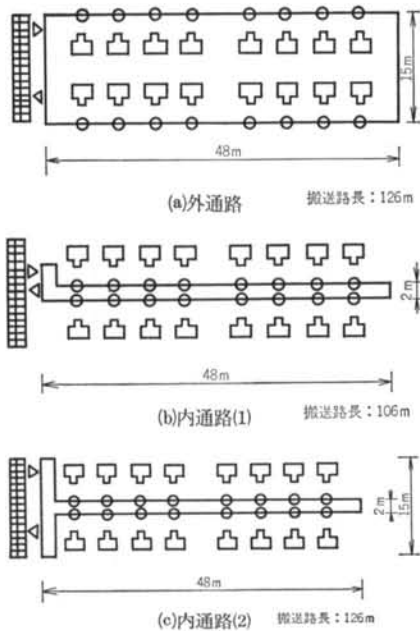


図-11 設定した外通路と内通路

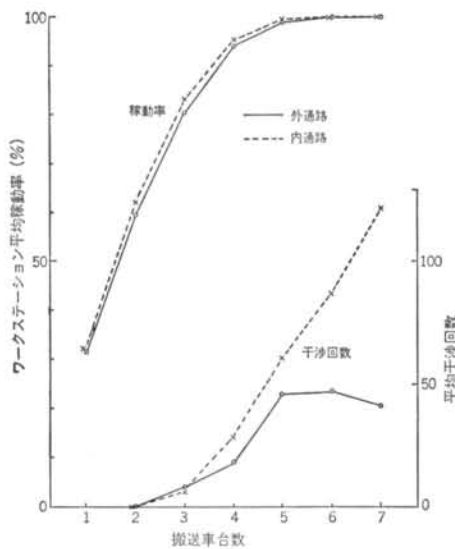


図-12 内通路と外通路のワークステーション稼働率の比較  
 倉庫での搬送車待機場所を設け、搬送路長を外通路と等しくしたケース（図-11(b)）では外通路とほぼ同じ結果となった。

### 3.3.2 ショートカット

搬送路間にショートカットを設けたケース（図-13 (a), (b), (c)）では、ワークステーションの稼働率向上に効果はなかった（表-2）。逆に、干渉回数が増加する結果となっている。これは、合流点での干渉が増加した

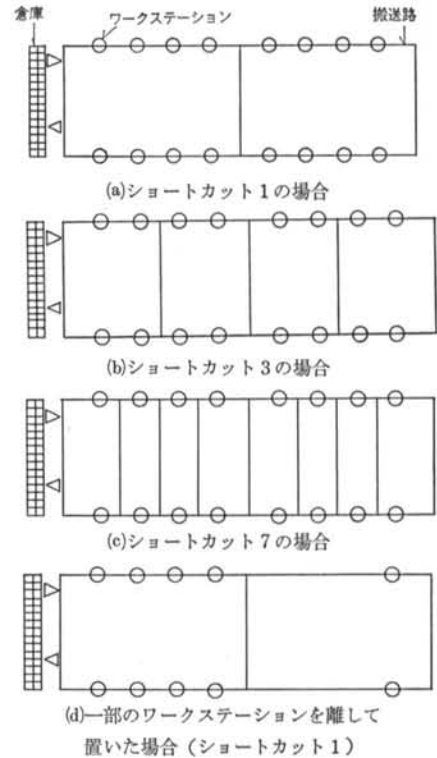


図-13 ショートカットの例

ショートカット数		搬送車数						
		1	2	3	4	5	6	7
ショートカットなし	ワークステーション平均稼働率(%)	31.5	59.3	80.6	93.8	98.9	99.6	99.6
	搬送車平均稼働率(%)	99.7	99.5	99.1	97.6	87.5	74.0	63.5
	平均干渉回数	—	0.0	7.7	18.0	46.0	46.7	40.7
ショートカット1	ワークステーション平均稼働率(%)	33.2	61.3	80.7	93.3	98.7	99.5	99.6
	搬送車平均稼働率(%)	99.6	99.4	96.4	91.9	82.5	69.7	59.9
	平均干渉回数	—	1.0	19.0	44.5	64.6	69.5	63.1
ショートカット3	ワークステーション平均稼働率(%)	34.0	63.5	81.1	94.0	99.2	99.5	99.6
	搬送車平均稼働率(%)	99.6	99.5	93.9	90.2	80.2	67.3	57.8
	平均干渉回数	—	0.0	34.0	58.3	79.6	81.7	72.1
ショートカット7	ワークステーション平均稼働率(%)	34.0	63.5	81.2	94.3	98.7	99.5	99.6
	搬送車平均稼働率(%)	99.6	99.4	94.0	90.6	79.5	67.2	57.7
	平均干渉回数	—	0.5	36.0	60.5	86.8	88.8	82.7

表-2 ショートカットによる効果

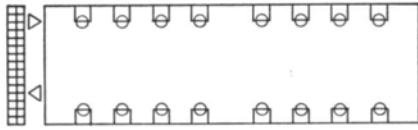


図-14 待避路を設けた場合

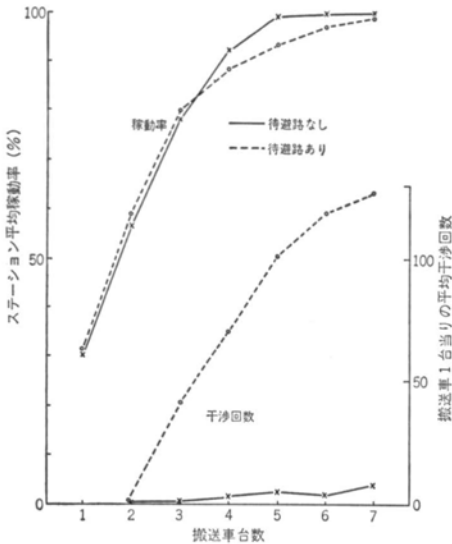


図-15 待避路による効果

ことによるほか、次のことが原因している。

- 目的地までの距離が短くなることにより、全ての場合に搬送車が通過する区間（倉庫付近）が相対的に混雑する。
- 搬送能力が向上したことにより、倉庫での荷降時間がボトルネックとなって入庫口で干渉を引き起こしている。

今回設定したモデルのレイアウトでは、機械群が比較的近ままとまった形をとったためショートカットの効果が認められなかったが、レイアウトの制約上、幾つかのワークステーションを離して置かざるを得ないとき（図-13(d)）には最大5.6%の効果認められた。

### 3.3.3 待避路

各ワークステーションに迂回路を設けたケース（図-14）では、干渉回数は無視できる程に小さくなり、ワークステーションの平均稼働率も十分に高くなる良好な結果を得た（図-15）。しかし、機械設置スペースは狭くなるため、通路スペースが十分とれる場合に限り有効な経路設定であるといえる。

### 3.4 干渉

搬送システムの能力を引き出すために、搬送車間の干渉を減少させることは重要な課題である。ここでは、搬

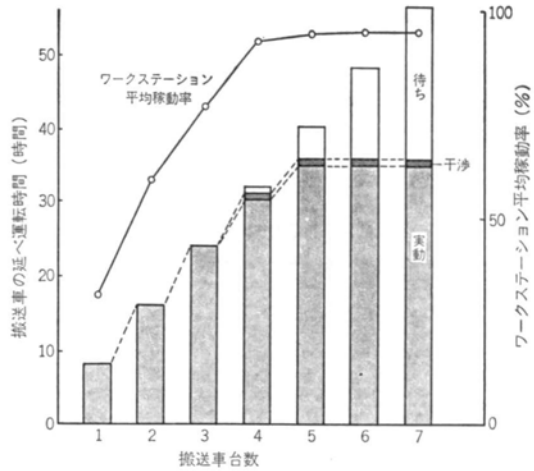


図-16 搬送車の状態別延べ運転時間

送車の状態を実働（実走・空走・積荷・荷降）、干渉による待ち、要求待ち（倉庫で待機）の3状態に分け、搬送車台数の変化による干渉時間の変化をみた。搬送車が4台のときにはワークステーションの稼働率が十分に得られていないにもかかわらず、干渉が原因して搬送能力がフルに生かされていないことが分かる（図-16）。ワークステーションの平均稼働率が最大値となる点から干渉時間が一定となっているのは、搬送車台数を必要以上に増やしても倉庫に待ち状態で待機する搬送車が増えるだけで、実働する搬送車の動きは変わらないことを意味する。

今回の実験から、干渉回数は次のパラメータによって変化することが明らかとなった（図-17）。

- (1) 搬送優先順位の入れ換え (3.2.3)
- (2) ショートカットの数 (3.3.2)
- (3) 待避路の有無 (3.3.3)
- (4) 倉庫の位置 (3.3.1)

干渉の回避は、搬送車制御の方法によっても可能である。

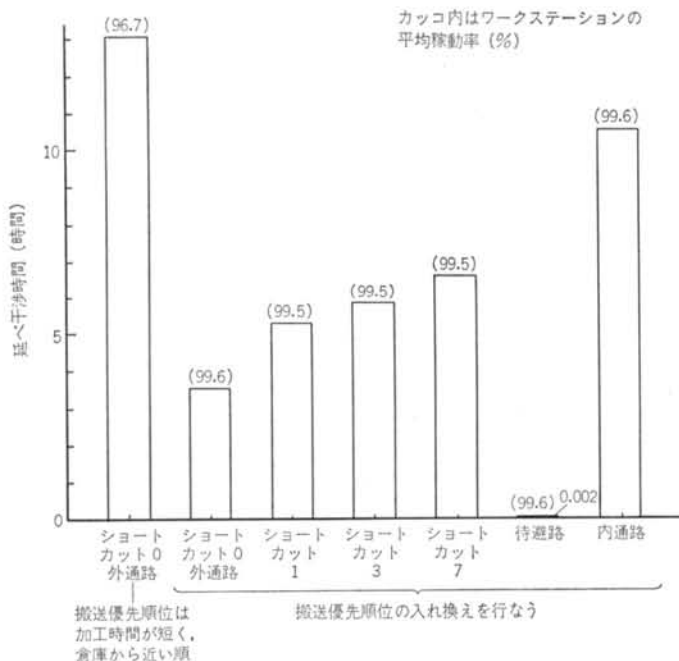
本システムによるシミュレーション結果のグラフィック表示は、様々な制御パターンに固有の制御ロジックを組み合わせる上でも有効に機能するものと考えられる。

### 3.5 実験・考察のまとめ

以上の実験結果から明らかとなった、シミュレータを用いた搬送システム設計のポイントをまとめると、次のとおりである。

- 1) 搬送車の必要台数は、台数を増やすことによるワークステーションの平均稼働率の向上が認められないと判断される最低の台数とする。





図一七 パラメータ、レイアウトを変化させたときの延べ干渉時間の比較

2)干渉回数が多い場合、他のパラメータ、レイアウトの変更によるシステム改善の余地がある。

3)搬送車台数を増やしてもワークステーションの平均稼働率が100%近くに達しないとき、倉庫での処理時間を平均要求間隔よりも十分に小さく設定することで改善を図れる可能性がある。

4)ステーションパッファは、各ステーションに1個設けることが稼働率向上に有効である。

5)搬送路パターンはシンプルな方が良く、ショートカットの設定は、機械群が分散している場合の他は干渉を増やすことになり、ワークステーションの稼働率の向上に効果は期待できない。

6)倉庫の手前には、搬送車待機場所を設ける必要がある。

7)待避路は、スペースが許す限り設けた方が良い。限られたモデルの範囲ではあるが、独立でない様々のパラメータが搬送効率に与える影響を整理した本実験の結果から、シミュレータを用いた搬送システムの設計を

効率よく行なうことができる。また、これらの作業を通じ、搬送システム設計案を絞り込む過程では、考えられる最善の状態からシミュレーションをスタートさせ、後に制約を加えて個々のパラメータを決定して行くプロセスが、シミュレータを用いたシステム設計作業をより効率よく進められるものとの認識を得た。

#### § 4. おわりに

シミュレータを用いた生産システム設計を行なうに当たっては、概略設計段階から詳細設計段階に至る各段階において変化する計画内容とその都度シミュレートし、評価することが望ましい。そのために、シミュレータは手軽に扱え、かつ汎用性に富んだものが要求される。しかし、手軽に扱えるシミュレータは詳細設計段階で十分に仕様を詰め切れない欠点を持ち、汎用性に富

むシミュレータはパラメータが多くなりシステムも大きくなることから、利用に際して熟練を要するという欠点をもつ。また、搬送システムを中心に考えれば上下階の搬送を扱う立体搬送シミュレータ、工場内搬送の前後工程(資材搬入、倉庫、出荷、製品物流)を含めたシミュレータ、実用の制御用コンピュータを用いたシミュレータ等が課題として考えられる。

これらの課題に対し求められるシミュレータとは、数多くの設計作業に適用しパラメータの削減・追加を行なった上で、設計段階毎、サブシステム毎に小規模にまとめたものとする。そして、これらを用いた設計作業とは、対話形式で各設計段階に用意されたシミュレーションを行ない、データを受け渡して次のシミュレーションに移るといったものになるであろう。

今回の開発は、その1ステップを手掛けたものであるが、今後上記の課題に取り組み、従来の経験的な手法をより科学的な手法に置き換えた生産システムの設計手法を確立していきたい。

#### <参考文献>

- 1) 油井: "ファクトリーオートメーション" 日本工業新聞社 (1984年)
- 2) 高橋, 久保: "プラントレイアウト" 建帛社 (1972年)
- 3) 遠藤, 江川, 那須: "無人車システムのシミュレータ" 神鋼電機 103号 (1983年)

- 4) 村野: "FA システムシミュレーションと適用例" オートメーション 第31巻, 第6号 (1986年)
- 5) 村野: "FA システムシミュレーション" 富士時報 Vol. 58, No. 9 (1985年)
- 6) 池田訳: "グラフ理論" 共立出版 (1971年)
- 7) 五百井, 荒木訳: "コンピュータによるグラフ理論の応用" 共立出版 (1981年)
- 8) 米田, 疋田: "PASCAL プログラミング" サイエンス社 (1979年)