スケジューリングの理論

古瀬大六

序

線型計画法を利用して各種の工場における生産計画を出来るだけ有利に決めようとする試みは，*OR* の中の有力な一手法として，一般実務担当者の間にも広く知られるようになつてきている。

しかし，単純な線型計画的手法によつて決めることの出来るのは，各週或は各月において生産されるべき各種生産物の種類と量とだけであつて，各種の機械にどのような順序で掛けるか，については全然答えてはいない。現実の生産活動は，何をどれだけ，だけでなく，何時どの機械でどういう順序で生産するのか決まらなければ，開始することができない。

これに答えようとするのが，最近キャリフオルニヤ大学（ロス・アンゼルス）の Management Science Research Project における Salveson, Jackson, Nelson, Barankin 等，及び Rand Corporation における Johnson, Bellman 達によつて研究されつつある scheduling の理論である。

本論文は Johnson の pioneering article を紹介し，その後の発展の現状と見通しについて簡単に触れたものである。より一般的な取扱いについては別の機会に譲りたい。

1 問題の所在

生産計画が直ちに現場において具体化されるためには，次の五つのことがらが決められていなければならない。

（1） 製品の種類と量 (product mix)。

（2） 加工順序の指定 (routing)。

（3） lot size の決定 (economic lot size)。

（4） 使用機械の指定 (machine allocation)。

（5） 作業時間表の作成 (scheduling)。

ここでは，このうちの最後のスケジューリングの問題について考えてみたい。これら五種類の決定は互に複雑な相互依存関係に立つているので，これらを同時に決めることのできるようなモデルを考えることが一番望ましい。然し，本論文においては，そのような理想的取扱法に到達するための第一段階として，(5) だけを他の (1)—(4) から切離して独立に取り扱うこととする。

われわれは問題の性質上やむを得ず数学的表現を使用するけれども，数学的推論それ自体に興味を持つているわけではないので，必要以上の数学的厳密さを追求することはせずに，できるだけわかり易く説明することに重点をおきたい。

スケジューリング問題とは，次のような特殊な問題を意味する：

「n 種類の部品を m 種類の機械にかけて加工する場合，着工から完了までの所要時間を最小にする為には，各機械における各部品の加工順序をどのように決めたらよいか？　但し，(1)各種部品はそれぞれ時間的に連続して加工され，それを途中で分割することはできない，(2)部品 $i=1, 2, \cdots\cdots, n$ の機械 $A, B, \cdots\cdots, M$ における加工所要時間を $a_i, b_i, \cdots\cdots, m_i$ とする，(3)各種部品が完成するまでに，$A, B, \cdots\cdots, M$ のうちのどの機械をどのような順序で通らなければならないか (routing)，ということは予め決定されている。」

この三つの条件のうちの第一は，lot size が既に与えられており，その分割が不可能であることを意味している。その部品が物理的に連続したソリッドなものであれば，上記の条件が完全にみたされることは言うまでもない。しかし，技術的・物理的には可分てあつても，機械や工具の準備に要する時間と費用 (set-up cost) とが，分割によつて二倍に増加するために，経済的な理由から分割を不利とする場合も実際には少くないであろう。後者の経済的分割不可能性についても，更に二つのケースを分けて考えることができる。その一つは，顧客又は他部門からの注文によつて lot の大さが事前に与えられている場合であり，他の一つは，その注文の lot が非常に大きいか又は市場向生産であるために lot の大きさを自由に選べるとき，set-up cost や保管費用などを

考慮した上でいわゆる economic lot size の決定に従つて最も有利な lot の大きさを計算して決める場合である。

これらの何れの場合であつても，スケジュール決定の手続としては，lot の大きさを所与として扱えばよいのであるから，(1)，(2) の仮定は決して非現実的なものではなく，実際のケースの大部分はこの仮定を満足するであろう。

(3)の routing が所与であるという前提も，現実に合致する場合が多い。書籍の生産を例にとれば，組版→印刷→製本という順序は技術的に定まつており，逆にするわけにはいかない。然し，そのうちの一つの作業をやることの出来る機械は必らずしも一種類とは限らない。それを決める問題が，いわゆる machine allocation problem である。ここでは，この使用機械の種類は既に決定ずみであるとする。

この種の問題を最初に提供したのは Management Science Research Project の前身である Industrial Logistics Research Project のリーダーであつた Salveson である。彼自身の外に，Nelson, Barankin 等の人々がその解決に努力したけれども，なかなか容易には実用的な解法を見出すことができなかつた。

問題自身の論理的性格は極めて簡単明瞭である。すなわち．routing の条件を無視すれば，n 種類の部品を m 種類の機械にかけるときのスケジュールの数は $(n!)^m$ を超えることはない。この中から，与えられた routing に合致するものを選び，更にその中から全所要時間の最短のものを抜き出せば，解に到達することができる。解の存在は明らかであるし，このような単純な手続の有限回の繰返しによつて解に到達することも明らかなのだから，これで万事解決ずみだと考えられるかもしれない。

だが，現実にこの種の問題を解こうとする現場の人々からすれば，このような目の子算式解法が一体どれだけの時間で解に到達できるかという問題を無視することはできない。10部品10機械の場合のスケジュールの数を計算してみると $(10!)^{10} \fallingdotseq 37(0)^{114}$ という尨大な数になり，その一つを一秒で処理できたとしても，$12(0)^{107}$ 年というとてつもない長い時間を必要とするから，実用的解法としては全然無価値である。このていどの問題であれば，遅くとも1時間以

内に答がでなければ，実用にはならないであろう。

連続函数の極値問題については既に充分な研究が完了しており，その計算法も確立している。不等式条件が加われば，若干の不連続迄が現れて，計算がいくらか複雑にはなるが，その不等式の数が200を超えなければ一日以内で解くことができる。然し，われわれが今問題にしているスケジュール決定の問題は，与えられた順列の中から或る函数の値を最大又は最小にするものを選ぶという，全く不連続な問題である。そのために，従来の解析的な手法が全然役に立たない。

スケジュール問題の一般的解法を発見するためには，このような順列極値問題についての基礎理論の建設から始めなければならない。この方面への努力は，Barankin や Kuhn によつて或る程度進められてはいるけれども，まだほんの出発点に辿りついた程度で，差当つてのわれわれの問題の解決には直ぐに役立ちそうもない。

現在までの成果としては，機械が二台又は三台の場合，しかも各部品の加工順序が何れの機械についても同じである場合，に限り Johnson によつてその一般的な実用的解法が与えられている。それ以上に機械の多い場合に対しては，順列論的取扱の困難さを回避するために，種々の簡便計算法が研究されつつある。必らずしも最短加工時間を保証するものでなくても，それに近いものが簡単な手続きで得られるならば，その実用的意義は少くないであろう。

そのような簡便法の一つとして，例のガント図表が古くから使われている。この方法の欠陥は，すべてが直観的判断によつて行われ，はつきりした計算手続きが与えられていない点である。

戦時中に盛んに使われた優先順位法もまた，この簡便法の一つであるが，これは，順位の最高のものを極端に優遇する反面，二位以下のものが甚しい悪影響をうけ，全体としての所要時間が却つて長くなる危険性をもつているので，あまり良い方法とは言えない。この単純な優先順位法を改善して，スケジュール決定の一方法に仕上げようとする努力が，現在 Jackson 及び Nelson によつて進行中である。

2 二機械の場合

以下において，二機械の場合における Johnson の解法を説明する。

前節において (1) lot の不可分性，(2) 所要加工時間の一定，(3) routing の所与性，を仮定してきたが，ここでは，問題を簡単にするために更に第四の仮定として，(4) 総べての部品は先ず機械 A で加工し終つた後に機械 B で加工される，こととする。この第四の仮定を除いたらどうなるのか，については再び後で触れるであろう。

これらの仮定の上に立つて，スケジュール問題を再び定式化するならば：

「1，……，n の部品があり，何れも機械 A で加工し終つた後，初めて，機械 B にかけることができる。機械 A，B はそれぞれ一台しかなく，それぞれ一時に一部品しか加工することができない。a_i，b_i，をそれぞれ部品 i が機械 A，B にかけられる際の，取附時間プラス正味加工時間とするならば，全所要時間を最短にする加工順序（スケジュール）は，どのようにして求めることができるか？」

そのような最適スケジュールが存在することは確実である。その最適スケジュールの A における順序と B における順序とは，必らずしも同じであるという保証は与えられていない。然し，若しもその順序が A と B とで違つているならば，全所要時間を増加させることなしに，その順序を同じになるように修正することができる。

或る任意のスケジュール S が与えられており，各部品は $A \to B$ という routing 条件を満足しているものとする。そのうちの部品 i と部品 j との順序が，A では $i \to j$，B では $j \to i$ というように反対になつていたとする。この部品 i を A で加工する作業を A_i，B で加工する作業を B_i とし，部品 j についての同様の作業をそれぞれ A_j，B_j とする。この四つの作業を時間の順序に並べるならば，それがスケジュール S 属にするものである限り：

$$A_i \to A_j \to B_j \to B_i$$

という順序になつており，それ以外の順序は存在し得ない。

そこで，B_i の直前に B にかけられる部品 i' と i との順序を入れ替えてみ

よう。隣り同志を入れ替えるのであるから，B におけるこの二部品の加工時間の和 (b_i+b_i') は不変であり，従つて，機械 B における他の部品のスケジュールは，このような入れ替えによつて何の影響をも受けることはないであろう。また，それによつて B_i' はその着手が時間的に遅くなるのであるから，最初のスケジュール S において $A_i'\rightarrow B_i'$ なる関係が保証されている以上，B_i' を更に右（時間的に日後）の方に移動させても，$A_i'\rightarrow B_i'$ なる関係は破られることはない。

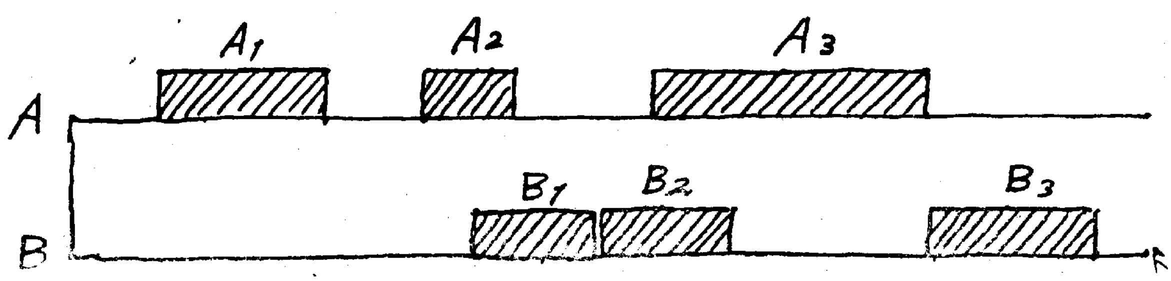
B_i の着手は反対に早くなるけれども，それは B_j の着手時点よりも早くなることはなく，しかも，$A_i\rightarrow A_j\rightarrow B_j$ なる関係はそのまま保たれているのだから，$A_i\rightarrow B_i$ もそのまま成立つ。それ故，このような入れ替えは，全体の所要時間を延ばすこともなく，また，技術的前後関係 (routing) を破壊することもなしに，実行することができる。

このようにして，B_i を次から次えと左隣りの作業と入れ替えていけば，有限回の入れ替えの後に，$B_i\rightarrow B_j$ なる順序に到達することができるであろう。B の順序をそのままにしておいて，反対に A の方の順序を入れ替えることによつて，B のそれに一致させることも出来る。以上の結論を次の補助定理にまとめておく。

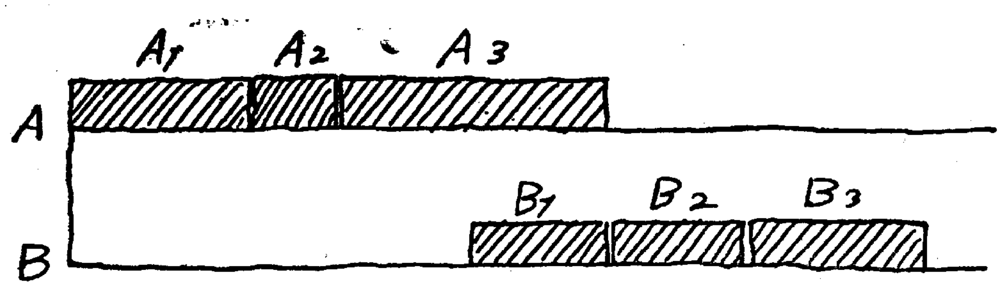
〔補助定理1〕　機械 A 又は B における加工順序を，時間上の損失なしに，他の機械 B 又は A における加工順序に一致させることができる。

この補助定理が成立つことを知つた以上は，最適スケジュールを求めるに当つても，A における各部品の加工順序と B におけるそれとが全く同じであるような場合だけについて考えればよいことになる。これは，計算の上からも，思考の上からも，大きな節約になるであろう。

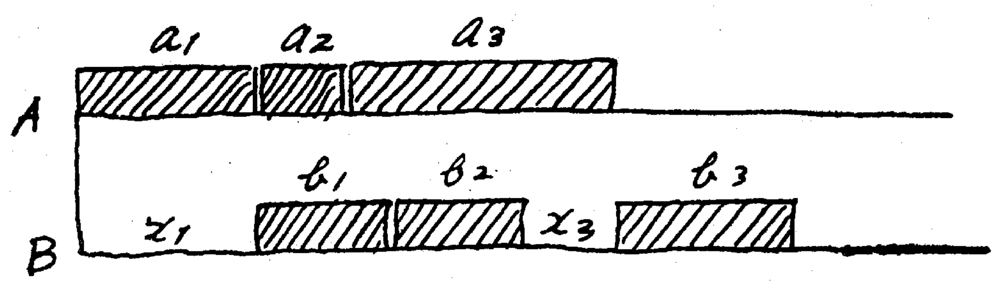
そこで，$A_i\rightarrow B_i$ なる技術的荷役関係を満足し，かつ，各部品の加工順序が A と B とで同一てあるところの，任意のスケジュール S を考えてみよう。



この S における所要時間を短縮するはどうしたらよいであろうか？ 何れの部品についても A の方が先に使用されるのであるから，$A_1, \cdots\cdots, A_n$ はできるだけ左の方へ動かしてやれば，それだけ $B_1, \cdots\cdots, B_n$ のうちのある部品の着工を早めることができ，しかも，$A_i \to B_i$ なる関係を保持することができるであろう。そこで先ず，A_1 の左端を原点，則ち 0 時点に移し，$A_2, \cdots\cdots, A_n$ は互に時間的なすきまのないように，ぴつたりとよせつけてしまうことにする。



次に，Bの方も，A の作業が終り次第，無駄な手待ちをすることなしに，直ちに加工にかかることとする。先ず，B_1 の左端を A_1 の右端の位置まで左に動かす。次の B_2 の左端を A_2 の右端に一致するところまで左にずらしてやろうとすると，A_2 の右端に到達する以前に，B_1 の右端に引かかつて動けなくなつてしまう。そこで，やむをえず，B_2 の左端は，B_1 の右端に一致させておく。更に B_3 を左に移動させていくと，こんどは，B_2 の右端につきあたることなしに A_3 の右端に到達することができる。



このような手続きを A_n まで繰返すならば，与えられたに S ついての最短所要時間を与えるスケジュールを求めることができる。上図からも明らかなように，$A_1, \cdots\cdots, A_n$ の間には空隙は全く存在しないが，0，$B_1, \cdots\cdots, B_n$ の間には幾つかの間隙，則ち手待ち時間が存在し得る。この B_i の直前における手待ち時間を x_i とすれば，全所要時間は

$$\sum_1^n (b_i + x_i)$$

で表わされる。そのうち $\sum^n x_i$ は所与であるから，手待ち時間の合計 $\sum^n x_i$ を最小にすれば，それは同時に全加工時間の最小を実現することになるであろう。

そこで，与えられた加工順序 S のもとにおける $\sum^n x_i$ の最小値を計算してみよう。

$$x_1 = a_1$$

$$x_2 = max\ (a_1 + a_2 - b_1 - x_1,\ 0)$$

この第二式は，0 と $(a_1 + a_2 - b_1 - x_1)$ とのうち，何れか大なる方をとることを意味する。この二つの式を辺々相加えれば：

$$x_1 + x_2 = max\ (a_1 + a_2 - b_1 - x_1 + a_1,\ a_1)$$

更にこの右辺括孤内の第一項の最終項 a_1 に $a_1 = x_1$ を代入すれば：

$$x_1 + x_2 = max\ (a_1 + a_2 - b_1,\ a_1)$$

次に x_3 については：

$$x_3 = max\ (a_1 + a_2 + a_3 - b_1 - b_2 - x_1 - x_2,\ 0)$$

となる。$(x_1 + x_2)$ は，a_1, a_2, b_1 が所与である以上，$max\ (\sum_1^2 a_i - b_1, a_1)$ なる関係を通じて一般的に定まるので，これを上式の両辺に加えても等号が成立つから，

$$\sum_1^3 x_i = max\ (\sum_1^3 a_i - \sum_1^2 b_i,\ \sum_1^2 x_i)$$

となり，更にこの右辺の $\sum_1^2 x_i$ に $max\ (\sum_1^2 a_i - j_i,\ a_1)$ を代入すれば：

$$\sum_1^3 x_i = max\ (\sum_1^3 a_i - \sum_1^2 b_i,\ \sum_1^2 a_i - b_i,\ a_1)$$

となる。この手続きを x_n に到達するまで繰返すことにより，次の式が得られ

る。

$$\sum_1^n x_i = \max_{1\leq u\leq n} \left(\sum_1^u a_i - \sum^{u-1} b_i\right)$$

$$= \max_{1\leq u\leq n} X_u$$

この式は，各部品の A 及び B における加工順除が与えられており，その順序に従つて各部品に $i=1,\cdots\cdots,n$ の番号を附けた場合の機械 B における遊休時間を表わすものである。われわれの目的は，このような所与の S に対する最短加工時間を計算することではなく，そのような加工時間を最短にするような最適順序 S_0 を発見することにある。そこで，上記の S の中の任意の部品 j と部品 $(j+1)$ との順序を入れ替えたならば，全加工時間（従つてまた全遊休時間 $\sum_1^n x_i$）がどのように変化するかを計算してみよう。

このような j と $(j+1)$ との入れ替えによつて変るのは，K_j と K_{j+1} とだけであつて，他の $K_1,\cdots\cdots, K_{j-1}, K_{j+2},\cdots\cdots, K_n$ は全然変らない。

この入れ替えた後の x の値を x' とすれば：

$$\sum_1^n x_i = \max\,(K_1,\cdots\cdots, K_{j-1}; K_j, K_{j+1}; K_{j+2},\cdots\cdots, K_n)$$

$$\sum_1^n x'_i = \max\,(K_1,\cdots\cdots, K_{j-1}; K'_j, K'_{j+1}; K_{j+2},\cdots\cdots, K_n)$$

となる。

(1) $\max\,(K_j, K_{j+1}) \leqq \max\,(K'_j, K'_{j+1})$ である場合には：

$$\max\,(K_1,\cdots\cdots, K_{j-1}, K'_j, K'_{j+1}, K_{j+2},\cdots\cdots, K_n)$$

$$= \max\,({}_1,\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots, K_n; K'_j, K'_{j+1})$$

であるから，$\sum_1^n x'_i$ の値は増加することはあつても減少することはあり得ない，則ち；

$$\sum_1^n x_i \leqq \sum_1^n x_i$$

(2) $\max\,(K_j, K_{j+1}) > \max\,(K'_j, K'_{j+1})$ である場合には；

K_j 又は K_{j+1} が $(K_1,\cdots\cdots, K_n)$ のうちの最大項である場合には所要時間は減少し，然らざる場合には，不変に保たれるであろう。

従つて，与えられた加工順序 S が最短加工時間を保証するものであるため

の充分条件は，

$$max\ (K_j, K_{j+1}) \leqq max\ (K'_j, K'_{j+1})$$
$$(j=1, \cdots\cdots, n-1)$$

が成立つことである。

与えられた非最適順序から出発して，最適順序 S_0 に到達するには，上記の条件に合致しない，則ち：

$$max\ (K_j, K_{j+1}) > max\ (K'_j, K'_{j+1}) \qquad (1)$$

なる関係にある j と $j+1$ とを発見したならば，その順序を入れ替えていけばよい。それを何回か繰返した結果，上記の不等式関係を示すような j が一つも存在しないようになれば，その時の順序が最適順序 S_0 に外ならない。

この K の値を一々計算するのは面倒なので，簡便な比較法を考えてみたい。K_{j+1}, K'_j, K'_{j+1} をそれぞれ K_j で表わせば：

$$K_{j+1} = K_j + a_{j+1} - b_i$$
$$K'_j = K_j - a_j + a_{j+1}$$
$$K'_{j+1} = K_j + a_{j+1} - b_{j+1}$$

となるから，これらの値を(1)に代入した上で各項から$(K_j + a_{j+1})$を差引けば：

$$max\ (-a_{j+1}, -b_j) > max\ (a_j, -b_{j+1})$$
$$\therefore\ min\ (a_{j+1}, b_j) < min\ (a_j, b_{j+1}) \qquad (2)$$

となるであろう。任意の j と $(j+1)$ 番目との部品につき (2) の関係が成立つならば，その加工順序を入え換えるのが有利である。総べての j について(2)の不等式が成立たない状態になつたならば，そのときにおける加工順序に最短加工時間を保証するであろう。この最適順序 S_0 を実際に計算する手続きとして Johnson が与えているのは，次のような形式的方法である：

1) $a_1, \cdots\cdots, a_n$ 及び $b_1, \cdots\cdots, b_n$ を次のように縦に並べる。

i	A	B
1	a_1	b_1
2	a_2	b_2
$\vdots$	$\vdots$	$\vdots$
n	a_n	b_n

2) これら $2n$ 個の数字のうち最小のものを選ぶ。

3) それがもし a_i であつたならば，その部品 i を第一番目の加工対象とする。

4) それがもし b_i であつたならば，その部品 i の加工を最後にまわす。

5) その部品 i についての a_i, b_i を，上記リストの中から消す。

6) 残りの a, b について同様の操作を行い，加工順序表のうちの空位の左端又は右端に入れる。

7) 同じ A 列又は B 列の中の最小値にタイのものがあるときは，番号 i の若い方を選び出す。

8) a_i と b_i とがタイのときは，A の方を選ぶ。

例えば，加工時間の一覧表を：

i	A	B
1	5	6
2	5	3
3	32	5
4	8	32
5	3	4

とすれば，最小値は $b_2 = a_5 = 3$ であり，その中の任意の一つ b_2 を第5番目の加工対象とする。残りのうちの最小値 $a_5 = 3$ をとつて，これを第1番目の加工物とする。更に残りの最小値 $a_1 = 5$をとつて，第2番目の加工対象とする。同様にして，$i=3$ が第4番目，残りの $i=4$ が第3番目となる。すなわち，最適順序は：

$$(5 \to 1 \to 4 \to 3 \to 2)$$

と，簡単に求められる。

最後に，われわれの特殊な仮定，すなわち，全部品の routing が何れも $A \to B$ であるという仮定について再検討してみよう。

$A \to B$ なる routing の部品を $1 \cdots\cdots, m$, 反対に $B \to A$ なる routing のものを $m+1, \cdots\cdots, m+n$ とする。A 機械については，$1, 2, \cdots\cdots, m$ の順序で連続的に加工し，B 機械では $m+1, \cdots\cdots, m+n$ の順序で連続的に加工し

た上で，今度は機械を互に取りかえて同じ順序で加工する。このようにすれば，部品1が機械 B にかけられるのは $\sum_{j=1}^{n} b_{m+j}$ 時点においてであり，従つて，それは A における加工の終了した時点 a_1 に比べて，非常に後になるのが普通であろう。このような事情は，他の $2, 3, \cdots\cdots, m$ についても同様であり，部品 i の A における終了時点 $(a_1 + a_2 + \cdots\cdots + a_i)$ は，Bにおける着手可能時点 $(b_{m+1} + \cdots\cdots + b_{m+n}) + (b_1 + \cdots\cdots + b_i)$ よりも遙かに前であるのが通常であろう。

このような推論が正しいとすれば，異つた routing の混在する場合には，然らざる場合にくらべて，手待ち時間なしに全作業を遂行し得る可能性が多い，と結論することができるであろう。この可能性は，$\sum_{1}^{m} a_i$ と $\sum_{j=1}^{n} b_{m+j}$ との値が近接しているほど，大きく，両者が一致する場合には，加工順序をどのように選んでも必らず，全然遊休時間を生ぜしめることなしに作業を進めることができる。

Johnson が単一 roting の場合だけをとりあげて考えたのも，このように，混合 routings の場合にはスケジュールの如何があまり作業時間に影響を及ぼさないことを考えたからではないてあろうか。勿論，混合 routings であつても，極端な場合には遊休時間を生ずる。その際には，先ず，$A \to B$ なる routing をもつ全部品について，上記の単一 routing のときの計算規則を適用してスケジュールを決める。次に $B \to A$ なる反対の routing をもつ全部品についてもまた，同様にして最適スケジュールを決める。この両者がきまれば，最初の $A \to B$ のスケジュールのうちの機械 B の加工における 0 時点をそのまま右へ $\sum_{j=1}^{n} b_{m+j}$ だけずらし，また同時に，$B \to A$ なるスケジュールのうちの機械 A の加工における0時点を $\sum_{i=1}^{n} a_i$ だけ右へずらすことによつて，全体としての最適計画を決めることができるであろう。

その結果として，$A \to B$ 又は $B \to A$ について部分的に計算されたスケジュールの中に含まれている遊休時間 x_i, x_{m+j} は，その多くのものを 0 にすることができるようになるから，$b_1, \cdots\cdots, b_m ; a_{m+1}, \cdots\cdots, a_{m+n}$ はいずれもできるだけ左側へ押しつけるように努力しなければならないこと勿論である。

3 三機械の場合

機械の種類が一つふえて，A，B，C の三種類になると，前のように簡単にはいかなくなつてくる。しかし，三種類の機械の場合でも，二種類の場合と同様に，各機械における各製品の着工順位を同じであると仮定することが可能である。

或るスケジュール S において，部品 i と j の順位が，A，B，C によつて相違しているものとする。そのうち二つは必らず同順位であり，一つだけが他の二つと異つた順位をもつているのであるから，その一つが A の場合，B の場合及び C の場合について考えてみよう。

まず，A が異つており，B と C とが同じ場合には，前節の二機械の場合の〔補助定理1〕に従つて，A の順位を入れかえて B の順位と同じに改めることができる。これによつて B における最終完了時点は増加せず，また C における routing に影響することもない。何となれば，A_i をその右隣りの A'_i と入れ替えることにより，A'_i の着工時点は a_i だけ早くなるのであるから，これによつて $A'_i \to B'_i \to C'_i$ という（i'）部品の routing は何等の影響をも受けない。従つて，このような手続を繰りかえせば，すべての routing を破ることなしに，また，全体の所要時間をふやすことなしに，A_i と A_j とを入れかえることができる。その結果，すべての機械における着工順位は（$j \to i$）に統一される。

A と B とが同じ（$j \to i$）であり，C だけが（$i \to j$）であるならば，C_j をその直前の C'_j と入れかえ，更にその直前のものと入れかえる，という手続きをくりかえすことによつて，全機械における着工順位を（$j \to i$）に統一することができる。これによつて，C における最終完了時刻が延びないのは勿論，C'_j は C_j だけ遅くなるのであるから，$A'_j \to B'_j \to C'_j$ の routing は依然として成立つであろう。それ故，この場合にもまた，全機械の着工順序は（$j \to i$）に統一される。

最後に，A と C とが（$i \to j$），B だけが（$j \to i$）である場合には，まず A_j を順番に直後の部品と入れ替えることによつて A と B とにおける順序を（j

$\to i$）に統一し，次に C_j をその直前の部品と入れかえることによつて B と C とを同じ（$j\to i$）に統一すればよい。

かくして，三機械の場合についても次の補助定理が成り立つ。

〔補助定理2〕 機械 A，B，C における加工順序を，時間上の損失なしに，悉く同じ順序に改めるとどができる。

このような簡単化が可能なのは三機械まででであつて，四機械になると，各機械における各部品の加工順位を変えた方が所要時間が少くてすむ場合が生じてくる。たとえば，

	a_i	b_i	c_i	d_i
$i=1$	3	3	3	3
$i=2$	3	1	1	3

とすれば，全部（1→2）ならば15時間，全部（2→1）でも同じく15時間であるが，A と B とでは（1→2），C と D とでは（2→1）というように機械によつて順序を変えることによつて所要時間を14時間に短縮することができる。そこで，再び三機械の場合にもどつて考えてみよう。前と同様に，全部品の routing は何れも $A\to B\to C$ であると仮定する。そうすれば，二機械の場合と同様に，A では全部品は遊休時間なしに引続いて加工することができるけれども，B と C とには手待ち時間を生ずる可能性がある。B における遊休時間を前回同様 $x_1,\cdots\cdots,x_n$ で表わし，C における遊休時間を $y_1,\cdots\cdots,y_n$ で表わすことにする。x の値は二機械の場合と全く同じ推論によつて：

$$\sum_1^n x_i = \max_{1\leq u\leq n}\left(\sum_1^n a_i - \sum_1^{n-1} b_i\right)$$

で表わされる。y の方は：

$$y_1 = x_1 + b_1$$

$$\quad = a_1 + b_1$$

$$y_2 = max\ (x_1 + x_2 + b_1 + b_2 - y_1 - c_1, 0)$$

……………………

$$y_n = max\ (\sum_1^n b_i + \sum_1^n x_i - \sum_1^{n-1} c_i - \sum_1^{n-1} y_i\ ,\ 0)$$

従つて：

$$y_1 + y_2 = max\ (x_1 + x_2 + b_1 + b_2 - c_1\ ,\ y_1\ )$$

....

$$\sum_1^n y_1 = max\ \ (\sum_1^n x_i + \sum_1^n b_i - \sum_1^{n-1} c_i\ ,\ \sum_1^{n-1} y_i\ )$$

この最後の式に

$$\sum_1^{n-1} y_i = max\ \ (\sum_1^{n-1} x_i + \sum_1^{n-1} b_i - \sum_1^{n-2} c_i\ ,\ \sum_1^{n-2} y_i\ )$$

を代入すれば：

$$\sum_1^n y_i = max\ \ (\sum_1^n x_i + \sum_1^n b_i - \sum_1^{n-1} c_i\ ,$$

$$\sum_1^{n-1} x_i + \sum_1^{n-1} b_i - \sum_1^{n-2} c_i\ ,\ \sum_1^{n-2} y_i\ )$$

これを順次繰りかえして右辺の y を消去すれば：

$$\sum_1^n y_i = max\ \ (\sum_1^n x_i + \sum_1^n b_i - \sum_1^{n-1} c_i\ ,$$

$$\sum_1^{n-1} x_i + \sum_1^{n-1} b_i - \sum_1^{n-1} c_i, \cdots\cdots,\ x_1 + b_1\ )$$

これに前記の

$$\sum_1^n x_i = \max_{1 \leq u \leq n}\ (\sum_1^u a_i - \sum_1^{u-1} b_i\ ) = \max_{1 \leq u \leq n} K_u$$

$$\sum_1^{n-1} x_i = \max_{1 \leq u \leq (n-1)} K_u$$

....

を代入していけば：

$$\sum_1^n y_i = \max_{1 \leq u \leq n}\ (max K_u + \sum_1^n b_i - \sum_1^{n-1} c_i\ ,$$

$$\max_{1 \leq u \leq n-1} K_u + \sum_1^{n-1} b_i - \sum_1^{n-2} c_i\ ,\ \cdots\cdots\cdots\cdots,\ K_1 + b_1\ )$$

ここに，$\sum_1^n b_i - \sum_1^{n-1} \equiv H_n$ と記すならば：

$$\sum_1^n y_i = max\ (\max_{1\leq u\leq u} K_u + H_n,\ \max_{1\leq u\leq n-1} K_u + H_{n-1},\cdots\cdots,$$
$$K_1 + H_1\ )$$
$$= \max_{1\leq u\leq v\leq n} (H_v + \max_{1\leq u\leq v} K_u\ )$$
$$= \max_{1\leq u\leq v\leq n} (H_v + K_u\ ) \qquad (3)$$

これが，二機械の場合の $\sum_1^n x_i = \max_{1\leq u\leq n} K_u$ に対応する，三機械の場合の遊休時間式に外ならない。仮定により各部品は C で最後の仕上げが行われるのであるから，C における遊休時間の合計値を最小にするようにスケジュールを組めば，それが同時に，全体としての所要時間を最小にする結果となることは，容易に推察できるであろう。

そこで，前回同様，j 番目と $j+1$ 番目の部品の順位を入れ替えてみて，それが $\sum_1^n y_i$ の値をどのように変えるか，を検討してみなければならない。

このような入れ替えによつて動くのは，a_j, a_{j+1}, b_j, b_{j+1}, c_j, c_{j+1} だけであるから，K と H のうちでは，K_j, K_{j+1}, H_j, H_{j+1}だけである。(3)式の右辺のうち，これら四つの項を含むものを取り出せば：

$$(H_{j+1}+K_{j+1}),\ (H_{j+1}+K_j\ ),\cdots\cdots,\ (H_{j+1}+K_1\ ),$$
$$(H_j+K_j\ ),\ (H_j+K_{j-1}),\cdots\cdots,\ (H_j+K_1\ )$$

の $(2j+1)$ 項だけである。この K_j, K_{j+1}, H_j, H_{j+1} の入れ替え後の値をそれぞれ，K'_j, K'_{j+1}, H'_j, H'_{j+1} とすれば，このような入れ替えが $\sum_1^n y_i$ を減少させるに充分な条件は：

$$max\ (H_{j+1}+K_{j+1},\ H_{j+1}+K_j,\cdots\cdots,$$
$$H_{j+1}+K_1\,;H_j+K_j,\ H_j+K_{j-1},\cdots\cdots,\ H_j+K_1\ )$$
$$> max\ (H'_{j+1}+K'_{j+1},\ H'_{j+1}+K'_j,\cdots\cdots,$$
$$H'_{j+1}+K_1\,;H'_j+K'_j,\ H'_j+K_{j-1},\cdots\cdots,\ H'_j+K_1\ )$$

が成り立つことである。

これら各項の内容を再び a, b, c の値に戻して考えてみると，$a_1,\cdots\cdots,a_{j+1}$；$b_1,\cdots\cdots,b_{j+1}$；c_j, c_{j-1} を含んでいる。二機械の場合にはその中に含まれるのは a_j, a_{j+1}, b_j, b_{j+1} だけであつたから，その入れ替えの有利不利を決めるために

はそれ以前の加工順位をもつ各部品がどのような順位で行われるかに関係なく，単にその問題となつている j と $j+1$ とだけの資料によつてこれを決めることができたのであつた。

ところが，三機械の場合になると，それ以前に加工された部品がどの部品であつたか，また，その順番がどうであつたか，によつて，或いはその入れ替えが有利となり，或いは不利となる。これら総ての場合について一々計算比較するということは，大変な手数がかかるので，現実的解法とはなり得ない。折角(3)式を求めてみたけれども，それからは，二機械の場合のような簡単な判定方法が得られないことがわかつた。

そこで，問題の一般性を更に限定することによつて，上記の判定式の中から $a_1,\cdots\cdots,\ a_{j-1},\ b_1,\cdots\cdots,\ b_{j-1},\ c_{j-1}$ を除く方法はないであろうか。それには，$\min a_i \geqq \max b_j$，すなわち，$a_1,\cdots\cdots,a_n$ のうちのどれをとつても，$b_1,\cdots\cdots,b_n$ のうちのどれよりも小さくない，と仮定すればよい。そうすれば

$$\begin{aligned}K_j &= \sum_1^j a_i - \sum_1^{j-1} b_i \\ &= \left(\sum_1^{j-1} a_i - \sum_1^{j-2} b_i\right)+(a_j - b_{j-1}) \\ &= K_{j-1}+(a_j - b_{j-1}) \\ &\geqq K_{j-1}\end{aligned}$$

すなわち K_j は j の非減少函数になる。従つて：

$$\max_{u\leqq v} K_u = K_v$$

$$\therefore\ \sum_1^n y_i = \max_{1\leqq v\leqq n}(H_v + K_v)$$

のように，非常に簡単化される。

ここで再び，j と $j+1$ との順位を入れ替えると，変化するのは H_j，H_{j+1}，K_j, K_{j+1}だけであるから，それを H'_j, H'_{j+1}，K'_j, K'_{j+1}とすれば $\sum_1^n y_i$ が減少するための充分条件は：

$$\begin{aligned}&\max\ (H_j + K_j\ ,\ H_{j+1}+K_{j+1}) \\ &\qquad > \max\ (H'_{j+1}+K'_{j+1},\ H'_j + K'_j\ )\end{aligned}$$

が成り立つことである。前の場合と違つて，u が1から v の間を自由に動くようなことはなく，必らず $u=v$ であることが保証されているから，前のように多数の項を比較してみる必要はない。この H と K とを元の a, b, c にもどすために：

$$H_{j+1}=H_j+b_{j+1}-c_j$$
$$H_{j+1}=K_j+a_{j+1}-b_j$$
$$H'_j=H_j-b_j+b_{j+1}$$
$$K'_j=K_j-a_j+a_{j+1}$$
$$H'_{j+1}=H_j+b_{j+1}-c_{j+1}$$
$$K'_{j+1}=K_j+a_{j+1}-b_{j+1}$$

を上の不等式に代入すれば：

$$max\ (H_j+K_j ; H_j+b_{j+1}-c_j+K_j+a_{j+1}-b_j)$$
$$>max\ (H_j+b_{j+1}-c_{j+1}+K_j+a_{j+1}-b_{j+1} ;$$
$$H_j-b_j+b_{j+1}+K_j-a_j+a_{j+1})$$

この各項から $(H_j+K_j+a_{j+1}+b_{j+1})$ を引けば：

$$max\ (-a_{j+1}-b_{j+1} ; -b_j-c_j)$$
$$>max\ (-b_{j+1}-c_{j+1} ; -a_j-b_j)$$
$$\therefore\ min\ (a_{j+1}+b_{j+1},\ b_j+c_j)<min\ (b_{j+1}+c_{j+1},\ a_j+b_j)$$

となる。二機械の場合よりも若干複雑にはなるけれとも，その計算は簡単である。

K_j が j の増加函数であるのと同様に，H_j が j の減少函数である場合にも，同じような簡単化が可能である。すなわち，$b_1, \cdots\cdots, b_n$ のどの一つをとつても，それは $c_1, \cdots\cdots, c_n$ のどの一つよりも大きくはない $(max\ b_i \leq min\ c_j)$ と仮定すれば：

$$H_j=\sum_1^j b_i-\sum_1^{j-1} c_i$$
$$=H_{j-1}+(b_i-c_{i-1})$$
$$\leq H_{j-1}$$

となるから：

$$\sum_{1}^{n} y_i = \max_{1 \leq u \leq v \leq n} (H_v + K_u)$$
$$= \max_{1 \leq u \leq n} (H_u + K_u)$$

なる関係式が得られる。これは，前の（$\min a_i \geqq \max b_j$）なる場合の式と全く同じであるから，

$$\min(a_{j+1}+b_{j+1},\ b_j + c_j) < \min(b_{j+1}+c_{j+1},\ a_j + b_j)$$

を満足するところの総べての j について，j と $j+1$ とを入れかえることによつて，最適解に到達することができる。

このような特殊な条件が現実に成立つような場合があるであろうか？ B を特定の機械と見ることをやめて，A から C への運搬時間であると考えるならば，$\min_{1 \leq i \leq n} a_i \geqq \max_{1 \leq j \leq n} b_j$ なる条件は，殆んど確実に実現されるであろう。また，$c_1, \cdots\cdots, c_n$ が，工具や材料の準備に必要な時間（set-up time）を含む場合，これらの set-up time をそれぞれ $-b_1, \cdots\cdots, -b_n$ とすれば，$a_i \geqq -b_i \geqq 0$ 又は $c_i \geqq -b_i \geqq 0$ が成立つならば，上記の判別公式をそのまま適用することができる。

4　連続函数による近似法

Johnson の解法を三機械の一般的な場合から，更に四機械以上の場合に拡張することは殆んど不可能と思われる。

一般に m 機械・n 部品の場合を目の子算でやるとすると，$(n!)^m$ の場合について一つ一つ計算してみなければならない。$m=3$，$n=10$ というような少数のケースを考えてみても，$37(0)^{114}$ という尨大な数になつてしまうこと，前に述べた如くである。

これを前節の手続きによつて：

$$\max (H_v + K_u) \qquad (3)$$

の形に改めるならば，$j+1$ と j との順位を判定するには，

$$(j+1)+j+(j+1)+j=4j+2$$

個の項を計算すればよく，従つて，高々：

$$n \times n \times (4n+2)$$

個の項，則ち，$n=10$ ならば 4200 の項の計算と比較で済むことになる。これは大変な利益であり，時間と労力の節約である。Johnson は，これを計算困難であるとして，その利用をあきらめているけれども，現在われわれの使用し得る最高の高速電子管式計算機を使うならば，恐らく数分で計算することのできる程度の問題でしかないであろう。

故に，われわれは，三機械までは，一般的に，短時間内で解くことができる，と言つてよい。然し，四機械以上となつてくると，そう簡単にはいかない。というのは，最適スケジュールにおける各部品の加工順序が，それぞれの機械によつて異なる可能性が存在するからである。このような条件のもとでは(3)のような簡単な式に到達することができない。

Bellman は，スケジューリング問題の今一つの解法として，Johnson の式を近似的に連続化する方法を提称している。まず，二機械の場合の Johnson の式：

$$\sum_{1}^{n} x_i = \max_{1\leq u\leq n} \left(\sum_{1}^{u} a_i - \sum_{1}^{u-1} b_i \right)$$

において，n を部品の種類の数と見ないで，二種類の部品の合計個数と考え，そのうちの k 個は第一部品，残りの $(n-k)$ 個が第二部品であるとする。部品の数はいくら多くても，種類は二つしかないから，a_i, b_i の値は a_1, a_2 ; b_1, b_2, の二通りしかあり得ない。

従つて，$\sum_{1}^{u} a_i$ なる函数は，i が 1 から 2, 3, 4,……と増加するにつれて，或るときは a_1 なる速さで，また他の時は a_2 なる速さで増加する。そこで，第一部品を加工中は 1 なる値をとり，第二部品を加工中には 0 なる値をとるステップ函数 ϕ を導入して：

$$a(u)=a_1\phi(u)+a_2\{1-\phi(u)\}$$
$$b(u)=b_1\phi(u)+b_2\{1-\phi(u)\}$$

とおけば，近似的に：

$$\sum_{1}^{u} a_i - \sum_{1}^{u-1} b_i \fallingdotseq \int_{0}^{u} \{a'(t)-b(t)\}dt$$

と書きなおすことができる。u の値，すなわち，加工される部品の数が多いほど，この近似は良好となる。

この函数 ϕ は，第1部品が何番目に加工され，第2部品が何番目に加工されるか，を示しているから，ϕ らを決めることは，スケジュールを決めることに外ならない。それ故，最適スケジュールを決めるには：

$$\begin{aligned}
\mathrm{I} &= \min_{\phi} \max_{0\leq u\leq n} \left[\int_0^n \{a(t)-b(t)\}dt\right] \\
&= \min_{\phi} \max_{0\leq u\leq n} \left[\int_0^u \{a_1\phi+a_2(1-\phi)\}\,dt-\int_0^u \{b_1\phi+b_2(1-\phi)\}\,dt\right] \\
&= \min_{\phi} \max_{0\leq u\leq n} \left[(a_1-a_2-b_1+b_2)\int_0^u \phi dt+(a_2-b_2)\int_0^u dt\right] \\
&= \min_{\phi} \max_{0\leq u\leq n} \left[(a_1-a_2-b_1+b_2)\int_0^u \phi dt+(a_2-b_2)u\right]
\end{aligned}$$

但し，

$$\int_0^n \phi(t)dt=k$$

$\phi=0$ 又は 1

これを解いて ϕ の値を決めるには，どうしたらよいか？　まず，$\alpha=(a_1-a_2-b_1+b_2)$ の値の正負に応じて，次の三つの場合に分けて考えてみる。

i)　$\alpha>0$ なる場合：

任意の $0\leq u\leq n$ に対して I の値は，第二部品を一括して最初に加工し，次に第一部品を一括して加工した場合（則ち，ϕ は区間〔0, $T-k$〕において 0，区間〔$T-k$, T〕において 1）の方が，反対の場合よりも小か又は等しい。

ii)　$\alpha<0$ なる場合：

任意の $0\leq u\leq n$ に対して，I の値は，第一部品を一括して最初に生産し，次に第二部品を一括して生産した場合（則ち，ϕ は区間〔0, k〕において 1，区間〔k, T において0）の方が，反対の場合よりも小か又は等しい。

iii)　$\alpha=0$ なる場合：

I の値は ϕ に無関係である。

以上の関係からして，α の符号を検討することによつて，最適スケジュールを決定することができる。この判定法はまた，

$$(a_1 - b_1),\quad (a_2 - b_2)$$

の値の小さいものを先に，大きいものを後に加工する，と表現しても同じことになる。

これをさらに，三つ以上の種類の部品の場合に拡張することは容易である。例えば，三種類の場合には問題は：

$$\phi_i = 0 \text{ 又は } 1$$

$$\phi_1 + \phi_2 + \phi_3 = 1$$

$$\int_0^u \phi_1\, dt = k_1,\ \int_0^n \phi_2\, dt = k_2,\ \int_0^n \phi_3\, dt = k_3$$

$$k_1 + k_2 + k_3 = 1$$

なる条件の下で，

$$\min_{\phi} I(\phi) = \min_{\phi} \max_{0 \leq u \leq n} \int_0^u \{(a_1 - b_1)\phi_1$$

$$+(a_2 - b_2)\phi_2 + (a_3 - b_3)\phi_3\}\, dt$$

を満足する ϕ を求めることであり，それはまた，

$$(a_1 - b_1),\ (a_2 - b_2),\ (a_3 - b_3)$$

を，その値の小さいものから順に並べることに外ならない。

次にまた，機械の種類を三種類に拡張すれば，$a(b)$，$b(t)$ の外に，新たに $c(t)$ が加わつて，

$$a(t) = a_1\phi + a_2(1-\phi)$$

$$b(t) = b_1\phi + b_2(1-\phi)$$

$$c(t) = c_1\phi + c_2(1-\phi)$$

となり，その最適スケジュールは：

$$\min_{\phi} I(\phi) = \underset{0 \leq u \leq v \leq n}{mvx} \left[\int_0^u \{a(t) - b(t)\}dt\right.$$

$$\left. + \int_0^v \{b(t) - c(t)\}dt\right]$$

但し　$\int_0^n \phi(t)dt = k$

で与えられる。

ここまでは順調に進むことができるが，機械が四種類以上になると，上記の $a(t)$，$b(t)$，$c(t)$ の右辺における ϕ を同じ形で考えることができなくなる。何となれば，ϕ を同じ形にとることは，各機械における部品の加工順序を同一と考えることに外ならないからである。前節にも説明したように，ϕ を同じ形にとることのできるのは三機械までに限られている。

Bellman は，不連続な組合せ問題を，連続函数化することに努力してはみたけれども，その一旦排除されたかに見えた不連続性が，実は再び ϕ 函数の形で頭を出してきていることに注目しなければならない。

一般的なスケジューリング問題の完全な解決への路は，未だ遠い将来に延びており，多くの有能な数学者の努力を待たなければならないであろう。

文　　献

(1) Johnson, S. M.: Optimal 2 and 3-stage Production Schedules with Step-up Times Included, Rand Paper, P-402, 5 May, 1953 (Revised, 20 Oct., 1953), pp.11.

(2) Bellman, R.: Dynamic Programming of Continuous Processes, Rand Report, R-271, July 1954, pp. 125—133.

(3) Jackson, J. R.: Notes on Some Scheduling Problems, University of California, Los Angeles, Management Science Research Project, Research Report, No.35, Oct. 1, 1954, pp.6.

(4) Rowe, A. J. and J. R. Jackson: Research Problems in Production Planning and Scheduling, UNCLA, MSRP, RR No.46, Oct. 1955, pp.

(5) Salveson, M.E.: On A Qualitative Method in Production Planning and Scheduling, Econometrica, Vol.20, No.9, Oct. 1952, pp. 571—574.