

# GAによるナーススケジューリング問題の一解法

小清水 誠\*

荒井 誠\*\*

## A solution of nurse scheduling problem by Genetic Algorithm

Makoto KOSHIMIZU

Makoto ARAI

Abstract - In this paper, we propose a new algorithm to solve the scheduling problem of nurse in the medical area. This problem is the quota problem of nurse that keep load balance of work, give the high quality service for the patients. In the case of many nurse, making satisfactory schedule by human power is very difficult. And the solutions by conventional methods are problem dependent and have limited capability when the characteristics and scale of the targeted problem change. The method presented here is a new approach method to solve that using Genetic Algorithm (GA). Some simulation-basis experiments are carried out and the results show good applicability of the proposed method.

Key word - Genetic Algorithm, nurse scheduling

### 1. はじめに

ナーススケジューリング問題とは、医療機関において患者に対して質の高いサービスができるように勤務負荷のバランスを取りながら勤務シフトを決定する問題である。看護師が多い場合、満足のいく勤務表を決定するのは大変困難な作業であり、パターン化が困難な上、短期間で頻繁にスケジュールの変更が必要であり、解析的な手法で解を求めることは不可能である。本研究は、この問題の解決ために遺伝的アルゴリズム (Genetic Algorithm) <sup>(1)(2)</sup>を用いた新しい手法を提案する。提案する手法は、解探索で必要となる看護師の勤務パターンを GA での遺伝子として表現し、その評価 (適応度) によって解 (平均勤務時間割) を効率的に求めるものである。

本研究は、提案する手法に基づいたシステムを構築し、その実験結果から本手法の適応可能性を論じる。

\* 釧路高専技術室

\*\* 釧路高専機械工学科

### 2. 遺伝的アルゴリズム (Genetic Algorithm)

GA とは、生物の進化の過程をコンピューター上でシミュレートすることで確立的に近似解探索を行う最適化アルゴリズムの1つである。

生物の進化とは、生殖により遺伝子が選択・交叉・突然変異を繰り返し与えられた環境に適応した個体として生成されることである。

GA ではコンピューター上での進化の過程をシミュレートするために、これまでいくつかの方法が提案されて

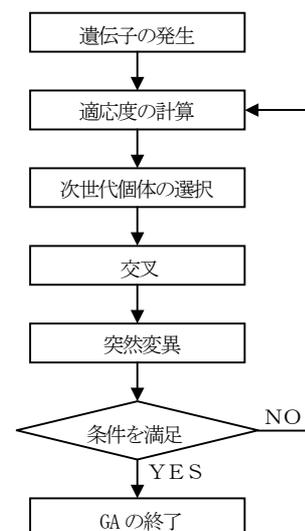


図1. SGA の概念.

いるが、その中でも最も多く使われているのが Goldberg<sup>(3)</sup> によって提案された Simple GA (SGA) である。図1に SGA の概念を示す。本研究での解析探索手法も SGA を基本としている。

### 3. 問題解決の手法

#### 3.1 勤務パターンの表現

GA で最も重要なことは、問題因子を GA での遺伝子形として表現することである。ここで、勤務表と遺伝子の関係について説明する。最も基本的な勤務表は人数 ( $m$  人) と日数 ( $n$  日) によって2次的に作成される。それを GA に適応させるために勤務表を図2のように、総数  $m \times n$  個をもつ遺伝子として表現することとした。

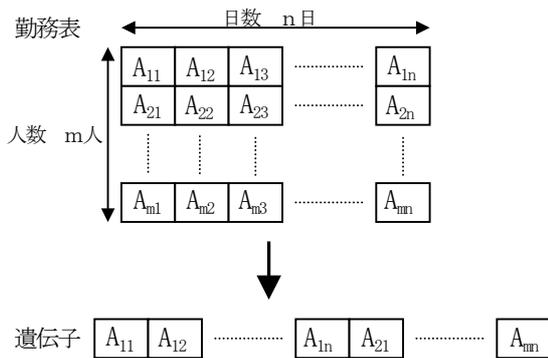


図2. 勤務表と遺伝子の関係

初期遺伝子の発生方法は、要素数が  $m \times n$  個の遺伝子として、勤務パターン数を  $P$  とすると式(1)により生成できる。

$$A_{M^N} = \text{rand}() \% P \quad \dots\dots(1)$$

$(M=1, 2, 3, \dots, m \quad N=1, 2, 3, \dots, n)$

ここで、rand() は整数型の乱数を生成するジェネレーターを示し、% は剰余の計算を表すものとする。すなわち、遺伝子 A には 0 ~ P-1 までの整数のいずれかが入る 1 次元配列として表現し、この遺伝子を進化過程における 1 個体とする。この個体を複数作成し、初期遺伝子列とした。

#### 3.2 適応度の算出

選択された遺伝子の評価のために問題に対する適応性を適応度として算出する。すなわち、図3に示すように縦(列)方向は、その日の勤務割り当て人数と等し

いかを比較し、等しければ遺伝子に対して1ポイントを与える。横(行)方向は、まず各々の勤務パターンの平均勤務数(AVE*i*)を式(2)により計算する。

$$AVE_i = [\text{割り当て人数} \times n \div m]_{\text{int}} \pm 2 \quad \dots\dots(2)$$

$(i=0, 1, \dots, P-1)$

ここで、[ ]<sub>int</sub> は整数化処理を表すものとする。±2 は収束のための自由度として加えている。式(2)により求められた値と表の勤務数を比較し等しければ、さらにこの遺伝子に1ポイント加える。

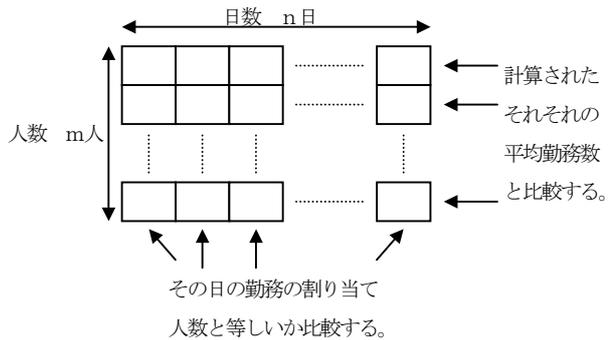


図3. 遺伝子に対するポイントの与え方

最終的に、このポイント数が  $m + n$  ポイントになった場合は最適解が計算されたことになる。すべての遺伝子のいずれかで最適解が求められなかった場合は、それぞれの遺伝子のポイント数 POINT*i* ( $i=1, 2, 3, \dots, IND$ ) より適応度を計算する。適応度を  $G_i$  とすると、本研究では式(3)により計算した。

$$G_i = \frac{POINT_i - \min}{\max - \min} \quad \dots\dots(3)$$

$(i=1, 2, 3, \dots, IND)$

ここで、max は POINT*i* の最大値、min は最小値、IND は生成した個体数とする。式(3)により  $G_i$  は 0.0 ~ 1.0 までの実数値をとり、ポイント数が一番高い遺伝子の適応度は 1.0 になる。

#### 3.3 選択処理

次は、次世代個体の選択処理である。選択の方法には、GA において代表的なルーレット方式を採用した。ルーレット方式とは、図4に示すように次世代の個体を選択する際、乱数により選択する方法である。しかし、これだけでは適応度の高い遺伝子が選択されない可能性がある。そこで、しきい値を設定し、適応度がそのしきい値より高い遺伝子はすべて選択し、不足分を乱数によるルーレット方式で選択するようにした。

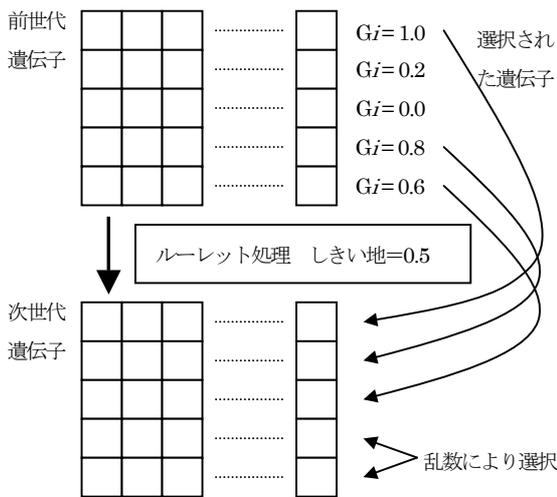


図4. 遺伝子の選択

### 3.4 交叉処理

次は、交叉処理である。交叉は最も基本的な交叉である一点交叉を採用した。一点交叉では交叉の起こる確率(交叉率)を設定する。その後、すべての遺伝子に対し0.0 ~ 1.0の実数型の乱数を発生させ、それが交叉率より低ければ交叉する。交叉する遺伝子には、さらに1 ~  $m \times n$ の整数型の乱数を生成し交叉点を決定、その交差点を基準にそれ以降を次の遺伝子と入れ替える。それにより次世代に新しい遺伝子が生成されることになる。

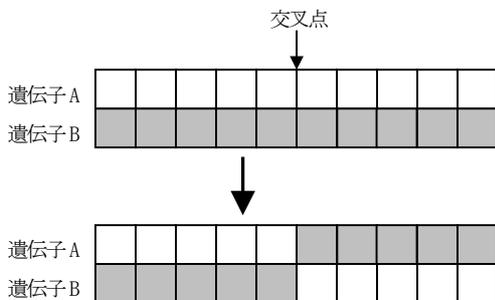


図5. 一点交叉

### 3.5 突然変異処理

次は、突然変異処理である。突然変異は局所解に陥るのを防ぐために使用する。突然変異の場合も一点交叉の場合と同様に、突然変異の起こる確率(突然変異率)を設定する。それぞれの遺伝子に対し0.0 ~ 1.0の実乱数を発生させ、それが突然変異率より低ければ突然変異させる。さらに、突然変異の対象となる遺伝子には1 ~  $m \times n$ の整数乱数を生成し遺伝子中で突然変異させる位置を決定する。その後、0 ~ P-1の

整数型の乱数によって新しい値と入れ替える。

以上の操作をSGAのフローチャートに従い、いずれかの遺伝子でポイント数(PPOINT*i*)が  $m+n$  ポイントになるまで繰り返す。この一連の作業により処理が終了した場合は、最適解が求められている。

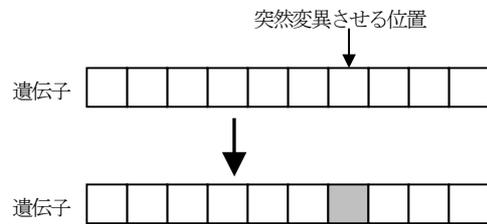


図6. 突然変異

## 4. 制約条件

本研究では、看護師の勤務表を作成することを目的にしているため、制約条件を設定する。夜勤の翌日は必ず休みにすることにする。これにより初期計算時には

$$\text{夜勤の人数} \leq \text{休みの人数}$$

を満足しなければならない。

## 5. 実験装置

以上までのアルゴリズムに基づいてシステムを構築した。今回の実験で使用した実験装置は以下である。

- PC ノート型パソコン(A4)
- CPU Intel Pentium Processor III (800MHz)
- メモリ 256MB
- OS Microsoft Windows Me
- 開発環境 Microsoft VC++ 6.0

## 6. 実験

構築したシステムの適用可能性を検証するためにいくつかの実験を行った。

### 6.1 実験(1)

解を計算する上で最適な交叉率と突然変異率との関係を調べた。生成した遺伝子の個体数(INV)を20とし、しきい値0.5に固定、交叉率を0.1 ~ 0.9、突然変異率を0.2 ~ 0.9まで変化させた。表1に、それぞれ計算20回での最適解が求まるまでの世代交代数の平均値を示す。

	突然変異率							
	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.1	24444	7476	15940	108495				
0.2	33930	13966	7551	9333				
0.3	132733	15888	6026	6623	11953			
交 0.4	56309	28959	8266	6483	7660			
又 0.5		29663	8366	5143	4755	5565		
率 0.6		13596	12268	5836	4142	4803	67062	
0.7			11822	7615	3558	3667	4798	22145
0.8			13067	9722	3583	3922	4694	9389
0.9			26994	4935	5436	3531	3431	7135

しきい値0.5 勤務日数20日 夜2人 休3人 昼4人

表1. 交叉率と突然変異率の関係(1)

その他のパラメーターは、日数20日・夜2人・休3人・昼4人である。交叉率が0.7～0.9 突然変異率が0.6～0.7 の場合は比較的少ない世代交代数で解が得られることがわかった。

次に、交叉率・突然変異率と選択処理のしきい値の関係を調べるために平均世代交代数が3000世代で最適解が求められた6箇所の値を使用し、しきい値を0.3～0.8まで変化させ、その関係を調べた。表2にそれぞれ20回計算させた時の平均値を示す。

交叉率	0.7	0.7	0.8	0.8	0.9	0.9
突然変異率	0.6	0.7	0.6	0.7	0.7	0.8
0.3	4341	23855	4014	13162	5334	42526
し 0.4	3948	5898	3641	4665	3753	6129
き 0.5	3558	3667	3583	3922	3531	3431
い 0.6	4847	3560	4446	3533	3044	2723
値 0.7	8544	4805	6649	4276	4948	3333
0.8	9395	4013	9669	3160	5285	3083

勤務日数20日 夜2人 休3人 昼4人

表2. 交叉率・突然変異率としきい値の関係(1)

## 6.2 実験(2)

さらに、もう少し実用的なパラメーターとして日数31日・夜2人・休3人・昼4人・朝3人で計算を行った。しきい値を0.5に固定し、交叉率を0.4～0.9まで、突然変異率を0.3～0.7まで変化させた。その結果を表3に示す。

	突然変異率				
	0.3	0.4	0.5	0.6	0.7
0.4	362587	53707	175284		
0.5		86459	65833		
交 0.6		133646	55895	256377	
又 0.7		141129	47623	70577	
率 0.8		209261	70657	44203	
0.9			117947	47410	170560

しきい値0.5 勤務日数31日 夜2人 休3人 昼4人 朝3人

表3. 交叉率と突然変異率の関係(2)

実験(1)結果と同様に、交叉率・突然変異率としきい値の関係を調べるために、平均世代交代数が40000世代で最適解が求められた3箇所の値を使用し、しきい値を0.4～0.8まで変化させ、その関係を調べた。その結果を表4に示す。

交叉率	0.8	0.7	0.9
突然変異率	0.6	0.5	0.6
0.4	158547		100747
し 0.5	44204	47623	47410
き 0.6	50187	81955	53657
い 0.7	49620		45987
値 0.8	47561		39917

勤務日数31日 夜2人 休3人 昼4人 朝3人

表4. 交叉率・突然変異率としきい値の関係(2)

以上の結果から、交叉率0.8 突然変異率0.6 しきい値0.5 の場合が比較的少ない世代交代数で解が得られることがわかった。

## 6.3 実験(3)

次に、勤務パターン数と収束の関係を調べるために勤務人数の合計を11人としてパターン数を2～5まで変化させて計算する。その他のパラメーターは勤務日数20日・交叉率0.8 突然変異率0.6 しきい値0.5とした。その結果を表5に示す。

パターン	夜	4(人)	3	2	2
休	7	5	3	3	3
昼	0	3	3	3	2
朝	0	0	3	3	2
夕	0	0	0	0	2
平均世代交代数	738(世代)	2091	9335	58790	

勤務日数20日・交叉率0.8 突然変異率0.6 しきい値0.5

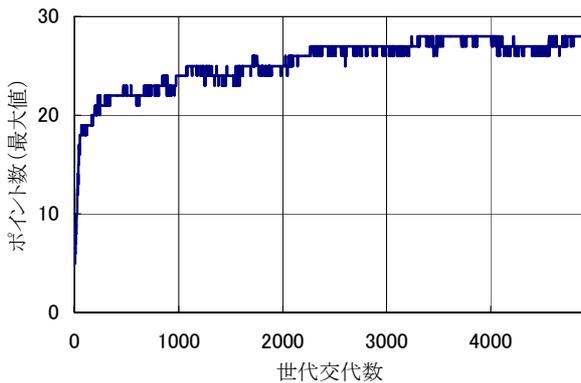
表5. 勤務パターン数と収束の関係

この結果より、合計勤務人数が同じでもパターン数が増えるにつれ、より多くの世代交代が必要なことがわかる。

#### 6.4 実験(4)

次に、解への収束までの世代交代数と最大ポイント数の関係を示す。図7は、勤務日数20日・交叉率0.8・突然変異率0.6・しきい値0.5・夜2人・休3人・昼4人での結果である。図8は、勤務日数31日・交叉率0.8・突然変異率0.6・しきい値0.5・夜2人・休3人・昼4人・朝3人での結果である。

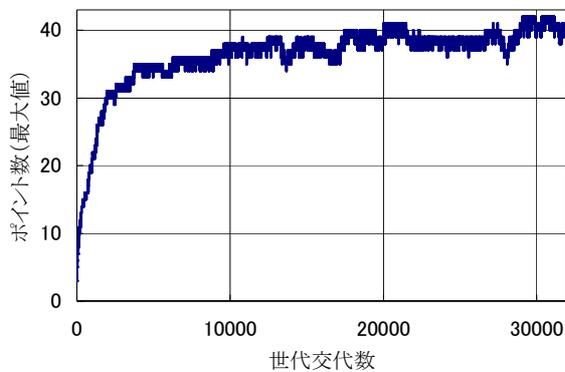
いずれの場合も同様に比較的早い回数で80%以上のポイント数を獲得する。これは、選択により優秀な遺伝子が優先的に選ばれるために80%近くまでは早く収束する。しかし、その後は同じ性質の遺伝子が残るために選択と交叉の影響が少なくなり突然変異への依存が大きくなる。その結果、80%を越えたあたりからポイントの獲得率が悪くなる。



勤務日数20日 交叉率0.8 突然変異率0.6

しきい値0.5 夜2人 休3人 昼4人

図7. 世代交代数とポイント数(1)



勤務日数31日 交叉率0.8 突然変異率0.6

しきい値0.5 夜2人 休3人 昼4人 朝3人

図8. 世代交代回数とポイント数(2)

#### 7. まとめ

ナーススケジューリング問題の解決のため GA を用いた新しい手法を提案し、その有用性を検証するために、いくつかの数値実験を行った。その結果から、以下の点が判明した。

- 1) 日数31日、勤務総人数15人以下では世代交代数が、100万世代までに解が得られることがわかった。
- 2) 勤務総人数が15人を超えると100万世代目でも解が収束しない場合がある。
- 3) 遺伝子のポイントの獲得率が80%を超えたところから収束させるには突然変異に依存する。

そのために、選択・交叉・突然変異の処理において以下のように、さらにポイントを獲得する工夫が必要である。

- 1) ポイントの獲得できなかった行(列)のみを突然変異させる。
- 2) 積極的に突然変異させるとポイントを獲得した行(列)を壊してしまう場合があるので、各パラメーターと突然変異率のバランスを求める。

などの検証が必要である。

今後、実用的な勤務表に近づくために氏名の入力や個人の都合による勤務の固定、変更などの要素が組み込まれるアルゴリズムの検討を行う予定である。

最後に開発したアプリケーションでの勤務表の出力例を図9・図10に示す。

#### 8. 参考文献

- (1) 米澤保雄: "遺伝的アルゴリズム 進化理論の情報科学", 森北出版株式会社
- (2) Lawrence Davis: "遺伝的アルゴリズム ハンドブック", 森北出版株式会社
- (3) David E. Goldberg: "Genetic Algorithms in Search Optimization & Machine Learning, ", Addison-Wesley Publishing Company
- (4) 林晴比古: "新 VC++5.0 入門 ビギナー編", ソフトバンク株式会社
- (5) 林晴比古: "新 VC++5.0 入門 シニア編", ソフトバンク株式会社

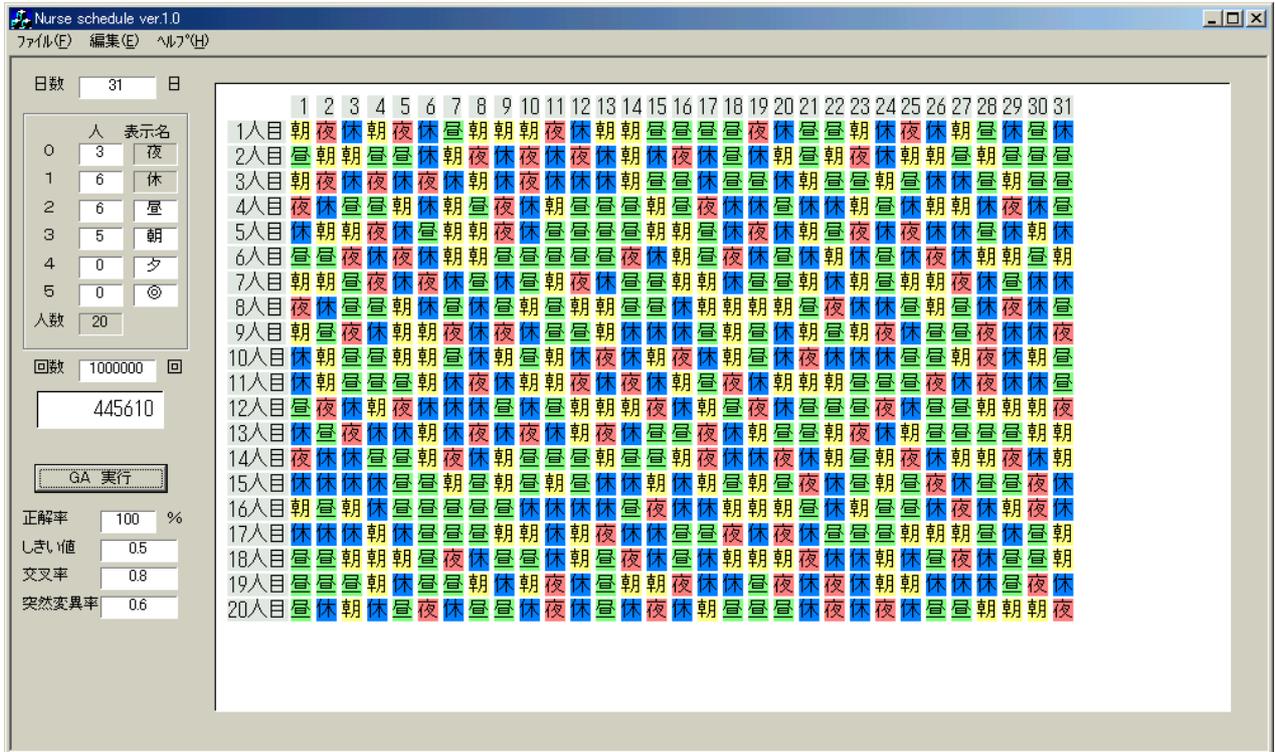


図9. アプリケーションの表示画面

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1人目	朝	夜	休	朝	夜	休	昼	朝	朝	朝	夜	休	朝	朝	昼	昼	昼	昼	夜	休	昼	昼	朝	休	夜	休	朝	昼	休	昼	休
2人目	昼	朝	朝	昼	昼	休	朝	夜	休	夜	休	夜	休	朝	休	夜	休	昼	休	朝	昼	朝	夜	休	朝	朝	昼	朝	昼	昼	昼
3人目	朝	夜	休	夜	休	夜	休	朝	休	夜	休	休	休	朝	昼	昼	休	昼	昼	休	朝	昼	昼	朝	昼	休	休	昼	朝	昼	昼
4人目	夜	休	昼	昼	朝	休	朝	昼	夜	休	朝	昼	昼	朝	昼	夜	休	休	昼	休	休	朝	昼	休	朝	朝	休	夜	休	昼	休
5人目	休	朝	朝	夜	休	昼	朝	朝	夜	休	昼	昼	昼	朝	朝	昼	休	夜	休	朝	昼	夜	休	夜	休	休	昼	休	朝	休	休
6人目	昼	昼	夜	休	夜	休	朝	朝	昼	昼	昼	昼	夜	休	朝	昼	夜	休	昼	休	朝	休	休	昼	休	夜	休	朝	朝	昼	朝
7人目	朝	朝	昼	夜	休	夜	休	昼	休	朝	夜	休	昼	昼	朝	朝	休	昼	昼	朝	休	朝	昼	朝	朝	夜	休	昼	休	休	休
8人目	夜	休	昼	昼	朝	休	昼	休	昼	朝	朝	朝	昼	昼	休	朝	朝	朝	朝	朝	昼	夜	休	休	昼	朝	昼	休	夜	休	昼
9人目	朝	昼	夜	休	朝	朝	夜	休	夜	休	昼	昼	朝	休	休	休	昼	朝	昼	休	朝	昼	朝	夜	休	昼	昼	夜	休	休	夜
10人目	休	朝	昼	昼	朝	朝	昼	休	朝	昼	朝	休	夜	休	朝	夜	休	朝	昼	休	夜	休	休	昼	昼	朝	夜	休	朝	昼	昼
11人目	休	朝	昼	昼	朝	休	夜	休	朝	朝	夜	休	夜	休	朝	昼	夜	休	朝	朝	朝	昼	昼	夜	休	昼	昼	夜	休	休	昼
12人目	昼	夜	休	朝	夜	休	休	休	昼	休	昼	朝	朝	朝	夜	休	朝	昼	夜	休	昼	昼	昼	夜	休	昼	昼	朝	朝	朝	夜
13人目	休	昼	夜	休	休	朝	休	夜	休	夜	休	朝	夜	休	昼	昼	夜	休	朝	昼	昼	朝	夜	休	朝	昼	昼	昼	朝	朝	朝
14人目	夜	休	休	昼	昼	朝	夜	休	朝	昼	昼	朝	昼	昼	朝	夜	休	休	夜	休	夜	休	朝	昼	朝	夜	休	朝	朝	夜	朝
15人目	休	休	休	休	昼	昼	朝	昼	朝	昼	朝	昼	休	休	朝	休	朝	昼	朝	昼	夜	休	昼	朝	昼	夜	休	昼	昼	夜	休
16人目	朝	昼	朝	休	昼	昼	昼	昼	昼	休	休	休	休	昼	夜	休	休	朝	朝	朝	朝	昼	休	朝	昼	昼	休	夜	休	朝	夜
17人目	休	休	休	朝	休	昼	昼	朝	朝	休	朝	夜	休	休	昼	昼	夜	休	朝	朝	朝	夜	休	昼	昼	朝	朝	朝	昼	休	朝
18人目	昼	昼	朝	朝	朝	昼	夜	休	昼	昼	休	朝	昼	夜	休	昼	休	朝	朝	朝	夜	休	休	朝	休	昼	夜	休	昼	昼	朝
19人目	昼	昼	昼	朝	休	昼	昼	朝	休	朝	夜	休	昼	朝	朝	夜	休	休	昼	夜	休	夜	休	朝	朝	休	休	休	休	昼	夜
20人目	昼	休	朝	休	昼	夜	休	昼	昼	休	夜	休	昼	夜	休	朝	昼	昼	昼	休	夜	休	夜	休	夜	休	昼	朝	朝	朝	夜

勤務日数31日 交叉率0.8 突然変異率0.6 しきい値0.5 夜3人 休6人 昼6人 朝5人

図10. 勤務表の出力例