

明 細 書

発明の名称：

スケジュール作成支援装置およびスケジュール作成支援方法

技術分野

[0001] 本発明は、スケジュール作成支援装置およびスケジュール作成支援方法に関する。

背景技術

[0002] 所定条件下で所望のパラメータを最大または最小とする解を探索する、いわゆる組合せ最適化問題の概念は、交通渋滞解消、グローバルサプライチェーンにおける物流コスト低減、など実社会における複雑な問題にも適用される。

[0003] 一方、そうした問題においては解候補が爆発的に多くなるため、スーパーコンピュータや量子コンピュータなど相応の計算能力を有した計算機でなければ、当該問題を実用的な時間内に解くことが難しい。

[0004] 例えば、量子コンピュータに関連する従来技術としては、全数探索を必要とするような逆問題や組み合わせ最適化問題に対して高速演算を可能にする計算機に関し、スピンを演算における変数とし、解こうとする問題をスピン間相互作用とスピンごとに作用する局所場で設定し、また、時刻 $t = 0$ において外部磁場により全スピンを一方向に向かせ、時刻 $t = \tau$ で外部磁場がゼロになるように外部磁場を徐々に小さくし、また、各スピンは時刻 t における各サイトの外部磁場及びスピン間相互作用のすべての作用で決まる有効磁場に従い向きが定まるとして時間発展させ、その際、スピンの向きが有効磁場に完全に揃うのではなく、量子力学的に補正された向きとすることにより、系が基底状態をほぼ維持するようにする技術（特許文献 1 参照）などが提案されている。

先行技術文献

特許文献

[0005] 特許文献1：WO 2 0 1 6 / 1 5 7 3 3 3

発明の概要

発明が解決しようとする課題

[0006] ところが、多くの人員が協働する際の全体スケジュールの作成業務に関して、上述のごとき量子コンピュータ技術を適宜に適用する形態は提案されていない。

[0007] 例えば、コールセンタにおけるスケジュール作成業務では、数百人規模のオペレータに関して週間、月間のシフトスケジュールを作成する必要がある。一方、現状では、経験のある担当者が、予め決まったルール（例：早番、遅番といったシフトパターンを、該当期間に関して固定的な順序、頻度で組合せて配置する）の下、人力でスケジュール作成を行っていた。

[0008] こうしたスケジュール作成は、一般的なPCで表計算ソフトウェアにおける関数を用いて相応に効率化することも可能である。ところが、人員数が一定規模を超えてくる場合、或いは、複数条件が影響し合う非線形な制約条件が存在する場合、現実的な時間で解を得ることができない。よって、ごく小規模な組織で線形の制約条件しか存在しない状況にしか対応できていない。

[0009] 多くの人員が協働する場合のスケジュール作成は、各人員の勤務日数規定、各日の必要人員数といった基本的な要因の他にも、各人員の都合や曖昧な希望、勤務体系、各種規則など様々な要因によって大きな影響を受ける。また、その要因自体も人員や所属組織、業務需要によって大きく変動し、かつ要因同士も競合し影響しあうケースがある。

[0010] しかも、業務需要の急増／急減、人員の急な欠勤など、要因の急変動が生じるケースもある。こうした状況に一般的なPCを採用して従来どおりスケジュール作成を行うとしても、上述の要因数や非線形な制約条件などに応じて計算量が指数関数的に増加し、膨大な時間を要するかオーバーフローに至ることとなる。つまり、必要なタイミングでスケジュール作成を行うことが期待できない。

[0011] そこで本発明の目的は、協働する多くの人員に関して非線形な制約条件を

踏まえたスケジュール作成を効率的に行う技術を提供することにある。

課題を解決するための手段

[0012] 上記課題を解決する本発明のスケジュール作成支援装置は、所定業務で協働する各人員の所定期間における総勤務長、前記期間における各タイミングでの必要人数、および前記業務に前記各人員を割り当てる際の制約条件、の各情報を格納した記憶部と、前記期間における前記総勤務長、前記必要人数、および前記制約条件が満たされる際に最小となる制約条件用関数、を項として含む目的関数に関して、前記各人員の出勤有無をスピンとし、前記制約条件用関数における変数間の感応度を前記スピンの間の相互作用の強度として設定したイジングモデルを演算する演算部とを有し、前記演算部は前記演算の結果に基づき、前記所定期間の前記各タイミングにおける前記各人員の出勤有無を規定したスケジュールを所定装置に出力することを特徴とする。

[0013] また、本発明のスケジュール作成支援方法は、所定業務で協働する各人員の所定期間における総勤務長、前記期間における各タイミングでの必要人数、および前記業務に前記各人員を割り当てる際の制約条件、の各情報を格納した記憶部を備える情報処理装置が、前記期間における前記総勤務長、前記必要人数、および前記制約条件が満たされる際に最小となる制約条件用関数、を項として含む目的関数に関して、前記各人員の出勤有無をスピンとし、前記制約条件用関数における変数間の感応度を前記スピンの間の相互作用の強度として設定したイジングモデルを演算し、前記演算の結果に基づき、前記所定期間の前記各タイミングにおける前記各人員の出勤有無を規定したスケジュールを所定装置に出力する、ことを特徴とする。

発明の効果

[0014] 本発明によれば、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成を効率的に行うことが可能となる。

図面の簡単な説明

[0015] [図1]本実施形態のスケジュール作成支援装置を含むネットワーク構成図である。

[図2]本実施形態におけるスケジュール作成支援装置のハードウェア構成例を示す図である。

[図3]本実施形態におけるタイミングチャート例を示す図である。

[図4]本実施形態におけるフロー例1を示す図である。

[図5]本実施形態における基本情報テーブルの構成例を示す図である。

[図6]本実施形態における制約条件テーブルの構成例を示す図である。

[図7]本実施形態における制約条件テーブルの構成例を示す図である。

[図8]本実施形態におけるスケジュール作成支援方法を示すフロー図である。

[図9]本実施形態における画面例を示す図である。

発明を実施するための形態

[0016] ーアーニーリングマシンについてー

上述の特許文献1にも示すように、本出願人は量子コンピューティング技術を開発し、例えば、ビッグデータに基づく全数探索問題（組合せ最適化問題の概念含む）における諸問題の解決を図ってきた。

[0017] こうした全数探索問題に対して、一般的には量子コンピュータへの期待が大きい。量子コンピュータは、量子ビットと呼ばれる基本素子からなり“0”と“1”を同時に実現する。そのためすべての解候補を初期値として同時に計算可能であり、全数探索を実現しうる可能性を持っている。しかし、量子コンピュータは全計算時間に亘って量子コヒーレンスを維持する必要がある。

[0018] こういった中で注目されるようになってきたのが断熱量子計算と呼ばれる手法である（参考文献：E. Farhi, et al., "A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem," Science 292, 472 (2001).）。この方法は、ある物理系の基底状態が解になるように問題を変換し、基底状態を見つけることを通して解を得ようとするものである。

[0019] 問題を設定した物理系のハミルトニアンを $\hat{H}_p$ とする。但し、演算開始時点ではハミルトニアンを $\hat{H}_p$ とするのではなく、それとは別に基底状態が明確で準備しやすい別のハミルトニアン $\hat{H}_0$ とする。次に十分に時間を掛けてハミルトニアンを $\hat{H}_0$ から $\hat{H}_p$ に移行させる。十分に時間を掛ければ系は基底状態に居続け、ハミルトニアン $\hat{H}_p$ の基底状態が得られる。これが断熱量子計算の原理である。計算時間を τ とすればハミルトニアンは式（１）となり、

[式１]

$$\hat{H}(t) = \left(1 - \frac{t}{\tau}\right) \hat{H}_0 + \frac{t}{\tau} \hat{H}_p$$

式（２）のシュレディンガー方程式に基づいて時間発展させて解を得る。

[0020] [式２]

$$i\hbar \frac{\partial}{\partial t} |\psi(t)\rangle = \hat{H}(t) |\psi(t)\rangle$$

断熱量子計算は全数探索を必要とする問題に対しても適用可能で、一方向性の過程で解に到達する。しかし、計算過程が式（２）のシュレディンガー方程式に従う必要があるならば、量子コンピュータと同様に量子コヒーレンスの維持が必要になる。

[0021] 但し、量子コンピュータが１量子ビットあるいは２量子ビット間に対するゲート操作を繰り返すものであるのに対して、断熱量子計算は量子ビット系全体に亘って一斉に相互作用させるものであり、コヒーレンスの考え方が異なる。

[0022] 例えば、ある量子ビットへのゲート動作を考えてみる。この時、もしその量子ビットと他の量子ビットとで相互作用があれば、それはディコヒーレン

スの原因になるが、断熱量子計算ではすべての量子ビットを同時に相互作用させるので、この例のような場合にはディコヒーレンスにならない。この違いを反映して断熱量子計算は量子コンピュータに比べてディコヒーレンスに対して頑強であると考えられている。

[0023] 以上述べたように、断熱量子計算は全数探索を必要とするような難問に対して有効である。そして、スピンを演算における変数とし、解こうとする問題をスピン間相互作用とスピンごとに作用する局所場で設定する。

[0024] 時刻 $t = 0$ において外部磁場により全スピンを一方向に向かせ、時刻 $t = \tau$ で外部磁場がゼロになるように外部磁場を徐々に小さくする。

[0025] 各スピンは、時刻 t における各サイトの外部磁場及びスピン間相互作用のすべての作用で決まる有効磁場に従い、向きが定まるとして時間発展させる。

[0026] その際、スピンの向きが有効磁場に完全に揃うのではなく、量子力学的に補正された向きとすることにより、系が基底状態をほぼ維持するようにする。

[0027] また、時間発展の際に各スピンを元の向きに維持する項（緩和項）を有効磁場に加え、解の収束性を向上させる。

[0028] 本実施形態におけるスケジュール作成支援装置としては、上述の断熱量子計算を行うアニーリングマシンを想定するが、勿論これに限定するものではなく、組合せ最適化問題を本発明のスケジュール作成支援方法に沿って適宜に解くことが可能なものであればいずれも適用可能である。

[0029] 具体的には、アニーリング方式において電子回路(デジタル回路など)で実装するハードウェアだけでなく、超伝導回路などで実装する方式も含む。また、アニーリング方式以外にてイジングモデルを実現するハードウェアでもよい。例えばレーザーネットワーク方式(光パラメトリック発振)・量子ニューラルネットワークなども含む。また、前述した通り一部の考え方が異なるものの、イジングモデルで行う計算をアダマールゲート、回転ゲート、制御NOTゲートといったゲートで置き換えた量子ゲート方式においても、本発明

を実現することができる。

[0030] ーネットワーク構成ー

以下に本発明の実施形態について図面を用いて詳細に説明する。図1は、本実施形態のスケジュール作成支援装置100を含むネットワーク構成図である。

[0031] 図1に示すスケジュール作成支援装置100は、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成を効率的に行うコンピュータ装置であり、具体的には、一例としてアニーリングマシンを想定する。

[0032] ただし、アニーリングマシンの概要は特許文献1に基づき既に述べたとおりであり、その具体的な構成や動作等の詳細については適宜省略する（以下同様）。

[0033] 本実施形態のスケジュール作成支援装置100は、インターネットなどの適宜なネットワーク10を介して、ユーザ端末200とデータ通信可能に接続されている。

[0034] 上述のユーザ端末200は、スケジュール作成支援装置100からスケジュールの提案を受ける端末である。

[0035] このユーザ端末200のユーザとしては、具体的には、多くのオペレータを抱えてコールセンタ業務を遂行する事業者で、金融機関や保険会社、或いは大手メーカーといった組織を想定できる。或いは、多くの看護師や介護スタッフを抱えて患者等の看護業務や介護業務を遂行する、医療機関や介護事業者も想定できる。

[0036] いずれにしても、相応規模の人員を日付や時間帯ごとに必要数だけ割り当てて、全体として業務を遂行する事業者であれば、上述のユーザに該当する。つまり、そうした事業者の業務に関しては、本発明が適用可能であると言える。

[0037] 具体例として想定できる、例えば銀行のコールセンタにおけるスケジュール作成に際しては、所定の経験を持った担当者が、数百人規模の全オペレータの月間シフトスケジュールを人力で作成しているのが現実であった。つま

り当該業務は属人化しがちであり、また人力での作業である。

[0038] そのため、従来どおり固定化された規則でのシフトを踏まえたスケジュール作成は可能であっても、イレギュラーなシフトや外的要因に対して臨機応変に対処したスケジュール作成は困難である。

[0039] 一方、表計算ソフトウェアにおける関数等を利用した場合、「休日設定は土日固定」、「一度に計算できる人員数は１００人まで」、などといった現実的な障壁から離れてスケジュール作成を行うことは難しい。

[0040] 上記の障壁は、数学的な困難さおよびその数学的困難さを解決する技術不在（厳密には未成熟）に起因する。表計算ソフトウェアは、線形の制約条件であれば数学的に高速に解けるものであるが、非線形の制約条件を解くには膨大な時間がかかるという性質を持つ。

[0041] このように従来であれば、スケジュール作成に際し、要素すなわち各人員などに関する非線形の制約条件の増加に対して計算量が指数関数的に増加し、計算完了までに長時間を要するが、アニーリングマシンを使用したスケジュール作成支援装置１００を採用することで、要素の増加にさほど依存せず計算を行うことが可能となる。

[0042] ــــــــハードウェア構成ــــــــ

また、本実施形態のスケジュール作成支援装置１００のハードウェア構成は、図２に以下の如くとなる。

[0043] すなわちスケジュール作成支援装置１００は、記憶部１０１、メモリ１０３、演算部１０４、および通信部１０５、を備える。

[0044] このうち記憶部１０１は、SSD（Solid State Drive）やハードディスクドライブなど適宜な不揮発性記憶素子で構成される。

[0045] また、メモリ１０３は、RAMなど揮発性記憶素子で構成される。

[0046] また、演算部１０４は、記憶部１０１に保持されるプログラム１０２をメモリ１０３に読み出すなどして実行し装置自体の統括制御を行なうとともに各種判定、演算及び制御処理を行なうCPUである。

[0047] また、通信部１０５は、ネットワーク１０と接続してユーザ端末２００と

の通信処理を担うネットワークインターフェイスカード等を想定する。

[0048] なお、スケジュール作成支援装置100がスタンドアロンマシンである場合、ユーザからのキー入力や音声入力を受け付ける入力装置、処理データの表示を行うディスプレイ等の出力装置、を更に備えたとすれば好適である。

[0049] また、記憶部101内には、本実施形態のスケジュール作成支援装置として必要な機能を実装する為のプログラム102に加えて、基本情報テーブル125および制約条件テーブル126が少なくとも記憶されている。ただし、これらテーブルについての詳細は後述する。

[0050] また、プログラム102、すなわちアニーリングマシンとしての動作を実装するアルゴリズムは、解くべき課題であるイジングモデル1021の情報を保持する。このイジングモデル1021は、情報提供の対象となる業務や人員、およびそれらに影響を与える他の各種情報に基づき管理者等が予め設定しておくものとなる。

[0051] なお、アニーリングマシンの概要にて述べた断熱量子計算は、別名で量子アニールとも呼ばれ、古典的な焼きなましの概念を量子力学に発展させたものである。即ち、断熱量子計算は本来古典的動作が可能で、高速性や解の正解率に関して性能を向上させるために量子力学的効果が付加されたものとも解釈できる。そこで本発明では、演算部そのものは古典的とし、演算過程に量子力学的に定まるパラメータを導入することにより、古典的であるが量子力学的な効果を含んだ演算方法・装置を実現する。ただし、演算部を量子コンピュータで構成する形態についても勿論採用しうる。

[0052] 以上の概念に基づき、以下の例では断熱量子計算との関連性を説明しながら解としての基底状態を得る古典的アルゴリズムと、それを実現するための装置に関して述べる。

[0053] こうした前提でのスケジュール作成支援装置100は、 N 個の変数 s_j ($j = 1, 2, \dots, N$) が $-1 \leq s_j \leq 1$ の値域を取り、局所場 g_j と変数間相互作用 J_{ij} ($i, j = 1, 2, \dots, N$) によって課題の設定がなされる。

[0054] また演算部104では、時刻を m 分割して離散的に $t = t_0$ ($t_0 = 0$) から t_m ($t_m = \tau$) まで演算するものとし、各時刻 t_k における変数 $S_j^z(t_k)$ を求めるに当たり、前時刻 t_{k-1} の変数 $S_j^z(t_{k-1})$ ($i = 1, 2, \dots, N$) の値と緩和項の係数 g_j あるいは $g_j + \delta g_j$ を用いて $B_j^z(t_k) = \{\sum_i J_{ij} S_i^z(t_{k-1}) + g_j + \delta g_j \cdot S_j^z(t_{k-1})\} \cdot t_k / \tau$ あるいは $B_j^z(t_k) = \{\sum_i J_{ij} S_i^z(t_{k-1}) + g_j + \delta g_j \cdot S_j^z(t_{k-1})\} \cdot t_k / \tau$ を求め、上述の変数 $S_j^z(t_k)$ の値域が $-1 \leq S_j^z(t_k) \leq 1$ になるように関数 f を定めて $S_j^z(t_k) = f(B_j^z(t_k), t_k)$ とし、時刻ステップを $t = t_0$ から $t = t_m$ に進めるにつれて上述の変数 S_j^z を -1 あるいは 1 に近づけ、最終的に $S_j^z < 0$ ならば、 $S_j^z = -1$ 、 $S_j^z > 0$ ならば、 $S_j^z = 1$ として解を定める。

[0055] 係数 g_j は、例えば $|J_{ij}|$ の平均値の50%から200%の値である。また、課題設定の局所場 g_j に関して、あるサイト j' に対してのみ補正項 δg_j を g_j に加え、該サイト j' に対してのみ g_j の大きさを大きくすることもできる。また、補正項 δg_j は、例えば $|J_{ij}|$ の平均値の10%から100%の値である。

[0056] 続いて、量子力学的な記述から出発して古典的な形式に移行することを通して、アニーリングマシンの基本的原理を述べる。

[0057] 式(3)で与えられるイジングスピン・ハミルトニアン基底状態探索問題はNP困難と呼ばれる分類の問題を含み、有用な問題であることが知られている(文献:F. Barahona, "On the computational complexity of Ising spin glass models," J. Phys. A: Math. Gen. 15, 3241 (1982).)。

[0058] [式3]

$$\hat{H}_p = -\sum_{i>j} J_{ij} \hat{\sigma}_i^z \hat{\sigma}_j^z - \sum_j g_j \hat{\sigma}_j^z$$

J_{ij} 及び g_j が課題設定パラメータであり、 σ^z はパウリのスピン行列の z 成分で ± 1 の固有値を取る。 i, j はスピンのサイトを表す。イジングスピンとは値として ± 1 だけを取りうる変数のことで、式 (3) では σ^z の固有値が ± 1 であることによりイジングスピン系となっている。

[0059] 式 (3) のイジングスピンは文字通りのスピンである必要はなく、ハミルトニアンが式 (3) で記述されるのであれば物理的には何でも良い。

[0060] 例えば、各人員の出勤有無を ± 1 に対応付けることや、ロジック回路の $high$ と low を ± 1 に対応付けることも可能であるし、光の縦偏波と横偏波を ± 1 に対応付けることや $0, \pi$ の位相を ± 1 に対応付けることも可能である。

[0061] ここで例示する方法では、断熱量子計算と同様に、時刻 $t = 0$ において式 (4) で与えられるハミルトニアンの基底状態に演算系を準備する。

[0062] [式 4]

$$\hat{H}_0 = -\gamma \sum_j \hat{\sigma}_j^x$$

γ は全サイト j に一様に掛かる外場の大きさで決まる比例定数であり、 σ_j^x は、パウリのスピン行列の x 成分である。演算系がスピンそのものであれば、外場とは磁場を意味する。

[0063] 式 (4) は、横磁場を印加したことに相当し、すべてのスピンの x 方向を向いた場合 ($\gamma > 0$) が基底状態である。問題設定のハミルトニアンは z 成分のみのイジングスピン系として定義されたが、式 (4) にはスピンの x 成分が登場している。従って、演算過程でのスピンはイジングではなくベクトルの (ブロッホベクトル) である。 $t = 0$ では式 (4) のハミルトニアンでスタートしたが、時刻 t の進行と共に徐々にハミルトニアンを変化させ、最終的には式 (3) で記述されるハミルトニアンにしてその基底状態を解として得る。

[0064] [式5]

$$\hat{H} = -\mathbf{B} \cdot \hat{\boldsymbol{\sigma}}$$

ここで $\hat{\boldsymbol{\sigma}}$ はパウリのスピン行列の3成分をベクトルとして表示している。基底状態はスピンの向きが磁場方向を向いた場合で、 $\langle \cdot \rangle$ を量子力学的期待値として $\langle \hat{\boldsymbol{\sigma}} \rangle = \mathbf{B} / |\mathbf{B}|$ と書ける。断熱過程では常に基底状態を維持しようとするので、スピンの向きは常に磁場の向きに追従する。

[0065] 以上の議論は多スピン系にも拡張できる。 $t=0$ ではハミルトニアンが式(4)で与えられる。これは全スピンに対して一様に磁場 $\mathbf{B}_j^x = \gamma$ が印加されたことを意味する。 $t>0$ では、磁場のx成分が徐々に弱まり $\mathbf{B}_j^x = \gamma(1-t/\tau)$ である。z成分に関してはスピン間相互作用があるために有効磁場としては式(6)になる。

[0066] [式6]

$$\hat{B}_j^z(t) = \frac{t}{\tau} \left(\sum_{i \neq j} J_{ij} \hat{\sigma}_i^z + g_j \right)$$

スピンの向きは $\langle \hat{\sigma}^z \rangle / \langle \hat{\sigma}^x \rangle$ で規定できるので、スピンの向きが有効磁場に追従するならば式(7)によりスピンの向きが定まる。

[0067] [式7]

$$\langle \hat{\sigma}_j^z \rangle / \langle \hat{\sigma}_j^x \rangle = \langle \hat{B}_j^z(t) \rangle / \langle \hat{B}_j^x(t) \rangle$$

式(7)は量子力学的記述であるが期待値を取っているので、式(1)～(6)とは異なり古典量に関する関係式である。

[0068] 古典系では量子力学の非局所相関(量子縫れ)がないので、スピンの向きはサイトごとの局所場により完全に決まるはずであり、式(7)が古典的スピン系の振る舞いを決定する。量子系では非局所相関があるために式(7)

は変形されることになるが、それに関しては後述することとし、ここでは発明の基本形態を述べるために式(7)で定まる古典系について記述する。

- [0069] 図3にスピン系の基底状態を得るためのタイミングチャート(1)を示す。図3の記述は古典量に関するものなので、サイト j のスピンを σ^j ではなく s^j により表した。またそれに伴い、図3の有効磁場 B^j は古典量である。 $t=0$ において全サイトで右向き(正)の有効磁場 B^j が印加され、全スピン S^j が右向きに初期化される。
- [0070] 時間 t の経過に従い、徐々に z 軸方向の磁場とスピン間相互作用が加えられ、最終的にスピンは $+z$ 方向あるいは $-z$ 方向となって、スピン S^j の z 成分が $s^j_z = +1$ あるいは -1 となる。時間 t は連続的であることが理想であるが、実際の演算過程では離散的にして利便性を向上させることもできる。以下では離散的な場合を述べる。
- [0071] ここで例示するスピンは z 成分だけでなく x 成分が加わっているためにベクトル的なスピンになっている。図3からもベクトルとしての振る舞いが理解できる。ここまで y 成分が登場してこなかったが、それは外場方向を xz 面に取ったために外場の y 成分が存在せず、従って $\langle \sigma^y \rangle = 0$ となるためである。
- [0072] 演算系のスピンとしては大きさ1の3次元ベクトル(これをブロッホベクトルと呼び、球面上の点で状態を記述できる)を想定しているが、図に示す例における軸の取り方では2次元のみを考慮すればよい(円上の点で状態を記述できる)。
- [0073] また γ は一定なので $B^j \times (t) > 0$ ($\gamma > 0$) あるいは $B^j \times (t) < 0$ ($\gamma < 0$) が成り立つ。この場合、2次元スピンベクトルは半円のみで記述できることになり、 $[-1, 1]$ で S^j_z を指定すれば S^j_z の1変数で2次元スピンベクトルが定まる。従って、ここでの例では、スピンは2次元ベクトルであるが、値域を $[-1, 1]$ とする1次元連続変数として表記することもできる。
- [0074] 図3のタイミングチャートでは時刻 $t = t_k$ においてサイトごとに有効磁

場を求め、その値を用いて式（８）により $t = t_k$ におけるスピンの向きを求める。

[0075] 〔式 8〕

$$s_j^z(t_k)/s_j^x(t_k) = B_j^z(t_k)/B_j^x(t_k)$$

式（８）は式（７）を古典量に関する表記に書き改めたものなので $\langle \cdot \rangle$ の記号が付いていない。次に、 $t = t_{k+1}$ の有効磁場を $t = t_k$ におけるスピンの値を用いて求める。各時刻の有効磁場を具体的に書けば式（９）及び（１０）となる。

[0076] 〔式 9〕

$$B_j^x(t_{k+1}) = \left(1 - \frac{t_{k+1}}{\tau}\right) \gamma$$

[式 10]

$$B_j^z(t_{k+1}) = \frac{t_{k+1}}{\tau} \left(\sum_{i \neq j} J_{ij} s_i^z(t_k) + g_j \right)$$

以下、図 3 のタイミングチャートで模式的に示した手順に従い、スピンと有効磁場を交互に求めていく。

[0077] 古典系ではスピンベクトルの大きさは 1 である。この場合スピンベクトルの各成分は、 $\tan \theta = B_j^z(t_k) / B_j^x(t_k)$ で定義される媒介変数 θ を用いて $S_j^z(t_k) = \sin \theta$ 、 $S_j^x(t_k) = \cos \theta$ と記述される。

[0078] これを書き直せば、 $S_j^z(t_k) = \sin(\arctan(B_j^z(t_k) / B_j^x(t_k)))$ 、 $S_j^x(t_k) = \cos(\arctan(B_j^z(t_k) / B_j^x(t_k)))$ である。

[0079] 式（９）から明らかなように $B_j^x(t_k)$ の変数は、 t_k のみであり、 τ

と γ は定数である。従って、 $S_j^z(t_k) = \sin(\arctan(B_j^z(t_k)/B_j^x(t_k)))$ 及び $S_j^x(t_k) = \cos(\arctan(B_j^z(t_k)/B_j^x(t_k)))$ は $B_j^z(t_k)$ と t_k を変数とする関数として $S_j^z(t_k) = f_1(B_j^z(t_k), t_k)$ 及び $S_j^x(t_k) = f_2(B_j^z(t_k), t_k)$ のような一般化した表現もできる。

[0080] スピンを2次元ベクトルとして記述しているので、 $S_j^z(t_k)$ と $s_j^x(t_k)$ の2成分が登場しているが、 $B_j^z(t_k)$ を式(10)に基づき決定するならば $S_j^x(t_k)$ は必要ない。

[0081] これは、 $[-1, 1]$ を値域とする $S_j^z(t_k)$ のみでスピン状態を記述できることに対応している。最終的な解 S_j^{zd} は、 $S_j^{zd} = -1$ or 1 になる必要があり、 $S_j^z(\tau) > 0$ ならば $S_j^{zd} = 1$ 、 $S_j^z(\tau) < 0$ ならば $S_j^{zd} = -1$ とする。

[0082] 図4に、上述のアルゴリズムをフローチャートにまとめたものを示す。ここで $t_m = \tau$ である。図4のフローチャートの各ステップ $s_1 \sim s_9$ は、時間 $t = 0$ から $t = \tau$ に到る図3のタイミングチャートの、ある時刻での処理に対応している。すなわち、フローチャートのステップ s_2 、 s_4 、 s_6 がそれぞれ、 $t = t_1$ 、 t_{k+1} 、 t_m における上記の式(9)及び(10)に対応している。最終的な解はステップ s_8 において、 $s_j^z < 0$ ならば $S_j^{zd} = -1$ 、 $S_j^z > 0$ ならば、 $S_j^{zd} = 1$ とすることにより定める(s_9)。

[0083] ここまでは課題が式(3)で表現された場合に如何に解かれるかを示した。次に具体的課題が如何に局所場 g_j と変数間相互作用 J_{ij} ($i, j = 1, 2, \dots, N$) を含む式(3)で表現されるかに関して具体例を挙げて説明する。

[0084] ここでの具体的課題すなわちイジングモデル1021は、例えば、コールセンタ業務に携わるオペレータ各々の所定月における総勤務日数、当該月における各日での必要人数、およびコールセンタ業務に各オペレータを割り当てる際の制約条件、の各情報をベースに、当該月における総勤務日数、当該

月における各日の必要人数、および上述の制約条件が満たされる際に最小となる制約条件用関数、を項として含む目的関数に関して、各オペレータの出勤有無をスピンとし、制約条件用関数における変数間の感応度をスピンの間の相互作用の強度として設定したイジングモデルを想定する。

[0085] この場合、局所場 g_j は、上述の当該月における総勤務日数、当該月における各日の必要人数、および上述の制約条件が満たされる際に最小となる制約条件用関数、における変数の値が目的関数へ与える影響度として設定されることを想定する。

[0086] 以上のような考察を通して、（上述の目的関数の各項の間に関する）変数間相互作用 J_{ij} と局所場 g_j を具体的に設定し、式（３）で表されるイジングモデル 1021 の基底状態探索、すなわち上述の当該月における総勤務日数、当該月における各日の必要人数、および上述の制約条件が満たされる際に最小となる制約条件用関数、からなる目的関数が最小となる基底状態の探索を通して、各オペレータの勤務シフトすなわち、コールセンタ業務における当該月の全体スケジュールを特定する。

[0087] なお、イジングモデルとアニーリング法で計算するのは、「目的関数を最小化する」ことだけである。そのため、目的関数を最小化する際に満たされる必要がある制約条件がある場合、それらを何らかの形で目的関数に足し込む必要がある。

[0088] 例えば、

[式 1 1]

$$\sum_i x_i = A$$

という制約条件を考えてみる。この制約条件を「制約条件が満たされる時に最小となる関数」に変換するとすれば、以下の式になる。

[0089] [式 1 2]

$$\left[\sum_i x_i - A \right]^2$$

二乗となっている部分は必ず正の値となるため、この式が最小値となるのは二乗の中身が0となる時だけである。中身が0となるのは $\sum x_i - A = 0$ 、の時だけであるので、この関数が最小となる最適化問題を解けば、 $\sum x_i = A$ が満たされている解が自動的に得られることになる。

[0090] また、例えば、上述のアニーリング法では、制約条件としたい項目も目的関数に入れ込んでしまう必要があるため、目的関数も制約条件も同じ重要度で扱うことになる。

[0091] 例えば、以下のような最適化問題があったとする。

[0092] [式 1 3]

$$\begin{aligned} \min & : \sum_k a_k x_k \\ \text{subject to} & : \sum_k b_k x_k = B \\ & \sum_k c_k x_k = C \end{aligned}$$

これをアニーリング用の定式化に変更すると、以下のようになる。

[0093] [式 1 4]

$$\min : \sum_k a_k x_k + P \left[\sum_k b_k x_k - B \right]^2 + Q \left[\sum_k c_k x_k - C \right]^2$$

ここでPとQは定数であり、どの項を優先的に最小化するかを決めるファクタとなる。例えば、3つの項すべてを平等に最小化する（つまり、制約条件の強さに偏りをつけないで問題を解く）場合、PとQを同等にするなど、項間でバランスをとるよう値の設定を行う。

[0094] 一方、「第2項の制約条件は厳密に守るが、第3項の制約条件はあまり重視しない」という問題設定であれば、重視する項の係数であるPの値を、Qの値より大きくすることで望みの解が得られることになる。

[0095] このようにアニーリング法によれば、制約条件に優先度を付し、あまり重視しない制約条件については「なるべく満たす」といった設定が可能になる

。

[0096] なお、イジングモデルに関する各種設定は、本実施形態のスケジュール生成の各条件、情報に応じ、既存の一般的概念に沿った形で適宜に行うものとする。

[0097] ــــــــــــــــスピン（変数）設定の具体例ــــــــــــــــ

＜タイムテーブルにおけるスピン＞

ある人員 i が、ある時間帯 j に、勤務種類 k を実施する場合に 1、実施しない場合に 0 となる変数 $x_{(i,j,k)}$ を考える（図 5 参照）。基本的にはこの変数 $x_{(i,j,k)}$ の 1 つ 1 つが CMOS アニーリングマシンにおけるスピン 1 つ 1 つに割り振られる。

[0098] 「勤務種類」を考えない場合、この変数は単純に、「ある人員 i が、ある時間帯 j に業務をするかしないか」を与える変数となる。つまり、下記の表 1 で 1 が入ったセルでは業務を実施し、0 が入ったセルでは業務を実施しない、ということになる。

[0099] [表 1]

時間帯	人員 1 ($i = 1$)	人員 2 ($i = 2$)	人員 3 ($i = 3$)	...	人員 30 ($i = 30$)	必要人数
9:00-9:10 ($j = 1$)	$x_{1,1,k}$	$x_{2,1,k}$	$x_{3,1,k}$	...	$x_{30,1,k}$	10 名
9:10-9:20 ($j = 2$)	$x_{1,2,k}$	$x_{2,2,k}$	$x_{3,2,k}$	...	$x_{30,2,k}$	12 名
9:20-9:30 ($j = 3$)	$x_{1,3,k}$	$x_{2,3,k}$	$x_{3,3,k}$	...	$x_{30,3,k}$	10 名
9:30-9:40 ($j = 4$)	$x_{1,4,k}$	$x_{2,4,k}$	$x_{3,4,k}$	...	$x_{30,4,k}$	10 名
⋮	⋮	⋮	⋮	⋮	⋮	⋮
20:50-21:00 ($j = 72$)	$x_{1,72,k}$	$x_{2,72,k}$	$x_{3,72,k}$	...	$x_{30,72,k}$	3 名
勤務コマ数	400 ± 50	400 ± 50	400 ± 50	...	400 ± 50	

[表 2]

$x_{1,1,1}$ の値	$x_{1,1,2}$ の値	該当するシフト
0	0	休憩
1	0	インバウンド業務
0	1	アウトバウンド業務
1	1	(該当なし) ※両方とも 1 になることは制約条件で禁止する

[0100] 「勤務種類」を考える場合、上記の表2において、勤務種類「インバウンド」を $k=1$ に対応させ、勤務種類「アウトバウンド」を $k=2$ に対応させ、2つの変数の組合せによってタイムテーブル上の1セルを表現する。つまり、あるセルに属する2つの変数 ($x_{(i,j,1)}$ と $x_{(i,j,2)}$) がともに0なら「休憩」、 $x_{(i,j,1)}$ が1で $x_{(i,j,2)}$ が0なら「インバウンド」、 $x_{(i,j,1)}$ が0で $x_{(i,j,2)}$ が1なら「アウトバウンド」を表すことになる。ここで、 $x_{(i,j,1)}$ も $x_{(i,j,2)}$ も1となった場合は「同一人物が同一時間帯に2つの異なる業務を実施することになってしまうため、これは制約条件として除外する必要がある。

[0101] <月間シフトにおけるスピ>

ここでは、ある人員 i が、ある日付 j に、勤務種類 k を実施する場合に1、実施しない場合に0となる変数 $x_{(i,j,k)}$ を考える。ただし、人員 i の勤務日数の下限を L_i 、上限を U_i 、日付 j に必要な人員の数を N_j としている。基本的にはこの変数 $x_{(i,j,k)}$ の1つ1つがCMOSアニーリングマシンにおけるスピ1つ1つに割り振られる(表3参照)。

[0102] [表3]

日付	人員 1 ($i = 1$)	人員 2 ($i = 2$)	人員 3 ($i = 3$)	...	人員 30 ($i = 30$)	必要人数
1 日 ($j = 1$)	$x_{1,1,k}$	$x_{2,1,k}$	$x_{3,1,k}$	...	$x_{30,1,k}$	N_1
2 日 ($j = 2$)	$x_{1,2,k}$	$x_{2,2,k}$	$x_{3,2,k}$	...	$x_{30,2,k}$	N_2
3 日 ($j = 3$)	$x_{1,3,k}$	$x_{2,3,k}$	$x_{3,3,k}$	...	$x_{30,3,k}$	N_3
4 日 ($j = 4$)	$x_{1,4,k}$	$x_{2,4,k}$	$x_{3,4,k}$	...	$x_{30,4,k}$	N_4
⋮	⋮	⋮	⋮	⋮	⋮	⋮
31 日 ($j = 31$)	$x_{1,31,k}$	$x_{2,31,k}$	$x_{3,31,k}$	...	$x_{30,31,k}$	N_{31}
勤務日数	$L_1 \sim U_1$	$L_2 \sim U_2$	$L_3 \sim U_3$	...	$L_{30} \sim U_{30}$	

「勤務種類」を考えない場合、この変数は単純に、「ある人員 i が、ある日付 j に業務をするかしないか」を与える変数となる。つまり、表3で1が入ったセルは出勤し、0が入ったセルは休日となる、ということになる。

[0103] 一方、「勤務種類」を考える場合、例えば、「早番」を $k=1$ 、「遅番」を $k=2$ 、「深夜番」を $k=3$ に対応させるようなことが考えられる。このとき、図7

における表の一番左上のセルに対応する変数は3つ存在する ($x_{1,1,1}$ 、 $x_{1,1,2}$ 、および $x_{1,1,3}$) ことになる。この3つの変数がそれぞれ0か1の値を取る
ので、考えられる組合せは、下記の表4における8つとなる。

[0104] [表4]

$x_{1,1,1}$ の値	$x_{1,1,2}$ の値	$x_{1,1,3}$ の値	該当するシフト
0	0	0	休日
1	0	0	早番
0	1	0	遅番
0	0	1	深夜番
1	1	0	× (制約条件で禁止する)
1	0	1	× (制約条件で禁止する)
0	1	1	× (制約条件で禁止する)
1	1	1	× (制約条件で禁止する)

このように、「3つの変数 ($x_{1,1,1}$ 、 $x_{1,1,2}$ 、および $x_{1,1,3}$) のうち、1
を取るのは1つまで」という制約条件を課して、上表の下4行分の結果が出て
しまうことを禁止する必要がある。

[0105] ———データ構造例———

続いて、本実施形態のスケジュール作成支援装置100が用いる各種情報
について説明する。図5に、本実施形態における基本情報テーブル125の
一例を示す。

[0106] 本実施形態の基本情報テーブル125は、コールセンタ業務で協働する各
オペレータの所定月における総勤務コマ数、当該月における各日の各時間帯
における必要人数、の各情報を格納したテーブルである。

[0107] この例では、時間間隔を10分として、横の行で指定している必要人数を
、各時間帯で満たし、オペレータごとの勤務時間数（勤務コマ数）を守る組
み合わせ最適化問題となる。区切る時間間隔が細かいほど、計算に必要なセ
ル（本テーブルにおけるセル＝要因）の数は増加し、より計算力が必要とな
る問題である。またセルの数が多いため、禁止パターンや最低連続コマ数（
連続勤務コマ数の最低基準）、上限連続コマ数（連続勤務コマ数の上限基準
）等の制約を入れる際、制約条件に応じた制約条件用関数の数が膨大となる
。ところがCMOSアニーリング法を採用すれば、現実的な時間内に好適な
結果すなわちスケジュールが特定可能である。

[0108] なお、図5の基本情報テーブル125で例示した例は、あくまで説明の都合上で限定的であり、その他の様々な事象の情報が格納されているものとする（以下同様）。

[0109] また、図6および図7に本実施形態における制約条件テーブル126の一例を示す。本実施形態の制約条件テーブル126は、上述のコールセンタ業務に各オペレータを割り当てる際の制約条件それぞれの情報を格納したテーブルである。

[0110] そのデータ構造は、各制約条件の識別情報（図中では“#”）をキーとして、当該制約条件の内容、最適化ソルバでの実装例、およびアニーリングマシンでの実装例（制約条件用関数）。といったデータから成るレコードの集合体である。

[0111] 本実施形態では、アニーリングマシンでの実装例すなわち制約条件用関数に加えて、従来技術との比較のため、最適化ソルバで当該制約条件を実装する際の関数例も対照表示している。

[0112] ーフロー例ー

以下、本実施形態におけるスケジュール作成支援方法の実手順について図に基づき説明する。以下で説明するスケジュール作成支援方法に対応する各種動作は、スケジュール作成支援装置100がメモリ等に読み出して実行するプログラムによって実現される。そして、このプログラムは、以下に説明される各種の動作を行うためのコードから構成されている。

[0113] 図8は、本実施形態におけるスケジュール作成支援方法のフロー例を示す図である。この場合、スケジュール作成支援装置100は、処理対象とするイジングモデル1021として、或る月における各オペレータの総勤務コマ数、必要人数、および制約条件用関数、を項として含む目的関数に関して、各オペレータの出勤有無をスピンとし、制約条件用関数における変数間の感応度をスピン間の相互作用の強度として設定したイジングモデル1021を演算するものとする。

[0114] そこで、スケジュール作成支援装置100は、各オペレータの希望条件を

ユーザ端末200から取得し、これを制約条件として制約条件テーブル126に格納する(s10)。或いは、この際、各オペレータから制約条件テーブル126中から、所望の制約条件に合致する制約条件の指定を受け付けるとしてもよい。

[0115] また、スケジュール作成支援装置100は、s10で得た制約条件を、既に述べたように制約条件用関数に変換する(s11)。この制約条件用関数は、当該制約条件が満たされる際に最小となる関数である。なお、上述のs10で制約条件の指定を受けた場合、当該制約条件に対して規定済みの制約条件用関数を制約条件テーブル126から抽出することとなる。

[0116] また、スケジュール作成支援装置100は、アニーリングマシンとして、上述の3つの項目（総勤務コマ数、必要人数、および制約条件用関数）に関する設定がなされたイジングモデル1021を課題とし、当該目的関数が最小となる基底状態を算定する(s12)。こうした基底状態の探索自体は、既存技術における処理と同様である。

[0117] つまり、各オペレータに関する制約条件と、上述の総勤務コマ数および必要人数といった基本情報の規定を満たしつつ、時間経過とともに（感応度に基づく理論上の）各時間帯の割り当て有無の最終状態に向け遷移し、各オペレータの割り当て結果が落ち着く状態を、基底状態として探索することとなる。

[0118] また、スケジュール作成支援装置100は、s12で特定した各時間帯における各オペレータの割り当て有無の情報（図9の画面900）を、スケジュールとしてユーザ端末200に送信し(s13)、処理を終了する。

[0119] 一方、こうした情報の提供を受けたユーザ端末200では、図9に示すようなスケジュール画面900をディスプレイ等の出力部に表示させる。

[0120] ユーザ端末200の操作者は、このスケジュール画面900を閲覧し、例えば、来月や明日、或いは直近1時間先のタイムテーブルについて認識し、コールセンタ業務の運営について適宜な確認、検討行うこととなる。

[0121] なお、制約条件の種類に応じて生成されるスケジュールの例は、以下のと

おりとなる。ここでは、こうした連続勤務に関する制約条件（式 15：勤務コマをなるべく連続させる）が無い前提でスケジュール生成した結果（表 5）と、当該制約条件を解いた場合のスケジュール（表 6）を対比のためそれぞれ示す。

[0122] [式 15]

$$\min - \sum_{i=1}^4 \sum_{j=2}^{15} x_{i,j-1} x_{i,j} \quad \text{式 15}$$

[表 5]

制約条件がない場合					
時間帯	オペレータ1	オペレータ2	オペレータ3	オペレータ4	必要人数
9:00 - 9:10	勤務	休憩	休憩	勤務	2
9:10 - 9:20	休憩	勤務	休憩	勤務	2
9:20 - 9:30	勤務	休憩	勤務	休憩	2
9:30 - 9:40	休憩	勤務	休憩	勤務	2
9:40 - 9:50	勤務	休憩	勤務	休憩	2
9:50 - 10:00	休憩	勤務	勤務	勤務	3
10:00 - 10:10	勤務	勤務	勤務	勤務	4
10:10 - 10:20	勤務	勤務	勤務	勤務	4
10:20 - 10:30	勤務	勤務	勤務	休憩	3
10:30 - 10:40	勤務	休憩	勤務	勤務	3
10:40 - 10:50	休憩	勤務	勤務	休憩	2
10:50 - 11:00	勤務	休憩	休憩	勤務	2
11:00 - 11:10	休憩	勤務	勤務	休憩	2
11:10 - 11:20	勤務	勤務	休憩	勤務	3
11:20 - 11:30	勤務	勤務	勤務	勤務	4
勤務コマ数	10	10	10	10	

[表 6]

制約条件がある場合					
時間帯	オペレータ1	オペレータ2	オペレータ3	オペレータ4	必要人数
9:00 - 9:10	勤務	休憩	休憩	勤務	2
9:10 - 9:20	勤務	休憩	休憩	勤務	2
9:20 - 9:30	勤務	休憩	勤務	休憩	2
9:30 - 9:40	休憩	勤務	勤務	休憩	2
9:40 - 9:50	休憩	勤務	勤務	休憩	2
9:50 - 10:00	休憩	勤務	勤務	勤務	3
10:00 - 10:10	勤務	勤務	勤務	勤務	4
10:10 - 10:20	勤務	勤務	勤務	勤務	4
10:20 - 10:30	勤務	勤務	勤務	休憩	3
10:30 - 10:40	勤務	勤務	勤務	休憩	3
10:40 - 10:50	休憩	勤務	休憩	勤務	2
10:50 - 11:00	休憩	勤務	休憩	勤務	2
11:00 - 11:10	勤務	休憩	休憩	勤務	2
11:10 - 11:20	勤務	休憩	勤務	勤務	3
11:20 - 11:30	勤務	勤務	勤務	勤務	4
勤務コマ数	10	10	10	10	

上記のように、制約がない場合は勤務コマがコマ切れになり、10分おき

に「勤務→休憩→勤務→休憩」となっている箇所が多く出現し、現実的なタイムテーブルとなっていない。

[0123] 一方、制約がある場合は、基本的に勤務コマが連続しており、まとまった時間、同じ勤務ができる現実的なタイムテーブルとなっている。

[0124] 続いて、不連続勤務に関する制約条件（式 16：休日コマを連続させすぎない）が無い前提でスケジュール生成した結果（表 7）と、当該制約条件を解いた場合のスケジュール（表 8）をそれぞれ示す。なお、ここでは、月間シフトにおける不連続勤務に関する制約条件の結果例を示す。今回考える制約条件の式の例を、以下に示す。ただし、今回は「勤務種類」については言及しないため、勤務種類に関する添え字（k）を省略した変数 $x_{i,j}$ について考える。また、Aは調整された任意の整数である。

[0125] [式 16]

$$\min - \sum_{i=1}^4 \sum_{j=2}^{14} (1 - x_{i,j-1})(1 - x_{i,j}) + A \sum_{i=1}^4 \sum_{j=3}^{14} \prod_{l=j-2}^j (1 - x_{i,l}) \quad \text{式 16}$$

[表 7]

制約条件がない場合					
日付	オペレータ1	オペレータ2	オペレータ3	オペレータ4	必要人数
1日	出勤	休日	出勤	出勤	3
2日	出勤	休日	出勤	出勤	3
3日	出勤	休日	出勤	出勤	3
4日	出勤	休日	出勤	休日	2
5日	出勤	出勤	出勤	休日	3
6日	出勤	出勤	出勤	休日	3
7日	出勤	出勤	出勤	休日	3
8日	出勤	出勤	休日	出勤	3
9日	出勤	出勤	休日	出勤	3
10日	出勤	出勤	休日	出勤	3
11日	休日	出勤	休日	出勤	2
12日	休日	出勤	出勤	出勤	3
13日	休日	出勤	出勤	出勤	3
14日	休日	出勤	出勤	出勤	3
勤務日数	10	10	10	10	

[表 8]

制約条件がある場合					
日付	オペレータ1	オペレータ2	オペレータ3	オペレータ4	必要人数
1日	休日	出勤	出勤	出勤	3
2日	休日	出勤	出勤	出勤	3
3日	出勤	出勤	休日	出勤	3
4日	出勤	出勤	休日	休日	2
5日	出勤	出勤	出勤	休日	3
6日	出勤	休日	出勤	出勤	3
7日	出勤	休日	出勤	出勤	3
8日	休日	出勤	出勤	出勤	3
9日	休日	出勤	出勤	出勤	3
10日	出勤	出勤	休日	出勤	3
11日	出勤	出勤	休日	休日	2
12日	出勤	休日	出勤	出勤	3
13日	出勤	休日	出勤	出勤	3
14日	出勤	出勤	出勤	休日	3
勤務日数	10	10	10	10	

上記のように、制約がない場合は出勤コマや休日が必要以上に連続となっ
てしまい、「10連勤」や「4連休」といった、法律面・就業規則面・各オ
ペレータの負担面から見ても、非現実的な月間シフトとなっている。一方、
制約がある場合は、程よく連勤と連休が散らばっており、現実的な月間シフ
トとなっている。

[0126] 続いて、禁止パターンに関する制約条件（式17：遅番翌日の早番禁止）
が無い前提でスケジュール生成した結果（表9）と、当該制約条件を解いた
場合のスケジュール（表10）をそれぞれ示す。ここでは、月間シフトにお
ける禁止パターンに関する制約条件の結果例を示す。今回考える制約条件の
式の例を、以下に示す。ただし、今回は「勤務種類」のうち、早番（k=1）と
遅番（k=2）についてのみ言及するため、勤務種類に関する添え字（k）は2
までを考える。

[0127] [式17]

$$\min \sum_{i=1}^4 \sum_{j=2}^{14} x_{i,j-1,2} x_{i,j,1} \quad \text{式 17}$$

この制約条件は「遅番の翌日に早番が入らないようにする」という意味と
なる。

[0128] [表9]

制約条件がない場合

日付	オペレータ1	オペレータ2	オペレータ3	オペレータ4	必要人数	
					早番	遅番
1日	休日	早番	早番	▲遅番	2	1
2日	休日	早番	▲遅番	早番	2	1
3日	▲遅番	早番	休日	早番	2	1
4日	早番	▲遅番	休日	休日	1	1
5日	早番	早番	▲遅番	休日	2	1
6日	▲遅番	休日	早番	早番	2	1
7日	早番	休日	早番	▲遅番	2	1
8日	休日	早番	▲遅番	早番	2	1
9日	休日	▲遅番	早番	早番	2	1
10日	▲遅番	早番	休日	早番	2	1
11日	早番	▲遅番	休日	休日	1	1
12日	早番	休日	早番	▲遅番	2	1
13日	早番	休日	早番	▲遅番	2	1
14日	早番	早番	▲遅番	休日	2	1
早番日数	7	7	6	6		
遅番日数	3	3	4	4		

[表 10]

制約条件がある場合

日付	オペレータ1	オペレータ2	オペレータ3	オペレータ4	必要人数	
					早番	遅番
1日	休日	早番	▲遅番	早番	2	1
2日	休日	早番	▲遅番	早番	2	1
3日	早番	早番	休日	▲遅番	2	1
4日	早番	▲遅番	休日	休日	1	1
5日	早番	▲遅番	早番	休日	2	1
6日	▲遅番	休日	早番	早番	2	1
7日	▲遅番	休日	早番	早番	2	1
8日	休日	早番	▲遅番	早番	2	1
9日	休日	早番	▲遅番	早番	2	1
10日	早番	早番	休日	▲遅番	2	1
11日	早番	▲遅番	休日	休日	1	1
12日	早番	休日	早番	▲遅番	2	1
13日	早番	休日	早番	▲遅番	2	1
14日	▲遅番	早番	早番	休日	2	1
早番日数	7	7	6	6		
遅番日数	3	3	4	4		

なお、上述の各表において、視認性向上のため、遅番の前に▲マークを付している。上記のように、制約がない場合は遅番の翌日に早番となってしまう箇所が存在し、十分な休息が取られないまま翌日の勤務を実施しなければならない、非現実的な月間シフトとなっている。一方、制約がある場合は、遅番の翌日は遅番か休日かのどちらかとなっており、現実的な月間シフトとなっている。

[0129] 続いて、人員間相性に関する制約条件（式 18：新人の同時出勤を避ける&指導員をセットにする）が無い前提でスケジュール生成した結果（表 11）と、当該制約条件を解いた場合のスケジュール（表 12）をそれぞれ示す。ここでは、月間シフトにおける人員間相性に関する制約条件の結果例を示す

。

[0130] ただし、今回は「勤務種類」については言及しないため、勤務種類に関する添え字 (k) を省略した変数 $x_{(i,j)}$ について考える。また、A、B は調整された任意の整数である。

[0131] [式 18]

$$\min \sum_{j=1}^{14} x_{3,j}x_{4,j} - A \sum_{j=1}^{14} x_{1,j}x_{3,j} - B \sum_{j=1}^{14} x_{2,j}x_{4,j} \quad \text{式 18}$$

[表 11]

制約条件がない場合					
日付	オペレータ1 (ベテランA) ※新人Aの指導員	オペレータ2 (ベテランB) ※新人Bの指導員	オペレータ3 (新人A)	オペレータ4 (新人B)	必要人数
1日	出勤	休日	出勤	出勤	3
2日	出勤	休日	出勤	出勤	3
3日	出勤	出勤	休日	休日	2
4日	休日	出勤	休日	出勤	2
5日	休日	出勤	出勤	出勤	3
6日	出勤	出勤	休日	休日	2
7日	出勤	出勤	休日	休日	2
8日	出勤	休日	出勤	出勤	3
9日	出勤	休日	出勤	出勤	3
10日	出勤	出勤	休日	休日	2
11日	休日	出勤	出勤	休日	2
12日	出勤	出勤	休日	出勤	3
13日	出勤	出勤	休日	休日	2
14日	休日	出勤	出勤	休日	2
勤務日数	10	10	7	7	

[表 12]

制約条件がある場合					
日付	オペレータ1 (ベテランA) ※新人Aの指導員	オペレータ2 (ベテランB) ※新人Bの指導員	オペレータ3 (新人A)	オペレータ4 (新人B)	必要人数
1日	出勤	出勤	休日	出勤	3
2日	出勤	出勤	出勤	休日	3
3日	出勤	休日	出勤	休日	2
4日	休日	出勤	休日	出勤	2
5日	出勤	出勤	休日	出勤	3
6日	休日	出勤	休日	出勤	2
7日	出勤	休日	出勤	休日	2
8日	出勤	出勤	出勤	休日	3
9日	出勤	出勤	出勤	休日	3
10日	休日	出勤	休日	出勤	2
11日	休日	出勤	休日	出勤	2
12日	出勤	出勤	休日	出勤	3
13日	出勤	休日	出勤	休日	2
14日	出勤	休日	出勤	休日	2
勤務日数	10	10	7	7	

上記のように、制約がない場合は「新人2人が同時に出勤する日」「新人Aが出勤しているのに、新人Aの指導員であるオペレータ1が出勤していない日」「新人Bが出勤しているのに、新人Bの指導員であるオペレータ2が出勤していない日」が存在し、現実的な月間シフトとなっていない。一方、

制約がある場合は、すべての日で新人は1人のみの出勤となっており、新人が出勤する際はその指導員が必ず出勤するという、現実的な月間シフトとなっている。

[0132] 続いて、人員希望反映に関する制約条件（式19：オペレータ2は1～7日に休日希望&オペレータ3は早番希望&オペレータ4は遅番希望）が無い前提下でスケジュール生成した結果（表13）と、当該制約条件を解いた場合のスケジュール（表14）をそれぞれ示す。

[0133] [式19]

$$\min - \sum_{j=1}^7 (1 - x_{2,j,k}) - A \sum_{j=1}^{14} x_{3,j,1} - B \sum_{j=1}^{14} x_{4,j,2} \quad \text{式 19}$$

この制約条件は、「オペレータ2は期間の前半（1日～7日）に休日を希望する」および「オペレータ3はできるだけ早番を希望する」および「オペレータ4はできるだけ遅番を希望する」という意味となる。

[0134] [表13]

制約条件がない場合						
日付	オペレータ1 (希望なし)	オペレータ2 (前半に休日希望)	オペレータ3 (早番希望)	オペレータ4 (遅番希望)	必要人数	
					早番	遅番
1日	休日	早番	▲遅番	早番	2	1
2日	休日	早番	▲遅番	早番	2	1
3日	早番	早番	休日	▲遅番	2	1
4日	早番	▲遅番	休日	休日	1	1
5日	早番	▲遅番	早番	休日	2	1
6日	▲遅番	休日	早番	早番	2	1
7日	▲遅番	休日	早番	早番	2	1
8日	休日	早番	▲遅番	早番	2	1
9日	休日	早番	▲遅番	早番	2	1
10日	早番	早番	休日	▲遅番	2	1
11日	早番	▲遅番	休日	休日	1	1
12日	早番	休日	早番	▲遅番	2	1
13日	早番	休日	早番	▲遅番	2	1
14日	▲遅番	早番	早番	休日	2	1
勤務日数	10	10	10	10		

[表14]

制約条件がある場合						
日付	オペレータ1 (希望なし)	オペレータ2 (前半に休日希望)	オペレータ3 (早番希望)	オペレータ4 (遅番希望)	必要人数	
					早番	遅番
1日	早番	休日	早番	▲遅番	2	1
2日	早番	休日	早番	▲遅番	2	1
3日	休日	早番	早番	▲遅番	2	1
4日	早番	▲遅番	休日	休日	1	1
5日	早番	休日	早番	▲遅番	2	1
6日	早番	休日	早番	▲遅番	2	1
7日	▲遅番	早番	早番	休日	2	1
8日	▲遅番	早番	早番	休日	2	1
9日	休日	早番	早番	▲遅番	2	1
10日	早番	早番	休日	▲遅番	2	1
11日	▲遅番	早番	休日	休日	1	1
12日	休日	早番	早番	▲遅番	2	1
13日	休日	早番	早番	▲遅番	2	1
14日	早番	早番	休日	▲遅番	2	1
勤務日数	10	10	10	10		

なお、上述の各表において、視認性向上のため、遅番の前に▲マークを付している。

[0135] 上記のように、制約がない場合は「期間の後半にもオペレータ2に休日」「オペレータ3に遅番」「オペレータ4に早番」といった、オペレータの希望から外れた月間シフトとなっている。一方、制約がある場合は、各オペレータの希望が叶えられた月間シフトとなっている。

[0136] 続いて、人員間平等勤務に関する制約条件（式20：勤務時間を人員間で平等とする）が無い前提でスケジュール生成した結果（表15）と、当該制約条件を解いた場合のスケジュール（表16）をそれぞれ示す。

[0137] [式20]

$$\min - \sum_{i=1}^4 \left[\sum_{j=1}^{15} x_{i,j} - D \right]^2 \quad \text{式 20}$$

この制約条件は、「各オペレータの勤務コマ数を、目安であるDになるべく近づける」という意味となる。

[0138] [表15]

制約条件がない場合					
時間帯	オペレータ1	オペレータ2	オペレータ3	オペレータ4	必要人数
9:00 - 9:10	勤務	勤務	休憩	休憩	2
9:10 - 9:20	勤務	勤務	休憩	休憩	2
9:20 - 9:30	勤務	勤務	休憩	休憩	2
9:30 - 9:40	勤務	勤務	休憩	休憩	2
9:40 - 9:50	勤務	勤務	勤務	休憩	3
9:50 - 10:00	勤務	勤務	勤務	休憩	3
10:00 - 10:10	勤務	勤務	勤務	勤務	4
10:10 - 10:20	勤務	勤務	勤務	勤務	4
10:20 - 10:30	勤務	勤務	勤務	勤務	4
10:30 - 10:40	勤務	勤務	勤務	休憩	3
10:40 - 10:50	勤務	勤務	勤務	休憩	3
10:50 - 11:00	勤務	勤務	休憩	休憩	2
11:00 - 11:10	勤務	勤務	休憩	休憩	2
11:10 - 11:20	勤務	勤務	勤務	休憩	3
11:20 - 11:30	勤務	勤務	勤務	休憩	3
勤務コマ数の実績	15	15	9	3	
勤務コマ数の目安	10	10	10	10	

[表 1 6]

制約条件がある場合					
時間帯	オペレータ1	オペレータ2	オペレータ3	オペレータ4	必要人数
9:00 - 9:10	勤務	休憩	休憩	勤務	2
9:10 - 9:20	勤務	休憩	休憩	勤務	2
9:20 - 9:30	勤務	休憩	勤務	休憩	2
9:30 - 9:40	休憩	勤務	勤務	休憩	2
9:40 - 9:50	休憩	勤務	勤務	勤務	3
9:50 - 10:00	休憩	勤務	勤務	勤務	3
10:00 - 10:10	勤務	勤務	勤務	勤務	4
10:10 - 10:20	勤務	勤務	勤務	勤務	4
10:20 - 10:30	勤務	勤務	勤務	勤務	4
10:30 - 10:40	勤務	勤務	勤務	休憩	3
10:40 - 10:50	休憩	勤務	勤務	勤務	3
10:50 - 11:00	休憩	勤務	休憩	勤務	2
11:00 - 11:10	勤務	休憩	休憩	勤務	2
11:10 - 11:20	勤務	休憩	勤務	勤務	3
11:20 - 11:30	勤務	勤務	勤務	休憩	3
勤務コマ数の実績	10	10	11	11	
勤務コマ数の目安	10	10	10	10	

上記のように、制約がない場合は人員間で勤務時間に偏りがあり、休憩時間がない人員と休憩時間ばかりの人員が混在している、非現実的なタイムテーブルとなっている。一方、制約がある場合は、各人員間でほぼ同じ勤務時間となっており、平等で現実的なタイムテーブルとなっている。

[0139] なお、例えば、突発的な需要急増に伴うオペレータ不足、或いはオペレータの突然の欠勤、といった緊急時に際し、スケジュール作成支援装置 100 は、当該事象に対応する制約条件をユーザ端末 200 から取得し、上述のフローを実行するとすれば好適である。

[0140] こうした状況での制約条件は、例えば、直近までに作成していたスケジュールでは、同日同時間帯に別の業務に割り当てられているオペレータを、「

出勤」の前提とする制約条件等が想定できる。

- [0141] その場合のスケジュール作成支援装置 100 は、当該制約条件に応じて s11 を実行して制約条件用関数を生成し、対応する目的関数に関するイジングモデルを解くものとする。
- [0142] 以上、本発明を実施するための最良の形態などについて具体的に説明したが、本発明はこれに限定されるものではなく、その要旨を逸脱しない範囲で種々変更可能である。
- [0143] こうした本実施形態によれば、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成を効率的に行うことが可能となる。
- [0144] 本明細書の記載により、少なくとも次のことが明らかにされる。すなわち、本実施形態のスケジュール作成支援装置において、前記演算部は、連続勤務に関する前記制約条件である、連続勤務長の上限または下限、連続勤務の追求、および連続勤務の抑制、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、としてもよい。
- [0145] これによれば、人員各々における連続勤務に関する条件（例：本人の希望や所属組織の規定。以下同様）を可能なかぎり踏まえつつ、人員コスト最小となるスケジュールを効率良く作成可能となる。ひいては、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成をより効率的に行うことが可能となる。
- [0146] また、本実施形態のスケジュール作成支援装置において、前記演算部は、不連続勤務に関する前記制約条件である、勤務タイミグの間隔の上限または下限、休日連続化の追求、および休日連続化の抑制、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、としてもよい。
- [0147] これによれば、人員各々における不連続勤務に関する条件を可能なかぎり踏まえつつ、人員コスト最小となるスケジュールを効率良く作成可能となる。ひいては、協働する多くの人員に関して非線形な制約条件を踏まえたスケ

ジュール作成をより効率的に行うことが可能となる。

[0148] また、本実施形態のスケジュール作成支援装置において、前記演算部は、禁止パターンに関する前記制約条件である、特定時間帯での勤務の連続禁止、および前記特定時間帯での勤務直後の所定時間帯での勤務の禁止、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、としてもよい。

[0149] これによれば、人員各々における勤務に関する禁止パターン（例：深夜勤務の直後に日勤は禁止）を可能なかぎり踏まえつつ、人員コスト最小となるスケジュールを効率良く作成可能となる。ひいては、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成をより効率的に行うことが可能となる。

[0150] また、本実施形態のスケジュール作成支援装置において、前記演算部は、人員間の相性に関する前記制約条件である、所定属性の人員の同時勤務の追求または抑制の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、としてもよい。

[0151] これによれば、人員各々における他人員との相性に関する条件（例：新人に対して指導可能な人員が同時出勤する。或る人員と或る人員とは同時出勤しない。）を可能なかぎり踏まえつつ、人員コスト最小となるスケジュールを効率良く作成可能となる。ひいては、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成をより効率的に行うことが可能となる。

[0152] また、本実施形態のスケジュール作成支援装置において、前記演算部は、人員の希望に関する前記制約条件である、所定パターンでの勤務の追求について、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、としてもよい。

[0153] これによれば、人員各々における希望に関する条件（例：所定日にできれば特定のシフトを割り当てる）を可能なかぎり踏まえつつ、人員コスト最小

となるスケジュールを効率良く作成可能となる。ひいては、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成をより効率的に行うことが可能となる。

[0154] また、本実施形態のスケジュール作成支援装置において、前記演算部は、人員間の平等に関する前記制約条件である、勤務パターンごとの出勤長に関する前記各人員の間での均等化について、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、としてもよい。

[0155] これによれば、人員間の平等に関する条件を可能なかぎり踏まえつつ、人員コスト最小となるスケジュールを効率良く作成可能となる。ひいては、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成をより効率的に行うことが可能となる。

[0156] また、本実施形態のスケジュール作成支援装置は、前記イジングモデルに関して組合せ最適化問題を解くCMOSアニーリングマシンであるとしてもよい。

[0157] これによれば、イジングモデルの動作を半導体のCMOS (Complementary Metal Oxide Semiconductor) などの素子を用いた回路で擬似的に再現し、互いに影響しあっている制約条件すなわち非線形な制約条件を踏まえた組合せ最適化問題の実用解を、室温下で効率良く求めることが可能となる。ひいては、協働する多くの人員に関して非線形な制約条件を踏まえたスケジュール作成を、さらに効率的に行うことが可能となる。

[0158] また、本実施形態のスケジュール作成支援方法において、前記情報処理装置が、連続勤務に関する前記制約条件である、連続勤務長の上限または下限、連続勤務の追求、および連続勤務の抑制、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、としてもよい。

[0159] また、本実施形態のスケジュール作成支援方法において、前記情報処理装置が、不連続勤務に関する前記制約条件である、勤務タイミングの間隔の上

限または下限、休日連続化の追求、および休日連続化の抑制、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、としてもよい。

[0160] また、本実施形態のスケジュール作成支援方法において、前記情報処理装置が、禁止パターンに関する前記制約条件である、特定時間帯での勤務の連続禁止、および前記特定時間帯での勤務直後の所定時間帯での勤務の禁止、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、としてもよい。

[0161] また、本実施形態のスケジュール作成支援方法において、前記情報処理装置が、人員間の相性に関する前記制約条件である、所定属性の人員の同時勤務の追求または抑制の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、としてもよい。

[0162] また、本実施形態のスケジュール作成支援方法において、前記情報処理装置が、人員の希望に関する前記制約条件である、所定パターンでの勤務の追求について、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するとしてもよい。

[0163] また、本実施形態のスケジュール作成支援方法において、前記情報処理装置が、人員間の平等に関する前記制約条件である、勤務パターンごとの出勤長に関する前記各人員の間での均等化について、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、としてもよい。

[0164] また、本実施形態のスケジュール作成支援方法において、前記情報処理装置が、前記イジングモデルに関して組合せ最適化問題を解くCMOSアニーリングマシンであるとしてもよい。

符号の説明

- [0165] 10 ネットワーク
100 スケジュール作成支援装置（アニーリングマシン）
101 記憶部
102 プログラム

- 1 0 2 1 イジングモデル
- 1 0 3 メモリ
- 1 0 4 演算部
- 1 0 5 通信部
- 1 2 5 基本情報テーブル
- 1 2 6 制約条件テーブル
- 2 0 0 ユーザ端末

請求の範囲

- [請求項1] 所定業務で協働する各人員の所定期間における総勤務長、前記期間における各タイミングでの必要人数、および前記業務に前記各人員を割り当てる際の制約条件、の各情報を格納した記憶部と、
- 前記期間における前記総勤務長、前記必要人数、および前記制約条件が満たされる際に最小となる制約条件用関数、を項として含む目的関数に関して、前記各人員の出勤有無をスピンとし、前記制約条件用関数における変数間の感応度を前記スピンの間の相互作用の強度として設定したイジングモデルを演算する演算部とを有し、
- 前記演算部は前記演算の結果に基づき、前記所定期間の前記各タイミングにおける前記各人員の出勤有無を規定したスケジュールを所定装置に出力することを特徴とするスケジュール作成支援装置。
- [請求項2] 前記演算部は、
- 連続勤務に関する前記制約条件である、連続勤務長の上限または下限、連続勤務の追求、および連続勤務の抑制、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、
- ことを特徴とする請求項1に記載のスケジュール作成支援装置。
- [請求項3] 前記演算部は、
- 不連続勤務に関する前記制約条件である、勤務タイミングの間隔の上限または下限、休日連続化の追求、および休日連続化の抑制、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、
- ことを特徴とする請求項1に記載のスケジュール作成支援装置。
- [請求項4] 前記演算部は、
- 禁止パターンに関する前記制約条件である、特定時間帯での勤務の連続禁止、および前記特定時間帯での勤務直後の所定時間帯での勤務の禁止、の少なくともいずれかについて、前記制約条件用関数を前記

目的関数の項に含め、前記イジングモデルを演算するものである、
ことを特徴とする請求項 1 に記載のスケジュール作成支援装置。

[請求項5]

前記演算部は、
人員間の相性に関する前記制約条件である、所定属性の人員の同時勤務の追求または抑制の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、
ことを特徴とする請求項 1 に記載のスケジュール作成支援装置。

[請求項6]

前記演算部は、
人員の希望に関する前記制約条件である、所定パターンでの勤務の追求について、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、
ことを特徴とする請求項 1 に記載のスケジュール作成支援装置。

[請求項7]

前記演算部は、
人員間の平等に関する前記制約条件である、勤務パターンごとの出勤長に関する前記各人員の間での均等化について、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算するものである、
ことを特徴とする請求項 1 に記載のスケジュール作成支援装置。

[請求項8]

前記イジングモデルに関して組合せ最適化問題を解く CMOS アニマリングマシンであることを特徴とする請求項 1 に記載のスケジュール作成支援装置。

[請求項9]

所定業務で協働する各人員の所定期間における総勤務長、前記期間における各タイミングでの必要人数、および前記業務に前記各人員を割り当てる際の制約条件、の各情報を格納した記憶部を備える情報処理装置が、
前記期間における前記総勤務長、前記必要人数、および前記制約条件が満たされる際に最小となる制約条件用関数、を項として含む目的

関数に関して、前記各人員の出勤有無をスピンとし、前記制約条件用関数における変数間の感応度を前記スピンの間の相互作用の強度として設定したイジングモデルを演算し、前記演算の結果に基づき、前記所定期間の前記各タイミングにおける前記各人員の出勤有無を規定したスケジュールを所定装置に出力する、

ことを特徴とするスケジュール作成支援方法。

[請求項10]

前記情報処理装置が、

連続勤務に関する前記制約条件である、連続勤務長の上限または下限、連続勤務の追求、および連続勤務の抑制、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、

ことを特徴とする請求項9に記載のスケジュール作成支援方法。

[請求項11]

前記情報処理装置が、

不連続勤務に関する前記制約条件である、勤務タイミングの間隔の上限または下限、休日連続化の追求、および休日連続化の抑制、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、

ことを特徴とする請求項9に記載のスケジュール作成支援方法。

[請求項12]

前記情報処理装置が、

禁止パターンに関する前記制約条件である、特定時間帯での勤務の連続禁止、および前記特定時間帯での勤務直後の所定時間帯での勤務の禁止、の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、

ことを特徴とする請求項9に記載のスケジュール作成支援方法。

[請求項13]

前記情報処理装置が、

人員間の相性に関する前記制約条件である、所定属性の人員の同時勤務の追求または抑制の少なくともいずれかについて、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、

- ことを特徴とする請求項 9 に記載のスケジュール作成支援方法。
- [請求項14] 前記情報処理装置が、
- 人員の希望に関する前記制約条件である、所定パターンでの勤務の追求について、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、
- ことを特徴とする請求項 9 に記載のスケジュール作成支援方法。
- [請求項15] 前記情報処理装置が、
- 人員間の平等に関する前記制約条件である、勤務パターンごとの出勤長に関する前記各人員の間での均等化について、前記制約条件用関数を前記目的関数の項に含め、前記イジングモデルを演算する、
- ことを特徴とする請求項 9 に記載のスケジュール作成支援方法。
- [請求項16] 前記情報処理装置が、
- 前記イジングモデルに関して組合せ最適化問題を解く CMOS アニールマシンであることを特徴とする請求項 9 に記載のスケジュール作成支援方法。