



Europäisches Patentamt
European Patent Office
Office européen des brevets



(11) **EP 0 947 962 A2**

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
06.10.1999 Bulletin 1999/40

(51) Int. Cl.⁶: **G07B 17/02**

(21) Application number: **99105376.0**

(22) Date of filing: **16.03.1999**

(84) Designated Contracting States:
**AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE**
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **17.03.1998 IT TO980233**

(71) Applicant: **ELSAG SPA**
16154 Genova (IT)

(72) Inventor: **De Leo, Guido**
16134 Genova (IT)

(74) Representative:
Jorio, Paolo et al
STUDIO TORTA S.r.l.,
Via Viotti, 9
10121 Torino (IT)

(54) **A method of balancing the load in a process for sorting objects**

(57) The sorting process comprises performing a first and at least a second consecutive sorting cycle by means of a postal machine and the present invention relates to a method of balancing to be performed before the commencement of the sorting process and comprising the step of performing, in at least one of the sorting cycles, at least one of the following procedures:

a) a first procedure of balancing the load of the outputs of the postal machine in the current sorting cycle based on at least one of the following balancing criteria:

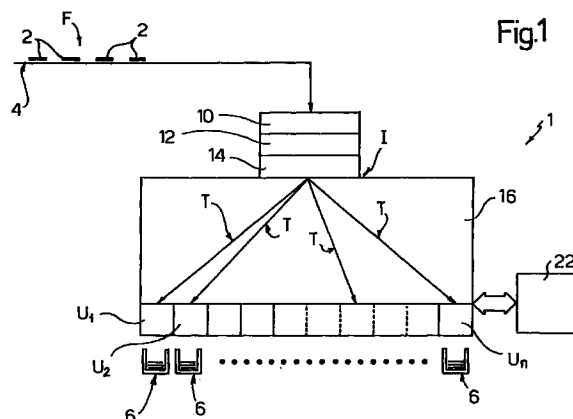
shifting the delivery addresses from the outputs to which they are assigned to respective logically contiguous outputs associated with the same postman;
bisection of delivery address into respective pairs of virtually separate delivery address and allocating them to a respective pair of logically contiguous outputs associated with the same postman; and
allocation of further outputs to postmen on the basis of an order of criticality of the outputs already allocated to the postman themselves; and

b) a second procedure of balancing the load of the outputs of the postal machine in the current sorting cycle based on at least one of the following balancing criteria:

distribution over outputs utilised in the current sorting cycle of delivery addresses allocated to

each output at the end of the subsequent sorting cycle;

shifting the delivery addresses from the outputs to which they are allocated to respective logically contiguous outputs;
exchange of delivery addresses between logically contiguous outputs; and
bisection of delivery addresses into respective pairs of virtually separate delivery addresses and allocating them to a respective pair of logically contiguous outputs.



EP 0 947 962 A2

Description

[0001] The present invention relates to a method of balancing the load in a process for sorting objects.

[0002] In particular, the present invention advantageously, but not exclusively, finds application in balancing the load in a process for sorting postal objects and the following discussion will make explicit reference thereto without any loss of generality thereby.

[0003] In fact, the method according to the present invention can also be utilised of balancing the load in any sorting process performed by one or more devices able to separate streams of objects to be arranged in a predetermined sequence for directing them to addresses or destinations, where the capacity of the said device or said devices is limited or expensive.

[0004] Machines are known for sorting and ordering post which receive at their input a stream of postal objects disposed in an entirely random manner and generate at their output an ordered stream of postal objects, that is to say a stream of postal objects disposed in a predetermined progressive order which allows their sequential distribution by one or more postman moving along a predetermined route.

[0005] The progressive order in which the postal objects are disposed at the outputs from the postal machine is generally defined by a sequence of adjacent destinations or addresses corresponding to street numbers or to groups of street numbers of buildings disposed along the route where these postal objects are to be delivered.

[0006] Known postal machines generally comprise an input, also called induction, receiving a set of postal objects, a plurality of outputs associated with respective containers into which groups of postal objects are disposed, and a transport and direction device (sorter) interposed between the inputs and the outputs and controlled by an electronic processing unit having the function of directing each postal object towards a respective output on the basis of a code generally printed on the postal object itself.

[0007] Moreover, each postman providing the service of postal object delivery is assigned a respective group of outputs of the postal machine different from that assigned to any other postman and from which at the end of the ordering process they extract the postal objects entrusted to them for delivery.

[0008] The ordering process performed by the postal machine generally involves the performance of a plurality of temporally consecutive sorting cycles by means of which groups of postal objects already subjected to preliminary sorting are reintroduced into the input of the postal machine and directed towards outputs associated with containers from which the postal objects deposited in a preceding sorting cycle have been removed.

[0009] When these sorting cycles have been completed groups of postal objects are taken from the

machine disposed in a predetermined progressive order which allows their sequential distribution by a postman who travels along a sub-section of a predefined route.

[0010] Postal machines also exist which have two or more inputs communicating with a single sorter device sending the postal objects towards groups of outputs associated with respective inputs.

[0011] Each input of the postal machine cannot utilise outputs in common with the other inputs and in this way, although the sorter device is shared, such postal machines in reality comprise several separate independent machines.

[0012] Common disadvantages of known sorting processes reside in the fact that such processes involve each output of the postal machine being associated with a predetermined number of adjacent addresses of places which the postman assigned to this output will have to serve and do not take account of the quantity of postal objects which must be delivered to the delivery addresses.

[0013] Consequently, during a sorting process it can happen that one output from the postal machine becomes filled, for example because there is a large user in one of the delivery addresses associated with this output receiving a large quantity of postal objects.

[0014] In cases of this type interruption of the current sorting cycle is therefore inevitable for the purpose of extracting the postal objects collected at the filled output.

[0015] Therefore, the filling of one output of the postal machine is enormously undesirable in that it involves significant losses of time both due to the interruption of the sorting cycle and to the numerous precautions which it is necessary to adopt in the management of the postal objects before restarting the sorting process itself.

[0016] This situation becomes particularly serious when the filling of one output of the postal machine is repeated several times during the various sorting cycles performed during one working shift.

[0017] The object of the present invention is that of providing a method of balancing the load in a process of sorting postal objects which makes it possible to avoid the most likely situations resulting in filling of the outputs of the postal machine which performs this process.

[0018] According to the present invention there is provided a method of balancing the load in a process for sorting objects, in particular postal objects, the said sorting process comprising a first and at least one second consecutive sorting cycle performed by means of a machine receiving a set of objects at its input and providing at its output the said objects identified and separated according to determined sorting rules; in one given sorting cycle the objects being provided to the outputs of the machine according to a respective predetermined sorting criterion and being again provided in an orderly manner to the input of the machine itself for the performance of a subsequent sorting cycle; the out-

puts of the machine being sub-divided, in the last sorting cycle of the sorting process, into separate subsets each subset being assigned to a plurality of respective destinations according to an assignment criterion correlated to the order that the objects are to be delivered to the destinations;

the said balancing method being characterised by the fact that it comprises the phase of performing, in at least one of the said sorting cycles, at least one of the following procedures:

a) a first procedure of balancing the load at the outputs from the machine at the end of the current sorting cycle based on at least one of the following balancing criteria:

- displacement of the destinations from the outputs to which they are assigned to respective logically contiguous outputs belonging to the same subset, according to a first predetermined logic criterion;
- assignment of further outputs to the said subsets on the basis of an order of criticality of the outputs already assigned to the subsets themselves; and
- bisection of destinations into respective pairs of virtually separate destinations and assignment of each pair of virtually separate destinations to a respective pair of logically contiguous outputs belonging to the same subset, according to a second predetermined logical criterion; and

b) a second procedure of balancing the load of the output of the machine at the end of the current sorting cycle based on at least one of the following balancing criteria:

- distribution over the outputs utilised in the current sorting cycle of destinations assigned to each output at the end of the successive sorting cycle;
- displacement of destinations from the outputs to which they are assigned to respective logically contiguous outputs according to a third predetermined logical criterion;
- exchange of destinations among logically contiguous outputs according to a fourth predetermined logical criterion; and
- bisection of destinations into respective pairs of virtually separate destinations and assignment of each pair of virtually separate destinations to a respective pair of logically contiguous outputs according to a fifth predetermined logical criterion.

rate destinations to a respective pair of logically contiguous outputs according to a fifth predetermined logical criterion.

[0019] For a better understanding of the present invention a preferred embodiment will now be described purely by way of non-limitative example and with reference to the attached drawings, in which:

Figure 1 is a schematic illustration of a machine for sorting postal objects; Figure 2 is a matrix constituting a matrix representation of the load of postal objects at each output of a postal machine performing a sorting process composed of two sorting cycles; and

Figures 3a-3k illustrate a flow diagram relating to the balancing method which forms the subject of the present invention.

[0020] In Figure 1 a postal machine is generally indicated 1, having an input I which can receive a stream F of postal objects 2 (for example letters, postcards, documents in envelopes or generally flat documents of approximately rectangular form) disposed in sequence (for example stacked) and conveyed towards the input I itself by means of a conveyor device 4 of known type (for example of belt type); and a plurality (N) of separate outputs U1, U2, U3, ..., UN, with each of which there is conveniently associated a removable container 6 (shown schematically) into which the postal objects 2 sent to this output are collected.

[0021] The stream F of postal objects 2 comprise a plurality of postal objects on which a code has been printed in a preceding phase, for example a bar code, which can identify the delivery address or destination of the postal object 2 itself; these postal objects 2 are in any event disposed in a sequence of "disordered" type, that is to say there is no progressive order nor any relation between the disposition of the postal objects 2 and the progressive order in which they are subsequently to be delivered.

[0022] The input I of the postal machine 1 is associated with a separator device 10 (schematically represented) receiving at its input the postal objects 2 supplied from the conveyor device 4 and acting to cause extraction of the postal objects 2 from the stream F and disposition of each postal object 2 into a spaced position with respect to the other postal object 2 of the stream F; a reading device 12 (schematically shown) receiving the postal objects 2 coming from the separator device 10 at its input and acting to read the code associated with the postal objects 2 themselves; a delay module 14 (shown schematically) receiving at its input the postal objects 2 coming from the reading device 12; and a sorter device 16 within the postal machine 1 interposed between the output of the delay module 14 and the outputs U1, U2, U3, ..., UN.

[0023] The postal machine is controlled by an elec-

tronic unit 22 of programmable type under the control of which the sorter device 16 directs the stream F supplied at the input I to all the N outputs of the postal machine 1, that is to say it operates a common mode of transport according to which each postal object 2 supplied to the input A can potentially be directed towards any of the N outputs.

[0024] The displacement of a postal object along the sorter 16, that is to say the path T followed by a postal object 2 along the sorter 16 from the input I to a generic output U_i , is determined by the code present on the postal object 2 read by the reading device.

[0025] To this end, as will be described in more detail hereinafter, the electronic unit 22 is provided with a plurality of electronic tables (not illustrated) receiving input data (for example from the reading devices 12) associated with the code printed on each postal object 2, and supplying at its output a set of data which identifies the selected output U_i towards which this postal object 2 must be directed.

[0026] The output data are then transmitted to the postal machine 1 to generate command signals for actuator members, for example, paddle selectors, transmission members etc (not shown) which contribute to provide, along the conveyor device 16, the path T which leads the postal object 2 towards the selected output U_i .

[0027] The method of balancing the load according to the present invention will hereinafter be described with reference to a postal machine performing a sorting process composed of two successive sorting cycles.

[0028] In particular, in the first sorting cycle the postal objects are provided at the input I of the postal machine 1 and then sorted towards the outputs U of the postal machine 1 according to a first sorting criterion determined by means of the present balancing method. The postal objects are then extracted in order from the outputs and reintroduced into the postal machine 1 itself through the input I in a predetermined reintroduction order for the second sorting cycle to be performed.

[0029] In the second sorting cycle the postal objects are sorted towards the outputs U according to a second sorting criterion determined by means of the present balancing method and then withdrawn from the outputs U to be delivered to the respective delivery addresses by means of the postmen.

[0030] For descriptive convenience the following description will make reference to a postal machine 1 having ten outputs without loss of generality thereby in that the inventive principle on which the balancing method which constitutes the subject of the present invention is based can be applied without any modification to a postal machine having any number of outputs.

[0031] The description of the method of balancing according to the present invention will be given by making reference to the matrix illustrated in Figure 2, which constitutes a matrix representation of the load of postal objects at each output U of the postal machine 1 at the end of each of the two sorting cycles constituting the

sorting process.

[0032] In particular, each row of the matrix represents the load of postal objects of a respective output U of the postal machine 1 at the end of the first sorting cycle whilst each column of the matrix represents the load of postal objects at a respective output U of the postal machine 1 at the end of the second sorting cycle.

[0033] The load of postal objects at the outputs of the postal machine 1 at the end of the performance of the present method of balancing therefore makes it possible to derive the sorting criteria to be used for the sorting process.

[0034] As illustrated in Figure 2, the matrix has ten rows and ten columns indicated with respective progressive identification numbers. In particular, the identification numbers of the columns are disposed in a progressively increasing order starting from the column furthest to the left of the matrix whilst the numbers identifying the rows are disposed in a progressively increasing order starting from the lowermost row of the matrix.

[0035] Given the correspondence between rows and columns of the matrix and outputs of the postal machine 1 in the first and second sorting cycle respectively, each number identifying rows and columns of the matrix of Figure 2 also identifies a respective output U of the postal machine 1 at the end of the first and second sorting cycle respectively. The effective physical position of the output U in the postal machine 1 does not however necessarily correspond to the progressive numbering of the rows and columns of the matrix of Figure 2, that is the outputs of the postal machine 1 are not necessarily disposed in a progressively increasing order corresponding to the progressively increasing order of the identification numbers of the rows and columns.

[0036] In other words, the output of the postal machine 1 represented by the column identified by the numeral "1" may not be physically the first output of the postal machine 1 itself, the output represented by the column identified by the numeral "2", which column in the matrix is adjacent and successive to the first column, may not be physically the second output of the postal machine 1 itself and nor need it be adjacent and subsequent to the output represented by the column identified by the numeral "1".

[0037] Similar considerations apply also to the other rows and columns of the matrix of Figure 2.

[0038] Therefore, the progressive numbering of the rows and columns is a numbering of "logic" type to which corresponds a "physical" disposition of the outputs related to it by a predetermined relation stored in the electronic control unit 22 and utilised in the sorting process to direct the postal objects towards the desired output.

[0039] In the following description, therefore, the term "logically contiguous outputs" will be understood to mean outputs of the postal machine 1 represented by rows or columns identified with successive identification numerals, such outputs not necessarily being physically

adjacent to one another nor their relative position being deducible from the identification numerals of the respective rows or columns.

[0040] Moreover, thanks to the unequivocal correspondence between the outputs of the postal machine 1 and the rows of the matrix in the first sorting cycle and between outputs of the postal machine 1 and columns of the matrix in the second sorting cycle, in the following description the term outputs of the postal machine 1 and the terms rows and columns of the matrix of Figure 2 will be utilised without distinction in the two sorting cycles and according to descriptive convenience.

[0041] With reference again to the matrix of Figure 2, the cells of the matrix further assume a precise significance related the delivery address or destination of the postal objects and in particular each cell of the matrix defines a respective virtual position within the matrix to which can be associated a real address of a place to which the postal objects must be delivered.

[0042] Since the cells of the matrix, as known, are unequivocally identified by respective pairs of numerical values indicative of the row and column of the cell itself, each virtual position to which a delivery address can be associated can therefore be represented by the pair of numerical values indicating the row and column of the respective cell.

[0043] Moreover, given the unequivocal correspondence between rows and columns of the matrix and outputs of the postal machine 1 in the two sorting cycles, each pair of numerical values indicative of the row and column of the respective virtual position also represents the output of the postal machine 1 which the postal objects having a given delivery address have to occupy at the end of the first and second sorting cycle respectively. In the light of what has been explained above, the present method of balancing is finalised by the creation of a table containing a plurality of data and defining an unequivocal relationship between all the possible codes printed on the postal objects 2 (and able to identify, as previously mentioned, respective delivery addresses for the postal objects 2 themselves), and corresponding virtual positions associated with the delivery addresses indicated by these codes and each identified by a pair of numerical values indicating the row and column of a respective cell of the matrix.

[0044] From this matrix, the method will easily deduce the rules for sorting the delivery addresses of the outputs of the postal machine 1 at the end of the first and second sorting cycle by assigning to a delivery address associated with a respective data cell of the matrix the output of the postal machine 1 corresponding to the number of the row of the cell in the first sorting cycle and the output of the postal machine 1 corresponding to the number of the column of the cell in the second sorting cycle.

[0045] Moreover, since the postal objects present in each output of the postal machine 1 at the end of the second sorting cycle are disposed in a predetermined

order which allows their sequential delivery by a postman travelling along a predetermined route, and this order of distribution being defined by a sequence of adjacent delivery addresses corresponding to house numbers or to groups of house numbers of buildings disposed along the route of the associated postman, the relation between all the possible codes printed on the postal objects 2 and the corresponding virtual positions defined by the said table is such as to define a criterion for allocation of the delivery addresses to the respective outputs of the postal machine 1 such as to allow the said order of delivery of the postal objects to be respected.

[0046] In detail, for each group of columns associated with the postmen the relationship defined by the table allocates the delivery addresses to the cells of the matrix according to an allocation criterion increasing in the sense of the columns and the rows in a manner described hereinafter.

[0047] In particular, the delivery addresses are allocated starting from the cell in the first row of the first column of the matrix (the cell positioned lowermost in the first column) up to the cell positioned in the last row of the first column of the matrix (the highest cell in the first column) and then again starting from the cell positioned in the first row of the second column up to the cell positioned in the last row of the second column, and so on for the successive columns.

[0048] The criterion by which the delivery addresses are allocated to virtual positions (that is to say to the cells of the matrix) therefore defines a constraint on the cells of the matrix of Figure 2 forbidding any exchange of positions between the numerical values represented in the cells of a column.

[0049] With reference again to the matrix of Figure 2, the first and second numerical value indicative of the row and column of each cell of the matrix of Figure 2 assume a precise significance related to the output U1 of the postal machine 1 which the postal object to be delivered at the delivery address associated with the virtual position identified by this pair of values occupies at the end of the two sorting cycles of which the sorting process is composed and in particular the first numerical value indicative of the row of the cell identifies an output Ui of the postal machine 1 at which the postal object to be delivered to the delivery address associated with the cell will have to be disposed at the end of the first sorting cycle, whilst the second numerical value indicative of the column of the cell identifies an output Uj of the postal machine 1 at which the postal object to be delivered to the delivery address associated with the cell will have to be disposed at the end of the second sorting cycle.

[0050] During each sorting cycle, the virtual position associated with it is determined and the pair of numerical values identifying the row and column defining this virtual position is utilised by the postal machine 1 to generate, via tables or sorting rules deduced directly

from the matrix, command signals for actuator members, for example paddle selectors, transmission members etc. (not shown) which contribute to form, along the sorter device 16, a path T which leads the postal object 2 towards the selected output U_i.

[0051] In particular, the postal machine 1 utilises the first numerical value identifying a virtual position (row number) to determine the destination output of the postal object at the end of the first sorting cycle and the second numerical value identifying the virtual position (column number) to determine the destination output of the postal object at the end of the second sorting cycle.

[0052] With reference again to the matrix of Figure 2, numerical values appear in the cells of the matrix itself to which a delivery address is allocated, these numerical values assuming a particular significance related to the traffic of postal objects, and in particular each numerical value can be correlated to the quantity of postal objects expected delivery at the delivery address associated with the cell in which it appears.

[0053] In particular, the numerical value appearing in a cell can be indicative of the quantity of postal objects in absolute or exact terms or else in terms of expected traffic.

[0054] Therefore, a numerical value appearing in a cell is indicative of the fact that this cell has had allocated to it a delivery address having expected traffic or real traffic indicated by the numerical value itself whilst the cells having no numerical values appearing in them represent virtual positions available for the allocation of a delivery address.

[0055] Therefore, for descriptive convenience, hereinafter, a cell containing a numerical value will be indicated by the term "occupied cell", meaning that it is associated with a respective delivery address, whilst a cell not containing any numerical value will be indicated with the term "free cell", meaning that there is no delivery address associated with it.

[0056] Moreover, the sum of the numerical values in each row and each column has associated with it a precise significance related to the load (that is to say the quantity of expected postal objects) at the output of the postal machine 1 corresponding to this row or column and in particular the sum of the numerical values of each row represents the load present at the output of the postal machine 1 associated with this row at the end of the first sorting cycle whilst the sum of the numerical values of each column represents the load present at the output of the postal machine 1 associated with this column at the end of the second sorting cycle.

[0057] Referring again to the matrix of Figure 2, to the side and above the matrix itself are illustrated first and second vectors each containing a number of fields equal to the number of rows and columns respectively of the matrix, that is to say ten fields. Each of the fields of the first vector shown to the side of the matrix is aligned with a respective row of the matrix itself and each of the fields of the second vector shown above the

matrix is aligned with a respective column of the matrix.

[0058] In each of the fields of the first vector there is shown a numerical value equal to the sum of the numerical values appearing in the cells of the row with which it is aligned whilst in each of the fields of the second vector there is shown a numerical value equal to the sum of the numerical values appearing in the cells of the column with which it is aligned.

[0059] In this way a first and second vector respectively represent the overall load at each output of the postal machine 1 at the end of the first and second sorting cycle which comprises the sorting process of the invention.

[0060] As it is possible to see by analysing the numerical values shown in the first and second above-mentioned vector, the matrix representation of Figure 2 is indicative of a load situation at the outputs of the postal machine 1 which is very unbalanced between the first and second sorting cycle. In fact, the first output (that identified with the numeral 1) will have to contain 290 postal objects at the end of the first sorting cycle and only 84 at the end of the second cycle. Therefore implementing the balancing method which is the subject of the present invention will avoid such heavily unbalanced situations which could involve, during the performance of the first sorting cycle, an overload of the first output of the postal machine 1 and therefore could make it necessary to interrupt the operation of the postal machine 1 itself for the purpose of removing the postal objects from the outputs in which they have accumulated.

[0061] In Figure 2, below the matrix there is also illustrated a third vector comprising four fields having different widths, aligned with respective groups of columns of the matrix and identified with the letters A, B, C and D. In particular each field of the third vector is associated with a respective group of columns with which it is aligned and the identifying letter shown in it is indicative of the postman to which this group of columns is allocated.

[0062] In detail, as illustrated in Figure 2, the field of the third vector identified with the letter A is associated with the first and second column of the matrix, the field identified with the letter B is associated with the third column, the field identified with the letter C is associated with the fourth, fifth, sixth and seventh column and the field identified with the letter D is associated with the eighth, ninth and tenth column.

[0063] Therefore, this type of representation illustrates the situation in which a first postman associated with the field identified with the letter A is allocated to the delivery of the postal objects disposed, at the end of the second sorting cycle, in the first and second output of the postal machine 1, a second postman associated with the field identified with the letter B is allocated to the delivery of the postal objects disposed in the third output of the postal machine 1, a third postman associated with the field identified with the letter C is allocated to the delivery of the postal objects disposed in the fourth,

fifth, sixth and seventh output of the postal machine 1, and a fourth postman associated with the field identified with the letter D is allocated to the delivery of the postal objects disposed in the eight, ninth and tenth output of the postal machine 1.

[0064] From the preceding description in relation to the matrix representation of Figure 2, it is evident how this matrix representation allows complete and clear illustration of the expected or real situation at the outputs of the postal machine 1 in each of the two sorting cycles which comprise the sorting process, as well as the traffic to each delivery address, the load of each postman and the postman who will have to deliver the objects disposed in each output of the postal machine 1.

[0065] The method of balancing which is the subject of the present invention will hereinafter be described by making reference to the flow diagram illustrated in Figures 3a - 3k and to a sorting process having the matrix representation of Figure 2.

[0066] Moreover, the method of balancing will be described with reference to a sorting constraint defined by the fact that the various postmen be assigned, at the end of the second sorting cycle, separate outputs of the postal machine 1 in such a way as to avoid the possible confusion which could arise if at the end of the sorting process postal objects relating to the routes of different postmen would have to be present in one output of the postal machine 1.

[0067] As illustrated in Figure 3a, the process initially starts at a block 10 in which general information is acquired on the characteristics of the batch to be handled, that is to say the sorting process to be performed, the number of postmen available for the delivery service, the number of delivery addresses to be served, the traffic of postal objects to each delivery address, known for example on the basis of historical data, the total number of outputs of the postal machine 1, the number of outputs of the postal machine 1 to be reserved for emergency situations, for example for use in substitution for one or more inoperative outputs for the diversion of postal objects, the maximum capacity of each output of the postal machine 1, a margin of tolerance on the capacity of each output in such a way as to guarantee that a desired capacity less than the maximum capacity of the output itself is not exceeded etc.

[0068] From block 10 the process leads then to a block 20 in which the number of outputs of the postal machine 1 usable for the performance of the balancing method and the desired capacity for each output of the postal machine 1 are determined.

[0069] In particular, the number of outputs usable for the performance of the balancing method is determined by subtracting, from the total number of outputs of the postal machine 1, a number of outputs reserved for emergency situations, whilst the desired capacity for each output of the postal machine 1 is determined by subtracting the respective tolerance from the maximum capacity of the output itself.

[0070] After the operations described with reference to block 20 there is performed a procedure for testing the sufficiency of the resources for the performance of the method of balancing and the sorting process.

5 **[0071]** If, using a method to be described below, this procedure establishes a positive result for the sufficiency of the resources for the performance of the balancing method and the sorting process, it generates, for example in the memory of the electronic unit 22 which
10 implements it, a matrix representation similar to that of Figure 2 which allows the allocation of the delivery addresses of the postal objects to the outputs of the postal machine 1 during the first and second sorting cycle of which the sorting process is composed.

15 **[0072]** In particular, from block 20 it leads to a block 30 in which it tests if the number of postmen available for the delivery service is sufficient for the performance of the balancing method, that is to say if the number of available postmen is less than or equal to the number of
20 outputs of the postal machine 1 effectively usable for the balancing method.

[0073] In particular, if the number of available postmen is less than or equal to the number of outputs effectively usable for the performance of the balancing method (output YES from block 30) then the method is potentially performable in relation to the number of postmen
25 which will be available to effect delivery, and therefore from block 30 it leads to block 40, otherwise if the number of postmen available is greater than the number of outputs effectively usable for the performance of the balancing method (output NO from block 30) then the balancing method terminates due to the impossibility of performance because of the insufficiency of the available number of outputs of the postal machine 1.

30 **[0074]** At block 40 there is determined the minimum number of separate outputs of the postal machine 1 to be allocated to each postman and the minimum number of outputs which it is necessary to utilise for the performance of the balancing method.

35 **[0075]** In particular, a minimum number of separate outputs to be allocated to each postman is determined on the basis of the number of delivery addresses which each postman must serve, the traffic to each delivery address and the number of delivery addresses which
40 can be allocated to each output of the postal machine 1, which is equal, as mentioned above, to the maximum number of virtual positions which can be allocated to each output of the postal machine 1 (that is to say equal to the number of cells of each column of the matrix of Figure 2).

45 **[0076]** Once the minimum number of outputs to be allocated to each postman is known the minimum number of outputs which it is necessary to utilise for the performance of the balancing method is determined by summing all the minimum numbers of outputs to be allocated to each postman.

50 **[0077]** From block 20 it then leads to a block 50 in which it is established whether the number of outputs of

the postal machine 1 usable for the performance of the balancing method is sufficient for the performance of the method itself, that is to say if the number of usable outputs for the performance of the balancing method is greater than or equal to the minimum number of outputs which it is necessary to use as determined in block 40.

[0078] In particular, if the number of outputs usable for the performance of the method of balancing is greater than or equal to the minimum number of outputs which it is necessary to use (output YES from block 50) then the balancing method is potentially performable in relation to the number of available outputs of the postal machine 1 and therefore from block 50 it goes to a block 60, otherwise if the number of outputs usable for the performance of the balancing method is less than the minimum number of outputs which it is necessary to use (output NO from block 50) then the balancing method is terminated due to the impossibility of performance because of the insufficient number of available outputs.

[0079] In block 60 the number of virtual positions available for the performance of the present balancing method is determined by subtracting from the theoretical number of available virtual positions, equal to the total number of cells of the matrix of Figure 2, the number of lost virtual positions (that is to say unusable positions) because of the said constraint on the allocation of separate outputs to the various postmen, which is determinable on the basis of the following considerations.

[0080] Having allocated a minimum number of separate outputs to each postman, that is to say a minimum number of separate columns of the matrix of Figure 2, inevitably each postman has been allocated a number of virtual positions, that is to say a number of cells of the matrix of Figure 2, equal to a whole multiple of the number of cells contained in a column, in particular equal to the product between the number of rows of the matrix and the number of separate columns allocated to it.

[0081] Generally, however, the number of virtual positions allocated to each postman is greater than the number of virtual positions which the postman himself would need on the basis of the number of delivery addresses allocated to him and consequently a given number of virtual positions allocated to each postman will not be usable by him and will thus not be utilised in the balancing method simply because of the constraints on the allocation of separate outputs to the postman.

[0082] Therefore, the number of virtual positions not utilised by each postman will be equal to the difference between the number of cells of the matrix allocated to him and the number of delivery addresses which must be served, and the total number of virtual positions lost because of the constraint on the allocation of separate outputs to the postmen will be equal to the sum of the numbers of virtual positions not utilised by each of the postmen.

[0083] From block 60 it then leads to a block 70 in

which it is established whether the number of available virtual positions is sufficient for the performance of the balancing method, that is to say if the number of effectively available virtual positions is greater than or equal to the number of delivery address to be served.

[0084] In particular, if the number of effectively available virtual positions is greater than or equal to the number of delivery addresses to be served (output YES from block 70) then the balancing method is potentially performable in relation to the number of available virtual position and therefore from block 70 it leads on to block 80, otherwise if the number of effectively available virtual positions is less than the number of delivery addresses to be served (output NO from block 70) then the balancing method is terminated due to the impossibility of performance because of the insufficiency of the number of available virtual positions.

[0085] In block 80 it is established whether the maximum traffic which can be sorted by the postal machine 1 is sufficient for the performance of the balancing method, that is to say if the sum of the capacities of the outputs of the postal machine 1 is greater than or equal to the number of postal objects to be sorted.

[0086] In particular, if the maximum sortable traffic is greater than the traffic to be sorted (output YES from block 80) then the balancing method is potentially performable in relation to the sorting capacity of the postal machine 1, the resource testing procedure is terminated and from block 80 it leads to a block 90, otherwise if the maximum sortable traffic is less than the traffic to be sorted (output NO from block) the balancing method is terminated due to the impossibility of performance because of the insufficient capacity of the postal machine 1 to sort delivery addresses.

[0087] In block 90 there is performed a preliminary allocation to each postman of a respective group of outputs of the postal machine 1 on the basis of the number of delivery addresses which each of them must serve and the number of virtual positions associated with each output of the postal machine 1.

[0088] The matrix of Figure 2 illustrates the situation at the outputs of the postal machine 1 after the above-described preliminary allocation phase; in particular, the matrix shows only the outputs of the postal machine 1 allocated to the postmen whilst the outputs not yet assigned to the postmen and therefore available during the balancing method for assignment of delivery addresses are not illustrated.

[0089] If the method of balancing has been terminated during the resource testing procedure, the information on the reason for this procedure having been abandoned (insufficient number of available postmen, insufficient number of available outputs of the postal machine 1, insufficient number of virtual positions, insufficient addressing capacity) are signalled in such a way that an operator can act consequently to modify the initial conditions which have not permitted the performance of the sorting process.

[0090] After the preliminary allocation phase there is performed a procedure for improving the balancing of the load of each of the outputs of the postal machine 1 at the end of the second sorting cycle which forms parts of the sorting process.

[0091] In particular, this procedure operates directly on the matrix of Figure 2 for the purpose of improving the balancing of the load in each of the columns of the matrix itself and involves the performance of a first and second sub-procedure which are consecutive in time.

[0092] In detail, from block 90 of Figure 3a the procedure leads to a block 100 of Figure 3b in which the first sub-procedure is performed, which will hereinafter be described with reference to Figure 3d.

[0093] The first sub-procedure involves performing, for each pair of adjacent columns of each group of columns associated with a postman, a simulation of the sorting of particular delivery addresses between the two columns considered and then real performance of the simulated sorting which determines the most consistent improvement in the balancing of the load in this pair of columns, with consequent modification of the configuration of the pair of columns considered.

[0094] It is evident that if none of the simulated sortings obtains an improvement in the balancing of the load with respect to the starting situation, the configuration of the pair of columns considered remains unchanged.

[0095] It is to be noted that these changes are made whilst respecting the initial allocation criterion corresponding to the order of distribution of the postal objects to the delivery addresses.

[0096] In practice the first sub-procedure causes a shift of the delivery address from the outputs of the postal machine 1 to which they are allocated to logically contiguous outputs of the postal machine 1 respectively associated with the same postman having consideration for the criterion for allocation of the delivery addresses to the respective groups of outputs.

[0097] It is evident that this first sub-procedure is only applicable when there are at least two columns associated with a postman.

[0098] In detail, with reference to Figure 3d, initially it leads to a block 110 in which a pair of adjacent columns to be examined are identified.

[0099] The columns are examined in succession in several iterations, starting from the first two columns of the first group up to the last two on the basis of their increasing numerical order. It is to be noted that the constraint on the allocation of the separate outputs of the postal machine 1 to the various postmen requires that the identified adjacent columns must belong to the same postman, that is to say to the same group of columns, in that it is not admissible that a delivery address assigned to one postman can be shifted into an output assigned to another postman.

[0100] From block 110 it then leads to a block 120 in which the loads TC1 and TC2 in the columns under

examination are determined, each of which is equal to the sum of the numerical values represented in the cells of the respective column. In block 120 there is moreover determined a numerical value TM1 equal to the greatest of the above determined loads TC1 and TC2.

[0101] From block 120 it then leads to a block 130 in which the following two shifts are separately simulated:

- the numerical value shown in the highest occupied cell in the first column of the pair (that is to say the column on the left) is shifted into the lowermost cell in the second column of the pair (that is the column on the right) and the numerical values shown in the cells of this second column before the simulated shift are moved upwards by one position;
- the numerical value shown in the lowest occupied cell in the second column is shifted into the free cell of the first column immediately above the highest occupied cell in the first column itself and the numerical values shown in the other cells of the second column before the simulated shift are moved down by one position.

[0102] These changes are simulated only if the cells of the destination column in to which a numerical value is shifted are not totally occupied.

[0103] In other words, taking for example into consideration the first and second column of the matrix of Figure 2, associated with the postman identified with the letter A, the two following changes are separately simulated:

- the numerical value "15" shown in the fourth cell of the first column is shifted into the first cell of the second column and the numerical values "25" and "22" shown in the first and second cells of the second column before the simulated shift are translated upwardly by one position;
- the numerical value "25" shown in the first cell of the second column is shifted into the fifth cell of the first column and the numerical value "22" shown in the second cell of the second column is moved down by one position.

[0104] Block 130 then leads to block 140 in which the loads TC3 and TC4 on by the two columns under examination in the first simulated shift and the loads TC5 and TC6 on the two columns under examination in the second simulated shift are determined. In block 140 there is moreover determined a numerical value TM2 equal to the greater of the loads TC3 and TC4 and a numerical value TM3 equal to the greater of the loads TC5 and TC6.

[0105] Block 140 then leads to block 150 in which is determined a numerical value Tm equal to the smaller of the numerical value TM1 determined in the block 120 and the numerical values TM2 and TM3 relating to the first and second simulated shift taking place in block

140.

[0106] If the numerical value T_m is equal to TM_2 or TM_3 then the simulated shift of the delivery address from one column to the other of the pair which makes it possible to obtain a load equal to T_m in one of the columns examined results in an improvement of the balancing of the load of the column examined, whilst if T_m is equal to TM_1 then neither of the two simulated changes of the delivery address from one column to the other of the pair makes it possible to obtain a load equal to T_m in one of the columns examined which means there is no improvement in the balancing of the load of the column examined and therefore the initial situation before the simulation is the best.

[0107] Therefore from block 150 it leads then to block 160 in which it is tested whether T_m is equal to TM_1 .

[0108] If T_m is equal to TM_1 (output YES from block 160) then none of the simulated shifts is performed, the configuration of the two columns is not changed and the process goes on to block 180, otherwise if T_m is equal to TM_2 or TM_3 (output NO from block 160) then from block 160 it leads to a block 170 in which the simulated shift which made it possible to obtain the smallest load equal to T_m is performed and the configuration of the two columns is varied in dependence on the simulated shift.

[0109] Block 170 then leads to block 180 in which it is established whether there is a further pair of adjacent columns to be examined in the same group of columns (that is to say associated with the same postman) or, if the columns of one group have already been all examined, in the next group of columns (that is to say associated with the next postman).

[0110] If there is a further pair of adjacent columns under examination (output YES from block 180) it leads from block 180 again to block 110 for repetition of the first sub-procedure for a subsequent pair of adjacent columns, otherwise if all the pairs of adjacent columns have been examined (output NO from block 180) then the first sub-procedure ends.

[0111] With reference again to Figure 3b, having terminated the first sub-procedure, it leads from block 100 to a block 190 in which it is tested if the improvement obtained with the first sub-procedure is sufficient or if a time t_0 has elapsed since this first sub-procedure was started.

[0112] In particular, the improvement obtained is considered sufficient if the loads in each group of columns associated with the various postmen are better balanced, by a predetermined quantity, with respect to the loads which existed before the performance of the first improvement sub-procedure.

[0113] The testing based on the time elapsed from the commencement of the first improving sub-procedure serves on the other hand to allow performance of the first sub-procedure only for a maximum predetermined time for the purpose of not excessively occupying the resources which in practice implement the present bal-

ancing methods.

[0114] In particular, if the improvement obtained with the first sub-procedure is sufficient or a time t_0 has elapsed since this sub-procedure was commenced (output YES from block 190) then it leads from block 190 to a block 200 to perform operations for allocation of further outputs to the postmen on the basis of an order of criticality of the outputs already allocated to the postmen themselves, otherwise if the improvement obtained with the first sub-procedure is not sufficient or a time t_0 from the commencement of this sub-procedure has not yet elapsed (output NO from block 190) then it leads again from block 190 to block 100 for repetition of the first improvement sub-procedure on all the columns of the matrix of Figure 2.

[0115] In block 200 it is tested whether from among the outputs usable for the performance of the method of balancing, determined in block 20, there are any available to be allocated.

[0116] If, among the outputs usable for performance of the balancing methods, there are some available for allocation (output YES from block 200) then it leads from block 200 to a block 210, otherwise if from among the outputs usable for the performance of the balancing method there are no more available to be allocated (output NO from block 200) then it goes on to block 240 in which the above-mentioned second sub-procedure is performed as described in detail hereinafter.

[0117] An order of criticality is assigned to the columns in block 210, which takes into account not only the number of delivery addresses associated with each of the columns but also the traffic at each of the delivery addresses. In particular, the said criterion involves allocation of an order of criticality to the columns based on the highest number of delivery addresses and, for the same number of delivery addresses, the greatest traffic to the delivery addresses.

[0118] In other words, the most critical column will be that having the highest number of occupied cells and the least critical column will be that having the lowest number of occupied cells. Whenever two or more columns have the same number of occupied cells the most critical column among these will be that containing the cell in which the highest numerical value is shown and the least critical column that containing the cell in which the lowest numerical value is shown.

[0119] From block 210 the process passes on to a block 220 in which the most critical column is identified on the basis of the criticality order.

[0120] From block 220 the process then leads to a block 230 in which to the group of columns containing the most critical column identified above (that is to say to the group of columns allocated to the most critical postman) there is assigned a free column, that is to say a column all the cells of which are free.

[0121] In practice, in block 230 a further output of the postal machine 1 to be utilised for the sorting process is allocated to the most critical postman (that is to say the

one which is associated with the above-identified most critical column).

[0122] The allocation of a further column to the group of columns containing the most critical column is obviously followed by a shift by one position of the columns of the subsequent groups in the matrix, from that in which such column was assigned.

[0123] From block 230 the process then again leads to block 100 for repetition of the first sub-procedure for improving the balancing of all the columns of the matrix of Figure 2 including that allocated in block 230.

[0124] As previously mentioned, whenever in block 200 it happens that there are no more outputs usable for performance of the balancing method available to be allocated (output NO from block 200) then from block 200 the process leads to block 240 in which the second sub-procedure is performed.

The second sub-procedure involves performing, for each pair of adjacent columns of each group of columns associated with the postman, a simulation of the bisection of particular delivery addresses belonging to the columns considered into respective pairs of virtually separate delivery addresses and their allocation to different columns.

[0125] According to the second sub-procedure, if a delivery address is bisected, it is removed from the matrix and replaced by two new delivery addresses: the first delivery address obtained by the bisection is allocated to the same cell in which the initial bisected delivery address was located and the second delivery address obtained by bisection is allocated to a cell disposed in the adjacent column to that in which the initial bisected delivery address was disposed and belonging to the same postman. Moreover, the traffic in the first and second delivery address obtained by bisection each have a reduced traffic, typically half, with respect to the traffic to the initial bisected delivery address in such a way that their sum is equal to the traffic to the initial bisected delivery address.

[0126] First and second delivery addresses therefore replace the initial bisected delivery address and in reality represent the same postal address which, via the said bisection, will be differently and more conveniently assigned to the outputs of the postal machine 1 in the first and second sorting cycle. In the light of the above the first and second delivery address obtained by bisection are therefore virtually separated.

[0127] In particular, each of the delivery addresses subjected to bisection is disposed, as better described hereinafter, in a particular position in the columns considered and is subdivided into first and second virtually separate delivery addresses, which are then separately assigned to the columns themselves.

[0128] The simulated bisection which causes the most consistent improvement of the load balancing of the pair of columns is then performed, with consequent modification of the configuration of the pair of columns considered.

[0129] It is apparent that if none of the simulated bisections causes an improvement in the load balancing, the configuration of the pair of columns considered remains unchanged.

[0130] In practice, by means of the second sub-procedure, delivery addresses are bisected into respective pairs of virtually separate delivery addresses and each pair of virtually separate delivery addresses obtained by bisection is allocated to a respective pair of logically contiguous outputs of the postal machine 1, associated with the same postman, comprising the output to which the delivery address from which this pair of virtually separate delivery addresses formed by bisection originated and a logically contiguous output allocated to the same postman.

[0131] Moreover, the allocation of each pair of virtually separate delivery addresses to the respective pairs of logically contiguous outputs is performed whilst still respecting the criterion by which the delivery address from which this pair of virtually separate delivery addresses has originated upon bisection is allocated to the respective group of outputs.

It is therefore evident that this second sub-procedure is only applicable when there are at least two columns associated with a postman.

[0132] In detail, the second sub-procedure is similar to the first sub-procedure described with reference to the flow diagram illustrated in Figure 3d and therefore will hereinafter be described solely as far as it differs therefrom.

[0133] The second sub-procedure differs from the first sub-procedure only by the fact that the simulations effected in block 130 are different from those described for the first sub-procedure.

[0134] In particular, in block 130 the two following subdivisions are separately simulated:

- the numerical value shown in the highest occupied cell in the first column is subdivided into first and second different but quantitatively similar numerical values of traffic, the sum of which is equal to the original numerical value. The first numerical value is then placed in the highest cell in the first column in place of the original numerical value whilst the second numerical value is placed in the lowermost cell in the second column. The numerical values in the cells of the second column before the simulated shift are then translated upwardly by one position;
- the numerical value in the lowermost occupied cell in the second column is subdivided into first and second separate, but quantitatively similar numerical values the sum of which is equal to the original numerical value. The first numerical value is then placed in the first free cell of the first column disposed immediately above the highest occupied cell in the first column itself whilst the second numerical value is placed in the lowermost cell in the second column in place of the original numerical value.

[0135] It is apparent that the subdivision of the numerical value shown in the cells can be performed in a different manner from that described above in that it is by no means essential for the satisfactory performance of the second sub-procedure than the two numerical values obtained by the subdivision be substantially equal to one another.

[0136] During the sorting cycles the postal objects to be delivered to the delivery address subjected to bisection will be directed by the postal machine 1 as if they belonged to one of the two virtually separate delivery addresses obtained by the bisection, by associating them with one or the other delivery address with a probability proportional to the respective traffic values.

[0137] Again, taking into consideration for example the first and second column of the matrix of Figure 2, associated with the postman identified with the letter A the following two simulations are effected separately:

- the numerical value "15" shown in the fourth cell of the first column is subdivided into a first numerical value equal to "7" and a second numerical value equal to "8". The first numerical value "7" is shown in the fourth cell of the first column in place of the original numerical value "15" and the second numerical value "8" is shown in the first cell of the second column. The numerical values "25" and "22" in the first and second cells of the second column before the simulated shift are translated up by one position.
- the numerical value "25" in the first cell of the second column is subdivided into a first numerical value equal to "12" and a second numerical value equal to "13". The first numerical value "12" is then placed in the fifth cell of the first column and the second numerical value "13" is placed in the first cell of the second column in place of the original numerical value "25".

[0138] The second sub-procedure makes it possible to obtain, for example, particularly pronounced results in the improvement of the load balancing when the load of each of the two columns is such that the simulated shifts in the first sub-procedure improve the load of the first column but worsen that of the adjacent column by an approximately equivalent amount and therefore the first sub-procedure does not allow an appreciable overall improvement to be obtained.

[0139] With reference to Figure 3b, having finished the second sub-procedure, the process leads from block 240 to a block 250 in which it is established whether the improvement obtained with the second sub-procedure is sufficient or whether a time t1 has elapsed since this sub-procedure was started.

[0140] If the improvement obtained with the second sub-procedure is significant or the time t1 since this sub-procedure was started has not elapsed (output NO from block 250), then the procedure leads on from block

250 again to block 240 for repetition of the second sub-procedure, otherwise if the improvement obtained with the second sub-procedure is not significant or a time t1 from when this sub-procedure was started has elapsed (output YES from block 250) then there is performed a load balancing improvement procedure of each of the outputs of the postal machine 1 at the end of the first sorting cycle which forms part of the sorting process.

[0141] The load balancing improvement procedure for each of the outputs of the postal machine 1 at the end of the first sorting cycle operates directly on the matrix of Figure 2 at the end of the load balancing improvement of each of the rows of the matrix itself and involves the performance of four consecutive sub-procedures described hereinafter.

[0142] In detail, block 250 of Figure 3b leads on to block 300 of Figure 3c in which the first improvement sub-procedure is performed, which will be described hereinafter with reference to Figures 3e and 3f.

[0143] The first sub-procedure involves performing, for each column of the matrix of Figure 2, a simulation of a redistribution of the delivery addresses of the column over all the cells of the column itself and then the simulation which causes the most consistent improvement in the balancing of the rows is actually performed, with consequent modification of the configuration of the column. It is evident that if none of the simulated shifts causes an improvement in the load balancing the configuration of the column under consideration remains unchanged.

[0144] In other words, the first sub-procedure effects a redistribution of the numerical values in the occupied cells of each column over all the cells of the column itself in such a way as to alternate the free cells and the occupied cells of the columns. In practice, the first sub-procedure performs a redistribution of the delivery addresses assigned to each output of the postal machine 1 at the end of the second sorting cycle on the outputs utilised in the first sorting cycle whilst maintaining respect for the allocation criterion by which the delivery addresses are allocated to the output itself in the second sorting cycle.

[0145] It is evident that this first sub-procedure is applicable only if the column in which the redistribution is effected has at least one free cell.

[0146] In detail, with reference to Figures 3e and 3f, the procedure starts at a block 310 in which a criticality order is allocated to the columns of the matrix of Figure 2 on the basis of a criterion which principally takes account of the number of delivery addresses associated with the columns themselves, that is to say the number of occupied cells in each column.

[0147] Therefore, the most critical column will be that having the greatest number of occupied cells and the least critical column will be that having the smallest number of occupied cells.

[0148] The criticality order of the columns is utilised in the following part of the sub-procedure to effect redistri-

bution of the numerical values shown in the cells of the columns themselves starting from the most critical down to the least critical and to distinguish the most critical column from the remaining columns in that for this column the distribution of the delivery addresses over the cells of the column itself is effected in a different manner from that used for the remaining columns.

[0149] Block 310 leads on to block 320 in which the most critical column is identified on the basis of the allocated criticality order.

[0150] Block 320 then leads onto a block 330 in which, for the most critical column, there is effected a shift, within the column itself, of the numerical values in the occupied cells in such a way as to distribute the free cells uniformly among the occupied cells.

[0151] Taking into consideration, for example, the tenth column of the matrix of Figure 2, which, as can be seen is the most critical column, this column has five occupied cells and five free cells and therefore a uniform distribution among occupied cells and free cells involves a shift of the numerical values in the occupied cells such as to have a free cell interposed between each pair of occupied cells.

[0152] Therefore, the numerical values "26", "48", "13", "11" and "2" shown in the five occupied cells of the tenth column are thus respectively shifted; the numerical value "2" shown in the fifth cell is shifted into the ninth cell, the numerical value "11" shown in the fourth cell is shifted into the seventh cell, the numerical value "13" shown in the third cell is shifted into the fifth cell, the numerical value "48" shown in the second cell is shifted into the third cell and the numerical value "26" shown in the first cell remains in this cell.

[0153] For each successive column from the most critical one the distribution takes place in a different manner described hereinafter.

[0154] Thus, from block 30 the process leads to a block 340 in which a subsequent column to be examined for the redistribution of numerical values shown in the respective occupied cells is identified starting from the second most critical column and down to the least critical column on the basis of the allocated order of criticality.

[0155] Block 340 leads on to a block 350 in which an order of criticality is allocated to the occupied cells of the examined column on the basis of a criterion which principally takes into account the traffic in the delivery addresses associated with the cells themselves, that is to say the numerical value shown in the occupied cells.

[0156] Therefore, the most critical cell will be that having the highest numerical value and the least critical cell will be that having the lowest numerical value.

[0157] Subsequently there are performed operations having the purposed of determining what will be the most convenient shifts within the column under examination of each of the numerical values shown in the occupied cells of the column itself for the purpose of minimisation of the sum of the numerical values shown

in the occupied cells in the rows belonging to the columns already examined.

[0158] In other words, by means of these operations, an optimum position for each of the numerical values within the column under examination is determined.

[0159] The optimum position for each numerical value shown in the occupied cells of the column under examination is determined, while still respecting the said constraint on the progressive order with which the virtual positions are associated with the respective delivery addresses (defined, as previously mentioned, by the sequence of adjacent delivery address corresponding to street numbers or to groups of street numbers, of buildings disposed along the route along which the postal objects must be delivered).

[0160] In particular, this constraint, by prohibiting any exchange in positions between the numerical values shown in the cells of the same column, imposes a condition that a shift in the numerical value shown in one cell corresponds, in the case of superimposition over another numerical value, an equivalent shift of the numerical value shown in the contiguous cells of the same column position above the cell (if the shift is upwards) or below the cell (if the shift is downwards).

[0161] Therefore, for determination of the optimum position of each of the numerical values shown in the occupied cells of the column under examination, block 350 leads on to a block 360 in which there is identified, in the column under examination, an occupied cell to be examined starting from the most critical cell and, in the subsequent repetitions, leading to the least critical cell on the basis of the previously assigned criticality order.

[0162] Block 360 then leads to a block 370 in which the load TC in the row RC in which the cell under examination is disposed is determined by considering, as previously mentioned, only the numerical values in the cells of the row, corresponding to columns already examined.

[0163] Block 370 leads then to a block 380 in which the possible shifts within the respective column of the numerical values in the cell under examination are determined, that is to say the changes for which it is possible to effect a corresponding shift in the numerical values shown in the other occupied cells of the same column.

[0164] Block 380 then leads on to a block 390 in which it is tested whether there is another possible shift to simulate within the same column as the numerical value in the cell under consideration.

[0165] If there is a possible shift to simulate (output YES from block 390) then from block 390 it leads onto block 400, otherwise if there is no possible shift to simulate (output NO from block 390) then from block 390 it leads to block 470 described hereinbelow.

[0166] In block 400 the sums of the numerical values shown in the cells belonging to the columns already examined are determined by row, excluding the column currently under examination and being limited to the

rows in which it is possible to effect a shift of the numerical value shown in the cell under examination and also including the row to which the cell under examination belongs.

[0167] From block 400 it then leads to a block 410 in which, from among the rows taken into consideration in block 400, the row having the minimum traffic sum is determined.

From block 420 it then leads to a block 460 in which the above-identified shift is performed, consequently performing, where necessary, the corresponding shifts of the other numerical values in the other occupied cells of the column.

[0168] It is apparent that no shift will be performed if the numerical value in the cell under examination belongs initially to the row which is that having the minimum traffic sum.

[0169] Block 460 then leads to a block 470 in which the numerical value in the cell under examination is constrained to remain in the position assumed, whether that is the initial position or that to which it has been shifted, in such a way that it is no longer subject to further shifts in the subsequent simulations effected for the other numerical values in the other cells of the column under examination. For example, taking into consideration the ninth column of the matrix of Figure 2, which as can be seen is the second most critical, the most critical cell is the first in which there is shown the numerical value "64". As described above, this numerical value is shifted to one of the second, fourth, sixth and eighth row of the column itself

[0170] The constraint on the progressive order of the virtual positions does not permit, for example, the shift of the numerical value "64" to the tenth cell in that the numerical value "30" shown in the second cell would consequently not be displaceable within the same column.

[0171] From block 470 it then leads to block 480 in which it is checked whether there is a further occupied cell to be examined in the column under examination on which to effect the simulation of the shift and repeat the operations described for the numerical value shown in this cell.

[0172] If there is a further occupied cell to be examined (output YES from block 480) then from block 480 the process moves back again to block 360 to repeat the operations described for the numerical value in the subsequent occupied cell under examination with the constraint of not further displacing the numerical values in the cells already examined, otherwise if all the occupied cells of the column under examination have already been examined (output NO from block 480) then block 480 leads on to block 490.

[0173] In block 490 it is checked whether there is a further column to be examined in the matrix of Figure 2.

[0174] If there is a further column to be examined (output YES from block 490) then from block 490 it leads to block 340 to repeat the operations described for the

occupied cells of the successive column under examination, otherwise if all the columns of the matrix have already been examined (output NO from block 490) then the first sub-procedure terminates.

[0175] With reference again to Figure 3c, having completed the first sub-procedure, it proceeds from block 300 to block 530 in which the second sub-procedure is performed which will be described hereinafter with reference to Figures 3g and 3h.

[0176] The second sub-procedure involves performing, for each row of the matrix of Figure 2 and in an order described in detail hereinafter, a simulation of a shift of each of the numerical values in the row itself into contiguous free cells of the same column and actual performance of the shift only when this makes it possible to obtain improvements in the balance of the rows.

[0177] In other words, the second sub-procedure simulates the shift of each of the numerical values in the occupied cells of each row into adjacent cells of the same column.

[0178] In practice, the second sub-procedure performs a shift of delivery addresses from the outputs of the postal machine 1 to which respective logically contiguous outputs are allocated at the end of the first sorting cycle whilst maintaining the allocation criterion by which the delivery addresses themselves are assigned to respective groups of outputs in the second sorting cycle.

[0179] It is evident that this second sub-procedure is only applicable when at least one of the occupied cells has at least one adjacent free cell.

[0180] In detail, with reference to Figures 3g and 3h, at block 550 the rows of the matrix of Figure 2 are allocated an order of criticality on the basis of a criterion which principally takes account of the number of delivery addresses associated with the rows, that is to say the number of occupied cells in each row.

[0181] Therefore, the most critical row will be that having the greatest number of occupied cells and the least critical row will be that having the smallest number of occupied cells.

[0182] The order of criticality of the rows is utilised to determine the most critical row on which to simulate the shift of the numerical values in the cells of the row itself

[0183] Block 550 leads then to a block 560 in which the most critical row RC to be examined for the shift in the numerical values in its occupied cells is identified.

[0184] From block 560 it then leads to a block 570 in which the load TC of the row RC under examination is determined by summing the numerical values in the occupied cells of the row itself.

[0185] Subsequently, for the purpose of minimisation of the load on the rows, operations are performed for determining the most convenient shift, within the same column, of each of the numerical values in the occupied cells of the row RC under examination from the position occupied in the row RC itself to positions in the rows RT immediately contiguous to the row RC.

[0186] In other words, by means of these operations, it is determined which will be the optimum position, for the purpose of minimising the load of the rows, of each of the numerical values in the occupied cells of the row RC under examination from among the original positions in the row RC itself, that is immediately above and immediately below in the same column.

[0187] It is apparent that, for a given numerical value these operations can only be performed when there is a free cell in the same column in at least one of the rows immediately contiguous to that in which this numerical value appears.

[0188] Therefore, for determination of the optimum position of each of the numerical values shown in these occupied cells of the row RC under examination, block 570 leads on to a block 580 in which there is identified, in the row RC under examination, an occupied cell to be examined, starting from the cell disposed in the first column and proceeding to that disposed in the last column.

[0189] From block 580 it then leads to a block 590 in which there are determined the possible shifts which the numerical value shown in the cell under examination can effect within the same column from the row RC to contiguous rows RT, that is to say the shifts which this numerical value can make towards free cells contiguous to that under examination.

[0190] It is evident that at most there are only two such shifts in that there are only two cells adjacent that under consideration.

[0191] From block 590 it then leads to a block 600 in which it is established whether there is any possible shift of this numerical value from the row RC to a contiguous row RT which can be simulated.

[0192] If there is a possible shift which can be to simulated (output YES from block 600) then from block 600 it leads on to a block 610, otherwise if all the possible shifts have already been simulated (output NO from block 600) then from block 600 it leads on to block 660 described hereinafter.

[0193] In block 610 one of the shifts determined in block 590 of the numerical value in the cell under examination from row RC to one of the contiguous rows RT is simulated and the new load TCN of the row RC and the new load TTN of the row RT after the shift are determined.

[0194] From block 610 it then leads to a block 620 in which a numerical value T_m equal to the greatest two loads TCN and TTN determined above is determined, that is to say T_m equals $\max(TCN, TTN)$.

[0195] From block 620 it then leads to a block 630 in which it is established whether the numerical value T_m is less than the load TC of the row RC under examination before the simulate shift, determined in block 570, that is to say it is checked whether the simulated displacement has achieved an improvement in the balance of the load in the rows RC and RT.

[0196] If the numerical value T_m is less than the load TC, that is to say if the simulated shift has caused

improvements in the balance of the load of the rows RC and RT (output YES from block 630) then from block 630 it leads to block 640, otherwise if the numerical value T_m is greater than or equal to the load TC, that is to say if the simulated shift has not caused any improvement in the balancing of the load of the rows RC and RT (output NO from block 630), then from block 630 it again goes to block 600 to test if there is a further possible shift to simulate from row RC to a contiguous row RT and the above described operations relating to this shift are repeated.

[0197] In block 640 the difference D between the load TC and the numerical value T_m is determined, which is correlated to the magnitude of the improvement of the load of the rows RC and RT obtained by means of the shift.

[0198] From block 640 it then leads onto a block 650 in which the above-determined difference D and the associated shift which allows this difference D to be obtained are memorised, as well as the position of the cell under examination.

[0199] From block 650 it then returns again to block 600 to test if there is further possible shift from row RC to a contiguous row RT to be simulated and to repeat the above-described operations relating to this shift.

[0200] In block 660, to which the process moves if there are no available shifts to simulate or if all the possible shifts of the numerical value shown in the cell under examination have already been simulated, it is checked if there is a further cell to be examined belonging to the row RC under examination.

[0201] If there is a further cell to be examined (output YES from block 660) then from block 660 it moves back to block 580 for repetition of the operations described for a subsequent occupied cell of the row RC, otherwise if all the cells have already been examined (output NO from block 660) then it moves onto a block 670.

[0202] In block 670 it is checked if at least one shift and a respective difference D relating to the row RC under examination has been memorised, that is to say if at least one of the simulated shifts has made it possible to obtain an improvement in the balance of the load on the rows RC and RT.

[0203] If at least one shift has been memorised, that is to say at least one of the simulated shifts has allowed a reduction in the load on the rows RC and RT (output YES from block 670), from block 670 it leads onto a block 680, otherwise if no shift has been memorised, that is to say no simulated shift has allowed a reduction in the load on the rows (output NO from block 670) from block 670 it leads to a block 700.

[0204] In block 680 it is identified which of the differences D memorised is the greatest and this consequently also identifies the shift which allows this maximum difference D to be obtained as well as the position in the cell involved of this shift.

[0205] From block 680 it then leads to a block 690 in which the above-identified shift is performed, with con-

sequent modification of the configuration of the matrix of Figure 2.

[0206] From block 690 it then moves again to block 550 for a new allocation of the order of criticality to the rows and repetition of the above-described operations for the most critical row.

[0207] In block 700, to which the process leads if no shift has been memorised, that is to say no shift in the row under examination has allowed an improvement in the balancing of the load on the rows, the row RC previously examined is considered excluded from the operations subsequently performed on the rows of the matrix of Figure 2 in that any shift in the numerical values of its cells does not allow any improvement in the balancing of the rows themselves.

[0208] From block 700 it then leads to a block 710 in which it is verified if all the rows have been excluded or if a maximum predetermined time has elapsed.

[0209] If not all the rows have been excluded and the maximum predetermined time has not yet elapsed (output YES from block 710), from block 710 it then leads to block 550 for a new allocation of a criticality order to the rows and repetition of the operations described above to the most critical row, otherwise if all the rows have been excluded or if not all the rows are excluded but a maximum predetermined time has elapsed (output NO from block 710) then the second sub-procedure terminates.

[0210] With reference again to Figure 3c, having finished the second sub-procedure, block 530 then moves on to block 720 in which it is tested if the improvement obtained with the second sub-procedure is sufficient or if a time T2 has elapsed since the second sub-procedure was started.

[0211] In particular, the improvement obtained is sufficient if the loads on each group of rows associated with the various postmen are better balanced by a predetermined amount with respect to the loads which they had before the performance of the second improvement sub-procedure.

[0212] If the improvement obtained with the second improvement sub-procedure is not sufficient or the time t2 has not elapsed since this second sub-procedure was started (output NO from block 720) then from block 720 it again returns to block 530 for repetition of the second sub-procedure, otherwise if the improvement obtained with the second improvement sub-procedure is sufficient or a time t2 has elapsed since this second sub-procedure was started (output YES from block 720) then from block 720 it moves on to block 730.

[0213] The third sub-procedure is performed in block 730, which will hereinafter be described with reference to Figures 3i, 3j and 3k.

[0214] The third sub-procedure involves performing a simulation of exchanges between contiguous rows of pairs of delivery addresses belonging to different columns and the performance of such exchanges only when they allow improvements in the balancing of the load of such pairs of columns to be obtained.

[0215] In other words, for each of the numerical values shown in row RC under examination, the third sub-procedure simulates the shifting, within a first column, of a numerical value shown in an occupied cell of row RC into a free cell of an immediately contiguous row RT and the simultaneous shifting, within a second column, of a numerical value shown in an occupied cell of row RT into a free cell of row RC.

[0216] In practice the third sub-procedure is performed by exchanging delivery addresses between contiguous outputs and each exchange is effected whilst still respecting the allocation criterion by which the exchanged delivery addresses are allocated to respective groups of outputs in the second sorting cycle.

[0217] It is apparent that this third sub-procedure is only applicable if there is at least one free cell in each of the said columns.

[0218] In detail, with reference to Figures 3i, 3j and 3k it initially starts at a block 750 in which a criticality order is assigned to the rows of the matrix of Figure 2 on the basis of a criterion which principally takes account of the number of delivery addresses associated with the rows themselves, that is to say the number of occupied cells in each row.

[0219] Therefore, the most critical row will be that having the greatest number of occupied cells and the least critical will be that having the smallest number of occupied cells.

[0220] The order of criticality of the rows is utilised to determine the most critical row to be considered for the simulation of the exchange of pairs of numerical values.

[0221] Block 750 leads onto a block 760 in which the most critical row RC to be examined for exchange of the numerical values in occupied cells thereof is identified.

[0222] From block 760 it then leads to a block 770 in which the load TC of the row RC under examination is determined by summing the numerical values shown in the occupied cells of the row itself.

[0223] Subsequently operations are performed having the purpose of determining, for the purposes of reduction of the load on the rows, which will be the most convenient shift within the said column of each of the numerical values in the occupied cells of the row RC under examination from the position in the row RC to a position in the immediately contiguous row RT and a shift in the opposite direction, contemporaneously, of another numerical value from its position in the row RT to a position in the row RC, obviously positioned in another column.

[0224] In other words, by means of these operations it is determined, for the purpose of minimising the load on the rows, which will be the most convenient positions to exchange of each of the numerical values in the occupied cells of the row under examination with another numerical value positioned in an immediately contiguous row.

[0225] It is evident that for a given numerical value these operations can be performed only if there are free

cells in at least one of the rows immediately above or immediately below the cell in which this numerical value appears.

[0226] Therefore, for the determination of the most convenient exchange of each of the numerical values in the occupied cells of the row RC under examination, block 770 leads onto a block 780 in which an occupied cell to be examined is identified in the row RC under examination, starting from the cell disposed in the first column and ending with that disposed in the last column.

[0227] From block 780 it then moves on to a block 790 in which the shifts which it is possible that the numerical value in the cell under examination can make within the respective column on the rows RT contiguous with the row RC are determined, that is to say the shifts towards the contiguous free cells.

[0228] It is evident that there can be at most two such displacements in that there are only two cells adjacent to that under consideration.

[0229] From block 790 it then leads to a block 800 in which it is established if there is a possible shift of such numerical value from the row RC to a contiguous row RT to be simulated.

[0230] If there is a possible shift to be simulated (output YES from block 800) then from block 800 it leads onto a block 810, otherwise if there is no possible shift to be simulated or if all the shifts have already been simulated (output NO from block 800) then from block 800 it leads onto a block 900 described hereinafter.

[0231] In block 810 one of the shifts determined in block 790 of the numerical value in the cell of row RC under examination and one of the contiguous rows RT is simulated and the load TT of the row RT before the shift is also determined.

[0232] From block 810 it then leads to a block 820 in which it is determined which of the numerical values shown in the occupied cells of the row RT can be shifted, within their column, into free cells of the row RC.

[0233] From block 820 it then leads to a block 830 in which it is checked if there is a possible shift of the numerical values shown in the occupied cells of the row RT from row RT itself to row RC to be simulated.

[0234] If there is a possible shift to be simulated (output YES from block 830) then from block 830 it leads onto a block 840, otherwise if there is no possible shift to be simulated or all the possible shifts have already been simulated (output NO from block 830) then from block 830 it leads back to block 800 to check if there is a further possible shift to be simulated of the numerical value shown in the occupied cell under examination from the row RC to another of the rows RT contiguous to row RC itself and therefore to repeat the above-described operations relating to this further shift.

[0235] In block 840 the shifting of one of the numerical values, determined in block 820, from row RT to row RC is simulated.

[0236] From block 840 it then leads onto a block 850

in which the new load TCN of the row RC after the simulated shift, and the new load TTN of the row RT after the simulated shift are both determined.

[0237] From block 850 it then leads onto a block 860 in which a first numerical value TM1 equal to the greater of the two loads TC and TT of the rows RC and RT before the simulated displacement is determined in blocks 770 and 810, that is to say $TM1 = \max(TC, TT)$, and a second value TM2 equal to the greater of the two loads TCN and TTN of the rows RC and RT after the simulated displacements are determined in the block 840.

[0238] From block 860 it then leads to a block 870 in which it is established if the second numerical value TM2 is less than the first numerical value TM1, that is to say if the exchange of numerical values between the rows RC and RT simulated above has caused an improvement in the loads in the rows RC and RT themselves.

[0239] If the second numerical value TM2 is smaller than the first numerical value TM1, that is to say if the exchange has caused an improvement in the balance of the loads on the rows RC and RT (output YES from block 870) then from block 870 it leads onto a block 880, otherwise if the second numerical value TM2 is greater than or equal to the first numerical value TM1, that is to say if the exchange has not made an improvement in the balance of the loads of the rows RC and RT (output NO from block 870) then from block 870 it leads again to block 830 to check if there is a further possible shift of a numerical value from the row RT to the row RC to be simulated and to repeat the above-described operations for this further shift.

[0240] In block 880 the difference D between the first and second numerical value TM1 and TM2 is determined, which is correlated to the magnitude of the improvement of the load in the rows RC and RT obtained by means of the exchange.

[0241] From block 880 it then leads to a block 890 in which the difference D mentioned above and the associated pair of shifts which allow this difference D to be obtained are memorised, as well as the positions of the cells involved in the said shifts.

[0242] From block 890 it then leads back to block 830 to check if there is a further possible shift of a numerical value from row RT to row RC to be simulated and to repeat the operations described above for the further shift.

[0243] As previously mentioned, whenever there is a further shift possible from the row RT to the row RC to be simulated (output YES from block 830) then from block 830 it leads onto block 840, otherwise if there is no possible shift to be simulated or all the possible shifts from row RT to row RC have been simulated (output NO from block 830) then from block 830 it leads to block 800 to check if there is a further possible shift from row RC to row RT to be simulated and to repeat the operations described above relating to this further shift.

[0244] If there is a further possible shift from row RC to row RT to be simulated (output YES from block 800) then, as previously mentioned, from block 800 it leads onto block 810, otherwise if there is no possible shift to be simulated or if all the possible shifts from row RC to row RT have been simulated (output NO from block 800) then from block 800 it goes onto block 900.

[0245] In block 900 it is checked to see if there is a further cell of the row RC to be examined.

[0246] If there is a further cell in row RC to be examined (output YES from block 900) then from block 900 it leads back to block 780 to repeat the described operations for a subsequent occupied cell of row RC, otherwise if all the cells of row RC have been examined (output NO from block 900) then it leads to block 910.

[0247] Block 910 checks if at least one pair of shifts and a respective difference D relating to the row RC under examination has been memorised, that is to say if at least one of the simulated exchanges has allowed an improvement in the balance of the load in rows RC and RT to be obtained.

[0248] If at least one pair of shifts has been memorised, that is to say at least one of the simulated exchanges has allowed the load in the rows RC and RT to be reduced (output YES from block 910) block 910 leads onto block 920, otherwise if no pairs of shifts have been memorised, that is none of the simulated exchanges has allowed the load on the rows RC and RT to be reduced (output NO from block 910) from block 910 it leads on to block 940.

[0249] In block 920 it is identified which of the memorised differences D is the greatest and consequently the pair of shifts which allow this differences D to be obtained are also identified as well as the positions of the cells involved in these shifts.

[0250] From block 920 it then leads to a block 930 in which the pair of shifts identified above are performed with consequent modification of the configuration of the matrix of Figure 2.

[0251] From block 930 it then leads back to block 750 for a new assignment of a criticality order to the rows and repetition of the above-described operations for the most critical row.

[0252] In block 940, to which the process leads if no pairs of shifts have been memorised, that is to say if no simulated shift allows an improvement of the balancing of the load in the rows RC and RT to be obtained, the previously examined row RC is considered excluded from the operations subsequently performed on the rows of the matrix of Figure 2 in that no shift of the numerical values of its cells allows any improvement in the balancing of the rows themselves to be obtained.

[0253] From block 940 it then leads to a block 950 in which it is checked if all the rows have been excluded or if a maximum predetermined time has elapsed.

[0254] If not all the rows have been excluded and a predetermined maximum time has not yet elapsed (output YES from block 950) from block 950 it leads back to

block 750 for a new allocation of an order of criticality to the rows and repetition of the above-described operations for the most critical row, otherwise if all the rows have been excluded or not all the rows have been excluded but a maximum predetermined time has elapsed (output NO from block 950) then the third sub-procedure terminates.

[0255] With reference again to Figure 3c, having terminated the third sub-procedure, from block 730 it leads to a block 970 in which it is checked if the improvement obtained with the third sub-procedure is sufficient or if a time t3 from when this third sub-procedure started has elapsed.

[0256] In particular, the improvement obtained is sufficient if the loads in each group of rows associated with the various postmen are better balanced, by a predetermined quantity, with respect to the loads which they had before execution of the third improvement sub-procedure.

[0257] If the improvement obtained with the third improvement sub-procedure is significant, or rather greater than a predetermined threshold, or if a time t3 from when the third sub-procedure is started has not elapsed (output NO from block 970) then from block 970 it leads back to block 730 for the repetition of the third sub-procedure, otherwise if the improvement obtained with the third improvement sub-procedure is not significant or if a time t3 from when this third sub-procedure is commenced has elapsed (output YES from block 970) then from block 970 it leads onto a block 980.

[0258] In block 980 the fourth sub-procedure is performed, which involves effecting, for each of the rows of the matrix of Figure 2 and in a predetermined order described in detail hereinafter, a simulation of a subdivision (bisection) of each of the delivery addresses disposed in these rows into two virtually separate delivery addresses and their separate allocation to nearby free cells of the same column and a performance of this subdivision only if it allows improvements in the balancing of the load in this pair of rows to be obtained.

[0259] In the fourth sub-procedure, therefore, if a delivery address is bisected, it is removed from the matrix and replaced by two new delivery addresses: the first delivery address obtained by bisection is allocated to the same cell in which the initial bisected delivery address was located and the second delivery address obtained by bisection is allocated to a cell disposed in the same column in which the initial bisected delivery address was disposed but in a nearby row, compatible with the available free cells of the matrix. Moreover, the traffic at the first and second delivery address obtained by bisection is reduced, typically to half, with respect to the traffic at the bisected initial delivery address in such a way that their sum is equal to the traffic at the bisected initial delivery address.

[0260] The first and second delivery address therefore replace the bisected initial delivery address and represent in reality the same postal address, which, via the

said bisection will now be differently and more conveniently assigned to the outputs of the postal machine 1 in the first and second sorting cycle. In the light of the above, the first and second delivery address obtained by bisection are therefore virtually separate.

[0261] Via the fourth sub-procedure bisection of a delivery addresses into respective pairs of virtually separate delivery addresses is effected and each pair of virtually separate delivery addresses obtained by bisection is assigned to a respective pair of outputs comprising the output to which is associated the original delivery address from which this pair of virtually separate delivery addresses has originated, and a second output identified by the position to which the second virtual delivery address is assigned.

[0262] Moreover, the allocation of each pair of virtually separate delivery addresses to the respective pairs of outputs is effected whilst respecting the allocation to the respective output in the second sorting cycle of the delivery address from which this pair of virtually separate delivery addresses has originated following bisection.

[0263] It is evident that this fourth sub-procedure is only applicable if at least one of the cells close to that under examination is free.

[0264] The fourth sub-procedure is similar to the second sub-procedure previously described with reference to Figures 3g and 3h and therefore will be described only inasmuch as it differs from the second sub-procedure.

[0265] The fourth sub-procedure differs from the second sub-procedure only by the fact that rather than simulate shifts on contiguous rows of the numerical value in the occupied cell under examination, the subdivision of this numerical value into separate and quantitatively similar first and second numerical values the sum of which is equal to the original numerical value is simulated.

[0266] The first numerical value is therefore represented in the cell in which the original numerical value is formed whilst the second numerical value is shown in a nearby free cell in the same column.

[0267] It is apparent that the subdivision of the original numerical value can be effected in a different manner from that described above in that it is not absolutely essential for a good result of the second sub-procedure that the two numerical values obtained by the subdivision be quantitatively similar to one another.

[0268] At the end of the simulation of the bisection of all the numerical values in the occupied cells of the row under examination the subdivision which is associated with the maximum difference D is then actually performed.

[0269] Having completed the fourth sub-procedure, from block 980 of Figure 3c the process leads to a block 990 in which it is checked if the improvement obtained with the fourth sub-procedure is significant or if a time t4 from when this fourth sub-procedure was initiated has

elapsed.

[0270] In particular, the improvement obtained is considered significant if the loads in each group of rows associated with the various postmen are better balanced by a greater quantity than a predetermined threshold, with respect to the loads which they had before performance of the fourth improvement sub-procedure.

[0271] If the improvement obtained with the fourth sub-procedure is significant or if a time t4 from when this fourth sub-procedure was initiated has not elapsed (output NO from block 990), then from block 990 it returns again to block 980 for repetition of the fourth sub-procedure, otherwise if the improvement obtained with the fourth sub-procedure is not significant or if a time t4 from when this fourth sub-procedure was initiated has elapsed (output YES from block 990) then the balancing method has terminates.

[0272] As initially mentioned, the balancing method described above relates to a postal machine 1 having a single input I.

[0273] If the postal machine 1 were to have more inputs for the postal objects the sorting process would be similar to that described and differs from this as follows.

[0274] In particular, if the postal machine 1 had several inputs, each of these would be assigned a first subset of the outputs at the end of the first sorting cycle and a second subset of the outputs at the end of the second sorting cycle. In other words, the postal objects entering into the postal machine 1 through the first input would be deposited in a first subset of the outputs of the postal machine 1 itself at the end of the first sorting cycle and in a second subset of the outputs of the postal machine 1 at the end of the second sorting cycle.

[0275] A similar argument holds true for the postal objects entering the postal machine 1 through the second input, the third input, and so on.

[0276] Therefore, by identifying the first and the second subset of the outputs of the postal machine 1 associated with each input on the rows and columns respectively of the matrix of Figure 2 the groups of rows and groups of columns respectively associated with each input of the postal machine 1 at the end of the first and second sorting cycle respectively are consequently identified.

[0277] By determining the intersection of each group of rows with the respective group of columns there are identified separate groups of cells of the matrix of Figure 2, each of which defines a respective sub matrix contained in the matrix of Figure 2.

[0278] Each sub matrix therefore constitutes a matrix representation of a postal machine to all intents having one input and the associated subset of outputs and can therefore be treated by the same standard as the matrix of Figure 2 without modifications with respect to what has been previously described, thus allowing the balancing of the subset of outputs associated with the

respective input to be improved.

[0279] As initially stated, the balancing method has been described with reference to a postal machine 1 performing a sorting process composed of two successive sorting cycles.

[0280] The inventive concept on which the present invention is based can nevertheless also be utilised in a sorting process composed of several successive sorting cycles.

[0281] In this case, in the last sorting cycle the first procedure will be applied in the manner described, in the first sorting cycle the second procedure will be applied in the manner described, whilst in the second sorting cycle (intermediate sorting cycle) it can be chosen not to apply any of the said procedures or one or two or all of them can be applied.

[0282] As initially stated, the method according to the present invention can also be utilised of balancing the load in any type of sorting process performed by means of one or more devices able to separate streams of objects to put them in a predetermined sequence for directing them to delivery addresses or destinations.

[0283] Therefore, the term "machine" utilised in the preceding description generally indicates, in these different applications of the present method of balancing, the device or devices utilised for the performance of the sequencing process.

[0284] In particular, in these different applications of the present balancing method, the machine is able to withdraw, ordered from an input, constituted by a generic number of expected zones, a sequence of objects the order of which can be unknown beforehand and each of which contains sufficient elements for a classification by the machine itself to guarantee the transport of the objects themselves towards an output constituted by deposition zones and/or release zones, which are dependent on the class of object identified according to predetermined rules and can also coincide with the said expected zones.

[0285] From a study of the characteristics of the balancing method according to the present invention the advantages which it allows are evident.

[0286] Finally, it is clear that the method described and illustrated here can have modifications and variations introduced thereto without by this departing from the protective ambit of the present invention.

[0287] For example, the criticality criteria adopted for ordering the rows and columns and the criteria on the basis of which the various sub-procedures are terminated can be different from those described.

[0288] Moreover, in the sub-procedures in which ordering criteria on the base of criticality are not applied, the method by which the rows and columns are examined can be different from that described.

[0289] Moreover, in the sub-procedures described with reference to blocks 530, 730 and 980 the shifts and exchanges between numerical values can be effected in different ways from those described, and in particular (if

possible) may be effected also in positions which are not immediately contiguous but spaced by two or more cells.

[0290] Similar arguments can be given for the sub-procedure described in block 980 and therefore if possible the virtually separate numerical values obtained by bisection can be assigned to cells positioned in rows which may even be not immediately contiguous.

[0291] Moreover, the criterion for allocation of the delivery addresses to the cells of the matrix can be different from that described.

By utilisation of a different allocation criterion from that described a series of variations in the manner of shifting the delivery addresses can be derived.

[0292] In particular, as previously mentioned, the criterion for allocation of the delivery addresses to the cells of the matrix defines a constraint prohibiting any exchange of positions between the numerical values in the cells of one column and this constraint requires the various shifts and bisections of numerical values described above in the sub procedures illustrated in blocks 100 and 240 to be effected whilst respecting this constraint. Therefore, by utilising a different criterion from that described it is apparent that any shift in the numerical values would have to respect this different allocation criterion and therefore the shifts would be effected between columns considered "contiguous" according to the allocation criterion adopted.

[0293] Consequently, even the "logical contiguity" of the outputs of the postal machine, which in the example described is defined by a memorised table and is constant in all the sorting cycles, and also at the level of the various sub-procedures, could be different and in each sub-procedure of the balancing method the outputs of the postal machine could be contiguous to one another according to a different logical criterion.

[0294] Finally, the balancing of the load of the outputs of the postal machine could be simplified by performing only one of the said first and second balancing procedures in the respective sorting cycle and this simplification, even if on the one hand it would cause an inevitable reduction in the efficacy of the balancing of the load of the outputs of the postal machine, it would nevertheless allow on the other hand a significant saving both in operating time and in occupation of resources required in the implementation of the method itself.

Claims

1. A method of balancing the load in a process of sorting objects (2), in particular postal objects, the said sorting process comprising the performance of a first and at least a second consecutive sorting cycle and being achieved by means of a machine (1) receiving a set of objects (2) at its input (I) and providing at its output (U) the said objects (2) identified and separated according to determined sorting rules; in a given sorting cycle the objects (2) being

delivered to the outputs (U) of the machine (1) according to a respective predetermined sorting criterion and being returned in their ordered state to the input (I) of the machine (1) itself for the performance of a subsequent sorting cycle; the outputs (U) of the machine (1) being subdivided, in the last sorting cycle of the sorting process, into separate subsets (A, B, C, D) and each subset (A, B, C, D) being allocated to a plurality of respective destinations according to an allocation criterion correlated to the order of delivery of the objects to the destinations;

the said balancing method being characterised by the fact that it comprises the phase of performing, in at least one of the said sorting cycles, at least one of the following procedures:

a) a first procedure (100, 210-240) of balancing the load at the outputs (U) of the machine (1) at the end of the current sorting cycle based on at least one of the following balancing criteria;

- displacement of destinations from the outputs (U) to which they are allocated to respective logically contiguous outputs (U) belonging to the same subset (A, B, C, D) according to a first predetermined logical criterion;
- allocation of further outputs (U) to the said subsets (A, B, C, D) on the basis of an order of criticality of the outputs already allocated to the subsets (A, B, C, D) themselves; and
- bisection of destinations into respective pairs of virtually separate destinations and allocation of each pair of virtually separate destinations to a respective pair of logically contiguous outputs (U) belonging to the same subset (A, B, C, D) according to a second predetermined logical criterion; and

b) a second procedure (300, 530, 730, 980) of balancing the load at the outputs (U) of the machine (1) at the end of the current sorting cycle based on at least one of the following balancing criteria;

- distribution over the outputs (U) utilised in the current sorting cycle of destinations allocated to each output (U) at the end of the subsequent sorting cycle;

- shifting of the destinations from the outputs (U) to which they are allocated to respective logically contiguous outputs (U) according to a third predetermined logical criterion;
- exchange of destinations between logically contiguous outputs (U) according to a fourth predetermined logical criterion; and
- bisection of destinations into respective pairs of virtually separate destinations and allocation of each pair of virtually separate destinations to a respective pair of logically contiguous outputs (U) according to a fifth predetermined logical criterion.

2. A method according to Claim 1, characterised in that the said first procedure (100, 210-240) comprises the phase of performing a first sub-procedure (100) of shifting destinations from the outputs to which they are allocated to respective logically contiguous outputs belonging to the same subset (A, B, C, D); the said first sub-procedure (100) comprising the phase of shifting the destinations whilst respecting the allocation criterion by which the destinations themselves are allocated to respective subsets (A, B, C, D) of outputs (U) in the last sorting cycle of the sorting process.

3. A method according to Claim 1 or Claim 2, characterised in that the said first procedure (100, 210-240) comprises the phase of performing a second sub-procedure (210-230) of allocating further outputs (U) to the said subsets (A, B, C, D) on the basis of an order of criticality of the outputs (U) already allocated to the subsets (A, B, C, D) themselves; the said order of criticality being a function of the number of destinations allocated to each output (U) and the traffic of objects to each destinations.

4. A method according to any preceding claim, characterised in that the said first procedure (100, 210-240) comprises the phase of performing a third sub-procedure (240) of dividing destinations into respective pairs of virtually separate destinations and assigning each pair of virtually separate destinations to a respective pair of logically contiguous outputs (U) belonging to the same subset (A, B, C, D); the said third sub-procedure (240) comprising the phase of assigning each pair of virtually separate destinations to a pair of logically contiguous outputs (U) comprising one output with which the destination from which this pair of virtually separate destinations has been formed by division is associated and a logically contiguous output (U) belong-

ing to the same subset (A, B, C, D).

5. A method according to Claim 4, characterised in that the said third sub-procedure (240) comprises the phase of allocating each pair of virtually separate destinations obtained upon bisection to respective pairs of logically contiguous outputs (U) whilst respecting the allocation criterion by which the destination from which this pair of virtually separate destinations was formed by bisection was assigned to respective subsets (A, B, C, D) of outputs (U) in the last sorting cycle of the sorting process. 5
6. A method according to any preceding claim, characterised in that the said second procedure (300, 530, 730, 980) comprises the phase of performing a fourth sub-procedure (300) of distributing destinations allocated to each output (U) at the end of the subsequent sorting cycle over the outputs (U) utilised in the current sorting cycle; the said fourth sub-procedure (300) comprising the phase of distributing the destinations whilst respecting the allocation criterion by which the destinations themselves are allocated to the associated output in the last sorting cycle of the sorting process. 10 15 20 25
7. A method according to any preceding claim, characterised in that the said second procedure (300, 430, 730 980) comprises the phase of performing a fifth sub-procedure (530) of shifting destinations from the outputs to which they are allocated to respective logically contiguous outputs; the said fifth sub-procedure (530) comprising the phase of displacing the destinations whilst respecting the allocation criterion by which the destinations themselves are allocated to the respective subsets of outputs in the last sorting cycle of the sorting process. 30 35 40
8. A method according to any preceding claim, characterised in that the said second procedure (300, 430, 730, 980) comprises the phase of performing a sixth sub-procedure (730) of exchanging destinations between logically contiguous outputs; the said sixth sub-procedure (730) comprising the phase of effecting each exchange by shifting a first destination from a first output to which it is allocated to a second logically contiguous output and simultaneously shifting a second destination from the said second output to the said first output; each of the said exchanges being effected whilst respecting the allocation criterion by which the first and second destination are allocated to respective subsets of outputs in the last sorting cycle of the sorting process. 45 50 55
9. A method according to any preceding claim, char-

acterised in that the said second procedure (300, 530, 730, 980) comprises the phase of performing a seventh sub-procedure (980) of bisecting destinations into respective pairs of virtually separate destinations and allocating each pair of virtually separate destinations obtained upon bisection to a respective pair of logically contiguous outputs; the said seventh sub-procedure (980) comprising the phase of allocating each pair of virtually separate destinations to a pair of logically contiguous outputs comprising the output with which the destination from which this pair of virtually separate destinations has originated upon bisection is associated and a logically contiguous output.

10. A method according to Claim 9, characterised in that the said seventh sub-procedure (980) comprises the phase of allocating each pair of virtually separate destinations obtained upon bisection to respective pairs of logically contiguous outputs whilst respecting the assignment criterion by which the destination from which this pair of virtually separate destinations has originated upon bisection is assigned to respective subsets of outputs in the last sorting cycle of the sorting process.
11. A method according to any of Claims 2, 3 or 4, characterised in that it further comprises the phases of:
 - verifying (190) if the improvement obtained with the said first sub-procedure (100) is not significant or whether a first maximum predetermined time (t_0) from when the first sub-procedure (100) commenced has elapsed; and
 - repeating the said first sub-procedure (100) if the improvement obtained is significant or the said first maximum predetermined time (t_0) has not elapsed and;
 - performing the said second and third sub-procedure (210-230, 240) if the improvement obtained is not significant or the said maximum predetermined time (t_0) has elapsed.
12. A method according to Claim 11, characterised in that it further comprises the steps of:
 - checking (200) if there are outputs (U) of the machine (1) available to allocate to the said subsets (A, B, C, D);
 - performing the said second sub-procedure (210-230) if there are outputs available to allocate to the said subsets (A, B, C, D); and
 - performing the said third sub-procedure (240) if there are no available outputs to allocate to the said subsets (A, B, C, D).
13. A method according to Claim 12, characterised in that it further comprises the steps of:

- repeating the said first sub-procedure (100) after the performance of the said second sub-procedure (210-230).

14. A method according to any of Claims from 11 to 13, characterised in that it further comprises the steps of:

- checking (250) if the improvement obtained with the said third sub-procedure (240) is not significant or if a second maximum predetermined time (t1) from when the second sub-procedure (240) commenced has elapsed;
- repeating the said third sub-procedure (240) if the improvement obtained is significant or if the said second maximum predetermined time (t1) has not elapsed; and
- performing the said second procedure (300, 530, 730, 980) for improvement of the balancing of the load of the outputs (U) of the machine (1) at the end of the first sorting cycle if the improvement obtained is not significant or the said second maximum predetermined time (t1) has elapsed.

15. A method according to any of Claims 6, 7, 8 or 9, characterised in that it further comprises the steps of:

- checking (720) if the improvement obtained with the said fifth sub-procedure (530) is not significant or if a third maximum predetermined time (t2) from when the fifth sub-procedure (530) commenced has elapsed; and
- repeating the said fifth sub-procedure (530) if the improvement obtained is significant or if the said third maximum predetermined time (t2) has not elapsed; and
- performing the said sixth sub-procedure (730) if the improvement obtained is not significant or if the said third maximum predetermined time (t2) has elapsed.

16. A method according to Claim 15, characterised in that it further comprises the steps of:

- checking (970) if the improvement obtained with the said sixth sub-procedure (730) is not significant or a fourth maximum predetermined time (t3) from when the sixth sub-procedure (730) commenced has elapsed; and
- repeating the said sixth procedure (730) if the improvement obtained is significant or if the said fourth maximum predetermined time (t3) has not elapsed; and
- performing the said seventh sub-procedure (980) if the improvement obtained is not significant or if the said fourth predetermined maxi-

imum time (t3) has elapsed.

17. A method according to Claim 16, characterised in that it further comprises the steps of:

- checking (990) if the improvement obtained with the said seventh sub-procedure (980) is not significant or if a fifth predetermined maximum time (t4) from when the seventh sub-procedure (980) commenced has elapsed; and
- repeating the said seventh procedure (980) if the improvement obtained is significant or if the said fifth predetermined maximum time (t4) has not elapsed; and
- terminating the said sorting process if the improvement obtained is not significant or if the said fifth predetermined maximum time (t4) has elapsed.

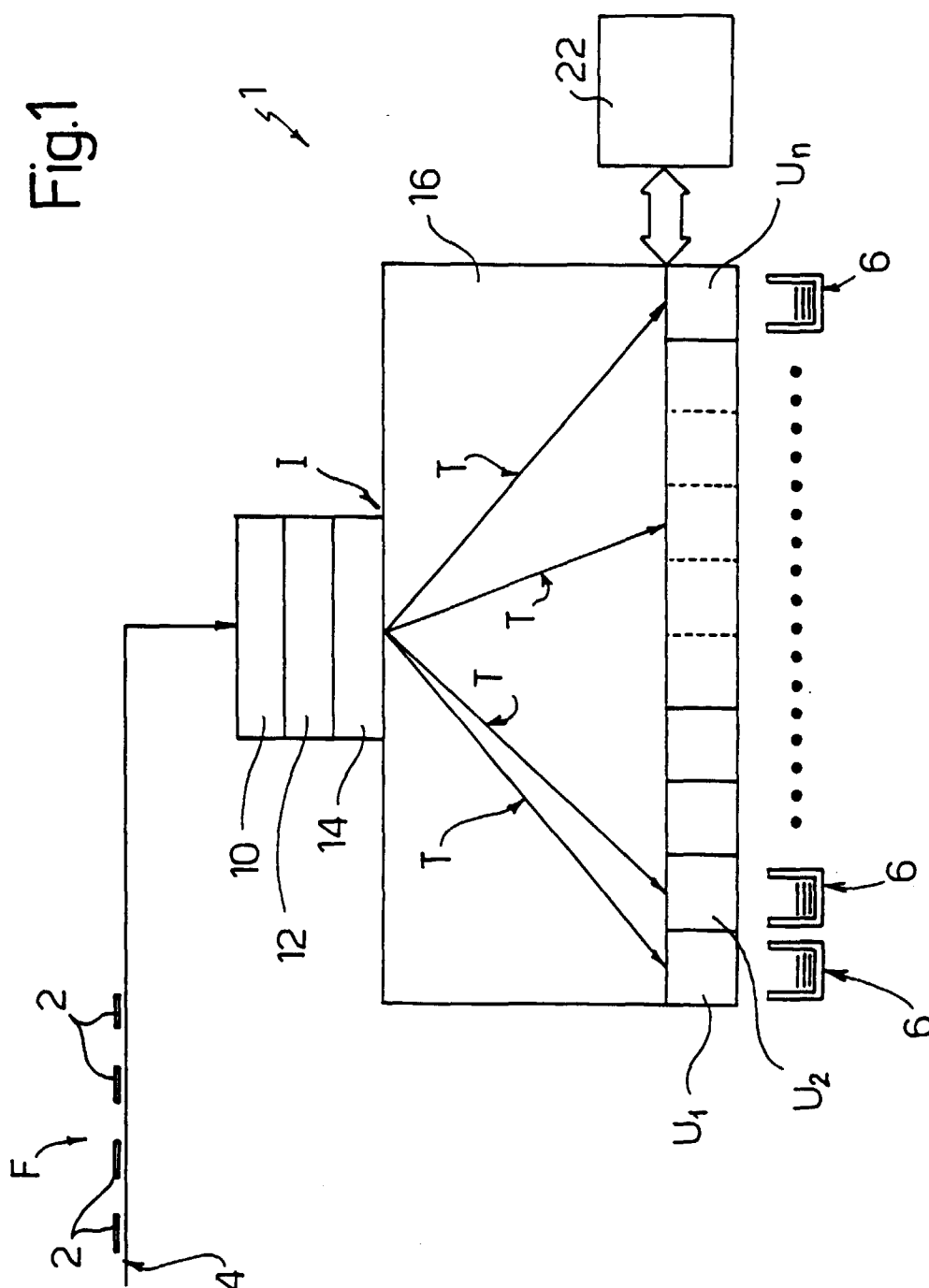
18. A method according to any preceding claim, characterised in that it comprises the steps of:

- performing the said fifth sub-procedure (530) after the said fourth sub-procedure (300).

19. A method according to any preceding claim, characterised in that it further comprises the steps of:

- before the said first and second procedure for improving the balancing of the load of the outputs of the machine (1) at the end of the second and first sorting cycle respectively, performing a third procedure (30-80) for testing for sufficiency of the resources for the performance of the sorting process;
- performing the said first and second procedure (100, 210-240, 300, 530, 730, 980) for improvement of the balancing of the load of the outputs (U) of the machine (1) at the end of the second and, respectively, the first sorting cycle if the said third procedure (30-80) has given positive results; and
- if the said third procedure (30-80) has given negative results terminating the said sorting process and signalling which insufficiency of resources has caused the non-performance of the sorting process.

Fig. 1



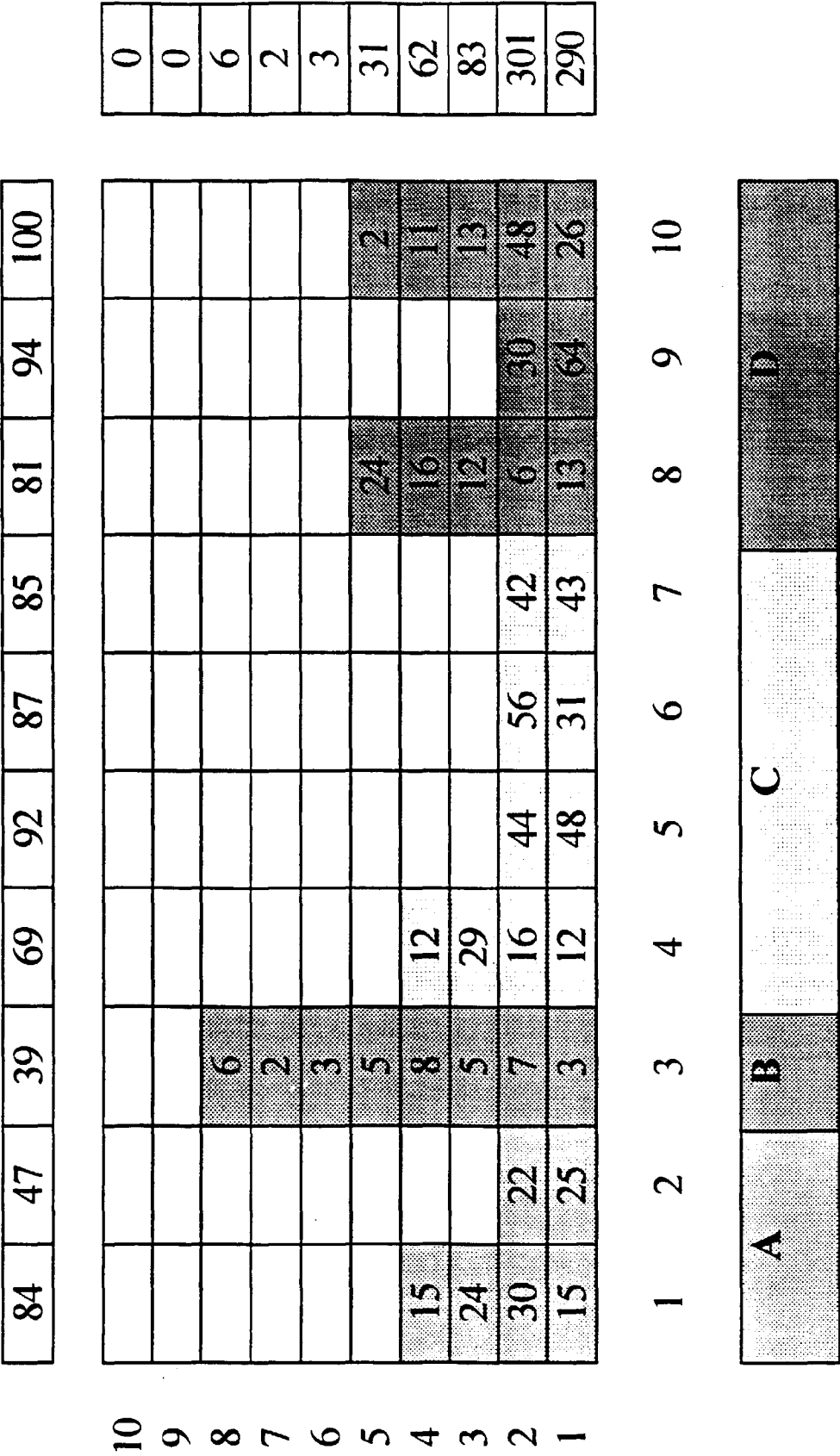


Fig. 2

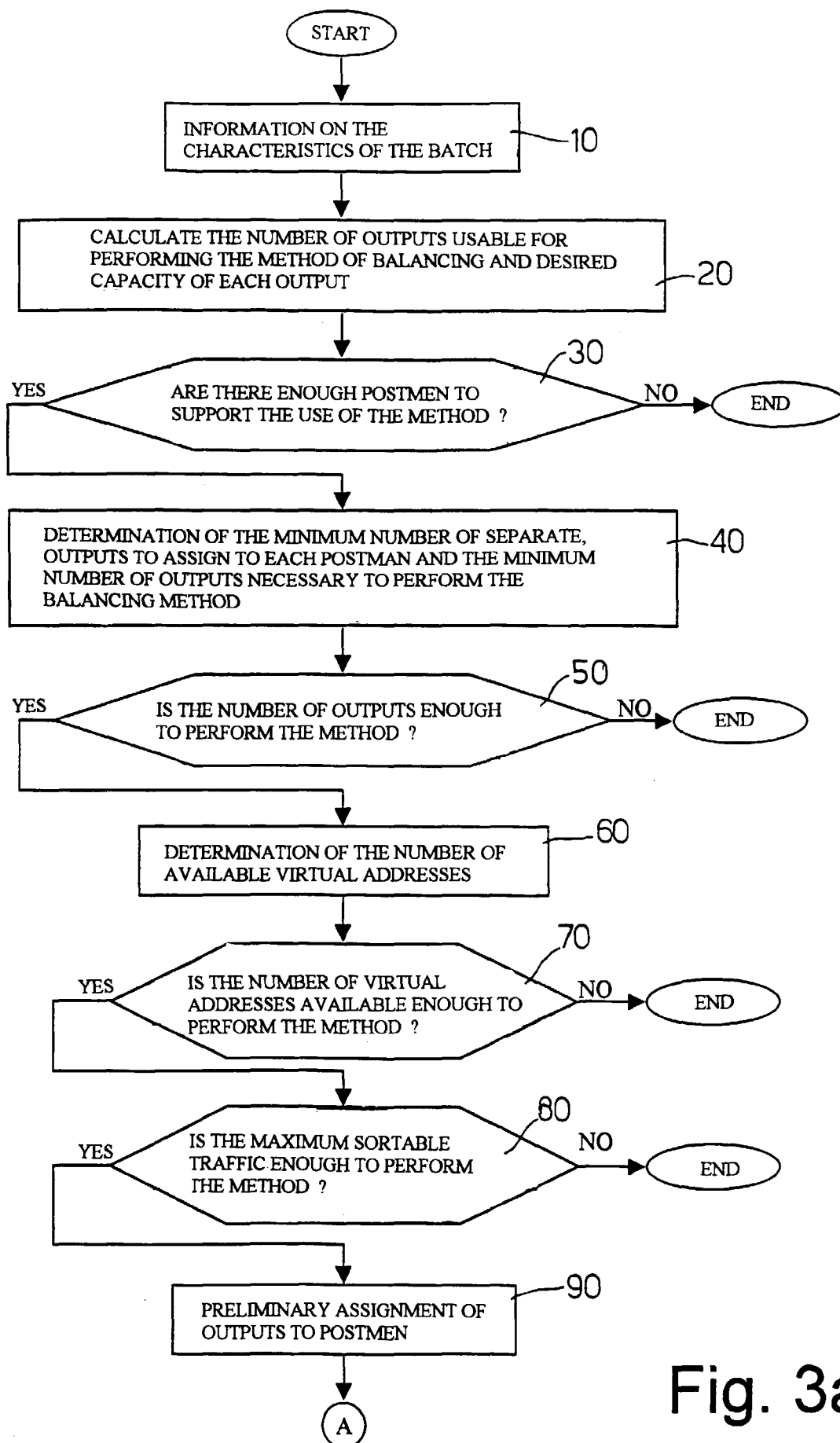


Fig. 3a

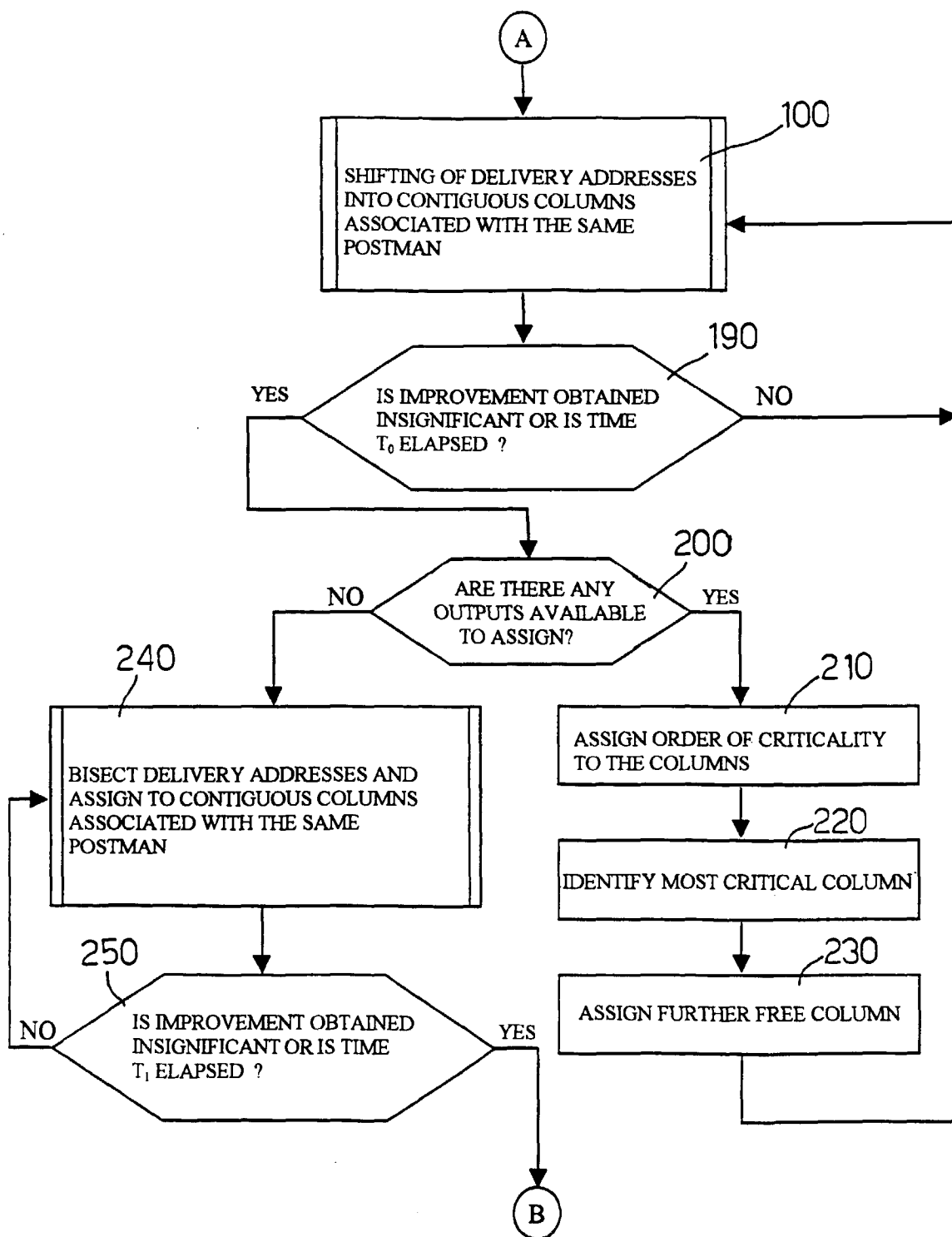


Fig. 3b

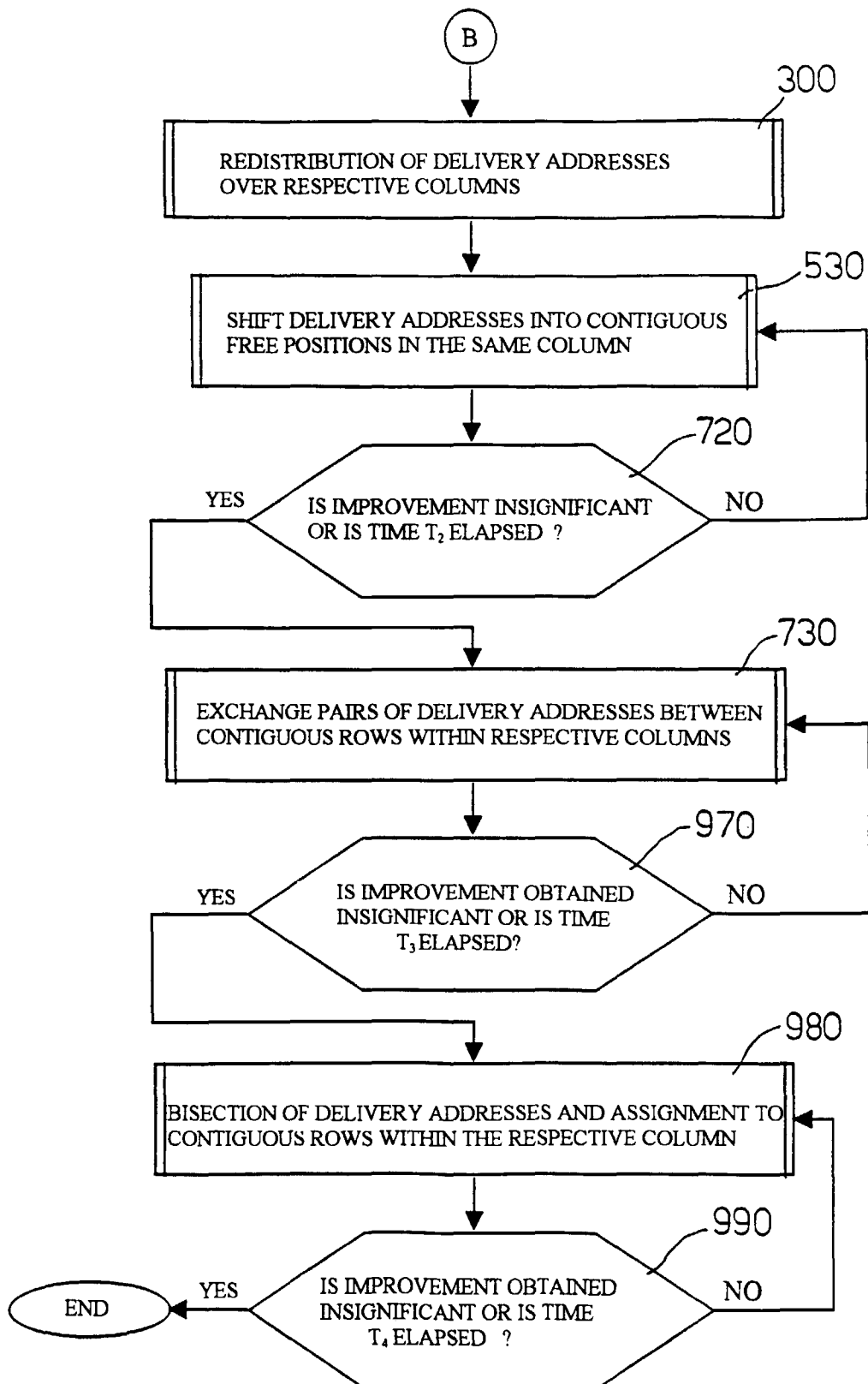


Fig. 3c

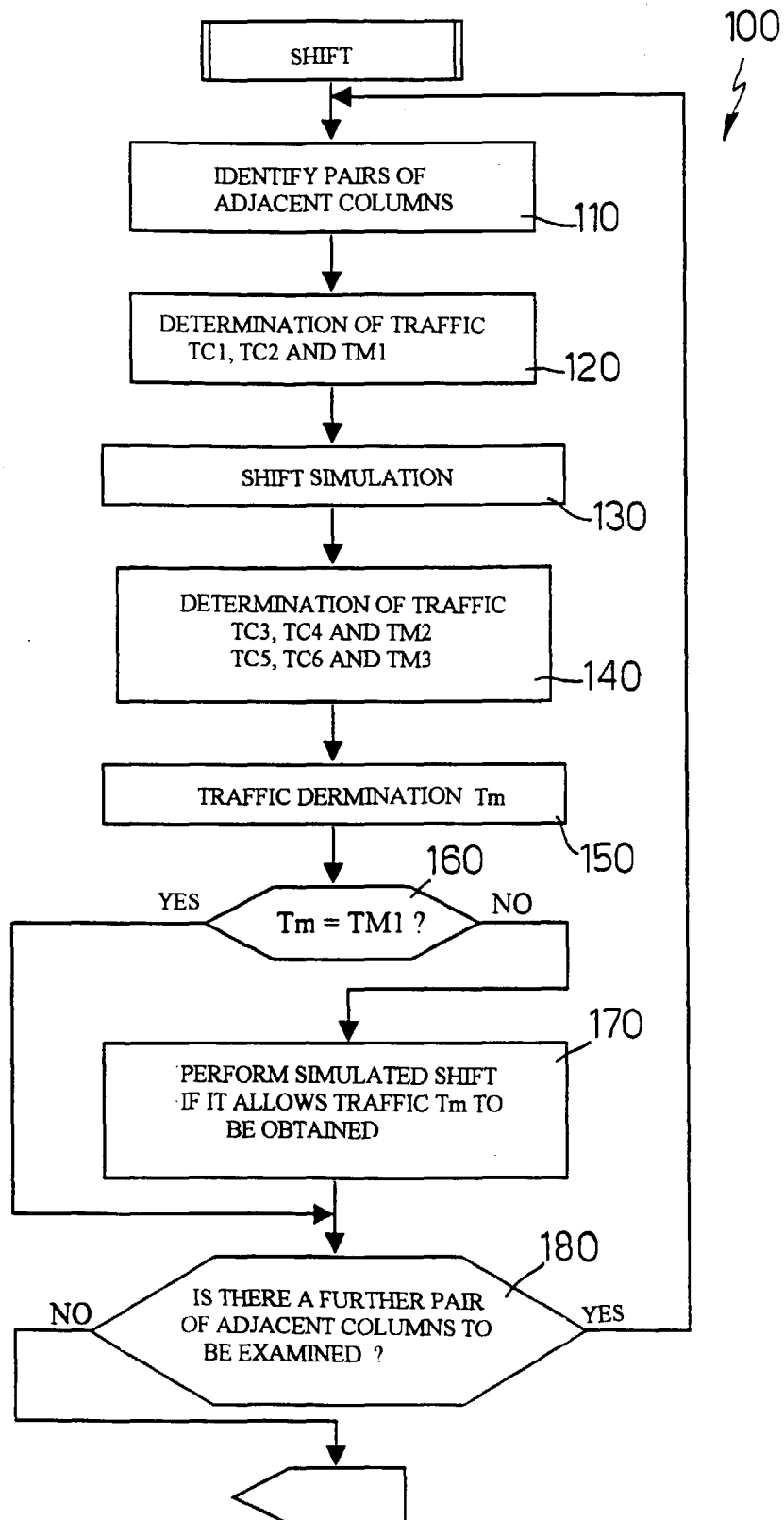


Fig. 3d

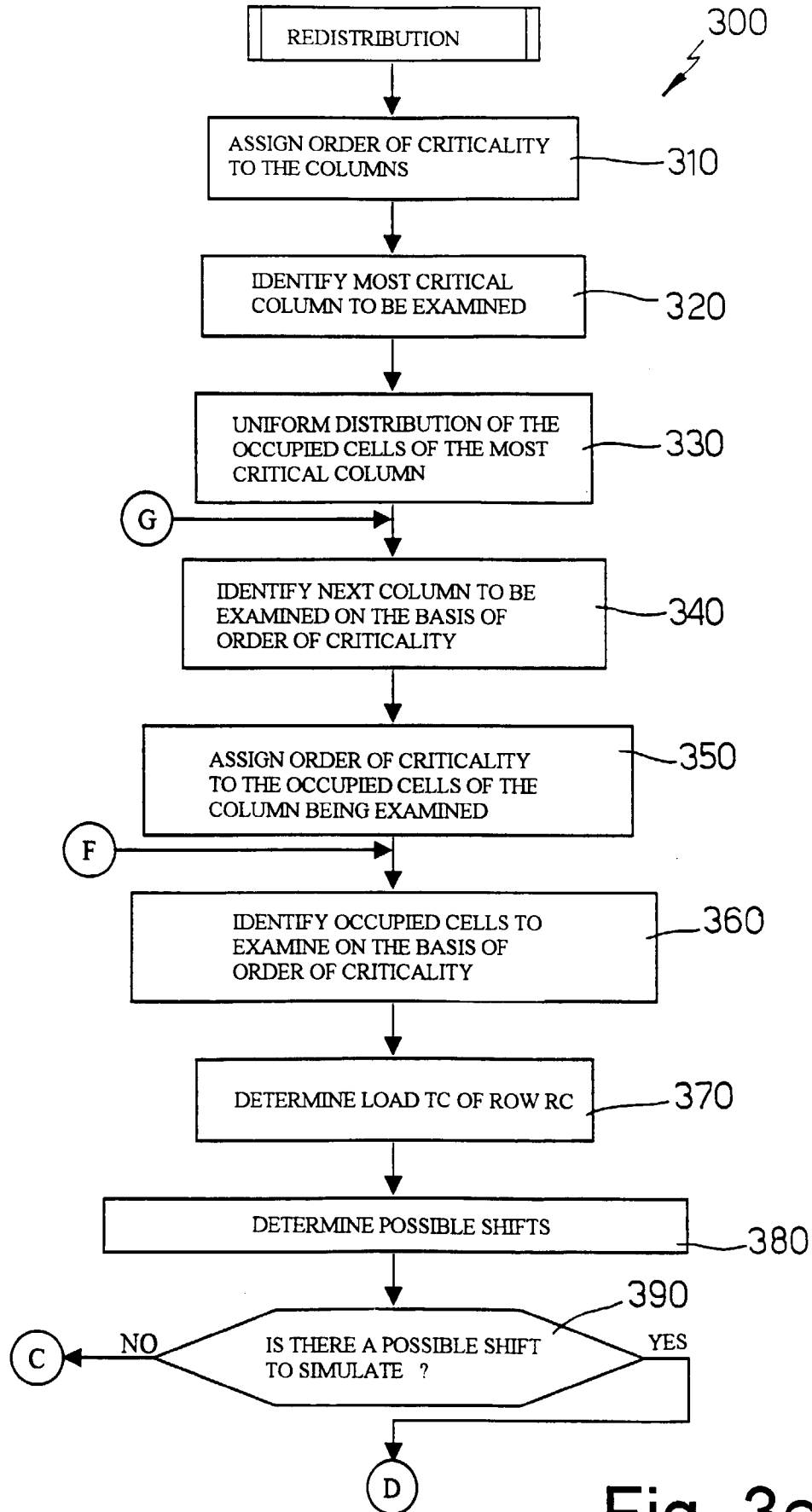


Fig. 3e

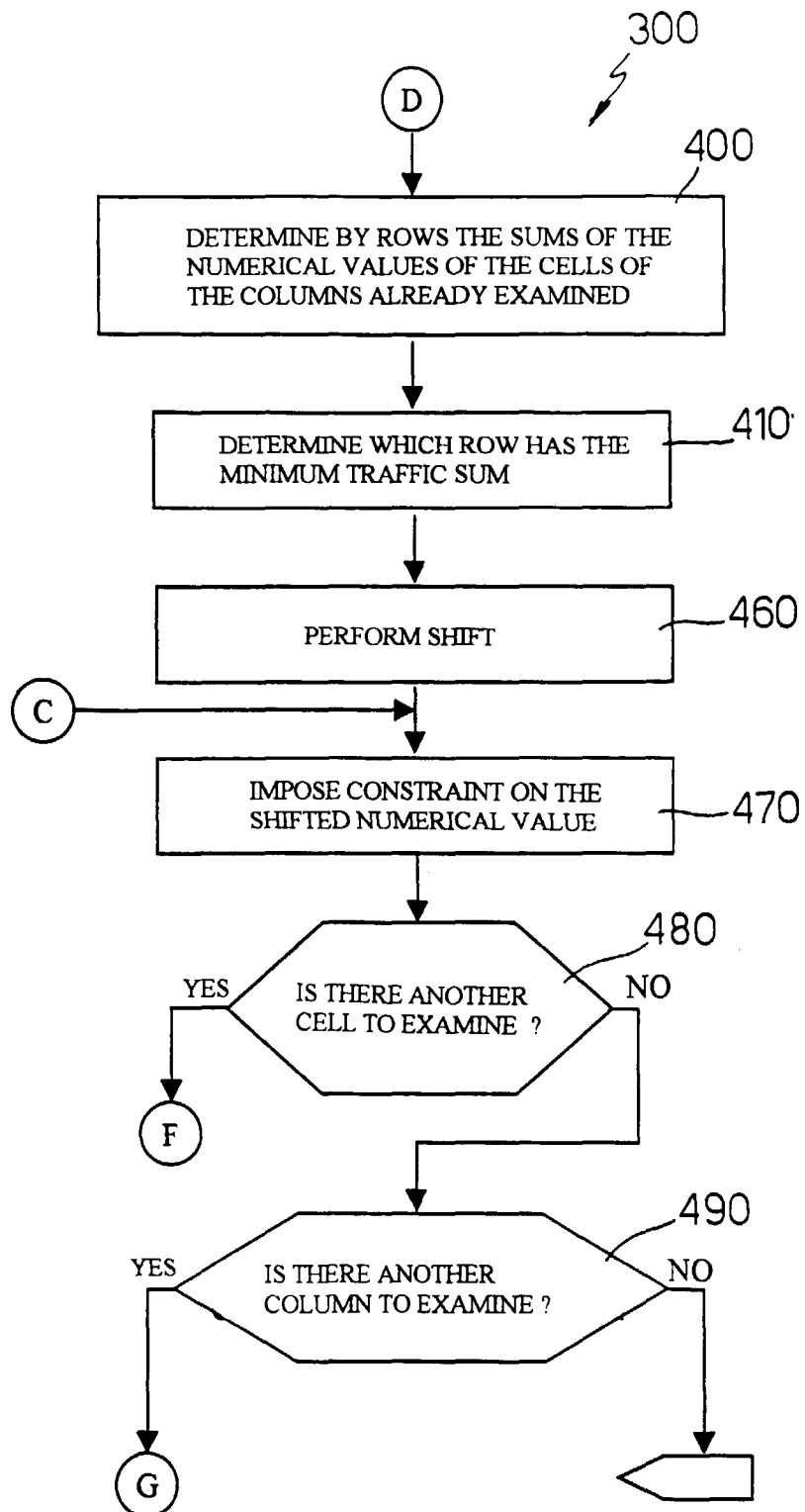
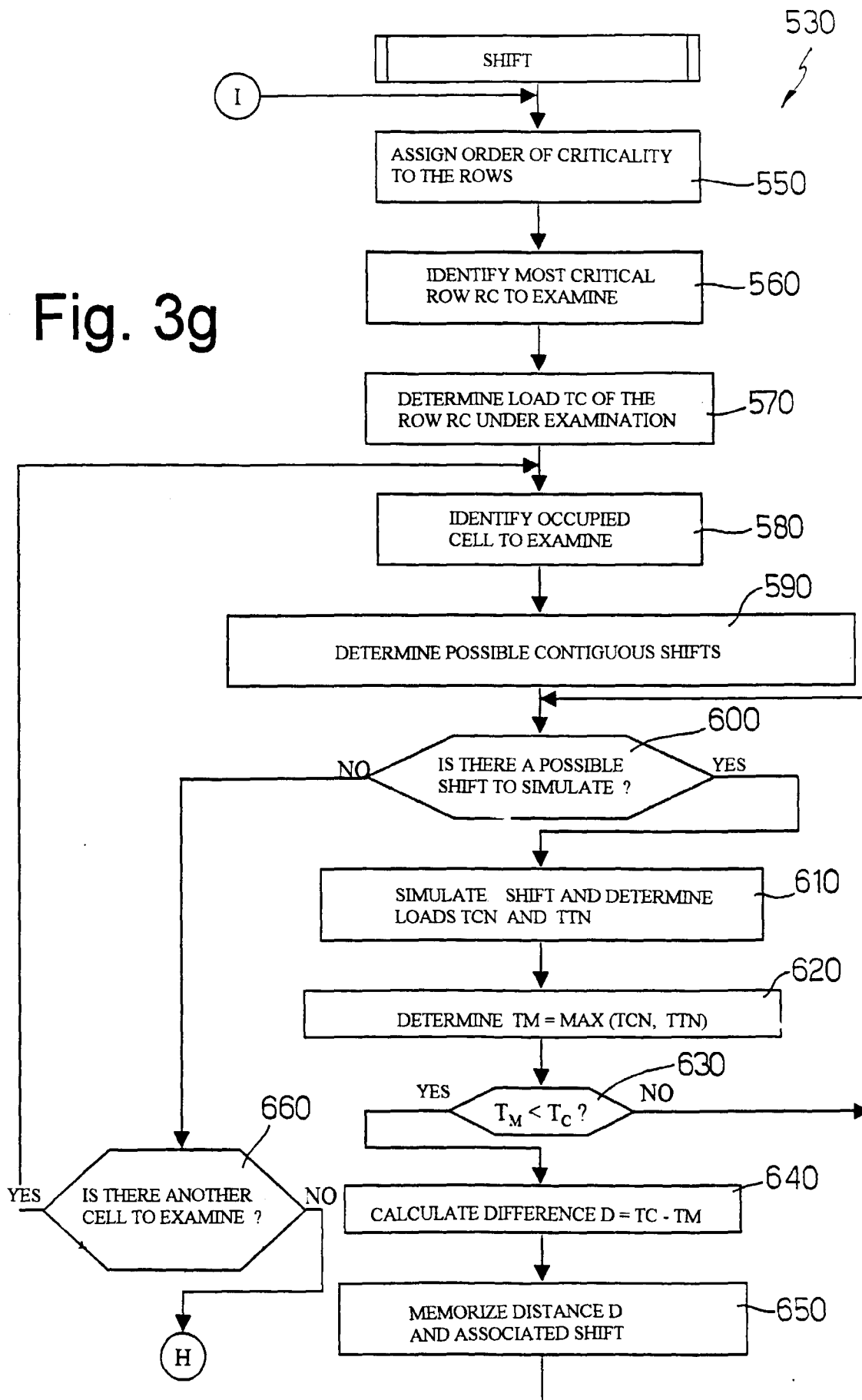


Fig. 3f

Fig. 3g



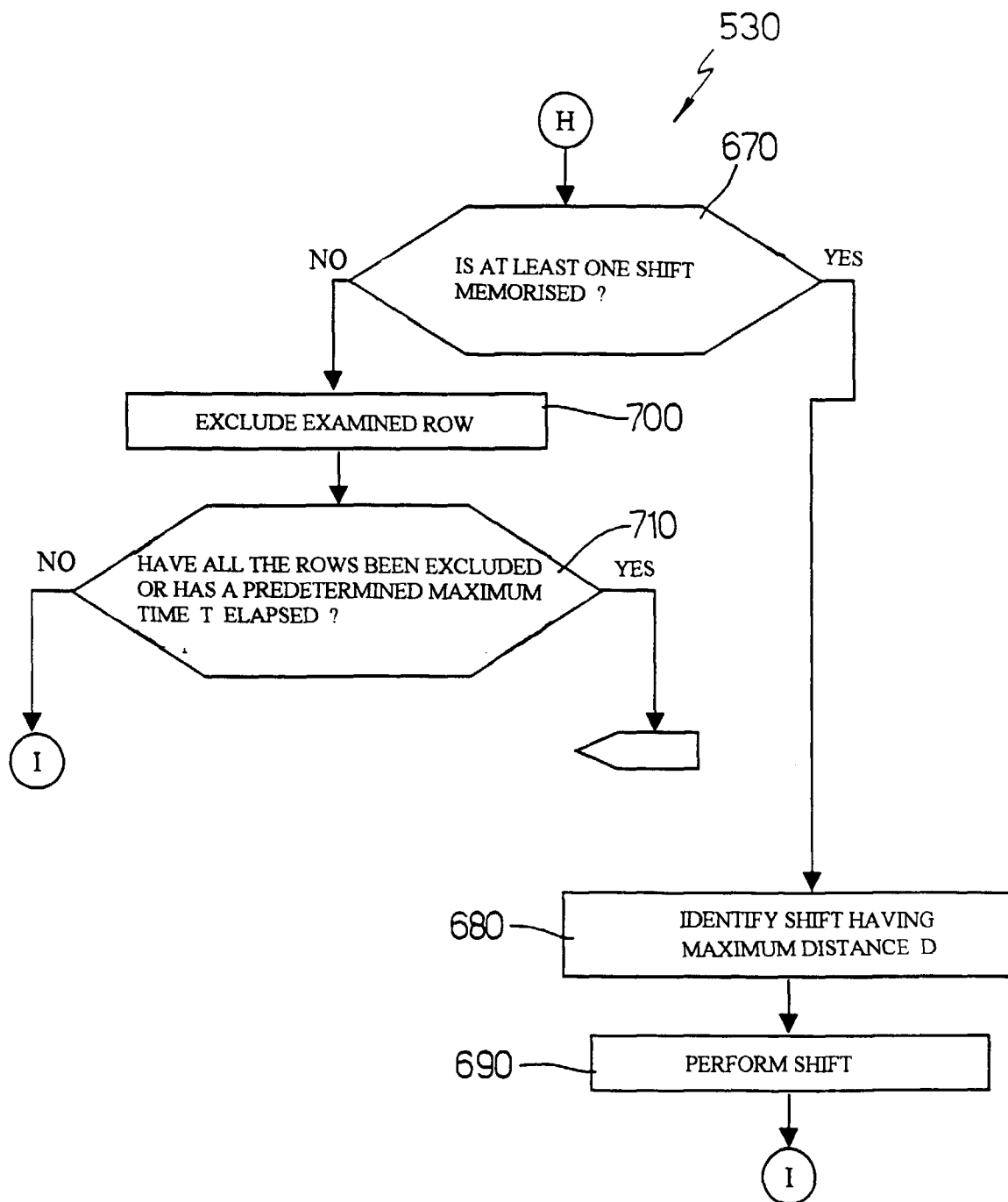


Fig. 3h

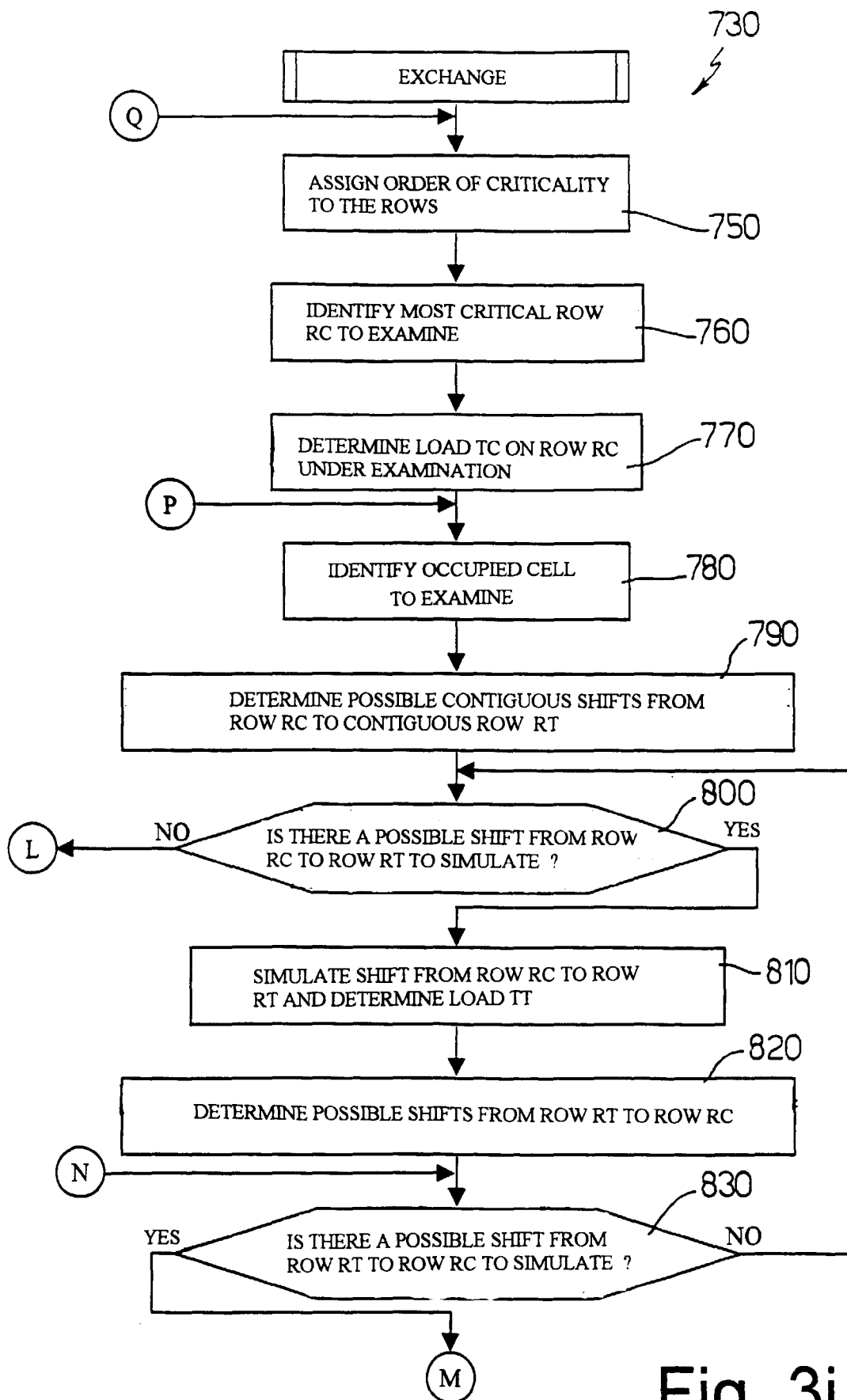


Fig. 3i

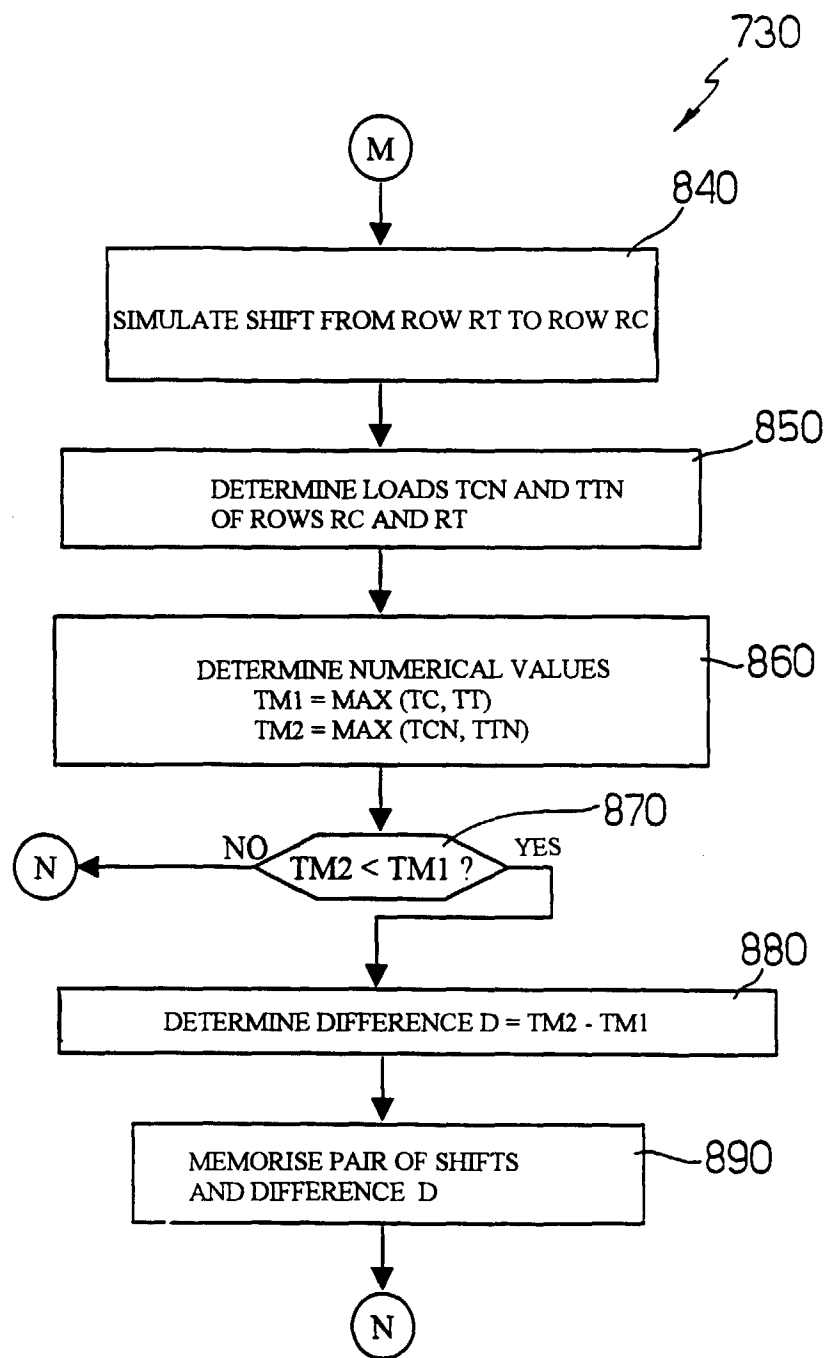


Fig. 3j

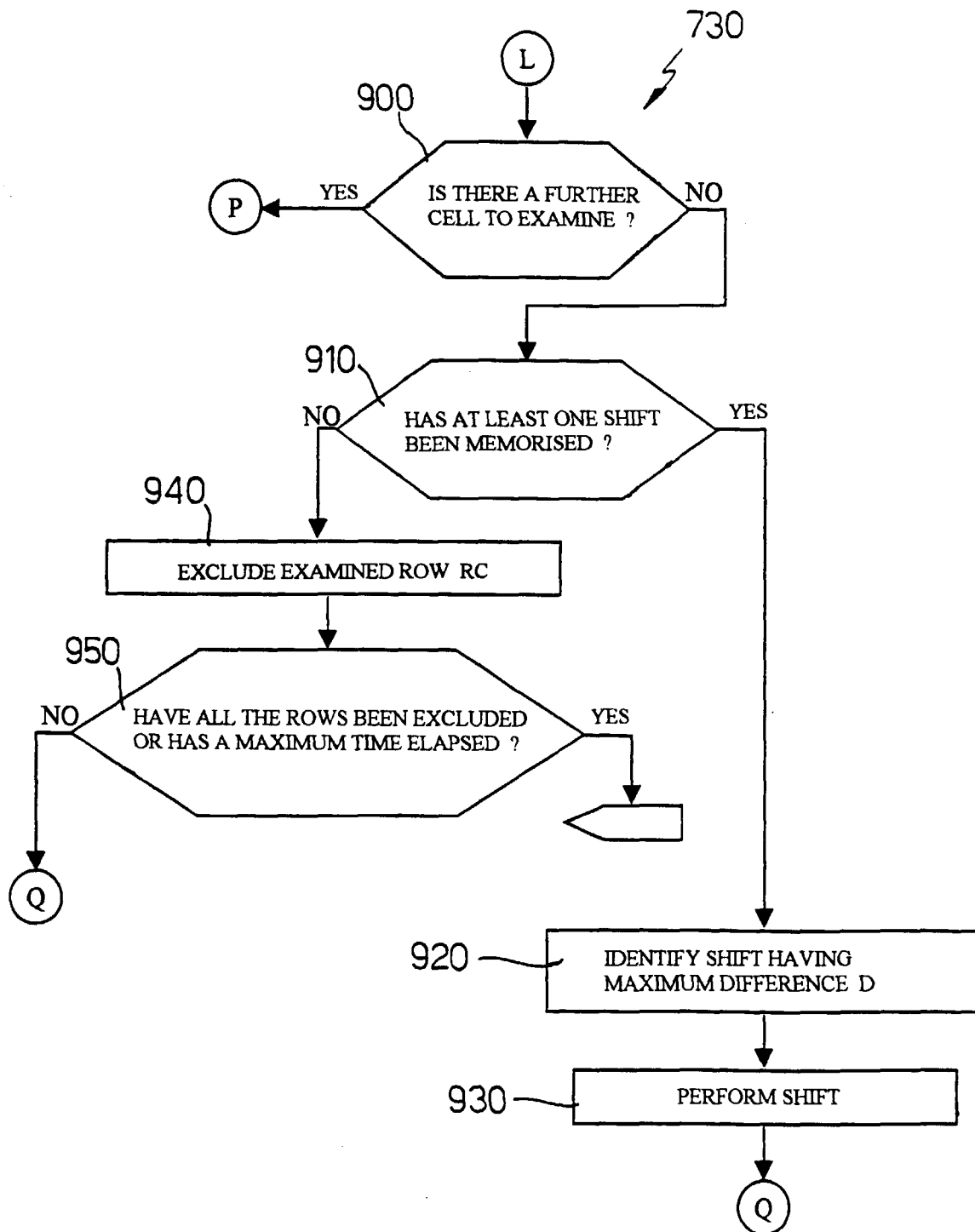


Fig. 3k