NBSIR 79-1738

EVACNET: Prototype Network Optimization Models for Building Evacuation

Richard L. Francis
Patsy B. Saunders

Operations Research Division
Center for Applied Mathematics
National Bureau of Standards
Washington, D.C. 20234

October 1979

U.S. DEPARTMENT OF COMMERCE
NATIONAL BUREAU OF STANDARDS

NBSIR 79-1738

EVACNET: PROTOTYPE NETWORK OPTIMIZATION MODELS FOR BUILDING EVACUATION

Richard L. Francis
Patsy B. Saunders

Operations Research Division
Center for Applied Mathematics
National Bureau of Standards
Washington, D.C. 20234

October 1979

U.S. DEPARTMENT OF COMMERCE, Juanita M. Kreps, ***Secretary***
Luther H. Hodges, Jr., ***Under Secretary***
Jordan J. Baruch, ***Assistant Secretary for Science and Technology***
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, ***Director***

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

The authors would like to acknowledge the information on building evacuation, flow rates, and capacities provided by J. L. Pauls of the National Research Council of Canada. We also appreciate the encouragement and support of B. H. Nelson and his staff in the NBS Center for Fire Research. Finally, we thank R. L. Paulson for her comments on earlier versions of this work.

ABSTRACT

This report presents the results of a pilot project conducted to explore how the evacuation and "evacuability" of buildings can be analyzed with the aid of mathematical network flow optimization models. As a research vehicle, Building 101, an eleven-floor building located on the Gaithersburg, Maryland campus of the National Bureau of Standards, has been studied; mathematical models pertinent to evaluating that building under a number of different circumstances have been developed and solved on the computer.

1.0 INTRODUCTION

This report presents the results of a pilot project conducted to explore how the evacuation and "evacuability" of buildings can be analyzed with the aid of mathematical network flow optimization models. As a research vehicle, Building 101, an eleven-floor building located on the Gaithersburg, Maryland campus of the National Bureau of Standards, has been studied; mathematical models pertinent to evacuating that building under a number of different circumstances have been developed and solved on the computer. The experience gained in studying network models of the evacuation of Building 101 has identified a number of research and development tasks which we believe can yield a powerful and broadly applicable methodology for the analytical study of building evacuation and its dependence on building design, operation, and codes.

We have tried in this report to make our results as accessible as possible, intentionally avoiding mathematics and relying heavily upon network drawings for motivation. In Section 2.0 we develop and explain necessary modeling ideas, using a simple three-floor model of a building for expository purposes. Section 3.0 describes Building 101 in some detail. Section 4.0 presents the results of a sequence of computer runs representing the evacuation of Building 101 under various conditions, and describes the conclusions resulting from this computational experience. Section 5.0 describes the computer codes utilized in the solution process. In that section we assume that the reader is acquainted with FORTRAN. Sample program inputs and outputs are given using, for expository purposes, the same simple three-floor building model described in Section 2.0. In the last section we draw general conclusions, identify key limitations of this modeling approach, attempt to give an overview of the modeling process we have used, and discuss the natural next steps in this applied research area. Appendix A is a graphical summary of the runs made on Building 101. Appendix B is the description of "A Simple Graphical Procedure for Estimating the Minimum Time to Evacuate the Building" which we have developed as a supplement to the rather detailed network flow models discussed elsewhere in our report. Computer listings of the programs developed during this study appear in Appendix C.

2.0 OVERVIEW OF MODEL

The purpose of this report is to present the results of a pilot project conducted to analyze the evacuation of buildings by means of computerized network flow optimization models. A major effort during this study involved constructing such an evacuation model of Building 101, an eleven-floor building located at the Gaithersburg, Maryland campus of the National Bureau of Standards. A "skeletal" network model of the building has been constructed which represents the following entities (as well as paths of movement between them): workplaces, halls, doors between workplaces and halls, stairwells, doors between halls and stairwells, doors between stairwells and the lobby, and lobby doors. The model determines by itself an evacuation routing of the people in the building so as to minimize the time to evacuate the building. Further, the model is dynamic, in the sense that it represents the pattern of the building evacuation over time. Just as one might imagine photographing an actual building evacuation using automatic time-lapse cameras which take pictures of relevant evacuation activities over regular time intervals, so the model depicts the evacuation of the building as it changes over time: time is divided into discrete time periods, and the model indicates the changes in the evacuation status during each time period, as well as the evacuation status at the end of each time period.

Data for the model include such things as the numbers of people in workplaces prior to evacuation, stairwell flow-rate capacities, hall and lobby flow-rate capacities, as well as static capacities such as the total number of people a hall, workplace, or stairwell can accommodate. By making repeated computer runs of the model with different data sets, "what if" questions of interest, such as the following, can readily be addressed:

- What if we want to determine the minimum time to evacuate the building, as well as routes people could follow so as to evacuate the building in the minimum time?
- What if there is a fire on the tenth floor?
- What if we could use "express elevators" to facilitate evacuating the building?
- What if a fire blocks a stairwell and/or some halls?
- What if we add more building exits?
- What if we add more stairwells?
- What if we want to identify evacuation bottlenecks?

The fact that the model is computerized greatly facilitates asking such "what if" questions: answering them usually entails only changing model data and then making a computer run. Such data changes can often be made by an operator sitting at a remote computer terminal. Computerization permits answering such questions quickly, and is particularly useful when the model is large enough to be unwieldy if dealt with manually. Such a computer model has clear advantages over such other approaches as the use of graphical models, pictorial representations of building evacuation, and actual trial building evacuations: the computer model is often quicker, cheaper, can handle larger problems, and greatly facilitates the comparison of many alternatives. In the long run, it is hoped that such computer models will facilitate the study of the interrelationships of building evacuation with building design, building redesign, and building evacuability, and also will lead to improvements in design for evacuation.

To provide some insight into computer network flow models, it seems appropriate to examine briefly the questions of why such models were initially developed, and what uses they have found. The initial development of network flow optimization was by Hitchcock [11] and Koopmans [13], but the development did not really take wing until work by Ford and Fulkerson [6] at the RAND corporation. Ford and Fulkerson were presented with the problem of determining the capacity (the bottleneck) of a railroad network. The model they constructed to address the problem consisted of "arcs" and "vertices", where each "vertex" represented the intersection of two or more rail lines, and each "arc" represented that portion of the rail line joining two adjacent rail intersections. Each arc had a capacity, e.g., box cars per time period, and could also have a unit cost, e.g., the cost to move a box car the length of the rail segment represented by the arc. Using their model it was possible to compute the maximum "flow" through the network, that is, the maximum number of box cars which could be sent through the network during the time period of interest. This maximum flow in turn identified network bottlenecks: arcs whose capacities placed a binding restriction upon the maximum flow.

Subsequently network flow optimization models have become of substantial economic importance [3], [8]. One generic model of interest is the so-called transshipment model, a good introductory discussion of which is given by Wagner [23]. Figure 1A depicts an example transshipment model, with known amounts available at nodes (circles) representing "origins," e.g., warehouses, and known demands for goods at nodes representing "destinations," e.g., customer locations, with arcs representing travel routes between nodes. Goods may either be shipped directly from origins to destinations (a possibility not illustrated in Figure 1A), or be transshipped via nodes representing intermediate points. Each arc of the network may have a travel cost and/or travel time, and a flow capacity. The transshipment problem is then to determine how all of the goods available at the origins should be routed through the network so as to meet demands at the destinations, and to minimize the total transportation cost (or transportation time, or some other relevant criterion) without exceeding arc capacities. A variation of the problem is to compute the maximum amount of goods which can be shipped via the network from origins to destinations. It is worth emphasizing that the important distinguishing aspect of such network problems is the incorporation of an <u>objective function</u>, such as total transportation cost, or total flow, which is to be optimized. Given the data, it often may not be difficult to determine by inspection some means of shipping goods from origins to destinations. It can be nontrivial, however, to determine an <u>optimum</u> pattern of shipments. To solve such problems a variety of optimization algorithms (well defined computational procedures) have been developed, [4], [6], [15] and computer programs [3], [8] of such algorithms are readily available to solve routinely problems having 50,000 to 100,000 arcs.

Given this account of the transshipment problem, it becomes evident that a building evacuation problem may be represented in a similar manner. As

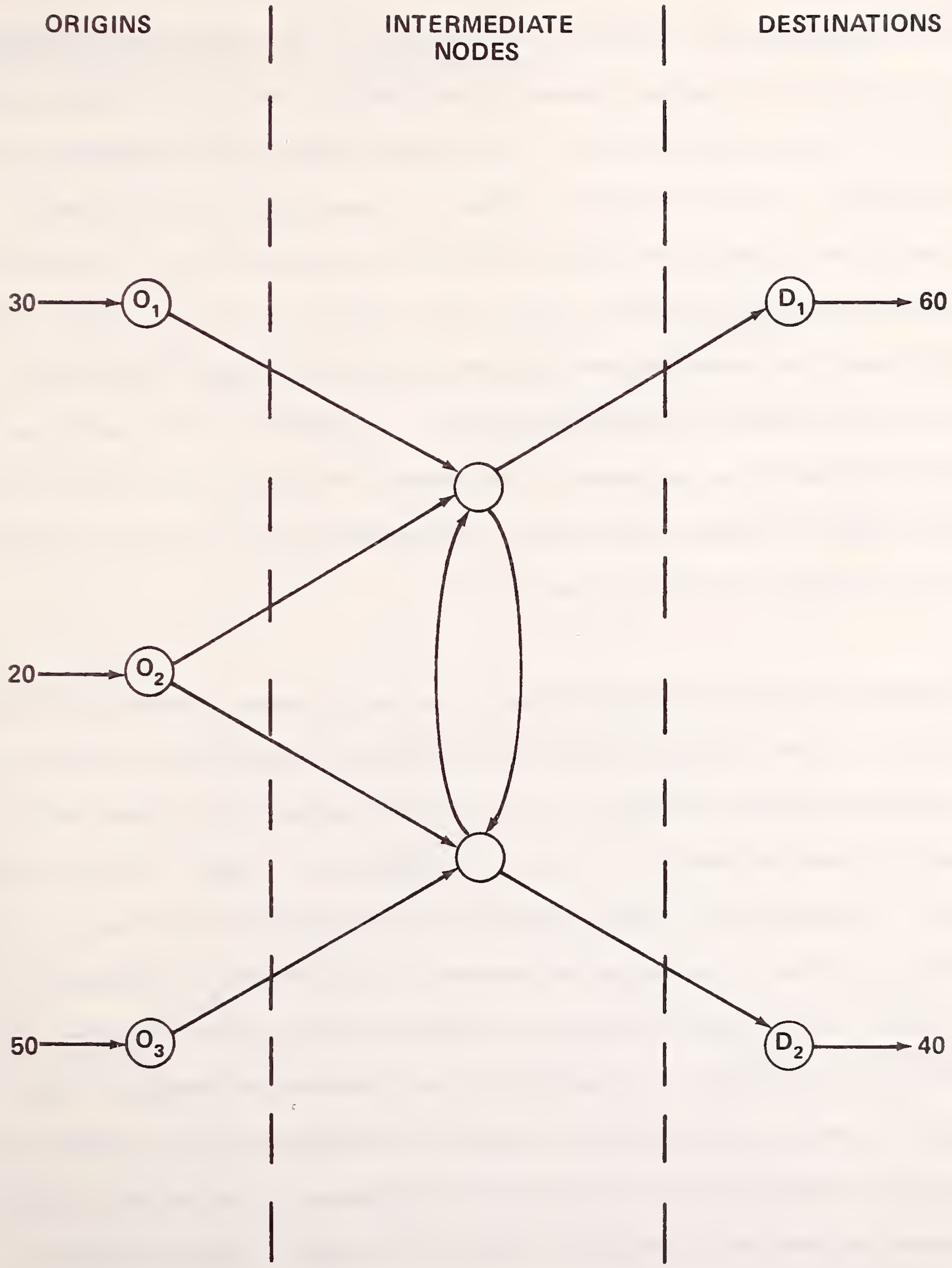


Figure 1-A EXAMPLE TRANSSHIPMENT NETWORK

Figure 1B indicates, instead of warehouses one has workplaces; instead of goods one has people; instead of destinations one has exit doors (or "safe" floors, or elevators). The building itself may be represented as a network. Data for arcs may include arc capacities, e.g., stairwell capacities in people per time period, arc travel times, e.g., the time to descend a flight of stairs; and, if appropriate, arc costs, e.g., the danger per person of being in some specific building location at a given time during the evacuation period. By means of such a model one can then determine, for example, the routes that people could take from the origins, through the building, to the destinations, so as to minimize the total time to evacuate the building.

It should be noted here that network flow optimization models are closely related to the "hydraulic models" of traditional and ongoing interest for studying building evacuation, and share the limitations of these models, such as those pointed out by Stahl and Archea [21], Stahl [22], and Pauls [18]. In particular, network models are <u>not</u> behavioral in nature: they make no attempt to describe the behavior of individuals in the event of a fire. Rather they demonstrate a course of action which, if taken, could lead to an evacuation of a building in an "appropriate" manner. Such models provide a benchmark, or standard of comparison, in the sense that they tell how quickly a building can be evacuated if the actual building evacuation pattern is the same as that of the model. One can think of such a model as recommending a desirable course of action; to be encouraged by suitable displays, instructions and training for the building occupants, etc. Behavioral models and network flow models are complementary, in that

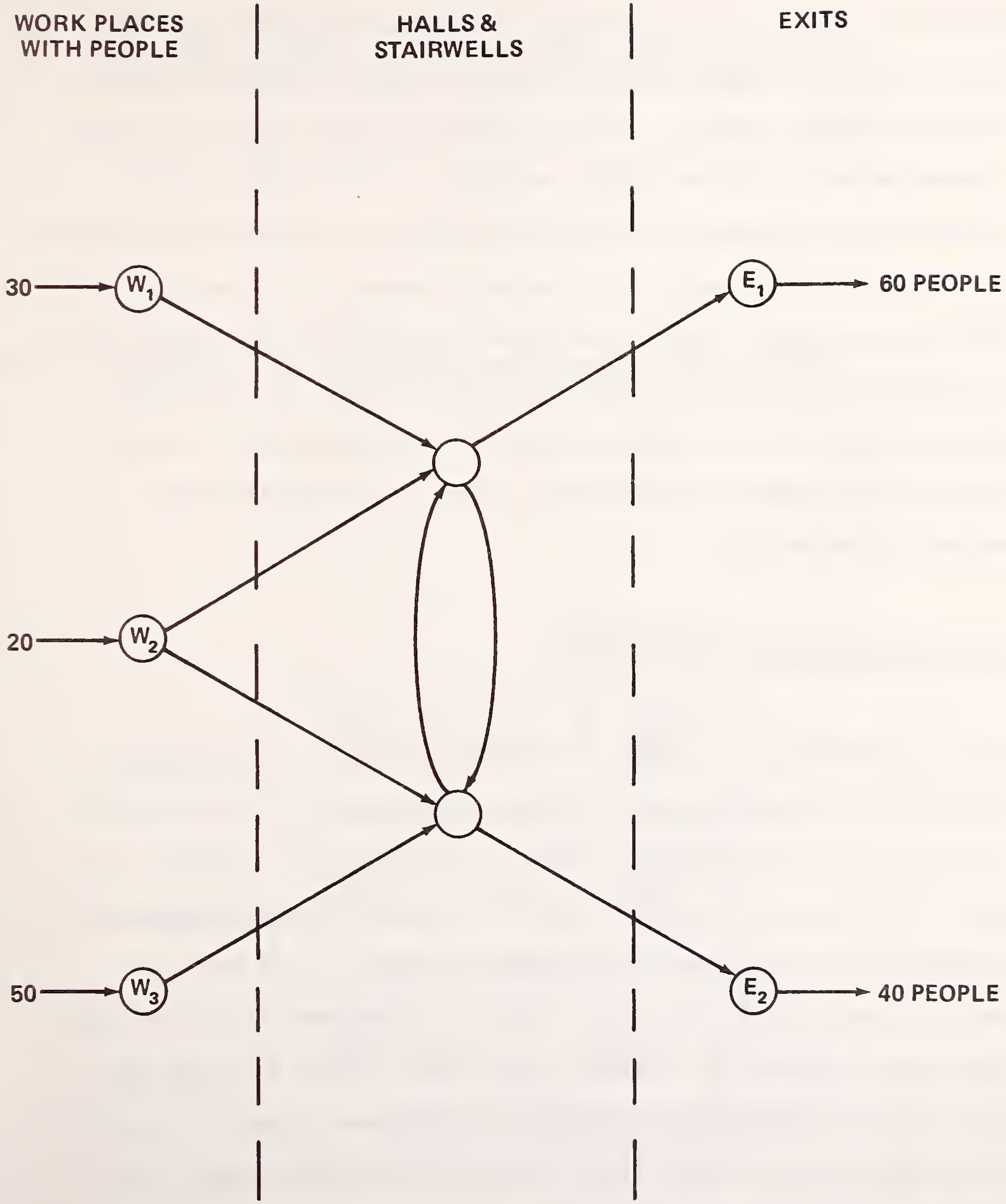


Figure 1-B **EXAMPLE BUILDING EVACUATION NETWORK**

each type of model has the potential of providing information the other cannot yield. For example, network flow models only address the question of how people might move in a building given that they decide to evacuate. To determine how long it might take occupants of a building to decide to evacuate the building once they hear an alarm, or to address the question of whether or not they will in fact decide to evacuate, one must turn to behavioral approaches. On the other hand, once people begin to move during a building evacuation, if they behave rationally, there is considerable reason to believe network models can be of use in predicting how long it will take to evacuate the building, as well as to suggest actual evacuation flow patterns.

Some Introductory Network Flow Models

Prior to discussing the modeling of Building 101, it is useful to obtain insight via some simpler models. Suppose, as an example, that 300 people are to evacuate via a stairwell, that the flow rate of the stairwell is 60 people per minute, and we wish to compute the time to evacuate the stairwell subsequent to the first person leaving the stairwell. This time, of course, is just 300/60 = 5 minutes. Figure 2 illustrates a means of interpreting this situation as a network flow problem. There are seven nodes, with each node numbered for a time period of a minute in length. (Allowing more than five time periods permits zero flows to be illustrated.) Also, there are six arcs, with the arrow-heads indicating flow directions. The "flow" in the horizontal arcs indicates the number of people remaining to be evacuated at the end of each time period, and the "flow" in the vertical arcs indicates the number of people evacuating during a time period. Thus,

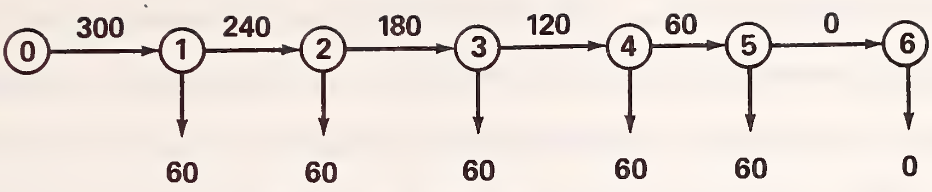


Figure 2 EXAMPLE OF NETWORK INTERPRETATION OF FLOW EQUATION

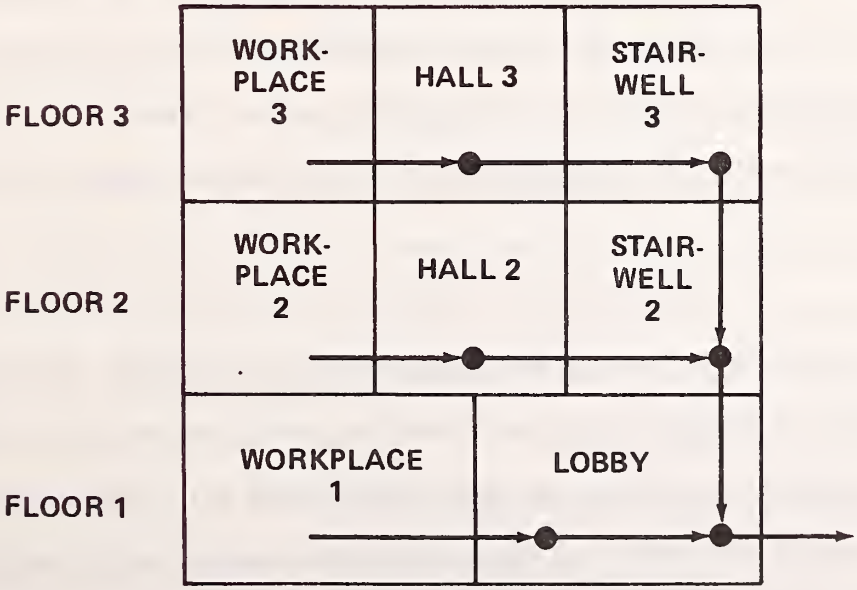


Figure 3 EXAMPLE OF FLOW PATTERN IN SIMPLE THREE FLOOR BUILDING

at the end of time period 1, 240 people remain to be evacuated, 60 people being evacuated during time period 1. At the end of time period 2, 180 people remain to be evacuated, 60 being evacuated during time period 2, etc. One can think of the problem literally as a flow problem in which the input flow is 300, the total of all the output flows is 300, and the output of 300 is to be obtained in as few time periods as possible. Note also that for each node the total flow into the node is equal to the total flow out of the node; a "conservation of flow" condition: for example the total flow into node 2 is 240, while the total flow out of node 2 is 60 + 180 = 240. Of course, for such a simple example it is unnecessary to have a network flow model in order to obtain the evacuation time result of 5 minutes; the real point of the example is to illustrate how the traditional flow computation of dividing the total number of people by the flow rate per time period may be represented as a network flow model. In addition, as will be seen, models such as this one occur as submodels ("holdover arcs") in subsequent more involved network flow models.

Pursuant to constructing such a more general model, consider Figure 3, which represents a simple three story building with one stairwell. Each floor has a workplace, each of the upper two floors has a hall, and stairwells connect the upper two floors with the lobby, which is used to exit from the building. In actuality one would expect each floor to have more than one workplace, and so the "workplaces" of the model may be viewed as composites of actual workplaces. Let us suppose a time period of ten seconds in length is chosen. Figure 4 then represents a network model of the building, with the rotation W, H, SW, and L denoting workplace, hall, stairwell, and lobby, respectively; subscripts indicate floor number. There are 20 people at workplace 1, and 16 people at each of workplaces 2 and 3; all 52 people are to leave the building via the exit. The data in Figure 4 indicate that it takes 1 time period to travel from the workplace to

the hall on floors 2 and 3, and 1 time period to travel from the workplace to the lobby on floor 1. It also takes 1 time period to travel from the hall to the stairwell on floors 2 and 3, and two time periods to descend one floor in the stairwell. The numbers in parentheses on the arcs represent upper bounds on flows per time period. At most 10 people can travel per time period from the workplace to the hall on floors 2 and 3, and at most 10 people per time period can travel from workplace 1 to the lobby. At most 8 people per time period can travel from the hall to the stairwell on floors 2 and 3, while at most 8 people per time period can travel from the lobby to the exit. The numbers in parentheses above the nodes represent upper bounds on the number of people who can be in the location represented by the node at any point in time; in other words, these numbers represent static capacities. For example, at most 20 people can ever be in workplace 3, at most 50 people can ever be in hall 3, at most 50 people can ever be in the stairwell between floors 3 and 2, at most 50 people can ever be in the stairwell between floors 2 and 1, and at most 40 people can ever be in the lobby. Thus it can be seen that the model really has two different types of capacities associated with it: while node capacities are static capacities, the arc capacities are dynamic capacities, since they are capacities per time period.

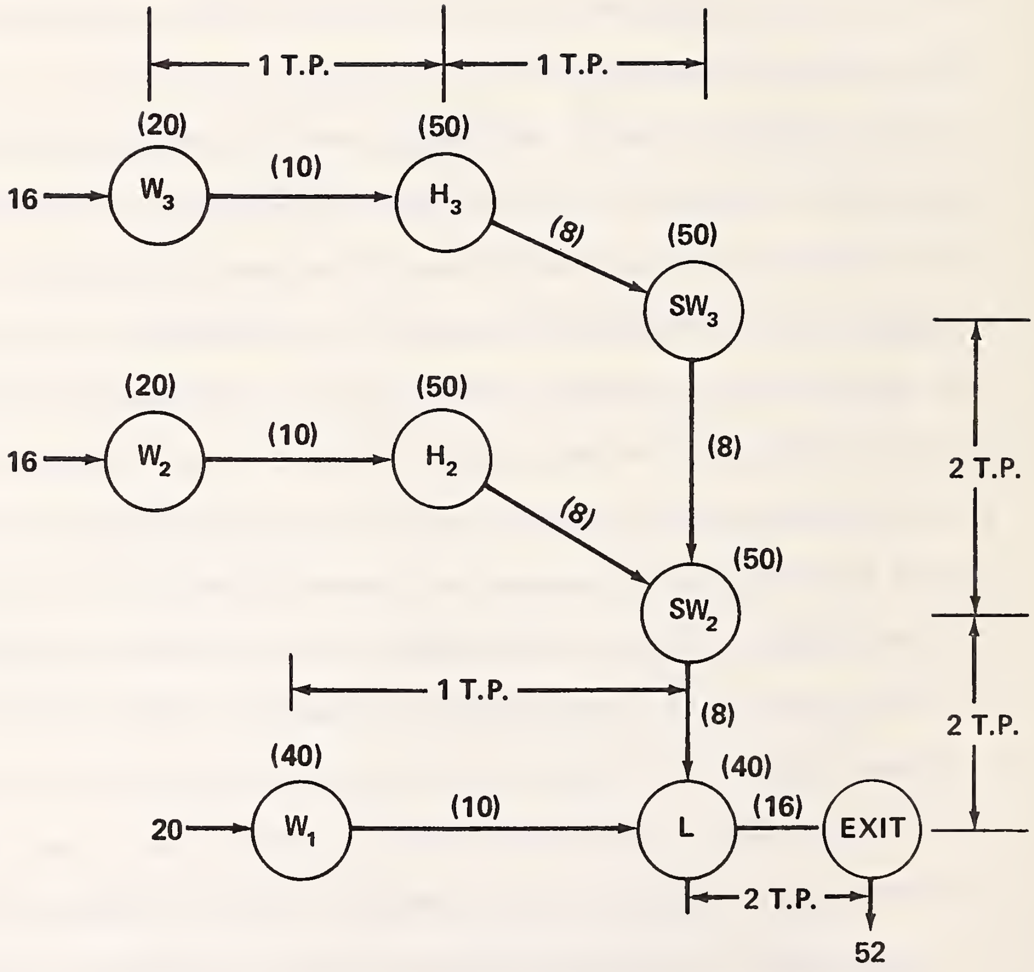


Figure 4 **STATIC NETWORK MODEL OF FLOW PATTERN IN SIMPLE THREE FLOOR BUILDING**

(W_i: WORKPLACE i. H_i: HALL i. SW_i: STAIRWELL i. L: LOBBY)

In order to obtain the dynamic network flow model of interest, the static network model of Figure 4 is expanded over time to get the dynamic network flow model shown in Figure 5. Examining Figure 5, one sees that for each node of Figure 4 there is a row of nodes in Figure 5, one per time period, with holdover arcs connecting the adjacent nodes in each row, and having the same capacities as the associated node of the static network. The holdover arcs in a row play a role much as in the first simple network flow example of Figure 2. (For example, flows in the five horizontal arcs in the top row represent the number of people remaining in workplace 3 at the end of the first, second, third, fourth, and fifth time period respectively.) The diagonal movement arcs in Figure 5 represent "copies" of arcs of the static network, and have the same capacities as the arcs they represent. (For example, the topmost set of diagonal arcs between workplace 3 and hall 3 are for "movement" flows from workplace 3 to hall 3 during periods 1 through 6 respectively.) Note that arcs between nodes representing stairwell 3 and nodes representing stairwell 2 cut across two time periods, since it takes two time periods to traverse a stairwell. Likewise, arcs between nodes representing stairwell 2 and nodes representing the lobby, and arcs between nodes representing the lobby and nodes representing the exit, cut across two time periods.

In order to obtain insight into the dynamic network, it is first helpful to consider an example where there is only one person in the building, and trace an actual route this person follows in the network from his starting point to the exit. We choose the route so as to illustrate a number of

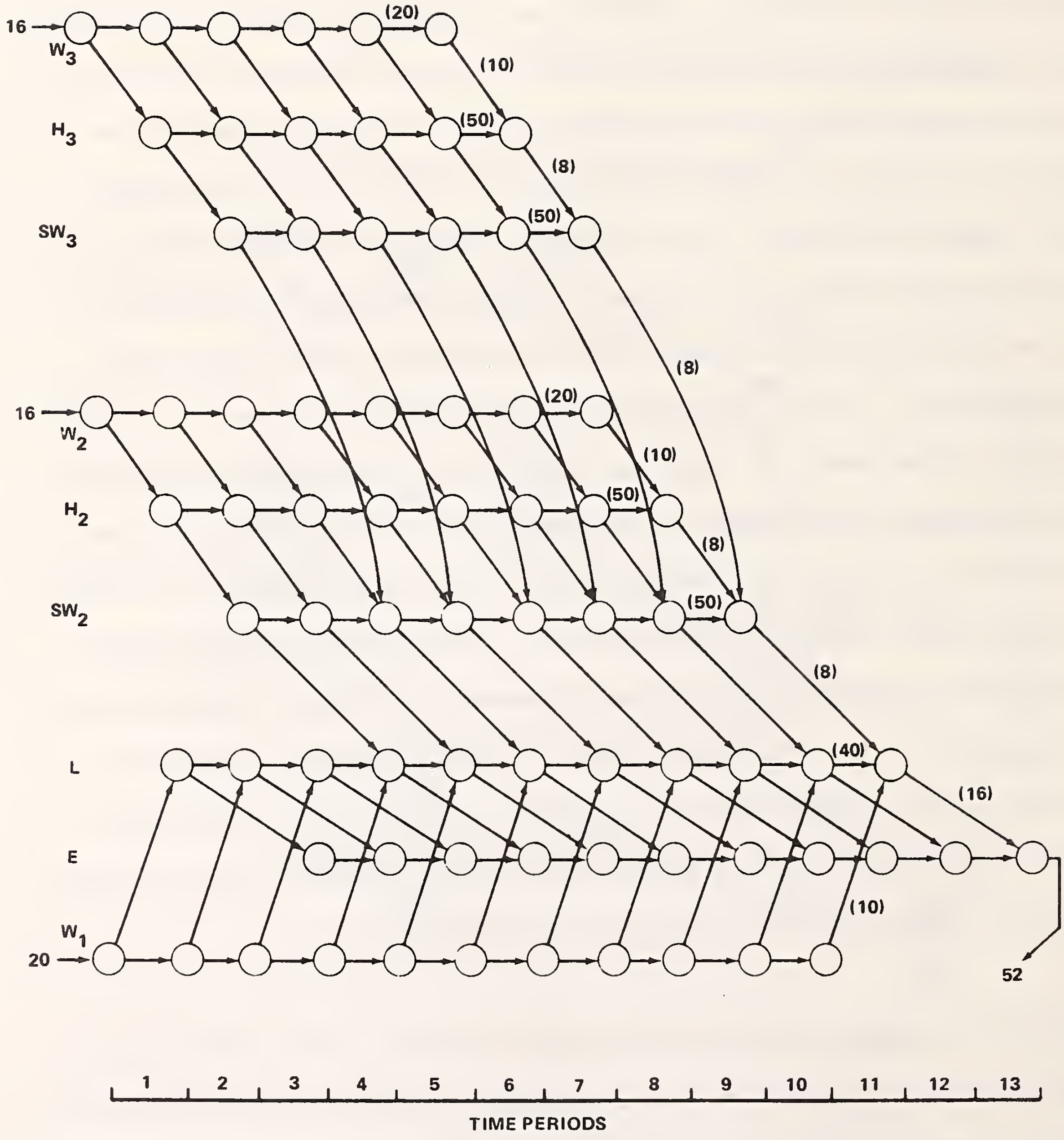


Figure 5 DYNAMIC NETWORK VERSION OF FIGURE 4 STATIC NETWORK

network features: there is no reason to think a person would actually follow such a route in evacuating the building. The dotted path in Figure 6 identifies the route followed by the person. The person, who is originally in workplace 3, remains in the workplace for periods 1 and 2, and then travels into the hall during period 3. He remains in the hall during period 4, and travels from the hall to the third floor stairwell during period 5. He then remains in the stairwell during period 6, prior to descending in the stairwell from floor 3 to floor 2 during periods 7 and 8. Next he remains in the stairwell during period 9, and then descends the stairwell from the second to the first floor during periods 10 and 11. He then travels from the lobby to the exit during periods 12 and 13. Thus by the end of period 13 the person is outside the building.

Next, we consider a more general example. Figure 7 shows an actual evacuation pattern imposed upon the network of Figure 5. Of the 16 people originally in workplace 3, 10 travel to hall 3 during period 1, while 6 remain in the workplace during period 1, and travel to the hall during period 2. Of the 10 people who travel to the hall during period 1, 8 travel to the stairwell during period 2, while 2 people remain in the hall to combine with the 6 who traveled into the hall during time period 2, giving 8 who travel from the hall to the stairwell during time period 3. Then, during periods 3 and 4, 8 people travel to the stairwell on the second floor, while 8 also travel between the two stairwells during period 4 and 5. Note that the flow pattern for the second floor is identical to that of the third floor. Observing the arcs representing travel from stairwell 2 to the lobby, note that there is a flow of 8 in each of the two-period intervals 3 and 4, 4 and 5,

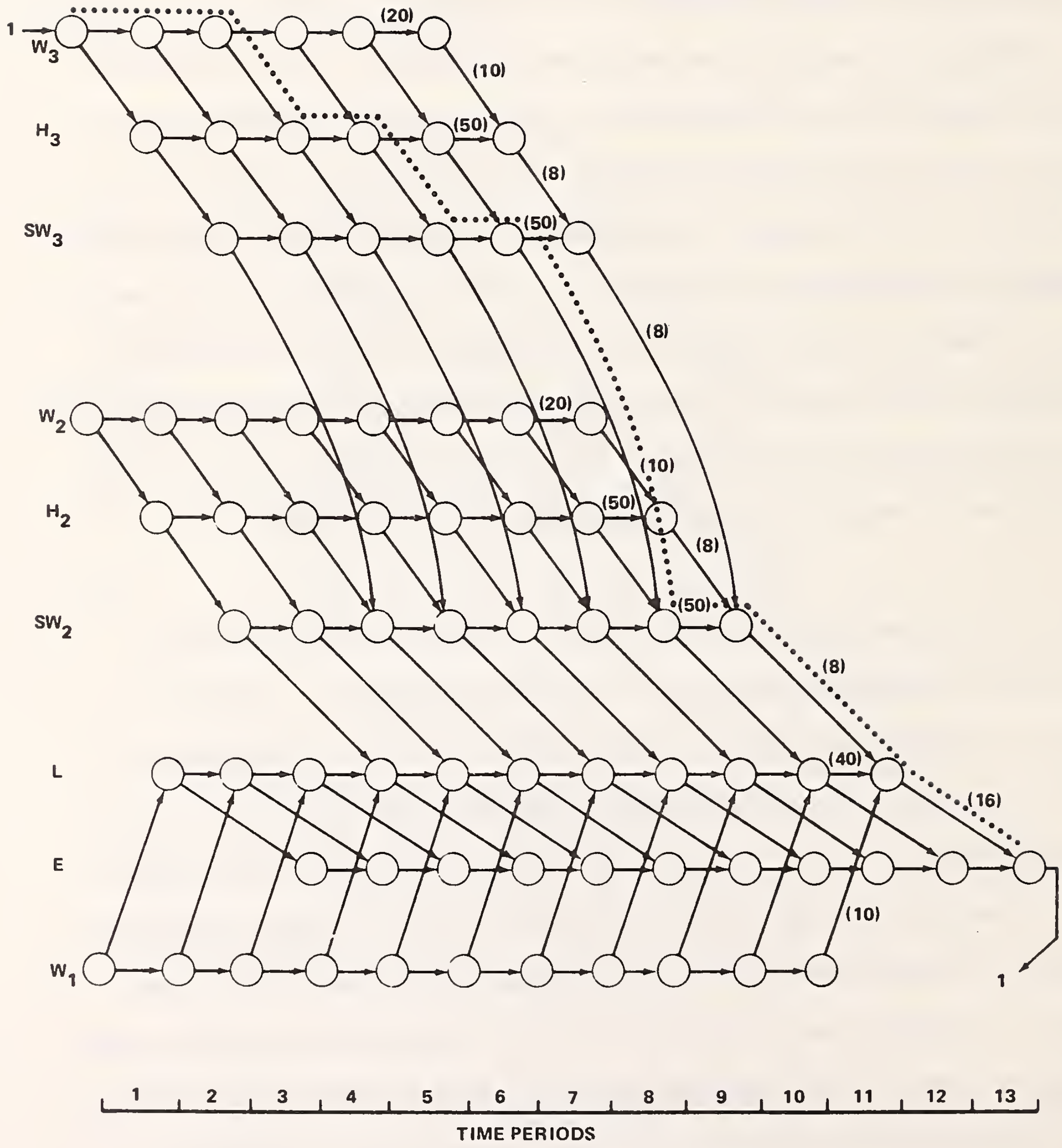


Figure 6 EXAMPLE EVACUATION ROUTE OF ONE PERSON SHOWN BY DOTTED PATH

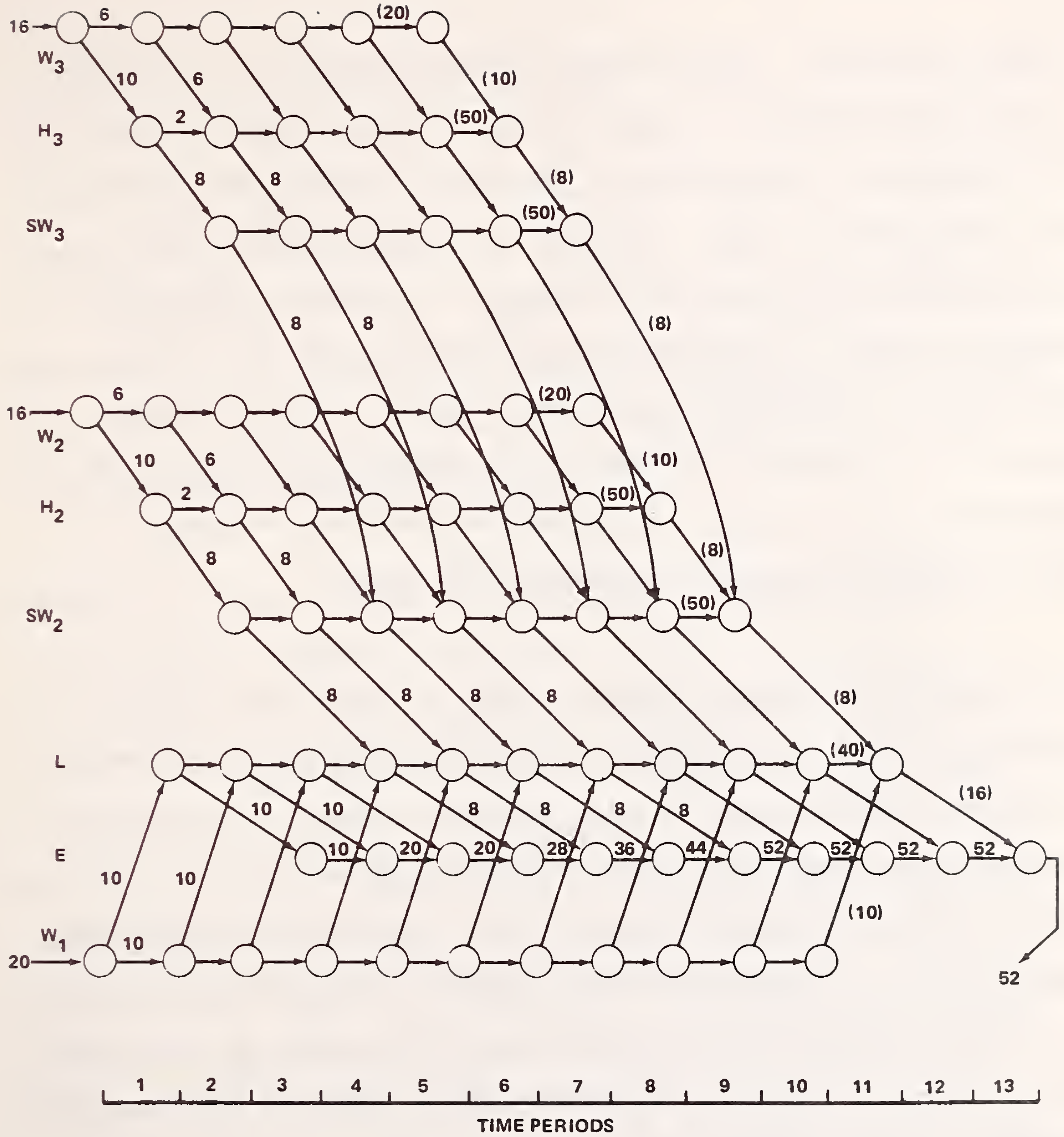


Figure 7 FIRST DYNAMIC VERSION OF FIGURE 4 STATIC NETWORK WITH COMPUTED FLOW PATTERN

5 and 6, and 6 and 7. Observing the flow from workplace 1, we find that 10 people leave workplace 1 during period 1, 10 people remain and then leave workplace 1 during period 2. The first 10 people travel to the exit during periods 2 and 3. Note that the horizontal arcs between nodes representing the exit act in effect as a turnstile, developing a cumulative count of the number of people exiting by the end of each time period: 10 people exit by the end of period 3, 20 by the end of period 4, 20 by the end of period 5 (no one exits during period 5), 28 by the end of period 6, etc. By the end of period 9 all 52 people exit, so that the building is evacuated in 90 seconds.

An examination of the flows from lobby nodes to exit nodes in Figure 7 identifies obvious "bottlenecks". One sees that arcs with capacities of 10 between workplace 1 and the lobby caused the flows of 10 between the lobby and exit nodes, while flows in arcs from stairwell nodes caused the flows of 8 from the lobby to the exit. For evacuation of the second and third floors, the stairwell flow rate capacities caused the bottlenecks, in the sense that more people could evacuate the building per time period if the stairwell flow rates were greater than 8. Figure 7 also illustrates a number of other properties of network flow models. Note that the model does not identify individual people; it only counts numbers of people. Thus, for example, of the initial 16 people in workplace 3, there is no way to tell which 6 remained in workplace 3 during period 1, and which 10 traveled to the hall. Likewise, of the 44 people who exit by the end of time period 8, the only way to identify where the 44 people come from is to "backtrack" the flows resulting in these 44 people; for this example one sees that 20 of the people come from workplace 1, 16 from workplace 2, and 8 from workplace 3. Such backtracking is simple for this small example, but can become onerous for a problem having, say, thousands of arcs. As a final note on this example, the indicated flows were in fact determined by a transshipment algorithm due to Bradley, Brown, and Graves [3]; Section 5.0 spells out the manner in which the problem is set up to be solved as a transshipment problem.

Figure 8 illustrates a variation of the problem of Figure 7, with the initial numbers of people in workplaces 1, 2, and 3 now being 35, 26, and 26 respectively. This example illustrates stairwell "merging", as well as the increased waiting flows in holdover arcs due to having more people in the building. For example, for the third node from the left for stairwell 2, 8 people arrive from stairwell 3, and merge with 8 more from the hall on floor 2; of the total of 16, 8 go to the lobby, while the other 8 remain in the stairwell. By examining the cumulative flows on the exit holdover arcs, one sees that all 87 people are not evacuated from the building until the end of time period 12, giving a building evacuation time of 120 seconds. This example problem was solved by the same algorithm as the previous example, and required only a change of 4 data cards of the input data deck used to solve the previous example.

It has been mentioned earlier that algorithms which solve the transshipment problem find a flow pattern which minimizes total cost. To this point no mention has been made of the use of arc costs in conjunction with the flow

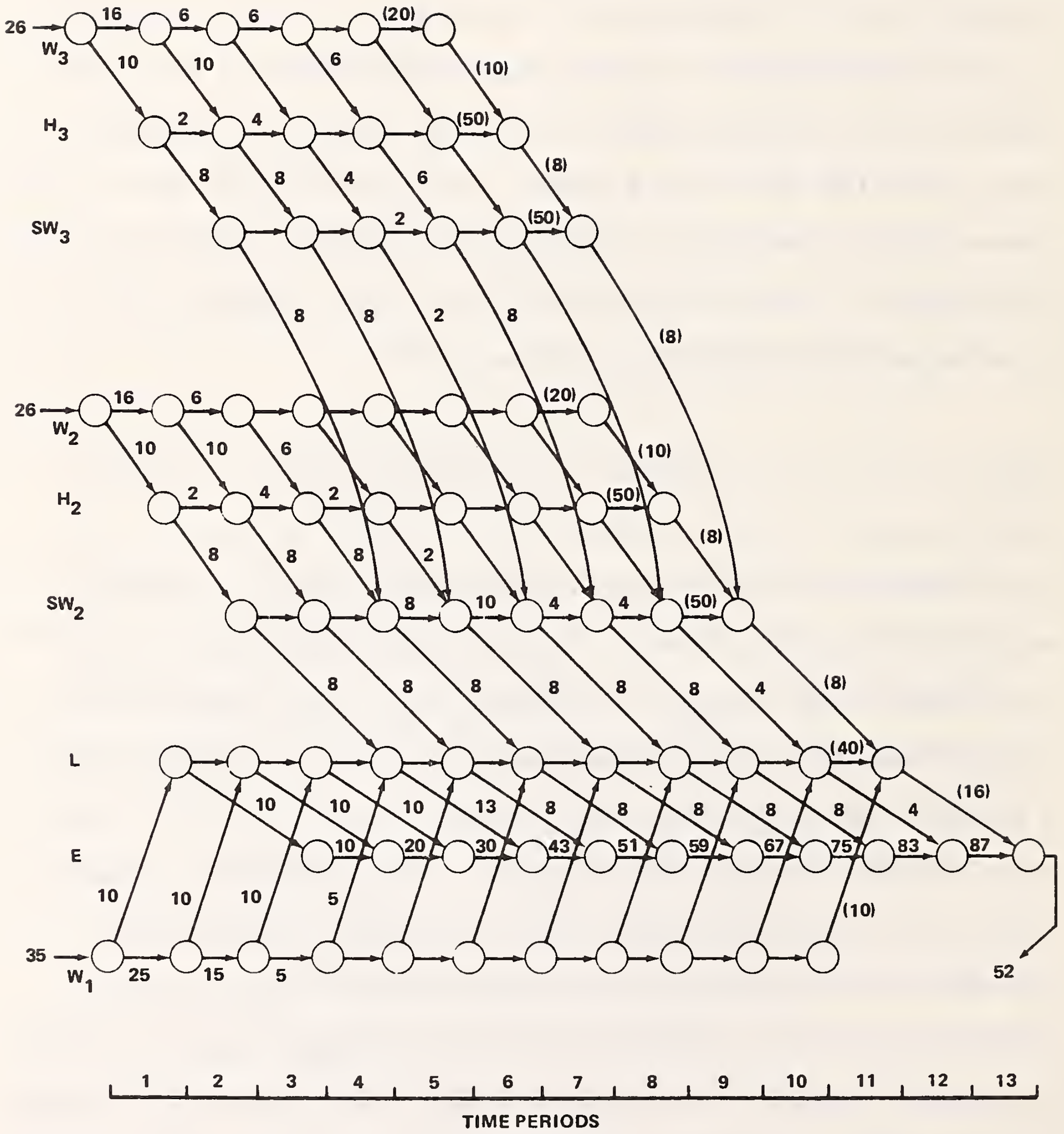
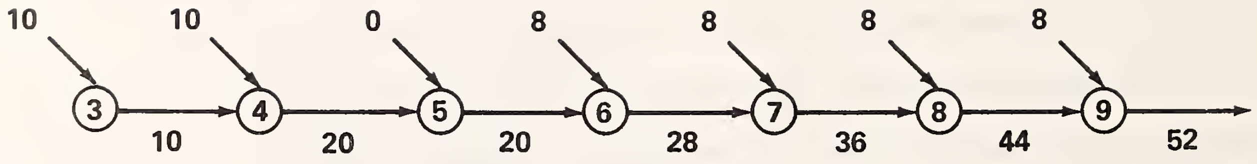


Figure 8 SECOND DYNAMIC VERSION OF FIGURE 4 STATIC NETWORK WITH COMPUTED FLOW PATTERN

patterns shown in Figures 7 and 8. A manner in which the costs may be assigned is illustrated in Figure 9, showing unit costs (costs per person) attached to diagonal movement arcs from the lobby to the exit, so that it is more expensive to exit in later time periods than in earlier time periods; the situation is exactly analogous to having a turnstile at the exit and charging people more to exit at a later period than at an earlier period. The algorithm which solves the problem determines a flow pattern that will evacuate the people so as to minimize the total amount the people would be charged. (An equivalent way to think of the turnstile feature is that people are paid more to evacuate in earlier time periods than in later time periods, and the algorithm determines a flow pattern that evacuates the people so as to maximize the total amount the people are paid.)

For the examples of Figures 7 and 8, all arc costs except turnstile arc costs are zero. Arc costs should always be zero except when there is good reason not to be. For example, if it becomes increasingly more dangerous, say, to remain in workplace 3 as time progresses, one might want to assign positive costs to holdover arcs representing workplace 3 in later time periods. As a second example, suppose it is of interest to model a situation where there is not enough time to evacuate everyone from the building: one way to do so is to construct an arc from the last holdover node copy for each hall to the last exit node copy and assign a cost to each such arc which is larger than any turnstile cost. Since the model chooses an evacuation flow pattern which minimizes total cost, it will route no more people via these arcs than is necessary. In effect, the flows in these extra arcs represent people who were trapped in the building.



TIME PERIOD	EXIT FLOW DURING PERIOD	UNIT FLOW COST	PERIOD COST	CUMULATIVE COST
3	10	6	60	60
4	10	8	80	140
5	0	10	0	0
6	8	12	96	236
7	8	14	112	348
8	8	16	128	476
9	8	18	144	620

Figure 9 EXAMPLE OF USE OF TURNSTILE

It should now be apparent that once a dynamic network model is constructed, modeling variations of the problem of interest can often be accomplished by small changes in the problem data only. For example, capacities could be changed in the dynamic network's arcs which represent replications of arcs between stairwell nodes, in order to introduce some variability in stairwell flow rates. More drastically, an arc can simply be removed. In fact, with reference to Figures 7 and 8, one can see there are more holdover arcs for floor 1 than for floor 2, and more holdover arcs for floor 2 than for floor 3. Extra holdover arcs could be included in the model for floors 2 and 3, but would be of no use in the example problems solved, as having extra holdover arcs would not permit quicker building evacuation. One might also want to omit arcs to preclude certain flows after specified time periods. For example, if a network model of a building includes two stairwells, it is easy to "shut down" a stairwell part way through an evacuation being modeled, in order to see what the effect would be on the building evacuation: alternatively, one stairwell capacity could be made less than the other to represent a situation where use of a stairwell by firemen causes the stairwell to be partially blocked.

As will be seen subsequently, to incorporate additional features into network models of building evacuation is often quite direct. There can be multiple stairwells, multiple workplaces, multiple halls, elevators (provided they run on fixed schedules which are multiples of the length of a time period), and multiple exits. Further, it is not essential (although it is simplest) to have every time period be of the same duration.

A final point to keep in mind about dynamic network models is that to model a large building for a substantial number of time periods the use of a computer is essential. For example, the eleven floor, fifty-eight time period model of Building 101 to be discussed subsequently has 5,543 arcs and 2,591 nodes: the dynamic network flow models discussed to this point are really quite small by comparison to the Building 101 model.

At this point it seems useful to formalize some of the preceding notions, and present general procedures for constructing both static and dynamic network flow models for building evacuation.

STATIC MODEL CONSTRUCTION

The static model is a direct network representation of the building, consisting of nodes (represented by circles) and directed arcs (represented by arrows) from one node to another. Nodes in the static network represent building locations of interest, such as workplaces, halls, stairwells, lobbies, and exits. Each node has a static capacity; the maximum number of people ever allowable at the location the node represents. Nodes are partitioned into three mutually exclusive and exhaustive classes: origins, intermediate nodes, and destinations. Origins represent sources of people to be evacuated, with the sources containing (known) specific numbers of people at the initiation of the evacuation activity being modeled. An origin node is identified by having exactly one arc pointing into the node, with the input into the arc being the number of people at the origin to be evacuated. As an example, with reference to Figure 5, the origin nodes are W_1, W_2, and W_3. An intermediate node is a node which is neither an origin nor a destination; each intermediate node has at least one arc from another node leading into it, and at least one arc leading out to another node. A destination node represents a location to which people are to be evacuated, and may be thought of as an exit. In the event there is only one exit, this exit is, of course, the destination, and should have exactly one arc pointing out of it, whose flow is the total number of people available at all the origins; Figure 4 illustrates a static model with only one destination. In the event there is more than one exit and a specified number of people is to evacuate via each exit, then each exit node becomes a destination and has an arc pointing out of it whose output flow is the number of people who are to use the exit: the total number of people evacuating via all the destinations must be equal to the total number of people coming from the origins. In the event there is more than one exit and the number of people to use each exit need not be specified, construct one additional node, which may be thought of as a "super destination," run an arc from each exit node to the super destination, and construct an arc pointing out of the super destination whose flow is the total number of people coming from all the destinations. Figures 10 and 11 illustrate the use of exit nodes with specified outputs, and the use of a super destination, respectively. Note that Figures 10 and 11 illustrate models of buildings with two stairwells. These models can easily be extended to represent buildings with more than three floors simply by replicating the model structure for floor 3 and by specifying the appropriate number of people for each additional floor.

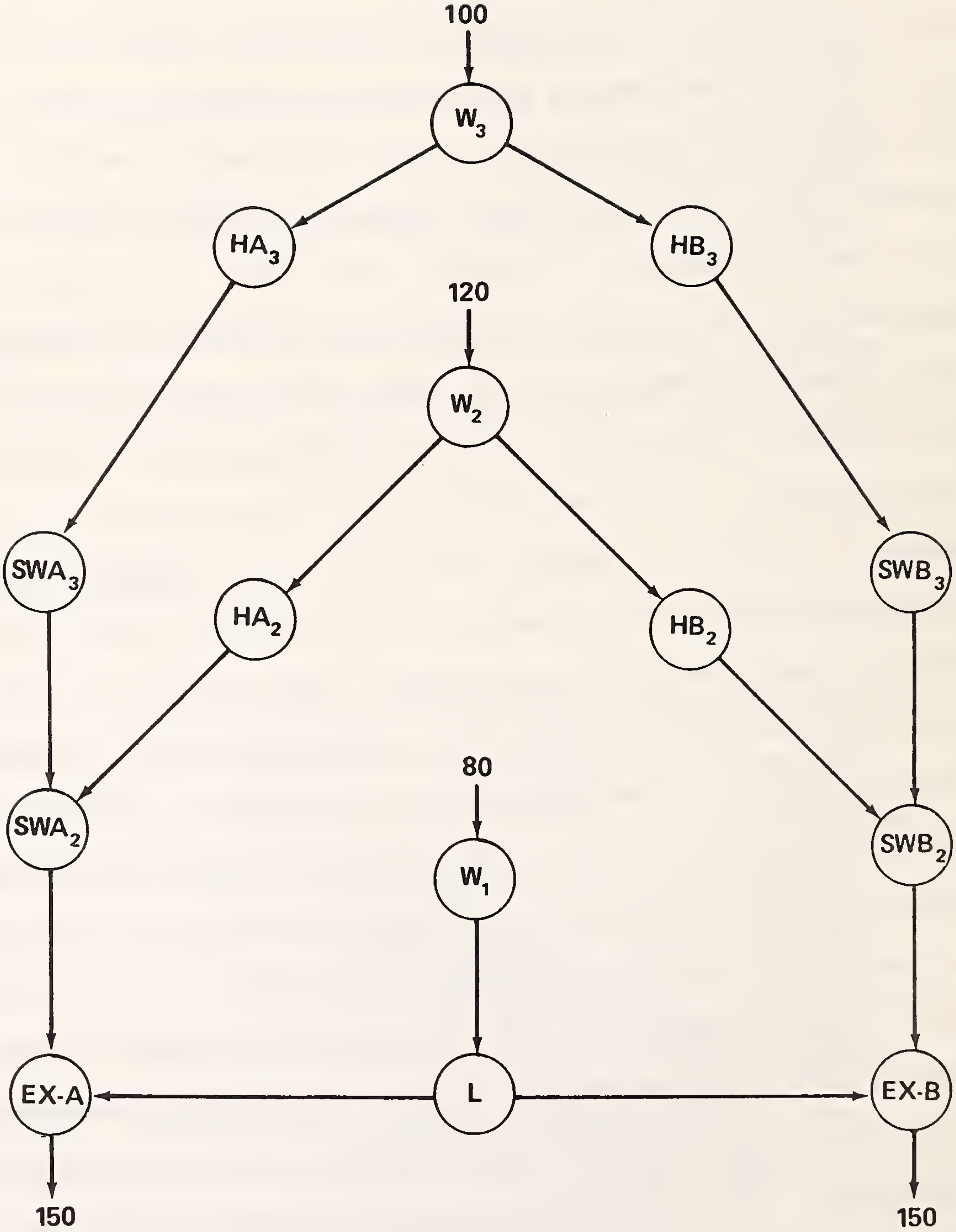


Figure 10 EXAMPLE STATIC MODEL WITH TWO STAIRWELLS & EXITS, & SPECIFIED EXIT OUTPUTS

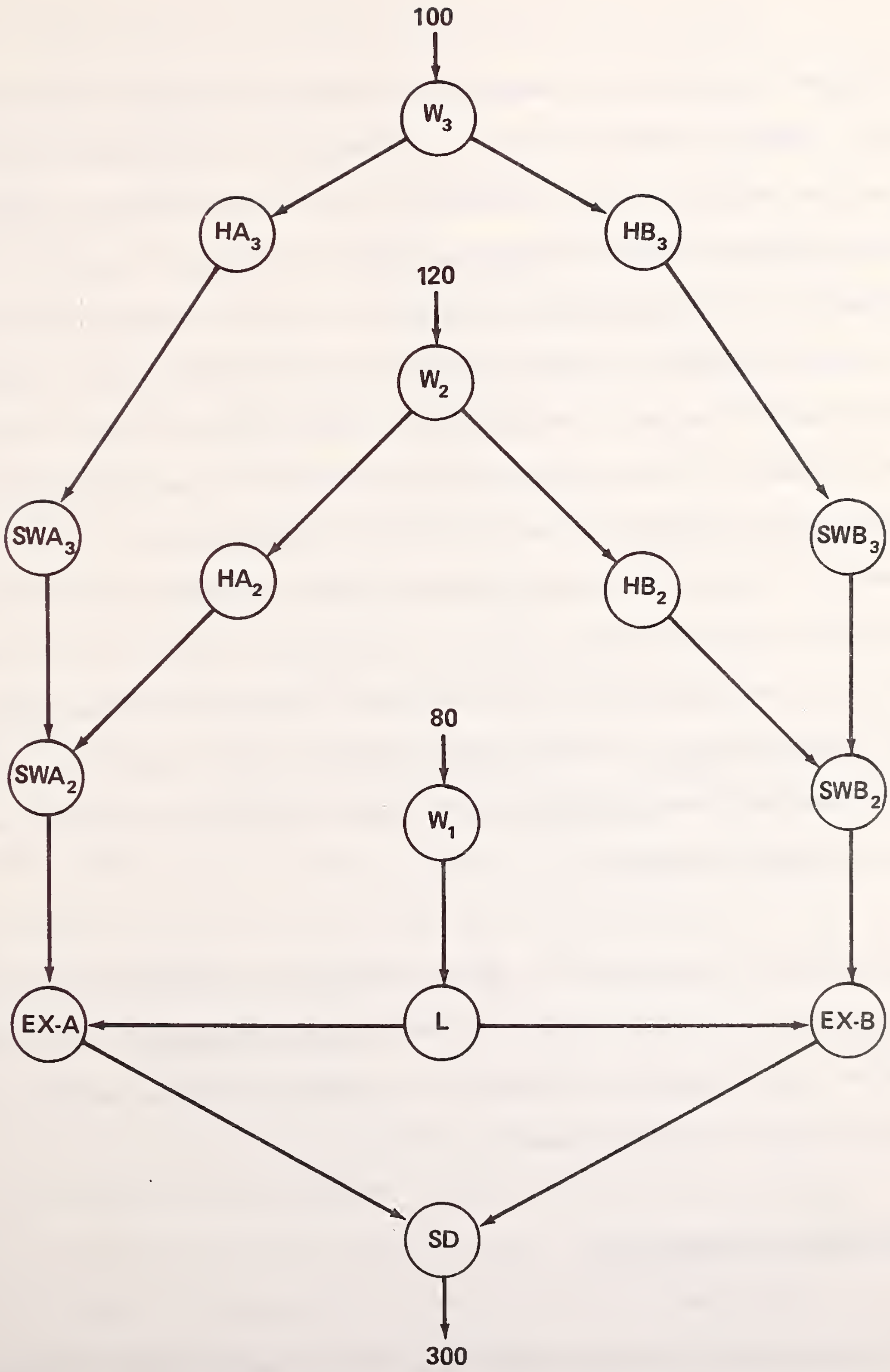


Figure 11 EXAMPLE STATIC MODEL WITH TWO STAIRWELLS & EXITS, & SUPER DESTINATION

A directed arc leads from one node to another node whenever it is reasonable to permit movement from the first location to the second, in the direction indicated by the arrow on the arc. Each directed arc represents one-way movement, and is entirely analogous to a one-way street. (It is permissible, as illustrated in Figure 2, to have two one-way arcs between two nodes, with the two arcs having opposite directions; such a situation is analogous to having a two-way street with one lane for each direction.) Each arc has the following data associated with it: the two nodes it joins; the arc traversal time (the number of time periods it takes one person to traverse the distance between the two locations the two nodes represent); the arc dynamic capacity (the number of people who can traverse the arc in one time period); the arc cost. When no cost is associated with traversing an arc, the arc cost should be zero. Generally, arc costs in the static network should be zero, unless they have a very real and direct physical meaning.

Other than zero arc costs, all the data associated with the arcs, as well as the static node capacities, should be positive integers. This condition that the data be positive integers is a requirement of the computer program used to solve the dynamic model.

DYNAMIC MODEL CONSTRUCTION

The crucial step in constructing a network flow model which is solvable as a transshipment model is the construction of the dynamic network model from the static network model.

Suppose the dynamic model is to have T time periods, where T would normally be chosen so that everyone could be evacuated from the building in T time periods. For each static node, say node s, construct T + 1 copies of node s, placed in a row and numbered 0, 1, ..., T - 1, T consecutively from left to right. Between any two adjacent node copies of node s, numbered say j and j + 1, construct a holdover arc from copy j to copy j + 1 whose capacity is the same as that of the static node s. For each static node which is an origin, let node copy 0 of the static node also be an origin, with the same flow input. For each static node which is a destination, let node copy T of the static node also be a destination, with the same arc flow output.

For each static directed arc, say from static node i to static node k, and having traversal time p (a positive integer), and for every integer t between 0 and T - p, construct a (directed) movement arc copy in the dynamic network which runs from copy t of node i to copy t + p of node k. As an example, if the static arc A runs from node 12 to node 30 in the static network, and has a traversal time of p = 2, and if T = 100, the movement-arc copies of A would run from copy 0 of node 12 to copy 2 of node 30, copy 1 of node 12 to copy 3 of node 30, ..., copy 97 of node 12 to copy 99 of node 30, and copy 98 of node 12 to copy 100 of node 30. The capacity of each movement arc is the same as the capacity of the arc it copies in the static network (except for the case where it is intended to permit the capacities of the arc copies to change over time). The costs of movement arcs should normally be zero, except for turnstile arcs, or when flow is to be precluded in an arc by assigning the arc a prohibitively high arc cost.

Once all of the arcs of the dynamic model have been constructed, it is easy to use the copy numbers of the nodes at the beginning and end of each arc to identify the time period(s) during which there can be a flow in the arc. For example, if a holdover arc is from copy t of node s to copy t + 1 of node s, whatever flow there is in the holdover arc occurs during period t + 1. If a movement arc is from copy t of one node to copy t + p of another node, whatever flow there is in the movement arc occurs during periods t + 1 through t + p.

After the dynamic model is constructed, costs may be assigned to turnstiles, as illustrated previously, in such a way as to minimize the total time to evacuate the building. In situations where there is more than one turnstile, care should be taken that all costs assigned during any given period to turnstiles are the same, regardless of the choice of the turnstile. Otherwise the "cheaper" turnstiles will receive disproportionately high use.

Figure 12 illustrates the construction of a dynamic network model from the static network model of Figure 4. With reference to Figure 12, it is convenient to refer to the arcs with slashes through them as "inessential" arcs. Such arcs are inessential in the sense that they may be deleted without changing the model, as either no flow can be put into an inessential arc, or any flow put into an inessential arc can never reach a destination. Note that deleting the inessential arcs from the network of Figure 12 gives the network of Figure 5: it should be clear that the two networks are equivalent, in the sense that they both have the same set of possible evacuation flow patterns. While inessential arcs may be deleted, if a

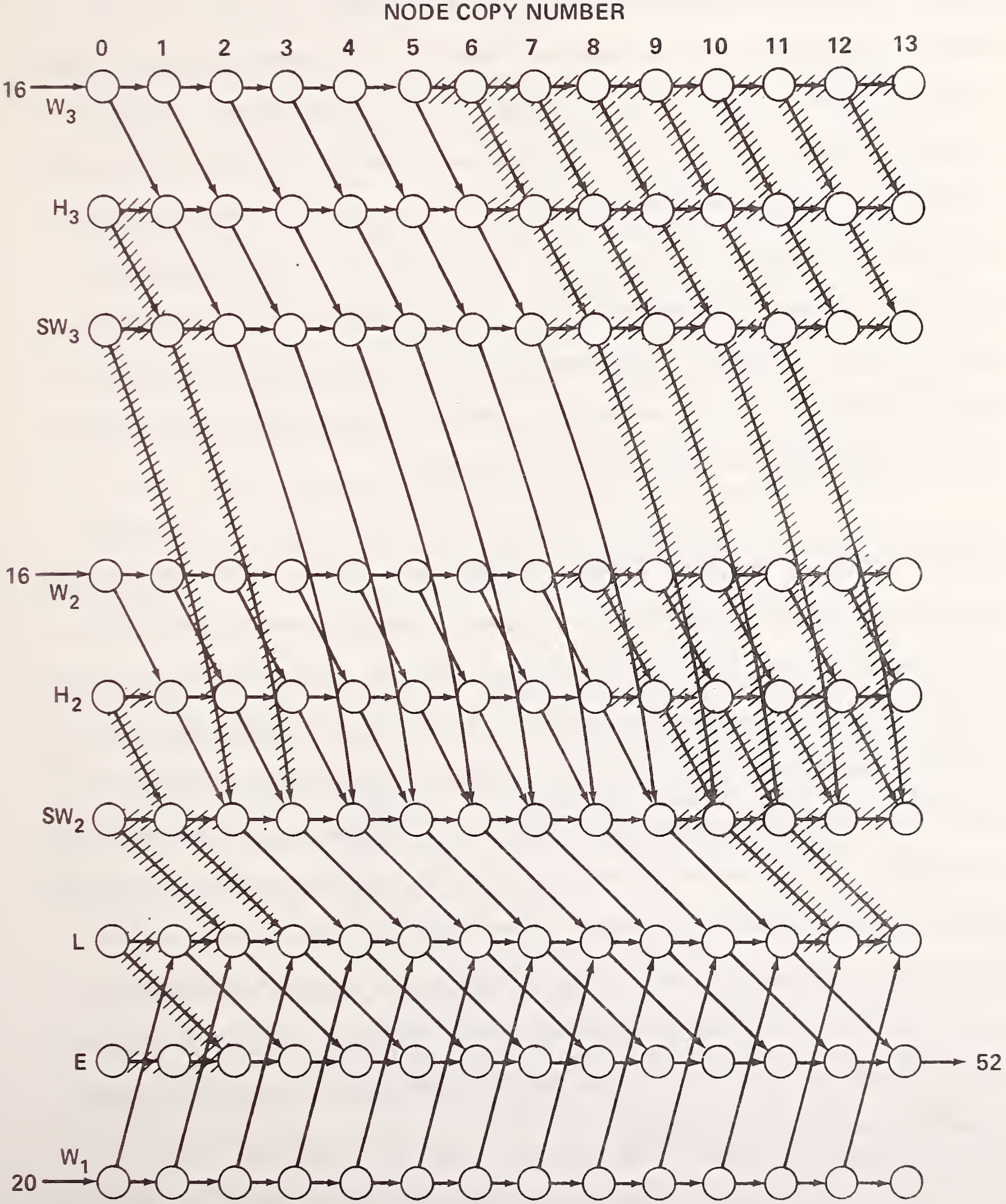


Figure 12 **DYNAMIC NETWORK VERSION OF FIGURE 4 STATIC NETWORK WITH INESSENTIAL ARCS**

dynamic model has a large number of arcs, say 30,000, it would be a non-trivial undertaking to go through and delete all the inessential arcs, even if the deletion is accomplished by means of a computer program. However, if the same model is to be used to make many computer runs, then the effort entailed in deleting inessential arcs may be justified by the consequent reduction in computer run times. It should also be noted, in any event, that when a dynamic network model has a large number of arcs, one would use a computer program to construct it from the static network: the procedure stated above for constructing a dynamic network model from a static network model would not be difficult to program.

Another aspect of using the dynamic network model is the appropriate choice of the duration of a time period. To model a building evacuation lasting 10 minutes in total, for example, would require 60 ten-second time periods, but only 30 twenty-second time periods. By choosing a smaller time period duration, additional model precision is gained, at the cost of having to work with a model with more arcs. Generally, it seems a good idea to choose the duration of a time period to be as large as possible without sacrificing model realism. However, it should be realized that such a choice is to some extent judgemental, and is highly problem-specific. For example, if for a problem of interest the stairwell flow rate is 0.7 people per second, then in order to work with integer data the length of a time period would have to be at least ten seconds.

3.0 A TOTAL EVACUATION MODEL OF THE ADMINISTRATION BUILDING AT NBS

This section is devoted to the construction of a total evacuation model of Building 101, the Administration Building of the National Bureau of Standards. The building is an eleven-floor building on the Gaithersburg, Maryland campus of the National Bureau of Standards. It was chosen as a convenient study vehicle to explore the applicability of network flow optimization models to building evacuation, and not because it is itself of any special interest in that context. While the building is perhaps among the simplest for which the construction of a network flow optimization model may be worthwhile, it is clear that experience gained in modeling Building 101 will facilitate the modeling of more complicated buildings.

In what follows in this section, we describe Building 101, present the static model of the building, discuss how the corresponding dynamic model is obtained, present summaries of a number of computer runs representing evacuation of the building under various conditions, and discuss the implications of the computer-run results.

In addition to floors 2 through 11, Building 101 contains a number of service functions on the first floor, such as auditoria, meeting rooms, a cafeteria, and a library. The building also has an extensive basement floor containing a number of services. Our model represents only floors 2 through 11 together with that part of the first floor which would actually be utilized during a building evacuation. Another reason for not modeling all the details of the first floor is that the lobby is two floors high and, in addition, an unoccupied floor containing the heating and air conditioning systems lies immediately below the second floor. Thus by the time people from the second floor begin to reach the first floor during an evacuation, many of the people on the first floor could already be outside the building.

Figure 13 shows the portion of the first floor of Building 101 which is of interest, the lobby, a crosswalk corridor ending in doors opening onto a crosswalk exit (the crosswalk connects Building 101 to an adjacent building, Building 225), and the "personnel corridor," which connects the lobby and the crosswalk corridor. (This last is the corridor onto which the offices of the Bureau's Personnel Division open.) Note that the lobby has four ("front") exit doors on its east side, and two ("side") exit doors on the north side. In addition, four elevators load and unload via the lobby. Further, one of the two stairwells of Building 101, called Stairwell A for convenience, opens onto the lobby. The second stairwell, called Stairwell B, opens onto the personnel corridor.

Figure 14 presents a rough sketch of the hall of a typical floor, floor 3, of Building 101. Note that the hall is roughly in the shape of an L, with the four elevator doors lying in the short leg of the L, and the two stairwell doors lying along the south side of the long leg of the L. The total floor space in the hall is about 1,650 square feet (153.3 square meters), of which about 175 square feet (16.26 square meters) is directly in front of the elevator doors. Some halls on other floors, particularly floors 6, 7, 8, 9, and 11, have less space than floor 3, due to the

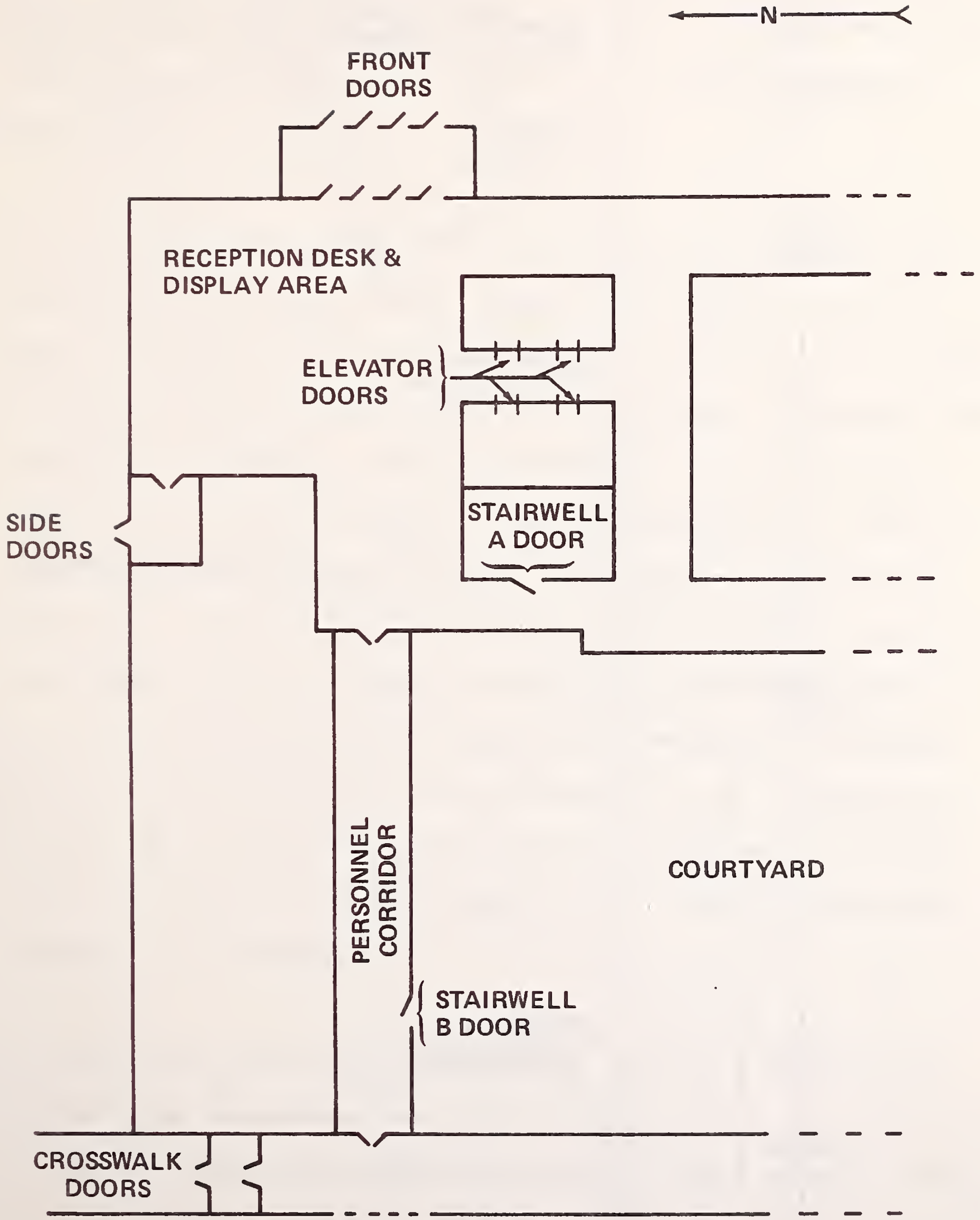


Figure 13 ROUGH SKETCH OF PART OF FIRST FLOOR OF BUILDING 101

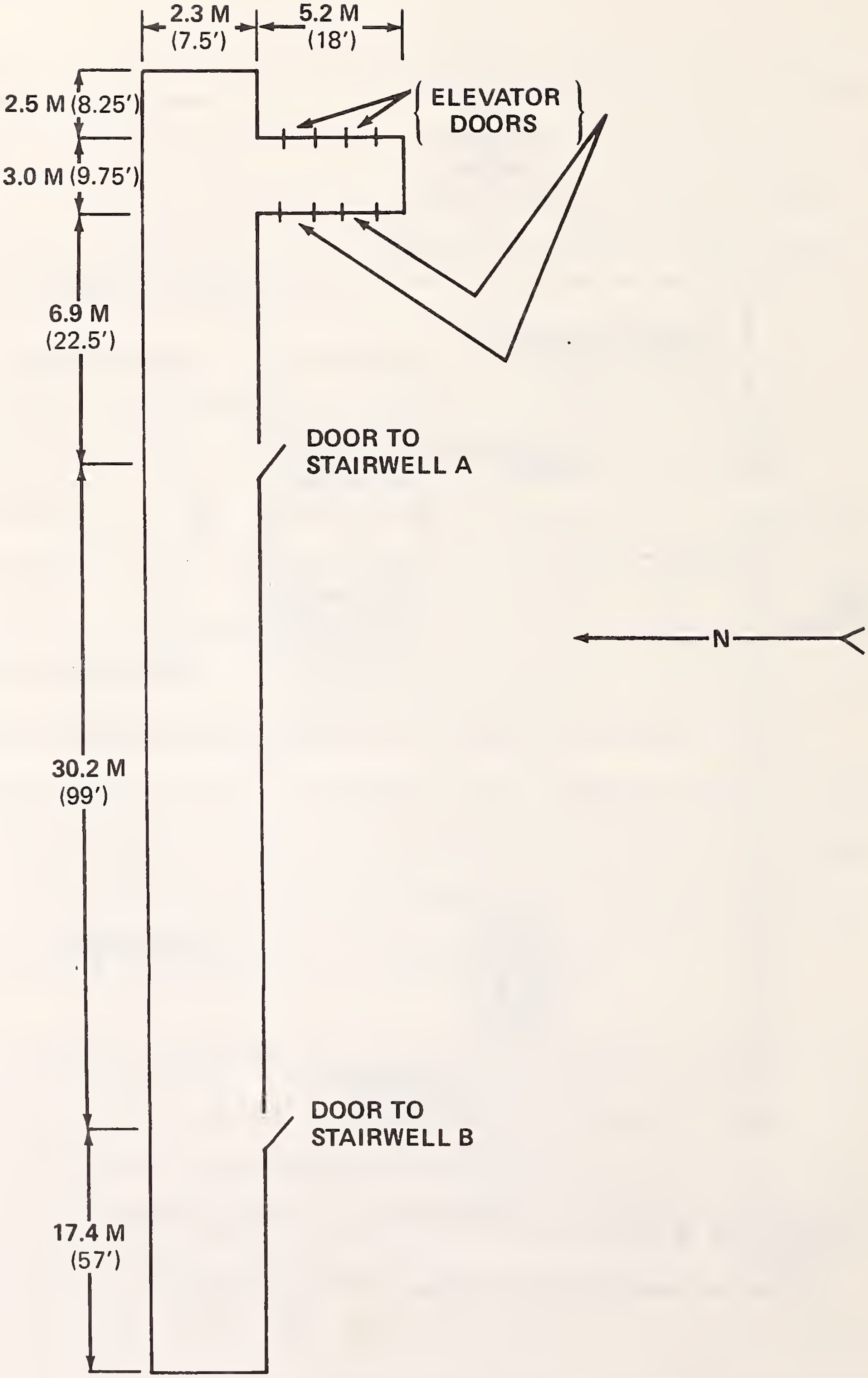


Figure 14 ROUGH SKETCH OF THIRD FLOOR HALL OF BUILDING 101

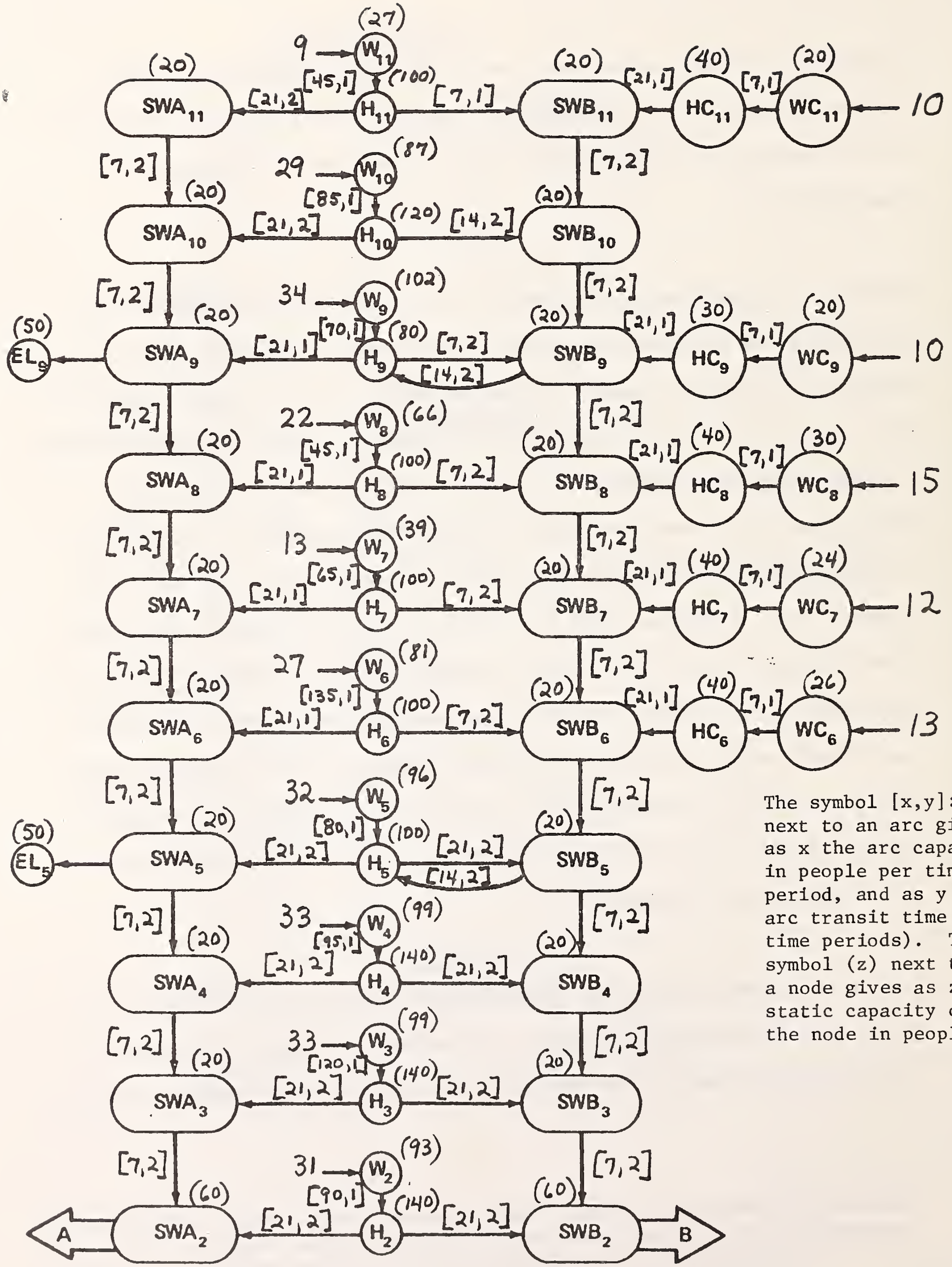
existence of offices at the west end of the floor. Although not shown in the drawing, each of the floors 2 through 11 has offices on both sides of the long leg of the L, with the depth of each office being about 17 feet (5.18 m.).

Figure 15 presents the basic static network flow model of Building 101, and also shows the number of people on each floor. The first page of Figure 15 represents floors 2 through 11, while the second page represents that part of the first floor which is of interest here. The actual offices on each floor are modeled as one or more composite work centers, each with a specified number of people.* Where the distribution of people is reasonably uniform throughout the floor, we use only one composite work center, e.g., Workcenter 3 (W_3) on floor 3. In some instances where the distribution of people on a floor is skewed, with more people on the west end than the east, there are two composite work centers as, for example, on floor 6, with one workcenter situated between the two stairwells (W_6), the other (WC_6) at the west end of the floor. We believe the representation of each floor's offices by only one or two composite workcenters is adequate for our purposes.

The model assumes that during an evacuation, people in a workplace will first move into the adjacent hall. For example, movement from W_6 will be into the middle part of the sixth-floor hall (H_6), while movement from WC_6

---

*Determined from office rosters (name tags near office doors) and inquiries; absenteeism and visitors were not considered in these analyses.



The symbol [x,y]: next to an arc gives as x the arc capacity in people per time period, and as y the arc transit time (in time periods). The symbol (z) next to a node gives as z the static capacity of the node in people.

Figure 15 BASIC STATIC MODEL OF BUILDING 101

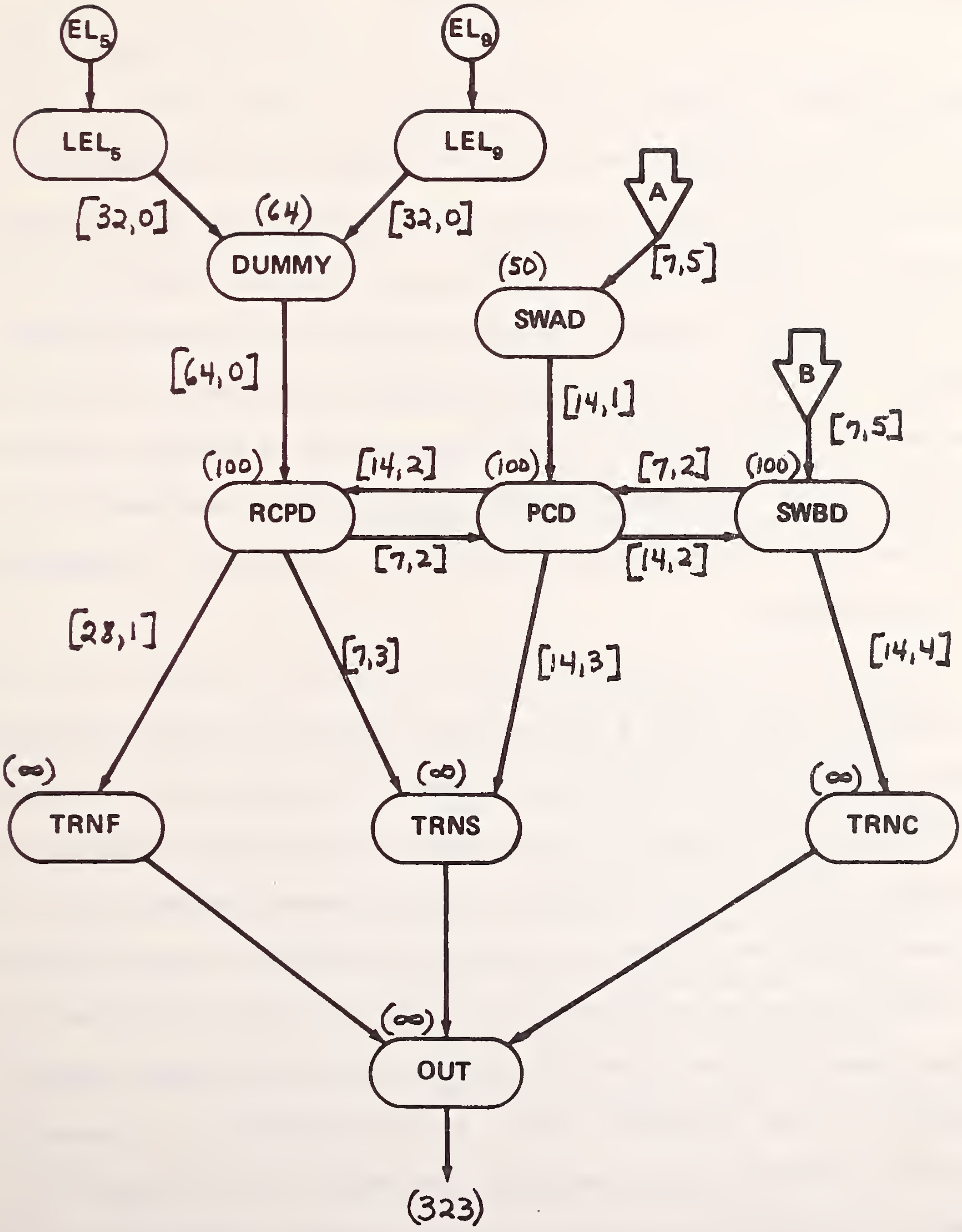


Figure 15 (Continued)

will be to the west part of the hall (HC_6). To continue with the example of floor 6, movement from H_6 can be to either stairwell (SWA_6 or SWB_6), while movement from HC_6 is always to SWB_6.

Next, consider the modeling of floors 5 and 9. These two floors have several extra features; one is to allow transfer from Stairwell B to Stairwell A, the other is to allow access to elevators. Thus the model permits the representation of people descending either Stairwell A or B and then taking an elevator from either floor 5 or 9. Pauls [18] has discussed the use of express elevators, running between the first floor and selected "safe" floors (i.e., floors which have special features to protect them in the event of a fire, such as being pressurized, so as to prevent smoke from entering the floors), to expedite the evacuation of the building.

In reality, floors 5 and 9 are not "safe" floors, and Building 101 probably does not have enough floors to merit the use of express elevators. The inclusion of elevators in this model is primarily hypothetical, with the aim of learning how to include them in a network representation. Alternatively, one may view the Building 101 model as a representation of some hypothetical building having a number of floors (excluding the first floor) which is an integer multiple of ten, and suppose floors 2 through 11 to be "composite" floors, each corresponding to a number of floors in the hypothetical building. One might then also suppose the number of people on each floor to be some "scaled" representation of the actual number of people, e.g., having 19 on floor 11 might actually represent 19 x 2 = 38 people, or 19 x 4 = 76 people, in which case capacities would need to be scaled accordingly. For such a larger "scaled" building, staging elevators might be a desirable feature.

The model assumes that elevators run on a regular schedule between the fifth floor and the lobby, and the ninth floor and the lobby. In particular, elevators leave each of the two floors once per minute for the lobby, beginning at the start of time period five for the fifth floor, and the start of time period six for the ninth floor. To simplify the model a bit, it was assumed that each of the two floors is served by a pair of elevators, operating in tandem. Supposing each elevator to have a capacity of 16 people, a pair of elevators has a capacity of 32. Figure 16 shows the representation of the elevators in the dynamic network model of Building 101.

With reference to Figure 16, we can see that elevators travel from floor 9 to the lobby during periods 6 through 9, 12 through 15, 18 through 21, 24 through 27, and 30 through 33. Likewise the pair of elevators serving floor 5 travel from floor 5 to the lobby during periods 5 through 8, 11 through 14, 17 through 20, 23 through 26, 29 through 32, and 35 through 38. Assuming the elevators travel in tandem allows a simpler network representation of the elevators, as one arc with a capacity of 32 can represent the travel of a pair of elevators in tandem from either floor 5, or floor 9, to the lobby.

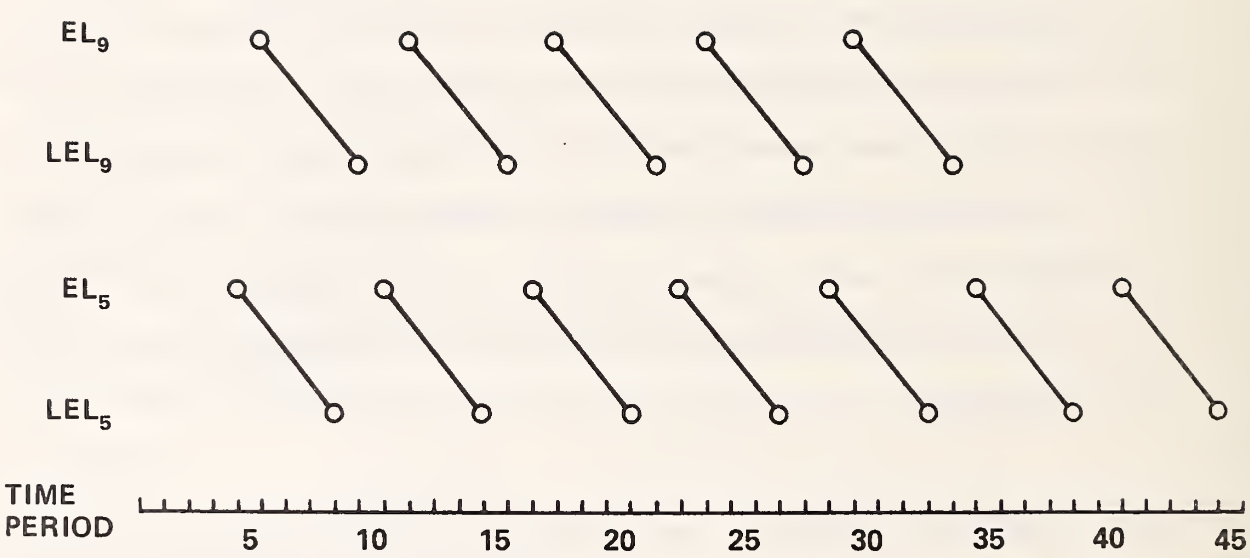


Figure 16 REPRESENTATION OF ELEVATORS IN DYNAMIC NETWORK MODEL OF BUILDING 101

The model allows four time periods for each pair of elevators to load, travel to the lobby, and unload. Likewise, two time periods are allowed for each pair of elevators to travel empty from the lobby up to either floor 5 or 9. (In actuality, the time between elevator doors beginning to close in the lobby until opening completely on floor 5 is about 14.5 seconds, while for floor 9 it is about 18.5 seconds. Since the model represents time by means of discrete time periods, each of 10-second duration, 20 seconds is the closest the model can come to representing the actual travel times.) Towards the end of this study we have concluded that allowing about ten seconds for loading, and ten seconds for unloading, is a bit optimistic; 15 to 20 seconds would be more realistic. While we did not have the opportunity to change the model accordingly, that change could be made rather easily by allowing either 5 or 6 time periods for loading, traveling to the lobby, and then unloading, so that the arcs representing elevator movement in Figure 16 would cut across either 5 or 6 time periods, rather than 4. More generally, it should be clear that the travel of individual elevators can be modeled if desired, providing only that each runs on a fixed schedule in multiples of the duration of a time period. (It does not seem possible to incorporate, in a network flow optimization approach, the case where elevators move in response to demand on individual floors. In any event, one would probably want to preclude such demand-actuated elevator movement during a building evacuation.)

As will be seen, the model has the facility either to use or not to use elevators during the representation of a building evacuation: elevator use is precluded simply by assigning large costs to the arcs representing elevator movement in Figure 16. Elevator use is allowed by assigning costs of zero to these arcs.

The second page of Figure 15 presents the static model of that part of the first floor of Building 101 of interest. The abbreviations have the following meaning:

LEL5 : doors of elevators running between lobby and floor 5
LEL9 : doors of elevators running between lobby and floor 9
DUMMY: a dummy node used to help represent movement from the floor space in front of the elevator doors to the area in the vicinity of the receptionist's desk
SWAD : door opening from Stairwell A onto the lobby
SWBD : door opening from Stairwell B onto the personnel corridor
RCPD : that part of the lobby in the vicinity of the receptionist's desk
PCD : that part of the first floor in the vicinity of the doors between the lobby and the personnel corridor
TRNF : "turnstile" used to record movement through the front lobby doors
TRNS : "turnstile" used to record movement through the side lobby doors
TRNC : "turnstile" used to record movement from the SWB door to the crosswalk doors
OUT : outside

Note that the model assumes people leaving Stairwell A may exit the building by passing through the personnel corridor door, walking along the personnel corridor past the Stairwell B door, and then leaving the building via the crosswalk exit (TRNC). Alternatively, they can pass by the personnel corridor door and travel directly to the side exit (TRNS), or pass by the personnel corridor door, pass by the area in the lobby near the receptionist's desk, and exit via the front doors (TRNF). Similarly, the model assumes that people arriving at the first floor via elevator or Stairwell B can exit the building by any one of the three exits. The total number of people leaving the building via all three exits must be the number of people in the building, 323.

Figure 17 shows a side view of that part of a stairwell joining any two adjacent floors (neither one the lobby). Stairwells A and B are mirror images of one another. Data of Pauls [16] indicate that stairwell widths play a crucial role in determining flow rates on stairwells. Based on observation of actual trial building evacuations, Pauls suggests the following empirically determined equation (see [16]) for predicting stairwell flow rates:

$$f = (0.206)\ w' \ (p/w')^{0.27} \ .$$

In this equation, w' is the effective width of a stairwell in meters, obtained by subtracting 0.3 m. from the actual width, while p is the number of people using the stairwell in total. The term f, the flow rate, has units of people per second. Since the actual width of each stairwell in Building 101 is 44 inches (1.12 m.), the effective width is 0.82 m. If $p = 165$ people use a stairwell, then f works out to be 0.7074 people per second, which when rounded and scaled to 7 people per ten seconds, was

STAIR WIDTH = 44 INCHES = 111.76 CM

TREAD DEPTH = 12.5 INCHES = 31.75 CM

RISER HEIGHT = 6.5 INCHES = 16.51 CM

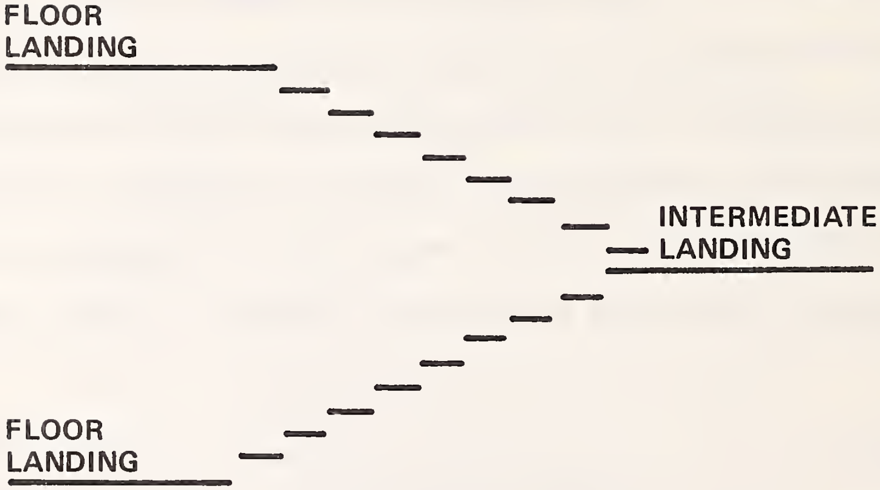


Figure 17 SIDE VIEW SKETCH OF STAIRWELL BETWEEN FLOORS

the stairwell flow rate capacity most often used in the model. As will be seen, the model is quite sensitive to the choice of a flow rate.

Parenthetically, it is interesting to note that the flow rate of 6 x 7 = 42 people per minute is less than half the figure of 90 people per minute recommended for 44-inch-wide stairs in an influential National Bureau of Standards study of 1935 [5]. We shall subsequently present further data suggesting that this flow rate (90 people per minute) is highly optimistic.

Perhaps the simplest means of describing arc and node data is to consider a specific floor. With reference to Figure 15, consider floor 3. The single number in parentheses next to each node represents the static node capacity: at most 99 people can ever be in W_3, at most 140 people in H_3, and at most 20 people in either SWA_3 or SWB_3. The pair of numbers in brackets adjacent to each arc gives, as first entry, the arc capacity per time period and, as second entry, the average number of time periods needed to traverse the arc. The duration of a time period is always ten seconds. Thus, for floor 3, it takes an average of one time period to prepare to travel and then to travel from a workplace into the hall, and as many as 120 people per time period can travel from the workplaces to the hall. Once a person is in the hall, it takes an average of two time periods to travel to either stairwell, and the associated flow rate capacity is 21 people per time period. It takes two time periods to descend one floor in either stairwell, and the associated flow rate capacity is 7 people per time period.

How were the node and arc data (say, for floor 3) determined? The figure for stairwell flow rate capacity was discussed above. The flow rate of 0.7 people per second led in turn to the choice of ten seconds as the duration of a time period, giving a flow rate of 7 people per time period. The arc travel times were determined by actually walking the distances involved at what seemed a moderate rate, and then rounding up the walking time to the next largest number of time periods. With reference to Figure 14, no office door is farther than 55 feet (16.76 m.) from a stairwell, and the model allows 2 time periods to travel 55 feet. Allowing each person in the hall 12 square feet (1.11 sq. m.) permits 140 people (to the nearest 10 people) in the hall. Because the hall is 7.5 feet (2.29 m.) wide, three files of people could walk abreast in the hall with little difficulty. One would expect each file to have a flow rate at least as high as that in the stairwells, giving a flow rate capacity of 3 x 7 = 21 for travel between the hall and the stairwell. Additional substantiation for this figure (21 people per time period) comes from Fruin [7], p.76, who points out that if the average pedestrian area occupancy is 10 to 15 square feet (.93 to 1.39 square meters), then the average flow volume will be between 15 to 20 PFM (pedestrians per foot width of walkway, per minute). Thus allowing each person in the hall 12 square feet would, since the hall is 7.5 feet wide, yield a hall flow rate between (7.5) x (15) = 112.5 and (7.5) x (20) = 150 people per minute, that is about 19 to 25 people per ten-second time period.

The stairwell static capacity of 20 comes from considering the number of steps between floors (see Figure 17) and the space available on the landings. In retrospect, allowing as many as 20 people on the stairwell between any two floors is probably a bit optimistic if a flow rate of 7 people per time period is to be realized. Fortunately, in the computer runs made there were only a few runs for which there were substantial waits in stairwells, so that the assumption of 20 people does not appear to be critical. Also, allowing two time periods to descend a floor may provide some compensation, since Pauls points out [18] that fifteen seconds (1.5 time periods) would be a typical time to descend one floor in a stairwell. The capacity of 120 people per time period on the arc joining W_3 and H_3 comes from counting the number of office doors in use on the floor (24) and assuming it would take two seconds to open and pass through a door, so that 5 people could open and pass through a door in a single time period, in turn permitting 5 x 24 = 120 people to pass through all 24 doors in a single time period.

Data for the floors other than floor 3 were determined in similar manner, with hall capacities being adjusted as necessary to reflect the amount of hall space available. Also, for floors 5 and 9, provision is made to represent people waiting for elevators. Further, the capacity of the arc from the SWB node to the H node is twice the capacity of the arc from the H node to the SWB node, to expedite the flow of people leaving SWB to wait for elevators.

We must emphasize that, despite our efforts to be careful and conservative in these selections of data, we would not wish these figures to be quoted out of context. With the exception of the data based upon Pauls' work, and upon Fruin's work, our figures are not substantiated by an appropriate body of observations and/or experiments. Our primary research effort has been devoted to developing the appropriate model structure, not to obtaining appropriate data for particular situations. In working with the computer model, any change of the model structure may well require the construction of new static and dynamic networks. Data changes, however, are quite simply made. These considerations dictate that a great deal of care goes into determining the appropriate model structure.

Figure 18 illustrates the representation of the turnstiles in the dynamic network model of Building 101. For these turnstiles the numbers on the holdover arcs may be thought of as payments received by individuals who "flow" through the arcs. Thus, for example, a unit of flow entering Turnstile C (TRNC) during period 5 would receive a total payment of 4 + ... + 56, a unit of flow entering during period 6 would receive a total payment of 5 + ... + 56, etc. Payments for Turnstile S work in exactly the same manner. Because it is impossible for any flow to arrive at Turnstile F after period 54, the modeling of Turnstile F stops after period 54. This fact makes it necessary to assign a cost of K = 53 + 54 + 55 + 56 to the holdover arc during period 54, so that payments made to units of flow entering Turnstile F during a given period will be the same as to units of flow entering either of the other two turnstiles during the same period. For example, a unit of flow entering Turnstile F during period 4 would receive a total payment of 4 + ... + 52 + K = 4 + ... + 56, which is the same payment a unit of flow entering either of the other two turnstiles during period 4 would receive.

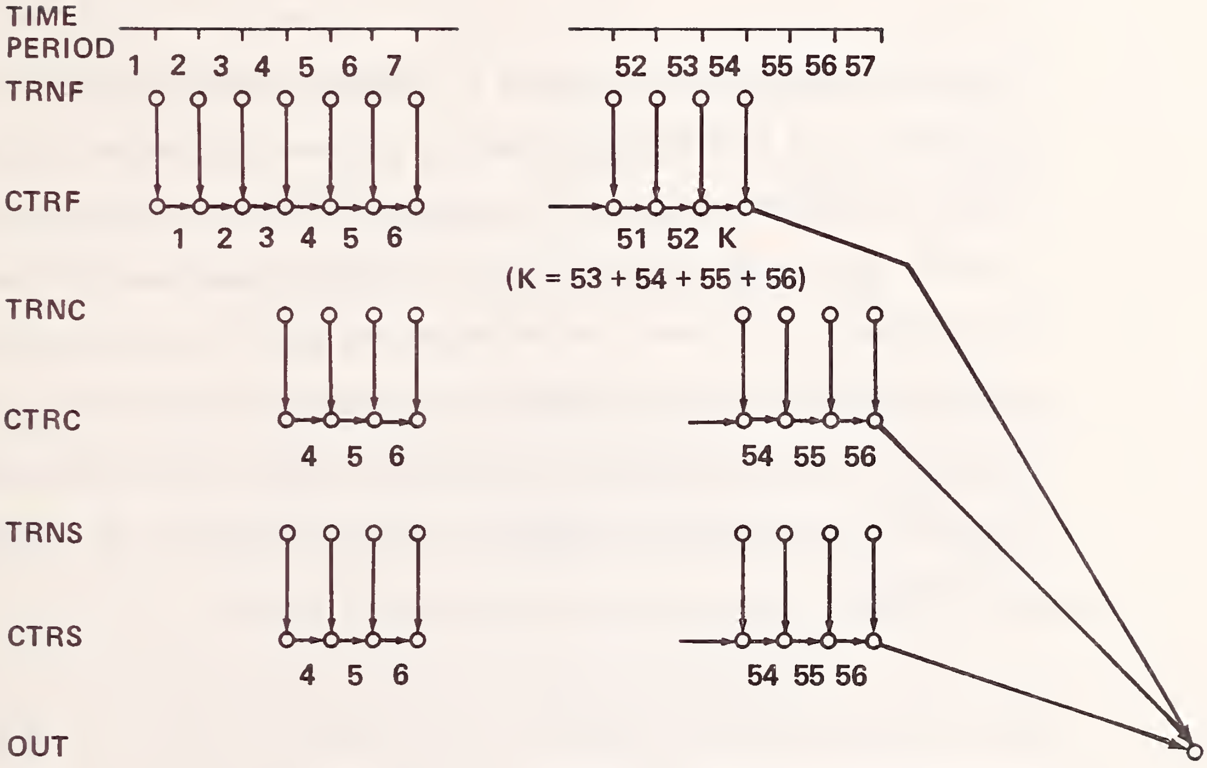


Figure 18 REPRESENTATION OF TURNSTILES IN DYNAMIC NETWORK MODEL OF BUILDING 101

The model elicits the routing of units of flow through the network so as to maximize the total amount of payments received by all 323 units of flow.

It should be noted that the means of assigning turnstile costs illustrated in Figure 18 is different from the one used earlier for the simple model of the three floor building. The approach of this section is a bit more complex, but has the advantage that it can be carried out with fewer arcs: the vertical arcs between the TRN nodes and the CTR nodes are actually unnecessary, and could be omitted. For example, if the TRNC nodes were omitted, the arcs coming from the replicated SWBD node copies could go directly to the associated CTRC node copies, permitting the elimination of 53 arcs.

4.0 COMPUTER RUNS OF EVACUATION SCENARIOS

Once a network optimization model of a building evacuation is contructed, it is easy to study a variety of evacuation scenarios by means of that model, making data changes as necessary to represent the different situations of interest. The study of these scenarios permit "What if?" questions of interest to be addressed. In this section we examine a number of scenarios of possible interest for Building 101. Our interest, again, is not so much in Building 101 <u>per se</u>, as in obtaining insight into the use of network optimization models to study building evacuation.

In summary, the contents of this section are as follows. We first identify the various runs made, using Table 1 both to define the runs and to indicate features of the model which can be changed to study various scenarios. There then follows a summary of each evacuation run, using the static network model of the building, which shows the total flow in each arc, e.g., the total number of people traveling from H_3 to SWB_3, as well as the last time period there is a flow in the arc, e.g, the last time period anyone traveled from H_3 to SWB_3. Data obtained from the runs are summarized in Tables 2 and 3.

Table 1 lists the seventeen computer runs made, which are numbered for reference convenience from 1 through 17. Each column except the last identifies a feature of the model which is changed during the runs. Column 1 lists the dynamic capacity of all the Stairwell A arcs, which was kept as 7 per time period except for runs 16 and 17. Column 2 lists the dynamic capacity of all the Stairwell B arcs, which were either all 7, all 15, or all 7 with the exception of the stairwell arcs from the second floor to the lobby (which were changed to be 3, 1, or 0 in some runs). Column 3 represents the decision to model a building evacuation either with or without the use of elevators. With reference to Column 4: to represent a situation where a particular floor had more than its customary number of people, in some runs 50 additional people were placed on floor 10 via the model. When 50 additional people were placed on floor 10, in some runs first priority (see Column 5) was given to the evacuation of floor 10 by assigning costs of 999 to all the holdover arcs on floor 10: in effect, it was made expensive to keep units of flow waiting on floor 10, as might be the case in actuality if, for example, a fire broke out on floor 10. In conjunction with giving first priority to the evacuation of floor 10, in some runs second priority was given to the evacuation of floor 11 (see Column 6) by assigning costs of 555 to all of the holdover arcs on floor 11. This second priority assignment was suggested by the possibility that a fire breaking out first on floor 10, might next threaten floor 11.

The model has the capability of using any combination of the following exits: front, side, and crosswalk. Whether or not an exit is available for use is represented by whether or not a large cost (typically 99999) is assigned to the arcs leading immediately to the node representing the exit. Note that in runs 1 through 17, the side lobby exit was kept unavailable. In preliminary runs made, because the walking time from either first-floor stairwell door is the same to the front exit as to the side exit, the model was routing flow indifferently via both exits. However, in reality the side exit is not used much in comparison to the

RUN #	SWA CAP.	SWB CAP. *(2nd TO 1st)	ELEVATORS?	+ 50 PPL. ON 10th?	1st PRIOR. TO 10th?	2nd PRIOR. TO 11th?	FRONT EXITS?	XWALK EXIT?	EVAC. TIME
1	7	7	NO	NO	NO	NO	YES	NO	35
2	↓	↓	↓	↓	↓	↓	↓	YES	35
3	↓	↓	↓	YES	↓	↓	↓	↓	38
4	↓	3*	↓	NO	↓	↓	↓	↓	44
5	↓	1*	↓	↓	↓	↓	↓	↓	52
6	↓	7	YES	YES	↓	↓	↓	↓	27
7	↓	↓	↓	NO	↓	↓	↓	↓	23
8	↓	3*	↓	↓	↓	↓	↓	↓	26
9	↓	0*	↓	↓	↓	↓	↓	↓	28
10	↓	7	NO	↓	YES	↓	↓	↓	35
11	↓	↓	↓	YES	↓	↓	↓	↓	38
12	↓	↓	YES	↓	↓	↓	↓	↓	27
13	↓	↓	NO	NO	↓	YES	↓	↓	35
14	↓	↓	↓	YES	↓	↓	↓	↓	38
15	↓	↓	YES	↓	↓	↓	↓	↓	27
16	15	15	NO	NO	NO	NO	↓	↓	30
17	↓	↓	YES	↓	↓	↓	↓	↓	22

Table 1. Run Listing with "What if" Features

front exit, due in part to the fact that the major parking lot serving building occupants is most readily accessible via the front exit. As it thus seems unlikely the side exits would receive much use even in an emergency evacuation,* the use of the side exits was precluded in the model. The possible preference of building occupants for particular exits is a behavioral phenomenon which the network optimization model cannot very well reflect.

The last column of Table 1 gives the number of time periods needed to evacuate the building for each run. Generally speaking, these times seem to vary in the way one might expect, with reduced flow rates causing evacuation times to increase, having more people in the building causing evacuation times to increase, and use of elevators causing evacuation times to decrease substantially. An interesting anomaly occurs in run 16, where the building evacuation time is 30 time periods, a decrease of 5 time periods from run 2, even though the stairwell capacities are more than twice those in run 2: the reasons for this anomaly will be considered in some detail subsequently.

Runs 1 through 17 are summarized graphically in Appendix A. Because of the number of these runs, a detailed discussion of each would make this part of the report long and tedious. Therefore we shall discuss only a few runs in detail, and comment briefly on the others. Then, rather than making detailed verbal comparisons of various runs, we shall summarize the comparisons in two tables, Tables 2 and 3.

---

*Unless specific instructions to the contrary were communicated and reinforced.

Run 2 may be considered the benchmark run, in the sense that it most closely represents the existing state of Building 101. With reference to Figure A-2, data presented in parentheses next to each arc, of the form (x,y), have the following meaning: x is the total amount of flow through the arc, while y is the last time period the arc is used. For example, for the arc from H_3 to SWA_3, there is a total flow on the arc of 14, and the last time period of use is during period 3. By examining the data for the stairwell arcs, we can easily determine the last time period during which portions of the stairwell are used. For example, for Stairwell A, the last time that part of the stairwell between floors 10 and 9 is used is during period 7, the last time that part between floors 9 and 8 is used is during period 14, etc. By the end of time period 31, all 162 people using Stairwell A have passed through the Stairwell A Lobby Door. Likewise, by the end of time period 30 all 161 people using Stairwell B have passed through that stairwell's Lobby Door. All the people who used Stairwell A exit via the front exits, while all the people who used Stairwell B exit via the crosswalk exit. Note that the numbers of people using the front and crosswalk exits are almost identical, as are the last time periods each exit is used.

With reference to floors 5 and 9 in Figure A-2, we can see some unnecessary transfers from Stairwell B to the Hall and then back to Stairwell B. This transfer facility was built into the model so that evacuation using elevators can be represented. The model in its current state does not reflect the fact that such transfers from the stairwell to the hall and back are unnecessary, as well as perhaps undesirable, when elevators are not used. However, such transfers can be precluded by attaching a high cost to the arc from the stairwell to the hall: one would then need to remember to reduce this cost to zero in runs in which elevators are to be used. It is well to remember that the driving force of the model for evacuating people from the building consists of the costs assigned to the turnstiles: the model as is will permit "looping" between Stairwell B and the Hall on floors 5 and 9 unless the looping is precluded by using high arc costs for one of the arcs in the loop. This anomaly has no effect on the objective function value, i.e. whether people are "waiting" or "looping" makes no difference in total evacuation time.

Examining the output results for the dynamic evacuation model shows that exactly 7 people per period use Turnstile C in each of 23 time periods: exactly 7 people use Turnstile F in each of 23 time periods as well, while there was one time period in which only one individual uses Turnstile F. Because Pauls' model predicted 7 people per time period, it is reassuring to find such close agreement.

Figure A-2 illustrates an interesting phenomenon which occurred in every run except Run 9 (in which elevators are used and Stairwell B "closed"): namely, the time periods during which each of the two exits "clear" differs by at most one. In many cases the "clearing times" are identical. For lack of a better term, we refer to this uniformity of clearing times as the "uniformity principle." The principle may be stated as follows: Given a building in which each occupant has reasonable access to every one of the evacuation routes, if the building is evacuated in minimum time then the times at which the evacuation routes clear will tend to be the same. For a simpler building model the uniformity principle can be

RUNS	DIFFERING FEATURE	COMMENTS
1,2	Use TRNC only in Run 2	Same building evacuation time, similar flow patterns
2,3	50 extra people on 10th floor in Run 3	Evacuation time 30" longer in Run 3
2,4	SWB capacity of 3 in Run 4	Noticeable change in Stairwell Utilization
2,5	SWB capacity of 1 in Run 5	Drastic change in Stairwell Utilization
2,6	Use elevators in Run 7	Building evacuation time decreases from 350" to 270"
2,10	10th floor has first evacuation priority	Evacuate 10th floor in 30" less
2,13	10th & 11th floors have first & second evacuation priority	Floor evacuation times decrease from 60" to 30"
2,16	SW capacities of 15 in Run 16	Building evacuation time decreases from 350" to 300". Flow rates of 15 seldom attained
3,6	Use elevators in Run 6	Building evacuation time decreases from 380" to 270"
3,11	10th floor has first evacuation priority	Floor evacuation time decreases from 80" to 50"
4,5	SWB capacity of 1 in Run 5	Building evacuation time increases from 380" to 520". Less use of SWB in Run 5
4,8	Use elevators in Run 8	Building evacuation time decreases from 440" to 230"
7,17	SW flow rate of 15 in Run 17	Building evacuation time decreases by 10"
8,9	SWB closed between second & first floor in Run 9	Building evacuation time increases by 30"
10,11	50 extra people on 10th floor in Run 11	Building evacuation time increases by 30"
11,12	Use elevators in Run 12	Building evacuation time decreases from 380" to 270"
13,14	50 extra people on 10th floor in Run 14	Building evacuation time increases by 30"
14,15	Use elevators in Run 15	Building evacuation time decreases from 380" to 270"
16,17	Use elevators in Run 17	Building evacuation time decreases from 300" to 210"

Table 2. Partial Listing of Runs Having One
Differing Feature, with Comments

Periods in Which Floors Clear	Floor #											# VIA ELEV.	# VIA SWAD	# VIA SWBD	# VIA TRNF	# VIA TRNC
Run #	11	10	9	8	7	6	5	4	3	2	1					
1	5	6	11	5	4	3	19	4	3	3	35	0	162	161	323	0
2	6	6	11	6	12	6	17	4	4	3	35	0	162	161	162	161
3	6	8	13	14	9	3	22	5	4	3	38	0	189	184	189	184
4	7	7	17	15	11	15	27	3	9	20	44	0	226	97	226	97
5	8	9	27	13	10	23	34	5	3	19	52	0	282	41	282	41
6	6	14	17	6	8	5	22	3	4	3	27	194	78	101	271	102
7	6	8	15	6	7	5	16	6	4	5	23	176	63	84	238	85
8	5	9	14	4	7	12	16	6	9	3	26	177	101	45	277	46
9	6	3	15	10	9	14	22	7	7	5	28	206	117	0	322	1
10	5	3	12	12	9	6	19	5	4	3	35	0	162	161	162	161
11	6	5	14	14	12	7	23	5	4	3	38	0	184	189	184	189
12	7	5	17	17	8	8	22	3	4	3	27	194	76	103	269	104
13	3	3	12	12	15	3	17	6	4	4	35	0	162	161	162	161
14	3	5	14	14	13	3	23	3	4	3	38	0	189	184	189	184
15	3	5	16	6	7	5	22	3	4	3	27	194	76	103	269	104
16	3	3	3	3	3	3	4	3	3	3	30	0	149	174	151	172
17	4	3	14	14	3	3	16	3	3	3	22	168	67	88	235	88

Table 3. Summary of Runs, With Periods in Which Floors Clear, & Numbers of People Using Various Routes

proven mathematically: see Appendix B for more detail. The principle is easy to motivate: if some evacuation route clears at a substantially later time than the others, then some of the people who use this route could instead use other routes (to which they have reasonable access), thus reducing the time to evacuate the building.

Figure A-1 summarizes Run 1, in which the use of the Crosswalk Turnstile is precluded. Note that the time to evacuate the building is the same as in Run 2 (35 time periods). This is because the walking time (in the model) from the Stairwell B Door to either the Front or Crosswalk Turnstile is the same, and the Front Turnstile has twice the capacity of the Crosswalk Turnstile.

For Run 3 (see Figure A-3) 50 additional people are placed on the tenth floor to be evacuated. It then takes 38 time periods (three more than in Runs 1 and 2) to evacuate the building, and 8 time periods (two more than in Runs 1 and 2) to clear the tenth floor.

The data for Run 4 are the same as for Run 2, except that the flow rate capacity of Stairwell B is changed from 7 to 3 in the stairwell between the first and second floors. This reduction might represent a partial blockage of Stairwell B. Note how the model adjusts for this reduction, sending only 97 people via Stairwell B, and 226 people via Stairwell A. Further, not only is the uniformity principle still true, but the times at which the two stairwells clear are the same. By comparison with Run 2, the flow rate capacity reduction in Stairwell B causes the building evacuation time to increase from 35 to 44 time periods.

Run 5 has the same data as Run 4, except that the flow rate capacity of Stairwell B is reduced further, from 3 to 1, between the second and first floors. Even fewer people (48) used Stairwell B than in Run 4. In spite of this, the two stairwells still clear at the same time, as do the two turnstiles. The number of time periods needed to evacuate the building increases to 55.

The data for Run 6 are the same as for Run 3, but there are 50 extra people on the tenth floor and the use of elevators is allowed. The elevators serving floor 9 make three round trips, while those serving floor 5 make four round trips. Note that 46 of the people on floors 9 through 11 do not use the elevators. Likewise, a number of people on floors 5 through 8 do not use the elevators, using the stairs instead. It is interesting to note that one individual entering the lobby via the Stairwell A Door exits the building via Turnstile C: an examination of the output for the dynamic model showed that this individual arrived during a time period when a large number of people who had arrived via the elevators were using the Front Turnstile to capacity. (The same phenomenon occurs in Runs 7, 8, 9, 12, and 15.) As a final remark, it is interesting to note that the uniformity principle holds, even though, due to the use of elevators, numerous people in the building do not have reasonable access to every evacuation route, e.g., people on floors 2, 3, and 4 do not have access to the elevators.

The data for Run 7 are the same as for Run 4, except that the number of people on the tenth floor is reduced back to 29. It is interesting to note that only one of the people on floors 9, 10, and 11 do not use the elevators running from floor 9. More people use Stairwell B than Stairwell A; one might expect this, because Stairwell A is closer to the elevators.

For Run 8, the flow rate capacity of that part of Stairwell B joining the second and first floors is taken to be 3, and the use of elevators is allowed. In this run everyone on floors 9 through 11 uses the elevators, and more than twice as many people use Stairwell A as used Stairwell B. Again, the uniformity principle holds.

The data for Run 9 are the same as for Run 8, except that the use of Stairwell B between the first and second floors was precluded by assigning very large costs to the associated stairwell arcs. Note that everyone on the ninth through the eleventh floor uses the elevators, and 54 people transfer from Stairwell B to Stairwell A on the fifth floor.

For Run 10, costs are assigned to the holdover arcs on the tenth floor, to represent a situation for which it is urgent to get everyone off the tenth floor. Indeed, everyone is off the tenth floor by the end of time period 3. (By comparison, it takes 6 time periods to get everyone off the tenth floor during Run 2.) This may be an appropriate place to remark that, unless costs are assigned to holdover arcs for floors, the model does not reflect the fact that it is better for people to be in the stairwells than to remain on the floors.

The data for Run 11 are the same as for Run 10, except that there are 50 extra people on the tenth floor. Note that the tenth floor is evacuated by the end of time period 5, as compared to 8 time periods during Run 3.

For Run 12 there are 79 people on floor 10, it is "expensive" to remain on Floor 10, and elevators can be used. More people use Stairwell B than use Stairwell A, and 46 of the people on floors 9 through 11 do not use the elevators.

The data for Run 13 are the same as for Run 2, except that holdover arc costs of 999 are used for the tenth floor, and holdover arc costs of 555 are used for the eleventh floor. Floors 10 and 11 both clear by time period 3, as compared to time period 6 during Run 2.

The data for Run 14 are the same as for Run 13, except that 50 extra people are placed on floor 10, causing the floor to clear in the fifth time period, as compared to the third time period in Run 14.

For Run 15, the data of Run 14 are changed to permit the use of elevators, accordingly effecting a drastic reduction, from 38 to 27, in the number of time periods needed to evacuate the building.

For Run 16, the data are the same as for Run 2, except that stairwell flow rate capacities are 15 per time period (90 people per minute). Even though the stairwell flow rate capacities are more than doubled, from 7 to 15, the time to evacuate the building only decreases from 35 to 30 time periods. An examination of the number of people passing through the Stairwell A and B doors per time period shows that flow rates of 15 people per time period are seldom attained: only for time periods 10 and 24 for Stairwell A, and 10 and 24 for Stairwell B. Figure 19 shows the number of people per time period passing through the Stairwell B door, for each of the time periods 8 through 26: the variation from one time period to another is quite noticeable. Effectively, what happened in Run 16 is that the Stairwell flow rate capacities are so large that they are hardly ever binding; in fact the 30 time periods needed to evacuate the building is just the number of time periods needed to walk from the eleventh floor to the exits. Additional insight can be obtained by observing that _if_ a flow rate of 15 people per time period could be attained, then it would take only 23 time periods to evacuate the building. The first people from floor 2 will reach the exits during the twelfth time period: if a steady flow rate of 15 could then be maintained, with 162 people using one exit and 161 using the other exit, it would take 162/15 = 10.8 and 161/15 = 10.73 time periods respectively for the files of people using the two stairwells to pass through the exit, giving 12 + 10.8 = 22.8 and 12 + 10.73 = 22.73 time periods, rounded up to 23 time periods, needed for the two turnstiles to clear. As these times are less than the time to walk from the eleventh floor to either exit, it is clearly impossible to attain a flow rate of 15 people per second in Building 101 for the conditions of Run 16. We speculate, but have had no opportunity to check, that if all the data for Run 16 were held constant except that more people were in the building, then there would be some "critical number" of people in the building, above which the flow rate of 15

Figure 19 FLOWS THROUGH SWB DOOR, RUN 16

could be attained, and below which it would not always be attained. If this critical number of people is x, we would expect $x/(2 \times 15) + 12 \geq 30$, so that $x \geq 30 \times 18 = 540$. Even if a flow rate of 15 could be attained in the model, there is no guarantee that it could be attained in an actual evacuation. It should be remembered that stairwell flow rate capacities are *upper bounds* on stairwell flow rates, and should be set accordingly. For most runs we set the stairwell flow rate capacities to be 7 because we believed this to be a reasonable upper bound on the flow rate.

The data for Run 17 are the same as for Run 16 except that elevators are used. We can see that all but 5 of the people on floors 9, 10, and 11 use the elevators, while 86 people used the elevators running from floor 5. A total of 88 people use Stairwell B, while a total of 67 people used Stairwell A; this might be expected, since the elevators are closer to Stairwell A than to Stairwell B. In this run the building was evacuated in 22 time periods, the smallest evacuation time of any of the runs, although only 1 time period less than in Run 7, which has identical data to Run 17 except for stairwell flow rate capacities of 7.

The reader is further referred to Tables 2 and 3 for further summaries of the runs. Table 2 lists a number of runs which differ by only one feature, identifies the feature, and gives comments on differences in results obtained. Table 3 lists the time period in which each floor is cleared of people in each run, and also summarizes the number of people evacuating the building by various routes. For further information we refer the reader to computer printouts included in the appendix.

At this point we summarize some of our conclusions from the runs discussed above, reserving for the following section a discussion on general conclusions, open questions, model limitations, and possible future work.

We believe our experience in modeling Building 101 demonstrates the fact that it is possible to construct "skeletal," i.e., network, models of an entire building and the people it contains: the models are quantitative, and recommend ways to route people out of the building so as to minimize the total evacuation time. Because these are optimization models, they provide standards of comparisons, or benchmark solutions; idealized goals towards which to strive. Further, such models can be designed so that they can be at least partially validated, in the sense that they give results (as in Run 2) which compare well with the empirical results of Pauls. As one might expect, stairwell flow rates turn out to be critical, and the use of elevators reduces building evacuation times drastically. Runs 16 and 17 cast considerable doubt (over and above the doubt already cast) on the widely accepted flow rate assumption of 90 people per minute for stairs 44" (1.12 m.) wide, and suggest that stairwell flow rates can be maintained at their upper limit only if there is a sufficiently large number of people in the building. Finally, we state again the uniformity principle:

> Given a building in which each occupant has reasonable access to every one of the evacuation routes, if the building is evacuated in minimum time, then the times at which the evacuation routes clear will tend to be the same.

We remark that the uniformity principle is not unrelated to one of the two principles of WARDRUP dealing with traffic flow [19], [24].

5.0 COMPUTER IMPLEMENTATION

This section presents a working level, first-attempt computer implementation of a network optimization model for building evacuation. The material in this section is intended for a programmer interested in utilizing the previously described solution procedure. Readers not familiar with computer programming may choose to omit this section. The computer codes described herein were created for the purposes of (1) examining the computational feasibility of that model, and (2) examining the behavior of the model under various evacuation conditions for Building 101 at the National Bureau of Standards. The current version of this material is viewed as only a rough first version of a software package which could ultimately provide a practical tool for understanding the relationship between the evacuation process and the building configuration (with modifications to represent emergency conditions). Such a tool could be useful to architects in designing buildings, to municipal officials responsible for building codes, and to safety officials interested in developing and promoting appropriate evacuation strategies.

The computer package consists of four programs. The evacuation optimization is accomplished using GNET, a primal network transshipment-problem code graciously made available to NBS by the authors Bradley, Brown, and Graves [3]. The other three programs were written at NBS to aid in data preparation and output analysis. SETUP helps the process of enlarging the static network representing the building to the time-expanded network which is input to GNET. CHECK performs the reverse operations, producing a convenient record of changes made to the static network associated with a specified time-expanded network. POSTPRO transforms the standard GNET output to a form more readily useful for analysis of the results. Both CHECK and POSTPRO contain some lines of code which are specific to the building or network under consideration. This part of the report will discuss these four programs, all of which are written in FORTRAN and were executed on the UNIVAC 1108 at NBS under the EXEC 8 Operations System. Use of these programs will be illustrated via a complete set of input and output data for the three-story building described in Section 2.0. Listings of SETUP, CHECK, and POSTPRO appear in Appendix C.

The Solution Procedure: GNET

The dynamic model described in Section 2.0 is a capacitated network flow model, in particular, a transshipment model. Problems of this type are readily solved by existing computer transshipment algorithm programs such as GNET. (Requests for copies of GNET should be addressed to the GNET authors.) For solving the building evacuation model, GNET was used as a "black box," i.e. no modifications were made at NBS to the code originally supplied by the authors. For all runs the output switch was fixed at 1 to produce only the summary output (described later) and a listing of arcs with non-zero flow in the final solution. This output served as input to the postprocessor, POSTPRO.

Input to GNET consists of the following information. The first record gives the number of nodes (in the time-expanded network) and the value of the output switch in the format 2I5. This is followed by one record for each arc (in the time-expanded network) giving the arc name (optional), origin node number, destination node number, cost, and capacity in the format (A4, 2X, 2I6, 2X, 2I10). (The "origin" and "destination" nodes for a particular arc are the nodes at the tail and head of the arc, respectively.) GNET input requires that nodes be numbered sequentially with no gaps, but the ordering of the arc input may be arbitrary. When the final solution is listed, arcs are grouped by destinations with lower numbered destinations first. For the model of Building 101, with time periods of 10 seconds duration, and allowing a maximum of 10 minutes to evacuate, there were 2591 nodes and 5543 arcs. GNET also requires two additional "super" nodes. The super source node (in this case node 2592) supplies people to each work place. The super sink node (2593 for this example) receives people from each exit. The appropriate arcs to and from these super nodes must be supplied. Costs on these arcs are zero. The capacity on each arc from the super source is equal to the number of people at the corresponding work place. Capacities on the arcs to the super sink must sum to the number of people in the building. The average time for computation (excluding input and output), over 17 runs on the Building 101 model, was approximately 30 seconds per run. Future research should be directed toward exploring the possibility of developing a solution procedure which, by exploiting the special structure of such models, could speed up computation time.

<u>The Preprocessor: SETUP</u>

A simple preprocessing code was written to assist a user in preparing the input for GNET. This preprocessor requires as input the following data for each arc ARC in the static network representation of the building.

(a) the name of the arc, NAME(ARC)

(b) the number of the origin node, ORIG(ARC)

(c) the number of the destination node, DEST(ARC)

(d) the cost, COST(ARC)

(e) the capacity, CAP(ARC)

(f) the number of copies of the arc needed in the time-expanded network, NUMB(ARC).

The code expands the network in the following way. For each arc, ARC, in the static network, the code generates NUMB(ARC) arcs in the time-expanded network. Consider, for example, the N-th copy of arc ARC, N = 0, 1, ..., NUMB(ARC)-1. The arc number TARC of the arc in the time-expanded network can be calculated as

$$\mathrm{TARC} = \sum_{I=1}^{\mathrm{ARC}-1} \mathrm{NUMB}(I)+N+1 \ .$$

The properties of arc TARC are calculated from the properties of arc ARC in the static network in accordance with the following:

$$\begin{aligned}
\mathrm{NAME(TARC)} &= \mathrm{NAME(ARC)} \\
\mathrm{ORIG(TARC)} &= \mathrm{ORIG(ARC)}+N-1 \\
\mathrm{DEST(TARC)} &= \mathrm{DEST(ARC)}+N-1 \qquad (5\text{-}1) \\
\mathrm{COST(TARC)} &= \mathrm{COST(ARC)} \\
\mathrm{CAP(TARC)} &= \mathrm{CAP(ARC)}
\end{aligned}$$

Exceptions to this method of expansion can be handled by editing the resulting data file for the time-expanded network.

A listing of this expansion program appears in Appendix C. The first input record contains the number of arcs in the static network (including arcs originating at the super source and arcs destined for the super sink) and the number of nodes in the time-expanded network (excluding the super source and the super sink), in the format 2I5. Thereafter the input consists of one record for each arc numbered above. Each record contains NAME(ARC), ORIG(ARC), DEST(ARC), COST(ARC), CAP(ARC), and NUMB(ARC) in the format (A4, 2X, 2I6, 2X, 3I10). Output from this program is directed to logical unit 7 and is in the format required for input to GNET.

The expansion procedure described above is extremely simplistic, as will be explained in the next paragraph. Its biggest advantage is the savings it affords in keypunching time and costs. To generate the 5543 arcs for the 11-story building described earlier, the user need only prepare data for the 175 arcs needed to represent the static network.

It should be noted and emphasized that preparing the arcs of the static network is not a quick and simple process. An appropriate conceptualization of the building must be developed to represent workplaces, stairwells, elevators, halls, interior doors, and exits, along with reasonable capacities and flow rates on connecting arcs. This conceptualization might best be developed as a drawing such as Figure 4. Next, the user must determine the number of expansion arcs needed for each arc in the static network. As stated in Sethis determination can be accomplished by determining a value for T, the maximum number of time periods needed to evacuate the entire building, and then expanding the network to have T copies of each arc in the static network plus holdover arcs connecting time-adjacent copies of each node. Although this procedure results in a network which includes some arcs ("inessential") which will never be used,[*] it is far simpler than a procedure which would delete all inessential arcs. Future study might well be devoted to evaluating the merits of each alternative.

After determining the number of expansion arcs needed, the user should give some thought to the numbering of nodes. Although GNET does not require any particular order when the arc data are input, there is a requirement that all nodes be numbered sequentially with no gaps. The post-processor and the expansion procedure both require that the origin and destination of the (n+1)-st copy of an arc be one greater than the origin and destination of the n-th copy. The user may, however, have preferences about the order in which these groups occur in output. For example, it may be desirable to see workplace-to-hall type arcs, beginning with the top floor and working down to the ground floor, followed by all hall-to-stairwell arcs in the same order, etc. If so, then nodes representing workplaces should have smaller numbers than nodes representing halls, and halls should have lower numbers than stairwells. Likewise, nodes on higher floors should have lower numbers than nodes on lower floors. A systematic

---

[*] In the very early time periods it would be impossible for people originating at work places to be traversing hall or stairwell arcs to exits. In the latter time periods arcs on higher floors are not used because everyone would have already evaculated to a lower floor.

procedure for ordering nodes can greatly facilitate the output analysis. It would not be difficult to automate the entire process of network expansion, including the counting and numbering of nodes, provided that each node has the same number of copies. (Determining how best to identify and delete inessential arcs requires further study.) Indeed, automating the expansion process is one of the top-priority efforts to be associated with the modelling of any additional buildings to be studied.

The Verification Procedure: CHECK

A typical application of the building evacuation model might consist of exercising the model on a number of different scenarios to represent various effects of disasters within the building: for example, a stairwell might be blocked above some floor after some time period, elevators might be out-of-service, etc. These situations are represented in the time-expanded network by modification of costs and/or capacities on selected arcs. Such changes are relatively easy to accomplish via an editing package available on most large computers. Particularly when many runs are to be made involving many data modifications, it is desirable to have some mechanism to help insure that the network model is and remains correct. A verification procedure was written to aid in this checking. The procedure accepts as input the time-expanded network and produces as output the corresponding static network. Checks are made to ensure that the relationships of (5-1) hold. Input to this program is identical to input for GNET as given earlier in this section. Output is identical to the input to the preprocessor, SETUP, described previously. A listing of this procedure appears in Appendix C.

It should be noted that the current version of this program contains several lines of code which are specific to the model of Building 101. These specificities, which concern legitimate exceptions to the relationships of (1), occur in the listing as several of the lines of code between the statement labeled 140 and the statement labeled 146. It would not be difficult to exclude these exceptions or to replace them with other exceptions.

The Post-Processor: POSTPRO

The standard output from GNET is a list of all arcs which have non-zero flow. These arcs are identified by the numbers of their origin and destination nodes. For the present application it was felt that node names rather than numbers would greatly facilitate the interpretation of output. Furthermore, in the GNET output arcs are ordered by destination, i.e. arcs with lower numbered destination appear prior to arcs with higher numbered destinations. For each destination, arcs are ordered by origin with lower numbered origins appearing first. This particular ordering scheme did not seem most appropriate for the current application. A post-processor/report generator was written to alleviate these difficulties.

The post-processor generates, at the user's option, any or all of three reports. In all reports, arcs are identified by 6-character names for both the origin and destination nodes. Both the flow and the time periods of the flow are printed. In the first report, the order in which the arcs are listed is by destination, exactly as described above. In the second report, the arcs are ordered by origin, i.e. all arcs with the same origin appear together in order of increasing destination. The order in which the origins occur is the order in which they would first appear in the first report. The third report lists, for each static arc with non-zero flow in at least one time period, the total flow over all time periods and the time period and amount of flow of the latest flow over that arc. The third report was found to be most useful in analyzing the results from the example problem. Either of the first two reports is helpful in tracing the time sequence of events of interest as represented by the dynamic model. In addition to the reports, the post-processor also echos all of the summary information produced by GNET. This information includes the number of nodes and arcs in the network, the total supplies and demands (people to be evacuated), the solution time, the number of "pivots" performed by the optimization algorithm, and the total cost, i.e. the optimal objective-function value.

The post-processor requires, as input, the output from GNET plus the node data necessary to determine node names and time periods. These data consist of the number of nodes in the static network and the following three pieces of information for each node N in the static network: LOWER(N) is the node number in the time-expanded network of copy 0 of node N. OFFSET(N) is a value used to determine the time period in the following way: if node K in the time-expanded network is some copy of node N in the static network, then the time period which K represents is calculated as K-OFFSET(N)+1 when K is an origin and as K-OFFSET when K is a destination. NAME(N) is the six-character name for node N. The number of nodes is read in as a single record in I5 format. For the remainder of the data there is one record per node with format (I5, 5X, I5, 1X, A6).

POSTPRO consists of a main program plus one subroutine which performs a binary search to locate each node in the time-expanded network with respect to the node it represents in the static network. A complete listing of the post-processor appears in the Appendix.

Illustrative Input and Output

The use of the programs discussed previously is perhaps best illustrated by examining the input and output data for the hypothetical three-story building described in Section 2.0. (Throughout this discussion the reader may find it useful to refer to Figure 20 and Figures 4-7.)

Each node was identified by the two-character name shown in Figure 4, except that the "W" was dropped from the name of each stairwell node and lobby and exit were named "LY" and "EX" respectively. The super source and super sink nodes were identified as "SO" and "SI" respectively. Using these abbreviated node names, arcs between two nodes were given four-character names with the first two characters being the name of the origin node and the second two characters being the name of the destination node. The node numbering scheme (refer to Figure 20) assigned numbers 1-6 to the 6 copies of W3, 7-14 to the 8 copies of W2, 15-25 for W1, 26-31 for H3, 32-39 for H2, 40-45 for SW3, 46-53 for SW2, 54-64 for LY, and 65-75 for EX. The super source and super sink become nodes 76 and 77 respectively.

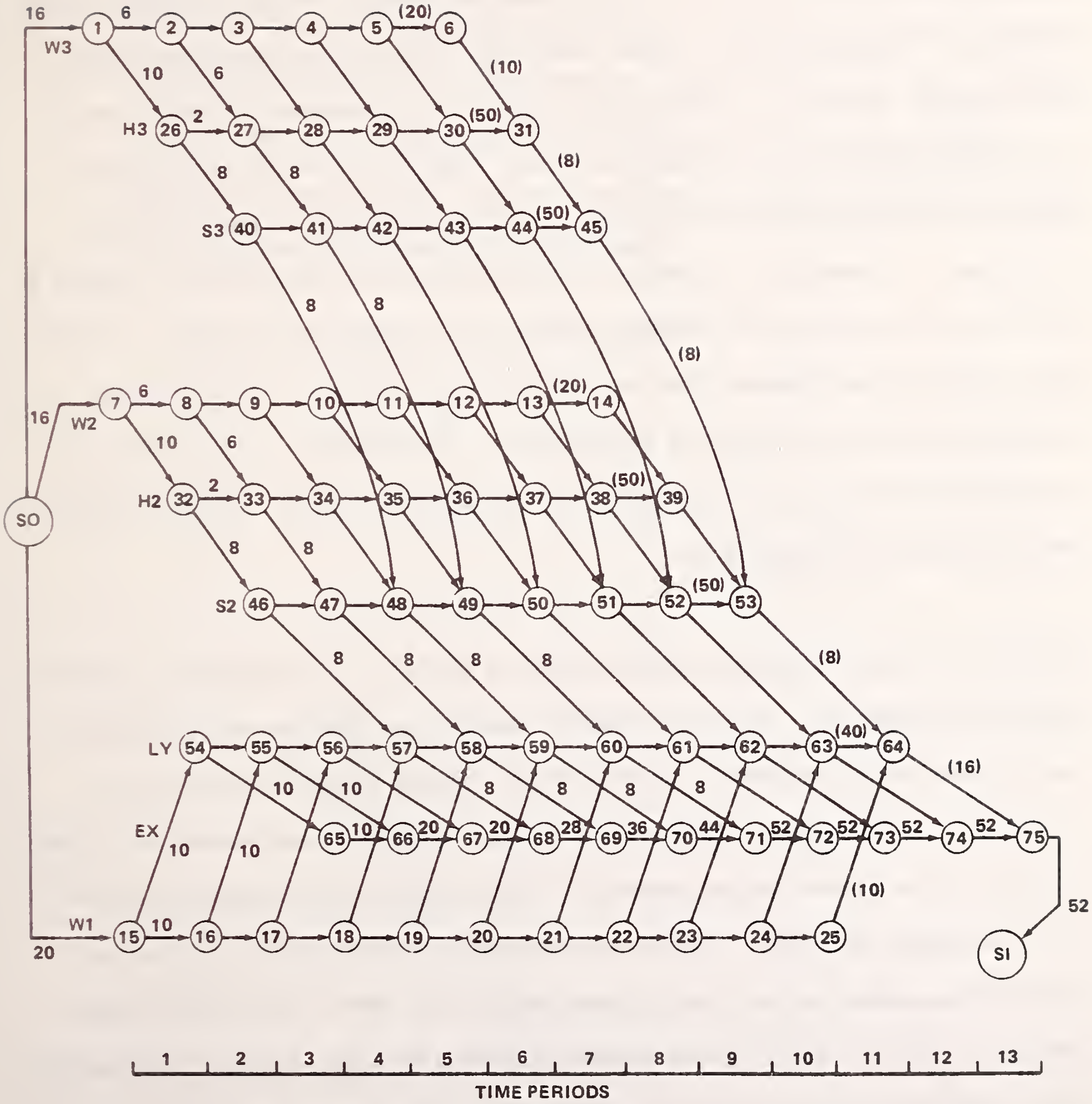


Figure 20 **DYNAMIC NETWORK AND SOLUTION FROM GNET FOR SIMPLE THREE FLOOR BUILDING**

Figure 4 illustrates the static network of 9 nodes and 8 arcs. Although GNET implicitly generates the two super nodes, arcs to and from these super nodes must be included in the input arc data. For the example, there is one arc leading from the super source to each of the three work areas. The capacity on each of these arcs is equal to the number of people initially located at the associated workplace. There is also an arc from the exit to the super sink with capacity of 52 (the number of people in the building). In addition, there is a hold-over arc for each node in Figure 4. Thus for the total static network there are 11 nodes and 21 arcs. For all arcs, except the holdover arcs at the exit, the costs are zero and the capacities are as illustrated in Figure 4. The holdover arcs at the exit are assigned costs of -1 through -10. The dynamic network and the solution are illustrated in Figure 20.

The input data for program SETUP appear in Table 4. The first row (record) gives the number of arcs in the static network and the number of nodes in the time-expanded network. The remaining records describe each of the 21 arcs by name, origin, destination, costs, capacity, and number of copies needed in the time-expanded network. Table 5 shows the output from SETUP, i.e. the input for GNET. The first record gives the number of nodes in the time-expanded network and a print switch for GNET. The other records describe each arc in the time-expanded network by name, origin, destination, cost, and capacity. The user should note that the exceptions to the rules for expanding the network, in this case the costs on the arcs named EXEX, were all dealt with using an editing routine after SETUP was executed.

The output from GNET, i.e. the input to POSTPRO, appears in Table 6. The additional data needed by POSTPRO to associate node numbers with node names and time periods appear in Table 7. The first record in Table 7 is the number of nodes. The remaining records give for each node the lowest number which represents that node, an offset value used to determine time periods, and the node name. Finally, Tables 8-10 give the three reports generated by POSTPRO.

```
      21    75
   W3W3          1         2            0          20          5
   W2W2          7         8            0          20          7
   W1W1         15        16            0          40         10
   H3H3         26        27            0          50          5
   H2H2         32        33            0          50          7
   S3S3         40        41            0          50          5
   S2S2         46        47            0          50          7
   L1L1         54        55            0          40         10
   EXEX         65        66           -1       99999         10
   SOW3         76         1            0          16          1
   SOW2         76         7            0          16          1
   SOW1         76        15            0          20          1
   EXSI         75        77            0          52          1
   W3H3          1        26            0          16          6
   W2H2          7        32            0          16          8
   W1L1         15        54            0          16         11
   H3S3         26        40            0           8          6
   H2S2         32        46            0           8          8
   S3S2         40        48            0           8          6
   S2L1         46        57            0           8          8
   L1EX         54        65            0          16         11
   @EOF
```

Table 4. Input to SETUP

```
 75      1
W3W3       1      2            0        20
W3W3       2      3            0        20
W3W3       3      4            0        20
W3W3       4      5            0        20
W3W3       5      6            0        20
W2W2       7      8            0        20
W2W2       8      9            0        20
W2W2       9     10            0        20
W2W2      10     11            0        20
W2W2      11     12            0        20
W2W2      12     13            0        20
W2W2      13     14            0        20
W1W1      15     16            0        40
W1W1      16     17            0        40
W1W1      17     18            0        40
W1W1      18     19            0        40
W1W1      19     20            0        40
W1W1      20     21            0        40
W1W1      21     22            0        40
W1W1      22     23            0        40
W1W1      23     24            0        40
W1W1      24     25            0        40
H3H3      26     27            0        50
H3H3      27     28            0        50
H3H3      28     29            0        50
H3H3      29     30            0        50
H3H3      30     31            0        50
H2H2      32     33            0        50
H2H2      33     34            0        50
H2H2      34     35            0        50
H2H2      35     36            0        50
H2H2      36     37            0        50
H2H2      37     38            0        50
H2H2      38     39            0        50
S3S3      40     41            0        50
S3S3      41     42            0        50
S3S3      42     43            0        50
S3S3      43     44            0        50
S3S3      44     45            0        50
S2S2      46     47            0        50
S2S2      47     48            0        50
S2S2      48     49            0        50
S2S2      49     50            0        50
S2S2      50     51            0        50
S2S2      51     52            0        50
S2S2      52     53            0        50
L1L1      54     55            0        40
L1L1      55     56            0        40
L1L1      56     57            0        40
L1L1      57     58            0        40
L1L1      58     59            0        40
L1L1      59     60            0        40
L1L1      60     61            0        40
L1L1      61     62            0        40
L1L1      62     63            0        40
L1L1      63     64            0        40
EXEX      65     66           -1     99999
```

Table 5. Output from SETUP---Input to GNET

EXEX	66	67	-2	99999
EXEX	67	68	-3	99999
EXEX	68	69	-4	99999
EXEX	69	70	-5	99999
EXEX	70	71	-6	99999
EXEX	71	72	-7	99999
EXEX	72	73	-8	99999
EXEX	73	74	-9	99999
EXEX	74	75	-10	99999
SOW3	76	1	0	16
SOW2	76	7	0	16
SOW1	76	15	0	20
EXSI	75	77	0	52
W3H3	1	26	0	10
W3H3	2	27	0	10
W3H3	3	28	0	10
W3H3	4	29	0	10
W3H3	5	30	0	10
W3H3	6	31	0	10
W2H2	7	32	0	10
W2H2	8	33	0	10
W2H2	9	34	0	10
W2H2	10	35	0	10
W2H2	11	36	0	10
W2H2	12	37	0	10
W2H2	13	38	0	10
W2H2	14	39	0	10
W1L1	15	54	0	10
W1L1	16	55	0	10
W1L1	17	56	0	10
W1L1	18	57	0	10
W1L1	19	58	0	10
W1L1	20	59	0	10
W1L1	21	60	0	10
W1L1	22	61	0	10
W1L1	23	62	0	10
W1L1	24	63	0	10
W1L1	25	64	0	10
H3S3	26	40	0	8
H3S3	27	41	0	8
H3S3	28	42	0	8
H3S3	29	43	0	8
H3S3	30	44	0	8
H3S3	31	45	0	8
H2S2	32	46	0	8
H2S2	33	47	0	8
H2S2	34	48	0	8
H2S2	35	49	0	8
H2S2	36	50	0	8
H2S2	37	51	0	8
H2S2	38	52	0	8
H2S2	39	53	0	8
S3S2	40	48	0	8
S3S2	41	49	0	8
S3S2	42	50	0	8
S3S2	43	51	0	8
S3S2	44	52	0	8
S3S2	45	53	0	8

Table 5 (Continued)

S2L1	46	57	0	8
S2L1	47	58	0	8
S2L1	48	59	0	8
S2L1	49	60	0	8
S2L1	50	61	0	8
S2L1	51	62	0	8
S2L1	52	63	0	8
S2L1	53	64	0	8
L1EX	54	65	0	16
L1EX	55	66	0	16
L1EX	56	67	0	16
L1EX	57	68	0	16
L1EX	58	69	0	16
L1EX	59	70	0	16
L1EX	60	71	0	16
L1EX	61	72	0	16
L1EX	62	73	0	16
L1EX	63	74	0	16
L1EX	64	75	0	16

Table 5 (Continued)

```
     75 NODES

TOTAL SUPPLIES =      52     TOTAL DEMANDS =      52     INITIAL COST =  0.
    130 ARCS

TIME=      .09 SECONDS, PIVOTS=     82, NNE=  32, NNS=  56, IPG=    8

FINAL SOLUTION...

FROM        TO          FLOW
   1         2             6
   7         8             6
  15        16            10
   1        26            10
  26        27             2
   2        27             6
   7        32            10
  32        33             2
   8        33             6
  26        40             8
  27        41             8
  32        46             8
  33        47             8
  40        48             8
  41        49             8
  15        54            10
  16        55            10
  46        57             8
  47        58             8
  48        59             8
  49        60             8
  54        65            10
  65        66            10
  55        66            10
  66        67            20
  67        68            20
  57        68             8
  68        69            28
  58        69             8
  69        70            36
  59        70             8
  70        71            44
  60        71             8
  71        72            52
  72        73            52
  73        74            52
  74        75            52
 COST=        -2434.
```

Table 6. Output from GNET---Input to POSTPRO

```
     9
     1          1 W3
     7          7 W2
    15         15 W1
    26         25 H3
    32         31 H2
    40         38 S3
    46         44 S2
    54         53 L1
    65         62 EX
```

Table 7. Input to POSTPRO

```
                    ARCS ORDERED BY DESTINATION

 6 PEOPLE FROM W3     TO W3      IN PERIODS    1 THROUGH    1
 6 PEOPLE FROM W2     TO W2      IN PERIODS    1 THROUGH    1
10 PEOPLE FROM W1     TO W1      IN PERIODS    1 THROUGH    1
10 PEOPLE FROM W3     TO H3      IN PERIODS    1 THROUGH    1
 2 PEOPLE FROM H3     TO H3      IN PERIODS    2 THROUGH    2
 6 PEOPLE FROM W3     TO H3      IN PERIODS    2 THROUGH    2
10 PEOPLE FROM W2     TO H2      IN PERIODS    1 THROUGH    1
 2 PEOPLE FROM H2     TO H2      IN PERIODS    2 THROUGH    2
 6 PEOPLE FROM W2     TO H2      IN PERIODS    2 THROUGH    2
 8 PEOPLE FROM H3     TO S3      IN PERIODS    2 THROUGH    2
 8 PEOPLE FROM H3     TO S3      IN PERIODS    3 THROUGH    3
 8 PEOPLE FROM H2     TO S2      IN PERIODS    2 THROUGH    2
 8 PEOPLE FROM H2     TO S2      IN PERIODS    3 THROUGH    3
 8 PEOPLE FROM S3     TO S2      IN PERIODS    3 THROUGH    4
 8 PEOPLE FROM S3     TO S2      IN PERIODS    4 THROUGH    5
10 PEOPLE FROM W1     TO L1      IN PERIODS    1 THROUGH    1
10 PEOPLE FROM W1     TO L1      IN PERIODS    2 THROUGH    2
 8 PEOPLE FROM S2     TO L1      IN PERIODS    3 THROUGH    4
 8 PEOPLE FROM S2     TO L1      IN PERIODS    4 THROUGH    5
 8 PEOPLE FROM S2     TO L1      IN PERIODS    5 THROUGH    6
 8 PEOPLE FROM S2     TO L1      IN PERIODS    6 THROUGH    7
10 PEOPLE FROM L1     TO EX      IN PERIODS    2 THROUGH    3
10 PEOPLE FROM EX     TO EX      IN PERIODS    4 THROUGH    4
10 PEOPLE FROM L1     TO EX      IN PERIODS    3 THROUGH    4
20 PEOPLE FROM EX     TO EX      IN PERIODS    5 THROUGH    5
20 PEOPLE FROM EX     TO EX      IN PERIODS    6 THROUGH    6
 8 PEOPLE FROM L1     TO EX      IN PERIODS    5 THROUGH    6
28 PEOPLE FROM EX     TO EX      IN PERIODS    7 THROUGH    7
 8 PEOPLE FROM L1     TO EX      IN PERIODS    6 THROUGH    7
36 PEOPLE FROM EX     TO EX      IN PERIODS    8 THROUGH    8
 8 PEOPLE FROM L1     TO EX      IN PERIODS    7 THROUGH    8
44 PEOPLE FROM EX     TO EX      IN PERIODS    9 THROUGH    9
 8 PEOPLE FROM L1     TO EX      IN PERIODS    8 THROUGH    9
52 PEOPLE FROM EX     TO EX      IN PERIODS   10 THROUGH   10
52 PEOPLE FROM EX     TO EX      IN PERIODS   11 THROUGH   11
52 PEOPLE FROM EX     TO EX      IN PERIODS   12 THROUGH   12
52 PEOPLE FROM EX     TO EX      IN PERIODS   13 THROUGH   13
```

Table 8. First Output Report from POSTPRO

```
                         ARCS ORDERED BY ORIGIN

     6 PEOPLE FROM W3        TO W3       IN PERIODS      1 THROUGH      1
    10 PEOPLE FROM W3        TO H3       IN PERIODS      1 THROUGH      1
     6 PEOPLE FROM W3        TO H3       IN PERIODS      2 THROUGH      2

     6 PEOPLE FROM W2        TO W2       IN PERIODS      1 THROUGH      1
    10 PEOPLE FROM W2        TO H2       IN PERIODS      1 THROUGH      1
     6 PEOPLE FROM W2        TO H2       IN PERIODS      2 THROUGH      2

    10 PEOPLE FROM W1        TO W1       IN PERIODS      1 THROUGH      1
    10 PEOPLE FROM W1        TO L1       IN PERIODS      1 THROUGH      1
    10 PEOPLE FROM W1        TO L1       IN PERIODS      2 THROUGH      2

     2 PEOPLE FROM H3        TO H3       IN PERIODS      2 THROUGH      2
     8 PEOPLE FROM H3        TO S3       IN PERIODS      2 THROUGH      2
     8 PEOPLE FROM H3        TO S3       IN PERIODS      3 THROUGH      3

     2 PEOPLE FROM H2        TO H2       IN PERIODS      2 THROUGH      2
     8 PEOPLE FROM H2        TO S2       IN PERIODS      2 THROUGH      2
     8 PEOPLE FROM H2        TO S2       IN PERIODS      3 THROUGH      3

     8 PEOPLE FROM S3        TO S2       IN PERIODS      3 THROUGH      4
     8 PEOPLE FROM S3        TO S2       IN PERIODS      4 THROUGH      5

     8 PEOPLE FROM S2        TO L1       IN PERIODS      3 THROUGH      4
     8 PEOPLE FROM S2        TO L1       IN PERIODS      4 THROUGH      5
     8 PEOPLE FROM S2        TO L1       IN PERIODS      5 THROUGH      6
     8 PEOPLE FROM S2        TO L1       IN PERIODS      6 THROUGH      7

    10 PEOPLE FROM L1        TO EX       IN PERIODS      2 THROUGH      3
    10 PEOPLE FROM L1        TO EX       IN PERIODS      3 THROUGH      4
     8 PEOPLE FROM L1        TO EX       IN PERIODS      5 THROUGH      6
     8 PEOPLE FROM L1        TO EX       IN PERIODS      6 THROUGH      7
     8 PEOPLE FROM L1        TO EX       IN PERIODS      7 THROUGH      8
     8 PEOPLE FROM L1        TO EX       IN PERIODS      8 THROUGH      9

    10 PEOPLE FROM EX        TO EX       IN PERIODS      4 THROUGH      4
    20 PEOPLE FROM EX        TO EX       IN PERIODS      5 THROUGH      5
    20 PEOPLE FROM EX        TO EX       IN PERIODS      6 THROUGH      6
    28 PEOPLE FROM EX        TO EX       IN PERIODS      7 THROUGH      7
    36 PEOPLE FROM EX        TO EX       IN PERIODS      8 THROUGH      8
    44 PEOPLE FROM EX        TO EX       IN PERIODS      9 THROUGH      9
    52 PEOPLE FROM EX        TO EX       IN PERIODS     10 THROUGH     10
    52 PEOPLE FROM EX        TO EX       IN PERIODS     11 THROUGH     11
    52 PEOPLE FROM EX        TO EX       IN PERIODS     12 THROUGH     12
    52 PEOPLE FROM EX        TO EX       IN PERIODS     13 THROUGH     13
```

Table 9. Second Output Report from POSTPRO

```
                  REPORT ON TOTAL AND LAST FLOW OVER EACH DISTINCT ARC

      ***** ARC ******      TOTAL FLOW       *********** LAST FLOW ************


      W3      TO W3              6            6 IN PERIODS     1 THROUGH      1
      W3      TO H3             16            6 IN PERIODS     2 THROUGH      2
      W2      TO W2              6            6 IN PERIODS     1 THROUGH      1
      W2      TO H2             16            6 IN PERIODS     2 THROUGH      2
      W1      TO W1             10           10 IN PERIODS     1 THROUGH      1
      W1      TO L1             20           10 IN PERIODS     2 THROUGH      2
      H3      TO H3              2            2 IN PERIODS     2 THROUGH      2
      H3      TO S3             16            8 IN PERIODS     3 THROUGH      3
      H2      TO H2              2            2 IN PERIODS     2 THROUGH      2
      H2      TO S2             16            8 IN PERIODS     3 THROUGH      3
      S3      TO S2             16            8 IN PERIODS     4 THROUGH      5
      S2      TO L1             32            8 IN PERIODS     6 THROUGH      7
      L1      TO EX             52            8 IN PERIODS     8 THROUGH      9
      EX      TO EX            366           52 IN PERIODS    13 THROUGH     13
```

Table 10. Third Output Report from POSTPRO

6.0 CONCLUSIONS AND NEXT STEPS

In this section we summarize our previous discussion, draw certain conclusions, and discuss appropriate next steps in the development of this network analysis approach.

Model Descriptors, Capabilities, and Limitations

As the key to our work is the dynamic model, we first list, and discuss one by one, some descriptors of the model which should be kept in mind in understanding its capabilities and limitations. The descriptors are as follows:

1. Prescriptive (and not behavioral)
2. Linear
3. Deterministic
4. Discrete Time
5. Computer Reliant
6. "Perfect knowledge," or "Forecast," of "future"

We have seen that the dynamic model finds a routing of people through the building from the workplaces to the exits that minimizes the time to evacuate the building (actually by minimizing turnstile costs as a surrogate measure for time to evacuate the building.) As such, the model prescribes, or recommends, an "ideal" course of action, providing a standard of comparison, or benchmark. (For example, an individual responsible for running evacuation drills of Building 101 would be able to judge how well he had done by comparing his time to evacuate the building with the time predicted by the model.) It is difficult to represent behavioral aspects in the model, although in some cases the initial design of the model may reflect some behavioral aspects, such as, for example, not permitting the use of side exits in the model due to the fact that they are seldom used in reality. Nevertheless, it should be realized that the model is really not a behavioral model, and is thus limited to the extent that behavioral aspects are important. For example, the model cannot be used to predict the time lapse between individuals hearing a fire alarm and actually beginning to evacuate the building.

The dynamic model is linear in its mathematics. This means, for example, that for each node, the total flow into that node is equal to the total flow out; for each arc, the flow in that arc must lie between zero and a fixed arc capacity. Then the model minimizes a linear cost involving turnstiles, computed by multiplying cost per turnstile arc by the arc flow and adding the resultant costs over all turnstile arcs. It should be recognized that in reality there are some nonlinearities which occur in building evacuation, e.g., in connection with congestion phenomena. For example, in a very crowded stairwell the flow rate is smaller than in an uncrowded stairwell. An ideal evacuation model would reflect this fact by having arc flow rate capacities vary suitably with arc flows, and not be independent of arc flows. Further, with reference to the static model, arc traversal times are chosen independently of arc capacities and of arc flows. In reality the time to traverse, for example, a stairwell, will depend upon how crowded the stairwell is.

Although reality presents some nonlinearities that the current network models do not reflect, one should keep in mind that the use of modeling techniques always involves a tradeoff between realism and tractability, i.e., the ease of obtaining problem insights from the model. The more realistic a model is, the less computationally tractable it is, and the more difficult it is to obtain problem insights by using the model. The linearity of the dynamic model permits the modeling of problems with literally hundreds of thousands of arcs, a necessary capability, considering the possible size of dynamic models of large buildings. We believe that models should be kept as simple as possible consistent with their use, and that linear models should be exploited in full before nonlinear models are considered. In particular, it does not yet seem appropriate to consider introducing nonlinearities into the network flow models.

There can be, of course, a great deal of uncertainty associated with building evacuation, including random variation in flow rates and arc traversal times, particularly if the evacuation takes place as the result of a fire or other unnerving event. The fact that our current network models are deterministic, that is, do not recognize probabilistic variation, is unquestionably a limitation, and a suitable topic for further research. We point out again, however, that possible gains in realism obtained by reflecting randomness in the model may well be offset by the resultant additional complexity and loss in computational tractability.

Ideally, one would like a model in which time is treated as continuous; there are difficulties and approximations which are intrinsic to the use of discrete time periods. In theory, one could construct dynamic network flow models where each time interval is as short as necessary--one second, for example. The resultant difficulty, of course, is an increase in the number of arcs of the network, causing the solution of the dynamic network model to require more computer time and "memory space" than would be the case if time periods were greater than a second. Again there is a trade-off between model realism and model tractability.

It is impossible, of course, to solve large network flow problems without use of the computer. There is some hope, however, of developing smaller, auxiliary models, which may not be computer reliant. One such model is given in Appendix B of this report; the model addresses the question of the allocation of people to stairwells that will minimize the time to evacuate the building. Nevertheless, there appears to be at present little prospect of having a model as flexible and general as the dynamic network flow model, that could be easily used by individuals such as building inspectors or fire department personnel who would not have ready access to computers. Ideally, one would like to provide such individuals with handbooks of graphs and charts they could use, and conceivably this may eventually be possible as further experience is gained in modeling evacuation problems, but such is not yet the case.

It is easy to fall into the habit of thinking of the dynamic model as one which evolves in time, with the first time period being worked with first, the second time period being worked with second, etc. Actually the model does not work this way at all: it operates on a representation of the building evacuation problem which simultaneously considers all the time periods of interest. At the outset, the model "knows" the relevant capacity and travel time information for all time periods. In this sense, the model assumes perfect knowledge of the future. Alternatively, one can consider that the model works with a forecast of the future, in which case it follows, of course, that the reliability of the model outputs is dependent upon the quality of the forecast. This dependency can, and should, be lessened by working with a number of different forecasts, but can never be completely avoided.

Using the Building 101 Model.

Next, we address the question of the use of our models by others. The particular dynamic model we have developed is specific to Building 101, and would not be directly useful to others unless they were interested in an identical building. However, the procedures we have developed for constructing the static and dynamic models, together with computer programs given in Section 5.0 for facilitating the use of these models, should be helpful to others wishing to construct dynamic network flow models of building evacuation. In particular, the Static Model Construction Procedure, and the Dynamic Model Construction Procedure presented earlier, should be directly useful in modeling building evacuations, regardless of the building of interest.

In Perspective, and Next Steps

It may perhaps be helpful at this point to offer a perspective on our work. To this end, consider Figure 21. As the figure illustrates, having people in buildings leads to evacuation problems. These problems may be studied in many ways. The way we have chosen involves constructing network models of a particular building, which has served as a prototype building for our initial work. We first construct a static network model, and then expand it to obtain a dynamic model. Using a computer representation of the dynamic model, we solve it by using a particular transshipment problem algorithm, in the process having to transform the computer representation of the dynamic model from one closely related to the building to one which is in a proper input format for the algorithm. The computer algorithm provides an output of one line of print for every arc of the dynamic network in which there is a positive flow, stating the amount of the flow, and identifying the arc by a pair of numbers which specify the nodes at the tail and head of the

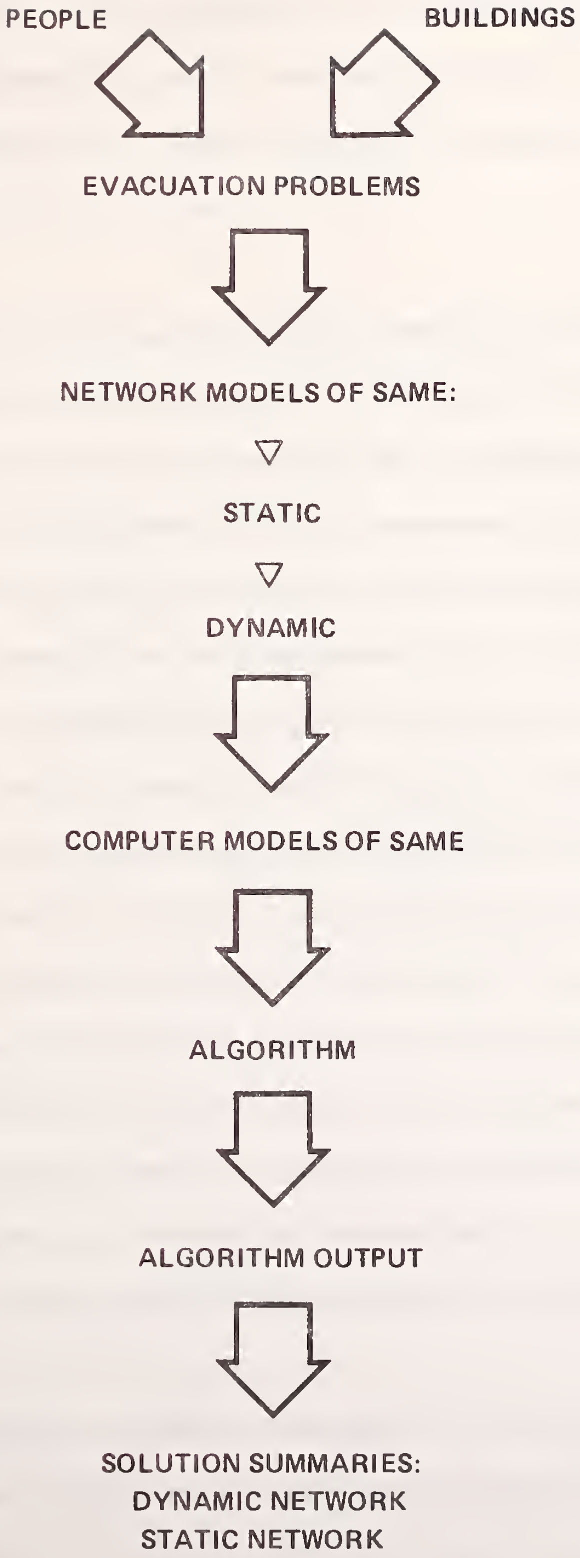


Figure 21 IN PERSPECTIVE

arc. It is then necessary to transform the algorithm output into an output directly related to the building. Such outputs can be dynamic as well as static. The run summaries given in Appendix A illustrate such static output, and details of the output formats appear in Section 6.0.

To this point the programs we have developed are somewhat building-specific, while the computer code of the solution procedure we have used is one made available to us as a professional courtesy, and available to others only with the approval of its developers. It should thus be evident that our procedures at this point are certainly not sufficiently general that they can be taken "off the shelf" and used by others for modeling the evacuation of buildings of interest to them. What should be useful, however, is the modeling process we have developed. Others having the necessary familiarity with computer programming and network flow optimization should find it a relatively direct task to use the procedures we have developed for modeling the evacuation of other buildings. In the long run, of course, what is desired is a computer program system sufficiently general that it can be used by others who do not wish to be concerned with computer programming and network optimization. To this end, therefore, we now discuss the further steps which we believe necessary to make our network modeling approach more useful.

A larger building than Building 101 should be modeled, in order to gain some feel for how large a building can be modeled at a computationally reasonable price. Further, one might want to consider modeling a building for which evacuation data are available, such as, for example, one of the buildings studied by Pauls, in order to compare "optimized" evacuations with observed evacuations.

In conjunction with additional modeling, the development of user-oriented software merits consideration. Automation of the Dynamic Model Construction Procedure would be most helpful, so that general static network models of buildings could be easily converted to the corresponding dynamic network models. Likewise, considerable thought should be given to a "post-processor", more general than the existing one, for converting the output of the transshipment algorithm into a building-related form which can easily provide insight into the output of the dynamic model. In the same vein it appears worthwhile to consider actually animating the output of the dynamic model, perhaps using a Video Display Terminal, so that the model results can be more easily displayed and explained. In addition, the prospect of large scale use of the network modeling approach would merit considering the development of a special-purpose algorithm to solve the dynamic network flow problem more efficiently by exploiting the special structure it has as a result of being obtained by replicating the static model. As an extra note, means of explicitly displaying waiting, such as waiting for elevators as illustrated in Figure 22, should be developed. For Building 101, due to the large ratio of building space to building population, relatively little waiting occurred in the use of the dynamic model, and therefore means of obtaining insight into waiting has perhaps been a bit

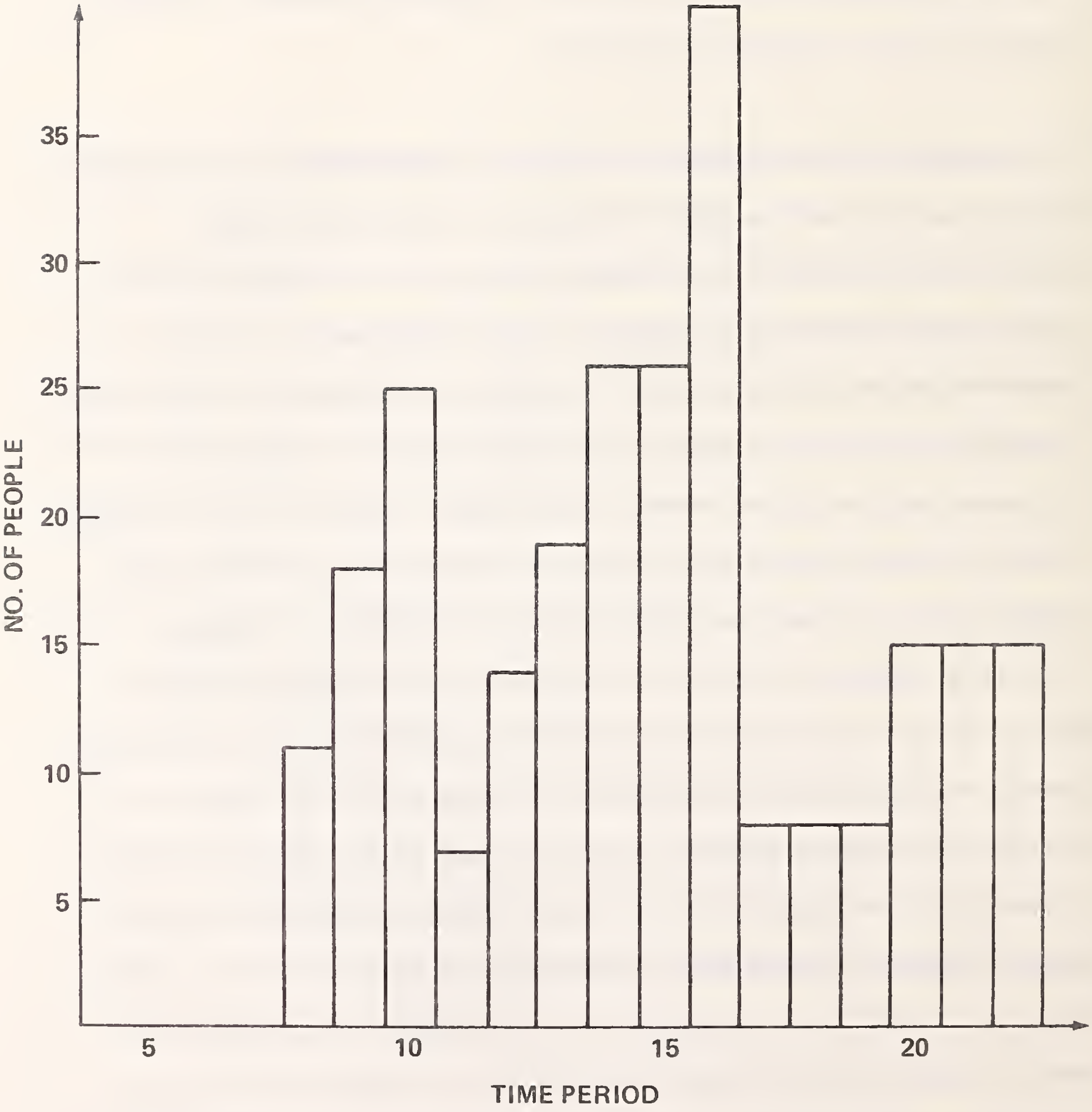


Figure 22 WAITING FOR ELEVATORS ON FIFTH FLOOR, RUN 9

neglected in our work to date.

While a number of computer program and algorithm developments thus merit consideration, there also are questions of modeling and theory to be pursued. For example, it should be possible to develop a number of guidelines to aid in choosing among alternative possibilities in constructing static models: there can easily be more than one static model of a given building, with consequent variations in model accuracy, computational efficiency, and information available from the dynamic model. Further, the fact that buildings tend to fall into various rather distinct categories, and to have common features, suggests it is possible to develop a "catalog" of static model parts, from which a modeler might pick and choose in constructing a static model of a particular building. It has been mentioned above that the current dynamic flow models are deterministic, and employ discrete time intervals. It seems of interest to explore to what extent data uncertainty, or other randomness associated with evacuation problems, would effect the accuracy of the model; hopefully it will be possible to account for randomness, to some extent, directly in future models. The effect of time-interval length upon the accuracy of the dynamic model should be studied. The use of turnstile costing to evacuate a building in a minimum number of time periods has worked well in our experience, but this approach is rather indirect, and one wonders if there may not exist more direct approaches which would work as well and lend themselves more readily to theoretical analysis. As an associated question, given some prediction of the spread of smoke and/or fire in a building, if some meaningful measurement of the resultant undesirability of being in particular locations in a building at particular times can be obtained, these "costs" could be converted to arc costs in the dynamic network model, in which case evacuating the building to minimize the total cost might quite literally suggest a safest building evacuation. (Run numbers 11 through 15 reflect a rough first attempt at attaching costs to being on particular floors during the evacuation of a building.)

There are variations of the building evacuation model which evacuate the maximum number of people from a building in a given number of time periods; see [6], for example. Such models can be solved by algorithms called "maximal dynamic flow" algorithms. In some cases these algorithms actually solve the dynamic problem by solving a variant of the static network problem, with consequent substantial savings in computer run time. In effect, with such maximum flow problems, one assumes an infinite number of people available, allowing them to be in the various workplaces in any manner, and then maximizes the total number of people which can be evacuated from the building in a fixed number of time periods. (The need to assume an arbitrarily large number of people available in the building arises because it may not be clear beforehand how many people can be evacuated.) Our experience with such models has been a bit disappointing. The model evacuates no more people from the upper floors than is a requirement of the problem statement, and routes as many people as possible from the lowest populated floor, uses the stairwells to capacity, and routes the people out of the exits which are the shortest walking times from the stairwell doors. In short, the answers given by the algorithm are obvious by inspection. It may well be, however, that for a more complicated building than Building 101, which has less symmetrical loadings of people on floors with respect to the stairwells, and less symmetrical placements of stairwells in the building, that the solutions would no longer be as obvious. Further, any procedure which can solve dynamic network flow problems without imposing the burden of constructing a dynamic network, merits a certain amount of respect, if nothing else.

Just as aggregation of individual offices into composite workplaces simplifies the development of the network models, one could consider the more drastic step of having composite floors, combining together two or more (pairwise) adjacent floors when they are quite similar. We suspect this procedure may work well; if it does, the computational savings consequent appear attractive. The literature on such approaches (referred to generally as network reduction methods) should be investigated for applicability or adaptability; if nothing relevant is found, devising original procedures for our purposes should be considered.

The problem of finding insightful ways to present the (substantial) output of the dynamic model has been mentioned previously. Use of animation to help in presenting such output, while attractive and promising, is basically a mechanical/electrical approach. We suggest that a more fundamental approach also be investigated. To make this suggestion more concrete, let us return to the maximum flow problem. There is associated theory for this problem which specifies the amount by which the maximum flow could be increased if the flow rate capacity of any arc were to be increased by one unit. Obviously it would be of value to know those arcs whose unit increase in flow rate capacity would cause the greatest increase in the maximum flow, as such an arc is, in a very real sense, "the" bottleneck. The existence of such a theory for the maximum flow problem suggests that analogous theoretical results may be obtainable for the dynamic flow problem of evacuating a building in minimum time. This could permit automating, to some extent, the analysis of the output of the dynamic model, while being assured that critical aspects of the output are identified and considered.

At present, besides the rather detailed static and dynamic network procedures for representing buildings, we also have, at the other extreme, a quite simple model, virtually graphical in nature (as discussed in the Appendix B), which specifies allocations of people to evacuation routes that minimize the time to evacuate the building. The existence of models at these two extremes of complexity naturally suggests that models of intermediate complexity can be developed, which can augment and/or supplement the dynamic network models, while providing more information than the graphical model. The graphical model, in effect, assumes that <u>any</u> allocation of people to evacuation routes is allowable; such an assumption may be invalid, of course, if, for a building evacuation problem certain evacuation routes are available only to certain floors, such as stairwells which serve only a wing of the building, or do not extend to the top floor of a building. At the other extreme, the dynamic network model permits any routing of people to the exits which is permitted by the directions implied by the arcs, taking into account the fact that if a unit of flow wants to exit a building from an exit remote from the workplace where it originates, a substantial travel time will be involved. We believe it will be possible to develop models with intermediate assumptions as to which exits will be accessible from which workplaces, which avoid the naive assumption of complete accessibility at one extreme and the complete detail of "route interconnectedness" at the other extreme, which can be made more readily usable than the dynamic network flow model, and more realistic than the graphical model. We further envision that such models, apart from ones we already have in mind, will begin to suggest themselves in a natural way as further experience is gained in working with the dynamic network model.

REFERENCES

[1] Bellmore, M., Vemuganti, R. R., "On Multi-Commodity Maximal Dynamic Flows," Operations Research, Vol. 21, 10-21, 1973.

[2] Bookbinder, J. H., Sethi, S. P., "The Dynamic Transportation Problem: a Survey," Working Paper 77-12, Faculty of Management Studies, University of Toronto, Toronto, Canada, 1977.

[3] Bradley, G. H., Brown, G. G., Graves, G. W., "Design and Implementation of Large Scale Primal Transportation Algorithms," Management Science, Vol. 24, 1-34, 1977.

[4] Christofides, N., Graph Theory: an Algorithmic Approach, Academic Press, New York, NY, 1975.

[5] Design and Construction of Building Exits, Washington, DC, U. S. Department of Commerce, National Bureau of Standards, Misc. Publication M151, 1935.

[6] Ford, Jr., L. R., Fulkerson, D. R., Flows in Networks, Princeton University Press, Princeton, NJ, 1962.

[7] Fruin, J. J., Pedestrian Planning and Design, Metropolitan Association of Urban Designers and Environmental Planners, Inc., P. O. Box 722, Church Street Station, New York, NY 10008, 1971.

[8] Glover, F., Karney, D., Klingman, D., "Implementation and Computational Comparisons of Primal, Dual and Primal-Dual Computer Codes for Minimum Cost Network Flow Problems," Networks, Vol. 4, 191 & ff., 1974.

[9] Glover, F., Karney, D., Klingman, D., Napier, A., "A Computational Study on Start Procedures, Basis Change Criteria and Solution Algorithms for Transportation Problems," Management Science, 20, 793 & ff., 1974.

[10] Grinold, R. C., Marshall, K. T., Manpower Planning Models, Elsevier North-Holland, Inc., NY, 1977.

[11] Hitchcock, F. L., "The Distribution of a Product from Several Sources to Numerous Localities," Journal of Mathematical Physics, Vol. 20, 224-230, 1941.

[12] Johnson, B. M., Pauls, J. L., "Study of Stair and Elevator Use in Jeanne Mance Building, Ottawa--an Interim Report," National Research Council of Canada, Division of Building Research, Ottawa, Canada, 1978.

[13] Koopmans, T. C., "Optimum Utilization of the Transportation System," Proceedings of the International Statistical Conferences, 1947, Washington, DC (Vol. 5 reprinted as Supplement to Econometrica, Vol. 17, 1949).

[14] Merchant, D. K., Nemhauser, G. L., "A Model and an Algorithm for the Dynamic Traffic Assignment Problem," Transportation Science, Vol. 12, No. 3, 183-199, 1978.

[15] Minieka, E., Optimization Algorithms for Networks and Graphs, Marcel Dekker, Inc., New York, NY, 1978.

[16] Pauls, J. L., Jones, B. K., "Case Studies of Building Evacuations" and "Building Evacuation Findings & Recommendations," submitted for publication in Behavior in Fires, D. Canter (ed.), J. Wiley & Sons, Inc., New York, NY, 1978.

[17] Pauls, J. L., "Management and Movement of Building Occupants in Emergencies," National Research Council of Canada, Division of Building Research, NRCC 16845, DDR Paper No. 788, Ottawa, Canada, September 1977.

[18] Pauls, J. L., "Movement of People in Building Evacuations," Chapter 21 of Human Response to Tall Buildings, D. J. Conway (ed.), Strausburg, PA: Dowden, Hutchinson, & Ross, Community Development Series, Vol. 34, 1977.

[19] Potts, R. P., Oliver, R. M., Flow in Transportation Networks, Academic Press, New York, NY, 1972.

[20] Roytman, M. Y., Principles of Fire Safety Standards for Building Construction, published for the National Bureau of Standards, U. S. Department of Commerce, and the National Science Foundation, Washington, DC by Amerind Publishing Co., Pvt. Ltd., New Delhi, India, 1975.

[21] Stahl, F., Archea, J., "An Assessment of the Technical Literature on Emergency Egress from Buildings," NBSIR 77-1313.

[22] Stahl, F. I., "Simulating Human Behavior in High-Rise Building Fires: Modeling Occupant Movement Through a Fire-Floor form Initial Alert to Safe Egress," NBS-GCR 77-92.

[23] Wagner, H. M., Principles of Operations Research, 2nd Edition, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1975.

[24] Wardrup, J. G., "Some Theoretical Aspects of Road Traffic Research," Proceedings of the Institute of Civil Engineers, Part II, 325-378, 1952.

[25] Wilkenson, W. L., "An Algorithm for Universal Maximal Dynamic Flows in a Network," Operations Research, Vol. 19, No. 7, Nov.-Dec. 1971.

APPENDIX A

GRAPHICAL SUMMARY OF RUNS

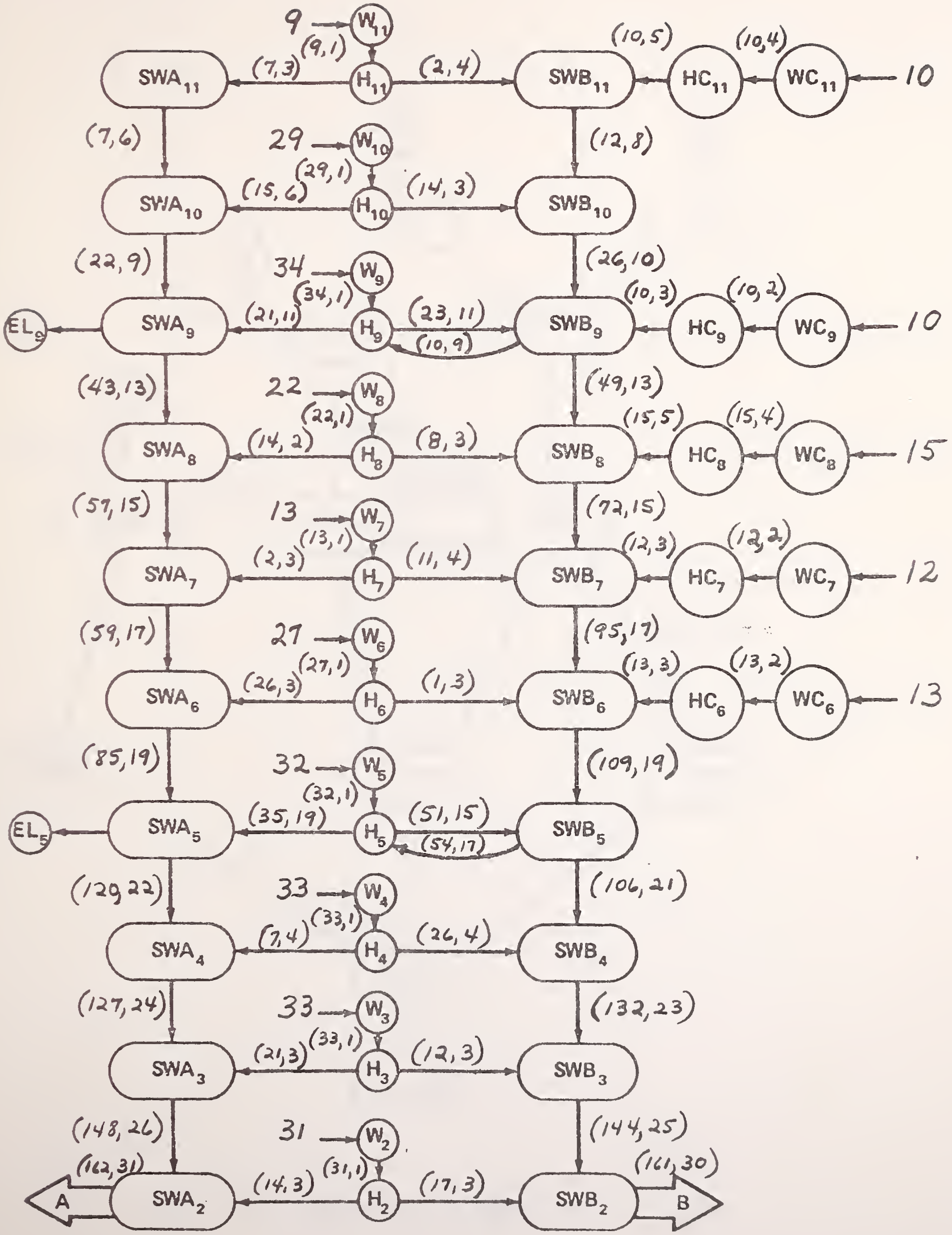


Figure A-1, Run 1 Summary

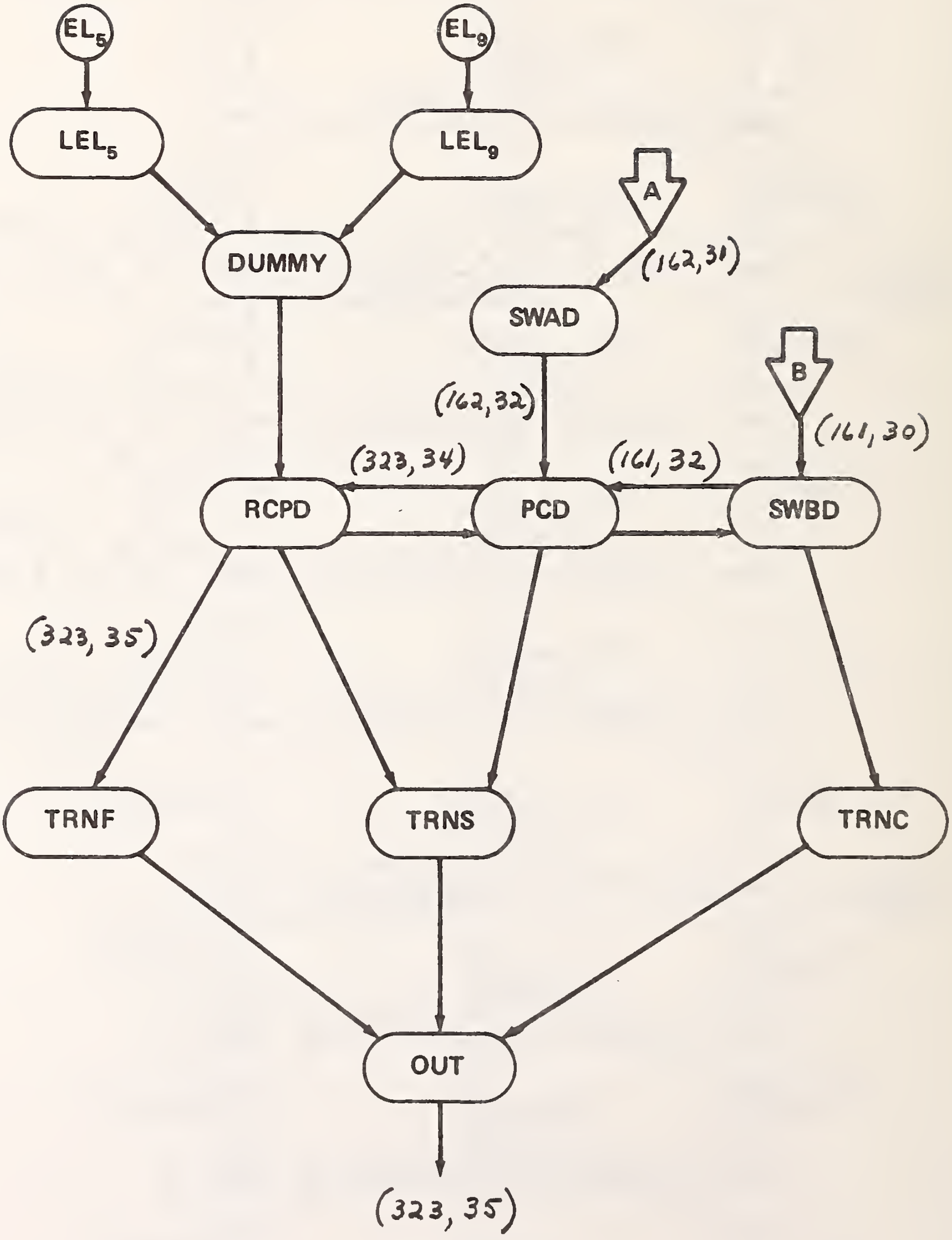


Figure A-1 (Continued)

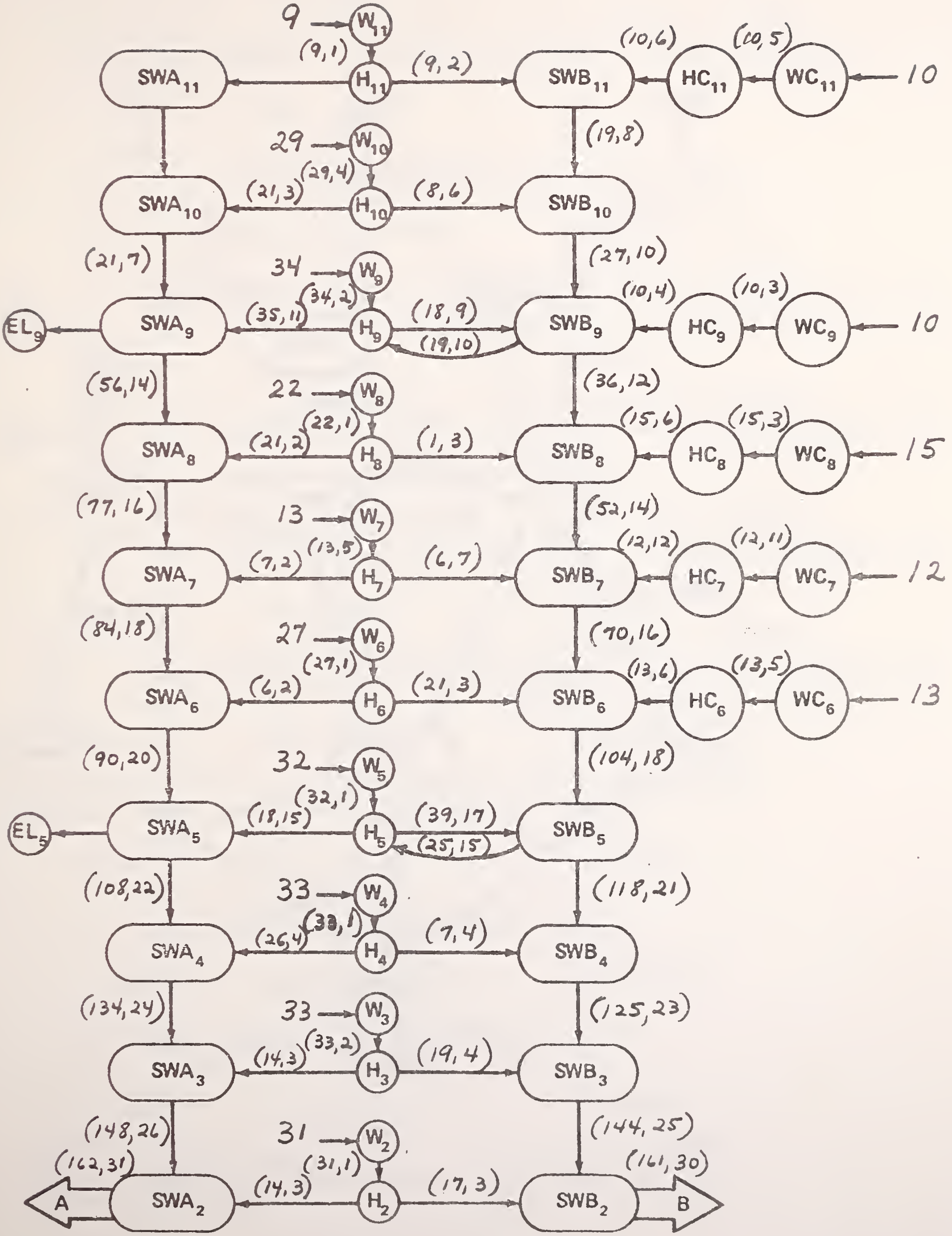


Figure A-2, Run 2 Summary

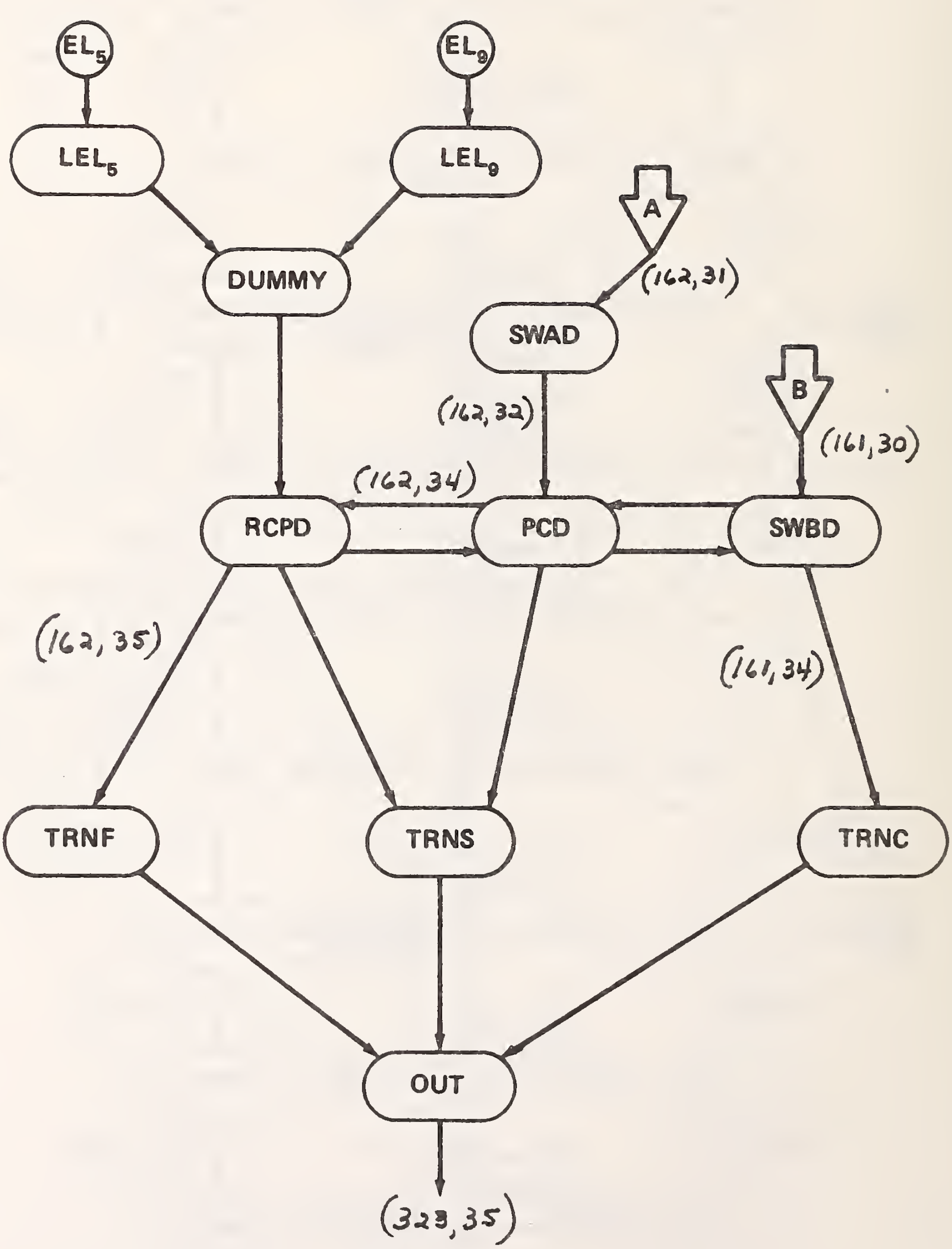


Figure A-2 (Continued)

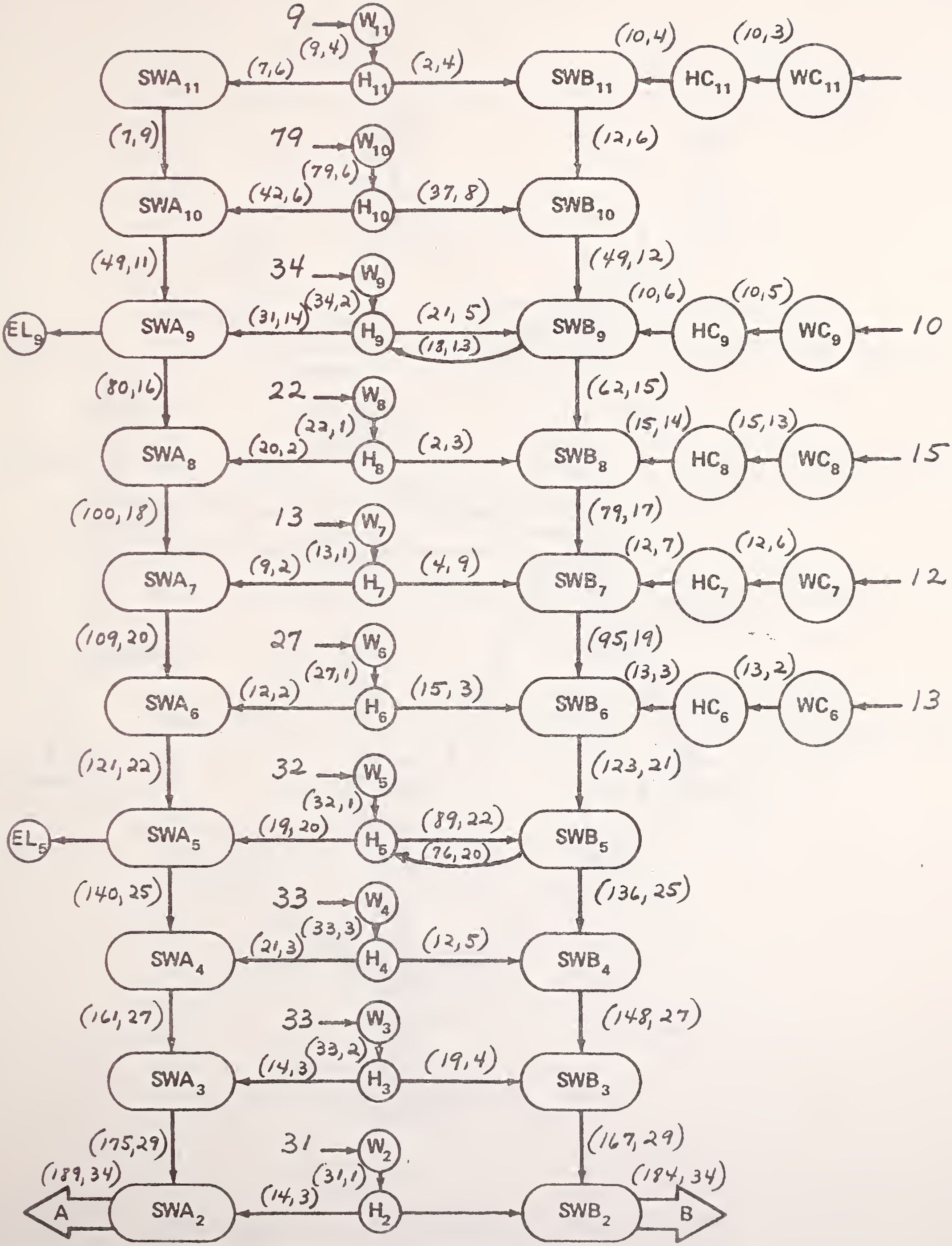


Figure A-3, Run 3 Summary

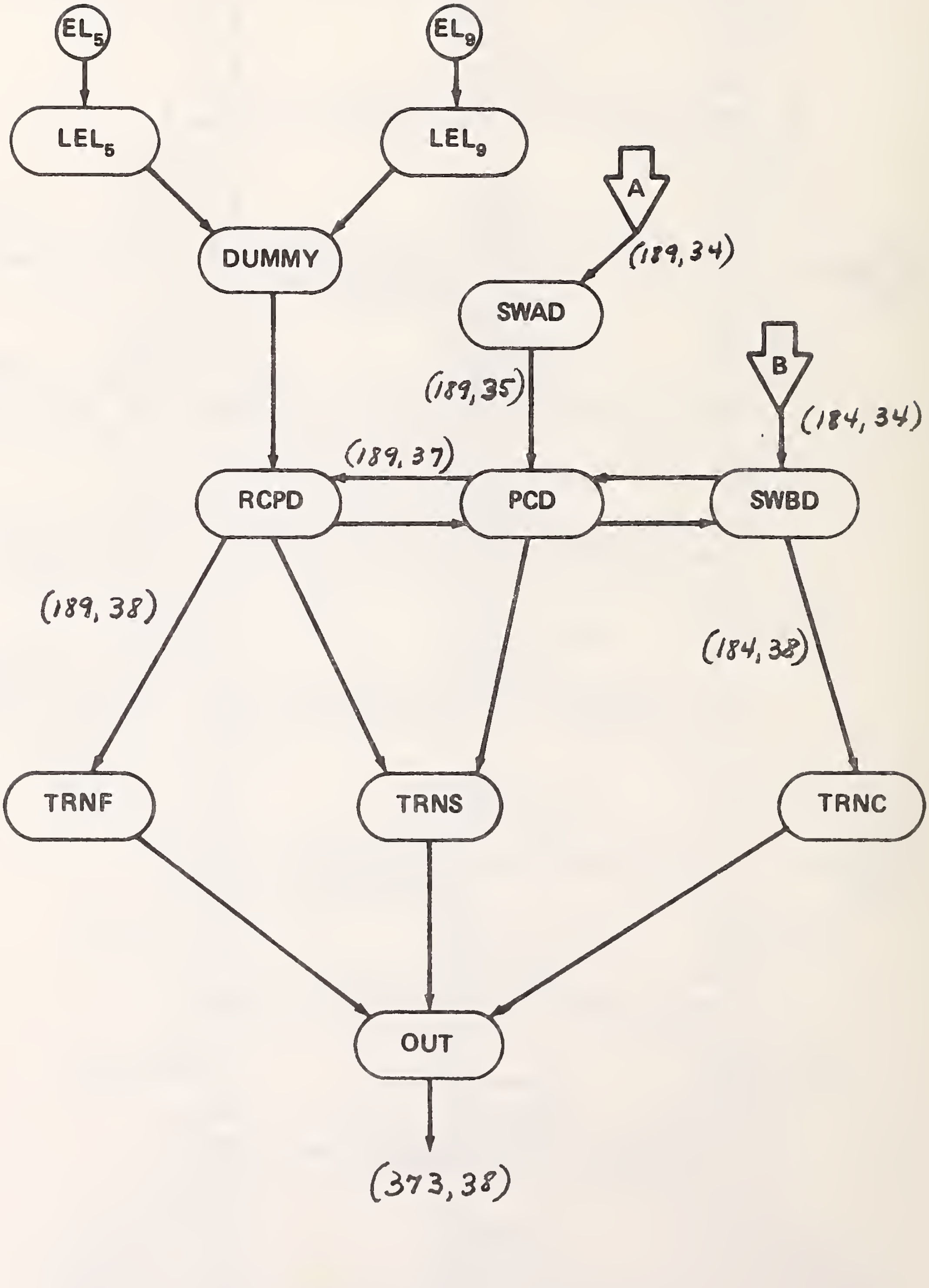


Figure A-3 (Continued)

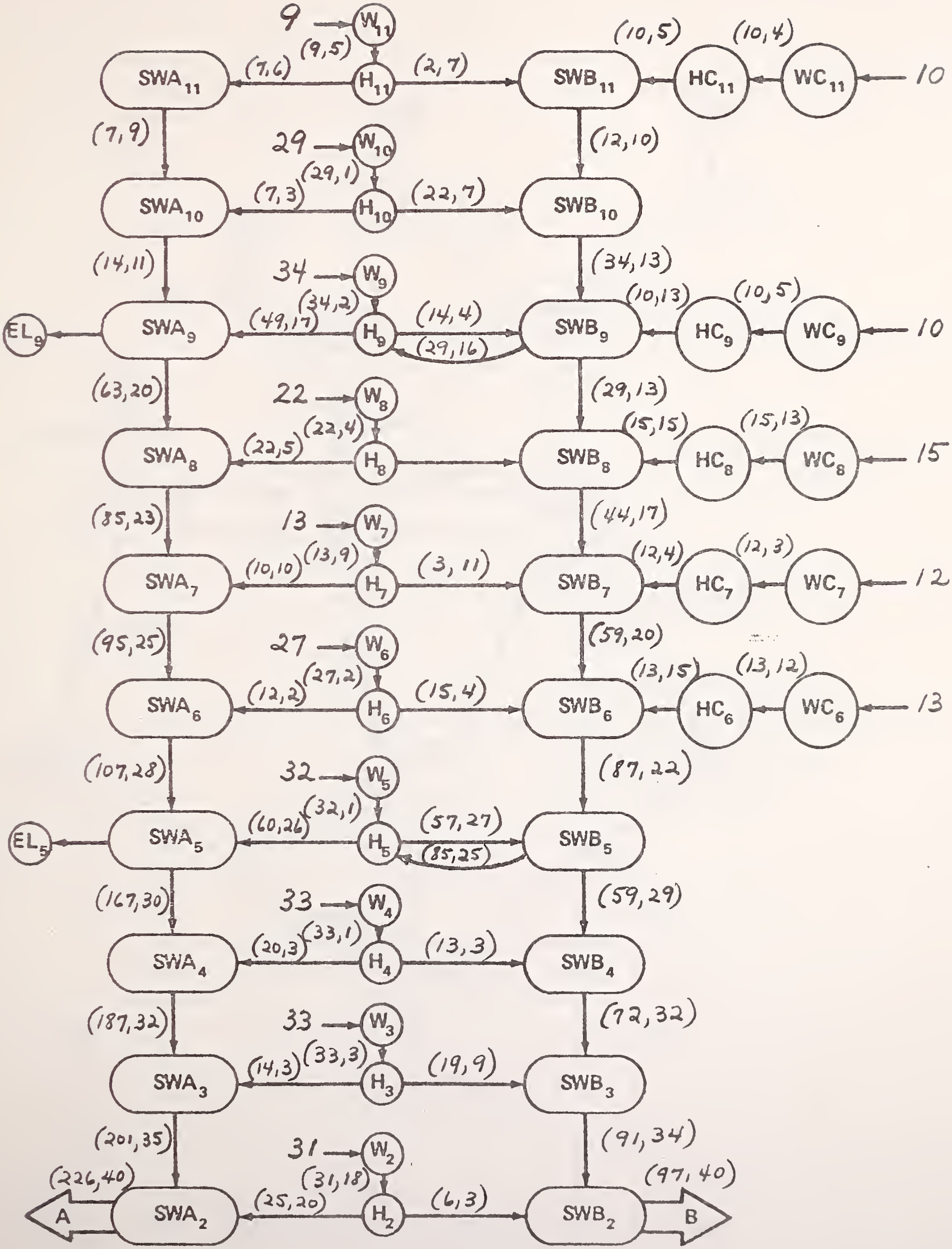


Figure A-4, Run 4 Summary

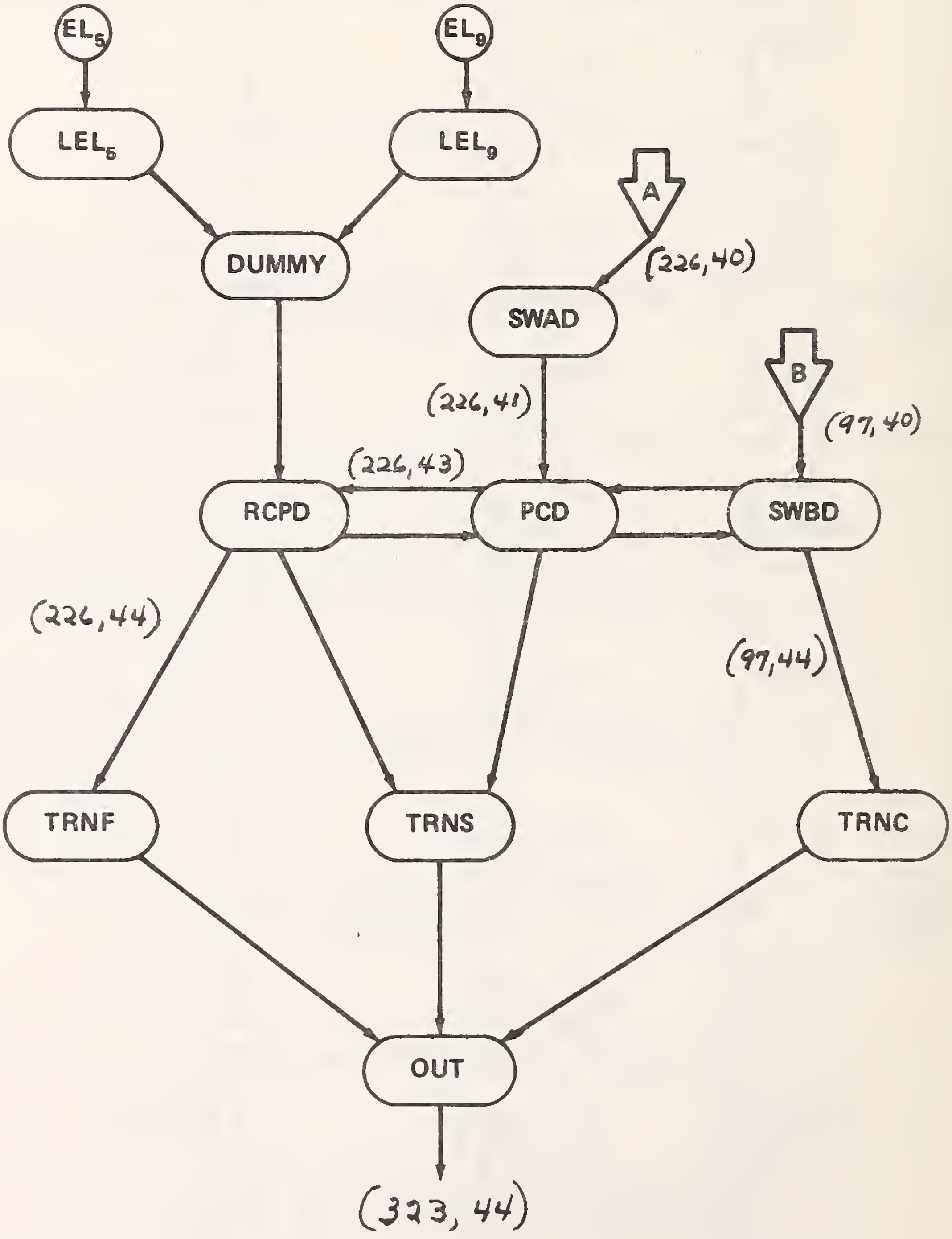


Figure A-4 (Continued)

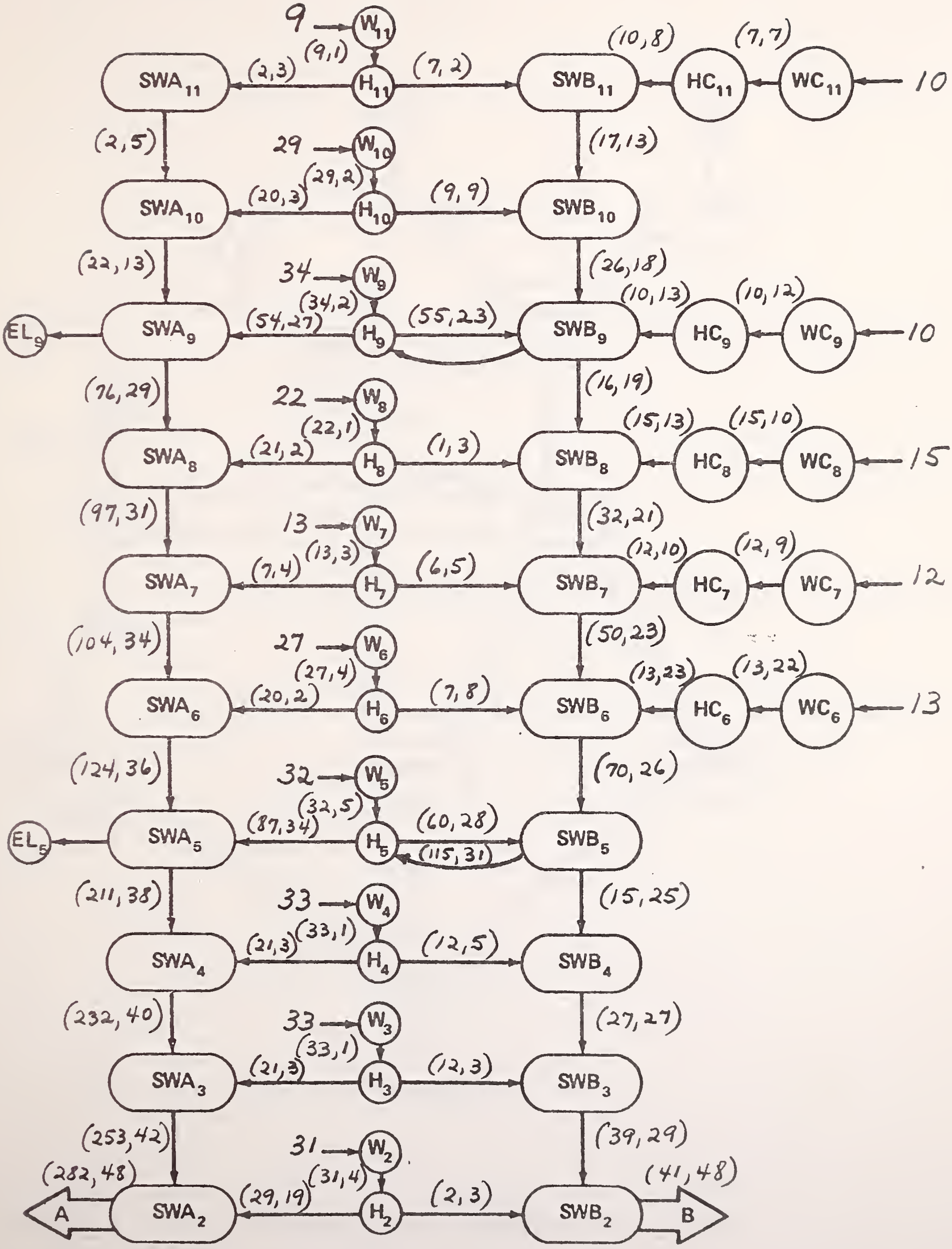


Figure A-5, Run 5 Summary

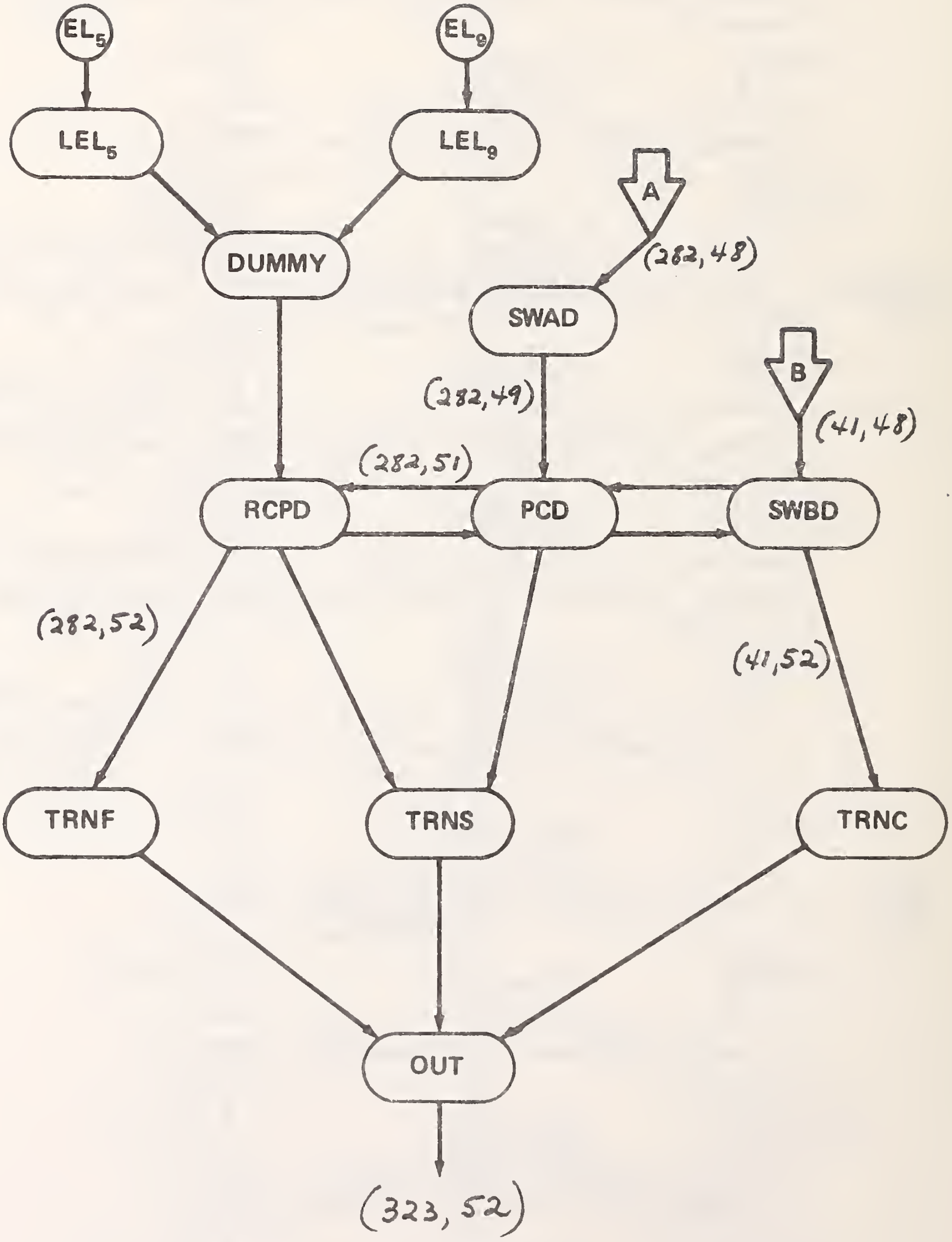


Figure A-5 (Continued)

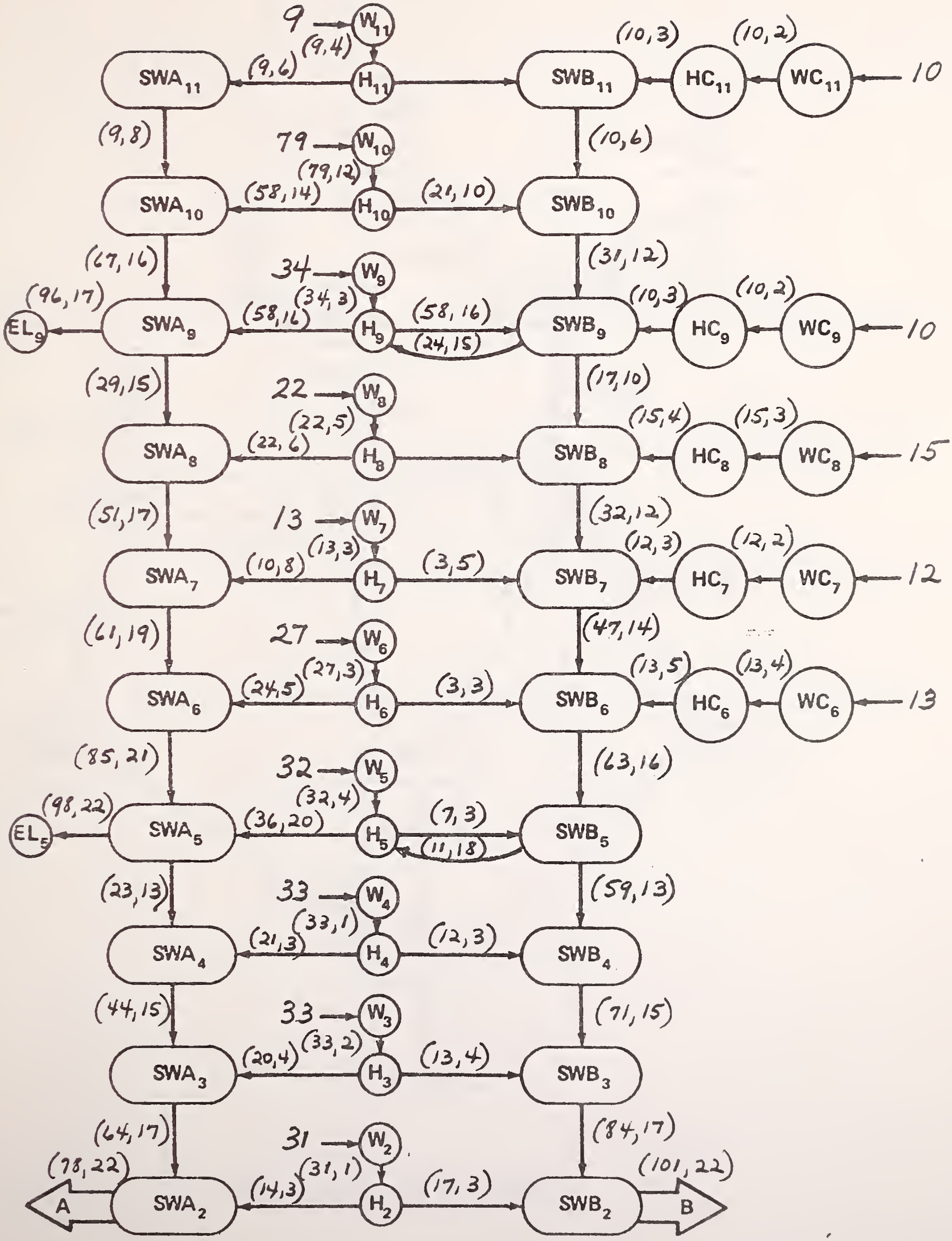


Figure A-6, Run 6 Summary

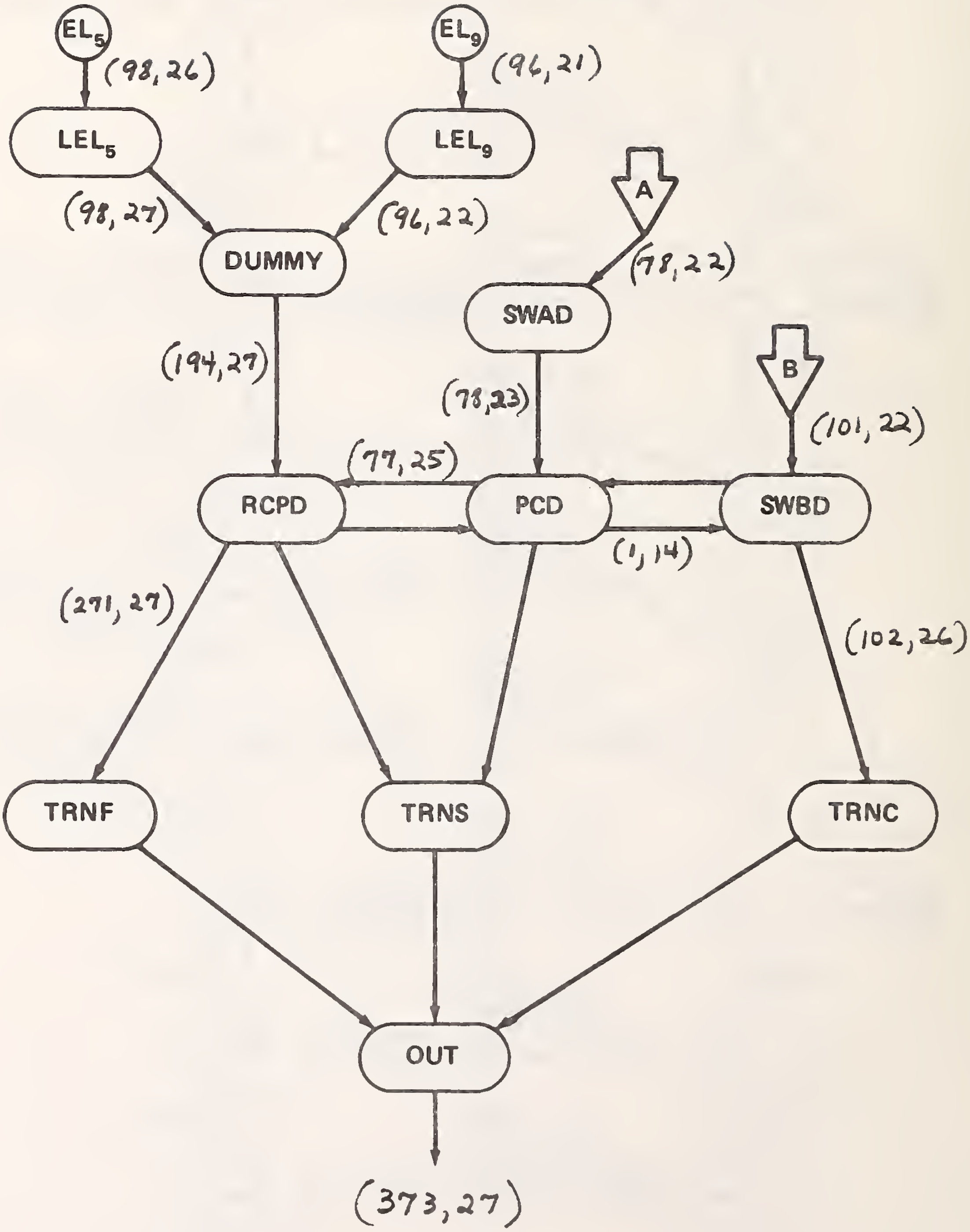


Figure A-6 (Continued)

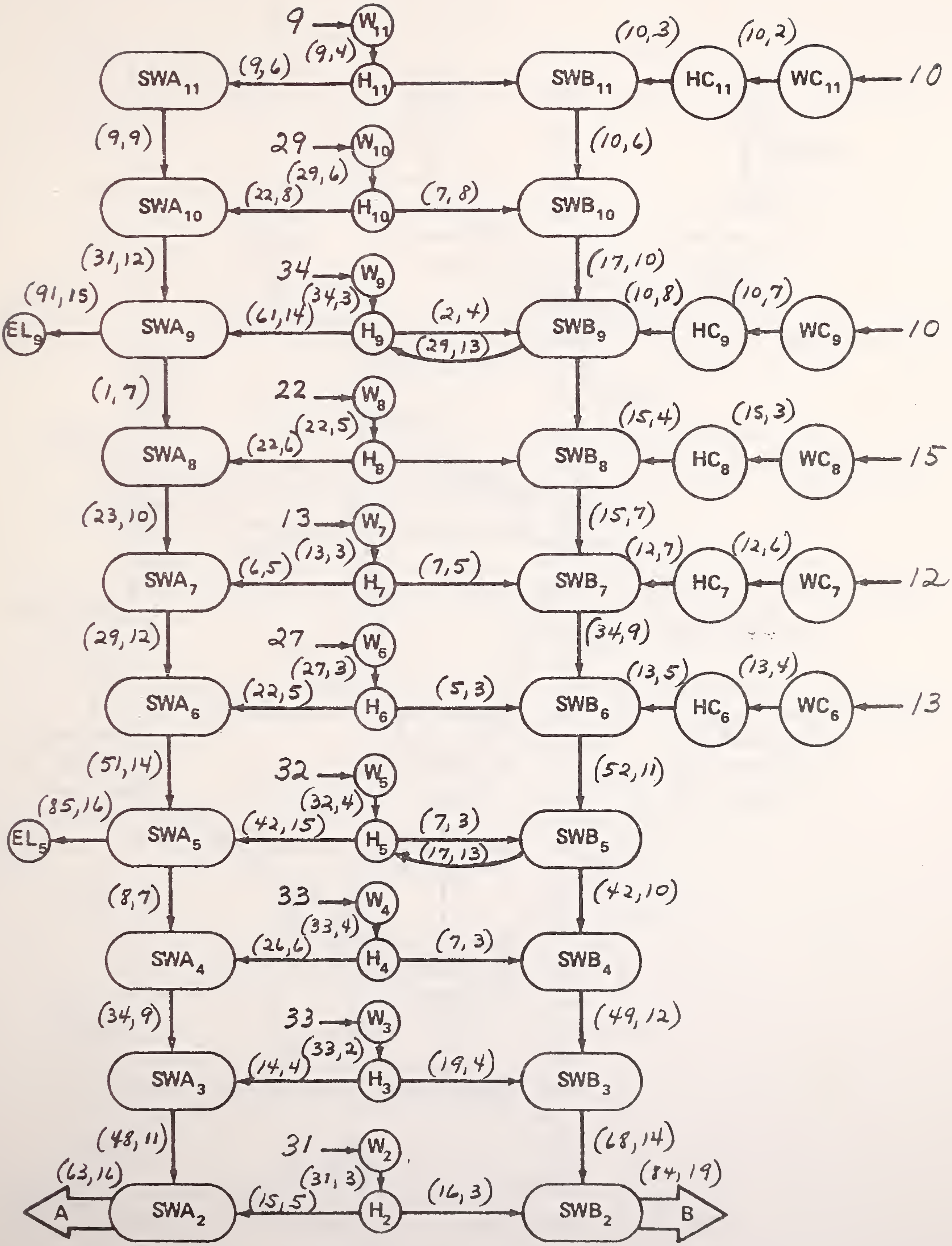


Figure A-7, Run 7 Summary

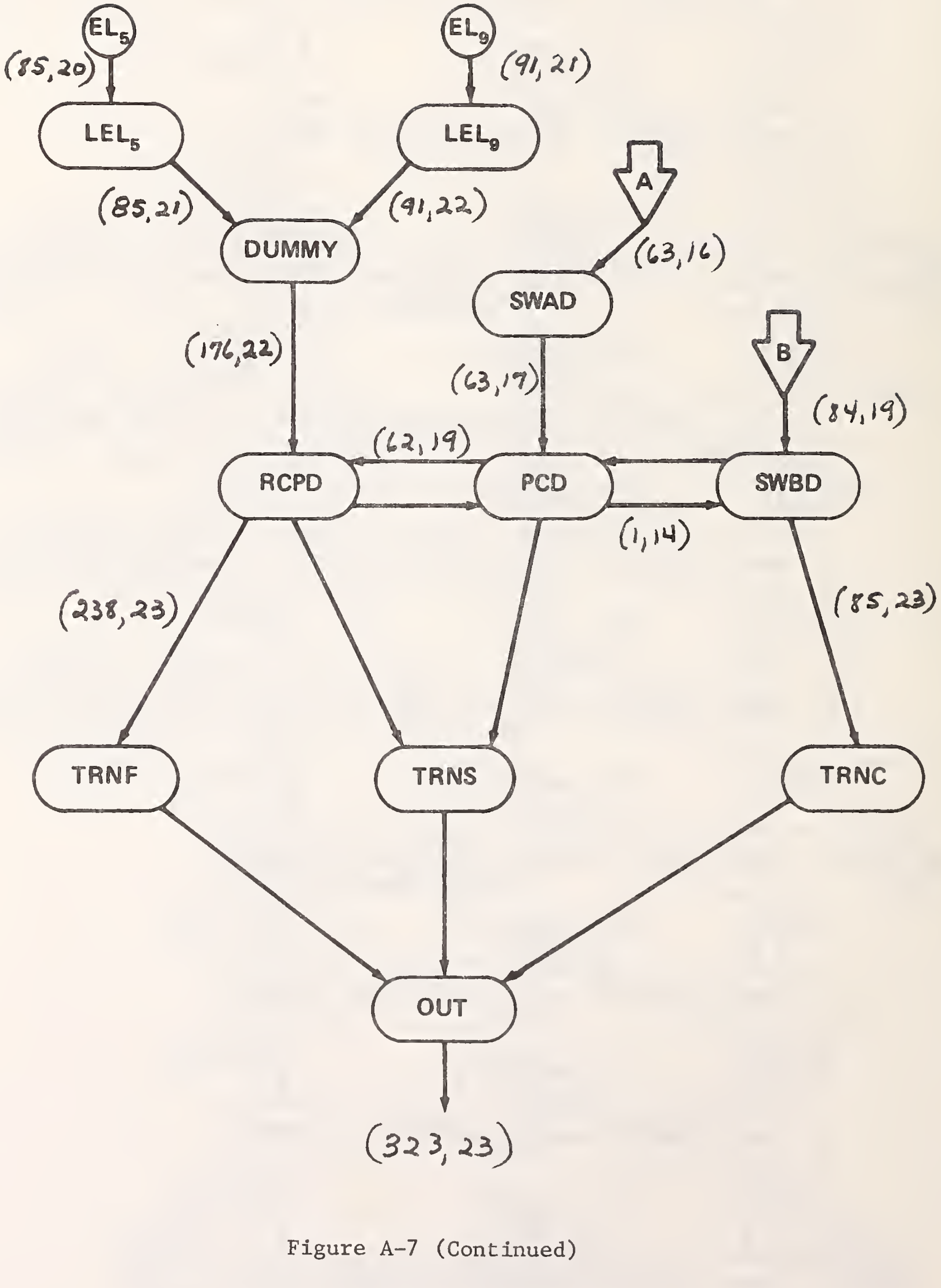


Figure A-7 (Continued)

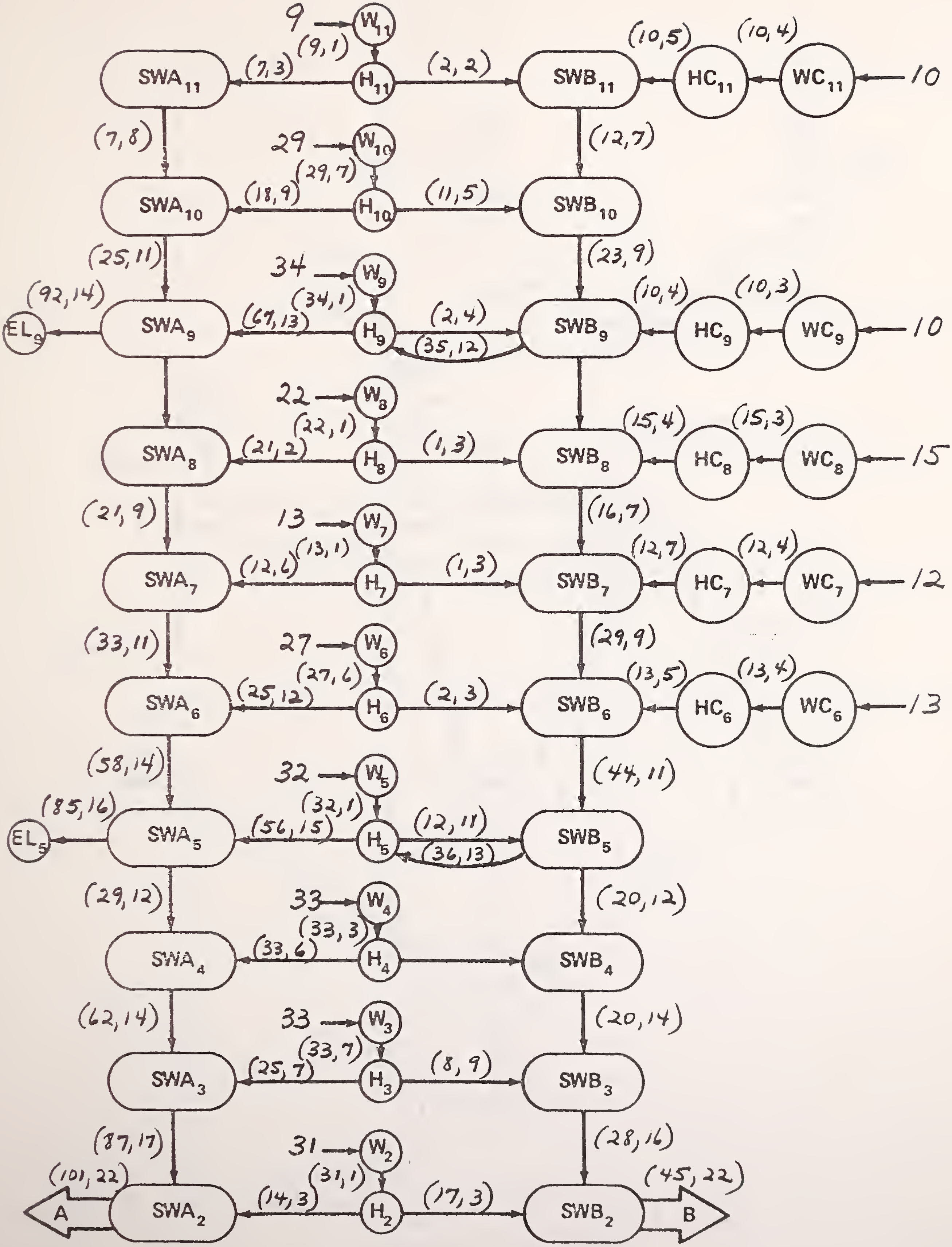


Figure A-8, Run 8 Summary

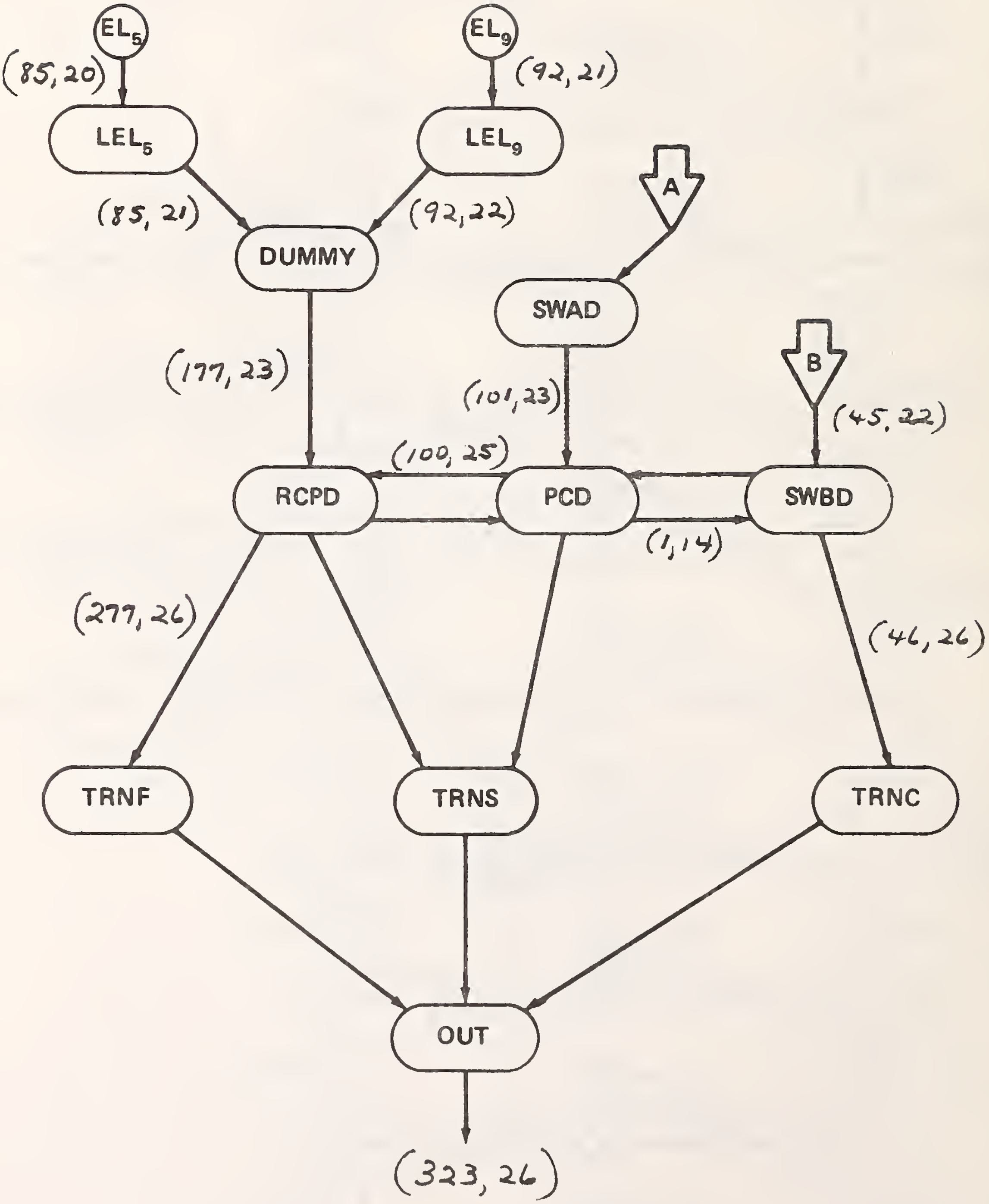


Figure A-8 (Continued)

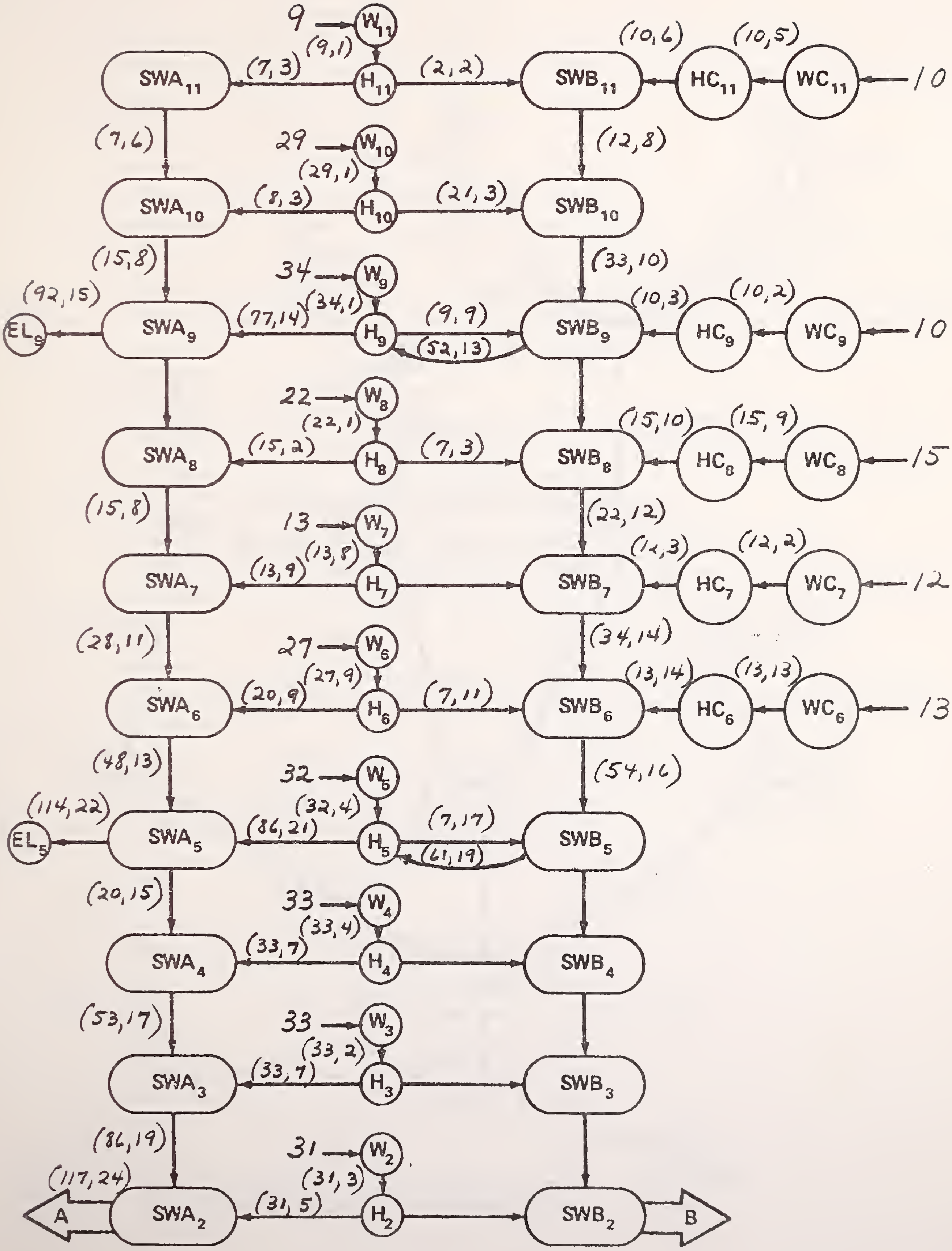


Figure A-9, Run 9 Summary

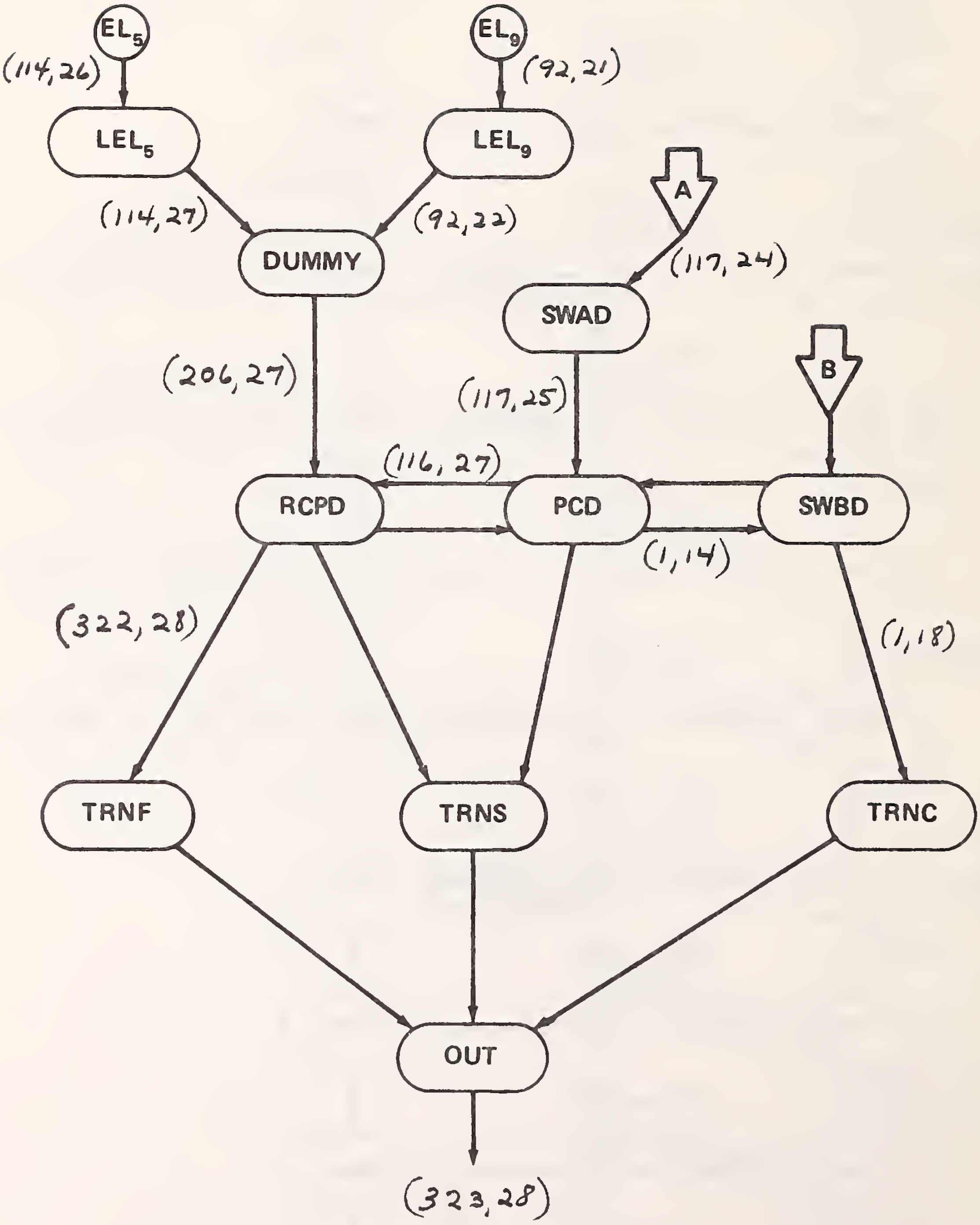


Figure A-9 (Continued)

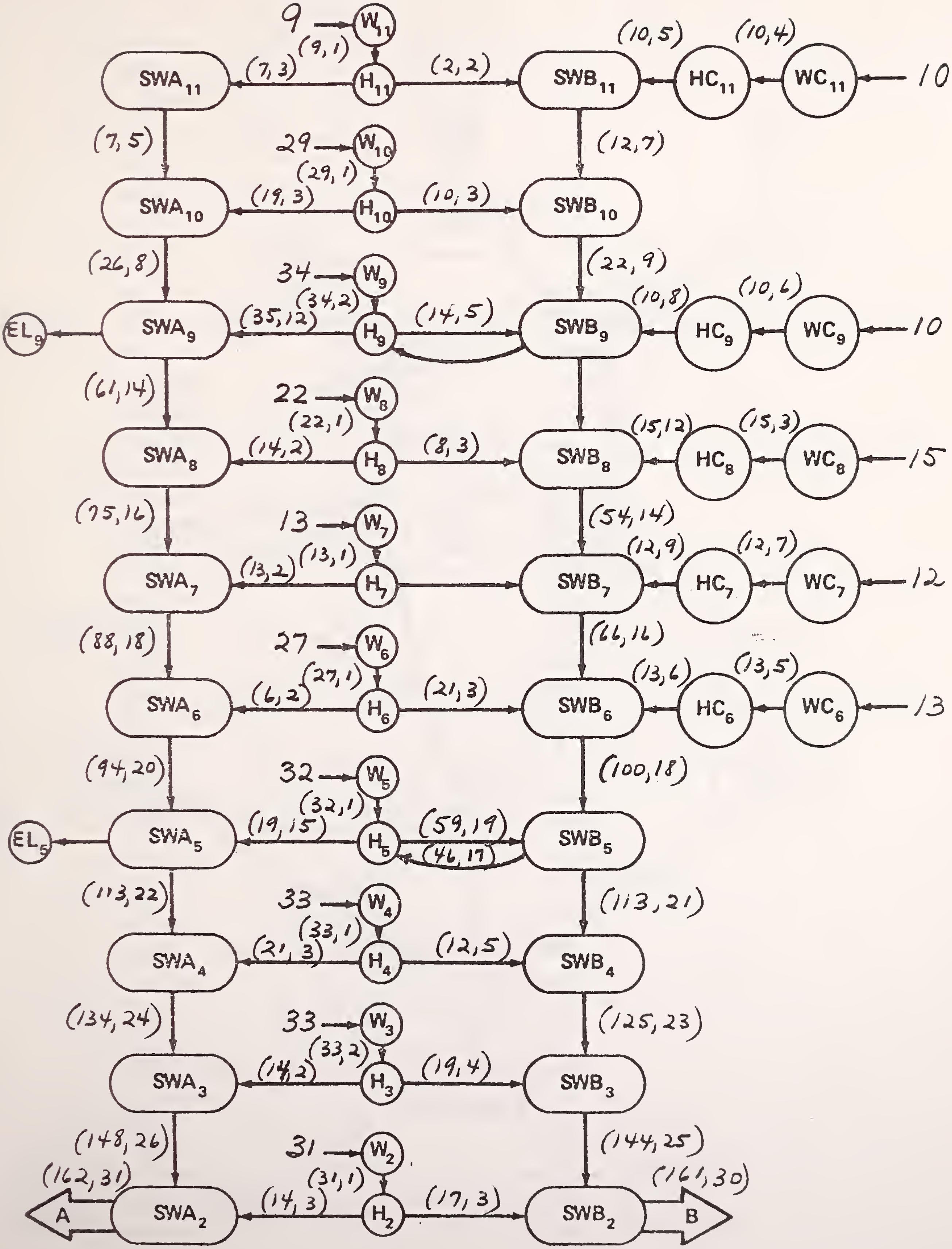


Figure A-10, Run 10 Summary

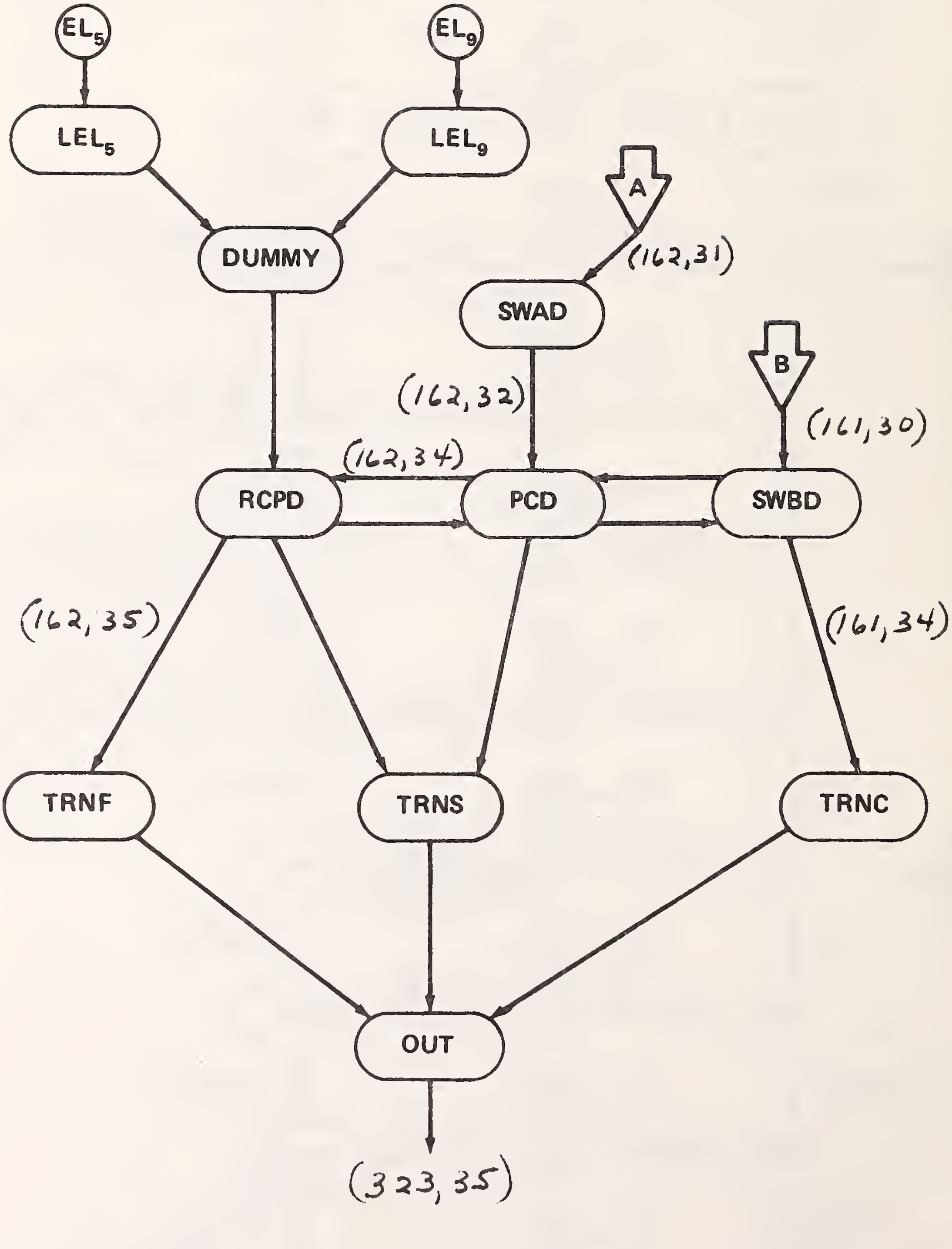


Figure A-10 (Continued)

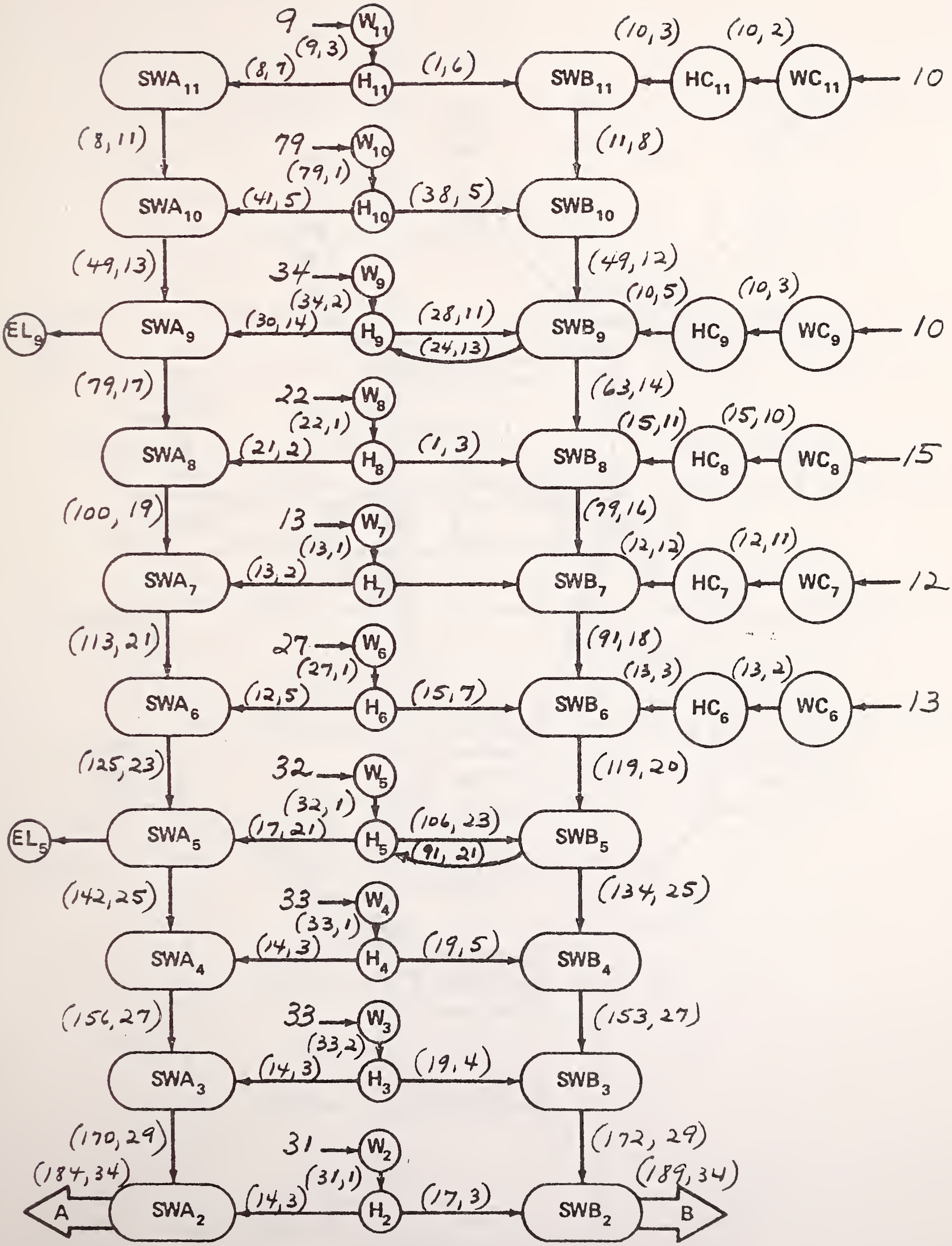


Figure A-11, Run 11 Summary

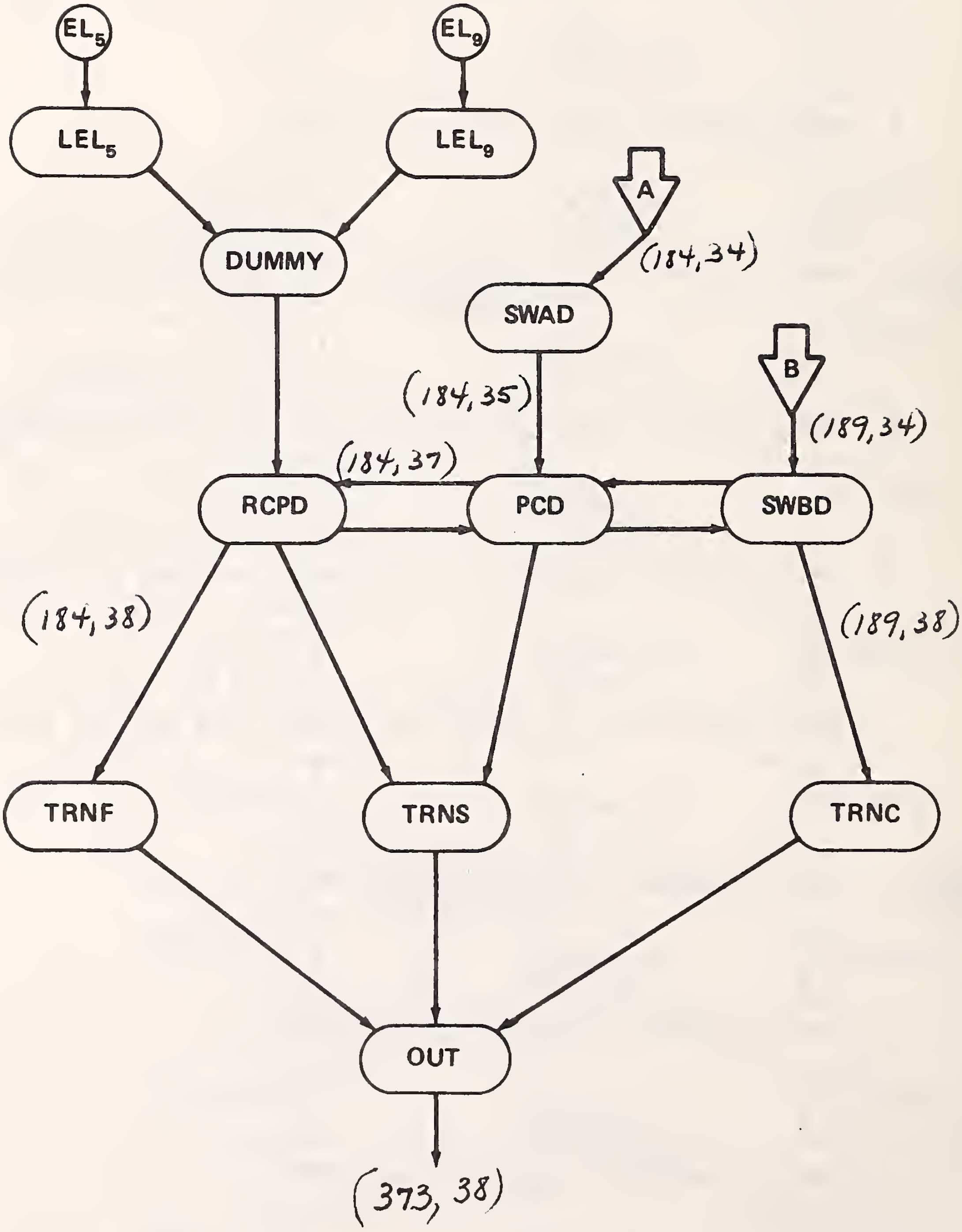


Figure A-11 (Continued)

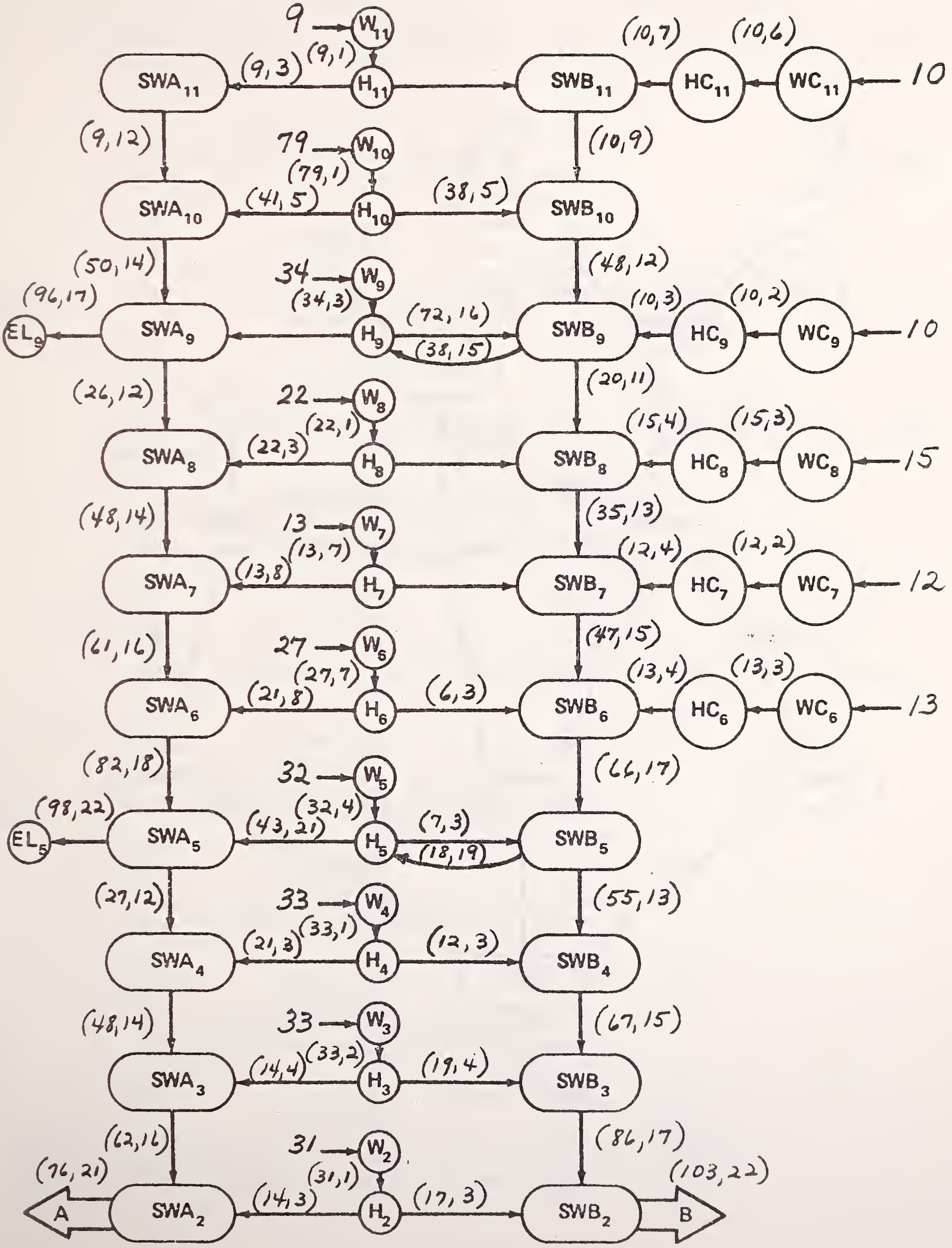


Figure A-12, Run 12 Summary

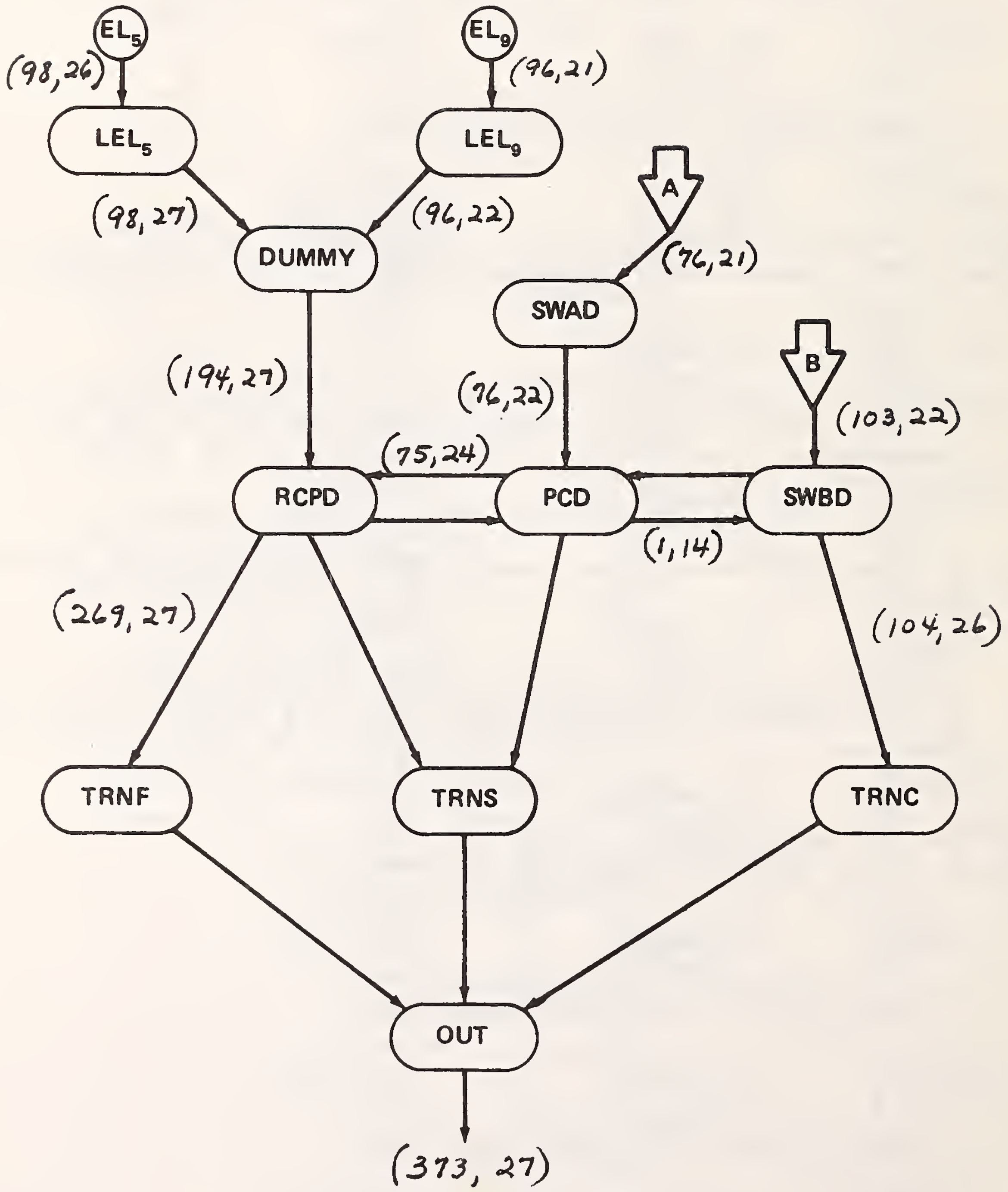


Figure A-12 (Continued)

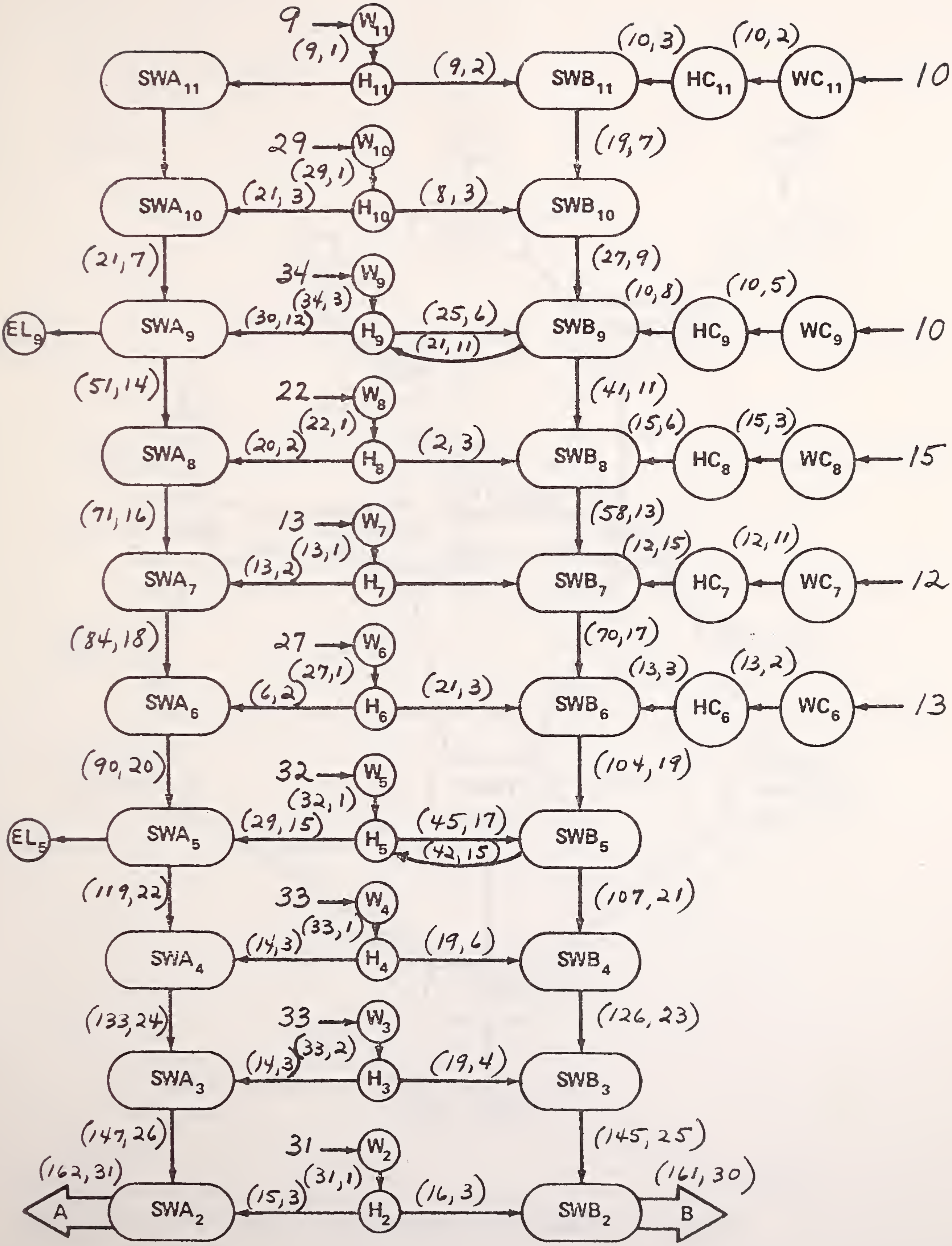


Figure A-13, Run 13 Summary

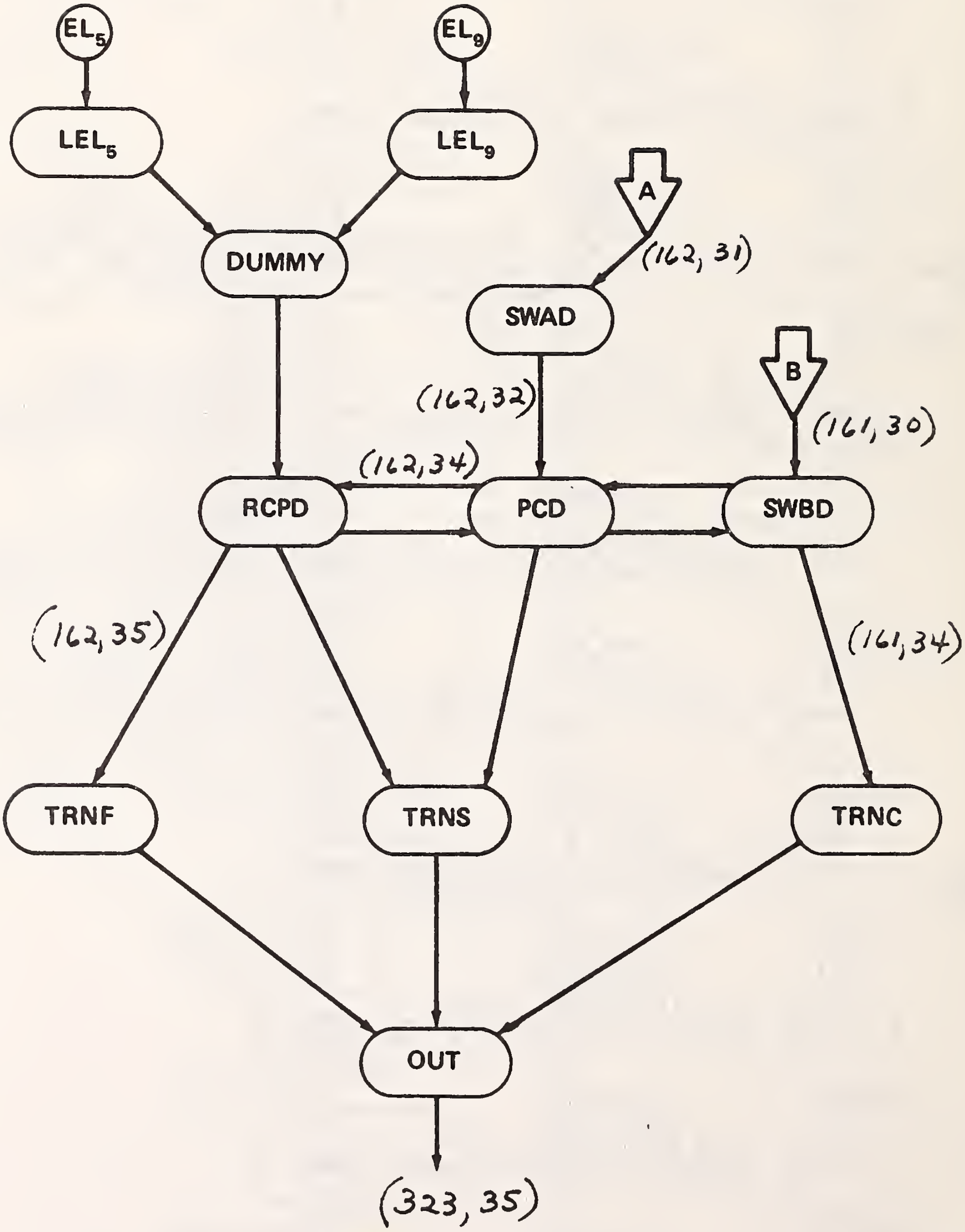


Figure A-13 (Continued)

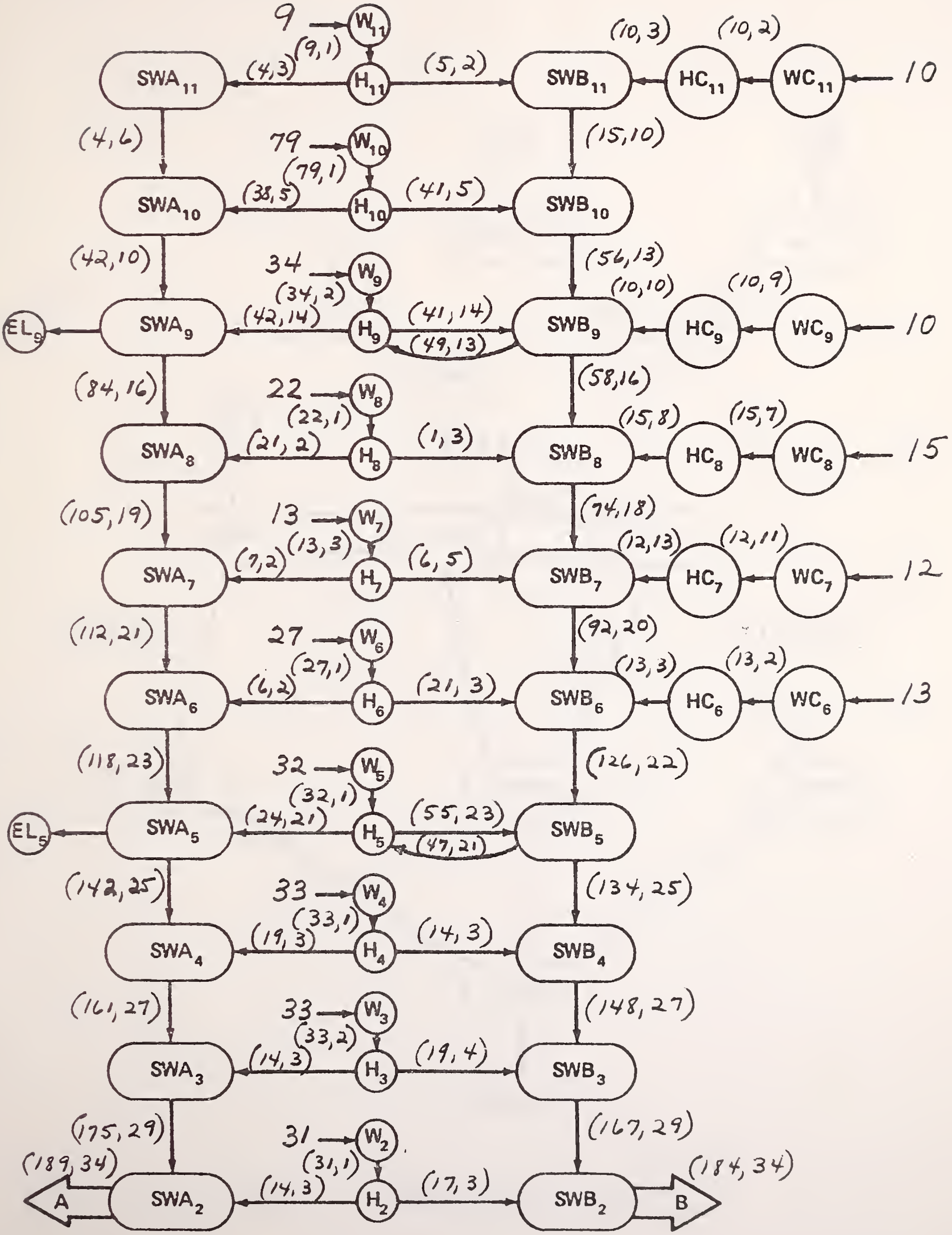


Figure A-14, Run 14 Summary

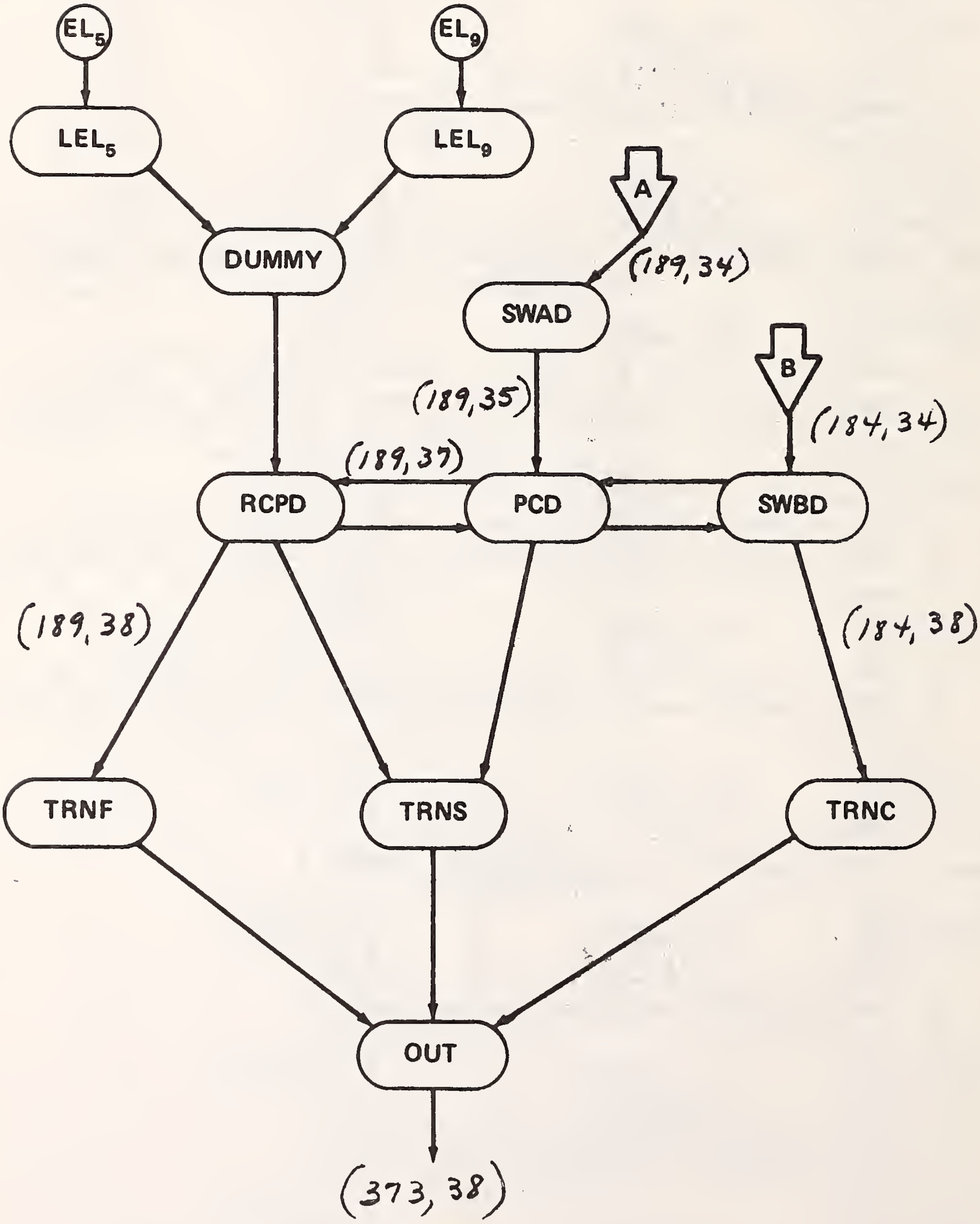


Figure A-14 (Continued)

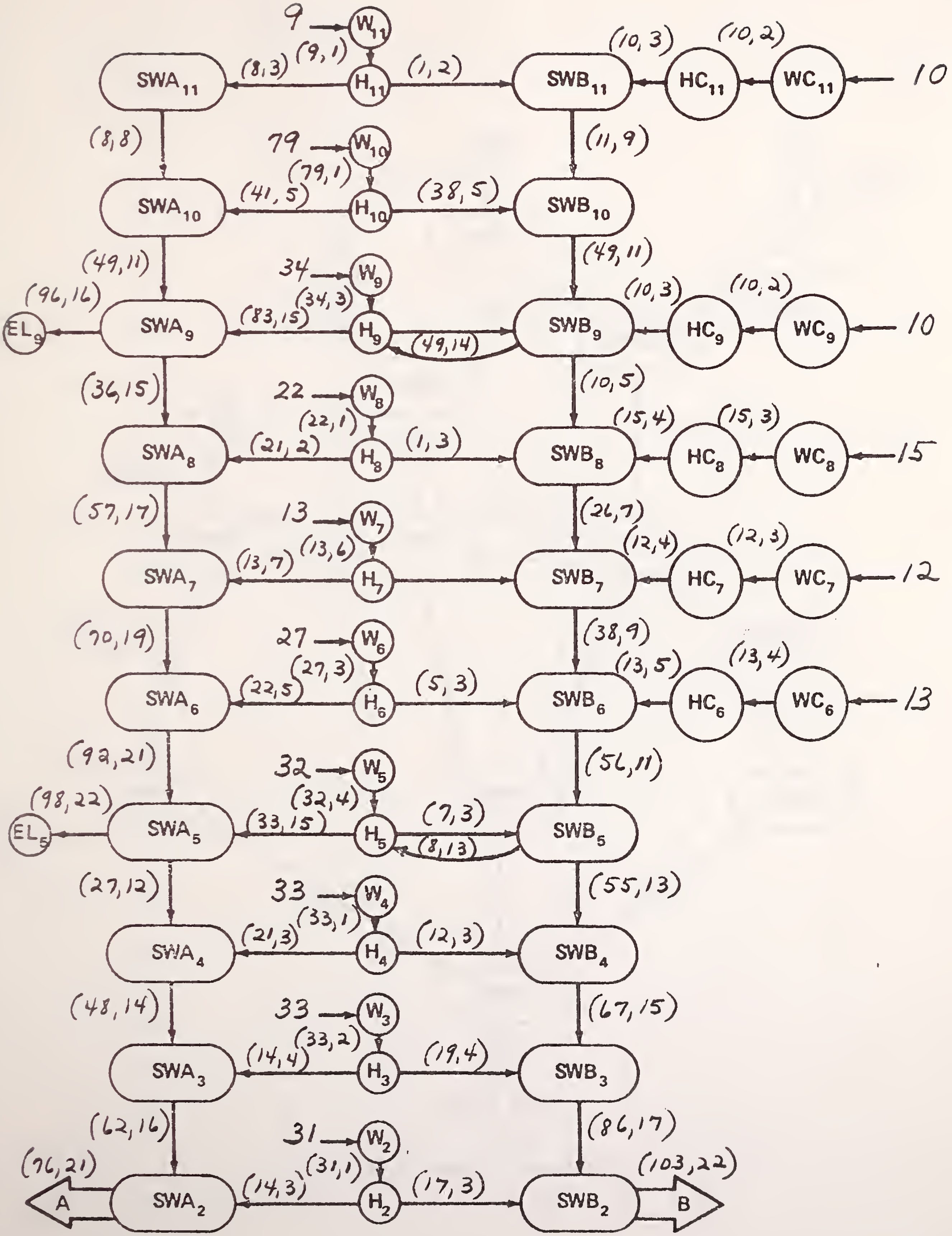


Figure A-15, Run 15 Summary

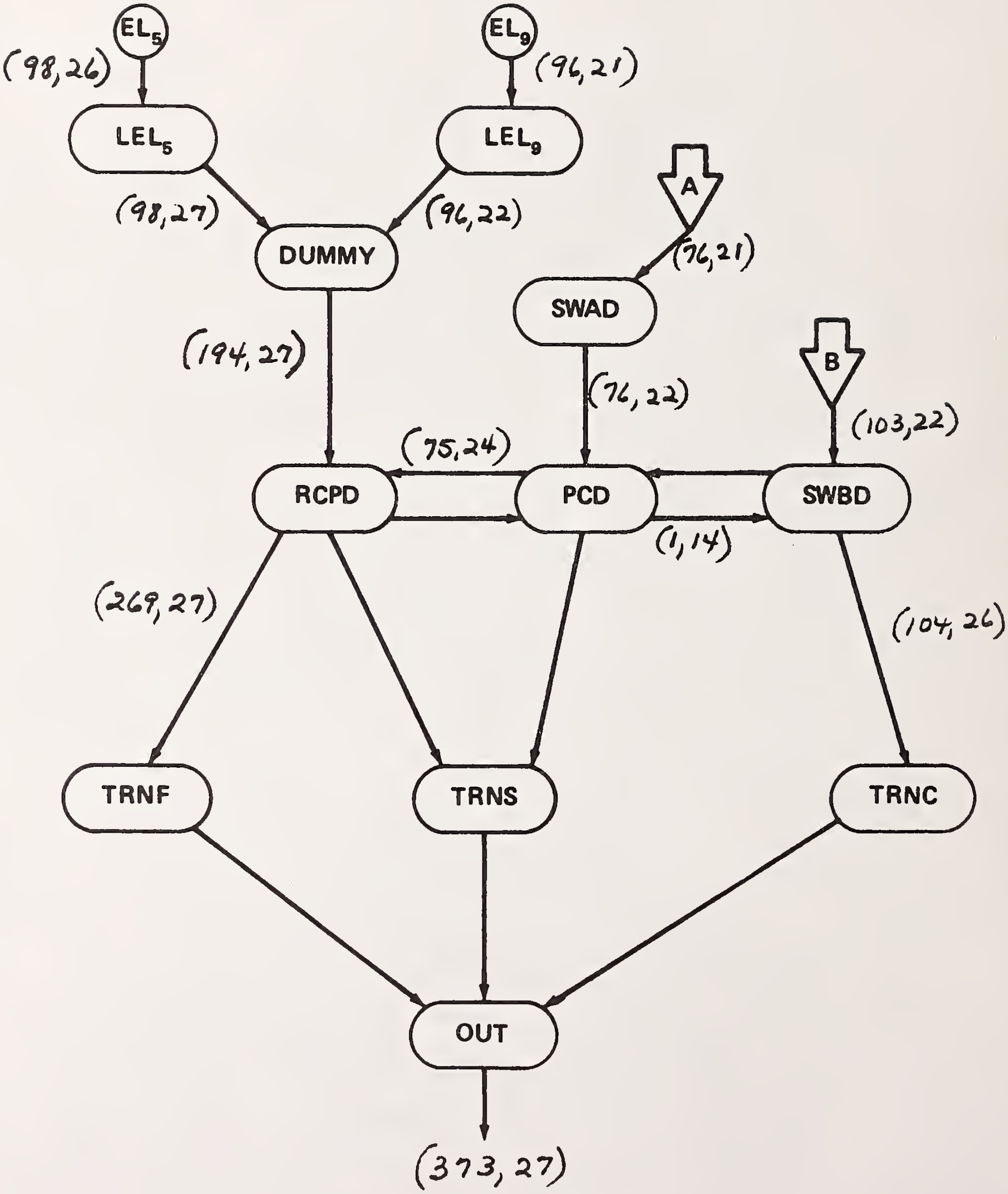


Figure A-15 (Continued)

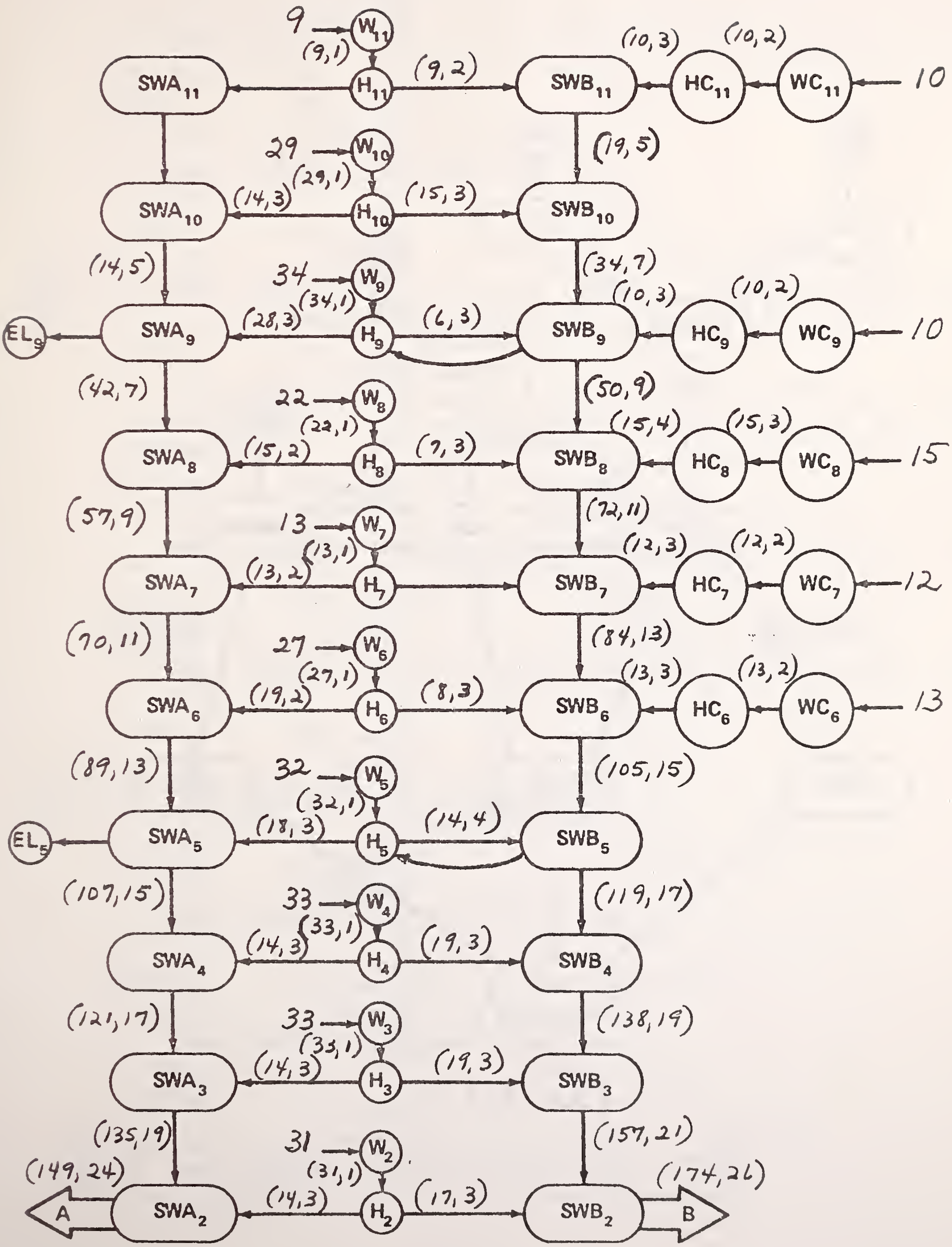


Figure A-16, Run 16 Summary

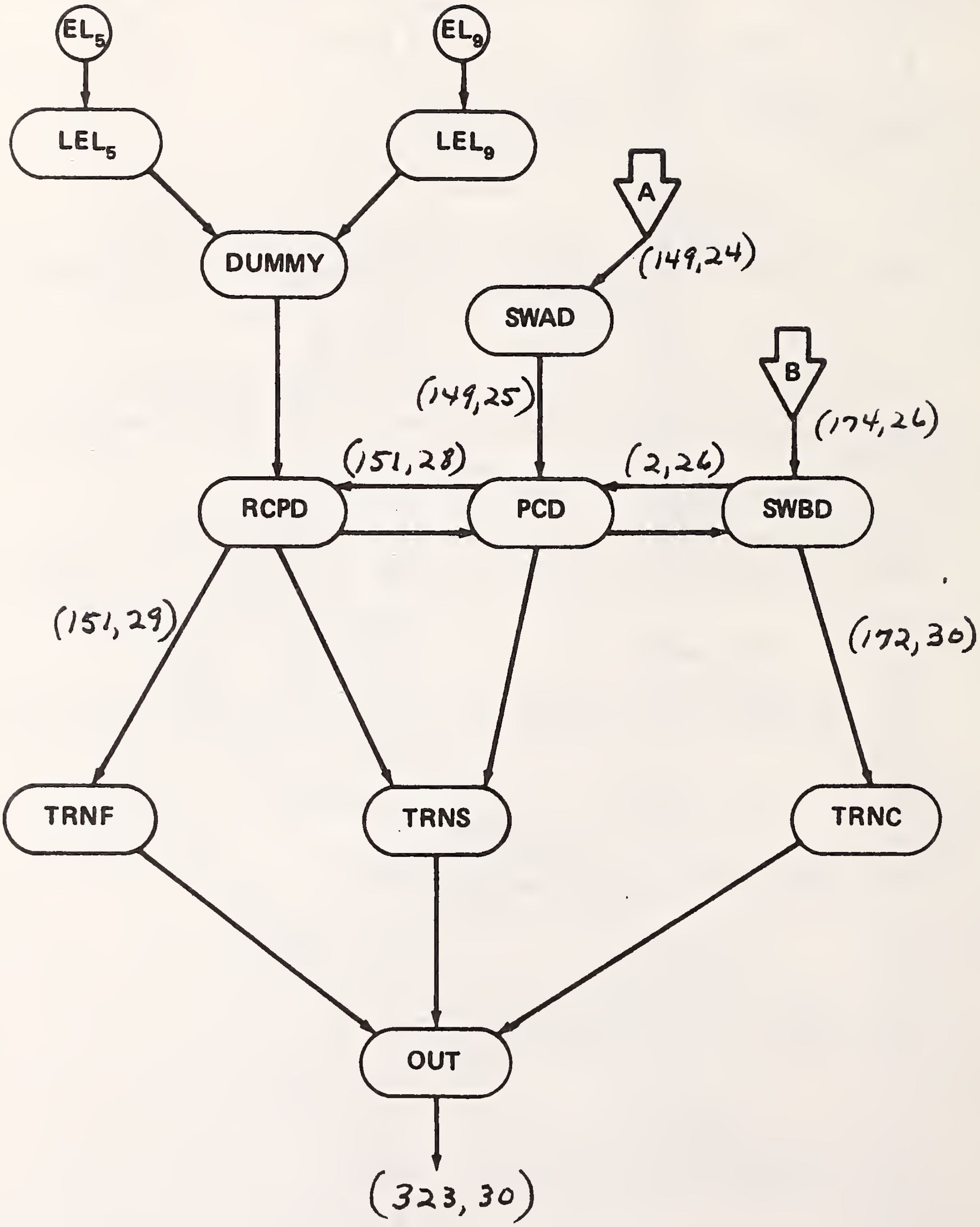


Figure A-16 (Continued)

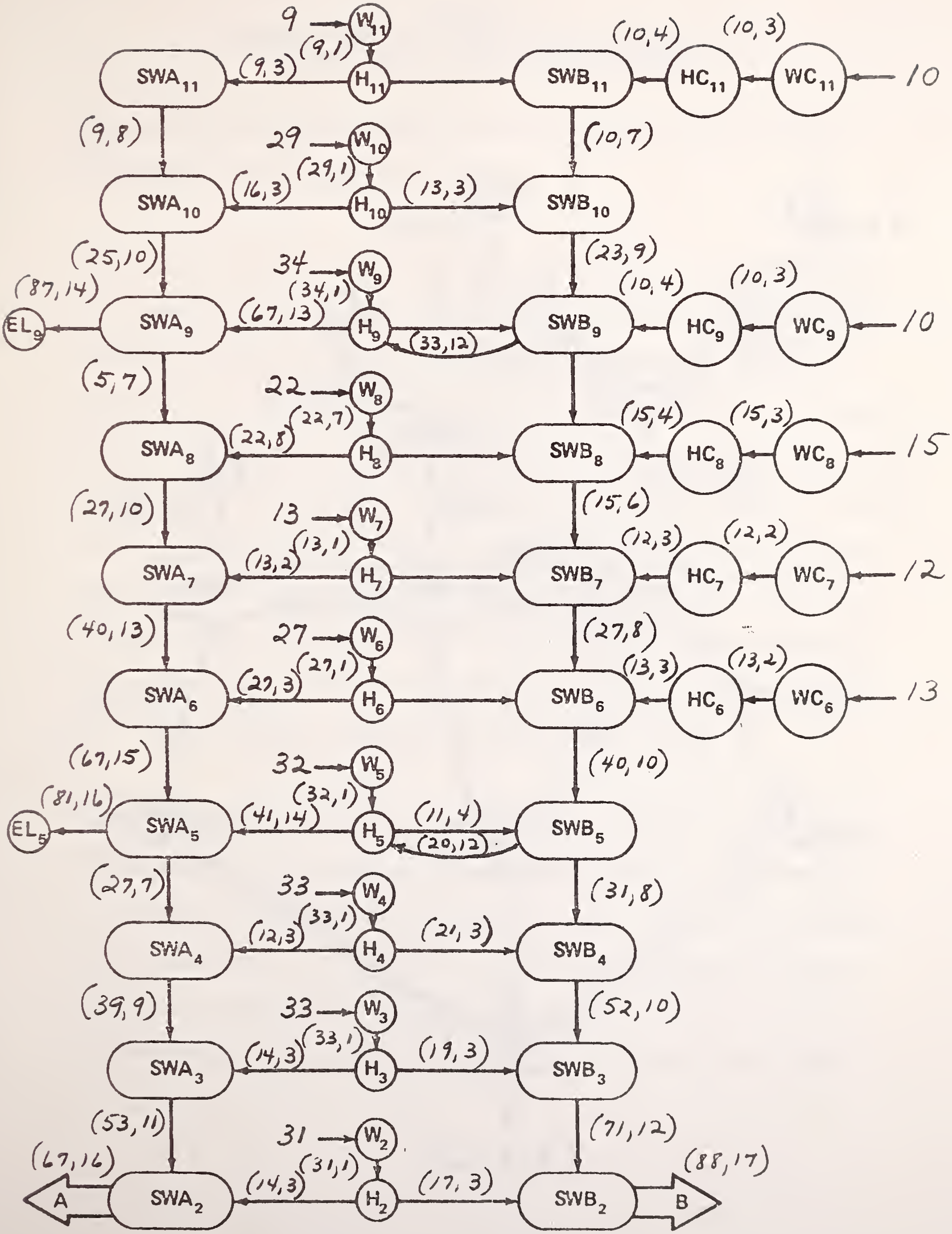


Figure A-17, Run 17 Summary

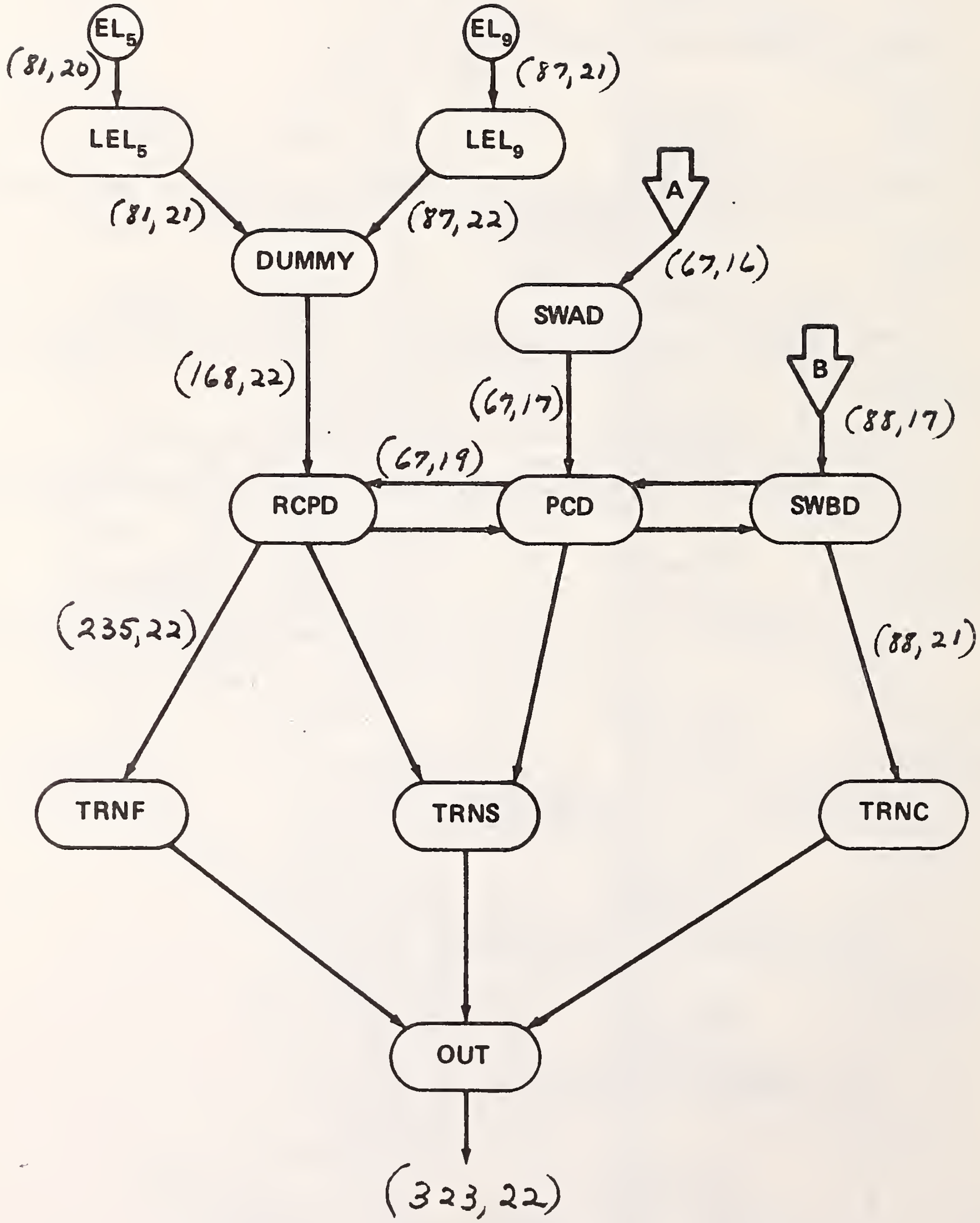


Figure A-17 (Continued)

APPENDIX B
A SIMPLE GRAPHICAL PROCEDURE FOR ESTIMATING THE MINIMUM TIME TO EVACUATE THE BUILDING

This discussion presents a simple graphical procedure for estimating the minimum time to evacuate a building.

The evacuation problem of interest is illustrated by Figure B-1. There are n evacuation routes, x_j people (a number to be determined) will be evacuated via route j, and K people will be evacuated in total. For each route j it is assumed that the time to evacuate x_j people via route j is given by a known "route evacuation time function," say $t_j(x_j)$. Since the building is not evacuated until all the routes are evacuated, the time to evacuate the building, say z, is the longest of the route evacuation times $t_1(x_1)$, ..., $t_n(x_n)$. The problem of interest is to compute the minimum building evacuation time, say z*, as well as the corresponding "route allocation," that is, the allocation of people to routes, say x_1^*, ..., x_n^*, yielding a minimum building evacuation time.

Several assumptions are needed about the route evacuation time functions. For each route j it is assumed (Assumption 1) that $t_j(0) = 0$, that is, zero people are evacuated in zero time. For each route j it is further assumed (Assumption 2) that $t_j(x_j)$ is a continuous function (it has no "jumps") and (Assumption 3) that if y_j is greater than x_j, then $t_j(y_j)$ is greater than $t_j(x_j)$: in other words, each route evacuation time function is strictly increasing. Assumption 3 may be paraphrased as follows: the larger the number of people using a route, the greater the time to evacuate the route. In order for Assumptions 2 and 3 to hold, it is sufficient that the slope of each route evacuation time function always be positive.

The graphical solution procedure can now be stated.

(1) On one side of a piece of translucent paper, graph the axes, representing the horizontal axis (which has units of number of people) by a dotted line, and representing the vertical axis (which has units of time) by a dashed line.

(2) In the first quadrant defined by the axes just constructed, graph each route evacuation time function. Call this graph Graph 1. (See Figure B-2.)

(3) Turn Graph 1 face down. While keeping Graph 1 face down, rotate the graph so that the vertical axis as seen through the translucent paper is now the dotted line (people), while the horizontal axis is now the dashed line (time). Leave Graph 1 in the position it now has from this point on.

Notice that the graph of the route j evacuation time function--when viewed through the paper--gives the number of people, say $p_j(z_j)$, which can be evacuated in a time of z_j. For each route j it is convenient to call $p_j(z_j)$ the "people evacuation function" for route j. Call Graph 1--as viewed through the paper--Graph 2. (See Figure B-3.)

(4) On Graph 2, construct the total (over all routes) of the people evacuation functions, and denote this function by $P(z)$. $P(z)$ is just the total number of people that can be evacuated via all routes in a time of z, so that $P(z) = p_1(z) + \ldots + p_n(z)$.

(5) Using Graph 2, identify the point K on the people axis, and use the function $P(z)$ to find the point z^* on the time axis for which $K = P(z^*)$. The time z^* is the minimum time to evacuate the building.

Continuing to use Graph 2, for each route j read on the people axis the number of people, say x_j^*, for which $x_j^* = p_j(z^*)$. The route allocation $x_1^*, \ldots, x_n^*$ gives the minimum building evacuation time, z^*: any other allocation will give a building evacuation time greater than z^*.

As a simple linear example, suppose there are two evacuation routes, that route 1 has an evacuation flow rate of 42 people per minute, while route 2 has an evacuation flow rate of 18 people per minute. Thus the number of minutes needed to evacuate x_1 people via route 1 is given by $t_1(x_1) = x_1/42$, while the number of minutes needed to evacuate x_2 people via route 2 is given by $t_2(x_2) = x_2/18$. Figure B-2 illustrates Graph 1 for this example. Note each route evacuation time function satisfies Assumptions 1, 2, and 3. Figure B-3 illustrates Graph 2 for this example, showing the people evacuation functions $p_1(z_1)$ and $p_2(z_2)$. For route 1, since 42 people can be evacuated per minute, the number of people which can be evacuated in z_1 minutes is just $p_1(z_1) = 42z_1$. Likewise, for route 2, $p_2(z_2) = 18z_2$. Thus the total number of people which can be evacuated via both routes in z minutes is given by $P(z) = p_1(z) + p_2(z) = 42z + 18z = 60z$. If the building has $K = 420$ people, then the time z^* for which $K = P(z^*)$ is computed from $420 = 60z^*$, giving $z^* = 7$ minutes as the minimum time to evacuate the building. Figure B-3 illustrates $P(z)$, K, and z^*. From Figure B-3 it also follows that $x_1^* = p_1(z^*) = 42z^* = 294$, while $x_2^* = p_2(z^*) = 18z^* = 126$, and so the allocation of 294 people to route 1 and 126 people to route 2 yields a minimum building evacuation time.

Note, for the example, that each of the two routes is completely evacuated in the same time, $z^* = 7$ minutes. It is in fact true for the general problem, because $x_j^* = p_j(z^*)$, that x_j^* people are evacuated via route j in a time of z^*, so that <u>all</u> of the routes are completely evacuated in the same time, z^*.

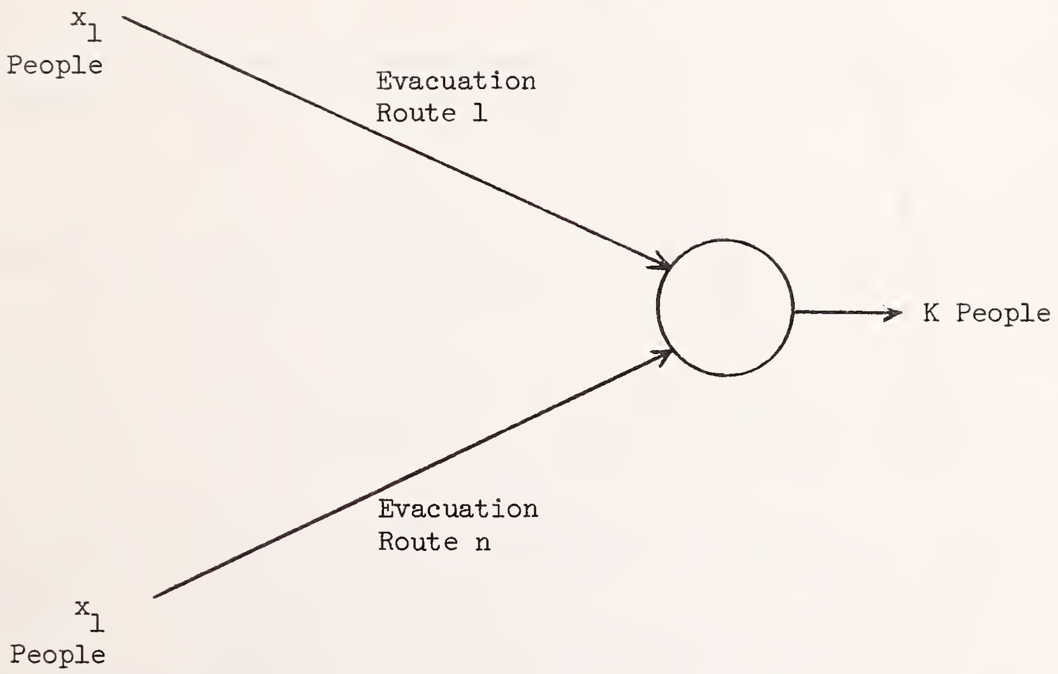


FIGURE B-1: Illustration of the Evacuation Problem

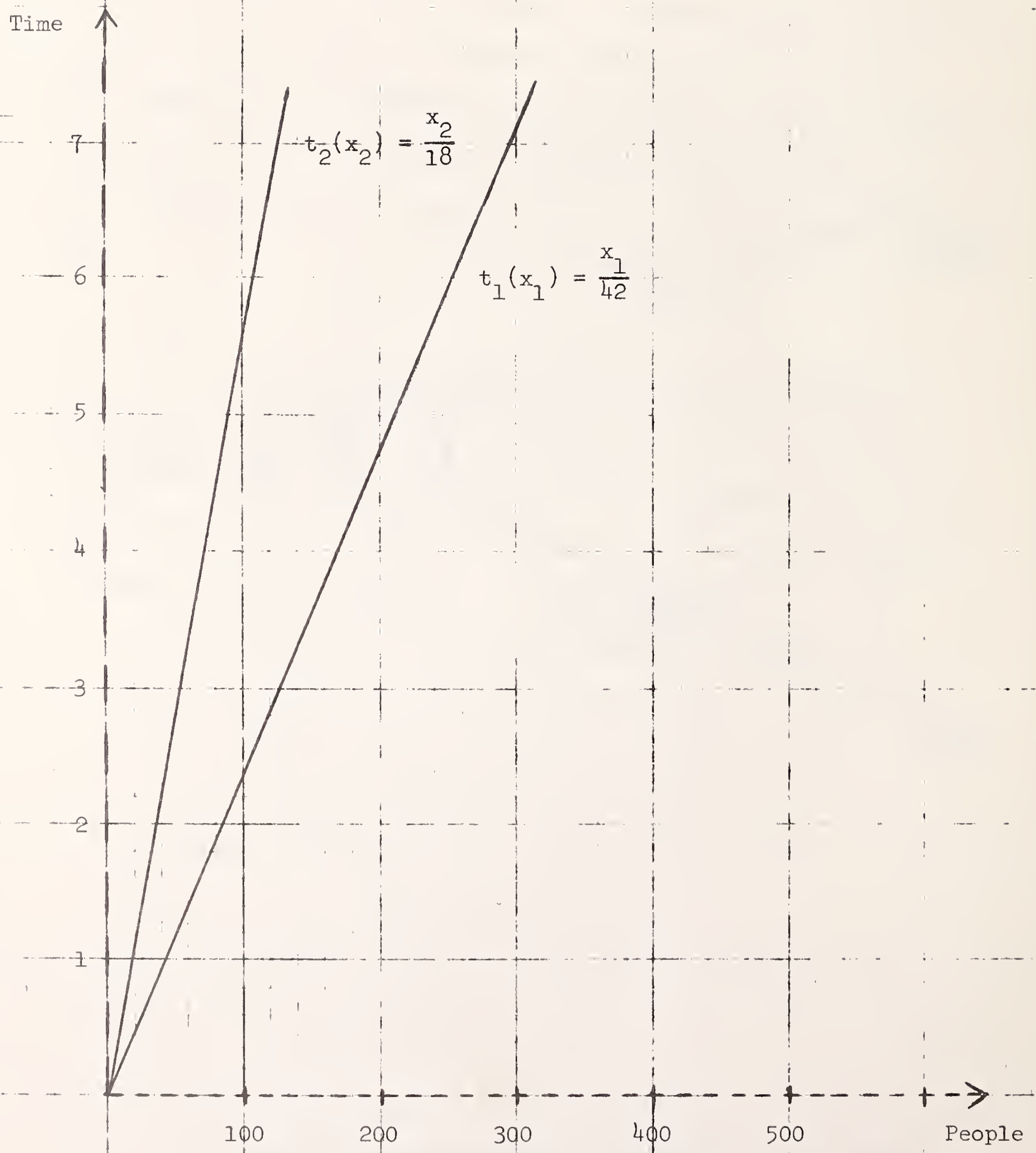


FIGURE B-2: Example of Graph 1

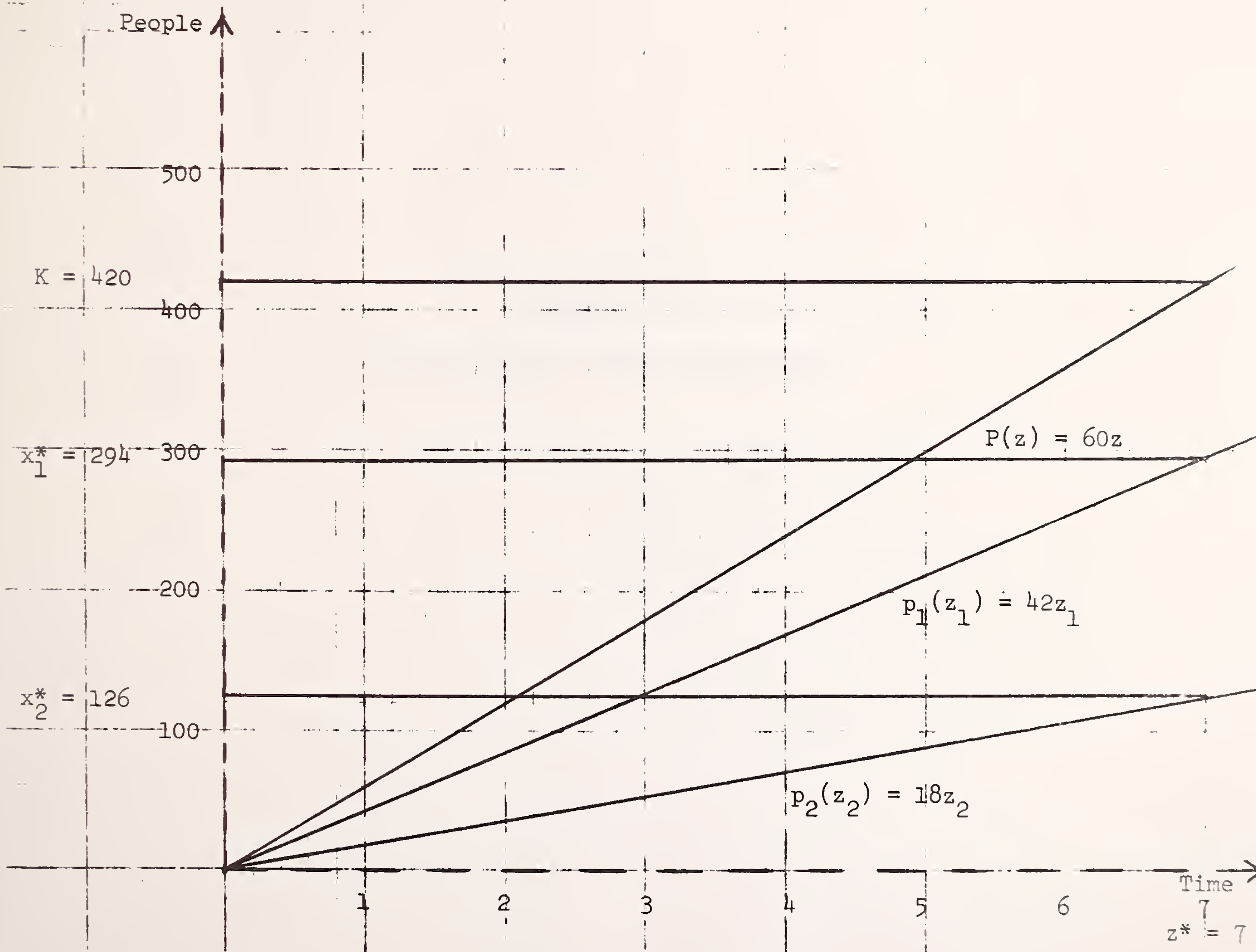


FIGURE B-3: Example of Graph 2

APPENDIX C

LISTING OF SETUP, CHECK, AND POSTPRO

Listing of SETUP

```
      IMPLICIT INTEGER (A-Z)
      OUT=7
      READ (5,100) NARCS,NODES
  100 FORMAT (2I5)
      WRITE (OUT,105) NODES
  105 FORMAT (I5,4X,'1')
      DO 200 I=1,NARCS
      READ (5,115) NAME,ORIG,DEST,COST,CAP,NUMB
  115 FORMAT (A4,2X,2I6,2X,3I10)
      DO 120 J=1,NUMB
      WRITE (OUT,115) NAME,ORIG,DEST,COST,CAP
      ORIG=ORIG+1
      DEST=DEST+1
  120 CONTINUE
  200 CONTINUE
      ENDFILE OUT
      STOP
      END
```

Listing of CHECK

```
      IMPLICIT INTEGER (A-Z)
      END=0
      FLAG=0
      OUT=6
      READ (5,100) NODES
  100 FORMAT (I5)
      ARC= '      '
      RECORD=0
      ACOUNT=0
  105 READ (5,110,END=200) TARC,TORIG,TDEST,TCOST,TCAP
      RECORD=RECORD+1
  110 FORMAT (A6,2I6,2X,2I10)
      IF (TARC .EQ. ARC) GO TO 140
      IF (RECORD .EQ. 1) GO TO 130
  115 ACOUNT=ACOUNT+1
      WRITE (OUT,120) ARC,ORIG,DEST,COST,CAP,ACOUNT
  120 FORMAT (1H ,A6,2I6,2X,2I10,I5)
      IF (END .EQ. 1) GO TO 210
  130 ARC=TARC
      ORIG=TORIG
      DEST=TDEST
      COST=TCOST
      CAP=TCAP
      ACOUNT=0
      GO TO 105
  140 ACOUNT=ACOUNT+1
      IF (TORIG .NE. ORIG+ACOUNT) GO TO 150
      IF (TARC.EQ.'SA09  '.AND.TORIG.EQ.258.AND.TDEST.EQ.384) GO TO 142
      IF (TARC.EQ.'SB09  '.AND.TORIG.EQ.288.AND.TDEST.EQ.416) GO TO 142
      IF (TDEST .NE. DEST+ACOUNT) GO TO 150
  142 IF (COST .GE. 0) GO TO 146
      IF (ARC.EQ.'2TRF  '.AND.TORIG.EQ.2481.AND.TDEST.EQ.2482.AND.TCOST
     1    .EQ.-218) GO TO 148
      IF (TCOST .NE. COST-ACOUNT) GO TO 150
      GO TO 148
  146 IF (TCOST .NE. COST) GO TO 150
  148 IF (TCAP .EQ. CAP) GO TO 105
  150 WRITE (6,160) RECORD,ACOUNT,ARC,ORIG,DEST,COST,CAP,TARC,TORIG,
     1                   TDEST,TCOST,TCAP
  160 FORMAT ('0BAD DATA IN RECORD NO. ',I5,' WHICH IS RECORD NO. ',I5,
     1           ' FOR THIS ARC.'/(1H ,A6,2I6,2X,2I10))
      FLAG=FLAG+1
      GO TO 105
  200 END=1
      GO TO 115
  210 WRITE (OUT,215) FLAG
  215 FORMAT (' *** NO. OF BAD RECORDS IS ',I5)
      STOP
      END
```

Listing of Main Program of POSTPRO

```
      IMPLICIT INTEGER (A-Z)
      REAL COSTV
      PARAMETER MAXN=2000,MAXA=5000
      DIMENSION NAME(MAXN),LOWER(MAXN),OFFSET(MAXN)
      DIMENSION NAMEO(MAXN),FIRST(MAXN),LAST(MAXN)
      DIMENSION FLOW(MAXA),ORIG(MAXA),DEST(MAXA),PO(MAXA),PD(MAXA)
      DIMENSION NEXT(MAXA)
      DIMENSION TITLE(20)
      DATA YES/4HYES /,NO/4HNO  /
      DATA FINAL/4HFINA/,FROM/4HFROM/,COST/4HCOST/
      DATA DUMY/4HDUMY/,TRNC/4HTRNC/,TRNF/4HTRNF/,TRNS/4HTRNS/
      OUT=6
C
C *** UNIT 47 IS AN ALTERNATE PRINT FILE, USED IN THIS PROGRAM
C *** TO KEEP THE REGULAR OUTPUT SEPARATE FROM THE QUESTIONS
C *** TO THE USER REGARDING WHICH OUTPUT REPORTS ARE DESIRED.
C
      ARC=0
      NORIGS=0
      READ (5,5) NODES
    5 FORMAT (I5)
      READ (5,10) (LOWER(N),OFFSET(N),NAME(N),N=1,NODES)
   10 FORMAT (I5,5X,I5,1X,A6)
      INDO=NODES
      INDD=NODES
   50 READ (5,55,END=350) (TITLE(I),I=1,20)
   55 FORMAT (1X,19A4,A3)
      WRITE (OUT,55) (TITLE(I),I=1,20)
      IF (TITLE(1) .NE. FINAL) GO TO 50
      READ (5,55) (TITLE(I),I=1,20)
      IF (TITLE(1) .EQ. FROM) GO TO 70
      WRITE (6,65) (TITLE(I),I=1,20)
   65 FORMAT ('0ERROR IN INPUT FILE. CARD READ IS AS FOLLOWS:  '/1X,
     1        19A4,A3)
      STOP
   70 READ (5,55,END=200) TITLE(1)
      IF (TITLE(1) .NE. COST) GO TO 73
          READ (5,71) COSTV
   71     FORMAT (6X,F12.0)
          WRITE (OUT,72) COSTV
   72     FORMAT (///' COST = ',F12.0)
          GO TO 70
   73 READ (5,74) TORIG,TDEST,TFLOW
   74 FORMAT (I5,2X,I5,1X,I10)
      IF (TORIG .LT. LOWER(INDO)) GO TO 76
          IF (INDO .EQ. NODES) GO TO 80
              IF (TORIG .LT. LOWER(INDO+1)) GO TO 80
   76 CALL SEARCH (TORIG,INDO,LOWER,NODES)
   80 IF (TDEST .LT. LOWER(INDD)) GO TO 82
          IF (INDD .EQ. NODES) GO TO 85
              IF (TDEST .LT. LOWER(INDD+1)) GO TO 85
   82 CALL SEARCH (TDEST,INDD,LOWER,NODES)
   85 IF (INDO .LT. 1) GO TO 100
      IF (INDD .LT. 1) GO TO 100
          ARC=ARC+1
          PO(ARC)=TORIG-OFFSET(INDO)+1
          NAME1=NAME(INDO)
```

```
          IF (NAME1 .EQ. TRNC .OR. NAME1 .EQ. TRNF .OR. NAME1 .EQ. TRNS)
     1        PO(ARC)=PO(ARC)-1
          PD(ARC)=TDEST-OFFSET(INDD)
          ORIG(ARC)=NAME(INDO)
          DEST(ARC)=NAME(INDD)
          FLOW(ARC)=TFLOW
          IF (NAME(INDO) .EQ. DUMY .OR. NAME(INDD) .EQ. DUMY)
     1        PD(ARC)=PD(ARC)+1
          IF (NORIGS .GT. 0) GO TO 90
              NORIGS=1
              NAMEO(1)=NAME(INDO)
              FIRST(1)=1
              LAST(1)=1
              NEXT(1)=0
              GO TO 70
   90     TNAME=NAME(INDO)
          DO 92 N=1,NORIGS
             NSAVE=N
             IF (TNAME .EQ. NAMEO(N)) GO TO 94
   92     CONTINUE
          NORIGS=NORIGS+1
          NAMEO(NORIGS)=TNAME
          FIRST(NORIGS)=ARC
          LAST(NORIGS)=ARC
          NEXT(ARC)=0
          GO TO 70
   94     TLAST=LAST(NSAVE)
          NEXT(TLAST)=ARC
          NEXT(ARC)=0
          LAST(NSAVE)=ARC
          GO TO 70
  100 WRITE (OUT,105) TORIG,TDEST,TFLOW
  105 FORMAT ('0UNABLE TO RECOGNIZE ORIGIN AND/OR DESTINATION OF THE '
     1        'FOLLOWING ARC:  '/1X,I5,2X,I5,1X,I10)
      GO TO 70
  200 NARCS=ARC
C     WRITE (6,205)
  205 FORMAT ('0DO YOU WANT A REPORT OF ALL ARCS ORDERED BY '
     1        'DESTINATION?  YES OR NO')
      READ (5,210) REPLY
  210 FORMAT (A4)
      IF (REPLY .EQ. NO) GO TO 250
          WRITE (OUT,220)
  220 FORMAT (1H1,21X,'ARCS ORDERED BY DESTINATION'//)
          WRITE (OUT,230) (FLOW(ARC),ORIG(ARC),DEST(ARC),PO(ARC),PD(ARC)
     1                   ,ARC=1,NARCS)
  230     FORMAT (I10,' PEOPLE FROM ',A6,' TO ',A6,' IN PERIODS',I5,
     1            ' THROUGH',I5)
  250 CONTINUE
C     WRITE (6,255)
  255 FORMAT ('0DO YOU WANT A REPORT OF ALL ARCS ORDERED BY '
     1        'ORIGIN?  YES OR NO')
      READ (5,210) REPLY
      IF (REPLY .EQ. NO) GO TO 350
          WRITE (OUT,260)
  260     FORMAT (1H1,21X,'ARCS ORDERED BY ORIGIN'/)
          DO 280 N=1,NORIGS
             WRITE (OUT,265)
```

Listing of Main Program of POSTPRO (Continued)

```
  265        FORMAT (1H )
             ARC=FIRST(N)
  270        WRITE (OUT,230) FLOW(ARC),ORIG(ARC),DEST(ARC),PO(ARC),
     1                       PD(ARC)
             ARC=NEXT(ARC)
             IF (ARC .GT. 0) GO TO 270
  280     CONTINUE
  350 CONTINUE
C     WRITE (6,355)
  355 FORMAT ('0DO YOU WANT A REPORT ON TOTAL AND LAST FLOW OVER EACH '
     1        /'DISTINCT ARC?  YES OR NO')
      READ (5,210) REPLY
      IF (REPLY .EQ. NO) GO TO 400
  359     WRITE (OUT,360)
  360     FORMAT ('1',10X,'REPORT ON TOTAL AND LAST FLOW OVER EACH '
     1            'DISTINCT ARC'//' ***** ARC ******',5X,'TOTAL FLOW',6X
     2        ,'*********** LAST FLOW ************'//)
          DO 390 N=1,NORIGS
             SUM=0
             ARC=FIRST(N)
             PDEST=DEST(ARC)
  365        IF (DEST(ARC) .NE. PDEST) GO TO 370
                SUM=SUM+FLOW(ARC)
                GO TO 380
  370        WRITE (OUT,375) ORIG(TARC),PDEST,SUM,FLOW(TARC),PO(TARC),
     1       PD(TARC)
  375        FORMAT (1H ,A6,' TO ',A6,3X,I10,3X,I10,' IN PERIODS',I5,
     1              ' THROUGH',I5)
             SUM=FLOW(ARC)
             PDEST=DEST(ARC)
  380        TARC=ARC
             ARC=NEXT(ARC)
             IF (ARC .GT. 0) GO TO 365
                WRITE (OUT,375) ORIG(TARC),PDEST,SUM,FLOW(TARC),
     1          PO(TARC),PD(TARC)
  390     CONTINUE
  400 STOP
      END
```

Listing of Subroutine SEARCH of POSTPRO

```
      SUBROUTINE SEARCH (NODE,INDEX,LOWER,LENGTH)
      IMPLICIT INTEGER (A-Z)
      DIMENSION LOWER(1)
      INDEX=0
      B2=-1
      T2=LENGTH
    2 IF (T2-B2 .GT. 1) GO TO 3
      INDEX=B2
      RETURN
    3 C=(T2+B2)/2
      KEYC=LOWER(C)
      IF (KEYC-NODE) 6,4,5
C --- FOUND
    4 INDEX=C
      RETURN
C --- LOWER HALF
    5 T2=C
      GO TO 2
C --- UPPER HALF
    6 B2=C
      GO TO 2
      END
```

NBS-114A (REV. 9-78)

U.S. DEPT. OF COMM.
BIBLIOGRAPHIC DATA SHEET

1. PUBLICATION OR REPORT NO.

2. Gov't. Accession No.

3. Recipient's Accession No.

4. TITLE AND SUBTITLE

EVACNET: Prototype Network Optimization
Models for Building Evacuation

5. Publication Date

6. Performing Organization Code

7. AUTHOR(S)

Richard L. Francis and Patsy B. Saunders

8. Performing Organ. Report No.

9. PERFORMING ORGANIZATION NAME AND ADDRESS

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, DC 20234

10. Project/Task/Work Unit No.

11. Contract/Grant No.

12. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS *(Street, City, State, ZIP)*

13. Type of Report & Period Covered

14. Sponsoring Agency Code

15. SUPPLEMENTARY NOTES

☐ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

16. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

This report presents the results of a pilot project conducted to explore how the evacuation and "evacuability" of buildings can be analyzed with the aid of mathematical network flow optimization models. As a research vehicle, Building 101, an eleven-floor building located on the Gaithersburg, Maryland campus of the National Bureau of Standards, has been studied; mathematical models pertinent to evaluating that building under a number of different circumstances have been developed and solved on the computer.

17. KEY WORDS *(six to twelve entries; alphabetical order; capitalize only the first letter of the first key word unless a proper name; separated by semicolons)*

Building evacuation; high-rise building design; network flow

18. AVAILABILITY ☒ Unlimited

☐ For Official Distribution. Do Not Release to NTIS

☐ Order From Sup. of Doc., U.S. Government Printing Office, Washington, DC 20402, SD Stock No. SN003-003-

☒ Order From National Technical Information Service (NTIS), Springfield, VA. 22161

19. SECURITY CLASS (THIS REPORT)
UNCLASSIFIED

20. SECURITY CLASS (THIS PAGE)
UNCLASSIFIED

21. NO. OF PRINTED PAGES
108

22. Price
$6.50