Selecting a Microcomputer for Process Control and Data Acquisition

AMI ARBEL

ABRAHAM SEIDMANN
SENIOR MEMBER, IIE
Department of Industrial Engineering
Tel-Aviv University
Tel-Aviv 69978, Israel

Abstract : A selection methodology is presented for evaluating and selecting a microcomputer for implementing computerized process control and data acquisition systems. The approach is based on establishing priorities that reflect the relative importance of the relevant factors in the system design. Resolution steps include the identification of major system tasks, definition of basic functional concerns with their system descriptors, and comparative evaluation of the candidate microcomputers. A detailed case study illustrates the application of the methodology in the context of energy management.

■ In recent years a wide variety of microcomputer equipment and supporting software has appeared on the market, with vendors competing for each potential application. It is likely that, for the same industrial application, different suppliers may present microcomputer systems which may be quite different both from a technology viewpoint and the adaptability of the computer to its operating environment.

The problem of microcomputer selection through process definitions and its inherent trade-offs has recently attracted considerable interest in the industrial engineering literature ([10, [16], and [17]). Existing computer selection methodologies deal primarily with large-scale systems, and they tend to concentrate on isolated aspects of the problem. These include queueing models [4], benchmark [3], simulation [8], functional and check-list approaches [2], and various scoring (or scaling) techniques [14]. The latter provide an attempt to deal with tangible as well as intangible factors of the selection process. Typically, one provides weights for the major selection categories and then proceeds to score each system element in its appropriate category. There are two basic deficiencies with such an approach. First, it cannot accommodate multicriteria selection problems. Second, the scoring process is quite arbitrary and does not provide a mechanism for detecting scoring inconsistencies. The approach taken in this paper is the Analytic Hierarchy Process (AHP) [11] as the selection process. The AHP methodology compares very favorably with other multiattribute decision approaches in its ability to capture the importance of the various elements of the problem and suggest a course of action. Also, user acceptability and confidence in the analysis provided by the AHP is high compared to other methods [13].

To date there is no single, unified, systematic, and accepted methodology that is capable of dealing with the tangible as well as the intangible aspects of the problem. This paper provides such an approach for the problem of selecting a microcomputer to be incorporated in industrial control and data logging systems ([7] and [10]). This approach considers functional, operational, and technological aspects of the problem in a way that incorporates all those elements into a unified assessment methodology.

In the discussion to follow, standard microelectronic terminology will be used. Interested readers may refer to [1], [5], [6], [9], [15], and [16] for the required technical definitions.

THE ASSESSMENT MODEL

The assessment model considers the tangible and intangible

Received September 1982; revised October 1983. Paper was handled by the Editor-in-Chief.

0740-817X/84/0300-0073/$2.00/0 © 1984 IIE

aspects of the microcomputer selection problem. The assessment is based on arranging the factors mentioned above in a hierarchy whose levels are those clusters of factors relevant to the selection problem. The analytical assessment of the hierarchy is carried out through the methodology of the Analytic Hierarchy Process developed recently by T. L. Saaty [11], [12]. This methodology, in addition to being a problem-structuring tool, provides a systematic approach for assessing priorities of the components of the problem. Details of the AHP are given in the Appendix.

The first step in building the model is the identification of clusters of factors or, in the AHP terminology, the *levels* of the hierarchy. These levels are described in Fig. 1.

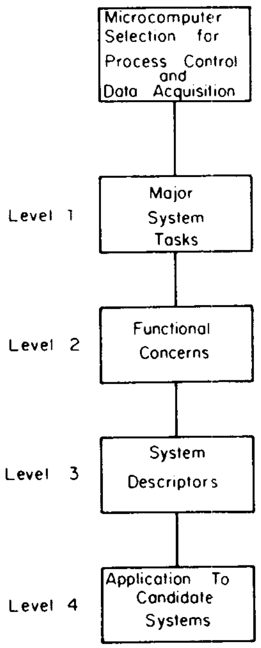

Fig. 1. The four system levels.

Level 1: Major Tasks

It should be emphasized that a listing of these major tasks *does not* imply that they are of equal importance to the operational success of the system. In fact, the model's main contribution is in assessing the *priorities* of these tasks (and of other members of the hierarchy) so that the microcomputer system selected is the one most suitable for these tasks. The major tasks of the system include four universally defined elements [7]: (1) *Process Monitoring* provides the microcomputer with the current status data from the processes to which it is connected. (2) *Data Processing* is concerned with the detection of process disturbances and computes the required response. (This task may also include sensor correction and calibration as well as calculation of inferred variables.) The actual changes in the process are conducted by the (3) *Corrective Actions* task. This task is responsible, for example, for changing the physical configuration of a valve or rotating a stepping motor through one complete revolution. (4) The *Data Logging* task corresponds to all the internal information management activities such as gathering, sorting, and filing data on the operational parameters of a given system. The designators of these tasks are given in Table 1. These tasks proved to be useful in several applications; however, other applications may require modifying to suit particular needs. This flexibility of adding or deleting elements is an inherent feature of the AHP methodology.

Table 1: Designators for major system tasks.

Designation	Tasks
T1	Process monitoring
T2	Data processing
T3	Corrective actions
T4	Data logging

Level 2: Functional Concerns

These functional concerns are comprised of the following elements. *Programmability* refers to the potential for implementing changes in the basic control modes by means of software modifications. *Data Conversion* is concerned with the analog-to-digital (A/D) and digital-to-analog (D/A) conversions required on the input and output signals to permit communication with the discrete microcomputer circuits [1]. Other data conversion functions treat parallel-to-serial conversions along with synchronous and asynchronous data communication. *Computational Ability* is the function of the system in detecting variables' departures from predetermined setpoints as well as continuously calculating the required adjustments. *Capacity* is a surrogate term for the performance capability of the system to simultaneously handle several calling sites with acceptable response times. It considers the amount of CPU time required to run the control algorithms, to perform the control programs, and to produce all the printed records. *Configuration Flexibility* is the ability to expand the current system to enhance the performance of the basic configuration with minimal changes in software and hardware. The final functional concern is the *System Reliability*; it is measured by the expected mean time between failures (MTBF) and calculated from actual experience or by probabilistic evaluation of the component parts. Table 2 summarizes the functional concerns of level 2 and their designators.

Table 2: Designators of basic functional concerns.

Designation	Functional concerns
F1	Programmability
F2	Data conversion
F3	Computational ability
F4	Capacity
F5	Configuration flexibility
F6	Reliability

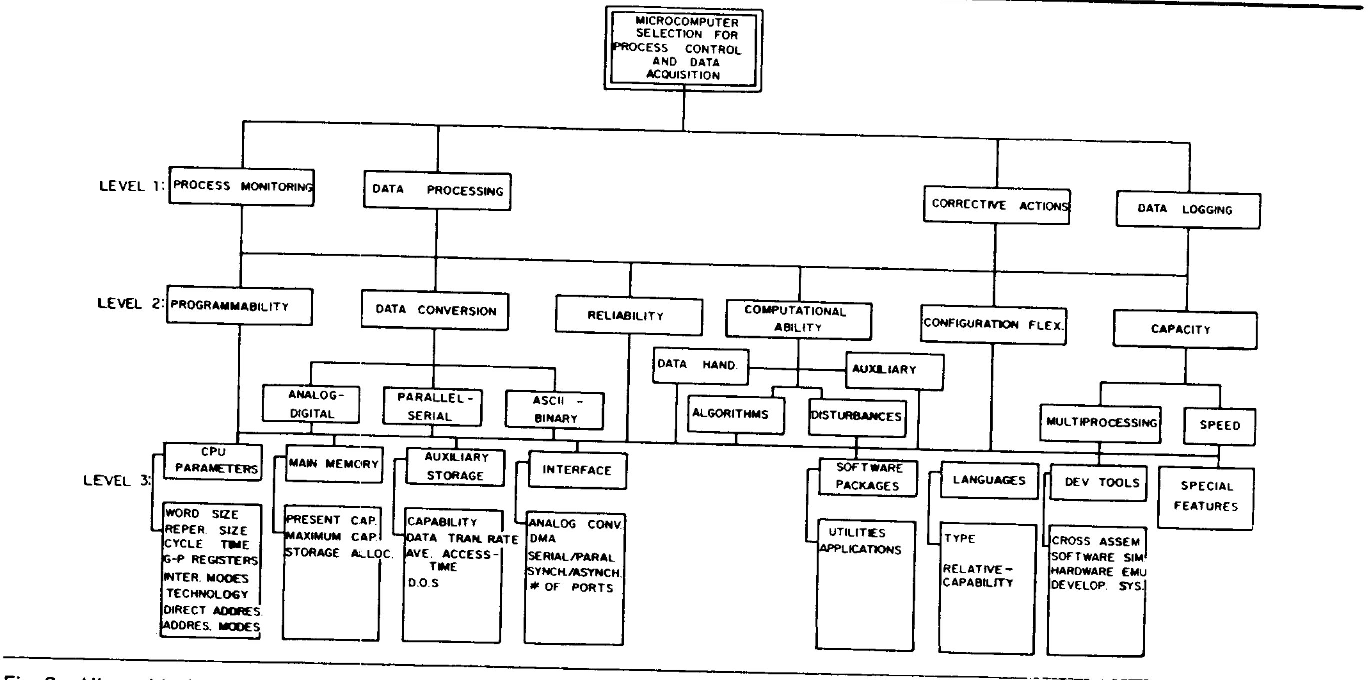

Fig. 2 . Hierarchical selection structure.

Level 3: System Descriptors

The *system descriptors* are broken into eight main technical categories. The *CPU Parameters* include the word size (bits), instruction repertoire, addressing methods, data transfer rate, data bus width, priority interrupt system, and clock cycle. These features can influence the effective computing speed, programming, and storage utilization. *Main Memory* is characterized by its capacity, addressing capability, and semiconductor technology. An important factor here is the relative allocation of storage space between the functional and technical ROM, RAM, and E(PROM) elements [16]. The *Auxiliary Storage* is usually a magnetic disk (floppy disk and winchester type) or a magnetic tape that stores data or programs not immediately required by the control computer. The important parameters here are the storage capacity, average access time, data transfer rate, and device operating system. The *Interface Options* are related to the communication and conversion of signals associated with the peripheral devices, data channels, and controllers. It includes such options as direct memory access (DMA), hand-shaking operation, vectored interrupts, analog conversion, and serial and parallel I/O with synchronous or asynchronous data transfer. The *Software Packages* consist of two types: system utilities and special application programs. The system utilities include the operating system and standard software products such as data bases, records sorting, or graphical displays. The application programs execute all the special algorithms and control actions of the given system. The *Programming Languages* enable the user to modify or upgrade the application programs on hand. They are defined by the type of languages available, the supporting software (e.g., compilers, interpreters, and assemblers), and by special elements such as floating point, complex variables, and interrupt-handling capabilities. The *Development Tools* aid in debugging both software and hardware during the design or modification phases. The final element of the "system descriptors" discussed here include *Special Features* which may be useful in certain applications. These include such items as self-contained diagnostic circuitry, special fail-safe design measures, stall alarm, or power failure detection. Table 3 depicts the major systems descriptors and their designators.

Table 3: Designators of major system descriptors.

Designation	Major system descriptors
S1	CPU parameters
S2	Main memory
S3	Auxiliary storage
S4	Interface options
S5	Software packages
S6	Programming languages
S7	Development tools
S8	Special features

Figure 2 depicts the complete assessment hierarchy. It should be pointed out, however, that the concept of hierarchical structuring and decomposition is flexible enough to accommodate selection problems somewhat different from the particular one considered here. This flexibility manifests itself through the ability to add or delete elements so that an appropriate presentation of the problem is provided.

The hierarchical structuring of the decision (selection) problem completes the first part of the problem. The second part, concerned with the analytical assessments of priorities, is presented next.

Table 4: Characteristic parameters for candidate systems.

Parameters	Systems A	B	C	D
CPU				
Word size (bits)	8	8	8	16
Repertoire size	38	42	28	72
CLK cycle time (μsec)	0.12	0.15	0.29	0.08
G-P registers	8	4	8	16
Interrupt modes	Vectored	Vectored & nonvectored	Fast & normal	Vectored
Technology	NMOS	NMOS	CMOS	NMOS
Direct addressing to (KBytes)	64	64	48	800
Addressing modes	6	10	8	14
MAIN MEMORY				
Present capability (KBytes)	32	48	32	256
Maximum capability (KBytes)	64	48	48	500
Storage allocation	Good	Fair	Very good	Very good
AUXILIARY STORAGE				
Capability (KBytes)	280	320	720	320
Data transfer rate (KBits/sec)	25	30	54	30
Average access time (msec/record)	175	175	310	250
DOS	Vendor's	CPM & vendor's	Vendor & partial CPM	CPM & vendor's
INTERFACE OPTIONS				
Analog conversion	Very good	Very good	Good	Fair
DMA	Available	Available	Available	Available
Serial/parallel conversion	Moderate	Slow	Moderate	Fast
Synchronous/asynchronous modes	Available	Not available	Available	Available
Number of ports	16	12	24	10
SOFTWARE PACKAGES				
System utilities	Few	Many	Many	Moderate
Application packages	Need to develop	Good	Very good	Need to develop
LANGUAGES				
Type[a]	A, PL, F, B	A, PS, CB, B	A, F, B	A, PL, PS, C
Relative capabilities	Moderate	Fair	Very good	Good
DEVELOPMENT TOOLS				
Cross assemblers	Not available	Available	Available	Not available
Software simulators	Very good	Good	Good	Fair
Hardware emulators	Good	Fair	Very good	Excellent
Development systems	Excellent	Very good	Good	Fair
SPECIAL FEATURES	Many/important	Few/important	Moderate/important	Few/important

[a] A–Assembler, B–Basic, C–C, CB–Cobol, F–Fortran, PS–Pascal, PL–PL/M.

PRIORITY ASSESSMENT AND MICROCOMPUTER SELECTION

This section considers a specific microcomputer selection application for the task of designing computerized energy-management systems. The proposed systems will control heating, ventilation, air-conditioning, and lighting in industrial plants, hotels, and office buildings. Interface with the operator is via line printer, CRT, and dedicated function keyboard. These three components, along with some special items such as multichannel serial communication units, were customized for these systems and therefore excluded from the selection process discussed here.

In particular, the system designer is faced here with the problem of selecting one system out of four available, competitive, configurations at the same price range that were proposed by vendors. In some cases, of course, one is faced with many more systems; however, a prescreening process eliminates many which do not provide certain minimal performance level requirements. The specifications of these four systems are shown in Table 4.

Table 4 provides a technical summary of four typical candidate microcomputer systems representing the current state of technology in the competitive market. In this case, the system design has defined the auxiliary storage device as a dual drive for 5¼-in. floppy discs. Some of the entries in the table can be extracted from the vendors' data sheets (for example, CLK cycle time or the number of ports), where other entries are the result of assessing the relative contribution of the microcomputer component to the overall system performance (for example, the allocation of storage space among the ROM, RAM, and PROM elements, or the capabilities of the high-level programming languages).

In order to select the most appropriate system for the particular application, the decision elements described in the hierarchy of Fig. 2 have to be prioritized. The prioritization process starts with the top level and progresses downward until the bottom level is reached and specific systems priorities are derived. The process will not be repeated here in detail but will be highlighted only.

The first step is concerned with prioritizing the *tasks* of the microcomputer system. The efforts here are directed toward identifying which task contributes more toward the overall operational success of the system. The answers to these questions are summarized in Table 5.

Table 5: Pairwise comparisons of tasks.

	T1	T2	T3	T4	Priorities (P_1)
Proc. Mon. (T1)	1	1/2	1	2/3	0.18
Data Proc. (T2)		1	2	1	0.34
Proc. Con. (T3)			1	2	0.25
Data Log. (T4)				1	0.23

$\lambda_{max} = 4.207$ C.R. = 0.076

Only the upper triangular part of the matrix is shown since the matrix is reciprocal; i.e., $a_{ij} = 1/a_{ji}$. The entries of the matrix itself are taken from the ratio scale shown in Table 6. For example, in comparing data processing (T2) and process monitoring (T1) relative to their importance to the system performance, a slight preference for T2 over T1 is indicated through the a_{21} element of the matrix. After the matrix is filled out, the largest eigenvalue (λ_{max}) is found, and the eigenvector corresponding to this eigenvalue is the required priority vector (see Appendix and [11] for details). The consistency check is provided through the *consistency ratio* (C.R.), which is required to be less than 0.1 for acceptable results ([11] and Appendix).

Table 6: Comparison scale.

Intensity of Importance	Definition
1	Equal importance
3	Moderate importance of one over another
5	Essential or strong importance
7	Very strong or demonstrated importance
9	Absolute importance
2, 4, 6, 8	Intermediate values between adjacent scale values

When the C.R. exceeds 0.1, the judgments leading to this evaluation are inconsistent and should be reconsidered. New judgments have to be provided to improve consistency. Improvement of consistency can also be provided (when all else fails) by mathematical means whereby a reciprocal matrix is fitted to the current judgments through a least-squares fit procedure [11].

Next, one considers the elements in level 2 relative to their contribution to *each* of the elements in level 1. Therefore, in considering elements of level 2 relative to *process control*, we have the results summarized in Table 7 (the designations are taken from Tables 1 and 2). Note that F5 was judged not to be relevant to task 1 (i.e., its priority relative to T3 is zero, and it is not included in the comparison matrix).

Table 7: Concerns relative to T3.

	F1	F2	F3	F4	F6	Local priority (w_1)
F1	1	1/2	1/3	1/3	1	0.10
F2		1	1	1	3	0.26
F3			1	1	2	0.26
F4				1	3	0.28
F6					1	0.10

$\lambda_{max} = 5.039$; C.R. = 0.0088

The priority vector w_1 is relative only to task 1. Continuing in this manner, one constructs a comparison matrix relative to each of the remaining tasks and obtains for each one its corresponding local priority vector w. To convert these local priorities to *global* measures we proceed as follows:

Let

$$W_{1,2} = [w_1 \ w_2 \ w_3 \ w_4] \tag{1}$$

be a matrix whose columns are the local priority vectors of the elements in level 2 relative to elements in level 1. If, in addition, the global priority vector of elements in level 1 is denoted by p_1, then the global priority vector of elements in level 2 is given by

$$p_2 = W_{1,2}p_1 \tag{2}$$

and, in general,

$$p_i = W_{j,i}p_j, \quad i > j, \tag{3}$$

where $W_{j,i}$ is defined by the product

$$W_{j,i} = (W_{i-1,i}) \ldots (W_{j+1,j+2}) \cdot (W_{j,j+1}). \tag{4}$$

Table 8 lists the four local priority vectors. Multiplying each column with the priorities of the tasks (taken from Table 5), the last column of global priorities is derived. The global priorities indicate the importance of each functional concern relative to the overall goal of selecting the system rather than to each specific task.

After establishing the global priorities for the functional concerns, one proceeds down the hierarchy to the level of system descriptors. Again, one compares all elements in this level relative to each of the elements in the preceding level. Summary of the local priorities of the eight groups of

Table 8: Summary for level 2.

	w_1	w_2	w_3	w_4	Global priority (p_2)
F1	0.10	0.14	0.00	0.27	0.13
F2	0.26	0.23	0.29	0.27	0.26
F3	0.26	0.20	0.13	0.09	0.17
F4	0.28	0.21	0.26	0.27	0.25
F5	0.00	0.12	0.10	0.00	0.06
F6	0.10	0.10	0.23	0.09	0.13

descriptors and their global priority is shown in Table 9. The global priority vector, p_3, is obtained from

$$p_3 = W_{2,3}\, p_2 = W_{2,3} W_{1,2}\, p_1 \,, \tag{5}$$

where $W_{2,3}$ is the matrix whose columns are the local priority vectors of elements in level 3 relative to elements in level 2 (these are the columns of Table 9).

We are now in a position to compare the systems themselves whose properties were summarized in Table 4. Usually, at this point, one is required to provide a "score" for each of the system parameters and, by weighting them somehow, one provides a global measure of system performance. The application of the AHP methodology allowed us to derive the weights for the system parameters in a systematic, consistent way. The "scores" for the systems are, again, derived through paired comparison relative to each of the parameters, which assures consistency and, therefore, quality of the overall process. For example, the CPU parameters are divided into word size (bits), repertoire, CLK cycle time, G-P registers, interrupt modes, technology, direct addressing, and addressing modes. By performing pairwise comparisons among these parameters one can assess their relative importance, which dictates how to split the overall rating of 0.29 (for CPU parameters) among them; for brevity we skip this simple step. In comparing the four systems relative to work size we obtain Table 10.

The first three systems are equal with respect to word size (8 bit), and, for the control application discussed here, D was judged to have a moderate dominance (3) over all of them, as is indicated in the pairwise comparison. Note that the comparison between the 8- and 16-bit capabilities was done based on judgment rather than technical ratio (16/8 = 2).

Table 10: Systems relative to word size.

	A	B	C	D	Local priority
A	1	1	1	1/3	0.17
B		1	1	1/3	0.17
C			1	1/3	0.17
D				1	0.50

$\lambda_{\max} = 4.00$; C.R. = 0.00

Again, continuing the comparison of the systems relative to all the parameters, the global priority for the system is obtained from

$$p_4 = W_{3,4}\, p_3 = W_{3,4} W_{2,3} W_{1,2}\, p_1 \,. \tag{6}$$

This yields the final ranking for the systems:

System A = 0.18
System B = 0.23
System C = 0.32
System D = 0.27.

Note that if one changes the relative importance of the tasks (i.e., changes p_1), the sensitivity at the system level is derived from

$$\hat{p}_4 = W_{3,4} W_{2,3} W_{1,2} (p_1 + \Delta p_1)$$

$$= p_4 + W_{3,4} W_{2,3} W_{1,2} \Delta p_1 \,, \tag{7}$$

where p_4 is the "old" priority vector and the Δp_1 are the changes in the task priorities. Small changes in p_1 did not change the overall rating, so that system C could be concluded to be a "robust" choice for the computerized energy-management system.

CONCLUDING REMARKS

This paper has presented a new methodology for the selection of a microcomputer system. The approach is based on the Analytic Hierarchy Process that provides the means of structuring and analyzing this complex, multifaceted problem. The novel feature of this approach is that, instead of merely providing a set of checklists that do

Table 9: Summary for level 3.

	w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_{10}	w_{11}	w_{12}	Global priority (p_3)
S_1	.18	.20	.46	.29	.44	.43	.32	.39	.29	.24	.10	.27	.29
S_2	.16	.00	.00	.20	.17	.16	.32	.17	.10	.00	.22	.11	.26
S_3	.08	.00	.00	.00	.00	.13	.11	.12	.00	.00	.10	.11	.11
S_4	.00	.60	.29	.35	.39	.00	.00	.00	.26	.55	.18	.24	.10
S_5	.18	.20	.25	.16	.00	.00	.12	.00	.12	.00	.20	.00	.10
S_6	.19	.00	.00	.00	.00	.28	.14	.32	.00	.21	.10	.13	.08
S_7	.15	.00	.00	.00	.00	.00	.00	.00	.22	.00	.10	.13	.05
S_8	.06	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.01

not aid the decision itself, we present a way that provides a structured, consistent, and accountable recommendation for a selection decision.

The results of the example presented in this paper should not be taken as a general recommendation for selection rules in these cases. Instead, one should use the proposed methodology with judgment to reflect one's own goals.

APPENDIX: The Analytic Hierarchy Process (AHP)

This approach is based on three major components:

1. The AHP starts by decomposing a complex decision problem into a hierarchy; each level consists of a few manageable elements and each element is, in turn, decomposed into another set of elements. The process continues down to the most specific elements of the problem, typically the specific courses of action considered, or the decision variables, which are represented at the lowest level of the hierarchy.

2. A measurement methodology is used to establish priorities among the elements within each stratum of the hierarchy.

3. A measurement theory is used to establish the priorities of the hierarchy and the consistency of the judgmental data provided by the group of respondents.

Deriving the actual local priorities of members in each level is done through a pairwise comparison between members of the level relative to a member of the adjacent upper level. These pairwise comparisons are summarized in an $n \times n$ comparison matrix A given by

$$A = \begin{bmatrix} w_1/w_1 & w_1/w_2 \; \ldots \; w_1/w_n \\ w_2/w_1 & w_2/w_2 \; \ldots \; w_2/w_n \\ \vdots & \vdots \qquad\quad \vdots \\ w_n/w_1 & w_n/w_2 \; \ldots \; w_n/w_n \end{bmatrix} . \tag{8}$$

The information displayed in this matrix is interpreted as follows: every element, a_{ij}, of the matrix A shows the relative contribution to the objective of the ith activity compared to the jth activity, i.e.,

$$a_{ij} = w_i/w_j, \qquad 1 \leqslant i \leqslant n, \qquad 1 \leqslant j \leqslant n . \tag{9}$$

The actual entries are derived by using the scale described in Table 6.

Note that the matrix is a *reciprocal* matrix, i.e., $a_{ij} = 1/a_{ji}$. Therefore, whenever the ijth element of the matrix is specified, the jith position is automatically determined by its reciprocal value. To actually recover the weights, w_i, rather than their ratios that are given in Equation (8), we proceed as follows. Note that the matrix A in Equation (8) is of unity rank and, therefore n - 1 of its eigenvalues are equal to zero; furthermore,

$$\sum_{i=1}^{n} \lambda_i = \text{Trace (A)} \triangleq \sum_{i=1}^{n} a_{ii} = n , \tag{10}$$

and therefore the nonzero eigenvalue is equal to n. It is easily verified that $Aw = nw$, from which it follows that w is the (normalized) eigenvector associated with the largest eigenvalue of the matrix A in Equation (8). This represents the perfectly consistent case, where $a_{ij} = a_{ik}a_{kj}$, $\forall\ i, j, k$. In practice, the elements of the matrix A are estimated through the use of the scale whose values are given in Table 6. In general the elements of matrix A satisfy $a_{ij} = w_i/w_j + \epsilon_{ij}$, where ϵ_{ij} is some error that represents inconsistencies in judgment, and then $a_{ij} \neq a_{ik}a_{kj}$. It can be shown that the largest eigenvalue of the matrix A, $\lambda_{\max}$, satisfies $\lambda_{\max} \geqslant n$, where equality holds for the perfectly consistent case only. A *consistency index* is now defined as

$$\text{C.I.} = \frac{\lambda_{\max} - n}{n - 1} , \tag{11}$$

which is zero in the perfectly consistent case. To assess the consistency derived in Equation (11), we compare it to the worst case, which will be the case of a pairwise comparison matrix whose entries are filled at random. Doing it for many samples and for various matrices, Saaty [11] has obtained the following:

n	1	2	3	4	5	6	7	8	9	10
R.I.	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49

(12)

where n represents the dimension of the matrix and R.I. is the *random index* evaluated through Equation (11) for these random matrices. Now one defines the *consistency ratio* (C.R.) as

$$\text{C.R.} = \text{C.I./R.I.}, \tag{13}$$

which is required to be less than 0.1 for acceptable results (more on this is found in [11]). A simple example will illustrate the basics of the calculations involved. Suppose that the comparison of three elements has yielded the following pairwise comparison matrix:

$$A = \begin{pmatrix} 1 & 4 & 5 \\ 1/4 & 1 & 2 \\ 1/5 & 1/2 & 1 \end{pmatrix} .$$

Using any computational routine to evaluate eigenvectors, or an approximation method ([11], p.19), yields the following vector of priority: $w = (0.683, 0.200, 0.117)$, with the largest eigenvalue $\lambda_{max} = 3.024$ (verify that $Aw = \lambda_{max} w$). Using Equation (11), the consistency index is C.I. = 0.012 and the consistency ratio, CR., is 0.021.

REFERENCES

[1] Berger, R. W., "Digital Logic and Arithmetic Background for Interfacing Microcomputers," *AIIE TRANSACTIONS* **12**, 1, 3-14 (March 1980).

[2] Booth, G. M., *Functional Analysis of Information Processing*, John Wiley & Sons, New York (1973).

[3] Borovits, I. and Neumann, S., *Computer Systems Performance Evaluation*, Lexington Books, New York (1979).

[4] Ferrari, D., *Computer Systems Performance Evaluation*, Prentice-Hall, Englewood Cliffs (1978).

[5] Givone, D. D. and Roesser, R. P., *Microprocessors/Microcomputers: An Introduction*, McGraw-Hill, New York (1980).

[6] Goodrich, J. L., "Interfacing Microcomputers for Control and Data Acquisition," *AIIE TRANSACTIONS* **12**, 2, 106-113 (June 1980).

[7] Harrison, T. J., *Minicomputers in Industrial Control*, Instrument Society of America, Pittsburgh (1978).

[8] Joslin, E. O., *Analysis, Design, and Selection of Computer Systems*, College Readings, Arlington, VA (1972).

[9] Marcy, W. M., "Digital Electronics for Microprocessor Applications in Control of Manufacturing Processes," *AIIE TRANSACTIONS* **12**, 1, 15-22 (March 1980).

[10] Marcy, W. M., "Preparing Functional Specifications for Computer Based Industrial Control Systems," *AIIE TRANSACTIONS* **12**, 4, 282-288 (December 1980).

[11] Saaty, T. L., *The Analytic Hierarchy Process*, McGraw-Hill, New York (1981).

[12] Saaty, T. L. and Vargas, L. G., *The Logic of Priorities*, Martinus Nijhoff Publishing, Boston (1982).

[13] Schoemaker, P.J.H. and Ward, C. C., "An Experimental Comparison of Different Approaches to Determining Weights in Additive Utility Models," *Management Science* **28**, 2, 182-196 (February 1982).

[14] Timmreck, E. M., "Computer Selection Methodology," *ACM Computing Surveys* **4**, 12, 199-222 (December 1973).

[15] Wolfe, P. M. and Yokley, M. T., "Microprocessor Based Process Control," *Fall IE Conference Proceedings*, 399-403 (1979).

[16] Wolfe, P. M., "Microprocessors Extend the Horizons for the Industrial Engineer," *Computers & Industrial Engineering* **3**, 4, 71-74 (December 1979).

[17] Zimmers, E. and Brown, L., "The Selection of Microcomputers for Industrial Data Acquisition," *Computers & Industrial Engineering* **4**, 4, 293-302 (December 1980).

Ami Arbel is a senior lecturer at the Department of Industrial Engineering, Tel-Aviv University. He received a BSME from the Technion, I.I.T., Israel, an MSME from the University of California at Berkeley, and an MS and a PhD degree from the Engineering-Economic Systems Department at Stanford University. Prior to joining Tel-Aviv University, Dr. Arbel was a senior research engineer with engineering consulting companies in the United States, consulting in areas of systems theory, optimization methods, decision analysis and strategic studies. He is a member of TIMS, ORSA, and IEEE.

Abraham Seidmann is a lecturer at the Industrial Engineering Department of Tel-Aviv University. He holds BSc and MSc degrees from the Technion, Israel Institute of Technology, and a PhD in industrial engineering from Texas Tech University. Dr. Seidmann's research interests include computerized production systems and microcomputers in operational systems and production management. He has published in these areas in such journals as *The International Journal of Production Research, Management Science*, and *Operations Research*. He is a senior member of IIE and a member of TIMS, ACM, and Alpha Pi Mu.