Extraction of Influential Programming Factors

Toshio Awano*

1. *Introduction*

A programming time calculation method has been reported, wherein elements which influence programming and debugging time were selected. [1]. As the distribution of programming time and debugging time is assumed to be a Weibull distribution, probability density function is

$$f(x)=\frac{\beta}{\alpha}(x-\gamma)^{\beta-1}\quad e^{-\frac{(x-\gamma)^{\beta}}{\alpha}}\qquad (x\geqq\gamma)$$
$$=0 \tag{1}$$

where

α : scale parameter

β : distribution form parameter

γ : distribution position parameter

Define 'risk factor' $g(x)=\int_t^{\infty} f(x)dx$ and, from Eq. (1),

$$g(t)=e^{-\frac{(t-\gamma)^{\beta}}{\alpha}}$$

If $\alpha=\mu_\alpha$, $\beta=1$, $\gamma=0$, we get

$$g(t)=e^{-\frac{t}{\mu_\alpha}}$$

$g(t)$ is decided by μ_α. In the above-mentioned paper, A: Composition of program, B: Number of I/O entries, C: Number of items and D: Difficulties in Process are adopted as the elements of μ_α.

This paper describes the method of selecting these elements.

2. *Analysis of influential programming factors*

First, it is important to construct perfect enquete items which can represent the field of these problems. In this paper, investigation covers the following 21 items selected after consideration.

2.1 *Inquiry*

Answers for the enquete are obtained from 66 programmers at the Central Electronic Computer Center of NTT, showing each programmer's classification judgement (Table 2.1). The classification, as shown in Table 2.1, separated into

This paper first appeared in Japanese in Joho-Shori (Journal of the Information Processing Society of Japan), Vol. 13, No. 3 (1972), pp. 161～169.

* Nippon Telegraph and Telephone Public Corporation

Table 2.1 Enquete table for factor analysis in programming and its summary

Item / Classification	(1) Number of Data items	(2) EDP system construction	(3) Number of input/output files	(4) Degree of generalized program development	(5) Standardization of programming technique	(6) Program language used	(7) Facilities of electronic computer	(8) Rate of machine operation time in center	(9) Job connected or not	(10) Data processing system	(11) Level of faculty of system program	(12) Operating system	(13) Experience in using electronic computer	(14) Job network complexity	(15) Program quantity	(16) Number of programmers	(17) Knowledge of the subject	(18) Composition of program	(19) Programmer skill	(20) Programming term	(21) Difficulties in process
a	2	2	0	3	4	4	3	3	2	1	5	5	2	1	3	1	18	10	13	6	8
b	4	8	11	9	8	7	14	14	13	12	12	12	17	17	16	24	12	18	24	25	29
c	18	21	20	20	24	33	20	22	28	33	26	26	28	32	30	20	21	29	20	28	22
d	16	28	26	22	18	13	27	15	19	18	19	13	17	15	14	13	11	5	5	7	7
e	26	7	9	12	12	9	2	12	4	2	4	8	2	1	3	8	4	4	4	0	0

5 categories for each item, namely, a; the most influential, b; considerable influence, c; ordinary, d; slight influence, e; least influence. And correlation matrix (Table 2.2) is obtained from its summary by differential production method.

The items selected are:

(1) Number of data items.

All input/output items of data processing.

(2) EDP system construction.

CPU only, main and sub CPU, common file plural CPU etc.

(3) Number of input/output files.

All input/output files indispensable for program (excluding working file).

(4) Degree of generalized program development.

The degree of development of generalized program and possibility of its utilization.

(5) Standardization of programming technique.

Standardization of programming technique and the degree of its utilization possibility.

(6) Program language used.

FORTRAN, COBOL, ALGOL, ASSEMBLER etc.

(7) Facilities of electronic computer.

Memory capacities, peripheral equipment etc.

(8) Rate of machine operation in Center.

The rate of the electronic computer operation that influences program debugging time.

(9) Job connected or not.

Whether there is a job connected with the subject or not.

(10) Data processing system.

Batch processing, real time processing etc.

(11) Level of faculty of system.

The degree that faculties of operating system program etc. influence programming.

(12) Operating system.

Operating system faculty level of one-job processing, multi-job processing, time sharing system etc.

(13) Experience in using electronic computer.

Whether programmer has experience in actual using electronic computer or not.

(14) Job network complexity.

Degree of complexity in system designing of job.

(15) Progrm quantity.

Number of processing steps and passes in program.

(16) Number of programmers.

Programmers who can work on the subject.

(17) Knowledge of the subject.

Programmer's knowledge of and experience in the subject.

(18) Program complexity.

Complexity of faculty elements; check, merge, calculation, general procedure, report program generator, edit etc.

(19) Programmer skill.

Programmer experience and training, practical sense etc.

(20) Programming term.

(21) Difficulties in process.

2.2 *Factoring of the centroid method*

Factoring correlation matrix (Table 2.2) is accomplished by using the centroid theorem. Factor matrix obtained is shown in Table 2.3.

Next, in order to find simple structure and simplify the factor interpretation, an axis rotation of factor loading obtained is enacted. (In this paper, orthogonal rotation applies).

Factor loading matrices obtained by the above-mentioned procedure is shown in Table 2.4.

2.3 *Factor interpretation*

Table 2. 2 Correlation matrix

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1		.372	.414	.570	.576	.394	.078	.474	.202	.195	.158	.274	.058	.070	.082	.005	−.074	−.160	−.530	−.262	−.442
2	.372		.986	.970	.854	.628	.944	.736	.832	774	.846	.632	.434	.658	.634	.410	.126	.158	−.168	.234	.072
3	.414	.986		.970	.848	.608	.928	.812	.832	.766	822	.640	.746	.678	.654	.548	.492	.176	−.114	.288	.146
4	.570	.970	.970		.930	.710	.836	.840	.818	.776	.960	.852	.698	.654	.642	.448	.004	.180	−.208	.218	.020
5	.576	854	.848	.930		.914	.710	.972	.882	.880	.870	.872	.772	.772	.778	.460	.226	.446	−.028	.190	.126
6	.394	.628	.608	.710	.914		.538	.858	.878	.920	.866	.976	.812	.856	.876	.476	.526	.736	.272	.586	.336
7	.078	.944	.928	.836	.710	.538		.702	.858	.790	.878	.626	.824	752	.720	.590	.270	.292	.126	.460	.368
8	.474	.736	.812	.840	.972	.858	.702		.918	.906	.864	.914	.888	.894	.892	.804	.146	.614	.284	.652	.466
9	.202	.832	.832	.813	.882	.878	.858	.918		.990	.990	.932	.974	966	.958	.710	.480	.684	.364	.712	.532
10	.196	.774	.766	.776	.880	.920	.790	.906	.990		.984	.964	.966	.970	.970	.676	.550	.748	.402	.740	.538
11	.158	.846	.822	.960	.870	.866	.878	.864	.990	.984		.912	.960	.948	.942	.642	.550	.668	.344	.688	.488
12	274	.632	.640	.852	.872	.976	.626	.914	.932	.964	.912		.914	946	.960	.652	.542	.820	.444	.756	.532
13	.058	.434	.746	.698	.772	.812	.824	.888	.974	.966	.960	.914		.992	.992	.812	.520	.772	.546	.982	.702
14	.070	.658	.678	.654	.772	.856	.752	.894	.968	.970	.948	.945	.992		.994	.794	.472	.840	.588	.870	.716
15	.082	.634	.654	.642	.778	.876	.720	.892	.958	970	.942	.960	.992	.994		.782	.464	.858	.592	.872	.704
16	.008	.410	.548	.448	.460	.476	.590	.804	.710	.676	.642	.652	.812	.794	.782		.120	.640	.656	.838	.808
17	−.074	.126	.492	.004	.226	.526	.270	.146	.480	.550	.550	.542	.520	.472	.464	.120		.746	.612	.618	.514
18	−.160	.158	.176	.180	.446	.736	.292	.614	.684	.748	.668	.820	.772	.840	.858	.640	.746		.844	.934	.846
19	−.530	−.168	−.114	−.208	.028	.272	.126	.284	.364	.402	.344	.444	.546	.588	.592	.656	.612	.844		.910	.956
20	−.262	.234	.288	.216	.190	.586	.460	.652	.712	.740	.688	.756	.982	.870	.872	.838	.618	.934	.910		.994
21	−.442	.072	.146	.020	.126	.336	.368	.466	.532	.538	.488	.532	.702	.716	.704	.808	.514	.846	.956	.994	

Table 2. 3 Factor matrix obtained

	A_1	A_2	A_3	A_4	h^2 obt.	h^2guess	difference
1	.18	.63	.32	.29	.61	.58	.03
2	.73	.63	−.28	−.24	1.04	.99	.05
3	.79	.48	−.27	−.20	.97	.99	−.02
4	.77	.69	−.20	.17	1.12	.97	.15
5	.84	.54	.24	.09	1.05	.97	.08
6	.88	.15	.41	.03	.96	.98	−.02
7	.79	.27	−.49	−.18	.96	.95	.01
8	.93	.23	.09	.38	1.05	.97	.08
9	.99	.12	−.06	−.05	.99	.99	.00
10	.99	.07	.07	−.07	.98	.99	−.01
11	.98	.15	−.11	−.13	1.00	.99	.01
12	.97	.03	.26	.13	1.01	.98	.03
13	.98	−.18	−.12	.18	1.03	.99	.04
14	.98	−.14	−.03	.05	.98	.99	−.01
15	.98	−.17	.05	.07	.99	.99	.00
16	.76	−.28	−.29	.34	.84	.84	.00
17	.52	−.30	.23	−.59	.75	.75	.00
16	.77	−.57	.36	−.07	1.03	.93	.10
19	.47	−.86	.06	−.06	.96	.96	.00
20	.79	−.67	−.12	.10	1.08	.99	.09
21	.62	−.75	−.18	.04	.97	.99	−.02

Table 2. 4 Final factor matrix

	F_1	F_2	F_3	F_4	h^2 obt.	h^2 guess	difference
1	.21	−.09	.64	−.39	.61	.58	.03
2	.81	.43	.08	−.47	1.06	.99	.07
3	.80	.48	.08	−.31	.97	.99	−.02
4	.90	.22	.47	−.27	1.15	.97	.18
5	.59	.48	.64	−.28	1.06	.97	.09
6	.34	.68	.61	.00	.95	.98	−.03
7	.87	.45	−.11	−.09	.97	.95	.02
8	.69	.39	.64	.16	1.06	.97	.09
9	.68	.68	.26	.07	.99	.99	.00
10	.58	.74	.32	.09	.99	.99	.00
11	.71	.69	.19	.01	1.01	.99	.02
12	.48	.67	.56	.18	1.02	.98	.04
13	.67	.58	.27	.45	1.05	.99	.06
14	.59	.67	.27	.33	.98	.99	−.01
15	.52	.69	.33	.36	.98	.99	−.01
16	.64	.30	.16	.58	.86	.84	.02
17	−.08	.86	−.12	.04	.76	.75	.01
18	.12	.83	.27	.55	1.07	.93	.14
19	−.02	.59	−.11	.78	.96	.96	.00
20	.34	.60	.02	.69	.95	.99	−.04
21	.27	.53	−.13	.79	.99	.99	.00

Now we interpret the final factor matrix (Table 2. 4). For factor F_1 (the 1st factor), items that have high factor loading are data items 2. EDP system construction, 3. Number of input/output files, 4. Generalized program development, 7. Electronic computer facilities, 8. Machine operation time rate in Center, 11. System program faculty level and 13. Experience in using electronic computer. These are closely connected with an electronic computer and manufacturing.

Consequently, these items are considered factors governed by the electronic

Fig. 2. 1. Clusters of four influential programming factors

H factor (axis of electronic computer)

EDP system construction.
Number of input/output files.
Degree of generalized program development.
Facilities of electronic computer.
Rate of machine operation time in Center.
Level of faculty of system program.
Experience in using electronic computer.
Memory capacity.
Number of I/O channels.
Electronic computer used.
Special equipment (display etc.)
Random access equipment.
Bits/word.
Memory access time.
Whether the electronic computer for programming and that for actual use are the same or not.
Method of machine operation.

S Factor (axis of subject system)

Whether job is connected or not.
Data processing system.
Complexity of job network.
Quantity of program (input, output).
Composition of program.
Obscurity of system design condition.
Requirement of originality for programming.
Quality of system design report.
Number of users in organization.
Condition of response time.
Reliability of system design.
Condition of on line.
Number of input mode.
Number of output mode.
Relation with other system interface.
Rate of business order.
Rate of input and output.
Rate of logical judgement.
Rate of automatic check.
Rate of faculty of information storage and retrieval.
Rate of input/output of data.
Rate of control.
Lack of user experience.
Number of information sources of system.
Degree of change of subject system.
Number of faculties of system.
Number of system components.
Number of unsupermanuated system components.
Rate of faculty of decision making.
Pressure of timing.
Number of all documents.
Program mode (business, scientific, utility, individual process etc.)
Initial programming or not.

Average turn around time.
Quality of information source documentation.
Number of words at table and number of constant that is not at data base.

P Factor (programmer axis)

Number of data items.
Number of input variable item.
Number of output variable item.
Standerdization of programming techniqe.
Program language used.
Operating system used.
Unsufficiency of operating.
Complexity of propram interface.
Number of objective program instructions.
Number of iterative objective program instructions.
Number of source program instructions.
Abondonment rate of objective programs.
Abondonment rate of source programs.
Number of conditions of branch.
Number of words in data base.
Number of items and sort in data base.
Rate of conversion faculty.
Rate of generation faculty.
Number of sub-program.
Rate of magnification of program oriented language.
Rate of utilization of support program.
Document generated by programming steps.
Range wherein document generated by programming steps can be utilized.
Assembler or compiler used.

M Factor (axis of management)

Number of programmer.
Skill of programmer.
Programming term.
Difficulties in process.
Rate of programmer (upper, middle, lower).
Average experience of programmer (for programming language).
Average experience of programmer (for subject business).
Rate of system design participation of programmer.
Average servicing term of programmer.
Maximum programmers for objective job.
Management ability.
Number of joint system design party.
Machine operation excluding programmer.
Programmer who works other than programming.
Human error.
Degree of receiving information about objective system design.
Rate of analysts (upper, middle, lower).
Rate of utilization of special equipment.
Standardization of plan and procedure.
Examination of documents.
Rotation of programmer.
Working condition of programmer.
Condition of grouping programmers.
Health.

computer itself, say, H factor.

For factor F_2 (the 2nd factor), the items that have high factor loading are 9. Job connected or not, 10. Data processing system, 14. Job network complexity, 15. Program quantity and 17. Knowledge of the subject. All these items are closely connected with the subject system. They are the S factor.

For factor F_3 (the 3rd factor), items that have high factor loading are 1. Data item, 5. Standardization of programming technique, 6. Program language used and 12. Operating system. All these items are those which the programmer himself must decide upon and develop. Consequently, these are considered as the factor closely connected with the programmer. They are the P Factor.

Lastly, for factor F_4 (the 4th factor), the items that have high factor loading are 16. Number of programmers, 19. Programmer skill, 20. Programming term and 21. Difficulties in process. These are greatly influenced by the management. Consequently, they are called the M Factor.

In conclusion, through the above-mentioned interpretation, four independent factors are found, H, S, P and M. Groups of these factors are shown in Fig. 2. 1.

3. *Decision on Elements which influence programming.*

In the result of factor analysis, these four factors influence programming and debugging. We get four clusters, where each cluster has one root, as shown in Fig. 2. 1.

Selecting comparably measurable and largely influential elements from each cluster, then the Number of I/O files from the 1st cluster, Program complexity from the 2nd cluster, Data items from the 3rd cluster and Difficulties in process from the 4th cluster are selected.

Elements A, B, C and D, which decide μ_α, are provisionally obtained by the above-mentioned procedure.

Next, it must be determined whether A, B, C and D interact with each other or not.

They are examined by analysis of variance of programming time (Table 3. 1) in which y is actual programming time (week). By analysis of variance (I), ratios of variance between the elements are $A\times B$: 0. 896, $A\times C$: 0. 024, $A\times D$: 1. 296, $B\times C$: 0. 274, $B\times D$: 0. 097, $C\times D$: 0. 035. They are all smaller than $F^1{}_5(0.05)=6.61$, therefore interation between them is negligible. Next, invariables greater than errors term e in column V are A, B, D and $A\times D$. To make sure of this fact, we pool all the element into the errors term e in column V, excluding the above invariables greater than errors term e. We get (II). In (II), F_0 of interaction $A\times D$ is 2. 193, which is smaller than $F^1{}_{11}(0.05)=4.84$, where interaction $A\times D$ can be considered negligible.

For the debugging term, the procedure is the same. As a result, similarly, Table 3. 1 Analysis of variance of programming time

Allocation

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	r
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1. 62
2	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	4. 51
3	1	1	1	2	2	2	2	1	1	1	1	2	2	2	2	2. 66
4	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1	4. 42
5	1	2	2	1	1	2	2	1	1	2	2	1	1	2	2	4. 34
6	1	2	2	1	1	2	2	2	2	1	1	2	2	1	1	2. 57
7	1	2	2	2	2	1	1	1	1	2	2	2	2	1	1	2. 72
8	1	2	2	2	2	1	1	2	2	1	1	1	1	2	2	4. 43
9	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	2. 57
10	2	1	2	1	2	1	2	2	1	2	1	2	1	2	1	9. 02
11	2	1	2	2	1	2	1	1	2	1	2	2	1	2	1	5. 00
12	2	1	2	2	1	2	1	2	1	2	1	1	2	1	2	5. 14
13	2	2	1	1	2	2	1	1	2	2	1	1	2	2	1	6. 11
14	2	2	1	1	2	2	1	2	1	1	2	2	1	1	2	7. 49
15	2	2	1	2	1	1	2	1	2	2	1	2	1	1	2	5. 30
16	2	2	1	2	1	1	2	2	1	1	2	1	2	2	1	12. 36
	A	B	$A\times B$	C	$A\times C$	$B\times C$		D	$A\times D$	$B\times D$		$C\times D$				

Analysis of variance (I)

Element	S	ϕ	V	F_0
A	41. 345	1	41. 345	7. 863*
B	6. 734	1	6. 734	1. 281
C	0. 903	1	0. 903	0. 172
D	24. 059	1	24. 059	4. 576
$A\times B$	4. 709	1	4. 709	0. 896
$A\times C$	0. 126	1	0. 126	0. 024
$A\times D$	6. 812	1	6. 812	1. 296
$B\times C$	1. 440	1	1. 440	0. 274
$B\times D$	0. 511	1	0. 511	0. 097
$C\times D$	0. 185	1	0. 185	0. 035
e	26. 290	5	5. 258	
			$F_5{}^1=(0.05)=6.61$	

Analysis of variance (II)

Element	S	ϕ	V	F_0
A	41. 345	1	41. 345	13. 311**
B	6. 734	1	6. 734	2. 168
D	24. 059	1	24. 059	7. 746*
$A\times D$	6. 812	1	6. 812	2. 195
$e'\begin{pmatrix} e & , B\times C \\ C & , B\times D \\ A\times R, & C\times D \\ A\times & \end{pmatrix}$	34. 164	11	3. 106	
			$F_{11}{}^1(0.05)=4.84$ $F_{11}{}^1(0.01)=9.65$	

N_1B_1 ** Level of significance 1%
 * Level of significance 5%

the interaction between A, B, C and D can be considered negligible.

Consequently, elements A, B, C and D are selected for the influential programming factor.

Next, the average time for each level must be calculated. Level is the class in one element, shown at Table 3. 2. It is very easy to calculate average time for every level, if the actual data are very abundant. However, at present, data are so few that they are calculated by the method of linear inner and outer inserts and we got Table 3. 3. Using Table 3. 3, we can get 2 numerical tables,

Table 3.2 Factor levels

	Element *A*	Element *B*	Element *C*	Element *D*
No. of level	4	5	3	5
Range of level	1∼4	2∼6	1∼3	1∼5

Table 3.3 Programming term (week).

Level	Programming term				Debugging term		
	Element *A*	Element *B*	Element *C*	Element *D*	Element *A*	Element *B*	Element *E*
1	3,409		4,907	2,887	1,648		3,131
2	5,246	4,581	5,058	3,489	4,345	4,053	3,519
3	8,920	4,927	5,968	4,694	7,719	4,123	4,004
4	12,594	5,619		5,899	11,093	4,120	4,489
5		6,311		7,104		4,296	4,975
6		7,003				4,383	
0	5,018	5,056	5,012	3,938	4,140	4,134	4,134

Table 3.4 Elements and Levels

Element	Illustration of Element	Judgment of Level		Level
A Composition of Program	① Check, ② File Merge, ③ Central Calculation, ④ Report Editing are selected as Program Function Elements. A Program has at least One Function Element. The Level used is decided by Number of Function Elements that the Program must have. Error List is included in 'Report Editing .	Number of Function Element that Program must have	1	1
			2	2
			3	3
			4	4
B Number of I/O Entries	Level is decided by All the Number of I/O Entries that the Program Requires, but the Work File to compensate for the Lack of Internal Memory in the Same Program is omitted.	Number of I/O Program Files	2	2
			3	3
			4	4
			5	5
			Above 6	6
C No. of Items	Level is Decided by All the Number of Items Included in Above-Mentioned I/O File, but Relay I/O Entries for the Succeeding Program is omitted.		1∼100	1
			101∼300	2
			301∼	3
D Difficulties in Process	Synthetic Supplement Factor of Above-Mentioned Factors Ex., ① Programming Itself ② Skill of Programmer ③ Subroutine etc. ④ Other Factors	In Programming	Very Easy	1
			Easy	2
			Ordinary	3
			Slightly Difficult	4
			Considerably Difficult	5

Eeach Above-Mentioned Factor has a 0 Level and it is used in the Case of Unreliable Information.

1) Correspondence of General 'Risk' and Individual 'Risk' and 2) Calculation of Individual Program and if Table 3.4 is given, we can estimate programming time [1].

4. *Summary*

We extracted four influential programming factors by centroid method, as we said above. We got four clusters, each of which has a principal axis of an influential programming factor. Next, we decided four elements, namely, A: Composition of Program, B: Number of I/O Entries, C: Number of Items and D: Difficulties in Process, each of which was selected from individual cluster. If a single element represents a cluster, any element is adoptable.

Four elements obtained in this way were used for Calculation and Estimation of EDPS System Design Time on more reliable basis than before [1]. Especially it is very effective for COBOL programming language of large scale business, and has already been used for material management, account, payment and personal statistics and estimation of construction works etc. which have each above ten thousand steps and in success at present.

Reference

[1] Awano, T: Calculation and Estimation of EDPS Sytem Design Time in Days, *Information Processing in Japan* Vol. 11. (1971).