Studia Humana
Volume 5:1 (2016), pp. 31—39
DOI: 10.1515/sh-2016-0003

Continuous-Logical Methods in Mathematical Economics

Vitaly I. Levin

Mathematics Department,
Penza State Technological University,
Russia

e-mail: vilevin@mail.ru

Abstract:
An application of continuous logic for the mathematical description of economical systems is given. Parallel, sequential, parallel-sequential and sequential-parallel systems are calculated using continuous logic (CL) methods.
Keywords: continuous logic, logical determinant, mathematical economics.

1.

In 1978 the author for the first time indicated a possibility to use continuous logic (CL) for solution of the optimal tasks of mathematical economics. The application of CL is represented by operations of maximum and minimum. Also, there was detected a possibility to evaluate industrial systems with the help of CL of various metrics: speed, productivity, and the modes of operations of these systems determined by ratio 'more' or 'less' between temporary parameters of operations, executed there. Nevertheless, there were not developed methods of the CL-analysis and synthesis of the optimal schedules of execution of operations in systems for a long time. Nowadays CL-methods in mathematical economics represent an independent branch of this science, with the research methodology and significant results. In the given paper let us review some of these results.

2.

Let us consider a sequential system with m blocks executing m various operations. In the system, n jobs consisting of m indicated operations simultaneously occur. The operation execution i for the job j is given by the matrix $A = \|a_{ij}\|$. These jobs are started in the system and pass in the blocks $1,\ldots,m$ in the same order $1,2,\ldots,n$. Thus, each job passes in the next block i at once after an output from the previous block and release of block i from a prior operation. We assume that the speed of the given system is characterized by time of passing of all jobs through all blocks:

$$T = A^{\vee} = |a_{ij}|^{\vee}, \tag{1}$$

where T is a disjunctive logical determinant (LD) $A^{\vee}$ from a matrix A of operations. The LD $|a_{ij}|^{\vee}$ is a function $\{a_{ij}\} \to a^r$, where a^r is the r-th element of a matrix A. The formula (1) reduces a calculation (analysis) of speed of the sequential system to a calculation (analysis) in the LD $A^{\vee}$. In the LD there is a CL function satisfying (1) which expresses time of operation a_{ij}. Due to T the speed of job (productivity) of the system is expressed by $v = n/T$.

3.

For a sequential system, the average load for the k-th block is defined thus:

$$R_k = \sum_{j=1}^{n} a_{kj} / T \tag{2}$$

and let $r_k(t)$ be an instant load at the arbitrary moment t, then the average load of the system and its instant load at the moment t are defined as follows:

$$R = \sum_{k=1}^{m} \sum_{j=1}^{n} a_{kj} / mT, \; r(t) = \sum_{k=1}^{m} r_k(t)/m . \tag{3}$$

Equations (1) – (3) for the calculation of characteristics of average load of the block and of the system show that first the LD $A^{\vee}$ from a matrix A of job time should be calculated. For finding the characteristics of instant load it is necessary to determine matrixes of moments of beginning of jobs in $\bar{T} = \|\bar{t}_{ij}\|$ blocks and ending of jobs in $\underline{T} = \|\underline{t}_{ij}\|$ blocks. Here $\underline{t}_{ij}$ ($\bar{t}_{ij}$) are moments of the beginning (ending) of the job j in the block i. Let $A_{rk}^{\vee}$ be a disjunctive LD from r first lines and k first columns of a matrix A and $A^* = \|A_{rk}^{\vee}\|$ be a matrix attached to A. We know that

$$\bar{T} = A^* . \tag{4}$$

It is clear that

$$\bar{T} - \underline{T} = A . \tag{5}$$

By calculating A^* with the help of wave algorithm, we can obtain $\bar{T}$ and then $\underline{T}$ from (5). Hence, a characteristic $r_k(t)$ of instant load of the block can be defined in the following manner:

$$r_k(t) = \begin{cases} 1, \; t \in \bigcup_{i=1}^{n} [\underline{t}_{ki}, \bar{t}_{ki}], \\ 0, \; \text{otherwise.} \end{cases} \tag{6}$$

4.

Let us consider a special class represented by sequential systems dependent on time by arrival of jobs, in which there is both order and moment of arrival of jobs in the system. If this order is $1, 2, \ldots, n$, and an appropriate moment is $\tau_1, \tau_2, \ldots, \tau_n$, time of passing of all jobs through the system is af follows:

$$T = \begin{vmatrix} \tau_1 & \tau_2 - \tau_1 & \cdots & \tau_n - \tau_{n-1} \\ a_{11} & a_{12} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{vmatrix}^{\vee} - \tau_1. \tag{7}$$

From (7) it is evident that any system of this class is equivalent to a sequential system by arrival of jobs having an extended matrix of job time.

5.

Let us concentrate now on a parallel system from m functional one-type blocks $1,2,\ldots,m$ for the execution of n one-type jobs $(n \geq m)$. The execution time of job j in the block i is given by a matrix $A = \|a_{ij}\|$. Jobs arrive in the system by the given order $P_n = (j_1,..., j_n)$. Thus, at the moment $t = 0$ the jobs $j_1,..., j_m$ boot in blocks $1,\ldots,m$, which begin jobs and arrive in the process of release by the consequent jobs $j_{m+1},..., j_n$. Any characteristics of speed of the given system are expressed by a vector $t = (t_1,..., t_n)$, where t_k is the moment of ending of job j_k. The execution time of all n jobs and speed of their execution (productivity) is defined thus:

$$T = \bigvee_{i=1}^{n} t_i, \qquad v = n/T, \tag{8}$$

where $\vee = \max$ is an operation of CL-disjunction. The vector t is calculated easily in the case of homogeneous system, where the blocks are identical on speed, so the matrix of job time A degenerates in a vector of job time $a = (a_1,..., a_n)$, where a_j is an execution time of job j in any block. Let $\tilde{a} = (\tilde{a}_1,..., \tilde{a}_n)$ be a vector distinguished from a vector a by rearrangements of elements according to the given order P_n of starting jobs. Then the moments t_k are connected by such a recurrence:

$$t_k = \begin{cases} \tilde{a}_k, \ 1 \leq k \leq m; \\ \begin{vmatrix} t_1 \\ \cdots \\ t_{k-1} \end{vmatrix}^{(k-m)} + \tilde{a}_k, \quad m+1 \leq k \leq n, \end{cases} \tag{9}$$

allowing sequentially to calculate these moments for the homogeneous system of jobs in the terms of CL, for serial LD of sort $\begin{vmatrix} a_1 \\ \cdots \\ a_N \end{vmatrix}^{r}$ in (9) expressed in these terms.

6.

Characteristics of load of the parallel system are defined as follows. Any block k of the parallel system begins to operate at the moment $t = 0$, gets the job 1, then at the moment $t_1(k)$ it gets the job 2, ends at the moment $t_2(k)$ ..., at last, at the moment $t(k)$ it ends the last job. From this, the expressions of average and instant load of the k-th block are defined in the following way:

$$R_k = t(k)/T, \quad r_k(t) = \begin{cases} 1, & 0 \leq t \leq t(k), \\ 0, & t > t(k), \end{cases} \Bigg| \ k = \overline{1,m}, \tag{10}$$

and expressions of average and instant load of the system:

$$R=[\sum_{k=1}^{m} t(k)]/mT, \quad r(t)=\begin{cases}1, & t\le t^1,\\ 1-(i/m), & t^i \le t\le t^{i+1},\ i=\overline{1,m-1},\\ 0, & t>t^m.\end{cases} \tag{11}$$

where $\{t^i\}$ is a set $\{t(i)\}$, ordered by increase.

7.

A special class of parallel systems dependent on time by arrival of jobs, in which the arrivals of jobs are given by the order $P_n=(j_1,...,j_n)$ and the moment $\tau_1,...,\tau_n$ is obtained as follows. The calculations and analysis of such systems is based on a formula-analogue (9) (the case of the homogeneous system):

$$t_k=\begin{cases}\tau_k+\tilde{a}_k, & 1\le k\le m,\\ \left(\tau_k+\begin{vmatrix}t_1\\ \cdots\\ t_{k-1}\end{vmatrix}^{(k-m)}\right)+\tilde{a}_k, & m+1\le k\le n.\end{cases} \tag{12}$$

8.

Let us consider a parallel-sequential system from M consisting of joint steps with m_i parallel one-type blocks of equal speed at the i-th step. At an input of the system, the sequence $P_{1n}=(j_{11},...,j_{1n})$ arrive from n jobs $1,\dots,n$, passing through it by steps, where appropriate jobs are executed. The execution time of operation i for the job j is given by the matrix $A=\|a_{ij}\|$. The order of job execution inside each step is determined by laws of operation of parallel systems, and transition step by step (laws of operation of sequential systems). Thus, the order of jobs at an output of step does not coincide with the order at an input in general case. Let us designate a moment of the ending of the q-th job operation started at the k-th operation by t_{qk}. The characteristic of speed of the whole system is expressed by time T of execution of all jobs expressed by t_{qk} with the help of a disjunction of CL:

$$T=\bigvee_{k=1}^{n} t_{Mk}, \tag{13}$$

so the calculation of T is reduced to a calculation of a matrix $\|t_{qk}\|$. The latter is reduced to a recurrence, allowing to get these characteristics in the terms of CL:

$$t_{qk} = \begin{vmatrix} t_{q-1,1} \\ \cdots \\ t_{q-1,n} \end{vmatrix}^{(k)} + \tilde{a}_{qk}, \quad \text{while} \quad 1 \le k \le m_q;$$
$$t_{qk} = \left(\begin{vmatrix} t_{q-1,1} \\ \cdots \\ t_{q-1,n} \end{vmatrix}^{(k)} \vee \begin{vmatrix} t_{q1} \\ \cdots \\ t_{q,k-1} \end{vmatrix}^{(k-m_q)} \right) + \tilde{a}_{qk}, \text{ while } m_q + 1 \le k \le n. \quad (14)$$

In (14), the matrixes $\tilde{A} = \|\tilde{a}_{ij}\|$ are obtained from the matrix $A = \|a_{ij}\|$ of rearrangements of elements in each q-th line according to the order of job start at the q-th job (the order of ending at the $(q-1)$-th operation).

9.

Le us examine a consecutive-parallel system consisting of parallel joint branches as a sequence of blocks. In the k-th branch there are m_k blocks. Each k-th branch can execute any of n ($n \ge M$) jobs, submitted to the system, by splitting at m_k several sequential jobs executed in appropriate blocks of the branch. The speed of any k-th branch is given by a matrix $A(k) = \|a_{ij}(k)\|$, where $a_{ij}(k)$ is a time execution in the k-th branch of operation i for the job j. The order of job execution in each branch is defined by laws of operation of sequential systems by a sequence of jobs $P_n = (j_1,..., j_n)$ among branches (laws of job of parallel systems). The allocation of a sequence of jobs P_n among branches depends on the order of release of the first blocks of branches. These blocks derivate the parallel system with a matrix of times of jobs as follows:

$$A = \|a_{kj}\|, \;\; a_{kj} = a_{1j}(k), \;\; k = \overline{1,M}, \;\;\; j = \overline{1,n}. \quad (15)$$

Calculating this system allows us to find the order and moments of release of blocks and through them to find an allocation of an entry sequence of jobs P_n among branches and moments of arrival of these jobs in various branches. After that the calculation and analysis of the whole system is reduced to the same procedures with separate branches in a mode dependent on time of arrival of jobs. So, the time execution of all jobs in the k-th branch T_k, and the load R_k of the common execution time of all jobs in the system are defined thus:

$$T = \mathop{\vee}_{k=1}^{M} t_k, \quad R = \sum_{k=1}^{M} R_k / M. \quad (16)$$

10.

Let us return to the tasks of calculation and analysis of synthesis of the whole system. They consist in a choice of set of acceptable procedures of job execution of optimal procedure, when characteristics of the system have the best values. The task of synthesis of static system is simple. The parallel system with m blocks is intended for the execution of n jobs $(n \ge m)$. The time execution of job j in the block i is given by a matrix $A = \|a_{ij}\|$. The set of jobs W is executed by acceptable splitting into subsets $W_1,..., W_m$, executed in appropriate blocks. Total operating time of all blocks is:

$$D=\sum_{i=1}^{m}\sum_{j\in W_i}a_{ij} \ . \qquad (17)$$

It is required to select an optimal splitting of jobs into subsets W_i where $D=\min$. Two variants of this problem are possible: 1) without limitations by a cardinal number of subsets W_i; 2) with limitations of the sort: $b_i\le|W_i|\le c_i$. We see that

$$D_{\min}=A^{1\wedge} \text{ (in the first case); } D_{\min}=A^{2\wedge} \text{ (in the second case),} \qquad (18)$$

Here $A^{1\wedge}$ and $A^{2\wedge}$ are conjunctive LD of the 1-st and 2-nd sort with limitations by the sums of elements from a time matrix A. By the definition of $A^{1\wedge}$ there is a function of sort $\underset{q}{\wedge}\sum_q^{/}a_{ij}$, where $\wedge=\min$ is a conjunction of CL and $\sum_q^{/}a_{ij}$ is a sum of elements a_{ij}, including only one element from each column of the matrix A. Further, $A^{2\wedge}$ is a function $\underset{q}{\wedge}\sum_q^{//}a_{ij}$, where $\sum_q^{//}a_{ij}$ is a sum of elements a_{ij}, including only one element from each column of the matrix A and p_i elements from the i-th line, where $b_i\le p_t\le c_i$. Thus, to solve this problem it is necessary to calculate an appropriate LD. Hence, the value of LD specifies the value of $D_{\min}$ and the optimal allocation of jobs in blocks (the presence of element a_{ij} in an expression of LD means an attachment of the j-th job to the i-th block). The representation of (18) also shows that the analysis of the optimal static system is reduced to the analysis of behaviour of appropriate LD with changes of elements. In (18), the LD is a CL-function from their elements, the value $D_{\min}$ expresses time a_{ij} in the terms of CL.

According to the simplified formula of calculation of LD, we have $A^{1\wedge}=\sum_j\underset{k}{\wedge}a_{kj}$ for an arbitrary k-th block of all those jobs, which execution time in this block is a minimum compared with other blocks.

11.

The problem of synthesis of the sequential system with m blocks which execute n jobs consists in searching the optimal order $P_{opt}=(j_1,...,j_n)$ of job execution with the execution time of all jobs $T=\min$. The solution of this task is most simple in the case of $m=2$. Any two jobs i,j in P_{opt} following the order $i\to j$, the execution conditions satisfies:

$$a_{1i}\wedge a_{2j}\le a_{1j}\wedge a_{2i}\ , \qquad (19)$$

where $\wedge=\min$ is a conjunction of CL. The condition of (19) enables to design simple deciding rules for finding the optimal order of jobs P_{opt} without searches. It is interesting that in the case of $m=2$ the solution of the problem of synthesis is searched in the class of the permutation schedules, i.e. sequences of jobs. It is connected to the following fact: P_{opt} with $m=2$ lies in the class of the permutation schedules. Thus, any two jobs i,j in P_{opt} following the order $i\to j$, the execution conditions are:

$$a_{1i}\wedge a_{2j}\le a_{1j}\wedge a_{2i},\quad a_{2i}\wedge a_{3j}\le a_{2j}\wedge a_{3i},$$
$$(a_{1i}+a_{2i})\wedge(a_{1i}+a_{3j})\wedge(a_{2j}+a_{3j})\le(a_{1j}+a_{2j})\wedge(a_{1j}+a_{3i})\wedge(a_{2i}+a_{3i}). \tag{20}$$

The search of P_{opt} in the system with $m=3$ blocks is carried out in the way: 1) the construction of graph of priorities of jobs linking by an arc $i\to j$, which represents jobs i,j satisfying the condition of (20); 2) finding in the graph any hamiltonian path, which gives P_{opt}. The given algorithm of searching the order P_{opt} in the system with $m=3$ blocks is more complex than solution rules for systems with $m=2$ blocks.

12.

The sequence of jobs $P=(i_1,...,i_k,i,j,...,i_n)$ is strongly (poorly) separable, if the rearrangement of any pair of jobs i,j increases (decreases) a time moment of the ending of subsequence of jobs $(i_1,...,i_k,i,j)$ in all the blocks q (even in one block q), $q=\overline{2,m}$. Such a limitation practically allows to formulate in the terms of CL and LD some general analytical conditions of optimality about jobs in systems with any number m of blocks.

1) For the sequence $P=(i_1,...,i_k,i,j,...,i_n)$ of passing of n jobs through m blocks it is optimal (and strongly separable) that at any ordered pair (i,j) of adjacent jobs from P the time of execution of jobs satisfies the following conditions:

$$\left.\begin{array}{ll} A^{\vee}_{s,s+1}(i,j)\le A^{\vee}_{s,s+1}(j,i), & s=\overline{1,m-1};\\ A^{\vee}_{s,s+2}(i,j)\le A^{\vee}_{s,s+2}(j,i), & s=\overline{1,m-2};\\ \dots\dots\dots\dots\dots\dots\dots\dots & \\ A^{\vee}_{1m}(i,j)\le A^{\vee}_{1m}(j,i), & \end{array}\right\}, \tag{21}$$

which appear in a special disjunctive logical determinant up to the i-th and j-th columns of the matrix $A=\|a_{ij}\|$ of job time.

$$A^{\vee}_{sr}(i,j)=\begin{vmatrix} a_{si} & a_{sj}\\ \cdots & \cdots\\ a_{ri} & a_{rj}\end{vmatrix}^{\vee},\quad 1\le s\le r\le m\ ; \tag{22}$$

2) For the sequence P it is optimal (poorly separable) that at any pair of adjacent jobs (i,j) the time of jobs satisfies the conditions with the same LD:

$$[A^{\vee}_{12}(i,j)\le A^{\vee}_{12}(j,i)]\cup[A^{\vee}_{s3}(i,j)\le A^{\vee}_{s3}(j,i),s=\overline{1,2}]\cup\ldots\cup[A^{\vee}_{sm}(i,j)\le A^{\vee}_{sm}(j,i),s=\overline{1,m-1}] \tag{23}$$

Since LD is a CL-function, the conditions of (21), (23) express jobs in the terms of CL. With $m=2$ the conditions of (21) and (23) coincide to give (19), therefore, we have the necessary and sufficient condition of an optimality of systems with two blocks. With $m=3$ the conditions of (21) is reduced to (20). With $m\ge 3$ the condition of (21) is harder than the conditions of (23). The search in systems with $m\ge 4$ blocks with the help of sufficient conditions of an optimality (21) is carried out as well as in the case of $m=3$, with using the graph of priorities of jobs. The search in the systems with $m\ge 3$ blocks with the help of necessary conditions of an optimality (23) is carried out in the following manner: 1) the creation of graph of priorities of jobs linking by an arc $i\to j$, which represent jobs satisfying the condition of (23); 2) searching in the graph all hamiltonian paths giving a sequences P, suspicious on an optimality; 3) in P_{opt} it is selected P, for which $T = \min$.

13.

For sequential systems with large numbers of blocks m and jobs n, and also for systems of other construction (parallel and more complex), analytical conditions of an optimality do not work or in general are absent. Therefore, synthesis of such systems is usually carried out by a branch and bound algorythm. The efficiency of this method essentially depends on the force of used estimations of time T of execution of all n operations, common for all possible sequences of jobs of sort $P^r = (R^r, Q^r)$, where R^r is a fixed sequence of r first operations, and Q^r is a set of all possible sequences of others $n-r$ jobs. The application of CL and LD allows us to receive strong estimations of characteristics of systems. It is clear that the value of characteristic for an initial site R^r of sequence P^r, as shown above, is expressed precisely by operations of CL and LD, and estimation of characteristic for the rest site Q^r turns out by a choice of the best (worst) case, that uses operations of continuous logic $\vee = \max$ and $\wedge = \min$. So, for the parallel system with m blocks and the matrix of job time $A = \|a_{ij}\|$ the lower bound of a characteristic T is defined thus:

$$T(P^r) \geq \frac{\sum\limits_{j \in \tilde{Q}_r} \bigwedge\limits_{i=1}^{m} a_{ij}}{m} + \begin{vmatrix} t_1(R^r) \\ \cdots \\ t_r(R^r) \end{vmatrix}^{(r-m)+1} . \qquad (24)$$

Here $\tilde{Q}_r$ is the set of all jobs from Q_r, $t_k(R^r)$ is the moment of the ending of the k-th job in the order of jobs from R^r, calculated according to section 5, $|...|_s$ is a serial LD-column of a rank s.

14.

Some economic models of an industrial type have been considered in [1] – [5]. CL-models of several concrete classes of other systems are studied in [6] – [8]. The generalizing consideration is undertaken in [9].

References

1. Pervozvansky, A.A. *Mathematical Models in Production Management*. Nauka: Moscow, 1975 [in Russian].
2. Granberg, A.G. *Simulation of Socialist Economy*. Economics: Moscow, 1988 [in Russian].
3. Wagner, H.M. *Principles of Operations Research*. Prentice-Hall: Englewood Cliffs, New Jersey, 1969.
4. Kaufmann, A. *Introduction a la combinatorique en vue des application*. Dunod, Paris, 1968.
5. Kaufmann, A. and Henry-Laborder A. *Methodes et modeles de la recherche operationelle*. Tome 3. Dunod, Paris, 1974.
6. Bulanov, A.F. *Logical Models of Solution of the Extremal Tasks*. Dissertation. Riga, 1988 [in Russian].
7. Lysak, S.A. *Logical Methods and Algorithms of Optimal Ordering of Jobs*. Dissertation. Leningrad, 1989 [in Russian].
8. Miretzky, I.Y. *Construction of the Optimal Schedules of Execution of Jobs for Technical Systems*. Dissertation. Moscow, 1993 [in Russian].

9. Levin, V.I. *Structural-Logical Methods of Research of Complex Systems*. Nauka, Moscow, 1987 [in Russian].