Robustness of Information Systems and Technologies

MARK BURGIN
Department of Mathematics
University of California, Los Angeles
Los Angeles, CA 90095
USA
mburgin@math.ucla.edu

Abstract: - Robustness of technological systems and processes is studied using tools of the mathematical theory of information technology. Results obtained in the paper allow one to analyze robustness of compound information technologies based on stability and robustness of their components. The aim is the development of efficient methods and algorithms of the computer aided design of information technologies.

Key-Words: - Information technology, mathematical model, technological operator, robustness, regular input, acceptable input

1 Introduction

When using a computer or software system, we obviously hope that it won't crash being sufficiently robust. That is why robustness as an important information and communication property has been studied for a variety of systems, technologies, and processes (cf., for example, [1,7,8]). Technological approach brings unification to these studies. For instance, when we evaluate properties of information hardware or software, we actually do this by evaluating corresponding properties of technologies based on this system and performance of this system in definite technologies.

The mathematical theory of technology developed in [2–5] allows one to study robustness as an intrinsic property of information systems and processes. The term robust used with regard to computer software or hardware refers to a software or hardware system, for example, the operating system, that performs well not only under ordinary conditions but also under unusual conditions that stress its designers' assumptions. Robustness is considered as a degree to which system can function correctly in the presence of inputs different from those assumed or it guarantees the maintenance of desired system characteristics despite fluctuations in the behavior of the system components or its environment [1].

An extended definition of robustness includes several components. For instance, robustness of an interactive system consists of:
1. Responsiveness, which is measured by the response time and modeled by temporal robustness.
2. Observability is a measure for how well internal states of a system can be inferred by knowledge of its external behavior, e.g., by its output.
3. Recoverability reflects the ability to restore the process (system) when a failure occurred.
4. Task conformance.

The latter also has four components:
(a) data task conformance, which is measured by the admissible variability of data and modeled by data robustness studied in this paper;
(b) procedural task conformance, which is measured by the admissible variability of the used algorithms, programs, and scenarios, being modeled by procedural robustness;
(c) technical task conformance, which is measured by the admissible variability of data and modeled by technical robustness;
(d) performer task conformance, which is measured by the admissible variability of data and modeled by performer robustness.

To build a mathematically rigorous model, we use specific mathematical notation. In particular, the symbol \( \forall \) means "for any" or "for all", while the symbol \( \exists \) means "there is" or "there exists".

2 Technological Operators

The basic concept of the mathematical theory of technology is a technological operator. Any technology is represented by some technological operator. Technological operators are blocks for building concrete technologies.

Definition 2.1. An elementary technological operator \( EO \) (or an elementary technology) is a triad \( (Cb, Op, Cf) \) where:
Cb denotes a description of the initial technological conditions or input specification of EO, that is, such conditions that must be satisfied for the beginning of EO execution;

Cf denotes a description of the finalizing technological conditions or output specification of EO, that is, such conditions that must be satisfied when EO gives its result;

Op denotes the operation realized by EO.

Technological conditions have a complex structure and their description includes three components:
- descriptions DAA of all active agents (performers and operational devices) AA of the technological process;
- descriptions DPA of all passive agents (objects of activity) PA of the technological process;
- descriptions DCC of the context conditions CC.

Active agents have of two types:
1. Performers or executors (producers, managers, and controllers).
2. Operational devices (completely controlled, partially controlled, and autonomous).

Passive agents consist of three types: raw materials (input data), work materials, products (output data).

Context conditions have different types: temporal conditions (such as requirements related to physical time, to system time, etc.), spatial conditions (such as requirements related to physical space, information space, state space, etc.), and combined conditions (such as velocity, acceleration, productivity, efficiency, power, etc.).

State space may include such parameters as temperature, light conditions, pressure, etc.

**Definition 2.2.** A complete technological operator CO is a triad of the following form:

\[
CO = (\text{Tpr}, \text{Tm}, \text{Tpt})
\]

Here:

Tpr is an operator that describes execution of preparatory operations;
Tm is an operator that describes execution of the main operation;
Tpt is an operator that describes execution of concluding operations.

This structure corresponds to the structure a transformation system operation: input, transformation, and output. As an example take an information processing system, concrete like computer or theoretical like abstract automaton. It consists of processor and input and output devices.

**Definition 2.3.** A composite technological operator A is a technological operator that consists (is built) of other technological operators.

In particular, complete technological operator is a kind of composite technological operators.

There are different operations for combining technological operators. One of the most important is the sequential composition or product of technological operators. Namely, if A and B are technological operators, then their product AB is an operator obtained by performing at first B and then applying A to the output of B. A composition of technological operators is called technological if it is determined by a technological operator.

In what follows, we primarily consider information technologies. The main input, output and working material of information technologies is information in the form of data and knowledge.

**Definition 2.4.** A technology is a technological composition of complete technological operators.

Thus, a general form of the mathematical model of technology is a technological operator that has the following structure:

\[
A = \{ A_1, A_2, \ldots, A_n \}
\]

Here \( A_c \) is the composition operator of the technology T (of the operator A) for operators \( A_1, \ldots, A_n \).

Computer programs can illustrate this construction. As it is demonstrated in [2], any technology is, in particular, an algorithm and a program. On the other hand, any program may be considered as an incomplete technology. Incompleteness means that program in itself does have for example directions what type of computer it needs for its execution, the volume of necessary memory etc. Thus, the structure of a program reflects the general structure of a technology. Really, a program P can be perceived as a collection of modules (operators \( P_1, \ldots, P_n \)) with a small, algorithmic core (operator Pє) that expresses how the modules are used to obtain a desired effect. In a sense, the core is expressed in an application specific language, with control constructs implemented by the modules.

To systematize technological compositions, we separate specific types with respect to the processes they determine.

A. A classification of technological compositions by organization of composition:

1. **External** technological composition is performed by special composition operators.
2. **Internal** technological composition is performed by the composed operators themselves. In this case, composition
operators only provide channels (media) for interaction, which make possible interaction of composed operators.

3. **Mixed** technological composition is partially performed by special composition operators and partially performed by the composed operators themselves.

**B. A classification of technological compositions by performance of operators:**
1. **Parallel** technological composition organizes parallel performance of the composed operators.
2. **Sequential** technological composition organizes performance of the composed operators one after another with the output conditions of the previous operator taken as the input conditions by consequent operator.
3. **Concurrent** technological composition organizes individual performance of the composed operators with possible interactions.

3. **Technological Robustness**

Important properties of any technology are robustness and stability. Usually, a technology (represented here by a technological operator) has some initial conditions (in particular, some definite input) for which this technology is designed and under which presumably it correctly works. However, in some cases, this technology (technological operator) works with objects that do not belong to its admissible input domain. This is considered technologically incorrect and may result in damage of equipment, inappropriate output, and/or injury of the staff. Informally, robustness means that if the initial conditions (input), for example, raw material, does not differ too much from the admissible input, then functioning of the technology (its output) does not differ too much from the functioning (output) with correct input. For instance, one of the main principles of software design is roughness that prescribes to build a software system in such a way that small variations in the environment or shock of unpredicted events do not cause systems failure [11]. It is also possible to say that robustness is the degree, measure or extent to which a system or its component can function correctly in the presence of invalid input data, existence of faults in its parts and/or stressful environment conditions [9].

Robustness of a system (technology) has three areas:

1. **Inner robustness** reflects changes in the system (technology).
2. **External robustness** reflects changes in the system (technology) environment.
3. **Intermediate robustness** reflects changes in the system (technology) connections to the environment.

For instance, an interactive program $P$, e.g., word processor, is designed to allow the user control $P$ with the mouse and keyboard. Intermediate robustness reflects how $P$ functions only with the mouse used, i.e., the keyboard does not function, or when only the keyboard is used, i.e., the mouse does not function.

To formalize and study such a property of technology as robustness, we need some mathematical concepts. In what follows, $R$ is the set of all real numbers and $R^+$ is the set of all non-negative real numbers.

**Definition 3.1.** A *generalized distance* on a set $X$ is a function that corresponds to each pair of elements from $X$ some element from a fixed partially ordered set $L$.

Usually the set of all non-negative $R^+$ is taken as $L$. Another important particular case is when $L$ is a vector space. This allows one, for example, to deal with robustness degree of a system as a vector that consists of robustness degrees of different system characteristics.

Examples of generalized distances are metrics and quasimetrics [10].

An important case of generalized distances is *characteristic distance* in $X$. Let us consider a characteristic (property) $c$ of elements in $X$. It is represented by a mapping $c: X \rightarrow R$. Then the numerical characteristic distance $d_c$ between two elements $x$ and $y$ from $X$ is defined as

$$d_c(x, y) = | c(x) - c(y) |$$

Several system characteristics induce a mapping $c: X \rightarrow R^+$. In this case, we have a vector characteristic distance $d_c$ between two elements $x$ and $y$ from $X$ defined as

$$d_c(x, y) = \| c(x) - c(y) \|$$

In what follows, any space is always a set with a generalized distance, e.g., a metric space. For instance, if we have a set $Q$ of some products or raw materials, generalized distance gives a measure (estimate) of differences between different elements from $Q$.

Each information technology is designed to work with regular input data where regularity is defined by corresponding conditions. In addition, it is specified what is regular output data of a certain information technology. Thus, when a technology $T$
is represented by a technological operator \(O_{p_f}\), we define the domain \(I_f\) from which regular input data for \(T\) are taken and the domain \(O_f\) which contain all regular output of \(T\). The domain of all possible input data for \(T\) is denoted by \(D_f\) and called the domain of the information technology \(T\). The domain of all possible outputs for \(T\) is denoted by \(C_f\) and called the codomain of the information technology \(T\). Generalized distances in the domains \(D_f\) and \(C_f\) usually are determined by properties/parameters of input and output data for a given information technology \(T\).

This discriminates two kinds of information technologies (technological operators): regular and stochastic.

**Definition 3.2.** A technology (technological operator) \(T\) is called regular if given regular input data, it gives regular output data.

**Definition 3.3** A technology (technological operator) \(T\) is called stochastic if given regular input data, it gives regular output data only with some probability \(p\).

The domain \(I_f\) is a subdomain of the domain \(D_f\) of the technology (technological operator) \(T\). The domain \(O_f\) is a subdomain of the domain \(C_f\) of the technology (technological operator) \(T\).

Let us take a technological operator (TO) \(A\). We suppose that the both domains \(D_A\) and \(C_A\) are spaces in which generalized distances \(d_i\) and \(d_o\) are defined, and \(I_0 \subseteq D_A\) is the admissible input domain. For instance, treating the perturbation parameter \(\pi\) in the sense of [1] as input and \(p\) as output of the system functioning, we can take the distance in the set of perturbation parameters as \(d_i\) and the characteristic distance \(d_o\) as \(d_o\).

**Definition 3.4.** a) The admissible input domain \(I_{0f} \subseteq D_A\) of a technological operator \(A\) consists of tentative input data such that given input data from \(I_{0f}\), operator \(A\) is normally functioning according to its design or utilization instructions.

b) The acceptable input domain \(I_{af} \subseteq D_A\) of a technological operator \(A\) consists of tentative input data such that given input data from \(I_{af}\), operator \(A\) performs some action.

c) The tolerable input domain \(I_{tf} \subseteq D_A\) of a technological operator \(A\) consists of tentative input data such that given input data from \(I_{tf}\), operator \(A\) performs some tolerable action, i.e., gives a tolerable output.

There is different understanding what normal functioning and what is tolerable functioning of a regular technological operator \(A\) is. Here we give three interpretations:

1) normal functioning of a regular technological operator \(A\) implies that it gives regular output;

2) normal functioning of a regular technological operator \(A\) implies that it gives acceptable output;

3) normal functioning of a regular technological operator \(A\) implies that it gives a tolerable output and does not harm other processes and systems [6].

Let us consider such an output characteristics as response time. When typing a letter on a computer keyboard, the user will usually expect these letters to appear immediately on the screen. This demands less than 0.1 seconds response time, which is a regular output characteristic. However, for other more indirect commands of the user the response time can be longer. For instance, when clicking a button in visual interface, the system should give an immediate (less than 0.1 second response time) visual feedback that the system has received that command. Nevertheless, the actual result of the intended operation, such as closing a dialog or deleting a file, the response time of around 0.5 seconds is perceived as acceptable.

In navigation in a web browser, the browser must give an immediate feedback that it has received the command. The regular response time for loading and rendering the new webpage is 0.5 second. However, acceptable response time for loading and rendering the new webpage can be 1 second.

**Theorem 3.1.** a) If normal functioning of a regular technological operator \(A\) implies regular output, then \(I_{0d} \subseteq I_{d} \subseteq D_A\).

b) If producing regular output implies normal functioning of a regular technological operator \(A\), then \(I_d \subseteq I_{0d} \subseteq D_A\).

Often regular input data are also admissible input data. However, it is possible that a technological operator (a system) gives regular output while harming some other systems or processes. This situation cannot be considered as normal functioning and thus, admissible input data will be more restricted than regular input data.

Taking computer robustness, we consider other kinds of input and output domains. For instance, now there are several industry standard voltages in use in processors. In the, so-called, STD (standard) demands regular (nominal) voltage 3.300 V, while acceptable voltage is from 3.135 V to 3.465 V. For the VRE (Voltage Reduced Extended), regular (nominal) voltage is 3.520 V (or 3.500 V), while acceptable voltage is from 3.450 V to 3.600 V (or from 3.400 V to 3.600 V).

Here we formalize and study such a property of information technology as data robustness. Let \(A\) be a technological operator (TO) and \(a, b \in R\).
Definition 3.5. A technological operator $A$ is called:

1) regularly $(a, b)$-robust in a domain $D_0 \subseteq D_4$ if for any $x \in D_0 (d(x, I_x) < a$ implies $d_0(A(x), O_A) < b)$.
2) regularly $a$-robust in a domain $D_0 \subseteq D_4$ if for any $x \in D_0 (d(x, I_x) < a$ implies $A(x) \in O_A$).

Informally, regular $(a, c)$-robustness means that if input data are not far from the regular input domain $I_x$, then the output data are not far from the regular output domain $O_A$. Regular $a$-robustness means that if input data are not far from the regular input domain $I_x$, then the output data still belong to the regular output domain $O_A$.

Lemma 3.1. If $a \geq d$ and $b \leq c$, then any regularly $(a, b)$-robust in $D_0$ technological operator $A$ is regularly $(d, c)$-robust in $D_0$.

Corollary 3.1. If $a \geq d$, then any regularly $a$-robust in $D_0$ technological operator $A$ is regularly $d$-robust in $D_0$.

Definition 3.6. The regular tolerance distance $rtd_A$ of a technological operator $A$ is defined as

$$d_A = \sup \{ a; A \text{ is regularly } d \text{-robust in } D_4 \}$$

Proposition 3.1. If $A$ is a regularly $a$-robust technological operator in $I_x$ and $B$ is a technological operator, then their sequential composition $AB$ is a regularly $a$-robust technological operator.

Corollary 3.2. $d_{AB} = d_A$.

Lemma 3.2. If the generalized distance $d_0$ is a quasimetric, then any regularly $a$-robust in $D_0$ technological operator $A$ is regularly $(a, c)$-robust in $D_0$ for any $c \geq 0$.

Lemma 3.3. If the generalized distance $d_0$ is a metric, then $A$ is a regularly $a$-robust in $D_0$ technological operator if and only if $A$ is regularly $(a, 0)$-robust in $D_0$.

Theorem 3.2. If $A$ is a regularly $(a, b)$-robust technological operator in $D_0 \subseteq I_x$ and $B$ is a regularly $(b, c)$-robust technological operator in $A(D_0) \subseteq D_0$, then their sequential composition $AB$ is a regularly $(a, c)$-robust technological operator.

Corollary 3.3. If $A$ is a regularly $(a, b)$-robust technological operator in $D_0 \subseteq I_x$, $B$ is a regularly $(d, c)$-robust technological operator in $A(D_0) \subseteq D_8$, and $b \leq d$, then their sequential composition $AB$ is a regularly $(a, c)$-robust technological operator.

Theorem 3.3. If a technological operator $A$ is regularly $(a, b)$-robust in $D_0 \subseteq D_4$ and regularly $(d, c)$-robust in $D_1 \subseteq D_4$, then $A$ is regularly $(u, v)$-robust in $D_0 \cup D_1$ where $u = \min \{a, d\}$ and $v = \max \{b, c\}$.

Let us take a set $C \subseteq R^* \times R^*$ of pairs of non-negative real numbers.

Definition 3.7. A technological operator $A$ is called regularly $C$-robust if $A$ is regularly $(a, b)$-robust for any pair $(a, c)$ from $C$.

In this case, $C$ is called a robustness set of $A$.

Proposition 3.2. If $H \subseteq C$ and a technological operator $A$ is regularly $C$-robust, then $A$ is regularly $H$-robust.

Proposition 3.3. If a technological operator $A$ is regularly $C$-robust and regularly $G$-robust, then $A$ is regularly $C \cup G$-robust.

Definition 3.8. A set ideal is a collection of sets closed with respect to subsets and unions.

Corollary 3.4. Robustness sets of a technological operator form a set ideal.

Definition 3.9. a) The left closure $L$ of a robustness set $C$ of a technological operator $A$ is equal to

$$LC = \{(c, b); 0 \leq c \leq a \text{ and } (a, b) \in C \}$$

b) The right closure $R$ of a robustness set $C$ of a technological operator $A$ is equal to

$$RC = \{(a, d); 0 \leq b \leq d \text{ and } (a, b) \in C \}$$

Lemma 3.4. a) $LLC = LC$ for any robustness set $C$.

b) $RRC = RC$ for any robustness set $C$.

c) $RLC = LRC \subseteq LC$, $RC$ for any robustness set $C$.

Theorem 3.4. A technological operator $A$ is regularly $C$-robust if and only if $A$ is regularly $RLC$-robust.

Definition 3.10. A technological operator $A$ is called:

1) regularly $(c, d)$-non-robust in $D_0 \subseteq D_4$ if

$$\exists x \in D_0 (d(x, I_x) < c \text{ and } d_0(A(x), O_A) > d)$$

2) regularly $c$-non-robust in $D_0 \subseteq D_4$ if

$$\exists x \in D_0 (d(x, I_x) < c \text{ and } A(x) \notin O_A)$$

Lemma 3.5. If $a \geq d$ and $b \leq c$, then any regularly $(d, c)$-non-robust in $D_0$ technological operator $A$ is regularly $(a, b)$-robust in $D_0$.

Corollary 3.5. If $a \geq d$, then any regularly $d$-non-robust in $D_0$ technological operator $A$ is regularly $a$-non-robust in $D_0$.

Lemma 3.6. If the generalized distance $d_0$ is a quasimetric, then for any $c \geq 0$, any regularly $(a, c)$-non-robust in $D_0$ technological operator $A$ is regularly $a$-non-robust in $D_0$.

Lemma 3.7. If the generalized distance $d_0$ is a metric, then $A$ is a regularly $a$-non-robust in $D_0$
technological operator if and only if $A$ is regularly $(a, 0)$-non-robust in $D_0$.

Let us take a set $D \subseteq \mathbb{R}^2 \times \mathbb{R}^2$.

**Definition 3.11.** A technological operator $A$ is called regularly $D$-non-robust if $A$ is regularly $(a, b)$-non-robust for any pair $(a, c)$ from $D$.

In this case, $D$ is called a nonrobustness set of $A$.

**Proposition 3.4.** If $H \subseteq D$ and a technological operator $A$ is regularly $H$-non-robust, then $A$ is regularly $D$-non-robust.

It is necessary to remark that technological robustness can represent system robustness. Indeed, let us consider a system $R$ with a system feature $p$ and take a perturbation parameter $\pi$ in the sense of [1]. Then it is possible to treat $\pi$ as input and $p$ as output of the system functioning. In this case, regular $(a, c)$-robustness shows how changes in $\pi$ impact the feature $p$, while regular $a$-robustness reflects admissible changes of the perturbation parameter $\pi$, which do not move $p$ from its regular domain.

## 4 Conclusion

We have used a mathematical model of information technology based on the concept of a technological operator, to formalize such an important property of information systems and technologies as robustness. We study robustness of technological operators with respect to their input data, which forms the data conformance component of robustness. Results obtained in the paper allow one to analyze robustness of compound technological operators through robustness of their components.

It would be also interesting to study the technical conformance component of robustness, i.e., robustness with respect to the system that is used by technological operators. That is, it might be useful to find, for example, how much it is possible to change tools and/or machines to get the same or even better quality product or in what range we can vary qualification of performers, e.g., programmers or software engineers, to get the same or even better quality results.

Besides, there are many other important properties of technologies, such as efficiency, interoperability, reliability, security, usability, maintainability, etc. The mathematical theory of technology provides means for formalization, study and evaluation of these properties.