# An Entropy-Based Uncertainty Measure of Process Models

Jae-Yoon Jung[a], Chang-Ho Chin[a], Jorge Cardoso[b]

[a]Department of Industrial and Management Systems Engineering, Kyung Hee University, Republic of Korea

[b]CISUC/Department of Informatics Engineering, University of Coimbra, Portugal

**ABSTRACT**

In managing business processes, the process uncertainty and variability are significant factors causing difficulties in prediction and decision making, which evokes and augments the importance and need of process measures for systematic analysis. We propose an entropy-based process measure to quantify the uncertainty of business process models. The proposed measure enables capturing the dynamic behavior of processes, in contrast to previous work which focused on providing measures for the static aspect of process models.

**Keywords**: concurrency, real-time system, entropy, process uncertainty, process measure

## 1. Introduction

A business process is a collection of tasks and decisions to produce products or services in an organization. The measurement of business processes has a great significance since a process does not only create value, but also costs. In recent years, many measures of process models have been devised to reflect or predict process characteristics such as understandability, reliability, usability, and maintainability [1]. A process model designed with the aid of these metrics as guiding principles is likely to be less error-prone, easier to understand, maintain, and manage, and more efficient [2].

Several process measures for complexity [3,4], density [5], coupling and cohesion [6] have been recently introduced to provide a quantitative basis for the design, development, and analysis of process models. In particular, most studies have addressed complexity measures. Higher complexity leads to more difficulty in understanding and interpreting process models. Mendling *et al.* [7] analyzed hundreds of SAP reference models to confirm that complexity seemed to be a key determinant for errors. Gonzalez *et al.* [1] and Muketha *et al.* [8] provide very good surveys of recent research done in this area.

*Published in Information Processing Letters, Vol. 111, No. 3, Jan 2011, pp. 135-141.*

In this paper, we focus on the uncertainty and variability of business processes. Uncertainty and variability are caused by events that force a system to deviate from a regular and predictable behavior [9]. In operational processes, reducing variability or uncertainty has been an important issue, since doing so enables the process to guarantee better predictability and managerial efficiency. On the contrary, systems with high variability and uncertainty have more difficulties in making more efficient planning and scheduling. In mathematical statistics, entropy is often used to measure uncertainty about the value of a random variable [10]. In a similar way, the concept of entropy can be applied to measure the uncertainty of execution scenarios in a process. See Section 2 for details.

In contrast to previous research emphasizing the static aspect of process models, in this paper, we propose an entropy-based measure which captures the dynamic behavior of processes. It enables experts to better understand the nature of processes at runtime. The proposed entropy-based process measure quantifies the uncertainty of executing business process models and the process uncertainty is defined in terms of the transition and execution of tasks. We provide explicit forms of measure for primitive control-flow patterns and illustrate it by a process model fractionated into separate blocks of primitive patterns.

## 2. Entropy-based uncertainty measure

A business process is a set of logically related tasks performed to achieve a defined business outcome [11]. The process has uncertainty because some parts of it are conditionally executed at runtime based on managers' decisions or process data. While it is not possible to know for sure if a particular task will be executed, it is possible to associate a probabilistic model to conditional tasks by analyzing the past behavior of processes' executions. In our research, we use the entropy measure to model the uncertainty associated with the execution (or not execution) of conditional tasks. The uncertainty becomes higher as the distribution over execution scenarios is more uniform. The larger number of scenarios leads to the higher uncertainty, when all scenarios have the same probability of occurring. These properties of process uncertainty can be reflected by the concept of entropy.

The uncertainty of information is generally calculated by Shannon's entropy [12]:

$$H(X) = \sum_{i=1}^{n} P(x_i)u(x_i) = -K\sum_{i=1}^{n} P(x_i)\log_2 P(x_i)$$

where $X$ is a discrete random variable taking possible states $x_1, x_2, ..., x_n$ with probabilities $P(x_1), P(x_2), ..., P(x_n)$, respectively (for $1 \leq i \leq n$, $P(x_i) \geq 0$, $\sum P(x_i)=1$). The entropy $H(X)$ is the expectation of $u(x_i)$ which is $x_i$'s uncertainty. Shannon interpreted the amount of information by the entropy measure when $K$=1 (the constant $K$ is merely the choice of measurement unit). Shannon's entropy is a simple quantitative measure of uncertainty in a data set. The entropy increases, as the state distribution is more uniform.

The intent of a process model is achieved through completing a set of process tasks in one of all possible execution scenarios. In this paper, we focus on the uncertainty of which scenario is executed to accomplish the intent of process model at each time. We define the execution scenario (ES) probability of a scenario as the probability that the specific one out of all possible scenarios is executed. The ES probability is obtained by repetitively multiplying the transition probabilities between two sequential tasks in the scenario. The process uncertainty increases, as the probability distribution over scenarios is more uniform. When all ES probabilities are same, the larger number of scenarios results in the higher uncertainty. This relationship between the process uncertainty and its transition probabilities can be formulated with Shannon's entropy and it can be viewed as a measure of the uncertainty of process model.

For illustrating how the concept of entropy can be applied to calculating the uncertainty of process models, consider a simple process model ***B*** consisting of task $t_0$ proceeding to one of tasks $t_1$, $t_2$, and $t_3$ in an XOR-split, where there exist three possible execution scenarios: $t_0 \rightarrow t_1$, $t_0 \rightarrow t_2$, and $t_0 \rightarrow t_3$. The corresponding ES probabilities are $P(t_1 \rightarrow t_2)$= 1/5, $P(t_1 \rightarrow t_3)$ =3/10, and $P(t_1 \rightarrow t_4)$=1/2, respectively. In this process model ***B***, the uncertainty of which scenario is executed is calculated using Shannon's entropy as follows:

$$U(\boldsymbol{B}) = -\sum_{i=1}^{3} P(t_0 \rightarrow t_i) \log_2 P(t_0 \rightarrow t_i) = -\left(\frac{1}{5}\log_2\frac{1}{5} + \frac{3}{10}\log_2\frac{3}{10} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1.48,$$

where $U(\boldsymbol{B})$ denotes the uncertainty of process model ***B***. The minimum value (=0) for the measure is attained with only one ES probability equal to 1 and the others equal to zero (e.g., $P(t_1 \rightarrow t_2)$=1, $P(t_1 \rightarrow t_3)$=0, $P(t_1 \rightarrow t_4)$=0), which means that the process model is always executed in the specific scenario of task $t_0$ proceeding to task $t_1$ with no uncertainty. The maximum value is attained with all equal ES probabilities (i.e., $P(t_1 \rightarrow t_2)$ =$P(t_1 \rightarrow t_3)$ =$P(t_1 \rightarrow t_4)$ =1/3), which maximize the uncertainty of which scenario is executed. ES probabilities can be estimated from the historical data of the process model and be updated with every new observation.

A process with high entropy indicates that it is more difficult to correctly predict which conditional tasks will be called for execution. On the other hand, a process with low entropy indicates that some conditional tasks have a higher probability of being executed than others. As a result, the uncertainty of which tasks will be called for execution is lower. If process model ***A*** has higher entropy than process model ***B***, then it is possible to predict with higher confidence and accuracy which conditional tasks will be executed in process model ***B***.

## 3. Explicit forms of measure for process models

This section provides the explicit forms of the uncertainty measure for five primitive control-flow patterns such as sequence, AND-split, OR-split, XOR-split, and loop. To describe the accurate behaviors of process models, Petri nets are used to represent them as shown in Fig. 1. A process model is a directed bipartite graph with two node types, tasks and places, which are depicted by rectangles and circles, respectively [6]. In the model, the state of a process is represented by the distribution of tokens, depicted by black dots. In a process model, a task is *ready* if each of its input places contains at least one token. A ready task can start its execution, and when the task *starts*, it consumes a token from each input place and produces a token for each output place. In this paper, we define a transition between two sequential tasks as the accomplished movement of tokens required to make the task that follows ready. The transition probability is obtained by simply multiplying all the probabilities of related token movements. If a place has only one output task, the transition probability to the following task is 1. However, if the place has more than one output task, the transition probabilities to the following tasks can vary between 0 and 1 according to the control-flow patterns of the process.

Each process model under analysis can be simply expressed with only one control-flow pattern containing multiple process blocks at a certain level of resolution as shown in Fig. 1(a) and, at the next finer level of resolution, the process block may reduce to only one individual task or expand to another group of process sub-blocks in one or more control-flow patterns as shown in Fig. 1(b). We assume that all process blocks are independently executed.

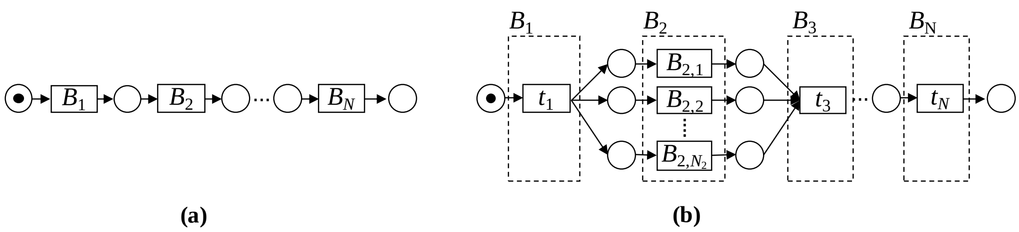


Fig. 1. A process model in the sequence pattern: (a) expressed with unexpanded process blocks and (b) expressed with expanded process blocks. $t_1$ and $t_N$ are individual tasks.

The notations used for the explicit forms of uncertainty measure for process models are as follows:

- $\boldsymbol{B}$ : process model under analysis, as a whole
- $N$ : total number of process blocks at a certain level of resolution where process model $\boldsymbol{B}$ can be expressed

as only one control-flow pattern containing multiple process blocks, as shown in Fig. 1(a)

- $M$ : total number of possible execution scenarios for the $N$ process blocks in process model $\boldsymbol{B}$
- $BS_k$ and $P(BS_k)$: $k^{th}$ execution scenario for the $N$ process blocks in process model $\boldsymbol{B}$ and its ES probability
- $B_g$ : $g^{th}$ process block in process model $\boldsymbol{B}$, $1 \leq g \leq N$
- $P(B_g)$ : probability that the $g^{th}$ process block $B_g$ is executed by all $M$ possible execution scenarios in order to accomplish the intent of process model $\boldsymbol{B}$, $1 \leq g \leq N$
- $S_{g,i}$ and $s_{g,i}$: $i^{th}$ execution scenario of process block $B_g$ and its ES probability, $1 \leq i \leq V_g$
- $R_{g,h}$ and $r_{g,h}$: transition between two sequential process blocks $B_g$ and $B_h$ and its transition probability
- $N_g$ : total number of process sub-blocks at a certain level of resolution where process block $B_g$ can be expressed as only one control-flow pattern containing multiple process sub-blocks, as shown in Fig. 1(b)
- $B_{g,j}$ : $j^{th}$ process sub-block in process block $B_g$, $1 \leq j \leq N_g$

Each explicit form of the uncertainty measure given in Sections 3.1 to 3.4 can be divided into two components: expectation of execution-related uncertainties, represented by $-\sum_{k=1}^{M} P(BS_k)\log_2 P(BS_k)$, and expectation of uncertainties within process blocks, represented by $\sum_{g=1}^{N} P(B_g)U(B_g)$. Therefore, the general form of the uncertainty measure of process model $\boldsymbol{B}$ is $U(\boldsymbol{B}) = -\sum_{k=1}^{M} P(BS_k)\log_2 P(BS_k) + \sum_{g=1}^{N} P(B_g)U(B_g)$.

### 3.1. Sequence

The sequence block is a set of process blocks executed in sequence. Fig. 2(a) shows sequence block $B_{\mathrm{SEQ}}$ with process blocks $B_g$ ($1 \leq g \leq N$). The explicit form of its uncertainty measure is $U(B_{\mathrm{SEQ}}) = \sum_{g=1}^{N} U(B_g)$.

**Proof.** Suppose that sequence block $B_{\mathrm{SEQ}}$ includes only two process blocks $B_1$ and $B_2$ (i.e. $N$=2). When process blocks $B_1$ and $B_2$ are sequentially and independently executed in scenarios $S_{1,i}$ and $S_{2,j}$ ($1 \leq i \leq V_1$ and $1 \leq j \leq V_2$) with transitions $R_{0,1}$ and $R_{1,2}$, its corresponding Shannon's entropy-based uncertainty is $\log_2(r_{0,1}s_{1,i}r_{1,2}s_{2,j})^{-1}$. Note that $r_{0,1} = r_{1,2} = 1$ and $\sum_{i=1}^{V_1} s_{1,i} = \sum_{j=1}^{V_2} s_{2,j} = 1$. Hereby, the uncertainty measure of sequence block $B_{\mathrm{SEQ}}$, defined as the expectation of uncertainties, is calculated as follows:

$$U(B_{\mathrm{SEQ}}) = \sum_{i=1}^{V_1}\sum_{j=1}^{V_2} P(R_{0,1})P(S_{1,i})P(R_{1,2})P(S_{2,j}) \log_2\{P(R_{0,1})P(S_{1,i})P(R_{1,2})P(S_{2,j})\}^{-1}$$

$$= -\{r_{0,1}r_{1,2}\log_2(r_{0,1}r_{1,2})\} + \left\{r_{0,1}r_{1,2}\sum_{i=1}^{V_1} s_{1,i}\log_2(s_{1,i})^{-1} + r_{0,1}r_{1,2}\sum_{j=1}^{V_2} s_{2,j}\log_2(s_{2,j})^{-1}\right\}$$

$$= -\sum_{k=1}^{1} P(BS_k)\log_2 P(BS_k) + \sum_{g=1}^{2} P(B_g)U(B_g) = \sum_{g=1}^{2} U(B_g)$$

The derivation can be easily extended to a sequence block with more than two process blocks (i.e. $N \geq 3$).□

### 3.2. AND-split

The AND-split block embraces two or more parallel branches all of which are executed concurrently after the preceding task. Fig. 2(b) shows AND-split block $B_{\mathrm{AND}}$ with process blocks $B_g$ ($1 \leq g \leq N$). The explicit form of its uncertainty measure is $U(B_{\mathrm{AND}}) = \sum_{g=1}^{N} U(B_g)$.

**Proof.** The AND-split block is equivalent to the sequence block in that all included process blocks are executed and the order of execution does not change the entropy. The proof follows directly from that of Section 3.1.□

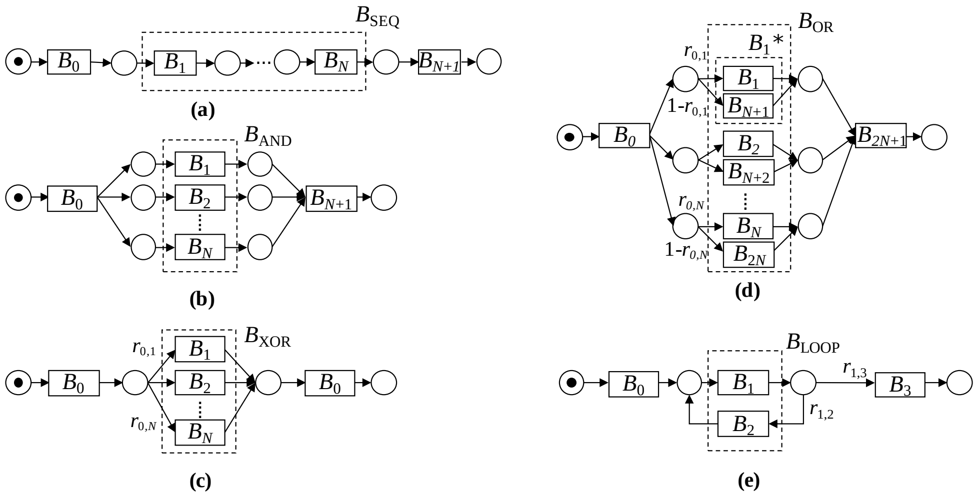


Fig. 2. Five primitive patterns: (a) Sequence block $\boldsymbol{B}_{\mathbf{SEQ}}$, (b) AND-split block $\boldsymbol{B}_{\mathbf{AND}}$, (c) XOR-split block $\boldsymbol{B}_{\mathbf{XOR}}$, (d) OR-split block $\boldsymbol{B}_{\mathbf{OR}}$, and (e) Loop block $\boldsymbol{B}_{\mathbf{LOOP}}$.

### 3.3. XOR-split

The XOR-split block embraces two or more branches only one of which is executed after the preceding task. Fig. 2(c) shows XOR-split block $B_{\mathrm{XOR}}$ with process blocks $B_g$ ($1 \leq g \leq N$). The explicit form of its uncertainty measure is $U(B_{\mathrm{XOR}}) = -\sum_{g=1}^{N} r_{0,g} \log_2(r_{0,g}) - \sum_{g=1}^{N} r_{0,g} \sum_{i=1}^{V_g} s_{g,i} \log_2(s_{g,i})$.

**Proof.** When process block $B_1$ is executed in scenario $S_{1,i}$ ($1 \leq i \leq V_1$) with transition $R_{0,1}$, its corresponding Shannon's entropy-based uncertainty is $\log_2(r_{0,1}s_1)^{-1}$. Note that $\sum_{g=1}^{N} r_{0,g} = 1$ and $\sum_{i=1}^{V_1} s_{1,i} = 1$. Hereby, the uncertainty measure of XOR-split block $B_{\mathrm{XOR}}$ is calculated as follows:

$$U(B_{\mathrm{XOR}}) = \sum_{g=1}^{N} \sum_{i=1}^{V_1} \mathrm{P}(R_{0,g}) \mathrm{P}(S_{g,i}) \log_2\{\mathrm{P}(R_{0,g}) \mathrm{P}(S_{g,i})\}^{-1}$$

$$= -\sum_{g=1}^{N} r_{0,g}\log_2(r_{0,g}) + \sum_{g=1}^{N} r_{0,g}\sum_{i=1}^{V_g} s_{g,i}\log_2(s_{g,i})^{-1}$$

$$= -\sum_{k=1}^{N} P(BS_k)\log_2 P(BS_k) + \sum_{g=1}^{N} P(B_g)U(B_g).\square$$

### 3.4. OR-split

The OR-split block embraces two or more branches each of which is decided to be executed or not concurrently and independently after the preceding task. Fig. 2(d) shows OR-split block $B_{\mathrm{OR}}$ with process blocks $B_g$ ($1\le g\le N$), dummy process blocks $B_h$ ($N+1\le h\le 2N$), and transitions between two sequential blocks. The dummy blocks are used to indicate that its corresponding process block is not executed. Note that $U(B_h)=0$ for $N+1\le h\le 2N$. The explicit form of its uncertainty measure is $U(B_{\mathrm{OR}}) = -\sum_{g=1}^{N}\left\{r_{0,g}\log_2(r_{0,g}) + (1-r_{0,g})\log_2(1-r_{0,g}) + r_{0,g}\sum_{i=1}^{V_g} s_{g,i}\log_2(s_{g,i})\right\}$.

**Proof.** As shown in Fig. 2(d), for convenience, let $B_g^*$ denote the $g^{th}$ logical block comprised of process block $B_g$, its dummy block $B_{N+g}$, and transitions $R_{0,g}$ and $R_{0,N+g}$. Note that $s_{N+g,1} = 1, \sum_{i=1}^{V_g} s_{g,i} = 1$. When process blocks $B_g$ and $B_{N+g}$ are exclusively executed in scenarios $S_{g,i}$ and $S_{N+g,j}$ ($1\le i\le V_1$ and $j=1$) with transitions $R_{0,g}$ and $R_{0,N+g}$, the uncertainty measure of logical block $B_g^*$ is calculated as a XOR-split block:

$$U(B_g^*) = -r_{0,g}\log_2(r_{0,g}) - (1-r_{0,g})\log_2(1-r_{0,g}) - r_{0,g}\sum_{i=1}^{V_g} s_{g,i}\log_2(s_{g,i})$$

$$= -P(BS_g)\log_2 P(BS_g) - P(BS_{N+g})\log_2 P(BS_{N+g}) + P(B_g)U(B_g)$$

Hereby, the uncertainty measure of $B_{\mathrm{OR}}$ is calculated as in an AND-split block with $B_g^*$ ($1\le g\le N$):

$$U(B_{\mathrm{OR}}) = \sum_{g=1}^{N} U(B_g^*) = -\sum_{g=1}^{N}\left\{r_{0,g}\log_2(r_{0,g}) + (1-r_{0,g})\log_2(1-r_{0,g}) + r_{0,g}\sum_{i=1}^{V_g} s_{g,i}\log_2(s_{g,i})\right\}$$

$$= -\sum_{k=1}^{2N} P(BS_k)\log_2 P(BS_k) + \sum_{g=1}^{2N} P(B_g)U(B_g).\square$$

### 3.5. Loop

The loop block embraces process blocks which are executed repeatedly until satisfying a certain condition. Fig. 2(e) shows Loop block $B_{\mathrm{LOOP}}$ with two process blocks $B_g$ ($1\le g\le 2$). The explicit form of its uncertainty measure is $U(B_{\mathrm{LOOP}}) = -\left\{1-(r_{1,2})^L\right\}\left\{\frac{r_{1,2}\log_2(r_{1,2})}{r_{1,3}} + \log_2(r_{1,3})\right\} + \left\{\frac{1-(r_{1,2})^{L+1}}{r_{1,3}}\right\}U(B_1) + \left\{\frac{r_{1,2}-(r_{1,2})^{L+1}}{r_{1,3}}\right\}U(B_2)$.

**Proof.** Note that $r_{0,1}= r_{2,1}=1$ and $r_{1,2}+r_{1,3}=1$. All possible block-level execution scenarios of loop block $B_{\mathrm{LOOP}}$ and corresponding ES probabilities are shown in Table 1. The $k^{th}$ block-level execution scenario $BS_k$ is equivalent to a sequence block containing ($k$+1) process blocks of $B_1$ and $k$ process blocks of $B_2$. Note that $B_0$ and $B_3$ are not included in $BS_k$ since those are not in the loop. Hence, the uncertainty measure of $BS_k$ is obtained by:

$$U(BS_k) = (k+1) * U(B_1) + k * U(B_2).$$

Loop block $B_{\text{LOOP}}$ can be viewed as an XOR-split with $L$ branches the transition probabilities of which are equal to the ES probabilities shown in Table 1. Note that $\sum_{k=0}^{L} P(BS_k) = r_{1,3} + r_{1,2}r_{1,3} + \ldots + (r_{1,2})^{L-1}r_{1,3} + (r_{1,2})^{L} = 1$ with $r_{1,3} = 1 - r_{1,2}$. Thus, calculating the uncertainty measure of loop block $B_{\text{LOOP}}$ can be written in terms of all possible block-level execution scenarios:

$$U(B_{\text{LOOP}}) = \sum_{k=0}^{L} P(BS_k)\log_2\left(P(BS_k)\right)^{-1} + \sum_{k=0}^{L} P(BS_k)\,U(BS_k)$$

The first term in the equation of $U(B_{\text{LOOP}})$ is further expanded as follows:

$$\sum_{k=0}^{L} P(BS_k)\log_2\left(P(BS_k)\right)^{-1} = \sum_{k=0}^{L-1}\left\{(r_{1,2})^k r_{1,3}\log_2\left((r_{1,2})^k r_{1,3}\right)^{-1}\right\} + (r_{1,2})^L\log_2\left((r_{1,2})^L\right)^{-1}$$

$$= -\left\{1-(r_{1,2})^L\right\}\left\{\frac{r_{1,2}\log_2(r_{1,2})}{r_{1,3}} + \log_2(r_{1,3})\right\}$$

The second term in the equation of $U(B_{\text{LOOP}})$ is further expanded as follows:

$$\sum_{k=0}^{L} P(BS_k)\,U(BS_k) = \sum_{k=0}^{L-1}\left[(r_{1,2})^k r_{1,3}\{(k+1)U(B_1) + kU(B_2)\}\right] + (r_{1,2})^L\{(L+1)U(B_1) + LU(B_2)\}$$

$$= \left\{\frac{1-(r_{1,2})^{L+1}}{r_{1,3}}\right\}U(B_1) + \left\{\frac{r_{1,2}-(r_{1,2})^{L+1}}{r_{1,3}}\right\}U(B_2) = P(B_1)U(B_1) + P(B_2)U(B_2)$$

Therefore, the explicit form of the uncertainty measure for $B_{\text{LOOP}}$ with two process blocks $B_g$ ($1 \leq g \leq 2$) is:

$$U(B_{\text{LOOP}}) = -\left\{1-(r_{1,2})^L\right\}\left\{\frac{r_{1,2}\log_2(r_{1,2})}{r_{1,3}} + \log_2(r_{1,3})\right\} + \left\{\frac{1-(r_{1,2})^{L+1}}{r_{1,3}}\right\}U(B_1) + \left\{\frac{r_{1,2}-(r_{1,2})^{L+1}}{r_{1,3}}\right\}U(B_2)$$

$$= -\sum_{k=0}^{L} \boldsymbol{P(BS_k)\log_2 P(BS_k)} + \sum_{g=1}^{2} \boldsymbol{P(B_g)U(B_g)}.\square$$

Table 1. Execution scenarios and ES probabilities of $B_{\text{LOOP}}$

| No. | Block-level execution scenarios of $B_{\text{LOOP}}$ | ES probabilities of $B_{\text{LOOP}}$ |
|---|---|---|
| 0 | $B_0 - B_1 - B_3$ | $P(BS_0) = r_{0,1}r_{1,3} = r_{1,3}$ |
| 1 | $B_0 - B_1 - B_2 - B_1 - B_3$ | $P(BS_1) = r_{0,1}r_{1,2}r_{2,1}r_{1,3} = r_{1,2}\, r_{1,3}$ |
| | ⋮ | ⋮ |
| $L$-1 | $B_0 - B_1 - B_2 - B_1 - \ldots - B_1 - B_2 - B_1 - B_3$ | $P(BS_{L-1}) = r_{0,1}r_{1,2}r_{2,1} \ldots r_{2,1}\, r_{1,2}r_{2,1}r_{1,3} = (r_{1,2})^{L-1}r_{1,3}$ |
| $L$ | $B_0 - B_1 - B_2 - B_1 - \ldots - B_1 - B_2 - B_1 - B_2 - B_1 - B_3$ | $P(BS_L) = r_{0,1}r_{1,2}r_{2,1} \ldots r_{2,1}r_{1,2}r_{2,1}r_{1,2}\, r_{2,1} = (r_{1,2})^L$ |

### 3.6. Uncertainty of process model comprised by tasks only

As shown in Table 2, the explicit forms of the uncertainty measure $U(\boldsymbol{B})$ given in Sections 3.1 to 3.5 reduce to ones for the corresponding control-flow pattern only, when each process block reduces to a task with its ES probability of 1 and its uncertainty measure of zero.

Table 2. Explicit forms of the uncertainty measure for control-flow patterns

| Control-flow pattern | Explicit form of the uncertainty measure for control-flow pattern only |
|---|---|
| Sequence | $U(B_{\text{SEQ}})=0$ |
| AND-split | $U(B_{\text{AND}})=0$ |
| XOR-slit | $U(B_{\text{XOR}})=-\sum_{g=1}^{N} r_{0,g}\log_2(r_{0,g})$ |
| OR-split | $U(B_{\text{OR}})=-\sum_{g=1}^{N}\{r_{0,g}\log_2(r_{0,g})+(1-r_{0,g})\log_2(1-r_{0,g})\}$ |
| Loop | $U(B_{\text{LOOP}})=-\left\{1-(r_{1,2})^{L}\right\}\left\{\frac{\log_2(r_{1,2})}{r_{1,3}}-\log_2(r_{1,3})\right\}$ |

For comparing the uncertainties of control-flow patterns in terms of the structure only excluding the effect of transition probabilities, assume that all splitting points of each control flow pattern in Table 2 have the equal transition probabilities. That is, $r_{0,g}$=1/$N$ for XOR-split, $r_{0,g}$=1/2 for OR-split, and $r_{1,2}$= $r_{1,3}$=1/2 for Loop. The sequence and AND-split control-flow have the uncertainty measure of zero since the execution order of tasks is known with no uncertainty. The loop control-flow converges to 1 as $L$ increases. The uncertainty of the XOR-split is minimized with the smallest number of branches (i.e. $g$=2) and increases as the number of branches increases. The OR-split always shows the higher uncertainty than the XOR-split at the same number of branches.

## 4. Illustration I – How to apply the entropy measure?

This section illustrates the uncertainty measure with a process model fractionated into separate blocks of primitive patterns. Fig. 3(a) shows a process model with three primitive control-flows and tasks $t_i$ $(1 \le i \le 8)$. The XOR-split has two branches with an equal transition probability and the loop has the recursive transition probability of 0.2 with no bounded number of iterations. The uncertainty measure of this process model is calculated by recursively implementing two steps: 1) identifying maximum-sized logical blocks consisting of tasks in only one primitive control-flow and 2) calculating the uncertainty measure of the identified logical blocks. The identified logical blocks are considered as tasks at the next iteration.

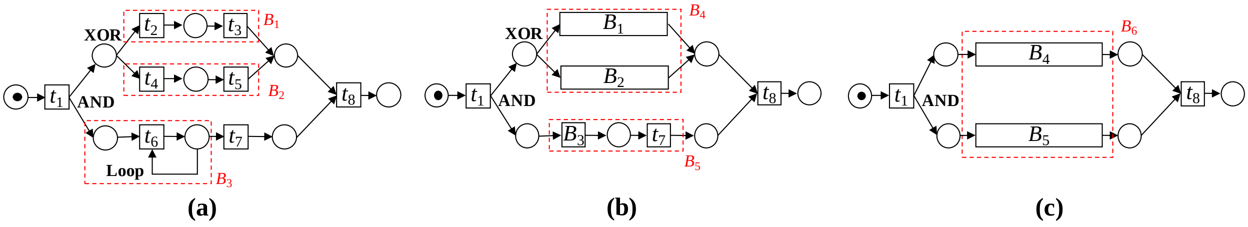


Fig. 3. Logical block-based approach for calculating the uncertainty measure: (a) logical blocks at the first iteration, (b) logical blocks at the second iteration, and (c) logical blocks at the third iteration.

At the first iteration, as shown in Fig. 3(a), $B_1$ and $B_2$, and $B_3$ are identified as maximum-sized logical blocks with only one sequence control-flow or only one loop control-flow. Note that $U(t_i) = 0$ for $1 \le i \le 8$. The uncertainty measures of sequence blocks $B_1$ and $B_2$ are equal to 0. As for the loop, $U(B_3) = \lim_{L\to\infty} -\{1 - (0.2)^L\}\left\{\frac{0.2\log_2(0.2)}{1-0.2} + \log_2(0.8)\right\} = 0.902$. At the second iteration, as shown in Fig. 3(b), $B_4$ and $B_5$ are identified as maximum-sized logical blocks with only one XOR control-flow or only one sequence control-flow. $U(B_4) = -2 * (0.5\log_2(0.5)) = 1$ with $U(B_1)= U(B_2)=0$. $U(B_5) = U(B_3) + U(t_7) = 0.902$. At the third iteration, as shown in Fig. 3(c), $B_6$ is identified as a maximum-sized logical block with only one AND-split control-flow. $U(B_6) = U(B_4) + U(B_5) = 1.902$. Finally, the process model can be viewed as a sequence block with $t_1$, $B_6$, and $t_8$. Hence, $U(\boldsymbol{B}) = U(t_1) + U(B_6) + U(t_8) = 1.902$.

**5. Illustration II – A healthcare use case**

Processes can coordinate and manage tens and even hundreds of tasks which require often expensive and scarce resources to be properly executed. In hospitals, for example, healthcare processes can manage several tasks which require specific and expensive resources that range from doctors, X-rays and CAT scan equipments, EKGs, ambulances, surgical rooms, digital records, etc. During the design of a process, experts associate with each task adequate roles and resources which are required for its execution.

For example, consider two patients transported to an emergency room by ambulance. Patients ***A*** and ***B*** are suffering fracture and dyspnea (shortness of breath), respectively. Before their arrival, a manager schedules the assignment of resources for their checkup processes. The number of possibly required resources varies according to their symptoms. So does the certainty for the use of each resource. In assigning X-rays, EKGs, CTs, and MRIs for their checkup processes, the manager is fully convinced that an X-ray is required for patient ***A***. However, there is no such resource for patient ***B*,** since the dyspnea can be caused by abnormalities in many organs such as the heart, lungs, and the brain. These two different uncertainties of checkup processes can be reflected and distinguished by our proposed measure. The manager has more difficulty in efficiently assigning limited resources, as the more checkup processes of high entropy are initiated. The entropy of the checkup process for patient ***B*** could be reduced by adding a task of reviewing his/her past history and finding out the frequent causes of his/her dyspnea.

For healthcare institutions, knowing beforehand that a healthcare process model has low entropy (i.e. low uncertainty) enables the creation of more accurate assignment schedules for resources and makes possible a better planning. Healthcare professionals, medical equipment, and physical facilities can be allocated ahead of time knowing that they will be indeed needed to provide an efficient and cost effective care to patients.

## 6. Conclusion

For assessing the predictability and managerial efficiency of process models, we propose an entropy-based measure to quantify the uncertainty of business process models. The proposed measure enables to estimate the process uncertainty and interpret it in terms of execution-related uncertainties and process block-related uncertainties. These types of measures can be used to guide business process designers and analysts in developing and improving processes to be more predictable, less complex, less prone to errors, and simpler to understand [3,13].

## References

[1] L.S. Gonzalez, F.G. Rubio, F.R. Gonzalez, and M.P. Velthuis, Measurement in business processes: a systematic review, Business Process Management Journal 16(1) (2010) 114-134.

[2] I.T.P. Vanderfeesten, H.A. Reijers, J. Mendling, Wil M. P. van der Aalst, and J. Cardoso, On a quest for good process models: the cross-connectivity metric, In: Proc. of Int. Conf. on Advanced Infor. Sys. Eng. (2008) 480-494.

[3] J. Cardoso, How to Measure the Control-flow Complexity of Web Processes and Workflows, in: Workflow Handbook 2005, Future Strategies Inc., Lighthouse Point, 2005. pp. 199-212.

[4] K.B. Rassen, W.M.P. van der Aalst, Complexity metrics for Workflow nets, Information and Software Technology 51(3) (2009) 610-626.

[5] J. Mendling, Testing Density as a Complexity Metric for EPCs, Technical Report JM-2006-11-15, Vienna University of Economics and Business Administration (2006).

[6] H.A. Reijers, I.T.P. Vanderfeesten, Cohesion and coupling metrics for workflow process design, Lecture Notes on Computer Science 3080, Springer-Verlag, Berlin (2004) 290-305.

[7] J. Mendling, M. Moser, G. Neumann, H.M.W. Verbeek, B.F. van Dongen, and W.M.P. van der Aalst, Faulty EPCs in the SAP Reference Model, Lecture Notes in Computer Science 4102 (2006) 451-457.

[8] G.M. Muketha, A.A.A. Ghani, M.H. Selamat, and R. Atan, A survey of business process complexity metrics, Inform. Technol. J. 9(7) (2010) 1336-1344.

[9]W.J. Hopp and M.L. Spearman, Factory Physics: Foundations of Manufacturing Management, 2nd Ed., UK:Irwin McGraw-Hill, London, 2001.

[10] J. V. Zidek and C. van Eeden, Uncertainty, Entropy, Variance and the Effect of Partial Information, Lecture Notes-Monograph Series 42 (2003) 155-167.

[11] APICS Dictionary, 8th ed., Falls Church: American Production and Inventory Control Society, Inc., 1995.

[12] C.E. Shannon, A mathematical theory of communication, Bell System Technical Journal 27 (1948) 379-423.

[13] J. Mendling, H.A. Reijers, J. Cardoso, What makes process models understandable?, Lecture Notes on Computer Science 4714, Springer-Verlag, Berlin (2007) 48-36.