

## LETTER

## Comparing Process Behaviors with Finite Chu Spaces

Xutao DU<sup>†\*a)</sup>, Student Member, Chunxiao XING<sup>††</sup>, and Lizhu ZHOU<sup>†††</sup>, Nonmembers

**SUMMARY** We develop a distance function for finite Chu spaces based on their behavior. Typical examples are given to show the coincidence between the distance function and intuition. We show by example that the triangle inequality should not be satisfied when it comes to comparing two processes.

**key words:** Chu spaces, processes, distance function, triangle inequality

## 1. Introduction

### 1.1 Motivation

Various applications require the comparison between process models. For example, when developers try to find a software module that is the best match for their requirements in a software module database, they need to compare the modules with their requirements in a certain way. Requirements can be described in different ways, such as an example software module, a temporal formula, a first-order formula, etc. Various methods for determining the *equivalence* or *satisfaction* between the target process and the requirements have been proposed. For example, bisimilarity checking has been used between two processes modeled by some kind of process calculi, modeling checking has been used between a process model and a temporal formula. However, all of them result in a 0 or 1 answer. In reality, a perfect 100% matching hardly exists, and we need a way to find out how close a process is to the requirements [1].

In modern computing paradigms (e.g. services computing and cloud computing), concurrency has been used extensively. However, it is well known concurrent programs are notoriously difficult to test and debug. Formal methods provide an effective way to tackle this problem. Chu spaces originate from category theory and have been proposed as a promising model of concurrency [2], [3]. Different from other formalisms (e.g. automata, Petri nets and process calculi), Chu spaces support both the true concurrency semantics and compositional algebraic manipulation simultaneously. Moreover, Chu spaces have unique

structures that support efficient verification algorithms for process-algebraic specifications [4]. Theoretical researches on Chu spaces have been fruitful (e.g. [5]–[7]). Applications of Chu spaces include the modeling of RTL (Register Transfer Level) [8], Verilog [9], general concurrent programs [4], WS-BPEL [10], and physical systems [11]. However, there is no research on how to measure the degree of similarity between Chu spaces until now. In this paper, we present a behavior-based distance function for Chu spaces, which will provide an effective way for the analysis of concurrent processes.

The rest of this paper is organized as follows: Section 2 introduces the basic theory of Chu spaces. Section 3 presents the distance function for finite Chu spaces together with typical examples. Section 4 presents concluding remarks and future research directions.

## 2. Chu Spaces

**Finite Chu Spaces** A finite Chu space  $C = (E, X)$  over the alphabet  $K$  consists of a finite set  $E$  of *events* and a finite set  $X \subseteq K^E$  of *states*.

Different  $K$  presents different modeling power and complexity of Chu spaces. In this paper, we use  $K = \{0, \angle, 1, \times\}$ , which is proposed in [5]. This  $K$  supports both true concurrency and branching time semantics, which means it is suitable for the refined modeling of concurrent behavior [12]. The structure of  $K$  is depicted in Fig. 1.

Example Chu spaces will be given in the next section when discussing the distance function.

## 3. Distance between Chu Spaces

### 3.1 Pseudometric Space

**Pseudometric Space** A pseudometric space is an ordered pair  $(S, d)$  where  $S$  is a nonempty set and  $d$  a distance function  $d : S \times S \rightarrow [0, \infty]$  such that for all  $x, y, z \in S$

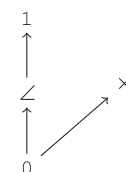


Fig. 1 The structure of  $K$ .

Manuscript received June 30, 2010.

<sup>†</sup>The author is with the Department of Computer Science and Technology, Tsinghua University, China.

<sup>††</sup>The author is with the Research Institute of Information Technology, Tsinghua University, China.

<sup>†††</sup>The author is with the Faculty of Department of Computer Science and Technology, Tsinghua University, China.

\*Presently, with Institute of Electronic System Engineering.

a) E-mail: du.xutao@gmail.com

DOI: 10.1587/transinf.E94.D.1321

1.  $d(x, y) \geq 0$
2.  $d(x, y) = d(y, x)$
3.  $d(x, z) \leq d(x, y) + d(y, z)$ , i.e. the triangle inequality

We will develop a distance function  $d_c$  for Chu spaces.  $d_c$  is not a metric because the third condition (i.e. the triangle inequality) is not satisfied. However, we will show that it is not only acceptable but also a natural result when comparing two processes.

### 3.2 A Distance Function for Chu Spaces

**Behavioral precision and recall** Let  $C_1 = (E_1, X_1)$  and  $C_2 = (E_2, X_2)$  be two finite Chu spaces. Then  $E = E_1 \cap E_2$  is the set of common events of  $C_1$  and  $C_2$ . We define

$$\text{precision}(C_1, C_2) = \frac{|X_1|E \cap X_2|E|}{|X_1|}$$

$$\text{recall}(C_1, C_2) = \frac{|X_1|E \cap X_2|E|}{|X_2|}$$

The name *precision* and *recall* is taken after those used in [1]. The symbol  $|$  means function restriction. A high  $\text{precision}(C_1, C_2)$  means most states of  $C_2$  are also states of  $C_1$ . And a high  $\text{recall}(C_1, C_2)$  means most states of  $C_1$  are also states of  $C_2$ . Using both of them, the distance function for Chu spaces is defined as follows.

**A distance function** Let  $\text{Chu}$  be the set of Chu spaces. The distance function  $d_c : \text{Chu} \times \text{Chu} \rightarrow [0, \infty]$  is defined by

$$d_c(C_1, C_2) = 1 - \frac{\text{precision}(C_1, C_2) + \text{recall}(C_1, C_2)}{2}$$

### 3.3 Properties of $d_c$

**Theorem 3.1:** For all  $C_1, C_2 \in \text{Chu}$ ,  $d_c(C_1, C_2) \geq 0$ .

**Proof**

$$\begin{aligned} d_c(C_1, C_2) &= 1 - \frac{\text{precision}(C_1, C_2) + \text{recall}(C_1, C_2)}{2} \\ &= 1 - \frac{\frac{|X_1|E \cap X_2|E|}{|X_1|} + \frac{|X_1|E \cap X_2|E|}{|X_2|}}{2} \\ &= 1 - \frac{|X_2||X_1|E \cap X_2|E| + |X_1||X_1|E \cap X_2|E|}{2|X_1||X_2|} \\ &= 1 - \frac{(|X_1| + |X_2|)(|X_1|E \cap X_2|E|)}{2|X_1||X_2|} \end{aligned}$$

Now suppose  $|X_1| \geq |X_2|$ , we have  $|X_1| + |X_2| \leq 2|X_1|$  and  $|X_1|E \cap X_2|E| \leq |X_2|$ . Then

$$d_c(C_1, C_2) \geq 1 - \frac{2|X_1||X_2|}{2|X_1||X_2|} = 0$$

A similar argument can be given when  $|X_1| \leq |X_2|$ . ■

**Theorem 3.2:** For all  $C_1, C_2 \in \text{Chu}$ ,  $d_c(C_1, C_2) = d_c(C_2, C_1)$ .

**Proof**

$$\begin{aligned} d_c(C_1, C_2) &= 1 - \frac{\text{precision}(C_1, C_2) + \text{recall}(C_1, C_2)}{2} \\ &= 1 - \frac{\frac{|X_1|E \cap X_2|E|}{|X_1|} + \frac{|X_1|E \cap X_2|E|}{|X_2|}}{2} \\ &= 1 - \frac{|X_2||X_1|E \cap X_2|E| + |X_1||X_1|E \cap X_2|E|}{2|X_1||X_2|} \\ &= 1 - \frac{(|X_1| + |X_2|)(|X_1|E \cap X_2|E|)}{2|X_1||X_2|} \\ d_c(C_2, C_1) &= 1 - \frac{\text{precision}(C_2, C_1) + \text{recall}(C_2, C_1)}{2} \\ &= 1 - \frac{\frac{|X_2|E \cap X_1|E|}{|X_2|} + \frac{|X_2|E \cap X_1|E|}{|X_1|}}{2} \\ &= 1 - \frac{|X_1||X_2|E \cap X_1|E| + |X_2||X_2|E \cap X_1|E|}{2|X_1||X_2|} \\ &= 1 - \frac{(|X_1| + |X_2|)(|X_1|E \cap X_2|E|)}{2|X_1||X_2|} \\ &= d_c(C_1, C_2) \end{aligned}$$

### 3.4 Examples

Consider the Chu spaces in Table 1.

Table 2 shows the distance between  $C_1$  and  $C_2$ ,  $d_c(C_1, C_2) = 3/5$ . Actually, only the initial state  $x_1$  and the final state  $x_5$  of  $C_1$  have their counter parts in  $C_2$  (i.e.  $u_1$  and  $u_5$ ). Hence, their distance are very far. From another perspective, since  $C_1$  is the sequential execution of  $a$  and  $b$  (i.e.  $a; b$ ) and  $C_2$  is the sequential execution of  $b$  and  $a$  (i.e.  $b; a$ ), they are quite different and then the distance between them should be far. Hence, the results match with our intuition.

Table 3 shows the distance between  $C_1$  and  $C_3$  is  $4/5$ , which is a relatively large value in  $[0, 1]$ .  $C_1 : a; b$  let  $a$  executes first then  $b$  whereas  $C_3 : a + b$  nondeterministically

**Table 1** Chu Spaces  $C_1 : a; b$ ,  $C_2 : b; a$ ,  $C_3 : a + b$  and  $C_4 : a||b$ .

$C_1 : a; b$   

$C_1$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$
$a$	0	$\perp$	1	1	1
$b$	0	0	0	$\perp$	1

$C_2 : b; a$   

$C_2$	$u_1$	$u_2$	$u_3$	$u_4$	$u_5$
$a$	0	0	0	$\perp$	1
$b$	0	$\perp$	1	1	1

$C_3 : a + b$   

$C_3$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$y_6$	$y_7$
$a$	0	0	$\perp$	1	$\times$	$\times$	$\times$
$b$	0	$\times$	$\times$	$\times$	0	$\perp$	1

$C_4 : a||b$   

$C_4$	$z_1$	$z_2$	$z_3$	$z_4$	$z_5$	$z_6$	$z_7$	$z_8$	$z_9$
$a$	0	$\perp$	1	0	$\perp$	1	0	$\perp$	1
$b$	0	0	0	$\perp$	$\perp$	$\perp$	1	1	1

**Table 2**  $\text{precision}(C_1, C_2)$  and  $\text{recall}(C_1, C_2)$ .

$$\text{precision}(C_1, C_2) = \frac{||x_1, x_5||}{||x_1, x_2, x_3, x_4, x_5||} = 2/5$$

$$\text{recall}(C_1, C_2) = \frac{||x_1, x_5||}{||u_1, u_2, u_3, u_4, u_5||} = 2/5$$

$$d_c(C_1, C_2) = 1 - \frac{2/5 + 2/5}{2} = 3/5$$

**Table 3**  $precision(C_1, C_3)$  and  $recall(C_1, C_3)$ .

$$precision(C_1, C_3) = \frac{|\{x_1\}|}{|\{x_1, x_2, x_3, x_4, x_5\}|} = 1/5$$

$$recall(C_1, C_3) = \frac{|\{x_1\}|}{|\{u_1, u_2, u_3, u_4, u_5\}|} = 1/5$$

$$d_c(C_1, C_3) = 1 - \frac{1/5 + 1/5}{2} = 4/5$$

**Table 4**  $precision(C_1, C_4)$  and  $recall(C_1, C_4)$ .

$$precision(C_1, C_4) = \frac{|\{x_1, x_2, x_3, x_4, x_5\}|}{|\{x_1, x_2, x_3, x_4, x_5\}|} = 1$$

$$recall(C_1, C_4) = \frac{|\{x_1, x_2, x_3, x_4, x_5\}|}{|\{z_1, z_2, \dots, z_9\}|} = 5/9$$

$$d_c(C_1, C_4) = 1 - \frac{1 + 5/9}{2} = 2/9$$

**Table 5** Chu Spaces  $C_a : a$ ,  $C_b : b$  and  $C_1 : a; b$ .

$C_a : a$				$C_b : b$			
$C_a$	$a_1$	$a_2$	$a_3$	$C_b$	$b_1$	$b_2$	$b_3$
$a$	0	$\angle$	1	$b$	0	$\angle$	1

$C_1 : a; b$					
$C_1$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$
$a$	0	$\angle$	1	1	1
$b$	0	0	0	$\angle$	1

chooses one of  $a$  and  $b$  to execute and cancel the other. Both  $a$  and  $b$  can finish their job in  $C_1$ , but only one can finish in  $C_3$ , therefore their distance should be very far from the intuitional point of view. Thus, results also match with our intuition in this case.

Table 4 shows the case for  $C_1$  and  $C_4$ .  $C_4 : a||b$  is the concurrent composition of  $a$  and  $b$ , in which both of them can run independently. It is well known that sequential composition is just a special case of concurrent composition. Therefore,  $C_1 : a; b$  should be not so far from  $C_4 : a||b$ . Thus the relatively small value  $2/9$  also matches with the intuition.

### 3.5 Why Triangle Inequality Is Not Required

We argue that for a distance function between processes, the triangle inequality is not required. Consider the Chu spaces in Table 5.  $C_a$  and  $C_b$  are the execution of  $a$  and  $b$ , respectively.  $C_1$  is the sequential execution of  $a$  and  $b$ .

From Table 6, we can see both  $d_c(C_a, C_1)$  and  $d_c(C_b, C_1)$  are  $3/10$ . It is also intuitive that the event  $a$  executes from 0 to  $\angle$  then to 1 in both  $C_a$  and  $C_1$ . And event  $b$  executes from 0 to  $\angle$  then to 1 in both  $C_b$  and  $C_1$ . Therefore, the distance  $3/10$  is reasonable.

However, when it comes to  $C_a$  and  $C_b$ , they even do not have any common events, therefore, their distance should be the most far, i.e.  $d_c(C_a, C_b) = 1$ . It then follows that

$$d_c(C_a, C_b) = 1 \geq d_c(C_a, C_1) + d_c(C_1, C_b) = 3/5$$

**Table 6**  $d_c(C_a, C_1)$ ,  $d_c(C_b, C_1)$  and  $d_c(C_a, C_b)$ .

$$precision(C_a, C_1) = \frac{|\{a_1, a_2, a_3\}|}{|\{a_1, a_2, a_3\}|} = 1$$

$$recall(C_a, C_1) = \frac{|\{a_1, a_2, a_3\}|}{|\{x_1, x_2, x_3, x_4, x_5\}|} = 2/5$$

$$d_c(C_a, C_1) = 1 - \frac{1 + 2/5}{2} = 3/10$$

$$precision(C_b, C_1) = \frac{|\{b_1, b_2, b_3\}|}{|\{a_1, a_2, a_3\}|} = 1$$

$$recall(C_b, C_1) = \frac{|\{b_1, b_2, b_3\}|}{|\{x_1, x_2, x_3, x_4, x_5\}|} = 2/5$$

$$d_c(C_b, C_1) = 1 - \frac{1 + 2/5}{2} = 3/10$$

$$d_c(C_a, C_1) + d_c(C_b, C_1) = 3/5$$

$$precision(C_a, C_b) = 0$$

$$recall(C_a, C_b) = 0$$

$$d_c(C_a, C_b) = 1 - 0 = 1$$

since  $d_c(C_1, C_b) = d_c(C_b, C_1) = 3/10$ .

Because the result of  $d_c(C_a, C_1)$ ,  $d_c(C_b, C_1)$  and  $d_c(C_a, C_b)$  all match with our intuition. It follows that it is a natural result in comparing two Chu spaces processes that the triangle inequality should not be satisfied by the distance function.

## 4. Conclusions and Future Work

We present a distance function between Chu spaces. To our best knowledge, it is the first one in the Chu spaces theory. We show how this distance function can be used to compare processes modeled by Chu spaces through typical examples. And show why the triangle inequality is not applicable when comparing processes' behaviors. Since Chu spaces are ideal new model of concurrency, we suppose our method can be used as a base for further study of comparing concurrent processes.

This work is partially supported by the National Basic Research Program of China (973 Program) No.2011CB302302.

## References

- [1] A.A. de Medeiros, W. van der Aalst, and A. Weijters, "Quantifying process equivalence based on observed behavior," Data & Knowledge Engineering, vol.64, pp.55–74, 2008.
- [2] V. Gupta, Chu Spaces: A Model of Concurrency, Ph.D. Thesis, Stanford University, Sept. 1994.
- [3] V.R. Pratt, "Chu spaces," Course Notes for the School in Category Theory and Applications, July 1999.
- [4] X. Du, C. Xing, and L. Zhou, "Modeling and verifying concurrent programs with finite Chu spaces," J. Comput. Sci., vol.25, no.6, pp.1168–1183, 2010.
- [5] V.R. Pratt, "Transition and cancellation in concurrency and branching time," Math. Structures in Comp. Sci., vol.13, no.4, pp.485–529, Aug. 2003.
- [6] H. Devarajan, D.J.D. Hughes, G.D. Plotkin, and V.R. Pratt, "Full completeness of the multiplicative linear logic of chu spaces," Proc.

- 14th Annual IEEE Symposium on Logic in Computer Science (LICS'99), pp.234–242, Trento, Italy, July 1999.
- [7] F.P. Huang, M. Droste, and G.Q. Zhang, “A monoidal category of bifinite chu spaces,” *Electron. Notes Theor. Comput. Sci.*, vol.212, pp.285–297, 2008.
- [8] J. Vinh and P.C. Bowen, “Semantics of rtl and validation of synthesized RTL designs using formal verification in reconfigurable computing systems,” *Proc. 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS 2005)*, pp.247–254, Greenbelt, Maryland, USA, April 2005.
- [9] L. Ivanov, “Modeling non-iterated system behavior with chu spaces,” *Proc. 2008 International Conference on Computer Design (CDES 2008)*, pp.145–150, CSREA Press, Las Vegas, Nevada, USA, July 2008.
- [10] X. Du, C. Xing, and L. Zhou, “A Chu spaces semantics of control flow in BPEL,” *Proc. 2009 Asia-Pacific Services Computing Conference (APSCC 2009)*, pp.142–149, Biopolis, Singapore, Dec. 2009.
- [11] S. Abramsky, “Big toy models: Representing physical systems as chu spaces,” *Tech. Rep. RR-09-08*, OUCL, Sept. 2009.
- [12] R. van Glabbeek and U. Goltz, “Refinement of actions and equivalence notions for concurrent system,” *Acta Informatica*, vol.37, no.4/5, pp.229–327, 2001.
-