**Flow**

Process Mining & RFID

Dennis Lin  
University Distinguished Professor  
Department of Supply Chain & Information Systems  
The Pennsylvania State University  
DennisLin@psu.edu

---

**What is a Process?**

- A series of operations performed in the making or treatment of a product
- A particular course of action intended to achieve a result

---

**How to Wash your Hands?**

1. WET: 脫下手中飾物，手沾濕
2. RUB: 取清潔劑搓洗指尖、指縫、手心、手臂，至少20秒
3. POUR: 在流動的自来水下，沖淨雙手。
4. HOLD: 撇水沖洗水龍頭
5. WIPE: 用乾淨的紙巾擦乾雙手，以用過的紙巾開水龍頭

---

**Process Mining**

- How to Wash Your Hands?  
  - Five Steps for Washing your hands
- When your company faces problems, is there any formal procedure to follow?  
  - Business Process Mining  
    - Via Strategy/Design  
    - Via Data
Wash your hands!

More Example: Process
- Quality Control
  - Plan-Do-Act-Check
- Six-Sigma
  - Define-Measure-Analysis-Improve-Control

- How to organize a big conference?
- How about Business Process?

Six Sigma Methodologies
- DMAIC
  - Define
  - Measure
  - Analyze
  - Improve
  - Control

- DMADV
  - Define
  - Measure
  - Analyze
  - Design
  - Verify

Develop a standard procedure/process when unexpected happens!
How to build up the process?

Is this an optimal process?

Statistical Process Mining

Data-Based Process Mining

Process Mining: Data

Table 1. A process log.

Fig. 1. A process model corresponding to the process log.
RFID: Radio Frequency Identification

Bar Code

Anatomy of a Barcode

The components of a 96-bit electronic product code (in Hex)

<table>
<thead>
<tr>
<th>Header</th>
<th>EPC Manager (will be assigned to companies)</th>
<th>Object Class</th>
<th>Product Code</th>
<th>Serial Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 bits</td>
<td>28 bits</td>
<td>24 bits</td>
<td>36 bits</td>
<td></td>
</tr>
</tbody>
</table>

The RFID tag responds to the reader by broadcasting its EPC, which is a 96-bit code consisting of:
- 8 bits of header information
- 28 bits identifying the organization that assigned the code
- 24 bits identifying the type of product
- 36 bits representing serialization information for the product

Source: Avicon white paper, 2003
**RF Communication**

- Electromagnetic waves modulated to carry data/signals
- Two different ways to generate ways
  - Inductive coupling
  - Close proximity electromagnetic wave
  - Propagating electromagnetic waves
- The fundamental RF communication theories apply—nothing new.
- New: the cost, size, signal processing capability.

**RFID**

- Radio Frequency Identification (RFID) technology has been in use since the 1950’s
- Advocated as Electronic Product Code™ or EPC™

---

**RFID Chokepoint Layout**

- Clearance Area
- Read Area
- Antennas
- Asset Flow

- 8 ft
- 4 ft
- 12 ft
Some Current Applications

Low Frequencies     High Frequencies

RF-ID Tags

Per Mr. Jun Takei
Auto Theft Protection

Waste Management (Shanghai, China)

Illegal Liquor Protection

Hotel Management Entrance Control
Security Management

13.56MHz High Frequency

APEC
Olympic at Beijing
International Meetings
People & Cars

Reformation (Jail) Management
Japan school kids to be tagged with RFID chips
The chips will be put onto kids' schoolbags, name tags or clothing to track the kids' movements.

By Jo Best
27 March, 2006
http://news.com.com/Japan+school+kids+to+be+tagged+with+RFID+chips/2100-1012_3-5266700.htm?tag=sas.email

Library Book Management (SNG)

Application Possibilities
- Shelf Reading
- Shelving
- Inventory Management
- Data Collection
- Determining Shelf Order
- Searching
- Finding Lost/Missing Items
- Pre-sort Activities
- Weeding Items

RFID in Animal Husbandry
Next Generation RFID!!

EPC
- LF: 100-150 KHz
  Hotel Card, Auto Protection, Manufacturing Line, Animal Husbandry
- HF: 13.56 MHz
  - 14443
    - 14443A—CashCard
    - 14443B—ID, Passport
  - 15693—10cm with password
- UHF (1m)
  - 915MHz: EPC ISO1800-6C, Inventory & Supply Chain
  - 2.45 GHz
  - 5.8 GHz: High Speed EZ-Pass

A Data Explosion is Coming!
Are You Ready?

Impact to Statistics
How to analyze the population data?
Bar-coding

- Bar coding is one of the most popular and cost effective forms of automated data collection systems.
- Fixed Barcode: Contains static information that is the same for all products of the same brand and type. An example is a UPC barcode on a 12 oz. can of coke.
- Variable barcode: A variable data bar code contains data that identifies a single product and changes for each separate product.

Lin, Dennis K.J. and Wadhwa, Vijay
(Cover Story of Quality Progress, Feb 2007)
**RFID**

- RFID uses radio waves to automatically identify people or objects.
- Old technology but increased affordability, scalability, data processing capability.
- Capable of identifying each and every object uniquely in a supply chain.

**Basic Structure**

- Manufacturing
  - Receiving
  - Storage
  - Picking
  - WIP
  - Shipping
  - PLM
  - Quality Control
  - Labor productivity
  - Inventory mgmt.
- Warehouse
  - Receiving
  - Storage
  - Picking
  - Shipping
  - Cross-Docking
  - Stock Visibility
  - Replenishment
  - Pricing
  - Theft Reduction
  - Training
- Retailer
  - Product Receiving
  - Checkout
  - Theft Reduction
  - Labor productivity

**Data Structure: S.I.T. Space**

- **State**
- **ID**
- **TimeStamp**

- Namely, {where, which, when}

- Particularly interested in the difference between Execution & Planning (Sense & Response)

**Sample RFID Data**

<table>
<thead>
<tr>
<th>Location</th>
<th>EPC</th>
<th>Date/time</th>
<th>Reader</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC 123</td>
<td>0023000.341815.500000024</td>
<td>08-04-05 23:15</td>
<td>inbound</td>
</tr>
<tr>
<td>DC 123</td>
<td>0023000.341815.500000024</td>
<td>08-09-05 7:54</td>
<td>conveyor</td>
</tr>
<tr>
<td>DC 123</td>
<td>0023000.341815.500000024</td>
<td>08-09-05 9:25</td>
<td>outbound</td>
</tr>
<tr>
<td>ST 987</td>
<td>0023000.341815.500000024</td>
<td>08-09-05 20:31</td>
<td>inbound</td>
</tr>
<tr>
<td>ST 987</td>
<td>0023000.341815.500000024</td>
<td>08-09-05 20:54</td>
<td>sales floor</td>
</tr>
<tr>
<td>ST 987</td>
<td>0023000.341815.500000024</td>
<td>08-10-01 1:10</td>
<td>sales floor</td>
</tr>
<tr>
<td>ST 987</td>
<td>0023000.341815.500000024</td>
<td>08-10-01 1:12</td>
<td>backroom</td>
</tr>
<tr>
<td>ST 987</td>
<td>0023000.341815.500000024</td>
<td>08-11-01 15:01</td>
<td>sales floor</td>
</tr>
<tr>
<td>ST 987</td>
<td>0023000.341815.500000024</td>
<td>08-11-01 15:47</td>
<td>sales floor</td>
</tr>
<tr>
<td>ST 987</td>
<td>0023000.341815.500000024</td>
<td>08-11-01 15:49</td>
<td>box crusher</td>
</tr>
</tbody>
</table>
The timeline in a supply chain

Process Mining & Flow Time Analysis

Example of a Supply Chain

What is new? e-Supply Chain
Wireless SCM

Manufacturing

Overseas Shipment

Ground Shipment (Warehouse/D.C.)

Customer

Cell Tower

RFID Retailer

RFID at Work

- Source tagging of goods/pallets/containers

- Port

- Movement of Goods

- Exchange of Data

- Warehouse/D.C.

- Ground Shipment (Truck or Train)

- Customer

- Wireless-enabled visibility at every link in the chain via phone/computer/PDA

- Cell Tower

- RFID Retailer

- Movement of Goods

- Exchange of Data

Lin and PSU Study Group (2006)

Challenges in RFID Enabled Supply Chain Management

Essence of event management

- Instance of real world
- Approximate
- Rich in assumptions
- Unmodeled dynamics
- Complex & dynamic
- Non linear

RFID At Work

- Tagged shipments inside of tagged containers

- Temperature condition, security and location data

RFID At Work

- Customs data for incoming shipments

- Equipment inventory control (containers, chassis, etc.)

RFID At Work

- Shipment status, location, estimated delivery time

- Vehicle contents

- Driver status/advice

- Routing/scheduling

- Inventory control

- Anti-theft

RFID At Work

- Dock/gate/yard management

- Vehicle status/location/routing

- Product/shipment notification

- Driver status

- Barcode scanning

- Wireless WMS network

Process Mining: Data

<table>
<thead>
<tr>
<th>Case Identifier</th>
<th>Task Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>case 1</td>
<td>task A</td>
</tr>
<tr>
<td>case 2</td>
<td>task A</td>
</tr>
<tr>
<td>case 3</td>
<td>task A</td>
</tr>
<tr>
<td>case 4</td>
<td>task B</td>
</tr>
<tr>
<td>case 5</td>
<td>task B</td>
</tr>
<tr>
<td>case 6</td>
<td>task C</td>
</tr>
<tr>
<td>case 7</td>
<td>task D</td>
</tr>
<tr>
<td>case 8</td>
<td>task E</td>
</tr>
<tr>
<td>case 9</td>
<td>task F</td>
</tr>
<tr>
<td>case 10</td>
<td>task G</td>
</tr>
<tr>
<td>case 11</td>
<td>task H</td>
</tr>
</tbody>
</table>

Table 1. A process log.
Process Mining: Model

![A process model corresponding to the process log.](image)

Mining Process Model: Mining Hidden Tasks

Even task A is removed from the log, it is clear that there has to be an AND-split if we assume tasks B and C to be in parallel. Similar for D.

Mining Process Model: Mining Duplicate Tasks

Questionable process model

Mining Process Model: Mining Non-free-choice Constructs

The choice between task D and task E is decided not only by their immediately precedent, but also some earlier choices (A and B). Such constructs are difficult to mine since the choice is non-local and the mining algorithm has to "remember" earlier events.
Mining Process Model-Mining loops

Mining loops can be difficult if the loops include many tasks (long span) and involve splits/joins.

Research Issues: Theoretical & Applications

- How to build up the model (flow-chart), for noise-free cases?
- How to build up the model (flow-chart), with noise?
- How to make use of the model?
- How to compare two models? What is the “optimal” model?
- Multiple Process Mining?

Standard language

- Standard language
- Unified Modeling Language (UML)
- Event-Driven Process Chain (EPC)
- Petri Nets

Petri Nets

- Places
  - Represents one or many objects. Each object is always in some state.
- Transitions
  - Represents one or many operations, which are only possible at specific states of objects and which change the state of specific objects
- Arcs
What kind of data?

- Process mining: event log

<table>
<thead>
<tr>
<th>Event</th>
<th>Case ID</th>
<th>Activity ID</th>
<th>Originator</th>
<th>Time stamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Case 1</td>
<td>Activity A</td>
<td>John</td>
<td>9-3-2004: 15.01</td>
</tr>
<tr>
<td>2</td>
<td>Case 2</td>
<td>Activity A</td>
<td>John</td>
<td>9-3-2004: 15.12</td>
</tr>
<tr>
<td>3</td>
<td>Case 3</td>
<td>Activity A</td>
<td>Sue</td>
<td>9-3-2004: 16.03</td>
</tr>
<tr>
<td>4</td>
<td>Case 3</td>
<td>Activity B</td>
<td>Carol</td>
<td>9-3-2004: 16.07</td>
</tr>
<tr>
<td>5</td>
<td>Case 1</td>
<td>Activity B</td>
<td>Mike</td>
<td>9-3-2004: 18.23</td>
</tr>
<tr>
<td>6</td>
<td>Case 1</td>
<td>Activity C</td>
<td>John</td>
<td>16-3-2004: 9.23</td>
</tr>
</tbody>
</table>

The resulting model

Literature review: Algorithms

- Software (most based on \( \alpha \)-algorithm)
  - Emit (Van Aalst et. al., 2002)
  - Little Thumb (Van Aalst et. al., 2004)
  - InWoLvE (Herbst et. al., 2001)
  - Process Miner (Schimm et. al., 2002)

Difficulties and Challenges

- Mining hidden activities
- Mining duplicate activities
- Mining non-free-choice constructs
- Mining loop
- Using time
- Visualizing results
- Mining different perspectives
- Dealing with noise
- Dealing with incompleteness
- Delta analysis

Van Aalst et. al., 2004
**α-algorithm**

- If a and b are **sequent** iff \( a > b \) (a is the input of b)
- If a and b are **alternative** iff \( a \triangleright b \) and \( b \triangleright a \)
  (a can’t be the input of b and b can’t be the input of a)
- If a and b are **concurrency** iff \( a > b \) and \( b > a \)
  (a can be the input of b and b can be the input of a)

---

### Optimal Model: Illustrative Example

**Event logs \( W:\)**

\{ABCD\}, \{ACBD\}, \{AED\}

---

**Model-1**

---

**Model-2**

---

---

**Event logs \( W:\)**

\{ABCD\}, \{ACBD\}, \{AED\}

---

---

---

---

---
The Process of Building a process model:

**Example: Event Logs**

Event logs

<table>
<thead>
<tr>
<th>Case 1: SABCDEF</th>
<th>Case 2: SACBDEF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 3: SAJCDEF</td>
<td>Case 4: SACJDEF</td>
</tr>
<tr>
<td>Case 5: SABCD</td>
<td>Case 6: SACBDEF</td>
</tr>
<tr>
<td>Case 7: SAJCDEF</td>
<td>Case 8: SACJDEF</td>
</tr>
<tr>
<td>Case 9: SABCD</td>
<td>Case 10: SABCDF</td>
</tr>
<tr>
<td>Case 11: SABCD</td>
<td>Case 12: SAJCDF</td>
</tr>
</tbody>
</table>

**Step1 : Build an n x n From-To Table**

<table>
<thead>
<tr>
<th>From/To</th>
<th>S</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>H</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>I</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>J</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Step 2: Find out the potential concurrent activities from From-To Table

\[
\begin{array}{l}
\{B,C\}, \{C, J\}, \{E, F\} \\
\end{array}
\]

Are they
Concurrence?!
Alternative?!
Loop?!

Concurrence → Both-And (not in order)
Alternative → Either-Or
Loop → Trouble*

Step 3: Modify the From-To Table

Concurrence

Simple Loop
Step 4: Count the number of the place

{B,C,J}

Concurrence!? Alternative?!
1. {B,C} are concurrence
2. {C,J} are concurrence
3. B and J didn’t happen together in any trace

Step 5: build an n×s Activity-Place Table

<table>
<thead>
<tr>
<th>Action</th>
<th>S0</th>
<th>S1</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
<th>S5</th>
<th>S6</th>
<th>S7</th>
<th>S8</th>
<th>S9</th>
<th>S10</th>
<th>S11</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>H</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>I</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>J</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Step 6: Build an $s \times n$ Place-Activity Table

<table>
<thead>
<tr>
<th>From/to</th>
<th>S</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>S0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S9</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>S10</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>S11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Step 7: we now can build up the process model, from Activity-Place Table and Place-Activity Table,

Contribution

Given events log (data), we are able to construct a process model.
But,
Not sure whether this is the optimal one!

Comparisons with Existing Algorithms

<table>
<thead>
<tr>
<th></th>
<th>EMit</th>
<th>Little Thumb</th>
<th>InWoLvE</th>
<th>Process Miner</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>Graph</td>
<td>Graph</td>
<td>Graph</td>
<td>Block</td>
<td>Graph</td>
</tr>
<tr>
<td>Time</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Basic parallelism</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Non-free choice</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Basic loops</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Arbitrary loops</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Hidden tasks</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Duplicate activities</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Noise</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Next</td>
</tr>
</tbody>
</table>
Summary: Process Mining

- What is available?
- What need to be done?
- Some initial results on
  - Simple Loop
  - Non-Free Choice
  - Over-General
  - Others

Future Work

- How to build a minimum process model?
- How to build the suitable model, when the event logs data were contaminated with noise?
- How to accommodate the time (and other) component?
- When the event logs come from two (mixed) processes, how to build the two process models?

Reference

- Lijie Wen · Wil M. P. van der Aalst · Jianmin Wang · Jiaquang Sun, Mining process models with non-free-choice constructs, Data Min Knowl Disc (2007) 15:145-180
- References Therein