The “Network Part Program” approach in view of co-evolving products, processes and systems

Outline

Outline of the presentation:
- Problem statement and motivation
- The Process in view of Integration and Evolution
- Real cases analysis
- Flexibility of the Network Part Program
- Methodologies and tools
- Conclusions and future developments
**Problem Statement**

The process is commonly considered as the set of operations that allow to transform raw parts in finished products.

The process is therefore the NC code that is given to the machine for executing the task.

---

**Problem Statement: integration**

However, considering the INTEGRATION among products processes and systems, the NC code does not contain only process information, thus is an hybrid entity.

Tool management info

Production System

Process parameters info
**Problem Statement: integration**

Consider the following examples...

1. The pallet configuration is embedded in the NC code.
2. The tool management rules are embedded in the NC code.
3. The rapid movements of the turret are coded in the NC code.

   **HOWEVER: these are all system management decisions**

Moreover...

1. The cutting parameters and the material removal strategies are pure process information which is mixed and lost in the NC code.

**Problem Statement: integration**

Is the integration among process and system currently structured in the NC codes correctly formalized?

**EXAMPLE 1: on-line parameters adjustment**

1. The process designer sets the cutting speed and the cutting depth.

2. The operator realizes on-line that the machine vibration amplitude is too high.

3. The operator enters the NC code and adjusts the cutting parameters.
Problem Statement: integration

Is the integration among process and system currently structured in the NC codes correctly formalized?

EXAMPLE 1: on-line parameters adjustment

Integration problems are manually addressed by an experienced person.

1. Who should ideally manage the adjustment procedure?
2. Where should the modified information been stored for re-use and learning scopes?
3. Who manages the information generated by the operator?
4. Should the adjustment loop be automatic?

EXAMPLE 2: unexpected disturbances and failures

Integration problems are addressed by NC code analysis

1. Where should the NC code restarted from?
2. Should the traceability be a system level information?
3. The failure is corrected
4. The NC Code can be restored only after having identified the last executed process operation
Problem Statement: evolution

In the scientific literature as well as in the industrial practice considerable effort has been spent toward the design of highly evolvable production systems.

Production System

Problem Statement: evolution

However the process still seems to be the rigid node in the achievement of co-evolving products-processes and production systems.

Production System

Is the process really the bottleneck for co-evolution? Consider the following real cases...
Co-evolution case in FMSs

Co-evolution case in Reconfigurable Transfer Lines
**Problem Statement**

Currently the process is the bottleneck in the co-evolution of products, processes and systems since high effort is required to adapt it to new product/system configurations.

Is it possible to provide some flexibility to the process to improve its adaptability and support the co-evolution framework?

Going deeper in details on the process side...
Currently adopted NC Code

- Impossible to change the order of execution of operations.
- Impossible to optimize the operation sequence in real time at shop floor level
- Impossible to split a specific part program on different machines
- Technological and managerial info is embedded in the same code.

Outline

Problem Statement

Real Case Analysis

Network Part Program

Methodologies and tools

Conclusions and developments
Main Issues of the proposed approach

Is it possible to propose a new adaptive process structure?

Main ideas:
- To clearly separate system and process information from the part program
- To provide a solid product, process and production system information formalization, considering their integration
- To provide some modularity and flexibility to the pure process structure in order to support the co-evolution idea

The Network Part Program concept

[Diagram showing network part program concept with nodes labeled as Program Start, Part_1, Part_2, etc.]
1. Possibility to change the execution order of the operations within a process plan (under technological constraints)
2. Possibility of optimizing and re-optimizing the operation sequence also after product modification
3. Possibility to split a specific part program on machines
4. Easiness in the recovering of an interrupted sequence
5. Possibility of performing operations in unmanned shift
Flexible Data Structure

Effects on product-process interaction

**STEP NC based separation of product and process data into two different objects**

- Product information ➔ Machining feature ➔ Machining workingstep
- Process information ➔ Machining operation

**Important consequences in view of co-evolution:**

- Changes in the product have eco only on “features”
- Changes in the process have eco only on “operations”
- A change in the workplan regards only one or a few workingsteps, without needs of changing all the part program
- Changes in precedence constraints mean only a different way of connecting two workingsteps: do not mean to rebuild all the part program.
**Feature recognition**

- **Problem Statement**

- **Conclusions and developments**

---

**Methodologies & Tools: Simulation**

Machining Simulation (VERICUT) to:
- Verify the process against collisions
- Quickly evaluate the impact of product modifications on the process

MORE RELEVANT IF THE PROCESS IS FREQUENTLY MODIFIED
<table>
<thead>
<tr>
<th>Product</th>
<th>Process</th>
<th>Production system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed</td>
<td>Flexibility of the process allows to resume the production of a pallet after the failure</td>
<td>Disturbances and failures affect the system operations</td>
</tr>
<tr>
<td>Fixed</td>
<td>The modularity of the process allows to change the order of execution of operations</td>
<td>Tool wear phenomenon causes the need for a tool change</td>
</tr>
</tbody>
</table>

---

Examples highlighting potential benefits evolution

<table>
<thead>
<tr>
<th>Product</th>
<th>Process</th>
<th>Production system</th>
</tr>
</thead>
<tbody>
<tr>
<td>One or more features are modified</td>
<td>Only the operations linked to the new features are modified</td>
<td>Depending on its ability to absorb evolutions is modified or not</td>
</tr>
<tr>
<td>The production volume or mix is changed</td>
<td>Flexibility of the process allows to process mixed part pallets</td>
<td>Better utilization of the system resources</td>
</tr>
<tr>
<td>The product remains unchanged</td>
<td>The process rapidly adapts by setting the new parameters</td>
<td>The system changes</td>
</tr>
</tbody>
</table>
Conclusions and future development

The co-evolution of products, processes and production systems can be potentially supported by NetPP approach:

- **Real-time management** of machining operation execution order
- **Shift of decisional aspects** to a level higher than the one of the CNC, in order to **increase the traceability** of the modifications in the part-programs
- **Split a specific part program on different machines**
- **Recovery of an interrupted machining sequence** without additional costs

**Relevant consequences:**

- **More flexibility** in the process may help in **better exploiting the system flexibility**. Possibly, the required machine flexibility may be reduced (FFMS).
- **Indeed, software flexibility** cheaper than hardware flexibility

---

Thank you

Tullio TOLIO - Politecnico di Milano  
Department of Mechanical Engineering