CIMR in University Politehnica of Bucharest – Working Together with Companies

Professor Theodor Borangiu

University Politehnica of Bucharest, Department of Automation & Applied Informatics
Director of CIMR Centre of Research in Robotics and CIM, cimr@cimr.pub.ro

CIMR Organization. Activities

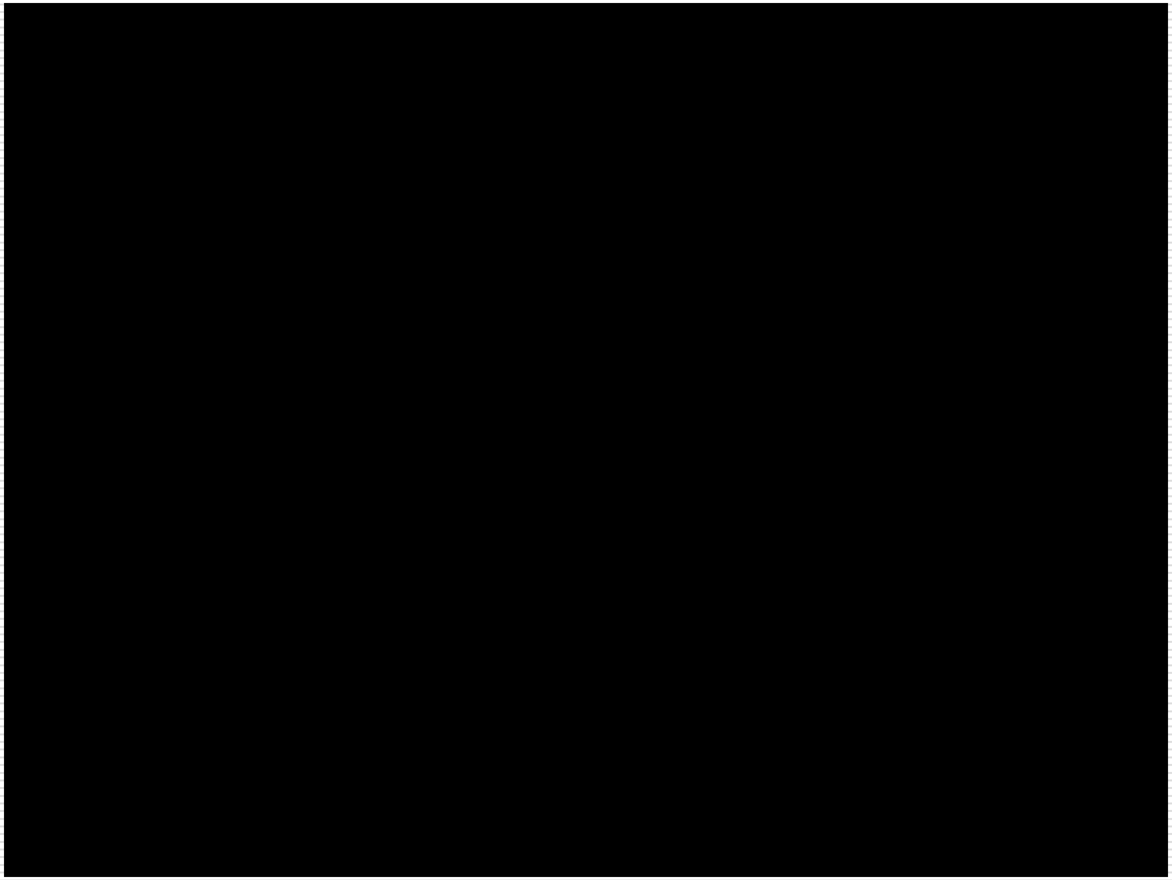
- **Founded in 1994** within the Faculty of Automatic Control & Computers (AC) of the Polytechnic University of Bucharest (PUB)
- **Activities**: fundamental and applied research, technology transfer, training
- **Certified as Excellency Research Centre** in 2003 by the Ministry of Education and Research and in 2008 by the National Agency for Scientific Research (ANCS)
- **32 academic researchers** – all PhD from the Automation & Industrial Informatics Dept. in PUB; 8 professors, 4 PhD directors
- **18 PhD students** working full time
- **3 Research Laboratories**:
  - Advanced Process Control and Communication Technologies
  - Business-Driven SW Development and Application Lifecycle Management
  - Robotics and Artificial Intelligence in Manufacturing Control
- **Partnership with industry**: ITC (IBM, Microsoft, AlsysData), Control (Siemens, ASTI, Festo, Motorola, NI), Robotics & Manufacturing (Adept, ABB, Kuka, East Electric)
- **Contact**: 313, Spl. Independentei, sector 6, Bucharest 060032, Romania
  Tel: +40 21 402 93 14, Fax: +40 21 317 09 12, E-mail: cimr@cimr.pub.ro

Research Programs, Strategies

- Advanced robot motion control and integration in cooperative manufacturing
- In line shape generation from robot-driven 3D scan patterns
- Guidance Vision for Robots (GVR) and Automated Visual Inspection (AVI)
- Distributed, semi-heterarchical holonic manufacturing control (distributed AI with MAS)
- Product-driven automation with intelligent, embedded devices
- SOA for integrated enterprise management and control [open standards]. Cloud [open education]

Advanced Robot Motion Control & Integration in Cooperative Manufacturing

Robot tended CNC milling machines



- **Company**: UPETROLAM Bucharest
- Complex trajectory generation on 4-axis CNC mills
- Procedural robot motion command
- CAD for robot – CNC applications
- Development for CAE

Advanced Robot Motion Control & Integration in Cooperative Manufacturing

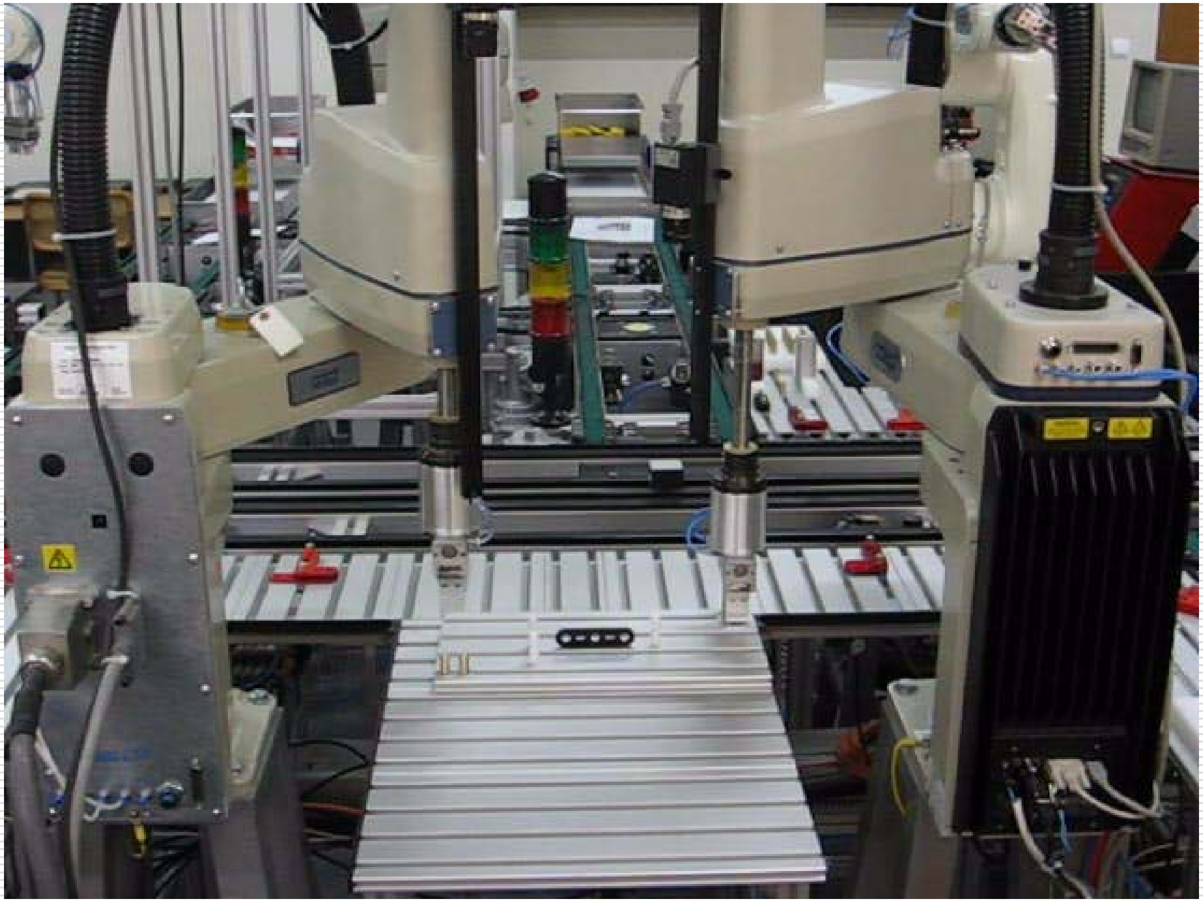
Shared robot workspace

- **Company**: East Electric
- Management of shared, multi-access factory workplaces
- Collision avoidance
- Task synchronization
- Multitasking operating mode
- Multi-robot CAD applications

Advanced Robot Motion Control & Integration in Cooperative Manufacturing

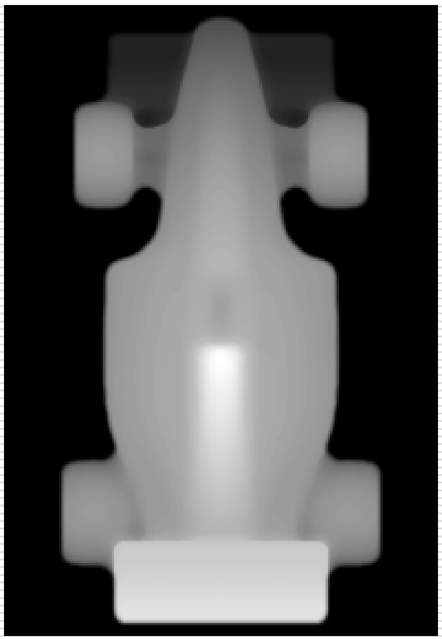
Robot cooperation

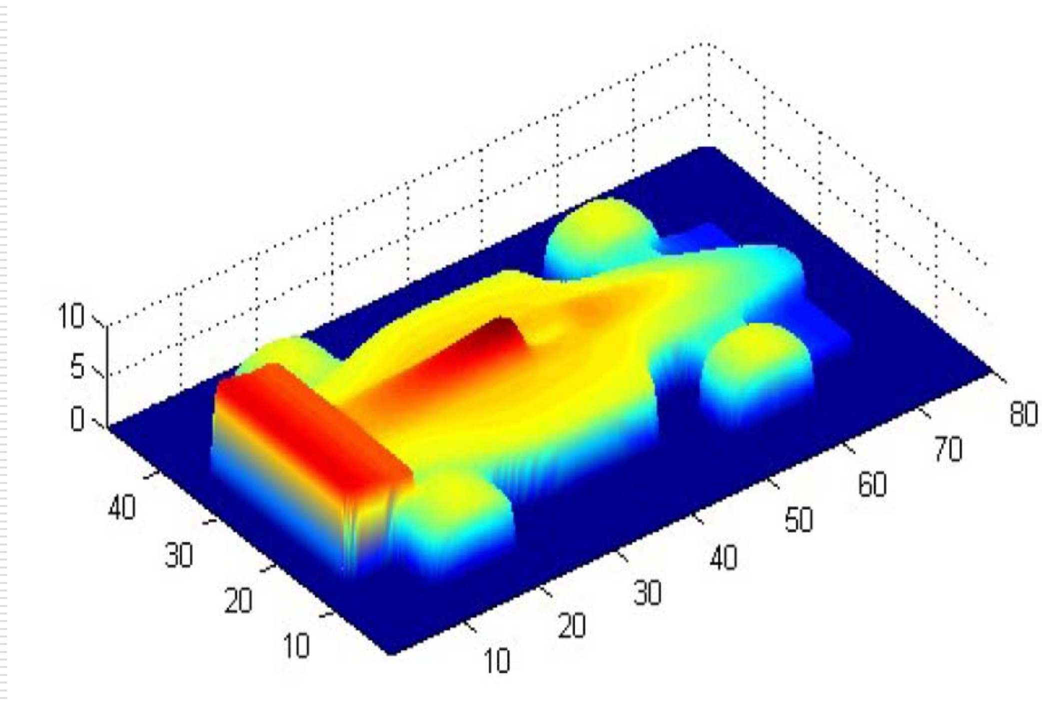
- **Company**: COMET Tecuci
- Part- and task-driven robot operating modes
- Master-slave control mode
- Hybrid position-force control
- Multitasking operating mode
- Dynamic interaction modes



In line complex shape generation from robot-driven 3D scan patterns

Height Map images





Generating CNC toolpaths from grey level images

In line complex shape generation from robot-driven 3D scan patterns

Obtaining height map images

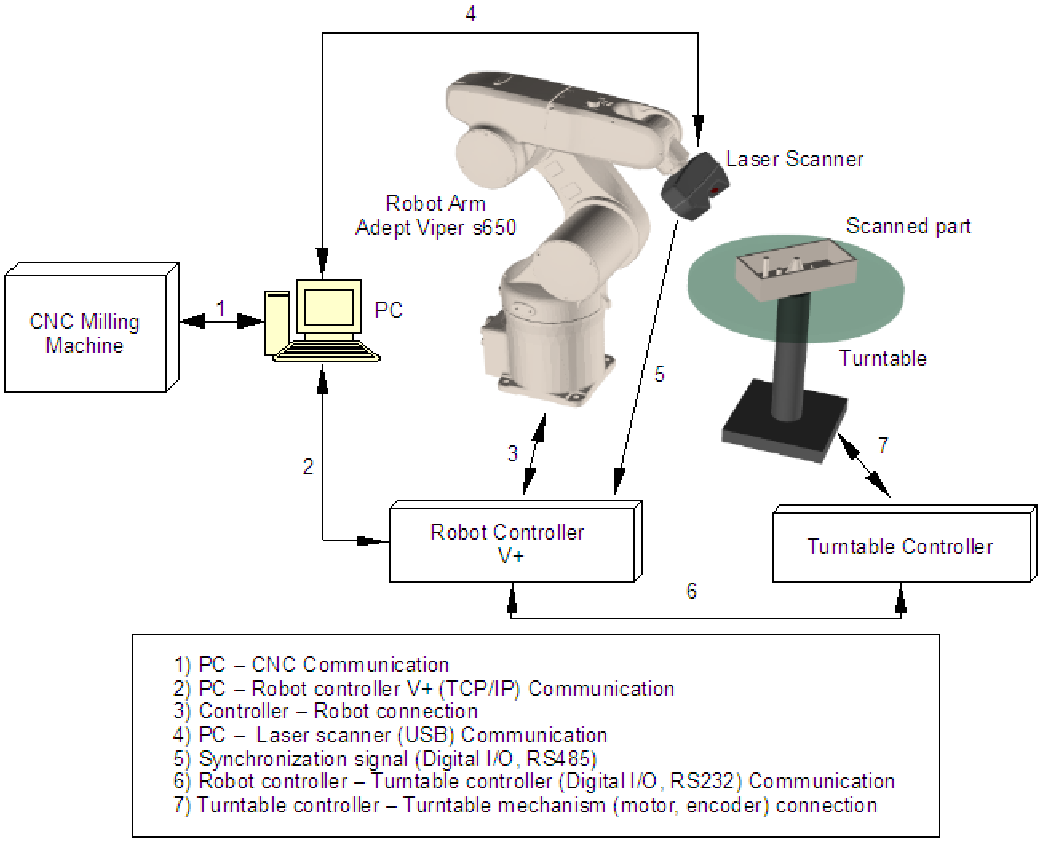
The 3D surface of a model to be machined is scanned with a laser range finder device:

- A vertical stripe of laser light is moved across the model object surface, and captured by a video camera. Along each horizontal scan line of the video frame, the *brightest* spot is taken to be the point at which the laser stripe "hits“ the surface (detection at sub-pixel resolution).
- The relative positions of the laser and the video camera are used to find the 3D coordinates of the brightest spot by triangulation.
- The *x*-coordinate of each point in the output depth image is determined by the position of the laser stripe for a particular video frame
- The *y*-coordinate corresponds to a raster line in the video frame,
- The depth value is computed from the brightness peak detected along the raster line in the video frame

In line complex shape generation from robot-driven 3D scan patterns

Arm-mounted laser range finder and robot motion patterns

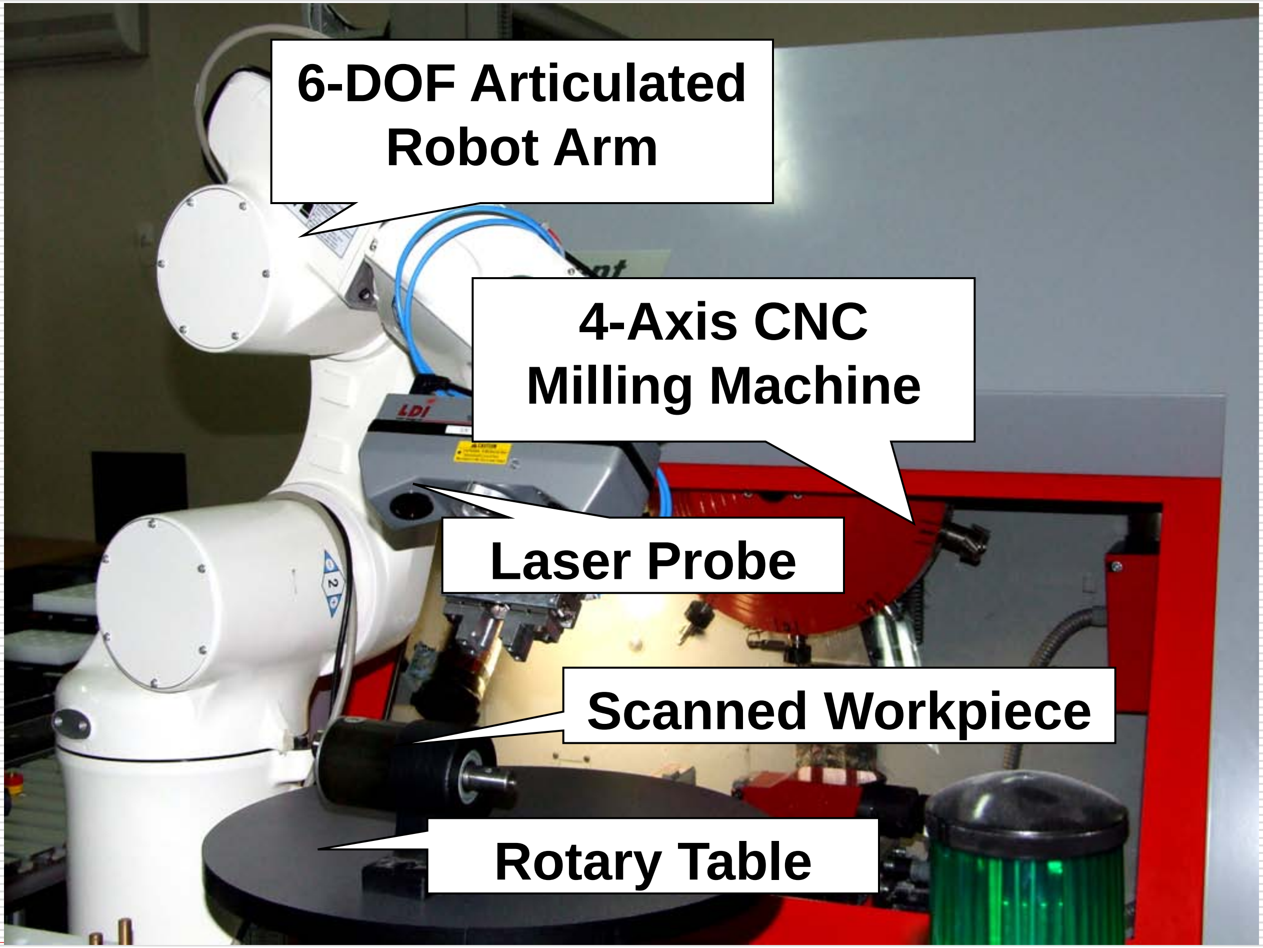
- Dual laser probe measure distances from 70 to 250 millimetres, with an accuracy of 30 µm
- The laser probe is arm-mounted on a 6-d.o.f. robot
- The scanning paths are computed in *real-time* by the robot controller from *predefined* or adaptive motion patterns
- The range finder device generates depth map-type information describing the object's surface, *synchronously* with the motion of the laser scanner probe
- Robot working envelope: spherical, 650 mm radius; resolution of rotary table: 0.03 deg
- Hardware controllers: robot-, rotary table-, CNC machine; PC integrated



In line complex shape generation from robot-driven 3D scan patterns

Multiple-axis scanning patterns

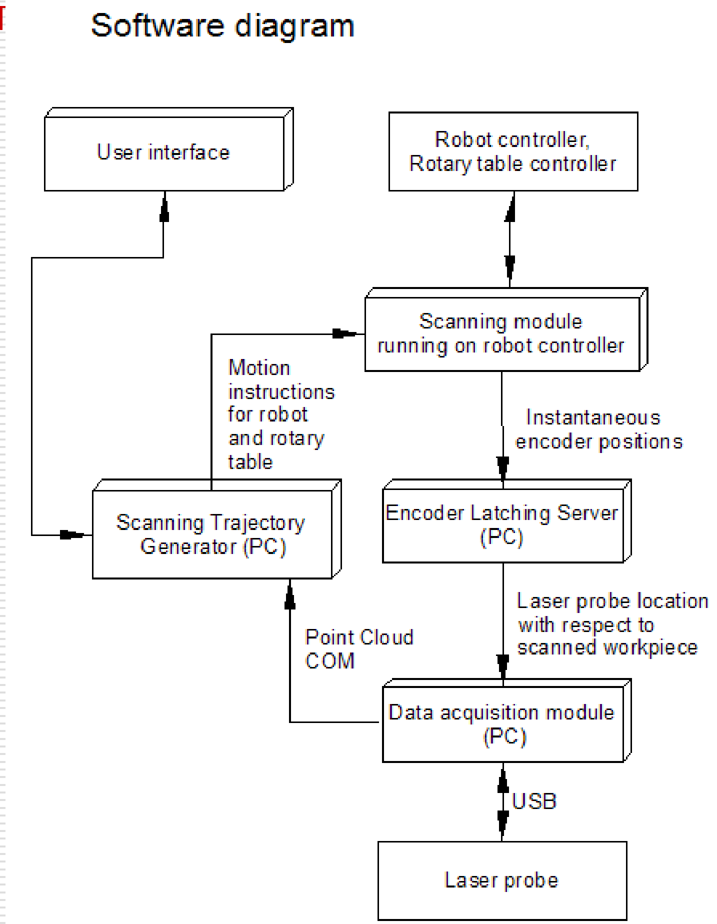
- Multiple object views
- Anthropomorphic 6 d.o.f. robot motion patterns, object-oriented
- Robot – rotary table synchronization
- Path optimization (power consumption, avoiding singularities, smoothness)



In line complex shape generation from robot-driven 3D scan patterns

Arm-mounted laser range finder and robot motion patterns

- A linear laser module projects a line of red light on the scanned object
- The line is detected by two cameras located on the laser probe
- The object's contour along the laser line is measured using Cartesian coordinates
- The robot arm and rotary table form a **7-DOF kinematic chain** that can move the laser probe to a precise location relative to the work piece, according to the *scanning trajectory*; from this location, a *measurement* is made
- The measured points *lie* in the laser plane
- Since the position of the laser probe (and therefore the laser plane) is known *relative to the work piece* at each measurement point, the measured points can be transformed into a *unique* reference frame, which is attached to the work piece
- In this way, a point cloud is obtained, which is the *3D representation of the scanned object*

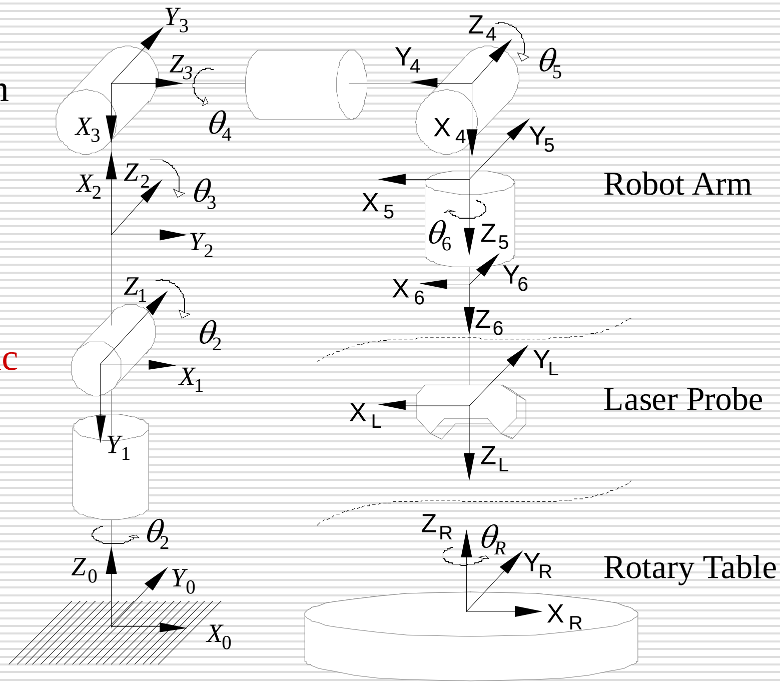


Software diagram of the laser scanning system

In line complex shape generation from robot-driven 3D scan patterns

Arm-mounted laser range finder and robot motion patterns

- **Communication**
  - **USB interface** connects laser probe to the PC
  - **Data acquisition module** (DAM) takes measurements from laser probe
  - Measurements are aligned in the *workpiece's frame*; the instantaneous pose of laser probe relative to rotary table is computed by the **Encoder latching server** (ELS) module
  - ELS receives the *encoder readings* from robot- and rotary table controllers (TCP/IP connection) and uses the kinematic model of the system
  - The location of the laser probe is sent to the DAM in **X-Y-Z-yaw-pitch-roll** format
- **Synchronization**
  - The robot- and rotary table controllers latch their current position using an external trigger signal sent by the laser probe every time a measurement is made
  - Scanning rate: 50-150 frames / sec (frame = line of scanned points; robot / table position latching: less than 1 msec



Denavit-Hartenberg reference frame assignment

In line complex shape generation from robot-driven 3D scan patterns

❑ Arm-mounted laser range finder and robot motion patterns

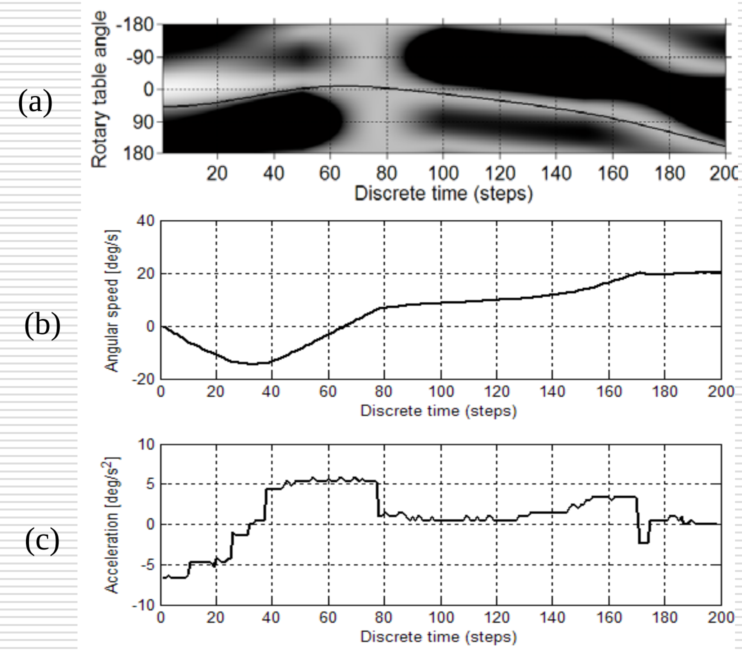
- **Path planning algorithms (example 3)**
  - **(3) Dijkstra-like algorithm**
    - Finds a minimal cost path through a graph with positive weights (*costs*) on its edges, from a given node to all other notes reachable from it
    - A 3rd dimension is added to the configuration space M_G: the angular velocity ω
    - A node in the graph will be expressed as a vector of discrete coordinates (i, j, k) , which maps to its continuous counterpart $(\theta_i, t_j, \omega_k)$
    - From node $(\theta_i, t_j, \omega_k)$, one may *advance* using the acceleration and reach the node corresponding to $\left(\theta_i + \omega_k \Delta t + a\frac{(\Delta t)^2}{2},\ t_j + \Delta t,\ \omega_k + a\Delta t\right)$
    - The **cost of a node** is:

$$C_{node} = k_\omega (\omega_k)^2 + 1 - f_R\left(\theta_i, T_L^{(j)}\right)$$

    - The **cost of an edge** is:

$$C_{edge} = k_a a^2 + k_\omega (\omega_k + a\Delta t)^2$$

– The planned path does not touch obstacle edges, the motion is smooth and the acceleration rates are much lower than those obtained with the Local Maxima (2)



Path computed by Dijkstra algorithm for a grayscale M_G. The path does not touch obstacles. (a) Position; (b) Angular speed; (c) Angular acceleration

In line complex shape generation from robot-driven 3D scan patterns

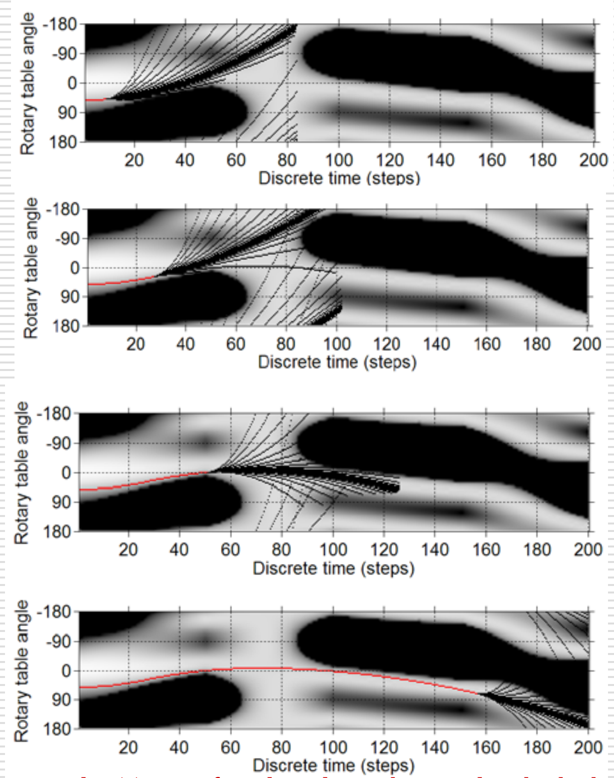
❑ Arm-mounted laser range finder and robot motion patterns

- **Path planning algorithms (example 4)**
  - **(4) "Ray-Shooting" heuristic algorithm**
    - Provides a better solution than the local maxima heuristic, with real time planning perspective
    - Computes a smooth motion, comparable to the one obtained with Dijkstra algorithm
    - At every time step *k*, the algorithm *looks ahead* *p* future time steps, that is, from *k*+1 through *k*+*p*. Over this range, performs a motion with constant angular acceleration, for planned path smoothness
    - A finite set of acceleration values $a_j, \; j=\overline{1,n_a}$, and for every acceleration a_j , a possible path is evaluated, starting from the current state and spanning on the following *p* time steps.
    - From the set of paths, the best one is chosen, for an acceleration $a_{j.max}$, and the motion from time *k* through time *k*+1 is done with this acceleration

(a)

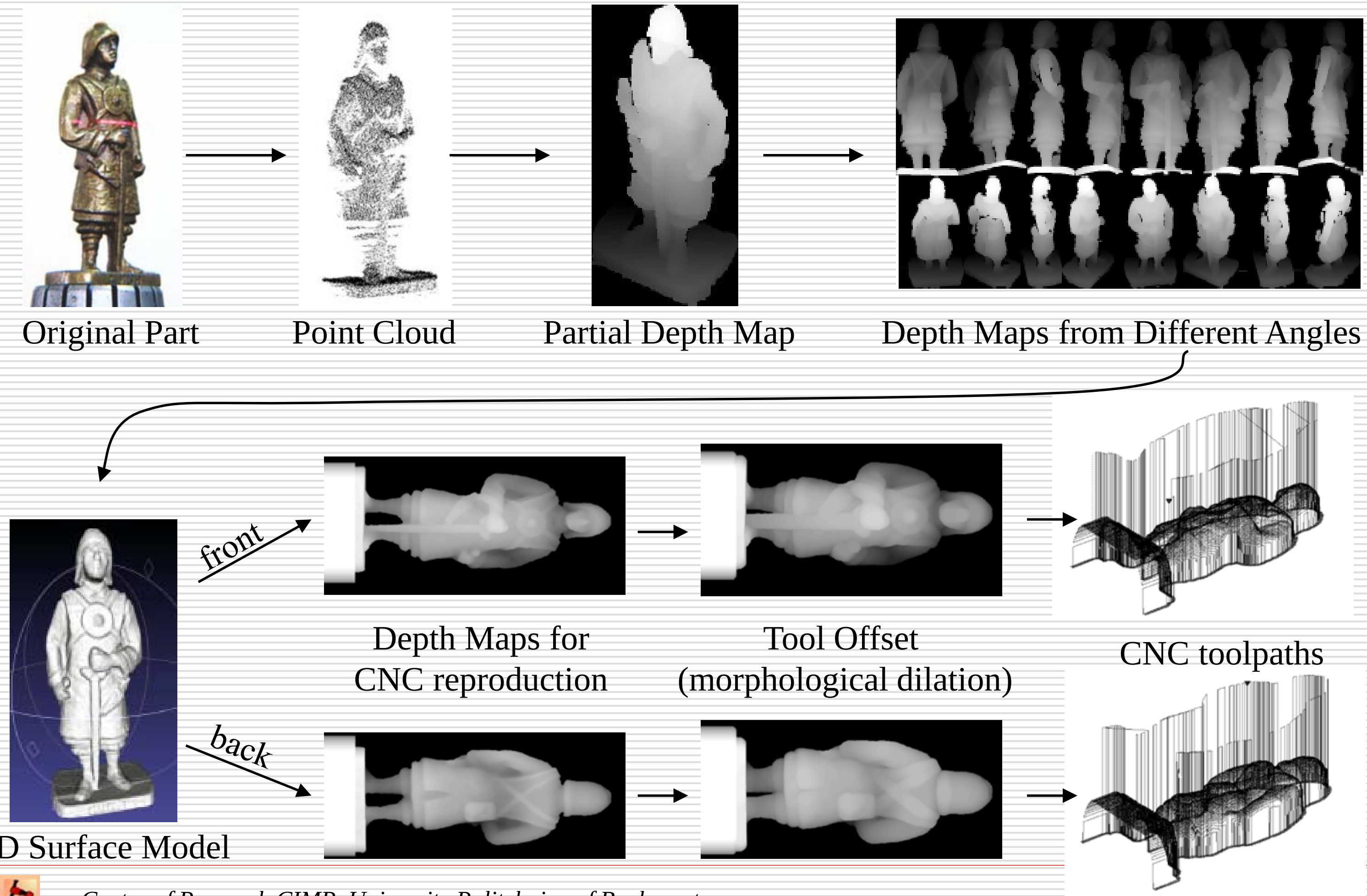
(b)

(c)



Ray Shooting example: (a) Rays found a solution by avoiding both obstacles on the left side; (b) One ray starts seeing an alternative path: avoiding the 2nd obstacle on the right side; (c) The 2nd alternative has lower cost than the 1st one, thus it is chosen; (d) Search almost finished. Rays wrap around on *Y* axis.

Complete Data Processing Chain for Tool Path Generation

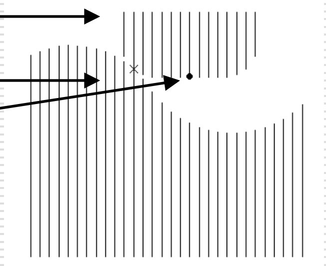


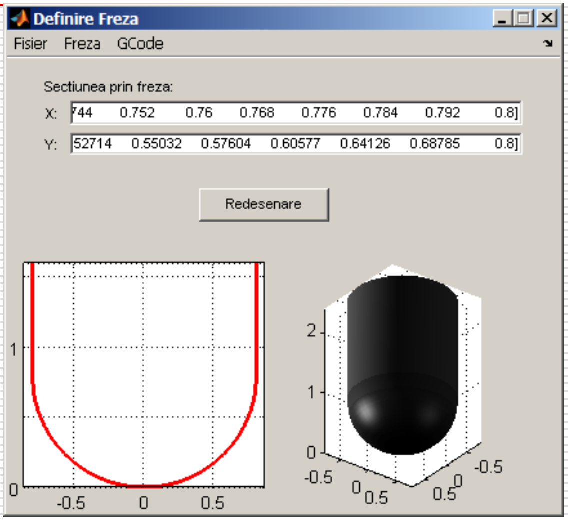
Computing 3D Tool Compensation with Image Processing Operations

Milling Tool

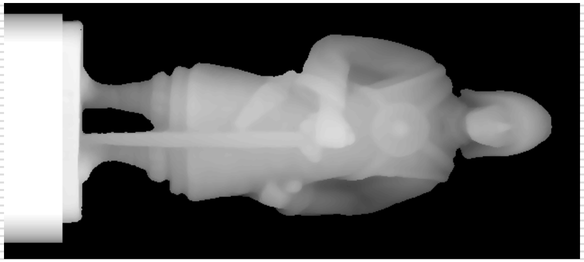
Part Surface

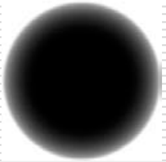
Milling Tool Tip





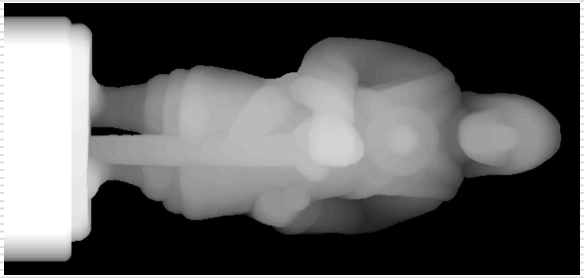
Part surface (depth map)





Milling Tool Shape (depth map)

Morphological dilation

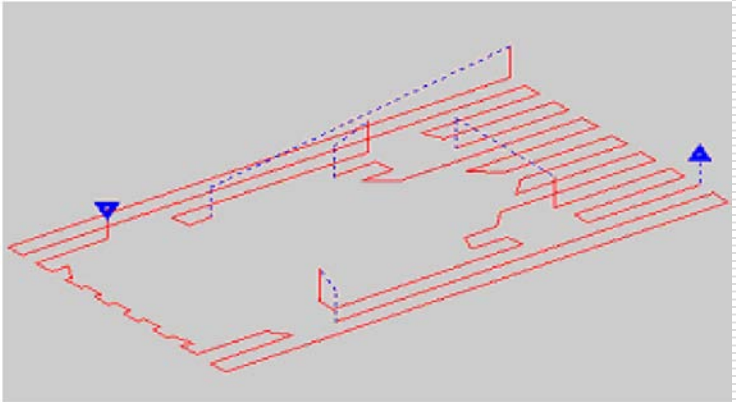


Milling Surface

The surface where the tip of the cutting tool can move in order to be always tangent to the part surface

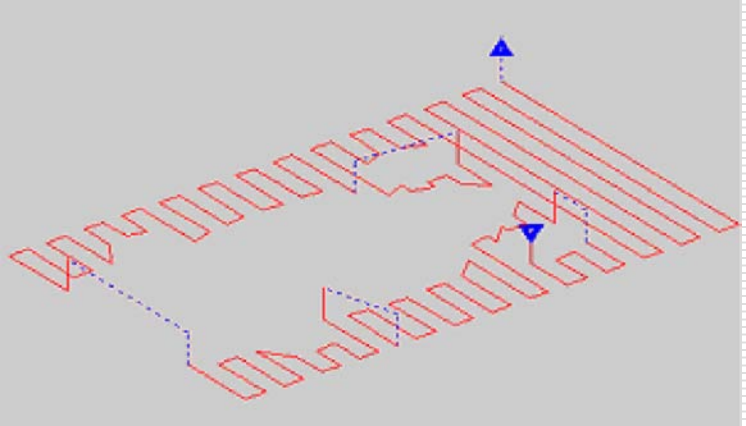
Generating CNC toolpaths from grey level image and pattern robot motion

Generating **Roughing** Toolpaths



- Each roughing stage is performed *at constant Z level*

- At a given Z level, selecting the region where the cutter should clean up is an *image thresholding* operation

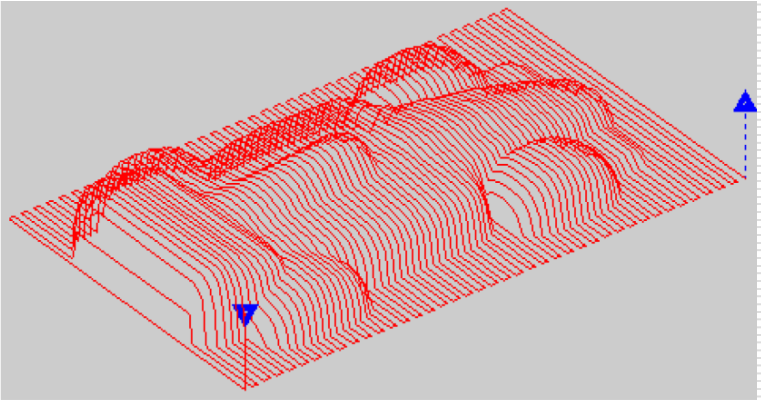


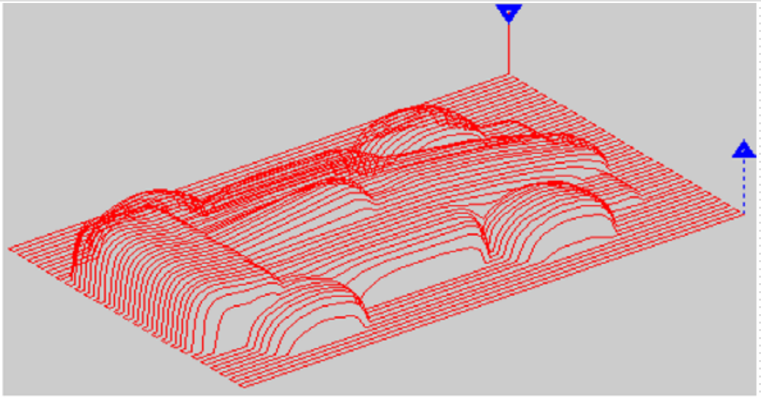
- For flat endmill cutters 2D offset compensation was used

Generating CNC toolpaths from grey level image and pattern robot motion

Generating **Finishing** Toolpaths – 1st method

- In XY plane, the tool moves parallel with one axis or direction
- The tool moves on the "safe surface"
- There is no need to compute the whole "safe surface"

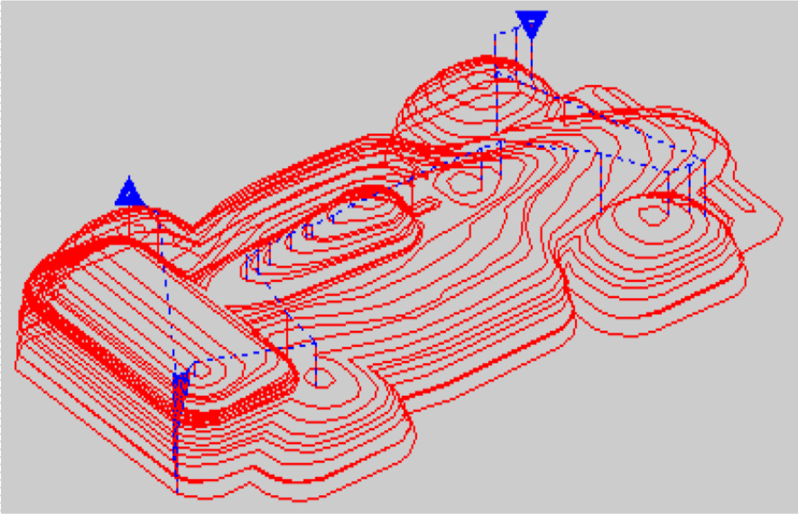




Generating CNC toolpaths from grey level image processing

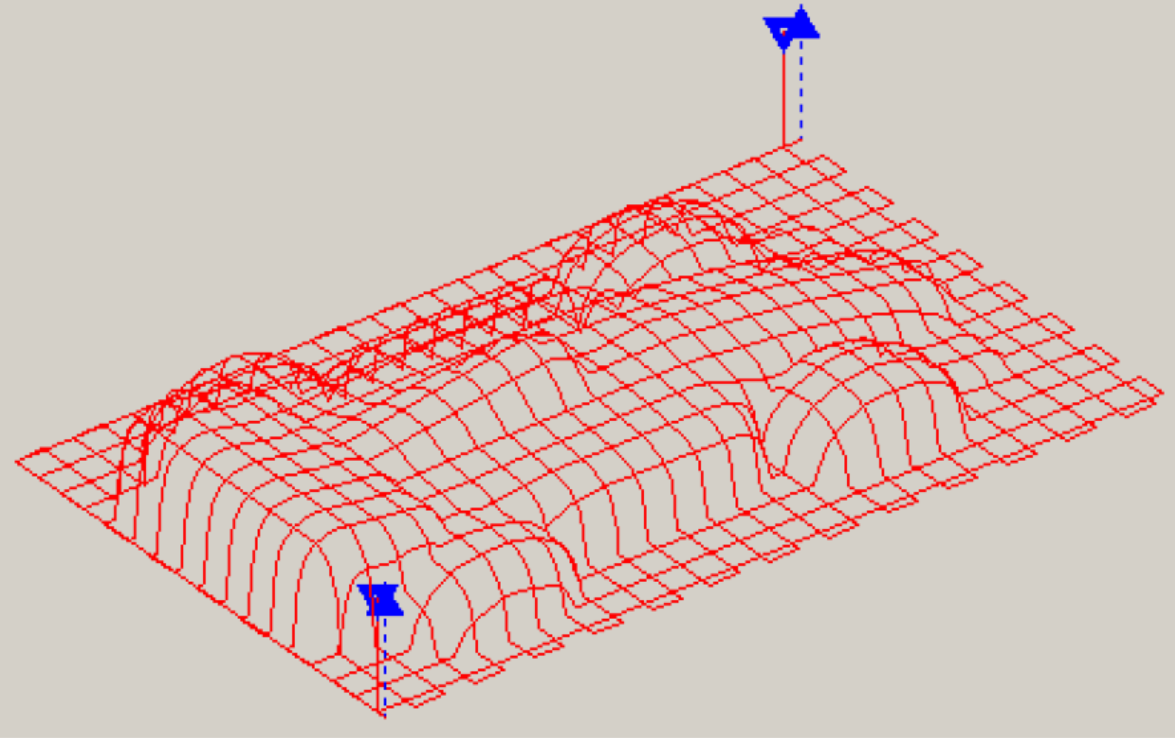
Generating Finishing Toolpaths – 2nd method

- Tool paths are at constant Z levels
- Because of the tool shape, one cannot use 2D compensation any more
- The whole surface needs to be computed!



Generating CNC toolpaths from grey level image processing

Finishing Toolpaths - Combined



Sample 3D Scanning Movie

Company:

ICECON Bucharest

Guidance Vision for Robots & Automated Visual Inspection

Intelligent Feeding Systems

- **Company**: East Electric
- Configurable material presentation modes
- Dual, flexible feeding system
- Robotized part feeding from unstructured storage
- High speed, real time machine vision qualifies parts
- Robot motion guidance through image processing

Guidance Vision for Robots

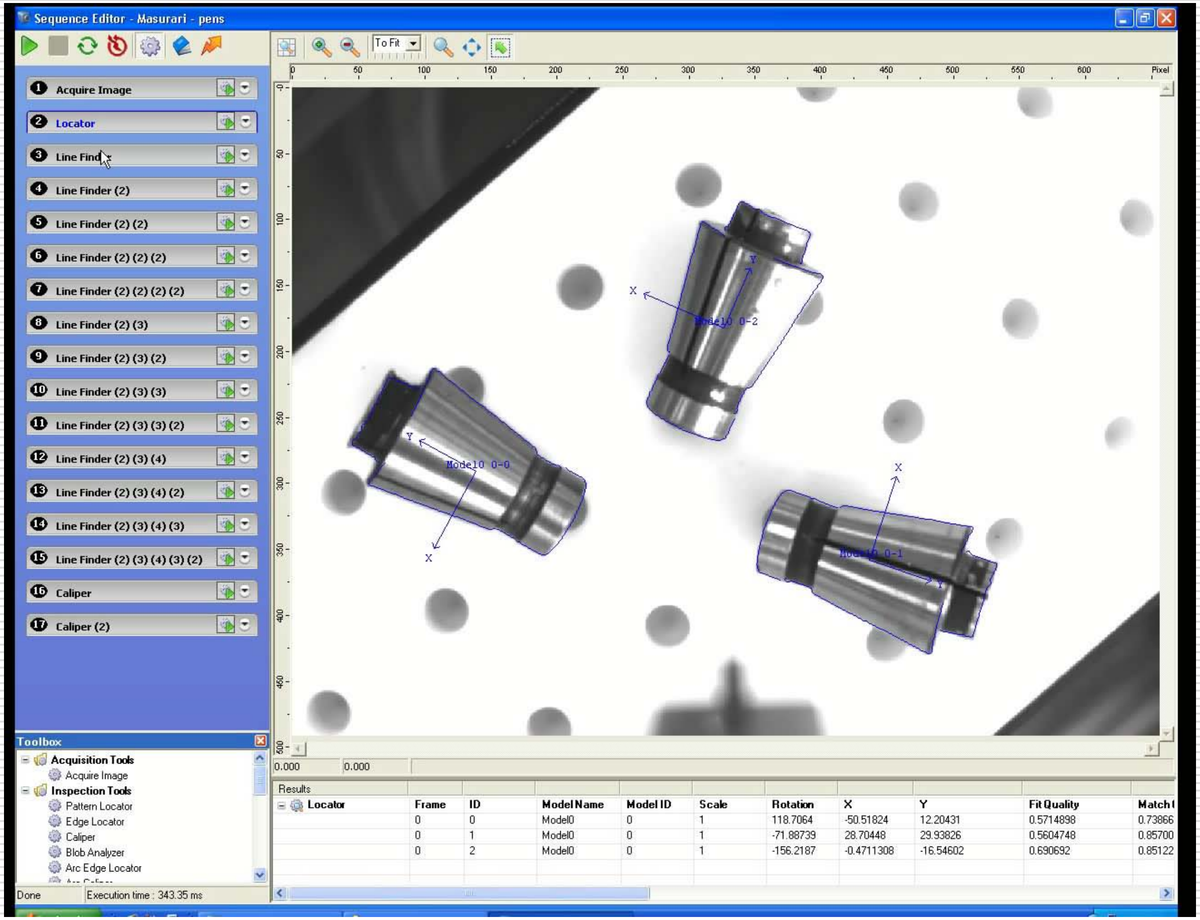
Motion tracking with dynamic visual feedback

- **Company**: East Electric
- Visual servoing
- Dynamic Look & Move
- Robot-scene & robot-object modelling
- Synchronizing pick-and-place tasks with moving material flow
- Application: Visual management of factory transportation system

Automated Visual Inspection

Part Geometry Analysis (1)

- Robust recognition
- Part-dedicated lighting system
- Construction of virtual cameras
- Software measurement tools:
  - Finders
  - Rulers
  - AOI
  - Calipers
- Aggregate quality control

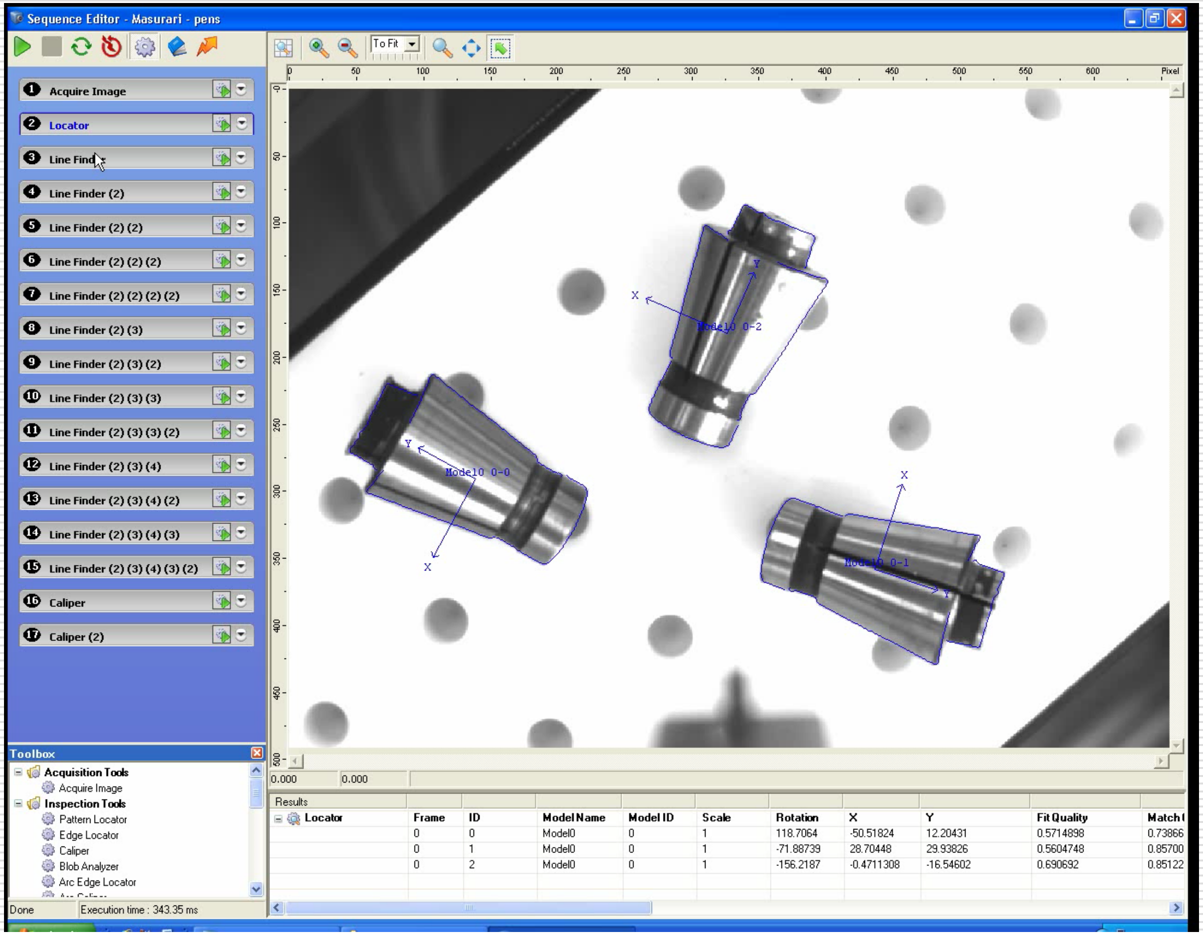


Automated Visual Inspection

Measurement demo (1) using:

- Line Finders
- Linear Rulers

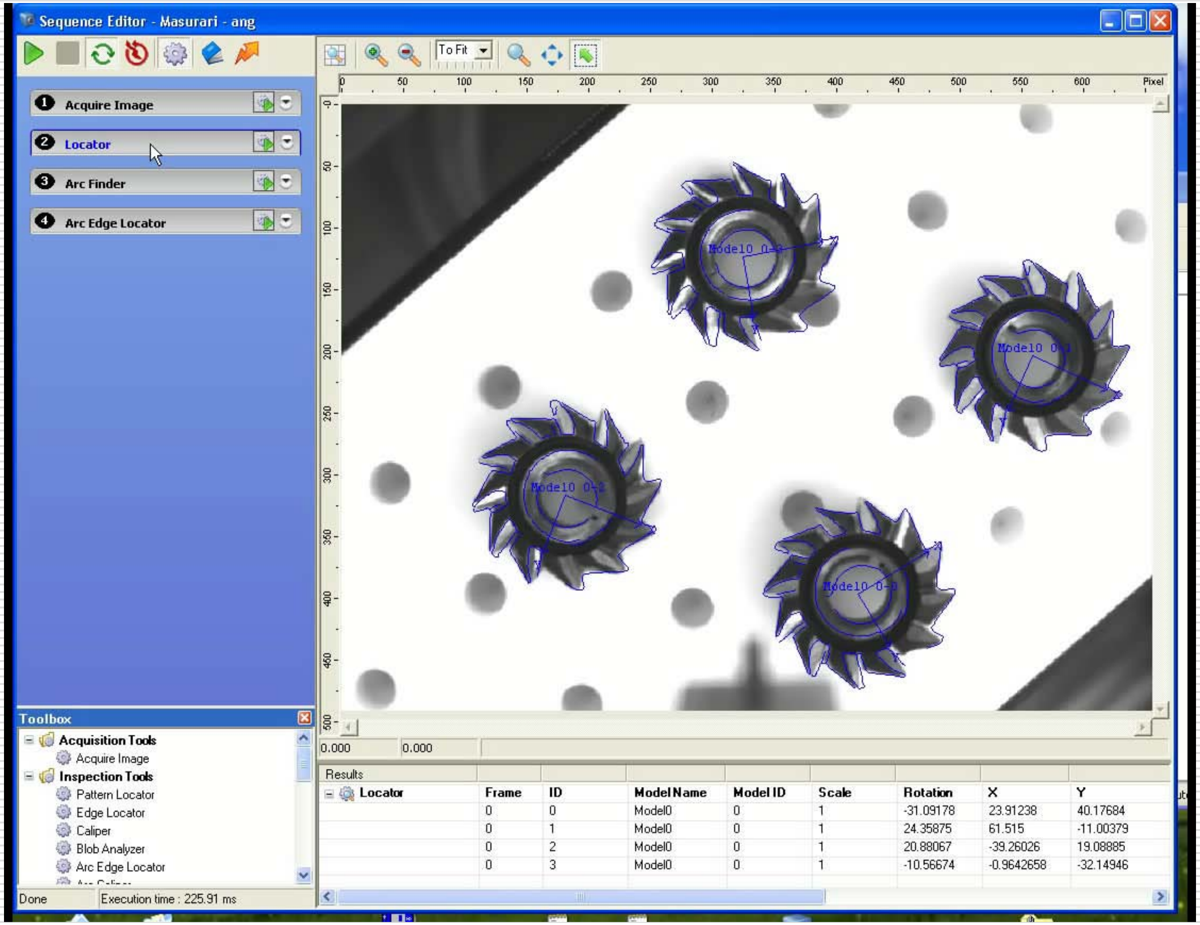
Company: Nuclear Fuel Factory



Automated Visual Inspection

Part Geometry Analysis (2)

- Visual measurements
- Shape descriptors
- Anchor features (point-based)
- Signature analysis
- Component sorting in material flow
- Structured workplace management

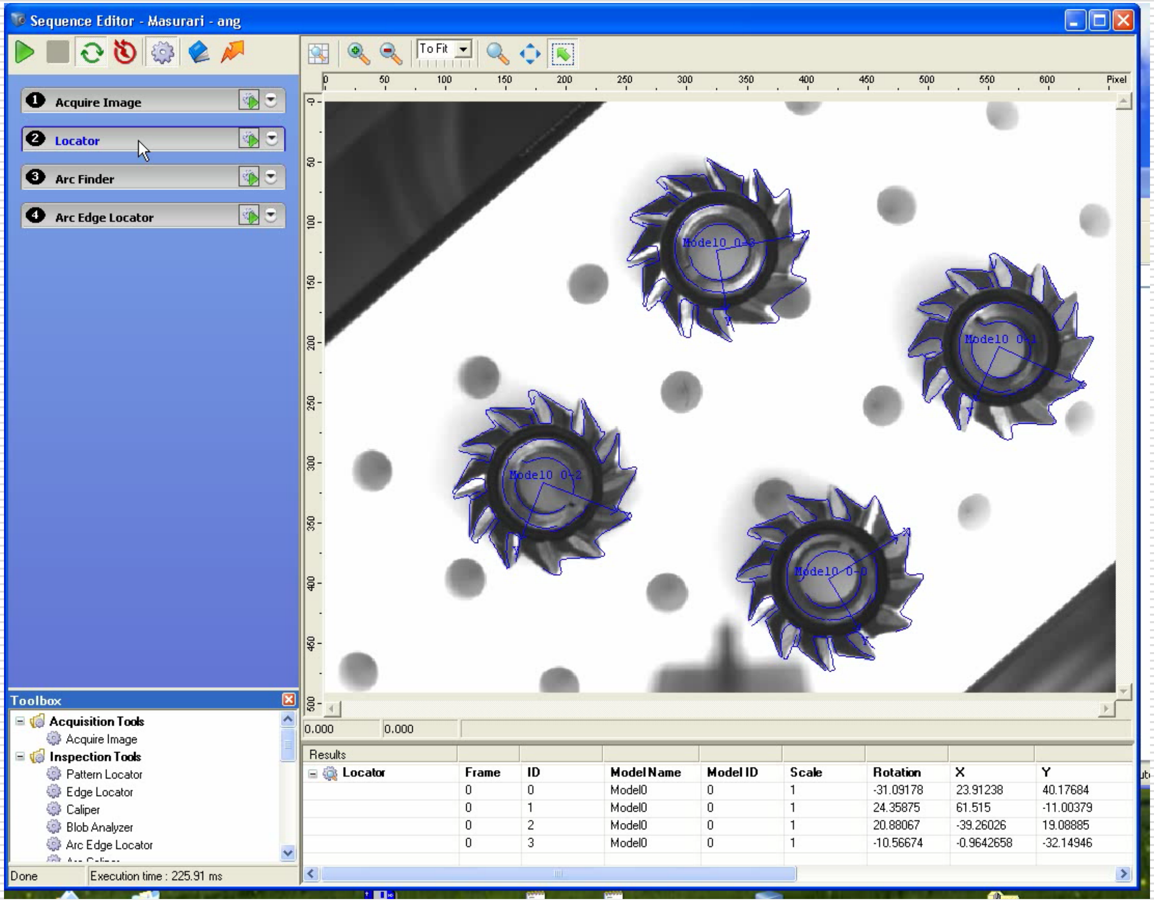


Automated Visual Inspection

Measurement demo (2) using:

- Arc Finder Tools
- Arc Rulers

Company: Nuclear Fuel Factory

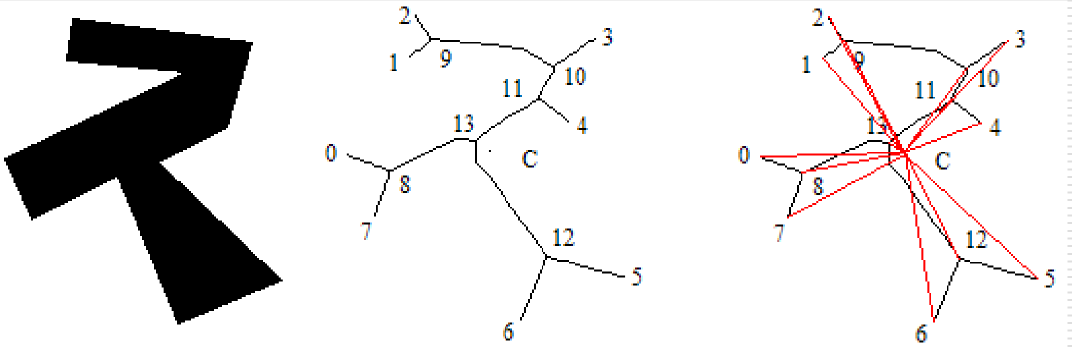


Automated Visual Inspection

- Signature type: 2
- Skeleton length: 330.5 mm
- Max center-node dist.: 70 mm
- INFO_12=(45, 19, 12, 3, 5, 35.5, 6, 13, 53)
  - Dist. From center: 45 mm
  - Angle: 19º
  - ID: 12
  - No. of neighborhood nodes: 3
  - Neighb_node_1_ID: 5
  - Branch length: 30.5 mm
  - Neighb_node_2_ID: 6
  - Branch length: 26 mm
  - Neighb_node_1_ID: 13
  - Branch length: 53 mm
- INFO_6=(65, 37, 6, 1, 12, 26)
- INFO_7=(49.75, 108,7, 1, 8, 19)
- …

Space Domain Descriptors:
Skeleton and skeleton signature

The computation of the signature starts by getting the information from the farthest node and then getting the information from the others nodes in clockwise order.

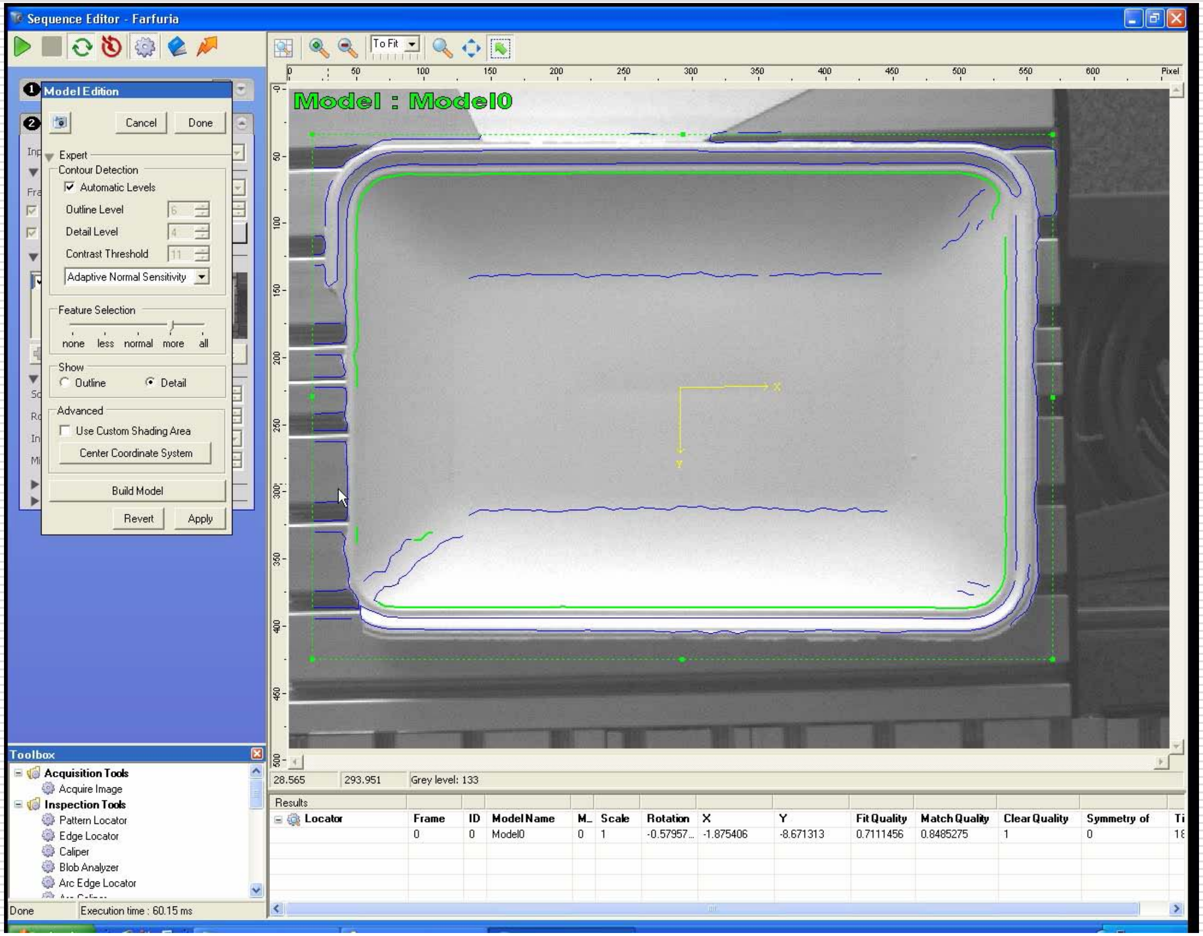


The Skeleton, nodes IDs, the center of gravity and the distances center-nodes.

Guidance Vision for Robots & Automated Visual Inspection

Real-time part locating for correcting robot grasp

- **Company**: IPEC (ceramic plate producer, Alba Iulia)
- **Robot type**: ABB 6 d.o.f. IRB 1600
- **Vision**: 2D, stationary digital monochrome camera
- **Application**: plate from press has position and orientation offsets relative to taught location, due to conveyor driving errors (merged AVI & GVR)



Distributed, semi-heterarchic holonic manufacturing control (d_AI/MAS)

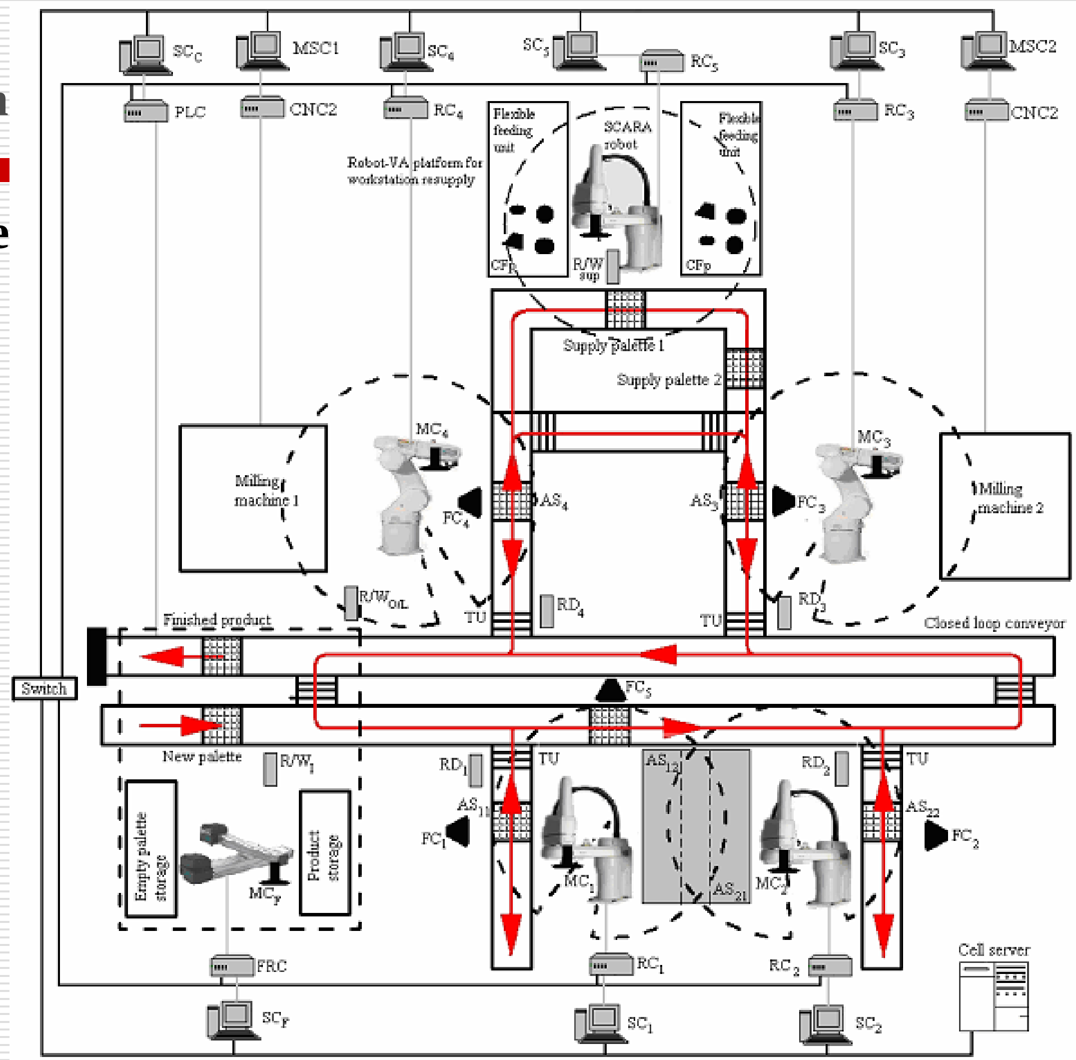
Key features of holonic manufacturing control:

1. Controlled process: networked robot assembly with in-line part- machining and supply
2. Manufacturing structure: job-shop type, transportation by closed-loop conveyor:
3. Use real-time, high-speed machine vision to condition materials/parts (GVR, AVI):
   - ✓ Visually Qualify, Recognize, Locate items in the workplace foreground
   - ✓ Feature-based product- and component measuring: in-line CAQC
   - ✓ Authorize part access based on clear fingerprint check: avoid collision
4. Physical infrastructure addressed: industrial robots, CNC machine tools, machine vision, material storages, intelligent feeding devices, tool holders
5. Semi-heterarchical control architecture, designed as HMES
6. PROSA reference architecture taken as model for HMES development
7. Holons:
   - ✓ Are autonomous and cooperating agents
   - ✓ Encapsulate an information part and a physical part
   - ✓ Holon type: Product- (PH), Resource- (RH), Order- (OH), and Expertise (EH) Holons
8. A Service Oriented Architecture (SOA) integrates 4 areas: (i) **Offer Request Management**; (ii) **Client Order Management**; (iii) **OH & SH Management**; (iv) **OH Execution** (Tracking)
9. Fault-tolerance and disaster recovery provided

d_AI/MAS: the pilot platform

Distributed control architecture

- Distributed architecture:
  information entities (holons) with physical counterparts: Cell- & St- Server, Comp, Ctrl
- Multiple-LAN communication:
  Ethernet, serial, ring, I/O
- Cell Server (IBM xSeries) for:
  - Client order management
  - Job-shop batch scheduler
- PC Station Computers for:
  - CNP data provider
  - Data and pg replication
  - St_Ctrl terminal: edit, track
- PLC for:
  - OH execution
  - Product traceability

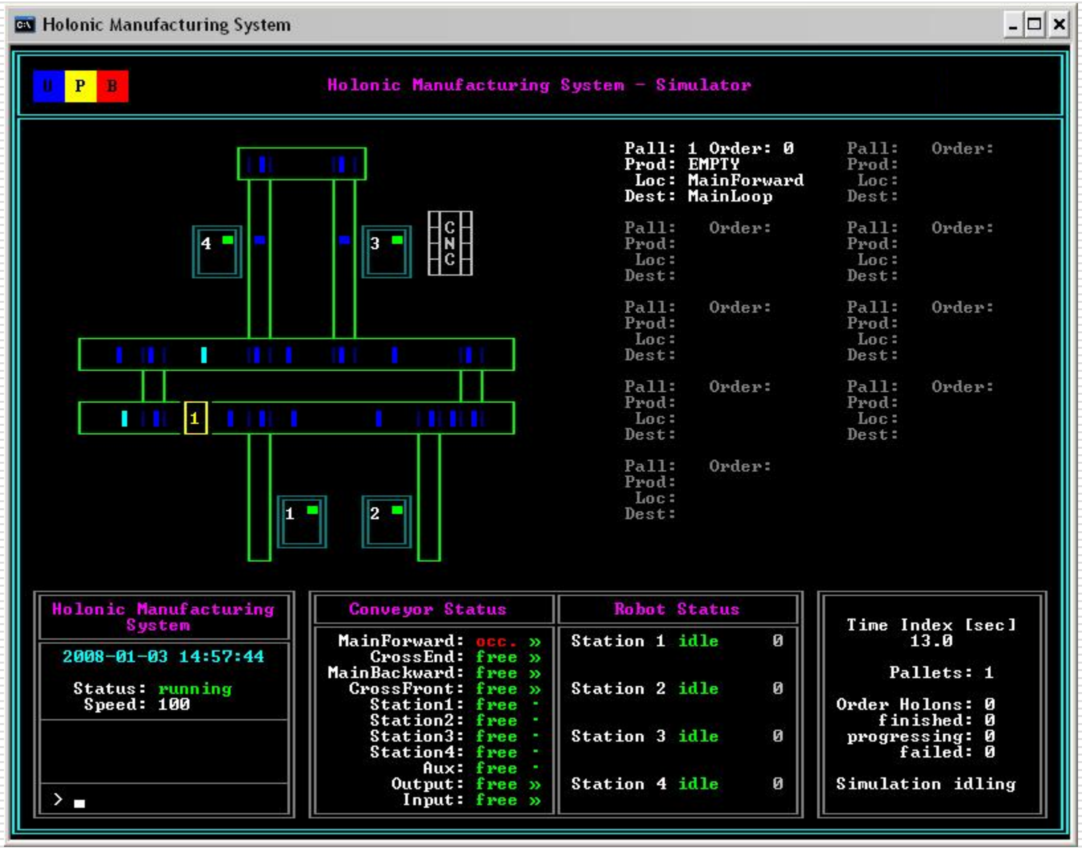


Holonic manufacturing system with self-supply of assembly parts

Distributed, semi-heterarchic holonic manufacturing control (d_AI/MAS)

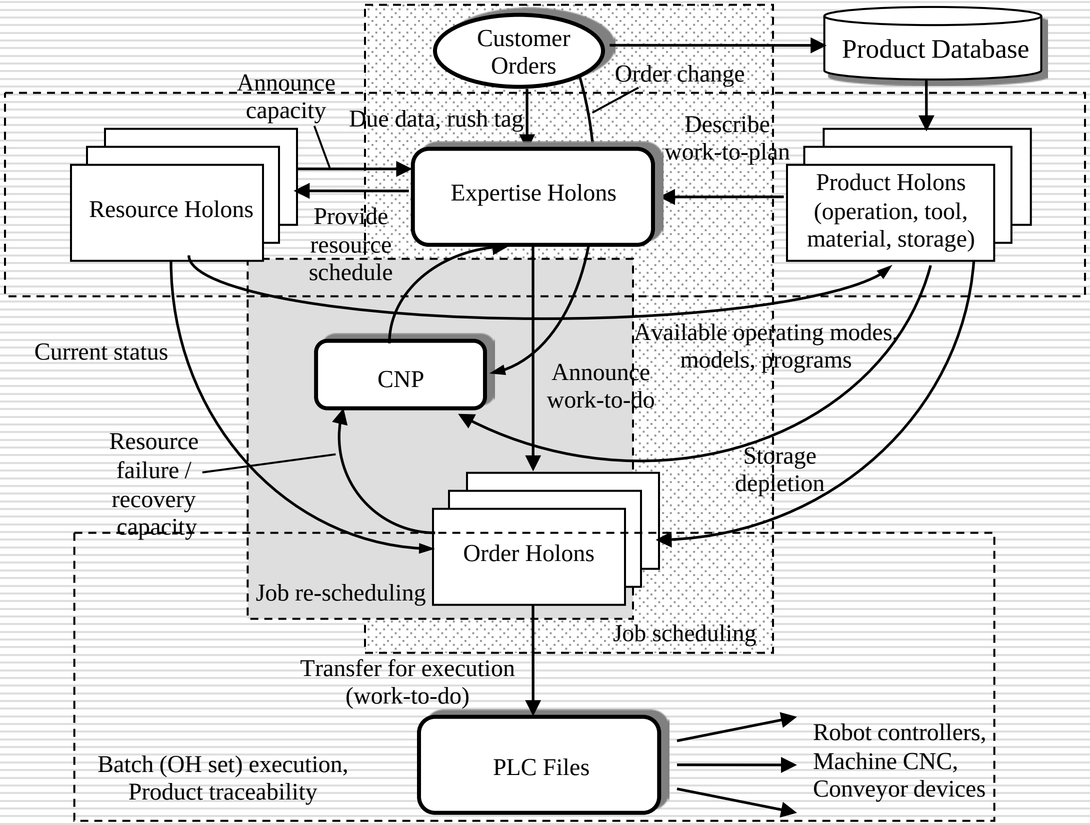
Dynamic simulation tool (DST)

- DST assists & stepwise validates:
  - the creation of OH (off-line)
  - production tracking (on-line)
- DST simulates product transport:
  - A *transportation time matrix* (TTM) is created by measuring the time used to move a pallet between points (stoppers)
  - Smallest time index (transport time slice) *t.s* = 0.5 seconds
- Usage of DST core routines with *variable time base*:
  - Visual simulation: timed mode run, event-driven TTM (0.5 s)
  - OH (re) scheduling: computed mode run, instant-driven TTM
- GUI - conveyor divided in sectors



The dynamic simulator is based on a process-oriented (*product_on_pallet* transport) Graphic User Interface

Distributed, semi-heterarchic holonic manufacturing control (d_AI/MAS)



Basic holon cooperation and communication structure in the semi-heterarchical control architecture

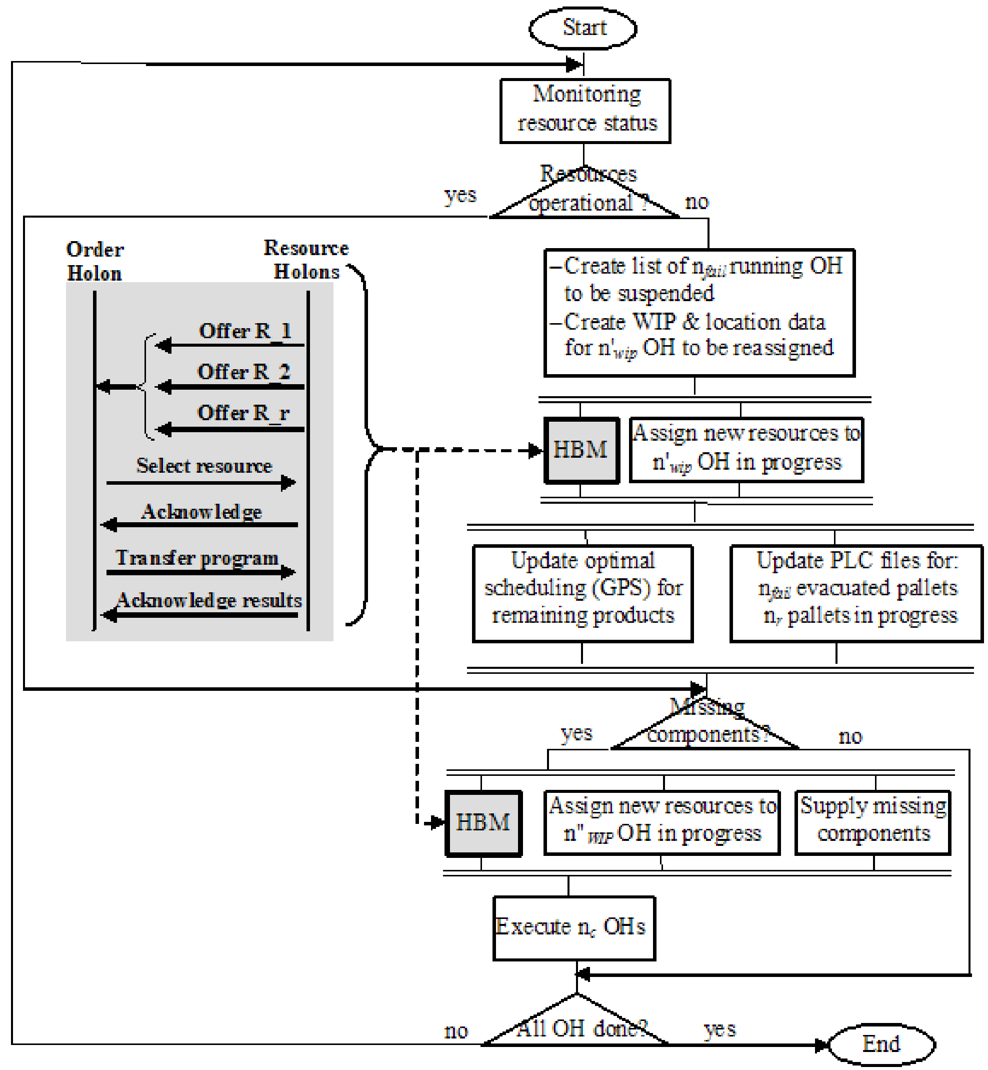
The Holarchy

- Holon types (PROSA reference):
  - *Basic*: Product-, Resource-, Order-
  - *Staff*: Expertise-
- Holarchy types (set of basic rules for holon cooperation for management & control of all manufacturing tasks):
  - Hierarchical (optimal planning)
  - Heterarchical (flexibility, quick response to changes)
- Automatic switch between the two holarchies:
  - *triggered* by: resource failure / recovery, operation failure, supply request and order change
  - based on *CNP mechanism*
- OO holon design: a class containing data fields and functionalities
- HolonManager: coordinates data exchange between holons

d_AI/MAS: Managing Resource Breakdown/Recovery

Failure Manager flowchart:

- HBM: Holonic Bidding Mechanism of CNP type, assigns operations to resources
- N: all scheduled OHs
- n_{WIP}: OH currently in execution
- n_{fin}: finished OHs
- n_d: OHs delayed due to part missing
- n_{fail}: OHs failed (in execution, cannot be finished)
- n_e: OHs not yet started, can no more be executed
- $n'_{WIP} = n_{wip} - n_{fail}$: OHs in execution, can be continued without rescheduling
- $n''_{WIP} = n_{wip} - n_d$: OHs that can be done by re scheduling to resources disposing of parts
- $n_c = N - n_{fin} - n_{WIP} - n_e$: OHs not yet started, can be processed



Dynamic OH rescheduling at resource failure/storage depletion with embedded CNP job negotiation (monex)

Distributed, semi-heterarchic holonic manufacturing control (d_AI/MAS)

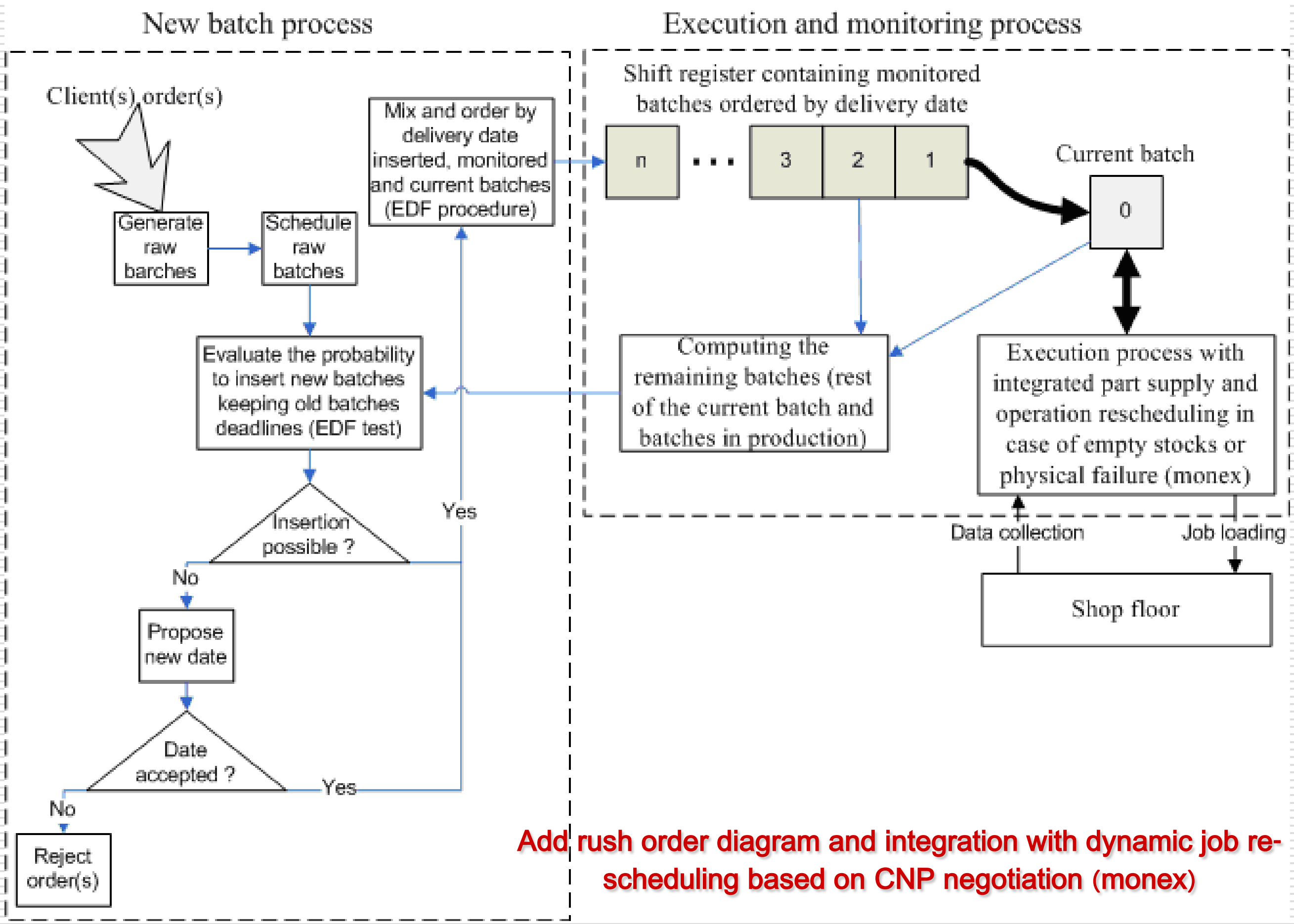
- **Management of Rush Orders**: *The EDF approach used to insert **rush orders** in a production already scheduled*:

1. Compute the remaining time for finishing the rest of the current batch (if necessary).
2. Insert new production data: product types, quantities, delivery dates.
3. Separate products according to their delivery date.
4. Form the entities "production batches" (a *production batch* is composed of all the products having the same delivery date).
5. Generate raw orders inside the production batches (APO lists).
6. Schedule the raw orders (using a GPS algorithm, e.g. KBS or Step Scheduler), compute the makespan and test if the inserted batch can be done (the makespan is smaller than the time interval to delivery date if production starts now).
7. Analyse the possibility of allocating the batches to the cell using the EDF and second equation for feasibility test.
8. Allocate batches to the system according to EDF.
9. Resume execution process with new scheduled OH.

Distributed, semi-heterarchic holonic manufacturing control (d_AI/MAS)

Management of Rush Orders

Companies:
East Electric
SIS



Add rush order diagram and integration with dynamic job re-scheduling based on CNP negotiation (monex)

Product-driven automation with intelligent, embedded devices

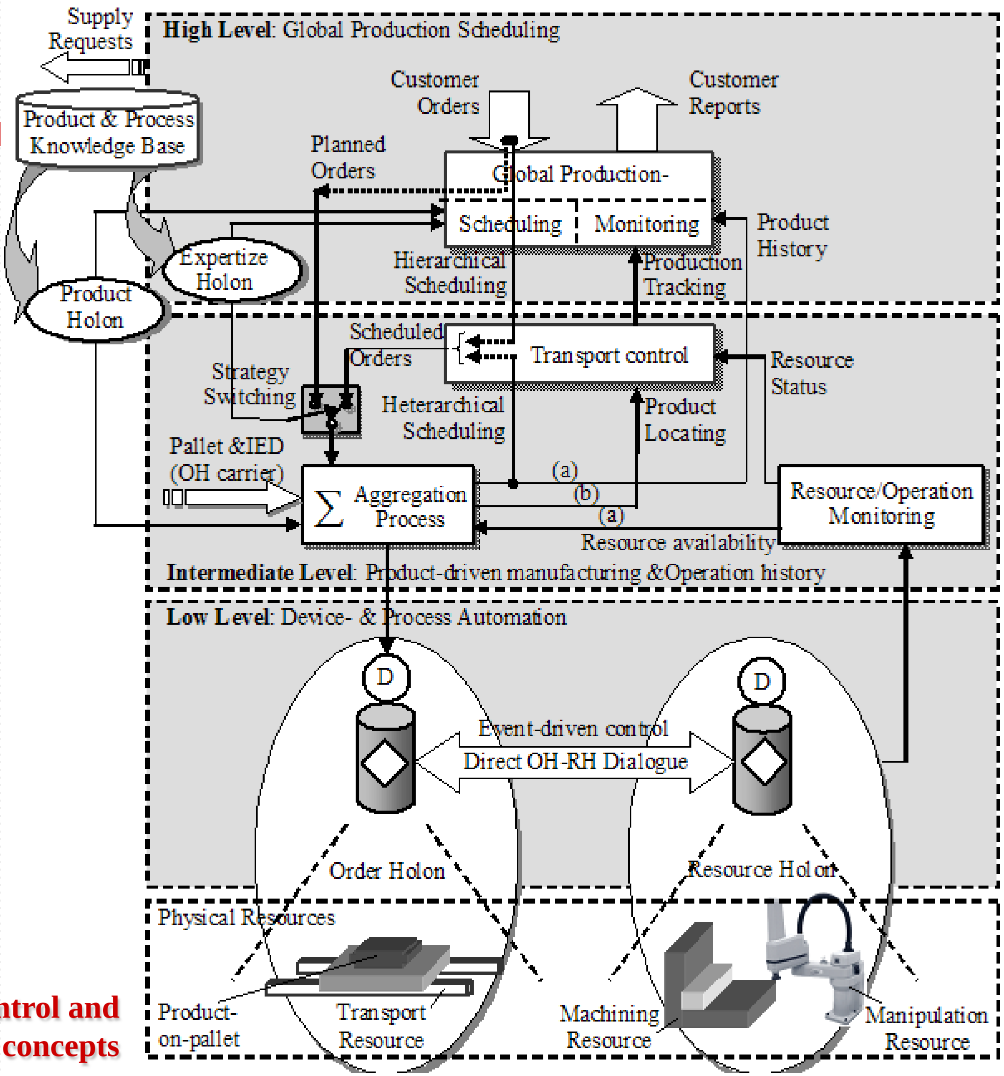
The Open Control (OC) Concept

A global control paradigm, in which traditional (explicit) control is augmented by a new type of control (implicit): *entities can be strictly controlled hierarchically and, at the same time, they can be influenced heterarchically by their environment (environmental) and/or by other entities (societal).*

Effect: allows designing distributed control systems that are both agile and globally optimized, thus reducing the myopic behaviour of self-organized architectures and increasing the agility of traditional architectures.

The Open Control uses IED to implement the **Intelligent Product** (IP) functionality (based on the "*Client-Server*" model (IP -the Server); **strategies**:

- Non-negotiated heterarchical;
- Negotiated heterarchical;
- *Semi-heterarchical*

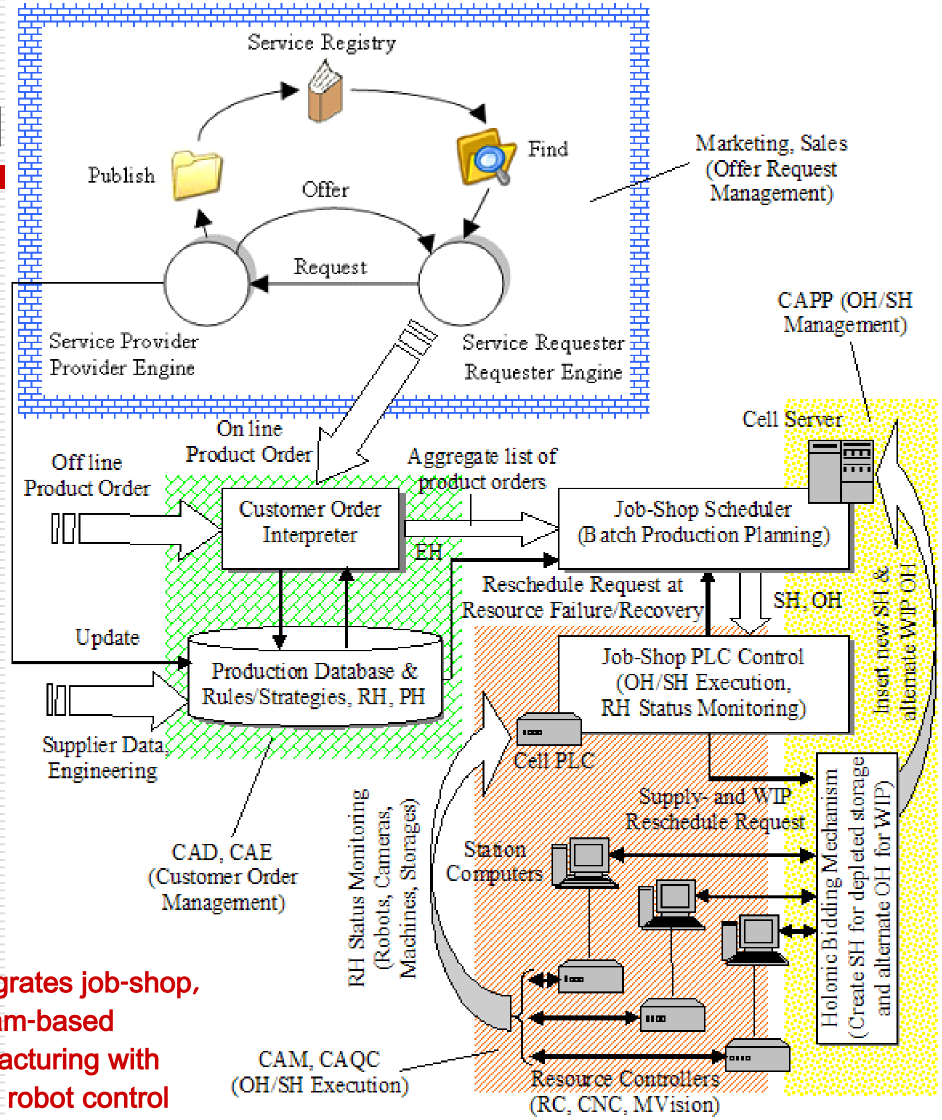


The Open Control and Intelligent Product concepts

SOA for integrated enterprise management & control [open standards]

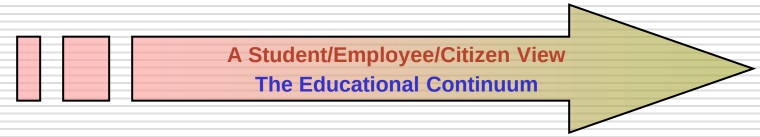
SOA of job shop manufacturing with Holonic control of RC, CNC and MV

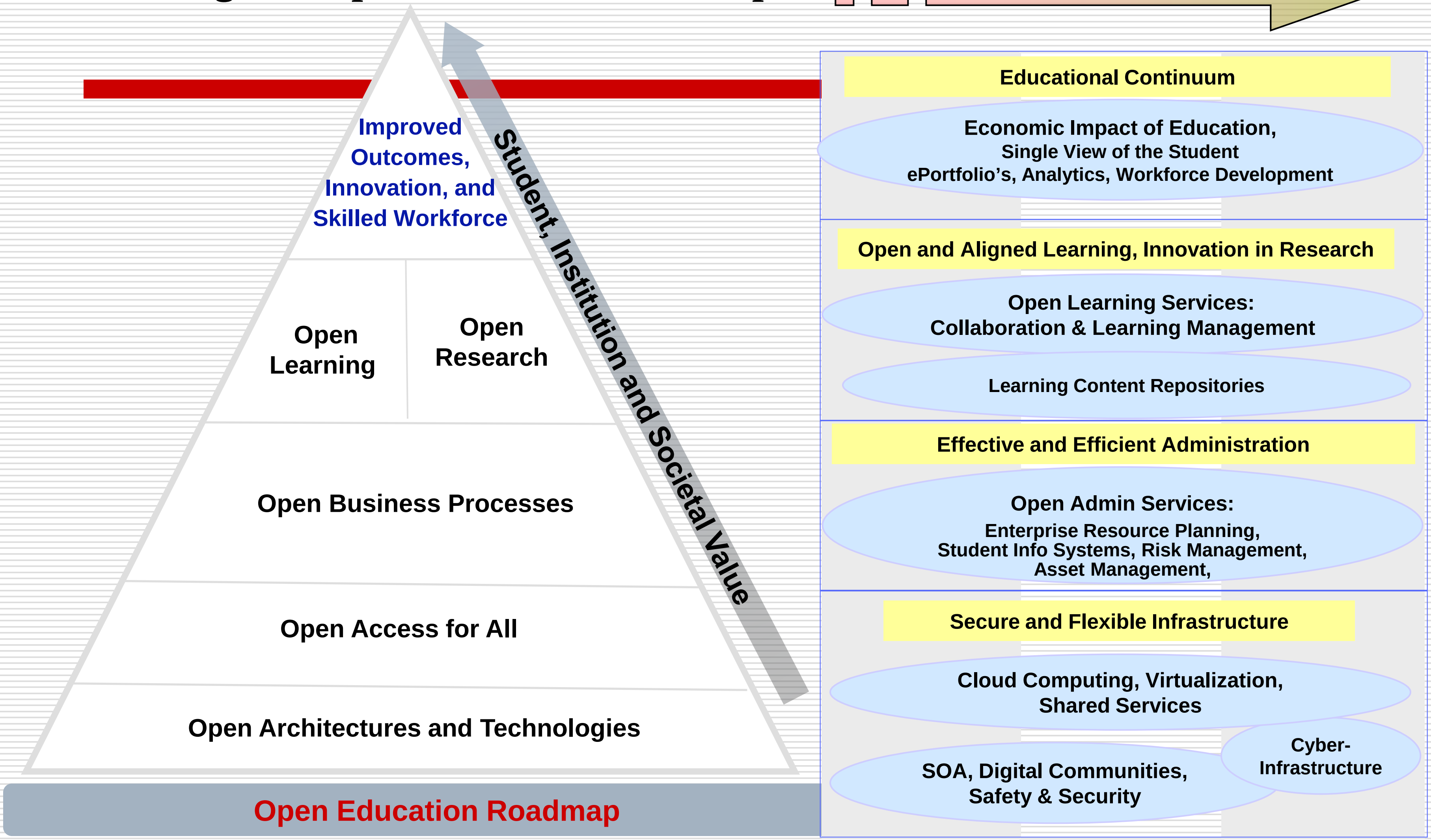
- Closed-loop conveyor
- Networked robot-vision (RV) workstations
- Dual-camera RV stations: stationary (*down looking*) & mobile (*arm-mounted*)
- CNC machine tools (3D, 4D)
- Magnetic RD/WR devices for pallet (product) traceability
- On-line created Supply Holons
- KB Management of Client Orders
- Fault tolerance and disaster recovery



SOA integrates job-shop, team-based manufacturing with holonic robot control

CIMR aligns to open education roadmap





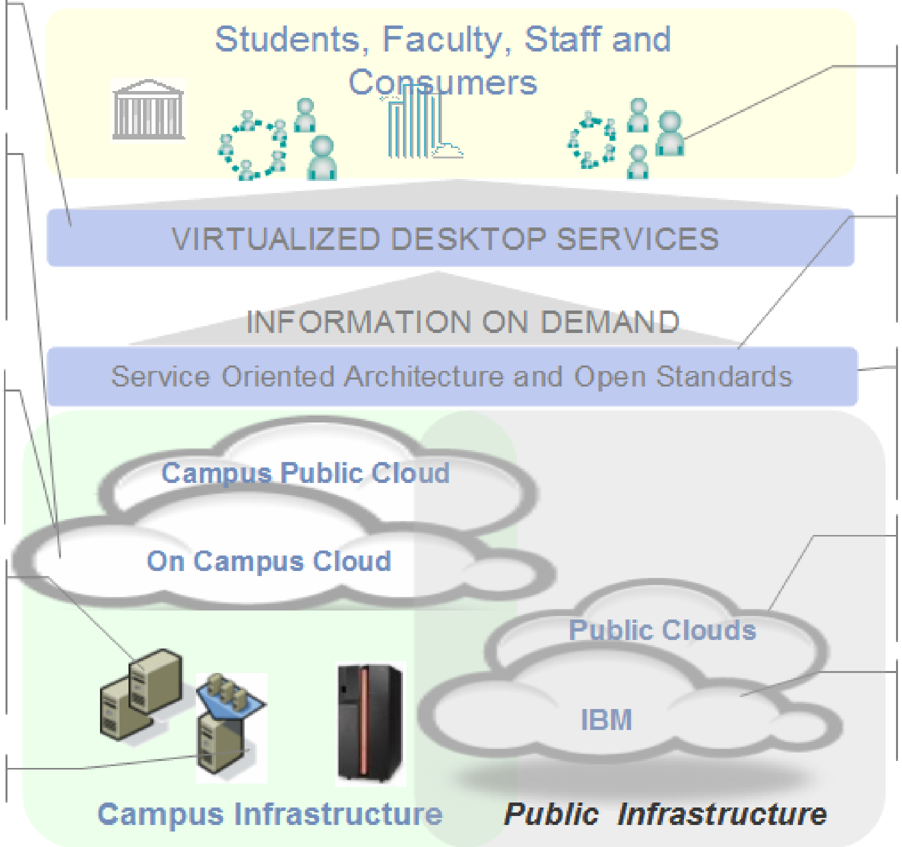
CIMR works in Shared University Research with IBM

Success story – **CIMR in PUB** and **IBM (2007-2009)**

- Over 25 free trainings and lectures organized by IBM for PUB professors and students on IBM and open-source technologies
- 2 new Master Programs started in 2009 with IBM support on curricula and software: *Service Engineering and Management* and *Service-oriented Architecture for Enterprise Management and Control*
- 1 IBM Shared University Research – ww IBM competition – granted to PUB
- 8 IBM Faculty Awards – ww IBM competition – granted to PUB:
  - Education and Research in SSME, an integrated project for the National Academic Network, 2008
  - Integrated Cooperation Space for Competitiveness and Innovation in SME (IN@SPACE), 2009
- 4 IBM PhD Fellowship awards – ww IBM competition – granted to PhD students from PUB:
  - Techniques for the Optimization of Communication Flows in Distributed Systems, 2008
  - Data Storage, Representation and Interpretation in Grid Monitoring Environments, 2009

CIMR builds Cloud Computing blocks with IBM support [open education]

- Virtual Client Services **(VCS)** and Virtual Infrastructure Access **(VIA)**
- Virtualized computer resources pool managed by Virtual Computing Lab **(VCL)** or Tivoli Provisioning Manager **(TPM)**
- **Systems management** for virtual environment (Tivoli Services Automation Manager)
- **eServers** coupled with **Virtualized Storage** provides efficient cost and energy management
- **IBM Blue Cloud**



- **Thin Clients** and **Mobile Devices** allow every user to access services easily
- **Business Intelligence** provides insights and runs processes
- **SOA** and **open standards** create a framework for application interoperability
- Financial and accounting transactions with **e-Commerce SW**
- **Web Services** and **Deep Computing Capacity on Demand**

The End

Thank you !