NRC·CNRC

Robot Programming by Demonstration (Pbd)

Iraj Mantegh
NRC Aerospace

ACAMP Robotic and Intelligent Systems Seminar
June 9,2014

Introduction

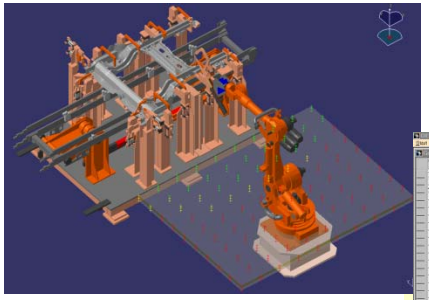
"Advanced manufacturing techniques and management practices that reduce cycle times are a real priority for the (aerospace) industry today"- Advanced Manufacturing, Nov 20??.

"One serious social issue that should be addressed by robotics is the decrease of labor power to maintain industrial activities and social services as the elderly population continues to Increases"
-K. Tanie, President, IEEE Robotics and Aut. Society, 20??)

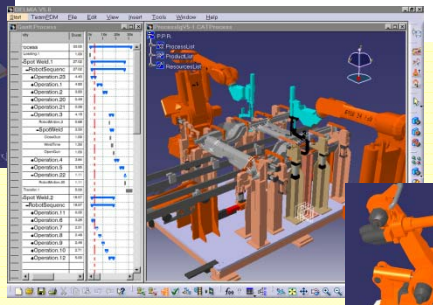
Background
Offline Programming (OLP)

- Does not capture process info
- Requires specific programming expertise

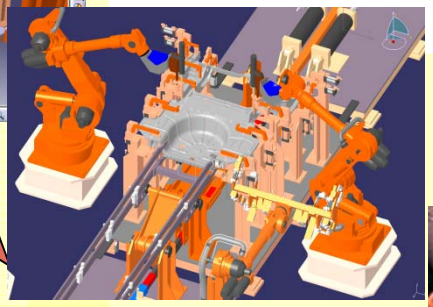
Robot Task Definition



Workcell Sequencing



Off-Line Programming



Excecution in production

Resource-centred tasks & layout (robot & aux. devices)

Validated process (with multiple resources)

Robot(s) programs in native naguage(s)

Robot PBD

- Robot programming by demonstration (PbD) refers to transfer of skills to robots by providing solutions for the required performance through demonstrations.

Traditional robot programming: (e.g. OLP software)

- Time consuming and cost intensive solutions (programming expert, facility, time consuming)
- Task specific
- Robot dependent
- Limited to structured working environment

Robot programming by demonstration:

- Programming by task experts
- Intuitive approach
- Adaptive for different tasks
- Independent of the robot platform
- Continually refine performance with repetition of demonstrations

Robot PBD

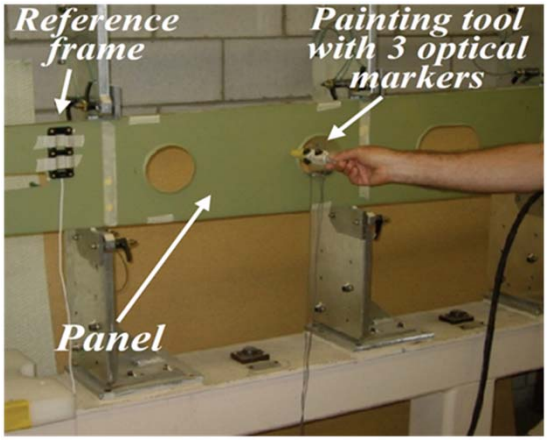
- **Robot PbD:**
  - quick, natural way of robot programming;
  - reduces costs for development of industrial applications;
  - provides framework for service robotics applications.

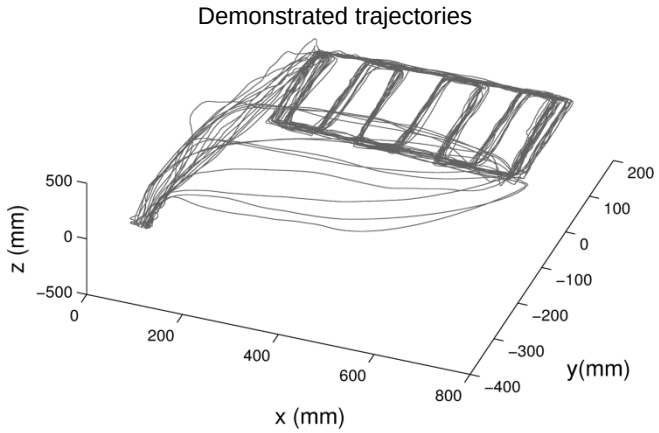
- **Challenges in robot PbD:**
  - learning generic tasks,
  - interpreting teacher's intention,
  - correspondence problem,
  - robust learning,
  - evaluation of learning performance,
  - human-robot interaction.

Methodology

- 1. Perception of demonstrations using optical tracking system

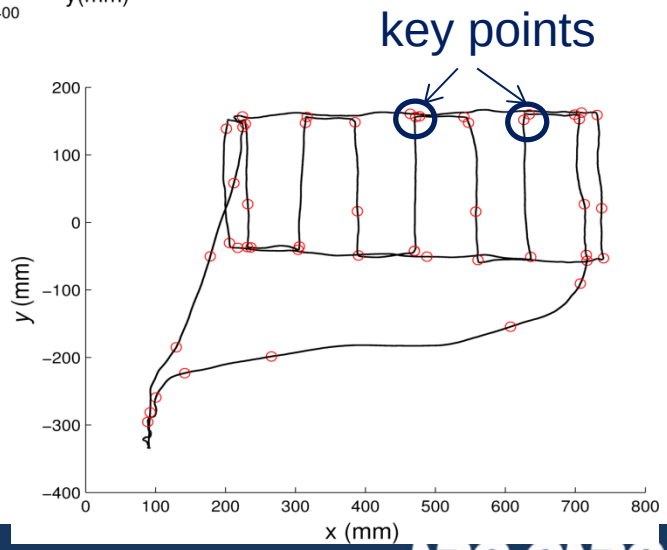
example: painting task





- 2. Selection of candidate key points

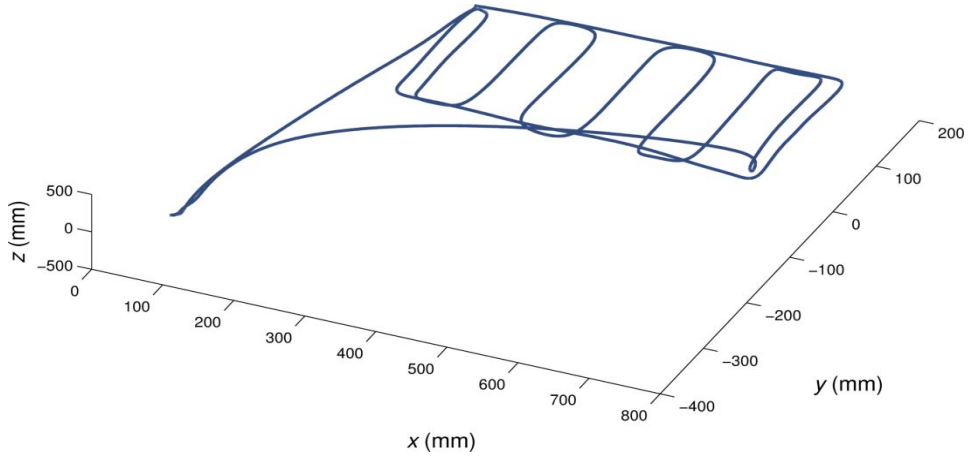
(Linde-Buzo-Gray (LBG) algorithm)



Methodology

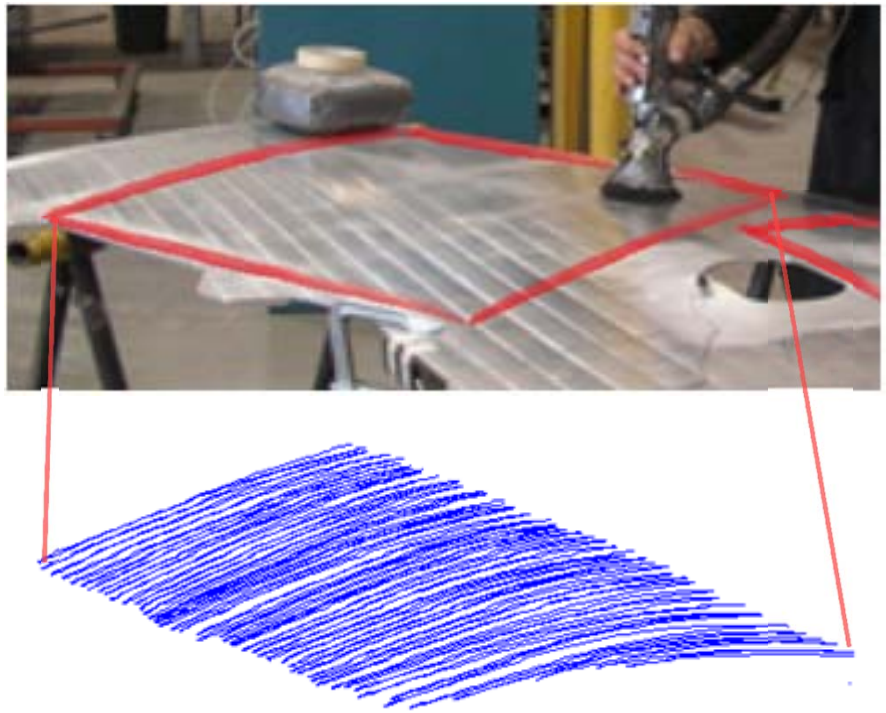
- 3. Segmentation of the trajectories (with HMM)
  - Key points: transitions between the states.
- 4. Temporal alignment of the key points (using DTW (dynamic time warping) algorithm)
- 5. Assigning weighting coefficients to key points' clusters
- 6. Fitting and interpolation between the key points across all demonstration

Generalized trajectory



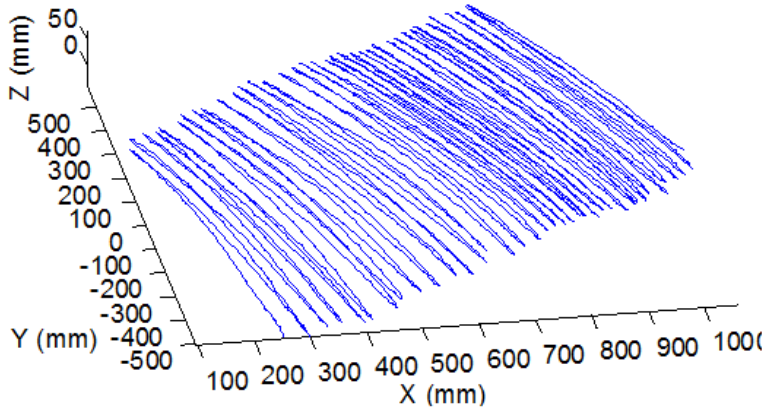
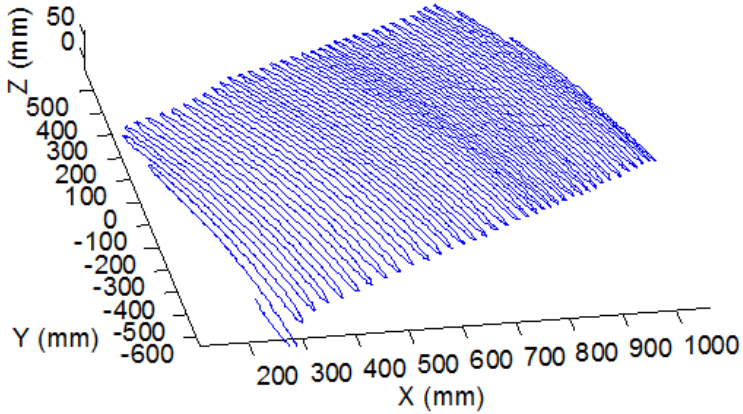
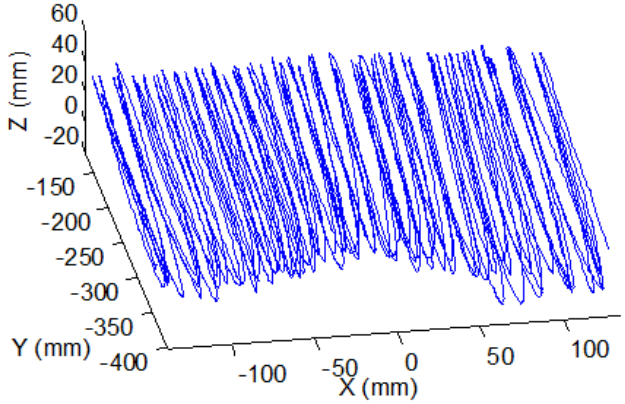
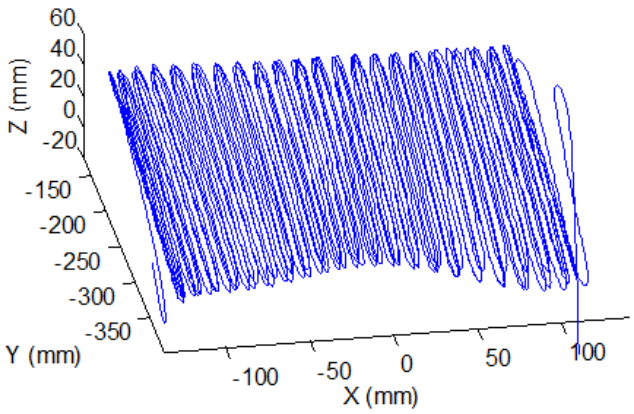
Example

- “Generalizations” drawing from multiple human demonstrations
- Example manufacturing processes:
  - Spray painting
  - Sanding
  - Polishing / Buffing
  - Shot peen forming
- It is shown that the generalized paths are both **more consistent** and **more effective** than the observed human demonstrations



A pseudo-periodic trajectory is followed when shot peen forming an airfoil.

Experiment in Shot Peen Forming: Results

task	Most Consistent Human Trajectories	Generalized Trajectories
1		
2		

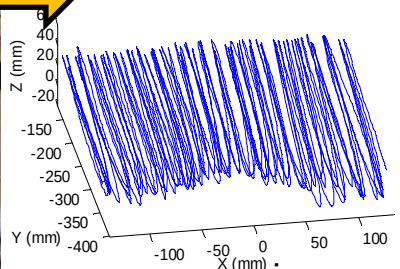
Trajectory Learning: Peen Forming example

Human Demonstrations Trajectories

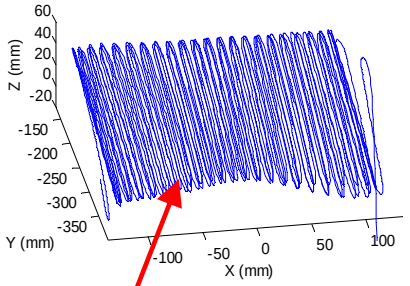
Generated Robot

*Generated trajectories are **smoother, less erratic**, and **more consistent throughout the part***

Sonaca Montreal



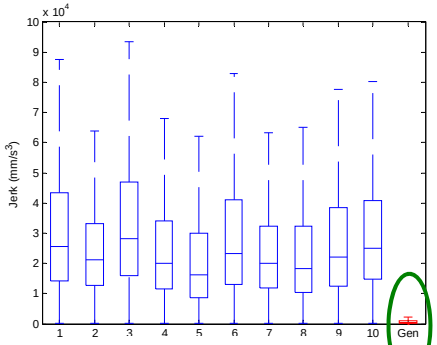
gaps in coverage



uniform coverage

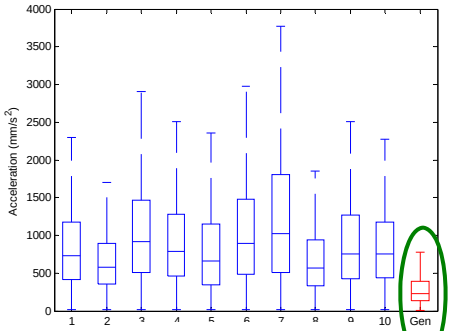
Robot trajectory- created directly from human demonstration

Jerk



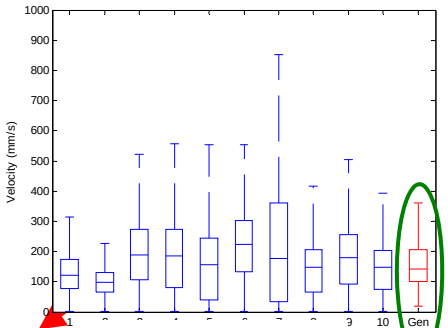
Test Subject

Acceleration



Test Subject

Velocity



Test Subject

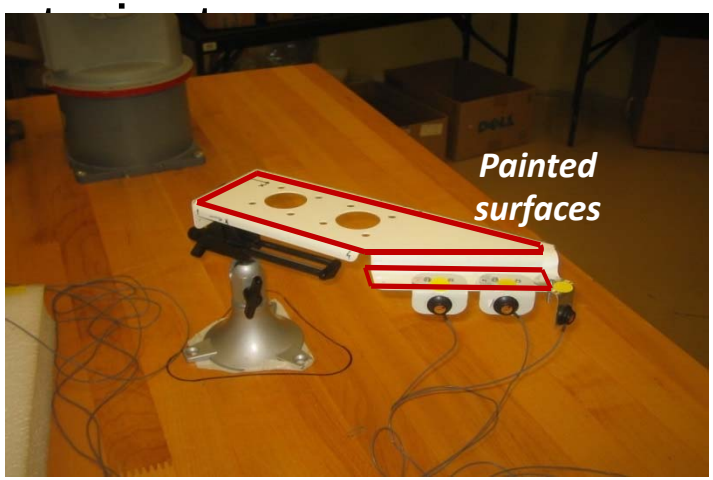
Generalted trajectory has lowest *jerks* and *accelerations* → smoothest!

Generated trajectory's *velocities* are average of all demonstrations (requirement)

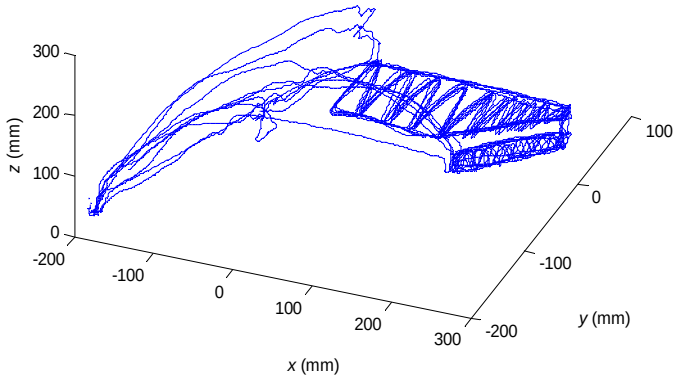
NRC·CNRC

Trajectory Learning: Complex geometry- Spray

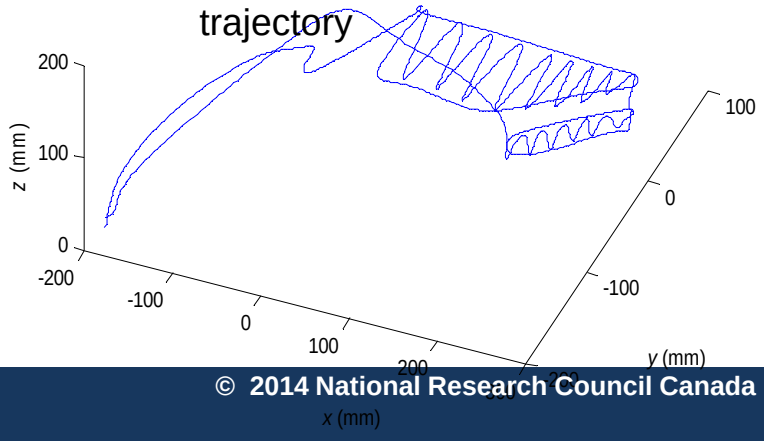
- Example: spray painting parts with more complex



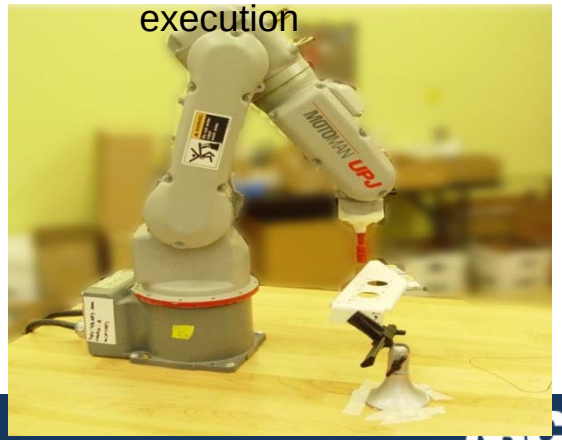
Demonstrated trajectories



Generalized trajectory

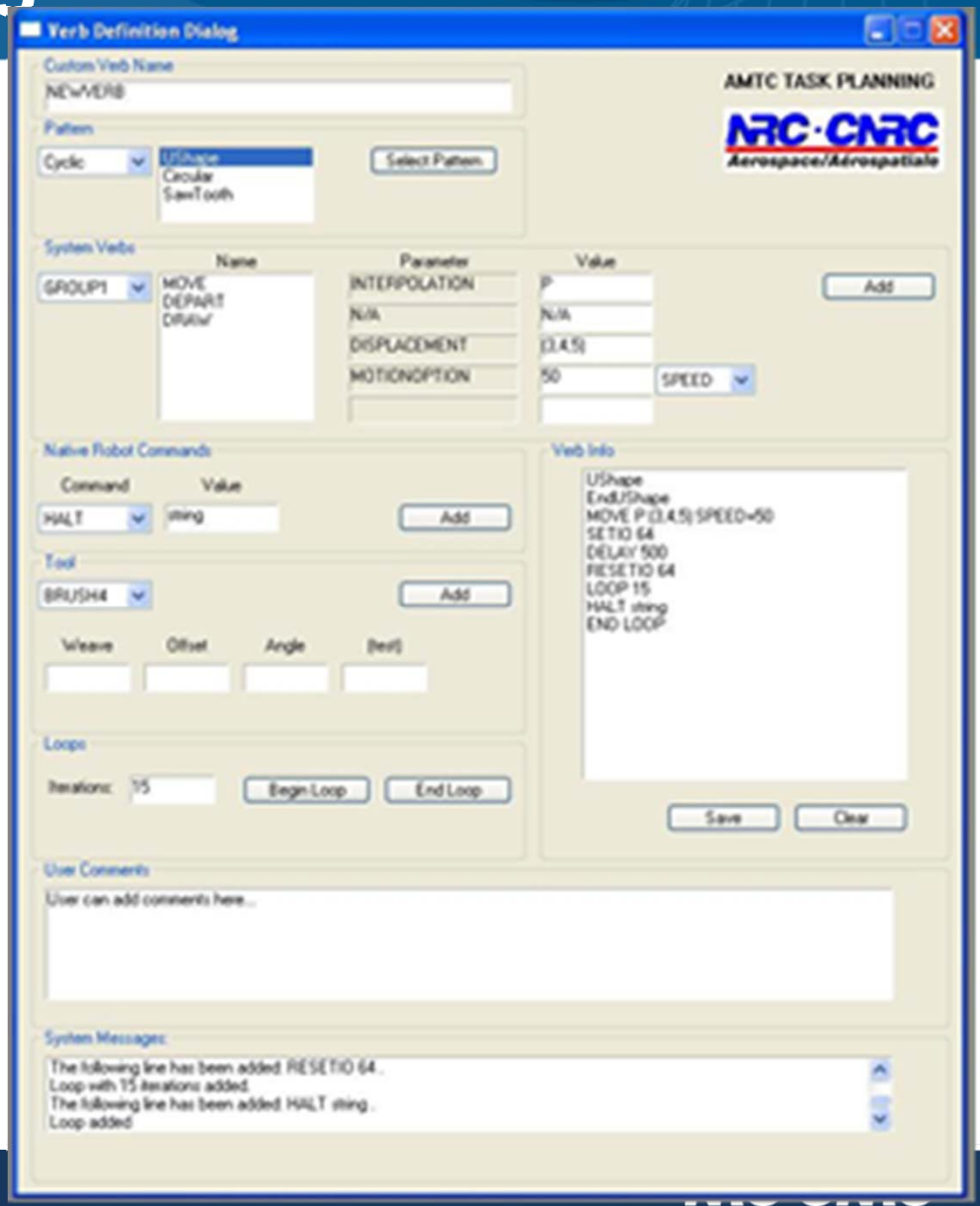


Robot task execution

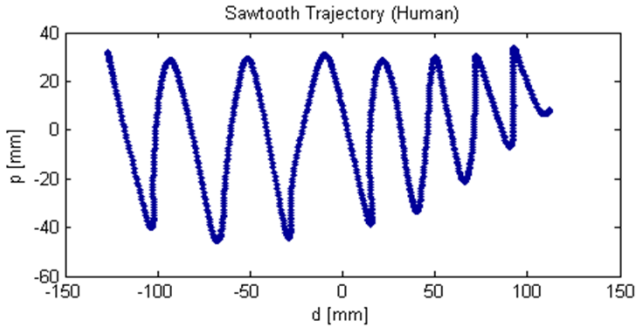


Robot Task Planning

- Define a “Task Template”
- Robot-independent
- Geometry-independent
- Task template include trajectory patterns
- Use the measured trajectory parameters
- Pattern and task reusability
- Invoke trajectory learned by PbD

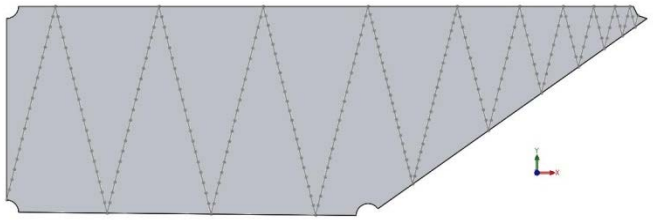


Trajectory Reconstruction (Trajectory Generation)



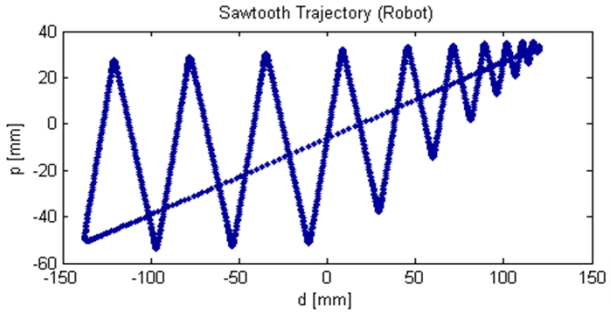
- Analysis of demonstrated trajectory
- Pattern identification
- Kinematic profiling
- Analysis of new geometry
- Trajectory reconstruction

Demonstrated Trajectory



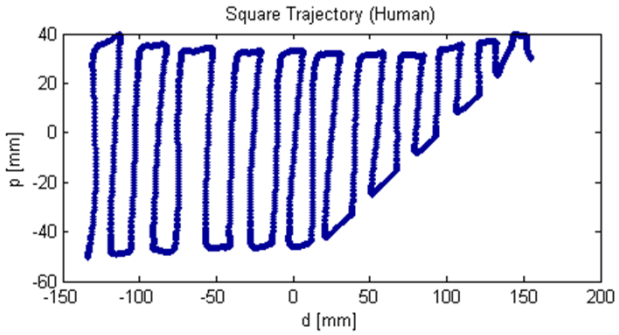
- Generation of robot code
- Upload to robot
- Record and verify with optical tracking

Reconstructed by PbD



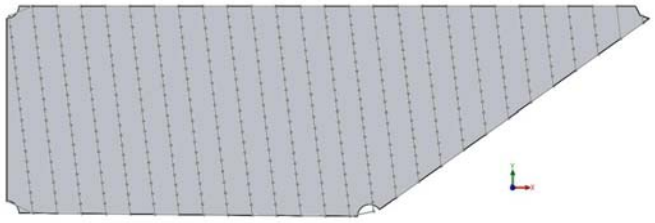
Executed on Robot

Trajectory Reconstruction
(Trajectory Generation)



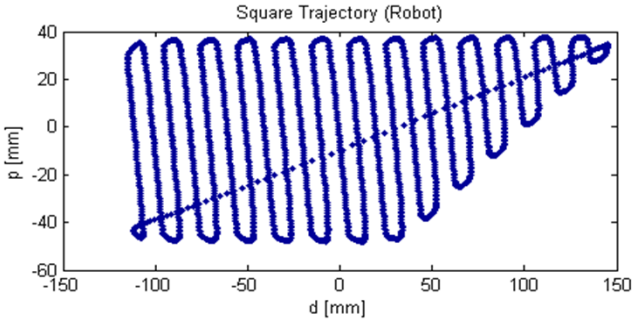
- Analysis of demonstrated trajectory
- Pattern identification
- Kinematic profiling
- Analysis of new geometry
- Trajectory reconstruction

Demonstrated Trajectory



- Generation of robot code
- Upload to robot
- Record and verify with optical tracking

Reconstructed by PbD



Executed on Robot

Thank you

Eric Lefebvre
Business Development / Client Relationship Lead
Tel: 613-949-7548
eric.lefebvre@nrc-cnrc.gc.ca
www.nrc-cnrc.gc.ca

Dr. Iraj Mantegh
Program Lead
Tel: 514-283-9240
Iraj.mantegh@nrc-cnrc.gc.ca