Industrial Manufacturing Automation
Leveraging ROS
Agenda

- Scan-N-Plan Evolution
  - Blending M1 – M4
  - Blending M4 (Demo)
  - Production System
- Production System
  - Overview
  - Challenges
  - Solutions
- Technology Leveraged
- Example Workcell
- Active Research
Scan-N-Plan

Current State
- Requires human interaction (offline) for every programming change
- Unable to adapt to as-built condition
- Offline inspection increases scrap and limits adaptability

Future State
- Enables real-time adjustment to as-built condition
- Eliminates manual programming — operator just specifies tasks
- Enables process feedback/adaptation via automated inspection

In-Process Feedback
Evolution
Production Systems
Production Systems

- Two axis gantry with 6DOF manipulator.
- Size: 6m x 4m
- Joint Effort with Integrator
  - Integrator design and built the system
  - SwRI developed the Scan-N-Plan solution
    - Offline
    - Online
Production System Offline Component

- Add new parts
- Define model data
  - Localization features
  - Verification features
  - Save to database
- Define job data
  - Dynamically generate
    - Surface Tool Path
    - Edge Tool path
  - Save to database
- Ability to reload part and modify data
Production System Online Component

• Process
  - Select parts from database
  - Scan booth
  - Localize
  - Detailed Scan
  - Motion Planning
  - Preview and Approval
  - Execution
• Logging
• Manual Manipulation
Production System Online Component

- Manual Manipulation
  - Open-Source
  - Group Selection
  - Joint and Cartesian Manipulation
  - Pkg: tesseract_rviz
Production System Contact Monitoring

• In large system it is difficult for operators to see everything while manually operating the robot.
• Mitigate this risk active contact monitoring is leverage.
• It currently publishes the contact results at 80hz for the PLC to be able to execute a safe stop to prevent operator error.
Production System Challenges

• Modeling System Constraints
  - Festooning
  - DCS Joint Exclusion Zones
    • If \((J1 < 10 \text{ and } J1 > -10)\) then
      - \(J2 > 60 \text{ and } J2 < 80\)
      - \(J3 > -30 \text{ and } J3 < 40\)
  - Configuration
  - Limit robot extension
  - Numerical rounding
    • Programs sent to Robot are at Joint limits or DCS Joint Limits cause robot faults
    • ROS Reading state at the same limits causing motion planning failures
  - Error Recovery
ROS-Industrial Technology Leveraged
ROS_SCXML

- State machine library based on Qscxml that loads a scxml state machine file definition in order to run a FSM.
- It allows attaching custom c++ function callbacks to state events and can be embedded into a qt gui application.
- Open sourced in the near future.
YAK (Yet Another Kinfu)

- Improvements
  - ROS Agnostic
    - Modern CMake
    - Upgraded CMake version for better CUDA support
Noether

- Tool path generation on well behaved surface meshes (pictures above).
- All waypoints have their z axis normal to the surface.
- Surface segmentation: can divide a mesh into multiple sub-meshes based on local surface features such as average normal direction, curvature and distance.
Noether (New)

• Filter Pipeline (PointCloud & Meshes)
  • Yaml Configuration
• B-Spline Surface Reconstruction
• PCL (Must build from source)
  • http://pointclouds.org/blog/trcs/moerwald/index.php
Tesseract (Planning Environment)

- **tesseract_geometry**
  - capsule, convex_mesh, sdf_mesh, octomap/PointCloud

- **tesseract_urdf**
  - Support new shape types & Quaternions

- **tesseract_kinematics** (Forward, Inverse, Jacobian)
  - IKFast & OPW Kinematics

- **tesseract_motion_planners**
  - TrajOpt, Descartes & OMPL Integration

- **tesseract_process_planners**
  - End-To-End Planners

- **tesseract_ros** (Full ROS support)

- **tesseract_ros2** (ROS2 support - Rviz pending)
• **tesseract_process_planners**
  - Framework that take a tool path generated on a surface and constructs a process tool path.
  - Process Definition
    - Start
    - Segments
      - Segment (Approach, Process, Departure)
    - Transitions
      - From-End
      - From-Start
    - End
Example Scan-N-Plan Systems

- Two axis external positioner with 6DOF manipulator.
- PushCorp Spindle with Compliance Device
- Panel to process
Example Scan-N-Plan Systems

Noether generated toolpath on the surface
Example Scan-N-Plan Systems

Tesseract Process Generator
Example Scan-N-Plan Systems

**Static Tasking**

**Dynamic Tasking**

**Conditional Tasking**

Leveraging `cpp-taskflow`
Current IR&D (Realtime Path Planning)

- TensorFlow
- BULLET
  - Physics Library
- PyTorch

Motion Planning
- GPU Accelerated
Cost/Constraint

GPU Accelerated Collision Checking
The heat method is a general principle that can be applied to any geometric data structure, as long as one knows how to take the gradient of a scalar function. It has been implemented on a variety of data structures including subdivision surfaces [de Goes et al 2016], voxel grids [Coeurjolly et al 2015], spline surfaces [Nguyen et al 2015], point clouds [Crane et al 2013], tetrahedral meshes [Belyaev & Fayolle 2015], and regular grids (using standard finite differences).
Contact Information

- **Levi Armstrong**
  - Senior Research Engineer
- **Southwest Research Institute**
  - 6220 Culebra Road
  - San Antonio, Texas 78238

- Phone: (210) 522-3801
  Email: levi.armstrong@swri.org
- [www.swri.org](http://www.swri.org)
- [www.ros-i.org](http://www.ros-i.org)