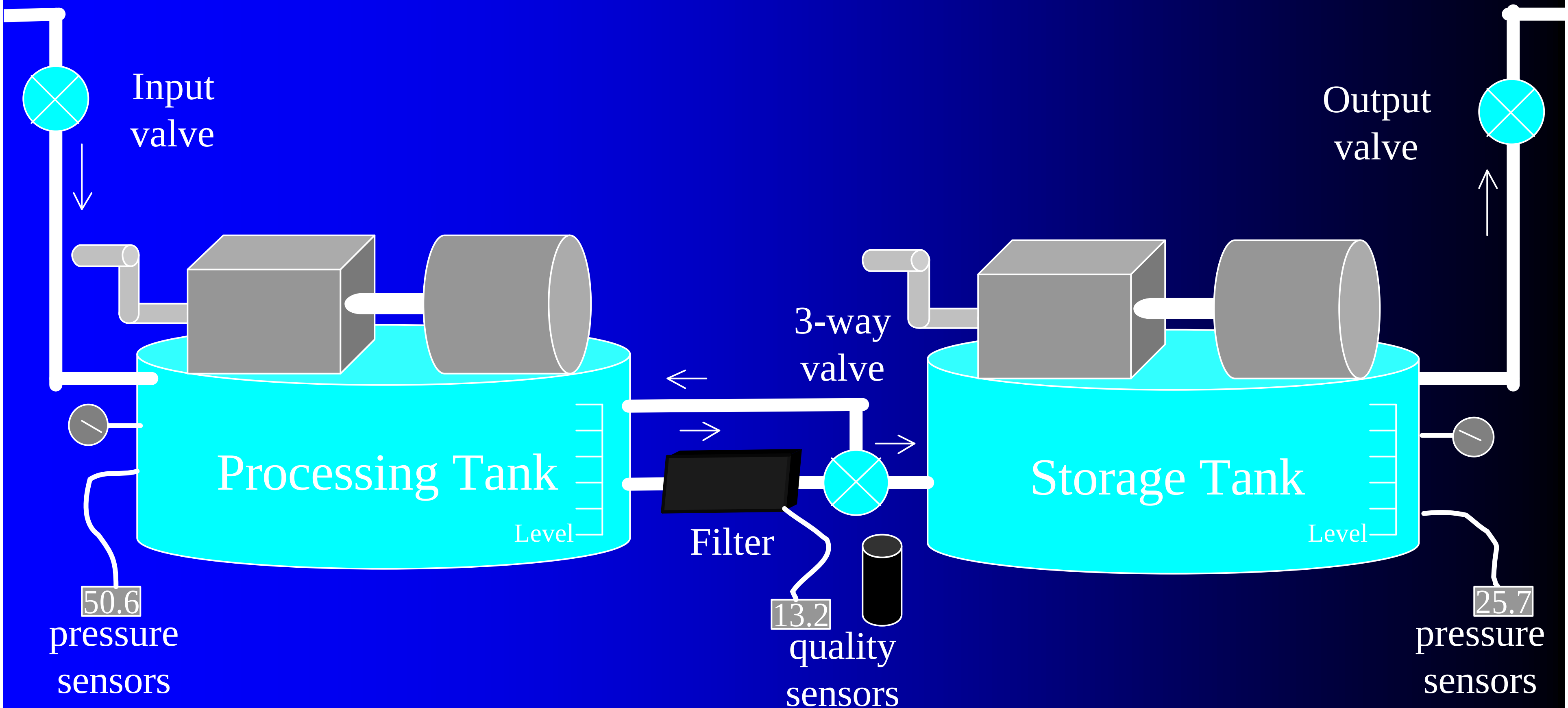
Requirements Outline

- Motivations
- Expanded Tank Example
- Accessibility of information
- Commandability
- Encode multiple methods to accomplish same task
- Knowledge of other available agents
- Changing responsibility: both accept and request

Motivations

- Adjustable autonomy places severe requirements on control systems
- Without a properly designed control system, adjustable autonomy can be ineffective and even dangerous
- Difficult to "retrofit" adjustable autonomy into existing autonomous or tele-operated control systems

Expanded Tank Example



Goal and Procedures for example

- Goal
  - process product from processor tank to storage tank
- Procedures
  - maintain higher pressure in processor tank than in storage tank
  - if product quality is bad then recirculate
  - if processor tank too full turn off input valve
  - if storage tank empty close output valve

Accessibility of information

- Human (or other agent) who wants to adjust autonomy needs to know what the control system knows and what it is doing
- This includes information about
  - state
  - goals
  - tasks
  - models

State information

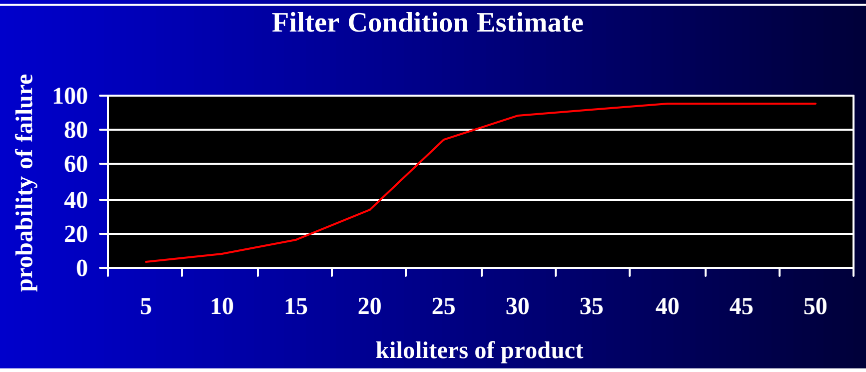
- State is a set of values that represent the current abstraction of the system (internal state) and its environment (external state)
- Human should be able to read and update:
  - internal states of control system
  - control system’s perceived state of the world
  - from our example:

Internal State	**External State**
system setpoints	levels of tanks
autonomy level	quality of product
	valve positions

Models

- Models define set of possible states and their relationships.
  - typical ways to express models include rules, logical statements, equations, procedures, probability distributions.
- Models should be presented in a way that is easily understood by humans.
  - this includes allowing the user to predict futures states given the current state and a course of action as well as modifying the models that make such predictions.

Examples of Models

Model type	example
rule	if filter quality < 10.5 then close valve to storage tank
logical statement	pressure status = $(p_{t1} < 15)$ and $(p_{t2} < 12)$
equation	inter-tank flow = $f(p_{t1} - p_{t2})$
procedure	while (quality sensor > 15) and $(p_{t1} > p_{t2})$ {set 3-way valve to open}
probability distribution	

Goal information

- Goal is a desired set of states.
- User needs to know the system's current goals and its progress in achieving those goals. System may need to explain nonlinearities, e.g., backtracking.
- For our example the main goal is to move water from the processing tank to the storage tank.
  - there may be sub-goals, e.g.,
    - recirculate product in processing tank
    - increase pressure in storage tank to 15

Task information

- Current tasks
  - task parameters
  - start time
  - estimated end time
- Finished tasks
  - finish time
  - success or failure
- Planned tasks
  - estimated start time

Commandability

- A system can be adjustably autonomous only if it can be commanded by an outside agent.
- Commanding can be divided into several types:
  - physical actuation, e.g., *turn pump 1 motor on*
  - goals, e.g., *maintain tank 1 pressure between 15 and 17 for 8 hours*
  - state, e.g., current product quality is 5
  - models, e.g., increase filter quality limit from 10.5 to 12.5 in the rule:
    if filter quality < 10.5, then close valve to storage tank
  - task procedure execution, e.g., *execute procedure 7a*

Encode multiple methods to accomplish same task

- Adjustable autonomy only applies when there are multiple methods (paths) to accomplish system tasks.

Tank example: *if the system only had one tank with a motorized pump and automated sensors and one tank with manual pump and sensors, you would still need to handle human-computer interactions but not adjustable autonomy.*

Encode multiple methods to accomplish same task cont.

- The multiple task methods can be created manually or by using an automated planner/scheduler. The methods can be predetermined or created on demand.
  - automated planning of methods is helpful when there are a wide variety of possible ways to achieve the goals or there are many constraints that disqualify potential methods.

Tank example: *The crew schedule may be such that crew members are only available for certain periods of times and each crew member may be only qualified to perform a subset of the possible manual tasks. Moreover, the electricity to run the motors may be highly constrained and vary over time. In such a case, an automated planner is useful to achieve goals.*

Encode multiple methods to accomplish same task cont.

- For predetermined manually-created methods, agents should be provided succinct information regarding when/where each method is best applied. This information should include, but not be limited to:
  - probability of success
  - criteria for recognizing success/failure/no progress
  - resources available/required
- For planner-generated methods, agents should be provided heuristics to help the planner find and select plans for a wide variety of situations.

Knowledge of other available agents

- Minimum: (1) Human and (1) Autonomous system
- Capabilities
  - in order for an agent to delegate a task to another agent, it must know if that agent has the necessary capabilities.
  - one technique is for each agent to dynamically "publish" its capabilities to a registry.

Tank example: *the system must know when a crew member is available to turn the pump and that a crew member can turn only one pump crank at a time at a maximum rate of x with a latency of y and a maximum duration of z.*

Knowledge of other available agents cont.

- Commanding
  - includes task request, constraints (e.g., temporal, conflicts, preconditions)
  - grants or rescinds authority to do task, use resources, issue commands to other agents

Tank example: *a crew member must be able to command the system not to turn on the storage pump motor for a period of time while the crew member performs maintenance on it. The system may ask if a crew member is available to crank the pump during this period and communicate the consequences if no pumping will be done.*

Knowledge of other available agents cont.

- Recognition of success/failure of these agents
  - success conditions, timeouts, resulting state

__Tank example:__ The system may request a crew member to manually operate the pump and recognize when the desired pressure is obtained by a specified time or event.

- Prediction
  - system should anticipate need to change responsibility, whether to override or request human control

__Tank example:__ An improved system would also recognize if the crew member is pumping at a rate that would enable the goal to be met. The system could then be proactive in making a correction, like switching to the electric motor, instead of waiting for the inevitable failure to occur.

Changing responsibility: both accept and request

- Probabilistic reasoning/latency/reactivity
  - under certain conditions, the user may want to select agent that minimizes risk, other times an agent that maximizes productivity

Tank example: *There is uncertainty about the accuracy of the continuous product quality sensor attached to the filter. Under certain conditions, the system may request a manual check of the product quality and stop production until confirmed.*

Changing responsibility: both accept and request cont.

- Safety
  - transition between responsible agents for goal/task
    - handoff should be planned to minimize risk when possible
    - system should verify with humans what their responsibilities are
    - system may support granting people different levels of command authority

Tank example: *When the system requests a crew member to maintain a specified tank pressure, the system should request that the crew member acknowledge acceptance of the responsibility. Otherwise, the human may assume the pressure is being maintained by the system and the system assumes it is maintained by the human. In such a situation, the pressure is uncontrolled.*

Changing responsibility: both accept and request cont.

- Consistency
  - the system should insure that constraints entered into the system are not violated regardless of the combination of methods being simultaneously executed by the system and users.

Tank example: *There may be a constraint such that both pumps cannot be operated simultaneously when the difference between the tank pressures exceed a specified value.*

Changing responsibility: both accept and request cont.

- Recognize unsafe transitions
  - system should warn human of potentially undesirable consequences of overriding a recommended autonomous behavior.
  - even when the system is not responsible for controlling certain sub-systems, it can advise users when constraints are being violated or prevent commands that would violate such constraints from being executed.

Tank example: *The system should either warn or prevent the user from pressurizing the tank above the set limit or opening certain valves when it is in an over-pressurized state.*