**Intro to Unmanned Vehicle Systems Test 1 Review**

**UVS Introduction Fall 2019 (8/21)**

* **Technological reasons for popularity of Unmanned systems**: Rapid increase of computing power, significant miniaturization of enabling technologies, significant cost reductions in enabling system components
* **Sociological/Economic reasons**: Reduction in risk and cost, reduction in tolerance for loss of life in military operations, potential to do productive work
* **Moore’s law**: 2019 $1000 computer will have power of human brain, 2029 $1000 computer has power of 1000 human brains, 2049 $1000 computer has power of human race. Low cost, high power computing is not a bottleneck
* Future communication includes vehicle to vehicle and wifi across metropolitan areas (WiMax)
* **Edholm’s law**: Bandwidth growing faster than Moore’s law (doubling every 12 months)
* English sent 8 unmanned burning ships in 1588 to confuse and defeat Spanish Armada
* Unmanned flying torpedoes developed but never used in WWI in 1918
* Radio controlled B-17 and B-24 bombers had limited use in WWII
* German Goliath (AKA Beetle Tank) in WWII to destroy tanks, building and bridges
* C.R England trucks control acceleration and gear shifting, and trucks behind will autonomously drive behind to cut down on drag and save fuel. Fuel savings help pay for systems.
* Crop dusting helicopters and harvesting tractors have unmanned systems
* Package delivery and transportation systems are some unmanned systems in Arlington
* **UxV Capability classes**: Teleoperated Vehicles (Searcher), Semiautonomous (Donkey), Platform-Centric Autonomous Vehicle (Wingman), Network-Centric Autonomous Vehicle (Hunter-Killer Teams)
* Teleoperated (Searcher): Human controls vehicle from distance. Relies on sensors that acquire information, communication links, and display technologies. Operator responsible for functions of system. Has no onboard reasoning. Used for whats over the hill or around the corner
* Semiautonomous (Donkey): Designed to follow markers (“breadcrumbs”) left by leader. Uses cognitive process to select best route from marker to marker. Preceder Donkey must have sufficient autonomy to move in advance. Most common used to move supplies
* Platform-Centric (Wingman): Can traverse between two waypoints without any help by human operator. Must be able to carry out mission in hostile environment with survivability and self-defense. Capable of identifying friends, foes, noncombatants. Capable of refueling itself. Used to work in team in military operations
* Network-Centric (Hunter-Killer): Must support coordination between ten to one hundred systems all connected to accomplish complex mission. “Tell us what to do and get out of the way”
* Humans need to be able to monitor many unmanned systems to gain economic advantage. If one human each monitored one system, there would be no economic advantage. Would be the same if the human were to fly the plane or drive the car.
* **UVS enabling technologies:** Human-Robot Interaction (HRI), Mobility, Communications, Power/Energy, Health Maintenance
* HRI: How humans will interact with multiple robots. Covers how intelligent agents work together
* Mobility: Ability of the vehicle to move about in a given operational environment
* Communications: Ability to communicate with a UVS (totally autonomous UVS that accepts no inputs is unrealistic and undesirable)
* Power/Energy: What it will run on and how long it will run. UVS usually require long endurance
* Health Maintenance: UVS must detect issues with system and suggest remedies
* **Autonomous Behavior:** Planning, Perception, Behavior & Skills, Navigation, Learning/Adaptation
* Perception: Takes data from sensors and develops a representation of the world around the UVS
* Navigation: Keeps track of the UVS current position and pose (roll, pitch, yaw) in an absolute coordinate system
* Planning: Decomposes the high-level general task commands into a series of subtasks or functions
* Behavior: combination of sensing and effecting into an atomic action. Can be innate, learned, or strictly a stimulus response
* Skill: collection of behaviors needed to follow a plan or execute a complex task
* Learning/adaptation: improve system performance by analyzing historical system performance statistics and adjusting Autonomous Behavior Subsystem control factors.

**Homework**

* Health/Maintenance: How much battery power is left in the system
* Address UxV Systems in diagram

**System Life Cycle (8/24)**

* Law of Unintended Consequences: When a decision is made, there will be an effect that may or may not be what you wanted
* **System Life Cycle Stages:** Establish system needs, Develop system concept, Design and develop the system, Produce the system, Deploy the system, Operate the system, Retire the system
* **The Systems Engineering Process:** Requirements Analysis, System Analysis and Control, Functional Analysis/Allocation, Synthesis
* Requirements Analysis is the primary focus of the systems engineering process
* Functional Analysis and Allocation: Process of arranging functions into logical sequences, decomposing higher-level functions into lower-level functions, and allocating performance from higher tot lower level functions
* Synthesis: Process which concepts or designs are developed based on functional descriptions from functional analysis and allocation
* Verification: Process confirms that the resulting architecture will satisfy the system requirements
* System Analysis and Control: management and technical activities to control the SE process
* System-of-systems: An integrated, distributed architecture where systems perform functions independently or in combination to meet related objectives.
* System: Works to meet a common objective. Functions with only the aid of human operators and standard support infrastructures.
* Sub-System: Performs a related set of functions. Must interface with other sub-systems and often with external systems to operate successfully
* Component: Performs a function and is the primary building block of the system and is designed to meet requirements and specifications
* Sub-Component: Performs tasks or elementary function. It is designed to meet specifications
* Part: Is combined with other parts to support tasks. It comes in standard configurations and can be obtained commercially
* System Environment: Interactions of the system with its environment from the main substance of system requirements
* Interactions include: Inputs and outputs, System operators, Operational Maintenance, Support systems, System Housing, Shipping and handling.
* System Boundary: What is under the control of the system designers. Can be manipulated to meet user needs. Outside the boundary can be influenced but not manipulated
* Component Design Elements: Electrical, Electro-optical, Electromechanical, Mechanical, Thermo-mechanical, Software
* Interface Elements: Connectors (facilitate the transmission of electricity, fluid, and force), Isolators (inhibit interactions), and Converters (alter the form of the interaction medium)
* **Risk:** Combination of the probability of an event occurring and the significance of the consequences of the event
* Risk management is the processes used to manage risk
* Risk dealing strategies: Avoidance, Transference, Management, Analysis
* Requirements analysis: Mandatory system requirements, performance system requirements
* Risk management process: Select risk management process, identify risk, analyze risks, perform risk abatement (mitigation), Track and evaluate risks
* What can go wrong? What is the likelihood that it would go wrong? What are the consequences if it does go wrong?
* Strategies for risk mitigation: Eliminate risks by using alternate solutions, transfer the risk (software to hardware), prevent escalation of risk by monitoring, accept the risk (do nothing), share the risk (insurance), develop alternatives for critical items, Reduce risk probability or severity

**UGV Introductory Laboratory (8/28)**

* Expendable UxV Platforms: Low cost “Disposable” systems constructed for Commercial-Off-The-Shelf components. Designed to support a narrow range of missions. Suitable for use in Swarming applications and marsupial deployment models. Potential applications include Humanitarian De-mining, Security protocols, Environmental/Natural Disaster damage assessment
* Skid-Steer UGV platforms: Differential drive platform has two independent drive systems. Moves by driving two wheels/tracks at independent velocities. Can be implemented using simple tri-state switch on each drive motor or use a full proportional speed controller for each motor. Skid-Steer platforms include M5 tank toy from Wal-Mart, IMI Bomb disposal robot. Skid Steer subsytems: Vehicle structure, Control Micro Processors, Communications, Drive train and energy storage components, navigation sensors, obstacle avoidance sensors, mission sensors, mission actuators, operator interface
* Inexpensive UxVs need inexpensive controllers
* Pixhawk Autopilot, UGV Control Processor PlugaPodS
* Controller Area Network (CAN) Bus communications
* Wireless communications ZigBee Mesh Networking Protocols
* Drive train & energy storage components: Amplifiers (electric systems), Motors, Gearboxes, Wheels/Tracks, Energy storage (batteries)
* Drive control subsystem for electrical skid steer UGVs: Micro Controller, H-Bridge amplifier, motor, gearbox, encoder
* How do we implement closed-loop control? Hypothesis: If we can determine how far a wheel on one side of the vehicle has turned vs. the other per unit of time we can get a measure of the rate of turn of the platform
* Navigation sensors: Wheel encoders, Digital compass, GPS units, IMUs
* Wheel encoders: counts encoder ticks and rotation direction, used to determine degrees of turn of wheel
* Digital compass: Heading (also pitch and roll on some units)
* GPS units: Global position values
* IMUs: measures linear and angular accelerations
* Obstacle avoidance sensors: IR range sensors, Ultra sonic range sensors, Single point laser range finders, Scanning laser range finders, Vision sensors
* Mission sensors: Image sensors, Metal detectors, Ground penetrating radar, Flame/heat/smoke detectors, Motion detectors, Chemical/biological agent detectors
* Mission actuators: Equipment carried by vehicle that can effect the vehicle’s immediate environment
* Operator interface designed to provide multi-level views or the system

**MATLAB (9/4)**

* clear variables to clear saved variables in workspace
* don’t name files class, path, system

**Dynamical Systems (9/11)**

* System: Transforms an input signal into an output signal

Static system: Output of the system at a given time depends only on the input at that instant  
v(t) = Ri(t)

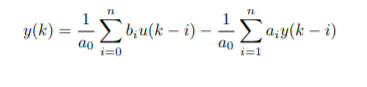
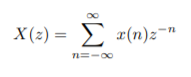
* Dynamic system: Output depends on current and past inputs
* Differential equations: describe evolution of a system signals along time
* Simplest method consists of separation of variables and integration to solve differential equations
* F = ma = m
* Laplace Transform: F(s) = ∫∞-∞e-stf(t)dt
* s = jω, ω = frequency of sinusoidal signal
* a0y’’ + a1y’ + a2y = b0u’’ + b1u’ + b2u -> Laplace -> a0s2Y(s) + a1sY(s) + a2Y(s) = b0s2U(s) + b1sU(s) + b2U(s) -> (a0s2 + a1s +a2)Y(s) = (b0s2  + b1s + b2)U(s) -> Y(s)/U(s) = (b0s2  + b1s + b2)/(a0s2 + a1s + a2)
* Transfer Complex: H(s) = Y(s)/U(s)
* Transfer function is only useful to model systems with linear differential equations and it always assumes initial conditions equal to zero

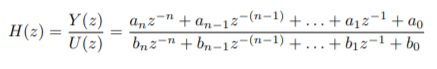
**Transfer Function & State Space (9/16)**

* mx’’ + bx’ + kx = Fu, x’’ is acceleration, x’ is velocity, x is position
* my’’ + by’ + ky = u -> Laplace -> ms2Y(s) + bsY(s) + kY(s) = U(s) -> transfer -> H(s) = Y(s)/U(s) = 1/(ms2 + bs + k)
* state variables: smallest set of variables that are necessary to describe the behavior of a dynamical system
* If differential equation is of order n, the system has n states
* mx’’ + bx’ + kx has two states, x1 = x, x2 = x’
* the derivative of x1 = x’ = x2
* the derivative of x2 = x’’ = -(b/m)x’ – (k/m)x + (1/m)Fu = -(b/m)x2 – (k/m)x1 + (1/m)Fu
* [x’2x’1] = [-k/m -b/m0 1] [x2x1] + [1/m0]Fu
* Linear system: x’ = Ax + Bu, x = n x 1 state vector, u = m x 1 input vector, A and B are matricies of appropriate dimensions
* Nonlinear system: x’ = f(x,u) or for some applications x’ = f(x) + g(x)u
* Things that belong to Transfer Function: Frequency domain, Linear systems, Algebraic polynomials, Zero initial conditions, Bell labs (1940s)
* Things that belong to State Space: Time domain, Linear and nonlinear systems, complex functions, arbitrary initial conditions, Lyapunov (1892), Kalman (1960),
* Kirchhoff’s laws: Vi = RiL + L(diL/dt) + Vc, iL = C(dVc/dt)
* x1 = Vc, x2 = iL [x’2x’1] = [-1/L -R/L0 1/c] [x2x1] + [1/L0]Vi
* Transitional state variables: ξ = [x, y, z]T
* Rotational state variables: ղ = [ψ, θ, φ]T
* Lagrangian model:

A screen shot of a clock

Description automatically generated

* sX(s) -> dx(t)/dt
* (1/s)X(s) -> ∫ x(t)dt
* Computers and microprocessors do not work with exact continuous signals, instead they discretize it
* Discrete-time systems: a0y(k) + a1y(k-1)…. + any(k-n) = b0u(k) + b1u(k-1)…. + bnu(k-n)
* 
* Z-transform of a signal x(k) for discrete-time systems:



* For linear systems: x(k+1) = Ax(k) + Bu(k)
* For nonlinear systems: x(k+1) = f(x(k),u(k))
* zX(z) -> x(k+1)
* z-1X(z) -> x(k-1)

**Modeling and Simulation of a Skid-Steer UGV in Simulink (9/18)**

* ωR = wheel angular speed
* rR = Wheel radius
* ωR rR = Wheel forward speed (if there is no slippage)
* 0 < SR < 1 slippage factor
* SR = 0 -> no slippage
* SR = 1 -> full slippage
* VR = (1 – SR)rRωR wheel forward speed with slippage
* VL = Left track/wheel speed
* VR = Right track/wheel speed
* (x,y) position of UGV
* V = UGV forward speed
* Θ = angle of UGV
* Θ’ = angular speed
* x’ = x-component of forward velocity
* y’ = y-component of forward velocity
* x’ = VcosΘ
* y’ = VsinΘ
* V = (x’)2+(y’)2
* x’ = ½[(1-SL)rLωL + (1-SR)rRωR]cosΘ
* y’ = ½[(1-SL)rLωL + (-SR)rRωR]sinΘ
* Θ’ = 1/b[(1-SL)rLωL + (1-SR)rRωR]
* Pulse Width Modulation (PWM): method for generating an analog signal using a digital source
* Two main components of PWM: Duty Cycle (DC) and Frequency
* Duty Cycle: Amount of time the signal is in a high (on) state as a percentage of its one complete cycle
* Frequency: How fast the PWM completes a cycle

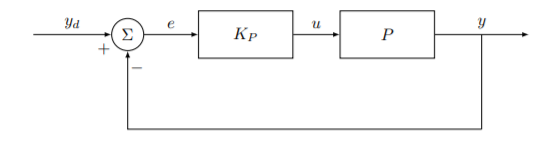
**Guidance, Naviagtion, and Control (GNC) of UGV for Travelling through Waypoints (9/23)**

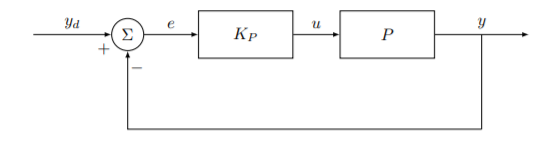
* Navigation: What are the position and orientation of the vehicle?
* Guidance: What should be the vehicle’s velocity (speed and direction) to get where it needs to go?
* Control: How to achieve the required velocity vector?
* Navigation -> Guidance -> Control -> Vehicle + sensors + acutuators
* GPS for position, Compass for orientation
* Encoder reading can be used to estimate position and orientation for indoor ugv’s
* For UGV, advancing position & orientation based on wheel speed estimates from encoder readings
* Encoder signals are decoded to increment or decrement the position counter
* counter increments when primary channel is ahead and decrements when its behind
* A switch in the phase relationship indicates a change of direction
* counter is 16bit integer and free flowing (overflows to -32768, underflows to 32767)
* Encoder gives 22-23 ticks per inch traveled by track, about 900 ticks per meter traveled, eTick = 900[1/m]
* wheel radius r=0.052959, y radian of wheel rotation, track travels y.r[m]
* Right and Left encoder counts moves forward, moves backward, turns left or right
* wheel speed = eCount(k) – eCount(k-1)/eTick.r.TsampleEncoder
* eTick = Ticks per [meter], r=wheel radius, TsampleEncoder=sampling period [sec], eCount(k)-eCount(k-1) = number of ticks in one sampling period
* When overflow/underflow, the counter jumps
* When |eCount(k)-eCount(k-1)|>32000, UGV cannot move that fast, must use [eCount(k-1)-eCount(k-2)]
* x^’=r/2(w^L + w^R)cosθ^
* y^’=r/2(w^L + w^R)sinθ^
* θ^ = r/b(w^L + w^R)

**Guidance, Naviagtion, and Control (GNC) of UGV for Travelling through Waypoints (9/23-25)**

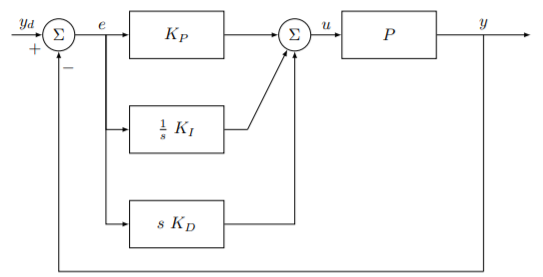
* θcom = tan-1(y-yi/x-xi),i={1,2….,n} angle to turn towards to the wayPoint
* |r| = sqrt[(y-yi)2 + (x-xi)2], distance to the waypoint
* when |r|<=rp2, i= i + 1 switch to the next waypoint
* Constant commanded speed always < MaxSpeed
* Commanded speeds is reduced when close to waypoint

**Brief Introduction to PID Controllers (9/30)**

* Closed loop controllers are used over open loop
* Feedback controller: Closed loop system, measures the state of a system and applies the appropriate input to correct the deviation of the output with respect to a desired value
* yd = desired trajectory for the output of the system
* e(t) = yd(t) – y(t), error signal
* u(t) = Kpe(t)
* 
* uI(t) = ∫40e(τ)dτ
* u(t) = Kpe(t) + KI ∫40e(τ)dτ

PI Control

* uD(t) = de(t)/dt
* u(t) = Kpe(t) + KI ∫40e(τ)dτ + KD(de(t)/dt)



PID Control

* Integral term guarantees a steady state error of zero by limiting the area below the error curve
* derivative term responds to the velocity of change of e(t) and applies a corrective action before it becomes too large
* We only need to select the gains of Kp, KI, and KD to obtain the desired performance of the closed-loop system.
* Speed = V = sqrt(xdot2 + ydot2) = r/2(wL + wR­)
* wtilde = wL + wR­, Δw = wL - wR­, wheel speed
* V = r/2(wL + wR­), translation speed θdot = r/b(Δw), Angular speed
* Total wheel speed to control V, Differential wheel speed to control θ, use PID control
* wR = (wtilde – Δw)/2, wL = (wtilde + Δw)/2

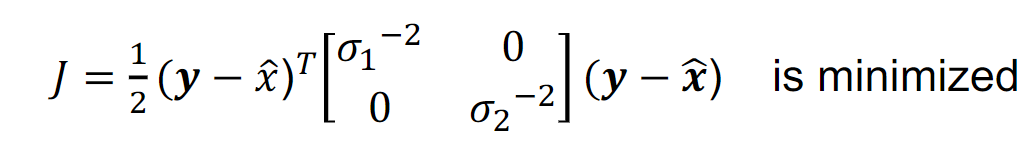
**Sensing and Perception (10/2)**

* Sensors are used to make vehicle aware of their status and surroundings
* Types of sensors: vehicle-centric internal sensors, external and state vehicle sensors, mission-centric sensors
* Sensor: device that converts a physical stimulus (temp, pressure, force, etc) into a compatible form (voltage, current, digital, stream) that can be interpreted by a control system for the purpose of measuring the stimulus
* Trend of sensors: Better, smaller, cheaper
* Desirable sensor features: High accuracy, high precision, wide operating range, high speed of response, ease of collaboration, minimum drift, high reliability, low cost, compatible with system architecture
* Software tasks: Routine platform operation, Diagnostics (why the system failed), Prognostics (when will the system fail next)
* Software development costs generally far exceed cost of physical sensor
* Internal/Vehicle-Centric sensing supports UxV Health Maintenance functions and Enable the operation of basic vehicle functions, independent of mission
* Health Maintenance functions: Platform operation monitoring, diagnostics, prognostics
* Basic vehicle functions: temperature sensor to regular amount of cooling, voltage sensor to monitor amount of charge remaining on battery
* Types of Internal/Vehicle-Centric Sensing: Temperature measurement (Thermocouple, Resistance temperature detectors, Infrared), Vibration measurement (displacement-based, velocity-based, accelerometers), Force measurement (load cell, strain gauge)
* Sensors enable the UxV to collect data about its environment, data gathered about the environment influenced influences UxV’s behavior
* Classes of External Vehicle Senors: Vehicle state sensors, Obstacle detection sensors
* Vehicle state sensor: position and orientation of vehicle with respect to one or more relative or absolute reference frames. For discussion we’ll include velocity and acceleration as vehicle state sensors
* Types of vehicle state senors: GPS recievers, digital compasses, Inclinometers (tilt sensors), Inertial measurements units (IMU), encoders
* GPS recievers: determined a globally referenced location of vehicle, needs to be locked on to at least 3 satelittes to calculate 2D position (latitude and longitude). Four or more satellites, receiver can determine 3D position (latitude, longitude, and altitude). GPS can calculate speed, bearing, track, trip distance, distance to destination, sunrise and sunset time and more
* GPS perform triangulation calculations to determine users position
* GPS is comprised of a network of 24 satellites
* GPS can be improved by fixed ground facilities. Enhancement strategies are referred to as differential GPS and commonly uses the WAAS (Wide Area Augmentation System)
* Ionospheric perturbations and magnetic storms cause GPS signal delays. Can be compensated using SBAS and L1/L2 decoding. Multipathing causes positioning errors. Multiple GNSS receivers are better suited to filter out multipathing due to higher number of visible satellites at the same time
* Digital Compass: Used to determine heading, pitch, and roll. It senses the difference in earth’s magnetic field from disturbance caused by external elements
* Digital compasses consist of 3-axis magneto resistive sensor, signal conditioning software and hardware, and tilt sensors or accelerometers.
* Embedded microprocessor and software transform sensor inputs into desired heading, pitch, and roll angles
* Inertial Measurement Units: Senses the type, rate and direction of motion
* Information provided includes: Current rate of acceleration, yaw, pitch, and roll
* System can use information to calculate current vehicle position and speed, given an initial speed and position
* IMUs do not need to communicate with any external systems
* IMUs suffer from unbounded accumulated error and calculations tend to drift and become inaccurate
* IMUs reduce mechanical complexity, are physically smaller, and increase computational complexity
* Fiber Optical Gyroscope: uses interference of light to measure angular velocity, consists of a large coil of optical fiber
* MEMS (Micro Electro Mechanical System): sensors built using silicon micro-machining techniques. Angular velocity is calculated based on the difference in time between two vibrations
* Accelerometer (mechanical): mass-spring system. Mass displacement and Force used to move mass is used to calculate acceleration (a = F/m)
* Three types of solid state accelerometer: surface acoustic wave, vibratory, and silicon and quartz devices
* Surface acoustic wave: Consists of cantilever beam resonating at a particular frequency, acceleration changes the frequency and acceleration is calculated from the change in frequency
* MEMS sensor advantages: small size, low weight, rugged construction, low power consumption, short start-up time, inexpensive to produce, high reliability, low maintenance, compatible with operations in hostile environments. MEMS disadvantage: far less accurate
* Encoders: Measure linear or angular displacement. Rotary encoders can be used to measure rotation of motor or wheel. With help of microprocessor software, they can be used to measure vehicle velocity and position.
* Incremental encoders measure change in position
* Unmanned vehicles operate in unstructured environments, there are going to be objects that will lie in the path of the vehicle. Obstacle detection sensors identify obstacles that impede motion of robot
* Types of obstacle detection sensors: Laser range finding (LADAR), Radar, machine vision, ultrasonic, infrared, tactile sensor
* Laser range finder: Uses laser beam to determine distance to a reflective object. Works by measuring time it takes for laser to be reflected off object and returned
* LADAR aka LiDAR, same as laser range finder except laser beam is steered in one or two axes
* LADAR can be used to created three dimensional maps of physical environment
* Negative obstacles (potholes, ditches etc) are problem for LADAR
* LADAR tend to be placed in front and high as possible
* Radar (Radio detection and ranging): uses electromagnetic waves to identify range, altitude, and direction of moving and fixed objects, used in larger UAV and USV systems
* Machine vision: process images from cameras to produce representation of objects in the world. Algorithms are used to analyze patterns of pixel values. Machine vision can be used to generate information that supports path planning

**Sensor Fusion (10/14)**

* Sensors provide information about state of system
* Encoders and other internal sensors allow for odometry
* GPS and external sensors allow for position based knowledge
* Sensor readings not always accurate
* Errors that measure change in state will add up over time
* UxVs using just odometry will eventually get lost
* Effect of uncertainty in sensors that measure state do not increase
* Jitter: Multiple readings from same or different sensors in same state produce inconsistent results
* Need to integrate sensor information to use sensors
* Fusion: process of combining information form multiple sources or readings
* Information fusion: Merging of information from heterogenous sources with differing representations
* Data fusion: process of integration of multiple data and knowledge representing the same real-world object
* Sensor fusion: combination of measurements from several sensors (Multi-Sensor Data Fusion)
* Information and data fusion are often employed as synonyms
* Data fusion is used for raw data (taken directly from sensors)
* Information fusion is used for processed data
* Types of data that can be combined:
  + Complementary: Information provided by input sources that represents parts of a scene that can combined to obtain more global information. Eg multiple cameras combined to produce a larger image
  + Redundant: Two or more input sources provide information about same target that can be fused to increment the confidence. Eg Overlap in cameras vision.
  + Cooperative: provided information is combined into new information that is typically more complex. Eg multi-modal (audio and video) data fusion is considered cooperative.
* One of most prominent applications of sensor fusion has been Navigation
* Navigation system: filter that integrates information from encoders, GPS, gyroscopes, accelerometers, and magnetometer measurements
* Challenging application because of the complexity and numerical burden (process data)
* If data is orthogonal (measures independent aspects) fusion is simple. Combine sensor readings into one sensor vector
* Redundant data involves multiple measurements of same properties
* Most UxV sensors provide complementary data. GPS and IMU measure position and velocity which are complementary
* When combining redundant data, we are interested in obtaining best estimate. Best estimate is having the lowest expected square error. If two or more readings are equally reliable, take mean (average) of readings

A close up of a clock

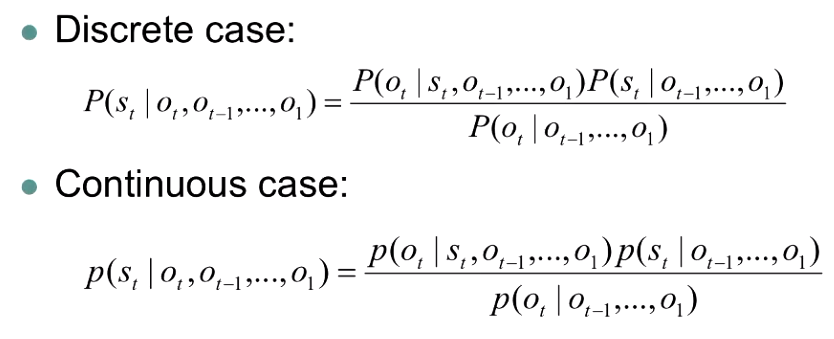
Description automatically generated

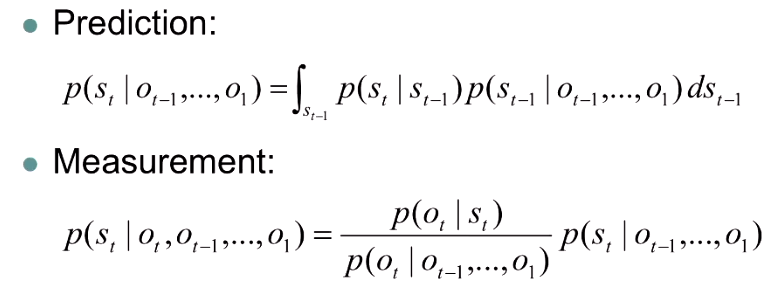
* y = Hx + n, H is covariance matrix (shown above), it models the uncertainty of sensors and correlations between errors of sensors

A close up of a logo

Description automatically generated

* GPS and laser provide position-related data
* Encoders and Accelerometers provide velocity-related data
* Lidar provides relative information data
* Need to model the effect of one type of information on the other to get best estimate (velocity influences position)
* Each sensor measurement can indicate a state with a certain probability P(s | o)
* Observation probability can be measured more easily, P(o | s) characterizes the precision of the sensor
* Uncertainty in relation between states over time can be captured in transition model P(st+1|st, at), also captures relation between velocity and position





* Benefits of Bayesian filter: Optimal estimates, no assumptions about distributions, uniform framework
* Problems with filter: Often computationally intractable, integral might not be analytically solvable
* Kalman filter is special case of recursive Bayesian filter following assumptions:
  + System and observation model are linear:
  + st = Ast-1 + Bat-1 + wt
  + ot = Hst + vt
* Uncertainty in the system and observation models are Gaussian
  + wt ~ N(0,Q)
  + vt ~ N(0,R)
* Kalman filter estimates posterior distribution in terms of mean and the covariance matrix
  + s’t = E[st] (apostrophe used for carat above s)
  + Pt = E[(st – s’t)(st – s’t)T]
* posterior distribution is a Gaussian distribution

Discrete Kalman filter

A screenshot of a cell phone

Description automatically generated

* Ball moving on track with unknown length and velocity and bounces perfectly at end example
  + we need position (x) and velocity (v)

A picture containing object

Description automatically generated

ot = st + vt

* UxS system is frequently not linear, Orientation introduces nonlinearity
* Extended Kalman Filter (EKF) relaxes requirement on linear models
* Uses Jacobian matrix, EKF does not always converge on correct solution

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of text

Description automatically generated

A screenshot of a cell phone

Description automatically generated

* Bayesian filtering/estimation provides unified framework for optimal estimation.
  + System module captures knowledge about system behavior
  + Observation module captures sensor characteristics
* If system module is linear and uncertainty is Gaussian, Kalman filter provides efficient solution
* If system is non-linear but uncertainty is Gaussian, Extended Kalman Filter can be used

**Path Planning (10/16)**

* Guidance with waypoints can be problematic
  + Waypoints have to lead to goal
  + waypoints need to avoid collisions
* Path and trajectory planning compute safe guidance in an environment as opposed to manually determining waypoints
* Path planning addresses problem of moving from initial position to goal without hitting obstacles. Assumes existence of map
* Shape of vehicle makes it more difficult to plan a path since no part of vehicle can hit object
* Configuration space is space spanned by degrees of freedom of the vehicle. The vehicle can be reduced to a single point in configuration space
* A vehicle can be reduced to a point by extending obstacles appropriately.
* Mapping obstacles into configuration space can be performed for any vehicle geometry, for vehicles that change shape, this gets more complex
* Basic Motion Planning Problem in configuration space is simplified path planning problem
  + Solid vehicle reduced to single point
  + Only static obstacles
  + Holonomic vehicle (if constraints are integrable into positional constraints that the variables are the system’s coordinates)
  + Only collision free paths are allowed
* Path planning approaches: Roadmap approaches, Cell Decomposition, Potential Field
* Roadmap approach
  + Construct a set of intersecting roads and determine the path by finding a sequence of roads that lead from start to goal
  + Similar to map-based navigation
  + First step requires construction of finite set of roads
  + Visibility graphs to create roads, connect all corners of obstacles and vehicle goal
  + Path search: Find a sequence of road segments that connects start and goal using best first search
  + Voronoi Diagram: Construct road segments that are equidistant to the two nearest obstacles. Connect start and goal to the nearest road. Find sequence of road segments that leads from start to goal
  + Approximate Voronoi diagram can be generated easier using distance propagation methods. Simultaneously starting wave fronts from all obstacles, record the points at which two wave fronts meet for the first time. In practice this requires a discretization of configuration space
  + Roadmaps provide efficient representation, Visibility graphs and Voronoi Diagrams have significant limitations like only work correctly in 2D configuration spaces and cannot be expanded to non-holonomic or dynamic trajectory planning
* Cell Decomposition
  + Decompose the workspace into freespace and obstacle cells. Vehicle can move freely in freespace. Path planning reduces to finding sequence of adjacent freespace cells with the first containing the current location and the final one containing the goal. Determining strategy to move within and between freespace cells
  + Cell decomposition approaches rely on construction of convex polygonal cells.
  + Convex cells allow for safe motion between two points in a cell along a straight line
  + Polygonal cells allow for safe transition between cells by moving through the midpoint of the shared edge between consecutive cells on a path
  + Cell Decomposition Approaches
    - Exact Convex Cell Decomposition
      * Construct convex freespace cells
      * Path search: Label cells and search for path
      * Path construction: Connect centers of cell connections
    - Approximate Regular Grid Cell Decomposition
      * Divide space into regular grid and mark cells that contain any piece of an obstacle
  + Cell decomposition are easy to compute. Approximate methods can divide space fast and allow for fast path calculations
  + Cell decomposition limitations:
    - Large spaces require large amounts of memory
    - Precision and completeness is limited by the resolution of the cells
* Potential Field
  + Construct a function U(q) over workspace, Q = {q}, of the vehicle that has large values at obstacle locations and small values at goal locations
  + Potential function defines surface which the vehicle can move downhill, away from obstacles and towards goal
  + Compute negative gradient, F(q) = -ΔU(q),
  + Potential field approaches:
    - Mixture of goal and obstacle potentials
      * Mixtures of potentials have local minima. There are situations where potential field-based path planner gets stuck
      * Navigation functions are potentials without local minima, require computation of entire potential
    - Manhattan distance
      * Discrete space into regular grid
      * Label goal cell with potential of 0
      * Propagate increasing distances to neighboring cells (cell will be labeled with number of cells away from goal cell)
  + Potential field approaches provide global path planning solution
    - Deviations from path are addressed
    - Movement direction at every point in space is easily computed as negative gradient of potential
    - Potential fields work in arbitrary dimensions
  + Computation of potential issues
    - Mixture potentials are easy to compute but have local minima
    - Navigation functions have no minima, but hard to compute
* Non-Holonomic Path Planning
  + Impose constraints on the path and do not fall into the basic path planning problem
  + Vehicle can only move forward and turn in place (cannot move sideways). Airplanes can not stop and turn, Non-Holonomic path planning does not work for airplanes
  + Non-holonomic constrains can be encoded in configuration space into the road map or into potential function
  + Manhattan Distance (See Echo360 October 23 how Manhattan Distance works)
  + Connectivity can be used to reflect non-holonomic as well as dynamic constraints
  + High-dimensional configuration spaces make it difficult to represent and search the space, Grid-based techniques require too much memory to represent the space or potential
  + Configuration space has higher dimensionality but non-holonomic constraints and collisions are easy to represent and determine
  + Cartesian space has low dimensionality but non-holonomic constrains and collisions are difficult to determine
  + To address high complexity of high-dimensional systems, roadmap based path/trajectory planners are usually used. Constraints are encoded in road segments. Only road segments that can be traversed will be constructed. Collisions computed during construction
* Random road construction used since exhaustive roadmap construction not possible in high-dimensional configuration
  + Rapidly Exploring Random Trees (RTTs) cover large spaces efficiently. Strength of heuristic regulates efficiency. Increased greediness (higher heuristic) results in faster search but increases chance path to goal not found. Higher heuristic value in complex environments not a good idea
  + Collision detection during road construction can increase cost significantly if space is dense with obstacles
  + Efficiency can be increased using bi-directional (build from start and goal and meet in the middle) road construction
  + Motion constraints are reflected in road segments. Both kinematic and dynamic constraints
  + First paths found tend to be inefficient, randomized path refinement can improve quality
  + RTTs address complex problems in high-dimensional configuration spaces
  + RTTs trade off quality of the found path against search and representation efficiency
* Path planning requires that the goal location and location of obstacles are known, any change in the map requires re-running path planner. Local path planning assumes perfect path tracking, any deviation from the path requires re-running path runner (safety margins can reduce this problem).
* Reactive navigation uses set of simple sensor-action rules to navigate an environment. Behavior-based control approaches are most common reactive architectures
* Reactive navigation does not rely on global map information
  + Behaviors range from reflexive (no map information) to reactive (local map/history information)
  + Fast reaction to unexpected map changes
* Reactive navigation suffers from local extrema in complex environments. Can be solved sometimes using randomization
* Hybrid reactive/planning approaches bridge the shortcomings of both approaches
* Hybrid approaches can plan with imprecise and incomplete maps

**Localization and Mapping**

* Mapping: Build a representation of the world from sensor readings given the position of the robot
* Localization: Determine position and orientation of the robot in the world from sensor readings and a given map of the world
* Simultaneous Localization and Mapping (SLAM): Determine both map and location simultaneously from sensor readings
* Notations:
  + Location: Xt = (x0,…xt)
  + Sensor readings: Zt = (z0,…zt)
  + Maps: m=(m0,…mt)
  + Actions: At = (a0,…at)
  + UxV motion model uncertainty: p(xt| Xt-1,At-1,m)
  + Sensor model: p(zt|Xt-1,Zt-1,At-1,m)
* Bayes filter is optimal estimator
  + Makes no assumptions about models
  + Allows to incrementally estimate model or location
  + Analytic solution of filter in general
* Types of Maps:
  + Occupancy grid maps
  + Landmark-based maps
  + Topological maps
* Bayes filter estimates probability distribution over locations
* Particle filter is a special instance of the Bayes filter that uses weighted sample sets as finite representation for a general probability density function
* In SLAM, Mapping assumes robot position is precisely known and Localization assumes the map is given
* Particle filter can be used for estimation of map in SLAM
* Particle filters can become unwieldy when environment is too large, Topological maps address this
* Topological map traits:
  + Avoid detailed geometry
  + Only contain map connectivity between landmarks
  + Good detectors for landmarks
  + Link between landmarks contains how to reach landmark
  + Only local space is represented geometrically

**Communications**

* UxVs need to transmit data to control center and control center needs to transmit commands to UxV
* UxV communications started at end of 19th century
  + Nikola Tesla created first wireless remote UxV in 1898, it was a boat
* Types of communication architecture:
  + One-way remote control platforms
    - simple to implement
    - low cost communications
    - significant need for human operators
    - requires line of sight
  + Two-way remote control platforms
    - allows operation without line of sight
    - higher power requirements
    - High bandwidth requirements
    - need for real-time communications
    - significant need for human operators
  + Partially autonomous platforms
    - Reduced real-time requirements
    - reduced operator load
    - transmission of selective information
    - need for significant processing power and sensor suites on platform
* Communication Modalities:
  + Wired communications
    - Very simple and high throughput
    - Relatively unaffected by environment
    - relatively secure
    - limits mobility of platform
  + Optical communications
    - well focusable to limit interceptability
    - works under water for reasonable distances and speeds
    - requires line of sight
  + Acoustic Communications
    - Sound can transmit information without need direct line of sight
    - works under water and in air
    - relatively low throughput
    - easy to intercept and interfere
  + Radio frequency communications
    - High throughput
    - can work without line of sight
    - can be focused to reduce interception and optimize signal loss
    - signal quality depends on environment conditions
    - Does not work under water
* Radio frequency most common communication modality except for submaries
* increasing RF carrier frequency leads to:
  + increased throughput
  + Better focus on signal
  + Reduced object penetration capabilities
  + increased straight line transmission behavior
  + increased sensitivity to weather
* RF communication modes:
  + Omni-directional
    - no antenna guidance
    - potential for signal broadcast
    - very fast signal strength
  + Directional point to point
    - large range with limited energy
    - reduced risk of interference, interception, and jamming
    - antenna guidance needed
  + Mixed mode (omni and directional)
    - Intermediate range
    - partial antenna tracking
  + Relayed communication (satellite)
* Networks allow more than two entities to communicate