Industrial Safety 2

Reflexive halt 2

Reflexive Halt Methods for Mobile Bases 2

Reflexive Halting for Manipulators 3

Recommendations for This Robot 4

Emergency Stop System 4

E-Stop Systems Used in This Lab 4

E-Stop Requirements for This Robot 4

Version 1 Prototype 5

Version 2 Design 6

# Industrial Safety

Mobile robots, particularly experimental platforms, require safety systems to disable them. These systems fall into two major categories. Reflexive halt and speed and separation monitoring systems keep robots from colliding with humans operators and workers interacting with the robot. Emergency stop systems are used to disable the robot in case it does something unsafe or unexpected. Both types of systems are necessary in an industrial robot, especially one that does not operate in a cage or with guards, neither of which are practical for a mobile robot.

[Write some stuff here about OSHA and NIST] [CITE Marvel]

## Reflexive Speed Limiting

A key safety feature of many robots is reflexive speed limiting or halting. Reflexive halting uses sensors to detect obstacles in the robot's path and prevent motion that would result in a collision. Reflexive speed limiting is particularly important for robots that operate in shared environments with people, because people can suddenly move into a previously clear robot path, endangering themselves and the robot. Whereas a planner will create safe paths for a robot around static obstacles, paths may be invalidated by moving obstacles such as people, and the planner may react too slowly to prevent a collision. Reflexive halting operates more quickly and on a lower level than planning and trajectory generation, and can override the velocity commands from the trajectory generator.

### Reflexive Halt Methods for Mobile Bases

Mobile robots often implement a reflexive halt as shown in ALGORITHM. For example, if a measurement source such as a LIDAR reports that there is an obstacle in front of the robot at close range, the robot's velocity would be limited to turning and reversing. This approach is effective, but can cause difficulties in navigating tight areas, where there may be enough space for a robot to navigate through obstacles if it does so carefully.

Given:  
 Sensor measurements, M  
Do:  
 for each measurement m in M:  
 if(dangerous(m)):  
 prevent motion in direction of m

Another approach to reflexive halting was described by Chad Rockey in his masters thesis [CITE ROCKEY]. This approach, called Reflexive Avoidance Plus, was developed for a smart wheelchair, which must operate in crowded areas around people. Reflexive Avoidance Plus uses velocity limiting rather than preventing motion altogether. When a sensor detects an obstacle in the robot's path, it limits the maximum velocity in that direction using a scaling function based on the distance of the obstacle from the robot. This approach prevents the robot from colliding with an obstacle (the maximum velocity is zero below a certain threshold distance), but allows low speed progress toward an obstacle. Because the robot approaches the obstacle at lower speed, it can safely get closer to obstacles.

### Reflexive Halting for Manipulators

In addition to the mobile base, ABBY's robotic arm poses a risk for collisions. In industrial situations, manipulators are kept inside safety cages to prevent people from interfering with them or getting injured. A safety cage is not a possibility for a mobile manipulator, so another solution is necessary. A reflexive halt for the manipulator allows it to operate safely in the presence of people and obstacles without a safety cage.

Like the mobile base planner, the planner for the arm generates collision-free paths. However, the planner for the arm does not replan at all once it commits to a trajectory. The trajectory is generated and sent to the IRC5 for execution, and then ROS waits for the trajectory to be executed. If something enters the path of the trajectory, the robot does not alter the current trajectory. This makes the robot unsafe to operate around humans above a certain joint speed.

Rethink Robotics, with their robot Baxter, [CITE BAXTER] solved the problem of operating an industrial robot without a safety cage with a mechanical and software solution that relied on force feedback and serial-elastic actuators. Because all of Baxter's joints are elastic and its arms are so light, it can safely collide with people and obstacles. It also uses force feedback in its joints to detect these collisions and become passive, allowing people to push it around. Because ABBY's robotic arm does not have serial elastic actuators or force feedback in the joints, this solution is not possible.

Using the Kinect sensor, it would be possible to implement one of several possible reflexive collision avoidance methods on ABBY. One method would be to halt arm motion when an obstacle enters the arm's work envelope and suspend arm motion until the obstacle leaves the work envelope. Although this is arguably the safest solution, it can cause the robot to become stuck in the stopped state. If an inanimate obstacle is brought into the robot's work envelope and left there, the robot will never re-enable the arm.

To resolve this problem, the reflexive halt behavior can be augmented as shown in ALGORITHM. In this algorithm, the currently planned path is repeatedly checked for dangerously close object (such as people) until execution is completed. If an object enters the dangerous area, the robot stops execution of the trajectory and waits. If the obstacle leaves the area before a timeout is reached, the robot resumes execution of the trajectory. If the obstacle does not move, the robot will attempt to retry planning to move around the obstacle to accomplish its goal. This reflexive halt algorithm has two advantages over the naïve algorithm described in the previous paragraph. First, it does not stop for obstacles that enter the work envelope but do not interfere with the planned motion. This allows humans to work alongside the robot and interact with it by giving it objects or taking objects from it. Second, it will plan around stationary objects that enter the work envelope, allowing a person to leave an object in the work envelope without stalling the robot.

Given:  
 Current Plan P, Measurements M  
Do:  
 for each state s in P, measurement m in M:  
 if(dangerous(m, s)):  
 halt  
 wait for obstacle to move or replan

The National Institute for Standards and Tests (NIST) has developed an algorithm to determine a safe separation distance S for a human to approach a robot. The equation [EQUATION], where KH is the speed of a moving human, KR is the speed of the robot, TR is the reaction time of the human, TB is the braking time of the robot, B is the robot braking distance, and C is a distance to account for measurement uncertainty. [CITE NIST]

NIST researchers used this equation, LIDAR scanners, and a Kalman filter to track humans moving through the robot's work envelope and determine whether a human had entered a danger zone around the robot (if the distance from the robot to the human is less than S). This method completes algorithms [ALGORITHM] by filling in the dangerous() function.

### Recommendations for This Robot

At this time, ABBY does not implement a reflexive speed limit for the base or the manipulator. Because both operate at very low speeds, the robot does not pose a safety threat to the operator, and can be easily stopped with the emergency stop system described below. However, this solution is not adequate for an industrial application.

Although this robot does not implement reflexive speed limiting, the mobile base reflexive halt method was tested on a robot of similar size and speed that was designed to operate safely alongside humans. As such, it is appropriate for this application.

## Emergency Stop System

### E-Stop Systems Used in This Lab

The Case Mobile Robotics Group has used a few emergency stop systems in its robots. All of the HARLIE-class robots developed for the Intelligent Ground Vehicle Competition used a commercially-available wireless relay system from Remote Control Technologies. This system, shown in FIGURE, consisted of the remote control relay in series with an onboard disable switch and a second relay switched by an active-high enable signal from the cRIO, which is software-controllable. These three switches (one manual and two relay) control the current through the coil of a solenoid, which in turn switches the power to the motor controller on and off. This system has one critical flaw, which is that there is no “heartbeat” from the wireless remote to the remote control relay. This means that if the battery in the wireless remote dies or the radio communication is lost between the wireless remote and the robot, there is no way to remotely stop the robot, nor is there any indicator to the operator that the robot cannot be wirelessly stopped.

OTTO the smart wheelchair used a custom remote mode switching system designed to interface with the Arduino-based control system used to control the wheelchair drivetrain. Using a pair of XBee 2 Pro wireless network modules, a GPIO signal is transmitted from a remote control unit to the input of an Arduino on the wheelchair, disabling the autonomous functions of the wheelchair when a button is pushed on the remote. Because the Xbee wireless modules' GPIO mirroring has a programmable timeout and default output state, this system automatically disables autonomous functions if communication is lost between the remote and the robot. However, this system does rely on an Arduino microcontroller and was not designed as an emergency stop system, but as a switch between autonomous operation and normal (joystick) operation of a wheelchair.

### E-Stop Requirements for This Robot

This robot has several requirements that motivated the development of a new emergency stop system combining the merits of the HARLIE-class stop system and the OTTO remote switching system. First, the emergency stop system needs to be able to switch the high current, 24 volt rail providing power to the motor controllers. Second, the system needs to be able to activate the 24 volt emergency stop input on the IRC5 robot controller, which must be electrically isolated from the rest of the robot's DC electronics. Third, the system must monitor four sources, stopping the actuators if any of them are disabled:

1. 5 volt active-high enable signal from the cRIO, which is controlled by the ROS software
2. 24 volt active-high emergency stop signal from the IRC5, which is controlled by the emergency stop switches on the IRC5 and FlexPendant and by RAPID software.
3. Twist-lock stop switch mounted on the robot (The robot is disabled if the switch is opened or disconnected.)
4. Wireless remote control with a heartbeat signal of at least 1Hz

Fourth, the system should be implemented entirely in hardware for safety reasons. Software faults in a safety system are unacceptable and adequate testing of a software system would be too time-consuming for this project. Fifth, the remote control unit should have some feedback as to the state of the four emergency stop sources described above.

### Version 1 Prototype

Given the requirements, an emergency stop system was designed and fabricated using printed circuit boards. The schematic of the system is shown in FIGURE. The system consists of two circuits, a remote control and an emergency stop circuit on the robot.

SCHEMATICS GO HERE

The remote circuit uses an XBee radio module's GPIO mirroring function is used to transmit the state of the emergency stop button to the emergency stop circuit on the robot in the same manner it was used on OTTO. This system also uses the GPIO mirroring function to send the states of the other emergency stop sources to the remote, where they are displayed on LEDs. Because a twist-lock style emergency stop button was not available, an S-R latch was used to latch the state of a normally-open momentary pushbutton, requiring that the remote be powered off and back on again to reset the wireless emergency stop.

The emergency stop circuit on the robot has inputs for the onboard emergency stop button, the cRIO's enable signal, and the emergency stop output of the IRC5. The input from the IRC5 goes into an optoisolator IC because the IO on the IRC5 is floating relative to the rest of the robot's DC systems. A 7400 series AND IC is used to generate logic signals to enable the drive base and the IRC5's emergency stop input. The drive base logic signal controls a Darlington transistor, which in turn switches the coil of a solenoid that controls the drive base in the same manner as on HARLIE-class robots. The IRC5 output logic signal switches the 24v General Stop input of the IRC5 using an optoisolator IC.

These circuits were prototyped and installed on the robot. The input from the IRC5's emergency stop was defeated by installing jumper J1 because the output had not been configured in RAPID software. Additionally, testing showed that the 4N35 optoisolator used to switch the IRC5's emergency stop could not switch enough current to enable the emergency stop circuit, causing the IRC5 to go into General Stop mode seemingly at random. This function was defeated by disconnecting the IRC5   
Stop output and shorting the General Stop input of the IRC5. These two changes completely decouple this emergency stop circuit from the IRC5, meaning it no longer meets requirements 2 and 3b described above. Furthermore, the wireless link between the XBee modules proved unreliable, causing the system to momentarily switch into emergency stop mode seemingly at random. Extensive bench testing of the system suggests that this problem is caused by an insufficiently reliable power supply to one or both of the XBee modules. In order to make the system usable, the wireless emergency stop was replaced with a twist-lock style emergency stop button on a ten foot wired tether. This modification means that the system no longer meets requirements 3d and 5 described above. The system does reliably control the power to the drive base, providing a level of safety for the robot, but revisions are required to make the system function as specified.

### Version 2 Design

A revised version of this emergency stop circuit was designed and portions of it prototyped, but it has not been tested. This version should fix the problems discovered in the first version of the emergency stop.

SCHEMATICS HERE

To integrate the system with the IRC5, the optoisolator on the output of the emergency stop circuit was replaced with a relay module, which will more reliably switch the General Stop input of the IRC5. To complete integration with the IRC5, the RAPID software must be modified to output the current General Stop state to a GPIO, which must be connected to the IRC5 input of the emergency stop circuit.

To solve the wireless communication issues, the power supply in the remote was replaced with a 3.3 volt boost supply, which should be much more reliable, and bypass capacitors were added to the power rails of the XBee modules on both the remote and the emergency stop circuit. Testing has shown that the XBee modules are reliable when a sufficiently clean and reliable DC supply is available to power them.

In addition to solving the problems described above, some small changes were made to improve the circuit. To reduce the power consumption of the emergency stop circuit and reduce the heat produced by the onboard power regulator, the Darlington transistor used to switch the coil of the drivebase enable solenoid was replaced with a MOSFET circuit that performs the same function. To make the system easier to use and more reliable, the momentary switch and latch used on the previous version was replaced with a twist-lock style emergency stop switch.

Although this system has not been constructed, its constituent parts have been tested individually. The MOSFET switching circuit has been confirmed to work with a resistive load equivalent to the coil resistance of the solenoid used to switch the drive base power rail. The power supply circuit in the remote has been tested and provides a reliable 3.3 volt power supply from a pair of AA batteries. The use of a relay instead of a transistor to control the General Stop input is in line with recommendations from ABB's documentation, and the relay used meets the requirements. If the necessary components can be acquired for this emergency stop circuit, it should be able to meet all of the requirements described above.