Exception Handling for Sensor Fusion

Gregory T. Chavez and Robin R. Murphy
Department of Mathematical and Computer Sciences
Colorado School of Mines
Golden, CO 80401-1887
gchavez@mines.colorado.edu rmurphy@mines.colorado.edu

Abstract

This paper presents a control scheme for handling sensing failures (sensor malfunctions, significant degradations in performance due to changes in the environment, and errant expectations) in sensor fusion for autonomous mobile robots. The advantages of the exception handling mechanism are that it emphasizes a fast response to sensing failures, is able to use only a partial causal model of sensing failure, and leads to a graceful degradation of sensing if the sensing failure cannot be compensated for.

The exception handling mechanism consists of two modules: error classification and error recovery. The error classification module in the exception handler attempts to classify the type and source(s) of the error using a modified generate-and-test procedure. If the source of the error is isolated, the error recovery module examines its cache of recovery schemes, which either repair or replace the current sensing configuration. If the failure is due to an error in expectation or cannot be identified, the planner is alerted. Experiments using actual sensor data collected by the CSM Mobile Robotics/Machine Perception Laboratory's Denning mobile robot demonstrate the operation of the exception handling mechanism.

1 Introduction

Since sensing is the only medium through which an autonomous mobile robot directly obtains data about the world, it is imperative that it be both certain and robust. Uncertainty results from a number of sources, including random sensor errors, systematic sensor errors, and occlusions. Robustness permits the sensing system to either maintain certainty or, at worst, degrade gracefully in the face of unanticipated problems such as sensor malfunctions, changes in the environment (e.g. the lights go off), or planning errors (e.g. robot is not in the right location). These unanticipated problems can also be referred to as sensing failures, indicating the inability of the perceptual process to provide reliable perception.

Sensor fusion systems which can combine the data extracted from multiple sensors types have been proposed as a means of reducing uncertainty. They are expected to be more certain because they typically provide broader coverage of the percept over larger ranges of environmental conditions and viewpoints. Unfortunately many existing sensor fusion systems do not effectively address the need for robustness.

The approach taken in this paper treats robust sensor fusion as a form of *exception handling*, where sensing failures are exceptions to the fusion process which must be dealt with immediately in order to allow the process to resume production of reliable percepts. Exception handling for sensor fusion is especially challenging for two reasons. First, the use of multiple types of sensors in an open world with the ensuing complex interactions make it impractical to construct a complete causal model relating all possible failures to its source or cause. Second, exception handling should be fast. If a mobile robot does depend on sensing to accomplish a task, it must either suspend action on the task until the failure is resolved or continue on blindly and risk damaging itself or the surroundings.

This paper presents a control scheme and representations for sensor fusion exception handling for use with SFX, called SFX-EH. However, SFX-EH is separate from the sensor fusion mechanism in SFX and as a result can be used with any system capable of detecting a sensing failure. Detection of failures will not be addressed here; the reader is directed to [7]. The remainder of the paper is organized as follows. Related work is discussed in Section 2 and the

SFX-EH architecture in Section 3. Demonstrations of SFX-EH with sensor data collected from the CSM mobile robot are described in Section 4. The paper concludes with a summary and description of future work.

2 Related Work

While little work in exception handling for sensor fusion has been published, more general work in error detection and recovery has been performed in other AI domains, such as planning. Planning systems deal with how to order a set of tasks based on predictions of what the state of the world will be at execution time for each step. Since robots are a physical implementation of a planning system, these exception handling techniques utilized for planning have some relevance for sensor fusion. This section discusses five papers which address some aspect of exception handling for either robotics or planning.

2.1 teleSFX

Murphy's modification of SFX for teleoperations (teleSFX) [6] implements an exception handling mechanism for sensor fusion. In order to reduce communication bandwidth, the remote machine under teleSFX attempts to autonomously repair or replace sensors using the generate and test paradigm to pinpoint the cause of the sensing failure. A simplified form of an information matrix [8] represents the proper operating and interaction characteristics for a particular arrangement of sensors. This matrix also is used to determine if alternative sensor(s) could be substituted.

TeleSFX indicates that the generate and test paradigm for error classification is viable. The modified information matrix [8] used to model the relationships between sensors, however, is too complex to extend to domains with larger numbers of heterogeneous sensors.

2.2 GTD and CHEF

Generate Test and Debug (GTD) [1] and CHEF [3] are planning systems which use similar approaches to error recovery in the geological and cooking domains, respectively. GTD uses a generate and test procedure supplemented with a debugger to repair denied hypotheses, creating new ones. One interesting aspect of GTD is that it challenges each of the assumptions made in the interpretation. This is useful for sensor fusion systems such as SFX which use a static plan, because of their dependence on assumptions. A static plan is one which is configured at the beginning of a task and is not permitted to change during execution. Perceptual systems which operate according a static plan tend to be efficient, but will produce errors if the assumptions made during construction are violated.

CHEF is a case-based planner that operates in the cooking domain. It generates plans for new situations by altering stored plans which only partially match the present situation. CHEF contains a plan modification library which constrains steps that can be substituted for existing steps. This is useful because it suggests a way for exception handling to rapidly recover from sensing failures by modifying or repairing the current configuration rather than by starting over.

A weakness of both GTD and CHEF is that they rely heavily on domain independent repair schemes and knowledge. This type of knowledge is not readily available in most sensor fusion domains where it is difficult to predict complex interactions between sensors. Furthermore, an exception handling module must be fast in order to restore sensing as soon as possible. Domain independent knowledge tends to cause the handler to make many guesses until the cause can be narrowed down and identified. Although it limits extensibility, basing exception handling on domain dependent knowledge appears to be a more realistic choice.

2.3 Weller et al sensor modules

Weller, Groen, and Hertzberger [9] deal directly with detecting and recovering from sensor errors. Weller et al's sensor system is broken down into sensor modules. Each module contains tests which verify the input data, the internal data used in computation, and the output data from the execution of the algorithms used to process the raw data.

Environmental conditions dictate whether certain tests are to be performed or not. Error recovery is handled by modifying the raw sensor data or the algorithms that manipulate the raw data.

The primary advantages of this approach are that it emphasizes modularity and the use of local expert knowledge. This local expert knowledge is one way of encoding the domain dependent information needed to streamline the exception handling process. However, this approach has a significant disadvantage for a sensor fusion application. It repairs exceptions by either modifying the raw sensor data or the sensor data processing algorithms. Adjusting invalid data is not an acceptable method for recovery, mainly because there is no obvious method for adjusting the invalid data. Further, the method should be expanded to include modifications or removal of sensors themselves if they are malfunctioning.

2.4 Firby and Hank's hybrid deliberative/reactive architecture

Firby and Hanks [4] have presented a planning architecture that addresses exception handling. This architecture joins Firby's reaction action packages (RAPs) [2] with Hanks' deliberation system. The action component handles the execution of plans and addresses exception handling for plan failures. Two types of plan failures can occur: an atomic action fails or there are no applicable methods for a RAP corresponding to a plan step. Firby and Hanks indicate that either an alternate method is selected or the same method is run again. A plan step is re-tried until satisfied or until the system assures itself that no available method can succeed. The system signals failure if the same method is run twice in the same world state without success. The failure is then passed up to the calling plan step which then deals with it.

The contribution of their approach for sensor fusion is to have an exception handling mechanism propose a number of candidate "next steps" and then select one based on satisfaction of pre-conditions for those steps. One major drawback is that no formal error classification scheme is presented; the system recovers by either choosing another method randomly whose pre-conditions are currently satisfied or by running the same method again. Given the need for rapid recovery, it is preferable to modify the current plan to save the start up costs associated with instantiating different sensors.

3 SFX-EH

The major goal of SFX-EH is to provide an exception handling mechanism for SFX which will respond quickly enough to preserve reactivity in an autonomous mobile robot. By the domain dependent nature of exception handling, SFX-EH is dependent on the organization of SFX; however, the error classification and recovery techniques used in SFX-EH can potentially be transferred to other architectures.

3.1 Sensor Fusion Effects Architecture

SFX-EH is intend to work within the confines of SFX; therefore, some background on SFX is needed before SFX-EH can be detailed. The SFX control scheme views sensor fusion as being a special case within a general perceptual process. A perceptual process is instantiated by a deliberative planner and is responsible for supplying a motor behavior with an accurate percept. The percept is determined by the planner. As shown in Figure 1, it begins by using percept-specific knowledge to select a set of sensors and sensing algorithms which are expected to provide acceptable certainty for the projected environment (*configuration* mechanism). The *execution* mechanism collects the sensor observations from the *sensor data blackboard* and fuses them into a percept. The perceptual process also monitors for sensing failures during execution, and if one is detected, it suspends execution and invokes the exception handling mechanism. The exception handler can either *repair* the sensing configuration (e.g., replace malfunctioning sensors with backups, eliminate some sensors altogether, etc.) or request that a new configuration be constructed.

The original implementation of SFX concentrated on the execution mechanism, which uses *fusion states* to compensate for minor disagreements (or conflicts) between sensors. Uncertainty is propagated using a Dempster-Shafer formalism, which exploits the weight of conflict metric inherent in Dempster's rule of combination. The exception handling mechanism was not implemented at all.

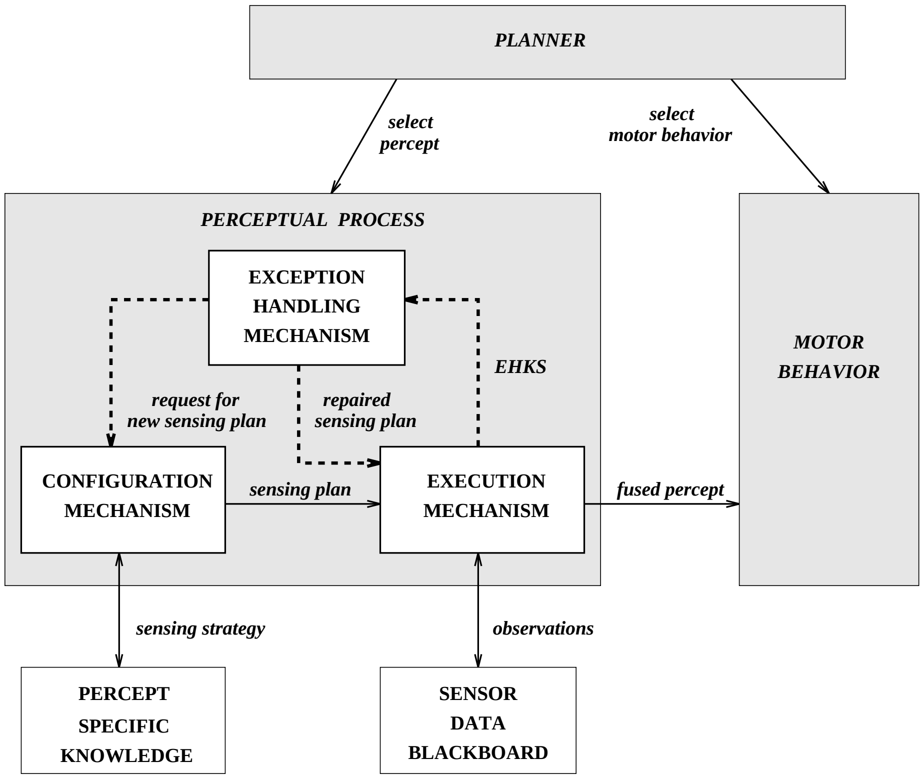

Figure 1: Layout of SFX.

One key aspect of SFX which is exploited by SFX-EH is the sensing strategy and sensing plan representation. They are important because SFX-EH attempts to repair sensing plans and if that is not possible, it then requests that the configuration mechanism select a new plan from the sensing strategy. The sensing strategy is a collection of sensing plans, activation conditions for each sensing plan, and an initial sensing plan pointer. There are two types of activation conditions: *sensor availability*, represented as a vector of bitmasks which delineate the required sensors for the corresponding sensing plan, and *environmental pre-conditions*, which specify the operating attributes of the environment needed to satisfactorily use the sensing plan. The initial sensing plan pointer points to a default sensing plan, if known.

The sensing plan represents all necessary information to detect a particular percept. It is a collection of of sensor-based models (called descriptions) of the percept which are built up from features and the specific sensors assigned to observe each description. Normally, a single sensor type is used for a single description, but multiple sensors of the same type can be employed to redundantly sense a single description. Each description-sensor pair is called a *body of evidence*. The sensing plan is represented by a directed acyclic graph. The root of the graph is the percept node which represents the total belief in the synthesized measurement of the percept. The next level (children of the percept node) contains a subgraph representing the collection of features (or model) of the percept being observed by each sensor.

3.2 SFX-EH

The basic architecture of SFX-EH is shown in Figure 2. When the execution mechanism detects a fusion failure, the exception handling mechanism is invoked and the relevant information is encapsulated in the *exception handling knowledge structure (EHKS)*. The error classification module uses EHKS to isolate the likely cause of the state failure. Next, the error cause is sent to the error recovery module if it can be handled locally, otherwise the planner must take action. If it can be handled locally, the error recovery module selects and executes a recovery scheme.

The actual design of the exception handling mechanism for SFX-EH is based on five assumptions:

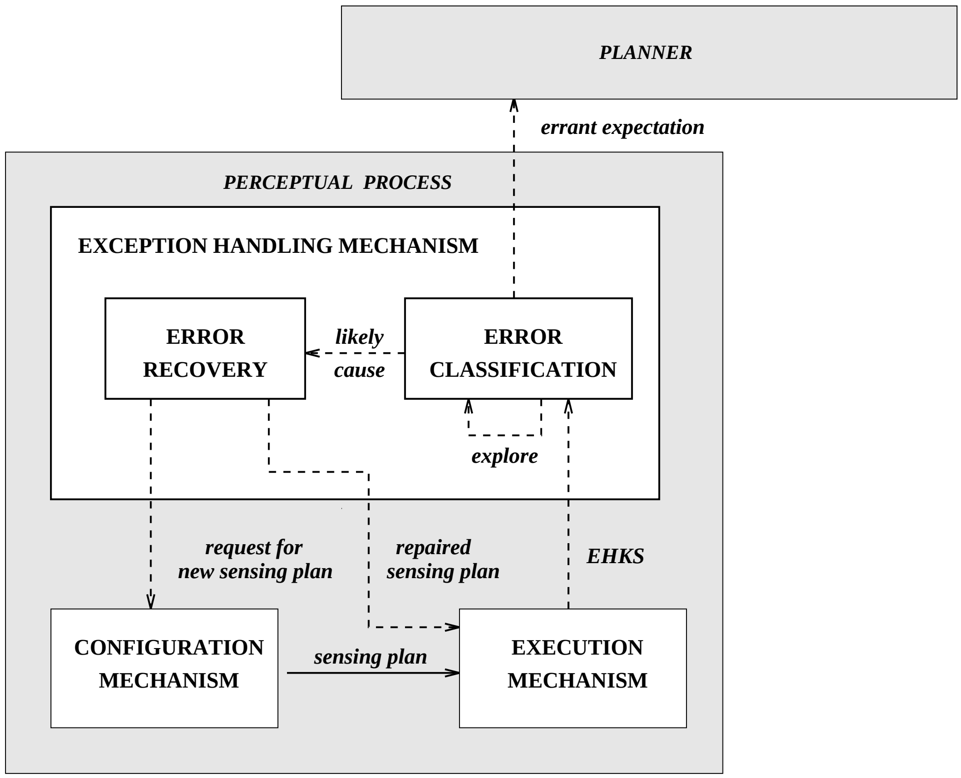

Figure 2: Layout of SFX-EH.

1. Sensing failures result from either sensor malfunctions, changes in the environment which degrade sensor performance, and/or errant expectations put forth by the robot planner.

2. The ways in which sensing failures manifest themselves may not be fully understood for an application; therefore the exception handling mechanism may not have access to a complete causal model of sensing failures. This importance of this assumption was illustrated in [7], where SFX found there was no direct correspondence between the cause of a sensing failure and whether it was detected as missing, highly uncertain, or conflicting data.

3. Exception handling can make use of the sensor data which led to the failure, on-board sensor diagnostics (either hardware or software), or it can actively use a sensor not in the sensing plan to externally corroborate whether a suspect sensor is still functioning correctly or that environmental pre-conditions remain satisfied.

4. The deliberative planner which instantiated the perceptual process is responsible for resolving failures due to errant expectations.

5. If the exception handling mechanism cannot identify the cause of a sensing failure, it will assume errant expectation. This is reasonable because the planner is responsible for any situation which cannot be resolved by the local exception handler.

3.2.1 Error Classification

Error classification is necessary because SFX-EH assumes that only a partial causal model of sensing failures is available. If the exception handling mechanism had access to a complete causal model, then the error classifier component could be eliminated.

The error classification module categorizes failure into three types. The first type is *sensor malfunctions*, which arise because one or more sensors are malfunctioning. The second is referred to as *environmental change*, where one or more sensors can be showing poor performance because the environment has changed from its condition at configuration time (e.g., lights turned off). This is an important distinction from a malfunction because it means that an environmental change can impact multiple sensors. The third category of errant expectation occurs when the performance is poor because the planner has set up an error in expectation. An error in expectation means that the percept which the robot is attempting to locate is not there because it has moved or the robot is not actually located where it thinks it is.

The error classification module uses the EHKS passed by the execution mechanism to identify the type of error. The EHKS is a frame that contains six slots. The **failure step** slot is a flag that describes the execution step where the state failure occurred. The **errors** slot gives the state failure condition encountered. The **bodies of evidence** slot is list of frames, each of which holds sensor data. The **environmental pre-conditons** slot also holds a list of frames, each of which describes the attribute of the environment (if any) which serves as a pre-condition for using that sensor, the expected value of that environmental attribute for acceptable performance of the sensors, and pointers to other sensors which share the same environmental precondition. The EKHS contains this so it can challenge the environmental pre-conditions.

Error classification in SFX-EH is accomplished by a generate and test algorithm similar to that popularized by Dendral [5]:

1. *Identify suspect bodies of evidence.*
   In cases like missing evidence, the suspect body of evidence is obvious. However, in other types of sensor fusion failures, the cause may not be apparent. For example, when two sensors show a high conflict or disagreement it is not clear which is the culprit. In that case, SFX-EH labels all sensors as suspect and relies on testing to isolate the real culprit.

2. *Generate an ordered list of possible hypotheses explaining the failure.*
   Associated with each hypothesis is the appropriate test, either a sensor diagnostic or an environmental pre-condition challenge. Note that the tests are application and sensor dependent.

3. *Perform the tests to see if the hypothesis can be confirmed or denied.*
   Once the test list has been generated, the tests are performed in the order that they appear in the list. Additional tests may be inserted in the list to confirm that any additional sensors being used to actively confirm a hypothesis are operating correctly.

4. *If hypothesis is confirmed, quit; otherwise return to step 1.*

Generate and test has several advantages for this application. Since it does not require formal operators for generating hypotheses, SFX-EH can use a rule-based method to select from a list of candidate hypotheses. The selection of a hypothesis from a list of candidate hypotheses is better for diagnosing errors for a one-step plan. A one-step plan is a plan consisting of one time step. The *plan* required for sensor fusion is a configuration of sensors which collect data in one instant in time which qualifies as a one-step plan. Further, the candidate hypotheses reflect the partial causal model and are easy to expand and modify.

The major disadvantage of generate and test is that it can be time-consuming if there is a large problem space and all possible hypotheses must be generated. As noted in [5], if the problem space can be constrained, generate and test can be efficient. SFX-EX overcomes this disadvantage because the list of candidate hypotheses does constrain the problem space. However, this constraint opens up the possibility that the classifier will encounter a situation it cannot resolve. In keeping with the philosophy of performing fast identification and repairs at the perceptual process level and sending difficult problems to the planner, if SFX-EH cannot identify the cause as being either a sensor malfunction or environmental change, it assumes an errant expectation and triggers replanning.

The generation, ordering, and execution of the tests has a direct impact on the efficiency of the system. Correctness is ensured by ordering the tests so that whenever an additional sensor is actively used (i.e. to test for an environmental change), that sensor's reliability must be checked out by a sensor diagnostic function first. The termination condition is that the tests are performed until either all the tests have been performed or an environmental change has been confirmed. The reason that the testing does not terminate upon a confirmed sensor failure is because an environmental change can make sensors appear to be malfunctioning and cause a sensor diagnostic function to incorrectly classify a sensor malfunction.

3.2.2 Error Recovery

A sensor fusion exception handling error recovery module should make use of a cache of recovery schemes (cases). The use of cases allows for instant mapping of error cause to recovery scheme based on the error classification. The recovery scheme consists of functions used to repair or replace the sensing configuration. The recovery schemes attempt to repair the sensing configuration first, but if that is not viable, they signal for the configuration module to select a new plan from its sensing strategy. Two types of sensing configuration repair are available: sensor removal and sensor replacement. Note that the graph structure of the sensing plan supports repair operations; both the pruning of a subgraph associated with a "bad" or the attachment of subgraphs for new sensors can be accomplished by simply updating pointers.

SFX-EH supports limited error recovery; it supports sensor removal and recommends sensing plan replacement for the case where sensor removal would result in the removal of all the sensors in the sensing configuration. SFX-EH also has a one-to-one relationship between an error cause and a recovery routine. Ideally, multiple recovery schemes should correspond to an error cause, and the error recovery module should reason which is the best recovery scheme to employ based on the robot and the world states. Sensor replacement, which is not supported by SFX-EH, is a more complicated but desirable form of error recovery than sensor removal. The simplified implementation of error recovery was chosen to reduce the scope of development of SFX-EH and allow for complete development of the error classification module.

4 Demonstrations

This section summarizes thirteen experiments which demonstrate the proper execution of each portion of SFX-EH and the error free execution of SFX after SFX-EH has corrected an exception. Demonstrations used real sensor data taken by the CSM Mobile Robotics/Machine Perception Laboratory's Denning MRV-III mobile robot for exception handling.

4.1 Methodology and Equipment

Demonstrations were conducted for the domain of indoor navigation, in particular for a scenario where a mobile robot was acting as a security guard or a delivery robot. In such a domain, the robot moves throughout a building into and out of rooms. This entails recognition of the open doorway, and its precise location, which is a perceptually intensive task that may require the use of multiple sensors to reduce uncertainty. For example, ultrasonics may be used to reduce uncertainty about which line segments correspond to a door frame.

The perceptual process for these experiments was responsible for identifying the **hallway-door** shown in Figure 3. This door is the actual door to the hallway from the MR/MP Laboratory. Since the focus of this work is on the direct operation of the exception handler rather than on the configuration mechanism, the sensing strategy consisted of a single sensing plan. The sensing plan for initially recognizing the door used easy to extract features from three sensors. Ultrasonics detected the corner, the color camcorder looked for the red fire extinguisher, and the b&w video camera extracted the strong vertical lines corresponding to the door frame and walls. Note that this set of descriptions of the door allows it to be identified even if the door is closed.

The sensor availability activation conditions for this plan were that all three sensors be available. The environmental pre-condition used for the demonstrations is light intensity. Light intensity is the only environmental pre-condition used for the demonstrations because a change in light intensity (the lights going out) is the known environmental change that impacts any of these sensors for an indoor environment.

Experimental data was collected from *Clementine* (Figure 4), a Denning MRV-3 mobile robot. Three types of sensors on Clementine were used: a black and white Elbex video camera, a Sony color camcorder, and a ring of 24 Polaroid ultrasonic sensors. Visual data from both cameras and ultrasonics data were collected with the robot at a

Figure 3: The **hallway-door** percept as seen by the color camcorder.

Figure 4: Clementine, the MR/MP Laboratory's Denning MRV-III robot.

single location near the door. The data was collected six feet from the door on a line 45 degrees from the plane of the closed door. The single data location was considered sufficient because only one data location is needed to generate sensing failures. Visual images originally were taken with the lights on and off. Due to the presence of the emergency lights in the lab, however, the room was not dark enough with the lights off to cause a detectable failure, so the images were darkened by performing two histogram compressions on each of the images.

Because SFX and all of its accompanying sensor data processing algorithms were not integrated with Clementine at the time of testing, failures were simulated by interfacing an interactive version of SFX, which allows the user to input the belief function that would have been computed by SFX for each body of evidence. The Shafer belief functions are entered as a set of values between zero and one for the *support*, *against*, and *dontknow* values that make up a body of evidence in each description/sensor node in the sensing plan. A value of 1.0 for *dontknow* results in a pre-processing step state failure indicating missing data. Missing data actually refers to completely uncertain data. A value between 0.7 and 1.0 for *dontknow* results in a pre-processing step state failure for high uncertainty. The threshold values for missing and highly uncertain data are part of the sensing plan for the **hallway-door** percept. A fusion step state failure for high conflict is simulated by entering conflicting values of *support* and *against* between bodies of evidence.

4.2 Demonstration of Entire SFX-EH System

The input vision data for this experiment simulated dim lighting. In this instance, the bodies of evidence contributed by the ultrasonics and color camcorder gave high belief for the **hallway-door** because the ultrasonics were unaffected and the camcorder compensated with its auto-gain function. However, the b&w video camera could not adjust for the dim lighting and did not detect strong vertical lines. This caused the b&w body of evidence to report a high belief against the presence of the door, in conflict with the other two bodies of evidence. The SFX simulator signaled a high conflict failure. SFX-EH classified the error correctly, repaired the plan, and resumed execution which then reported a fused belief for the presence of the door.

Figure 5a. shows the screen output of SFX-EH after beliefs computed by the SFX simulator have been entered. SFX detected a high conflict sensing error, which by definition involves all sensors. Figure 5b. shows the abbreviated output of SFX-EH during error classification. First, it correctly identified all bodies as being suspect. Next, it generated the ordered list of hypotheses. First to be tested was whether an environmental pre-condition had changed. Recall that environmental pre-conditions are tested first. The only environmental pre-condition for this sensing plan was the light intensity. Next on the list were individual sensor diagnostics. The final step of error classification is test execution. For this demonstration, the environmental pre-condition was challenged and confirmed an environmental change. Upon confirmation of the environmental change, test execution terminates. Recall that the termination condition for test execution is to run tests until an environmental change is confirmed or all tests in the lists have been executed, whichever comes first.

Figure 6a. shows that SFX-EH responds to the error with the recovery scheme for the b&w camera when its environmental pre-condition is violated. That recovery scheme repairs the sensing plan by pruning the body of evidence from the b&w camera from the sensing plan. After error recovery is complete, the SFX simulator is called with the repaired plan. As shown in Figure 6b., it performed fusion and did not encounter any sensing failures. It returned a high belief that the door is indeed present.

4.3 Other Demonstrations

Additional tests of the error classification module were performed, concentrating on the selection and ordering of the candidate hypotheses (tests) and correct termination. Only one case was showed a failure, when the generation of tests and ordering strategy resulted in SFX-EH missing an ultrasonics malfunction because of a previously confirmed environmental change. On the other hand, one of the demonstrations showed SFX-EH exceeding expectations by correctly identifying a simultaneous malfunction of multiple sensors (both the b&w camera and ultrasonics).

Tests which concentrated on the error recovery module were also performed. Specific error recovery schemes successfully demonstrated were: pruning of description/sensor nodes that correspond to errant bodies of evidence,

```
Enter belief for Cornerness (Ultrasonics):
Support: 1.00
Against: 0.00
Don't know: 0.00
        Failure Status: HIGH CONFLICT

Enter belief for Edges (Black and White Camera):
Support: 0.00
Against: 1.00
Don't know: 0.00
        Failure Status: HIGH CONFLICT

Enter belief for Color (Color Camera):
Support: 1.00
Against: 0.00
Don't know: 0.00
        Failure Status: HIGH CONFLICT
```

a.

```
*******************************************************
Welcome to error classification!!!
Body of evidence 0:  Sensor type is ultrasonics
Body of evidence 1:  Sensor type is black and white camera
Body of evidence 2:  Sensor type is color camcorder

STEP 1:  Identification of suspect bodies of evidence
Suspect body of evidence:  0
Suspect body of evidence:  1
Suspect body of evidence:  2

STEP 2:  Generation of candidate hypotheses (tests)
Check intensity
This test challenges an environmental pre_cond

Ultrasonic diagnostic
This is a suspect sensor diagnostic

B and W camera diagnostic
This is a suspect sensor diagnostic

Color camera diagnostic
This is a suspect sensor diagnostic

STEP 3:  Execution of tests

Test 1:  testing for environmental change
This is an intensity challenge function
thresholds bandw:  110  color:  100
Average pixel value is for color image:  113.089714
Average pixel value is for b&w image:  72.088486
*****VIOLATION:  B&W MINIMUM INTENSITY THRESHOLD*****
returnval for intensity is: 0x2

******CONFIRMED TEST LIST********
Intensity change confirmed

Testing complete/successful

ERROR CLASSIFICATION COMPLETE!!!
```

b.

Figure 5: a) Inputs for SFX-EH demonstration. b.) Output of SFX-EH for error classification.

```
*******************************************************
Welcome (bienvenidos) to error recovery!!!

Description/sensor nodes in original sensing plan
Desc/sensor node: 0 ultrasonics
Desc/sensor node: 1 black and white camera
Desc/sensor node: 2 color camera

Intensity change recovery

Description/sensor nodes in repaired sensing plan
Desc/sensor node: 0 ultrasonics
Desc/sensor node: 1 color camera
ERROR RECOVERY COMPLETE
```

a.

```
*******************************************************

BEGIN SFX EXECUTION

        Type: description
        Description: Cornerness
        Sensor: Ultrasonics

        Belief:
                Support: 1.00
                Against: 0.00
                Don't Know: 0.00
                Conflict: 0.00

        Failure Status: NO_PS_FAILURE


        Type: description
        Description: Color
        Sensor: Sony- Videocam

        Belief:
                Support: 1.00
                Against: 0.00
                Don't Know: 0.00
                Conflict: 0.00

        Failure Status: NO_PS_FAILURE

No failures
Fusion step returned SUCCESS


        Type: percept
        Name: Door

        Total Belief:
                Support: 1.00
                Against: 0.00
                Don't Know: 0.00
                Conflict: 0.00

        Failure Status: NO_FS_FAILURE
******NO EXCEPTIONS*******
```

b.

Figure 6: a) Output of SFX-EH for error recovery. b) Abbreviated output of SFX after plan repair.

pruning of individual sensors from a set of like sensors used to contribute a body of evidence, and recommendation of new sensing plan selection.

5 Conclusions and Future Work

The paper has presented the SFX-EH exception handling mechanism for sensor fusion. The purpose of SFX-EH is to add robustness to sensor fusion by classifying and recovering from sensing failures. It is intended for use as part of the SFX architecture, however most of the techniques used (such as generate and test) are transferable to other systems.

SFX-EH has many advantages. It emphasizes a fast response to sensing failures whenever possible. It does this by repairing existing plans, avoiding the configuration overhead, and by using specialized, domain-dependent knowledge first, using domain-independent knowledge at the planning level as a last resort. It is able to use a partial causal model of sensing failure, reducing the amount of domain-dependent knowledge actually needed. Another advantage is that SFX-EH, at the worst case, maintains a graceful degradation of sensing by allowing sensing, even though it may be reduced, to continue, hopefully enabling the robot to complete its task.

SFX-EH has two disadvantages. First, as a fundamental design decision, it relies on domain-dependent knowledge. This allows exception handling to be faster but makes it application specific. Second, the SFX-EH recovery module does not contain multiple recovery schemes for each error cause. Multiple recovery schemes would allow the system to decide which is the best one to select based on the state of the world. Also, the recovery schemes do not attempt to ensure that the plan repair will result in the same level of certainty as the failed configuration.

Work is in progress on integrating all components of the SFX and SFX-EH system on the MR/MP Laboratory's mobile robot. Future work will specifically address the shortcomings of the error recovery module. Additional research is called for in three areas: planning, learning, and parallelism. Planning is needed to address the errors that cannot be resolved by the exception handling mechanism. Given the domain specific nature of exception handling for sensor fusion, the ability to learn new error recovery strategies and save repaired plans for future use by the configuration mechanism is needed to reduce the domain dependent knowledge currently required. One potential advantage of the SFX architecture is that it allows a robot to support independent perceptual processes for multiple tasks. Since these processes are independent, parallel implementation is expected to improve the efficiency and speed of perception, thereby improving the ability of the robot to perceive and respond to the events in the world.

Acknowledgements

The first author is supported by a State of Colorado Graduate Fellowship. The authors would like to thank Elizabeth Nitz for her helpful comments.

References

[1] Doyle, R. J., Atkinson, D. J., Doshi, R. S., "Generating Perception Requests and Expectations to Verify the Execution of Plans." *Proceedings of the National Conference on Artificial Intelligence*, 1986, pp. 81-88.

[2] Firby, R. J., "An Investigation Into Reactive Planning in Complex Domains," *Proceedings of 6th National Conference on Artificial Intelligence*, July 13-17, 1987, Seattle, WA, pp. 202-206.

[3] Hammond, K. J., "CHEF: A Model of Case-Based Planning." *Proceedings of the National Conference on Artificial Intelligence*, 1986, pp. 267-271.

[4] Hanks, S., Firby, R. J., "Issues and Architectures for Planning and Execution." *Proceedings of a Workshop on Innovative Approaches to Planning, Scheduling and Control*, San Diego, CA, November 5-8, 1990.

[5] Lindsay, R. K., Buchanan, E. A., Feigenbaum, E. A., Lederberg, J., Applications of Artificial Intelligence for Organic Chemistry: The Dendral Project, New York: McGraw-Hill, 1980.

[6] Murphy, R.R., "Robust Sensor Fusion for Teleoperations", *1993 IEEE International Conference on Robotics and Automation*, invited special session on Multisensor Fusion, Atlanta, GA, May 2-6, 1993, pp 572-577.

[7] Murphy, R.R, and Arkin, R.C., "SFX: An Architecture for Action-Oriented Sensor Fusion", *1992 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Raleigh, NC, July 7-10, 1992.

[8] Vander Velde, W. E., and Carignan, C. R., "Number and Placement of Control System Components Considering Possible Failures." *Journal of Guidance and Control*, vol. 7, no. 6, Nov-Dec, 1984.

[9] Weller, G. A., Groen, F. C. A., Hertzberger, L. O., "A Sensor Processing model Incorporating Error Detection and Recovery." In T. C. Henderson, editor, *Traditional and Non-traditional Robotic Sensors*, Maratea, italy, Aug. 28- Sep. 2 1989, Springer-Verlag.