Case Based Disruption Monitoring

Joe Kann, Matthew Molineaux and Bryan Auslander

Knexus Research Corp.
174 Waterfront Street, Suite 310, National Harbor, MD 20745

firstname.lastname@knexusresearch.com

Abstract. Mine Countermeasures Missions (MCM) take place in very complex and uncertain environments which poses complexity for planning and explanation algorithms. In order to keep a mission on target, constant disruption monitoring and frequent schedule adjustments are needed. To address this capability gap, we have developed the Case-Based Disruption Monitoring and Analyzing (CDMA) algorithm. The CDMA algorithm automatically detects disruptions within a mission and attempts to determine possible root causes. Once confirmed, our second developed algorithm, CLOSR modifies existing schedules to compensate for these root causes. Evaluation of CDMA on simulated MCM operations demonstrates the effectiveness of case-based disruption monitoring. Both the CDMA and CLOSR algorithms, along with simulator, are enclosed with our KRePE system.

1 Introduction

Unforeseen disruptions occur when planning in the real world. When monitoring for such disruptions and providing an explanation as to why the disruption occurs, better insight is provided in order to fix the plan. Mine Countermeasure Missions (MCM) for example, uses planning constantly. MCM planning uses a variety of resources and each resource has its own set of capabilities and operational constraints, as well as characteristic failure points.

Mine Countermeasure Missions (MCM) must respond to frequent disruptions, and recovering from these disruptions can be complex. MCM missions involve the location, identification, and neutralization of enemy explosive ordnance in a maritime context. This is key to naval power projection and sea control, two core capabilities of U.S. maritime power, as characterized by *A Cooperative Strategy for 21st Century Seapower* [4]. Due to high complexity and uncertainty when scheduling MCM missions, accurate plans must be created and frequently revised once a mission has started. Frequent disruptions in MCM operations can occur due to many types such as: changes in sea state, visibility, weather, equipment failure, etc. Situations like these interfere with resource availability and/or readiness. Therefore, schedules for MCM operations require frequent changes and updates where the disruptions are monitored in order to keep the success of the mission. Current practice calls for manually observing all incoming data for detection of issues that could cause a mission to fail. The manual process of monitoring for disruptions can be tedious and prone to error.

To meet this need, we are developing a system for MCM operation decision making and planning support called KRePE. KRePE builds upon a foundation of cognitive architecture components, algorithms and simulations. Housed within the KRePE architecture the Case-Based Disruption Monitoring and Analyzing (CDMA) algorithm performs monitoring and analysis of disruptions and Case-Based Local Schedule Repair (CLOSR) reschedules tasks that MCM planner operators perform on a frequent basis. Both the CDMA and CLOSR algorithms fall in a problem solving paradigm known as Case-based reasoning (CBR) by relying on general and specific knowledge of MCM operations, how operations might be disrupted, and how to fix these interruptions.

In this paper, we discuss the challenges of continuous situation monitoring, and root cause analysis of mission disruptions through case-based reasoning. We close with an empirical study that demonstrates this effective anomaly detection in order to generate schedule modifications that achieve mission success.

2 Mine Countermeasures Mission Scheduling & Operations

MCM operations involve the location, identification, and neutralization of sea mines [5]. These operations employ surface vehicles, aircraft, divers, and unmanned surface and underwater vehicles, and can take weeks to plan and execute. While the operations are taking place, they are disrupted early and often by events such as unforeseen weather conditions, technological failures, and incorrect enemy course of action estimations. While technology exists to automatically create an initial schedule, distribute tasks, and track task completion, the critical monitoring and rescheduling tasks have been, to date, poorly supported [6].

MCM operations involve a unique set of specialized tasks that must be scheduled to minimize the risk to ships from sea mines. What follows is a brief description of the tasks in an MCM operation and their characteristics. The schedule for an MCM operation tasks multiple vehicles to repeatedly *hunt* and/or *sweep* subsections of a specified *threat area* where mines are expected, slowly transiting back and forth in a lawnmower-like search pattern, until the risk of remaining mines is reduced to an acceptably low level. The paths followed by these search vehicles are referred to as *tracks*.

Hunting is a search and destroy activity that encompasses use of specialized sensors to find underwater objects that are *mine-like*, identification of mine-like objects as *mines* or *non-mines*, and neutralization of all discovered mines. The *probability of detection* describes the equipment's sensitivity within that range to the size and reflectivity of mine casings. Because mines may be missed, missions are commonly evaluated according to a *percent clearance* objective. Percent clearance is defined as the probability that a mine at any given position in the search area will be detected.

Sweeping is an activity that uses specialized apparatus to destroy all mines present in a given area either by cutting the chains that connect them to the ocean floor or employing signal generators which mimic the magnetic and acoustic signatures, of ships, to trigger mines that are activated by those signatures.

The operation schedule, which may consist of hundreds of tasks of heterogeneous types, must be repeatedly adjusted over the course of the operation in response to unexpected events which invalidate it. The task of keeping the schedule up to date despite hundreds of interrelated tasks is complex, difficult, and laborious, particularly given the constant time pressure of typical operations. Modifications to schedules are kept to a minimum, in order to reduce expense and opportunities for error; we refer to this characteristic as *minimal operational disruption*. However, modified schedules must also fulfill operational requirements such as percent clearance, time limits, and risk to equipment. These difficult tasks (i.e., monitoring, response, and rescheduling) can be greatly aided by new computational tools.

3 CDMA

One way to reduce the burden on MCM human operators is to help with constant monitoring of disruptions that will impact the mission. Constant monitoring of a vast array of disruption types can be quite difficult. In addition to detecting the disruption, diagnosing the root cause of the problem can be daunting, or easily overlooked. Case-Based Disruption Monitoring and Analyzing (CDMA) within the KRePE architecture handles both disruption monitoring and providing possible root causes.

Case-based reasoning (CBR) is a problem solving paradigm that relies on general cases of a problem domain along with specific domain cases. These cases consist of a mapping between problems and a solution. When a new problem is introduced, generally CBR systems map and provides this new problem to the most similar problem already stored in its case base and provides a solution associated with the known problem. We describe the case representation and the CDMA algorithm in detail in the following subsections.

3.1 CDMA Representation

CDMA uses case-based reasoning for monitoring and analysis of disruptions that will impact an ongoing operation. Based on limited information of the world state, the CDMA algorithm determines if a disruption has occurred. A disruption case in our system are generated manually and consists of five parts: violated expectations, parameters, root cause likelihood, root cause questions and new assumptions.

The case applies when all of the violated expectations are met; and the parameters indicate which variables are applied to a specific problem instance. An example problem representation is shown in Table 1. In this example, there is a disruption where the operator has not heard from the unit within the past 15 minutes while it was out in the field performing a task.

The likelihood and list of root cause questions provide information that can be accessed by an operator through an interactive decision making process. The likelihood provides an apriori probability of how likely a particular root cause is for a given disruption. The root cause tests constitute a set of questions that can help the operator deduce what is causing the disruption. The parameters defined by the violated expectations populate the variables within the questions, detailing the questions to a specific unit, piece of equipment, etc. If these questions are answered, the likelihoods for the root causes adjust to this information. Using the example from above, Table 1 provides the entire case representation. The new assumptions are a set of suppositions or beliefs as to which root cause explains the disruption. The parameters defined from the violated expectations instantiate the problem information into these new assumptions.

	Case Example
Violated Expectations	current_time(?curTime) ^ unit_last_check_in(?unit,?lastCheck) ^ subtract(?curTime,?lastCheck,?difference) ^ greater_equal(?difference,15.0) ^ unit_assigned_task(?unit,?task) ^ ¬equal(?task,'Unassigned')
Parameters	?curTime, ?unit, ?lastCheck, ?difference, ?task
Likelihood	0.9
Questions	Is ?unit communicating on short-wave?
New Assumptions	unit_capability_failure(?unit, 'Communications', ?lastCheck)

Table 1. Case Representation for CDMA algorithm.

With the use of a standard relational database called the Integrated Rule Inference System (IRIS) [8], CDMA can reuse case(s) in the problem space without having to generate new cases for each set of parameter values. Therefore similarity metrics are not being used. From the example, we do not need to create new cases for each type of equipment or unit, as it can handle all of the parameters. When monitoring detects a disruption, it alerts human operators with a message. The operator then decides the root cause of a given disruption. CDMA adds this confirmed root cause assumptions to the case base providing more information to its case base. These new assumptions trigger schedule repair to occur because the disruption affects the mission.

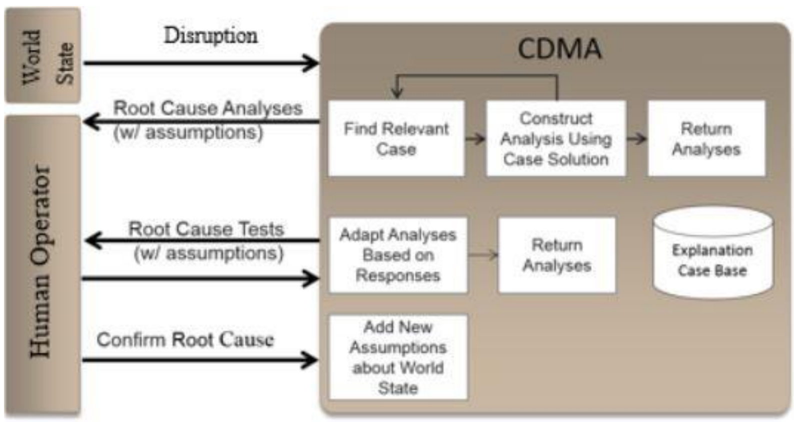

Fig. 1. Workflow for CDMA algorithm.

3.2 CDMA Algorithm

CDMA performs the following steps for disruption monitoring and analysis as shown in Figure 1:

1. ***Find Relevant Case***: To find a possible disruption, CDMA searches through the list of cases to find a relevant case that matches a violated expectation. Each case that matches provides a possible root cause for the disruption.
2. ***Construct Analysis Using Case Solution***: To analyze a disruption, the parameter values indicated by a specific violated expectation are substituted for the parameters specified by an individual case problem.
3. ***Return Analyses***: Each possible disruption is provided on screen for the user to review, detailing the types of root causes for the disruption, along with additional information such as root cause tests and likelihood for each cause.
4. ***Adapt Analyses Based on Responses***: Users can answer these root cause test questions in order for the system to better understand the disruption for future root causes.
5. ***Return Analyses***: The system returns updated likelihoods, sorted with highest likelihood first, along with clearing out infeasible causes.
6. ***Add New Assumptions about World State***: After user selection of the root cause for a disruption, the system creates new assumptions about the world and why the disruption occurred. These new assumptions are added into the case base, providing new information that can be used to generate schedule repair if necessary.

4 CLOSR

To repair schedules that don't meet the criterion of minimal operation disruption, we use the Case-Based Local Schedule Repair (CLOSR) algorithm [10]. This Case base reasoning algorithm in the KRePE architecture creates new assumptions and generates repairs. These repairs strive for "minimal disruption" meaning changes to the schedule should be kept at a minimum while rescheduling to fix a disruption. For example, in MCM operations, repairing a vehicle communication disruption might try to resolve the problem without leaving its search area to minimize transiting time and fuel. Subsequent to case reuse, an adaptation process examines and resolves conflicts created by the schedule repair procedure which is useful for its flexibility. For more detail, please see [10].

5 Evaluation

We hypothesize that the discrepancy monitoring and analysis capabilities of CDMA outperforms ablations that ignore alerts or acts on randomly-selected root causes. To demonstrate this, we ran the CDMA algorithm in an automated manner on a series of simulated MCM operations. For each operation, we measured and compared the performances of three decision makers that: (1) ignores all alerts from CDMA and keeps the original schedule, (2) acknowledges CDMA found disruptions and chooses a random root cause from those suggested therefore rescheduling randomly and (3) acknowledges CDMA found disruptions and chooses the root cause with the highest likelihood. Difference between decision makers indicate the performance improvement that can be achieved by adopting the recommendations made by the CDMA algorithm.

Our study examines an MCM mission with a mine clearing objective. As it is impossible to ensure that 100% of mines are removed in the real world, missions are planned to achieve a high level of percent clearance. This means that there is a high chance that a mine at any given point in the search area would be observed if it existed. The operations conducted in our study are intended to achieve a 95% clearance level; in other words, we would expect 95% of the mines present to be removed. We hypothesize that the decision maker using KRePE's case base will achieve these performance objectives, and that the decision maker that ignores the disruptions will not. This will demonstrate both that monitoring and analyzing disruptions is necessary to achieve an acceptable level of performance under simulated conditions, and that the system is sufficient to achieve that performance.

5.1 Experimental Framework

A simulator for MCM operations, Search and Coverage Simulator (SCSim), another component of KRePE, supports rapid and repeated evaluation and testing of MCM decision support systems and component algorithms. SCSim simulates search missions involving multiple heterogeneous search units, including ships and helicopters, each with different available equipment configurations. Mines and mine-like objects are distributed randomly by SCSim in fields and lines according to pre-set distributions with variable density and object counts. This facilitates evaluation of algorithm performance under varying operating conditions. As a benchmark, automated testing of a two month operation takes less than one minute.

SCSim simulates the assignment of parameterized tasks to units according to a schedule, including transit, sweep, and hunt tasks. Task parameters include, for example, the equipment to use for sweeping, and sensor depth for hunting. To simulate a mission, SCSim automatically generates appropriate tracks for each task and simultaneously changes the position of each vehicle along its assigned tracks. Observations (e.g., contacts) are generated based on vehicles' positions and the sensor equipment in use. Interactions of deployed sweeping equipment is also simulated, and changes the internally represented status of mines. In addition to the scheduled tasks, SCSim is responsible for simulating random events the unexpected difficulties that invalidate an existing schedule (e.g., equipment failure, bad weather, operator errors).

An individual mission test using SCSim is controlled by a scenario description. Scenario descriptions include, at a minimum, the vehicles and equipment available for use, threat areas to be cleared of sea mines, and task areas where vehicles will operate. Other elements of the scenario specify random distributions for mine like objects, mine line placements, and events that may occur. To mimic the real world as closely as possible, SCSim provides only partial observations for the purposes of rescheduling. For example, when a helicopter's communications system fails, its position is no longer reported to the system. As a result, the helicopter appears not to move.

Experiments are driven by a test harness that integrates with SCSim as shown in Figure 2. The test harness generates scenarios defining: the area of operations, available assets, and the ranges of random experimental variables, such as what mine types will be deployed and when events will trigger. The Test Generator applies an appropriate decision maker that acts as a user of the system. Each decision maker encodes different responses to situations, such as alerts, that arise during the mission simulation. After all simulated missions are complete, the Performance Evaluator tabulates and summarizes these results in a human readable form.

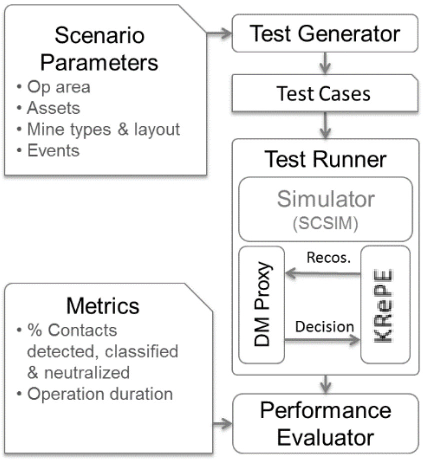

Fig. 2. KRePE simulation driven evaluation

5.2 Experiment Setup

Our experiments used three decision makers and ten randomly generated test scenarios. The first decision maker, “KRePE DM”, confirms the correct root cause with the highest disruption likelihoods and selects a new schedule from those generated to activate. The second decision maker, “Random DM”, randomly chooses a root cause and selects a new schedule from that root cause. The third decision maker, our baseline, “Ignore DM”, ignores KRePE’s recommendations, never changing its schedule when prompted. Comparing performance of these three decision makers allows us to measure the efficacy and correctness of schedules generated by case-base disruption monitoring system.

The performance of each decision maker was evaluated in each of ten randomly generated scenarios, generated. (See Table 2). Scenarios differ primarily in the thirty random events that occur and the positions of mines and mine-like objects. Each event was additionally parameterized with a trigger time (chosen randomly over the first six-hundred hours of the mission) and target unit (chosen randomly among the six tasked assets). The times were chosen in this fashion because events that occur when a unit has already performed all its tasks cause no problems, and therefore are uninteresting to our study. Four mine lines, each with a mine count between ten and thirty, at various depths and mine types were placed randomly in each scenario.

The fixed parameters used in all scenarios included the area searched, and seven assets, consisting of four helicopters, two MCM ships, and one support ship that could assist in tasks if necessary. Each ship and helicopter has available equipment for hunting mines, contact sweeping, detection, and mine neutralization.

5.3 KRePE Metrics

We evaluated KRePE DM, Random DM, and Ignore DM using the following three metrics: (1) *percent contacts detected:* This measures the percentage of mines detected by a unit; (2) *percent mines neutralized:* Percentage all mines are neutralized by a unit and (3) *operation duration:* Total simulation time required to complete the operation.

The first two metrics are calculated based on the true number of mines and mine-like objects generated in the scenario. These summarize the plan's effectiveness in terms of how well the MCM mission goal of searching for and eliminating mines was achieved. Each scenario generated includes a large number of non-mine mine-like objects uniformly spread throughout the threat area, so the percent contacts detected value is an approximation of the percent clearance, or probability that a mine would be detected at any given location. The third metric, operation duration, illustrates a plan's efficiency by measuring the total simulation time required to complete all tasks.

5.4 KRePE Results

Experiments were run on an i7 processor laptop, taking one hour to complete. Figure 3 shows a scatter plot that displays the percentage of existing contacts that were classified correctly and duration of each mission operation measured in simulation hours. The duration of an operation performed by Ignore DM varies little, as the original schedule is never updated, whereas the duration of KRePE DM and Random DM missions can vary greatly. A schedule can be lengthened dramatically when new mine types have been discovered; to ensure safety, many new hunt and/or sweep tasks must be introduced to clear the additional mines. Similarly, if vehicles are damaged beyond repair, the diminished resources can greatly increase mission length. The increased time and repaired schedules allow KRePE DM to outperform Ignore DM by classifying between 95 and 100% of the mine like objects in every mission. Random DM, like KRePE DM, responds to disruptions, but because it does not choose the most likely cause, its task performance is not as high as KRePE DM's. Note that neither Ignore DM nor Random DM represents any real human decision maker; rather these results should be interpreted to show the difficulty of the task and that CDMA's suggestions are benefitting mission performance.

Table 2 shows one-tailed t-test with paired examples. The results include the average and standard deviation for each metric and decision maker. Note: indicate the (small) likelihood that Ignore DM might on average achieve higher values than KRePE DM if many more experiments were undertaken.

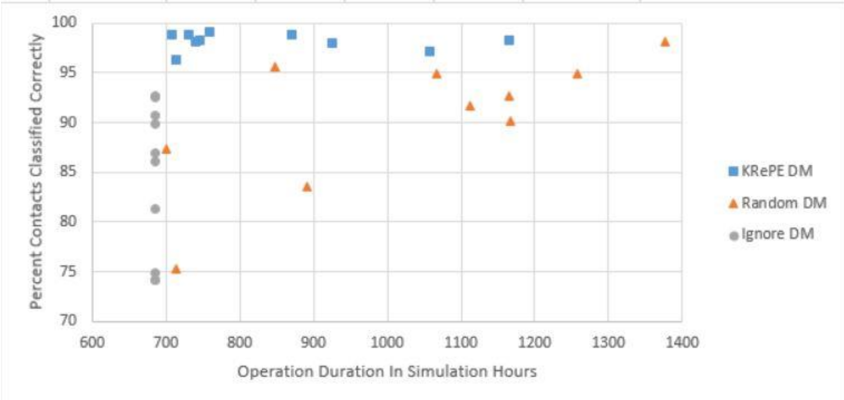

Fig. 3. Scatter Plot of Operation Duration to Percent Contacts Classified Correctly

Table 2. KRePE Results

Metric	KRePE DM	Random DM	Ignore DM
Percent Contacts Detected	98.2 ± **0.8**	90.4 ± **6.5**	84.3 ± **7.2**
Percent Mines Neutralized	93.7 ± **8.1**	88.3 ± **9.6**	81.5 ± **14.4**
Operation Duration	841.6 ± **152.2**	1030.0 ± **218.8**	685.5 ± **0.3**

6 Related Work

Case-based reasoning [1] is a problem solving process based on the adaptation and application of known solutions to new problems. It has been applied to many different domains and problems besides disruption detection.

DISCOVERHISTORY [9] looks for explanations of observations through abductive reasoning, where it maps an observation to a hypothesis that accounts for the observation. DISCOVERHISTORY has been shown to be effective over a large problem space, but is slow with determining disruptions. This is not sufficient for quick detection of immediate issues required by mine countermeasures operations.

A case-based reasoning system, CHEF [7] creates food recipes and explains its own failures. The system tries strategies to see which one can be used to fix the recipe plan. CHEF uses causal rules to explain why its own plan fails. However, the system does not handle constrained resources present in a typical scheduling problem.

The system described in [3] is a CBR system that focuses on wartime equipment maintenance by analyzing feature sets of equipment for maintenance. The system automates the process of deciding the quality of the equipment. CDMA, in contrast, supports a “man-in-the-loop” in order to allow operators to have control over what should be done about disruptions.

7 Conclusion

We presented the CDMA algorithm within the KRePE system that supports monitoring for disruptions and disruption analysis in mine countermeasures operations. Scheduling in this domain is challenging due to the complexities resulting from a large number of tasks that must be allocated over numerous resources. CDMA includes components that assist operation planners by constantly monitoring the environment for changes and providing analysis of discrepancies. Once disruption detection occurred CDMA made it possible for the CLOSR algorithm to reschedule without the need to replan by recommending alternative schedules. We introduced the requirement of minimally disruptive repair as a key operational requirement for automatic schedule repair algorithms in MCM applications.

Our results indicate the efficacy of a case-based strategy; schedule repair was rapid, and created new schedules on demand that ensured the elimination of all mines and increased clearance to a reasonable level. This presents a novel and measurable increase in automated MCM rescheduling capabilities. In the future, we want to apply our system to Unmanned Combat Logistic missions in order to demonstrate effective case-base disruption monitoring with other domains.

8 Bibliographic References

1. Aamodt, A., & Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, *7*(1), 39-59.
2. Boyd, John, R. (1995). The essence of winning and losing. 28 June 1995.
3. Cai, Jiwei., Jia, Yunxian., Gu, Chuang., and Wu, W. (2011). Research of Wartime Equipment Maintenance Intelligent Decision-making Based on Case-Based Reasoning. In *Procedia Engineering* (Volume 15, 2011, pp. 163-167. CEIS 2011).
4. Chief of Naval Operations, Commandant of the Marine Corps, & Commandant of the Coast Guard. (2007). *A Cooperative Strategy for 21st Century Seapower*.
5. Cummings, Mary, and Collins, Angelo. (2010). Autonomous Aerial Cargo/Utility. In *Concept of Operations, Department of the Navy, ONR, Science & Technology*.
6. Garcia, G. A., & Wettergren, T. A. (2012). Future planning and evaluation for automated adaptive minehunting: A roadmap for mine countermeasures theory modernization. In *SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics.
7. Hammond, Kristian J. (1986). CHEF: A Model of Case-Based Planning. In *Proceedings of the Fifth National Conference on Artificial Intelligence*. Philadelphia, Pennsylvania.
8. *IRIS. Program documentation. IRIS Reasoner. Vers. 0.6. N.p., 3 Apr. 2008*. Web. 1 Aug. 2015. <http://www.iris-reasoner.org/pages/user_guide.pdf>.
9. Molineaux, M., Kuter, U. and Klenk, M. (2012). DiscoverHistory: Understanding the past in planning and execution. In *Proceedings of the Eleventh International Conference on Autonomous Agents and Multiagent Systems* (pp. 989–996. ACM Press, Valencia).
10. Molineaux, M., Auslander, B., Moore, P. G., & Gupta, K. M. (2015). Minimally disruptive schedule repair for MCM missions. In *SPIE Defense+ Security*. International Society for Optics and Photonics.