SANDIA REPORT
SAND2020-4198
Printed April 2020

The Grey Zone Test Range Integrated Urban Simulation Environment

Andjelka Kelic, Walt Beyeler, Roger Mitchell, Michael Bernard, Casey Doyle, Alisa Rogers, Chris Frazier, Thushara Gunda, Katherine Klise

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico
87185 and Livermore,
California 94550

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology & Engineering Solutions of Sandia, LLC.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@osti.gov
Online ordering:	http://www.osti.gov/scitech

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Rd
Alexandria, VA 22312

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.gov
Online order:	https://classic.ntis.gov/help/order-methods/

ABSTRACT

Sandia National Laboratories is part of the government test and evaluation team for the Defense Advanced Research Projects Agency Collection and Monitoring via Planning for Active Situational Scenarios program. The program is designed to better understand competition in the area between peace and conventional conflict when adversary actions are subtle and difficult to detect. For the purposes of test and evaluation, Sandia conducted a range of activities for the program: creation of the Grey Zone Test Range; design of the data stream for a user experiment conducted with U.S. Indo-Pacific Command; design, implementation, and execution of the formal evaluation; and analysis and summary of the evaluation results. This report details Sandia's activities and provides additional information on the Grey Zone Test Range urban simulation environment developed to evaluate the performer technologies.

ACKNOWLEDGEMENTS

Many additional people contributed countless hours to this work and it would not have been possible without their effort. The additional Sandia team includes: Mika Armenta, Cheryl Bolstad, Bill Fogleman, Becky Jeffers, Ernest Kim, Rob Kittinger, Will Peplinski, Paul Schutte, La Tonya Walker, Jeremy Wendt, and our program manager, Rossitza Homan. The authors would like to thank the entire DARPA team for their support during this effort: Fotis Barlos, Ashley Calder, Sri Kumar, Anna Skinner, Scott Peeke, and Wade Pulliam. We also wish to thank Ken McCullough, Gregg Current, David Cassidy, David Hansen, and Steve Cohen from NIWC Atlantic without which the user event would not have been possible, Bob Mohan from USINDOPACOM who has been a key advocate for COMPASS, and last but not least the performer teams and our integration partners from Lockheed Martin: Jim Starz, Mark Hoffman and Kevin Spivey, whose dedication and hard work made this all possible.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

This page left blank

ACRONYMS AND DEFINITIONS

Abbreviation	Definition
API	Application Programming Interface
COMPASS	Collection and Monitoring via Planning for Active Situational Scenarios
DARPA	Defense Advanced Research Projects Agency
DYMATICA	DYnamic Multi-scale Assessment Tool for Integrated Cognitive-behavioral Actions
GZTR	Grey Zone Test Range
INDOPACOM	United State Indo-Pacific Command
LOE	Line of Effort
SES	Socio-Economic Status
TA1	Technical Area 1
TA3	Technical Area 3
TAZ	Transportation Analysis Zone

This page left blank

1. INTRODUCTION

Sandia National Laboratories (Sandia) is part of the government test and evaluation team for the Defense Advanced Research Projects Agency Collection and Monitoring via Planning for Active Situational Scenarios (DARPA COMPASS) program. The program is designed to better understand competition in the area between peace and conventional conflict when adversary actions are subtle and difficult to detect. The goal of the program is to provide theater-level operations and planning staff with analytic and decision support tools that reduce ambiguity about adversarial actors and objectives. These tools recommend "probing" actions tailored to elicit responses from adversaries to reveal additional information about their goals and strategies.

Sandia conducted several different activities for the program: creation of the Grey Zone Test Range (GZTR); design of the data stream for a user experiment conducted with U.S. Indo-Pacific Command (INDOPACOM) based on INDOPACOM exercise data; design, implementation, and execution of the formal evaluation; and analysis and summary of the evaluation results. This report details Sandia's activities and provides additional information on the GZTR urban simulation environment, called COMPASSville, developed to evaluate the performer technologies.

The program teams consisted of four performer teams (called Technical Area 1, or TA1, teams) that were developing the software to conduct adversary intent and strategy detection and determination and an integration team (called Technical Area 3, or TA3) that was responsible for both the interface between the GZTR and the performer software and the user interface to the performer tools.

This page left blank

2. GREY ZONE TEST RANGE

The COMPASS GZTR is a controlled mid-sized urban simulation environment with adversary activity, which is created to evaluate performer technologies' ability to detect an adversary and their intent. To determine the boundaries of detection limits, the GZTR is engineered with controlled difficulty of the detection problem. The following sections provide an overview of the urban environment, called COMPASSville, and contain details about the architecture and models comprising the GZTR that create the simulation environment. The section concludes with a description of the scenario scripts that provide the detection problem for the TA1 performers and the model that feeds those scripts into the GZTR.

2.1. Overview

The GZTR consists of four core interdependent infrastructure models: electric power, food, fuel, and transportation. Interacting infrastructure models provide a core of services to the urban population of about 500,000 people. Coupled with the infrastructure models are socio-behavioral models of poor and non-poor segments of the population. The population's reactions to stimuli in the environment, such as from infrastructure disruptions or changes in government policy, are captured by socio-behavioral models. Population sentiment varies by location within the city and by socioeconomic status. Layered on top of the urban environment are models of the local government and a set of adversaries executing strategies to accomplish goals.

The urban environment of the GZTR is called COMPASSville and is geospatially located in Hargeisa, Somaliland. The general location is selected to present performers with a plausible synthetic analog of challenges that might be encountered in a real-world theater of operations. The core urban simulation is loosely based on the city of Hargeisa Somaliland. However, detailed information about the city – its geography, infrastructure extent and condition, demographics, and economic conditions – is modified to represent more-developed cities. This approach ensures that specific knowledge of the region and associated known conditions cannot be used by the performers when interacting with the GZTR to develop COMPASS technologies. All necessary information to conduct actor goal and strategy assessment is contained within the GZTR. Although the city of COMPASSville is the central stage for interacting with the physical infrastructures and the urban population, the overarching political dynamics are inspired by broader dynamics of major state and non-state actors in the Horn of Africa.

The GZTR includes: (1) detailed geographic data about COMPASSville, including road networks, a power grid, food and fuel supplies and locations where the population conducts day to day activities such as going to work and school, and (2) population demographics including age, socioeconomic status and employment rates. COMPASSville provides a synthetic environment that contains information that would be available to performer technologies in the real world.

The basic unit within COMPASSville is called a TAZ (transportation analysis zone). The 420 TAZs of COMPASSville are shown in Figure 2-1. A TAZ is a logical geospatial region made up of on average 185 households divided by poor and non-poor to get the desired socio-economic distribution and total population numbers (480,000). Population in the TAZ is measured by the residential, or night time, population numbers. The collection of TAZs make up COMPASSville, with 400 primarily residential TAZs and 20 primarily business districts that are the source of many of the jobs and do not have a residential population. Similar to the socio-economic distribution, jobs are distributed and allocated to the population to simulate the desired employment rate.

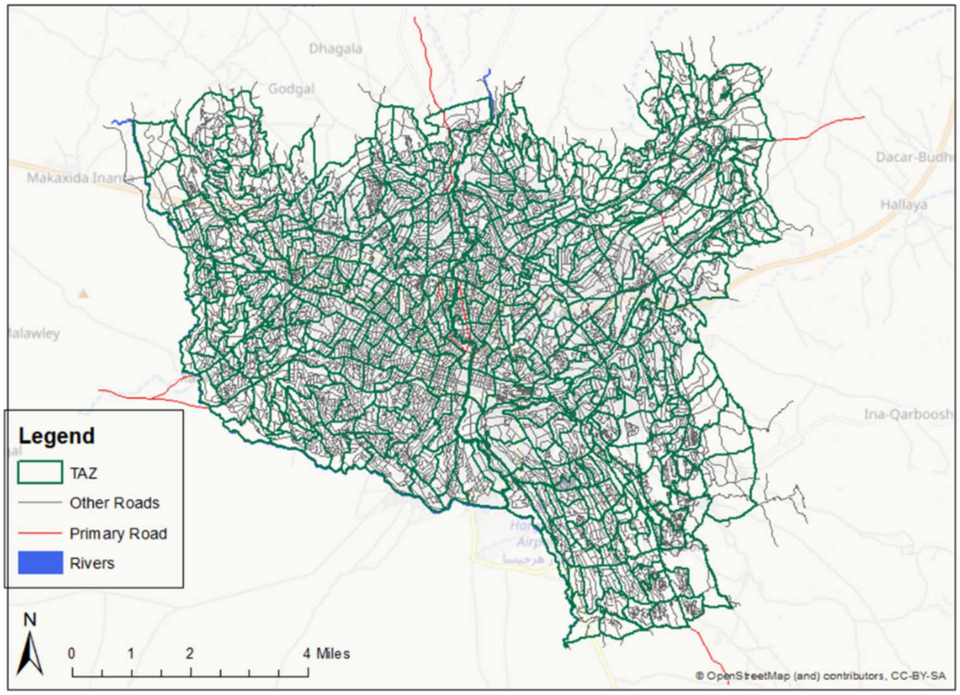


Figure 2-1. COMPASSville TAZs

The Road to Crisis sets the stage for the detection problem presented to the performers. It provides additional background information including: details about Somaliland's independence; political and economic activities in Somaliland and Somalia; Somaliland's recent interactions with Somalia; details on the violent extremist organization (al-Shabaab) operating in the region and its recent activities; and Russian and Chinese interests in the region. This background information is designed to provide additional information on motivations behind possible Red actors' actions in the GZTR.

Two year-long scripts contain Red actors' events and actions, for use in performer training and evaluation. Events are encoded in a Sandia-developed formalism called CAMEO+: an extension of the CAMEO formalism used to encode geopolitical events. Since the COMPASS program is focused on ambiguous Gray Zone activities, including infrastructure events and associated Red actions, the CAMEO formalism is extended to include such activities. Script actions are implemented using the same construct available to the performers for interacting with the simulation.

2.1. COMPASSville Data Sets and Models

COMPASSville contains detailed models of electric power, food, fuel, transportation, and the local population. The four infrastructure models are a core, interactive set of lifeline infrastructures that allow for a variety of dynamic Red actor actions. The population model governs the behavioral dynamics of the COMPASSville citizens as well as their interactions with and reactions to unfolding events.

The geographic information for the GZTR is constructed from cities of a similar population size using asset data from OpenStreetMap. The baseline for the number of infrastructure assets necessary to support the population is based on proxy cities with similar populations. Note that OpenStreetMap does not provide sufficient information for creating a model: asset information is incomplete and; specifics associated with assets modeling the provision of goods and services by that infrastructure such as capacity and usage, do not exist in OpenStreetMap. To overcome this challenge, model assets are augmented and additionally specified, as needed, to create the food, fuel, electric power, and transportation models and match that infrastructure to the population characteristics, such as consumption rates and fuel usage based on socioeconomic status.

Population demographics and the associated behavioral characteristics are constructed using modified World Bank and Somaliland government publications. Specifics related to the behaviors of the population are created based on Horn of Africa regional culture and history supported by relevant socio-behavioral theories of human behavior. Demographics and behaviors are then co-simulated with the infrastructure models to allow for appropriate interactions and sizing of the infrastructure.

The environment also includes a basic household model that accounts for how a household allocates resources. Population demographics, augmented with Somaliland government and regional information, are used to construct the model.

The simulation environment creates a series of structured data streams, with real world analogs, to facilitate transition of the COMPASS performer technologies to an operational environment.

- Population demographics are in a format similar to CIA World Factbook or other publications.
- Asset information and status are in a format similar to MIDB or OpenStreetMap. Infrastructure status is model generated.
- Public perception information, on elements such as government and infrastructure services, is similar to information that could be found in media reports, governmental assessments, and survey data.
- Household economic information is similar to survey data (for example, the American Household Survey).
- Indicators of commodity prices and samples and averages of commodity inventories as are based on real world pricing and commodity information.
- News feeds are text similar to news banners summarized by CAMEO-like information (actor, action, and object).
- Bulletin reports contain text similar to security and intelligence reports issued by the U.S. State Department to its personnel summarized by CAMEO-like information (actor action object).
- Twitter feeds include trending social media keywords and counts, with keywords similar to the news feed and bulletin reports.

2.2. Grey Zone Test Range Simulator Architecture

The COMPASS simulator consists of a Controller program, multiple Java or Python models, and the MongoDB database. Figure 2-2 is a diagram of the COMPASS simulator architecture. A development user interface that is part of the Controller and utilized file system files are not shown. The GZTR simulator may be run on either the Linux or Windows operating systems.

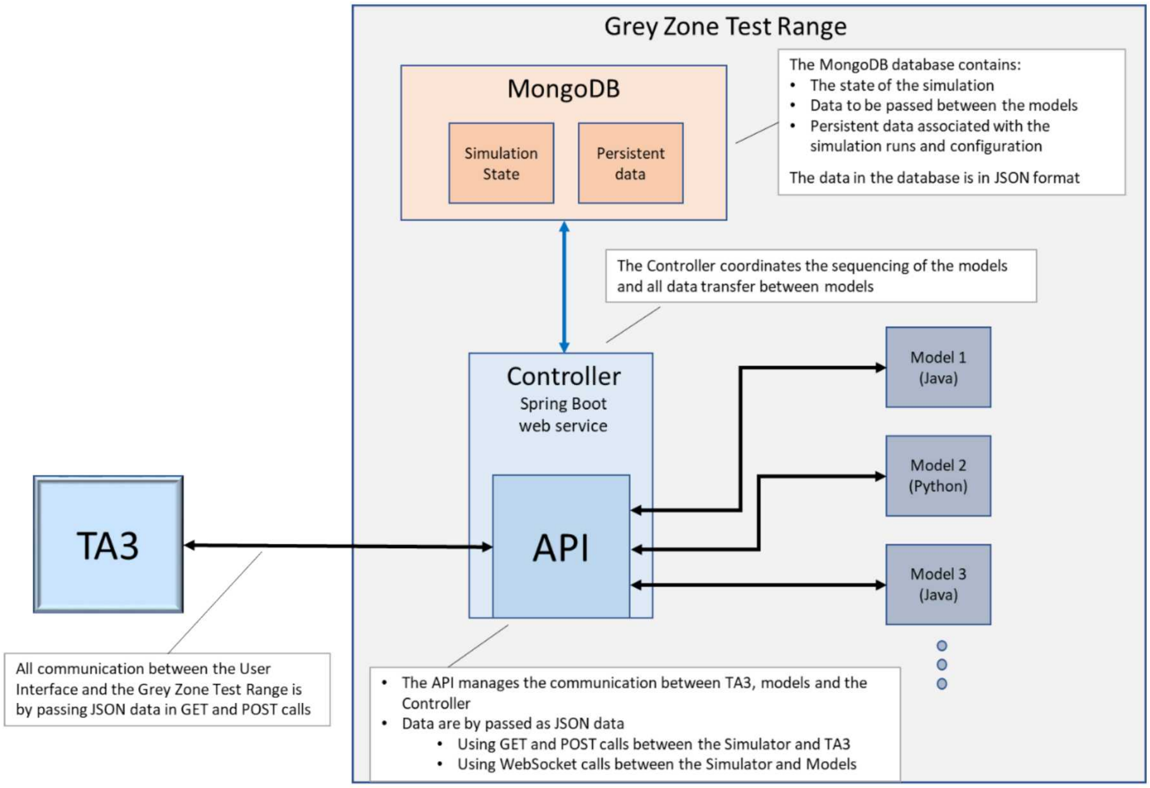

Grey Zone Test Range
MongoDB
Simulation State
Persistent data
The MongoDB database contains:
• The state of the simulation
• Data to be passed between the models
• Persistent data associated with the simulation runs and configuration
The data in the database is in JSON format
The Controller coordinates the sequencing of the models and all data transfer between models
Controller
Spring Boot web service
API
Model 1 (Java)
Model 2 (Python)
Model 3 (Java)
TA3
All communication between the User Interface and the Grey Zone Test Range is by passing JSON data in GET and POST calls
• The API manages the communication between TA3, models and the Controller
• Data are by passed as JSON data
• Using GET and POST calls between the Simulator and TA3
• Using WebSocket calls between the Simulator and Models

Figure 2-2. GZTR Simulator Architecture

The Controller is a Spring Boot Java web service program. The Controller provides the Application Program Interface (API) for communication between the TA3 interfaces and the GZTR, coordinates the running of the models, provides communication between the models, and stores data to the MongoDB database.

The Controller provides the API for all communication between TA3 and the GZTR. This API uses a combination of standard web GET and POST calls to control the running of the simulation and to retrieve data from the simulation. There are many commands and requests defined in the API, but the primary calls used are to:

- Start the models in the GZTR
- Have the GZTR run for one, or more, time steps
- Retrieve data from the GZTR
- Reset the GZTR

The communication between TA3 and the Controller is asynchronous where TA3 may have to poll the GZTR status to determine what state the GZTR is in and if data is available for a specific time step.

The Controller is responsible for coordinating the operation of the multiple models in the GZTR. The models are implemented as either Java or Python programs that are run as executable programs. The models are responsible for modeling a specific area of COMPASSville infrastructure or socio-behavior or providing key data components of the simulation such as the event or actions stream. The event stream is the CAMEO+ information that is provided as part of the scenario and the actions are Red or other activities that have a direct impact on the simulation.

The Controller communicates with the models through multiple WebSockets channels. These WebSockets channels provide synchronous communication between each model and the Controller. The models communicate through the Controller and not directly.

When the Controller receives the start command through the API it automatically starts all the models as either a Java or Python executable, depending on the model. Prior to starting the models, the Controller attempts to kill each model, if running, by first sending a reset command and then doing a hard kill command. The API provides a reset command that stops all the models, again by first sending a reset command and then doing a hard kill command.

The Controller is responsible for the sequencing of the models when the GZTR runs. This is accomplished by sending messages through the WebSockets. The models are sequenced by all models running at the same time using the data from the previous time step while the Controller waits for all of the models to complete their processing before moving on to the next time step. If a model does not respond in time, it times out and an error is reported. This method is used over methods where each model would run synchronously and in a specific order to improve the running time of the simulator.

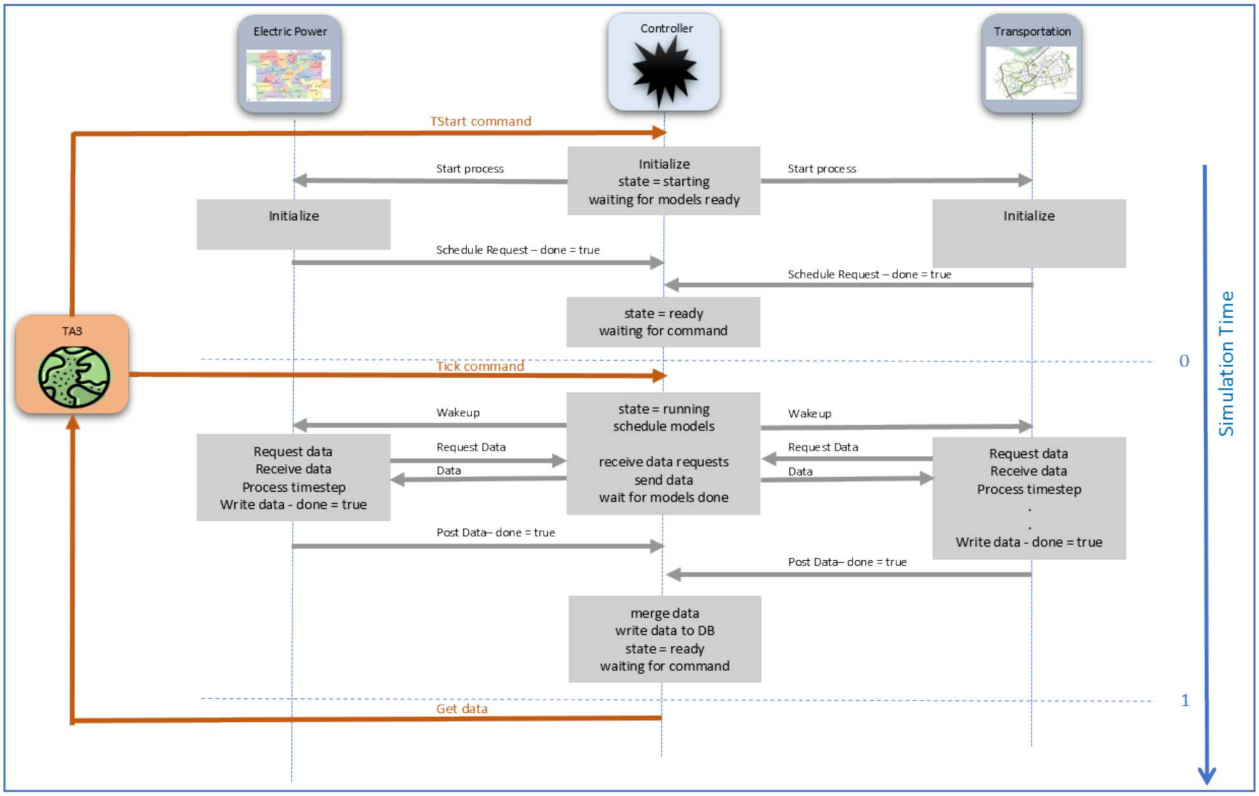


Figure 2-3. GZTR Model Sequencing

Figure 2-3 shows a diagram of the model sequencing. Here only two models shown as an example, with the Controller and the TA3 interface. Simulation time is shown as moving down the page. Functions performed by the models and the Controller are shown in the boxes, the API to TA3 interface is shown as orange arrows, and gray arrows show messages across the WebSockets between the models and the Controller. The diagram shows the model startup and a single time step sequence. The steps are:

1. TA3 sends a start command through the API.
2. The Controller starts all the models and waits for them to respond with a Schedule Request message with the done flag set to true.
3. Each model starts and performs any initialization tasks required.
4. Each model sends out a Schedule Request message with the done flag set to true.
5. The Controller sets its state to ready and waits for commands.
6. TA3 sends a tick command through the API.
7. The Controller changes its state to running.
8. The Controller sends a Wakeup message to all models that have requested to run at this time.
9. The models request any required data from the Controller.
10. The Controller services any data requests and provides data messages to the models. The data comes from the global Results data store from the previous step time.
11. The Controller waits for all models to finish and sends out the done flag set to true. A model times out if it does not respond in time and an error is reported.
12. The models perform their processing for the time step.
13. The models write out their data for the time step with the done flag set to true.
14. When all the models have reported back with the done flag set to true, the Controller merges the data into the global Results data store and writes it to the MongoDB database. The Controller sets its state to ready.
15. TA3 has been polling the Controller and now knows that the step is done and requests data for the time step.
16. The Controller responds to TA3's data request and waits for new commands.

Multiple data collections from the GZTR are stored in the MongoDB database in a hierarchical JSON format. The primary data collections are: Results and Configuration. The Results collection contains all the data from the simulation runs. This data is stored as a single JSON record for each time step that represents the entire state of the GZTR and all the models contained in the GZTR. The Configuration collection stores data that does not change with each simulation run. Examples of Configuration data are the GeoJSON files that specify the assets in COMPASSville and the demographics for COMPASSville. Both the Results and Configuration data may be accessed via the API calls.

The various models in the GZTR may not run on every time step and may not produce data for each time step. The Controller keeps one Results data store for all the models of the entire system. At the end of each time step, the Controller merges the data from the various models into the single data store. Values that already exist from a previous time step are replaced by the new values from the current time step, values that were not already in the data store are added. As a result of this merging, the Results data store has the current state of the system, but no information is given for the time step when a specific value was set. If this information is required, the user must query for data at multiple times to find when the value changed. At the end of each time step the Controller writes the Results data to the MongoDB database.

When models are running during a time step the data that they request and receive from other models is from the previous time step. Since the API requests data after the time step is complete, information is returned for the most recently completed time step. This merging approach for the results data storage allows the user to always get the entire state of the system with one data request and not have to maintain the total memory state of the system themselves.

2.3. Electric Power Model

The GZTR electric power model is a Python package which integrates disruption and restoration into power system modeling. It uses pandapower and PYPOWER to define the power model, interact with the power model to update loads, generation, disruptions and restorations and solve optimal power flow at each time step.

The electric power model includes background disruptions, scheduled restoration with a repair crew, and interaction with external dependencies such as additional infrastructure models and the ability for actors to break and restore model components. The electric power system contains external and internal generators. Distribution lines connect generators to substations. Transmission lines connect substations to customers, who are grouped into TAZs. An illustration of these connections is shown in Figure 2-4.

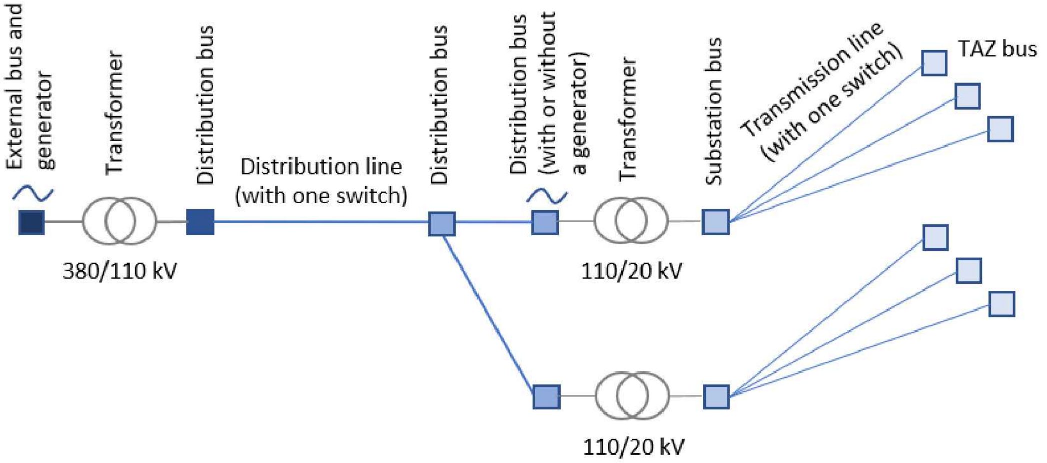

External bus and generator
Transformer
Distribution bus
Distribution line (with one switch)
Distribution bus
Distribution bus (with or without a generator)
Transformer
Substation bus
Transmission line (with one switch)
TAZ bus
380/110 kV
110/20 kV
110/20 kV

Figure 2-4. Sample Power System Configuration

The electric power system consists of the following components:

- 538 buses (14 external, 59 distribution, 45 substation, 420 TAZ)
- 420 loads (one for each TAZ)
- 24 generators (14 external generators, 10 internal generators)
- 507 lines (87 distribution lines, 420 transmission lines)
- 507 switches (one for each distribution line, one for each transmission line)
- 59 transformers (14 at external buses and 45 at substations buses)

Each TAZ is assigned a base load and load pattern using synthetic residential and commercial loads. The total peak system load is expected to be 100 MW. Each generator is rated to 5 MW for a total system generation capacity of 120 MW. A diagram of the COMPASSville electric power system can be found in Figure 2-5.

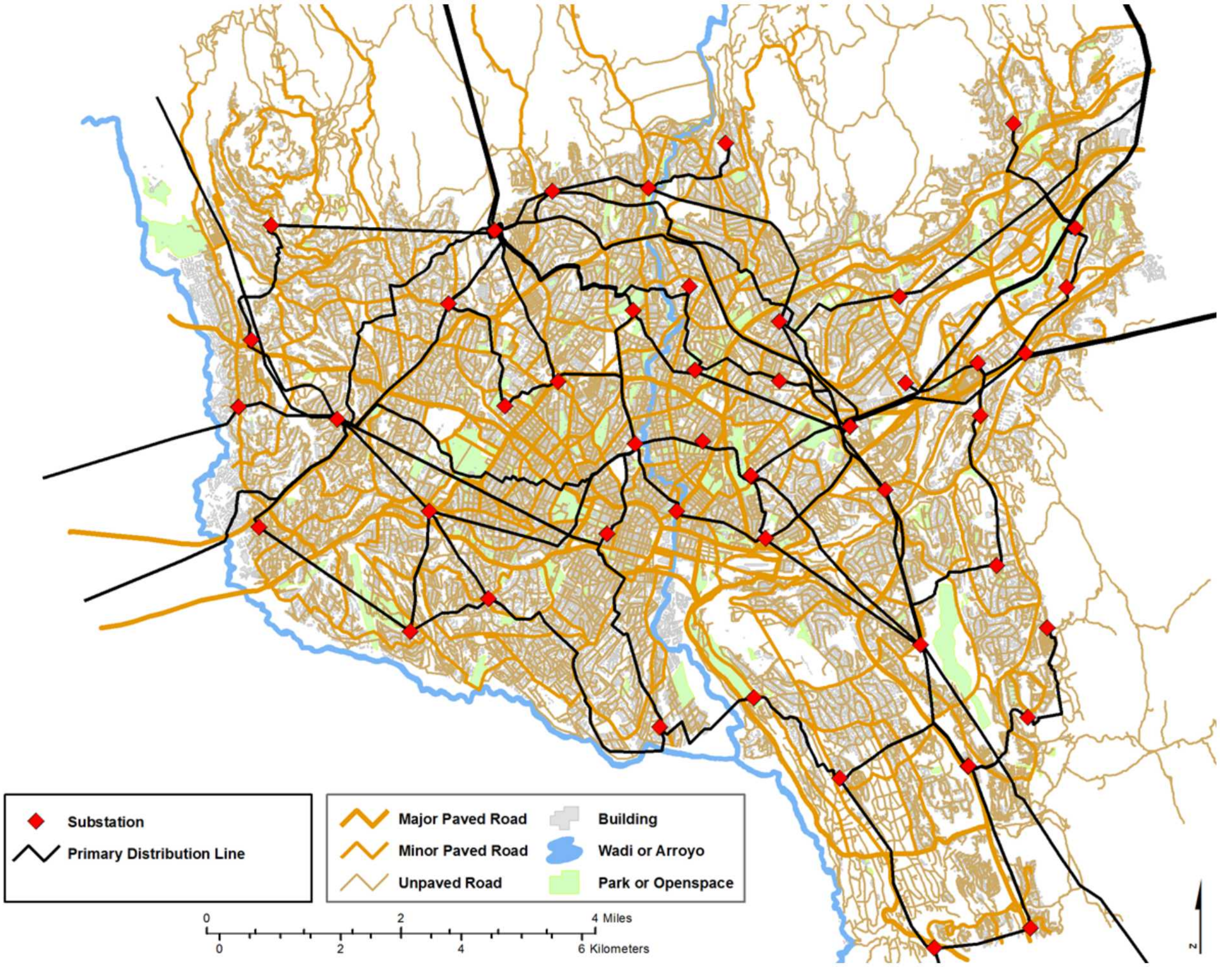


Figure 2-5. COMPASSville Electric Power System

Under normal conditions, background failures are included in the simulation. Network components subject to background failure include buses, loads, generators, switches, lines, and transformers (538+420+24+507+507+59 = 2055 components). The probability of failure is selected from a lognormal distribution with a threshold of 2.85e-5 (1 failure every 4 years). The duration of most background failures are expected to be short. In the simulation, the duration of each failure is selected from the categories shown in Table 2-1 using an exponential distribution.

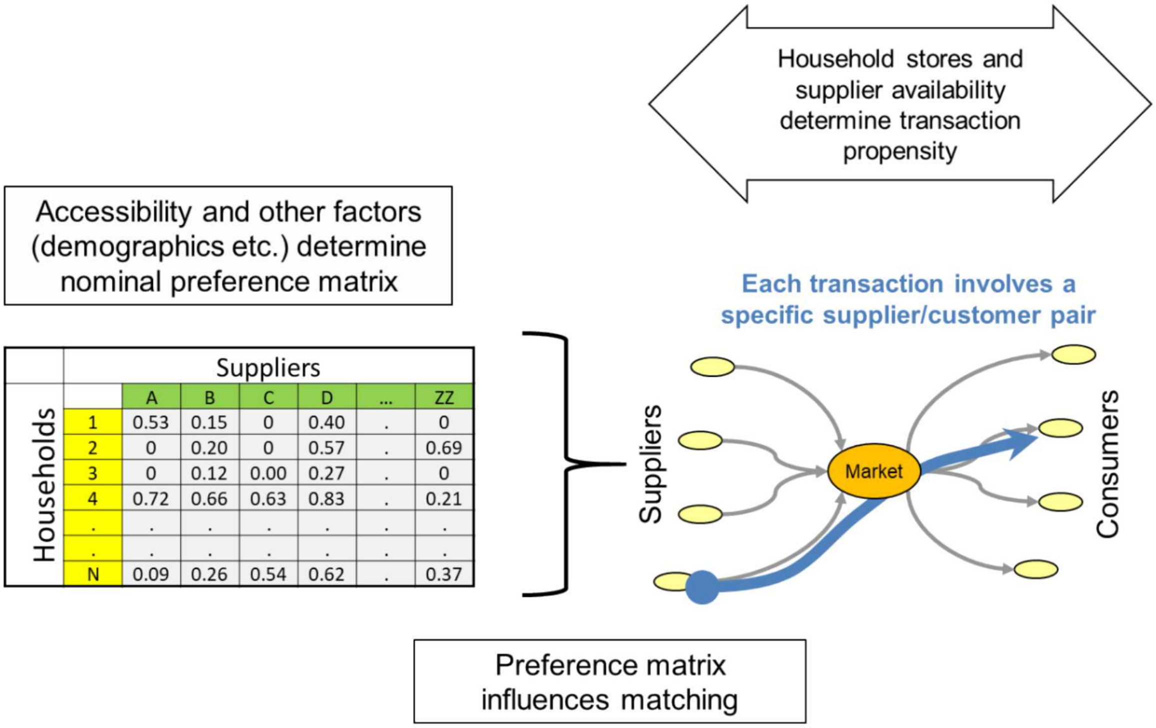
Table 2-1. Electric Power Component Failure Duration Categories

Category	1	2	3	4	5	6	7	8	9	10
Duration	1-2 hours	2- 5 hours	5- 12 hours	12- 24 hours	1- 2 days	2-4 days	4-7 days	7-15 days	15-30 days	1-2 months

The electric power model is dependent on several other models. The fuel model supplies diesel fuel for the generators in the system. The fuel storage at generators is measured in time of operation. If fuel levels are less than 6 hours of operation, the generator does not turn on. The model is also dependent on the transportation model for restoration services and access from the service yard to the damaged asset. If there is no access or access is delayed, then the restoration is either not possible or is delayed, respectively.

2.4. Food Model

The GZTR food model is a configuration of the Exchange model [1] that represents sources for meat, grain, and produce, consumption of food by households, and the grocers and restaurants that connect supply and consumption. Each producer, importer, household, and intermediary holds inventories of food. They manage these inventories through transactions in markets that connect prospective buyers with prospective sellers as shown in Figure 2-6. Disruptions to food supplies naturally induce increases in prices, and substitution away from scarce food types and toward more available types.


Household stores and supplier availability determine transaction propensity
Accessibility and other factors (demographics etc.) determine nominal preference matrix
Each transaction involves a specific supplier/customer pair
Suppliers
Market
Consumers
Preference matrix influences matching

		Suppliers					
		A	B	C	D	...	ZZ
Households	1	0.53	0.15	0	0.40	.	0
	2	0	0.20	0	0.57	.	0.69
	3	0	0.12	0.00	0.27	.	0
	4	0.72	0.66	0.63	0.83	.	0.21
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	N	0.09	0.26	0.54	0.62	.	0.37

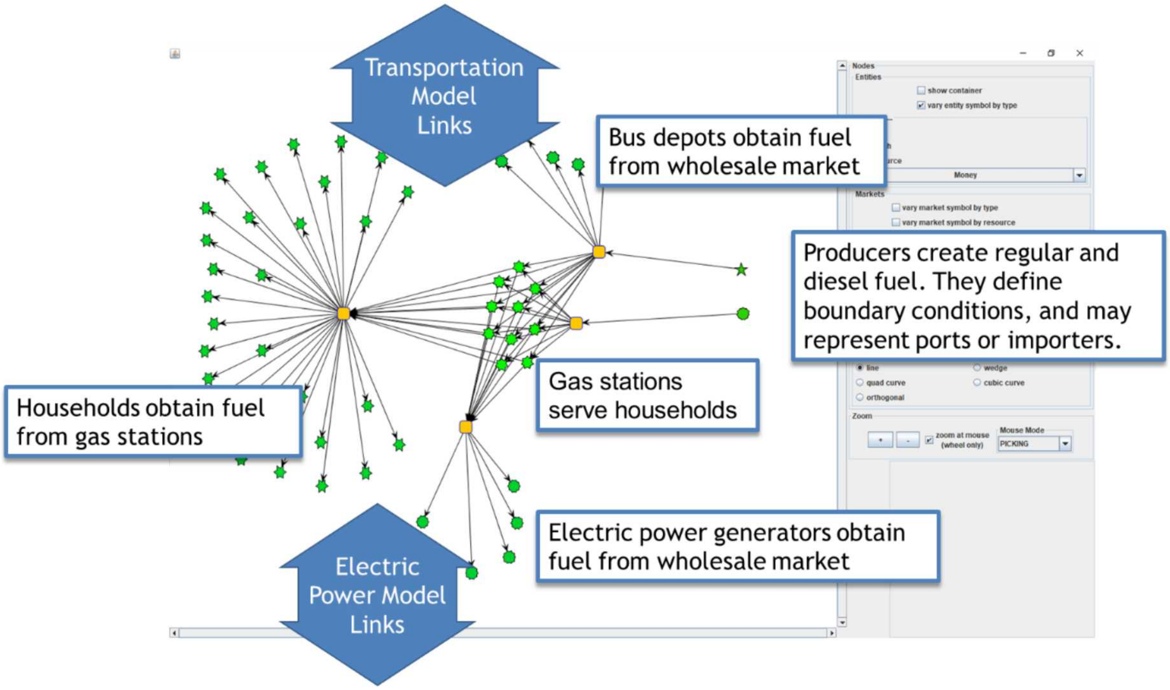
Figure 2-6. Sample Preference and Market Matrix

Household entities represent the collection of individual households in a specific TAZ that fall into a specific economic category (poor or non-poor). Poor and non-poor households are distinguished by the baseline composition of their diet and by money inflows. There are two aggregated importers for each food type – one notionally representing the port at Berbera and the other overland trade from Somalia and other border countries. Food importers supply grocers, who in turn sell the basic commodities to households. Restaurants are also supplied by importers. They produce meals, which form a fourth consumption category for households. A large central market also supplies food to households. It is fed by a set of 10 small producers for each food type, representing regional agricultural output.

Households change their consumption patterns in response to relative scarcity of food, and also based on shifts in their disposition. A perception of food scarcity can lead to increase in target inventory of food, thereby increasing purchasing rates and further exacerbating scarcity. The overall food consumption rate for a household determines its health. Changes in health are used to trigger transitions in sentiment in the population model.

2.5. Fuel Model

The GZTR fuel model is also implemented as a configuration of the Exchange model. Both regular and diesel fuel production and consumption are modeled. Both are imported from the Berbera port, which is modeled as an aggregated importer for each fuel type. Regular is sold to gas stations, who in turn serve households. Diesel is also sold to households via gas stations however the primary consumers for diesel fuel are electric power generators. Transportation depots (for both busses and utility truck yards) also use diesel fuel this is illustrated in Figure 2-7.


Transportation Model Links
Electric Power Model Links
Nodes
Entities
show container
vary entity symbol by type
Money
Markets
vary market symbol by type
vary market symbol by resource
line
wedge
quad curve
cubic curve
orthogonal
Zoom
zoom at mouse (wheel only)
Mouse Mode
PICKING

Bus depots obtain fuel from wholesale market

Producers create regular and diesel fuel. They define boundary conditions, and may represent ports or importers.

Gas stations serve households

Households obtain fuel from gas stations

Electric power generators obtain fuel from wholesale market

Figure 2-7. Fuel Model Interdependencies

Shortages of fuel have the potential to disrupt electric power and transportation. As in the food model, household consumption is modeled using two household objects per TAZ, each representing an income category. Consumption rates are higher in non-poor than in poor households. As in the food model, changes in population sentiment regarding fuel availability can induce hoarding, and sustained problems in obtaining fuel induce changes in household perception.

2.6. Transportation Model

The GZTR transportation model captures the vehicular and non-motorized travel in the region. Behavioral in nature, the model estimates trips, splits them into distinct travel modes, and then assigns the trips to the road network to capture congestion and travel times. The model is structured as a standard four-step travel demand model consisting of trip production/attraction, trip distribution, mode choice and time-of-day split, and assignment. The model outputs include travel times and distances between TAZs, network link congestion, and aggregated in-vehicle times experienced by the population.

In the model, the basic unit of travel is the trip, defined as uninterrupted travel between an origin and a destination. Because a majority of travel is round-trip in nature, the model uses the concept of trip productions and attractions, where productions can be thought of as the origin of a round-trip, and attractions the destination. One modeled production-attraction (PA) trip represents two physical origin-destination (OD) trips: the outbound and inbound segment of the round-trip. While not an exact representation of all travel (which may include intermediate stops), the aggregate travel behavior of a region can be adequately captured using this abstraction.

With this in mind, the following discusses in detail the four steps of the model:

1. Trip Production/Attraction: The number of productions and attractions per zone are separately calculated for both home-based trips and non-home-based trips. Home-based trips start and end at home and are further segmented into work, school, and other.
2. Trip Distribution: Production and attraction vectors from step 1 are distributed into a PA matrix via an algorithm that inversely weights travel time between TAZs (so TAZs farther apart are less likely to be connected than those closer).
3. Mode Choice and Time-of-Day Split: PA trips are separated into travel modes via a nested logistic regression model which calculates mode probabilities based on travel times, travel costs (e.g. fuel, bus fare, vehicle wear-and-tear), and comparative mode preferences. The modes available are walk, bicycle, auto, scooter, shared truck (truxi), and transit. A shortest-path optimization is used to build the transit paths. Once the mode split has occurred, the PA matrices are split into time-period specific OD matrices by performing weighted sums of PA matrices and their transposes – the PA matrix represents the outgoing OD trip, and the transpose the return OD trip.
4. Assignment: The vehicular modes – auto, scooter, and shared truck – are assigned to the network with proportionality factors (e.g. scooters count as 0.6 of an auto) to calculate congestion and travel times. The assignment procedure iteratively averages the results of assigning all trips to the shortest path by time between TAZs. The effect of congestion on travel times is calculated using volume-delay functions, which vary by road type.

The transportation model is sensitive to a variety of inputs from other GZTR models, such as the preferences outputted by the socio-behavioral population model, or the demand fulfillment from the fuel model. To reduce model runtimes, for a given network configuration, the model calculates results for all of the extreme point combinations of outputs from other models, and calculates its actual results as a linear combination of these extreme points based on the other models' actual outputs.

2.7. Twitter Model

The GZTR Twitter model communicates some basic level of sentiment across the entire COMPASSville society including other entities such as the Somaliland or Somalia government, or the population of Somalia. The Twitter model generates hashtags into a feed; these hashtags can be generated intrinsically within the code, triggered by another model (e.g., Populace model), or generated by the scenario scripts or the performers. The model consists of Python code as well as a data store that contains hashtag distribution data for each of the infrastructures and a noise category. These distributions contain roughly one-day long lists of hashtag volume per hour with a sample hashtag for the distribution as the file name. The data was created based on real world Twitter data, then matched with appropriate names for use in the model. These data files contain all of the information the Twitter model uses to populate its feed during a run.

The Twitter model generates a feed consisting of ten hashtags that are dynamically updated each time step (i.e., one hour). To generate a new hashtag, three parameters have to be specified: bucket ('transportation', 'food', 'fuel', 'power', 'noise'), tag (to be added to the feed), and intensity (to scale the distribution curve). Any of these items may be omitted, leading to default values for each. If no bucket is selected (or a nonexistent bucket), then the default is 'noise'. If no tag is selected then a random tag is taken from the selected bucket, and if a nonexistent tag is selected then a random distribution is chosen from the bucket and paired with the provided tag for the feed. Intensity values vary between 1(low), 2(med), and 3(high) and default to the lowest intensity (1) in the case of missing or out of bounds intensity values.

At startup, the Twitter model selects ten hashtags from the noise bucket to populate the first feed. Then, as time advances, the associated hashtag intensities and time series are stepped through until each one reaches the end of its duration. When a hashtag completes its duration, a new hashtag is generated in its place. Additionally, at each time step, any exterior actions that spawn hashtags as well as three randomly generated hashtags from the noise bucket are added to the feed. These items are then sorted by their current intensity, highest to lowers, so that only the top 10 hashtags are within the Twitter feed at any time. If, at any point, a new hashtag is spawned (whether via external action or normal operation) that is already in the feed, the current item is reset to a new distribution and a new item from the same bucket is spawned as well.

Intensities of hashtags control the number of mentions produced. Upon selecting a distribution, the data is then normalized to a new maximum value based on intensity. The Twitter model takes in actions for new hashtags to enter into the feed. These can come from TA1 performers, the scenario script, or other GZTR models.

2.8. Socio-Cognitive Representation of Population Behaviors

To simulate how grey-zone conflict behaviors affect a population, both the Somaliland poor and non-poor populations were modeled. The poor Somaliland population represents people at the lowest levels of socio-economic status (SES). The non-poor Somaliland population represents people at all other levels of SES, which includes the small middle class and wealthier people. The entire population of the city is divided into groups based on TAZs.

To represent the decision making and behaviors of these populations, a reduced version of the socio-cognitive modeling framework, DYMATICA, was used. DYMATICA (DYnamic Multi-scale Assessment Tool for Integrated Cognitive-behavioral Actions) is a computational modeling and simulation approach that helps decision makers better understand and anticipate the decision calculus of populations, groups, and governments within societal systems. DYMATICA simulates the dynamic psychosocial, geopolitical, and socioeconomic interactions within and between actors. The decision-making of actors is represented within socio-cognitive models that are embedded within the larger GZTR framework. In addition to SES, groups in the GZTR are based on members' home TAZ.

Groups within the populace model act as a singular community, where at each time step, the community assesses the state of the system (e.g., local food and fuel availability, traffic patterns, and power) and develops a general plan of action for the day (i.e., go to work, obtain gas, go to the grocery store, etc.). Groups have both static (i.e., TAZID, socioeconomic status [poor or non-poor]) and dynamic (i.e., sentiment) properties. The GZTR populace model incorporates inputs from all four infrastructure models and generates outputs that are digested by all four infrastructure models and the Twitter model.

Each modeled population entity can perceive and respond according to their modeled decision calculus, which is affected by such things as their modeled perceptions, expectations and discordance (the difference between their expectations to what they are perceived to be experiencing), intentions, and ultimately behaviors. Modeling the decision calculus of each entity comprises of capturing and representing key decision elements pertinent to those entities and the scenario. These are derived from reports and data with respect to a modeled entity's perceptions, motivations, norms, and behavioral intentions associated with the scenario and include modeled entity's history and culture. This information is structured to characterize both the shorter-term (12 hours) and longer-term (such as one year) decision processes of populations. The behaviors of the modeled entities are based on actual behavioral responses to grey zone conflict activities that have occurred in the past within similar circumstances and environments.

The update process for the model happens on a discrete time basis every 12 (simulated) hours and involves: 1) assessing the infrastructure state of each TAZ, 2) comparing these states to baseline values, 3) calculating sentiment, and 4) generating a list of priority actions. In the model, the state of the infrastructure, at each given time stamp, provides a weighted set of cues. Together, the cues provide a general perception of the modeled populations' environment and their welfare. Infrastructure state assessment involves reading in different variables for each of the infrastructure models. For example, for the power model, this involves reading in the current fraction load served and fraction load served mean variables, and interpreting, from the values, the number of power-related issues the TAZ has experienced during the last 12 hours. Similarly, for transportation the hours traveled, state of the roads, and congestion in the TAZ are considered, while for food and fuel the deficiency level and consumption rate for the period are considered. These values are transformed into a [0,1] range of how bad the perceived state of the infrastructure is, then scaled based on the general sentiment value (such that TAZs that are already frustrated are more sensitive to further disturbances). This value is then compared to a baseline expectation value to determine the discordance in the population. The discordance is adjusted based on the poor vs non-poor breakdown of the TAZ (where non-poor populations get upset faster with a lower ceiling relative to poor populations) and a decay function (to include memory of past sentiment value for each infrastructure that persists weakly in time). Sentiment values are calculated for each infrastructure and then combined to form a general sentiment value; these values can be queried for a given TAZ using surveys (EconomicSurvey for food and fuel sentiment and SocialSurvey for transportation and power sentiment).

Once all the sentiment values have been calculated, they are used to create a list of priority actions for the TAZ for the next 12-hour period. The priority actions include going to work, going home, going to the grocery store, and going to get gas. By default, the TAZs alternate between home (for PM) or work (AM) as the priority. As infrastructure degrades, these priorities shift; for instance, if concern about fuel goes up, so does the desire to go to the gas station. These location-specific priorities are read in as behavioral triggers in the infrastructure models; power demands increase when people stay home, food and fuel have increased demand with more TAZs going to their respective stores, and traffic patterns change as new trips are spawned based on new priorities and needs of the communities.

Additionally, when TAZs increase in sentiment beyond predefined thresholds, they contribute to the overall unrest within the entire society. As societal unrest increases, actions are sent to the Twitter model to spawn new hashtags relating to the infrastructure that caused the unrest. These hashtag triggers can happen at different intensity levels and can happen multiple times in a time step depending on how widespread the current unrest is.

2.9. Player Utility

The GZTR Player utility coordinates interactions between the GZTR models and both scripted and TA1 performer actions that perturb or query the infrastructure and population models. Some scripted actions represent intentional actor behavior, such as manifestations of the LOEs being pursued by Red actors and actions by White and Blue actors. Some scripted actions correspond to natural disasters or other exogenous shocks (see Section 2.10 below). Player uses a set of rules that it reads on initialization to route the actions it receives to the appropriate model(s). Models receiving actions respond with acknowledgement messages back to Player. Performers' probing actions are also mediated by the Player utility in the same way as scripted actions.

Some categories of probing actions involve protecting or monitoring parts of the infrastructure. Player has an active role in implementing such actions because they involve the way future actions are processed. Protective probing actions block disruptive actions directed at protected portions of the infrastructure. Monitoring actions result in the disclosure of actions to performers within the scope of the monitoring action.

Besides processing scripted and performer actions, Player is also responsible for executing the scenario's event script. This involves relaying scripted events (CAMEO+) to performers via the Controller at the appropriate time. Event scripts can include probabilistic events and event chains; however, this feature was not exercised during the evaluation.

2.10. The Grey Zone Scenario Scripts

For the GZTR to simulate grey zone-type conflicts amidst day-to-day activities, Sandia developed two grey zone scenario scripts, which enabled the training and evaluation of the performers' technologies. These scripts were used to simulate grey zone activities and the typical dynamics of a medium-sized city within the Horn of Africa over the course of a one-year period per script. The scripts also guided the behaviors of Red (Somalia, China, Russia, and al-Shabaab), White (Somaliland government), and Blue (U.S.) actors which unfolded within the scenario.

The first scenario focuses on China's desire to acquire land in Somaliland in order to build a naval base. In order to do this, China engages in activities that increase Somaliland's need for Chinese political and economic support. China also engages in activities that make it easier for Somaliland to become a fully independent state. Conversely, Somalia engages in activities that make it more difficult for Somaliland to become independent. Countries such as Russia also engage in behaviors that promote their interests. The violent extremist organization, al-Shabaab, seeks to take advantage of conditions and events occurring in Somaliland to promote their position.

The second scenario extends the first scenario into a second year with China securing a naval base and having permission to explore and mine Somaliland for oil and minerals. In the scenario, China discovers a very important strategic metal, beryllium. China seeks to hide the fact that beryllium was found. However, when the Somaliland government uncovers this, they void the agreement with China. In response, China engages in activities aimed at punishing and persuading the Somaliland government to allow them to mine the beryllium. Somalia, asserting ownership of this territory, engages in activities to disrupt the mining of this strategic metal. Al-Shabaab also seeks to disrupt mining by engaging in terrorist activities. Moreover, Russia seeks to persuade Somaliland government officials to lease the land to a Russian mining company. In addition, the U.S. seeks to persuade Somaliland government officials to lease the land to a U.S. mining company.

The scripts consist of daily event descriptions that are relevant to the grey zone scenario along with events that serve as noise. Noise events are meant to simulate realistic day-to-day activities and were generated from various news and media sources; noise events range from social/community events, like concerts or sports events, to natural disasters which impact daily functioning to varying degrees. Each simulated day consists of events that occur within Somaliland in the first half (am) and/or latter half of the day (pm). These events can be:

1. Acts of nature, such as a rainstorm, sandstorm, or plague. These types of events are randomly placed within the scenario (typically lasting for multiple days to weeks) and can serve as a catalyst for grey zone activities, along with potential Somaliland population dissatisfaction with the Somaliland government.
2. Acts of governments, such as formal and informal pronouncements, trade and military agreements, infrastructure repair and development, and military actions. These acts can come from simulated Red and Blue actors and the Somaliland government in response to the actions of each other, the actions of al-Shabaab, the populations of Somalia and Somaliland, man-made disruptions and acts of nature that include such things as infrastructure interruptions and failings.
3. Acts of a violent extremist organization (al-Shabaab), such formal and informal pronouncements, military actions against governments, and acts of terrorism against populations. Acts by al-Shabaab are typically in response to inadvertent opportunities, changes in status, or acts of a government.
4. Acts of the Somalia populations, such as protests against the Somaliland government and social media behaviors attacking the Somaliland government and citizens. Most of the Somaliland population behaviors are modeled via the socio-cognitive behavioral model described in the prior section. The scripted actions of Somaliland populations that are not modeled in the socio-cognitive model are represented in the scripts

Each simulated half-day might consist of one or more events that occur simultaneously with other events. In its entirety, the events play out as "acts in a play" with plots and subplots where performers need to uncover grey zone-related actions within the larger number of natural and man-made events and conditions. Information available to performers for analysis, at each time-step, consist of fictitious stimuli, such as:

1. External media broadcasts consisting of television text banners from Somaliland government-influenced media outlets. The text banners consist of a) information directly related to the scenario, providing cues to the grey zone events that are occurring in the scenario; b) information that provides misleading evidence (which was called "tease information"), making it more difficult to determine actual grey zone activities[1] and; c) information completely unrelated to the scenario and considered to be noise. The degree of one type of banner information versus another can be adjusted to increase or decrease the difficulty of uncovering actual grey zone intent and activities.

[1] Tease events were classified as either "diverting" (in that it provided information that potentially diverted the assessor away from the correct path of reasoning) or "reinforcing" (in that it provide information that reinforces the primary storyline) to describe the extent to which they challenge the performer's ability to parse out grey zone activity from misleading evidence within the scenario.

2. Social media reports that provide overall social media trends from the Somaliland population. This information, coupled with actions that drive the Twitter model, shows the length and degree to which specific social media topics are trending. These trends are from the perspective of Somaliland populations.
3. U.S. State Department intelligence and security-related bulletins. These bulletins represent formal documents that are either derived from the U.S. Consulate in Somaliland or from U.S. intelligence and military agencies that issue reports. These documents are from the perspective of the U.S. Department of State, Department of Defense, or the intelligence community.
4. Polling data that provide the general sentiment of the Somaliland population.

The information described above was first developed in a natural language format and was then reconfigured into a triplet keyword format consisting of actor-action-object summaries. Events were encoded in a Sandia-developed formalism called CAMEO+: an extension of the CAMEO formalism used to encode geopolitical events. Since the COMPASS program is focused on ambiguous grey zone activities, including infrastructure events and associated Red actions, the CAMEO formalism was extended to include such activities. CAMEO+ formalism can now be applied to a wide range of non-military actions, greater diversity of social and political activity, and naturally occurring events that potentially impact grey zone activities. Many of these events also involve specific actions that take place in infrastructure models, such as creating an electric power outage, fuel shortage, or a Twitter feed on a specific topic. These scripted actions are implemented using the same construct that is available to the performers for probing actions.

2.11. Interacting with the Grey Zone Test Range

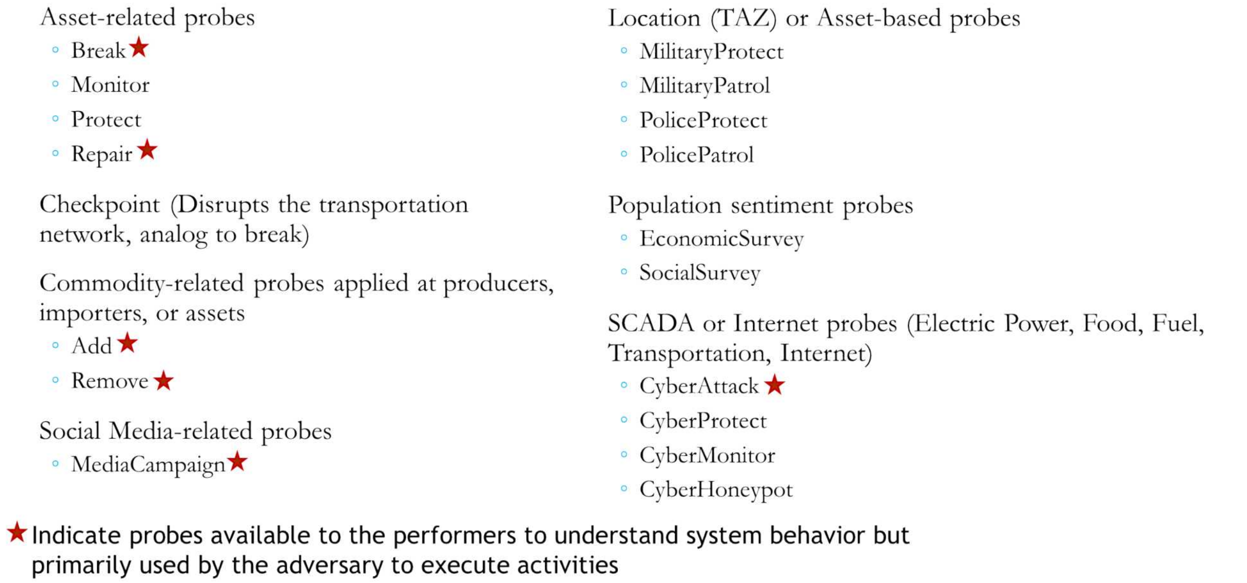
Interactions with the GZTR happen through actions that are also called probes. Probes are designed to either impact the GZTR or gather additional information from the GZTR. The scenario script implements probes to cause particular effects within the GZTR models, while TA1 performers can both impact GZTR models or execute probes to obtain additional information from the GZTR.

Some information is readily available in the GZTR and is provided via simulation variables and data feeds. This is information that would be easily observable in a real-world environment such as:

- Average pricing information for commodities (e.g. grain, produce, regular fuel)
- Random sentiment measure of the population
- Operating status of assets (e.g. restaurants, grocery stores, and gas stations)
- Power outages
- Traffic congestion
- Twitter feed

Additional information on the environment, that is not generally available, can be gathered by probing. Probes are available to the TA1 performers to watch an asset, TAZ or collection of TAZs, or a network. These probes could be used to provide clues about Red actor activities. While control system and Internet networks were not directly modeled in the GZTR, information on Red activities on control systems for electric power, food, fuel, and transportation and general Internet denial of service activities could be obtained by probing. Probes can also provide pricing and inventory information for an asset. For example, probing a gas station can provide the price and inventory for regular and diesel fuel at that gas station. Social and economic survey probes can provide sentiment related to food, fuel, electric power, and transportation for the population in a TAZ.

Probes that impact the GZTR models are primarily used to implement the scenario scripts as Red actor actions or natural events; however, these probes can also be used by TA1 performers to examine and change model behavior. These include the ability to repair or break assets, add or remove commodities, and execute social media campaigns. A full list of available actions and probes are show in Figure 2-8.



Asset-related probes

- Break ★
- Monitor
- Protect
- Repair ★

Checkpoint (Disrupts the transportation network, analog to break)

Commodity-related probes applied at producers, importers, or assets

- Add ★
- Remove ★

Social Media-related probes

- MediaCampaign ★

Location (TAZ) or Asset-based probes

- MilitaryProtect
- MilitaryPatrol
- PoliceProtect
- PolicePatrol

Population sentiment probes

- EconomicSurvey
- SocialSurvey

SCADA or Internet probes (Electric Power, Food, Fuel, Transportation, Internet)

- CyberAttack ★
- CyberProtect
- CyberMonitor
- CyberHoneypot

★ Indicate probes available to the performers to understand system behavior but primarily used by the adversary to execute activities

Figure 2-8. Available GZTR Actions or Probes

This page left blank

3. USER EXPERIMENT

In support of the user experiment, conducted in December 2019, Sandia created a storyline for Red actors pursuing distinct courses of action developed from existing exercise information. A specific subset of exercise information was identified by INDOPACOM to represent storylines that they wanted the TA1 performers to be able to interpret.

In order to turn the existing information into a consistent storyline, Sandia focused on a specific region and threads of interest and selected elements that supported a coherent storyline consistent with the COMPASSville simulation environment, four possible actors, and CAMEO+ events. The storyline focused on four actors -a near-peer adversary, a smaller nation-state, and two violent extremist organizations. One of the actors was introduced by Sandia in order to separate activities from a single actor focused on both outright destruction and more subtle activities. This resulted in an actor that was easier to detect and one that was more difficult. Exercise material was selected to allow for variation in the difficulty to detect the actors in play- (1) one actor employed very consistent tactics and was easily detectable; (2) another actor employed more subtle actions; (3) detection of the third actor required some inference by the performers; (4) the fourth actor had a minor role with very few activities and was included to bring the total to four and align it with the COMPASSville simulation. The CAMEO+ event stream was created from the corresponding exercise injects, changing the actor's name, as required. The CAMEO+ event stream was provided via a streamlined version of the GZTR through the TA3 interface to the GZTR. The streamlined version did not include infrastructure, socio-behavioral, or Twitter models.

This page left blank

4. TEST AND EVALUATION

As part of the test and evaluation team, Sandia designed and implemented a testing process to assess performers' solutions against the program objectives. Sandia created definitions for the key aspects of adversaries' intention – goals and strategies – which were the objects of the performers' estimation. Sandia designed and created the GZTR simulation environment as a testbed for performers' solutions. Data, collected during the evaluation, is analyzed against program criteria. This section presents the evaluation approach and specific measures that capture the qualities laid out in the COMPASS BAA. It provides specific definitions for the objects to be estimated, discusses the structure of performers' estimates, and describes the metrics that compare estimates to the ground truth. The ground truth for the evaluation is provided through the GZTR.

4.1. Approach

COMPASS technologies are designed to discover the goal of Red actors, if they exist, and the strategies they use to pursue their goal. TA1 performers are to infer goals and strategies by observing the behavior of the GZTR, and by introducing probes designed to induce responses that clarify goals and strategies without provoking unwanted reactions. The evaluations are designed to quantify the quality of the estimates produced by performers' technologies, and the effectiveness of the probing actions they recommend.

The evaluation approach is designed to allow an adjustable level of complexity within the GZTR, since it is unclear which aspects of the objectives are the most difficult, nor how technologies differ in performance among component task. It is important for the evaluation framework to support a range of complexities so that the training and evaluation can be made difficult enough to learn from.

4.2. Defining the Objects to be Estimated

"Goal," "intent" and "strategy" are used with different meanings in different contexts, and with greater or lesser precision. Precise definitions are essential for determining the output from the performers' technologies, and for developing reasonable measures for the degree of departure of that output from actual properties of the system. Operational definitions, grounded in the (in principle) observable state of the system, are necessary for this purpose. However, the conception of "goal" and "strategy" must also correspond, as far as possible, to the use of these terms in other discourses. The quality of the estimates of goals and strategies, as those terms are defined here, should be good indicators of performance in the field.

The following definitions are adopted to meet these criteria:

Goal: A state of the system that an agent seeks to bring about by acting on it

Strategy: Rules describing how an agent acts on the system to pursue their goal

These definitions were selected since they are sufficiently grounded in potential observations to lead to clear measures of estimation quality. They are also sufficiently flexible to admit conceptions of goals and strategies covering a wide range of scopes and time frames.

4.2.1. Goals

Limitations on what can be defined as a goal derive from the boundaries of the system, and the time frame of concern. System boundaries, in turn, are established by the sources of information used to feed performers' tools. For the GZTR this scope is limited to the geography and actors explicitly included in the simulation. Goals refer to states that might exist in the infrastructure or population of COMPASSville, or in the dispositions and actions of the governments and other actors included in the model. For example, establishing a government aligned with a fundamentalist Red actor is an admissible goal: this state can be achieved and observed within the system. However, positing that Red's real goal is global hegemony, and that establishing a sympathetic COMPASSville government is merely instrumental to that goal, is not admissible. Pursuit of hegemony cannot be detected through observing the GZTR alone, so it cannot be set as a target of estimation. Global hegemony could be defined as a goal under the proposed definition, however a global GZTR would then be required to create room for Red's pursuit of goals of that scope.

The definition of "goal" entails that agents care about the system being in a specific state. We denote the complete system state by s(t), where s is a vector in a high-dimensional state space comprising the possible values for the tens of thousands of system variables. The functional $Z(s(x) \mid 0<x<t)$ maps system states into a low-dimensional valuation space V. The dimensions of V are the set of indices or conditions that Red might potentially care about, for example the degree of popular support for the local government, the existence or not of official recognition of COMPASSville independence by Blue, and the growth rate of the local economy. Red's goal is described as a preference function over V. We simplify specification of this preference function by first discretizing any continuous components of the valuation space V into a small number of bins, and then making the preference value for being in a bin binary.

Defining Red's goals as sub-regions of a discrete low-dimensional space explicitly connected to the system state space creates a clear analytical target for performers' estimation algorithms. Even when applied against the GZTR, the complex interacting dynamics in the system are expected to make goal identification a challenge. Should this not be the case, the problem can be made more difficult by increasing the dimensionality of the goal space or even eliminating the discretization and defining goals as continuous preference functions over valuation space.

4.2.2. Strategies

Red's actions in pursuit of their goal are assumed to be governed by a set of rules, which constitute the persisting strategy object to be estimated by performers' technologies. Rules connect the current state of the system, including Red's internal state, to an action to be performed. The game-theoretic definition of "strategy" posits a persistent logical structure governing Red's actions in pursuit of their goal. Red's actions are assumed to depend on conditions they observe in the system, and on their experience. This broad definition emphasizes empirical verifiability grounded in the system state. It covers both "strategic" decision-making (i.e. long-term behavior with only general specification of the actions taken) and "tactical" decision-making (i.e. short-term behavior with detailed action specification).

Agents' actions are influenced by their past actions, the outcomes they've observed, the resources available to them at the time, and other factors known to them but generally unobservable. These are properly aspects of the total model state. However, it seems clearer to formally distinguish the agent's internal state information from the generally-available information about the external system. This factoring allows for a common condition space to be defined for all agents, and for an agent's decision rules to be described using a state diagram in which distinct states are associated with distinct rule sets mapping conditions to actions. This factorization allows specification of an agent's overall strategy in terms of sequential sub-goals, experimental actions whose outcomes steer their subsequent pathways, and other constructs used to describe and plan complicated activities. An agent's state diagram may be hierarchical, with a top-level state associated with a sub-state diagram that represents finer-scale sequences or contingencies.

To communicate more clearly with prospective system operators, state diagrams used in COMPASS are organized into Lines of Effort (LOEs) while component states correspond to tasks in these LOEs.

4.3. Estimates

Performers' technologies provide estimates for Red actor goals and associated strategies based on their observation of the system's evolution, and interpretation of the effects of probes they introduce. Performers should be able to convey the level of confidence they associate with these estimates, and to provide alternatives that represent the range of possibilities they consider consistent with the information available to them when the estimate is made. In addition to these descriptions of uncertainty, performers may wish to provide estimates having greater or lesser precision as a way of conveying confidence.

Because the goal space is known, a goal estimate corresponds to a region of goal space supposed to coincide with Red's goal. Estimate uncertainty can be conveyed by enumerating alternatives and assigning likelihoods to each.

Red's unknown strategy leads them to take a sequence of actions intended to accomplish their goal. By observing the model state space, performers form hypotheses about Red's goals (as discussed above) and, in conjunction with these possible goals, infer certain actions by Red. These actions are minimal descriptions of Red's strategy, in the sense of being estimates of the strategies outputs when the observed system trajectory defines its input. Estimated action sequences may allow the rule sets that define strategies more generally to be estimated, provided there are a set of a priori strategies available to constrain the space of possibilities.

A strategy estimate consists of an enumerated set of possibilities, each associated with a weight or likelihood. Each possibility is minimally a description of a sequence of actions but may be a more general function connecting conditions to action descriptions. Few constraints are placed on the form of performers' strategy descriptions to allow flexibility in expressing uncertainties and imprecision about their estimates of this functional. Any functional form that produces a sequence of action descriptions from an input sequence of system conditions can be used. Weights or probabilities can be assigned to specific alternative hypotheses, as with goals. The fundamental measure of similarity of a hypothetical strategy to the real strategy is operational.

4.4. Metrics

The quantitative evaluation is designed to characterize the quality of performers' estimates of goals and strategies. Because the program criteria were specified in terms of classifier performance (e.g. accuracy, precision, recall) the objects of estimation were construed as binary functions over a discrete space of possibilities. In this framework the ground truth is described as a function whose value is 1 in the regions of goal space that satisfy Red's goals, and false outside of that region. Hypotheses about goals are then a set of binary functions and associated probabilities. For each element, the distance from the hypothesized goal to the ground truth is measured by comparing the correspondence of true and false elements using the classifier metrics.

Given system observations up to some time, T, performers provide goal estimates as a list of descriptions of the subregion of goal space defining Red's goal, along with an associated likelihood or weight reflecting their level of confidence in this description. Identifying the real goal is correctly classifying each of the points in the discrete goal space. Construed in this way, metrics designed to describe the performance of classifiers (precision, recall, accuracy, specificity) were used to measure the distance between the true goal and any specific hypothesis. Because performer's goal estimates are a set of specific alternatives with associated weights or likelihoods, each estimate produced a distribution for each of the four measures.

Passive observation of the system, over time, provides information for improving goal estimates. Both the rate at which the estimate improves, and the asymptotic limit to the quality of the estimate, are important measures of the convergence of the goal estimation process. The maximum learning rate for any estimation process, however, may be limited by the dynamics of the system itself. Learning rates are however useful for comparing among approaches.

These same measures of information acquisition characteristics can be directly extended to characterize probes. The purpose of a probing action is to clarify ambiguities that the available system information allows in interpreting goals and strategies. An effective probe should lead to a clear improvement in these estimates.

The information gain, in the sense of reduction in estimation error, and latency can be measured by observing the effect of the probe on goal (and strategy) estimates over time.

A strategy estimate approximates the behavior of Red's strategy. As with goals, the estimate consists of a set of alternative strategy descriptions with associated weights or likelihoods, allowing performers to express different degrees of evidential support. Also paralleling goal evaluation, each alternative strategy is compared to the single actual strategy by construing strategies as a binary function over a space defined by the actors in the system and the set of lines of effort they might employ.

The COMPASS program created a library of possible strategies to provide a-priori for those performers who did not creating their own. Strategies were defined in terms of component LOEs, possibly related by ordering rules or logical conditions. An LOE described the kinds of actions that might be taken if it were pursued, but did not completely specify the actions by fully defining conditionality, timing, and parameterization.

4.5. Evaluation Findings

The BAA established numerical objectives for several desirable properties of an automated system for discerning the presence and objectives of a malevolent Red actor using probes. Later, constraints were introduced and opportunities were pursued that obviated some of these quantitative goals. The remainder were given an actionable interpretation and were measured during the evaluation exercise.

Based on hypotheses generated in the last portion of the one-year simulation period, three of the four teams met quantitative criteria for accuracy, precision, and (with respect to at least one kind of hypothesis) recall. The fourth team's final estimates did not meet some of the criteria; however, their approach did produce high-scoring hypotheses early in the simulation period.

Some outcomes indicate the need for additional refinement of techniques and/or additional rigor in testing before their performance in the field could be reasonably anticipated.

1. First, there was no observed tendency for the deployed probes to improve the quality of hypotheses. This outcome reflects the very limited time that performers had to assimilate the rules and pragmatics for deploying probes in the simulator, and to understand the responses of active probes in the simulator, which is necessary information to infer possible causative Red actions. This outcome does not implicate the value of probing in general or the potential quality of the probes the

technologies might recommend. However, the lack of informational value from probing in the constrained context of the experiment was not recognized by the technologies, which would ideally have discovered that probing would produce little value.

2. Second, while there was some tendency for recall scores to improve for some technologies during the course of the simulation, there was otherwise no evidence that observation of the system improved the quality of hypotheses. This finding needs to be explained. Note that prior information about the various Red actors and their propensities to engage in different lines of effort (LOEs) was a common body of information used to initialize or condition each technique. If technologies gave this information a large weight (either directly or indirectly) relative to new observations, that might have created considerable inertia in their hypotheses. This possibility could be tested by looking at the dynamics of hypothesis structures rather than simply of their scores. A second possibility is that the information available through the simulation was not sufficient, in principle, to implicate the correct set of hypotheses to the exclusion of alternatives.

This page left blank

5. SUMMARY

The GZTR and CAMEO+ were developed for the COMPASS program as part of the controlled evaluation platform. The GZTR can be extended to dynamic adversaries to provide an enriched set of challenges for performer capabilities. Additional scenarios could be developed to assist performers with improved tuning of their technologies and to achieve improvements in recall, information gain, and convergence. The GZTR could also be used as a training platform for operators to gain familiarity with the COMPASS technologies under a variety of possible operational contexts.

The evaluation completed successfully with various lessons learned and new paths forward for the performer teams to continue to evolve and improve their technologies.

REFERENCES

[1] Walt E. Beyeler, Michael Mitchell, A General Model of Interacting Specialists, SAND2019-2743

DISTRIBUTION

Email—Internal

Name	Org.	Sandia Email Address
Mike Valley	01810	mtvalle@sandia.gov
Justin Garretson	06617	jrgarr@sandia.gov
Rossitza Homan	08724	rhoman@sandia.gov
Technical Library	01977	sanddocs@sandia.gov

Email—External ([redacted]

Name	Company Email Address	Company Name
Ashley Calder	ashley.calder.ctr@darpa.mil	DARPA
Ken McCullough	kenneth.h.mccullough@navy.mil	NIWC
John Paschkewitz	john.paschkewitz@darpa.mil	DARPA