**Athena Analyst's Guide**

Athena S&RO Simulation, V6.3

October, 2015

William H. Duquette

Robert G. Chamberlain

David R. Hanks

Brian J. Kahovec

Jet Propulsion Laboratory

*Copyright 2008-2015, by the California Institute of Technology. ALL RIGHTS RESERVED.*

*United States Government Sponsorship acknowledged.*

Table of Contents

Models 6

1. Introduction 7

1.1 Overview 7

1.2 Other Documents 7

1.3 Changes for Athena 6 8

1.3.1 Changes for Athena 6.1 8

1.3.2 Changes for Athena 6.2 8

1.3.3 Changes for Athena 6.3 9

2. Athena Concepts 10

2.1 Model Parameters 10

2.2 Simulated Time 10

2.3 Geography 10

2.3.1 Neighborhoods 11

2.3.2 Neighborhood Proximity 12

2.3.3 Local vs. Non-Local Neighborhoods 12

2.4 Actors 13

2.4.1 Strategies 13

2.4.2 Support, Influence, and Control 13

2.4.3 GOODS Production Infrastructure 13

2.4.4 Stability 14

2.5 Groups 14

2.5.1 Civilian Groups 14

2.5.2 Force Groups 15

2.5.2.1 Mobilization, Deployment, and Assignment 15

2.5.3 Organization Groups 15

2.5.4 Force, Security, and Volatility 16

2.6 Modeling Areas 16

2.6.1 Physical 16

2.6.2 Time 17

2.6.3 Political 17

2.6.4 Military 17

2.6.5 Economics 17

2.6.6 Social 17

2.6.7 Infrastructure 18

2.6.8 Information 18

2.7 Simulation States and the Advancement of Time 18

3. Relationships and Control 20

3.1 Relationships and Affinity 20

3.2 Horizontal Relationships 21

3.2.1 Force and Organization Group Affinities 21

3.2.2 Computing Horizontal Relationships 21

3.3 Vertical Relationships 21

3.3.1 Baseline Vertical Relationships 22

3.4 Actor Support and Influence 22

3.4.1 Direct vs. Derived Support 23

3.4.2 Direct Support 23

3.4.3 Derived Support 24

3.4.4 Total Support 25

3.4.5 Influence 26

3.5 Control of a Neighborhood 26

3.5.1 When Control Shifts 27

4. Force Analysis 29

4.1 Overview 29

4.2 Force Group Stance and Effective Relationships 30

4.2.1 Stance 30

4.2.2 Group Discipline 30

4.2.3 Effective Relationships 31

4.3 Background Criminal Activity 32

4.3.1 Law Enforcement Personnel 32

4.3.2 Suppression of Criminal Activity 33

4.3.3 Nominal Criminal Fraction 34

4.3.4 Actual Criminal Fraction 35

4.4 Measuring Force 36

4.4.1 A Group's Own Force 36

4.4.1.1 Civilian Group Force 36

4.4.1.2 Organization Group Force 37

4.4.1.3 Force Group Force 38

4.4.1.4 Non-Criminal and Criminal Force 39

4.5 Friends and Enemies 39

4.6 Volatility 41

4.7 Security 41

4.7.1 Security and Empty Groups 42

5. Effects of Unit Activities 43

5.1 Force Presence and Activities 43

5.2 Organization Activities 46

5.3 Civilian Activities 46

5.4 Activity Situations 46

6. Abstract Situations 48

6.1 Athena 4 Changes 48

7. Services 49

7.1 Overview 49

7.1.1 Services vs. Abstract Situations 49

7.1.2 Levels of Service 49

7.1.3 Service Cases 50

7.2 Abstract Infrastructure Services (AIS) 50

7.3 Essential Non-Infrastructure (ENI) Services 51

7.3.1 Service vs. Funding 51

7.3.2 Funding by Individual Actors 53

7.3.3 ENI Services and Empty Civilian Groups 54

7.4 Expected Level of Service 54

7.4.1 Categorize the Actual Level of Service 55

7.5 Satisfaction Effects 55

7.5.1.1 The Needs Factor 56

7.5.1.2 The Expectations Factor 58

7.6 Vertical Relationship Effects 59

8. Attrition 61

8.1 Overview 61

8.1.1 Posture 62

8.1.2 Personnel Involved in Engagements 62

8.1.3 Hiding Groups 62

8.1.4 Engagement Life Cycle 63

8.2 Personnel Involved in Combat 63

8.2.1 Visibility and Detection 63

8.2.2 Effective Force 66

8.3 Allocating Force Group Personnel to Engagements 67

8.4 Lanchester Attrition in Athena 67

8.4.1 Force and Personnel Thresholds 67

8.4.2 Determine Postures 68

8.4.3 Attrition Coefficients 68

8.4.4 Combat Time 70

8.4.5 Compute and Apply Attrition 71

8.5 Casualties 72

8.5.1 Force Group Casualties from one Combat 72

8.5.2 Total Force Group Casualties 72

8.5.3 Civilian Casualties 73

8.6 Attrition and Mobilized Troops 74

8.7 Magic Attrition 74

8.7.1 Magic Attrition to Groups 74

8.7.2 Magic Attrition to Neighborhoods 74

8.8 Assessing the Attitude Implications 74

8.8.1 Satisfaction Effects of Attrition 74

8.8.2 Cooperation Effects of Attrition 75

8.8.3 Horizontal Relationship Effects of Attrition 76

8.8.4 Vertical Relationship Effects of Attrition 76

8.9 Deriving the Lanchester Solutions 76

9. Demographics 80

9.1 Connections with Other Models 80

9.2 Simplifying Assumptions 80

9.3 Population and Units 81

9.4 Civilian Group Population over Time 81

9.4.1 Natural Population Change 81

9.4.2 Civilian Attrition 82

9.5 Population Breakdowns 82

9.5.1 Subsistence Population 83

9.5.2 Consumer Population 83

9.5.3 Labor Force 83

9.6 Aggregate Statistics 84

9.6.1 Neighborhood Population 84

9.6.2 Regional Population 84

9.6.3 Labor Security Factor 84

9.6.4 Consumer Security Factor 85

9.7 Unemployment 85

9.7.1 Disaggregation to Neighborhoods 86

9.7.2 Disaggregation to Civilian Groups 88

9.7.3 Geographic Unemployment 89

9.7.4 Unemployment Situations 89

9.8 Consumption of Goods 90

9.8.1 The Actual Level of Consumption (ALOC) 91

9.8.2 The Expected Level of Consumption (ELOC) 92

9.8.3 The Expectations Factor 92

9.8.4 The Required Level of Consumption (RLOC) 93

9.8.5 The Lorenz Curve 94

9.8.6 The Consumption Curve 95

9.8.7 The Poverty Fraction 96

9.8.8 The Poverty Factor 97

9.8.9 Consumption Situations 98

10. Economics 99

10.1 Sectors 99

10.2 The Social Accounting Matrix (SAM) Tableau 100

10.2.1 SAM Matrix Inputs 102

10.2.2 Other Inputs 104

10.2.3 Balancing Flows Matrix (T-matrix) 105

10.2.5 SAM Outputs 106

10.3 Computing the Actors Sector 108

10.3.1 Computing Actor’s Expenditures 109

10.3.2 Computing Actor’s Income 110

10.4 Text Notation 112

10.5 Shape vs. Size 112

10.6 Production Functions 113

10.7 Calibrating the CGE 113

10.7.1 Set the Base Consumption 114

10.8 Scenario Inputs 115

10.9 Run-time Inputs 116

10.10 Outputs 118

10.11 Ways to Affect the Economy 118

10.12 Ways the Economy Affects Athena 119

11. GOODS production infrastructure 121

11.1 Initial Laydown of GOODS Production Infrastructure 121

11.1.1 Allocated to Neighborhoods 121

11.1.2 Allocated to Actors 122

11.2 Degradation and Maintenance of Infrastructure 123

11.3 Building New Infrastructure 124

11.4 Damaging Existing Infrastructure 125

12. Communications Infrastructure 126

12.1 Communications Asset Packages 126

12.2 CAP Capacity 126

12.3 CAP Neighborhood Coverage 126

12.4 CAP Group Penetration 127

12.5 CAP Audience 127

12.6 Broadcast Cost 127

12.7 CAP Ownership and Access Control 128

13. Information Operations 129

13.1 Semantic Hooks, Congruence, and Resonance 129

13.2 Payloads 130

13.3 Broadcasting an IOM 131

13.3.1 Access to the Broadcast CAP 132

13.3.2 IOM Acceptability 132

13.3.3 Payload Magnitudes 133

13.3.4 Summary 134

13.4 IOM Example 134

Appendices 137

14. Acronyms 138

# Models

## Introduction

This document presents the models and related constructs implemented in version 6.3 of the Athena Regional Stability Simulation. The models are described in sufficient detail to allow implementation; neither the derivations nor the implementation itself are in the scope of this document. Readers are advised to read this section and Section 2 (Athena Concepts) first, and then the other sections as desired. In addition, Part I of the *Athena User’s Guide* should be read before this document, as topics which are fully and adequately discussed there are not repeated here.

### Overview

The Athena simulation is a decision support tool designed to allow an analyst to consider the intended and unintended consequences of various courses of action that might be taken during stability operations in some region of interest. Athena is a descendant of the Joint Non-kinetic Effects Model (JNEM), but includes many new models. In addition, where JNEM is a federated simulation intended for use in training, Athena is a stand-alone single-user application intended for use in analysis.

The intent of Athena’s models is to capture and make explicit a wide variety of first-order causal links, each of which makes sense on the face of it, and then present the second-, third-, and nth- order consequences of events while preserving the causal chain.

### Other Documents

Documentation on using Athena may be found in the on-line help; invoke the Athena Simulation, and select Help Contents from the Help menu. Additional documentation may be found in the "docs" subdirectory of the Athena installation directory tree; open "docs/index.html" in a web browser, and follow the links.[[1]](#footnote-1) The documentation is included with the installed software. Documentation can also be obtained directly from the Athena project; contact [William.H.Duquette@jpl.nasa.gov](mailto:Willam.H.Duquette@jpl.nasa.gov).

Documentation in the "docs" directory include:

*Athena User’s Guide*

This document combines a high-level introduction to the Athena models and philosophy with information about using the Athena application and hints on how to assemble Athena scenarios.

*Athena Rules*

This document describes the events and situations (drivers) that affect group attitudes, and the Driver Assessment Model (DAM) rule sets that assess attitude change.

*Mars Analyst's Guide*

Athena is built upon a software infrastructure layer called Mars. Models implemented in Mars, including the Unified Regional Attitude Model (URAM), are documented in the *Mars Analyst's Guide* (MAG), which may be found in the Athena documentation tree.

### Changes for Athena 6

This section describes significant changes to the models described in this document. For a complete overview of the changes in Athena 6, see the *Athena User's Guide*.

#### Changes for Athena 6.1

Athena 6.1 includes the following model changes:

* Goods Production Infrastructure
  + The **goods** sector now has production infrastructure modeled explicitly, allowing actors to own, build, and maintain production "plants".
  + Unemployment is disaggregated to neighborhoods according to the distribution of plants.
* Economics Model
  + The capacity of the **goods** sector is computed from the amount of goods production infrastructure in neighborhoods.
  + Geographic Unemployment (GU), which is the number of workers unwilling to travel to available jobs in neighborhoods they deem “too far away”, is computed by the demographics model and further constrains the capacity of the workforce in the **pop** sector.
  + The views of the Economy are now referred to as “Calibrated Values from Base SAM”, “Constrained by Total Labor”, “Constrained by Labor and Goods Capacities” and “Constrained with Geo. Unemp. and Sec. Factors”.

#### Changes for Athena 6.2

Athena 6.2 includes the following model changes:

* "Environmental" Situations are now called "Abstract Situations".
* The bulk of the Athena Attrition Model has been removed; only magic attrition remains. The model as it existed covered only a small part of the force-on-force spectrum, and did so based on assumptions that were shown to be incorrect. In the future we hope to insert a broader model of attrition.

#### Changes for Athena 6.3

Athena 6.3.0 includes the following model changes:

* A new Athena Attrition Model has been added, replacing the one that was removed in Athena 6.2. The new model is a highly-aggregated Lanchester model covering a much wider range of conflicts. The Athena CIVCAS rule set has also been updated to cover a wider range of attitude effects (e.g., vertical relationships with actors, horizontal relationships with force groups).
* Neighborhood volatility (a component of security) is intended to range from 0 to 100, but the equation in use allowed it to exceed 100 in certain cases. The equation has been modified to prevent this.
* We now compute for each neighborhood the aggregate security of all civilians in the neighborhood.
* Three abstract infrastructure services (AIS) are introduced along with a SERVICE tactic that actors can use to set the actual level of those services. AIS are called “abstract” because there is no infrastructure explicitly modeled (ie. the power grid). The three services are WATER, ENERGY and TRANSPORT.

Athena 6.3.1 includes the following model changes:

* Output and Causality Analysis.
  + Athena can now detect and display the significant differences between the beginning and end of a run or between the end states of two related runs.
  + Athena can now trace the causes of output differences back through the model and present the resulting “causality chain” to the user.

## Athena Concepts

Athena’s models describe, within a region of interest, the significant political, social and economic actors, the actions they take, the effects of those actions on the civilian population of the region, the corresponding changes in political support for the actors, and the resulting effects on the stability and control of the region. Within this feedback loop Athena tracks actor assets, military force levels and activities, civilian attitudes, regional demographics, and the regional economy. Information flows between actors and civilians are modeled implicitly.

This section gives an overview of Athena and its parts, and of the basic concepts that Athena uses. The discussion is kept to a high level; see Sections 3 and following for the detailed models.

### Model Parameters

Many of the Athena models contain numeric parameters that are used to calibrate and tune the results of the models. Athena stores these values as *model parameters* in the *model parameter database*. Although not part of an Athena scenario proper, the model parameters are accessible to the Athena user, who can view them and set them on a scenario-by-scenario basis. When such a parameter appears in a model description, below, a footnote will give its name in the model parameter database. See the *Athena User’s Guide* for more information.

### Simulated Time

Athena uses the following measures of simulated time.

Athena's clock measures time in integer *weeks* since time 0. The week is the smallest time interval with which Athena is concerned; simulation time always advances week-by-week. Weeks are also sometimes referred to as time *ticks*.

Time 0 is mapped to a calendar date by a *start date* set by the user. Athena then outputs simulated time as either some number of integer weeks or as a Julian week string based on the start date (e.g., "2012W02" is the second week of the year 2012.)

Athena is a time-step simulation; many computations take place at each time tick. In past versions of Athena, some models were triggered every so many ticks; these trigger points were called *tocks*. In this version of Athena, all models are triggered every tick.

### Geography

Each Athena scenario is associated with a geographical region called the Athena *playbox*, which is divided into polygonal areas called *neighborhoods*. (Indeed, the playbox is simply the collection of neighborhoods.)

#### Neighborhoods

Neighborhoods are simply a way of dividing the playbox into a number of reasonably homogeneous areas, and may be of any size: country, province, city, town, zip code, or neighborhood proper. Almost everything that happens in Athena happens in the context of a particular neighborhood.

Geographically, neighborhoods are defined as polygons on a map, using some appropriate coordinate system.[[2]](#footnote-2) Neighborhoods may stack, e.g., a city may be a neighborhood within a larger province, and the city may contain several neighborhoods. When stacked, two neighborhoods are said to *nest* if the first is completely contained within the second, and to simply *overlap* if one is stacked on another but is not properly nested. In the diagram below, for example, A is an urban area surrounded by suburban areas B and C; all three lie within D, a county, which abuts E, another county. B and C are nested in D, and A overlaps B and C.

A

B

C

D

E

Note that locations within an inner neighborhood are not also part of the outer neighborhood; in the diagram above, D effectively has a hole cut out of it by B and C, and A cuts sections out of both B and C. Consequently, if a neighborhood is completely tiled by nested neighborhoods, it can be omitted as it contains no locations. If D's surface were entirely covered by suburbs, for example, there would be nothing left of D and no reason to define D within Athena.

Overlapping neighborhoods should be avoided, as they cause visualization problems—when a neighborhood is drawn transparently so that a map is visible underneath, the borders of any neighborhoods it overlaps are also visible, and this can be confusing. In the diagram, then, the borders of B and C should go around A, rather than A being overlaid on top of them.

In practice, however, stacked neighborhoods of either kind are rarely used.

#### Neighborhood Proximity

In Athena, simulation events take place within neighborhoods, and affect the people in the neighborhoods. An event taking place within a neighborhood can have ripple effects in other neighborhoods; the geographic spread of these ripples depends on how nearby other neighborhoods are presumed to be, socially and psychologically rather than geographically. This psycho-social nearness of one neighborhood to another is called *neighborhood proximity*.

A

B

C

D

E

There are four proximity levels: *here, near, far,* and *remote*. In the above diagram, for example, suppose that A is the capital city. From A's point of view, A is *here*, B and C are probably *near* A, and outlying area D is *far* from A. Neighborhood E is *remote*. In other words, a person in A would be concerned by violent events in A, might worry that violent events in B or C could spread to A, might be mildly concerned with riots out in the boondocks of D, and might not think at all of events in E.

Proximity need not be symmetric. A sees D as *far*, but D might see A as *near* because of its prominence in the country as a whole.

#### Local vs. Non-Local Neighborhoods

The playbox usually consists of a single contiguous region, e.g., a country, part of a country, or a set of adjacent countries. It is possible, however, to define *non-local* neighborhoods, representing countries (or simply staging areas) in other parts of the world. For example, the United States could be represented as a neighborhood, both as a place to stash US troops and as a way to track US civilian attitudes with regard to actions elsewhere in the world.[[3]](#footnote-3)

However, such neighborhoods of convenience are outside the economy of the region of interest, also called the *local region*. Thus, neighborhoods are flagged as being *local* or *non-local*, based on whether or not they participate directly in the economy of the local region. If the region of interest is Pakistan, for example, we might include portions of India and Afghanistan in the playbox but exclude them from the local economy.

### Actors

An *actor* is an individual or group of individuals that functions as a significant decision-maker in the politics, economy, and society in the playbox. The relevant actors in a scenario depend on the scope of the scenario. If the action takes place in a single city, the mayor of the city (or those he represents) is likely to be an actor. If the playbox consists of an entire country, individual city mayors might not matter.

Actors can be of different levels of abstraction in the same scenario. If the United States is intervening in the playbox, the “US” will be an actor, even if the playbox represents a portion of a province and most of the other actors are specific individuals. And if the US leads a coalition, the coalition might be represented as a single actor or as a number of actors, depending on what is needed for the scenario.

#### Strategies

Actors use *strategies* to achieve their goals. A strategy contains *tactics*, which when executed affect the simulation in various ways, and *conditions*, which determine when tactics are eligible for execution. Tactics and conditions are grouped into *blocks*, which are executed in priority order. See the *Athena User's Guide* for a complete description of strategies.

#### Support, Influence, and Control

In general, most of the actors’ goals will ultimately involve who is in control of the neighborhoods in the playbox. Note that we are speaking of power on the ground, not of governments or elections. The real power in Pakistan, for example, is not the Pakistani government as such, but the Pakistani military. In some provinces, control lies with the tribes who live there, and the central government is largely ignored.

To this end, we say that actors receive *support* from people in neighborhoods. Actors can use this support for themselves, or lend it to other actors. Actors with enough support in a neighborhood, whether direct or indirect, are said to have *influence* in that neighborhood. When an actor gains sufficient influence, he is said to be in *control* of the neighborhood.

These concepts will all be made precise in Section 3.

#### GOODS Production Infrastructure

Actors can own goods production infrastructure in the form of “plants”. Plants are the unit of production capacity of the **goods** sector in the economic model. Actors that are not automatically maintaining infrastructure must allocate resources to have them maintained. Plants that are not maintained can negatively impact the capacity of the goods sector, which may lead to unemployment and an inadequate amount of goods for the population. Actors can also build new goods production infrastructure to expand the capacity of the goods sector to meet demands of a growing population. Section 11 contains the details of the goods production infrastructure in Athena.

#### Stability

Given that we know who is in control in a neighborhood, we may also wish to know whether the neighborhood is *stable*. When the United States pulls out of a country where it has intervened, we would like to ensure that the regime we leave behind is both friendly and stable.

As yet, we do not have a model[[4]](#footnote-4) for computing stability; this is a topic for future work.

### Groups

The people in the playbox are divided into *groups*, of which there are three kinds: civilian groups, force groups, and organization groups.

#### Civilian Groups

Civilian groups represent the population of the playbox, i.e., the people who actually live in the neighborhoods. This population may be broken into groups by ethnicity, religion, language, social class, political affiliation, or any other demographic criteria the analyst deems necessary. Civilian groups are similar to the “market segments” used to target advertising: a group is a collection of people who may be assumed to have similar biases, interests, and behaviors due to their demographic similarity.

Each civilian group resides in some neighborhood; consequently, civilian groups are sometimes referred to as *neighborhood groups*.

Each civilian group’s population is represented in the Demographics model (Section 9), broken down in various ways: the total population, the number of consumers, the number of workers, the subsistence agriculture population, and so forth. In addition, all civilians are also represented in the Ground model in the form of civilian units. People present in units are usually referred to as *personnel*, rather than as *population.* There is no real difference between the two, however; units are simply an aid to visualization of where people are and what they are doing.

Athena models civilian groups in detail, tracking the attitudes of each group as the group’s members are affected by a variety of events and situations.

Since Athena 5, civilian groups are allowed to be *empty*, that is, they can have a population of zero. Empty groups serve as placeholders for population movements; for example, suppose the scenario involves a large fraction of group A fleeing to neighborhood N. The analyst can include an empty group B in neighborhood N; then, the FLOW tactic can be used to move population from A to B.

#### Force Groups

Force groups represent military forces, such as the U.S. Army, and other groups whose purpose is to apply force in support of policy. There are five types of force group; the force group type affects the degree to which a force group’s units are able to project force vs. other force groups. The types are as follows:

* Regular military, e.g., the U.S. Army
* Paramilitary, e.g., SWAT teams and other combat-trained police units
* Police, e.g., normal civilian police
* Irregular military, e.g., militias
* Criminal, e.g., organized crime

##### Mobilization, Deployment, and Assignment

Every force group is owned by an actor; when that actor executes his strategy each week, he may mobilize, demobilize, deploy, and assign the forces under his command.

Force group personnel moved into the playbox are said to be *mobilized*; when removed from the playbox, they are said to be *demobilized.* Once in the playbox, all personnel must either be *deployed* to a particular neighborhood or demobilized. Once in a neighborhood, personnel may be *assigned* to do particular activities during the course of the week.

#### Organization Groups

Organization groups represent organizations that are present in the playbox to help the civilians. There are three kinds: Non-Governmental Organizations (NGOs), International or Inter-Governmental Organizations (IGOs), and Contractors (CTRs). NGOs are groups like the Red Cross or Doctors Without Borders who do humanitarian relief, development, and so forth. IGOs are international organizations like UNESCO. Contractors are commercial firms who are doing development work in the playbox at the behest of some actor.

Like force groups, every organization group must belong to some actor; this may be a real actor, or a fictive actor that exists only to own and direct one or more organization groups. Organization personnel are mobilized, deployed, and assigned using the same mechanisms as force groups, but can only be assigned a limited set of activities.

#### Force, Security, and Volatility

Civilian and force groups, and to a much lesser extent organization groups, have the ability and willingness to project and use force. Athena analyzes the balance of forces in each neighborhood, taking into account the populations and personnel present in the neighborhood, the types of each, and the ability of each group and unit type to project force. As the result of this analysis Athena computes the *security* of each group in each neighborhood. Closely related to security is *volatility*, a measure of the likelihood of random violence within the neighborhood.

A force or organization group’s security in a neighborhood will determine which activities it can perform, if any. Further, a group's security determines the extent to which it can support actors politically.

### Modeling Areas

Athena's models are loosely grouped into a handful of modeling areas based on the PMESII-PT paradigm.[[5]](#footnote-5) The models themselves will be described in detail in the body of this document; this section gives a brief description of each area and the models within it.

Several of the modeling areas include attitudes which are tracked by the Unified Regional Attitude Model (URAM), which is described in the *Mars Analyst's Guide*.

#### Physical

The most basic area is the Physical area. It includes the neighborhoods and units, as described above, and also the following specific models:

* Force, security, and volatility
* Group activity analysis, including activity coverage and the resulting Activity Situations (actsits).
* Abstract Situations (absits)
* Abstract Events (abevents)
* Services
* Consumption of Goods

#### Time

There are no specific models in the Time area; but as Athena is a time-step simulation, many models take time explicitly into account.

#### Political

The Politics area covers actors and their strategies; vertical relationships of groups with the actors; and the computation of actor support, influence, and control.

#### Military

The Military area covers force groups and their activities, as well as the effects of magic attrition.

#### Economics

The Economics area tracks employment and the production of goods and services in the local region. The economy changes in response to changes in neighborhood demographics and production capacity, as well as actor expenditures. It also drives actor incomes.

#### Social

The Social area includes the civilian groups, their beliefs (and the stated beliefs of actors), their satisfaction levels, their relationships with other groups, and their demographics. Organization groups also appear here.

Both horizontal and vertical relationships are based on group-to-group and group-to-actor affinities computed from the groups' and actors' belief systems by the Mars Affinity Model (MAM), which is described in the *Mars Analyst's Guide.*

The Driver Assessment Model (DAM) is responsible for assessing the effects of each driver of attitude change, and giving related inputs to URAM. DAM primarily consists of a large collection of rule sets; each rule set is devoted to one particular kind of driver, e.g., civilian casualties or presence of a force group. The DAM rule sets are described in the *Athena Rules* document. A wide variety of drivers already exist: activity situations, abstract situations, demographic situations, and civilian and organization casualties; in addition, the user may define *Complex User-defined Role-based Situations and Events* (CURSEs), effectively writing their own rule sets on the fly.

Attitudes for empty civilian groups are fixed at their natural levels.

The Demographics model tracks the civilian population of the playbox by group and neighborhood, and breaks down each group's population in a variety of ways. Populations change as civilian lives are lost due to collateral damage and direct attrition, and as civilians flow to other groups in other neighborhoods. It provides population statistics, e.g., the number of consumers and the size of the labor force, to the Economics model, and creates demographic situations (demsits) in response to the rest of Athena.

#### Infrastructure

The Infrastructure area includes all physical infrastructure in the playbox. At present, that consists of Communications Asset Packages (CAPs), used by the Information area, and **goods** production infrastructure.

#### Information

The Information area covers information flow to and among the civilian population, including propaganda and other information operations. Information flow is modeled in three ways:

* In the spread of indirect satisfaction and cooperation effects in URAM, reflecting implicit communication among the civilians.
* In the use of cooperation in the Athena Attrition Model, reflecting implicit communication (or the lack thereof) between civilian groups and force groups.
* Explicitly, when actors conduct information operations campaigns to change the attitudes of the civilian population.

### Simulation States and the Advancement of Time

The Athena simulation has a number of states. When a new scenario is created, Athena is in the Scenario Preparation or **PREP** state. In this state, the analyst creates neighborhoods, groups, and so forth, and sets up their initial attributes.

When the analyst is satisfied with the scenario, the scenario is *locked*. At this time, Athena initializes the simulation resulting in a complete view of the simulated world at time 0. In particular, Athena:

* Initializes the various models
* Executes all tactics whose "on lock" flag is true, to establish the initial state of affairs
* Assesses attitude drivers and gives attitude inputs to URAM, to get the appropriate set of transient attitude effects at time 0.
* Saves the historical data for time 0.
* Enters the **PAUSED** state.

In the **PAUSED** state the analyst may examine the state of the simulation and make changes if desired (though it is usually wise to return to the **PREP** state first). Then, the analyst may ask the model to run time forward by some number of weeks. The simulation state changes to **RUNNING**, and time advances week by week.

During each week, Athena performs the following steps:

* Increments the simulation time by one tick (i.e., one week)
* Executes actor's strategies
* Determines the new state of affairs "on the ground".
* Assesses attitude drivers and gives attitude inputs to URAM.
* Advances URAM, thus applying all of the new attitude inputs.
* Saves a variety of historical data for later access (e.g., plotting)
* Returns to the **PAUSED** state if the stop time has been reached, allowing the analyst to examine the current state and make course corrections.

When Athena is **PAUSED** at time *t*, all simulation events for time *t* have already occurred and all history has been saved. Any steps the analyst might take will have their effect at *t*+1 or later.

## Relationships and Control

Athena models the relationships between groups and between groups and actors. This section describes how relationships are defined and how they vary over time. In addition, it describes the closely related issue of actor support and influence in neighborhoods, and how influence leads to neighborhood control. More particularly, this section addresses the following topics:

* Inter-group relationships, also called “horizontal” relationships.
* Group/actor relationships, also called “vertical” relationships.
* How relationships vary with time.
* Support for actors by groups in neighborhoods.
* Support for actors by other actors.
* The influence of actors in neighborhoods due to the support they receive.
* The circumstances under which control of a neighborhood shifts from one actor to another (or to or from no actor at all).
* What happens when control of a neighborhood shifts.

### Relationships and Affinity

Athena defines two kinds of relationship, horizontal relationships and vertical relationships. In each case, a relationship is expressed as a number *X* where . The magnitude of the number indicates the strength of the relationship and the sign indicates whether the related parties are friends or foes. We usually assume that an entity’s relationship with itself is 1.0.

All relationships are tracked by URAM and are subject to change due to attitude drivers, as described in the *Athena Rules Document*. The baseline and natural levels of each relationship default to the affinity *A* computed by comparing the belief systems of the entities involved, where , as described in the *Mars Analyst's Guide*, though this default may be overridden by the analyst.

Athena uses *quality* data types to relate numeric values to symbolic constants. The qaffinity data type is used with both affinity and relationship values; it is defined as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Narrative | Symbol | Value | Range |
| *a* supports *b* | SUPPORT | 0.8 | +0.7 < *value* ≤ +1.0 |
| *a* likes *b* | LIKE | 0.4 | +0.2 < *value* ≤ +0.7 |
| *a* is indifferent to *b* | INDIFF | 0.0 | –0.2 < *value* ≤ +0.2 |
| *a* dislikes *b* | DISLIKE | –0.4 | –0.7 < *value* ≤ –0.2 |
| *a* opposes *b* | OPPOSE | –0.8 | –1.0 ≤ *value* ≤ –0.7 |

### Horizontal Relationships

The horizontal relationship between group *f* and group *g* is denoted . This is the familiar group relationship from earlier versions of Athena and from JNEM. It is a static, unchanging value that reflects the amity or enmity between the pair of groups. It is not symmetric; it will often be the case that . Athena establishes the natural and baseline levels of horizontal relationships on group affinity as described in this section. Horizontal relationships are tracked in URAM, and may vary dynamically.

#### Force and Organization Group Affinities

Force and organization groups have neither belief systems nor affinities. They are, however, owned by actors, which do. For the purposes of computing initial relationships, we assume that force and organization groups inherit the belief systems and affinities of their owning actors.

To put it another way, each group gets its affinities from its associated *belief system entity*, where:

* The belief system entity for each force and organization group is the actor that owns the group.
* The belief system entity for each civilian group is the group itself.

Thus, when we speak of the affinity of group *f* for group *g*, we are really speaking of the affinity of group *f’*s belief system entity with group *g’*s belief system entity.

#### Computing Horizontal Relationships

The initial baseline horizontal relationship between two groups is then defined as follows:

Basing relationships on affinity provides two ways to allow relationships to vary dynamically: if a group’s belief about a topic changes (perhaps due to information operations), its affinities will change; and relationships can change due to URAM inputs.

The natural level of the relationship is the same as the initial baseline.

### Vertical Relationships

The vertical relationship is a number that determines the extent of group *g*’s opposition to or support for actor *a*. As with , ; unlike , the vertical relationship is unidirectional. We are measuring *g’*s opposition to or support for *a*; *a’*s opposition to or support for *g* will be revealed by *a’*s choice of tactics. Like horizontal relationships, vertical relationships are tracked in URAM and may vary dynamically over the course of the simulation.

The foundation for is the affinity of *g* for *a* based on their belief systems, where force and organization groups inherit their belief systems and affinities from the owning actors, as stated above. is usually static when *g* is a force or organization group[[6]](#footnote-6); for civilian groups, is dynamic, varying over time according to *a*’s actions and *g*’s circumstances. Groups expect things from actors, and the relationship will depend on how well the actors meet those expectations.

See the *Athena Rules* document for attitude drivers that affect vertical relationships (notably the CONTROL, ENI, and MOOD rule sets).

#### Baseline Vertical Relationships

The initial baseline vertical relationship for civilian group *g* with actor *a* is simply the affinity, unless this is overridden by the analyst:

Force and organization groups are owned by actors, and share the relationships of their actors. When *g* is a force or organization group owned by actor *b*, we define the initial baseline vertical relationship, as follows:

That is, a force or organization group has a perfect relationship with its owner, and its vertical relationship with any other actor is simply the affinity of its owner for that actor.

The natural level of the relationship is the same as the initial baseline.

### Actor Support and Influence

An actor receives *support* in a neighborhood from the groups in the neighborhood, and also possibly from other actors. Given enough support, the actor can gain *influence* in the neighborhood, and perhaps even control the neighborhood. This section defines the terms “support” and “influence” as they are used in Athena.

#### Direct vs. Derived Support

An actor’s political influence in a neighborhood derives from the support he receives from the groups in the neighborhood. An actor may make use of this support himself, or he may choose to give his support to another actor. We will call the support an actor receives directly from groups his *direct support*, and support an actor receives from other actors his *derived support*.

#### Direct Support

Actor *a’*s direct support in *n* is determined by:

* The number of people in *n* who favor actor *a* (including personnel from all three kinds of group)
* The strength of their favor, as determined by the vertical relationship between them and actor *a*
* Their ability to move and work within the neighborhood, as determined by their security (see Section 4)

Thus,

where

= The fraction of *n’*s population that is willing and able to support *a*.

= The number of people belonging to *g* in *n*. Group *g* can be a civilian, force, or organization group.

= The total number of people of all groups in *n*.

= The vertical relationship between group *g* and actor *a*.

= The minimum relationship needed to qualify as “support”, nominally 0.2.[[7]](#footnote-7)

= Group *g*’s security factor in neighborhood *n*, as computed from its security using a Z-curve.

The security factor is a number that indicates the fraction of a group that is able to actively support actors given the group’s security level. The security of group *g* in neighborhood *n* is an integer in the range ; the number is interpreted according to the following scale:

|  |  |
| --- | --- |
| **Bin** | **Range** |
| High | 25 < *security* ≤ 100 |
| Medium | 5 < *security ≤* 25 |
| Low | -25 < *security* ≤ 5 |
| None | -100 < *security* ≤ -25 |

We compute the security factor from the security level as follows:

where is a Z-curve:[[8]](#footnote-8)

This Z-curve is based on the assumption that groups with a security of “None” (-25 or less) can’t support anyone, and once a group has a security of “High” (25 or more) it is supporting as much as it can. Increasing security in between increases the group’s support per person directly, as shown. The parameters are defined in the model parameter database, and so can be changed to reflect different assumptions.

#### Derived Support

Actor *a* may do one of three things with his direct support in neighborhood *n*:

* He may use it himself, to try to take control in the neighborhood. In this case, we say that *a* supports himself in *n*.
* He may give it to another actor *b*. In this case, we say that *a* supports *b* in *n*, and that *b* receives derived support from *a*.
* He may choose not to use it at all, that is, he may be *apolitical*. For example, organization groups in Athena must be owned by an actor, but that actor will not usually involve himself in neighborhood politics. He might receive support from the groups in a neighborhood, but will not use that support to take control in the neighborhood. In this case, we say that *a* supports no one in *n*.

Note that derived support is not transitive. If *a* supports *b*, and *b* supports *c*, that does *not* mean that *a* supports *c*. Rather, it means that *b* gets *a’*s direct support, and *c* gets *b*’s direct support.

An actor chooses which actor to support in two ways. First, each actor has an actor that he “usually supports”; this may be **SELF**, **NONE**, or the name of a specific actor. This value is set as part of the definition of the actor during scenario preparation; it can be changed by the analyst as the simulation runs. In the absence of other input, the actor will support the given actor (or none) in all neighborhoods. Thus, for example, an actor that owns humanitarian organization groups like the Red Cross would support **NONE**in all neighborhoods, and would thus not be included in the political process.

Next, an actor can execute the **SUPPORT** tactic[[9]](#footnote-9) to explicitly support another actor. The **SUPPORT** tactic has the following inputs:

* The actor to support
* A list of neighborhoods in which to support him.

The effect of the **SUPPORT** tactic lasts until the next strategy execution, when it can be repeated.

Thus, the political relationships between the actors depend entirely on the actor’s strategies. This is a general rule—**an actor is what an actor does**, and hence **relationships between actors are always mediated by the actors’ strategies**.

#### Total Support

Actor *a*’s total support in *n*, also called his *effective support* in *n*, is computed as follows:

|  |  |
| --- | --- |
|  |  |

Note that while can be interpreted as the fraction of the population of *n* that supports *a*, can not. It is simply a score used to compute influence.

Some observations:

* If every actor supports himself, then for all a.
* If no actor supports actor *a*, including himself, then , even if .
* If an actor receives derived support, then his can be greater than 0, even if he gives his direct support away or chooses not to use it.
* If an actor gives his direct support away and receives no derived support, his will be 0.

The third bullet is particularly interesting: an actor can be “in control” of a neighborhood because of support from some other powerful actor, whether he wants to be or not.

#### Influence

Actors have influence in a neighborhood in proportion to their support relative to other actors. In particular, if only actor *A* has support in neighborhood *n*, then only *A* will have influence in *n*. We require an actor to have a minimum of *SupportThreshold* support[[10]](#footnote-10) to have any influence; this prevents an actor with a trivial (though positive) degree of support from taking control of a neighborhood in the absence of other candidates.

Let be the set of actors *a* for which . Actor *a*’s influence in neighborhood *n* is then

Note that and that unless is the empty set, in which case all influence values in *n* are 0.0. Influence is thus a zero-sum game; an actor can increase his influence only at the expense of some other actor.

### Control of a Neighborhood

When the residents of neighborhood *n* see actor *a* as being responsible for causing or fixing the current state of affairs in *n*, then *a* is said to be in control of neighborhood *n*. For *a* to be in control in *n* might involve an official governmental position, or it might simply be a *de facto* status, as when a military commander or warlord takes a neighborhood by force. As shown in Section 3.3, the civilians in *n* will hold *a* responsible for increases or decreases in their mood, and also for the provision of ENI services.

The actor in control of *n* will usually be the actor that has sufficient influence[[11]](#footnote-11) to dominate *n*; however, this is not always the case. Suppose actor *A* has a dominating degree of influence, and is in control of *n*. *A’*s rival, actor *B*, moves troops into *n*, causing a crisis. *B’*s action decreases security levels in *n* for everyone but *B’*s own troops, thus eroding *A’*s influence significantly. Although *A* may no longer be able to dominate the neighborhood, the civilians will continue to hold *A* responsible until he either becomes irrelevant (because he is dominated by *B*) or *B* is able to dominate the neighborhood altogether. If *A* manages to drive *B’*s forces out of *n*, he can maintain control throughout the crisis, even though his actual degree of influence may have been very low at times.

Thus, we define the following rules. Suppose, first of all, that actor *a* is in control of neighborhood *n* before we re-assess each actor’s influence in *n*. Who is in control after the re-assessment is determined by applying these rules:

1. If , then *a* remains in control.[[12]](#footnote-12)
2. If but , then *a* no longer dominates the neighborhood, but remains in control.
3. If for some actor *b*:
   1. If , then actor *b* now dominates the neighborhood and is in control. **Control has shifted.**
   2. Otherwise, actor *a* has clearly lost control, being dominated by *b*, but no other actor dominates the neighborhood. No actor is in control of neighborhood *n*. **Control has shifted.**
4. If no actor is in control of neighborhood *n*, and some actor *b* gains influence such that , then naturally *b* is now in control. **Control has shifted.**

In the future we might want to take time into account: if actor *a* is in control but no longer dominates the neighborhood (i.e., *a* has influence less than the *ControlThreshold*), then after some period time he is no longer deemed to be in control even if no other actor has gained control. Similarly, actor *b* might need to dominate the neighborhood for some period of time to gain control.

#### When Control Shifts

When control shifts:

* Support for actors is now relative to the new political situation, which leads to new values for , , and .
* The change in control is a monitored event that has an effect on the attitudes of every civilian group resident in the neighborhood.

The attitude effects are assessed by the CONTROL rule set, which is documented in the *Athena Rules* document.

## Force Analysis

### Overview

Athena analyzes the forces present in each neighborhood along with the neighborhood demographics to determine the following quantities for each group and neighborhood:

**Security**: A group's level of security in a neighborhood, measured as a score from -100 (very dangerous) to +100 (very safe). A group’s security determines the kinds of things the group can safely do within the neighborhood.

**Volatility:** The volatility of a neighborhood is the likelihood of spontaneous violence within the neighborhood, measured as a score from 0 (least volatile) to 100 (most volatile).

**Force**: A group’s force in a neighborhood is its physical ability to use its assets to control that neighborhood through force. The major component of force is military strength.

In general:

* The primary component of group *g*'s force in neighborhood *n* is group *g*'s assets in the neighborhood. In Athena, a group's assets are its people.
* Friendly military personnel in neighborhood *n*, i.e., those with a positive relationship with *g*, should increase *g*'s force in proportion to their numbers.
* Friendly civilians in *n* should also increase *g*'s force, but by a smaller amount per person than soldiers or militia. However, an aggressive demeanor and a negative mood () can increase a civilian group's force multiplier.
* Friendly contractor personnel in *n* may also increase *g*'s force, but personnel from friendly IGOs and NGOs are expected to remain neutral in case of conflict and do not contribute to the force of other groups.
* Friendly personnel in nearby neighborhoods should also increase *g*'s force, but to a smaller extent than personnel in neighborhood *n*.
* Also of interest is the force aligned against group *g*, from the groups that oppose group *g*. And just as *g* can call in its friends from nearby neighborhoods, so can *g*'s enemies.
* Force groups can be ordered to act as though they had a different relationship than they really do. The extent to which they are able to act as ordered depends on their training and discipline. The resulting *effective relationship* is used when determining friends and enemies.
* The activities performed by force group personnel should influence the degree of force they are able to project.

Using these concepts we build up a notion of force which can be used to define neighborhood volatility, and then a group's security within the neighborhood.

### Force Group Stance and Effective Relationships

Historically, the focus of the security model was on a force group providing for its own security in order to conduct its various activities. However, a force group can also be tasked to provide security in a neighborhood for the sake of the residents. The previous version of the security model handled this scenario poorly, because it did not take into account the actor’s intent in sending the force group to the neighborhood.

Instead, the security model determined the force group’s intent purely from its relationship with the resident groups. If these relationships were largely positive, then the force group was presumed to be well-disposed toward the civilians, and security would generally increase. If these relationships were largely negative, then the force group was presumed to be hostile towards the civilians, and security would generally decrease. As a result, a force group bent on conquest and subjugation might increase civilian security, and one tasked to do peace-keeping might make security worse.

#### Stance

Athena allows the actor to express his intent via the **STANCE** tactic, which sets , force group *f’*s stance toward group *g*. The variable ranges from -1.0 to +1.0, and can be thought of as a *designated relationship*—essentially, group *f* has been ordered to act as though its relationship with group *g* is rather than , *f’*s true horizontal relationship with *g*.

Thus, the actor can set to a positive number for peace-keeping, and to a negative number for more aggressive activities.

Group *g* can be any group in the playbox (other than *f*, of course). Thus, two antagonistic force groups who are ordered to work together can be given positive stances.

#### Group Discipline

Suppose force group *f* is told to adopt a stance of 1.0 toward group *g*, but *f’*s horizontal relationship with *g*, , is −0.5. It’s one thing to tell *f* to love *g*, but it’s another thing for them to do it, and in such a case we’d expect a certain amount of friction between *f* and *g* due to members of *f* who really don’t want to make nice with *g*. Consequently, we introduce the notion of *discipline*, a force group’s ability to obey orders. Further, we assume that discipline increases with training.

To that end, we define:

= The training level of force group *f*, an enumerated value.

= The discipline of force group *f*, a numeric value from 0.0 to 1.0.

The following table gives the default mapping from training levels to discipline values.[[13]](#footnote-13)

|  |  |
| --- | --- |
| Training Level | Discipline |
| Proficient | 1.0 |
| Fully Trained | 0.9 |
| Partially Trained | 0.7 |
| Not Trained | 0.4 |

At some future time, training level will likely be redefined as a continuous variable managed by URAM. This will allow training levels to change as the simulation runs, e.g., training activities can increase training levels; prolonged service without training can decrease them.

#### Effective Relationships

Group *f’s* effective relationship with group *g* in neighborhood *n* is then a function of his horizontal relationship with *g*, , his stance toward *g* in *n*, and his degree of discipline. Therefore, we define:

If force group *f* has no specified stance, then will simply be . If *f*’s discipline is perfect, then will be exactly . Otherwise, it will be something in between the horizontal relationship and the stance. For non-force-groups, of course, the effective relationship is simply the horizontal relationship.

We will use when determining a group's friends and enemies in a neighborhood.

### Background Criminal Activity

In Athena 3 and prior, we computed security based on the assumption that almost all violence was due to friction between groups with negative relationships, or in other words, that there was no intra-group violence. However, Rob Crowson of TRISA has pointed out that law enforcement plays a significant role in security through the suppression of criminal activity. Therefore, since Athena 4 the overall security model includes a model of background criminal activity in the civilian population, along with suppression of criminal activity by law enforcement activities.

Organized crime is represented in Athena by means of criminal force groups owned by actors; hence the kind of crime we discuss here is crime among the civilian population, excluding organized crime.

In a nutshell, we assume that some fraction of every civilian group will engage in criminal activities, and that some part of these criminal activities can be suppressed by an appropriate degree of law enforcement activities. Moreover, criminals cause friction not only with their own group, but with every group in the neighborhood.

#### Law Enforcement Personnel

First we determine the effective number of law enforcement personnel in neighborhood *n*, denoted . In computing we consider the following:

* The force group personnel present in *n*
* The activities to which they are assigned
* Each force group’s suitability for law enforcement activities given its force type
* Each force group’s efficiency at performing law enforcement activities given its degree of training.

First, we assume that some force group types are better suited for law enforcement activities than others. Suitability as a function of group type is shown in the following table:[[14]](#footnote-14)

|  |  |
| --- | --- |
| Group Type | Suitability |
| REGULAR | 0.8 |
| PARAMILITARY | 0.6 |
| POLICE | 1.0 |
| IRREGULAR | 0.3 |
| CRIMINAL | 0.6 |

Each member of an actual police group counts fully; members of other group types count less.

Next, we assume that a force group’s efficiency at enforcing the law depends on the group’s training level, as defined in Section 4.2.3.[[15]](#footnote-15) Efficiency as a function of group type is shown in the following table:[[16]](#footnote-16)

|  |  |
| --- | --- |
| Training Level | Efficiency |
| Proficient | 1.2 |
| Fully Trained | 0.9 |
| Partially Trained | 0.7 |
| Not Trained | 0.4 |

Next, some force group activities are more effective at law enforcement than others; some do not contribute to law enforcement at all. This is captured as the parameter in the following table:[[17]](#footnote-17)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Activity *a* | Description |  | Activity *a* | Description |  |
| CHKPOINT | Checkpoint/Control Point | 0.5 | COERCION | Coercion | 0.3 |
| CONSTRUCT | Construction | 0.0 | CRIME | Criminal Activities | 0.3 |
| CURFEW | Curfew | 1.2 | EDU | Schools | 0.0 |
| EMPLOY | Provide Employment | 0.0 | GUARD | Guard | 1.0 |
| INDUSTRY | Support Industry | 0.0 | INFRA | Support Infrastructure | 1.0 |
| LAWENF | Law Enforcement | 1.0 | MEDICAL | Healthcare | 1.0 |
| PATROL | Patrol | 1.0 | PSYOP | Psychological Operations | 0.3 |
| RELIEF | Humanitarian Relief | 0.3 | NONE | None | 0.0 |

We then compute the effective number of law enforcement personnel, , as follows:

where is the number of group *g*’s personnel assigned to do activity *a* in neighborhood *n*.

#### Suppression of Criminal Activity

Law enforcement does not eliminate the criminals; it simply suppresses their criminal activities. Therefore, we define as the degree of suppression of background criminal activities in neighborhood *n*, where ranges from 0.0 to 1.0. At 0.0, no background criminal activities are suppressed; at 1.0 all crime that can be suppressed is suppressed. Per doctrine, suppression of criminal activity is a function of the number of law enforcement personnel relative to the size of the population and the urbanization of the neighborhood. Consequently, we compute suppression using a family of coverage functions,[[18]](#footnote-18) as shown in the following table:[[19]](#footnote-19)

|  |  |
| --- | --- |
| Urbanization | Coverage Function |
| ISOLATED | 1/1000 |
| RURAL | 1/1000 |
| SUBURBAN | 2/1000 |
| URBAN | 3/1000 |

Thus, in an urban neighborhood an of 3.0 per thousand residents will achieve 2/3rds coverage and hence 2/3rds suppression of background criminal activity.

#### Nominal Criminal Fraction

Our initial model of background criminal activity in civilian populations is that the level of criminal activity in a group depends on the group’s demeanor and on the unemployment rate. Athena’s demographic model defines a statistic called , the unemployment per capita percentage. This is the percentage of the group's population that is currently unemployed, and so it is a measure of the degree to which unemployment affects the group at large.

Specifically, a group’s nominal criminal fraction, , is the fraction of the group that will engage in criminal activities in the absence of law enforcement. It is the output of a family of Z-curves.[[20]](#footnote-20) There is one curve for each group demeanor (AGGRESSIVE, AVERAGE, APATHETIC); the curves return given the . The default curves are shown in the following chart:



#### Actual Criminal Fraction

A civilian group *g’*s actual criminal fraction, , is the fraction of the group involved in criminal activities given the current degree of law enforcement in *g*’s neighborhood, *n*:

where

*n* = A neighborhood

*g* = A civilian group residing in *n*

= Suppression of crime by law enforcement in *n*, as defined above.

*suppfrac* = Suppressible fraction: the fraction of crime that can be suppressed by law enforcement activities,[[21]](#footnote-21) nominally 0.6.

= Nominal Criminal Fraction of group *g*.

= Actual Criminal Fraction of group *g*.

In short, the nominal criminal fraction is reduced by law enforcement suppression, but not past the suppressible fraction. The formula for can be simplified as follows:

### Measuring Force

The security of group *g* in neighborhood *n* is based on the balance of forces in the neighborhood. To compute security, then, we must first define each group's projection of force in the neighborhood.

#### A Group's Own Force

First, group *g*'s own force in neighborhood *n*, denoted , is the degree of force projected by its own personnel. Different groups can project a different amount of force per person. In general, then, a group's own force is defined as

where

= A force multiplier for group *g*.

= The number of group *g'*s personnel that are present in neighborhood *n*.

The details differ by group type.

##### Civilian Group Force

The own force of a civilian group is defined to be

where

*w* = A multiplier, [[22]](#footnote-22) nominally 0.01, indicating the fraction of the population available to participate in a violent fracas at any given time.

= The demeanor multiplier.[[23]](#footnote-23)

= The mood multiplier.[[24]](#footnote-24)

= The number of group *g'*s personnel that are present in neighborhood *n*.

It is assumed that the more aggressive *g'*s demeanor and the worse *g'*s mood, the more likely it is that civilians will use force to aid their friends and hinder their foes. Therefore,

Here, *b* is a factor that determines how strongly a group's mood contributes to its force.[[25]](#footnote-25) If *b* is 0.2, for example, the effect of mood will range from 0.8 (when group *g* is perfectly satisfied) to 1.2 (when group *g* is perfectly dissatisfied).

##### Organization Group Force

The own force of an organization group is defined to be

where

= The demeanor multiplier, as above.

= The organization type multiplier, as shown in the following table. [[26]](#footnote-26)

= The number of group *g'*s personnel that are present in neighborhood *n*.

| Organization Type | Multiplier |
| --- | --- |
| NGO | 0 |
| IGO | 0 |
| Contractor | 2 |

##### Force Group Force

The own force of a force group depends not only on its demeanor and group type but also on what it is doing. A group that is patrolling projects more force than one that is providing health care.

where

= The demeanor multiplier, as above.

= The force type multiplier, as shown in the table below.[[27]](#footnote-27)

*a* = A force group activity (see Section 5).

= An activity force multiplier, as shown in the table below.[[28]](#footnote-28)

= The number of group *g'*s personnel that are present in neighborhood *n* and performing activity *a*.

The force type multiplier is defined as follows, given group *g*'s force type.

| Force Type | Multiplier |
| --- | --- |
| REGULAR |  |
| PARAMILITARY |  |
| POLICE |  |
| IRREGULAR |  |
| CRIMINAL |  |

The activity force multiplier is defined in the following table. The multipliers for the activities (other than law enforcement) were originally much less than 0.8; e.g., healthcare was originally given a multiplier of 0.3. After discussion, we decided that a value of 0.3 would make sense at JNEM’s minute-by-minute timescales, but not at Athena’s weekly timescales. Troops assigned to do activities during a current week can be temporarily reassigned to handle security risks over the course of the week as need arises. Thus, while troops assigned to the more humanitarian activities project less force than those assigned to other activities, the loss is small.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Activity | Description |  | Activity | Description |  |
| CHKPOINT | Checkpoint/Control Point | 1.5 | COERCION | Coercion | 1.2 |
| CONSTRUCT | Construction | 0.8 | CRIME | Criminal Activities | 0.8 |
| CURFEW | Curfew | 1.2 | EDU | Schools | 0.8 |
| EMPLOY | Provide Employment | 0.8 | GUARD | Guard | 1.7 |
| INDUSTRY | Support Industry | 0.8 | INFRA | Support Infrastructure | 0.8 |
| LAWENF | Law Enforcement | 1.5 | MEDICAL | Healthcare | 0.8 |
| PATROL | Patrol | 2.0 | PSYOP | Psychological Operations | 1.0 |
| RELIEF | Humanitarian Relief | 0.8 | NONE | None | 0.0 |

##### Non-Criminal and Criminal Force

As stated in Section 4.3, some fraction of each civilian group consists of unsuppressed criminals who are effectively everyone's enemy. Consequently, for each each group we divide its own force into two components: the group's non-criminal force, , and its criminal force, . For force and organization groups, which we assume not to contain criminals, these are simply:

For civilian groups, the actual criminal fraction contains unsuppressed criminals; the remainder are (possibly reluctantly) law-abiding. Thus, we define

### Friends and Enemies

Violence in a neighborhood is caused by enemies (including criminals) and mitigated by friends. Friends and enemies are determined by their effective horizontal relationships, , where friends have a positive relationship and enemies have a negative relationship. We assume that group *f* will assist or oppose group *g* according to the strength of the effective relationship from *f'*s point of view. For convenience we define

Then,

In other words, your friends in the same neighborhood are those groups (including yourself) with whom you have a positive relationship, in proportion to the strength of the relationship, but excluding the criminals in each group. Then, your enemies in the same neighborhood are the non-criminals in those groups with whom you have a negative relationship, in proportion to the strength of the relationship, plus all criminals in every group.

Note that need not be symmetric; if we were to replace with in , then we'd be assuming that group *f* would assist group *g* based on the strength of *g*'s feeling for *f*, which might not be at all the same thing as *f'*s feeling for *g*.

Next, we take neighborhoods into account:

Here, *h* is a factor, nominally 0.3, that reduces the effect of friends and enemies in nearby neighborhoods.[[29]](#footnote-29) The phrase "*m* near *n*" denotes those neighborhoods *m* whose is *near*, that is, those neighborhoods *m* whose inhabitants regard neighborhood *n* as being nearby. As with , need not be symmetric.

In order to create scores, we'll need to normalize these values by the total force in the neighborhood:

Then,

### Volatility

The volatility of a neighborhood is the likelihood of spontaneous violence within the neighborhood. Volatility depends on the balance of forces in the neighborhood, and is a key component of security. For example, an ORG group may hesitate to go into a neighborhood with high volatility, even if the ORG group is on friendly terms with all of the parties present in the neighborhood, simply because the chance of getting caught in a cross-fire is too high.

The volatility of neighborhood *n* can be regarded as a measure of the number of individuals potentially in conflict in neighborhood *n*, using the force values defined above:

ranges from 0 to because it counts every conflict from the point of view of each of the parties involved. Scaling to the range 0 to 100 yields

where , and 0.0 otherwise.

### Security

A group's security in a neighborhood determines the kinds of activities the group can perform within that neighborhood. Different levels of security are needed for different kinds of operations. The group's level of security depends on the assets present in the neighborhood, including both those belonging to the group and those belonging to friendly and unfriendly groups, as well as on the nature of both assets and groups and the volatility of the neighborhood. In general:

* The presence of friendly forces should increase group *g*’s security in neighborhood *n*.
* The presence of enemy forces should decrease group *g*’s security in neighborhood *n*.
* Increased volatility should decrease group *g*’s security in neighborhood *n*.

The security of group *g* in neighborhood *n* is defined as follows:

where *v* is the volatility scaling factor, nominally 1.0, which can be used to reduce the effects of volatility on security.[[30]](#footnote-30) The denominator scales so that it ranges from -100 to +100.

For use in rules, will often be translated to a symbolic value, as shown in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| Range | | | Symbol |
|  |  |  | High |
|  |  |  | Medium |
|  |  |  | Low |
|  |  |  | None |

For some uses, we convert to a multiplicative factor using a Z-curve.

We also compute the average security of the civilian groups in *n*, weighted by their populations, as:

#### Security and Empty Groups

The math allows us to compute even if group *g* has no personnel present in neighborhood *n*. One can think of this as the security of one member of group *g* were he to be dropped into neighborhood *n*; but the presence of significant members of group *g* affects in a non-trivial way, and so this zero-personnel number isn't very useful.

For this reason, and also because non-zero security levels look very odd for empty civilian groups, Athena no longer displays for groups *g* with no personnel in *n*.

## Effects of Unit Activities

An actor’s strategy can deploy group personnel to neighborhoods, and then assign those personnel activities. The end result is a collection of units that contain the personnel and are tagged with the relevant activities.

### Force Presence and Activities

Force units present in a neighborhood can affect civilian attitudes in a neighborhood in two ways: by their mere presence, and by engaging in other activities.

**Force Presence:** A force group's presence in a neighborhood is measured as the total number of personnel in units that belong to the group and are present in the neighborhood.[[31]](#footnote-31)

**Force Activities:** A unit assigned an activity is assumed to carry out the activity if it can possibly do so. Each activity requires that the acting group have a minimum level of security in the neighborhood. The following table lists the force activities and the nominal minimum security level for each activity.[[32]](#footnote-32)

| Activity | Long Name | Minimum Security  Required | Coverage Function |
| --- | --- | --- | --- |
| PRESENCE | Presence | n/a | 25/1000 |
| CHKPOINT | Checkpoint/Control Point | Low | 25/1000 |
| CONSTRUCT | Construction | High | 20/1000 |
| COERCION | Coercion | Medium | 12/1000 |
| CRIME | Criminal Activities | Medium | 10/1000 |
| CURFEW | Curfew | Medium | 25/1000 |
| EDU | Schools | High | 20/1000 |
| GUARD | Guard | Low | 25/1000 |
| INDUSTRY | Support Industry | High | 20/1000 |
| INFRA | Support Infrastructure | High | 20/1000 |
| LAWENF | Law Enforcement | Medium | 25/1000 |
| MEDICAL | Healthcare | High | 20/1000 |
| PATROL | Patrol | Low | 25/1000 |
| PSYOP | Psyop | Medium | 1/50000 |
| RELIEF | Humanitarian Relief | High | 20/1000 |

Thus, a force group can enforce a curfew or provide law enforcement in a neighborhood only if the group’s security is at least Medium in that neighborhood. The extent to which a group is engaging in an activity in a neighborhood is measured by the total number of personnel in units that:

* Belong to the group
* Are present in the neighborhood
* Are assigned to do the activity

This figure is automatically 0 if the minimum security requirement is not met.

**Coverage Fractions:** Whenever a group is present or conducting an activity in a neighborhood, the extent to which its presence or activities affect the local population is termed its *coverage fraction* for the activity. The coverage fraction is a value from 0 to 1; it is computed by comparing the group’s personnel (present, or conducting an activity) with the total population of the neighborhood, using a function like that shown below.

c = 25

1.0

2/3

0.0

In this example, if the group has deployed 25 troops per 1000 people in the neighborhood then the coverage fraction is 2/3, rising exponentially toward 1.0 as the troop density goes toward infinity. The coverage fraction determines how much of the potential satisfaction change is gained or lost by the group’s presence or activity. The x-axis is usually troops per 1000 population, as shown, but can vary; for PSYOP, it is troops per 50,000 population. Thus, the two parameters required to compute a coverage fraction are {*c*, *d*} where *c* is the troop density at the 2/3rds point, and *d* is the population denominator, e.g., 1000 or 50,000. The coverage function is often specified in the form *c*/*d*. In the example shown above, then, the coverage function may be specified as either 25/1000 or {25, 1000}. The actual function is as follows. First, the troop density is

where *p* is the number of personnel engaged in the activity. Then the coverage fraction is

**Example:** Force group *g* has 750 troops in neighborhood *n*, which has a total population of 40,000 people:

*p* = 750

= 40,000

The coverage fraction parameter, *c/d*, is nominally 25/1000 for mere presence. The troop density for group *g* is therefore

The coverage fraction is therefore

A force group *g*'s coverage fraction for activity *a* in neighborhood *n* is denoted *.*

**Composite Coverage Fractions:** If two force groups were cooperating in enforcing a curfew, we would expect the curfew to suppress civilian activities based on the total personnel used in enforcing it, independent of the fact that two groups are involved. In cases like these it is useful to employ a *composite coverage fraction*. This fraction can be computed based on the total troop density across all relevant groups; it can also be computed from each group's coverage fraction for the activity, as follows:

At the present time, Athena does not generally make use of composite coverage fractions.

### Organization Activities

Units belonging to organization groups have no effect on civilian satisfaction due to their mere presence. Organization units may perform the following activities; the effect of these activities is modeled in the same way as force activities are, with these distinctions:

* Organization groups generally require high security to be effective; this can be change in the model parameter by activity and organization type.[[33]](#footnote-33) Note that this differs from JNEM, where it was assumed that CTR groups brought their own security personnel and could thus function with only medium security.
* Organization units can perform only a subset of the activities that a force unit can perform.

| Activity | Long Name | Minimum Security  Required | Coverage Function |
| --- | --- | --- | --- |
| CONSTRUCT | Construction | High | 20/1000 |
| EDU | Schools | High | 20/1000 |
| INDUSTRY | Support Industry | High | 20/1000 |
| INFRA | Support Infrastructure | High | 20/1000 |
| MEDICAL | Healthcare | High | 20/1000 |
| RELIEF | Humanitarian Relief | High | 20/1000 |

### Civilian Activities

Up through Athena 4, civilian units were allowed to perform the DISPLACED and IN\_CAMP activities. This was a stopgap, only; it allowed personnel to be moved around, but did not properly account for the attitudes of the displaced population. Since Athena 5, the FLOW tactic allows civilian personnel to move from group to group, and hence from neighborhood to neighborhood; consequently, these activities are no longer needed.

### Activity Situations

When a group has an activity with coverage greater than 0.0 in a neighborhood, we say that an *activity situation* exists in that neighborhood. The effects of these situations on local attitudes are assessed by the DAM rules; see the *Athena Rules* document. Note that no activity situation is created for the "In Camp" activity.

## Abstract Situations

Abstract situations (absits) are on-going circumstances in a neighborhood that affect all of the civilian personnel resident in the neighborhood. Abstract situations have a coverage fraction, just as activity situations do; however, for abstract situations the coverage fraction is set by the analyst when the situation is created. The complete set of abstract situation types (and the accompanying DAM rules) is given in the *Athena Rules* document.

### Athena 4 Changes

The absit rule sets changed in Athena 4 due to the switch from GRAM to URAM as the underlying attitude model. In past versions, an absit had these effects on satisfaction during its life-time:

* On inception, an inception penalty (a level effect)
* At each time step (including the first), a satisfaction drain (a slope effect)
* On resolution, a resolution benefit (a level effect)

The notion was that the absit caused satisfaction to decline until the situation was resolved; the resolution benefit would cause satisfaction to bounce back. The resolution benefit also included another effect: resolution by locals usually caused an added benefit to the group's Autonomy satisfaction.

URAM defines transient and persistent effects rather than level and slope effects (see the *Mars Analyst's Guide*) and so the rule sets had to change. An absit now has these effects:

* In the first week: a large transient effect, representing the sum of the inception penalty and the first week's drain.
* In subsequent weeks, a smaller transient effect.
* On resolution, Autonomy effects due to local resolution.

There is no need to have an explicit "bounce back" effect; when the on-going transient effects cease, the satisfaction curves naturally bounce back.

## Services

A service is something provided to the civilians, the level of which affects civilian attitudes.

### Overview

At present Athena models two types of services: abstract infrastructure services (Section 7.2), which do not require the expenditure of funds and Essential Non-Infrastructure (ENI) services (Section 7.3) which do. Services that do not have any infrastructure explicitly modeled in Athena (ie. energy delivery infrastructure) are abstract. Athena currently has three abstract infrastructure services: Water, Energy and Transportation.

#### Services vs. Abstract Situations

Athena has historically modeled services (e.g., power and water) using abstract situations, or “absits” (Section 6). In the absit model, the service is presumed to be provided unless the related absit exists, in which case the service is out and there is a satisfaction penalty. However, absits are not well-suited for Athena time horizons because they do not reflect changing expectations of service. In addition, they do not allow actors to have a role in providing services to the civilians—something an actor does to influence, benefit, and ingratiate the civilians, and to make them dependent on the actor’s leadership. Consequently, we have developed the notion of a provided service. Actors may be directly responsible for providing the service, or may indirectly facilitate provision of the service.[[34]](#footnote-34)

#### Levels of Service

A service is provided to the civilians by one or more actors. It can be provided to a greater or lesser extent, and the primary driver of attitude and relationship change is whether the provided level of service meets or fails to meet the expectations of the civilians. In particular, a service has:

* An Actual Level of Service (*ALOS*): the amount of the service currently being provided. This is a continuous variable which rises and falls over time. The natural units for the level of service will of course vary from one kind of service to another.
* A Required Level of Service (*RLOS*). If the *ALOS* is less than the *RLOS*, serious hardship results.
* An Expected Level of Service (*ELOS*). The *ELOS* will approach the *ALOS* over time. In general, the population grows happier when the *ALOS* exceeds the *ELOS*, and unhappier when the opposite is true.
* A Saturation Level of Service (*SLOS*). When the *ALOS* reaches saturation, the population has all the service they can use; increases beyond SLOS cause no net gain in satisfaction.

For simplicity’s sake, we normalize all of these variables to the Saturation Level of Service. That is, *SLOS* is scaled to be 1.0, and the other variables are all represented as fractions of *SLOS*. As a result,

Note that we allow *ALOS* to exceed the saturation level; the actors might indeed provide a superfluous quantity of the service, although they get no benefit from doing so.

#### Service Cases

As an aid to analysis of attitude and relationship effects, we define a number of cases. At any given time, with respect to a particular group *g* in a particular neighborhood *n*, one of the four following cases obtains:

* **Case R−**: The actual level is less than the required level. The people are hurting, and the expected level doesn’t matter.
* **Case E−**: The actual level is at least the required level, but is less than expected. The people will be unhappy.
* **Case E**: The actual level is close to the expected level. The people are getting what they expect.
* **Case E+**: The actual level is well above the expected level. The people are getting more than they expect, and they like it. (Note that the expected level can approach but never exceed saturation.)

### Abstract Infrastructure Services (AIS)

Actors can set the actual level of water, energy or transportation service provided to civilians, with concomitant attitude effects following the level-of-service model described in section 7.1.2. The actual infrastructure required to support the level of service is not modeled, nor are the costs associated with providing and maintaining the infrastructure; hence, we refer to these services as abstract infrastructure services (AIS).

The default actual and required levels of service for each service depend on the neighborhood’s urbanization level, and are controlled by model parameters.[[35]](#footnote-35) Initially the expected level of service for each AIS is set to the actual level of service, which means that at time 0 all groups expect the amount of service they are receiving.

Subsequently, actors can use the SERVICE tactic to modify the actual levels of service for the three services in each neighborhood; this may result in changes to civilian satisfaction and vertical relationships, as described in the *Athena Rules Document*.

### Essential Non-Infrastructure (ENI) Services

Among the tactics actors can use when trying to achieve their goals is funding of those essential civilian services that do not require significant amounts of infrastructure.[[36]](#footnote-36) To avoid repeating that mouthful every time we want to refer to them, we are calling them “essential non-infrastructure” or “ENI” services, for short. Civilian groups’ support for actors will be affected and their satisfaction levels may change. This section describes how to compute those changes. The model takes into account that the actors will usually be competing.

ENI services include such things as welfare payments, courts, public education, and any other service for which funding alone adequately describes what is needed to “make it happen.” The model can accommodate services that require other assets as well as money if those other assets are used in strict proportion to the funding, but details of how to do so are not suggested here. Athena will not execute any tactic an actor cannot afford (in terms of any asset).

Modeled services do not include provision of water, food, electrical power, or other utilities; shortages of those services are still modeled as *absits*, which can be mitigated by tactics that specify appropriate force group activities. Nor do they include law enforcement, which can be provided by assigning the LAWENF activity.

Other essential services that require substantial amounts of infrastructure, such as public transit and postal service are not covered by this model, nor are the rather distant governmental services of defense, central banking, and law-making.

#### Service vs. Funding

Because ENI services by definition do not require investment in infrastructure, we assume that the amount of service provided at any given time depends on the total funding for ENI services at that time, summed over all actors’ contributions. In particular, let

= A civilian group to whom ENI services are provided.

= The actual level of service for group *g*, as a fraction of the saturation level of service.

= The funding required to saturate the supply of ENI services to group *g*, in $/week.

= Total funding of ENI services for group *g*, in $/week.

= Shape parameter[[37]](#footnote-37) for the service vs. funding curve, where . Values greater than 1.0 imply economies of scale. A value of 1.0 indicates that service is directly proportional to funding.

= The urbanization level of the neighborhood in which *g* lives, one of ISOLATED, RURAL, SUBURBAN, or URBAN.

The saturation level of funding, , depends on the quantity of service required per person in the group and the number of people in the group:

= The population of group *g*.

= The per capita cost of providing saturated ENI services in a neighborhood with urbanization *r*.[[38]](#footnote-38)

The nominal values for are as follows:

|  |  |
| --- | --- |
| **Urbanization** |  |
| URBAN | 0.40 |
| SUBURBAN | 0.20 |
| RURAL | 0.10 |
| ISOLATED | 0.01 |

The resulting cost curve is shown in the figure, below. As drawn, it indicates that there are economies of scale (). Note that very large funding levels can be expected (in the real world) to show diminishing returns; the model can be modified to reflect that expectation if necessary, but the anticipated operating range is well below saturation.



#### Funding by Individual Actors

We assume that an actor can provide ENI services in any neighborhood in which he has at least minimal direct support.[[39]](#footnote-39) Thus, multiple actors can choose to provide ENI services to a particular group. The total funding for ENI services to group *g* is then simply

where

= Actor *a*’s funding for ENI services to group *g*.

Actors choose how much funding to provide for ENI services using the FUNDENI tactic, which is described in the *Athena User’s Guide*.

#### ENI Services and Empty Civilian Groups

An empty civilian group—a group with zero population—can clearly neither receive ENI services nor have attitudes about them. The actual, required, expected, and saturation levels of service for such a group are all defined to be 0.0.

When population is transferred into such a group, so that it is no longer empty, then the expected level of service, will be started at the actual level of service, , just as for groups that are non-empty on lock.

### Expected Level of Service

For both ENI and AIS we compute the expected level of service, , by exponential smoothing from the actual level of service and the previous expected level, capping the expectation at the saturation level:

where

= Group *g*’s expected level of service

= Group *g*’s expected level of service at the previous week.

= The actual level of service received by group *g*.

= The exponential smoothing parameter, where

The value of can be thought of as the reciprocal of the average age of data in weeks, i.e., if then the expectation is based on two weeks of data. Greater values of imply that expectations adjust more quickly than lower values. If is 1, the expected level of service will simply be just the actual level of service; the civilians will expect whatever they get. If is 0, then the expected level of service never changes.

We assume that civilians more rapidly grow accustomed to good fortune than to hardship, and consequently the value of depends on whether or not exceeds :

Here, and are model parameters[[40]](#footnote-40), nominally 0.5 and 0.25 respectively.

The initial value for at time 0 is simply the actual level of service at time 0. Thus, the civilians initially expect what they are currently getting.

#### Categorize the Actual Level of Service

To compute the attitude effects of ENI and AIS, we use the four service cases described in Section 7.1.3: **R−**, **E−**, **E**, and **E+**. The first step is to determine which of the four cases applies:

* If , then the actual is less than required; case **R−** applies.
* If , then the actual is about the same as expected; case **E** applies.
* If , then the actual is less than expected; case **E−** applies.
* Otherwise, the actual is greater than expected, and case **E+** applies.

where

= The actual level of service received by *g*.

= The expected level of service for *g*.

= The required level of service for residents of neighborhoods with urbanization *r*, where *r* is the urbanization of *g*’s neighborhood, as defined above.

= A model parameter, the half-width of the band around the expected value that defines the notion “approximately equal to”.[[41]](#footnote-41)

### Satisfaction Effects

The rate of change of civilian satisfaction due to the availability of services depends on two major considerations: (1) The difference between the actual level of service and the civilians’ expectation of service and (2) how the actual level compares to minimum (required) and maximum (saturation) levels. In general, satisfaction levels should go down if the civilians are receiving less than the minimum level of service. If the civilians are receiving at least the minimum level of service, then satisfaction should go down if they are receiving less than they expect, and should go up if they are receiving more than they expect.

To capture these two effects, we define two factors: the needs factor, , and the expectations factor, . These two factors are then used in the service rule sets to assess satisfaction effects; see the *Athena Rules* document for the specifics.

##### The Needs Factor

The factor measures the degree to which the actual level of service received by group *g* is less than or greater than that required for survival. It is based on the piecewise-linear function shown in the following figure:



In this diagram,

= The “needs” factor for group *g*.

= The actual level of service received by *g*.

= The required level of service for residents of neighborhoods with urbanization *r*, where *r* is the urbanization of *g*’s neighborhood.[[42]](#footnote-42)

= A model parameter,[[43]](#footnote-43) the gain on , nominally 2.0.

The nominal values for are shown in the following table:

|  |  |
| --- | --- |
| **Urbanization** |  |
| URBAN | 0.60 |
| SUBURBAN | 0.40 |
| RURAL | 0.20 |
| ISOLATED | 0.00 |

The function is computed using the following algorithm:

If ,

Let .

If ,

Let .

If , (the only remaining case),

Let .

The result is a number between 0 and , or nominally between 0 and 2.

The current model presumes that providing service in excess of saturation has no additional effect. This is unlikely. Providing excess service might decrease satisfaction with respect to AUT, for example. On the other hand, it might not really be possible to provide service in excess of saturation…which means that excess money spent on it is going for something else, and likely improving *somebody*’s satisfaction, and probably illicitly. In light of the abstract nature of ENI services as modeled in this version of Athena, then, we will continue to assume that the effect of excess service is flat.

##### The Expectations Factor

The rate of change of civilian satisfaction should depend on the difference between the expected and actual levels of service. When expectations are not met, satisfaction should go down; when they are exceeded, satisfaction should go up. The expectation factor, , ranges from and , or nominally between -2 and 2, and is computed simply as follows:

where

= The “expectations” factor for group *g*.

= The actual level of service received by *g*.

= The expected level of service for *g*.

= A model parameter,[[44]](#footnote-44) the gain on , nominally 2.0.

### Vertical Relationship Effects

For both ENI and abstract services, vertical relationship effects between a civilian group and an actor depend on the service case, (**E+**, **E**, **E−**, **R−**), and whether the actor is in control of the group’s neighborhood.

For ENI services, the funding provided by an actor must also be considered as an input to vertical relationship. First, we compute the credit, , given to each actor *a* for the level of ENI services provided to group *g* as follows.

If there is no funding, nobody gets any credit. This especially includes the actor in control of the neighborhood, if any:

* If , for all *a*.

Otherwise, we give credit to each actor according to the fraction of the total funding that he provides, but only up to the amount needed for saturation. Moreover, we give credit to the actor in control of the neighborhood *first*; if he provides a saturation level of service all by himself, the other actors get no credit. If he provides less than the saturation level of service, the other actors can only take credit for the increment of service provided up to the saturation level.

Let *C* be the actor in control of *g*’s neighborhood. Then,

where

= The funding required to saturate the supply of ENI services to group *g*, in $/week.

= Total funding of ENI services for group *g*, in $/week.

= Actor *a*’s funding for ENI services to group *g*.

We then categorize these contributions as shown in the following table:[[45]](#footnote-45)

|  |  |  |  |
| --- | --- | --- | --- |
| **Contribution** | | | **Categorization** |
| 0% ≤ |  | ≤ 20% | **N**: Actor *a* is providing **negligible** funding. |
| 20% < |  | ≤ 50% | **S**: Actor *a* is providing **some** funding. |
| 50% < |  | ≤ 100% | **M**: Actor *a* is providing **most** of the funding. |

The complete details of how service level and actor credit affect vertical relationship is documented in the *Athena Rules* document.

## Attrition

The Athena Attrition Model (AAM, pronounced “aim”) models attrition to force and civilian groups. AAM provides a basic framework for attrition and attrition-related modeling that includes most areas, but models each area in a simple way that can be improved in later versions.

The Athena Attrition Model uses the Lanchester attrition equations to determine the number of casualties suffered by each group. [[46]](#footnote-46) Traditional attrition models have a much smaller timescale than Athena, updating the attrition rates every few minutes. The smallest time-step in Athena is one week. Most combat doesn’t last that long, and the attrition rates will change several times over the course of one week.

Athena's attrition model also provides for magic attrition (via the ATTRIT tactic) of force, civilian, and organization groups in neighborhoods. All personnel present in the simulation are present in the form of units. Consequently, all attrition comes out of units; but it also comes out of each group’s pool of mobilized troops, thus reducing the personnel available for deployment and assignment the following week. The following sections will speak only of attrition to units; the effect on the pool of mobilized troops is implicit.

### Overview

AAM concerns itself with *engagements* between pairs of force groups (*f, g*) in a neighborhood *n*. An engagement occurs when both *f* and *g* have personnel in *n* and either *f* or *g* or both has been directed to attack the other. Engagements are governed by the following variables:[[47]](#footnote-47)

= Group *f*’s ROE with group *g*. The ROE may be ATTACK or DEFEND.

= Group *f’*s attack-to-defend force ratio with respect to group *g*. Group *f* can only attack *g* if the ratio of *f’*s force to *g’*s is at least this threshold.

= Group *f’*s defend-to-withdraw force ratio with respect to group *g*. Group *f* can only defend against *g* if the ratio of *f’*s force to *g’*s is at least this threshold.

= Group *f’*s concern for civilian casualties when fighting group *g*, one of HIGH, MEDIUM, LOW, or NONE

= Group *f*’s total personnel in the neighborhood.

= Group *f’*s personnel allocated to the engagement with group *g*.

Thus, we say that there is an engagement between *f* and *g* in *n* if is ATTACK or is ATTACK (or both).

By default, the ROEs and force ratio thresholds are set as follows:

= DEFEND

= model parameter: aam.defaultDefendThresh

= model parameter: aam.defaultCivcasConcern

Group *f’*s values with respect to group *g* in *n* can be set by *f’*s owning actor using the ROE tactic. Since combat only occurs in the presence of an ATTACK ROE, all combat is initiated by the ROE tactic and hence on an actor’s orders.

#### Posture

Group *f*’s posture with respect to group *g*, , is one of ATTACK, DEFEND, or WITHDRAW. If ATTACK, then *f* is actively attacking *g*. If DEFEND, *f* is actively defending against *g*. If WITHDRAW, *f* is attempting to withdraw from combat with *g*. Group *f*’s posture is determined by its ROE and force ratio with respect to *g* and its and thresholds.

For example, if the attack/defend ratio is 3:1, then in order for group *f* to actually attack group *g* its force ratio must be at least 3.0 at the start of the engagement. As the engagement proceeds and each group takes casualties the force ratio will change and a group’s posture might decrease from ATTACK to DEFEND to WITHDRAW as the group becomes more and more out-numbered.

A group’s attrition coefficient in the Lanchester equations depends on its posture.

#### Personnel Involved in Engagements

Force group *f* can be involved in engagements in neighborhood *n* with several force groups simultaneously. In this case, we allocate a fraction of *f*’s personnel to each engagement. Unless group *f* is hiding (see below), we assume that all of *f*’s personnel are available to fight.

#### Hiding Groups

A force group may attempt to avoid combat in a neighborhood via the HIDE tactic; an example would be a group made up of terrorist cells scattered among the civilian population. The hiding group’s effective ROE will be DEFEND; this cannot be overridden by an ROE tactic. If a hiding group is performing activities, the group is not hiding very well and is more visible to other groups in the neighborhood.[[48]](#footnote-48)

Before a force group can attack a group that is hiding, it must first detect some of the hiding group’s personnel. We assume that only the detected personnel in the hiding group will be engaged with the attacking group.

Note that since a hiding group cannot have an ROE of ATTACK, it can only be attacked by a group that is not hiding.

#### Engagement Life Cycle

An engagement between groups *f* and *g* in *n* begins when one or both groups attack the other, and lasts until neither group can fight or until a time limit is reached. After all engagements are over we total the casualties and reduce the number of personnel available in each group the following week. We also compute and apply civilian casualties.

### Personnel **Involved** in Combat

The first step in assessing attrition is determining which engagements exist, and for each engagement how many of each group’s personnel are involved. First we discuss how group *f* detects personnel belonging to hiding group *g*; second, we discuss how *f*’s personnel are allocated to engagements if it is involved in multiple engagements simultaneously.

#### Visibility and Detection

If neither group is hiding then each is perfectly visible to the other, and each group detects all of the other’s personnel. In this case the number of *f’*s personnel detected by *g*, , is defined for both groups as follows:

If group *g* is hiding, then *f* must find some or all of *g*’s personnel before it can attack them.

Any of *g*’s troops that are plainly visible are detected immediately; *g*’s other troops can be detected only with the cooperation of the neighborhood.

The visibility of a force group depends on the group’s activities[[49]](#footnote-49) and the urbanization and population of the neighborhood.[[50]](#footnote-50) We compute the number of visible personnel in hiding group *g* as

where

= A multiplier that depends on the urbanization of the neighborhood. [[51]](#footnote-51)

= A multiplier that is determined by activity *a*. [[52]](#footnote-52)

= The number of group *g’*s personnel performing activity *a* in the neighborhood.

Thus, is the number of *g*’s personnel that are “in the open.” However, *g* is assumed to be mingling with the civilian population, and so how visible the visible personnel are depends on the size of the civilian population. Consequently, we assess the visibility of *g*’s personnel with respect to size of the neighborhood by using a coverage function in the usual way:

where

*POP* = The total civilian population of the neighborhood

= A fraction of the personnel in group *g*, from 0.0 to 1.0.

Attacking group *f* ‘s ability to detect members of group *g* depends on *f*’s situational awareness in *n*. If *f* has no situational awareness, it can only detect the fraction of *g* that is visible. If *f* has perfect situational awareness, it can detect the entire group *g*. But *f’*s situational awareness depends on its ability to collect intelligence from the civilians in the neighborhood, which is to say that it depends on the cooperation of the neighborhood with *f*:

Group *f*’s ability to detect *g* is then defined as follows:

where

= The average cooperation of the civilian groups in the neighborhood with force group *f*.

This equation is simply a line segment whose endpoints are defined by the range:

If multiple groups are attacking hiding group *g*, it is possible (given these equations) for the attacking groups to detect a total number of personnel that is greater than the number of personnel in *g*. In that case, we define the actual detection fraction, , by normalizing the detection figures in the usual way.

Then, we multiply this fraction by the number of personnel in *g* to find the number of personnel in *g* detected by *f*.

Next, we assume that only some of group *f* will detect personnel in group *g*. The others are looking elsewhere in the neighborhood or are designated to other conflicts. The fraction of *f* that is involved in the conflict is determined by the percentage of *g* that *f* detects, as well as a multiplier nominally set to 1. We compute the number of personnel in *f* involved in the conflict as

where *detectionGain* is a model parameter[[53]](#footnote-53) that allows the user to adjust the number of personnel in the attacking group that are involved in combat.

The detected personnel of each group are used to compute the effective force of each group.

#### Effective Force

The number of casualties that group *f* inflicts upon group *g* does not depend just on the size of group *f*, but rather on its effective force multiplier, , a measure of the degree of force projected by one person in the group. Effective force is similar to the force projections computed in the security model in that it depends on the group’s force type, demeanor, training level, and equipment level. A group’s equipment level encompasses both the quality and quantity of its available equipment, of which there are four different levels: BEST, GOOD, FAIR, and POOR. is defined as

where

= A multiplier that depends on the equipment level of group *f*.[[54]](#footnote-54)

= A multiplier that depends on the training level of group *f*. [[55]](#footnote-55)

= A multiplier that depends on the forcetype of group *f*. [[56]](#footnote-56)

= A multiplier that depends on the demeanor of group *f*. [[57]](#footnote-57)

The force multipliers are model parameters that modify group *f*’s effectiveness due to the given axis; a highly trained force will have a higher than one that doesn’t.

Then, we compute *f*’s effective detected force by *g* as the force projected by the members of *f* detected by *g*:

### Allocating Force Group Personnel to Engagements

At this point, we build a set of potential engagements for the neighborhood, based on the ROEs of the force groups present in the neighborhood. If either group in a pair has an ATTACK ROE, we define an engagement and allocate personnel to it.

If groups *f* and *g* are engaged only with each other in neighborhood *n*, then the personnel allocated to the engagement will be all detected personnel:

Otherwise we need to allocate each group’s personnel across its engagements. We will do so in proportion to the effective detected force of the group’s opponents:

In either case, group *f’*s force allocated to the engagement with *g* is

### Lanchester Attrition in Athena

The Lanchester equations depend on the effective force multiplier for each group, the engaged personnel in each group, and the posture of each group. A group’s posture is determined by its ROE, and it actual and minimum force ratios. The groups’ postures may change as combat goes on. Thus, we apply the following algorithm:

Repeat

Determine personnel ratios for each group.

Determine the posture of each group

Assess attrition up to the next posture transition by either group

Until fighting ends due to posture transitions or the maximum combat time is exceeded.

#### Force and Personnel Thresholds

The ROE tactic governing group *f*’s engagement with group *g* specifies two force thresholds:

= Attack/Defend Force Ratio of group *f* with group *g*

= Defend/Withdraw Force Ratio of group *f* with group *g*

If *f* is attacking *g*, and the force ratio between *f* and *g*, , drops below , then *f* must cease attacking and defend; and if it drops below then *f* must withdraw.

The Lanchester equations are stated in terms of personnel rather than force levels, and so it is convenient to compute the equivalent ratios in personnel terms:

Similarly, we define the personnel ratio between *f* and *g* as

#### Determine Postures

Given the above definitions, when is ATTACK then the posture of group *f* with respect to *g* is

When is DEFEND, the posture is computed in the same way but is capped at DEFEND.

Group *g*’s posture is determined in the same way.

#### Attrition Coefficients

As combat continues, group *f* causes attrition to group *g*; the amount of attrition depends on the attrition coefficient, . Similarly, group *g*’s attrition to *f* depends on .

We take many different factors into account when computing for a particular engagement. The first is the nominal attrition coefficient, , which depends on the postures of groups *f* and *g*, e.g., *f* will cause more damage to *g* if *f* is attacking rather than defending, and no damage at all if both groups are withdrawing. The values of by posture are set by model parameters.[[58]](#footnote-58)

We then adjust the nominal attrition rate by a number of factors. Some of these factors are based on the same force multipliers we used to describe group *f* above; others represent environmental effects on force. In each case, the factor is a force multiplier divided by a constant to create a factor that is exactly 1.0 in the nominal case. The factors are defined as follows.

**Urbanization:**[[59]](#footnote-59)The degree of urbanization alters the effective force level of the groups involved. The nominal urbanization level is URBAN. Therefore,

**Concern for Civilian Casualties:** [[60]](#footnote-60) A group with a high degree of concern for civilian casualties will be able to inflict less attrition than one with a low degree of concern for civilian casualties. The nominal degree of concern is NONE:

**Equipment Level:** The nominal equipment level is BEST.

**Force Type:** The nominal force type is REGULAR:

**Training Level:** The nominal training level is PROFICIENT:

**Demeanor:** The nominal demeanor is AVERAGE:

Given these factors, we then define as follows:

Thus, when *f* is a fully trained, fully equipped REGULAR force group with average demeanor and no concern for civilian casualties engaged with *g* in an urban neighborhood.

#### Combat Time

The Lanchester Attrition equations are differential equations used to update the number of personnel in each group as combat progresses. The following algorithm incorporates the solutions to the differential equations. The interested reader may find the Lanchester system of equations, as well as the derivation of its solutions, in Section 8.9.

We solve the Lanchester equations for the time *h* in combat hours until one group no longer meets the minimum force ratio for its current posture. These solutions to the Lanchester equations depend on the attrition rates and . The four possible cases are described below.

Case 1: and . In this case, there is no attrition. STOP.

Case 2: and . We solve for the combat time measured in hours as:

Case 3: and .

Case 4: and . In this case, we begin by computing two constants, and that are found in the solution to our differential equations.

The number of combat hours until the next posture threshold is crossed then depends on the signs of and .

It can be shown that these restrictions on guarantee a feasible solution. Note that cannot be zero, hence the ratio cannot be zero.

No matter which case we are in, we have to make sure that we do not exceed the maximum number of combat hours per week, which is specified by a model parameter[[61]](#footnote-61). Initially, the value of is set to this this parameter. Then we compute:

We will use to update the number of personnel in each group. Then, if is greater than 0 we will loop through this algorithm again, starting from the beginning of Section 8.3.

#### Compute and Apply Attrition

Given the combat time in hours, we must compute the actual attrition to each group and reduce its numbers. Once again our solution depends on the values of the attrition coefficients, and there are four cases.

Case 1: and . If we are in this case, the algorithm has already been stopped.

Case 2: and . Group *f* loses personnel:

Case 3: and . Group *g* loses personnel.

Case 4: and . In this case, we use the same constants as before, and to update the number of personnel in each group:

At this point in the algorithm one group has either died out, changed its posture, or the maximum combat time has elapsed. If the group has no more personnel then the combat is over. If there is still time left ( > 0), then we will repeat the algorithm from the top of Section 8.3 onward. We will re-evaluate the posture of each group, compute new attrition coefficients, find the time until the next posture change, and update the number of personnel.

### Casualties

#### Force Group Casualties from one Combat

An engagement ends when the two groups have postures that do not allow them to fight anymore, when the maximum combat time has been exceeded, or when one group has no more personnel. At the end of each combat, we compute the number of casualties to each group in the engagement as the difference in personnel from the previous week.

where *t* is the current simulation time.

#### Total Force Group Casualties

After all combats have ceased and we have computed the casualties that result from each engagement, we compute the total number of casualties to each group *f* and update its number of deployed personnel:

From this equation, it is possible for a hiding group to take more casualties than it has personnel, but this is not realistic. In this case, the personnel is 0 and the number of casualties that each group inflicts on the hiding group should be reduced proportionally.

If ,

This ensures that no more than the entire group can be killed.

#### Civilian Casualties

When two groups engage in combat there will be some collateral damage. We compute civilian casualties caused by force group *f* as a fraction of the force group casualties caused by *f*. This fraction will be determined by the urbanization of the neighborhood, as well as *f*’s force type and training level.

where

= A multiplier that depends on the urbanization of the neighborhood. [[62]](#footnote-62)

= A multiplier that depends on the force type of *f*. [[63]](#footnote-63)

= A multiplier that depends on the training level of *f*. [[64]](#footnote-64)

Then we compute the total civilian casualties caused by *f* in this neighborhood this week as

where

= A multiplier that depends on *f*’s concern for civilian casualties when engaged in combat with *g*. [[65]](#footnote-65)

= The maximum fraction of the civilian population that can be attrited by a force group in a single week. [[66]](#footnote-66)

= The total population of the neighborhood.

This number is divided proportionally amongst the civilian groups in the neighborhood. Then, we subtract the number of civilian casualties in each group to get the number of civilians next week.

### Attrition and Mobilized Troops

All attrition comes out of the attrited units; but it also comes out of each force group’s pool of mobilized troops in the neighborhood, thus reducing the personnel available for deployment and assignment the following week. The following sections will speak only of attrition to units; the effect on the pool of mobilized troops is implicit.

### Magic Attrition

The analyst can attrit units and groups magically. All attrition to civilian units will be assessed by the relevant DAM rule set.

#### Magic Attrition to Groups

The analyst can attrit a specific group in a specific neighborhood. Only units present in the neighborhood will be attrited.

#### Magic Attrition to Neighborhoods

The analyst can also choose to apply attrition to all civilian units that happen to be in a neighborhood; this is equivalent to collateral damage incurred during combat.

### Assessing the Attitude Implications

Once attrition has been computed and applied, it is necessary to assess the implications for civilian attitudes via the CIVCAS rule set in the Driver Assessment Model (DAM). The rule set is documented in the *Athena Rules Document*; the remainder of this section describes the inputs to the rule set.

#### Satisfaction Effects of Attrition

We will assess the satisfaction effects of attrition on a civilian group as follows. First, all attrition occurring during the week, both magic and normal, to group *f* in neighborhood *n* will be accumulated in the **attrit\_nf** table.

When the CIVCAS rule set is triggered to assess satisfaction effects it has access to the following values:

*n* = The neighborhood in which combat occurred

*f* = The attrited civilian group.

*casualties* = The total number of casualties to *f* in *n* during the week.

= A Z-curve which converts a total number of casualties into a casualty multiplier used in the CIVCAS satisfaction rules.[[67]](#footnote-67)

*M* = The casualty multiplier.

We compute the casualty multiplier, *M*, as follows:

#### Cooperation Effects of Attrition

Athena only tracks the cooperation of neighborhood groups with force groups; consequently, we assess cooperation effects only for attrition to civilian groups. The CIVCAS rule set attends to this.

All normal civilian attrition is (at present) due to collateral damage resulting from fighting between two force groups. Magic attrition can optionally be attributed to one or two force groups. As attrition occurs, Athena accumulates the total casualties to group *f* in neighborhood *n* in which force group *g* was in some way involved in the **attrit\_nfg** table. If two force groups are involved in an altercation, as is usually the case, the total civilian casualties are attributed equally to both.

When the CIVCAS rule set is triggered to assess cooperation effects it has access to the following values:

*n* = The neighborhood in which combat occurred.

*f* = The attrited civilian group, resident in *n*.

*g* = A force group.

*casualties* = The total number of casualties to *f* in *n* during the past week, in which *g* was involved.

= The relationship between civilian group *f* and force group *g*.

= A Z-curve which converts a total number of casualties into a casualty multiplier used in the CIVCAS cooperation rules.[[68]](#footnote-68)

*M* = The resulting casualty multiplier.

We compute the casualty multiplier, *M*, as follows:

#### Horizontal Relationship Effects of Attrition

The horizontal relationship effect between force groups responsible for civilian casualties and the civilian group that suffered the casualties is similar to the cooperation effect in Section 8.8.2 above. The only difference being that the casualty multiplier is computed from a different Z-curve[[69]](#footnote-69).

#### Vertical Relationship Effects of Attrition

The vertical relationship between the actor that owns a force group responsible for civilian casualties is also affected.

When the CIVCAS rule set is triggered to assess vertical relationship effects it has access to the following values:

*n* = The neighborhood in which combat occurred

*f* = The attrited civilian group.

*a* = The actor that owns a force group *g* responsible for casualties.

*casualties* = The total number of casualties to *f* in *n* during the week.

= A Z-curve which converts a total number of casualties into a casualty multiplier used in the CIVCAS vertical relationship rules.[[70]](#footnote-70)

*M* = The casualty multiplier.

We compute the casualty multiplier, *M*, as follows:

### Deriving the Lanchester Solutions

The Lanchester Attrition equations are first-order differential equations used to compute the number of casualties that a group receives in a time-step. The Lanchester equation for two groups using “aimed fire” is represented by

where *x* and *y* are the number of personnel in each group and *a* is a positive multiplier. Basically, this equation says that the number of casualties to group *x* is determined by the number of people shooting at them in group *y*. We use a similar differential equation to compute attrition for group *y*.

Our solutions are subject to the following constraints:

where and are the minimum force ratios. First, let us consider Cases 2 and 3 as described in Sections 8.3.3 and 8.3.4. One group has an attrition coefficient of zero. WLOG, suppose that group is x. Then and we have the following system of differential equations:

We can easily tell that y is constant, which means that x will decrease at a constant rate. Group y will never reach its minimum force ratio, so we only have to solve for t when . We find:

where and are the initial populations of each group (at the beginning of the current week). Then using this value of t, we can find the new number of personnel in x.

Note that we could also use as the new value for x. We have solved the Lanchester equations when one attrition coefficient is zero. Now, we will solve the system of equations for when both coefficients are non-zero. This corresponds to Case 4. We have:

subject to the following constraints:

We will solve this system of differential equations by finding the eigenvalues and eigenvectors. The eigenvalues are given by

Whose eigenvectors are

This gives us the solutions

Setting , we have

Solving for the constants we find

where and are the initial values of each population. We want to find the time when our constraints fail. So we set and solve for *t* to find

And when, we have

We only care about positive times, so we want the inside of the natural logarithm to be greater than 1. It can be shown[[71]](#footnote-71) that one of these conditions always holds true and the other always fails depending on the sign of . Specifically, if , then our constraints imply that and On the other hand, if , then our constraints imply that and . To guarantee a feasible solution, we include the following conditions when computing *t*:

If , then .

If , then .

We substitute this value of *t* into the and listed above in order to find the new populations of each group.

## Demographics

The Athena Demographics model is responsible for:

* Tracking the civilian population of the playbox, by group, as it changes:
  + Due to births and natural deaths
  + Due to collateral damage
  + Due to explicit flows from one civilian group to another
* Breaking down the population of a group into subcategories, e.g.,
  + The number of consumers
  + The number of workers
  + The number of subsistence farmers and ranchers
* Rolling up the group demographics to the neighborhood and playbox levels.

The initial playbox population is input to the scenario as the *base population* of each civilian group. In addition, each group has a rate of change due to births and natural deaths.

### Connections with Other Models

The demographics model is intended to be as simple as possible while meeting the needs of the other Athena models. In particular:

* URAM requires the current population of each civilian group.
* The Athena attrition model requires that the civilian population can take attrition as indicated by the ATTRIT tactic. In the future, we might also have collateral damage as the result of combat between forces, as well as direct targeting of civilians by terrorists.
* The Economics Model requires the number of people who participate in the regional economy (the consumers) and the number of people in the potential labor force (the workers).
* The Ground Model requires that civilians can be moved from one group to another, possibly in another neighborhood.

### Simplifying Assumptions

We make the following simplifying assumptions:

* The histogram of group population by age is approximately flat, i.e., if 10 children are added to the population then there are also 10 more people of working age.
* The subsistence agriculture population is simply the population of each subsistence agriculture group.
* The subsistence population is outside the regional cash economy.
* The labor force is a simple fraction of the total consumers in each non-subsistence group.

### Population and Units

Athena 1 had the distinction between explicit population (represented in units) and implicit population (the default). Starting in Athena 3, units are an output rather than an input: the ground model "staffs" units based on resident populations and deployed forces, both for the purpose of visualization and to ease subsequent computations. As a result, every civilian is represented both implicitly and explicitly, and the distinction is no longer of interest.

### Civilian Group Population over Time

The population of civilian group *g* over time is governed by the following equations:

where

= The population of group *g* at time *t*

= The base population of group *g* at time zero, as defined in the scenario.

= The change in the population of group *g* at time *t* due to births and natural deaths.

= The number of people flowing into group *g* from other groups at time *t*.

= The number of people flowing out of group *g* into other groups at time *t*.

= The total number of casualties suffered by group *g* due to collateral damage or other violence at time *t*.

#### Natural Population Change

Athena includes a coarse model of population change. Each group *g* has a yearly rate of population change, , representing the net effect of births and natural deaths. It will be positive if births outnumber deaths, and negative if the reverse is true. It is assumed to be constant for the duration of the simulation.

Since we do not track population cohorts and assume that a constant fraction of each group is in the labor force, we are further assuming that the rate has been constant since the birth of those entering the work force at time zero.

The rate is expressed as the percentage change per year. The population figures will be updated on a weekly basis; thus,

Note that we are effectively compounding the rate weekly; thus the effective annual rate will be very slightly higher than

The other models in Athena expect the population to be an integer; however, the value of natural includes a fractional part, and (especially for small groups) the sum of those fractional parts can be significant. Rather than rounding or truncating , consequently, we will retain the fractional part, so that the fractional change can accumulate over time. Thus we define

so as to have an integer number to work with. Other Athena models will thus refer to .

If , then all of the actual people are gone. We will accordingly set both and to zero; otherwise we will have the embarrassing situation (given a long enough run) of having a person pop out of nowhere due to the application of .

#### Civilian Attrition

Athena's attrition model targets personnel in units created by the ground model, but the actual attrition is applied to the pool of people from which the units are staffed. For civilians, consequently, attrition to a civilian group's population is handled by subtracting the casualties from as shown above.

### Population Breakdowns

The Demographic model breaks each group's population down into categories.

#### Subsistence Population

The subsistence population of civilian group *g*, , is total resident population of the group, or 0 if the group does not do subsistence agriculture:

Note that displaced personnel are *never* part of the subsistence population. When civilians flee their lands, they can no longer raise crops or herd cattle on them. Displacement is modeled by flowing population from the subsistence group into another group; the second group should not be a subsistence group.

On the other hand, a group of nomadic herdsman that travels from neighborhood to neighborhood over the course of a year can be modeled as two (or more) subsistence groups in different neighborhoods; population can flow between them as appropriate.

#### Consumer Population

The consumer population of a civilian group of a group is simply the resident population if the group participates in the regional economy, and 0 otherwise. Thus, we define civilian group *g*'s consumer population as follows:

#### Labor Force

In principle, different civilian groups contribute to the labor force to different degrees, depending on their attributes. At present, however, we assume that the number of workers in a group is a simple fraction of the number of consumers in the group. Thus, we define the labor force[[72]](#footnote-73) of non-subsistence group *g* as follows:

where is the Labor Force Percentage for group *g*: the percentage of consumers who are in the labor force.

In Athena 4 and prior, the *LFP* was affected by the activities performed by the civilian group's population; for example, some or all of the group's members could be *displaced*, which reduced the *LFP*. The civilian activities model has been removed from Athena in preference to the new ability to flow personnel from one group to another; and a group that represents displaced persons can simply have a lower *LFF* than a group that is "at home".

### Aggregate Statistics

#### Neighborhood Population

The total population of neighborhood *n* is simply the population of all resident civilian groups:

The consumer population and labor force are computed similarly:

#### Regional Population

The regional population, consumers, and labor force are simply summed up across the “local” neighborhoods in the obvious way.

#### Labor Security Factor

The labor security factor, *LSF*, is the fraction of the playbox labor force that is available to work given group security levels; or, to put it another way, is the fraction of the labor force that is afraid to go to work due to poor security.

First, let be the fraction of group *g*'s labor force that is willing to go to work given group *g*'s security level. It is defined according to the following table:[[73]](#footnote-74)

|  |  |
| --- | --- |
|  |  |
| None | 0.90 |
| Low | 0.95 |
| Medium | 0.98 |
| High | 1.00 |

Then, *LSF* is simply the actual fraction of the labor force available to go to work:

#### Consumer Security Factor

The consumer security factor, *CSF*, is the fraction of the consumers in the playbox who are willing to go shopping given group security levels; or, to put it another way, is the fraction of the consumers that are afraid to go shopping due to poor security. As such, it is completely parallel to the labor security factor.

First, let be the fraction of group *g*'s consumers that are willing to go shopping given group *g*'s security level. It is defined according to the following table:[[74]](#footnote-75)

|  |  |
| --- | --- |
|  |  |
| None | 0.88 |
| Low | 0.92 |
| Medium | 0.98 |
| High | 1.00 |

Then, *CSF* is simply the actual fraction of the consumers that are willing to go shopping, aggregated across the playbox:

### Unemployment

Unemployment can drive attitude change. The Economics Model computes the demand for jobs, *QSpop*, for the region of interest. This gives rise to unemployment by neighborhood and civilian group.

#### Disaggregation to Neighborhoods

Athena assumes that jobs exist where goods production infrastructure exists. Because of this, unemployment can be disaggregated to neighborhoods based upon the total production capacity of the goods sector and the goods production capacity of each neighborhood as computed by the goods production infrastructure model. The demand for jobs by neighborhood is:

where

= The number of jobs in neighborhood *n*.

= The capacity constrained demand for jobs as computed by the Economic model.

= The **goods** production capacity of neighborhood *n*.

= The production capacity of the entire **goods** sector.

However, only those in the active labor force are available to fill the demand for jobs. The active labor force by neighborhood is given by:

where

= The active labor force in neighborhood *n*.

= The total labor force in neighborhood *n*.

= The turbulence fraction, those “in between” jobs.[[75]](#footnote-76)

The turbulence fraction is set to 0.04 representing 4% of the labor force in between jobs.

Athena assumes that workers prefer to work as close to home as possible, so we assign workers to jobs so that as many as possible work in their own neighborhoods. Of those left over, as many as possible commute to work in *near* neighborhoods followed by *far* and then *remote* neighborhoods.[[76]](#footnote-77) After workers have been assigned to jobs any workers left over are part of the unemployed work force and are referred to as being “geographically unemployed”. Note that setting the maximum commute distance to *remote* will always result in a geographic unemployment of zero.

Disaggregation of unemployment is an iterative process that progresses by increasing proximity. Before iteration begins the available labor force by neighborhood and the available jobs by neighborhood is initialized.

for all neighborhoods, m.

for all neighborhoods, n.

Iterating over each *proximitymn*, neighborhood *n* has jobs for which there is a work force in neighborhood *m* available to work them:

where

= The total available labor force for the jobs in neighborhood *n*.

= The available labor force in neighborhood *m* given *proximitymn.*

Once the labor force available for jobs in a neighborhood is determined, job offers can be made to the labor force in each neighborhood *m* given the those jobs in neighborhood *n*:

,

otherwise

After all job offers are determined for each pair of neighborhoods with the given proximity relationship, positions can be filled by allocating the available labor force to those jobs. First, the total number of job offers to *m* is summed:

Then, filled positions are allocated:

for

otherwise

Note that if there are no job offers at all to *m*, then there are no filled positions. Finally, the number of filled positions is decremented from both the jobs available and the labor force available:

This process is repeated at each level of proximity using the updated *AvailLFm* and *AvailJobsn* for all pairs of neighborhoods *m* and *n* until the available labor force in *m* multiplied by the available jobs in *n* is zero:

for all *m*, *n*

Upon reaching this condition, either all available jobs have been filled or all available labor force has been exhausted. The unemployment in neighborhood *n* is then the workforce remaining after all iterations at each proximity level is completed along with those workers in turbulence:

The unemployment rate in the neighborhood is:

#### Disaggregation to Civilian Groups

Once unemployment has been disaggregated to neighborhoods, disaggregation to groups is based on the size of the group populations in each neighborhood:

where

= The number of unemployed workers in group *g*.

= The population of group *g.*

= The population in neighborhood *n* (the sum of the population of all groups *g* in *n*.)

= The number of unemployed workers in neighborhood *n.*

The unemployment rate for each group is then:

#### Geographic Unemployment

It is possible that the unemployment computed by the disaggregation algorithm is different than the unemployment computed in the labor and goods capacity constrained view of the economic model. This is because the disaggregation algorithm takes into consideration the proximities that neighborhoods have with each other. The amount of unemployment in the disaggregation model is necessarily greater than or equal to the unemployment in the economic model, thus *geographic unemployment*, or GU for short is given by:

where

= The number of unemployed workers disaggregated to neighborhood *n*

= The number of unemployed workers as computed by the labor and goods capacity constrained view in the CGE

This difference, which may be zero, is then fed into the CGE on the next tick and applied to possibly further constrain the available labor in the “Constrained with Geo. Unemp. and Sec. Factors” view of the economy.

#### Unemployment Situations

How does unemployment affect the civilians?

* If there are many unemployed civilians, the fact of their presence will become a problem for their neighbors due to increases in crime and other social conflict. This effect will be negligible for small unemployment rates, and dominant for large ones (for some value of "small" and "large"). This will particularly affect Safety (SFT), and possibly Autonomy (AUT), and should depend on the effective unemployment rate of the neighborhood as a whole.
* Unemployed civilians earn no money, and hence cannot afford to buy goods. Prior to Athena 5, this effect was computed based on the degree of unemployment; now it is computed directly. See Section 9.8.

If unemployment in a neighborhood is sufficiently dire, it will engender an UNEMP (unemployment) demographic situation. We define the unemployment attitude factor (UAF) as a multiplier used to scale the magnitudes in the situation's rule set; it ranges from 0.0 to 2.0 , and is computed from the unemployment per capita using a Z-curve[[77]](#footnote-78), as follows: 

The above curve says that the unemployment rules will not begin to affect general attitudes until about 5% of the general population are unemployed; the rules reach their nominal magnitude when 10% of the population are unemployed; and the rules have their maximum effect when 15% are unemployed. This is not to say that problems do not continue to get worse when more than 15% of the population is unemployed; but very high unemployment brings other problems that can't simply be addressed by an attitude rule set.

Using the above curve, we compute the UAF for the neighborhood as a whole:

Then, an unemployment situation will exist in neighborhood *n* if . The situation is assessed by the UNEMP rule set; see the *Athena Rules* document for details.

Given that QOL effects due to insufficient consumption are handled by the Consumption of Goods model (Section 9.8), we assume that the primary effect of unemployment is on group SFT and AUT due to the number of unemployed workers in the neighborhood as a whole. For example, one would expect the crime rate to rise with the number of unemployed workers.

### Consumption of Goods

The Consumption of Goods model covers the effects on civilian attitudes of consuming a greater or lesser quantity of goods. It is similar to the ENI Services model (Section 7) in that it takes both minimum requirements and expectations into account. Unlike the Services model, however, we do not assume that every member of population receives the same amount of goods. Instead, we explicitly model income disparity and compute the fraction of each group that is living in poverty.[[78]](#footnote-79)

We deal with two distinct kinds of effect.

* Negative attitude effects due to the fraction of the group living in poverty, i.e., effects due to actual hardship.
* Attitude effects due to the actual level of consumption being higher or lower than expected, i.e., effects due to perceived hardship or prosperity.

**Subsistence agriculture groups:** Groups that live by subsistence agriculture do not participate in the regional economy, and hence do not consume goods. In the remainder of this section, the word "group" refers only to non-subsistence agriculture groups.

#### The Actual Level of Consumption (ALOC)

The economy's total consumption of goods, measured in goods baskets, is computed by the Economics model each week as a yearly rate of consumption, denoted *QD.goods.pop*. We disaggregate this total consumption to individual groups by assuming that group income comes predominantly from employment, and that consequently only employed civilians can buy goods. Thus, each group consumes goods in proportion to its share of the employed labor force:

where

= The total consumption of goods by group *g* this week, in goods baskets.

*QD.goods.pop* = The total consumption of goods in the regional economy, this week, expressed in goods-baskets/year.

= The number of employed workers from group *g* this week.

The goods are purchased by the employed using their wages, but are consumed by the group as a whole. Thus, the actual level of consumption (ALOC) for group *g*, expressed in goods baskets per capita per week, is

This is our basic measure of consumption of goods for group *g*.

#### The Expected Level of Consumption (ELOC)

Group *g'*sexpected level of consumption, , is the number of goods baskets per capita per week that group *g* expects to be able to purchase and consume, given past history. As with the ENI Services model we use exponential smoothing to compute expectations; and as with the ENI Services model we assume that civilians get used to a higher level of consumption more quickly than they get used to a lower level of consumption. First, let be the difference between the actual level of consumption this week and the expectations resulting from last week's consumption.

then

where

= The smoothing coefficient when actual consumption is higher than expected.[[79]](#footnote-80)

= The smoothing coefficient when expectations are higher than the actual consumption.[[80]](#footnote-81)

#### The Expectations Factor

Civilian attitudes are affected by actual consumption *vs.* expectations. As is usual, we define a multiplicative factor, , to drive a rule set:

= The expectation gain, nominally 3.0.[[81]](#footnote-82)

The main term yields a number between −1.0 and 1.0 when is greater than or equal to 1.0 goods basket per capita per week; the lower bound moves from −1.0 to 0.0 as moves from 1.0 to 0.0.

This equation differs considerably from that used to compute the ENI Services expectations factor. In the Services model, we assume a saturation level of service (SLOS), and normalize all level-of-service values to SLOS=1.0. For consumption of goods, however, a moment of reflection will show that there is no saturation level of consumption; few people would object to consuming more than they currently do. Thus, the term is in theory unbounded above.

But a moment's further reflection will show that the effect on one's attitude *is* bounded. Twice my current consumption is quite a big change; ten times my current consumption is unlikely (in the short term, at least) to have any larger an effect on my attitude.

The outer min() function in the definition of , then, rather arbitrarily cuts off at +1.0, corresponding to .

As usual, we prefer the magnitude symbols in our rule sets to correspond to intermediate inputs rather than to extreme inputs. Hence we include the term to stretch the range of from [−1.0,+1.0] to [−3.0,+3.0].

#### The Required Level of Consumption (RLOC)

The required level of consumption () is that degree of consumption (in goods baskets per capita per week) that demarcates the regional poverty line. Individuals consuming less than the RLOC are living in poverty; those consuming more are not. We define as follows:

where

*u* = The urbanization level of *g*'s neighborhood

= = The required goods baskets per capita for that urbanization level, expressed in goods baskets per year.[[82]](#footnote-83)

The RGPC naturally varies from region to region; the default values are as follows:

|  |  |
| --- | --- |
| Urbanization | RGPC, goods baskets/year |
| ISOLATED | 0 |
| RURAL | 350 |
| SUBURBAN | 400 |
| URBAN | 450 |

#### The Lorenz Curve

The Lorenz Curve is a common method for describing the distribution of income among a population.[[83]](#footnote-84) The *x*-axis of the curve is the fraction of the total population in order from poorest to richest; the *y*-axis is the percentage of total income earned by that fraction of the population:

The straight line shows what the curve would look like if every person in the population earned the same income. The line "From Quintiles" shows what the distribution actually was, from data; and the "Approximation" line is plotted using a power function, , one of the common approximations for the Lorenz curve.

In public sources, income distribution is often given as the Gini coefficient[[84]](#footnote-85), a fraction from 0.0 to 1.0 (or, commonly, from 0.0% to 100.0%). A coefficient of 0.0 would indicate perfect equality (i.e., a straight line); a coefficient of 1.0 would indicate that one person has all of the income. According to Wikipedia, the United States had a Gini coefficient of 0.467 in 2008.

When is used to approximate the Lorenz curve, we define *n* as

This equation has a singularity at ; but as noted above, when one person gets all of the income. In that exceptional case, we can simply assume that everyone is in poverty.

Now, the basic Lorenz curve relates fraction of population to fraction of total income. For our purposes, we want to know the income in absolute terms, rather than as a fraction of the total. Thus, we define the modified Lorenz curve for group *g* as

where is the total weekly income of the **pop** sector of the economy, disaggregated to groups:

The function tells us the income of the poorest fraction *x* of group *g*. We will use this to construct a consumption curve.

#### The Consumption Curve

Group *g* doesn't spend all of its income on goods, but only a fraction. Thus, we can say that the poorest fraction *x* of group *g* consumes the following number of goods baskets each week:

where is the fraction of income the poorest fraction *x* uses to purchase goods. In this model, we assume that is constant for all fractions *x*,[[85]](#footnote-86) and is simply

The consumption curve is then just

or, equivalently,

where is the total cost of consumption for group *g*.

#### The Poverty Fraction

Our goal in using the Lorenz curve is to determine the fraction of group *g* that is living in poverty. The poverty line, as defined in Section 9.8.4, is , the required level of consumption, expressed in goods baskets per capita per week. We can equivalently express this as , the required cost of consumption in dollars per capita per week, where

where is the cost of one goods basket in dollars.

Then, we want to find the fraction of the group that receives less than or equal to .

The civilians in *g* lie along the *x*-axis of the consumption curve, which has been normalized by *g*'s population. For person with index *r* at position *x*, we have

This is the cumulative distribution of *g*'s consumption. We take the derivative with respect to *r* to find the consumption of the individual with index *r* corresponding to *x*:

We set , and solve for *r*, which yields the index of the person living right at the poverty line:

We can then compute the fraction of the group living below the poverty line by dividing by the population of *g*:

The fraction contains *P.goods* in both the numerator and the denominator, so we can further simplify this to

As drops below the value of as defined here can exceed 1.0, rising to ∞ at Thus, we define

#### The Poverty Factor

As with expectations, we define a multiplicative factor, based on the poverty fraction, for use in rule sets. The poverty factor, *povf*, is defined using a Z-curve, :[[86]](#footnote-87)

The default parameters of the Z-curve are {}, i.e., *povf* is 0.0 for poverty fractions less than or equal to 5%, and scales smoothly up to 1.0 from there.

The Z-curve has two purposes: first, it allows the analyst to scale the effect of the poverty fraction as desired; second, the default parameters reflect the fact that some degree of poverty is inevitable, and is unlikely to affect attitudes significantly.[[87]](#footnote-88)

#### Consumption Situations

The *Athena Rules* document defines the CONSUMP rule set, which assesses consumption situations. A consumption situation obtains for group *g* if or for group *g*. See the *Athena Rules* document for details.

## Economics

The Athena Economics area models the economy of the region of interest, which can be an entire country, a portion of a country, or several small countries taken together. We will refer to this as the *local economy*. The core of the Economics area is a 6-sector Computable General Equilibrium (CGE) model solved as a system of non-linear equations using the Gauss-Seidel algorithm. The inputs to the CGE come from a Social Accounting Matrix (SAM) that the user populates with “base case” data to determine the shape of the economy. Other inputs come from the definitions of the actors and their strategies and other scalars such as remittances. Once all inputs are defined, the CGE is calibrated using initial values from the SAM. This document will give an overview of the Economics area, with focus on how the SAM is computed and how actors’ incomes, expenditures and strategies affect the economy. It then explains how the CGE is embedded in it and how it relates to the rest of Athena.

For comprehensive and detailed documentation on the CGE and the theory behind it, see JPL Publication 12-28, “Athena’s Computable General Equilibrium Model,” by Chamberlain, Duquette and Kahovec, which is included with the Athena documentation.

### Sectors

Athena partitions the local economy into six sectors: **goods**, **pop**, **black**, **actors**, **region** and **world**.

**The "goods" Sector:** The **goods** sector includes all production of goods and services in the local region. The unit of production is the *goods basket* (GBasket), a notional basket of goods and services sized so that it costs $1.

**The "pop" Sector:** The **pop** sector includes all labor by the workers in the local region, and all consumption by the population of the local region. The unit of production is the work-year of an *average worker*. Just as the goods basket represents a notional bundle of goods and services, the average worker represents a notional bundle of skills and kinds of work.

**The** **"black"** **Sector**: The **black** sector includes any illicit products that are appropriate for the scenario. This could include narcotics, illegal weapons or human trafficking. The unit of production is the metric ton (or tonne). The **black** sector has a unit price, which is exogenous (i.e. it is determined by international competition), and is based on one tonne of finished product. The black sector may also need an imported feedstock, the material from which the finished product is made. So, for example, if the product of the black market is opium then the feedstock might be imported poppies or even imported opium. The feedstock price and how many units of feedstock are required for one unit of product are also parts of the data for this sector. If the feedstock is not imported, it is obtained from the goods sector and is not modeled explicitly.

**The** **"actors"** **Sector**: The **actors** sector is an aggregation of each of the actor's revenues and expenditures as they are defined in Athena. This sector’s revenue is determined by individual actors’ incomes from the various sectors and what cut, if any, an actor receives from black market profits.[[88]](#footnote-89) The expenditures in this sector are determined by actors’ strategies. Actors in the scenario defined as BUDGET actors can infuse money into the economy increasing the GDP. Actors in the scenario defined as INCOME actors cannot since the money they spend came from the economy.

**The (rest of the) "region" Sector:** The **region** sector is involved in every transaction in the local economy that isn't included in the other sectors. That is, the **region** sector includes transactions involving transportation, electrical power, small businesses, education, religion, and everything else other than products of the **goods** sector, **black** market goods, goods or services provided by **actors**, or transactions with the **world** outside of the study region. The (rest of the) **region** sector of a CGE with higher resolution (that is, more sectors) would not include those transactions accounted for in the additional sectors.

**The (rest of the) "world" Sector:** The **world** sector is involved in every monetary or value transaction that flows across the study region's borders. In practice, this means remittances, imports, exports and foreign aid. Products (including the **black** market) and jobs filled by people from the **pop** sector and paid for by the **world** sector are exports. Products supplied by the **world** sector are imports. Black market feed stocks can also be imports. Flows of money from the **world** sector to the **region** or **actors** sectors are considered to be forms of foreign aid.

### The Social Accounting Matrix (SAM) Tableau

During scenario preparation, the Econ tab displays a matrix of money flows and other inputs used to compute salient outputs and the parameters that define the shape of the economy. This matrix, along with other inputs, is the Social Accounting Matrix (SAM) for the economy in question. As the inputs to the SAM are made or modified, we say the “shape” of the economy is determined by computing a set of shape parameters. The shape parameters and other outputs are used as inputs to Athena’s Computable General Equilibrium (CGE) matrix which shows the current state of the economy as simulated time advances.

In short, the size of the economy is based upon some assumed number of consumers. Then, the SAM is used to determine the shape of the economy which is then handed off to the CGE. Given a varying number of consumers, the CGE is used to compute money flows and derived values based on those money flows. The Gross Domestic Product and Unemployment Rates are used to calibrate parameters in the CGE, so they must correspond to the region of interest. If possible, the SAM should also correspond to the region. The following sub-sections describe the SAM inputs and its computed outputs.

#### SAM Matrix Inputs

In scenario prep, the “input flows” matrix is used to define money flows between the sectors in a base case. Using the values in the input flows matrix, Athena computes the “balancing flows” matrix to take into account remittances, graft, black market feedstock and black market profit. These two matrices are then summed to arrive at the Base SAM, which is used to determine the size and shape of the economy. A discussion of the size and shape of the economy is beyond the scope of this document.[[89]](#footnote-90) The form of these inputs is in a spreadsheet-like tableau[[90]](#footnote-91) (note that some values, indicated by non-italicized font, are computed from other values, which are italicized).

The input flows matrix:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **goods** | **black** | **pop** | **actors** | **region** | **world** | **Revenue** | **Price** | **Demand** |
| **goods** |  |  |  |  |  |  |  |  |  |
| **black** |  |  |  |  |  |  |  |  |  |
| **pop** |  |  |  |  |  |  |  |  |  |
| **actors** |  |  |  |  |  |  |  |  |  |
| **region** |  |  |  |  |  |  |  |  |  |
| **world** |  |  |  |  |  |  |  |  |  |
| **Expense** |  |  |  |  |  |  |  |  |  |

where

= The payment in $/year from sector *j* to sector *i*.

= The total expenditure, in $/year, of sector j.

= The total revenue, in $/year, of sector j.

= The price, in $, of one unit of production of sector i.

= The quantity of *i'*s product demanded per year.

Note: The units of BQDi are goods-baskets/year for the goods sector, tonne/year for the black sector and work-years/year for the pop sector.

*BXbg* is set to 0.0 since we assume the goods sector would not use any black market product. Also, *BXww* is set to 0.0 since the flow of money completely outside the region is meaningless as far as Athena’s economic model is concerned.

Since actor's revenues and expenditures are computed by Athena from the actor definitions and their strategies, this row and column cannot be edited. Section 10.3 goes into detail as to how those cells are computed once the scenario is locked.

Each time any input in the matrix is changed, the cells on the SAM tab are recalculated and the new values displayed. The following equations are true by definition:

#### Other Inputs

Other inputs to the SAM are summarized in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| **Cell Label** | **Cell Name** | **Meaning** | **Units** |
| Black mkt Feedstock Price | *PFwb* | The price of black market feedstock used to make the final product | $/tonne |
| Feedstock per Unit Product | *AFwb* | The number of units of feedstock needed to make one unit of final product | feedstock units/tonne |
| Max Feedstock Avail. | *MFwb* | The maximum amount of feedstock available | tonnes/year |
| Black Market Capacity | *CAPblack* | The maximum capacity of the black market | tonnes/year |
| Base Consumers | *BaseConsumers* | The number of consumers upon which the SAM data is based | number of people |
| Base Unemployment Rate | *BaseUR* | The base unemployment rate to which the economy is calibrated | % |
| Base Subsisters | *BaseSubsisters* | The number of people engaged in subsistence agriculture in the base case | number of people |
| Subsistence Wage | *BaseSubWage* | The wage someone engaged in subsistence agriculture is considered to have earned for the purpose of GDP calculation | $/year per person |
| Remittances | *BREM* | The amount of remittances flowing into the local economy | $/year |
| REM Change Rate | *REMChangeRate* | The percentage per annum that remittances change up or down | %/year |

These values affect some of the outputs from the SAM and some are used directly by the CGE. The following sections detail the outputs as they are computed from the SAM Matrix and other inputs.

#### Balancing Flows Matrix (T-matrix)

After the scenario is locked the values in the balancing flows matrix are computed from the input flows matrix and other inputs. The balancing flows matrix (aka the T-matrix):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **goods** | **black** | **pop** | **actors** | **region** | **world** |
| **goods** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **black** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **pop** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **actors** | 0.0 |  | 0.0 | 0.0 | 0.0 |  |
| **region** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |  |
| **world** |  |  |  | 0.0 | 0.0 | 0.0 |

where

= Black market profits to the actors sector if actors receive them

= , ensures the goods sector balances

= Black market feedstock price plus black market profits if no actors receive them

= , ensures the pop sector balances

=

=

#### SAM Outputs

As values in the input flows matrix and other inputs are made or modified by the user and the scenario is locked, the shape of the economy and other salient outputs are computed. The resultant shape parameters are displayed in a spreadsheet-like tableau:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **goods** | **black** | **pop** | **actors** | **region** | **world** |
| **goods** | *fgg* | *Agb* | *fgp* | *fga* | *fgr* | *N/A* |
| **black** | *fbg* | *Abb* | *fbp* | *fba* | *fbr* | *N/A* |
| **pop** | *fpg* | *Apb* | *fpp* | *fpa* | *fpr* | *N/A* |
| **actors** | *tag* | *tab* | *tap* | *faa* | *far* | *N/A* |
| **region** | *trg* | *trb* | *trp* | *fra* | *frr* | *N/A* |
| **world** | *twg* | *twb* | *twp* | *fwa* | *fwr* | *N/A* |

where

= Fraction of total revenue (Cobb-Douglas coefficients where relevant) for payments from sector *j* to sector *i*.

= Tax-like rate for payments from sector *j* to sector *i*.

= Leontief coefficient for payments from sector *j* to sector *i*.

These parameters are computed to represent the particular shape of the economy. The CGE is initialized with these values which then determine how much money flows from one sector to another based upon certain algorithms. The following equations are true by definition:

where

where

where and

where the notation *XT* indicates the summation of the *BX* (input flows) and *T* (balancing flows) matrices.

*fbg* is always 0.0 since, by definition, the goods sector does not use any black sector product. There is no need for shape parameters for expenditures by the world sector, since the money flows from the world sector to other sectors are treated separately using the *BEXPORTSi* described below. A complete discussion of the *fij*, *tij*, and *Aij* coefficients and their use by the CGE is beyond the scope of this document.[[91]](#footnote-92)

The remaining SAM outputs are described in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| **Cell Label** | **Cell Name** | **Meaning** | **Units** |
| Foreign Aid to Actors | *FAA* | Money from the world sector used as income by the actors sector | $/year |
| Foreign Aid to Region | *FAR* | Money from the world sector to the region sector | $/year |
| Exports of Goods | *BEXPORTSg* | The amount of exports from the goods sector to the world sector | GBasket/yr |
| Exports from Black Market | *BEXPORTSb* | The amount of exports from the black market sector to the world sector | tonnes/yr |
| Exported jobs | *BEXPORTSp* | The number of jobs exported from the pop sector to the world sector | work-yr/yr |
| Base GDP | *BaseGDP* | The base GDP to which the economy is calibrated, this results in the computation of the base wage. |  |
| Per cap. Demand for Goods | *BAgp* | The average per capita demand for goods | GBasket/yr |
| Base Sub. Agriculture | *BaseSA* | The product of the number of subsisters and the subsistence wage | $/year |
| Black Market Profit | *BNRb* | The net revenue in the black market sector | $/year |
| Black Feedstock | *FEEDb* | The total cost of feedstock for the black market | $/year |

The cells in this table have the following relationships to other inputs on the SAM tab:

### Computing the Actors Sector

The **actors** sector expenditures and revenues are computed solely from the actors’ definitions and their tactics and strategies. The **actors** sector revenue is determined from the actor definitions alone, but only for actors defined as INCOME actors. It is important to note that when discussing the sector in the economy for actors (which is an aggregation of all actors’ expenditures and revenues), this document uses a bold font. When discussing individual actors or actors together, this document uses a regular font.

The **actors** sector expenditures are determined by how actors choose to employ their tactics and strategies. This includes the SPEND tactic which can be used by BUDGET actors to infuse money into the economy and by INCOME actors to simulate the expenditure of money for overhead costs. Overhead costs would be those costs not explicitly associated with any of the other tactics defined in Athena.

The **actors** sector income is determined from actor definitions. Each INCOME actor has the following attributes related to income:

|  |  |
| --- | --- |
| **Name** | **Description** |
| *income, GOODS* | Income in dollars per week from the **goods** sector. |
| *income, BLACK NR* | Shares of income from the net revenue of the **black** market. |
| *income, BLACK Tax* | Income in dollars per week from the **black** sector, computed as a tax on revenue. |
| *income, POP* | Income in dollars per week from the **pop** sector. |
| *income, Graft* | Income in dollars per week in graft, money skimmed from foreign aid to the **region** sector. |
| *income, WORLD* | Income in dollars per week in foreign aid from the **world**. |
| *cash-on-hand* | Cash on hand, in dollars: the money the actor has immediately available to fund tactics. Unspent cash is carried over to the next week. |
| *cash-reserve* | Cash reserve, in dollars. The actor can reserve funds for later use. |

Each BUDGET actor has a set amount of money specified that is available to spend each week. The money does not come from within the economy but rather from somewhere else (such as from a government engaged in providing monetary aid to a region).

Note that if the economic model is disabled, then the income an INCOME actor has is simply the sum of the incomes specified without regard to whether the money is available within the economic model.

#### Computing Actor’s Expenditures

Athena computes base case expenditures for the SAM when the scenario is locked using the base case revenue. When simulated time advances, expenditures are computed for each tactic that costs money. Athena has a family of parameters that govern to which sectors actors’ expenses are allocated depending on tactic.[[92]](#footnote-93) When the scenario is locked, the base case expenditures in the SAM are computed from tactics that are set to execute on lock. Then the CGE is initialized with these base case expenditures.

Athena’s “cash” module tracks these expenditures and allocates them to the designated sector as defined by the parameters described above and in the associated footnote. When the economic model executes, these expenditures are then updated in the CGE:

for

where

∑*α* = Indicates that the sum is over all actors.

= Sum of all expenditures by all actors made to sector *i* at the current timestep.

*sharesi* = The number of shares allocated for sector *i* given the expenditure class[[93]](#footnote-94).

*totshares* = Total number of shares given an expenditure class.

*fraci* = The fraction of money in the expenditure class allocated to sector *i.*

*dollarsα* = The total amount of money spent by an actor on an expenditure class.

The constant 52.0 appears in the equation because actors expend cash on a week to week basis, but the CGE accounts for money in per annum quantities.

As Athena advances simulated time, actors will expend money as their strategies are executed and expenditures in the actors sector are computed again and again using the same equations.

#### Computing Actor’s Income

In scenario **PREP**, income for an actor is specified as some amount of money from each sector or, in the case of black market net revenues, shares of the total amount. Those amounts of income should be input with the understanding that it is based on an economy with a *BaseConsumers* number of consumers. In Athena it’s possible the actual number of consumers is different from *BaseConsumers* when the scenario is locked. Because of this, total base revenue from a sector is scaled to the actual number of consumers and, thus, total revenue for the sector as a whole is scaled:

for

for

where

= Base rate of money from sector i to the actors sector.

= Sum over all actors income from sector i.

*BaseConsumers* = The base number of consumers from SAM inputs.

*ActualConsumers* = The actual number of consumers as computed by Athena.

= Base revenue in the **actors** sector.

*BNRB* = Base net revenue in the **black** sector.

If no actor is designated to receive any income from shares of the black market net revenue then all black market net revenue is assumed to go the world sector, *BXwb.*

Once base rates and revenues are determined, each actor has tax-like rates computed for use in determining income from each sector as simulated time advances. These tax-like rates represent the shape of the revenue on an actor-by-actor basis which, when summed together, represent the shape of the **actors** sector revenue as a whole. Thus, as revenue streams in all other sectors fluctuate (for whatever reason), the revenue stream in the **actors** sector fluctuates with them. The actor-by-actor tax-like rates are computed using the following equations (the constant 52.0 is for converting from weeks to years):

for

where

= tax-like rate for actor α and sector *i*.

*Iαi* = Weekly income of actor α from sector *i*.

*Iαgraft* = Weekly income of actor α from graft.

*BPi* = Base price in sector *i*.

*BQDi* = Base quantity demanded in sector *i*.

= Base revenue in the **world** sector.

*FAR* = Foreign aid to the region.

= cut of black market net revenue for actor α expressed as a rate.

= Shares of black market net revenue for actor α.

= Total shares of black market net revenue.

= Total net revenue of the black market.

Then, as time advances and revenue in the sectors changes, income to each actor from each sector is determined:

for

where

= tax-like rate for actor α and sector *i*.

= Total revenue in sector *i*.

*Iαi* = Weekly income of actor α from sector *i*.

*Iαgraft* = Weekly income of actor α from graft.

*cutαb* = Fraction of black market net revenue to actor α.

*NRB*= Black market net revenue.

*Iαcut* = Weekly income of actor α from black market net revenue.

### Text Notation

The SAM and CGE are implemented as *cell models*, a pseudo-spreadsheet in which every cell has a name rather than a row and column. The cell model is contained in a text file, and so we also have a plain text notation for each of the mathematical symbols given above:

= X.i.j

= EXP.j

= REV.i

= P.i

= QD.i

When we are referring to a particular sector or sectors, the sector names are spelled out in full in the plain text form:

= X.goods.pop

### Shape vs. Size

For a complete discussion of shape vs. size see “Athena’s Computable General Equilibrium Model” by Chamberlain, Duquette and Kahovec included with the Athena documentation. For reference, a brief description is provided here.

We distinguish between the *size* of the economy, which can be roughly thought of as total revenues, and the *shape* of the economy, or the proportion of revenues across the sectors. The size of the economy is driven by consumption: increase the number of consumers, or the amount consumed by each, and the economy must increase in size. Decrease the amount of consumption, and the economy must shrink. But as the economy increases and decreases in size, its basic shape remains the same, because the basic industries and technologies in use remain the same.

The size is driven by the number of consumers and how much they to spend, which affects their per capita consumption. The number of consumers is given to the Economic model by the Demographic model. The shape of the economy is determined by providing base case data in the SAM; see Section 10.2.

### Production Functions

The shape of the economy is largely determined by the technologies used by the sectors to produce their products; and these technologies are described by *production functions*.

A production function determines how much product a sector consumes (from its own and other sectors) to produce one unit of its own product. The **goods** and **pop** sectors are modeled using Cobb-Douglas production and utility functions, which implies that, when ingredient quantities are chosen to minimize costs, the sector will spend a fixed proportion of its money on each of the three sectors, i.e., 0.5 on **goods**, 0.4 on **pop**, and 0.1 on **black**. As prices change, the Cobb-Douglas function allows the sectors to trade off their requirements for the product of one sector for the product of another.

These fixed proportions are called the Cobb-Douglas parameters, and are denoted .

The **black** sector is modeled using the Leontief production function, which assumes that input requirements are strictly proportional to the output quantity for every ingredient in an attempt to maximize profit. This is particularly useful for the black market since the price in that sector is exogenous (determined completely outside the economy being modeled).

### Calibrating the CGE

The CGE has three views of the economy: Unconstrained, Capacity Constrained and Capacity Constrained with Security Factors. The unconstrained economy is one where there is exactly enough production capacity to completely fill all demand. The capacity constrained economy takes into account that the capacity to manufacture goods is limited. The capacity constrained with security factors economy takes into account consumer security and labor security factors; these further reduce the size of the economy because some people are either too afraid to go out and shop or go to work. Athena takes the base GDP and base unemployment rate inputs and calibrates the CGE to the capacity constrained case arriving at a wage on a per capita consumption rate that will match those inputs. The shape parameters, money flows and other inputs from the base SAM are used during this calibration step:

* XT.i.j, the money flows used to determine the shape parameters.
* f.i.j, the Cobb-Douglas parameters for all sectors.
* A.i.j, the Leontief coefficients for the black sector
* t.i.j, the tax-like rates in the **goods**, **black** and **pop** sectors.
* BREM, the base remittances in $/year.
* PF.world.black, the black market feedstock price.
* AF.world.black, the black market feedstock per unit of product.
* MF.world.black, the maximum amount of black market feedstock available.
* BaseConsumers, the number of consumers upon which the BX.i.j are based.
* BaseUR, the unemployment rate at time 0.
* BaseGDP, the GDP based upon base flows of money in all sectors.
* BaseSubWage, the wages associated with subsistence agriculture for the purpose of computing gross domestic product. This is probably near the poverty level.

#### Set the Base Consumption

Next we need to provide the link between shape of the economy and its size. We do this by setting the base number of consumers.

The base case per capita demand for goods is determined from XT.goods.pop (which includes remittances), BP.goods and BaseConsumers. Adjusting any of these inputs in the SAM will cause the base case demand to go up or down. The actual demand, however, is determined when the scenario is locked. It is at that time that the actors sector is computed and the CGE is calibrated using the BaseGDP and BaseUR along with the true demographics which include the actual number of consumers and workers.

### Scenario Inputs

The following input values are plugged into the CGE by the ECON model as part of the base scenario. All these values come from the SAM:

|  |  |  |
| --- | --- | --- |
| Cell | Source | Description |
| BaseConsumers | DEMOG | The total number of consumers in the local region at time 0. This number is used to size the economy, and calibrates a number of constants used subsequently. |
| BP.i | ECON | Base prices for sectors that have product: **goods**, **pop** and **black**. |
| TX.i.j | ECON | Base case money flows from sector *j* to sector *i*. This includes the base values computed in the **actors** sector. |
| BQD.i.j | ECON | Base case quantities demanded in the **goods**, **pop** and **black** sectors. |
| f.i.j | ECON | Cobb-Douglas coefficients for the pertinent sectors. |
| t.i.j | ECON | Tax-like rates for the pertinent sectors. |
| A.i.j | ECON | Leontief coefficients for the pertinent sectors. |
| k.goods | ECON | After tax fraction of sales in the **goods** sector. |
| k.pop | ECON | After tax fraction of sales in the **pop** sector. |
| FAA | ECON | Amount of foreign aid to the **actors** sector. |
| FAR | ECON | Amount of foreign aid to the **region** sector. |
| BEXPORTS.i | ECON | Amount of exports from the **goods**, **pop** and **black** sectors. |
| BA.goods.pop | ECON | Base case per capita demand for goods. |
| Bgraft | ECON | Base case graft, the amount skimmed off FAR by the actors sector. |
| BaseUR | ECON | Base unemployment rate. |
| BREM | ECON | Base case remittances. |

### Run-time Inputs

The following input values are plugged into the CGE by the ECON model at each "tock", that is, at each update of the CGE as time passes (the “::” notation is a software construct, it can be thought of, for purposes of this document, as part of the cell name):

| **Cell** | **Source** | **Description** |
| --- | --- | --- |
| In::Consumers | DEMOG | The number of consumers in the local region at the current time. This number drives consumption, which determines the size of the unconstrained economy. |
| In::CAP.goods | ECON, Ground | The maximum capacity of goods production. |
| In::LF | DEMOG | The number of workers in the local economy at the current time. This number determines the production constraint for the **pop** sector, and also drives the computation of unemployment. |
| In::GU | DEMOG | The number of workers who are geographically unemployed. See Section 9.7.3. |
| In::LSF | ECON,  Ground | The Labor Security Factor, a number from 0.0 to 1.0. The LSF decreases with neighborhood security; (1-LSF) is the fraction of the work force that stays home from work out of fear, or to protect their property or families. Thus, this number also affects the production constraint for the **pop** sector. |
| In::CSF | ECON,  Ground | The Consumer Security Factor, a number from 0.0 to 1.0. The CSF decreases with neighborhood security; (1-CSF) is the fraction of consumers that stay home out of fear for their lives. Thus, this number also affects the effective number of consumers from the **pop** sector. |
| In::Subsisters | DEMOG | The number of people actually engaged in subsistence agriculture. These people are not part of the **pop** sector, since they do not make real wages. They do, however, contribute to GDP based upon the subsistence agriculture wage, BaseSubWage. |
| In::REM | ECON | The amount of remittances. If the remittance change rate is set to 0.0, this does not change during runtime. |
| In::X.world.actors,  In::X.region.actors,  In::X.goods.actors,  In::X.black.actors,  In::X.pop.actors | ECON | Based upon each actors strategies and overhead spending, these are the aggregate money flows from the **actors** sector to the various other sectors. |

### Outputs

The CGE produces the following output values.

| **Cell** | **Used By** | **Description** |
| --- | --- | --- |
| Out::P.*i* | Display | The price of one unit of sector *i*, in dollars. |
| Out::QS.*i* | Display | The quantity supplied for sector i, i.e., the number of units produced. |
| Out::REV.*i* | Display | The revenue of sector *i*, that is, Out::P*i*\*Out::QS.*i* in dollars. |
| Out::EXP.*i* | Display | The expense of sector *i*, that is, the dollars spent on the ingredients for the product of sector *i*. |
| Out::QD.*i.j* | Display | The quantity of sector *j*'s output purchased by sector *i*. |
| Out::X.*i.j* | Display | The dollars spent by sector *i* on sector *j'*s output. |
| Out::LATENTDEMAND.*i* | Display | The additional quantity of sector *i'*s product that the economy would purchase if it could be produced. (**goods**, **pop** and **black** only) |
| Out::IDLECAP.*i* | Display | The additional quantity of sector *i'*s product that the sector could produce if only there were demand for it. (**goods**, **pop** and **black** only) |
| Out::Unemployment | Display | The number of workers who are currently unemployed, including normal turbulence. |
| Out::UR | DEMOG | The unemployment rate, as a percentage. |
| Out::GDP | Display | The Gross Domestic Product, in dollars: the total value of final goods and services in the regional economy. |
| Out::CPI | Display | The Consumer Price Index, which measures changes in buying power since the start of the simulation. |
| Out::DGDP | Display | The Deflated Gross Domestic Product, i.e., the GDP divided by the CPI. This is the current size of the economy, in "time 0" dollars. |

### Ways to Affect the Economy

The Economy is affected at each economic tock by the inputs listed in Section 10.9. Consequently, the following things taking place in Athena as a whole will affect the economy:

* Civilian casualties can decrease the number of consumers and workers.
* Population growth or reduction can increase or decrease the number of consumers and workers.
* Remittance growth or reduction over time causes the amount of money the population spends to increase or decrease changing the size of the economy.
* Subsistence population, when displaced from their land, willy-nilly become consumers; they might not be able to contribute to the work force, depending on the manner of their displacement.
* Geographic unemployment, the unwillingness of workers to travel to where the jobs are, can reduce the size of the economy.
* When a civilian group's security in a neighborhood decreases by too much, workers stay home out of fear, thus reducing the effective size of the work force. This is measured by the Labor Security Factor (LSF).
* When a civilian group's security in a neighborhood decreases by too much, consumers stay home out of fear, thus reducing the number of people buying goods. This is measured by the Consumer Security Factor (CSF).
* Actors may build new goods production infrastructure increasing the capacity of the goods sector.
* Actors may maintain or fail to maintain goods production infrastructure causing the capacity of the goods sector to increase or decrease.
* Actors strategies change which may, in turn, change how their money is spent on the other sectors. Actors may also become bankrupt which means they cannot afford strategies that cost money.

### Ways the Economy Affects Athena

There are many ways in which the economy *should* affect Athena; at present, the only implemented effects are:

* Consumption of goods by the civilian population.
* Unemployment on the civilian population. This is done in the Demographic model. See Section 9.
* Income to actors may be affected by changes in revenue in the sectors from which they draw income.
* Income to actors may be affected if black market profits can be diminished.

## GOODS production infrastructure

Athena contains a model of goods production infrastructure in the form of “plants”. A plant is the basic unit of production, which can produce a number of goods baskets each week. Aggregated together, plants form the total capacity of the goods sector. Actors can own goods production infrastructure as a share of the total capacity of the goods sector. They can also build new infrastructure and be responsible for the maintenance of existing infrastructure. If plants that need to be maintained aren’t, then the capacity of the goods sector can be negatively impacted. Likewise, building new plants can cause the capacity of the goods sector to be expanded. This section describes the goods production infrastructure in detail.

### Initial Laydown of GOODS Production Infrastructure

Athena determines how many plants are needed throughout the play box based upon a number of factors that are described in detail below. First the number of plants per neighborhood is determined and then, in each neighborhood, the number owned by actors based on shares of ownership is determined. This way, the analyst does not need to know how many plants actors should own in order to have the total production capacity at initialization match the initial production capacity in the economic model. Athena will determine the exact number of plants to own by actor and neighborhood for the analyst. After initialization, the total production capacity of all plants matches the production capacity of the goods sector as calibrated by the economic model.

#### Allocated to Neighborhoods

Neighborhoods have a production capacity factor (PCF), which is a number greater than or equal to 0.0 but is nominally 1.0. The PCF is specified on a neighborhood-by-neighborhood basis and allows the analyst a way to indicate that some neighborhoods should be allocated more goods production infrastructure than others. The total number of plants in a neighborhood at time 0 is determined from the PCF, the initial capacity percentage of the entire economy[[94]](#footnote-95), the number of goods baskets produced per plant[[95]](#footnote-96), and the number of consumers:

where

*plantsn* = The number of plants allocated to neighborhood *n*.

*PCFn* = The production capacity factor of neighborhood *n.*

*initCapPct* = The initial capacity of the entire economy, this should be less than 100% if the economy is degraded at time 0

= The calibrated demand for goods in goodsBKTs from the econ model

= The number of consumers in neighborhood *n*.

= The total number of consumers in the local economy.

Note that if a neighborhood consists entirely of groups engaged in subsistence agriculture, there are no consumers and, hence, no goods production infrastructure allocated to that neighborhood.

#### Allocated to Actors

Plant ownership in each neighborhood is then further broken down by shares allocated to each actor:

where

*plantsna* = The number of plants allocated to actor *a* in neighborhood *n*.

*plantsn* = The number of plants allocated to neighborhood *n.*

*sharesa* = The number of shares of plant ownership actor *a* gets in neighborhood *n*.

= The total number of shares of plant ownership in neighborhood *n*.

The *ceil* function is used to round up to the nearest integer since Athena does not model fractional plants. Because of the rounding up, the initial average repair level[[96]](#footnote-97), , of plants owned by actor *a* in neighborhood *n* is then adjusted so the goods production capacity in each neighborhood when summed matches the initial output capacity as determined by the economic model. For neighborhoods that have the ability to produce goods but no shares of ownership are explicitly input, the SYSTEM agent will own all plants and they will require no maintenance.

### Degradation and Maintenance of Infrastructure

Any plants owned by an actor will degrade unless the actor that owns them is not automatically maintaining them. Automatically maintained plants will continue to have the same goods production capacity as they do at time 0 so far as the SYSTEM agent does not damage them. For other actors, plants will degrade each simulated week by a parameter-controlled amount:

Where is the degradation amount and *lifetime* is the goods production plant lifetime in weeks[[97]](#footnote-98). Thus, each week the average repair level, , of the plants owned by an actor degrade to . This repair level is then used as a multiplier to the production capacity of the plants owned by an actor in a neighborhood resulting in a reduction of their capacity. The repair level will remain at this degraded amount until the next simulated week unless the actor allocates resources to maintain owned infrastructure.

Maintenance of infrastructure is accomplished by having an actor execute MAINTAIN tactics as part of his overall strategy. This tactic serves to increase the average repair level of goods production infrastructure thereby maintaining a certain level of goods production capacity. The most the average repair level of goods plants can be increased in a given week is controlled by a parameter:

where is the maximum repair amount and *repairtime* is the amount of time in weeks it takes to bring a plants from an average repair level of 0.0 to 1.0[[98]](#footnote-99). This maximum amount of repair bounds the cost of doing repairs to whatever amount of money it would take to do the repairs on all plants owned by an actor in a neighborhood. Thus, the actual amount of repair is the lesser of 1) the maximum amount of repair or 2) the amount of repair that the money allocated to repair will bring:

where

= The actual amount of repair completed.

= The maximum amount of repair possible.

*cost* = The cost of the maximum amount of repair to one plant[[99]](#footnote-100).

*dollars* = The amount of money allocated to repair in a given week.

*nplantsna* = The number of plants owned by actor *a* in neighborhood *n*.

Thus the new repair level after all maintenance is completed is .

### Building New Infrastructure

Actors can also allocate money to build new goods production infrastructure, which increases the production capacity of the **goods** sector. This is accomplished through the use of the BUILD tactic as part of an actors overall strategy. The time it takes to complete the construction of one plant to become fully operational takes a parameter-controlled number of weeks[[100]](#footnote-101). Thus the maximum amount of construction that can take place on any given plant in one week is given by:

The maximum cost to build one new plant is then bound by the maximum amount of construction that can take place:

Like maintenance, construction of new infrastructure must have resources allocated to it and the amount of those resources may be less than the amount that it would take to complete the maximum amount of construction possible. Thus, the actual amount of construction that takes place in a given week is the lesser of 1) the maximum amount of construction possible 2) the amount of construction that the allocated amount of money could possible bring:

where

= The actual amount of construction completed.

= The maximum amount of construction possible.

*costmax* = The cost of the maximum amount of construction of one plant[[101]](#footnote-102).

*dollars* = The amount of money allocated per plant.

The number of plants is not found in any of the equations above. This is because Athena will continue to build as many plants as it can given resources allocated to it working on the most complete plants first. If an actor has enough resources to do the maximum amount of construction on *n* plants every week until construction is completed, then those *n* plants will begin to produce goods at the same time. However, resources may be constrained, thus, the number of plants worked on may vary from week to week. In other words, plants are constructed on a “best efforts” basis. This means that plants may begin to produce goods at different times depending on how resources end up being allocated.

### Damaging Existing Infrastructure

The average repair level of GOODS production infrastructure owned by any actor in any neighborhood can be set by the SYSTEM agent. In this way, the capacity of plants can be adjusted either up or down. It was envisioned that this would likely be used to simulate the sudden damage of plants due to manmade or natural phenomenon. Thus, the SYSTEM agent has a DAMAGE tactic that allows the analyst to trivially set the average repair level, ρ, of plants owned by an actor in a neighborhood.

## Communications Infrastructure

Athena contains a simple model of broadcast communications infrastructure, as an adjunct to the information operations model (Section 13). The model conflates all forms of broadcast media, with an emphasis on coverage areas; in short, by choosing a broadcast outlet, an actor chooses an audience. This section describes the communications infrastructure model in detail.

### Communications Asset Packages

A *communications asset package* (CAP), is a collection of hardware, personnel, and other infrastructure that in some way affects communications to and among the civilian population. Examples of CAPs include newspapers, television stations, the cellular telephone network, the Internet, websites of various kinds, and so forth. We focus strictly on the broadcast usage mode, and (except in the simplest possible way) we ignore damage to, repair and maintenance of, and investment in, the actual infrastructure. This allows us to take a simple view of CAPs for the near term.

### CAP Capacity

Each CAP *k* has a capacity value, denoted , where . This indicates the CAP's degree of repair. When , the CAP is fully functional; when

, the CAP is non-functional.

defaults to 1.0, and is changed only at the analyst's request. In particular, the analyst may write an **EXECUTIVE** tactic to change a CAP's capacity using the **CAP:CAPACITY** order.[[102]](#footnote-103) This is a proxy for a more detailed infrastructure model.

In particular, if an actor wishes to shut down a CAP, and is (in the analyst’s view) in a position to do so, the actor may use an **EXECUTIVE** tactic to reduce the CAP’s capacity to 0.0.

### CAP Neighborhood Coverage

Every CAP has a coverage area: the neighborhoods in which it can be heard (including partial neighborhoods). If the CAP is a web site, this would be every neighborhood with internet access; if the CAP is a traditional television station, this would be every neighborhood in range of the station's transmitter.

Thus, we define to be the fraction of the people in neighborhood *n* who can in theory receive messages from CAP *k*; thus, the values for for CAP *k* define *k*’s coverage area. The CAPs that are likely to be of interest cover many neighborhoods, so the value of this metric will usually be 1.0. However, numbers less than 1.0 may be appropriate for large neighborhoods or for neighborhoods on the fringe of a CAP’s coverage area.

is a scenario input.

### CAP Group Penetration

Just because a CAP's signal can reach everyone in a neighborhood doesn't mean that everyone in the neighborhood pays attention. Thus, we define to be CAP *k*’s market penetration into group *g*, i.e., the fraction of the people in group *g* that will watch, listen to, or read messages that are offered by CAP *k*, given the opportunity.

is a scenario input.

### CAP Audience

The primary output of the CAP model is CAP *k*’s audience, denoted : the fraction of group *g* that receives and understands messages transmitted by CAP *k* (independent of the message content). is computed as follows:

In other words, if the CAP is in good repair () and has complete coverage of a given neighborhood (), then the CAP's audience in the neighborhood is simply that fraction of each group in the neighborhood that pays attention to that CAP. If the neighborhood coverage is less than one, the audience decreases accordingly; and if the CAP's capacity is reduced, the audience decreases accordingly.

**Note:** in earlier memos, and in the Athena 4 code, the variable , called the "CAP coverage", was used instead. This caused confusion with the neighborhood coverage, , and so the term was adopted late in the development cycle.

### Broadcast Cost

Each CAP *k* has a broadcast cost, , in dollars per message per week. An actor must pay this cost in order to send a message via CAP *k* during a given week.[[103]](#footnote-104)

### CAP Ownership and Access Control

Each CAP is owned by some actor in the scenario, and of course the owning actor may use his CAPs to broadcast his information operations campaigns (Section 13) as he pleases. Often, however (as when a CAP is owned by a party outside the region of interest), an actor will need to make use of CAPs belonging to others. Thus, we need a model of access control.

The owner of a broadcast CAP can choose to grant access to other actors, or alternatively to block access. In practice there can be many forces acting on the owner of the CAP that will determine whether or not he chooses to grant access, or is forced to grant access. To take one example, consider a small town newspaper. Suppose that a warlord takes control of the entire town. Can the warlord force the CAP owner to grant him access? Possibly; or possibly the CAP owner might choose to destroy the CAP’s infrastructure. Or, possibly, the warlord might seize the CAP and its infrastructure, thus effectively becoming the new owner.

Rather than trying to model relationships between actors, we model their behavior:

* The warlord can use an **EXECUTIVE** tactic with the **CAP:UPDATE** order to make himself the owner of the CAP. As owner, he automatically has all of the access he needs.
* The CAP owner can use the **GRANT** tactic to give other actors access to a given CAP or set of CAPs.
* Whether these tactics are executed will depend on the attached conditions. Thus, the CAP owner might choose to grant access to the warlord if and only if the warlord controls the neighborhood containing the CAP.[[104]](#footnote-105)

See the *Athena Analyst's Guide* and on-line help for information on tactics and orders.

## Information Operations

Actors can use *information operations campaigns* to affect civilian attitudes. In particular:

* An actor may send an *information operations message* (IOM) via a *communications asset package* (CAP).
* The IOM will be heard by the audience of the chosen CAP (see Section 12.5).
* The IOM consists of:
  + A description: a brief English-language statement of the message, as an aid to the analyst.
  + A *semantic hook*, by which it appeals to the groups in the CAP's audience.
  + One or more *payloads*, by which it affects specific attitudes.
* The IOM results in attitude inputs to URAM. These inputs depend on:
  + The audience.
  + The *resonance* of the semantic hook with the audience.
  + The relationship of the audience with the actor perceived to be sending the IOM.
  + The payloads.

An Athena IOM is a highly abstract creature. A real-world IOM can be a smoothly produced piece of work with significant informational and emotional content. Athena is not able to parse and understand even the informational content, let alone the choice of colors, graphics, music, timing, and so forth in a typical radio, television or magazine spot. It is up to the analyst to analyze the message and its effects into terms that Athena can do something with.

The following subsections will define the semantic hook and the payloads, and how the actual effect is determined. Finally, Section TBD gives a complete example. The example presumes the material that precedes it; nevertheless, it might be useful to refer to it before proceeding.

### Semantic Hooks, Congruence, and Resonance

An IOM captures an audience by appealing to the audience's beliefs. We use the Theory of Homophily[[105]](#footnote-106) to assert that the group will accept the payload to the extent that there is a high degree of resonance between the group’s beliefs and the semantic hook. Thus, a semantic hook is just a set of positions on one or more belief system topics, using the same scale as is used when defining a group or actor’s position on a topic. The topics used in a hook may include those used for computing group-to-group and group-to-actor affinity; however, it is also possible to define additional topics purely for use in semantic hooks. (See Section 3.1 and the discussion of the Mars Affinity Model in the *Mars Analyst's Guide* for more information about belief systems and affinity.)

A semantic hook may be more or less congruent with a given civilian group’s belief system, as described in the *Mars Analyst's Guide*. Looking only at the topics included in the hook, we compute the civilian group’s affinity with the beliefs expressed in the hook. We call this affinity-like measure the *congruence* of the hook with the group’s belief system.

There is one additional input needed to compute the congruence. In its belief system, each group and actor has its *entity commonality fraction*: its degree of participation in the dominant culture in the playbox. The semantic hook’s entity commonality derives from the perceived source of the IOM, as discussed in Section 13.3.

We denote the congruence of message *m*'s hook with group *g'*s belief system This is later used to compute resonance, as described in Section 13.3.2.

### Payloads

Every IOM has one or more payloads: attitude effects on the receiving civilians. The payloads are specified by the analyst, and Athena determines (based on the choice of CAP, the perceived source of the IOM, the hook, the payloads themselves, and the recipient) the actual magnitude of the effect.

At present, there are four payload types; each identifies a particular set of the recipient’s attitudes to modify, and a nominal magnitude by which to modify them. When the message is broadcast, the payloads are applied to the civilian groups *f* covered by the CAP.

Cooperation Payload (COOP)

The COOP payload affects civilian group *f’*s cooperation with some specific force group *g*.

Horizontal Relationship Payload (HREL)

The HREL payload affects civilian group *f’*s horizontal relationship with some specific group *g*, of any group type.

Satisfaction Payload (SAT)

The SAT payload affects civilian group *f’*s satisfaction with some specific concern *c*.

Vertical Relationship Payload (VREL)

The VREL payload affects civilian group *f’*s vertical relationship with some specific actor *a*.

Each payload has a nominal magnitude *mag*, stated using one of Athena’s usual magnitude symbols, and representing a percentage change in the attitude curve from its current level toward the relevant extreme, i.e., a 5.0 point change moves a satisfaction level 5% of the way toward +100.0.

It might seem odd that the analyst must enter the magnitude of the change. As noted in Section 13, however, a real-world IOM has informational and emotional content that Athena simply cannot assess. Establishing the magnitude for each payload is part of the analysis the analyst must do while defining the IOM. Note, however, that the payload magnitude is only a nominal magnitude. It must be adjusted by the CAP coverage, the resonance of the semantic hook, and the regard of the recipient group for the perceived sender, as described in Section 13.3, before it is given to URAM.

### Broadcasting an IOM

Actor broadcast IOMs using the **BROADCAST** tactic, which has the following parameters:

* The CAP by which the IOM will be broadcast.
* The Perceived Source.
* The IOM to broadcast.
* The production cost.

The CAP and IOM have been described above.

**Perceived Source.** The Perceived Source is the actor who will be seen as the source of the IOM, the actor who is trying to influence public opinion. This might be the same as the True Source, the actor who executes the BROADCAST tactic; it might be some other actor; or there might be no Perceived Source, as when an IOM is seen as straight reporting.

The True Source might very well wish to hide their identity, and make it look like the IOM is coming from some other source. Athena does not attempt to adjudicate this. Rather, it is up to the analyst to decide which actor (if any) will be perceived as the source, whether this perception accords with the True Source’s wishes or not.

**Production Cost.** The cost of broadcasting an IOM is assumed to have two components: the production cost and the transmission cost. The transmission cost is a function of the chosen CAP; the CAP’s cost per message per week is part of the CAP’s definition. However, the cost of producing the message for transmission is a function both of the message itself and of the medium by which it is broadcast, and can’t be defined as simply a part of either CAP or the IOM. Thus, we allow the analyst to enter it here.

Note that the magnitude of the production cost affects only the resources consumed by the broadcast, not how effective the broadcast is. That is determined by the payload magnitudes.

Because different media have different production costs, and different CAPs may have different media types, it may be necessary to define multiple versions of an IOM that is intended for use in multiple media. Alternatively, the analyst can set the production costs to $0 and assume that production is part of the transmission cost.

#### Access to the Broadcast CAP

An actor may always broadcast using his own CAPs; however, CAP ownership can change and access to CAPs belonging to other actors can be granted or revoked as the result of strategy execution, and hence the IOMs cannot actually be broadcast until after strategy execution has concluded. At that time Athena broadcasts all IOMs for which the sending actor has access to the chosen CAP. If an actor's IOM cannot be broadcast on the chosen CAP then the money he spent on it is refunded.

#### IOM Acceptability

When IOM *m* is successfully broadcast, we next determine the acceptability of the IOM to each group *g* in the audience of the CAP used to broadcast the IOM, that is, each group *g* for which , where

where

*m* = The IOM

*g* = A group in the CAP's audience

*a* = The actor perceived to have sent the IOM

= The resonance of the IOM's semantic hook with group *g*

= The regard ofgroup *g* for actor *a*.

**Resonance.** The purpose of the IOM's semantic hook is to catch the civilians in the audience; the greater the value of , the more civilians are caught. However, we can't use congruence directly, as (being an affinity value) it ranges from −1.0 to +1.0. While it is possible that negative values of congruence should negate the attitude effects of the payloads, this effect should be muted. Hence, we define the resonance of the message as

where is a Z-curve with default parameters *lo*=0.0, *a*=0.0, *b*=0.6*,* and *hi*=1.0.[[106]](#footnote-107) With these settings, groups ignore messages with negative congruence, and a message achieves a resonance of 1.0 for any congruence greater than 0.6. The parameters can be adjusted to give negative congruence a slightly negative effect, as shown in the following graph:

*hi*

*lo*

*a*

*b*

+1.0

+1.0

–1.0

–1.0

0

0

*congruencegm*

*resonancegm*

**Regard.** A group's willingness to be affected by messages from an actor depends on how much they trust that actor; we use the vertical relationship as a proxy for that trust. Like congruence, ranges from −1.0 to +1.0, which is too broad a range; if a group has no trust at all for the sending actor, should be zero. As it happens, our *relationship multiplier functions*, defined in the *Mars Analyst's Guide*, were designed for exactly this purpose. Using the rationale that messages should be able to affect even those who hate the perceived source to some extent, but that friends should be affected much more suggests using the **frmore** function.

However, the group might not know the source. Messages ascribed to anonymous sources are not ignored; in fact, a great deal of credence is usually given to what "everyone says." Thus, we define as follows:

#### Payload Magnitudes

Each payload affects a particular set of attitude curves, as described in the *Athena User's Guide*, and has a nominal magnitude *M* chosen by the analyst. The analyst chooses *M* using one of the same magnitude symbols (e.g., **XL+**) used in the *Athena Rules* document to state the effect of various rule sets. The analyst should choose *M* based on the following assumptions:

* The IOM will be completely acceptable () to the targeted group.
* The IOM will have 2/3rds coverage of the targeted group.[[107]](#footnote-108)

The actual magnitude given to URAM by the IOM rule set is then

where defaults to 2/3rds. This is consistent with the use of coverage in the activity situation rule sets.

#### Summary

Thus, when an IOM *m* is successfully broadcast via CAP *k*, Athena does the following:

For each group *g* in *k*'s audience:

Compute

For each payload in *m*

Compute the magnitude, .

Give inputs to URAM for each attitude curve affected by the payload.

### IOM Example

The small country of Elitia is divided into two provinces, Elitia proper and Peonia, with one major city, the Capital. There are two significant ethnic groups in Elitia, the Elitians and the Peons. The provinces are predominantly rural. Elitia is inhabited only by Elitians, and Peonia only by Peons, while the Capital is inhabited by both. Thus, we have four civilian groups: Urban and Rural Elitians (ELU and ELR), and Urban and Rural Peons (PEONU and PEONR).

There are three significant actors in the country: the Government Party (GOV), which dominates the country and consist mostly of Elitians; the Peonian Liberation Front (PELF), which is working toward Peonian independence and a democratic Peonian state; and the Elitian People’s Party (EPP) which is trying to break the Government Party’s stranglehold on the state and bring about a democratic revolution. The EPP is weakly in favor of Peonian independence, but mostly because PELF activities keep government forces focused on Peonia and out of Elitia, the EPP stronghold.

The rural Elitians are great dog breeders, and love their dogs very much; the Elitian Hound is the mascot of the EPP, and government forces have been increasingly aggressive toward dogs when putting down protests. Hence, the EPP might wish to broadcast the following IOM with the intent of decreasing support for the Government Party:

**Gov’t ministers eat puppies for breakfast!**

It is up to the analyst to take this description, and turn it into a semantic hook and one or more payloads.

One of the belief system topics is **DOGS**; actors and civilians can be for or against dogs. Both the EPP and the rural Elitians are passionately for **DOGS**, and this forms our semantic hook:

**DOGS: Passionately For**

Note that this topic need not have any role in determining group and group/actor affinities; but it is something that the civilian groups care more or less about, and which can be used to hook them.

Next, we must determine the payload. In the context of Elitian politics, these are some likely ones: the rural Elitians will support the government less, like the Elitian army less (because it’s a tool of the government), cooperate with the army less, and feel that their ancient dog-breeding culture is being denigrated.

* **Change vertical relationships with GOV by XXL−**
* **Change horizontal relationships with the ARMY by L−**
* **Change cooperation with the ARMY by L−**
* **Change satisfaction with CUL by XL−**

Note that the analyst must specify the nominal magnitude of each effect. This is because the effect of a real-world IOM depends on many factors, only a few of which are visible to Athena. Is the content of the message about something vitally important, or about something small? Is the message well written and produced, or is it lacking? Thus, the payload magnitudes reflect the analyst’s own assessment about the importance of the message content and the quality of production. Athena can then go on to assess the relative effects of the message on the recipients given the congruence of the semantic hook with their beliefs.

Eventually the EPP elects to send this message using their party newspaper, the Dog-Breeder’s Digest. They write it up as straight reporting, but the analyst determines that everyone who receives it will know that the Digest is an EPP publication, and sets EPP as the perceived source of the message. (They’d get wider circulation with a television spot, and might be able to present a more powerful message, but the state television station is controlled by the Government, and is too expensive besides.)

The Digest has wide circulation in Elitia, reaching 85% of the population; it is less widely read in the Capital, where it reaches 25% of the urban Elitians. Peons are generally cat people, and having their own concerns tend to ignore the Digest completely, though a few of the urban Peons read it just to keep an eye on things.

The semantic hook has high congruence with the beliefs of rural Elitians, and hence has a high resonance. The urban Elitians are much less in favor of dogs, having mostly left Elitia for the city to get away from them; still, they do not hate dogs, and so the hook has a low but not zero resonance with them. The urban Peons are against dogs, and hence the hook has no resonance with them at all.

Rural Elitians have a high vertical relationship with and hence high regard for the EPP. Only a vanishingly small fraction of urban Elitians have power in the government (even though many of them are government employees); hence, many of them secretly support the EPP even if not very strongly. Regard is low, but positive.

Thus, the above payloads will have a strong effect on the rural Elitians, a weak effect on the urban Elitians, and no effect on any of the Peons.

The Government Party, faced with a public relations disaster, orders its troops to stoop killing dogs, and puts out its own message, which might be stated as follows:

**Gov’t ministers are dog owners; only cat owners hate the government!**

This message might have a more complex semantic hook:

**DOGS: Weakly For**

**CATS: Passionately Against**

**GOVT: Strongly For**

And naturally it will have payloads intended to increase support for the Government Party and its forces, and decrease support for the PELF. The effect might not be what was intended, however. The hook fails to catch the rural Elitians, who are passionately for dogs, don’t care much about cats either way, and truly hate the government. The urban Elitians, who are the real target of the message, are only weakly caught. They are weakly for dogs themselves, but they are only weakly against cats (associating them with the lower class urban Peons), and they are only weakly for the current government. Thus, resonance is low. Further, their vertical relationship with GOV is rather lower than it could be, so their regard is low, too. The message will go some way to propping up the current regime, but is unlikely to be terribly effective.

(The Peons, naturally, reject the government and its anti-cat message; and expecting nothing better from these evil tyrants, are not affected at all.)

## Output and Causality Analysis

This section presents an architecture and set of models for comparing two states of the Athena simulation[[108]](#footnote-109) in order to find significant differences and to trace the reasons for those differences back through the Athena models. Ultimately we wish to trace causality from outputs all the way back to the scenario inputs; the current models get us part of the way.

This section describes the basic framework for analyzing outputs and their inputs. The specific output and input variable types are documented in Athena’s on-line help, so that the information can be made available to the user of the software.

### Definitions

Athena has many models, each with its own inputs and outputs. The forms of these models vary greatly; hence, the first step is define a notation for describing the Athena models and there inputs and outputs so that we can work with them generically.

An output *z* of the Athena model is defined as:

where

*z* = An output of the Athena model, regardless of data type

= One of the *n* inputs to *z*.

The function *f* might be arbitrarily complex, and the number of inputs *n* might be quite large.

When comparing two scenario states, case A and case B, we will compare matching outputs *Z*, where *Z* is the ordered pair , the values of *z* in the two cases.

A variable type *T* is a particular kind of variable with its meaning, range of values, and indices, e.g., the satisfaction of group *g* with respect to concern *c* or the mood ofgroup *g*.

For each variable type *T* we define a delta function , which is a non-negative measure of the difference between and . The nature of the delta function depends on the type, but the most usual definition is the absolute difference between the two values:

Every output pair *Z* is a function of *n* input pairs , where

Each input pair of type *T* naturally has a delta function .

Finally, we will define for each input pair a significance score , where and . The score is used to rank input pairs by their value in explaining the magnitude of . The essential part of this model is the manner in which the scores are computed.

### Significance Scoring

Each input pair plays some part in explaining , the difference between the output values and . Our goal is define a significance score , , such that input pairs with higher scores have correspondingly higher explanatory values. We say that input pairs with a sufficiently high score are *significant*; the other input pairs are *insignificant*. In other words, input pair is significant if , where is the *significance threshold*.

We compute the values in several steps. First, we compute a normalizer for each input type *T* based on the input pairs of that type. Second, we assign a score to each input pair of type *T* based on its and the normalizer ; this score ranks the input pair relative to the other input pairs of that type. Third, we compute the scores based on the values.

We will go into each of these steps in detail in the following subsections.

#### Delta Functions

As noted above, the most usual delta function is simply the absolute difference of the values from case A and case B:

However, many other delta functions are possible. For example, some types of Athena outputs have symbolic values (e.g., the actor *a* in control of neighborhood *n* at the end of the run), and this means that we have to define special delta functions for these types. The simplest function is just

This function says that all differences are equally significant. More complicated delta functions can be defined if desired.

Ideally, delta functions should depend only the values and . One can imagine delta functions that take into account the context (i.e., the output values and or the values of the other input pairs), but such complications should be avoided if possible.

#### Q-scores and Normalizers

We rank the input pairs of type *T* as follows:

The value of the normalizer is intended to bring the deltas into a nominal range of 0.0 to 1.0 so that the resulting Q-score has a nominal range of 0.0 to 100.0. The normalizer depends on the input type. At present, we define three distinct kinds of normalizer.

First, can be a constant, usually (but not always) the maximum value of . For cooperation levels, for example, the maximum is 100.0 cooperation points; dividing by that gives a Q-score that ranges exactly from 0.0 to 100.0.

Some input types have a significant zero-point. Negative relationships and satisfaction levels are in a way qualitatively different than positive relationships and satisfaction levels. For these we set to half of the possible magnitude; this yields Q-scores ranging from 0.0 to 200.0, and gives greater weight to output pairs that span the zero-point. Thus, for horizontal and vertical relationships rather than 2.0.

Second, some input types reflect arbitrary arbitrarily large quantities; there is no obvious maximum delta value to use as a constant normalizer. For such types we want to normalize against the maximum value in the two scenarios. There are two subcases: where the values of the type are naturally cumulative, and where values of the type are not naturally cumulative.

The prototypical cumulative type is the population of civilian group *g*. When looking at the significance of a particular population delta in the context of the neighborhood mood, it makes sense to normalize it against the total population of that neighborhood. Similarly, when looking at the significance of a population delta in the context of the playbox mood it makes sense to normalize it against the total population of the playbox. In cases like this,

That is, we compute the total of the relevant set of populations for each case, and then take the larger of the two.

On the other hand, the unemployment rate in neighborhood *n* is non-cumulative: it doesn’t make sense to add up unemployment rates across neighborhoods. In this case we will normalize against the largest absolute value in the set:

#### Computing Significance Scores

Finally, we define the significance score as follows:

where

is a type-specific parameter, nominally 1.0, that is used to raise or lower the significance of input pairs of type *T* relative to input pairs of other types. It is usually chosen to reflect the role that inputs of type *T* play in the computation of output *z*, but it can also be used to adjust the relative rankings if the default rankings seem not quite right.

### Significant Output Detection

TRADOC G2-7 analysts have provided us with a set of primary Athena outputs, the outputs they look at when assessing and comparing Athena runs. We must be able to determine which of the vast number of primary outputs have changed significantly over the course of a single run, or that differ significantly between two distinct runs.

Thus far we have spoken about assigning significance scores to input pairs of an output pair *Z*; we will use the same scheme to rank Athena scenario outputs for presentation to the analyst. Each output pair , for , will be assigned a score using exactly the equations shown in Section 3. In this case the values used in the final step will be chosen to reflect the importance of output pair ’s type *T* to the analyst, and output will be considered significant in the context of the scenario as a whole if .

### Causality Chaining

Our goal in computing the significance of input pairs of an output pair *Z* is to explain the magnitude of , that is, to explain why the output changed as it did. But an input value *x* is often the output of some other model, and has its own inputs. We can compute significance scores for these inputs, and for their inputs, and so on, until we reach variables that are truly Athena scenario inputs.

The result is a tree of input pairs and scores:

*Z* ⇐ , ⇐ , ⇐ …

,

,

…

⇐ , ⇐ …

,

,

…

⇐ …

…

This tree can be extremely deep and bushy; however, the analyst can prune it automatically by setting the significance level . If , the tree will include all inputs to *Z*, whether direct or indirect, that changed at all between cases A and B. If , then the tree will include only the very most significant inputs at leach level.

# Appendices

## Acronyms

AAM Athena Attrition Model

AUT Autonomy (concern)

CIV Civilian

CSF Consumer Security Factor

CTR Contractor

CUL Culture (concern)

DAM Driver Assessment Model

ECDA Expected Collateral Damage per Attack

ECDC Expected Collateral Damager per NF Casualty

ELER Expected Loss Exchange Ratio

FRC Force

IED Improvised Explosive Device

IGO International or Inter-Governmental Organization

JNEM Joint Non-lethal Effects Model

LER Loss Exchange Ratio

LFF Labor Force Fraction

LSF Labor Security Factor

MAD Magic Attitude Driver

MAG Mars Analyst's Guide

NF Non-uniformed Force

NGO Non-Governmental Organization

ORG Organization

QOL Quality of Life (concern)

ROE Rules Of Engagement

S&RO Stability & Recovery Operations

SFT Safety (concern)

UF Uniformed Force

UNESCO United Nations Educational, Scientific, and Cultural Organization

URAM Unified Regional Attitude Model

1. When installed on the Windows operating system, this documentation is also available from the Windows start menu. [↑](#footnote-ref-1)
2. Athena uses neighborhoods as homogeneous bins; map coordinates and distances are of interest only for visualization and have no effect on the model results. Internally, locations are stored using latitude/longitude coordinates in decimal degrees.. [↑](#footnote-ref-2)
3. Note that Athena does not directly model the effect of foreign actions on US civilian attitudes, but “magic” attitude inputs could be used for this purpose. [↑](#footnote-ref-3)
4. Or models! Stability has social, economic, and political dimensions. [↑](#footnote-ref-4)
5. Political, Military, Economics, Social, Infrastructure, Information plus Physical and Time. [↑](#footnote-ref-5)
6. Static, that is, unless magic attitude drivers (MADs) are used to change them. [↑](#footnote-ref-6)
7. Model parameter: control.support.vrelMin, nominally 0.2. [↑](#footnote-ref-7)
8. Model parameter: control.support.Zsecurity [↑](#footnote-ref-8)
9. See the *Athena User’s Guide* for specifics about tactic types. [↑](#footnote-ref-9)
10. Model parameter: control.support.min , nominally 0.1. [↑](#footnote-ref-10)
11. We will make this notion more precise below. [↑](#footnote-ref-11)
12. Model Parameter: control.threshold, nominally 0.5. [↑](#footnote-ref-12)
13. Model parameters: force.discipline.\*. [↑](#footnote-ref-13)
14. Model parameters: force.law.suitability.\*. [↑](#footnote-ref-14)
15. It is true that different kinds of force group are trained for different things; however, this is accounted for under suitability. [↑](#footnote-ref-15)
16. Model parameters: force.law.efficiency.\*. [↑](#footnote-ref-16)
17. Model parameters: force.law.beta.\*. [↑](#footnote-ref-17)
18. See Section 6 for more on coverage functions. [↑](#footnote-ref-18)
19. Model parameters: force.law.coverage.\*. [↑](#footnote-ref-19)
20. Model parameters: force.law.crimfrac.\*. [↑](#footnote-ref-20)
21. Model parameter: force.law.suppfrac [↑](#footnote-ref-21)
22. Model parameter: force.population. [↑](#footnote-ref-22)
23. Model parameter: force.demeanor.\*. [↑](#footnote-ref-23)
24. See the *Mars Analyst's Guide* for the computation of mood. [↑](#footnote-ref-24)
25. Model parameter: force.mood. [↑](#footnote-ref-25)
26. Model parameters: force.orgtype.\*. [↑](#footnote-ref-26)
27. Model parameters: force.forcetype.\* [↑](#footnote-ref-27)
28. Model parameters: force.alpha.\* [↑](#footnote-ref-28)
29. Model parameter: force.proximity. An appropriate value of *h* should be selected during exercise construction to reflect the criteria used for selection of neighborhood boundaries. Normally, a playbox with large neighborhoods would have a smaller value for *h* than one with small neighborhoods. [↑](#footnote-ref-29)
30. Model parameter: force.volatility [↑](#footnote-ref-30)
31. Ideally, the nature of each unit would also be taken into account, at least at the level of "boots, wheels, or tracks," i.e., the number of personnel on foot, in wheeled vehicles, and in tracked vehicles. Tanks tear up the pavement, for example, and thus should have an increased negative effect on QOL. At present, Athena does not model units to this level of detail. [↑](#footnote-ref-31)
32. Model parameter database, activity.FRC.\* [↑](#footnote-ref-32)
33. Model parameter: activity.ORG.\* [↑](#footnote-ref-33)
34. In future versions, actors may also be able to interfere with the provision of a service. [↑](#footnote-ref-34)
35. service.\*.required.*urbanization*, where \* is either WATER, ENERGY or TRANSPORT [↑](#footnote-ref-35)
36. Services that require infrastructure are significantly more complicated; and moreover, each kind of service is likely to require its own unique infrastructure model. Thus, for this version we have limited ourselves to the simplest case. [↑](#footnote-ref-36)
37. Model Parameter: service.ENI.beta.*urbanization*, nominally 1.0 for all urbanization levels. [↑](#footnote-ref-37)
38. Model Parameter: service.ENI.saturationCost.*urbanization*. [↑](#footnote-ref-38)
39. Model parameter: service.eni.minSupport, nominally 0.0. [↑](#footnote-ref-39)
40. Model parameters: service.\*.alphaA, service.\*.alphaX. [↑](#footnote-ref-40)
41. Model parameter: service.\*.delta, nominally 0.1. [↑](#footnote-ref-41)
42. Model parameter: service.\*.required.urbanization. [↑](#footnote-ref-42)
43. Model parameter: service.\*.gainNeeds. [↑](#footnote-ref-43)
44. Model parameter: service.\*.gainExpect. [↑](#footnote-ref-44)
45. This table is defined in the code using the qcredit(n) quality type. [↑](#footnote-ref-45)
46. Prior versions contained a prototype model of attrition between "uniformed" and "non-uniformed" force groups. [↑](#footnote-ref-46)
47. AAM considers only one neighborhood at a time, so we drop the *n* subscript throughout this discussion. [↑](#footnote-ref-47)
48. In the future, we may only allow hiding groups to perform a limited set of tactics and activities. [↑](#footnote-ref-48)
49. Activities can only increase visibility, so the activity multipliers should be greater than 1. [↑](#footnote-ref-49)
50. In the future, we may also include force type and training level. [↑](#footnote-ref-50)
51. Model Parameters: aam.visibility.\* [↑](#footnote-ref-51)
52. Model Parameters: activity.FRC.\*.visFactor [↑](#footnote-ref-52)
53. Model parameter: aam.detectionGain [↑](#footnote-ref-53)
54. Model parameters: aam.FRC.equiplevel.\* [↑](#footnote-ref-54)
55. Model parameters: aam.FRC.discipline.\* [↑](#footnote-ref-55)
56. Model parameters: aam.FRC.forcetype.\* [↑](#footnote-ref-56)
57. Model parameters: aam.FRC.demeanor.\* [↑](#footnote-ref-57)
58. Model Parameters: aam.lc.\* [↑](#footnote-ref-58)
59. Model parameters: aam.FRC.urbcas.\* [↑](#footnote-ref-59)
60. Model parameters: aam.FRC.civconcern.\* [↑](#footnote-ref-60)
61. Model parameter: aam.maxCombatTimeHours [↑](#footnote-ref-61)
62. Model Parameter: aam.civcas.urbanization.\* [↑](#footnote-ref-62)
63. Model Parameter: aam.civcas.forcetype.\* [↑](#footnote-ref-63)
64. Model Parameter: aam.civcas.discipline.\* [↑](#footnote-ref-64)
65. Model Parameter: aam.FRC.civconcern.\* [↑](#footnote-ref-65)
66. Model Parameter: aam.civcas.limit [↑](#footnote-ref-66)
67. Model parameters: dam.CIVCAS.Zsat [↑](#footnote-ref-67)
68. Model parameter: dam.CIVCAS.Zcoop [↑](#footnote-ref-68)
69. Model parameter: dam.CIVCAS.Zhrel [↑](#footnote-ref-69)
70. Model parameter: dam.CIVCAS.Zvrel [↑](#footnote-ref-70)
71. The detailed proof is in the Appendix of memo bjk15\_001\_aam. Contact [Brian.Kahovec@jpl.nasa.gov](mailto:Brian.Kahovec@jpl.nasa.gov) for more info. [↑](#footnote-ref-71)
72. By "labor force" we mean that portion of the population that seeks to be employed, whether they are in fact employed or not. [↑](#footnote-ref-73)
73. Model parameter: econ.secFactor.labor.*security*. [↑](#footnote-ref-74)
74. Model parameter: econ.secFactor.consumption.*security*. [↑](#footnote-ref-75)
75. Model parameter: demog.turFrac. [↑](#footnote-ref-76)
76. The maximum distance is controlled by a model parameter called demog.maxcommute and has a default of “far”. [↑](#footnote-ref-77)
77. Model parameter: demog.Zuaf. [↑](#footnote-ref-78)
78. The income disparity model is based on "Wage Distributions and Survival," by Rebecca Lawler; Lawler was a Summer Undergraduate Research Fellow (SURF) on the Athena team the summer of 2010. The rest of the model was developed by Brian Kahovec with inputs from the other members of the modeling team. [↑](#footnote-ref-79)
79. Model Parameter: demog.consump.alphaA, nominally 0.5. [↑](#footnote-ref-80)
80. Model Parameter: demog.consump.alphaE, nominally 0.25. [↑](#footnote-ref-81)
81. Model parameter: demog.consump.expectfGain. [↑](#footnote-ref-82)
82. Model parameter: demog.consump.RGPC.\*. [↑](#footnote-ref-83)
83. See "Wage Distributions and Survival", by Rebecca Lawler. [↑](#footnote-ref-84)
84. Model Parameter: demog.gini. [↑](#footnote-ref-85)
85. During the derivation of this model, the modeling team examined the possibility of allowing *f* to vary with *x* and concluded that the effect wasn't significant. [↑](#footnote-ref-86)
86. Model parameter: demog.consump.Zpovf. [↑](#footnote-ref-87)
87. In some regions, it's quite possible that the *lo* parameter should be much higher. [↑](#footnote-ref-88)
88. Actors may have up to two forms of income from the black market: a tax-like rate on gross revenue and shares of net revenue. [↑](#footnote-ref-89)
89. See “Athena’s CGE Model” by Chamberlain, Duquette and Kahovec included with the Athena documentation. [↑](#footnote-ref-90)
90. The cell variable names all begin with ‘*B’* to indicate that these are “base” values used to initialize Athena’s Computable Equilibrium Model (CGE) when the scenario is locked. [↑](#footnote-ref-91)
91. See “Athena’s CGE Model” by Chamberlain, Duquette and Kahovec included with the Athena documentation. [↑](#footnote-ref-92)
92. econ.shares.*tt.i* is the family of parameters where *tt* is tactic type or and *i* is the sector. [↑](#footnote-ref-93)
93. An expenditure class is a tactic type (ASSIGN, DEPLOY, etc…)or “overhead”. [↑](#footnote-ref-94)
94. This is a model parameter: econ.initCapPct and is nominally 100% [↑](#footnote-ref-95)
95. This is a controlled by a model parameter: plant.bktsPerYear.goods [↑](#footnote-ref-96)
96. This is a number between 0.0 and 1.0 where 0.0 corresponds to inoperable and 1.0 corresponds to maximum production capacity. [↑](#footnote-ref-97)
97. The parameter is plant.lifetime [↑](#footnote-ref-98)
98. The parameter is plant.repairtime [↑](#footnote-ref-99)
99. The maximum cost is a fraction of the build cost: plant.repairfrac × plant.buildcost [↑](#footnote-ref-100)
100. The parameter is plant.buildtime [↑](#footnote-ref-101)
101. The maximum cost is plant.buildcost [↑](#footnote-ref-102)
102. See the *Athena User's Guide* and on-line help for details on orders and tactics. [↑](#footnote-ref-103)
103. Given the wide variety of CAPs in the real world, and the vastly different economics of running a television station vs. a newspaper vs. a website, it is difficult to come up with a one-size-fits-all cost model that truly makes sense. Hence, our goal here is simplicity, in accordance with the Metivier Principle. [↑](#footnote-ref-104)
104. Because CAPs do not yet have explicit infrastructure, they do not yet have home neighborhoods. However, the identity of the CAP’s neighborhood is presumably known to the analyst, and can be used when defining the conditions on the GRANT tactic. [↑](#footnote-ref-105)
105. The “Theory of Homophily” may have a good formal definition somewhere, but we have not taken the time to search the literature to find it. Wikipedia defines homophily as “the tendency of individuals to associate and bind with similar others.” We are extending the notion beyond willingness to associate with other people to willingness to associate with other ideas. Wikipedia says the “original formulation” was in [Lazarsfeld 1954] and that [McPherson 2001] is an “extensive review paper”. [↑](#footnote-ref-106)
106. Model parameter: dam.IOM.Zresonance. [↑](#footnote-ref-107)
107. Nominal coverage defaults to 2/3rds, but is set by the model parameter dam.IOM.nominalCAPcov. [↑](#footnote-ref-108)
108. Typically the beginning and end of a run, or the end states of two related runs. [↑](#footnote-ref-109)