DRAFT
Stratmas Models
DRAFT

January 31, 2007

This document contains descriptions of some of the models implemented in Stratmas. It should currently be considered as a work in progress. The distant goal is that it should continue to grow until it eventually contains all model documentation relevant to Stratmas.

It should be noticed that the author of this document is not the designer of the models described in it, but simply the one putting them on paper. It should also be pointed out that the outline and layout of this document may differ from what perhaps is expected from documents concerning similar topics. This is intentional and an explicit requirement from the assigner.

1 Military Units

The following sections describes the military units in Stratmas. The attributes tied to military units are described and explained as well as the orders that units may be given and what they mean.

1.1 Unit Attributes

This section provides descriptions of all attributes tied to Military units.

affiliation indicates which faction a unit belongs to. This must be one of the factions defined in the scenario's faction list. A unit's faction determines which other factions it is hostile towards according to the scenario's faction list. A unit u will only engage in battle with a unit e if u considers e an enemy and/or vice versa.

attackFactor is a factor that is multiplied to the unit's basic strength if the unit has an attack order. The purpose of the attack factor is to introduce the possibility to give a unit different attack and defend capabilities.

attritionCoefficient is no longer used and will be removed in future versions.

casualties is the current number of casualties for a unit. It is cumulative for the entire simulation.

defenseFactor is a factor that is multiplied to the unit's basic strength if the unit has a defend order. The purpose of the defend factor is to introduce the possibility to give a unit different attack and defend capabilities.

departTime is the time that a unit will be removed from the simulation. When this time has passed, the unit will no longer have any effect on the simulation. Notice that this does not imply that the unit's subunits are removed from the simulation.

deployTime is the time that a unit will enter the simulation. Before this time has passed, the unit will not have any effect on the simulation. Notice that this does not necessarily imply that the unit's subunits enter the simulation at the same time.

deployment is a distribution that represents how a unit distributes its strength over the area it currently occupies.

location is the area the unit currently occupies. It may take the shape of a circle or a polygon.

maxVelocity is a unit's maximum velocity in kilometers per hour. Units normally travel at an average speed of half their maximum velocity.

activity is a list of orders that a unit should carry out. Orders are described in greater detail in section 1.3.

personnel is the current number of personnel a unit has. It may be decreased when a unit is engaged in combat and thus affected by another unit or by insurgents.

present is a flag that indicates if a unit is currently present in the simulation. A unit is considered present if the current simulation time falls between the units deployTime and departTime.

strengthFactor is a number used to calculate a unit's basic strength[1] from its personnel as:

$$basicStrength = personnel \cdot strengthFactor$$

subunits is a list of this unit's immediate subunits.

symbolIDCode is the MilStd 2525B/APP-6A symbol id code describing the symbol for a unit. See the documents [4] and [5] for more information about symbol id codes.

withdrawThreshold is a number indicating which losses a unit can take before retreating from combat. A unit will retreat as soon as its strength falls below withdrawThreshold percent of its initial strength.

[1] The basic strength concept is introduced by Thorssell in [2].

1.2 Unit States

In order to describe the orders a unit may be given and how the combat model works we introduce a number of different states that a unit may be in. Notice that the states are not mutually exclusive.

alive A unit u is alive if $personnel > 0$

present A unit u is present if the current simulation time is later than u's deploy time and earlier or equal to u's depart time.

capable A unit u is capable if it has more than $withdrawThreshold$ percent of its initial personnel.

moving A unit u is moving if its location changed during the last time step.

combat situation A unit u is in a combat situation if u overlaps at least one cell that is also overlapped by a unit that is hostile towards u.

critical insurgent situation A unit u is in a critical insurgent situation if it is not in a combat situation and if it did lose more than 0.1 percent of its current personnel due to insurgent actions the last time step.

untouchable A unit u is untouchable if it can not be harmed by any of its enemies. This state has been introduced in order to fill the requirements of the RetreatOrder and the AmbushOrder, see section 1.3.

1.3 Orders

The following sections describes different aspects of the orders that may be given to military units in Stratmas.

1.3.1 Order Execution

An order o will not be executed until the simulation time has passed o's start time. However, it is not guaranteed that o is executed as soon as the simulation time has passed o's start time. A unit u may have to create an Attack, Defend or Retreat order or it may not be capable or there may be more than one order that is defined to be executed at the same time. If the latter is the case the order with the earliest start time will be executed. If both orders have the same start time it is unspecified which of the orders that will be executed.

1.3.2 Stratmas Orders

This section describes all the orders that may be given to a military unit in Stratmas and what behavior they lead to. The attributes tied to the different orders are also described. In the sections below u denotes a unit and o the order that is currently described.

GoTo Order

The GoTo order will move u with half its maximum velocity in a straight line towards o's location. When the center point of u's location equals the center point of o's location u's location changes to o's location. For example, if u's location is a circle and o's location is a square then u's location will change into a square when u arrives at the center point of o's location. o is carried out when u's location equals o's location.

Attributes

active is a flag indicating if a GoTo order is currently executed. A GoTo order is considered active if the simulation time has passed the start time of the order and the goal for the order is not yet reached.

carriedOut is a flag indicating whether a GoTo order has been carried out or not. A GoTo order is considered carried out if the simulation time has passed the start time of the order and the goal for the order is reached.

location is the area to go to. The location may take the form of a circle or a polygon. When the performing unit reaches the center of the location area it regroups, e.g it sets its own area to the location area of the order.

start is the start time of a GoTo order. The order will not be carried out until the simulation time passes this time.

Attack Order

If o is an Attack order and o has a location then move u to the location of o. Combat is handled as described in section 1.5. o is considered to be carried out when u's location equals o's location (if o has a location) and when u is not in a combat situation.

Attributes

active is a flag indicating if an Attack order is currently executed. An Attack order is considered active if the simulation time has passed o's start time and the goal for the order is not yet reached or if the goal is reached and u is in a combat situation.

carriedOut is a flag indicating whether an Attack order has been carried out or not. An Attack order is considered carried out if the simulation time has passed the start time of the order, the goal for the order is reached and u is not in a combat situation.

location is the area to go to. The location may take the form of a circle or a polygon. When u reaches the center of the location area it regroups, e.g it sets its own area to the location area of the order.

start is the start time of an Attack order. The order will not be carried out until the simulation time passes this time.

Defend Order

If o has a location then move u to the location of o. Combat (if any) as described in section 1.5. o is considered to be carried out when:

- If o has a location - the simulation time has passed o's end time.
- If o has no location - u is no longer in a combat situation.

Attributes

active is a flag indicating if a Defend order is currently executed. A Defend order is considered active if the simulation time has passed o's start time but not its end time or if o has no location - as long as u is in a combat situation.

carriedOut is a flag indicating whether a Defend order has been carried out or not. A Defend order is considered carried out if the simulation time has passed the end time of the order or if o has no location - when u is no longer in a combat situation.

end is the end time of an order. When the simulation time passes this time the order will no longer be carried out.

location is the area to go to. The location may take the form of a circle or a polygon. When u reaches the center of the location area it regroups, e.g it sets its own area to the location area of the order. When u is attacked and has to defend itself the defend order will not get an area but is assumed to be executed in u's current location.

start is the start time of a Defend order. The order will not be carried out until the simulation time passes this time.

Ambush Order

Transfer u to the untouchable state and move u to the location of o. After the simulation time has passed the startAmbush time and before it has passed the endAmbush time, ambushes will be performed according to the model described in section 1.6. Between the endAmbush time and the end time u is again transferred to the untouchable state. o is considered to be carried out when the simulation time has passed o's end time.

Attributes

active is a flag indicating if an Ambush order is currently executed. An Ambush order o is considered active if the simulation time has passed o's start time but not its end time.

carriedOut is a flag indicating whether an Ambush order has been carried out or not. An Ambush order is considered to be carried out if the simulation time has passed the end time of the order.

end is the end time of the order. When the simulation time passes this time the order will no longer be carried out.

endAmbush After this time has passed u will no longer perform any ambushes and can not be detected by other units until the simulation time passes o's end time.

location is the area to go to. The location may take the form of a circle or a polygon. When u reaches the center of the location area it regroups, e.g it sets its own area to the location area of the order.

start is the start time of an Ambush order. The order will not be carried out until the simulation time passes this time.

startAmbush No ambush will be performed and the performing unit u may not be detected by other units the simulation time passes the startAmbush time. After this time has passed u will perform ambush according to the model described in section 1.6.

Presence, Control, Secure, Freedom of Movement, Provide Civilian Functions

These orders have in large parts the same meaning for the unit performing them. Since they all affects process variables in an area they are as a group called *area orders*. What differs is their respective combat readiness[2] plus the magnitude of the influence they have on the process variables. The process variables that may be modified by these orders are listed in appendix A. How process variables are affected is described in section 3.

If o is one of the orders Presence, Control, Secure, Freedom of Movement or Provide Civilian Functions and o has a location then move u to the location of o else set o's location to u's location. Thereafter modify the given process variables with the given magnitudes. o is considered to be carried out when the simulation time has passed o's end time.

Attributes

active is a flag indicating if o is currently executed. o is considered active if the simulation time falls between o's start and end time.

carriedOut is a flag indicating whether o has been carried out or not. o is considered carried out if the simulation time has passed o's end time.

end is the end time of o. When the simulation time passes this time the order will no longer be carried out.

location is a location that may be tied to o. The location may take the form of a circle or a polygon. The location is an optional attribute. If a location is given, it is the area in which the order should be carried out. If u is not initially located in this area it will start by moving there, regrouping, e.g. set its own location to the location of the order, when it arrives. If no location is given the order will be carried out at the current location of u.

start is the start time of o. The order will not be carried out until the simulation time passes this time.

[2] See section 1.3.3.

target is an optional attribute indicating which population faction to target. Specifying a faction means that for process variables that are divided into factions, the order will only affect the specified faction. If no faction is specified, the order will affect all factions.

TerroristAttack Order

Transfer u to the untouchable state and move u to the location of o. When the simulation time passes the actionTime, u risks to be detected by the set of units E where each unit $e \in E$ overlapps the center cell c of u and is hostile towards u. For $e \in E$ the detection probability P is:

$$P = 1 - e^{-\frac{e_c}{population_c} \cdot \gamma}$$

where e_c is the personnel of e in cell c, $population_c$ is the number of inhabitants in c and gamma is a constant defined as:

$$\gamma = -10 \cdot \log 0.5$$

The basic assumption is that the probability to detect a terrorist attack should be 0.5 if there are 1 soldier per 10 inhabitants and square kilometer.

If u is detected it is destroyed immediately. If u is not detected it will modify process variables according to the area model described in section 3, that is — in the same way as the area orders described above. o is considered to be carried out when the simulation time has passed o's end time.

Attributes

actionTime is the time that the action will take place, e.g. the time when the specified process variables will be modified.

active is a flag indicating if o is currently executed. o is considered active if the simulation time falls between o's start and end time.

carriedOut is a flag indicating whether o has been carried out or not. o is considered carried out if the simulation time has passed o's end time.

end is the end time of o. When the simulation time passes this time the order will no longer be carried out.

location is a location that may be tied to o. The location may take the form of a circle or a polygon. The location is an optional attribute. If a location is given, it is the area in which the order should be carried out. If u is not initially located in this area it will start by moving there, regrouping, e.g. set its own location to the location of the order, when it arrives. If no location is given the order will be carried out at the current location of u.

start is the start time of o. The order will not be carried out until the simulation time passes this time.

target is an optional attribute indicating which population faction to target. Specifying a faction means that for process variables that are divided into factions, the order will only affect the specified faction. If no faction is specified, the order will affect all factions.

Retreat Order

Retreat orders may not be given to units by the user. They are only created when a unit is in a combat situation and it is no longer capable.

If o is a Retreat order, move u to a location l that has the shape of a circle with radius 10 km and is located 50 km from u's position (when given the order) in the direction that has the greatest angular distance from any of u's current enemies. When u has reached that location and if u has a superior unit then transfer u to the untouchable state and move u with 25 percent of its maximum velocity to the center point of u's superior unit. o is carried out when:

- If u has a superior unit - u has reached the center point of the superior.
- If u has no superior - u has reached l.

1.3.3 Order Influence on Unit Strength and Civilian Capabilities

A unit u has different combat readiness depending on which order it is currently executing. This is represented by a number in the range $[0, 1]$ where 1 is maximum readiness. The combat readiness number for the different orders may be found in table 1. If u is attacked when its combat readiness is below 1.0 it will get a reduced combat value in the first part of the battle. A unit is assumed to gain 20 percentage points combat readiness per hour.

The combat readiness also affects how large part of a units capacity that may be used to perform non combat related activities. For example if a unit has a Secure order it may only use $1.0 - 0.6 = 0.4$ that is 40 percent of its total capacity for civilian functions.

Order	**Combat Readiness**
AmbushOrder	1.0
AttackOrder	1.0
ControlOrder	0.8
DefendOrder	1.0
FreedomOfMovementOrder	0.4
GoToOrder	0.2
PresenceOrder	1.0
ProvideCivilianFunctionsOrder	0.2
RetreatOrder	0.0
SecureOrder	0.6

Table 1: The combat readiness parameter for different orders in Stratmas.

1.4 Unit Behavior

This section describes how units act during a time step. It is divided in two subsections – one for the setup and one for the actual execution.

1.4.1 Setup

For each unit u that is present and alive determine its order o according to the following:

- If u is in a combat situation do:
  - Calculate u's modified strength.
  - If u is capable, o will be an Attack order if u is moving and a defend order if u is not moving. If u is not capable the order will be a Retreat order.
- Else if u was ambushed during the last time step then try to find the ambushing units in order to defeat them. This may be seen as giving u a search order even if this order is not explicitly implemented as the other orders. The ambush model is described in greater detail in section 1.6.
- Else go through the list of orders for u and pick the first one that is active at the current time step. Thereafter - set u's modified strength.

A unit u's modified strength is calculated as:

$$personnel \cdot strengthFactor \cdot factor$$

where factor is:

- If u is in a combat situation the factor is 1 or slightly less than 1 depending on the order the unit had in the last time step. For example, if the last order was a Secure order and the unit now is in a combat situation it has been surprised and its strength should thus be reduced based on the fact that it must regain its combat readiness with 20 percentage points per hour.
- Else if u has an order the factor is determined by that order's combat readiness, see table 1.
- Else the factor is 1.

1.4.2 Execution

When a unit acts the following steps are performed:

- If u has an order then perform that order.
- If u is neither in a combat situation nor a critical insurgent situation, u recovers one $1/strengthFactor$ personnel per day.

1.5 Force on Force Combat

This section describes how combat between military units is handled in Stratmas. During combat each unit u goes through its current enemies and for each enemy e exposes e for an attack of magnitude m:

$$m = u.modifiedStrength \cdot u.attackDefendFactor \cdot factor$$

where $u.modifiedStrength$ is u's modified strength, $u.attackDefendFactor$ is the attack or defend factor if u has an attack or defend order respectively and

$factor$ is the amount that e's strength constitutes of the sum of the strength of all u's enemies. If we define e's final strength s as:

$$s = e.modifiedStrength \cdot e.attackDefendFactor \cdot f$$

where f is the amount that u's strength constitutes of the sum of the strength of all e's enemies, then the damage d in persons that e suffers from the attack by u during a time step is calculated as:

- If e has a Defend order the quote q is defined as:

  $$q = \begin{cases} \frac{m}{s} & \text{if } s \neq 0 \\ 6 & \text{if } s = 0 \end{cases}$$

  and the damage d as:

  $$d = (0.025 \cdot quote + 0.025) \cdot personnel \cdot f \cdot fractionOfDay$$

- If e has an Attack order the quote q is defined as:

  $$q = \begin{cases} \frac{s}{m} & \text{if } m \neq 0 \\ 6 & \text{if } m = 0 \end{cases}$$

  and the damage d as:

  $$d = max((-0.025 \cdot quote + 0.15) \cdot personnel \cdot f \cdot fractionOfDay, 0)$$

where $fractionOfDay$ is the fraction of a day that the current time step constitutes. It should be noted that this results in that the outcome will become different depending on the size of the time step.

The damage model is based on the assumption that a unit may be either attacker or defender and that the magnitude of the damage may be determined by the quote between the attacker's and the defender's[3] modified strengths. The attacker has a more difficult task that the defender and will thus get more damaged unless it has a very significant dominance. Figure 1 shows graphically how large damage attacker and defender will get based on the quote between their respective modified strengths. In the case where both units are attackers both units gets damaged according to the Attacker scale. Two units battling each other can not both be defenders at the same time.

It should be noted that this is an approximation of a modification of Thorssell's table based damage scale[4]. The number 6 is chosen since it is the largest quote defined in Thorssell's paper. There is thus an anomaly when s is close to 0 allowing the damage to get larger than if s is exactly 0.

1.6 Ambush

A unit that has been given an ambush order follows different rules than the ordinary combat rules. A unit u with an ambush order o has the following

[3]There may also be combats involving only attackers, in which case the quote between the own and the enemy's modified strength is used.

[4]This model is only documented in an informal note written in Swedish

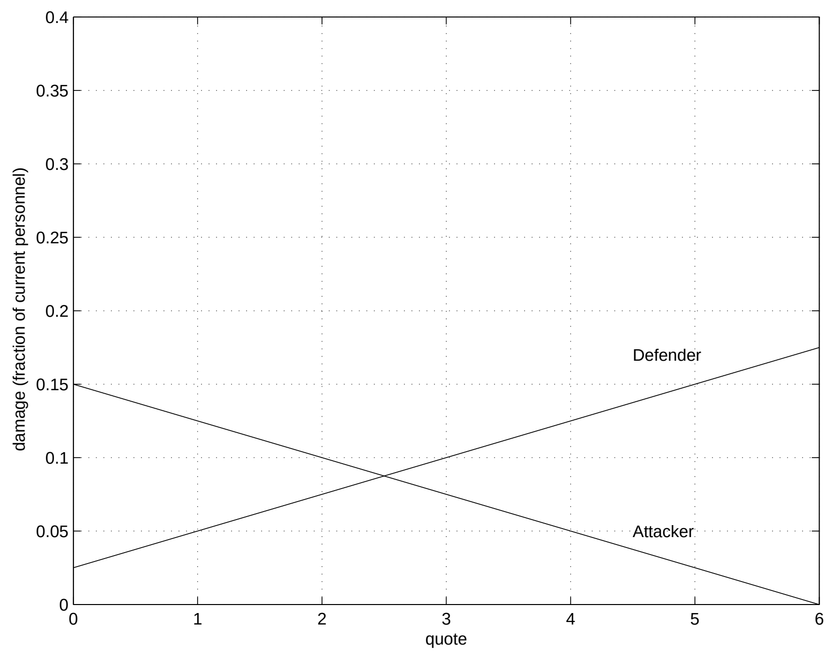


Figure 1: The damage in fraction of current personnel for military units.

characteristics. From the start time to the startAmbush time u is in the untouchable state. This is supposed to represent the fact that u is hidden. When the simulation time is between the startAmbush and endAmbush times u may perform ambushes as described below. After the endAmbush time and before the end time u is again transferred to the untouchable state.

To find out where an ambush will take place and which units u will ambush the following steps are performed:

- Find the set of cells C overlapped by u where the personnel of u is denser than 0.5 persons per square kilometer.

- For each cell $c \in C$ find out the set of units E_c that overlaps c and that is hostile towards u.

- For each cell c the probability p_e that u will ambush any of its enemies $e \in E_c$ is:
$$p_e = min\Big(\frac{personnel_c}{10 \cdot cellArea}, 1\Big)$$
where $personnel_c$ is u's personnel in c and the cell area is measured in square kilometers. The assumption this model is based on is that the probability for 1000 men to perform an ambush in a cell with an area of 100 square kilometers should be 1.

- Select a unit e from E_c that should be targeted for the ambush. A unit is selected with a probability that is proportional to its total personnel.

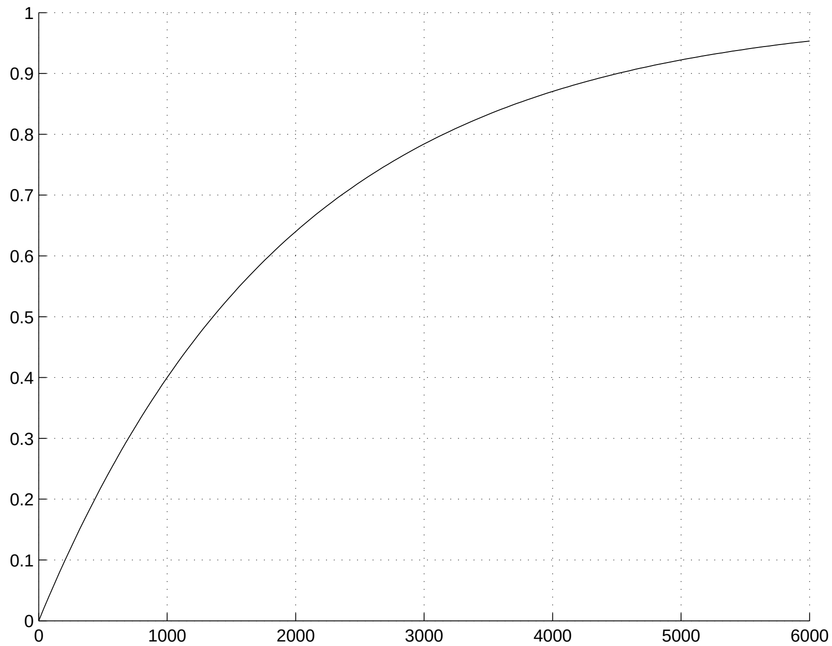


Figure 2: The probability for an ambushed unit e to find the ambushing unit u. The x-axis represents the product of the number of personnel for e and the number of persons per square kilometer for u.

- The damage d_u (in persons) suffered by the ambushing unit u is calculated as:

$$d_u = 0.05 \cdot personnel_c$$

and the damage d_e (in persons) suffered by the ambushed unit e is calculated as:

$$d_e = 4 \cdot d_u$$

If startAmbush and endAmbush are equal it should be interpreted as one single ambush performed at exactly that time. Thus for such orders an ambush is considered to be carried out during the specified time step in all cells $c \in C$.

A unit e that has been ambushed by a set of units U has during the following time step for each unit $u \in U$ the probability p_u to find u and engage it in regular combat. If we let a be the total personnel of e and b be the number of persons per square kilometer that u has then p_u is calculated as:

$$p_u = 1 - e^{-a \cdot b \cdot \gamma}$$

A plot of the probability may be seen in figure 2. The assumption this model is based on is that the probability for a unit with 1000 persons to find a unit with 1 person per square kilometer should be 0.4[5], e.g.

$$1 - e^{-1000 \cdot 1 \cdot \gamma} = 0.4$$

[5]It is further assumed that the searching unit really is of battalion size and that the search is performed in an area of 100 square kilometers. The model should thus not be considered usable for any other situations than that.

Hence the γ constant is calculated as:

$$\gamma = -\frac{\log(1-0.4)}{1000}$$

If u is found e will engage u in combat and e will then become attacker and u defender, that is e will be given an attack order and u a defend order. The combat then follows the force on force combat rules described in section 1.5. If found, u will no longer be untouchable. It will not become untouchable again until the battle is over and it has either won or lost and regained enough strength to become capable and thus able to continue executing its ambush order.

1.7 Insurgent Combat

There may be combat between a military unit u and insurgents that are hostile towards u that are located in the same cells as u overlaps. Insurgents are hostile towards u (and vice versa) if the faction u belongs to and the faction the insurgents belong to are defined as enemies in the scenario's faction list. Combat between military units and insurgents will occur in a cell if:

- If u is blue:[6]
  - There are more than 0.3 insurgents per square kilometer.
  - The quote between the number of insurgents and u's personnel is greater than 0.03.
- If u is not blue:
  - There are more than 10 insurgents per square kilometer.
  - The quote between the number of insurgents and the sum of the personnel of all units that are hostile towards the insurgents is greater than 0.1.
  - The faction the insurgents belong to constitutes more than 35 percent of the total population in the cell.

The damage that a unit u inflicts on the insurgents from faction F (in number of insurgents) if u is not involved in force on force combat is:

$$damage = modifiedStrength \cdot fractionOfDay \cdot fractionOfUnit \cdot f$$

where $fractionOfDay$ is the fraction of a day that the current time step constitutes, $fractionOfUnit$ is the fraction of the unit's personnel that is located in the cell and f is the fraction of the sum of the insurgents from all factions that combats this unit that the insurgents from faction F constitutes.

The damage (in persons) that the insurgents from faction F inflicts on a unit u in a cell is:

$$damage = insurgentStrengthFactor \cdot I^F \cdot fractionOfDay \cdot g$$

where $insurgentStrengthFactor$ is the insurgentStrengthFactor simulation parameter, I^F is the number of insurgents from faction F in the cell and g is

[6]That is if the second letter in u's symbol id code is an 'F'

the fraction of the sum of the strengths of all units in this cell that are hostile towards the faction F that u's strength constitutes.

The $fractionOfUnit$ parameter is normally determined by u's distribution over the area that u occupies. However, if u is in a critical insurgent situation u focuses with c percent of its total strength in the cells where the losses occurred during the last time step, where c is:

$$c = max(95, 100 \cdot combatReadiness)$$

The combat readiness is defined by the unit's current order as described in section 1.3.3.

2 Events

Stratmas events represents the events that take place during a simulation regardless of what else that happens. How events affects the simulation is described in greater detail in section 3. This section lists all attributes tied to events.

- *active* is a flag indicating if an event is currently executed. An event is considered active if the simulation time falls between the start and end time of the event.
- *end* is the end time of an event. When the simulation time passes this time the event will no longer be executed.
- *location* is the location where the event should take place. It may take the shape of a circle or a polygon.
- *start* is the start time of an event. The event will not be carried out until the simulation time passes this time.
- *target* is an optional attribute indicating which population faction to target. Specifying a faction means that for process variables that are divided into factions, the event will only affect the specified faction. If no faction is specified, the event will affect all factions.

In addition to the attributes defined above an event may influence zero or more process variables. The process variables that may be modified by events are listed in appendix A

3 Area Influence Model

The following sections describes the area influence model in Stratmas. This model is used for letting area orders and events affect the process variables in the grid. Since the area orders and events conceptually are very similar, we introduce the term *activity* representing them both. To avoid unnecessary repetition of the words Process Variable we also introduce the abbreviation PV.

3.1 Definitions

- E – The final effect of the modification that is then used to modify the actual PV value. More precisely it alters the value v of a PV according to:
$$v_{n+1} = v_n + E \cdot (v_{max} - v_n) \tag{1}$$
if $E > 0$ and
$$v_{n+1} = v_n + E \cdot (v_n - v_{min}) \tag{2}$$
if $E \leq 0$, where v_n, v_{n+1}, v_{min} and v_{max} is the current, new, minimum and maximum value of the PV respectively. The range of E is limited to $[-0.5, 0.5]$.

- C_A – The set of grid cells that is covered by the area A.

- s – The severity of a PV modification as given in the TacLan2 file.

- u_p – The personnel of the unit u.

- I – The set containing the following PV: *Displaced, Fraction crime victims, Perceived threat, Protected, Violence.*

- $pop_{i,c}$ – The population that should be taken into consideration when influencing PV_i $i \in I$ in cell c. If i is *Protected* then $pop_{i,c}$ maps to the number of displaced persons plus the number of protected persons, otherwise it maps to the total population.

3.2 Model Description

There are three different cases to consider in the Stratmas area influence model. These cases are described in the three following sections.

Case 1

This is the case where the activity has no performer, i.e. what is called an Event. This type of activity is used to represent the kind of events that is not connected to any actor in the simulation.

Let A be the area of the activity. Then the effect in cell $c \in C_A$ is:

$$E_c = 0.05 \cdot s \cdot frac_c$$

where $frac_c$ is the fraction of the activity's effect that affects the cell c. This fraction is determined by the distribution of the activity in the area A. This is assumed to be a normal distribution with

$$\sigma = \frac{\sqrt{|C_A|} \cdot c_{size}}{4}$$

where $|C_A|$ denotes the number of cells in C_A and c_{size} is the length in meters of the side of a cell.

Case 2

This case represents the situation where we have an area order (that is an activity that has a performer) that modifies a PV_i $i \notin I$.

Let A be the area of the performing unit u. The effect in cell $c \in C_A$ is:

$$E_c = 0.05 \cdot s \cdot frac_c$$

where $frac_c$ is the fraction of the activity's effect that affects the cell c. This fraction is determined by the distribution of the unit u over the area A.

Case 3

This case represents the situation where we have an area order (that is an activity that has a performer) that modifies a PV_i $i \in I$.

Let A be the area of the performing unit u. The effect in cell $c \in C_A$ is:

$$E_c = 0.05 \cdot \frac{10 \cdot s \cdot u_p}{pop_{i,c}} \cdot frac_c$$

where $frac_c$ is the fraction of the activity's effect that affects the cell c. This fraction is determined by the distribution of the unit u over the area A.

4 Process Variables and Grid Models

This section covers the grid related models in Stratma e.g. the models that define how the process variables (PV) in the grid are initialized and updated.

4.1 Conventions

The value of a PV for a time step t is written as the PV's name subscripted with t, for example $population_t$ means the population in a cell at time step t. The time step for the models below may not be shorter than one day. The values in a single equation refers to the same grid cell if nothing else is said explicitly.

For PVs that are divided in factions, the value for faction i is written as the PV's name superscripted with i, for example $population^i$ means the population of faction i. If no superscript is given we refer to all factions.

4.2 Definitions

This section contains some definitions that will be used in the following sections to describe the various process variables.

$cellArea$ is the area of a grid cell in square kilometers.

HDI is the Human Development Index simulation parameter.

$nfac$ is the number of ethnic factions in the simulation.

M is the modified mortality rate. In order to define it we must first define the hunger factor F as:

$$F = \begin{cases} 1 & \text{if } foodDays \geq 2 \\ \frac{1}{2 \cdot foodDays} & \text{if } foodDays < 2 \end{cases}$$

The modified mortality rate M is now defined as:

$$M = min(0.5, mortalityRate \cdot F)$$

where $mortalityRate$ is the mortality rate parameter of the disease for the simulation.

H is the high mortality rate. It is calculated as:

$$H = min(0.9 \cdot (1 - recoveryRate), 5 \cdot M)$$

where $recoveryRate$ is the recovery rate parameter of the disease for the simulation.

$atHome^i$ is the number of people of faction i that is neither displaced nor protected. It is calculated as:

$$atHome^i = population^i - displaced^i - protected^i$$

$atHomeIll^i$ is the number of people of faction i that is neither displaced nor protected and that is infected. It is calculated as:

$$atHomeIll^i = fractionInfected^i \cdot atHome^i$$

$atHomeImmune^i$ is the number of people of faction i that is neither displaced nor protected and that is immune. It is calculated as:

$$atHomeImmune^i = fractionRecovered^i \cdot atHome^i$$

$atHomeDeadD^i$ is the number of people of faction i that is neither displaced nor protected and that dies from disease. It is calculated as:

$$atHomeDeadD^i = M \cdot fractionInfected \cdot atHome$$

$atHomeDeadE^i$ is the number of people of faction i that is neither displaced nor protected and that dies from ethnic violence. It is calculated as:

$$atHomeDeadE^i = \begin{cases} \frac{3.33}{10^5} \cdot violence^i \cdot atHome^i & \text{if } violence^i > 80 \\ 0 & \text{if } violence^i \leq 80 \end{cases}$$

$shelteredIll^i$ is the number of people of faction i that is sheltered and infected. It is calculated as:

$$shelteredIll^i = fractionInfected^i \cdot sheltered^i$$

$shelteredImmune^i$ is the number of people of faction i that is sheltered and immune. It is calculated as:

$$shelteredImmune^i = fractionRecovered^i \cdot sheltered^i$$

$shelteredDeadD^i$ is the number of people of faction i that is sheltered and that dies from disease. It is calculated as:

$$shelteredDeadD^i = M \cdot fractionInfected \cdot sheltered$$

$unsheltered^i$ is the number of displaced people of faction i that is not sheltered. It is calculated as:

$$unsheltered^i = displaced^i - sheltered^i$$

$unshelteredIll^i$ is the number of displaced people of faction i that is infected. It is calculated as:

$$unshelteredIll^i = fractionInfected^i \cdot unsheltered^i$$

$unshelteredImmune^i$ is the number of displaced people of faction i that is immune. It is calculated as:

$$unshelteredImmune^i = fractionRecovered^i \cdot unsheltered^i$$

$unshelteredDeadD^i$ is the number of people of faction i that is not sheltered and that dies from disease. It is calculated as:

$$unshelteredDeadD^i = H \cdot fractionInfected \cdot unsheltered$$

$unshelteredDeadE^i$ is the number of people of faction i that is not sheltered and that dies from ethnic violence. It is calculated as:

$$unshelteredDeadE^i = \begin{cases} \frac{violence^i \cdot unsheltered^i}{10^4} & \text{if } violence^i > 80 \\ 0 & \text{if } violence^i \leq 80 \end{cases}$$

$campDelta^i$ Displaced people will under certain circumstances move to and from cells in accordance with the Refugee movement model, described in detail in section 4.6. $campDelta^i$ denotes the delta for a cell in number of displaced persons from faction i from this type of movement.

$threatDelta^i$ Displaced people will move to and from a cell due to so-called threat driven movement. Details about the threat driven movement model may be found in section 4.7. $threatDelta^i$ is the delta for a cell in number of displaced persons from faction i from this type of movement.

4.3 Process Variables

The following sections cover the various models for how to update the process variables (PV) in the grid from one time step to another. It consists of one subsection for each PV sorted in alphabetic order. Each such subsection has itself two subsections. One describing how the PV is initialized and one describing how the PV is updated from time step t to $t + 1$.

4.3.1 Daily Dead

The daily dead PV holds the number of people that has died as a consequence of diseases, ethnic violence and military engagement. However, the military engagement part of the dead is based on the number of shots fired in a cell – an indicator that is no longer valid since the combat model has changed. It is therefore always zero and hence left out from this description. The range of the $dailyDead$ PV is $[0, \infty]$.

Initialization

This PV is initialized to 0.

Update

Let D^i_{t+1} be the number of people from faction i that has died from diseases during time step $t+1$. We calculate D^i_{t+1} as:

$$D^i_{t+1} = atHomeDeadD^i_t + shelteredDeadD^i_t + unshelteredDeadD^i_t$$

Let E^i_{t+1} be the number of people from faction i that has died due to ethnic violence during time step $t+1$. Then E^i_{t+1} is calculated as:

$$E^i_{t+1} = atHomeDeadE^i_t + unshelteredDeadE^i_t$$

Finally, we calculate the daily dead PV as:

$$dailyDead^i_{t+1} = D^i_{t+1} + E^i_{t+1}$$

The total daily dead is the sum of the daily dead from all factions.

4.3.2 Disaffection

Description The range of the $disaffection$ PV is $[0, 100]$.

Initialization

This PV is initialized to 0.

Update

For faction i and time t let

$$\begin{aligned} A^i_t &= fractionNoFood_t + fractionNoWork_t + \\ &+ \frac{unsheltered^i_t}{population^i_t} + \frac{perceivedThreat^i_t}{100} \end{aligned}$$

and

$$B_t = fractionNoWater_t + fractionNoMedical_t + fractionCrimeVictims_t$$

Then the $disaffection$ for faction i for time step $t+1$ is calculated as follows:

$$disaffection^i_{t+1} = 100 \cdot \left(1 - \left(1 - \frac{A^i_t}{5}\right) \cdot \left(1 - \frac{B_t}{4}\right)\right)$$

and the total $disaffection$ is calculated according to:

$$disaffection_{t+1} = \frac{\sum_{i=1}^{nfac} disaffection^i_{t+1} \cdot population^i_t}{population_t}$$

4.3.3 Displaced

Description The range of the $displaced$ PV is $[0, \infty]$.

Initialization

This PV is initialized to 0.

Update

Let D^i_{t+1} be the number of displaced people from faction i that has died from diseases during time step $t+1$. We calculate D^i_{t+1} as:

$$D^i_{t+1} = shelteredDeadD^i_t + unshelteredDeadD^i_t$$

Let N^i_{t+1} denote the number of people from faction i that becomes displaced during time step $t+1$ due to ethnic violence. Then N^i_{t+1} is calculated as:

$$N^i_{t+1} = \begin{cases} violence^i_t \cdot 0.001 \cdot atHome^i_t & \text{if } violence^i_t > 80 \\ violence^i_t \cdot 0.0005 \cdot atHome^i_t & \text{if } 50 < violence^i_t \leq 80 \\ 0 & \text{if } violence^i_t \leq 50 \end{cases}$$

With the definitions above, the Displaced PV is updated as follows:

$$\begin{aligned} displaced^i_{t+1} &= displaced^i_t + campDelta^i_{t+1} + threatDelta^i_{t+1} + \\ &+ N^i_{t+1} - D^i_{t+1} - unshelteredDeadE^i_t \end{aligned}$$

4.3.4 Ethnic Tension

Description The range of the $ethnicTension$ PV is $[0, 1]$.

Initialization

Let V_0 denote the maximum level of violence among the factions in a cell at time step 0. Then the ethnic tension for time step 0 is calculated as.

$$ethnicTension_0 = \frac{0.01 \cdot (V + polarization_0) + fractionNoWork_0}{3}$$

It should be noticed that $polarization_0$ and $fractionNoWork_0$ is always 0 so $ethnicTension_0$ will always be 0.

Update

Let V_t denote the maximum level of violence among the factions in a cell. Let E_t denote the expected tension for time step t. It is calculated as follows:

$$E_t = \frac{0.01 \cdot (V + polarization_t) + fractionNoWork_t}{3}$$

Then the ethnicTensionfor time step $t+1$ is calculated as:

$$ethnicTension_{t+1} = ethnicTension_t + 0.05 \cdot (E_t - ethnicTension_t)$$

4.3.5 Food Days

Description The range of the $foodDays$ PV is $[0, 18]$.

Initialization

The food days PV is initialized as follows:

$$foodDays_0 = 3 + \left(15 - \lg\left(1 + \frac{population_0}{cellArea}\right)\right) \cdot HDI$$

Update

Define C_t as

$$C_t = \begin{cases} foodDays_t - 1 & \text{if } foodDays_t \geq 2 \\ \frac{foodDays_t}{2} & \text{if } foodDays_t < 2 \end{cases}$$

The food days PV is then updated as follows:

$$foodDays_{t+1} = C_t + infrastructure_t \cdot \left(3 + \left(15 - \lg\left(1 + \frac{population_t}{cellArea}\right)\right) \cdot HDI - C_t\right)$$

4.3.6 Fraction Crime Victims

Description The range of the $fractionCrimeVictims$ PV is $[0, 1]$.

Initialization

This PV is initialized to 0.

Update

Define C_t as:

$$C_t = \frac{0.01 \cdot (disaffection_t + violence_t) + fractionNoWork_t}{3}$$

Further, let D_t be:

$$D_t = \begin{cases} C_t \cdot min\left(\frac{population_t}{500 \cdot cellArea}, \ 1\right) & \text{if } \frac{population_t}{cellArea} > 100 \\ C_t & \text{if } \frac{population_t}{cellArea} \leq 100 \end{cases}$$

Then the fractionCrimeVictim PV is updated as follows:

$$\begin{aligned} fractionCrimeVictims_{t+1} &= fractionCrimeVictims_t + \\ &+ 0.2 \cdot (D_t - fractionCrimeVictims_t) \end{aligned}$$

4.3.7 Fraction Infected

Description The range of the $fractionInfected$ PV is $[0, 1]$.

Initialization

This PV is initialized to 0.

Update

Diseases spread from cell to cell according to the disease spread model described in section 4.5. Let $nearbyInfectious_t$ denote the number of infectious persons in nearby cells. Define $noWater$ as:

$$noWater = fractionNoWater_t \cdot population_t$$

Further, define $homeAtRisk_t$, $shelteredAtRisk_t$ and $unshelteredAtRisk_t$ as:

$$\begin{aligned}
homeAtRisk_t &= min(noWater_t, atHome_t - \\
&- atHomeIll_t - atHomeImmune_t) \\
shelteredAtRisk_t &= min(noWater_t, sheltered_t - \\
&- shelteredIll_t - shelteredImmune_t) \\
unshelteredAtRisk_t &= min(noWater_t, unsheltered_t - \\
&- unshelteredIll_t - unshelteredImmune_t)
\end{aligned}$$

and let

$$mixingRate = \frac{6}{cellSideKm}$$

where $cellSideKm$ is the length of the side of a cell in kilometers. We can now define I_t as:

$$I_t = \left(fractionInfected_t + \frac{mixingRate \cdot nearbyInfectious_t}{population_t} \right) \cdot infectionRate$$

where $infectionRate$ is the infection rate parameter of the disease of the simulation. I_t is then truncated as:

$$I_t = min(I_t, 0.5)$$

Then we continue by letting $homeCaught_{t+1}$, $shelteredCaught_{t+1}$ and $unshelteredCaught_{t+1}$ denote the number of persons at home, the number of sheltered and the number of unsheltered persons respectively that has become infected during time step $t+1$ and calculate them as:

$$\begin{aligned}
homeCaught_{t+1} &= min(homeAtRisk_t, \text{Poisson}(homeAtRisk_t \cdot I_t)) \\
shelteredCaught_{t+1} &= min(shelteredAtRisk_t, \\
&\quad \text{Poisson}(shelteredAtRisk_t \cdot I_t)) \\
unshelteredCaught_{t+1} &= min(unshelteredAtRisk_t, \\
&\quad \text{Poisson}(unshelteredAtRisk_t \cdot I_t))
\end{aligned}$$

where $\text{Poisson}(\lambda)$ is a number that is Poisson distributed with intensity λ. We now calculate the total number of infected persons that are at home, that are sheltered and unsheltered as:

$$\begin{aligned}
totHomeIll_{t+1} &= atHomeIll_t \cdot (1 - recoveryRate) + homeCaught_{t+1} - \\
&- atHomeDeadD_{t+1} - atHomeDeadE_{t+1} \\
totShelteredIll_{t+1} &= shelteredIll_t \cdot (1 - recoveryRate) +
\end{aligned}$$

$$
\begin{aligned}
& + shelteredCaught_{t+1} - shelteredDeadD_{t+1} \\
totUnshelteredIll_{t+1} &= unshelteredIll_t \cdot (1 - recoveryRate) + \\
& + unshelteredCaught_{t+1} - unshelteredDeadD_{t+1} - \\
& - unshelteredDeadE_{t+1}
\end{aligned}
$$

Finally we can calculate the total number of infected persons as:

$$
\begin{aligned}
totalInfected_{t+1} &= totHomeIll_{t+1} + totShelteredIll_{t+1} + \\
&+ totUnshelteredIll_{t+1}
\end{aligned}
$$

and the fraction infected as:

$$
fractionInfected_{t+1} = \frac{totalInfected_{t+1}}{population_t}
$$

4.3.8 Fraction No Food

Description The range of the $fractionNoFood$ PV is $[0, 1]$.

Initialization

This PV is initialized to 0.

Update

The fractionNoFood PV is updated as follows:

$$
fractionNoFood_{t+1} = \begin{cases} 1.3^{-foodDays_t} & \text{if } foodDays_t > 0 \\ 1 & \text{if } foodDays_t \leq 0 \end{cases}
$$

4.3.9 Fraction No Medical

Description The range of the $fractionNoMedical$ PV is $[0, 1]$.

Initialization

The fractionNoMedical PV is initialized to:

$$
fractionNoMedical_0 = (1 - HDI) \cdot \sqrt{1 - min\left(0.01 \cdot \frac{population_0}{cellArea},\ 1\right)}
$$

Update

The fractionNoMedical PV is always given the same value as in the previous time step. Thus:

$$
fractionNoMedical_{t+1} = fractionNoMedical_t
$$

4.3.10 Fraction No Water

Description The range of the $fractionNoWater$ PV is $[0, 1]$.

Initialization

This PV is initialized to 0.

Update

Define the total amount of water T in time step t as:

$$T_t = waterDays_t + \frac{suppliedWater_t}{population_t}$$

Then the fractionNoWater PV is updated as follows:

$$fractionNoWater_{t+1} = \begin{cases} 3^{-T_t} & \text{if } waterDays_t > 0 \\ 1 & \text{if } waterDays_t \leq 0 \end{cases}$$

4.3.11 Fraction No Work

Description The range of the $fractionNoWork$ PV is $[0, 1]$.

Initialization

The fractionNoWork PV is initialized to the same value as the unemployment simulation parameter.

Update

The fractionNoWork PV is always given the same value as in the previous time step. Thus:

$$fractionNoWork_{t+1} = fractionNoWork_t$$

4.3.12 Fraction Recovered

Description The range of the $fractionRecovered$ PV is $[0, 1]$.

Initialization

This PV is initialized to 0.

Update

The fractionRecovered PV is updated as follows:

$$\begin{aligned} fractionRecovered_{t+1} &= atHomeImmune_t + shelteredImmune_t + \\ &\quad + unshelteredImmune_t + recoveryRate \cdot \\ &\quad (atHomeIll_t + shelteredIll_t + unshelteredIll_t) \end{aligned}$$

where $recoveryRate$ is the recovery rate parameter of the disease for the simulation.

4.3.13 Infrastructure

Description The range of the $infrastructure$ PV is $[0, 1]$.

Initialization

The infrastructure PV is initialized as:

$$infrastructure_0 = 0.2 \cdot HDI \cdot min\left(0.01 \cdot \frac{population_0}{cellArea}, 1\right)$$

Update

The infrastructure PV is always given the same value as in the previous time step. Thus:

$$infrastructure_{t+1} = infrastructure_t$$

4.3.14 Insurgents

The insurgent PV is meant to represent the number of insurgents in a grid cell. The base assumption is that high levels of disaffection among the population generates more insurgents. The current model will increase the number of insurgents in a cell if the disaffection reaches above the *insurgentDisaffectionThreshold* and reduce the number of insurgents otherwise. Further, an assumption has been made that there cannot be more insurgents in a cell than a certain percentage of the total population. This percentage is called the *fractionPotentialInsurgent* parameter and is given as indata to the simulation. There is also another simulation parameter called the *insurgentGenerationCoefficient* that controls the magnitude of the increase or decrease. The range of the $insurgents$ PV is $[0, \infty]$.

Initialization

There is currently no insurgents at the start of a simulation.

Update

Define the maximum number of potential insurgents N_t as:

$$N_t = (F \cdot population_t^i) - insurgents_t^i$$

where F is the *Fraction potential insurgents* simulation parameter. Then the number of insurgents in a cell is updated according to:

$$insurgents_{t+1}^i = insurgents_t^i + U \cdot G \cdot (disaffection_t^i - T) \cdot N_t$$

where U is a random number uniformly distributed in the interval $[0, 0.01]$, G is the *Insurgent generation coefficient* and T is the *Insurgent disaffection threshold*.

4.3.15 Perceived Threat

Description The range of the $perceivedThreat$ PV is $[0, 100]$.

Initialization

This PV is initialized to 0.

Update

For time step t define $smoothedDead_t$ as:

$$smoothedDead_t = 0.75 \cdot smoothedDead_{t-1} + 0.25 \cdot dailyDead_{t-1}$$

assuming that $smoothedDead_0 = 0$. Further, define $mortality_t$ as:

$$mortality_t = \frac{1000 \cdot smoothedDead_t}{population_t}$$

and the $danger_t^i$ for faction i as:

$$danger_t^i = 0.1 \cdot violence_t^i \cdot \sqrt{1 - \frac{0.01 \cdot population_t^i}{population_t}}$$

Then the $dThreat$ is defined as:

$$dThreat = \frac{danger_t}{2} + mortality_t - \frac{perceivedThreat_t^i}{10} - \lg\left(1 + \frac{population_t}{cellArea}\right)$$

and thus the $perceivedThreat$ for faction i is updated as follows:

$$perceivedThreat_{t+1}^i = perceivedThreat_t^i + dThreat$$

4.3.16 Polarization

Description The range of the $polarization$ PV is $[0, 100]$.

Initialization

This PV is initialized to 0.

Update

Define $meanDisaffection_t$ as:

$$meanDisaffection_t = \frac{\sum_{i=1}^{nfac} disaffection_t^i}{nfac}$$

and $meanSquareDisaffection_t$

$$meanSquareDisaffection_t = \frac{\sum_{i=1}^{nfac} (disaffection_t^i)^2}{nfac}$$

then the $polarization$ for time step $t + 1$ is

$$polarization_{t+1} = \sqrt{meanSquareDisaffection_t - (meanDisaffection_t)^2}$$

4.3.17 Population

The Population PV represents the number of inhabitants in an area. The range of the $population$ PV is $[0, \infty]$.

Initialization

The Population PV is initialized according to the population distribution model described in section 4.4.

Update

The Population PV is updated as follows:

$$population^i_{t+1} = population^i_t - dailyDead^i_{t+1} - campDelta^i_{t+1} - threatDelta^i_{t+1}$$

4.3.18 Protected

The Protected PV represents the number of displaced people in an area that is considered to be under military protection. This number may be increased by military units that are stationed in the area and that has an order that affects the Protected PV according to the area influence model, see section 3. Only displaced people may be protected. Protected people are assumed to resettle and thereby stop beeing displaced and become ordinary inhabitants again. 0.5 percent of the protected population in an area is assumed to resettle every day. People are resettling to cells proportionally to the size of the initial population of the cells. The range of the *protected* PV is $[0, \infty]$.

Initialization

This PV is initialized to 0.

Update

As stated above, the increment - if any - of the Protected PV comes from military units via the area influence model described in section 3.

The resettling model works as follows. Let $totalResettled^i_t$ be the total amount of resettling people of faction i during time step t in the entire simulated area. Let $initialPopulation^i_c$ be the initial population of faction i in cell c and let $initialPopulation^i$ be the total initial population of faction i in the simulated area. Then in cell c the Protected PV is updated as follows;

$$protected^i_{t+1} = 0.995 \cdot protected^i_t + \frac{initialPopulation^i_c}{initialPopulation^i} \cdot totalResettled^i_t$$

4.3.19 Sheltered

Description The range of the *sheltered* PV is $[0, \infty]$.

Initialization

This PV is initialized to 0.

Update

The value of the Sheltered PV in a cell c can be modified in two ways. Either by a shelter agency team that has built a camp in c - see section *** - or by a military unit that is stationed in the area and that has an order that affects the Sheltered PV according to the area influence model, see section 3.

4.3.20 Supplied Water

Description The range of the $suppliedWater$ PV is $[0, \infty]$.

Initialization

This PV is initialized to 0.

Update

The value of the SuppliedWater PV in a cell can oly be increased by a water agency team that is stationed in the cell, see section ***.

4.3.21 Total Dead

The TotalDead PV is simply the sum of the DailyDead for all time steps up to the current simulation time. The range of the $totalDead$ PV is $[0, \infty]$.

Initialization

This PV is initialized to 0.

Update

The Total Dead PV is updated as follows:

$$totalDead_{t+1} = totalDead_t + dailyDead_{t+1}$$

4.3.22 Violence

Description The range of the $violence$ PV is $[0, 100]$.

Initialization

This PV is initialized to 0.

Update

The violence for faction i and time step $t + 1$ is calculated as:

$$violence^i_{t+1} = violence^i_t \cdot 0.75 + \text{Poisson}(0.20 \cdot disaffection^i_t)$$

where Poisson(λ) is a number that is Poisson distributed with intensity λ.
The violence for time step $t + 1$ for all factions is calculated as follows:

$$violence_{t+1} = \frac{\sum_{i=1}^{nfac} violence^i_{t+1} \cdot population^i_{t+1}}{population_{t+1}}$$

4.3.23 Water Days

Description The range of the $waterDays$ PV is $[0, 5]$.

Initialization

Define the $bestCapacity_t$ as:

$$bestCapacity_t = \frac{500 \cdot cellArea \cdot HDI}{population_t}$$

Then the Water days PV is initialized to:

$$waterDays_0 = \begin{cases} 2 & \text{if } bestCapacity_0 \leq 2 \\ bestCapacity_0 & \text{if } 2 < bestCapacity_0 < 5 \\ 5 & \text{if } bestCapacity_0 \geq 5 \end{cases}$$

Update

Define the $bestCapacity_t$ as in the previous section and the $shortfall_t$ as:

$$shortfall_t = bestCapacity_t - waterDays_t$$

Then the Water days PV is updated as follows:

$$waterDays_{t+1} = \begin{cases} waterDays_t + 0.1 \cdot HDI^2 \cdot shortfall_t & \text{if } shortfall_t > 0 \\ waterDays_t & \text{if } shortfall_t \leq 0 \end{cases}$$

4.4 Population Distribution

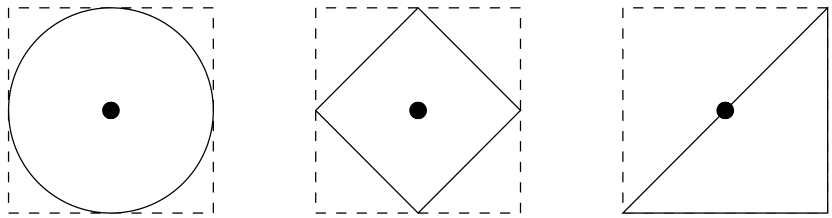
This section describes how the population from the cities provided to the Stratmas server is distributed accross the simulation grid. The basic assumption is that a city T is represented by an area A_T, a distribution D_T and a set E_T of ethnic groups that have a total population I_T. If there are n ethnic groups we refer to the i:th group as E_T^i and the number of inhabitants of group i as I_T^i where $i = 1, 2, \ldots, n$. In order to explain the distribution model we also define the set of grid cells C_T to contain all cells that is overlapped by the area A_T.

People in T are distributed across the cells in C_T based on the provided distribution D_T. For some types of distributions, such as a normal distribution, we need to know the center point of the city so we can center the distribution around that point. We define the center point P_T of T as the center of the bounding box for A_T. As may be seen in Figure 3 where dashed lines denotes bounding boxes and black dots the center points this definition of the center point is reasonable for some areas (3.a and b) but less reasonable for others (3.c).

If we let d_Q denote the distance in meters from the point Q to P_T we can define $D_T(d_Q)$ as the value of the distribution D_T at d_Q meters from its center point. For a description of how this value is calculated for different distributions, see section 5.

If we let d_c denote the distance from the center point of cell $c \in C_T$ to P_T we can calculate the number of inhabitants $N_{T,c}^i$ from ethnic group i of city T that should be placed in c as:

$$N_{T,c}^i = I_T^i \cdot \frac{D_T(d_c)}{\sum_{c \in C_T} D_T(d_c)}$$



a b c

Figure 3: Bounding boxes and center points for different city areas in Stratmas.

Defining T_{all} as the set of all cities in the simulation the total number of inhabitants I_c^i of ethnic group i that should be placed in cell c is simply:

$$I_c^i = \sum_{T \in T_{all}} N_{T,c}^i$$

4.5 Disease Spreading

People in a cell c_0 may be incefcted by people in a set of neighboring cells C. We define C as follows. Let

$$a = Round\left(\frac{50}{cellSideKm}\right)$$

where $cellSideKm$ is the length of the side of a cell in kilometers. C is then made up of the cells in a square with side $2 \cdot a + 1$ cells centered around c_0 excluding c_0, see Figure 4. If we let row_0 and col_0 denote the row and column of c_0 and row_c and col_c the row and column of $c \in C$ then we can calculate the weight w_c for cell c as:

$$w_c = \frac{e^{-\frac{\sqrt{(row_0 - row_c)^2 + (col_o - col_c)^2} \cdot cellSideKm}{30}}}{30}$$

Then $nearbyInfectious$ as described in section 4.3.7 for cell c_0 is calculated as:

$$\sum_{c \in C} w_c \cdot fractionInfected_c \cdot population_c$$

where $fractionInfected_c$ and $population_c$ are the values for cell c.

4.6 Refugee Movement

Displaced people in Stratmas may move towards refugee camps if there are any. Refugee camps are built by shelter agency teams. This process is described in greater detail in section ***.

If there are at least one refugee camp present in the simulation displaced people in a cell c may move towards it. If there are more than one camp people move towards the nearest one. People is assumed to move with an average speed

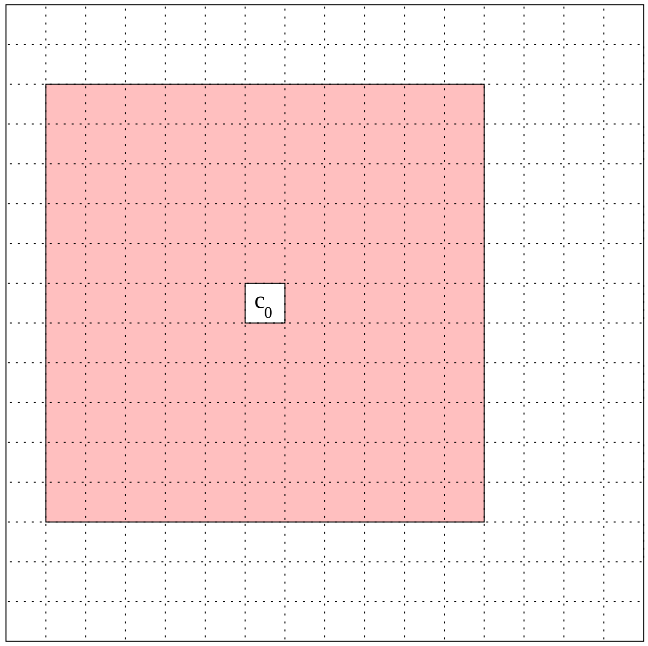


Figure 4: The shadowed area visualizes the cells in C when the side of a cell is 10 kilometers.

of 30 kilometers per day. The number of people $movingPop$ moving from c is calculated as:

$$movingPop = 0.4 \cdot e^{-0.01 \cdot d} \cdot unsheltered$$

where d is the distance in kilometers between the center point of c and the nearest refugee camp. Let row_c and col_c denote the row and column of c and row_T and col_T the row and column of the target cell T e.g. the cell in which the camp is located. We can now define the row and column of the cell a as:

$$\begin{aligned} row_a &= row_c + min\left(1, \frac{30}{d}\right) \cdot (row_T - row_c) \\ col_a &= col_c + min\left(1, \frac{30}{d}\right) \cdot (col_T - col_c) \end{aligned}$$

The greater part of the moving population from c will thus reach the cell a during this time step. However, to display diffusion the people are spread around a using an approximate normal distribution.

Let S be the set of cells that are neighbors to c according to Figure 5 and let σ be a diffusion measure set to 15 kilometers. Further, define N_s as:

$$N_s = e^{-\frac{1}{2 \cdot \sigma^2} \cdot D_s^2}$$

where $s \in S$ and D_s is the distance from the center of s to the center of a. The number of persons P_s moving to $s \in S$ is now calculated as:

$$P_s = movingPop \cdot \frac{N_s}{1 + \sum_{i \in S} N_i}$$

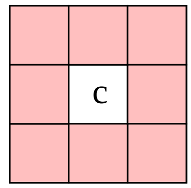


Figure 5: The shadowed cells are considered neighbors of c.

and the number of persons P_a moving to a is:

$$P_a = movingPop \cdot \frac{1}{1 + \sum_{i \in S} N_i}$$

4.7 Threat Driven Movement

If the value of the perceived threat process variable in a cell is high, people may move to a neighboring cell where the perceived threat is lower. Figure 5 shows which cells that are considered neighbors of a cell c.

People may move from a cell c if $perceivedThreat_c > 65$ and there is a neighboring cell n such that $perceivedThreat_n < perceivedThreat_c$. The target cell n will be the one cell among the neighboring cells that has the lowest perceived threat.

The number of persons A^i from faction i that moves from cell c to n is calculated as:

$$A^i = 0.2 \cdot unsheltered_c^i + 0.1 \cdot atHome_c^i$$

5 Distributions

References

[1] Ahlin, Daniel: *Implementation comments on the technical note "A Force Deployment and Combat Attrition Model for STRATMAS Phase V.6"*

[2] Thorssell, Åke: *STRATMAS STRIDSMODELL, dokument 1*

[3] Woodcock, Ted: *A Force Deployment and Combat Attrition Model for STRATMAS Phase V.6*

[4] *APP-6A Military Symbols for Land Based Systems*

[5] *Department of Defense - Interface Standard - Common Warfighting Symbology*

A Area Order Modifiable Process Variables

The list below contains the names of all process variables that may be modified by area orders.

- disaffection
- displaced
- ethnicTension
- foodDays
- fractionCrimeVictims
- fractionInfected
- fractionNoMedical
- fractionNoWork
- fractionRecovered
- infrastructure
- insurgents
- perceivedThreat
- protected
- sheltered
- violence
- waterDays