available_commodities__on_site[1](t) = available_commodities__on_site[1](t - dt) + (arrival_at_the_delivery_points[vehicle_type] - inquired_amount[vehicle_type]) * dt
INIT available_commodities__on_site[1] = init_inventory_on_site[1]

available_commodities__on_site[2](t) = available_commodities__on_site[2](t - dt) + (arrival_at_the_delivery_points[vehicle_type] - inquired_amount[vehicle_type]) * dt
INIT available_commodities__on_site[2] = init_inventory_on_site[2]

available_commodities__on_site[3](t) = available_commodities__on_site[3](t - dt) + (arrival_at_the_delivery_points[3] - inquired_amount[vehicle_type]) * dt
INIT available_commodities__on_site[3] = init_inventory_on_site[3]

INFLOWS:

- arrival_at_the_delivery_points[1] = SMTH1(supply_rate[1], transport__delay)
- arrival_at_the_delivery_points[2] = SMTH1(supply_rate[2], transport__delay)
- arrival_at_the_delivery_points[3] = SMTH1(supply_rate[3], transport__delay)

OUTFLOWS:

- inquired_amount[1] = demand[1]+add_demand[1]+need_of_relief_units__and_evacuees
- inquired_amount[2] = demand[2]+add_demand[2]+need_of_relief_units__and_evacuees
- inquired_amount[3] = demand[3]+add_demand[3]+need_of_relief_units__and_evacuees

avg_rate_of_consumption[1](t) = avg_rate_of_consumption[1](t - dt) + (change_in_the_rate_of_consumption[vehicle_type]) * dt
INIT avg_rate_of_consumption[1] = 475

avg_rate_of_consumption[2](t) = avg_rate_of_consumption[2](t - dt) + (change_in_the_rate_of_consumption[vehicle_type]) * dt
INIT avg_rate_of_consumption[2] = 475

avg_rate_of_consumption[3](t) = avg_rate_of_consumption[3](t - dt) + (change_in_the_rate_of_consumption[vehicle_type]) * dt
INIT avg_rate_of_consumption[3] = 475

INFLOWS:

- change_in_the_rate_of_consumption[vehicle_type] = (inquired_amount-avg_rate_of_consumption)/time_to_react_wait_and_see

commodities_in_transport[1](t) = commodities_in_transport[1](t - dt) + (supply_rate[vehicle_type] - arrival_at_the_delivery_points[vehicle_type]) * dt
INIT commodities_in_transport[1] = 475

commodities_in_transport[2](t) = commodities_in_transport[2](t - dt) + (supply_rate[vehicle_type] - arrival_at_the_delivery_points[vehicle_type]) * dt
INIT commodities_in_transport[2] = 475

commodities_in_transport[3](t) = commodities_in_transport[3](t - dt) + (supply_rate[vehicle_type] - arrival_at_the_delivery_points[3]) * dt
INIT commodities_in_transport[3] = 475

INFLOWS:

- supply_rate[1] = number_of__vehicles[1]*capacity_per_vehicle_type[1]
- supply_rate[2] = number_of__vehicles[2]*capacity_per_vehicle_type[2]
- supply_rate[3] = number_of__vehicles[3]*capacity_per_vehicle_type[3]

OUTFLOWS:

- arrival_at_the_delivery_points[1] = SMTH1(supply_rate[1], transport__delay)
- arrival_at_the_delivery_points[2] = SMTH1(supply_rate[2], transport__delay)
- arrival_at_the_delivery_points[3] = SMTH1(supply_rate[3], transport__delay)

measures_conducted[Evacuation_of_victims](t) = measures_conducted[Evacuation_of_victims](t - dt) + (conducting_measures[Aktivität]) * dt
INIT measures_conducted[Evacuation_of_victims] = 0

measures_conducted[Distribution_of_goods](t) = measures_conducted[Distribution_of_goods](t - dt) + (conducting_measures[Aktivität]) * dt
INIT measures_conducted[Distribution_of_goods] = 0

INFLOWS:

- conducting_measures[Evacuation_of_victims] = smth1(number_of_relief_units_conducting_measures[Evacuation_of_victims]*performance_per_hour_and_person[Evacuation_of_victims], sum_delay_evac, 0)

- conducting_measures[Distribution_of_goods] = smth1(number_of_relief_units_conducting_measures[Distribution_of_goods]*performance_per_hour_and_person[Distribution_of_goods], sum_delay, 0)

number_of_measures_units_that_have_to_be_conducted[Evacuation_of_victims](t) = number_of_measures_units_that_have_to_be_conducted[Evacuation_of_victims](t - dt) + (need_for_measures[Aktivität] - conducting_measures[Aktivität]) * dt
INIT number_of_measures_units_that_have_to_be_conducted[Evacuation_of_victims] = number_of_measure__units_that_have_to_be_conducted[Evacuation_of_victims]

number_of_measures_units_that_have_to_be_conducted[Distribution_of_goods](t) = number_of_measures_units_that_have_to_be_conducted[Distribution_of_goods](t - dt) + (need_for_measures[Aktivität] - conducting_measures[Aktivität]) * dt
INIT number_of_measures_units_that_have_to_be_conducted[Distribution_of_goods] = number_of_measure__units_that_have_to_be_conducted[Distribution_of_goods]

INFLOWS:

need_for_measures[Evacuation_of_victims] = Pulse(number_of_measure__units_that_have_to_be_conducted[Evacuation_of_victims],Stoptime)

need_for_measures[Distribution_of_goods] = Pulse(number_of_measure__units_that_have_to_be_conducted[Distribution_of_goods],cm_water_gauge_start_to_conduct_measures[Distribution_of_goods],STOPTIME)

OUTFLOWS:

conducting_measures[Evacuation_of_victims] = smth1(number_of_relief_units_conducting_measures[Evacuation_of_victims]*performance_per_hour_and_person[Evacuation_of_victims], sum_delay_evac, 0)

conducting_measures[Distribution_of_goods] = smth1(number_of_relief_units_conducting_measures[Distribution_of_goods]*performance_per_hour_and_person[Distribution_of_goods], sum_delay, 0)

number_of_relief_units_conducting_measures[Evacuation_of_victims](t) = number_of_relief_units_conducting_measures[Evacuation_of_victims](t - dt) + (relief_forces_allocated_to__response_operation[Aktivität] - back_to_stock_from_measure[Aktivität]) * dt

INIT number_of_relief_units_conducting_measures[Evacuation_of_victims] = 0

number_of_relief_units_conducting_measures[Distribution_of_goods](t) = number_of_relief_units_conducting_measures[Distribution_of_goods](t - dt) + (relief_forces_allocated_to__response_operation[Aktivität] - back_to_stock_from_measure[Aktivität]) * dt

INIT number_of_relief_units_conducting_measures[Distribution_of_goods] = 0

INFLOWS:

relief_forces_allocated_to__response_operation[Evacuation_of_victims] = Smth1(discrepancy[Evacuation_of_victims]/(performance_per_hour_and_person[Evacuation_of_victims]*exhausting_factor), delay_due_to_transportation+time_needed_for__decision, 0)

relief_forces_allocated_to__response_operation[Distribution_of_goods] = SMTH1(discrepancy[Distribution_of_goods]/(performance_per_hour_and_person[Distribution_of_goods]*exhausting_factor), delay_due_to_transportation+time_needed_for__decision, 0)

OUTFLOWS:

back_to_stock_from_measure[Evacuation_of_victims] = (number_of_relief_units_conducting_measures[Evacuation_of_victims])/(working_time[Evacuation_of_victims])

back_to_stock_from_measure[Distribution_of_goods] = (number_of_relief_units_conducting_measures[Distribution_of_goods])/(working_time[Distribution_of_goods])

number_of__vehicles[1](t) = number_of__vehicles[1](t - dt) + (vehicle_request[vehicle_type] - used__vehicles[vehicle_type]) * dt

INIT number_of__vehicles[1] = 22

number_of__vehicles[2](t) = number_of__vehicles[2](t - dt) + (vehicle_request[vehicle_type] - used__vehicles[vehicle_type]) * dt

INIT number_of__vehicles[2] = 22

number_of__vehicles[3](t) = number_of__vehicles[3](t - dt) + (vehicle_request[vehicle_type] - used__vehicles[vehicle_type]) * dt

INIT number_of__vehicles[3] = 22

INFLOWS:

vehicle_request[vehicle_type] = (adjustment_of_vehicles+add_need_for_vehicles)

OUTFLOWS:

used__vehicles[vehicle_type] = (number_of__vehicles/possible_driving_time)

relief_units_stock(t) = relief_units_stock(t - dt) + (back_to_stock_from_measure[Aktivität] + back_to_stock_from_measure[Aktivität] - relief_forces_allocated_to__response_operation[Aktivität] - relief_forces_allocated_to__response_operation[Aktivität] - loss_of_volunteers_due_to_long_lasting_disaster) * dt

INIT relief_units_stock = 50000

INFLOWS:

back_to_stock_from_measure[Evacuation_of_victims] = (number_of_relief_units_conducting_measures[Evacuation_of_victims])/(working_time[Evacuation_of_victims])

back_to_stock_from_measure[Distribution_of_goods] = (number_of_relief_units_conducting_measures[Distribution_of_goods])/(working_time[Distribution_of_goods])

back_to_stock_from_measure[Evacuation_of_victims] = (number_of_relief_units_conducting_measures[Evacuation_of_victims])/(working_time[Evacuation_of_victims])

back_to_stock_from_measure[Distribution_of_goods] = (number_of_relief_units_conducting_measures[Distribution_of_goods])/(working_time[Distribution_of_goods])

OUTFLOWS:

relief_forces_allocated_to__response_operation[Evacuation_of_victims] = Smth1(discrepancy[Evacuation_of_victims]/(performance_per_hour_and_person[Evacuation_of_victims]*exhausting_factor), delay_due_to_transportation+time_needed_for__decision, 0)

relief_forces_allocated_to__response_operation[Distribution_of_goods] = SMTH1(discrepancy[Distribution_of_goods]/(performance_per_hour_and_person[Distribution_of_goods]*exhausting_factor), delay_due_to_transportation+time_needed_for__decision, 0)

relief_forces_allocated_to__response_operation[Evacuation_of_victims] = Smth1(discrepancy[Evacuation_of_victims]/(performance_per_hour_and_person[Evacuation_of_victims]*exhausting_factor), delay_due_to_transportation+time_needed_for__decision, 0)

relief_forces_allocated_to__response_operation[Distribution_of_goods] = SMTH1(discrepancy[Distribution_of_goods]/(performance_per_hour_and_person[Distribution_of_goods]*exhausting_factor), delay_due_to_transportation+time_needed_for__decision, 0)

loss_of_volunteers_due_to_long_lasting_disaster = IF (end_of_need_for__distribution=1) then (relief_units_stock*availability_of_volunteers)/8 else 0

water_level_stock(t) = water_level_stock(t - dt) + (inflow_water_level - outflow_water_level) * dt

INIT water_level_stock = 0

INFLOWS:
inflow_water_level = water_level*1+SMTH1(water_level*increase_of_water_level*duration_of_flood, 48, 0)
OUTFLOWS:
outflow_water_level = delay(inflow_water_level,1,0)
weather__condition(t) = weather__condition(t - dt) + (increase_precipitation - decrease_precipitation) * dt
INIT weather__condition = 0
INFLOWS:
increase_precipitation = ((precipitation+precipitation2+precipitation3)/3)/10*increasing_bad_conditions
OUTFLOWS:
decrease_precipitation = Delay(increase_precipitation, 1, 0)
additional_amount_of_common_demand[1] = effect_1
additional_amount_of_common_demand[2] = effect_2
additional_amount_of_common_demand[3] = effect_3
add_demand[1] = if reaction[1] = 1 then (increase[1]) else 0
add_demand[2] = if reaction[2] = 1 then (increase[2]) else 0
add_demand[3] = if reaction[3] = 1 then (increase[3]) else 0
add_need_for_vehicles[vehicle_type] = desired_amount_of_vehicles-number_of__vehicles
adjustment_of_vehicles[vehicle_type] = SMTH1(used__vehicles, resting_time_for_drivers, 0)
availability_of_volunteers = GRAPH(TIME)

(0.00, 0.083), (2.06, 0.0836), (4.11, 0.0842), (6.17, 0.0849), (8.23, 0.0855), (10.3, 0.0861), (12.3, 0.0867), (14.4, 0.0873), (16.5, 0.088), (18.5, 0.0886), (20.6, 0.0892), (22.6, 0.0905), (24.7, 0.0918), (26.7, 0.0931), (28.8, 0.0945), (30.9, 0.0958), (32.9, 0.0971), (35.0, 0.0984), (37.0, 0.0992), (39.1, 0.0999), (41.1, 0.101), (43.2, 0.102), (45.3, 0.102), (47.3, 0.103), (49.4, 0.104), (51.4, 0.105), (53.5, 0.105), (55.5, 0.106), (57.6, 0.107), (59.7, 0.108), (61.7, 0.109), (63.8, 0.111), (65.8, 0.112), (67.9, 0.113), (69.9, 0.115), (72.0, 0.116), (74.1, 0.118), (76.1, 0.119), (78.2, 0.12), (80.2, 0.122), (82.3, 0.123), (84.3, 0.125), (86.4, 0.126), (88.5, 0.127), (90.5, 0.129), (92.6, 0.13), (94.6, 0.132), (96.7, 0.133), (98.7, 0.135), (101, 0.136), (103, 0.137), (105, 0.139), (107, 0.14)...
average_demand_of_commodities[1] = LOGNORMAL(average__demand[1], average__demand[1]* 0.005)
average_demand_of_commodities[2] = LOGNORMAL(average__demand[2], average__demand[2]* 0.005)
average_demand_of_commodities[3] = LOGNORMAL(average__demand[3], average__demand[3]* 0.005)
average_hourly_need_for__the_population = 0.00033
average__demand[vehicle_type] = average_hourly_need_for__the_population*(inhabitants-share__that_is_evacuated)
avg_hourly_need_for__evacuees = 0.000625
avg_hourly_need_for__response_operations = 0.0033
capacity_per_vehicle_type[1] = 22
capacity_per_vehicle_type[2] = 22
capacity_per_vehicle_type[3] = 22
cm_water_gauge_start_to_conduct_measures[Aktivität] = 0
coordination_witihin_the_relief_forces[Aktivität] = GRAPH(number_of_relief_units_conducting_measures)

(0.00, 0.00), (50.0, 0.18), (100, 0.5), (150, 0.98), (200, 1.30), (250, 1.60), (300, 1.92), (350, 2.28), (400, 2.70), (450, 3.26), (500, 3.94)
delay_due_ot__lacking_mobility = GRAPH(mobility_of_the__population)

(0.00, 0.535), (0.1, 0.903), (0.2, 1.22), (0.3, 1.52), (0.4, 1.80), (0.5, 2.18), (0.6, 2.36), (0.7, 2.83), (0.8, 3.28), (0.9, 3.81), (1.00, 3.98)
delay_due_to_lacking_cooperation_population = GRAPH(not_cooperative_share_of_population)

(5.00, 0.535), (5.50, 0.798), (6.00, 1.02), (6.50, 1.20), (7.00, 1.64), (7.50, 2.02), (8.00, 2.39), (8.50, 2.88), (9.00, 3.39), (9.50, 3.76), (10.0, 4.00)
delay_due_to_lacking_coordination_relief[Aktivität] = GRAPH(coordination_witihin_the_relief_forces)

(0.00, 0.107), (0.1, 0.271), (0.2, 0.457), (0.3, 0.579), (0.4, 0.829), (0.5, 1.06), (0.6, 1.31), (0.7, 1.56), (0.8, 1.76), (0.9, 1.91), (1.00, 2.00)
delay_due_to_road_condition = GRAPH(road__condition)

(0.00, 40.0), (0.1, 40.0), (0.2, 39.4), (0.3, 37.6), (0.4, 35.1), (0.5, 29.6), (0.6, 21.9), (0.7, 14.8), (0.8, 8.00), (0.9, 3.26), (1.00, 0.15)
delay_due_to_transportation = delay_due_to__circumnavigate_the_flooded_area+delay_due_to_road_condition

delay_due_to__circumnavigate_the_flooded_area = GRAPH(flooded_area)
(0.00, 1.23), (0.1, 2.69), (0.2, 4.60), (0.3, 7.67), (0.4, 12.5), (0.5, 22.6), (0.6, 28.9), (0.7, 35.0), (0.8, 38.8), (0.9, 39.9), (1.00, 40.0)

delivery_amount[vehicle_type] = avg_rate_of_consumption+desired_amount_for_delivery
demand[1] = average_demand_of_commodities[1]
demand[2] = average_demand_of_commodities[2]
demand[3] = average_demand_of_commodities[3]
desired_amount_for_delivery[vehicle_type] = (avg_rate_of_consumption*transport_duration-available_commodities__on_site)/replenishment__time
desired_amount_of_vehicles[vehicle_type] = smth1 (delivery_amount/estimated_capacity,planning_delay,0)
discrepancy[Evacuation_of_victims] = if water_level_stock>cm_water_gauge_start_to_conduct_measures[Evacuation_of_victims] then (number_of_measures_units_that_have_to_be_conducted[Evacuation_of_victims]-measures_conducted[Evacuation_of_victims]) else 0
discrepancy[Distribution_of_goods] = if water_level_stock>cm_water_gauge_start_to_conduct_measures[Distribution_of_goods] then (number_of_measures_units_that_have_to_be_conducted[Distribution_of_goods]-measures_conducted[Distribution_of_goods]) else 0
duration_of_flood = if time >from AND time<to then 1 else 0
effect_1 = GRAPH(available_commodities__on_site[1])
(0.00, 28.0), (1296, 26.0), (2592, 23.4), (3888, 20.3), (5184, 16.6), (6480, 12.9), (7776, 8.81), (9072, 6.22), (10368, 4.36), (11664, 1.97), (12960, 0.00)

effect_2 = GRAPH(available_commodities__on_site[2])
(0.00, 27.1), (1296, 24.5), (2592, 22.3), (3888, 18.9), (5184, 16.8), (6480, 14.0), (7776, 11.0), (9072, 9.33), (10368, 6.53), (11664, 2.28), (12960, 0.00)

effect_3 = GRAPH(available_commodities__on_site[3])
(0.00, 27.0), (1296, 24.2), (2592, 21.3), (3888, 17.5), (5184, 14.9), (6480, 12.6), (7776, 9.64), (9072, 6.84), (10368, 4.04), (11664, 1.45), (12960, 0.00)

end_of_need_for__distribution = if time >504 then 1 else 0
estimated_capacity[1] = 22
estimated_capacity[2] = 22
estimated_capacity[3] = 22
evacuated_people_need_for_beeing_supplied = number_of_measure__units_that_have_to_be_conducted[Distribution_of_goods]
exhausting_factor = GRAPH(TIME)
(0.00, 0.998), (1.03, 0.998), (2.06, 0.998), (3.09, 0.998), (4.11, 0.998), (5.14, 0.998), (6.17, 0.998), (7.20, 0.998), (8.23, 0.997), (9.26, 0.997), (10.3, 0.997), (11.3, 0.997), (12.3, 0.997), (13.4, 0.997), (14.4, 0.997), (15.4, 0.997), (16.5, 0.997), (17.5, 0.996), (18.5, 0.996), (19.5, 0.996), (20.6, 0.996), (21.6, 0.996), (22.6, 0.996), (23.7, 0.996), (24.7, 0.996), (25.7, 0.996), (26.7, 0.996), (27.8, 0.995), (28.8, 0.995), (29.8, 0.995), (30.9, 0.995), (31.9, 0.995), (32.9, 0.995), (33.9, 0.995), (35.0, 0.995), (36.0, 0.995), (37.0, 0.994), (38.1, 0.994), (39.1, 0.994), (40.1, 0.994), (41.1, 0.994), (42.2, 0.994), (43.2, 0.994), (44.2, 0.994), (45.3, 0.994), (46.3, 0.993), (47.3, 0.993), (48.3, 0.993), (49.4, 0.993), (50.4, 0.993), (51.4, 0.993), (52.5, 0.993), (53.5, 0.993)...

flooded_area = GRAPH(water_level_stock)
(0.00, 0.00), (120, 0.215), (240, 0.359), (360, 0.437), (480, 0.507), (600, 0.563), (720, 0.607), (840, 0.67), (960, 0.733), (1080, 0.826), (1200, 0.9)

from = 48
increase[vehicle_type] = step ((additional_amount_of_common_demand/100)*average_demand_of_commodities, 0)
increase_of_water_level = 0.5
increasing_bad_conditions = 1
inhabitants = 1636287
init_inventory_on_site[vehicle_type] = 1
mobility_of_the__population = random(0,1)
need_of_relief_units__and_evacuees = ((evacuated_people_need_for_beeing_supplied*avg_hourly_need_for__evacuees)+(avg_hourly_need_for__response_operations*sum_of_relief_units))*(OffOn)
not_cooperative_share_of_population = random(5,10)
number_of_measure__units_that_have_to_be_conducted[Evacuation_of_victims] = 20400
number_of_measure__units_that_have_to_be_conducted[Distribution_of_goods] = (inhabitants*0.12)
OffOn = 0
performance_per_hour_and_person[Evacuation_of_victims] = LOGNORMAL(10,10*0.2)
performance_per_hour_and_person[Distribution_of_goods] = LOGNORMAL(10,10*0.2)
planning_delay[vehicle_type] = RANDOM (0.5, 2)

possible_driving_time[vehicle_type] = 11

precipitation = GRAPH(TIME)

(0.00, 0.00), (2.79, 0.00), (5.57, 0.00), (8.36, 0.00), (11.1, 0.00), (13.9, 0.00), (16.7, 0.00), (19.5, 0.00), (22.3, 0.00), (25.1, 0.00), (27.9, 0.00), (30.6, 0.00), (33.4, 0.00), (36.2, 0.00), (39.0, 0.00), (41.8, 0.00), (44.6, 0.00), (47.4, 0.00), (50.1, 0.00), (52.9, 0.00), (55.7, 0.00), (58.5, 0.00), (61.3, 0.00), (64.1, 0.00), (66.9, 0.00), (69.6, 0.00), (72.4, 0.00), (75.2, 0.00), (78.0, 0.00), (80.8, 0.00), (83.6, 0.00), (86.4, 0.00), (89.1, 0.00), (91.9, 0.00), (94.7, 0.00), (97.5, 0.00), (100, 0.00), (103, 0.00), (106, 1.00), (109, 0.00), (111, 0.00), (114, 0.00), (117, 0.00), (120, 0.00), (123, 0.00), (125, 0.00), (128, 4.00), (131, 4.00), (134, 1.00), (136, 0.00), (139, 0.00), (142, 0.00), (145, 0.00)...

precipitation2 = GRAPH(TIME)

(0.00, 0.00), (2.79, 1.00), (5.57, 0.00), (8.36, 0.00), (11.1, 0.00), (13.9, 0.00), (16.7, 0.00), (19.5, 0.00), (22.3, 0.00), (25.1, 0.00), (27.9, 0.00), (30.6, 0.00), (33.4, 0.00), (36.2, 0.00), (39.0, 1.00), (41.8, 0.00), (44.6, 0.00), (47.4, 0.00), (50.1, 0.00), (52.9, 0.00), (55.7, 0.00), (58.5, 1.00), (61.3, 0.00), (64.1, 0.00), (66.9, 1.00), (69.6, 0.00), (72.4, 0.00), (75.2, 1.00), (78.0, 0.00), (80.8, 1.00), (83.6, 0.00), (86.4, 77.0), (89.1, 0.00), (91.9, 1.00), (94.7, 0.00), (97.5, 0.00), (100, 0.00), (103, 0.00), (106, 6.00), (109, 6.00), (111, 1.00), (114, 0.00), (117, 0.00), (120, 0.00), (123, 1.00), (125, 0.00), (128, 0.00), (131, 0.00), (134, 0.00), (136, 0.00), (139, 0.00), (142, 0.00), (145, 0.00)...

precipitation3 = GRAPH(TIME)

(0.00, 1.00), (2.79, 3.00), (5.57, 0.00), (8.36, 0.00), (11.1, 0.00), (13.9, 0.00), (16.7, 0.00), (19.5, 0.00), (22.3, 0.00), (25.1, 0.00), (27.9, 0.00), (30.6, 0.00), (33.4, 0.00), (36.2, 0.00), (39.0, 1.00), (41.8, 0.00), (44.6, 0.00), (47.4, 0.00), (50.1, 2.00), (52.9, 1.00), (55.7, 0.00), (58.5, 0.00), (61.3, 0.00), (64.1, 0.00), (66.9, 0.00), (69.6, 0.00), (72.4, 0.00), (75.2, 0.00), (78.0, 0.00), (80.8, 0.00), (83.6, 0.00), (86.4, 0.00), (89.1, 0.00), (91.9, 0.00), (94.7, 0.00), (97.5, 0.00), (100, 0.00), (103, 2.00), (106, 1.00), (109, 0.00), (111, 0.00), (114, 0.00), (117, 0.00), (120, 0.00), (123, 1.00), (125, 4.00), (128, 5.00), (131, 1.00), (134, 0.00), (136, 0.00), (139, 0.00), (142, 0.00), (145, 0.00)...

reaction[1] = if (available_commodities__on_site[1] <5000) then 1 else 0

reaction[2] = if (available_commodities__on_site[2] <5000) then 1 else 0

reaction[3] = if (available_commodities__on_site[3] <5000) then 1 else 0

replenishment__time[vehicle_type] = 12

resting_time_for_drivers[vehicle_type] = 10

road__condition = GRAPH(weather__condition)

(0.00, 1.00), (0.75, 0.944), (1.50, 0.822), (2.25, 0.607), (3.00, 0.407), (3.75, 0.319), (4.50, 0.241), (5.25, 0.185), (6.00, 0.115), (6.75, 0.067), (7.50, 0.019)

share__that_is_evacuated = 0.12*inhabitants

sum_delay = delay_due_to_transportation+delay_due_to_lacking_coordination_relief[Distribution_of_goods]

sum_delay_evac = delay_due_to_lacking_cooperation_population+delay_due_ot__lacking_mobility+delay_due_to_lacking_coordination_relief[Evacuation_of_victims]+delay_due_to_transportation

sum_of_relief_units = number_of_relief_units_conducting_measures[Evacuation_of_victims]+number_of_relief_units_conducting_measures[Distribution_of_goods]

time_needed_for__decision = random(0.5,2)

time_to_react_wait_and_see[vehicle_type] = 12

to = 142

transport_duration[1] = 4

transport_duration[2] = 4

transport_duration[3] = 4

transport__delay = delay_due_to_transportation

water_level = GRAPH(TIME)

(0.00, 256), (0.0574, 256), (0.115, 256), (0.172, 256), (0.23, 256), (0.287, 256), (0.345, 256), (0.402, 256), (0.459, 256), (0.517, 256), (0.574, 256), (0.632, 256), (0.689, 256), (0.747, 256), (0.804, 256), (0.862, 256), (0.919, 256), (0.976, 256), (1.03, 256), (1.09, 256), (1.15, 255), (1.21, 255), (1.26, 255), (1.32, 255), (1.38, 255), (1.44, 255), (1.49, 254), (1.55, 254), (1.61, 254), (1.67, 254), (1.72, 254), (1.78, 254), (1.84, 253), (1.90, 253), (1.95, 253), (2.01, 252), (2.07, 252), (2.13, 252), (2.18, 252), (2.24, 252), (2.30, 252), (2.35, 251), (2.41, 251), (2.47, 251), (2.53, 251), (2.58, 251), (2.64, 251), (2.70, 251), (2.76, 251), (2.81, 251), (2.87, 250), (2.93, 250), (2.99, 250)...

working_time[Evacuation_of_victims] = 8

working_time[Distribution_of_goods] = 8