Social networks module

J.Hackney IVT/ETH

MATSim User Meeting 23.04.2009

TU Berlin

Social Nets in MATSim

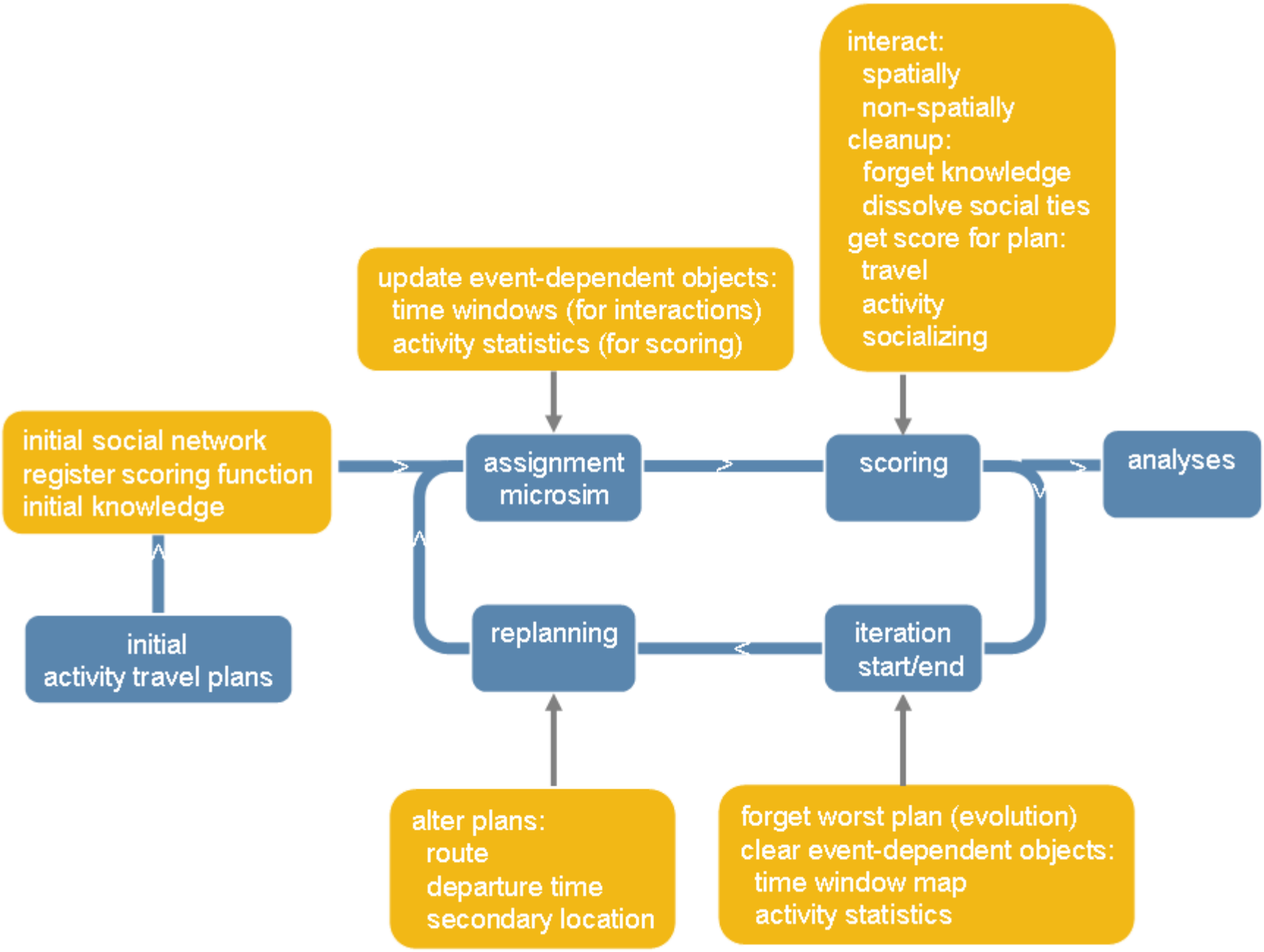
- ... Created or evolving with activity travel
- ... Used as a substrate for generating joint activity travel

MATSim vs. Behavior Model

- Result vs. Process
- MATSim is a search algorithm
  - Random changes to plans and culling worst individuals
  - "Optimality" surface changes with feedback
  - "Strategy" == dimension in which to search utility space
    - Time/duration
    - Route
    - Location
- "Behavior": decision-making
  - "Strategy" may include risk perception and management, various time horizons
  - Constrained information and uncertainty about information
  - Reinforcement, habits
  - Learning (targeted search in solution space)

Conflicting model goals

- MATSim agents do not experience time (beyond the timespan of the plan), so they cannot learn
- Building a social network (process) takes time and repeated efforts
- → Social networks should be an initial condition and a static parameter for MATSim's goals
- Building social relationships is another model and should be done outside MATSim optimization loop

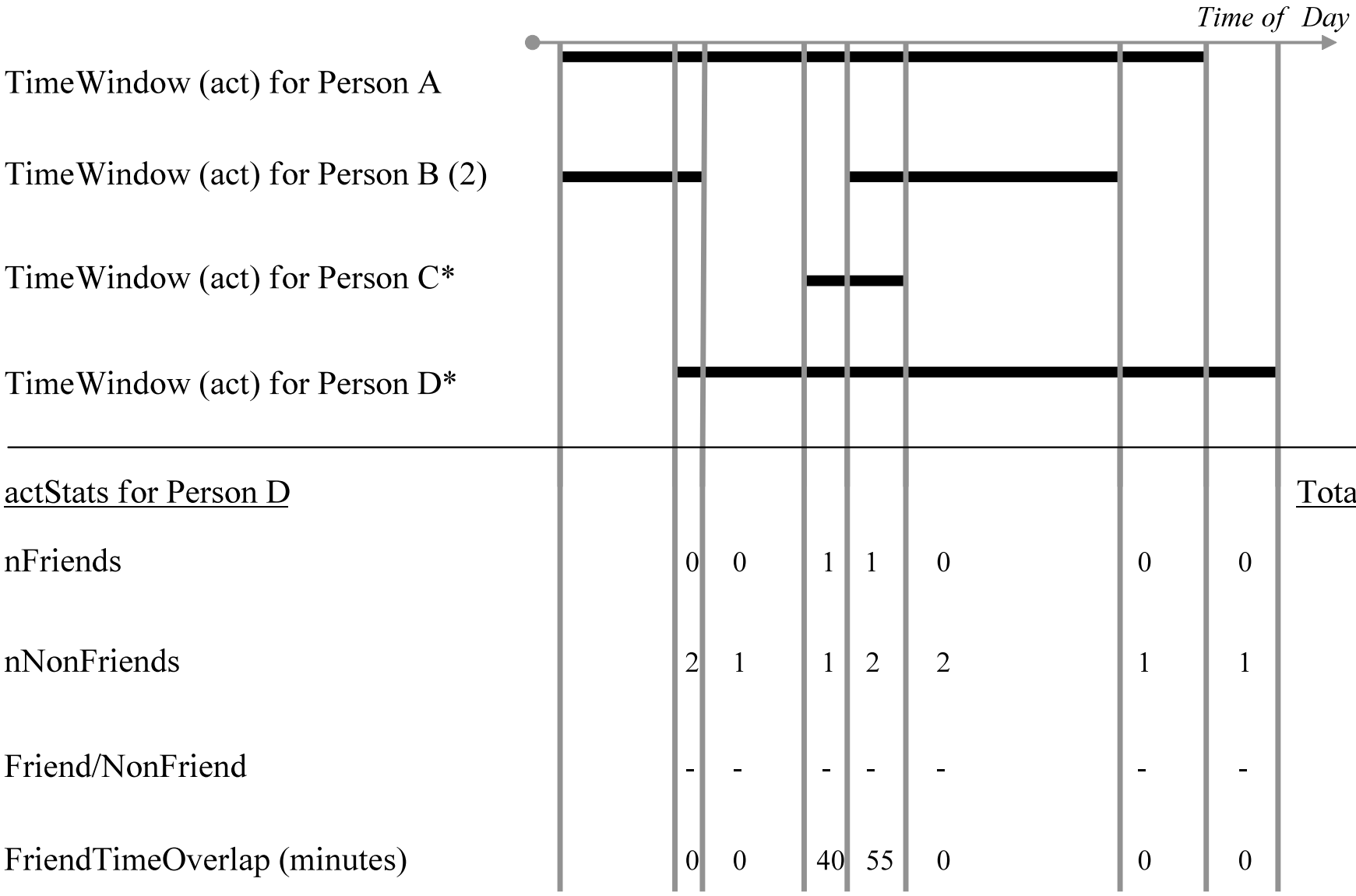

interact:
spatially
non-spatially
cleanup:
forget knowledge
dissolve social ties
get score for plan:
travel
activity
socializing
update event-dependent objects:
time windows (for interactions)
activity statistics (for scoring)
initial social network
register scoring function
initial knowledge
assignment microsim
scoring
analyses
initial activity travel plans
replanning
iteration start/end
alter plans:
route
departure time
secondary location
forget worst plan (evolution)
clear event-dependent objects:
time window map
activity statistics

Information Sharing

- Facilities (update Knowledge)
- Alters: introducing friends (EgoNet)
- Desired activity times and durations
- Other activity participants

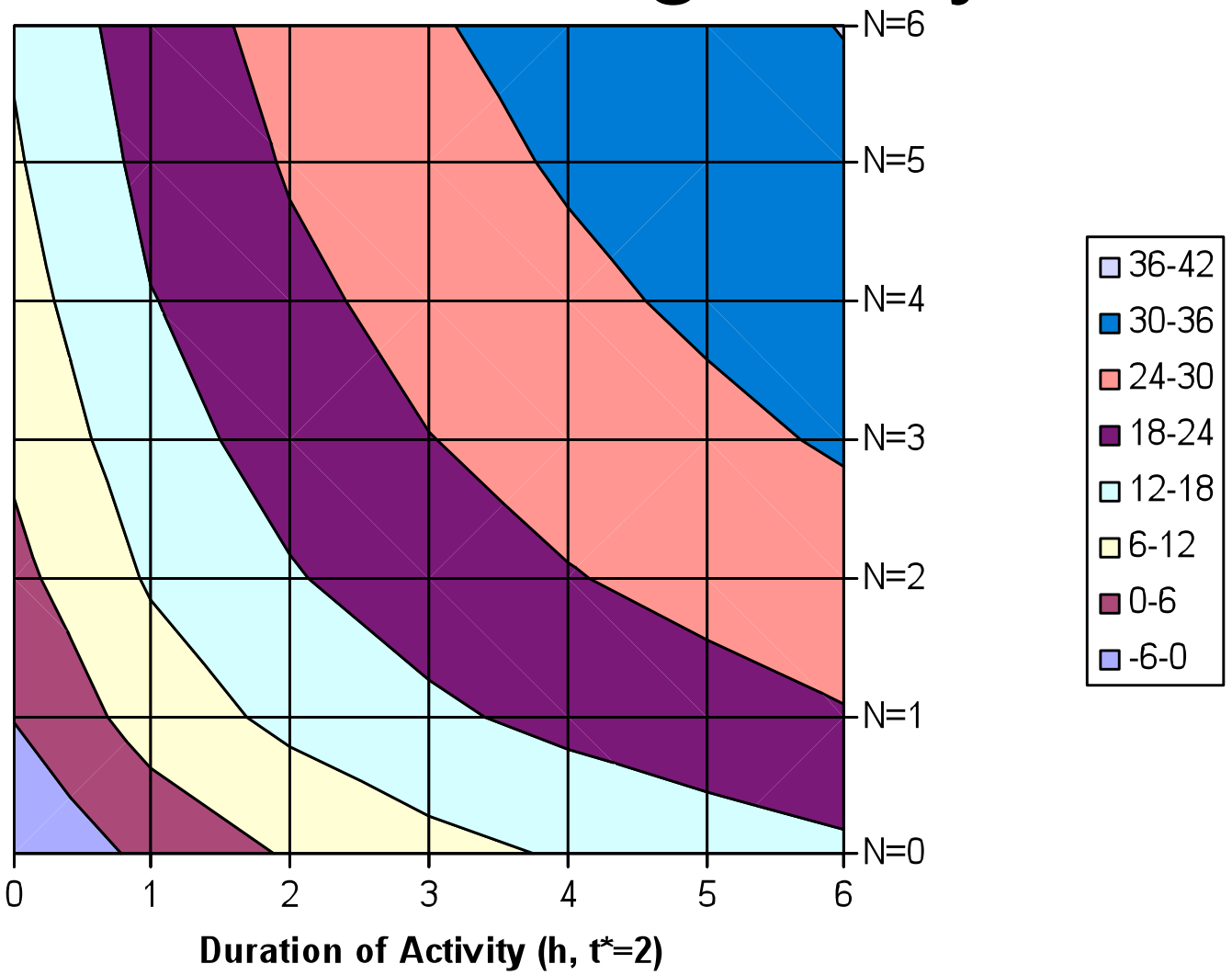
Scoring

- Individual scoring of joint plan
  - U = Ustd + β1soc*(FriendFoeRatio * δleisure)
  - U = Ustd + β2soc*(Nfriends * δleisure)
  - U = Ustd + β3soc*(*ln*(Nfriends+1) * δleisure)
  - U = Ustd + β4soc*(l*n*(TotalTimeWithFriends(hr)) * δleisure)



actStats for Person D								Total for act
nFriends	0	0	1	1	0	0	0	2
nNonFriends	2	1	1	2	2	1	1	2
Friend/NonFriend	-	-	-	-	-	-	-	0.67
FriendTimeOverlap (minutes)	0	0	40	55	0	0	0	95

Socializing Utility



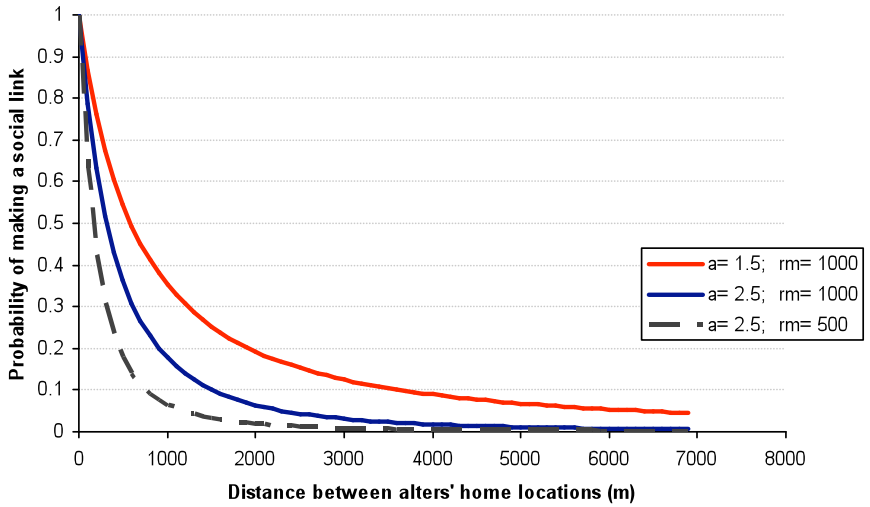
Replanning

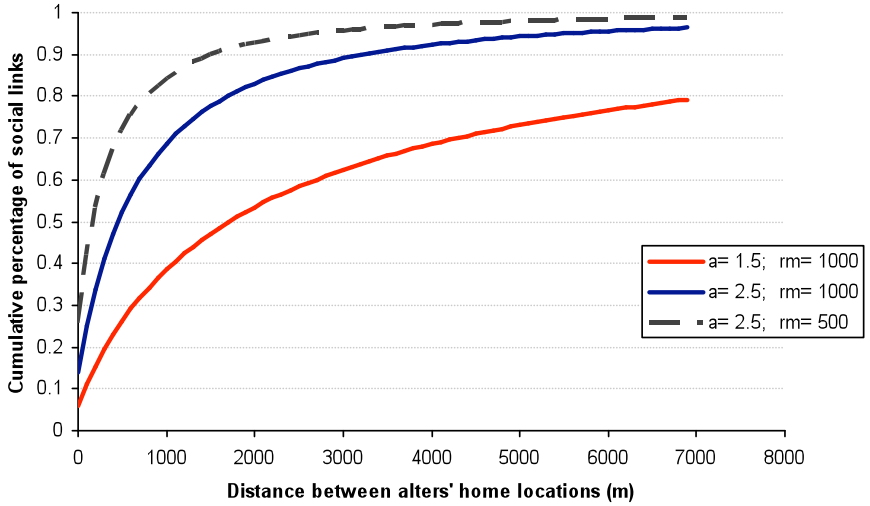
- Facility replanning
  - New facility from Knowledge
    - (Reroute, clear score)
- Joint time replanning
  - Activity start time set to group average
    - (Reschedule other acts, reroute, clear score)

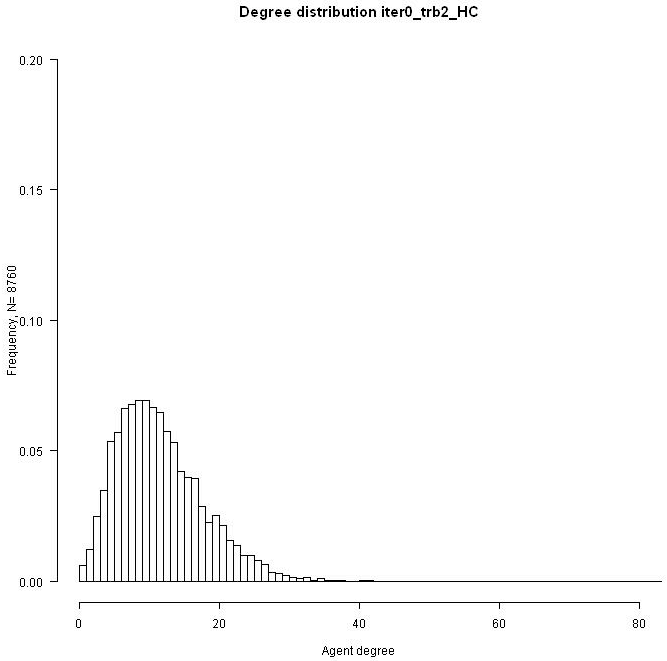
```
<module name="strategy">
<!-- Add Modules here, Probability values get all summed up and normalized to 1.0 -->
		<param name="Module_1" value="SelectExpBeta" />
		<param name="ModuleProbability_1" value="0.7" />
		<param name="Module_2" value="ReRoute" />
		<param name="ModuleProbability_2" value="0.1" />
		<param name="Module_3" value="TimeAllocationMutator7200_ReRouteLandmarks" />
		<param name="ModuleProbability_3" value="0.1" />
		<param name="Module_4" value="KSecLoc" />
		<param name="ModuleProbability_4" value="0.1" />
		<!-- param name="Module_4" value="FSecLoc" / -->
		<!-- param name="ModuleProbability_4" value="0.1" / -->

</module> <!-- strategy -->
```

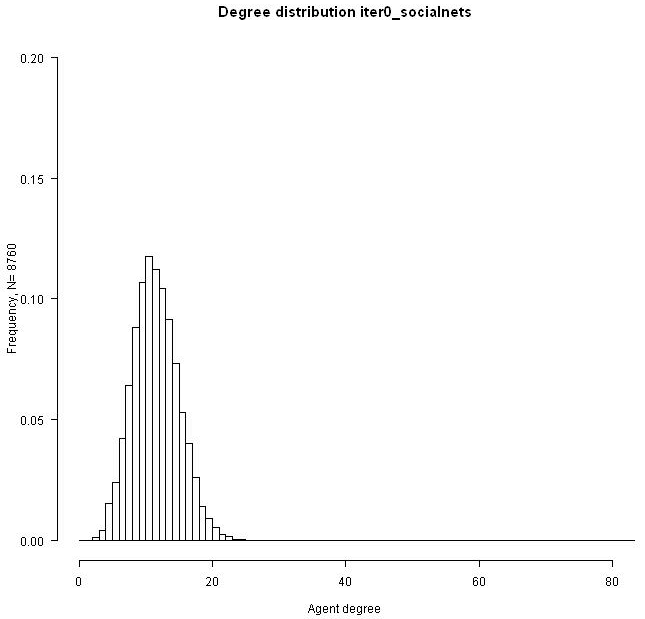
```
<module name="socialnetwork" >

        <!-- How many replanning iterations occur before letting agents socially interact again -->
        <param name="replanning_interval" value="1" />
        <param name="reporting_interval" value="50" />

        <!-- OUTPUT DIRECTORY -->
        <param name="outputDirSocialNets" value="&OUTPUTBASE;/socialnets" />

        <!-- INITIALIZE THE SOCIAL TIES -->
        <param name="socnetalgorithm" value="euclidrandom" />
        <param name="kbar" value="12" />
        <param name="edge_type" value="UNDIRECTED" />
        <param name="inputSocNetDir" value="&INPUTBASE;/socialnets" />
        <param name="inputIter" value = "0" />
        <param name="euclid_alpha" value ="1.5" />
        <param name="euclid_rmin" value ="1000" />
```

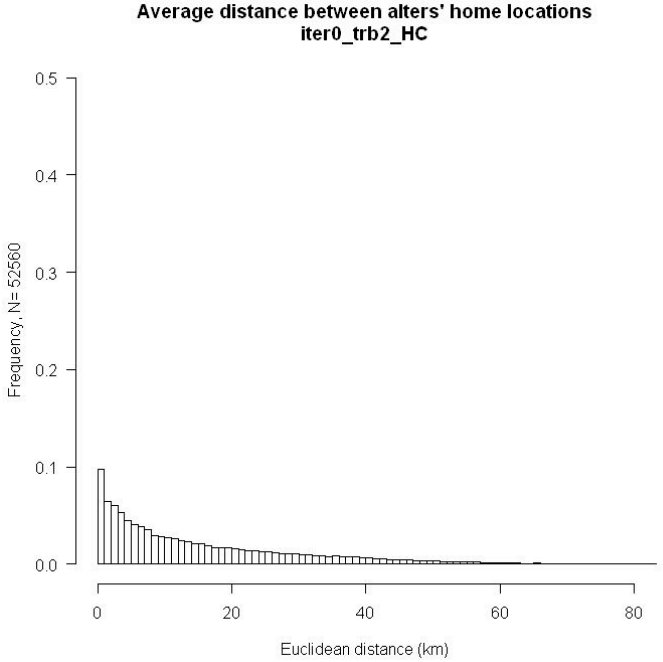
Figure 1 The distance-dependent probability of a social tie being constructed, given agent *i* and *j*, α=1.5, r_{min}=1000m.

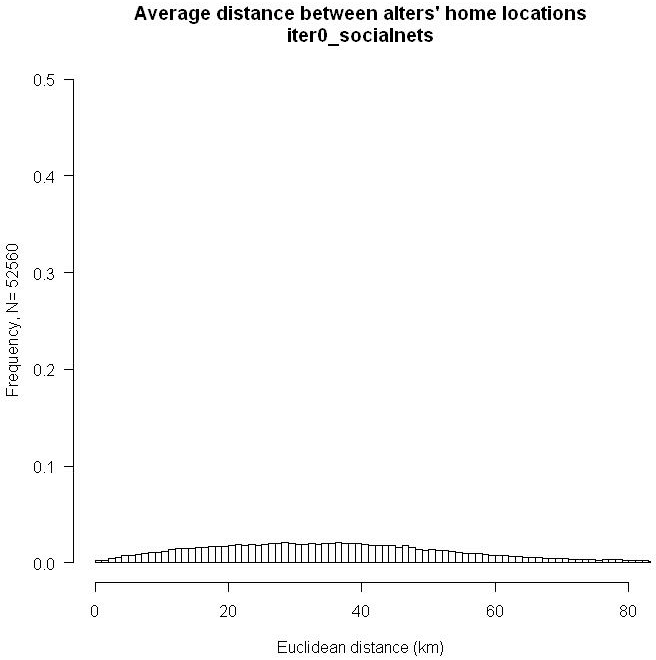




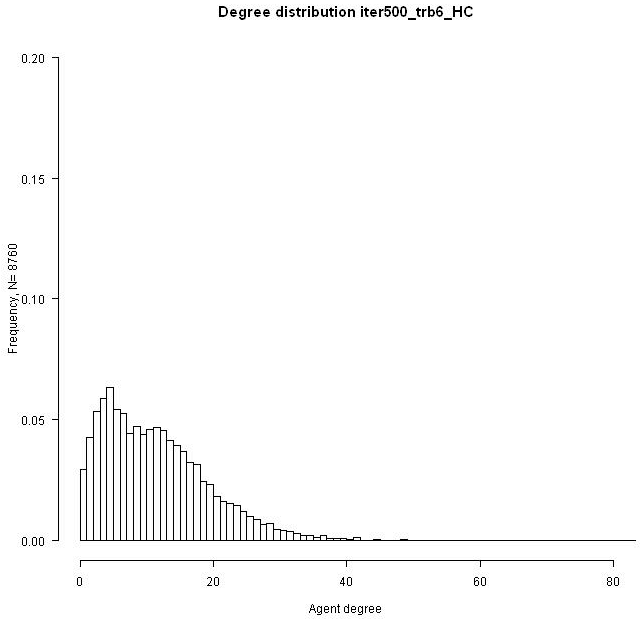

Degree distribution iter0_trb2_HC
Frequency, N= 8760
0.00
0.05
0.10
0.15
0.20
0
20
40
60
80
Agent degree

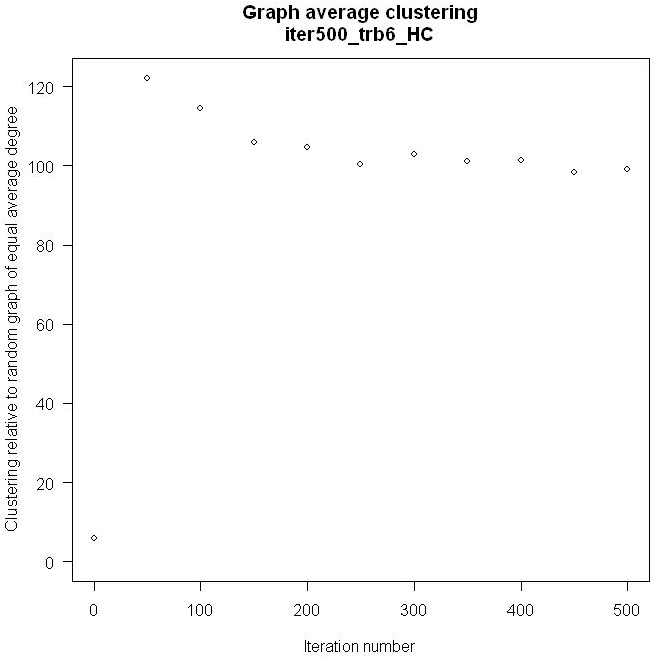

Degree distribution iter0_socialnets
Frequency, N= 8760
0.00
0.05
0.10
0.15
0.20
0
20
40
60
80
Agent degree

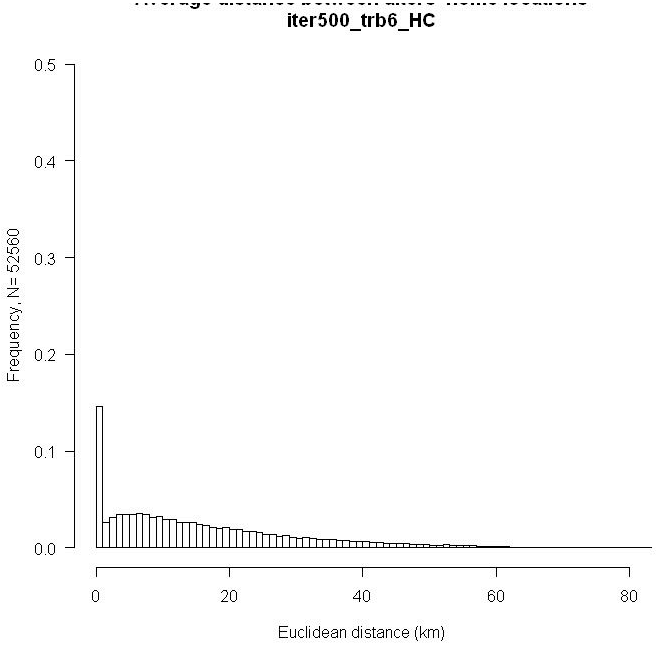

Average distance between alters' home locations
iter0_trb2_HC
Frequency, N= 52560
0.0
0.1
0.2
0.3
0.4
0.5
0
20
40
60
80
Euclidean distance (km)


Average distance between alters' home locations
iter0_socialnets
Frequency, N= 52560
0.0
0.1
0.2
0.3
0.4
0.5
0
20
40
60
80
Euclidean distance (km)

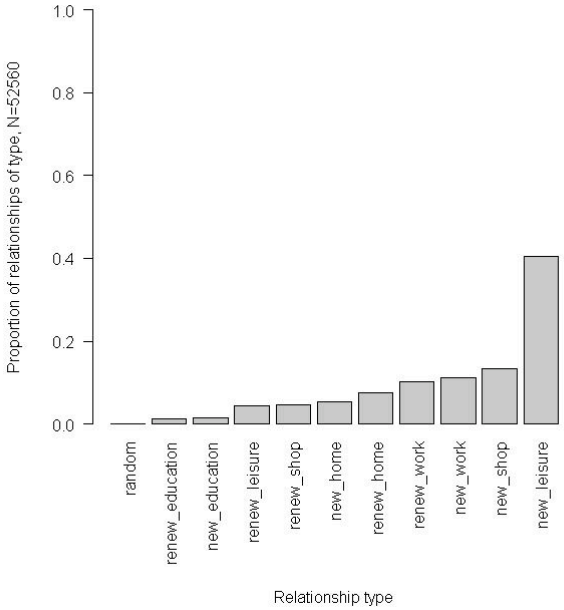
```
<!-- SOCIAL NETWORK DYNAMICS -->
                        <param name="socnetlinkremovalage" value="0" />
                        <param name="socnetlinkremovalalgorithm" value="constantkbar" />
                        <param name="socnetlinkremovalp" value="0.05" />
                        <param name="socnetlinkstrengthalgorithm" value="constant" />

                        <param name="prob_befriend" value="1." />
                        <param name="degree_saturation_rate" value="0" />
```

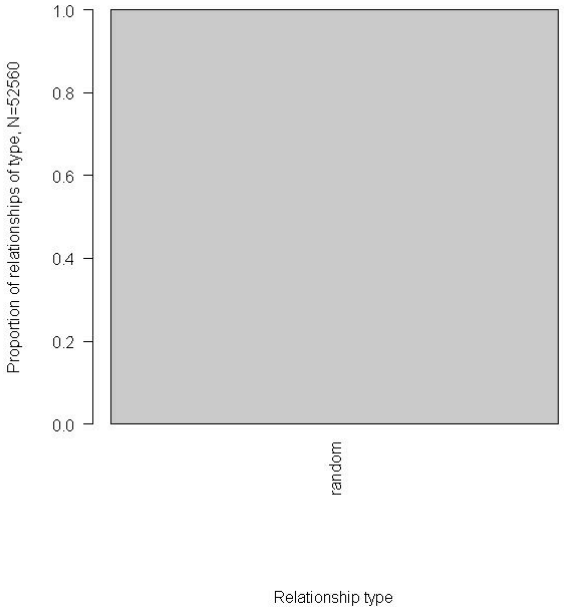

Degree distribution iter500_trb6_HC
Frequency, N= 8760
0.20
0.15
0.10
0.05
0.00
0
20
40
60
80
Agent degree

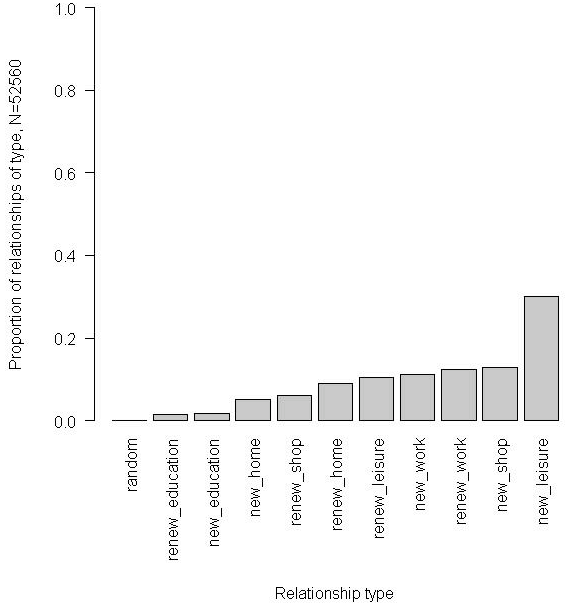

Graph average clustering
iter500_trb6_HC
Clustering relative to random graph of equal average degree
120
100
80
60
40
20
0
0
100
200
300
400
500
Iteration number


iter500_trb6_HC
Frequency, N= 52560
0.5
0.4
0.3
0.2
0.1
0.0
0
20
40
60
80
Euclidean distance (km)

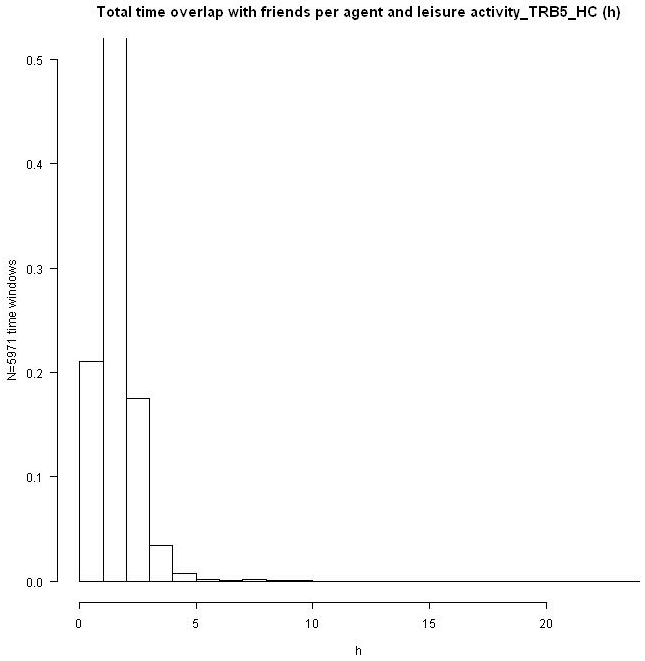
```
        <!-- INFORMATION EXCHANGES IN SOCIAL SPACE -->
                        <param name="factype_ns" value="home,leisure,shop,education,work" />
                        <param name="fract_introduce_friends" value="0.0" />
                        <param name="fract_ns_interact" value="1.0" />
                        <param name="num_ns_interactions" value="1" />
                        <param name="memSize" value="0" />


<!-- INFORMATION EXCHANGES IN GEOGRAPHICAL SPACE-->
                                <param name="spatial_interactor_type" value="timewindowrandom" />
                                <param name="s_weights" value="0.,0.,0.,0.," />
                                <param name="act_types" value="home,work,shop,education,leisure" />
```

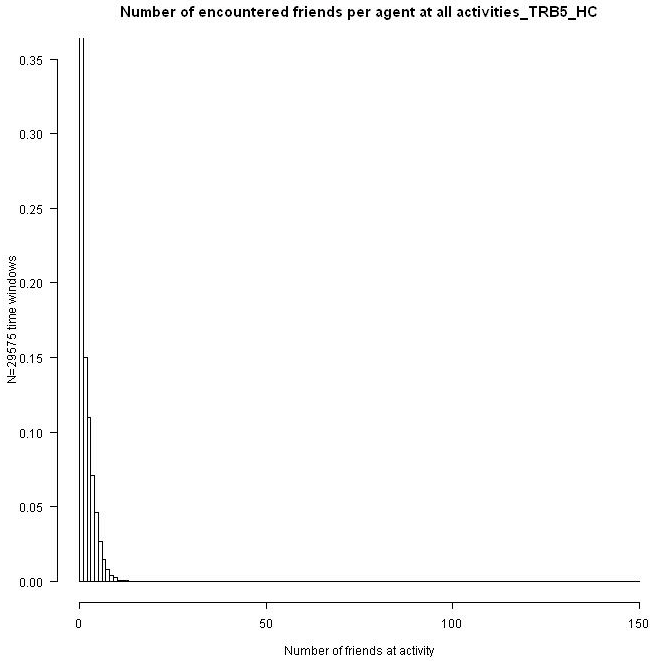

Proportion of relationships of type, N=52560
1.0
0.8
0.6
0.4
0.2
0.0
random
renew_education
new_education
renew_leisure
renew_shop
new_home
renew_home
renew_work
new_work
new_shop
new_leisure
Relationship type

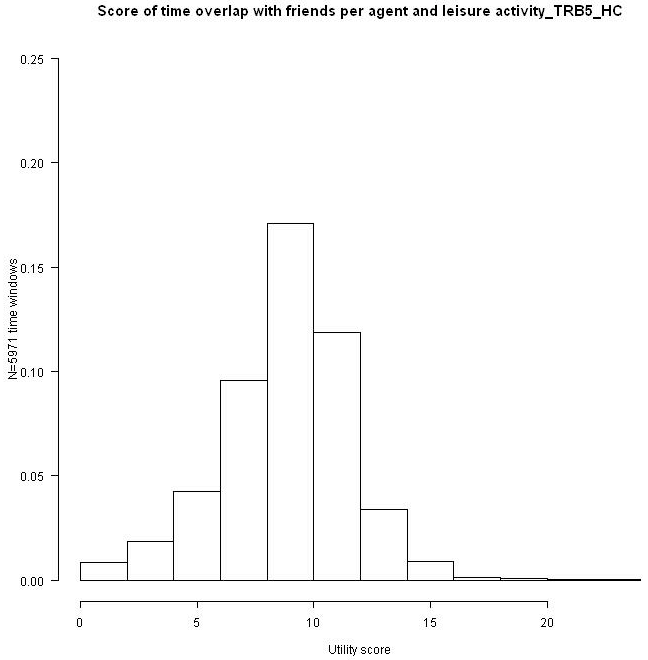

Proportion of relationships of type, N=52560
1.0
0.8
0.6
0.4
0.2
0.0
random
Relationship type

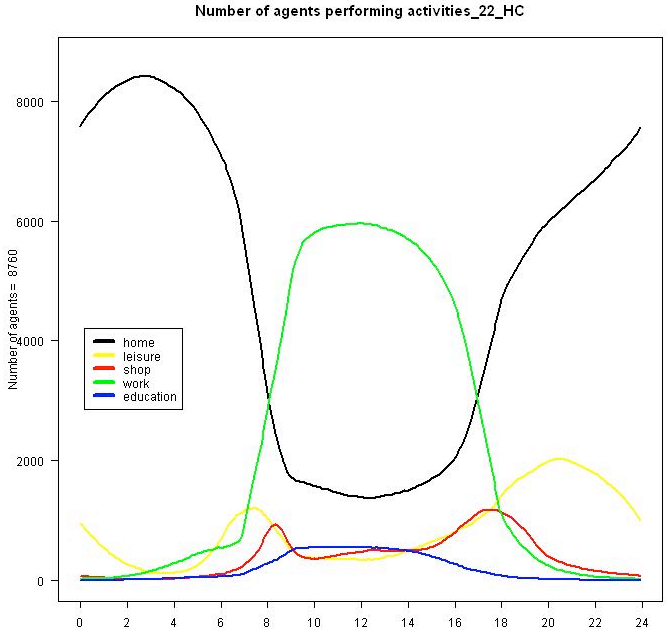

Proportion of relationships of type, N=52560
1.0
0.8
0.6
0.4
0.2
0.0
random
renew_education
new_education
new_home
renew_shop
renew_home
renew_leisure
new_work
renew_work
new_shop
new_leisure
Relationship type

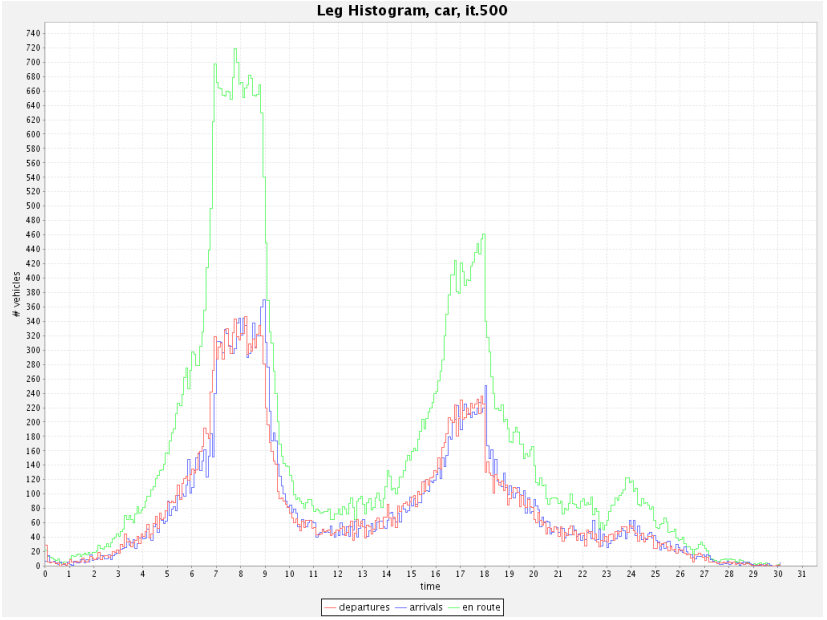
```
<!-- SCORING -->
                        <param name="betafriendfoe" value = "0" />
                        <param name="betanfriends" value = "0" />
                        <param name="betalognfriends" value = "0" />
                        <param name="betatimewithfriends" value = "24" />

<!-- REPLANNING -->
<!-- switch_weights: Replanning parameter. Probability of switching out these
                        facility types:
                        "home","work","shop","education","leisure".
                        Syntax is "0.,1.0,0.5,0.,0.5" with commas and no spaces.
                        Automatically normalized to 1.0.
                        Only one facility is switched out per plan per replanning iteration -->

                        <param name="switch_weights" value=".0,.0,1.0,.0,1.0" />
```

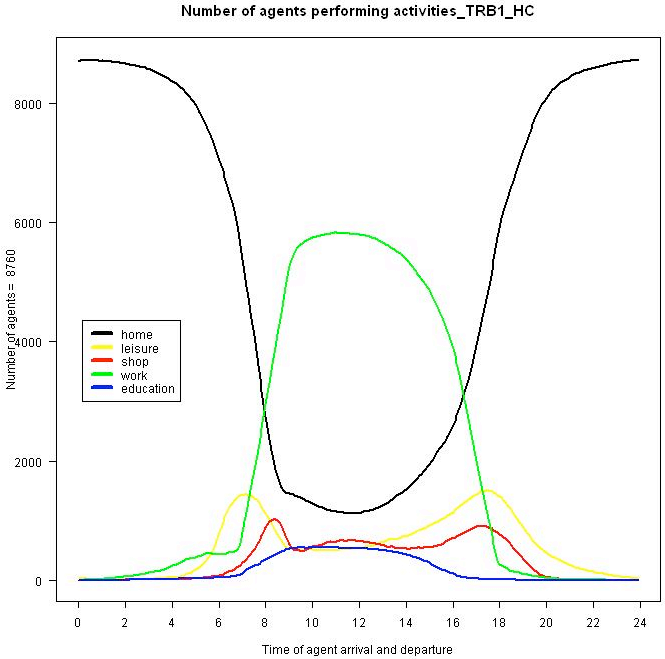

Total time overlap with friends per agent and leisure activity_TRB5_HC (h)
N=5971 time windows
0.0
0.1
0.2
0.3
0.4
0.5
0
5
10
15
20
h

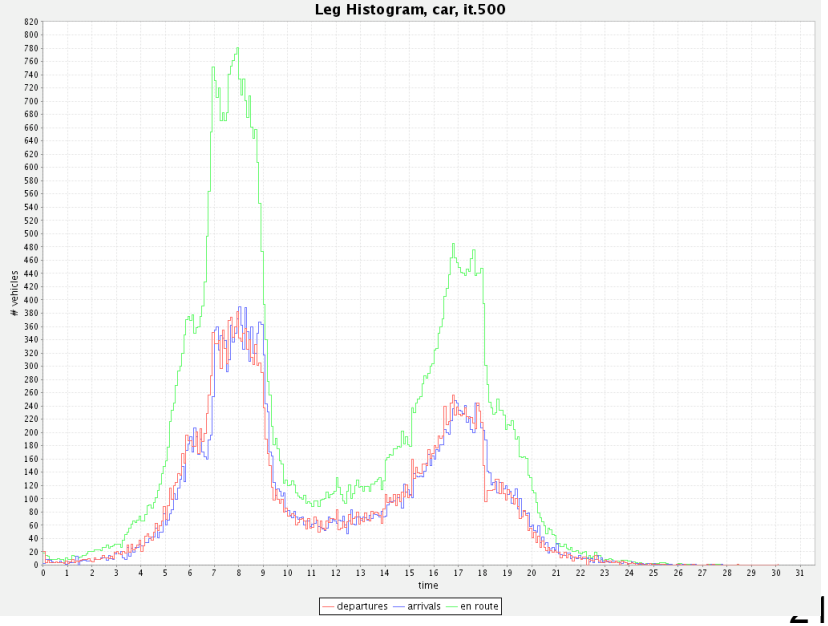

Number of encountered friends per agent at all activities_TRB5_HC
N=29575 time windows
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0
50
100
150
Number of friends at activity


Score of time overlap with friends per agent and leisure activity_TRB5_HC
N=5971 time windows
0.00
0.05
0.10
0.15
0.20
0.25
0
5
10
15
20
Utility score


Number of agents performing activities_22_HC
Number of agents = 8760
home
leisure
shop
work
education


Leg Histogram, car, it.500
vehicles
time
departures
arrivals
en route

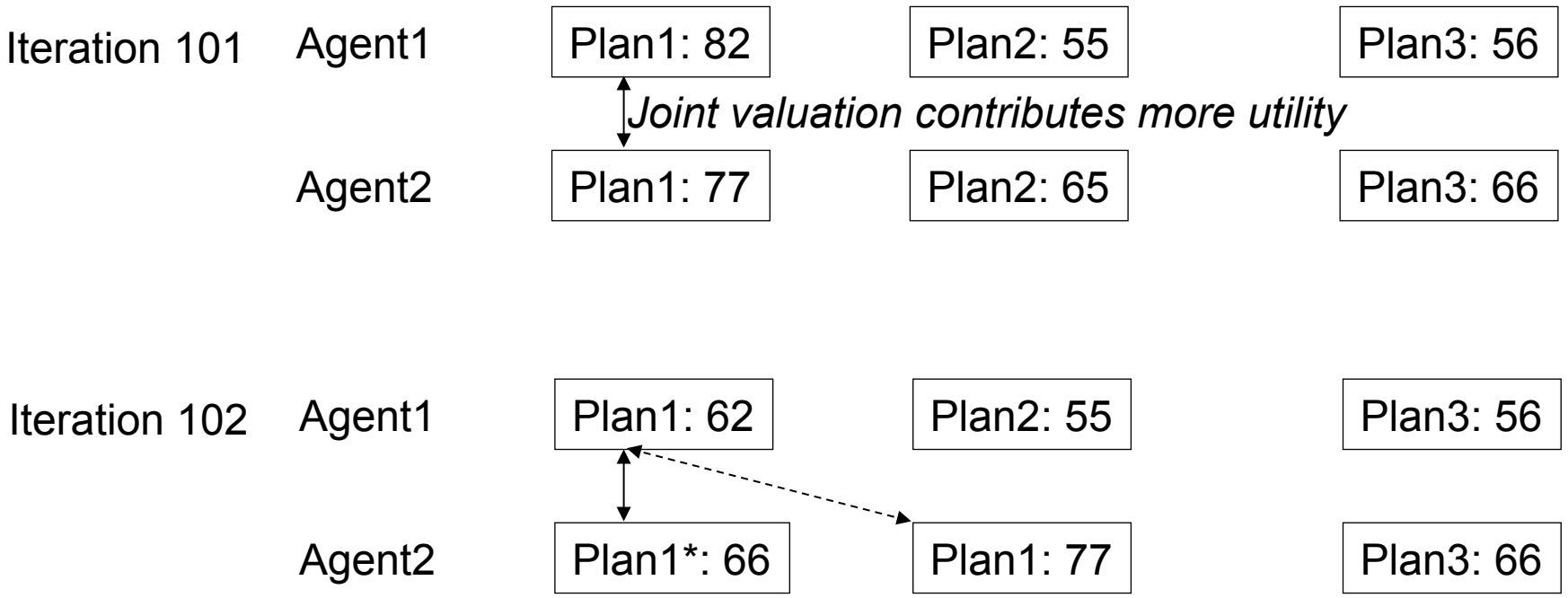

Number of agents performing activities_TRB1_HC
Number of agents = 8760
home
leisure
shop
work
education
Time of agent arrival and departure


Leg Histogram, car, it.500
vehicles
time
departures
arrivals
en route

Truly Joint Plans

- Joint intent (plans)
  - Activity start time
  - Activity duration
  - Others there
  - Cost sharing
- Joint evolution?
  - Joint plan collapses if culling individual plans
  - Replaced with another joint plan
  - Intent cannot be maintained –> "working point"

Truly Joint Plans



Plan1 of Agent2 is replanned and Plan1 of Agent1 is not.
The joint utility valuation is lower for both.

Existing Social Networks Objects

- Social network
- Ego net
- Mental Map
- Spatial Interactor
- Non-spatial Interactor
- Replanning
- Scoring
- Statistics

Desirable Social Network Objects

- Joint plans
  1. "AppointmentBook"
  2. "SocialAct"
     - Desired activity start time
     - Desired activity duration
  3. Joint plan evolution
     - Maintain agents in lock-step