AGENT-BASED MODELING, SIMULATION, AND CONTROL—SOME APPLICATIONS IN TRANSPORTATION

Montasir Abbas, Virginia Tech
(with contributions from past and present VT-SCORES students, including:
Zain Adam, Sahar Ghanipoor-Machiani, Linsen Chong, and Milos Mladenovic)

Workshop III: Traffic Control

New Directions in Mathematical Approaches for Traffic Flow Management

IPAM

October 27, 2015

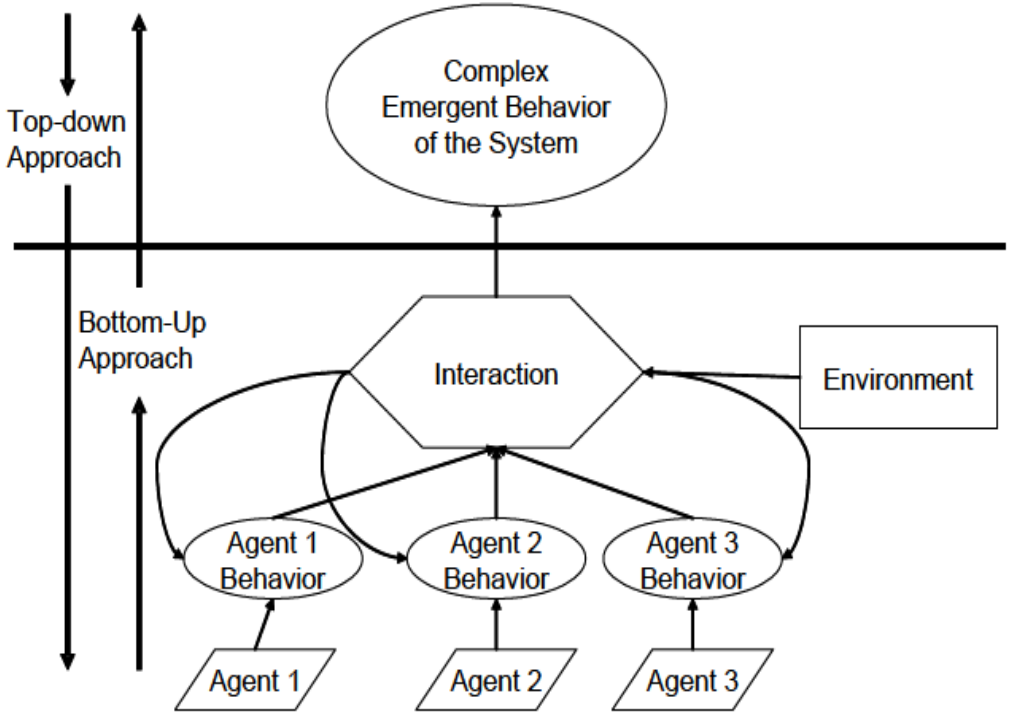
VT-SCORES Lab

Virginia Tech Signal Control & Operation Research and Education Systems Lab

Presentation Outline

- Agent based modeling… what? why? And how?
- What is the learning framework? What are the techniques?
- Examples of learning:
  - Controller agents
  - Driver behavior agents
  - Vehicle agents
- What if we don't incorporate learning?
- Conclusions

Background


Top-down
Approach
Complex
Emergent Behavior
of the System
Bottom-Up
Approach
Interaction
Environment
Agent 1
Behavior
Agent 2
Behavior
Agent 3
Behavior
Agent 1
Agent 2
Agent 3

Learning

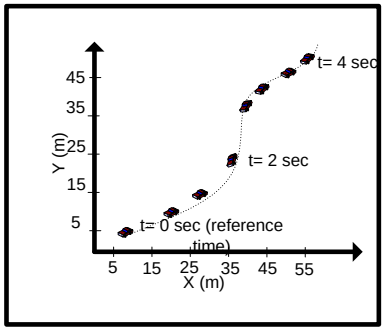
- Can we predict a condition or a behavior/response from a wealth of data?
- Can we model and interpret a phenomenon in a state-action framework?
- The same input data can lead to different performance measures, and we are the reason!

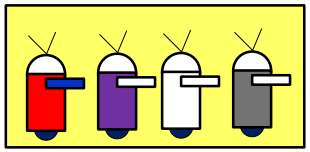
Motivation

Varying Traffic Behavior Maneuvers

Naturalistic Data

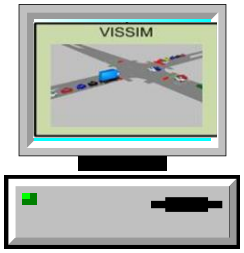
Detailed Behavioral Data Trajectories





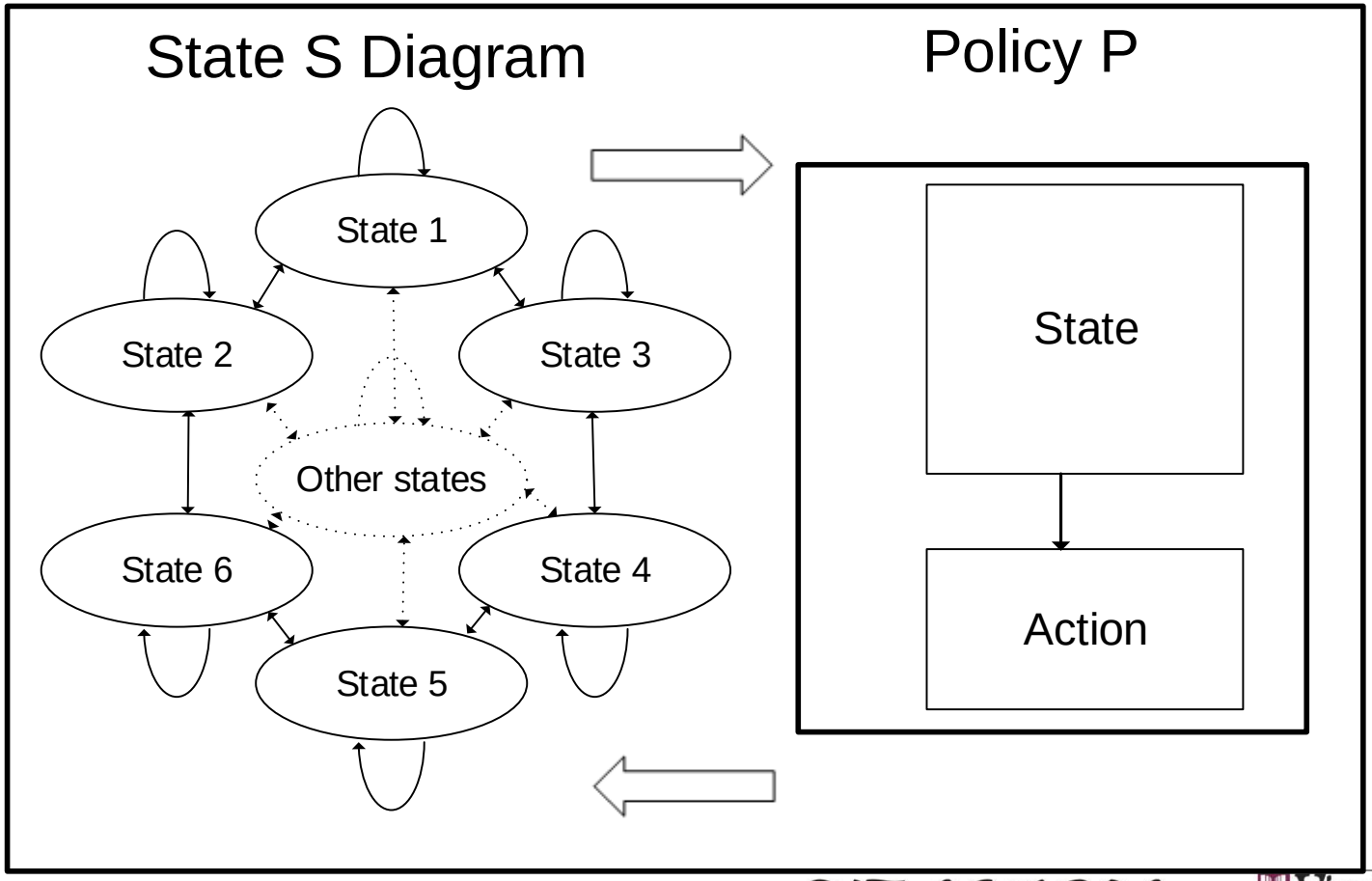
Trained Agents

Advanced VISSIM API-Agent Interface



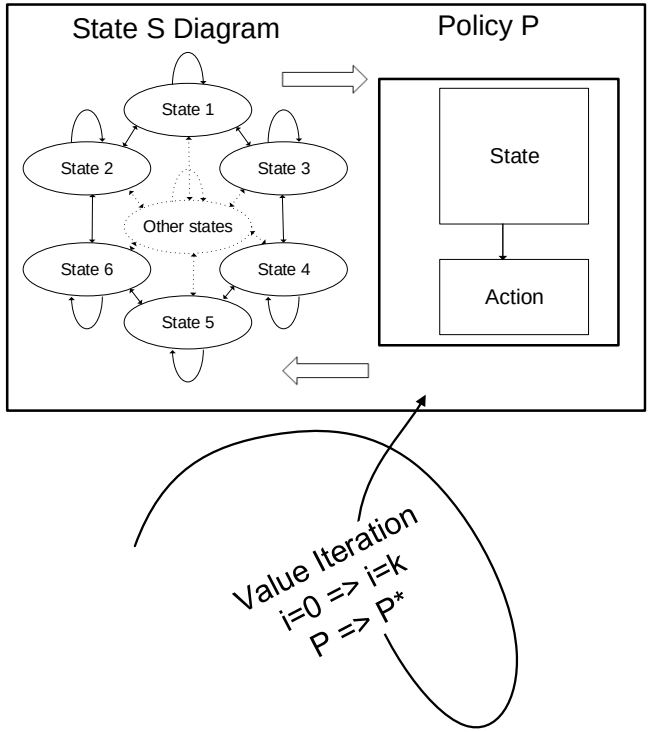
VISSIM Simulation

A Learning Framework



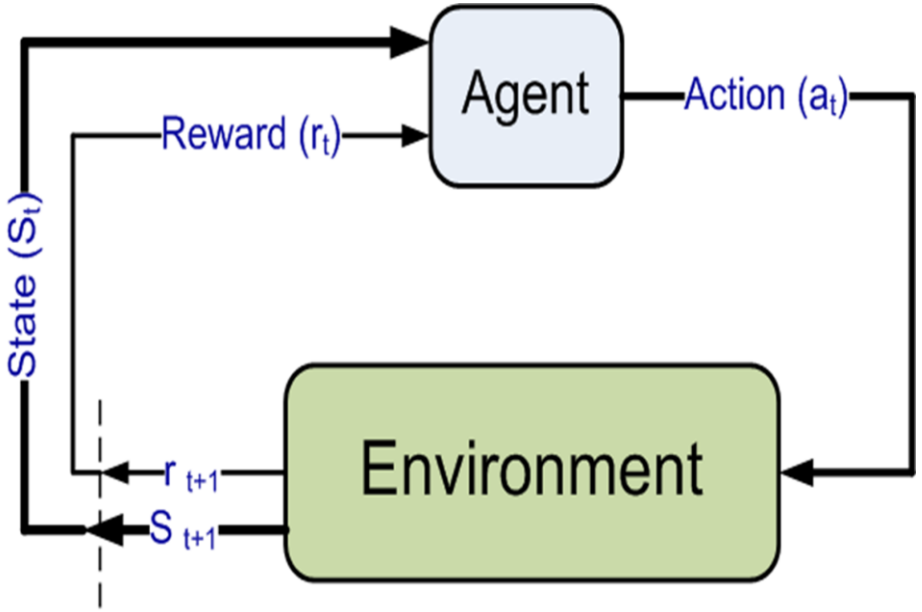
Learning Techniques

- Machine Learning
- Q-Learning
- Reinforcement Learning
- Etc.



Q-Learning

Acting on environment, receiving rewards, selecting actions to reach a goal



$$Q_{t+1}(s_t, a_t) = \underbrace{Q_t(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha_t(s_t, a_t)}_{\text{learning rate}} \cdot \left(\overbrace{\underbrace{R_{t+1}}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \underbrace{\max_a Q_t(s_{t+1}, a)}_{\text{estimate of optimal future value}}}^{\text{learned value}} - \underbrace{Q_t(s_t, a_t)}_{\text{old value}} \right)$$

Application: Dilemma Zone Problem

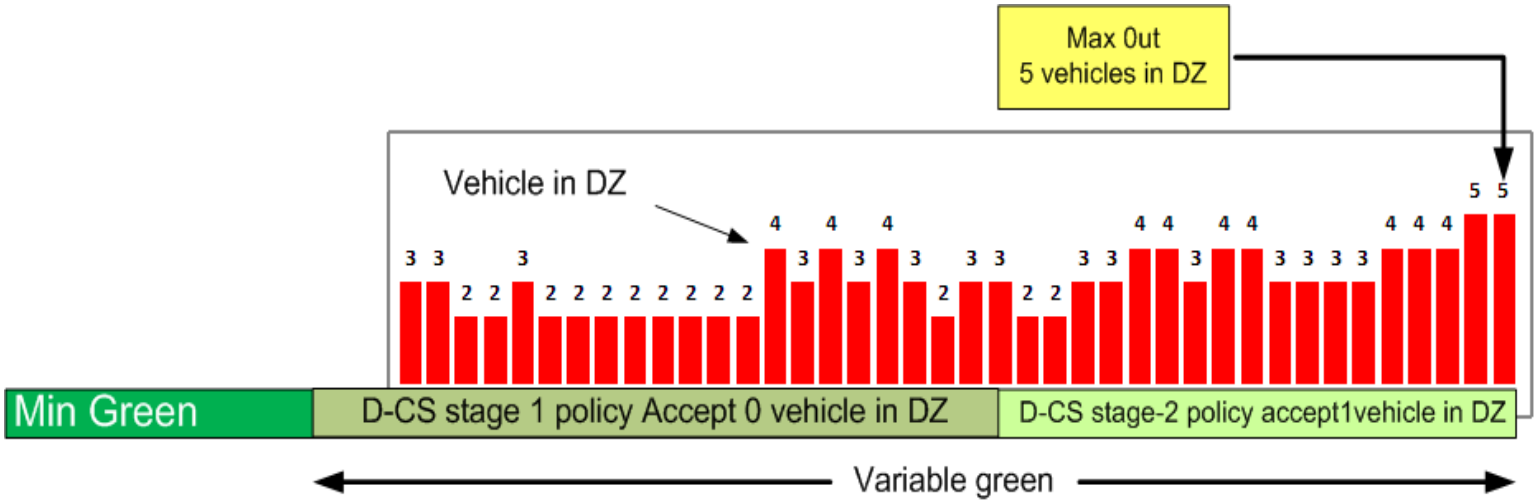
- Application of learning to controller and to humans
  - Controllers making decisions
  - Humans learning from mistakes

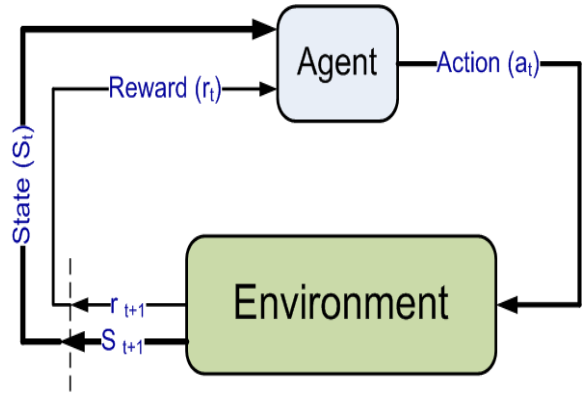
To stop or not to stop? That is the question!

To stop

To go

Controller Agent—Learning the Policy





$$Q_{(s,a)} \leftarrow Q_{(s,a)} + \alpha\left(r_{(s,a)} + \gamma.\max Q_{-1(s,a)} - Q_{(s,a)}\right)$$

Environment's State Variables:

Total number of vehicles in DZ

Agent's Actions:

- End the Green
- Extend the Green

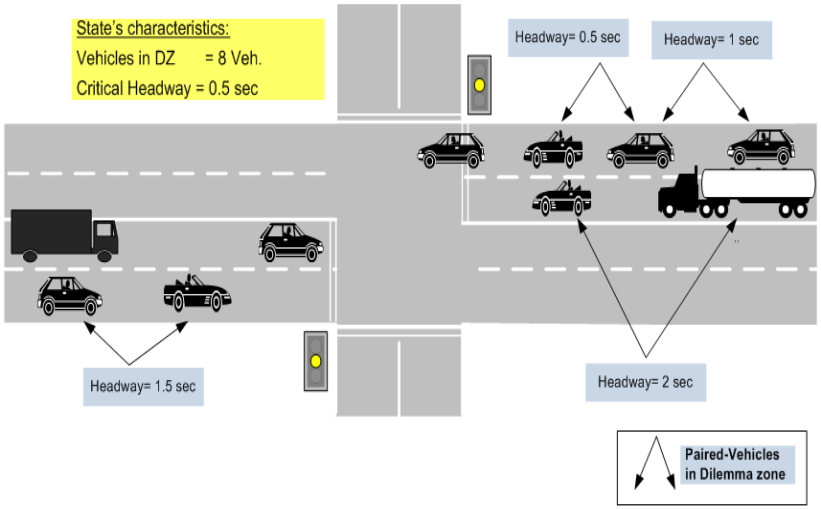
Reward:

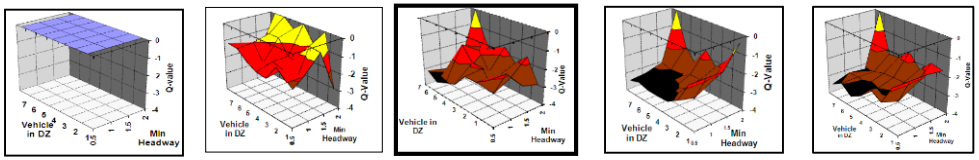
Vehicles caught in DZ

Q-learning algorithm parameters:

Learning rate: 0.01

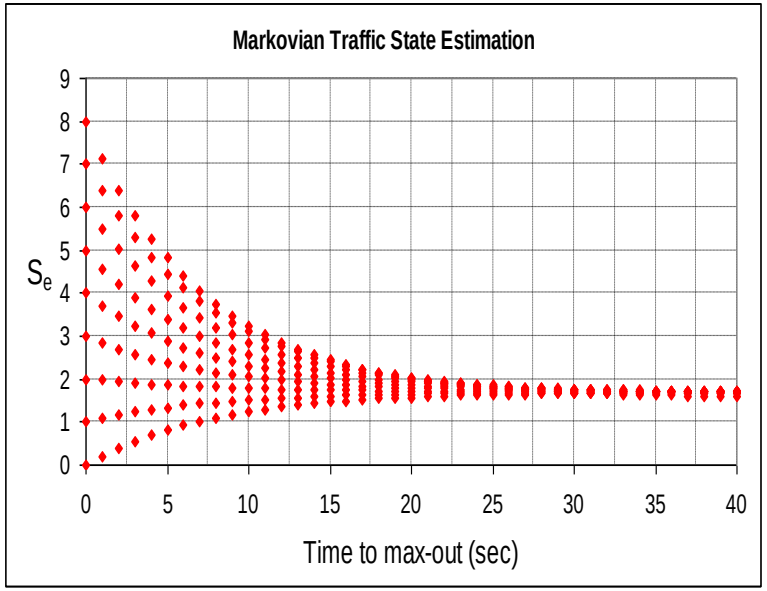
Discount rate: 0.5

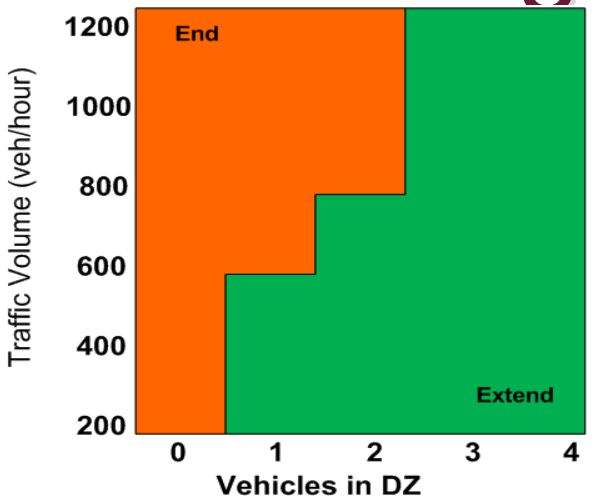


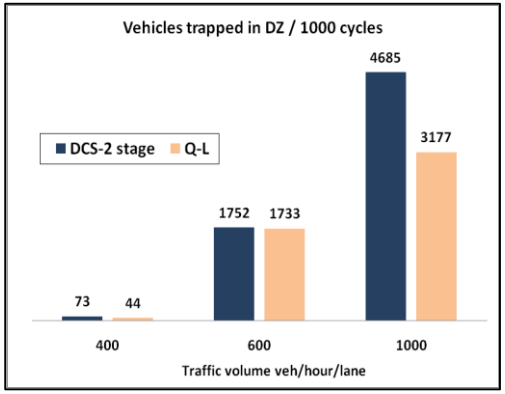


Off-line and Online Learning

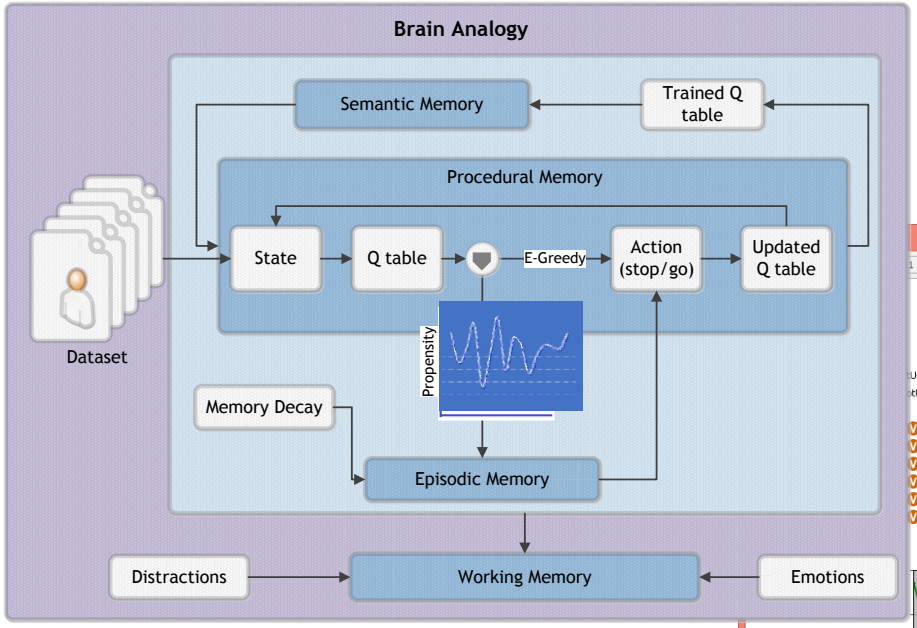
- Find P* with simulation
- Update Q-table with real data

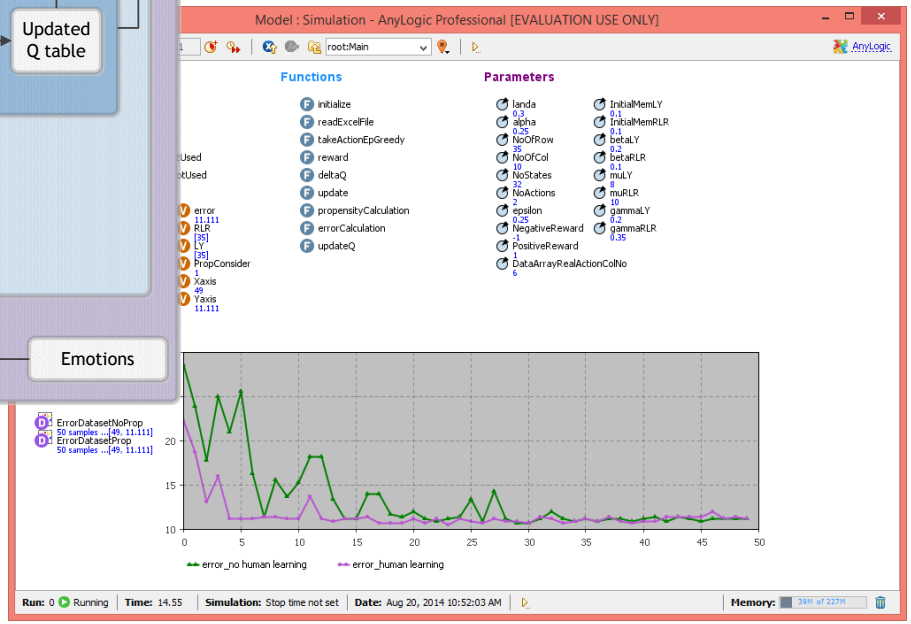






Human Learning Model



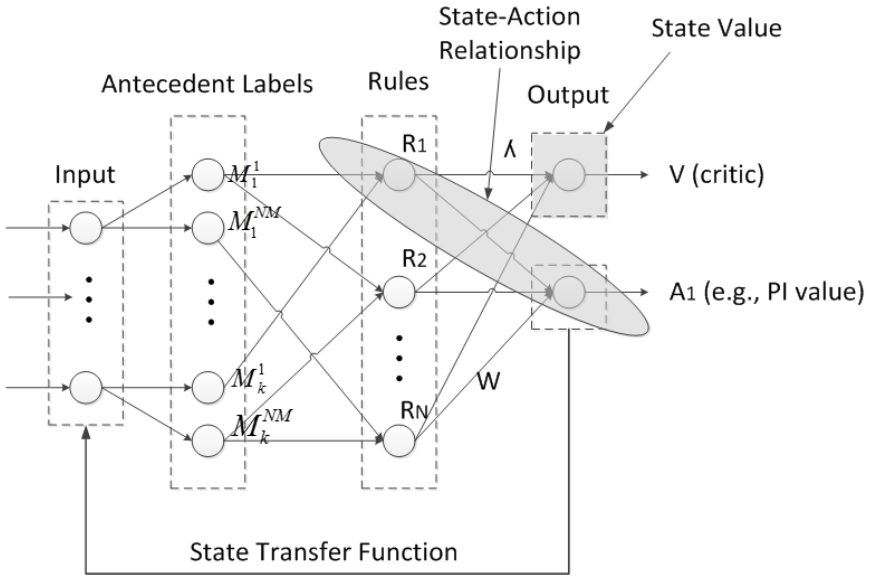


Dealing with High State Dimensionality (Naturalistic driving behavior study)*

- Training input: traffic states and actions
- Training output: acceleration and steering
- Input variables discretized using fuzzy sets
- Continuous actions are generated from discrete actions
- Uses all the safety critical events available in training

*Safety and Mobility Agent-based Reinforcement-learning Traffic Simulation Add-on Module (SMART SAM)

NFACRL Framework



S_i=the i^{th} input variable (state variable)

K =number of input variables

NM_i=number of fuzzy sets or membership functions for the S_i

$M_i^{a(i)}$=$a(i)^{th}$ fuzzy set or membership function for the i^{th} input variable

R_j=the j^{th} fuzzy rule

N=number of fuzzy rules

λ_j=weight between j^{th} fuzzy rule and critic

w_q^j=weight between j^{th} fuzzy rule and action q

V =critic value

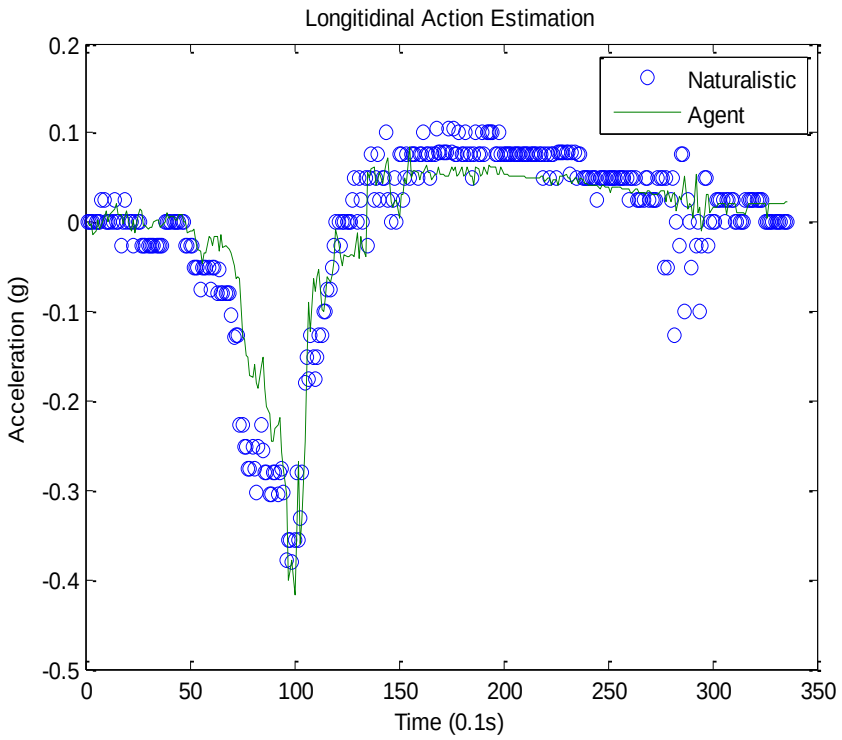
A_q =output of q^{th} action

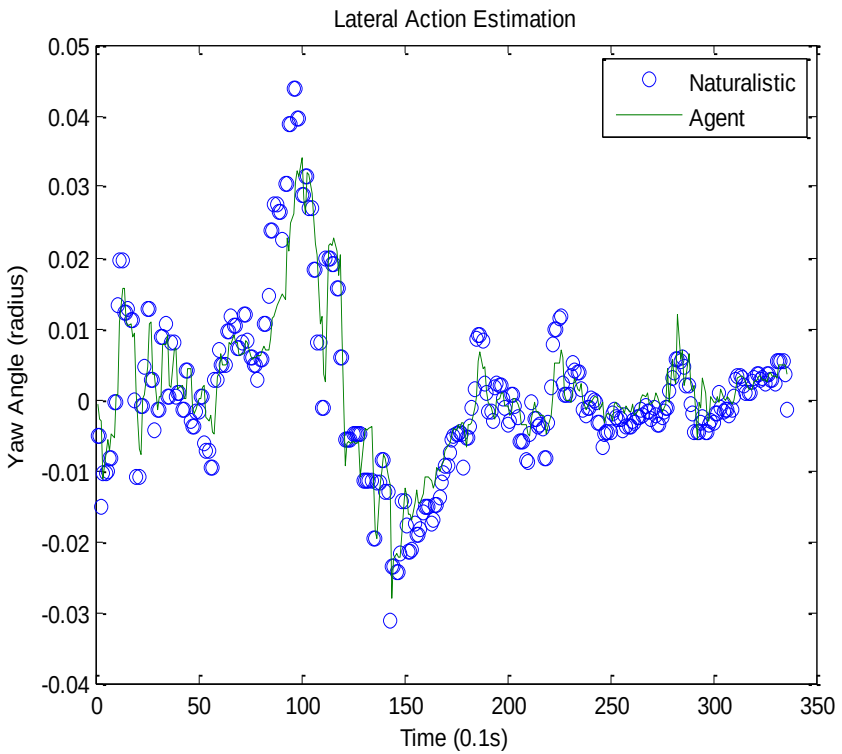
Where $i = 1, \ldots K, a(i) = 1, .. NM_i, j = 1, \ldots, N$ and $q = 1, .., P$

Applications and Cross Validation

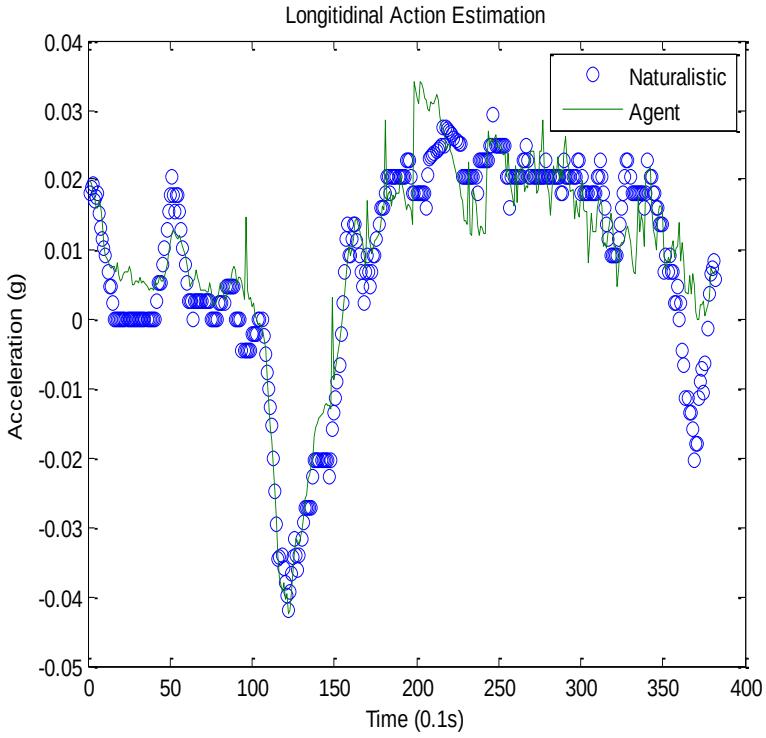
- Test the heterogeneity of the drivers
- Training: Used the data from Agent A in training with its behavioral rules as output
- Validation: Used the output rule of Agent A and applied it to driver B
- Heterogeneity of Agent A, B is represented by degree of accuracy in validation

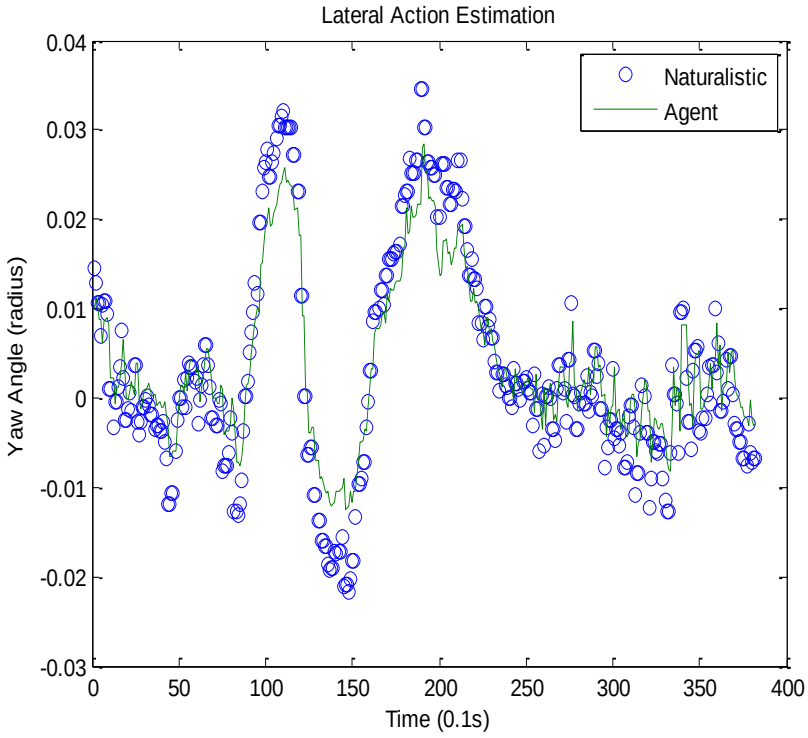
Agent A: Event 1



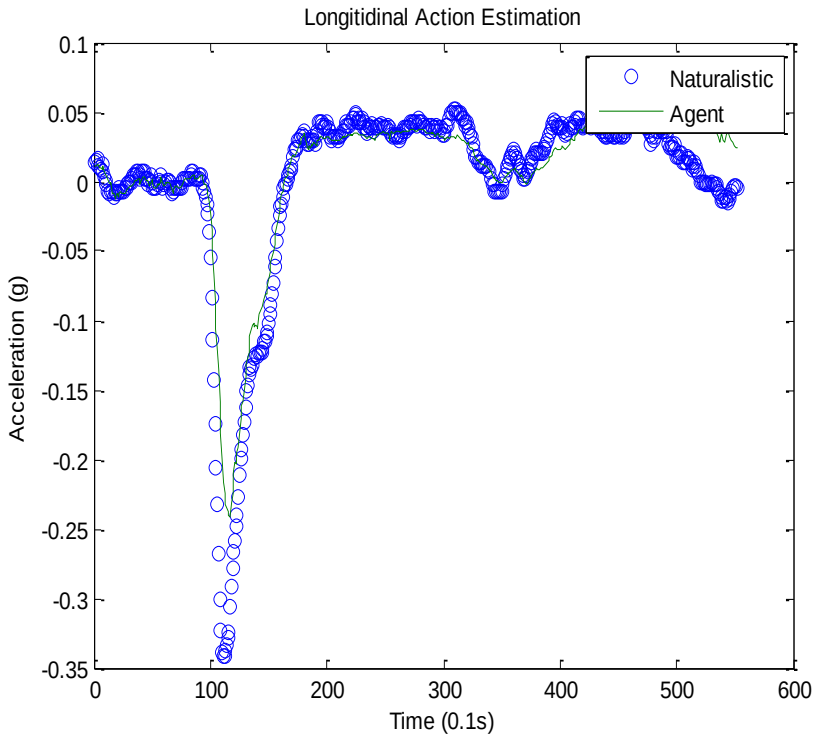


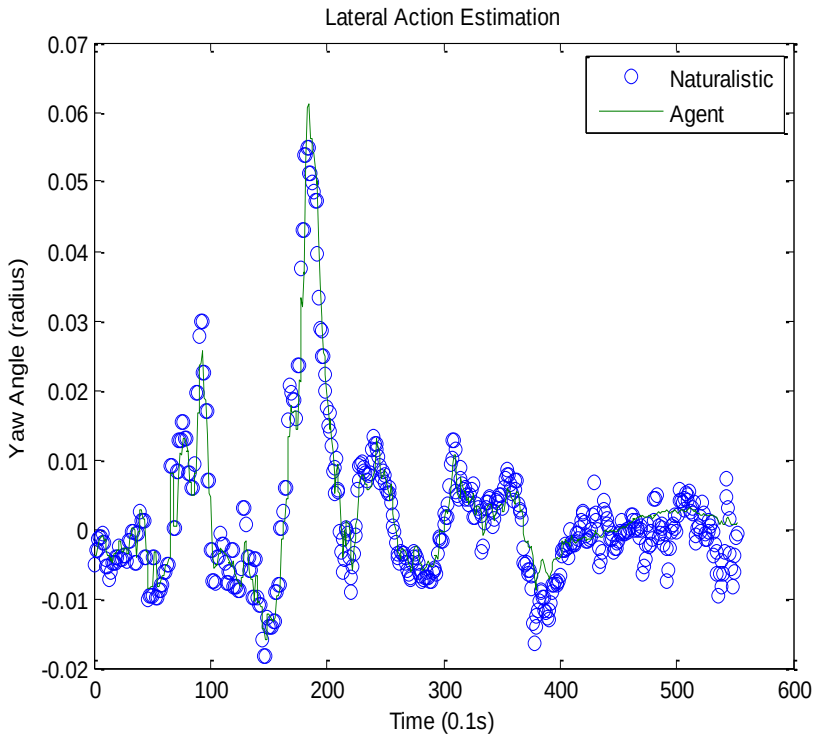
Agent A: Event 2



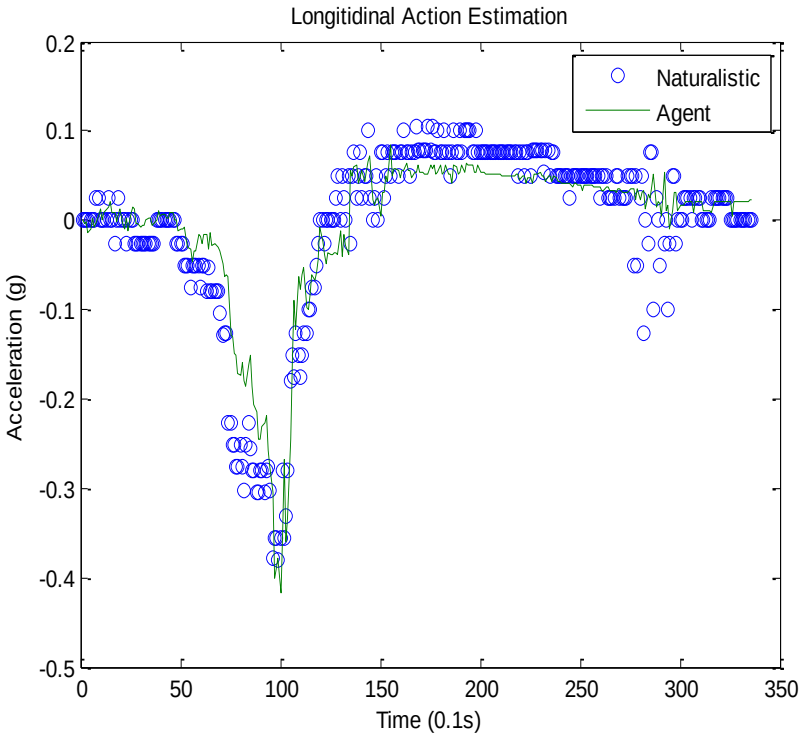


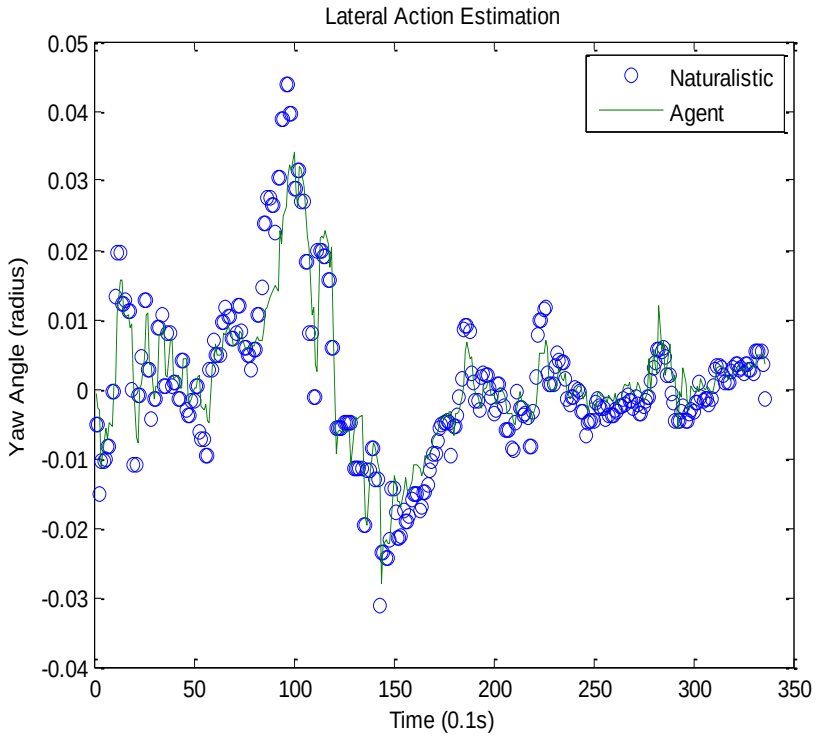
Driver Agent B



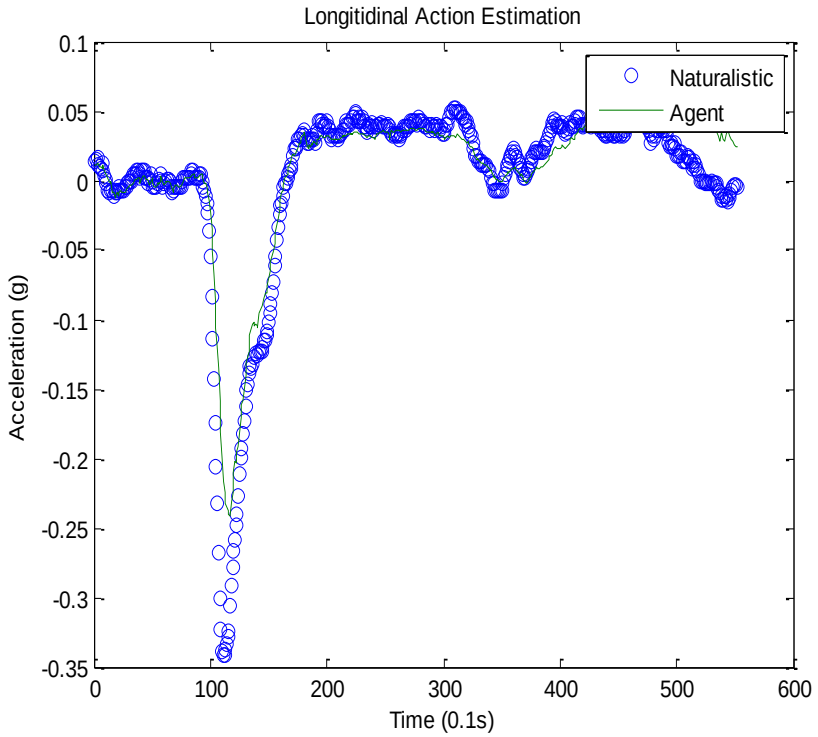


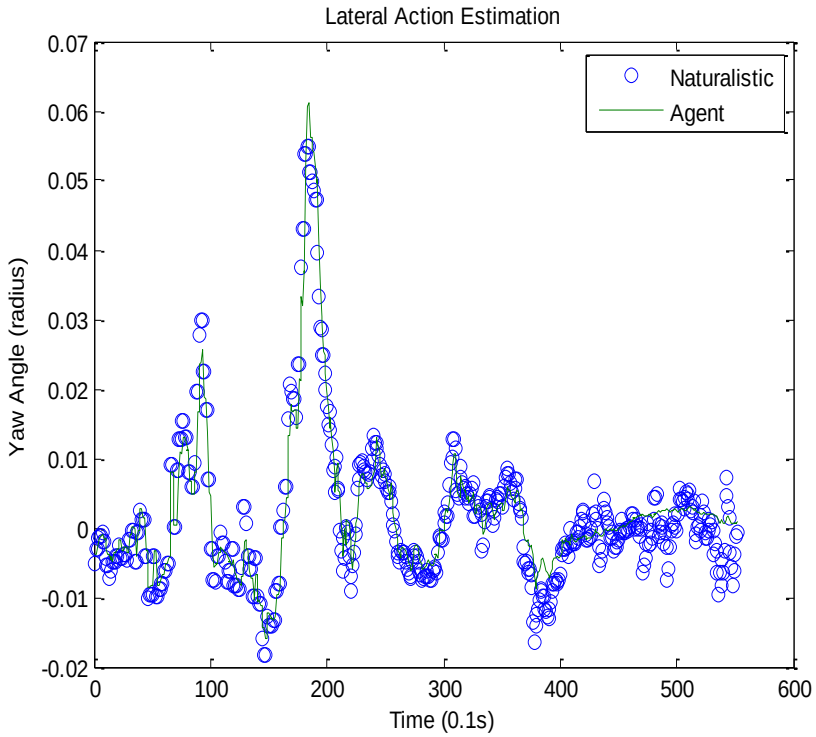
Driver Agent A: Own Behavior



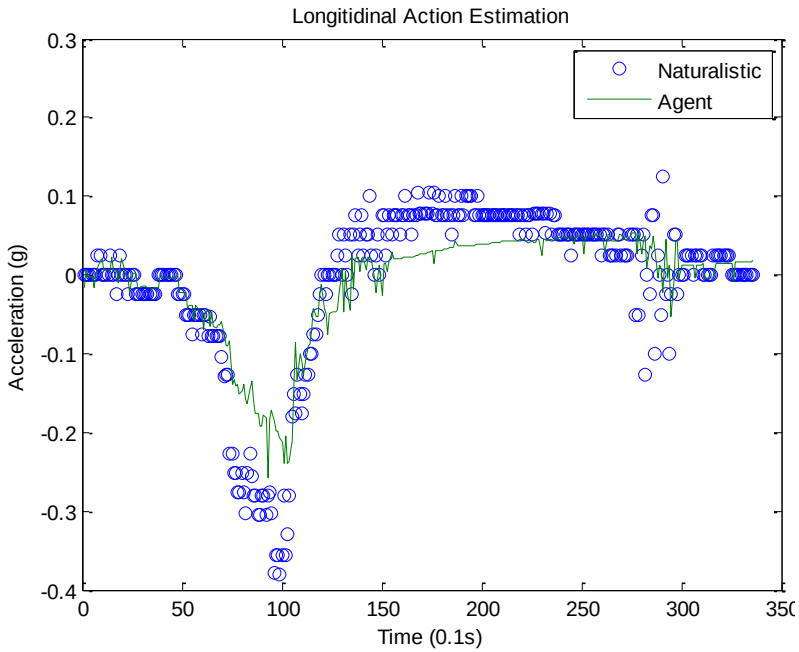


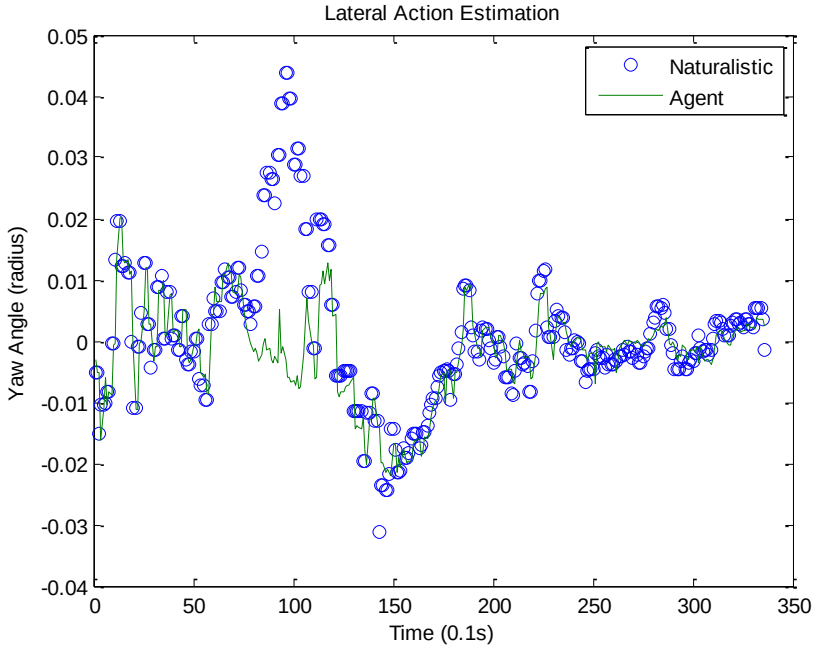
Driver B: Own Behavior





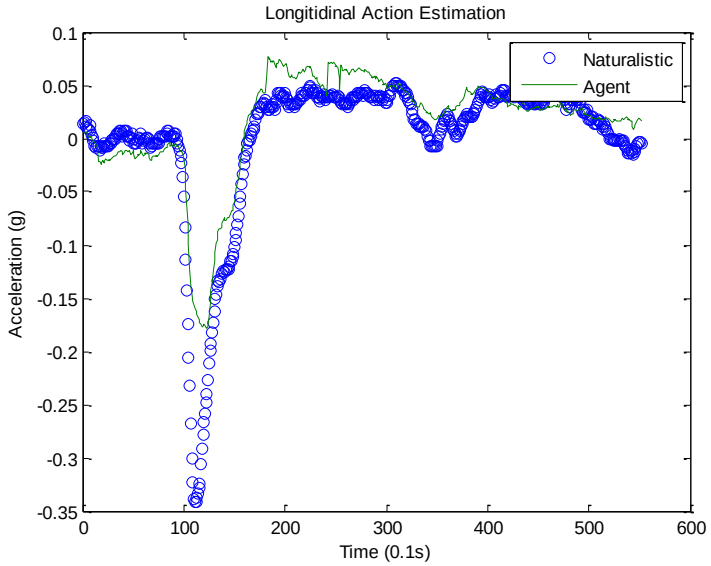
Driver A: Using Behavior from B

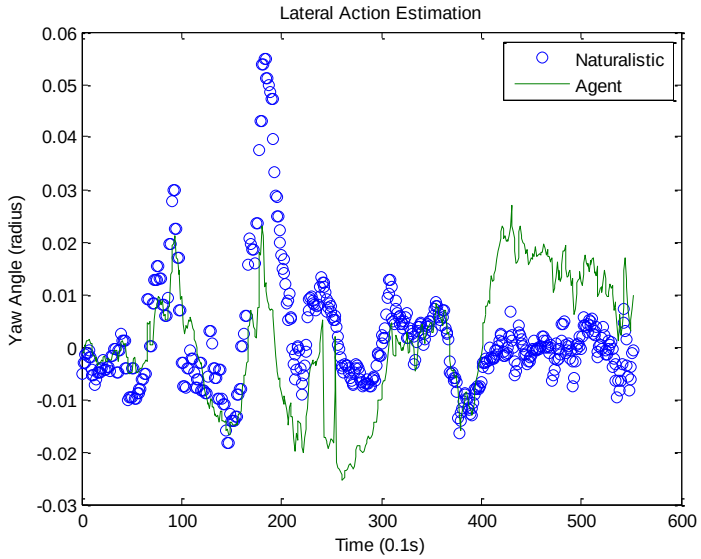




Driver B: Using Behavior from A

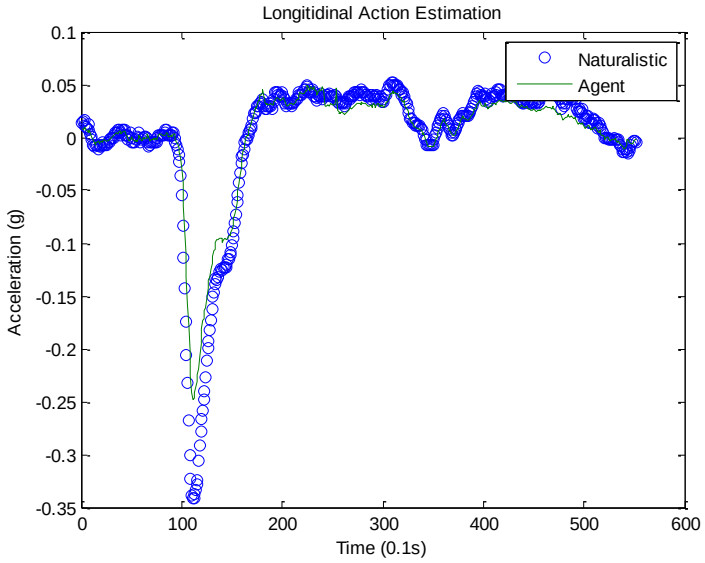
- Heterogeneity is clear

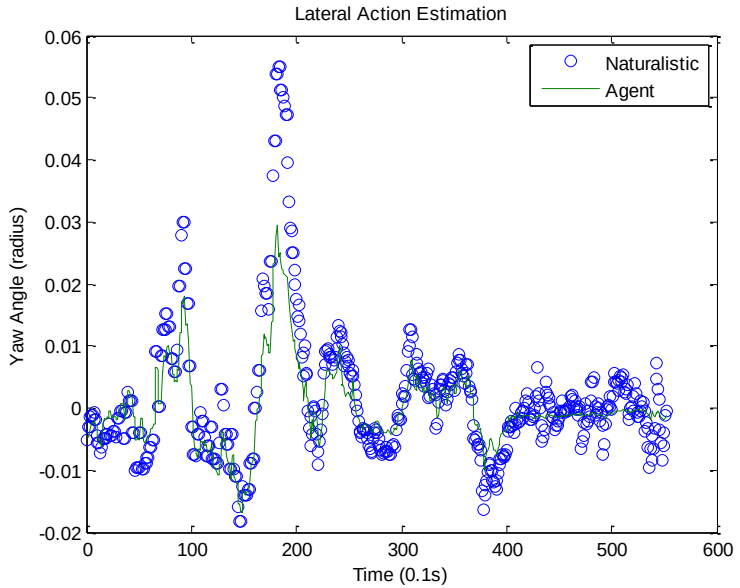




Mega-Agent Behavior

- Mega-Agent behaves as Driver B

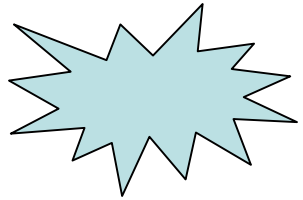




Comparison of Mega-Agent to Cross Validation Result

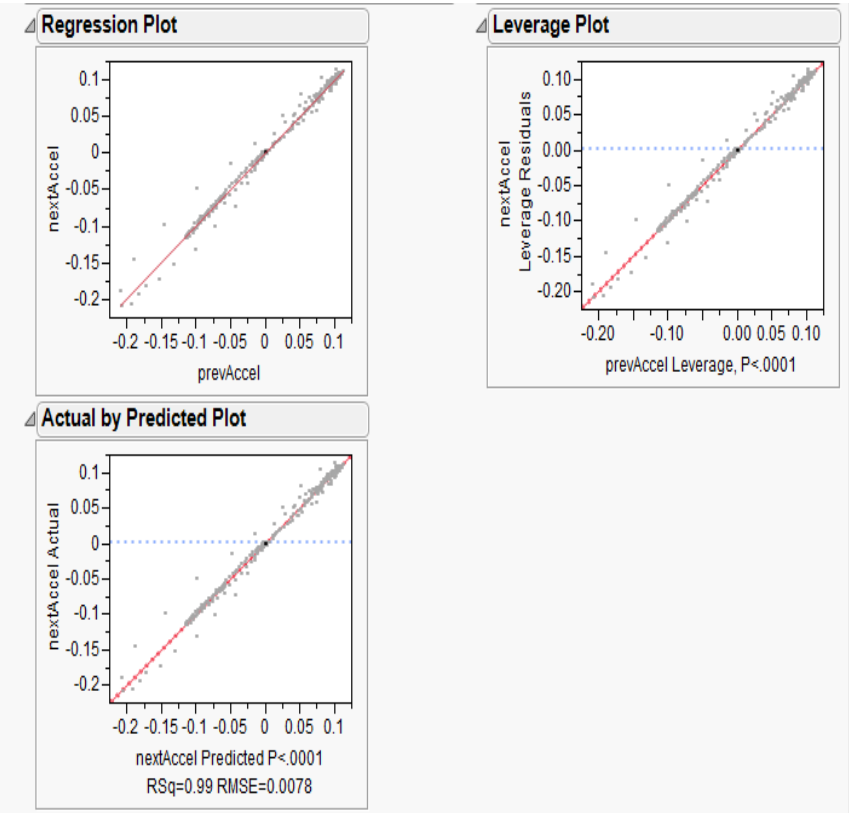
- Degree of accuracy: R square

Event	Agent A		Agent B		Mega	
	long	lat	long	lat	long	lat
Event A	0.98	0.967	0.81	0.83	0.98	0.95
Event B	0.82	0.6	0.97	0.92	0.97	0.9



But…Why NOT Statistical Modeling?

- Would lead to wrong conclusions!



Summary of Fit

RSquare	0.987222
RSquare Adj	0.987183
Root Mean Square Error	0.007776
Mean of Response	0.002671
Observations (or Sum Wgts)	333

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	1	1.5461451	1.54615	25572.66
Error	331	0.0200126	6.046e-5	Prob > F
C. Total	332	1.5661577		<.0001*

Lack Of Fit

Source	DF	Sum of Squares	Mean Square	F Ratio
Lack Of Fit	164	0.01603214	0.000098	4.1014
Pure Error	167	0.00398041	0.000024	Prob > F
Total Error	331	0.02001255		<.0001*
				Max RSq
				0.9975

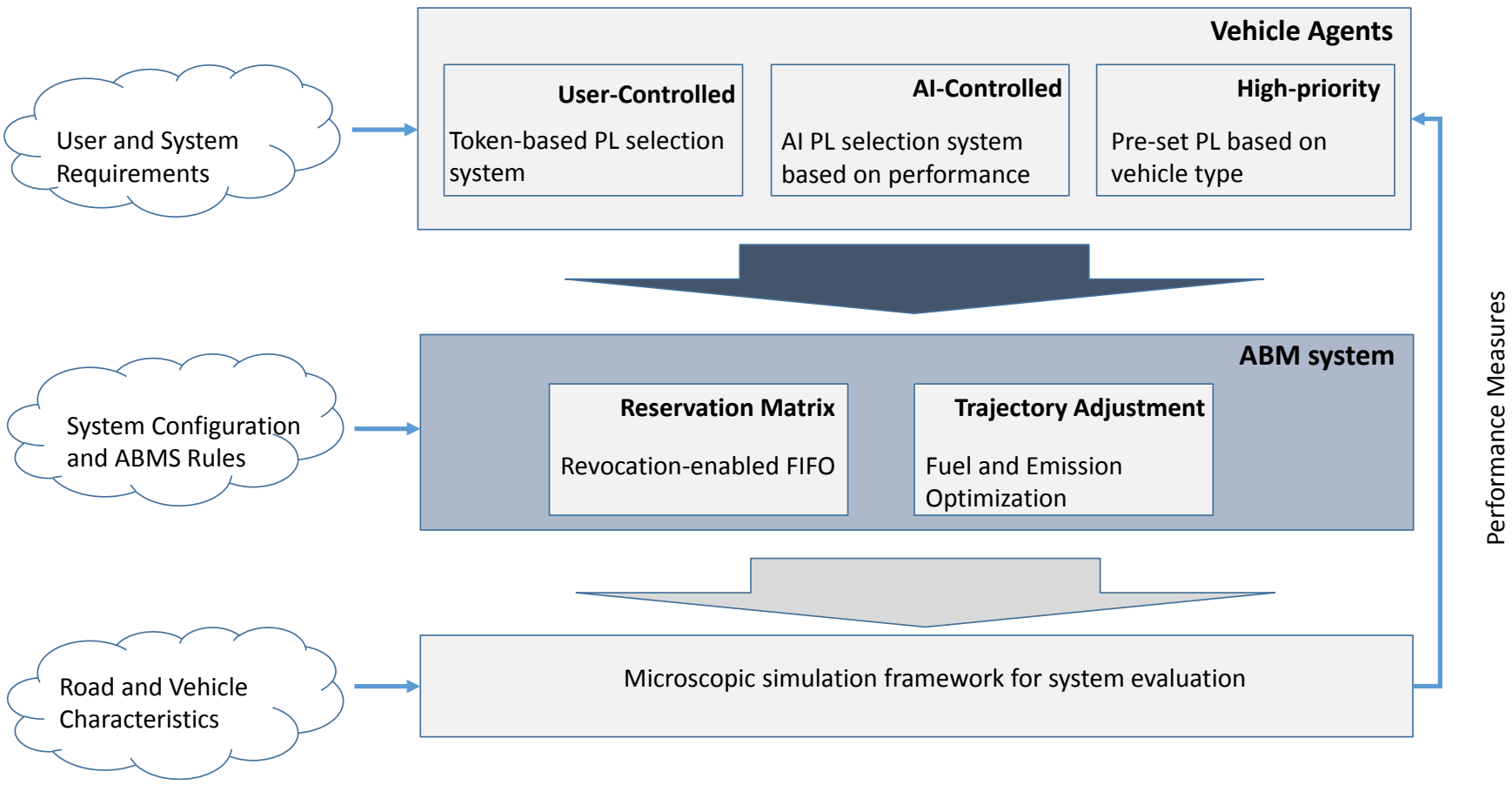
Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	1.8364e-5	0.000426	0.04	0.9657
prevAccel	0.9935897	0.006213	159.91	<.0001*

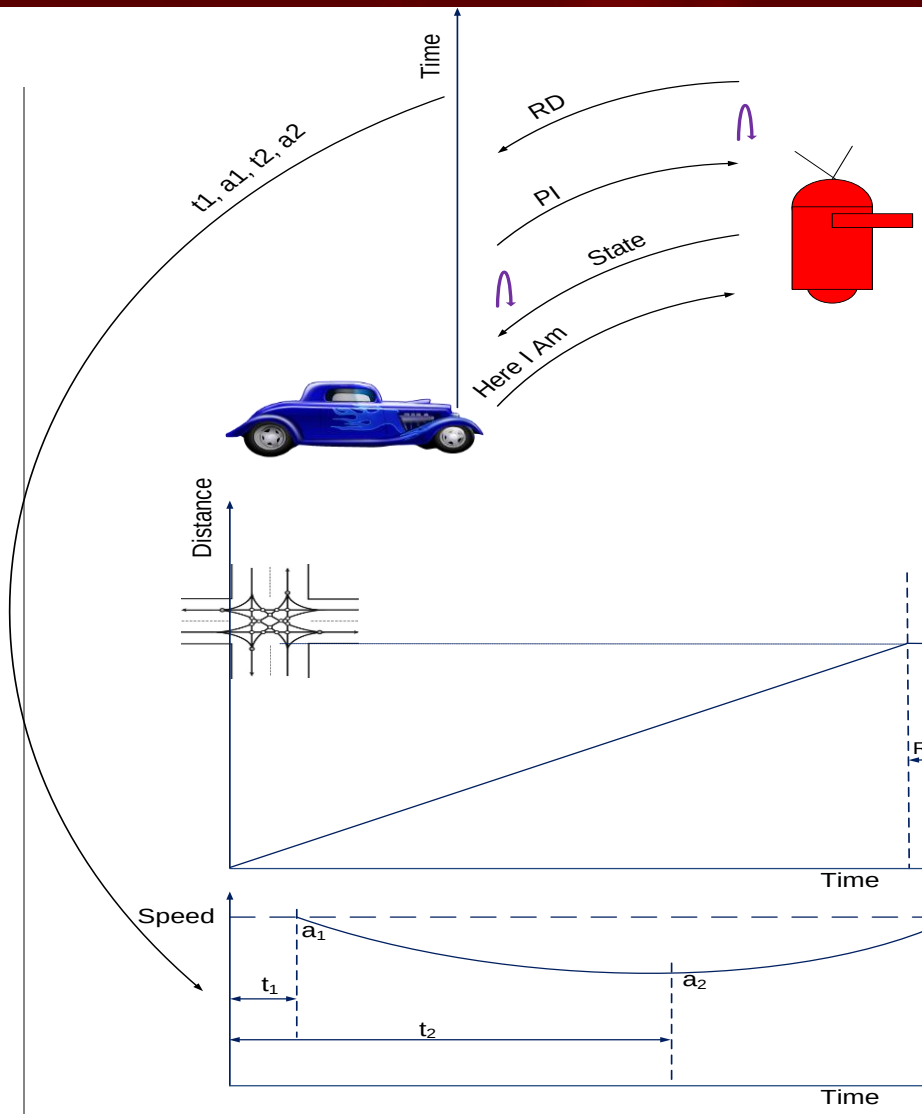
Future CV/AV Applications

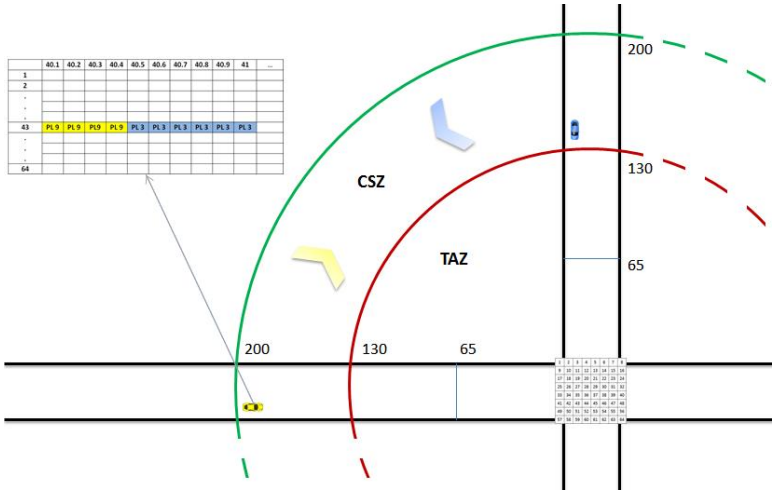
- Multi-modal applications: modeling, simulation, and optimization
- Accounting for different priorities, including emergency vehicles
- Utilization of the computing capabilities of CV/AV
- Linking arterial control to freeway management scenarios
- Characterizing and changing network performance

Multi-agent System Framework



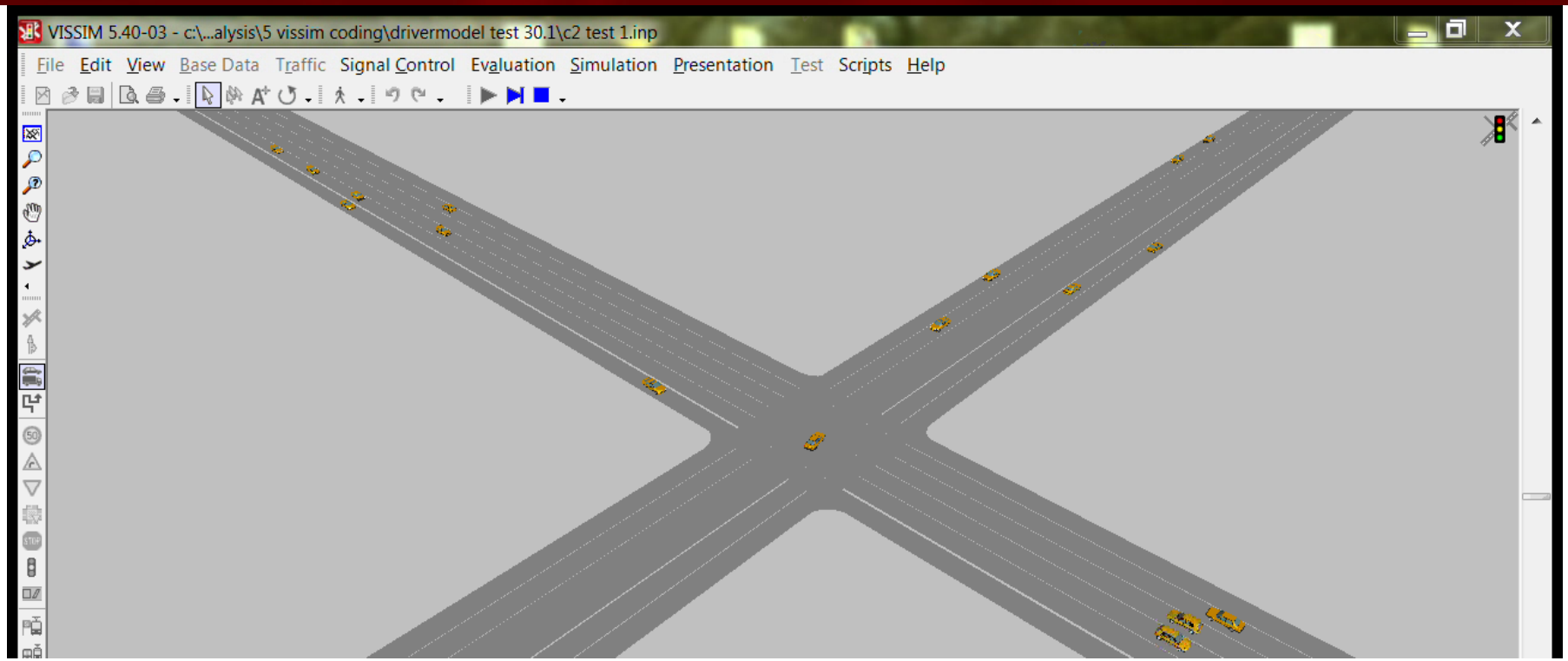
Multi-agent System Framework



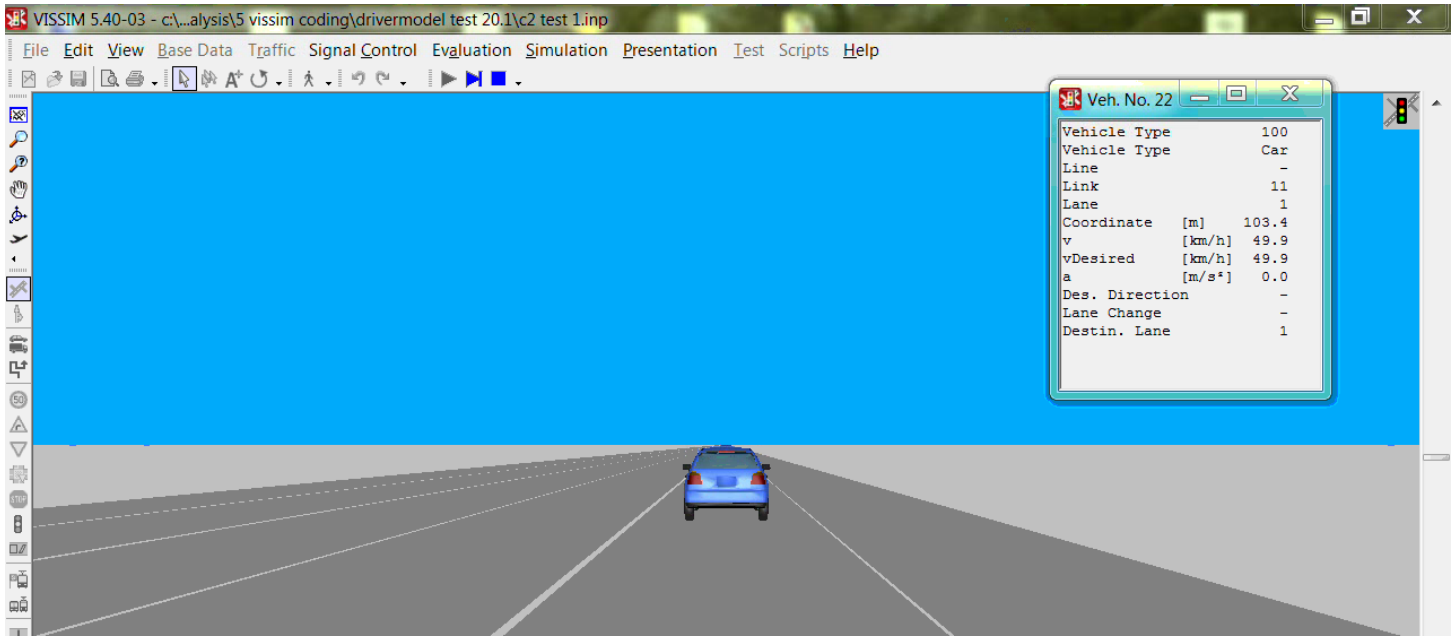


Required Delay for a vehicle after arriving at the intersection until higher priority vehicles clear all conflict tiles

Rather than driving with constant speed, come to a complete stop for a duration of RD before resuming speed, a vehicle follows a modified trajectory to delay its arrival by RD

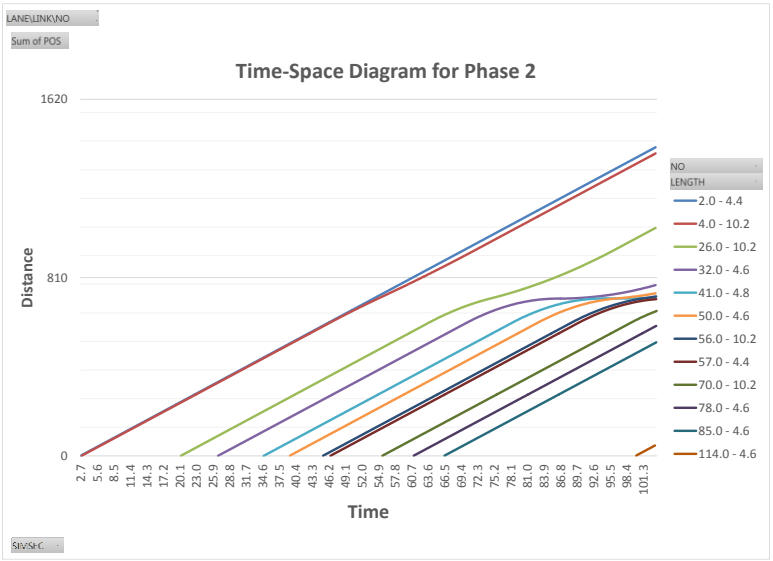

VISSIM 5.40-03 - c:\...alysis\5 vissim coding\drivermodel test 30.1\c2 test 1.inp
File Edit View Base Data Traffic Signal Control Evaluation Simulation Presentation Test Scripts Help

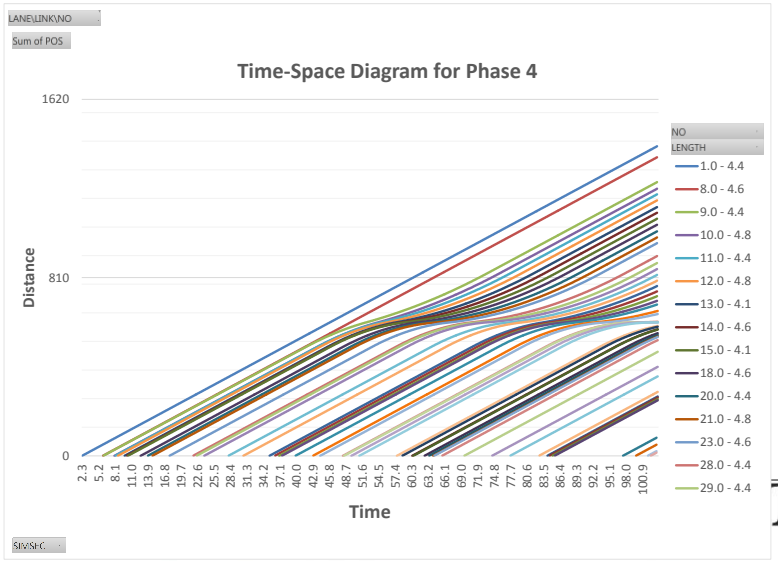
Negotiating an Intersection



Experiment Setup

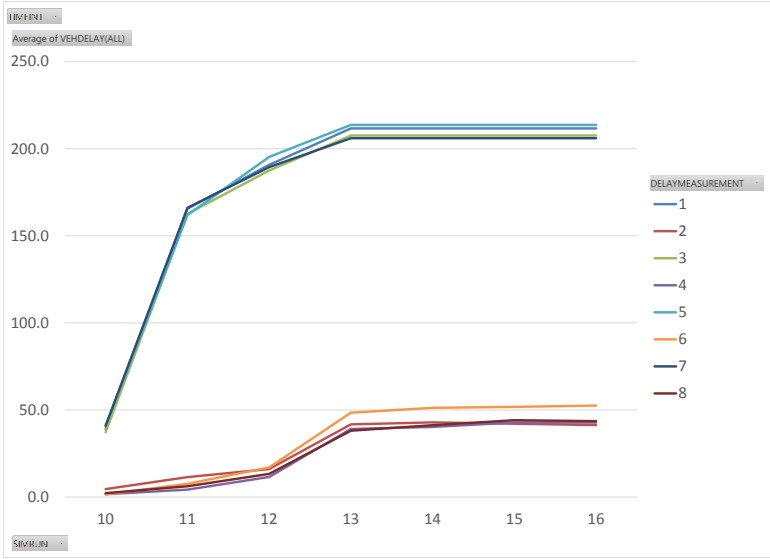
- Simulating high and low priority levels in some approaches
- Tabulated delay values and vehicle trajectories for different approaches





Experiment Results

- Agents adapt by forming dense platoons to pass through large gaps more efficiently
- Interesting emergent behavior can be observed from simple interaction rules
- Low priority agents are sensitive to traffic demand level
- Frequent EV calls re-synch the EV approach



Scenario	Phase								% EV, Ph2
	1	2	3	4	5	6	7	8	
	PL								
	1	2	1	3	1	2	1	3	4
10	200	200	200	200	200	200	200	200	0
11	400	400	400	400	400	400	400	400	0
12	400	600	400	600	400	600	400	600	0
13	400	800	400	800	400	800	400	800	0
14	200	200	200	200	200	200	200	200	10
15	400	400	400	400	400	400	400	400	20
16	400	600	400	600	400	600	400	600	30

Concluding Remarks

- Intelligent agents can capture individual learning, and agent-based modeling can capture the emerging system behavior
- Think state-action framework…it can explain a lot of things
- Win the chess game, not just the next move