RDECOM

Integration of Hardware-in-the-loop Facilities Over the Internet

Hardware-in-the-loop User Group Meeting
April 15, 2009

TARDEC

TECHNOLOGY DRIVEN. WARFIGHTER FOCUSED.

Mark J. Brudnak, Ph.D., U.S. Army RDECOM-TARDEC
Wilford Smith, SAIC
Jarrett Goodell, SAIC

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **15 APR 2009**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**

4. TITLE AND SUBTITLE **Integration of Hardware-in-the-loop Facilities Over the Internet**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Mark J. Brudnak; Wilford Smith; Jarrett Goodell**	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **US Army RDECOM-TARDEC 6501 E 11 Mile Rd Warren, MI 48397-5000**	8. PERFORMING ORGANIZATION REPORT NUMBER **19796RC**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) **TACOM/TARDEC** 11. SPONSOR/MONITOR'S REPORT NUMBER(S) **19796RC**

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
Presented at the Hardware-in-the-loop User Group Meeting, 15 APR 2009, Wright Patterson AFB, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**SAR**	**57**	

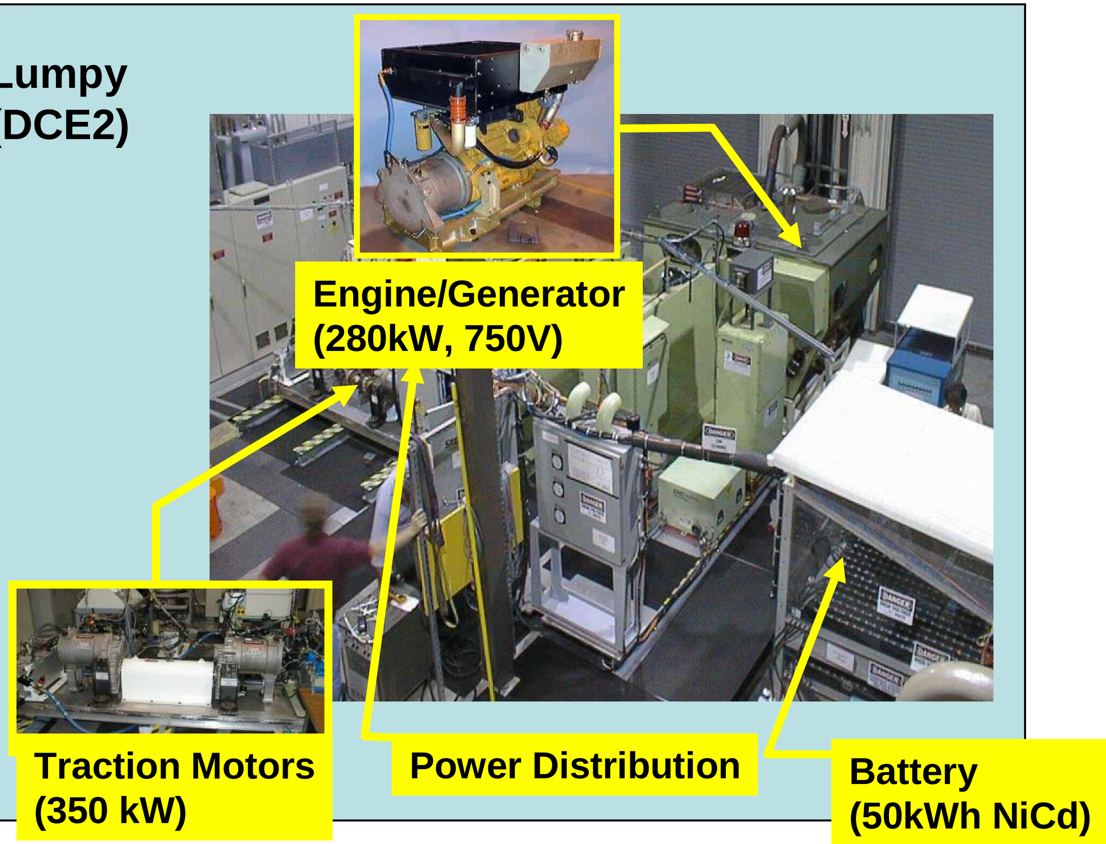
Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Power & Energy SIL

P&E SIL

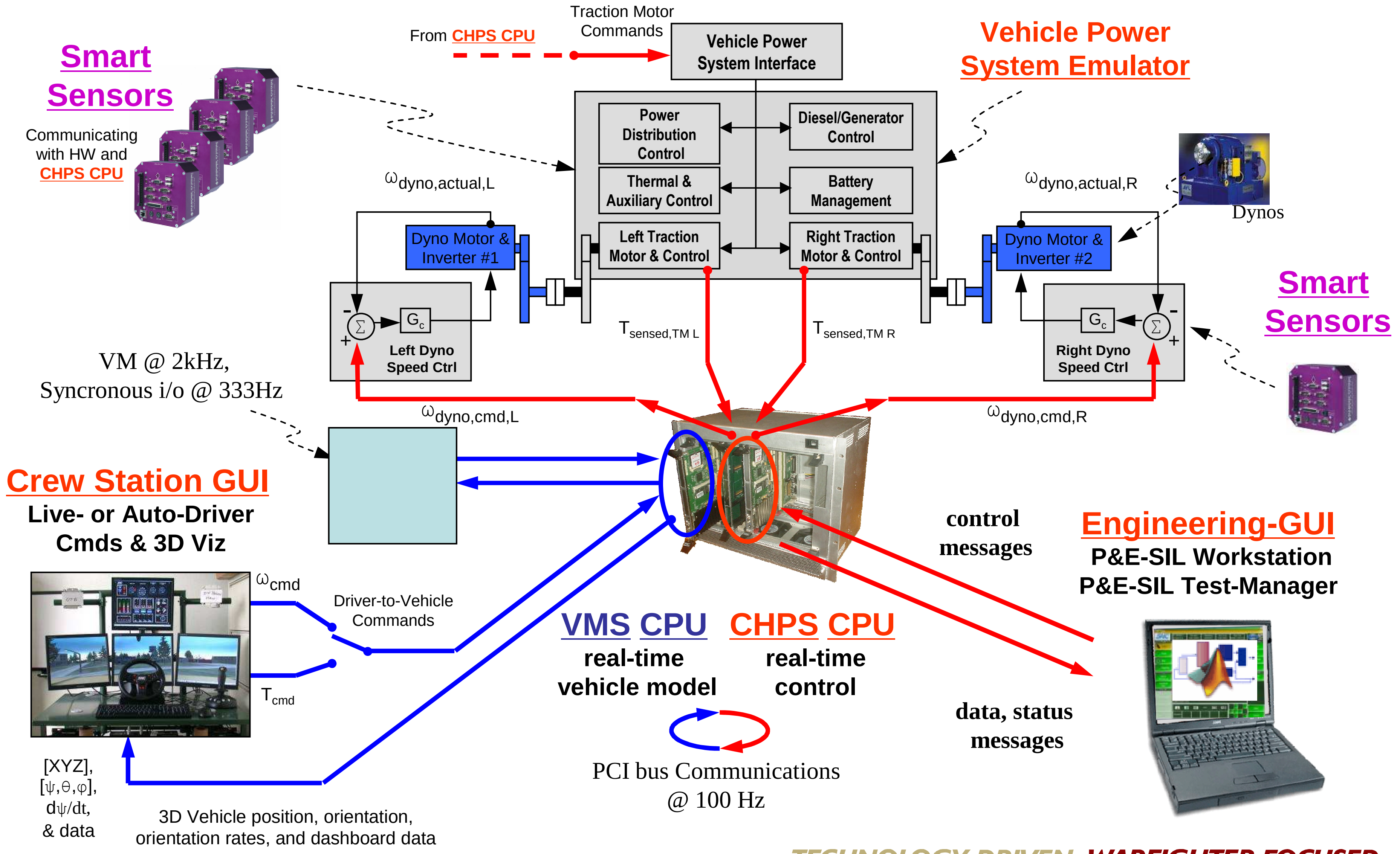
- Series Hybrid-electric power system
- Laboratory based evaluation of design alternatives
- Driven by automated controller
- Requires a-priori duty cycle

Lumpy (DCE2)

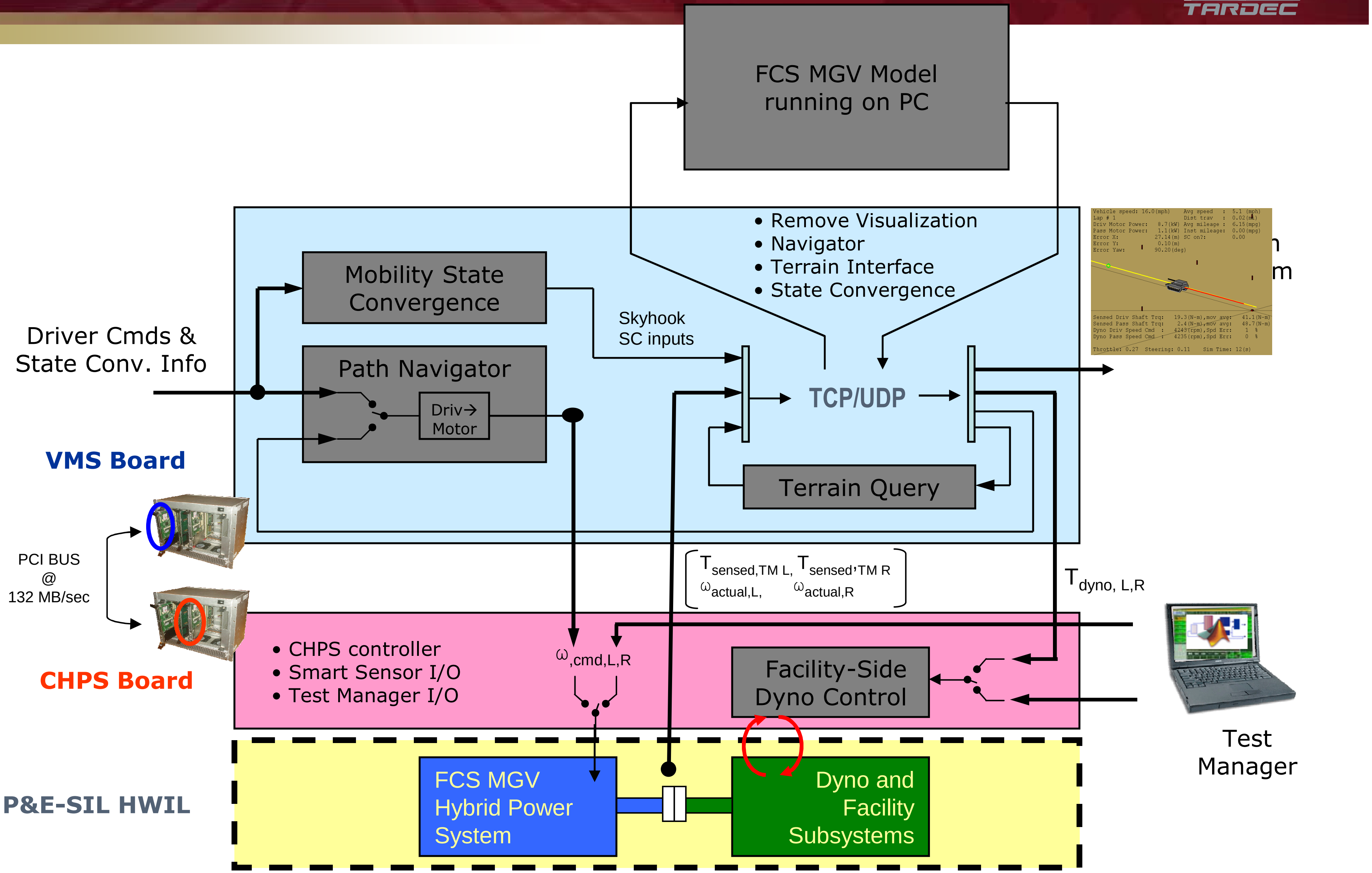


Hermit (DCE3)

SIL HWIL and Driver-in-the-Loop Layout

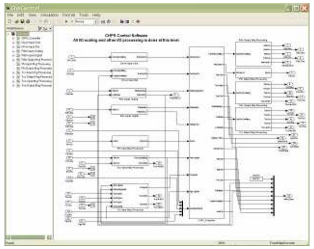


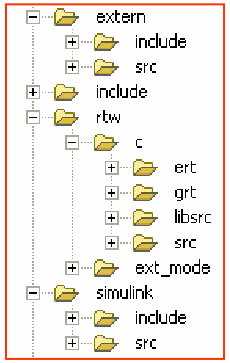
P&E-SIL Architecture



P&E-SIL – Automation of Code Generation and Integration

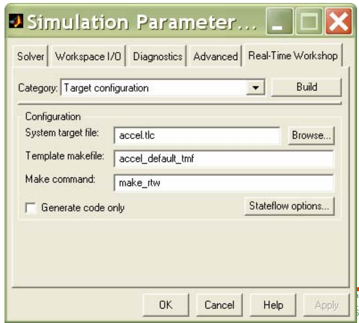
Graphical code



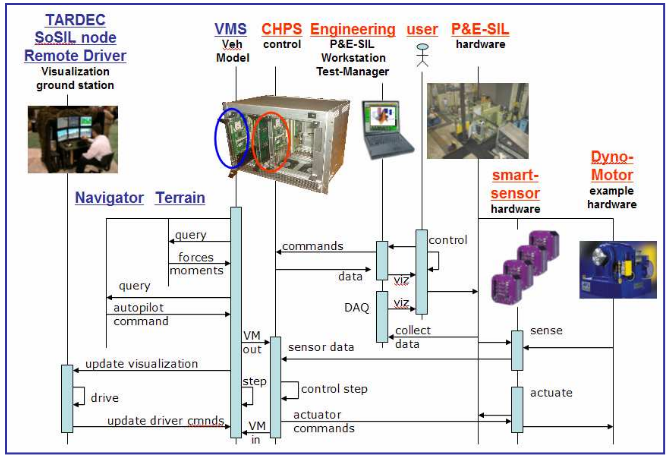


compiled libraries

Conversion to Real-time



embedded hardware



Embedded C/C++ code

```c
...te for enable system:
...e State'
...tate'
void ChpControl_Enable_State(real_T fu0, real_T FaultCmd,
 rtB_ChpControl_Enable_State *localB, rtP_ChpControl_Enable_State *localP)
{
  if (fu0 > 0.0) {

    /* Gain: '<S186>/Gain1'
     *
     * Regarding '<S186>/Gain1':
     *   Gain value: localP->Gain1_Gain
     */
    localB->Gain1 = FaultCmd * localP->Gain1_Gain;
  }
}

/* Output and update for trigger system:
 *   '<S216>/Subsystem3'
 *   '<S206>/Subsystem1'
 */
void ChpControl_Subsystem3(real_T fu0, real_T reset, rtB_ChpControl_Subsystem3
 *localB, rtP_ChpControl_Subsystem3 *localP, rtZCE_ChpControl_Subsystem3
 *localZCE)
{
  if (rt_ZCFcn(RISING_ZERO_CROSSING, &(localZCE->Subsystem3_ZCE), fu0)) {

    /* Switch: '<S220>/Switch' incorporates:
     *   Constant: '<S220>/clear'
     *   Constant: '<S220>/set'
     */
    if (reset >= localP->Switch_Threshold) {
      localB->Switch = localP->clear_Value;
    } else {
      localB->Switch = localP->set_Value;
    }
  }
}

/* Model step function */
void ChpControl_step(void)
{
  /* local block i/o variables */
  real_T rtb_All_battery_switches_open;
  real_T rtb_Unit_Delay3_a;
  real_T rtb_Rounding_Function_a;
  real_T rtb_Math_Function_a;
  real_T rtb_Math_Function_b;
  real_T rtb_Unit_Delay1_a[6];
  real_T rtb_Logical_Operator1_a;
  real_T rtb_Logical_Operator8_a;
```

OS-interface code

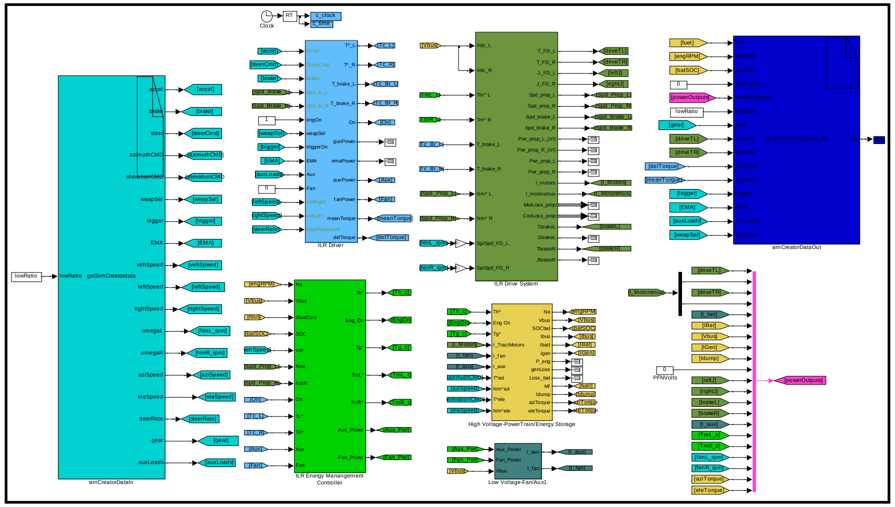
```cpp
////////////////////////////////////////////////////////////////////////
template<class T>
void SAICrtw<T>::rt_OneStep(void)
{
    static int missedFrameCounter=0;

    // **********************************************
    // * Check and see if base step time is too fast *
    // **********************************************
    if (OverrunFlag++)
    {
        rtmSetErrorStatus(RT_MDL, "Overrun");

          missedFrameCounter++;
          printf("choking on OVERRUNFLAGS:  %i\n",missedFrameCounter);

        //return;
    }
    // ------------------------------------------------

    // ************************************************
    // * Check and see if an error status has been set *
    // * by an overrun or by the generated code.       *
    // ************************************************
    if ( rtmGetErrorStatus( RT_MDL ) != NULL )
    {
        printf("\n rt_OneStep() :: overrun problems \n");
        return;
    }
    // ------------------------------------------------

    // **************
    // * Step model *
    // **************
     MODEL_STEP();

    // ************************
    // * Decrement overrun flag *
    // ************************
    OverrunFlag--;

    return;

} /* rtOneStep */
////////////////////////////////////////////////////////////////////////
```

RDECOM

Simulation Design – Power System

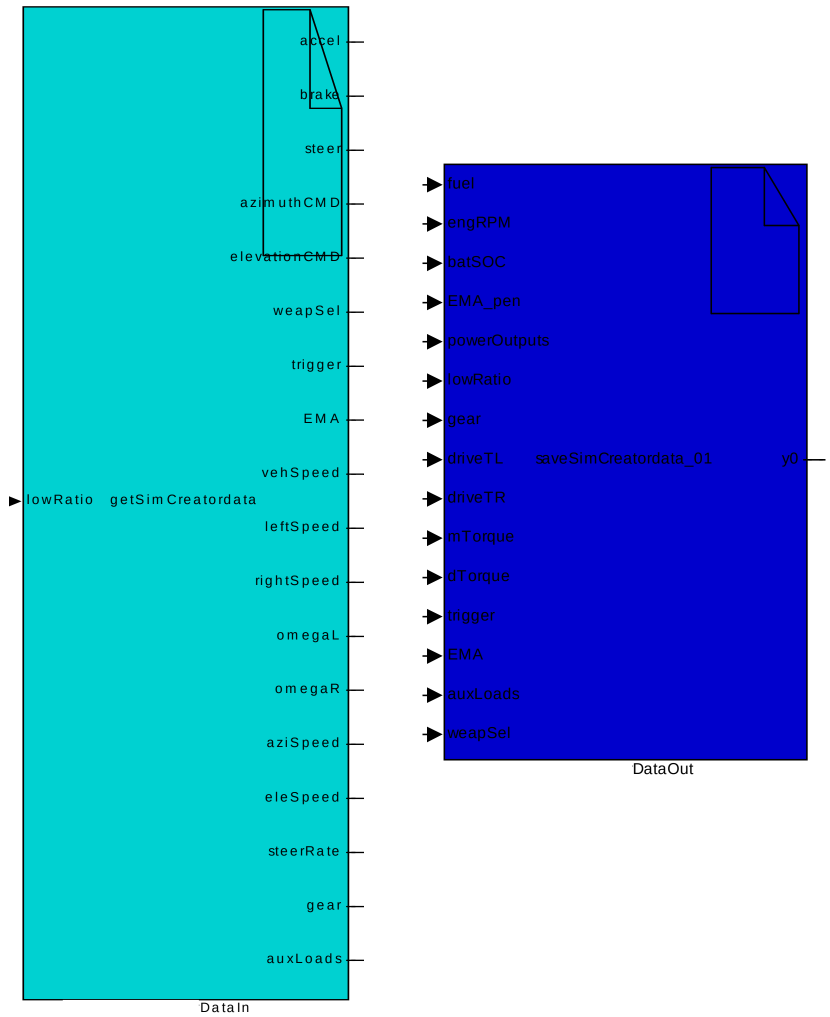
TARDEC

- Model: GVSL
  - Assumptions: SS torque transfer in gear train,
  - Current Power Systems: 420kW Engine, 410kW Generator, 500kW Battery (consistent with 24-ton FCS-like vehicle)
- Blocks:
  - Input
    - *Inputs from the Vehicle*
  - Output
    - *Outputs to the Vehicle*
  - High Voltage Powertrain & Energy Storage
    - *CHPSPerf- Engine, Generator, & Battery*
    - *420kW Turbo-Diesel Engine/ 410kWGenerator*
    - *Li-Ion Battery direct connected to 510V Bus*
    - *Dump Resistor*
    - *Turret Azimuth & Gun Elevation Motors*
  - Drive System
    - *Independent Left/Right Motors (ILR), transmission, & brakes*
    - *Gears include Coulombic, Viscous, and Mesh gear losses*
  - Energy Management
    - *Power Generation and Motor controllers (translates commanded torques to machine torques)*
  - Driver
    - *Speed-based mobility control (throttle/steer commands to torque commands) for low vehicle speeds*
    - *Torque-based mobility control for high vehicle speeds*
    - *Blended mobility control for mid vehicle speeds*
  - Low Voltage (Fan/Aux)
- Input File *gvsl_Input.m* defines input parameters



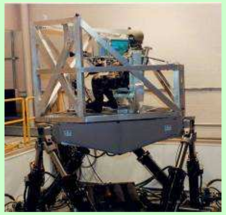
Simulation Design: Power System/SimCreator Interface

- Links with shared workspace with the vehicle module
- Input Block:
  - Commands/States from vehicle model: *accel, brake, steer, gun azimuth/elevation, gun trigger, EMA, speeds/rates (vehicle, track, sprocket, turret azimuth/elevation, steer), gear, aux. loads*
  - Includes speed sign management for neutral and reverse
- Output Block:
  - Power outputs (*propulsion & turret motor torques, sprocket & brake torques, mean & delta torque, fan/battery current & voltage, generator & dump resistor current, PFN voltage, Aux system current*), fuel, engine speed, battery SOC, EMA/gun readiness, Left & Right MOIs, gear.
  - Includes torque sign management for neutral and reverse



RemoteLink Internet ICD,
v1.11

GVSL
Warren
MI



VehModelFaults > 0 for Inf or NaN trapped on inputs/outputs, max speed or Euler angle exceeded

udp ports 5100, 5101

udp ports 5115, 5116

GVSL_out

52 floats, 208 bytes

Net QOS	38		sim_time_gvsl (s)
	39		sim_time_sil (s)
	40		round trip delay gvsl (s)
	41		round trip delay sil (s)
	42		GVSL_out update rate at SIL (Hz)
	43		SIL_out update rate at GVSL (Hz)
Health Status	50		Veh_dynamics_up (0/1)
	51		Power_system_SC_up (0/1)
	52		ESS_up (0/1)

SIL_out

104 floats, 116 bytes

87		sim_time_gvsl (s)	Net QOS
88		sim_time_sil (s)	
89		round trip delay gvsl (s)	
90		round trip delay sil (s)	
91		GVSL_out update rate at SIL (Hz)	
92		SIL_out update rate at GVSL (Hz)	
102		VMS/veh dyn Status (0/1)	Health Status
103		HWIL Status (0/1)	
104		SC Status (0/1)	

Initiate SIL HW shutdown
if any below are TRUE:

- `GVSL_Veh_Dyn_up        == 0`
- `GVSL_Pwr_Sys_SC_up     == 0`
- `GVSL_ESS_up            == 0`
- `Net_data_delay > 10s   == 1`

SIL
San Jose
Ca

Signal SIL HW health with:

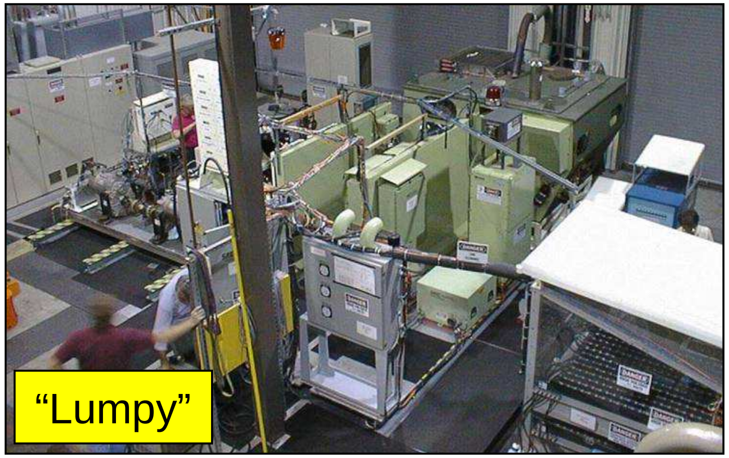
- `SIL_Veh_Dyn_Faults  == 0`
- `SIL_HWIL_Faults     == 0`
- `SIL_SC_Faults       == 0`

Motion Base Simulators

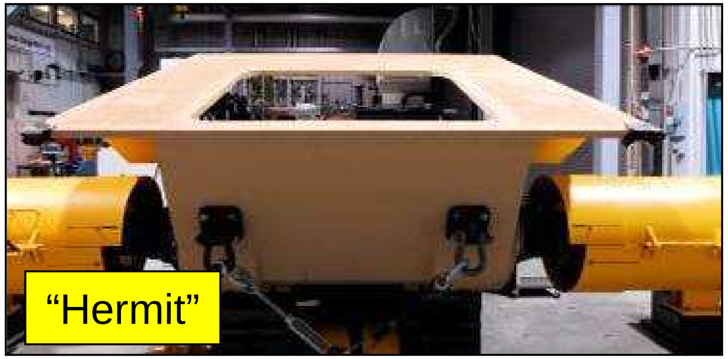
- Man-rated motion base simulator
- Integrated immersive simulation environment
- Real-time vehicle model
- Integrated CAT Crewstation
- Ideal facility for capturing soldier behavior (i.e. duty cycles)

Duty Cycle Motivation

- The Power & Energy SIL in Santa Clara, CA.
  - Series hybrid electric power system
  - Mobility loads:
    - Traction drive motors
  - Non-mobility loads:
    - Constant on/off loads
    - Time varying loads
    - Pulse power loads
- Non-mobility and mobility loads need to share the available power.
- What is the impact of power management choices?
- How should components be sized?
- Simple drive cycles were inadequate.
- → Need a relevant *Duty Cycle*



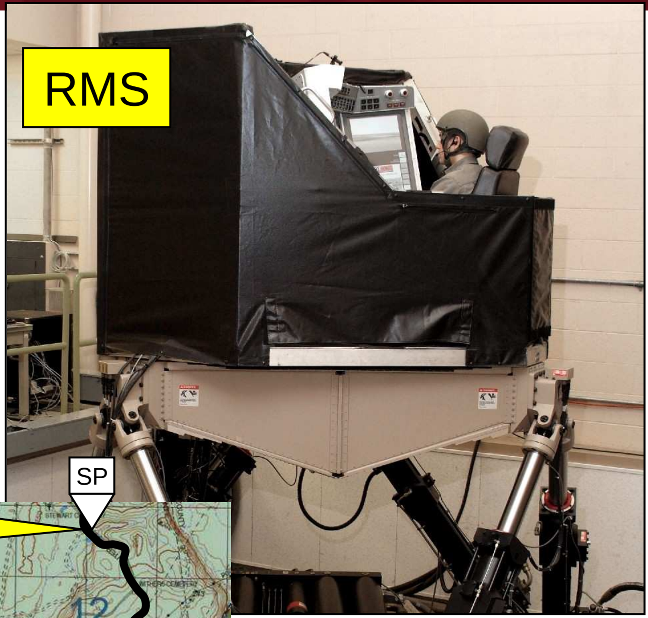
“Lumpy”

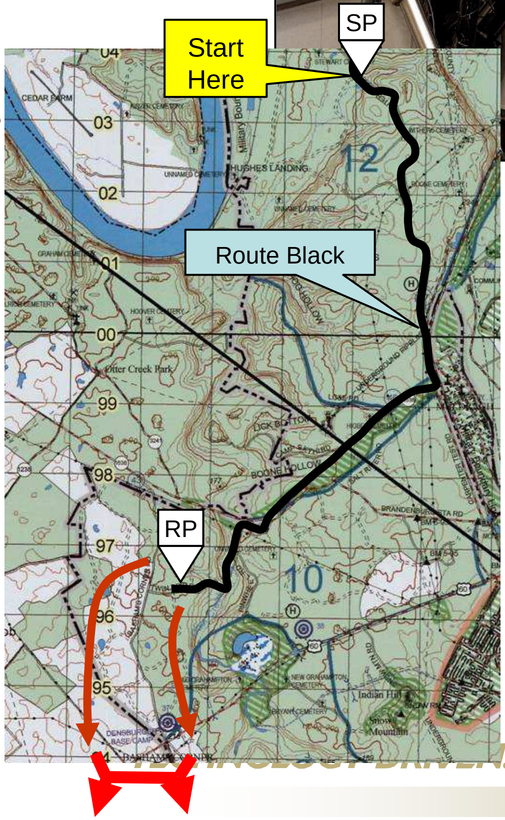


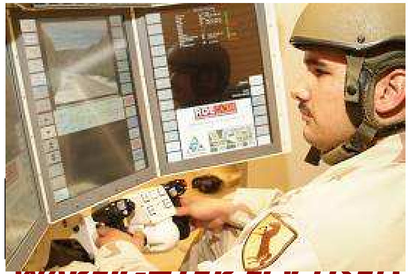
“Hermit”

Duty Cycle Experiment 2

- 12 trained Army driver/gunner subjects
- 13 km route
- Avg 42 minutes driving
- Grades greater than 30%
- 7 engagements with OneSAF opposition force infantry and vehicles







Historical Survey

- Duty Cycle Experiment 2 (DCE2): June 2006

Driver/Gunner/Chassis (TARDEC)

Power Train (SIL)

- Duty Cycle Experiment 3 (DCE3): June 2007

Driver/Gunner/Chassis (TARDEC)

Power Train (SIL)

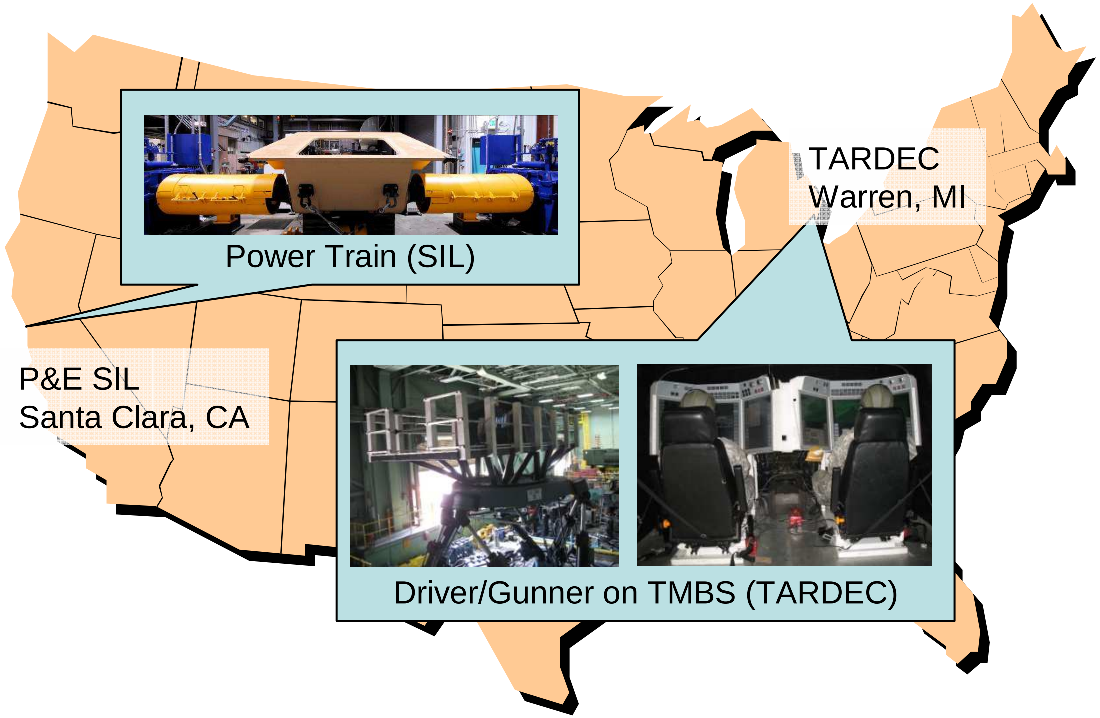
- ILIR: FY08

Driver/Gunner/Chassis (TARDEC)

Engine (Ann Arbor, MI)

Long Haul Motivation

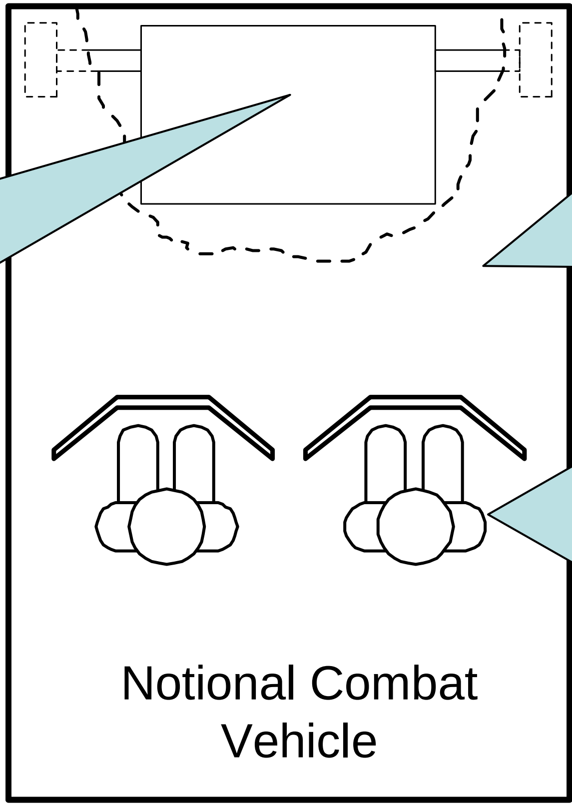
- Geographically disbursed.
- High-fidelity.
- Integration improves fidelity of experiments.
- Dynamical systems, the RMS and P&E SIL like tight loops. Substantial delays introduce instabilities.



2,450 Miles by Roads
2,080 Miles by Direct Route

Concept

Power Train (SIL)

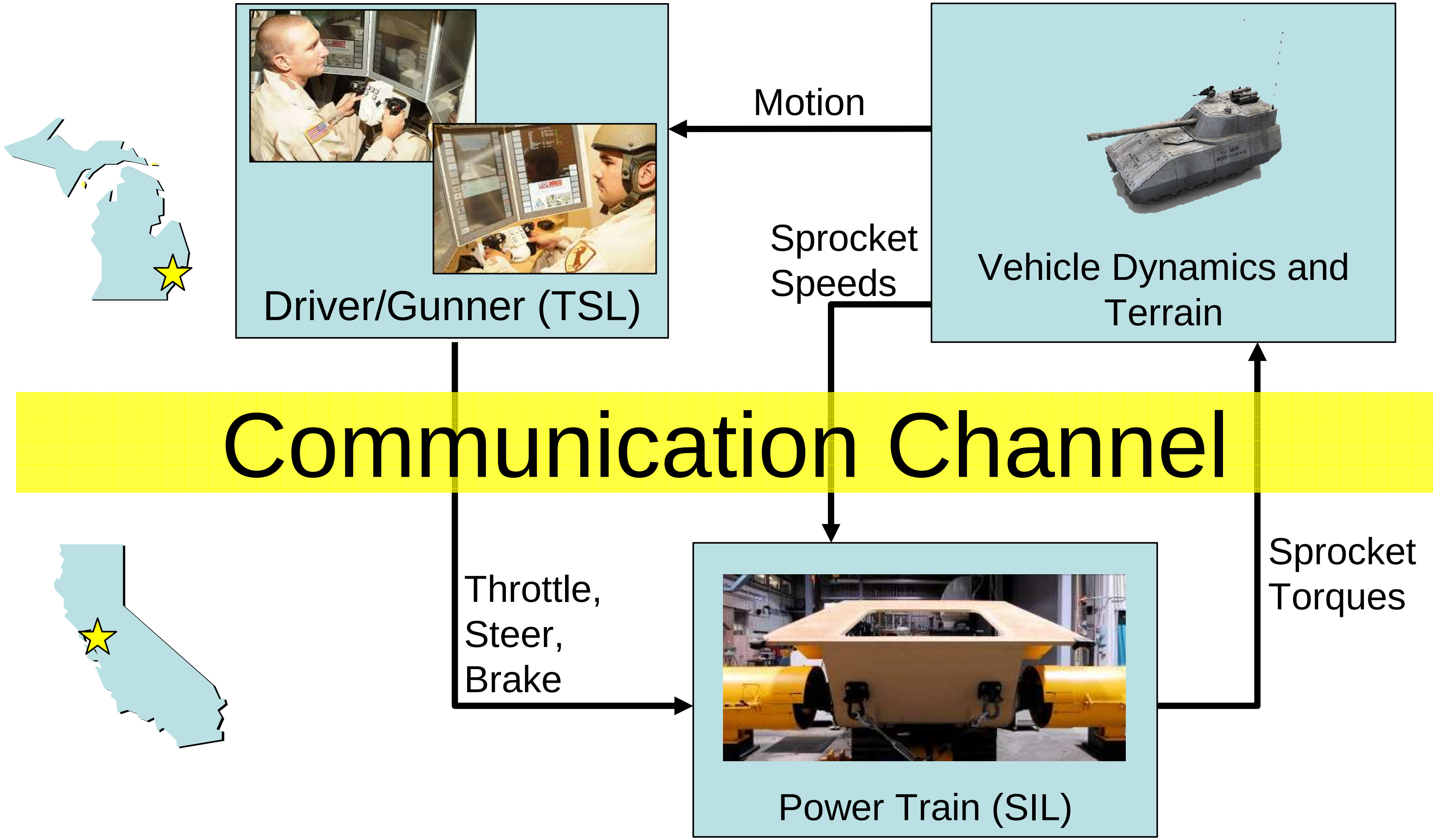


Vehicle Dynamics and Terrain (TARDEC)

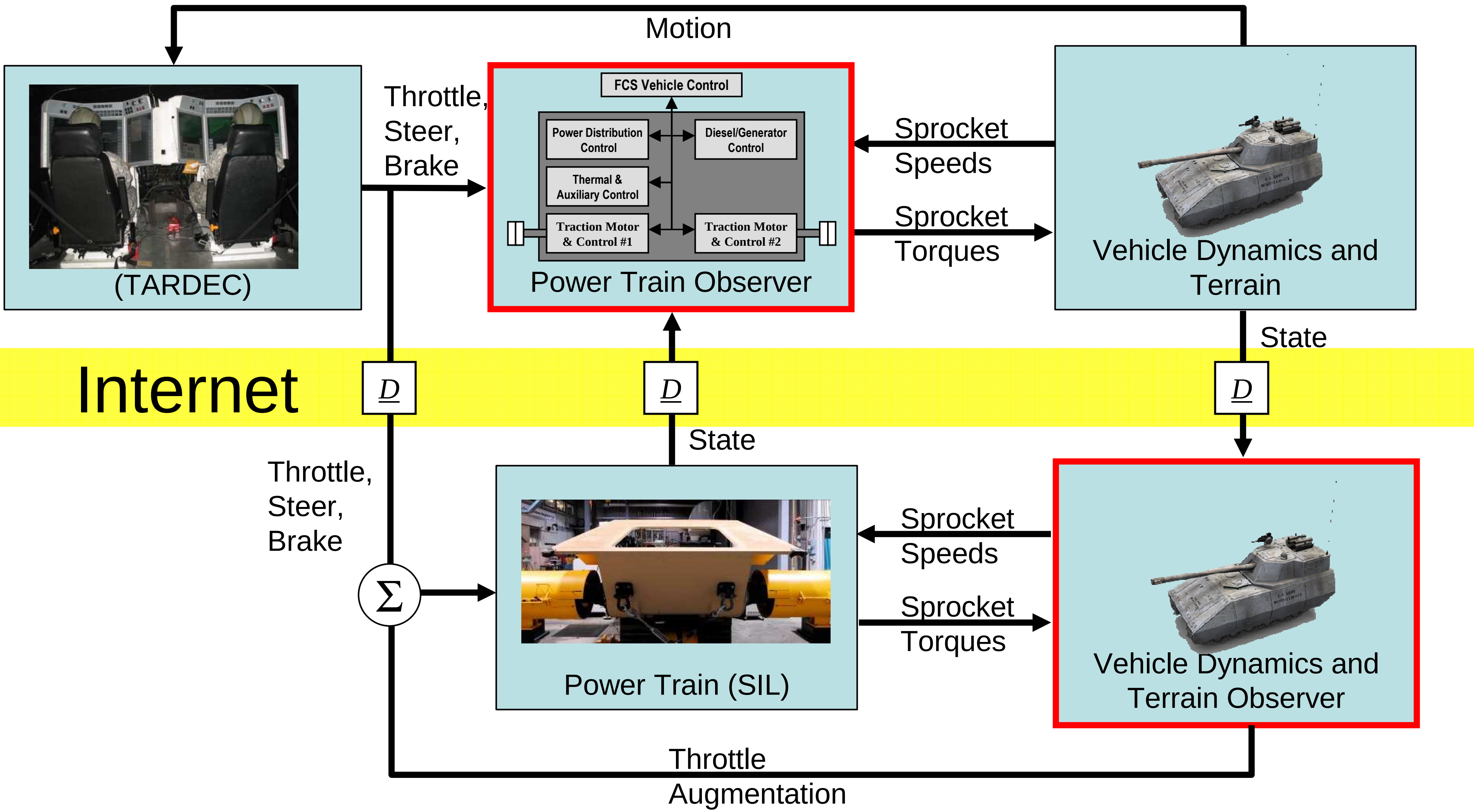
Driver/Gunner (TARDEC)

- Remote location of power system is transparent to the operators.

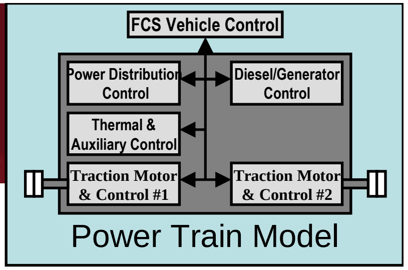
Interconnections



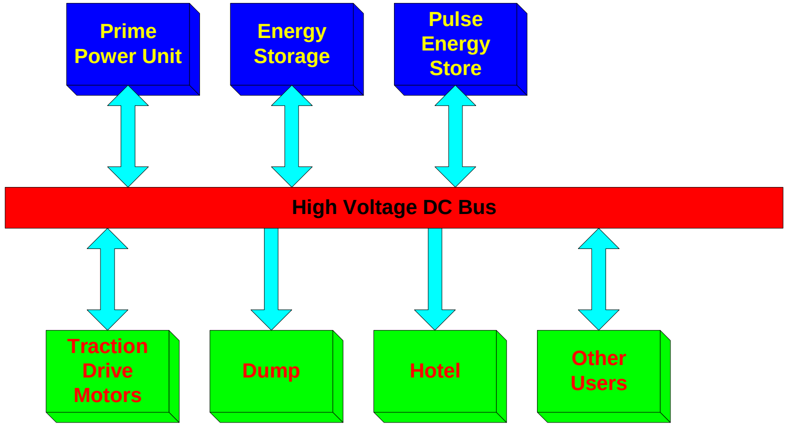
End Design

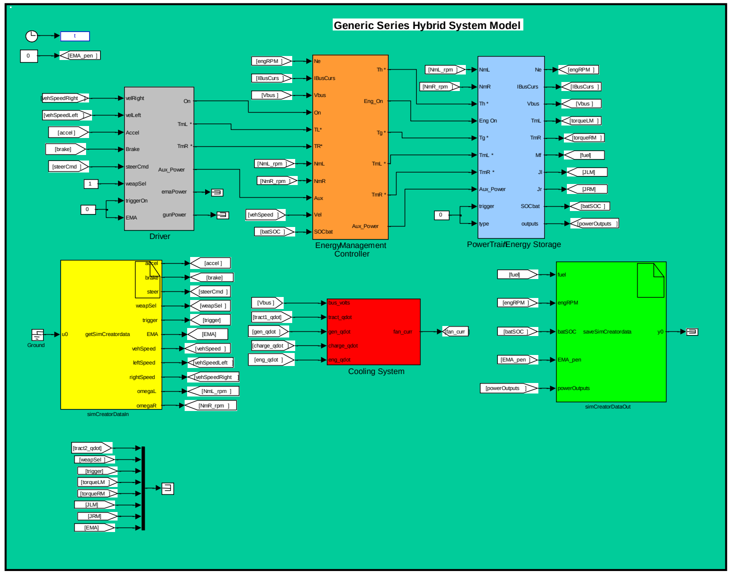

Motion
Throttle,
Steer,
Brake
FCS Vehicle Control
Power Distribution Control
Diesel/Generator Control
Thermal & Auxiliary Control
Traction Motor & Control #1
Traction Motor & Control #2
Power Train Observer
Sprocket Speeds
Sprocket Torques
Vehicle Dynamics and Terrain
(TARDEC)
State
Internet
D
D
D
State
Throttle,
Steer,
Brake
Σ
Power Train (SIL)
Sprocket Speeds
Sprocket Torques
Vehicle Dynamics and Terrain Observer
Throttle Augmentation

Power System Model



- Series Hybrid Power System for MCS
- Independent Left/Right
- Diesel Engine/Generator
- 600 V bus w/Battery
- Two 300kW traction motors.
- Includes thermal model
- Implemented in graphical modeling tool and converted to real-time code.



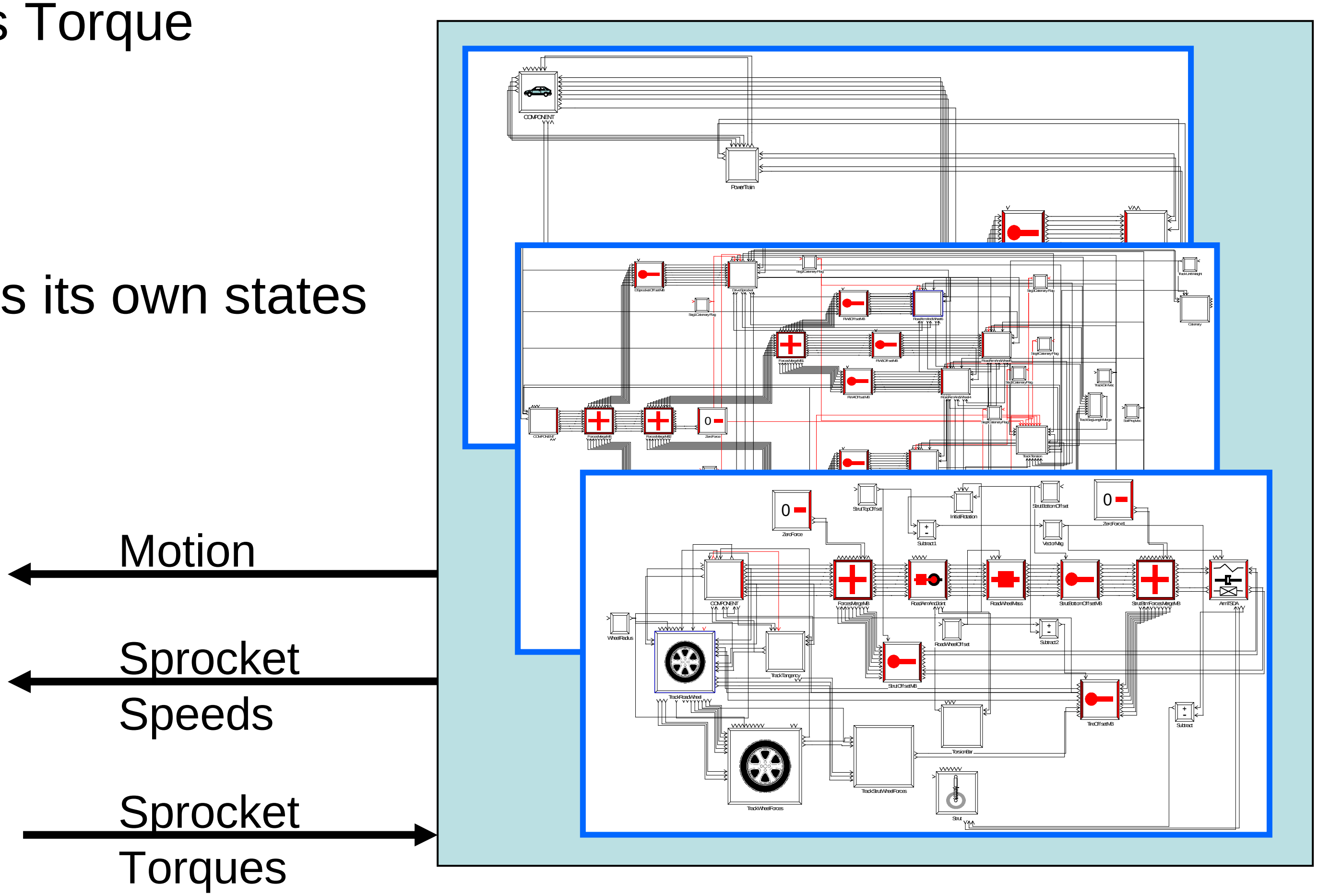


RDECOM

Vehicle Dynamics and Terrain

Vehicle Dynamics and Terrain

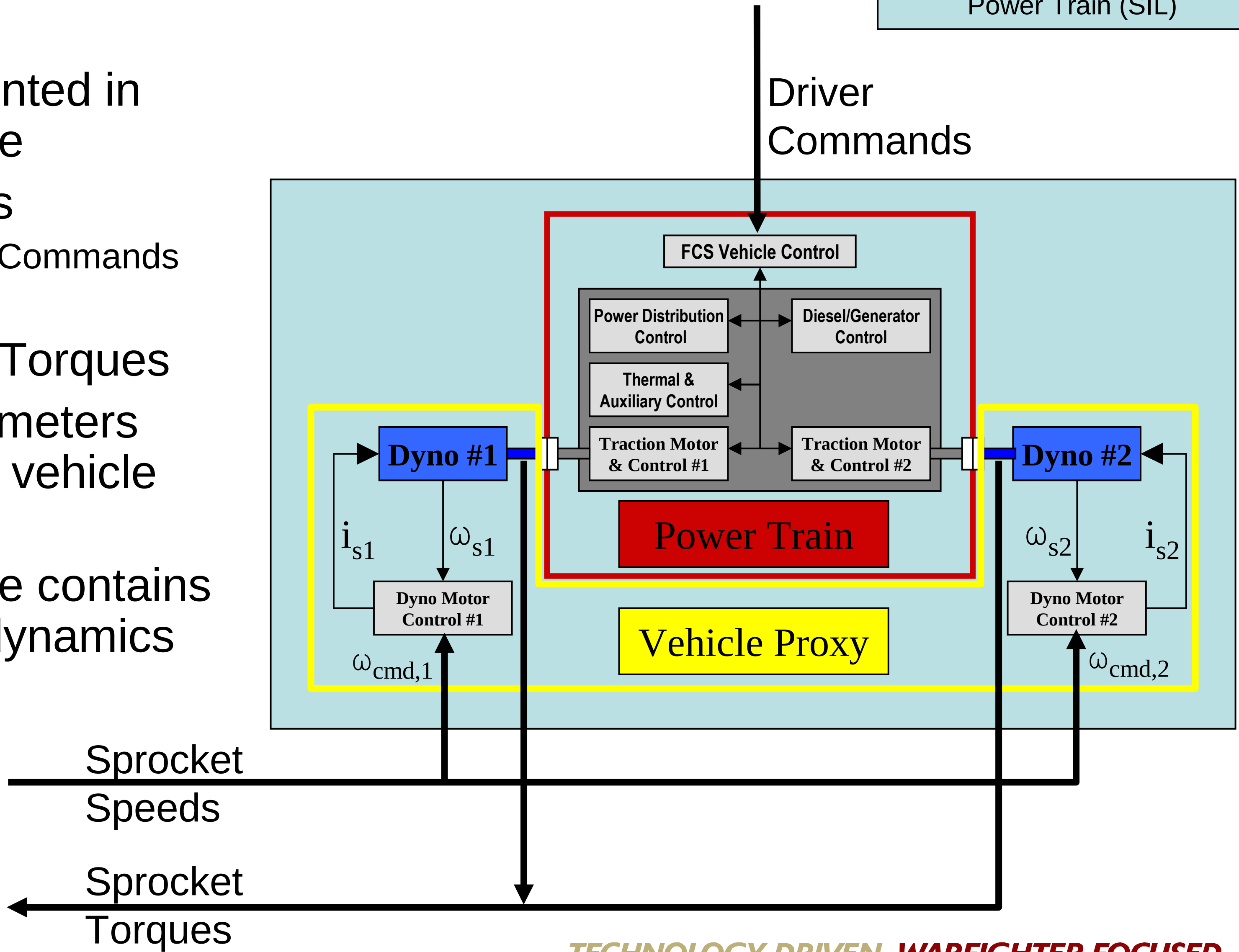
- Implemented in Dynamics Modeling Tool
- Receives Torque
- Outputs
  - Speed
  - Motion
- Integrates its own states



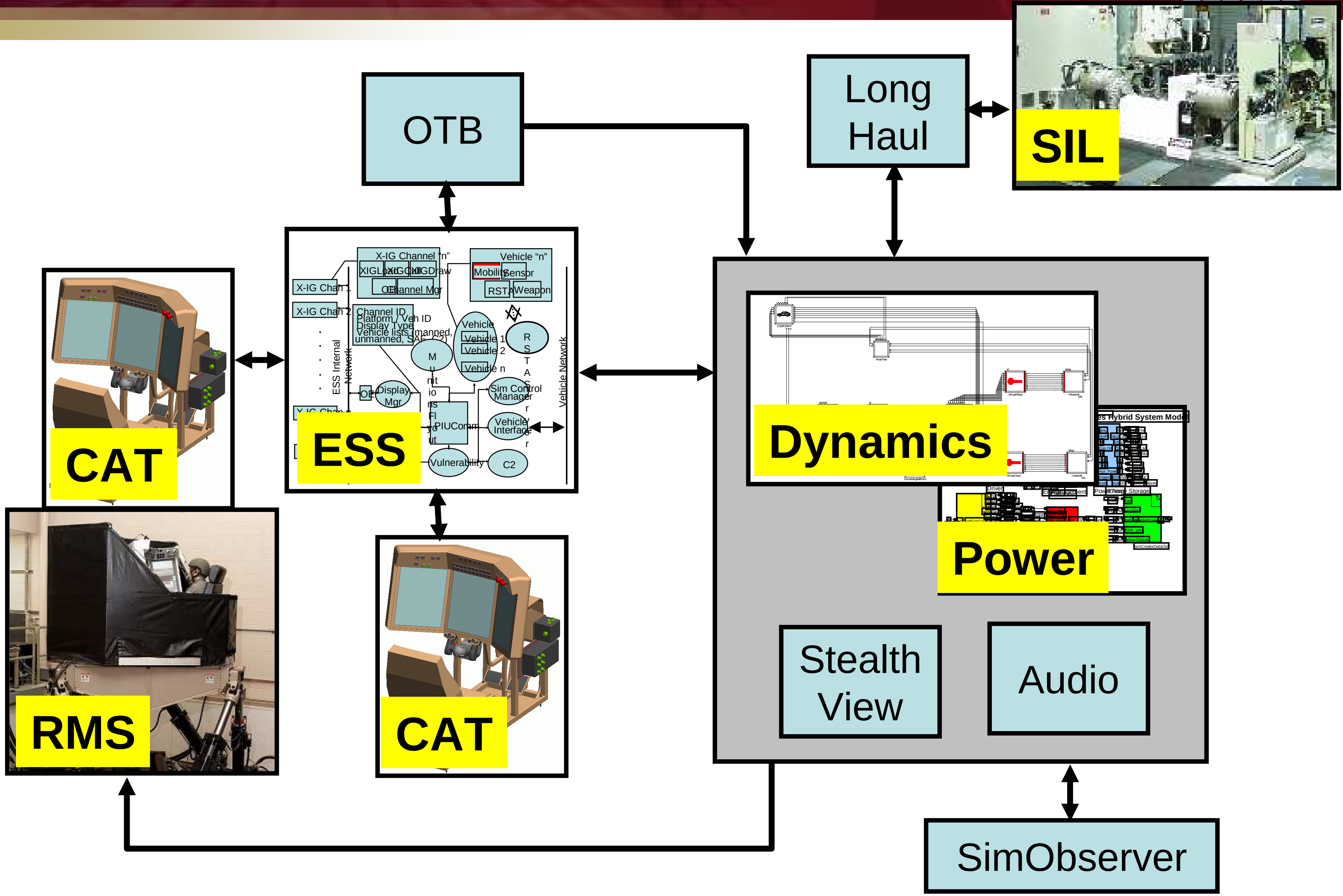
Power Train


Power Train (SIL)

- Implemented in Hardware
- Receives
  - Driver Commands
  - Speed
- Outputs Torques
- Dynamometers serve as vehicle proxy
- Hardware contains implicit dynamics



DCE Top Level Design



Communication Channel Choice

Modem (56k bps)

- Analog/Digital
- Dedicated channel
- Connection-based
- Reliable
- No firewall
- Noise-based corruption
- ~350 ms round trip
- 1.4% loss rate

Internet

- Digital
- No dedicated channel
- Packet-based
- Moderately Reliable
- Firewall configuration required
- Dropped packets
- ~94 ms round trip
- 0.1% loss rate

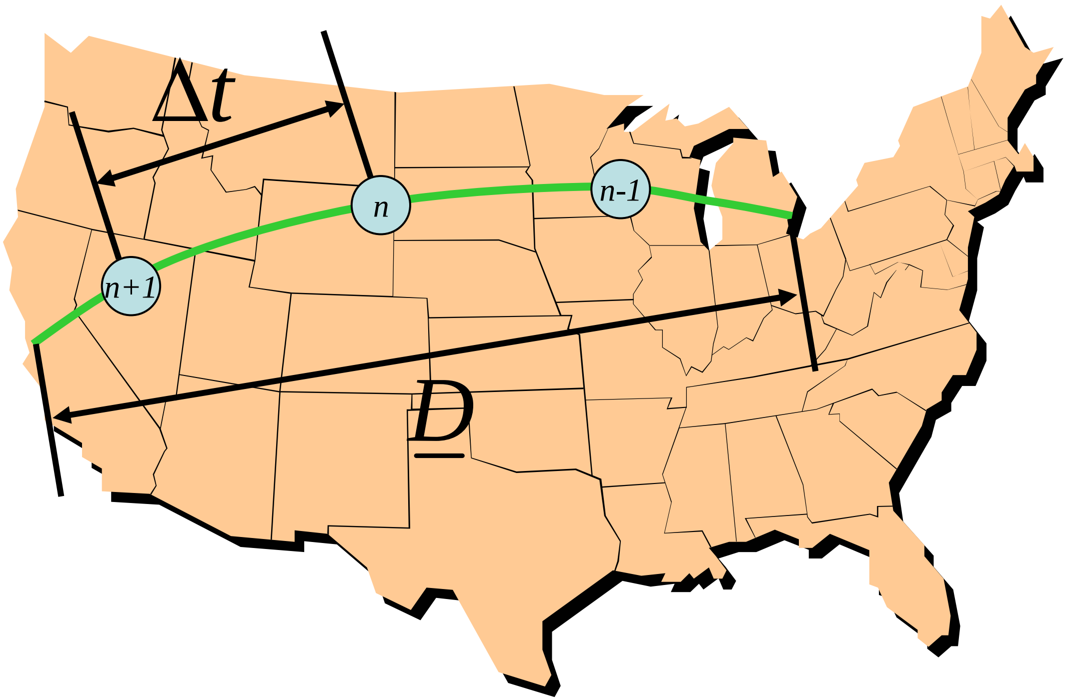
Protocol Choice

TCP $\Delta t \; ? \; \underline{D}$

- (Virtual) Connection
- Stream
- Reliable

UDP $\Delta t = \underline{D}$

- Connectionless
- Packet
- Unreliable



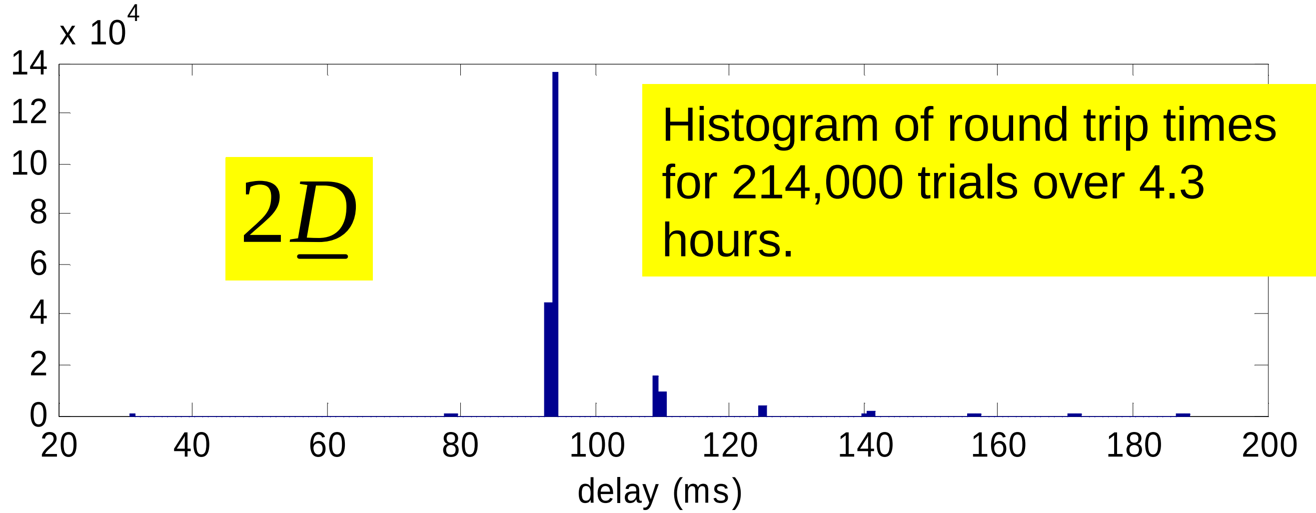
UDP Performance

- Round trip times
  - 78 ms to 188 ms
  - Most at 94 ms
  - Limit 26 ms
- 209 packets dropped
- Vehicle dynamics ~2 ms
- SIL ~10 ms

- Problems
  - Substantial delay
  - Delay jitter
  - Data loss

⇒ System Instability

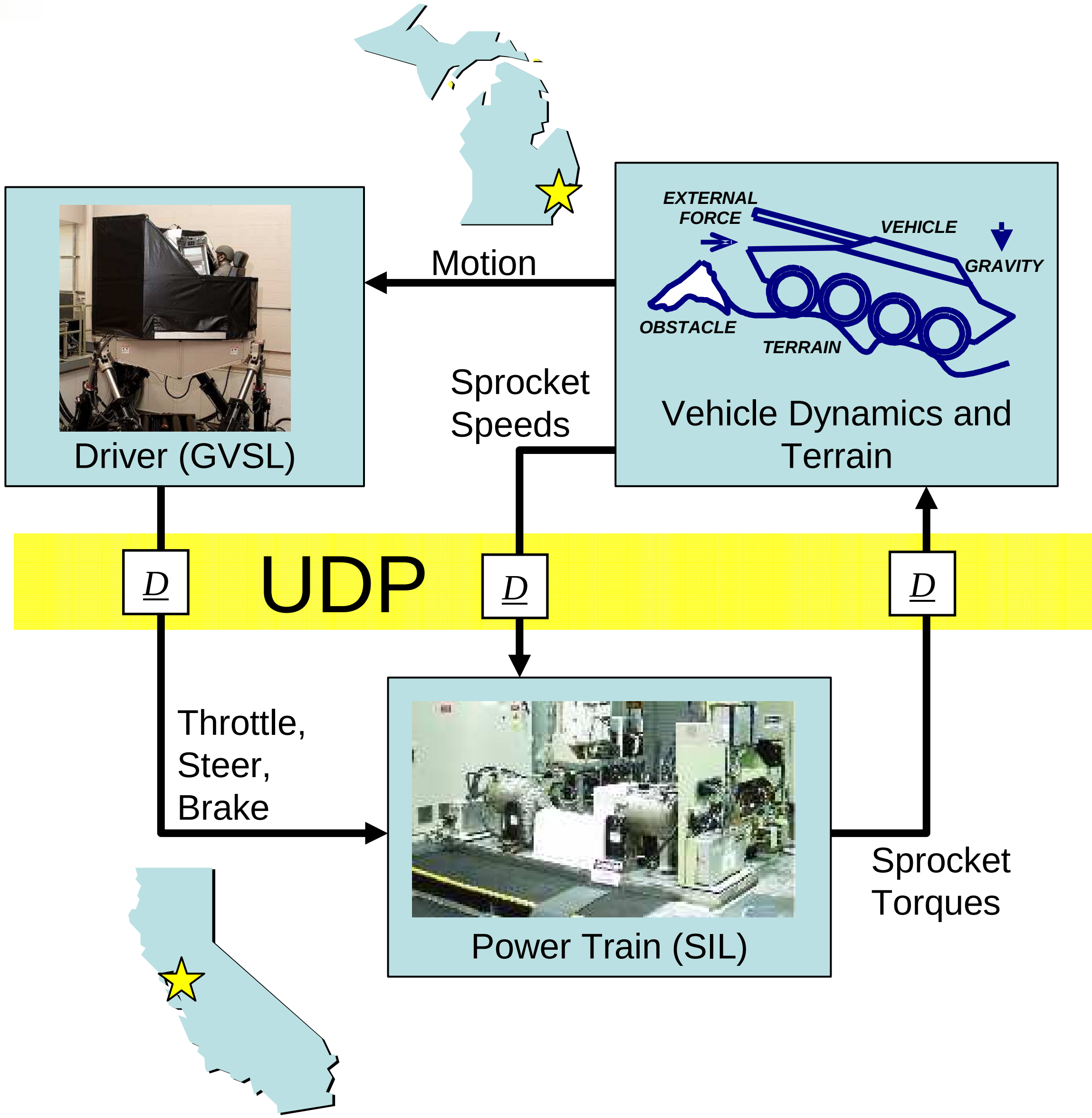
$\underline{D}$ is a random variable



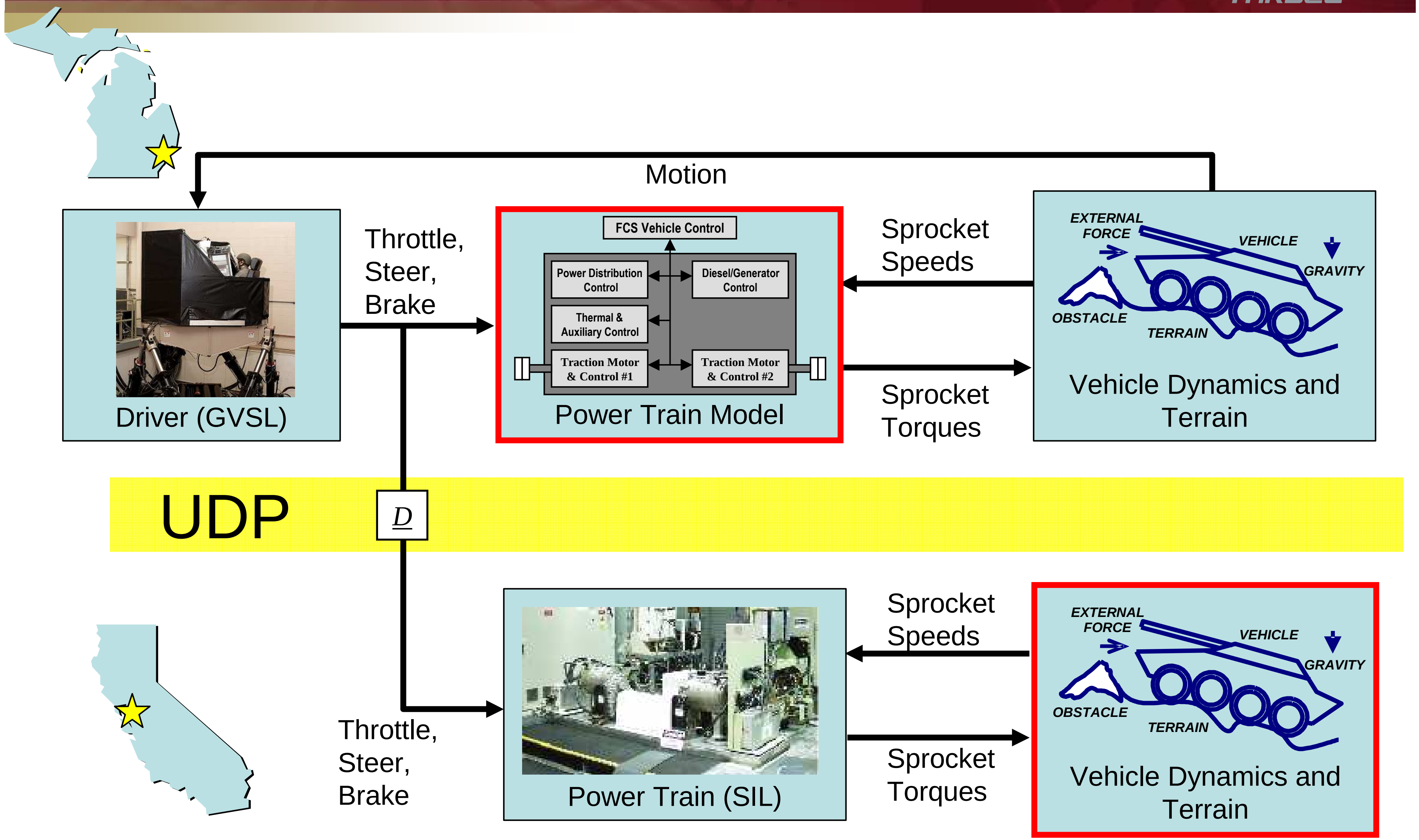
Histogram of round trip times for 214,000 trials over 4.3 hours.

Design A – Naïve Approach

- Delay > Dynamics
- Delay > SIL
- Simulator response
  - Driver → Motion
  - Increased by *2D*
- Safety risk to driver
- Damage risk to SIL
- Experimental quality degraded
- Potential instabilities



Design B – Parallel Simulations



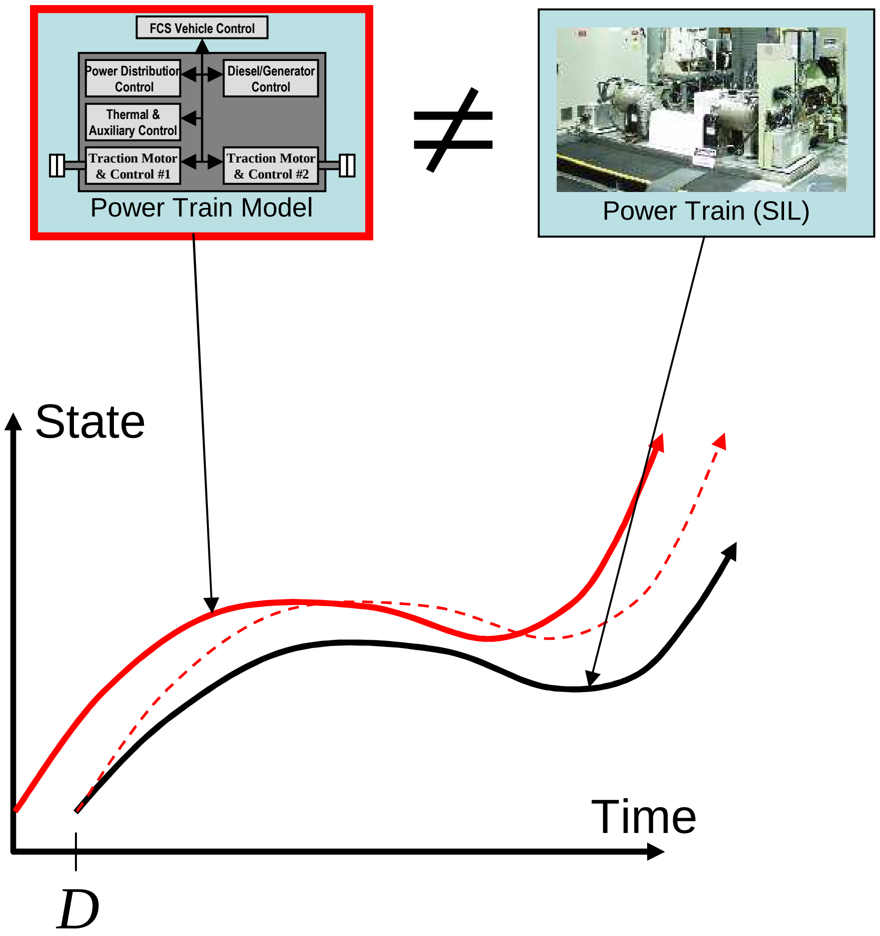
Design B – Evaluation

Pros

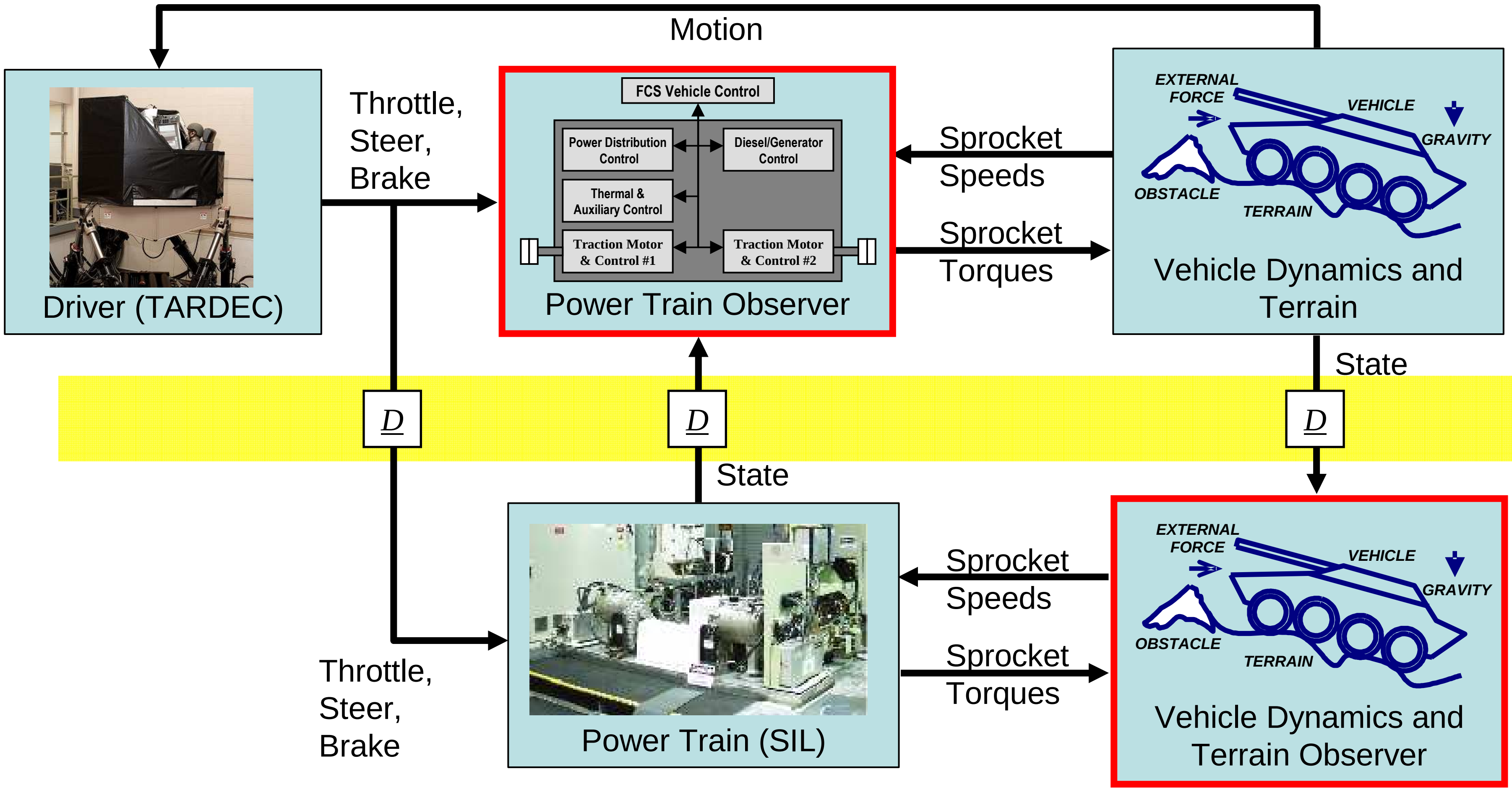
- SIL will receive proper commands delayed by D
- Immediate response
- The GVSL and SIL are not coupled

Cons

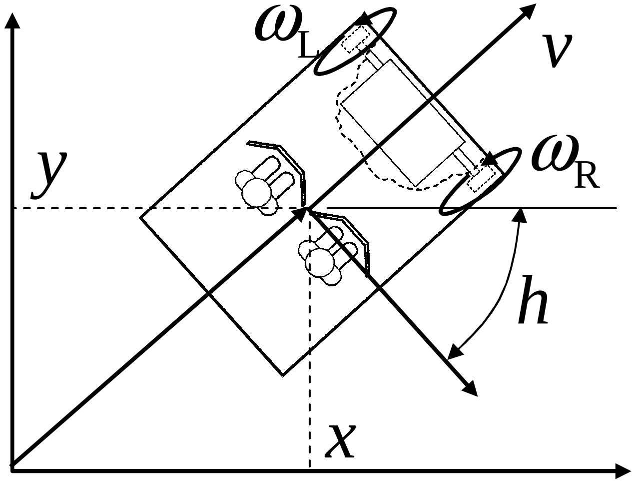
- The power train model does not exactly match the SIL
- The GVSL and the SIL will tend to drift apart over time.



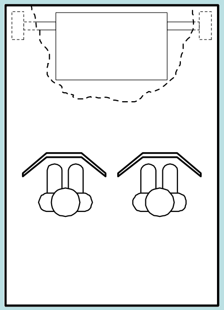
Design C – Observers



Design C – Vehicle dynamics



$$\boldsymbol{\tau}^v = \begin{bmatrix} \tau_L \\ \tau_R \end{bmatrix}$$

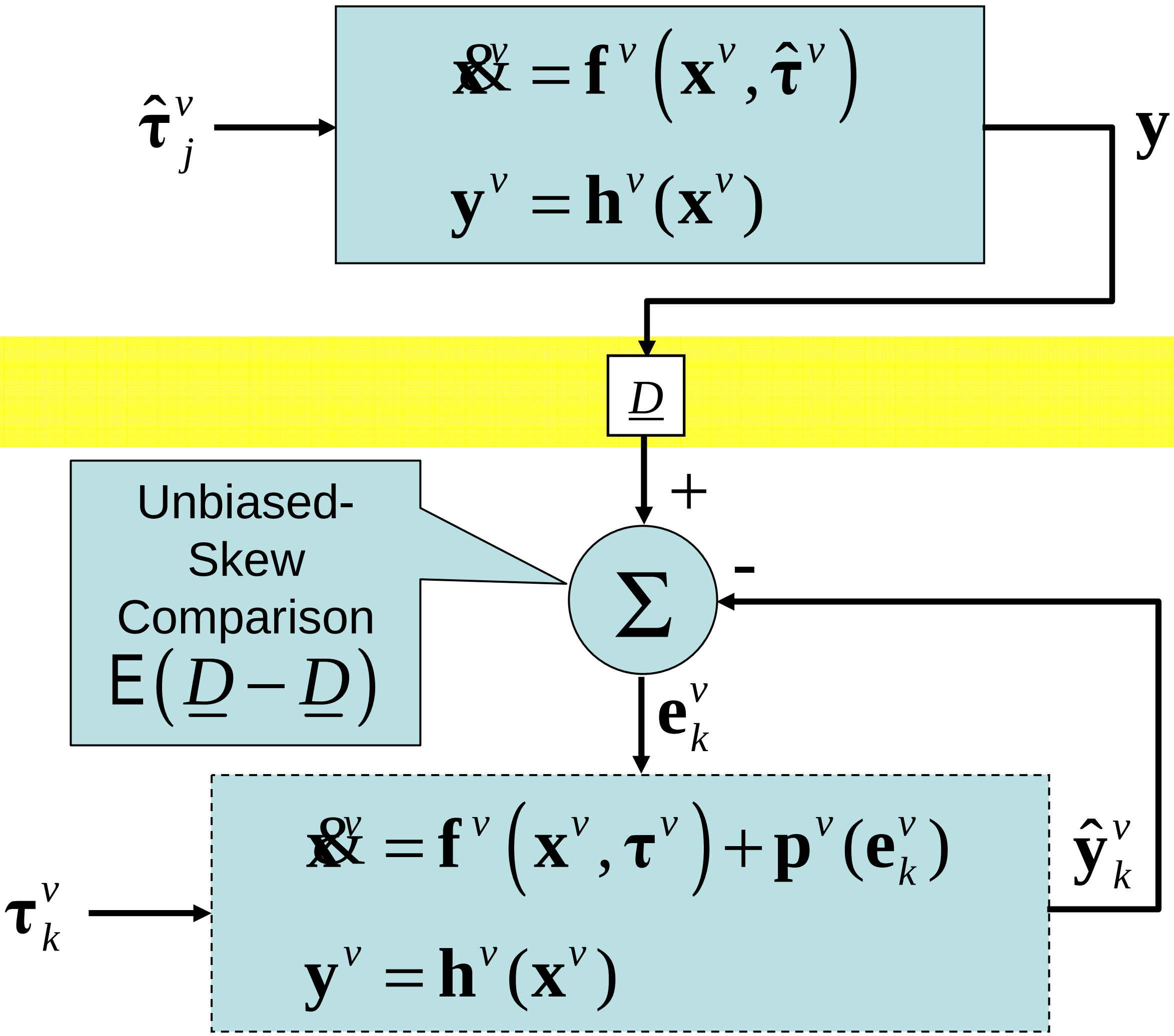


$$\dot{\mathbf{x}}^v = \mathbf{f}^v\left(\mathbf{x}^v, \boldsymbol{\tau}^v\right)$$

$$\mathbf{y}^v = \mathbf{h}^v(\mathbf{x}^v)$$

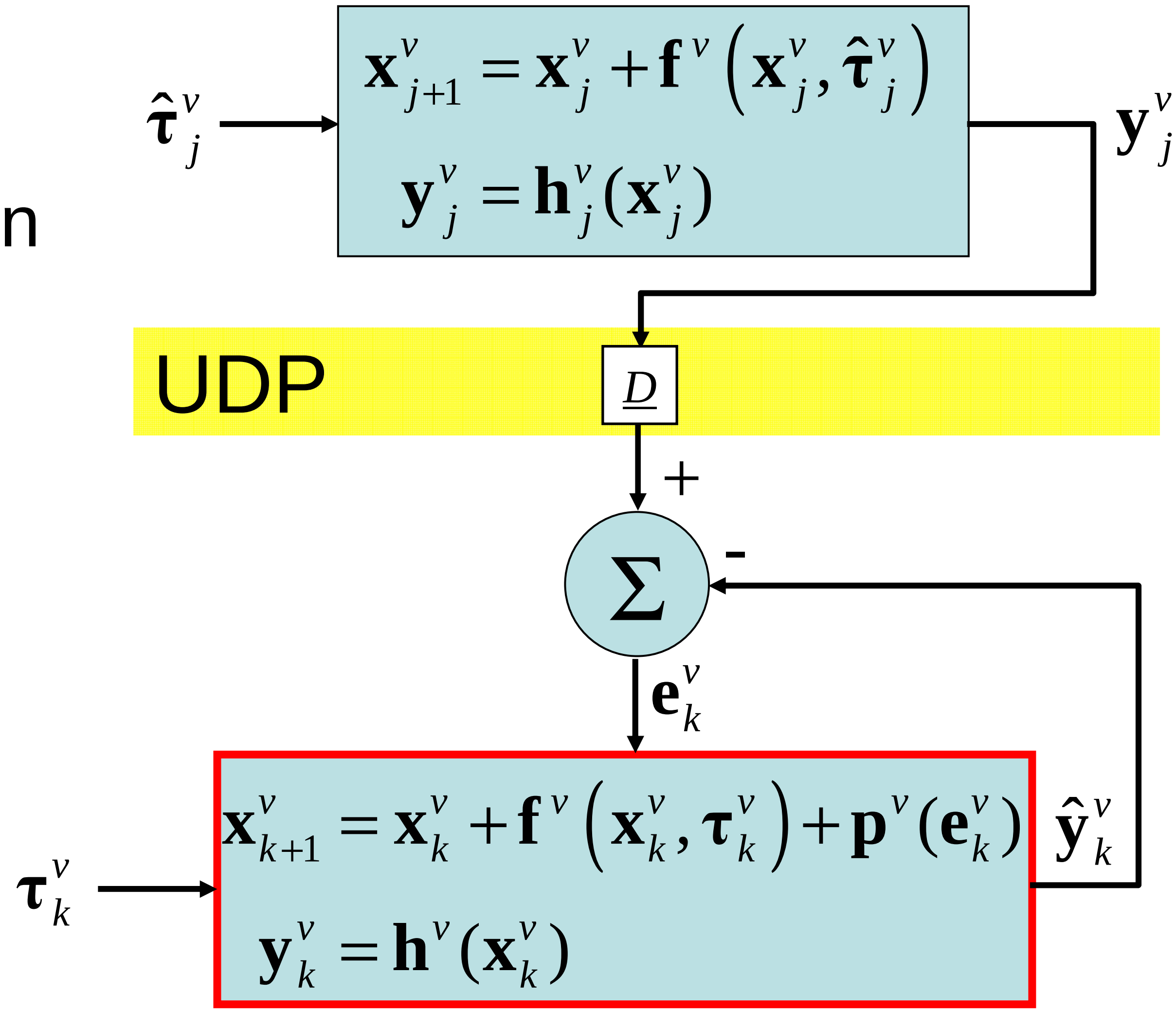
$$\mathbf{y}^v = \begin{bmatrix} x \\ y \\ h \\ v \\ \omega_L \\ \omega_R \end{bmatrix}$$

Design C – Vehicle Observer

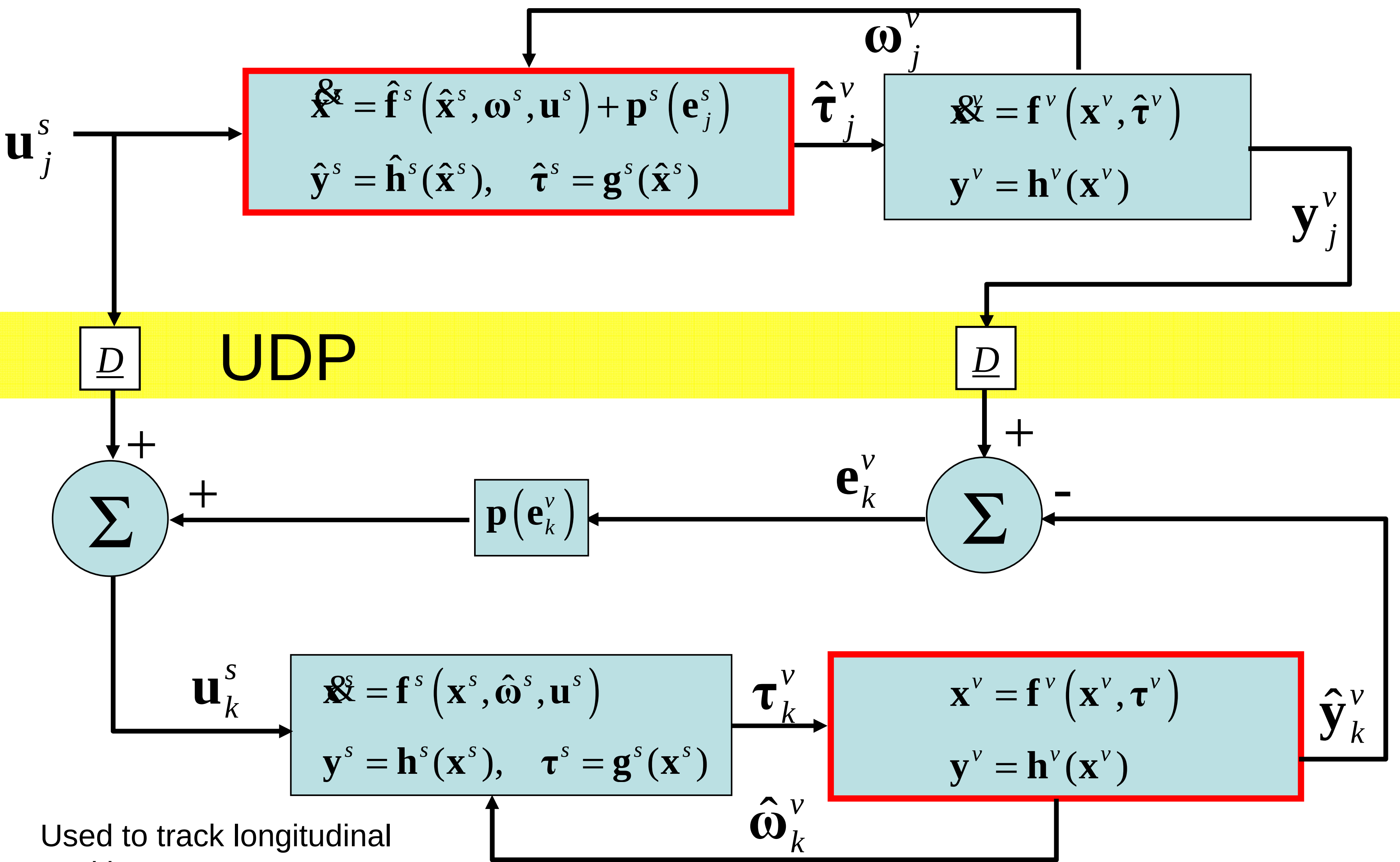


Vehicle Observer – Direct (“Skyhook”)

- Imposes an artificial force on the vehicle
- Used to track
  - Lateral position
  - Heading
  - Sprocket speed

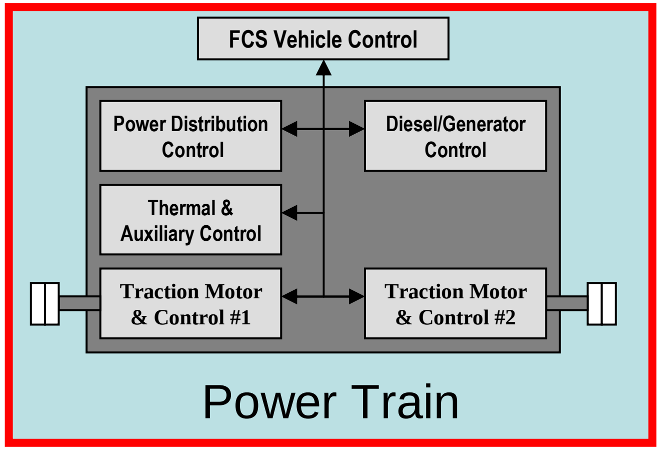


Vehicle Observer – Indirect ("Augmented Input")



- Used to track longitudinal position

Design C – Power train dynamics



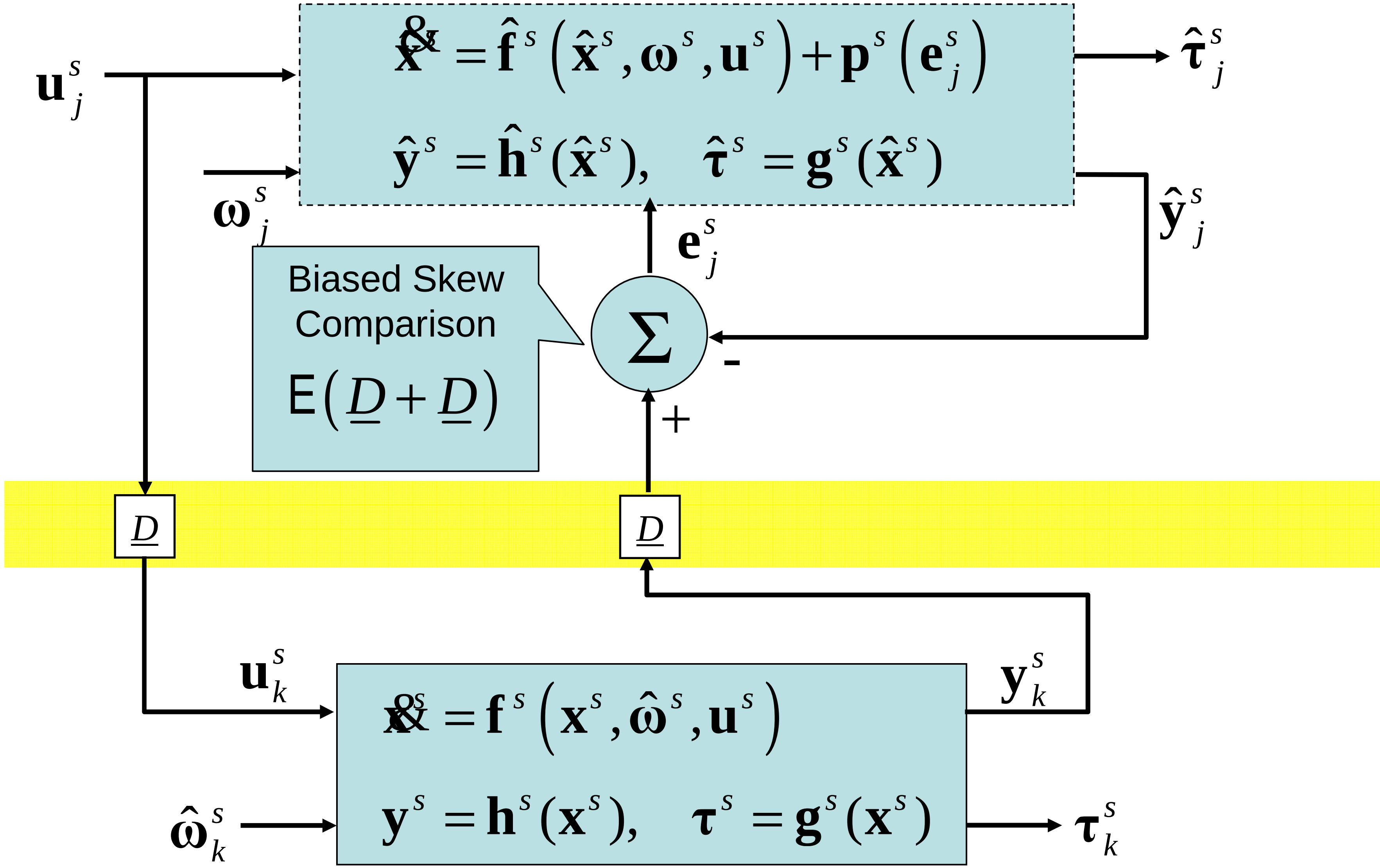
$$\mathbf{u}_k^s = \begin{bmatrix} t \\ s \\ b \end{bmatrix}, \qquad \boldsymbol{\omega}_k^s = \begin{bmatrix} \omega_L \\ \omega_R \end{bmatrix}$$

$$\dot{\mathbf{x}}^s = \mathbf{f}^s\left(\mathbf{x}^s, \boldsymbol{\omega}^s, \mathbf{u}^s\right)$$

$$\mathbf{y}^s = \mathbf{h}^s(\mathbf{x}^s), \quad \boldsymbol{\tau}^s = \mathbf{g}^s(\mathbf{x}^s)$$

$$\mathbf{y}_k^s = \begin{bmatrix} SOC_b \\ V_{bus} \\ \omega_e \\ l_f \\ T_b \\ M \end{bmatrix}, \qquad \boldsymbol{\tau}_k^s = \begin{bmatrix} \tau_L \\ \tau_R \end{bmatrix}$$

Design C – Power Train Observer



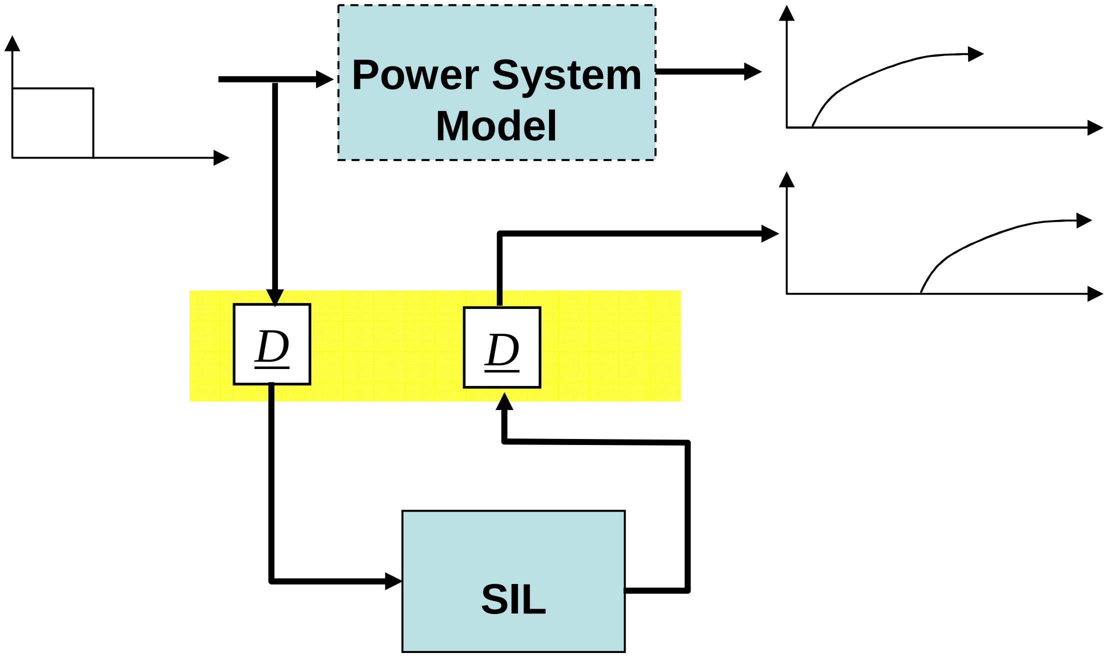
Design C – Evaluation

Pros

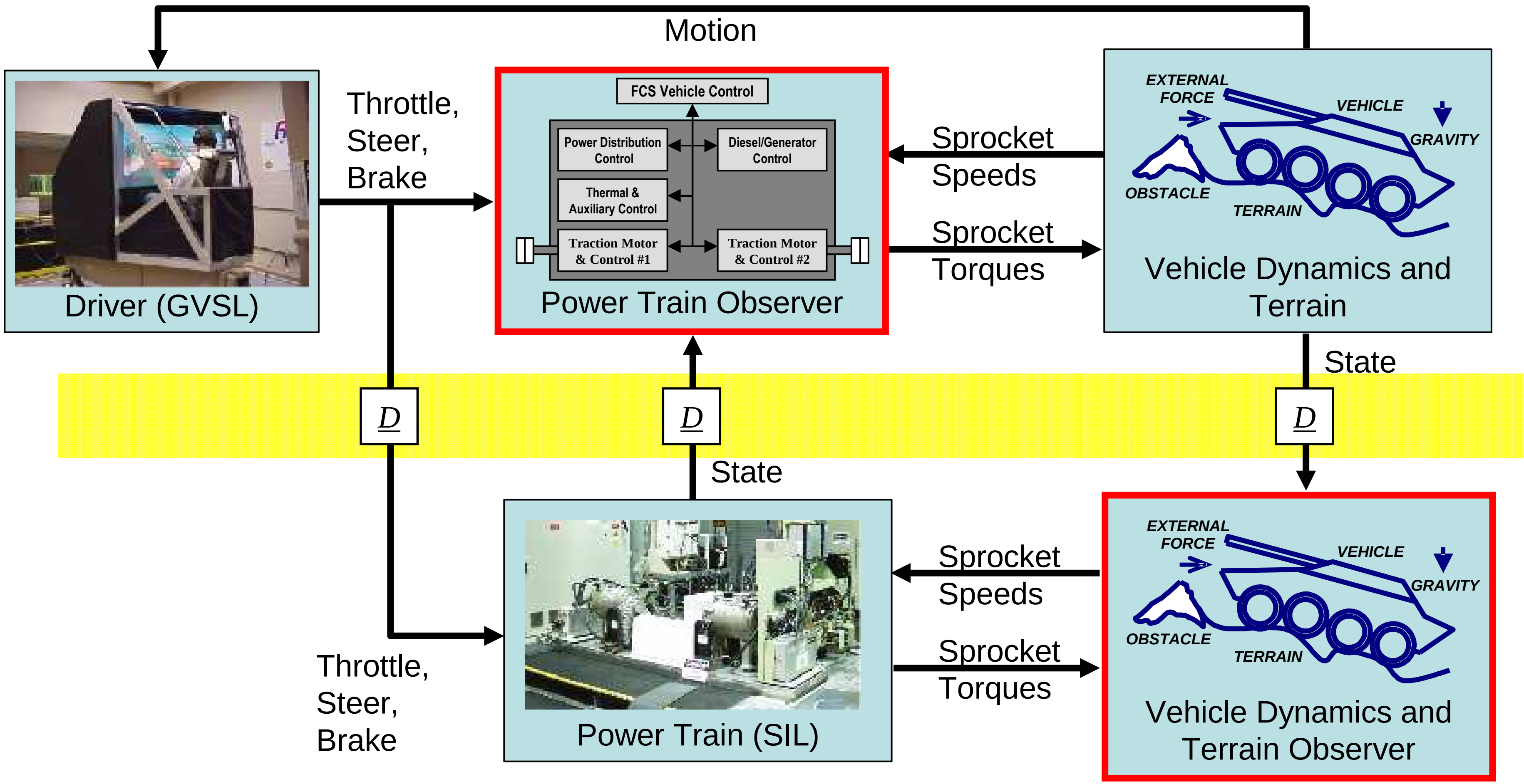
- States should track
- Delay is approximately negated in vehicle error

Cons

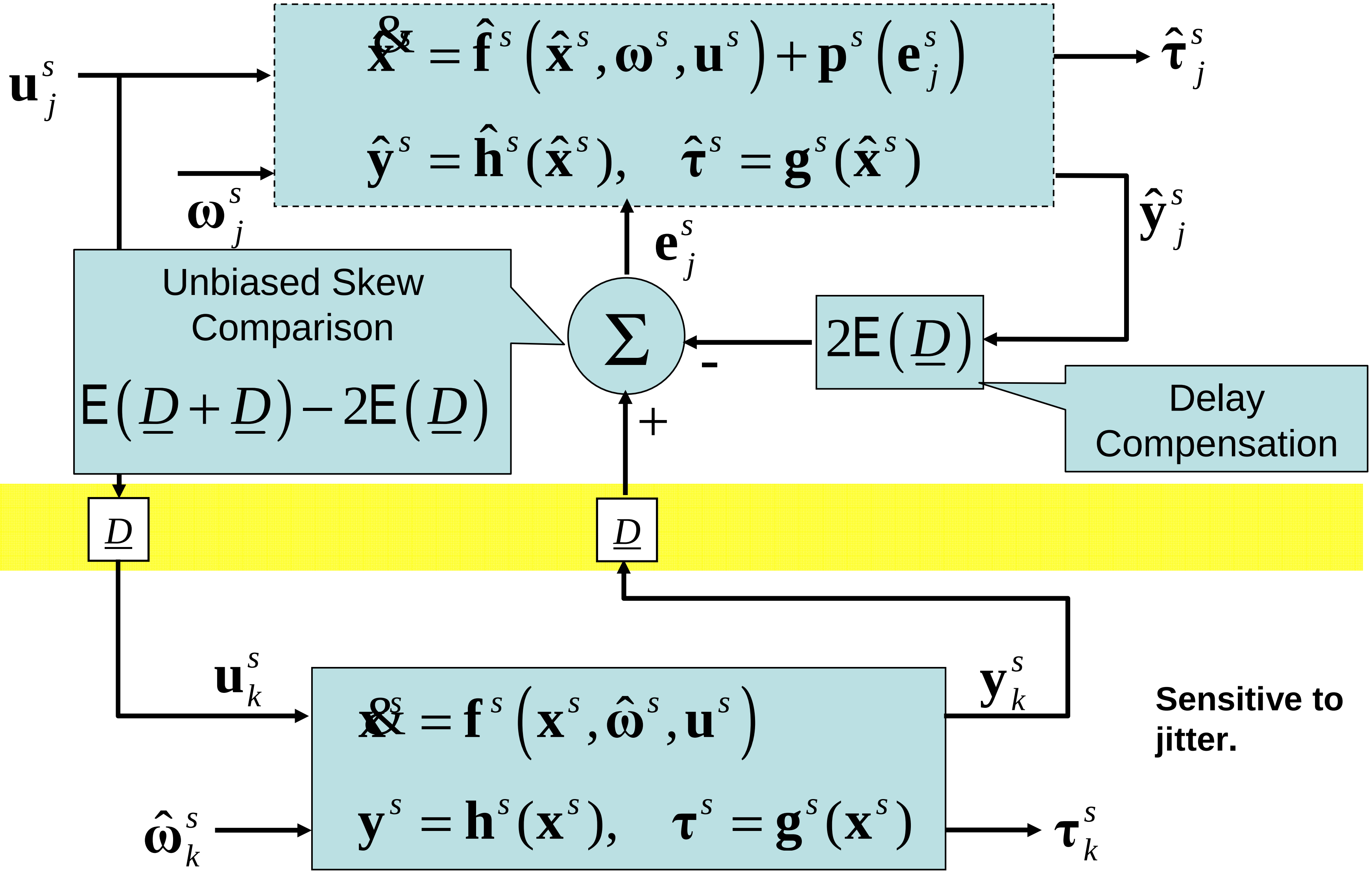
- Delay is approximately doubled in power train error
- Error contains time skew



Design D – Delay Compensation



Design D – Power Train Observer Delay Compensation



Sensitive to jitter.

Design D – Evaluation

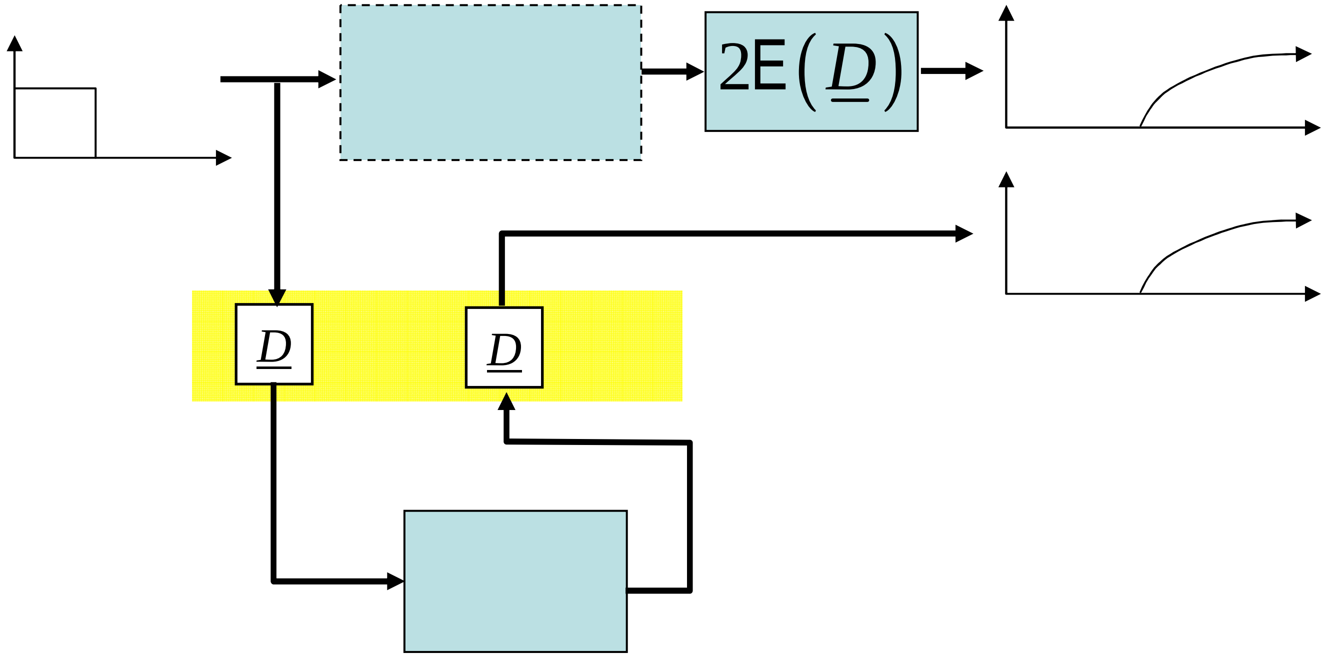
Pros

– Unbiased estimate of the error

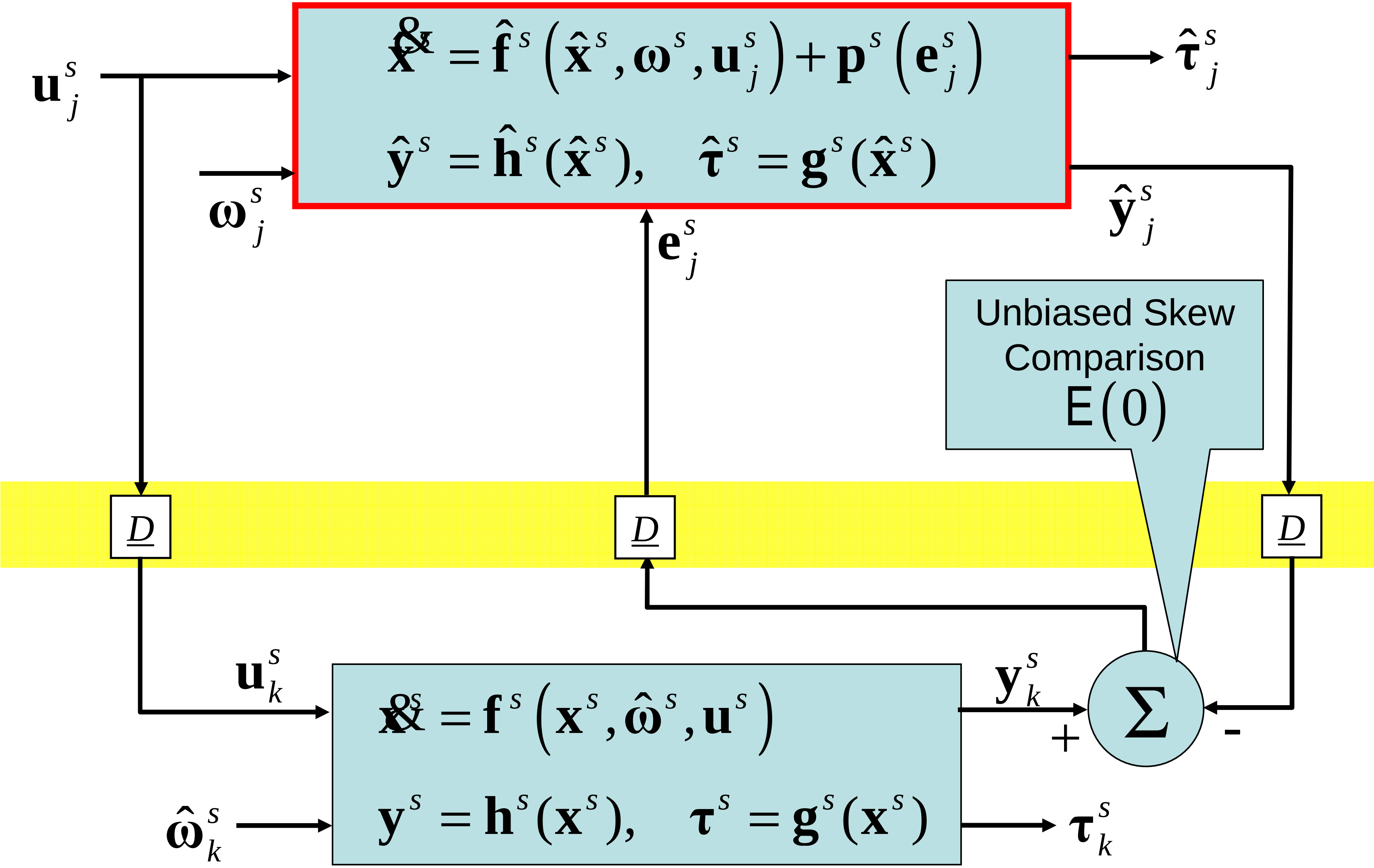
Cons

– Does not account for variance in delay (jitter).

– Does not account for data loss.



Power Train Observer (un-skewed, robust)



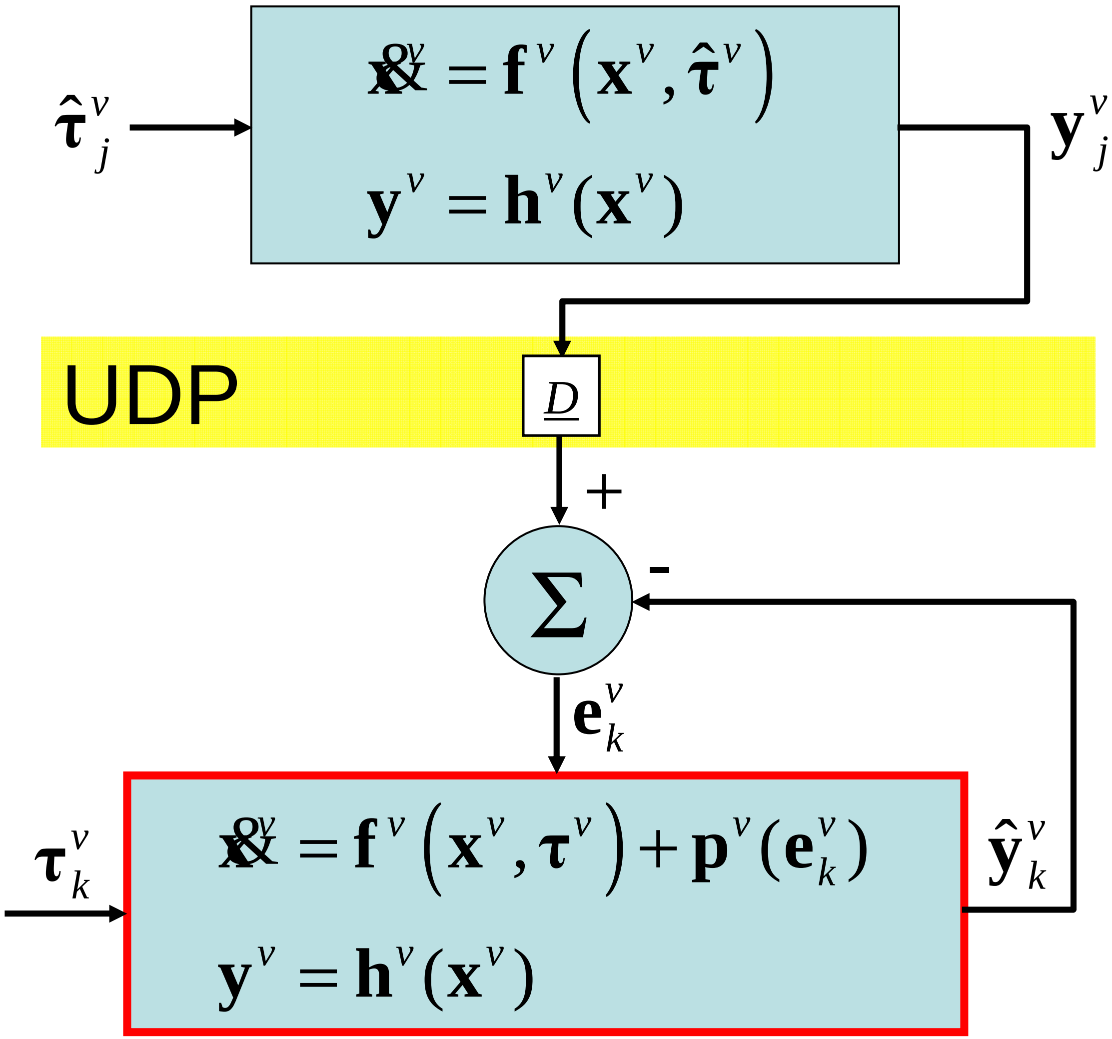
The Correction Term

- Ideas from sliding mode control were used to define the correction term.
- We used the sliding surface

$$s = \left(\frac{d}{dt} + \lambda \right) e$$

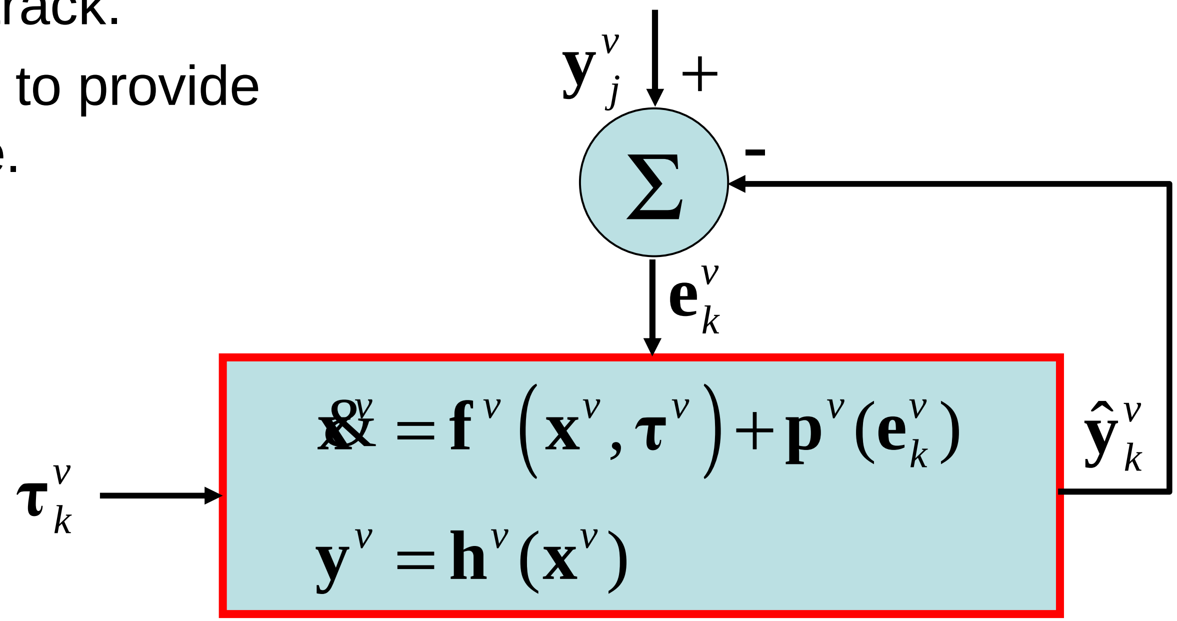
for some degree of freedom .

- We then devise a controller to drive s to 0.
- Our control action targets the acceleration terms of $\dot{\mathbf{x}}$
- We use the “robust” term of the SMC with a transition region to avoid chattering.



Aggressiveness of Control

- $\mathbf{p}^{v}(\mathbf{e}_k^{v})$ directly affects state rates.
- States are fully accessible (via rates).
- If allowed, may directly manipulate states.
- It is best to manipulate states in a rational way (i.e., IAW non-holonomic constraints)
- It is best to allow the correction term to gently keep system on track.
- Allow forward dynamics to provide instantaneous response.

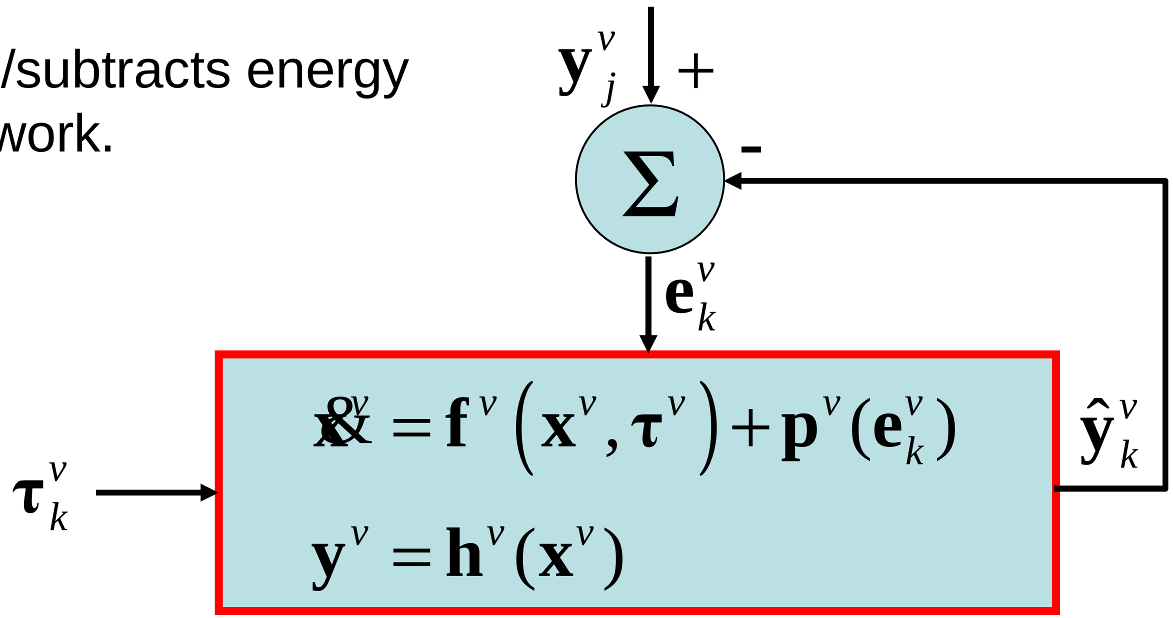


Leaked Energy

- $\mathbf{p}^v(\mathbf{e}_k^v)$ imposes an artificial "force" on the vehicle
- It affects the location and velocity.
- Ideally $\mathbf{f}^v\left(\mathbf{x}^v,\boldsymbol{\tau}^v\right) = \mathbf{p}^v(\mathbf{e}_k^v)$
- If $\mathbf{p}^v$ affects only acceleration terms then $\mathbf{v}^T\mathbf{p}^v$ is an energy like term (normalized to unit mass).
- The correction term adds/subtracts energy from the system; it does work.
- We use

$$\int_0^t \mathbf{v}^T\mathbf{p}^v d\tau$$

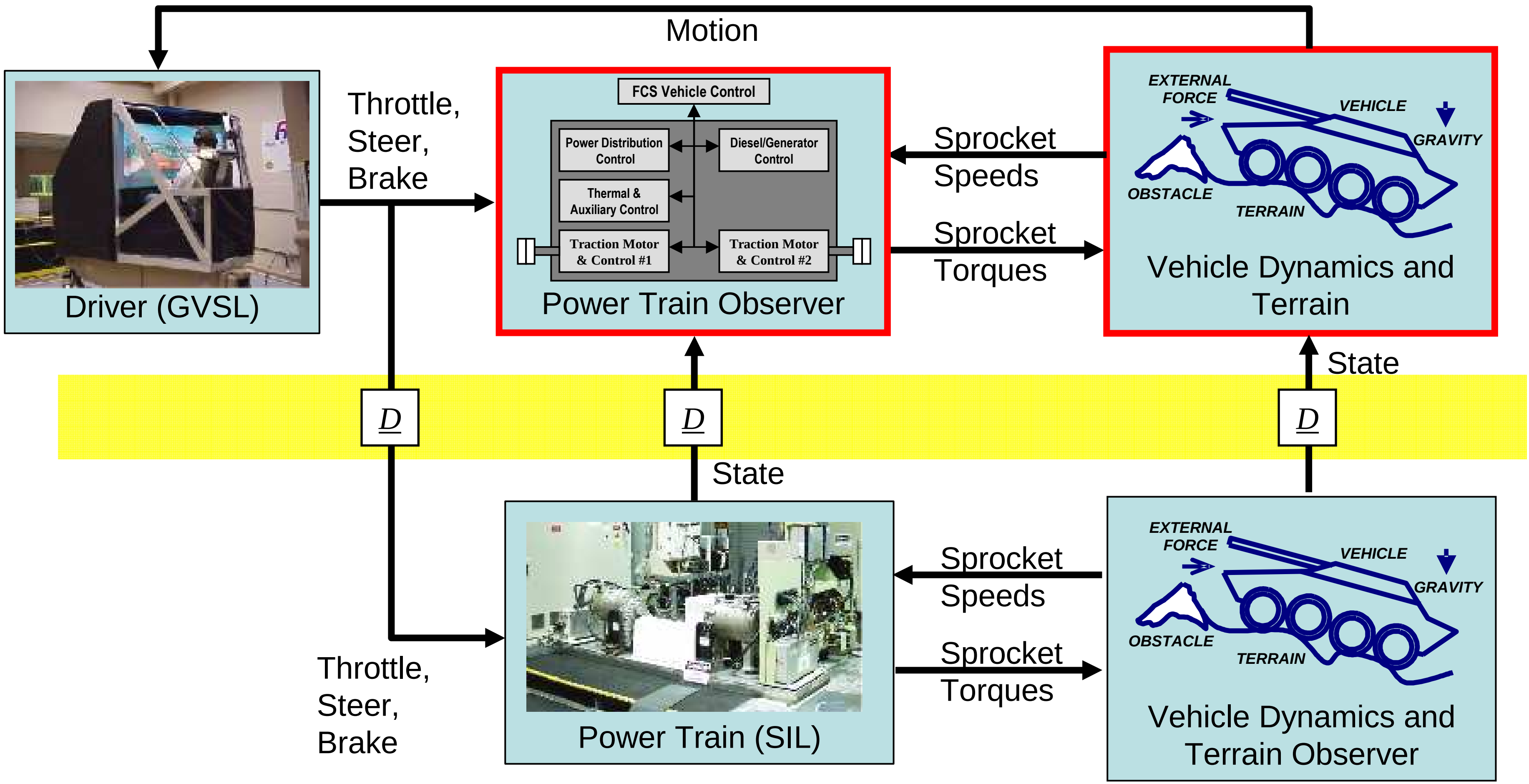
to measure the accuracy of the observer.



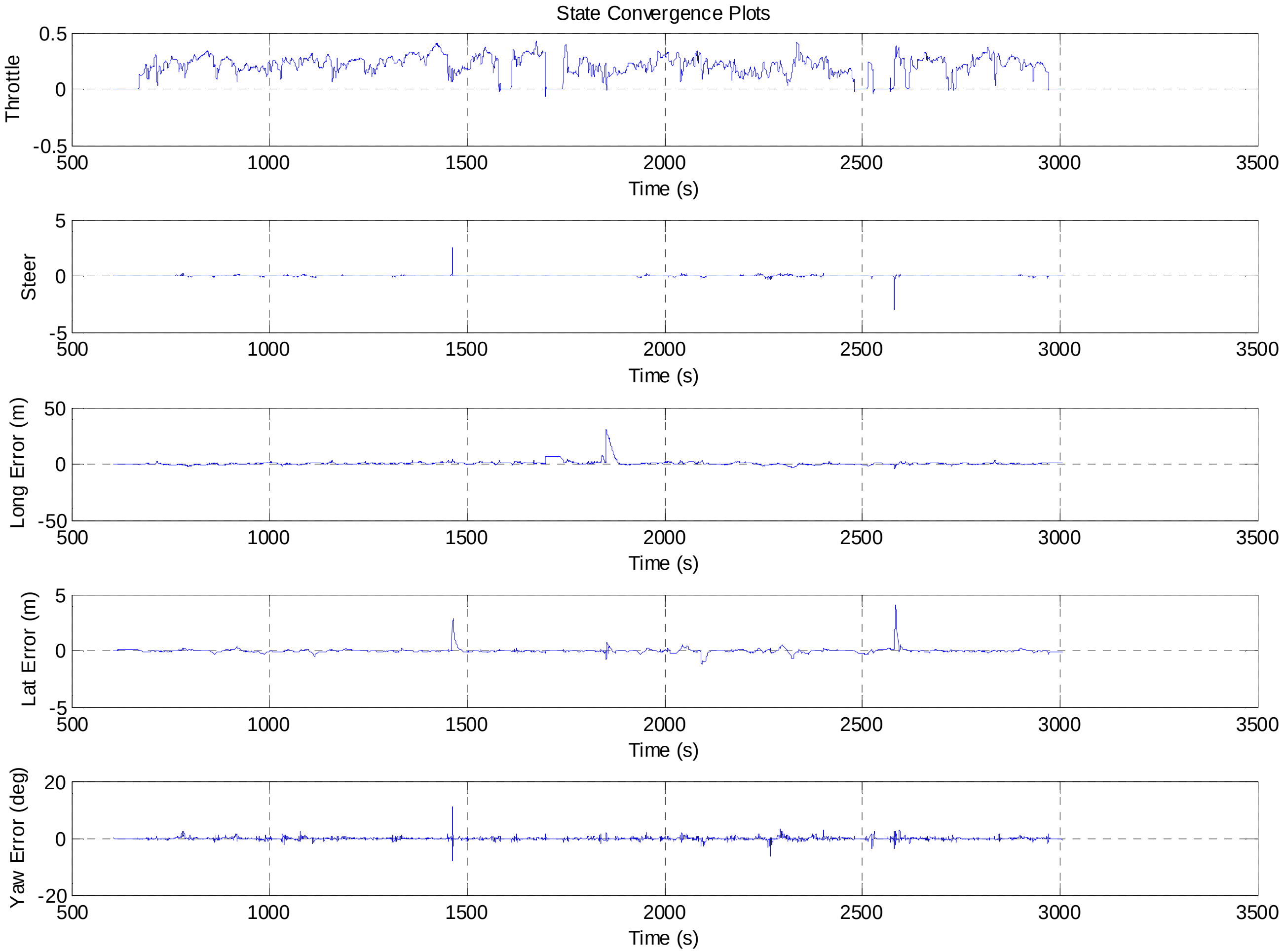
Safety Robustness

- Both TARDEC and SIL have parallel simulations running.
- Only TARDEC has human operator.
- Both sides set thresholds on state convergence error.
- If threshold is exceeded, the SIL is dropped off line and TARDEC continues.
- Additionally, health flags are sent back and forth regarding major system readiness.

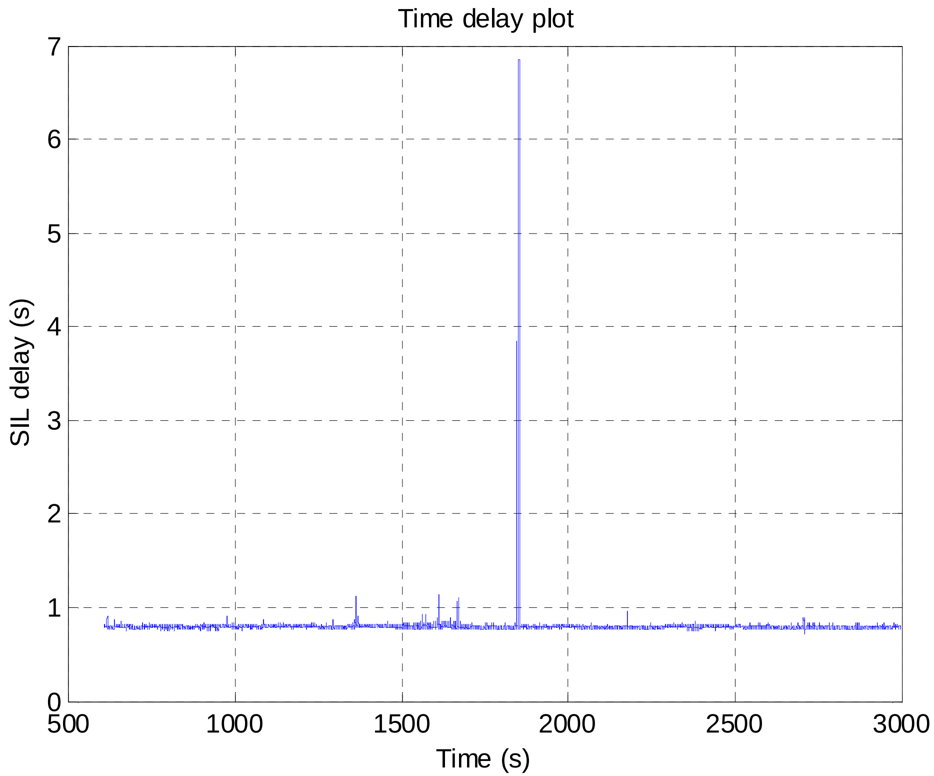
Design
E – Local Observers



Driver commands and SC errors

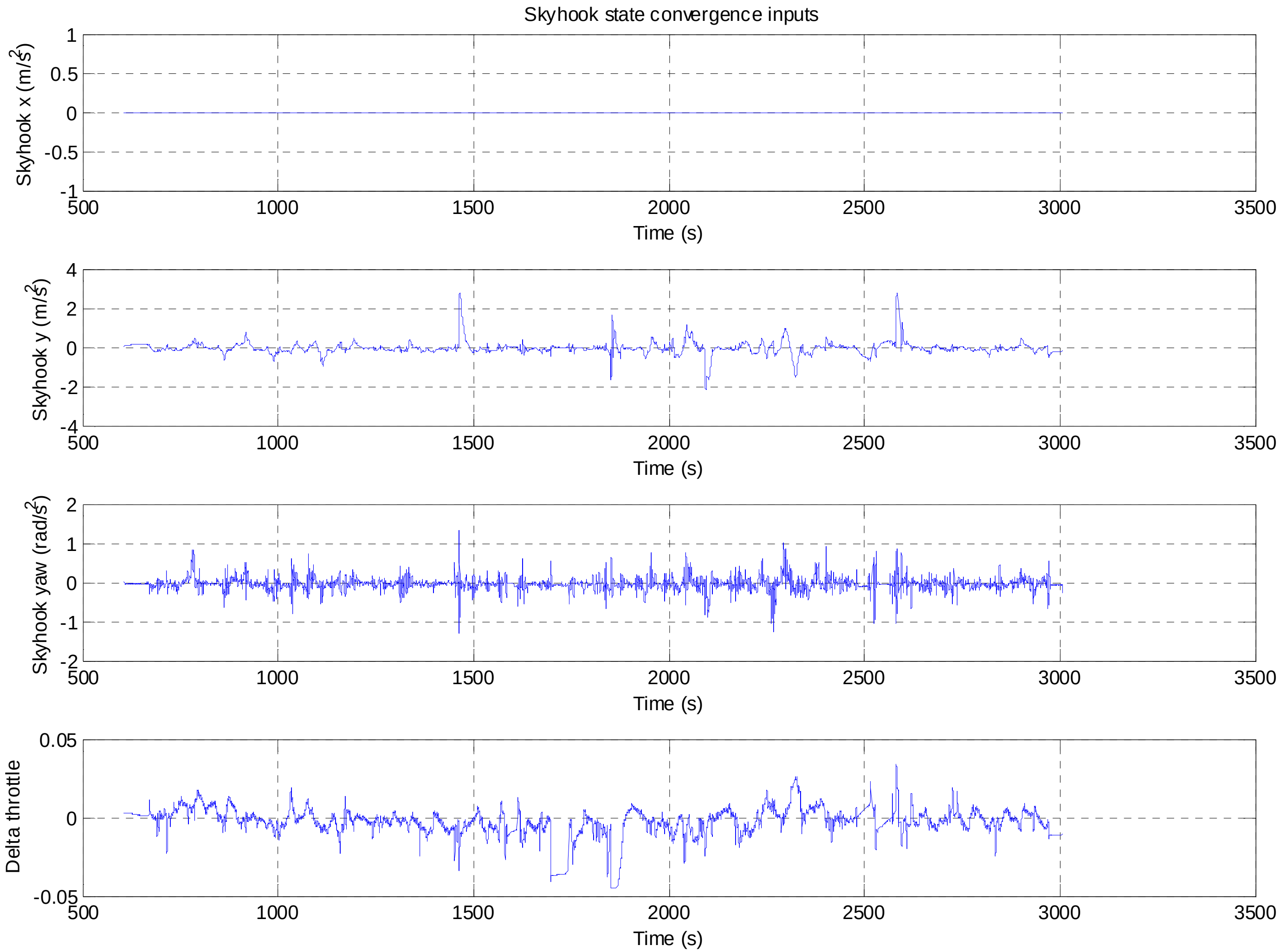


Time Delay

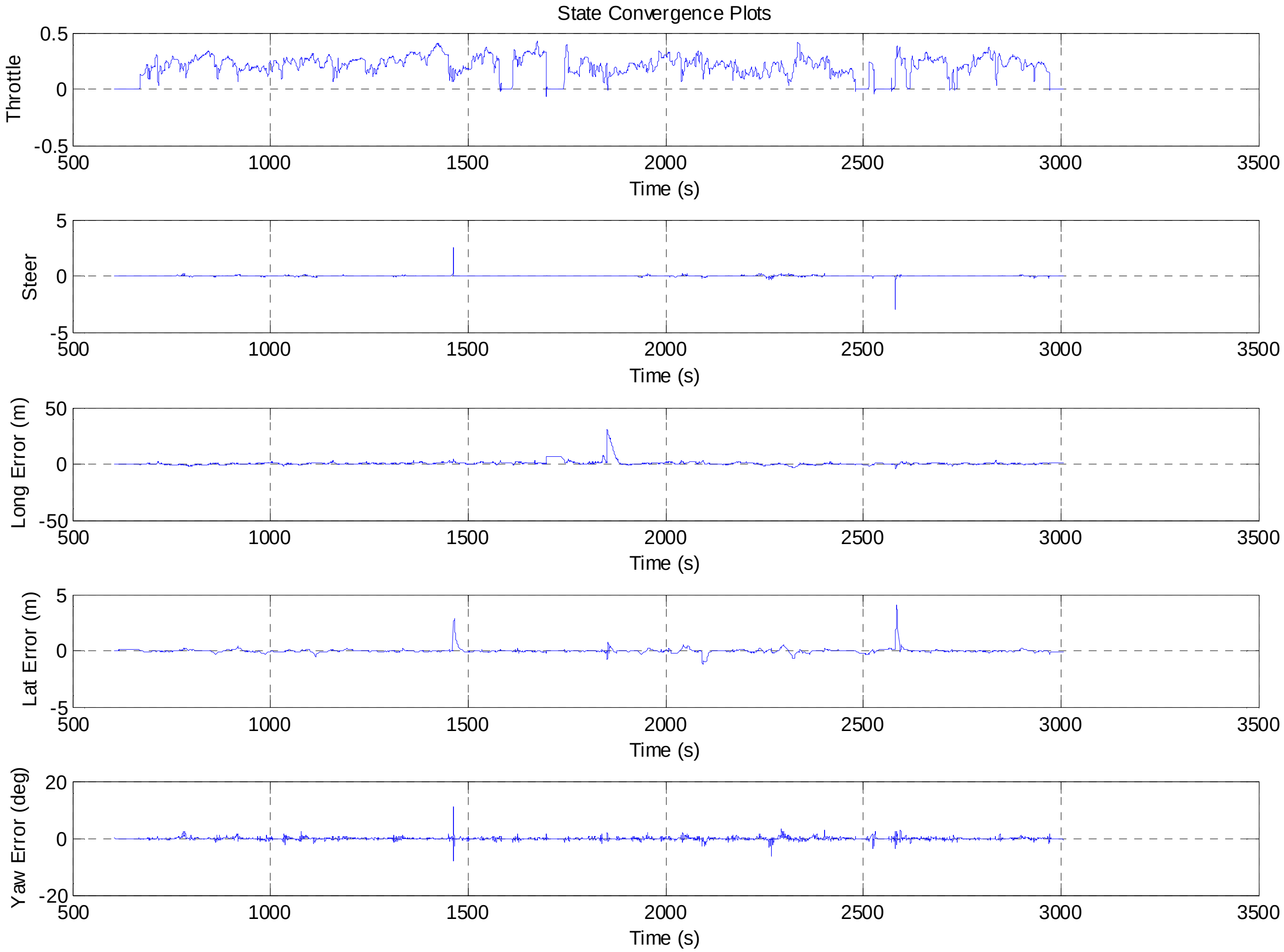


Time delay plot
SIL delay (s)
Time (s)
0
1
2
3
4
5
6
7
500
1000
1500
2000
2500
3000

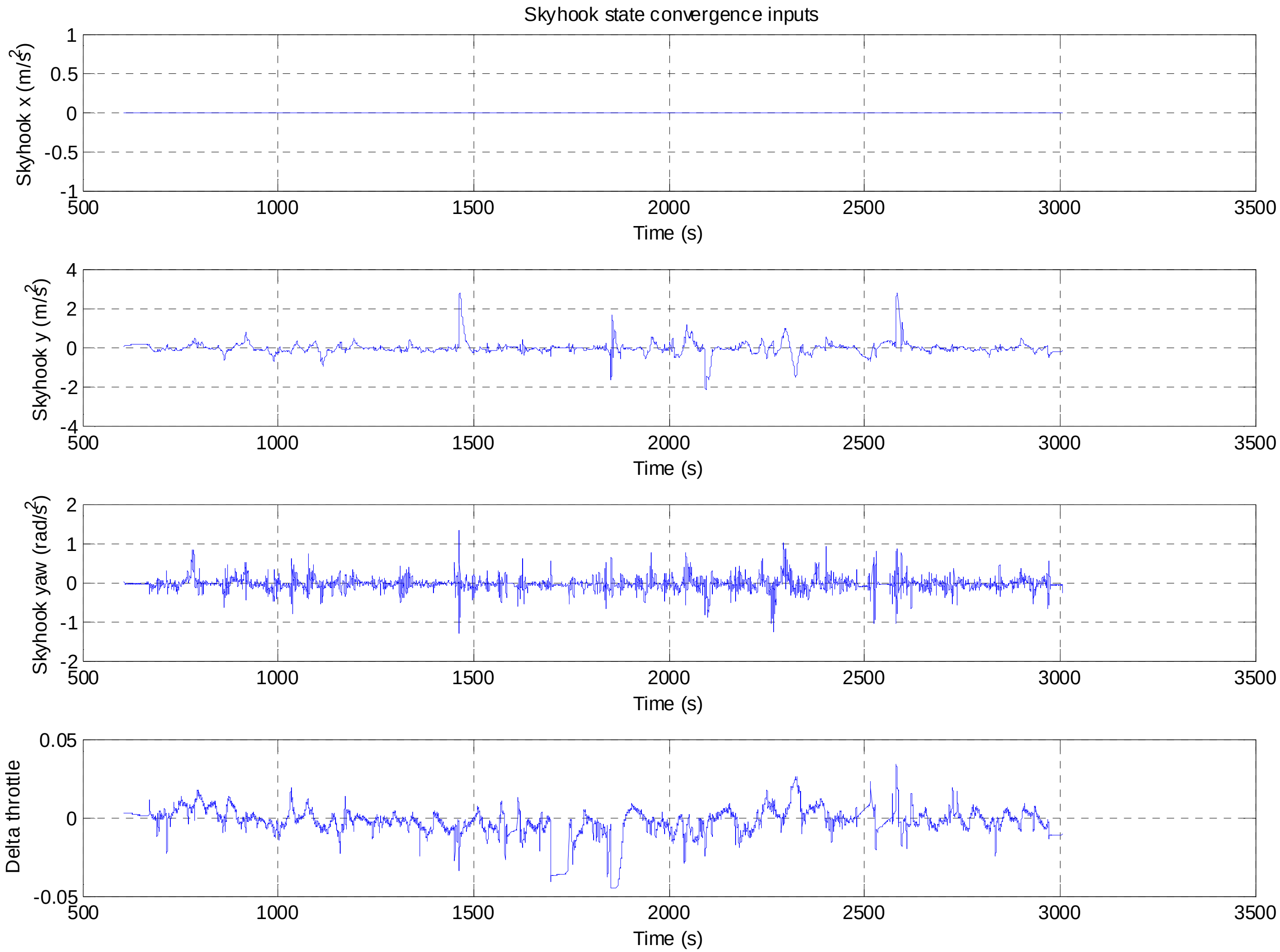
Artificial inputs



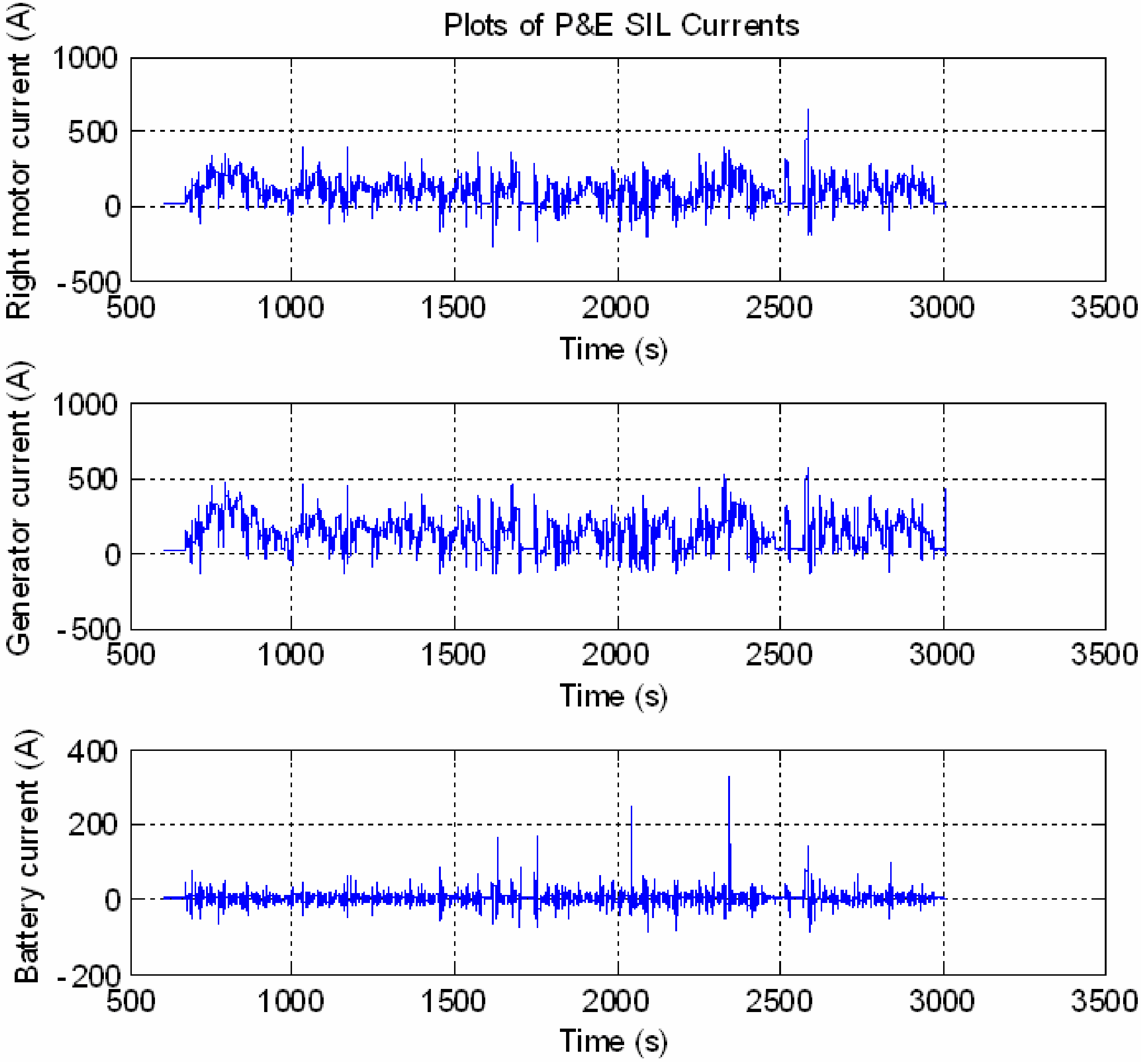
Driver commands and SC errors



Artificial inputs



SIL Performance



Main Results

The only large state convergence errors occurred during either a significant time delay or an extreme driving event.

Leaked mobility energy: Less than 3% of total input energy

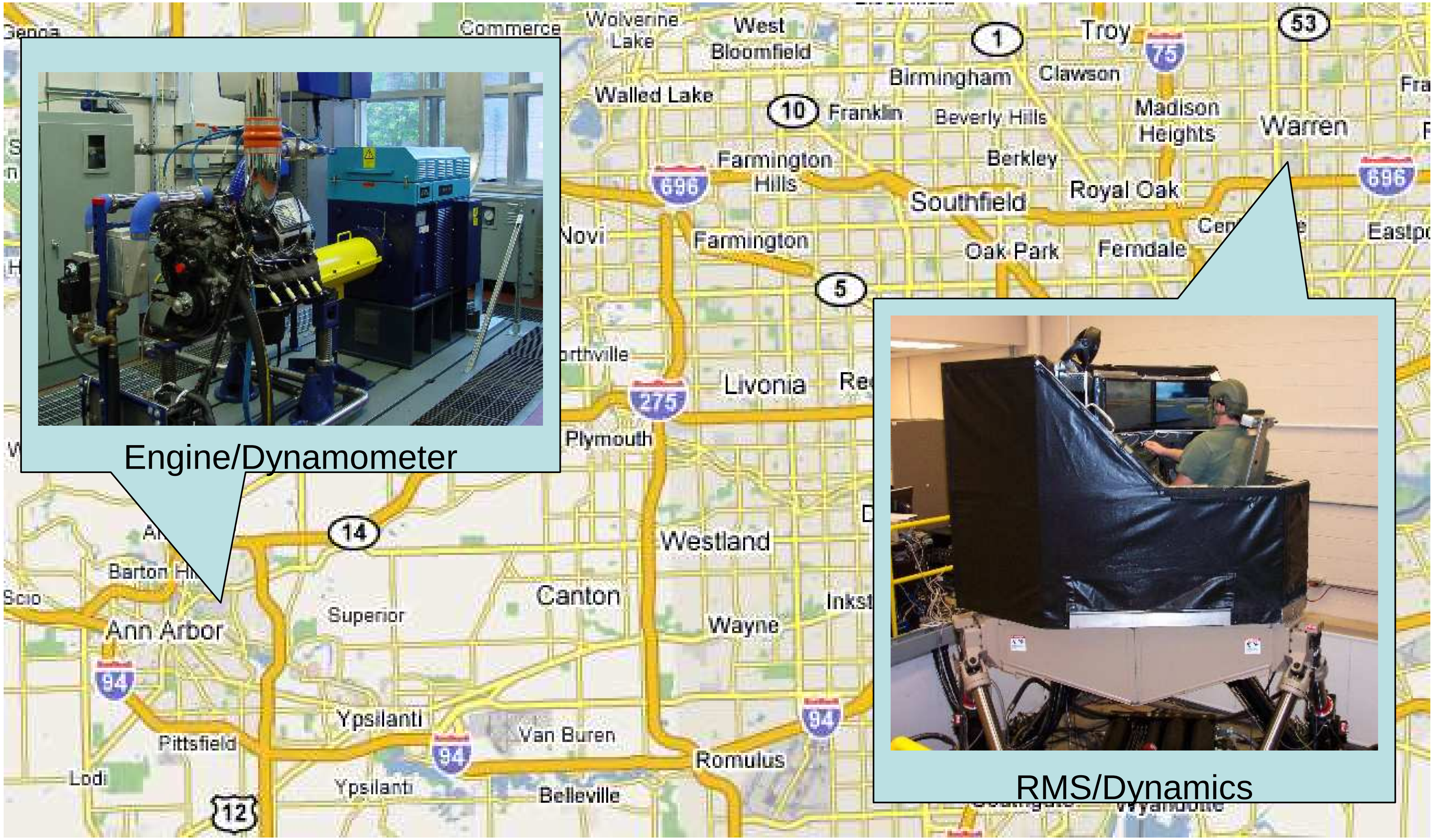
Leaked powertrain energy: Less than 2% of total input energy

Turret/Gun errors: Less than 10 degrees

Bus voltage errors: Less than 80 volts

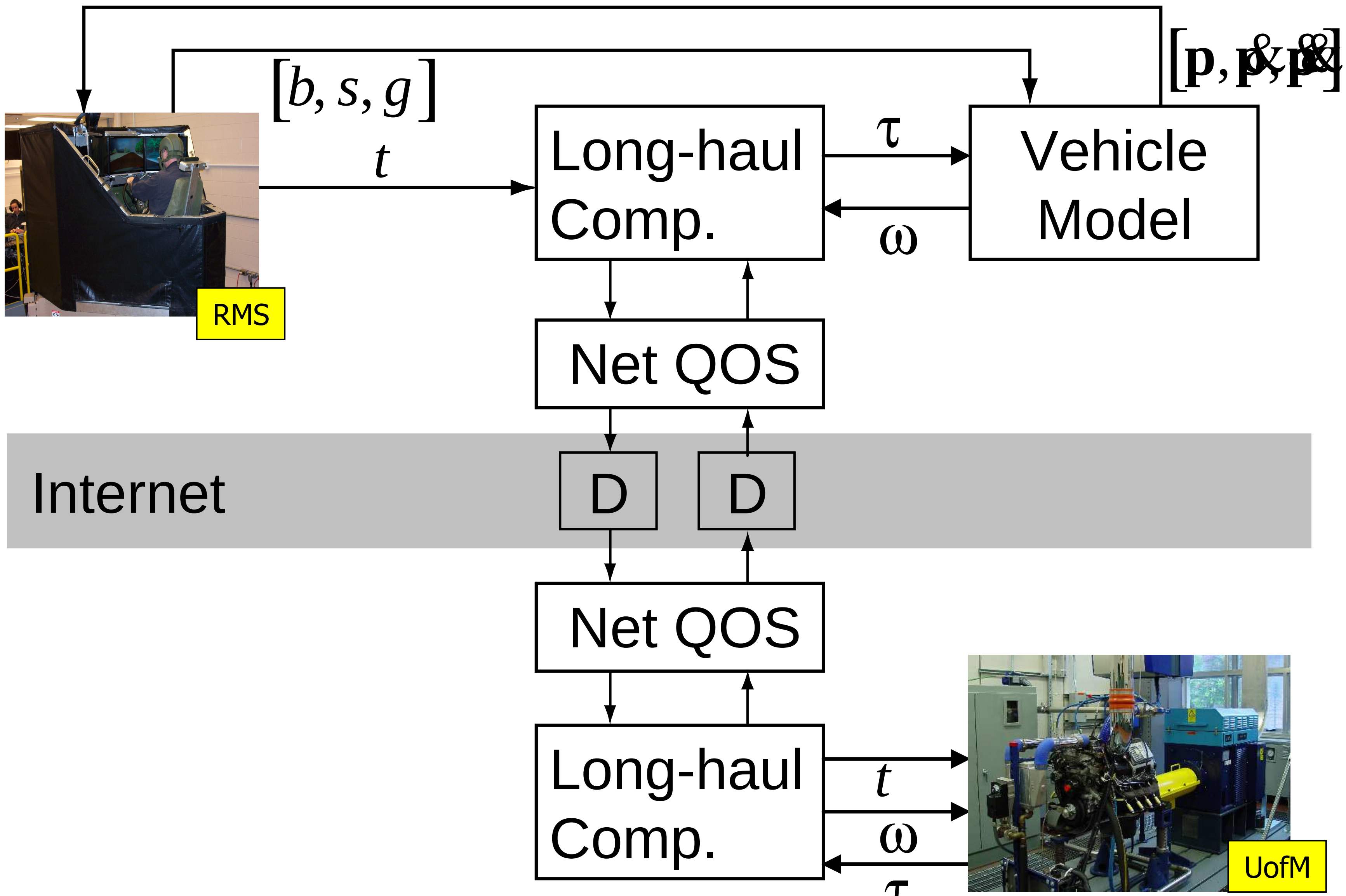
Driver didn’t notice any SC-induced oscillations

Continuing Work



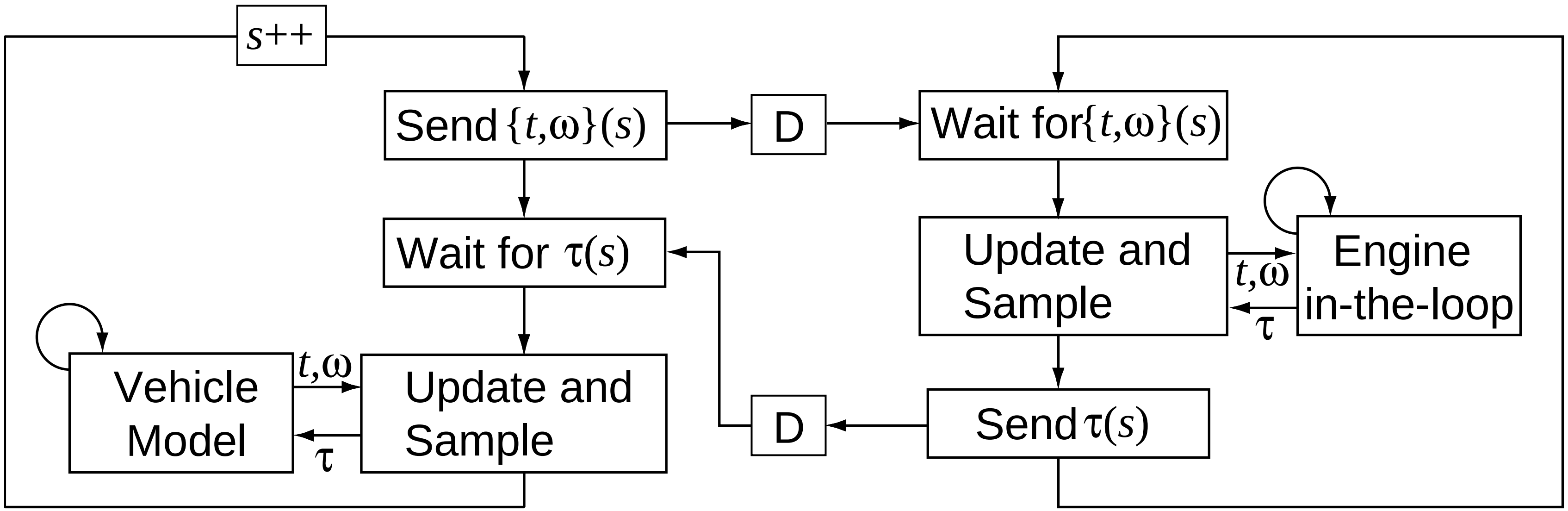
Map Courtesy of Google™ Maps

Observer Free Approach



Description: Approach

- Each lab executes its respective real-time thread
  - Real-time thread executes in the time domain, *t*
  - Non-blocking communications initiate events in the sample domain, *s*
- Do not allow data to accumulate in queue.
- Events are driven by sending and arrival of packets.
- Each update has a limited effect with regard to time.

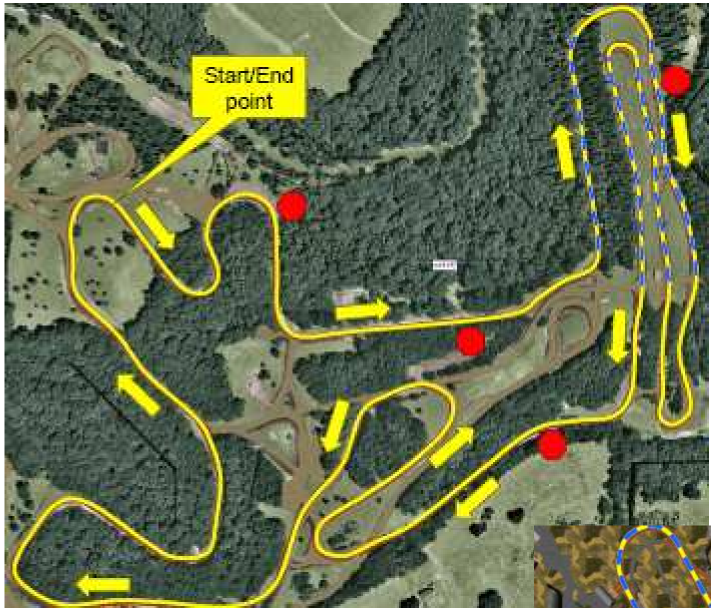


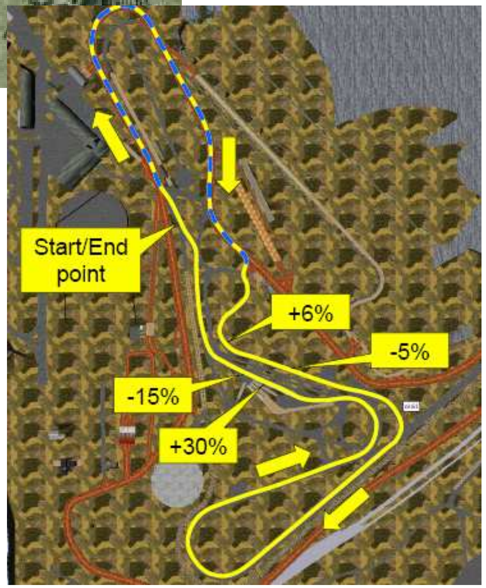
Description: Development plan

- Model-based internet integration with the actual engine
  - Preview added to driver model
- Replace internet model with LAN
- Replace LAN with internet
- Replace VESIM vehicle with TARDEC vehicle
  - Increased transmission damping ($\zeta \approx 0.014$)
  - Scaled engine by 50%
  - Redesigned driveline
  - Increased delay (up to 5x)
- Bring in the Motion Simulator

Description: Experiment

- 4 different drivers
- 2 different delay conditions
  - 25 ms
  - 125 ms
- 2 different closed courses at Aberdeen
  - Munson SFC
  - Churchville B





Results: Internet Delay Benchmark