1. Progress on Task 2, 3, 4, and 5

T-2. Collect data, formulate and estimate models of communication failure.
T-3. Collect data, formulate and estimate models of electrical failure.
T-4. Collect data, formulate and estimate models of synchronized failure and investigate possible causes.
T-5. Collect data, formulate and estimate models and algorithm to diagnose and classify loop detector faults systematically.

(1) The Caltrans freeway sensor network consists of two components: the “sensor system” of 25,000 inductive loop sensors and the “communication network” over which the sensor measurements are transported to Caltrans Traffic Management Centers. We performed a systematic analysis of sensor system failures in District 4, 7 and 11 in California using detector health data from PeMS. We used simple metrics that can be easily computed for very large systems: “productivity” is the fraction of days that sensors provide reliable measurements; “stability” is the frequency with which sensors switch from being reliable to becoming unreliable; and “lifetime” and “fixing time” are the number of consecutive days that sensors are continuously working or failed, respectively.

(2) Productivity captures the underlying sensor system performance and Stability capture communication network performance. The latter is related to the choice of communication technology. Various conclusions were drawn from the study, most prominent of which were: there is huge variability in productivity among districts; all three Districts suffer short-term outages, likely due to the communication network technology used in the different Districts.; the Detector Fitness Program (DFP) is unlikely to be cost-effective, since no District’s stability improved after the DFP, although there is some improvement in productivity.

(3) We identified the following needs for the next phase of work:
• Identify components of communication network of each District (E.g., GPRS modem vs. telephone line; UDP (user datagram protocol) vs. TCP (transmission control protocol), etc.)
• Investigate why certain District has better sensor network health than other Districts and provide ideas on “best practice” on sensor network health management.
• Develop a simple method that relates observed data failures to a particular component(s) in the sensor network.
• Develop a cost-effective method for prioritizing which detectors to fix first.

2. Progress on Task 7, 8, and 9

T-7 Preliminarily Code the “Reporter Generator” software and establish data connection with PeMS.
T-8. Code the software tool for scheduling and tracking the suspect loop detectors for on-site diagnostics.

(1) A preliminary version of the Report Generator program was created which includes an Excel interface to Matlab, Matlab scripts for processing 5-minute and loop diagnostics data from PeMS, and Excel and Powerpoint based loop health reports. The current version of the program produces two separate reports. The first consists of a Powerpoint file with a timeseries plot of the aggregate loop health of the entire freeway over the past year. The plot highlights the healthiest days of the year, both in terms of the PeMS loop health index and the total number of samples collected per day. Figure 1 shows a sample plot for I-210 East from February 2006 to February 2007. The plot evidences the expansion of the detector network beginning in August 2006, as well as a prolonged interruption in communications in January 2007.

![Figure 1: Health history report for I-210 East](image-url)
The health history report also provides a list of the 20 best days of the previous year. On I-210, all of the 20 best days occurred after December 10th 2006.

The second report consists of a color-coded matrix in Excel in which each row represents a VDS station, every column a day, and the color of the cell depends on the PeMS loop health diagnostic. This report serves to quickly identify consistently bad detectors. It can also be used for simulation purposes, to select the best day from which to construct a baseline simulation.

The Report Generator is in an early stage. Future areas of development include:

- providing a direct and automatic connection to PeMS.
- developing a format for reporting different types of failure.
- connecting the program to the models of tasks T-2, T-3, and T-4.

(2) An integrated Matlab/excel software toll is under development that, given an excel file with the list of VDS in a freeway section and some basic freeway topology, achieves the following tasks

- Creates a macroscopic CTM model of the freeway.
- Determines a preliminary set of VDS stations that may contain faulty loop detectors, based on the downloaded PeMS data and the “Reporter Generator” code described above,
- Imputes missing on-ramp and off-ramp flows and calibrates the CTM model based on data obtained from VDS with healthy loop detectors.

This CTJM model will be subsequently used as the basis for an observer-based fault detector.

A lane-by-lane CTM macroscopic density estimator and loop fault detector is under development and testing.

### 3. Progress on Task 9

**T-9. Hardware and relevant software development.**

- **9.1. Purchase and install retractable pole on a PATH van for mounting PTZ camera.**
- **9.2. Develop software for wireless communication and synchronization of two computers: at cabinet and on the van.**
- **9.3. Develop an interface with 170 controller cabinet.**

(1) Installation of a laptop with Linux for

- real-time image analysis for vehicle tracking;
- wireless communication software development; IEEE 802.11a will be used for data passing between the computer for vehicle tracking using video camera and the computer interface with the control cabinet;
It is believed that the real-time image analysis for vehicle tracking is the hardest part for the Onsite Detection Tool development. Two image analysis software systems are under consideration:

- The one used in Berkeley Highway Lab: This software was previously developed several years ago for vehicle tracking only when the cameras were installed very high looking into the freeway downward. In such a circumstance, vehicle tracking is much easier since all the vehicles could be approximated with a Box of certain size and there was no occlusion problem. Since then, some newer versions have been developed at PATH to some extent for vehicle tracking when the camera was mounted on vehicle (for lane and vehicle tracking for Frontal Collision Warning System), or mounted at road side for intersection monitoring. However, either case is quite different from the situation proposed in this project: the camera is to be mounted on a mast on the roadside of a freeway where traffic is much heavier.

- The second software is an open source developed in the University of Reading in U. K. We are trying to make this software work as a backup or an alternative. The reasons to install this software are (a) It was ready developed under Linux for real-time operation; (b) it developed both for people tracking and vehicle tracking, which could be used for some other Caltrans supported project such as CICAS; and (3) the overall structure of the software is simple and clear and well-documented, which would be suitable for further development. Until now, we have not succeeded to make it work in Linux yet, but we are approaching it.

Interface with 170 Control Cabinet:

The essential idea behind Fault Detection is to synchronize ground truth measurement coming from reliable sensors with the loop detector data in order to compare the two data streams for diagnosis. This is proposed to be accomplished by interfacing with existing infrastructure on freeway and freeway on-ramps by bypassing the 170E traffic controller. This approach was chosen because it is easy to implement in the field and because it is nonintrusive. It also allows determining if the 170E controller is part of the problem. The 170E controller will continue to function normally while the fault detection software is running and getting data directly from the loop detector cards.

In order to implement fault detection it is necessary to have a 170E Traffic Controller Real-Time Bypass system. The main idea of the system is to by-pass the existing controller to acquire the loop detector data directly from the cabinet using a PC and interfacing hardware. This hardware will Y-interface the bus between the sensors and actuators and the controller; thus having access to the data coming from the loop detectors cards without disturbing any of the other functions of the 170E controller, like ramp metering light control. The following figure shows the system architecture with its main components.
• Model 222 Inductive Loop Detector Sensor Unit
• Model 200 Traffic Light Actuator Unit
• Communications Bus
• 170E Controller
• Interfacing Circuitry
• Digital Input Output Board
• The Computer with Bypass Software

An important part of the bypass system is the interfacing circuitry. This device will allow the connection between the cabinet and the data acquisition board connected to the PC. Below is a diagram and an explanation of the functions of the circuit.

**Line 1:** The sensor data is transmitted both to the controller and the computer for processing.

**Line 2:** Unaltered communication is maintained between the controller and the cabinet.

Besides the hardware interface, software in C code is under development for real-time decoding the 170 messages according to loop map and time stamps.

(4) Selection candidate test site for future work at control cabinet level: Some candidate loop stations in Berkeley Highway Lab have been selected which could be used for Onsite Detection Tool development.