Yi-Chen Lai

Implementation of a Realtime Multisensor System

Duration of the Thesis: 6 months
Completion: Jan. 2014
Tutor: M.Sc. Rainer Schützle
Examiner: Dr.-Ing. Martin Metzner

Background

In the Institute of Engineering Geodesy Stuttgart (IIGS), a multisensor system consisting of different sensors is settled on a vehicle for surveying applications. With the dynamic measurements, a vehicle trajectory can be determined by several calculated values derived from the raw data of sensors. Therefore the transformation from acquired measurements into the final required values is one of the most important concepts in the operation.

The multisensor system is executed mainly by the environment of LabVIEW, which is a kind of visual programming languages. In order to combine all the necessary tasks into a complicated program structure, there will be a high possibility to have the unstable operating states or the unexpected mistakes in the realtime implementation; thus the related applications for the solution functions offered in LabVIEW are essentially discussed.

Objective

Originally, the multisensor system has an existing LabVIEW program for the implementation, but the functions included are not sufficient for the updated operation. In the thesis, to advance the original program into a real-time and user-friendly environment is the most important task. For these enhancement purposes, it should be improved for mainly three parts: time-stability for the synchronization problem, real-time implementation of estimation algorithm and more convenient user interface.
The original program includes only the acquisition functions without data processing procedures. The post-processing is additionally applied after the program executions are totally terminated and the acquired measurements are all recorded as data files.

Presently the objective in the new updated execution is to convert the raw data into required value as well as to estimate the state of dynamic system by applying Kalman Filter in realtime applications. As the complicated structure with several sensors applied in parallel, the synchronization problem may be happened if the separate executions are not simultaneously run in the same time sequence. And to have a presentation of the operation status and result in an easier way to understand, the realtime visualization is also another important task as one of the improvements. In addition, the measuring tasks of the system are all operated directly in the driving vehicle. Accordingly, a convenient and friendly interface which can be used in the realtime environment is definitely necessary for the system operators.

Eventually, the main tasks then can be concluded as three topics: synchronization problems, realtime data analysis and visualization and user interface improvements.

Figure 2: The simplified programs of original and updated ones

Figure 3: The updated program structure with main tasks presented
Synchronization Problem

According to the program structure with parallel-running loops, the first synchronization problem is about the relationship among data acquisition sections and filter calculation part.

![Figure 4: The program structure of parallel data acquisition problem](image)

At the same time sequence, the acquired measurements from each operating sensors shall be communicated to the calculation section within the specified interval. In the case of data acquisition with errors, the measurement data have the possibility to be missed in the execution and the outliers will be shown in the measurement comparison graph.

![Figure 5: The measurement comparison graphs in normal situation and with error occurred](image)

Therefore to solve the problem, it is able to concentrate on two principle issues: measurement delivering and correct executing time sequence. To deliver the measurement values between separate loops, the function of Notifier, which is used to communicate data between two or more independent parts of code in the same execution, is selected to satisfy the condition of parallel-running executions; it means the measurements are able to be sent from the acquisition parts to the calculation section by the specific notification without normal wire connecting in LabVIEW programming.
After successfully communicating the measurements among several parallel-running loops, the correct time sequence is also important in the program design. The part of filter calculation is only executed if the measurements are all acquired in each time period. Thus the function of Rendezvous, which synchronize several separate tasks at a specific point of execution, is applied for the situation.

According to the dataflow shown by the connecting line in the program, the part of filter calculation is later than the point which receives four available measured values from the acquisition loops, and all the processes will be completed in the specified time interval.
The parallel data acquisition is much more stable with the applications of Notifier and Rendezvous; the sensor measurements are communicated without wire connecting among separate executions by Notifier functions, and the time sequence between acquiring and calculating processes is arranged in order with Rendezvous operations. With the complicated programming structure, it is an essential advance for the realtime implementation.

**Realtime Data Analysis**

In the original program design, the filtering calculation is not included and only additionally implemented after all the data acquisition processes completed; nevertheless, it is not efficient for the realtime implementation. According to the improving purpose in the thesis, the filter processing is necessary to be applied synchronously with other operations in realtime.

After the issue of realtime data communication mentioned above, the measurements are able to be delivered from the parallel executed acquisition tasks to the data analysis section. In the LabVIEW programming environment, it is available to insert the MATLAB code for mathematical purpose by MATLAB script node. According to the algorithm procedures of the selected estimation method of Kalman Filter, the equations are separate into three steps: prediction, innovation and update. And for the circular value communication between the current and next time epoch, the operation of *shift register*, which is possible to communicate information between each continuous loop-iteration, is applied to solve the problem for the extended use of while loops.

After the implementation of Kalman filter, the estimation result as realtime visualization is able to be presented by the covariance matrix of updated state vector. The matrix is iterated in the circular filtering processes and includes all the information of influences and observations offered in the algorithm.

![Figure 8: The presented covariance matrix of updated state vector](image)

The diagonal elements show the variance values of updated state vector. If any unexpectedly large number is appeared in one of the variances, it should be highly concerned whether the execution state of the multisensor system is still in normal or not.
User Interface

For the purpose of connecting instruments with operating users, a controlling interface is required to input the required data settings into the program execution. To have a more convenient operation in the program design, the functions shall be improved.

Firstly, the interval settings are different for each operated sensor and calculation loop, and the time period used for the estimation algorithm is automatically computed and shown in the setting panel. Secondly, as part of the terminate process of the whole program, the recording file path for the measurement data of each sensor is also important. After choosing the folder path, the name of data file will be automatically composed with the existing information and the customized message data. Lastly, in order to convert the raw data into required measurements, the offset value and transfer factor shall be entered into the corresponding blanks.

Except for the controlling interface, the output representation is another important task to display the outcome diagrams. An appropriate layout of signal visualization will make the controllers much easier to understand the operating state in the realtime implementation. For the basic comparison purpose, the acquired measurements from different sensors are plotted by each time epoch. And another graph is applied to plot two trajectories from different sources: the acquired GPS data and the filtering result from other realtime measurements.

Figure 9: The front panel representation of the updated operation program
Conclusion

The multisensor system consists of several sensors and used to acquire the spatial measurements. With these measured values, two trajectories of the system are able to be determined respectively with GPS coordinate data and estimation result derived from other sensors. Compared with the original program concept, the realtime estimation process shall be added in the updated program.

In order to contain all the acquisition processes for every sensor and also the estimation calculation simultaneously in the same time sequence, some synchronization problems have the possibility to be occurred. To make sure the possible mistakes are able to be avoided, the topic of data communication in LabVIEW is essential to make the structure with parallel-running loops in order. After the required data is acquired and delivered to the data processing section, the selected estimation method is able to be operated with recursive filtering in each iteration period.

Constructing a user-friendly interface is another principle improvement; especially in the realtime implementations, a clear and easy-understanding executing interface is important for the updated program. Between the instruments and operators, the input settings are designed to be a control panel, and the output representation displays the diagrams as the program results.

In the thesis, the original program is advanced with improvements for the realtime purpose. The implementation of the realtime multisensor system in IIGS is able to be achieved with more useful function operations in the new LabVIEW program.