Redundancy and Synchronization Scheme of Surveillance Data Processing System in ATCS

Hyounkyoung Kim and Daekeun Jeon*
CNS/ATM Team of Satellite Navigation · Application Technology R&D Center
Korea Aerospace Research Institute, Daejeon, Korea
{kimhk, bigroot}@kari.re.kr

Abstract

In this paper, the redundancy and synchronization scheme for the high reliable surveillance data processing system for air traffic control is suggested. Two indices for reliability of surveillance data processing system are identified and full triple redundancy scheme and high availability software for the redundancy control are proposed. The synchronization issues among triple redundant nodes are analyzed and a mixed synchronization scheme according to the data characteristics is suggested. The performance test for the two indices of reliability is fulfilled and the test results are described.

Keywords: Surveillance Data Processing, Availability, Triple Modular Redundancy, Synchronization

1 Introduction

Air traffic control(ATC) is the real-time service by ground systems and controllers, for aircraft on the airport and in the air to avoid collision and provide smooth air traffic. According to the aircraft position, air traffic control is divided into three control sections - airport, approach, and en-route. Air traffic control system(ATCS) for en-route and approach control generally consists of surveillance sensors such as radar and ADS-B, flight data processing system(FDPS), surveillance data processing system(SDPS), controller working position(CWP), system management control system(SMC), recording and playback system, and so on. SDPS is a core system which generates track information and proper alerts to ATCS controllers using sensor data. In this paper, the triple modular redundancy design to enhance reliability the SDPS is discussed. And proper synchronization scheme to clear the issues considering redundancy is proposed. The remainder of this paper is as follows. Triple redundancy and synchronization design of the developed system is presented in Section 2 and 3 respectively, and the test results are described in Section 4. Finally, concluding remarks are indicated in Section 5.

2 Triple redundancy in SDPS

2.1 Reliability index of SDPS

ATCS should provide various information to ATC controllers without discontinuance because it is mainly related to the safety of the passengers. For the seamless service, the safe critical subsystems composing ATCS, such as SDPS and FDPS, should be highly reliable. To increase SDPS subsystem reliability, hardware based triple modular redundancy (TMR) is applied for SDPS in this study. Performance specification for SDPS, published by Eurocontrol in 2012[2], defines the ATM surveillance system requirements

IT CoNvergence PRActice (INPRA), volume: 3, number: 1 (March), pp. 43-50

*Corresponding author: CNS/ATM Team of Satellite Navigation·Application Technology R&D Center, 169-84 Gwahak-ro, Yuseong-gu, Daejeon, Korea, Tel: +82-42-860-2183

in terms of accuracy and reliability. Requirements related to accuracy in present study are verified during system performance tests and the results of the tests are described in [5], [4]. The reliability of SDPS is split into continuity and availability. Continuity for one node can be computed with mean time between critical failure(MTBCF) as follows[2].

$$Continuity = 1/MTBCF \tag{1}$$

The critical failure for MTBCF is the loss of horizontal positions of all aircrafts during at least 10 seconds at the output of the surveillance system. Availability is related to not only MTBCF but also mean time to repair(MTTR) and mean report time(MRT) which means the time between error recognition time and repair ready time. Availability values of one SDPS and N-modular redundant SDPS are calculated by [6], [2]

$$Availability_1 = MTBCF/(MTBCF + MTTR + MRT) \tag{2}$$

$$Avaiability_N = 1–(1–Availability_1)^N \tag{3}$$

Availability is the ratio of MTBCF and (MTBCF+MTTR+MRT). In order to maintain the the high availability, MTTR and MRT should be shorter if MTBCF is short. In addition, total system availability is highly affected by the number of redundancy, N. If N = 3, even though $Availability_1$ is only 99%, $Availability_N$ can be 99.9999%.

However, the exact MTBCF estimation for the software based system is not possible. Therefore, the continuity and availability of software should be verified during long-term operation test. Operation test requires more than one year to verify system function, performance, stability, and availability. The period includes the system modification and system on and off test. Therefore, continuity test may be conducted during less than 1 month. This is the reason that many legacy ATCS systems have used the following indices instead of continuity and availability to come up with system reliability considering time limit problem.

- Automatic switchover time: time difference between error occurrence time in the active node and time to provide stable system track to controllers from new active node without human intervention
- MTTR and power restart time: Power restart time means hardware power-off and start-up time and MTTR is the sum of power restart time and time to replace a hardware component.

The adoption of the indices above for system reliability is quite reasonable for a fault tolerant system. This is due to the facts that automatic and immediate switchover to new active node without recognition of controllers will support the seamless operation and the small MTTR may guarantee acceptable level of availability in spite of uncertainties due to imperfect knowledge of MTBCF.

The minimum requirement of automatic switchover time defined by [1] is that the failure detection of active node and switch over to standby node should be done in less than one time interval between information updates of the display. The requirement finally determined in this study is 1 second for the conservativeness. The performance requirements for MTTR and power restart time are set to 30 minutes and 20 minutes, respectively.

2.2 State of TMR nodes

In TMR SDPS, each SDP is called a node. The definition of node state and state transition diagram of SDPS is shown in Table 1 and Figure 1.

In Figure 1, “Degraded” is not considered as a state because it is not a stable state. A hot-standby node can be active node only if the two conditions are met.

Table 1: Definition of node state

state	Power On	Wait for Command	Processing	Output
active	◯	◯	◯	◯
hot-standby	◯	◯	◯	×
cold-standby	◯	◯	×	×
power off	×	×	×	×

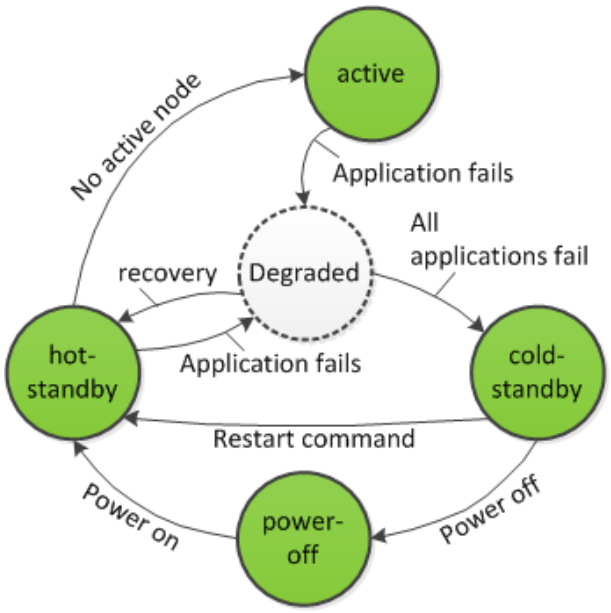


Figure 1: Node transition diagram of SDPS

- There is no active node
- The priority is the highest among hot standby nodes (the lowest IP address)

For the operating SDPS, the following time elements should be considered. Each element is added to state transition according to the state of node. For example, (a) to (d) should be added for the transition from "power off" to "active", and only (d) is required for the transition from "hot-standby" to "active".

(a) System start-up time

(b) Software start-up and initialization time

(c) Processing plot data and calculated stable system tracks

(d) Switchover time (hot-standby to Active)

Time (a), (b), and (d) are fixed once system is setup, and are affected by system configurations. But time (c) is variable because more than two successive plot data are required to generate stable system tracks. The time between plot data is determined by the number of sensors and sensor update period. When a node in cold-standby state changes its state to active, it cannot assure the continuity of system because its total transition time will be variable. At least one hot-standby node should exist to back up the active node within continuity time threshold. For that reason, "active, hot-standby, hot-standby" TMR configuration is suggested for the fault tolerant SDPS. The default state of each SDPS node, which represents the automatic state after the node is powered on, is "hot-standby" not "cold-standby".

2.3 HASW(High Availability Software)

For TMR, the state of applications of each node and the state of each node should be separately managed. Also, each node should communicate with external system to receive command and send the state. The HASW can be implemented into a function inside each application process or a separate process in each node or in an external system. In any cases, the function of HASW should be repeated in the appropriate time even though all other application functions are not in operation. The main application functions of SDPS, which consists of generating system track and safety alerts to controllers, consume much hardware resource to process highly complex algorithms. That may bring about the delay, so a simple software, HASW is suggested in this study. HASW is a software program that controls the application and node state according to the heart beat information from each application in its own node and from other HASW in other nodes. This interface scheme is also used in [3]. Also, it can receive user command from SMC. The current state is sent periodically to SMC and other nodes. The scheme of HASW is shown in Figure 2. HASW is independent of other software. TMR can be configured as various ways as shown in Figure 3. We adapted Figure 3(a) configuration according to the developed system policy.

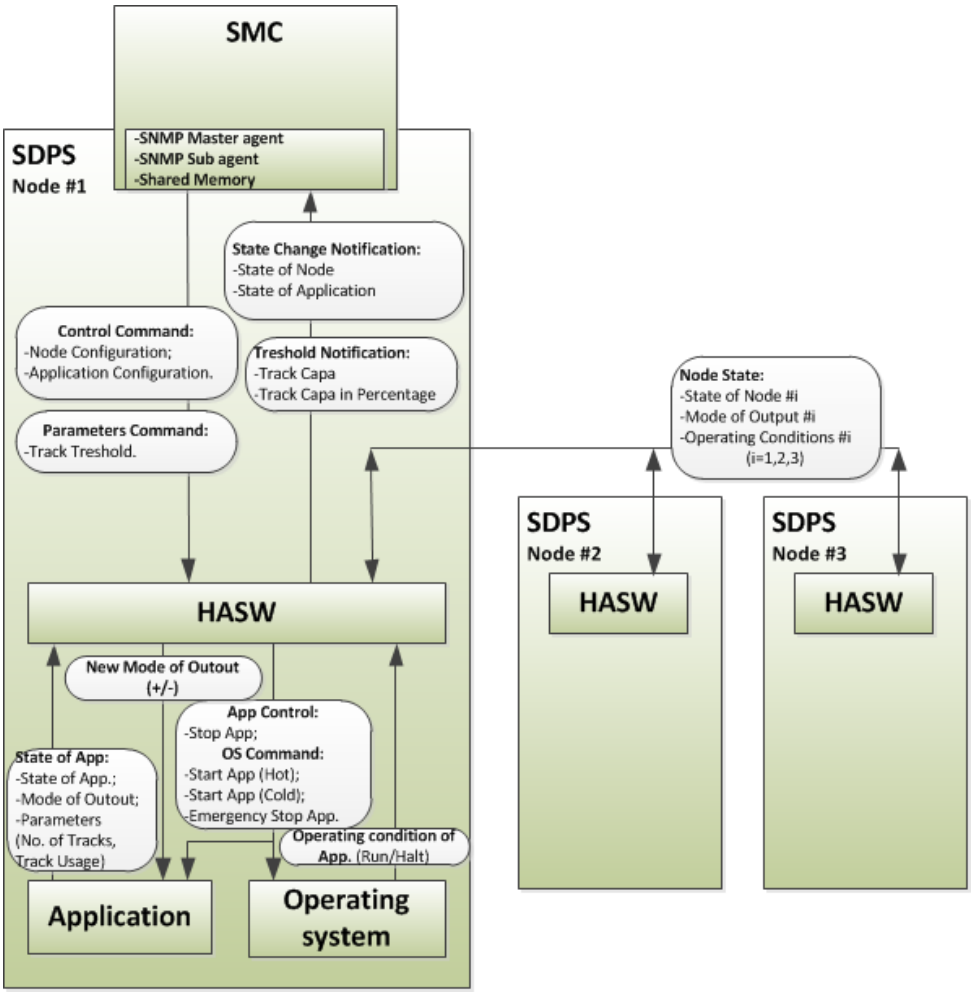
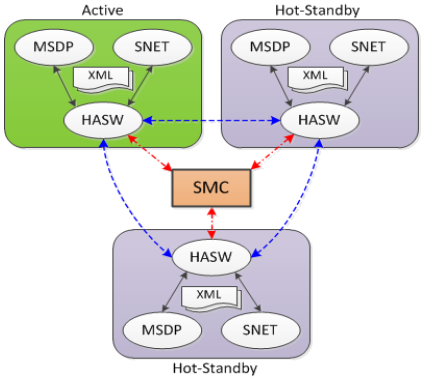
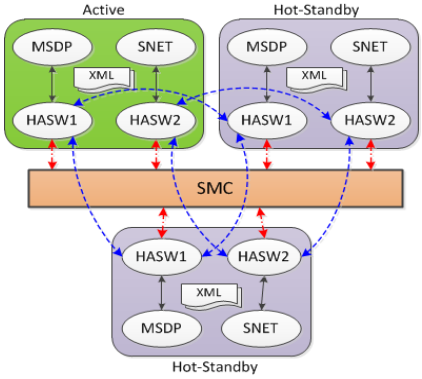


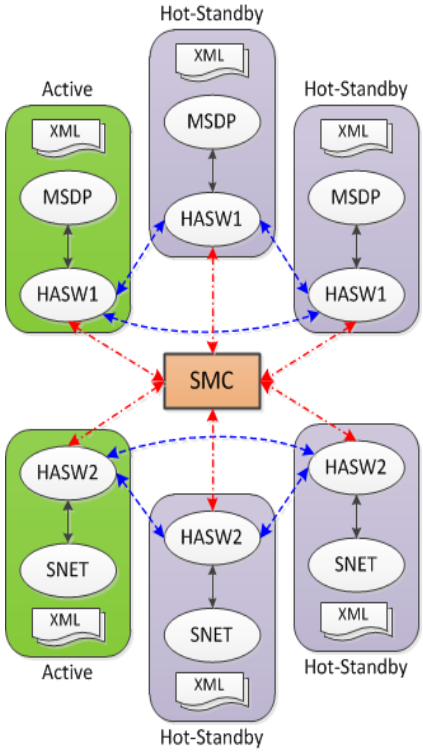
Figure 2: Scheme of HASW



(a) 1 HASW and N applications in a computer



(c) N HASWs and N applications in a computer



(b) 1 HASW and 1 application in a computer, N computers for 1 node

Figure 3: Possible TMR configurations of SDPS

3 Data Synchronization

All information processed in each node should be synchronized among nodes as if only one node is alive even though in switchover condition. Information in SDP can be categorized in three types according to the data update period.

(a) Long-term : Information set up by local adaptation when the system is installed and fixed during operation – such as aerodrome parameters, system performance, and so on

(b) Short-term : Information set up by local adaptation and updated in case of necessity during operation – such as sensor parameters

(c) Real-time : Information periodically updated by SDPS – such as track data

Both of Information (a) and (b) should be saved in a physical storage as to be used in the application initialization step after restart. Information (a) should not be updated during system operation but the data should be updated by maintenance plan. Information (b) may be updated by system operator periodically. Information (c) should be calculated by SDPS applications such as multi-sensor data processing(MSDP). It is automatically updated during operation. Figure 4 shows the information flow diagram in TMR SDPS to secure consistency among nodes.

To determine the policy of synchronization, the generator of the data is put ahead of everything. We suggested that the generator should be the owner who is responsible for the latest version of data and for retransmitting the data on demand or periodically. The followings are brief synchronization process of each information type.

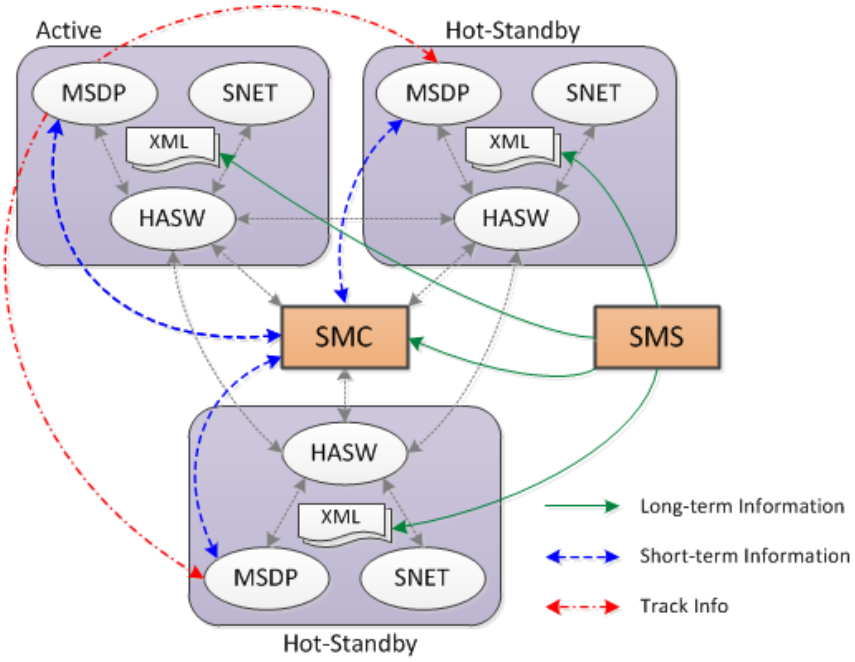


Figure 4: Information flow for the synchronization in SDPS

A. Long-term information

It is generated and distributed as a set of XML files by an off-line system, called SMS(Software Management System). After installation, only SMS can update the information with a permitted maintenance plan. However, some files and their items, which correspond to short-term information, are also parts of XML files so they should be updated without a maintenance plan. For this reason, originally distributed files are saved in a stable directory and copied to working directory. System uses working files during operation, but whenever the system is about to recall the original state, system operator can restart in cold mode with the original file set.

B. Short-term information

Sensor related parameters such as bias, blanking area, and sensor enable mode can be changed by SMC operator. So, SMC is the owner and responsible for the latest data. When SDPS receives the sensor parameters from SMC, it updates the XML files in working directory. Figure 5 shows the synchronization process of short-term data processing. The dashed green box is added for state transition from cold-standby to hot-standby. The remains are same for the normal operation. Only adding the data exchange in green box, the extension to other subsystems can be easily made.

C. Real-time information

Because the track number for an aircraft is used as a key for CWP and FDPS, it should be same in all nodes. The track information including track number is generated by MSDP. It should be broadcasted to CWP, FDPS, and other SDPS nodes. Whenever a track information is received, hot-standby node update its track number according to squawk code and horizontal position. This information is on the memory but is not saved as a file.

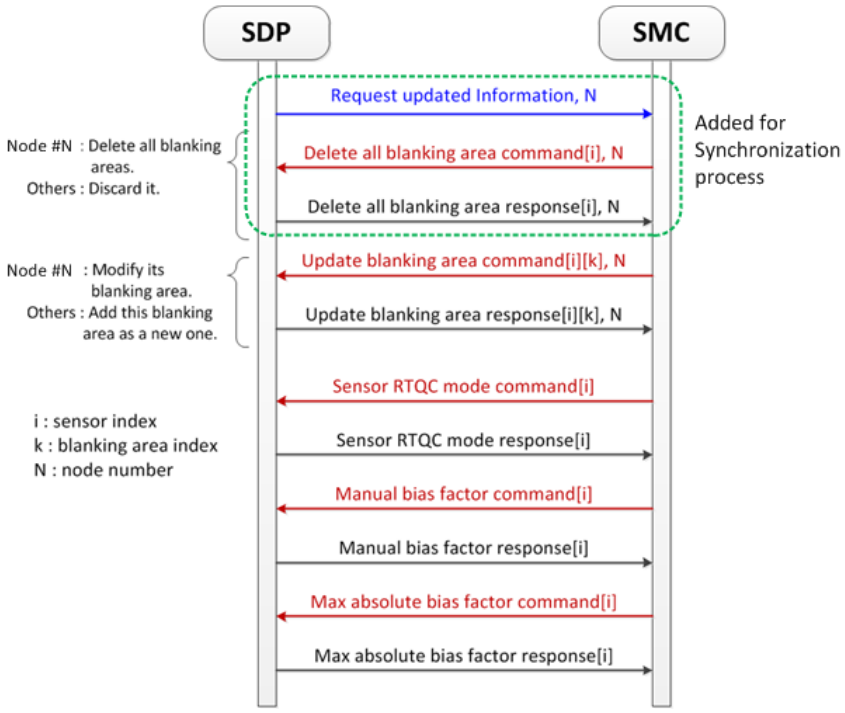


Figure 5: Synchronization of Short-term information

4 Test Results

As specified in Section 2.1, the performance requirement for the switchover time is 1 second. Using SMC event logs during performance test and operational test, it is confirmed that the switchover time was less than 200ms in average. As mentioned in Section 2.1, the total availability of TMR SDPS seems to be very high even in case the availability of each SDPS node is not high enough. So, the critical case, such as power-down of all ATCS, is considered. In this case, only MTTR and power restart time are the key element because $Availability_1$ is same as $Availability_N$. The performance requirements for MTTR and power restart time are 30 minutes and 20 minutes, respectively. With the spare part for SDPS hardware, the time for MTTR and power restart time was less than 20 minutes and 10 minutes.

5 Concluding remark

In this paper, the effective triple redundant scheme and synchronization in triple redundant nodes of SDPS are suggested. To meet the smooth switchover requirement, true TMR of active-hot standby-hot standby is suggested. It is confirmed that the switchover time meets the system performance requirement and remarkably short comparing the minimum requirement of continuity. Also, a mixed synchronization scheme considering characteristics of information in SDPS is suggested. Although this research is based on the SDPS in ATCS, the redundancy and synchronization scheme in this paper are not dependent upon the developed SDPS configuration. With their flexible characteristics, the schemes are expected to be effectively applied for general DMR and TMR systems.

References

[1] Eurocontrol standard document for radar surveillance in en-route airspace and major terminal areas. Technical Report SUR.ET1.ST01.1000-STD-01-01, EUROCONTROL, 1997.

[2] Eurocontrol specification for atm surveillance system performance. Technical Report EUROCONTROL-SPEC-0147, EUROCONTROL, 2012.

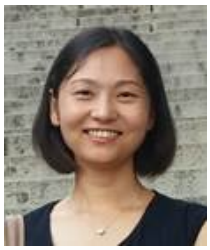
[3] F. Cristian, B. Dancey, and J. Dehn. Fault-tolerance in the advanced automation system. In *Proc. of the 20th International Symposium Fault-Tolerant Computing (FTCS'90), Newcastle Upon Tyne, UK*, pages 6–17. IEEE, June 1990.

[4] Y. Eun, D. Jeon, H. Ko, and C. Yeom. Multi radar tracking performance evaluation of surveillance data processing system for air traffic control. In *Proc. of the Korea Society for aeronautical & Space Sciences Conference (KSAS'11)*, September 2011.

[5] D. Jeon, Y. Eun, H. Kim, and C. Yeom. Multi-sensor data processing for air traffic control system. In *Proc. of the 32nd IASTED International Conference on Modelling, Identification and Control (MIC'13), Innsbruck, Austria*, February 2013.

[6] M. Rausand and A. Høyland. *System Reliability Theory. Models, Statistical Methods and Application*. Wiley-Interscience, 2003.

Author Biography



Hyounkyoung Kim received the B.S. degree in computer engineering and the M.S. degree in information and communication engineering from Chungbuk National University, Cheongju Chungbuk, Korea in 1999 and 2001 respectively. She is currently a senior researcher with Korea Aerospace Research Institute, Daejeon, Korea. From 2000 to 2005, she participated unmanned air vehicle system projects. Her current research interests are mainly in the field of CNS/ATM systems.

Daekeun Jeon received the B.S. and M.S. degrees in aerospace engineering from Seoul National University, Seoul, Korea in 1993 and 1995, respectively. He received his Ph.D. in aerospace engineering from Korea Advanced Institute of Sceience and Technology, Daejeon, Korea in 2013. From 1995 to 2000, he was an aircraft performance engineer for T-50 supersonic advanced pilot trainer at Samsung Aerospace Industries. He participated in the development of T-50 full flight simulator at Dodaam systems from 2000 to 2005. He is currently a principal researcher with Korea Aerospace Research Institute. His current research interests include multi-target multi-sensor target tracking algorithms and air traffic management.