

論文

DOI:10.5139/JKSAS.2010.38.8.781

저급 프로세서에 적용 가능한 무인기용 GPS/INS 고장검출 알고리즘 구현

유장식*, 안종선*, 성상경*, 이영재*, 천세범**

A Realization of Applicable GPS/INS Fault Detection Algorithm for UAV using Low Grade Processor

Jangsik Yoo*, Jongsun Ahn*, Sangkyung Sung*, Young Jae Lee* and Sebum Chun**

ABSTRACT

In the GPS/INS integrated system fault detection, algorithm based on a chi-square distribution is commonly used. In this paper, it has been proposed simplified GPS/INS fault detection algorithm that is combined conventional RAIM (Receiver Autonomous Integrity Monitor) and algorithm based on chi-square distribution for UAV using low-grade processor. It use a fault model to verify the proposed algorithm and produced the result.

초 록

GPS/INS 통합시스템의 고장검출에는 카이제곱 분포 기반의 알고리즘이 일반적으로 사용된다. 본 논문에서는 기존의 RAIM (Receiver Autonomous Integrity Monitor)과 카이제곱 분포 기반의 알고리즘을 결합하여 무인기용 저급 프로세서에 적용하기 위해 단순화시킨 GPS/INS 고장검출 알고리즘을 제안한다. 제안한 알고리즘의 검증을 위해 고장모델을 사용하여 그 결과를 나타내었다.

Key Words : AAIM(항공기 단독 무결성 모니터링), RAIM(수신기 단독 무결성 모니터링), Protection Level, GPS/INS, Fault Detection(고장검출)

1. 서 론

GPS는 현재 측정치에만 의존하는 스냅샷 방식의 항법을 수행하여 INS에서 발생하는 드리프트 오차 없이 항법 해를 산출할 수 있다. INS는 단기간 정확도는 GPS보다 우수하지만, 일정 시간이 경과되면 오차가 누적되어 발산하게 된다. 두 시스템은 서로 상호보완적으로 통합되어 사용되기도 한다. 일반적으로 GPS/INS 통합 방법은 약결합, 강결합, 초강결합이 사용되며 본 논문에서는 구현이 간단한 약결합 방식을 GPS/INS 통합에 사용하였다.

카이제곱 분포 기반의 시험기법은 카이제곱 분포 기반의 검정통계량을 생성하고 요구되는 오경보율 (False Alarm Rate)과 자유도로 계산한 한계치와 비교하여 고장검출을 수행한다[4]. 이 기법은 일반적으로 SCST (State Chi-Square Test) 시험기법과 카이제곱 시험기법과 GLR (Generalized Likelihood Ratio) 시험기법으로 이루어져있다.

RAIM은 GPS 단독 수신기만을 사용하는 고장검출 기법이며 여러 가지 고전 RAIM 기법들은 서로 수학적으로 동치라는 것이 증명되었다[1]. 기존의 GPS/INS 고장검출 알고리즘 중에서 의사거리의 잔차를 이용한 방법과 해 분리기법 (Solution Separation Method)을 이용한 방법이

RAIM은 GPS 단독 수신기만을 사용하는 고장검출 기법이며 여러 가지 고전 RAIM 기법들은 서로 수학적으로 동치라는 것이 증명되었다[1].

기존의 GPS/INS 고장검출 알고리즘 중에서 의사거리의 잔차를 이용한 방법과 해 분리기법 (Solution Separation Method)을 이용한 방법이

† 2010년 2월 19일 접수 ~ 2010년 7월 23일 심사완료

* 정희원, 건국대학교 항공우주정보시스템 공학과

교신저자, E-mail : younglee@konkuk.ac.kr

서울시 광진구 화양동 1번지

** 정희원, 한국항공우주연구원

있다[4,9]. 각각의 알고리즘은 저급 프로세서를 사용하는 무인기용 고장검출에는 복잡성으로 인해 적용이 어려운 단점이 있다. 본 논문에서는 기존의 GPS/INS 고장검출 알고리즘의 한계점을 분석하고 새롭게 제안한 구조의 특징을 이용해보다 단순한 구조의 알고리즘 구조를 설계하였고, 실제 데이터에 적용해 그 결과를 분석하였다.

II. 고장검출 알고리즘

2.1 기존의 GPS/INS 고장검출 알고리즘의 한계

기존의 의사거리 잔차를 이용한 GPS/INS 고장검출 알고리즘은 순수항법을 통해 얻은 사용자의 위치와 GPS 궤도력으로 계산된 위성사이의 거리를 의사거리 측정치와 차분하여 잔차를 생성하고 고장검출에 이용한다[4]. GLR 기법은 고장이 발생에 대한 가설검정으로 고장의 원인이 GPS인지 INS인지 구별하여 제거할 수 있지만 동시에 고장이 난 경우 의사거리 잔차가 0이 되는 경우를 피할 수 없는 단점이 존재하며 알고리즘의 복잡성으로 인해 저급 프로세서를 사용하는 시스템에는 적용하기 어렵다. 그림 1은 의사거리 잔차를 이용한 알고리즘의 구조도를 나타낸 그림이다.

기존의 해 분리 기법을 이용한 GPS/INS 고장검출 알고리즘은 GPS의 항법 해와 순수항법 해의 차분 값을 상태벡터로 사용한다. GPS의 고장을 찾기 위해서 전체 집합의 해와 부분 집합의 해의 차이를 이용해서 고장위성을 판별하게 된다.

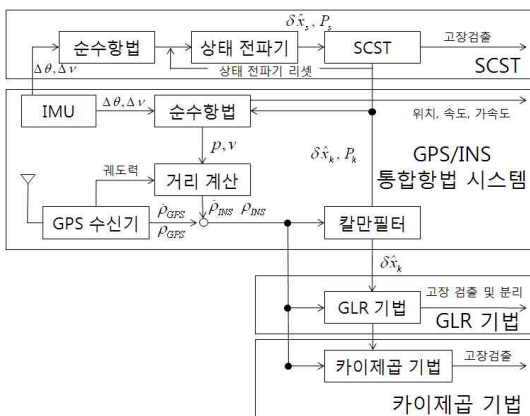


그림 1. 의사거리 잔차를 이용한 알고리즘[4]

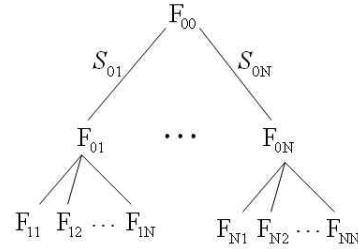


그림 2. 해 분리기법을 이용한 알고리즘[9]

그림 2에서 F_{00} 은 모든 위성을 고려한 전체 집합의 항법 해와 INS 항법 해의 차분 값이고 F_{0N} 은 N번째 위성을 제거한 부분 집합의 항법 해와 INS 항법 해의 차분 값을 의미하며 S_{0N} 은 F_{00} 과 F_{0N} 사이의 차분 값을 의미한다. 만약 1번 위성에서 고장이 발생한 경우 F_{01} 을 제외한 나머지 부분 집합과 전체 집합의 해는 고장 난 측정치를 사용하게 된다. 따라서 모든 차분 값들 중에서 S_{01} 이 가장 큰 값을 갖게 되는 특성을 이용하여 고장검출을 수행한다.

해 분리 기법을 이용한 알고리즘은 고장검출을 위해서는 많은 수의 칼만필터링이 요구되기 때문에 의사거리 잔차를 이용한 알고리즘과 마찬가지로 저급 프로세서에 적용하기에는 무리가 있다.

2.2 저급 프로세서 적용 가능한 무인기용 GPS/INS 고장검출 알고리즘

기존의 의사거리 잔차를 이용한 방법에서 GPS/INS의 고장을 구별하기 위해서 GLR 시험 기법을 사용하였다. 본 논문에서는 RAIM을 사용하여 GPS의 무결성 정보를 검증하고 이를 기반으로 INS의 고장을 검사하는 구조로 알고리즘을 설계하였다. 이러한 구조는 GPS정보와 INS 정보를 통합하는 것이 아니기 때문에 기존의 의사거리 잔차를 이용했을 때 잔차가 0이 되어 GPS와 INS에 동시에 발생한 고장을 감지하지 못하는 경우를 피할 수 있게 된다. 또한 본 논문에서는 2000년도 이전에 주로 브라운[2]에 의해 연구된 고전 RAIM을 적용했지만 이 뿐만 아니라 현대 RAIM의 적용만으로 간단하게 GPS/INS 고장검출 알고리즘의 성능을 개선시킬 수 있다. 그림 3은 본 논문에서 단순화시킨 GPS/INS 고장검출 알고리즘의 구조를 나타낸 그림이다.

그림 3에서 p_{GPS} 와 v_{GPS} 는 단독 GPS로 추정된 위치와 속도이며 RAIM을 통해 고장이 없다고 검증된 값이다. p_{INS} 와 v_{INS} 는 INS 순수항법으로

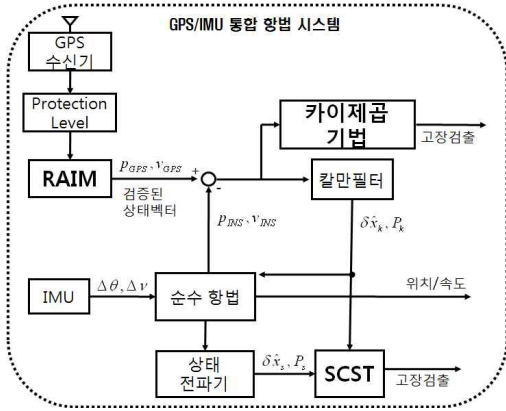


그림 3. 고장검출 흐름도

추정한 위치와 속도이고 $\Delta\theta$ 와 Δv 는 IMU의 각 가속도와 가속도 출력을 나타낸다. $\delta\hat{x}_s$ 와 P_s 는 상태 전파기로 추정된 위치오차와 공분산이고 $\delta\hat{x}_k$ 와 P_k 는 칼만필터로 추정된 위치오차와 공분산이다.

기존의 잔차를 이용하는 알고리즘과의 차이는 GLR기법을 대신하여 RAIM을 적용하였다는 것이고 기존의 해 분리 기법을 이용한 알고리즘과의 차이는 다수의 칼만필터를 사용하지 않고 GPS/INS 고장검출을 수행할 수 있다는 것이다.

본 논문에서 제안한 알고리즘은 GPS의 신뢰성을 향상시켜 전체적인 알고리즘의 무결성을 향상시키는 구조가 된다.

2.3 GPS/INS 고장검출 알고리즘의 세부 알고리즘 설명

2.3절에서는 그림 1과 그림 3에서 나타난 카이제곱 기법과 SCST 알고리즘에 대해 설명하고 그림 1에 나타난 RAIM 기법에 대해 설명한다.

2.3.1 카이제곱 시험기법

본 논문에서는 오차모델을 추정하는 방식의 확장형 칼만필터 (Extended Kalman Filter)를 사용하였고 시스템 오차모델과 측정식 오차모델은 식 (A.1)의 15차 모델을 사용하였고 부록에 자세히 기술하였다.

카이제곱 시험기법은 칼만필터 내에 잔차를 사용하여 검정 통계량을 생성한다. 그림 3에서 GPS와 INS의 위치와 속도로 구성된 6차 모델의 차분 값을 잔차로 사용하였다. 고장이 없는 경우의 측정치로 계산된 잔차는 정규분포를 따르며 공분산과 함께 이용되어 검정통계량을 산출한다. 다음 식 (1)은 검정통계량 값을 나타낸다[4].

$$l(k) = \sum_{i=k-N+1}^k r(k)^T V^{-1}(k)r(k) \quad (1)$$

여기서,

$r(k)$ = 칼만필터 잔차

$V(k)$ = 칼만필터 공분산

N = 윈도우 크기

카이제곱 시험기법은 윈도우 기법을 사용하여 고장검출을 수행하며 윈도우 크기가 카이제곱 시험기법의 성능을 결정한다[4]. 본 논문에서는 가용한 윈도우 크기를 얻기 위해 문헌을 참고하였고 윈도우 크기를 10으로 설정하였다[4].

카이제곱 시험기법에서 잔차는 GPS와 INS의 상태벡터 차분하여 사용한다. GPS/INS 통합 시스템에 고장이 발생하면 잔차가 크게 증가하여 미리 설정한 한계치를 초과하여 고장을 검출한다. 한계치는 시스템의 자유도와 오경보율로 계산된다[4]. 본 논문에서는 상태벡터 6개와 윈도우 크기 10을 곱하여 자유도 60과 FAR (Federal Aviation Regulations) 기준을 참고하여 0.005의 오경보율을 적용하였다.

2.3.2 SCST 시험기법

SCST 시험기법은 칼만필터와 상태전파기를 이용하여 검정통계량을 생성한다. Kerr는 두 타원체 중첩 시험기법을 이용한 방법을 제안했고, Da는 두 개의 상태 전파기를 일정 주기로 리셋하여 상태 전파기의 정확도를 향상시키는 방법을 제안했다[4,5,6].

본 논문에서는 칼만필터의 상태벡터와 상태전파기의 상태벡터 모두 식 (A.1)의 15차 모델을 사용하였다. SCST 시험기법도 카이제곱 시험기법과 마찬가지로 카이제곱분포를 이용하며 검정통계량은 식 (2)와 같이 계산한다.

$$l(k) = \beta^T(k)B^{-1}(k)\beta(k) \quad (2)$$

$$\beta(k) = \hat{x}_f(k) - \hat{x}_m(k) \quad (3)$$

$$B(k) = P_m(k) - P_f(k) \quad (4)$$

여기서,

$\hat{x}_f(k)$ = 칼만필터 상태벡터

$\hat{x}_m(k)$ = 상태전파기 상태벡터

$P_f(k)$ = 칼만필터 공분산

$P_m(k)$ = 상태전파기 공분산

SCST 시험기법에서 사용된 잔차는 칼만필터

의 상태벡터와 상태전파기에 의해 계산된 상태 벡터의 차분 값이다. 상태전파기의 상태벡터는 칼만필터의 상태벡터에 고장이 발생했는지 검사 할 때 사용되는 비교대상이 된다. SCST 시험기 법은 상태전파기의 정확도가 성능을 결정한다 [4]. 한계치 계산은 카이제곱 시험기법과 동일하 며 본 논문에서는 자유도가 15이고 오경보율이 0.005일 때의 값을 적용하였다.

2.3.3 GPS RAIM 분석

RAIM은 크게 최소자승기법, 패리티 공간기 법, 의사거리비교법으로 나눌 수 있다[1]. 패리티 공간기법의 RAIM은 의사거리 오차를 패리티 공간으로 투영하여 고장을 검출하는 기법이다. 검정통계량과 비교되는 한계치는 카이제곱 시험 기법과 SCST 시험기법처럼 카이제곱 분포에서 자유도와 오경보율에 의해서 결정된다. 식 (5)는 GPS 측정식을 나타낸다.

$$y = Hx + \epsilon \tag{5}$$

여기서,

H = 관측행렬

x = 사용자위치변화량

ϵ = 노이즈

관측행렬 H 를 QR분리를 통하여 GPS 측정식 을 패리티 벡터로 변환하는 패리티 변환행렬 P 를 계산할 수 있다. 패리티 변환행렬은 관측행 렬과 곱하여 영행렬이 되는 특성이 있어서 식 (5)에서 Hx 와 곱해져서 사라지고 측정식의 오차 에 비례한 값을 생성한다. 이것이 패리티 벡터 이며 다음과 같이 계산된다.

$$p = Py = P\epsilon \tag{6}$$

만약 의사거리 측정치에 고장이 발생하면 패리티 벡터의 크기는 증가한다. 이런 특성을 이용해서 패리티 벡터의 제곱을 고장 감지용 검정통계량으로 사용하며 식 (7)에서 나타낸 다.

$$TS_D = p^T p \tag{7}$$

TS_D 는 Test Statistics Detection을 의미하는 것 으로 고장 감지용 검정통계량을 의미한다.

RAIM의 한계치는 고장 감지용 검정통계량과 비교되어 고장을 판별하는데 기준이 되는 파라 메터이다. 한계치 결정은 가시위성의 개수, 오경 보율에 의해 계산되고 식 (8)은 한계치 계산식을 나타낸다.

$$\text{한계치} = \sqrt{\left(2 \times \ln \frac{1}{r}\right) \times \frac{(\sigma_{meas})^2}{(n-4)}} \tag{8}$$

여기서,

r = 오경보율

σ_{meas} = 측정치오차표준편차

n = 가시위성개수

고장을 감지한 후 패리티 벡터와 패리티 변환 행렬의 열벡터들을 비교하여 고장위성을 분별해 할 수 있다. 다음 식 (9)는 고장 분리용 검정통 계량을 나타낸다.

$$TS_i = \frac{p^T \cdot P_{:j}}{\|P_{:j}\|^2} P_{:j} = \frac{p^T \cdot P_{:j}}{\|P_{:j}\|} \tag{9}$$

식 (10)에서 $P_{:j}$ 는 패리티 변환행렬의 j 번째 열 벡터를 의미한다. 고장이 발생한 경우 패리티 벡터와 고장위성의 패리티 변환행렬의 열벡터는 방향성이 같은 특징이 있다. 이러한 특징을 이 용하여 고장위성을 감지하고 분리한다.

Protection Level은 GPS 위성배열 상태로 가 용성검사를 위해 사용되며, $SLOPE_{MAX}$ 선과 비 중심성 모수 (Non-centrality Parameter), 의사거 리 표준편차 모델로 계산된다. 비중심성 모수는 비중심 카이제곱 분포의 비중심성 정도를 나타 낸 파라미터이며 각 위성의 SLOPE 중 최댓값을 $SLOPE_{MAX}$ 로 정의한다. 다음 식 (10)은 SLOPE 계산식을 나타낸다.

$$SLOPE(i) = \frac{\sqrt{A_{1i}^2 + A_{2i}^2}}{\sqrt{S_{ii}}}, i = 1, 2, \dots, n \tag{10}$$

여기서,

$$A = (H^T H)^{-1} H^T$$

$$S = P^T P$$

식 (10)에서 A_{ii} 는 A 행렬의 첫 번째 행에 i 번 째 열의 성분을 의미하고 S_{ii} 는 S 행렬의 i 번째의 대각성분을 의미한다.

Protection Level은 현재 에폭에서 예측된 사 용자의 위치오차를 나타내고 가용성 분석의 핵 심적인 지표가 된다. 다음 식 (11)은 Protection Level 계산식을 나타낸다[2].

$$\text{Protection Level} = SLOPE_{MAX} \cdot \sqrt{\mu} \cdot \sigma_{pr} \tag{11}$$

여기서,

μ = 정규화된 p_{bias_B}

σ_{pr} = 의사거리 측정치의 표준편차

표 1. 항공용 운용기준[7]

운용기준	경계경보 제한 (수평/수직)	정확도 (수평/수직)
비정밀접근	556m/NA	220m/NA

Protection Level이 경계경보 제한을 초과하지 않으면 GPS가 가용한 상태라고 판단한다. 경계경보 제한은 운용기준에 따라 다르며 본 논문에서는 비정밀접근 (Non-Precision Approach) 기준을 적용하여 가용성을 평가하였고 표 1에 나타내었다.

가용성이 검증되면, RAIM을 수행하여 GPS 고장을 검출하고 GPS 상태벡터를 기준으로 INS의 고장을 판별한다. 만약 카이제곱 시험기법과 SCST에서 고장으로 검출되면 INS의 고장으로 판단하고 분리한다. 이런 과정을 통해서 두 시스템의 고장을 감시하고 검출을 수행하였다.

III. 고장검출 결과 분석

3.1 실험환경 및 고장모델

고장검출 성능평가를 위해서 두 가지 실험 데이터를 사용하였다. 그림 4는 건국대에서 수행된 실험궤적을 나타내고 그림 5는 서울시 마포구 소재 하늘공원에서 수행된 실험궤적을 나타낸다.

건국대 실험 데이터는 높은 DOP (Dilution of Precision)에 의해 발생하는 오차를 RAIM의 검정통계량과 Protection Level로 검출이 가능한지 알아보기 위해서 건국대 실험 데이터에는 고장모델을 인가하지 않았다. 이론적으로는 RAIM의 검정통계량은 의사거리에 발생한 고장만 감지할 수 있기 때문에 높은 DOP에 의한 고장은 검출

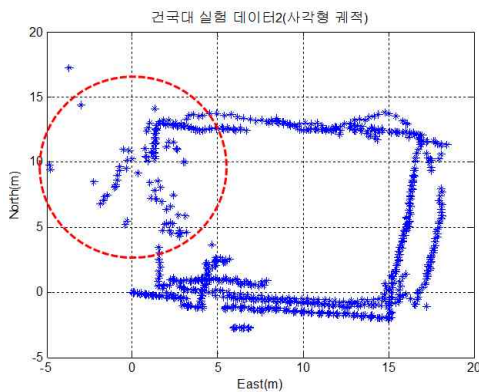


그림 4. 건국대 실험 데이터

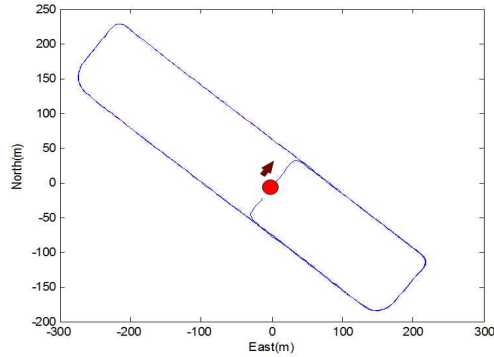


그림 5. 하늘공원 실험 데이터

할 수 없다. 하지만 Protection Level은 DOP을 고려하여 예측한 위치오차이기 때문에 경계경보 제한과 비교되어 높은 DOP에 의한 고장을 검출할 수 있고 3.2절에서 실험을 통해 확인하였다. 경계경보 제한은 검정통계량과 비교되어 고장을 판별하는 한계치처럼 Protection Level과 비교되어 가용성을 판단하는 기준이 된다. 본 논문에서는 항공기 이착륙과 같이 높은 가용성이 요구되는 실험이 아니기 때문에 비정밀접근 기준의 경계경보 제한을 사용하였다.

하늘공원 실험 데이터는 높은 DOP이 발생하는 구간이 없기 때문에 DOP에 의해 발생하는 오차와는 독립적으로 의사거리 고장에 의한 고장검출 성능을 확인할 수 있다. 그래서 하늘공원 실험 데이터에 고장모델을 적용하였고, 그 결과를 3.2절에 나타내었다.

건국대 실험 데이터는 GPS 단독으로 수행되었고 노바텔 사의 OEM-4를 사용하였다. 하늘공원 실험 데이터는 GPS와 INS로 수행되었고 GPS는 OEMV를 사용했고 INS는 허니웰 사의 HG 1900을 사용하였다.

본 논문에서 고장검출 알고리즘 성능 검증을 위해 적용한 고장모델은 다른 문헌[4,8]에서 사용한 것보다 작은 고장모델을 사용하였고 표 2는 인가된 고장모델을 나타낸 것이다.

표 2. 하늘공원 실험데이터에 적용 고장모델

고장 인가 측정치	고장형태	고장인가 시간(초)	고장 크기
GPS 의사거리	임펄스	100	100m
	바이어스	300-400	100m
	드리프트	500-600	1m/s
자이로 Roll축	바이어스	800초 이후	30deg/hr
			60deg/hr
			90deg/hr

3.2 고장검출 성능 분석

그림 6은 건국대 실험 데이터의 HDOP 과 HPL (Horizontal Protection Level)을 나타낸 그림이다. 그림 6에서 굵은 직선은 표 1에서의 수평면 경계경보 제한을 나타낸다.

HDOP이 480에폭과 790에폭에서 4이상의 높은 값을 가졌고 이 영향이 반영되어 HPL이 경계경보 제한 (Alert Limit)을 초과했음을 확인할 수 있다. 그림 4에서 점선 원 부분에서 위치오차가 발생하는 것을 볼 수 있다. 이 부분에서 GPS 안테나가 옥탑에 가려져 동쪽 위성들을 잃어버렸기 때문이었다. 그림 7은 건국대 실험 검정통계량 변화를 나타낸 그림이다.

그림 6에서 높은 DOP에 의한 영향을 HPL로 가용성판단을 수행할 수 있었다. 그림 7에서 검정통계량은 높은 DOP에 의한 오차를 검출하지 못한다는 것을 알 수 있다. RAIM의 검정통계량을 생성할 때 DOP과 관련된 관측행렬은 소거시키고 의사거리 오차만 고려하기 때문이다. 따라서 Protection Level을 통해 가용성 판단이 요구

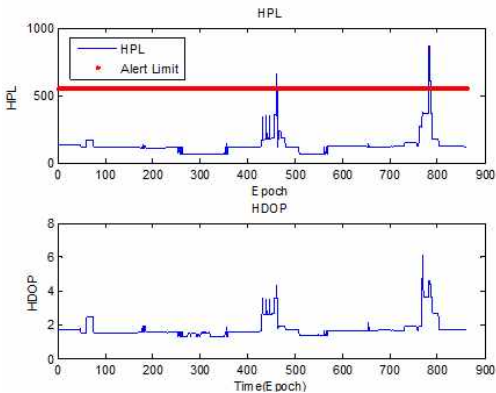


그림 6. 건국대 실험 HDOP과 HPL

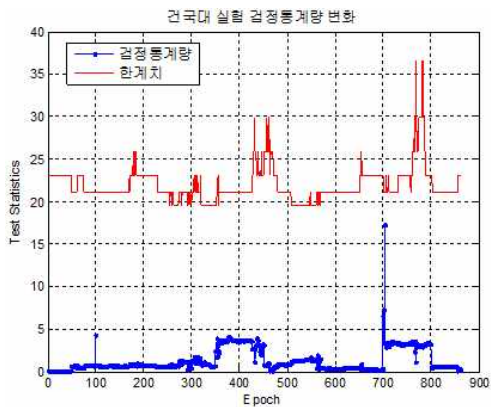


그림 7. 건국대 실험 검정통계량 변화

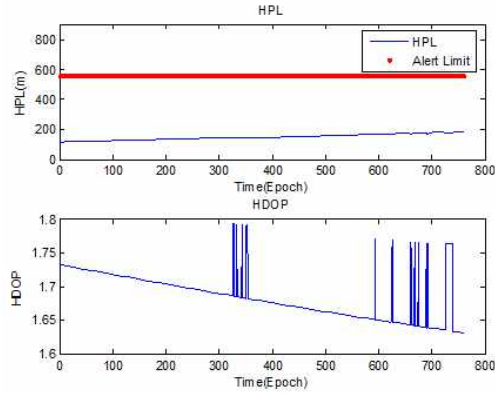


그림 8. 하늘공원 실험 HDOP과 HPL

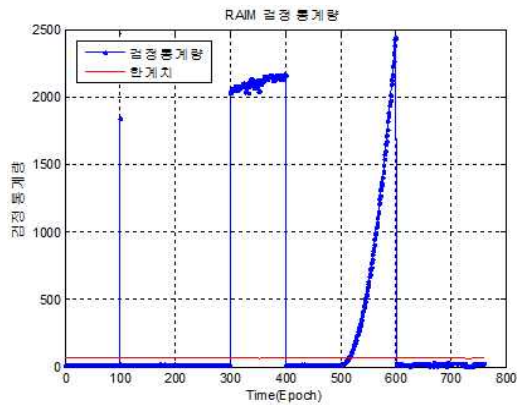


그림 9. 하늘공원 실험 검정통계량 변화

됨을 확인할 수 있었다.

그림 8은 하늘공원 실험 데이터의 HDOP과 HPL을 나타낸 그림이다.

그림 8을 통해 모든 구간에서 1.8이하의 HDOP를 가짐을 알 수 있고 HPL이 모든 구간에서 경계경보 제한을 초과하지 않았기 때문에 비정밀접근 기준에서 모든 구간에서 가용하다는 것을 확인할 수 있다.

그림 9는 하늘공원 실험에 적용한 RAIM의 검정통계량 변화를 나타낸 그림이다.

그림 9를 통해 임펄스와 바이어스와 드리프트 고장이 인가된 각각의 구간에서 고장검출이 수행되었음을 알 수 있다. 그림 10은 카이제곱 시험기법으로 INS의 고장을 검출한 결과이고 그림 11은 SCST 시험기법으로 INS의 고장을 검출한 결과를 나타낸 것이다.

그림 10은 자이로 센서가 각각 30deg/hr와 60deg/hr와 90deg/hr의 고장이 발생했을 때 카이제곱 시험기법의 검정통계량 변화를 나타낸

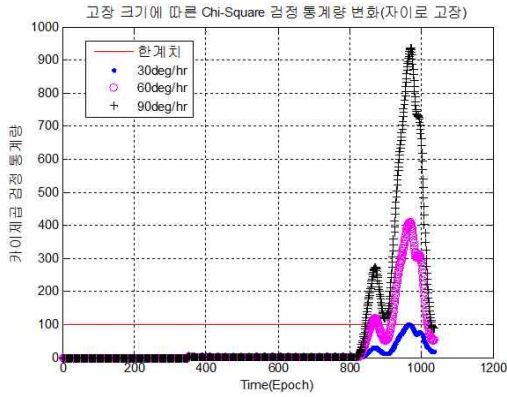


그림 10. 카이제곱 시험기법 검정통계량 변화

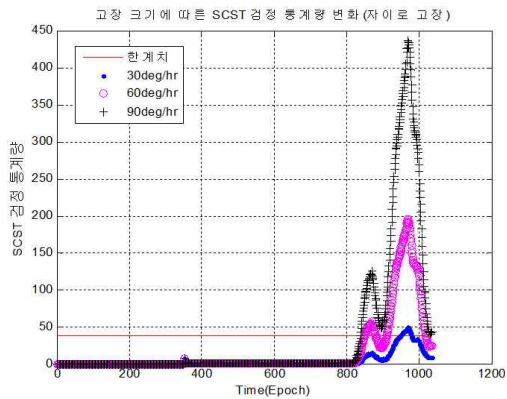


그림 11. SCST 시험기법 검정통계량 변화

것이다. 고장이 클수록 한계치를 초과하는 시간이 짧아진다는 것을 확인할 수 있다.

그림 11은 그림 10과 동일한 고장이 발생했을 때 SCST 시험기법의 검정통계량 변화를 나타낸 것이다. 그림 10의 결과와 마찬가지로 고장검출 시간은 고장의 크기에 비례하고 있음을 확인할 수 있다. 표 3은 카이제곱 시험기법과 SCST 시험기법의 고장검출 시간을 서로 비교한 표이다.

표 3에서 SCST 시험기법이 카이제곱 시험기법보다 고장검출시간이 빠르다는 것을 알 수 있다. 카이제곱 시험기법에서 설정한 윈도우 크기가 SCST 시험기법에 비해 둔감하게 설정되어

표 3. INS 고장검출 알고리즘 검출시간 비교

고장크기	카이제곱 검출시간(초)	SCST 검출시간(초)
30deg/hr	159	149
60deg/hr	59	49
90deg/hr	44	39

SCST 검출시간이 더 작게 나왔다고 분석할 수 있다. 표 3의 결과로 SCST 시험기법이 카이제곱 시험기법보다 절대적으로 우수하다고 할 수 없다. SCST 기법은 상태전파기를 사용하기 때문에 사용자의 동적특성이 강한 경우에는 상태전파기의 정확도가 떨어져서 카이제곱 시험기법을 사용한 경우보다 낮은 성능을 나타낼 수 있기 때문이다. 그래서 본 논문에서는 각각의 알고리즘의 단점을 서로 보완하기 위해서 두 가지 알고리즘을 동시에 적용하는 구조를 사용하였다.

IV. 결 론

저급 프로세서를 사용하는 무인기에 적용하기 위한 GPS/INS 고장검출 알고리즘 구조를 설계하였다. RAIM을 통해 검증된 GPS 정보를 기반으로 INS의 고장검출을 수행하는 구조로 의사거리 잔차를 이용한 알고리즘에서 잔차가 0이 되는 현상을 피할 수 있었고, 간단한 구조의 설계로 해 분리 기법의 복잡성을 피할 수 있었다.

의사거리 오차로 인해 발생한 위치오차만 감지가 가능했던 고전 RAIM의 검정통계량에 기존의 Protection Level을 이용하여 높은 DOP으로 인해 발생하는 위치오차까지 반영하도록 설계하였고 비정밀접근 기준에서 가용성 검사도 수행하였다.

설계한 알고리즘의 성능을 검증하기 위해 고장모델을 실제 데이터에 적용하였고 알고리즘 수행을 통해 고장이 인가된 부분에서 검출됨을 확인할 수 있었다. 또한 높은 DOP에 의해 발생한 위치오차는 검정통계량 변화로는 검출하지 못한다는 것을 확인할 수 있었고 Protection Level을 통해서도 검출할 수 있다는 것을 확인할 수 있었다. 고장이 없는 경우 비정밀접근 기준의 가용성을 가짐을 확인할 수 있었다.

INS 고장검출 알고리즘 성능을 검증하기 위해 고장모델을 적용하였고 고장의 크기가 클수록 검출시간이 줄어든다는 것을 확인할 수 있었다.

후 기

이 논문은 2009년도 지식경제부 근접감시용 무인항공기 시스템 개발 사업 및 교육과학기술부의 BK21 사업의 지원을 받아 수행하였으며 지원에 감사드립니다.

참고문헌

- 1) 이은성, "다중 안테나를 이용한 GPS 반송파 신호의 다중 고장 검출 및 분리", 건국대학교 박사학위 논문, 2005.
- 2) R. Grover Brown, Gerald Y. Chin "GPS RAIM: Calculation of Threshold and Protection Radius Using Chi-Square Methods-A Geometric Approach", Navigation: Journal of The Institute of Navigation, Vol. 39, No 3, Fall 1992.
- 3) 유창선 "GPS 다중고장에 대한 무결성 모니터링 기법", 충남대학교 박사학위 논문, 2003.
- 4) 김성태 "GPS/INS 통합항법시스템의 고장 검출 및 분리", 충남대학교 석사학위 논문, 2000.
- 5) Ren Da, "Failure Detection of Dynamical Systems with the State Chi-Square Test", Journal of Guidance, Control, and Dynamics. Vol 17, No 2, March - April 1994.
- 6) 김종원, "GPS 반송파 위상과 INS를 상호 보완적으로 이용한 자세 추정", 서울대학교 박사학위 논문, 1998.
- 7) International Civil Aviation Organization Eastern And Southern African Office, "Summary of the Discussions and Conclusions of the Second Meeting of the AFI Gns Implementation Task Force", Johannesburg, 22 - 23 JUNE 2004.
- 8) 김광훈, "내고장성 항법 시스템을 위한 적응 필터 설계", 서울대학교 대학원 박사학위 논문, 2006.
- 9) M. BRENNER, "Integrated GPS/inertial detection availability", Journal of The Institute of Navigation, Sept. 1995.

부 록

본 논문에서는 확장형 칼만필터를 이용하여 고장검출을 수행하였고 부록에서는 사용한 시스템 오차 모델과 추정식 오차모델에 대해 설명하였다. 식 (A.1)은 칼만필터의 추정식 오차모델을 나타낸다.

$$z(t) = H_{LC} \delta x_{INS}(t) + v_{LC}(t) = \begin{bmatrix} p_{INS} \\ v_{INS} \end{bmatrix} - \begin{bmatrix} p_{GPS} \\ v_{GPS} \end{bmatrix} \quad (A.1)$$

$$v_{LC} \sim N(0, R_{LC})$$

여기서,

$$\delta x_{INS} = [\delta Llh \ \delta V_{NED} \ \epsilon_{NED} \ \nabla_{XYZ} \ \eta_{XYZ}]^T$$

$$w_{LC} = [w_{ax} \ w_{ay} \ w_{az} \ w_{gx} \ w_{gy} \ w_{gz}]^T$$

$$H_{LC} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0_{1 \times 9} \\ 0 & 1 & 0 & 0 & 0 & 0 & 0_{1 \times 9} \\ 0 & 0 & 1 & 0 & 0 & 0 & 0_{1 \times 9} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0_{1 \times 9} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0_{1 \times 9} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0_{1 \times 9} \end{bmatrix}$$

식 (A.1)에서 상태벡터 δx_{INS} 의 각 성분의 의미는 다음과 같다.

$$\delta Llh = \text{위치오차}$$

$$\delta V_{NED} = \text{속도오차}$$

$$\epsilon_{NED} = \text{자세오차}$$

$$\nabla_{XYZ} = \text{가속도계 바이어스}$$

$$\eta_{XYZ} = \text{자이로 바이어스}$$

식 (A.1)에서 잡음 v_{LC} 는 평균이 0이고 공분산이 R_{LC} 인 정규분포를 따르며 잡음의 공분산이며 H_{LC} 에서 $0_{1 \times 9}$ 는 1×9 크기의 영행렬을 의미한다. 상태벡터의 성분의 밑첨자 중 a는 가속도계를 나타내고 g는 자이로를 나타내며 XYZ는 각 축 방향을 나타낸다.

식 (A.2)는 시스템 오차 모델을 나타낸다.

$$\delta \dot{x}_{INS}(t) = F_{INS}(t) \delta x_{INS}(t) + G_{LC}(t) w_{LC}(t) \quad (A.2)$$

$$w_{LC} \sim N(0, Q_{LC})$$

식 (A.2)에서 잡음 v_{LC} 는 평균이 0이고 공분산이 Q_{LC} 인 정규분포를 따른다.

여기서,

$$G_{LC}(t) = \begin{bmatrix} O_{3 \times 3} & O_{3 \times 3} \\ C_b^n & O_{3 \times 3} \\ O_{3 \times 3} & -C_b^n \\ O_{3 \times 3} & O_{3 \times 3} \\ O_{3 \times 3} & O_{3 \times 3} \end{bmatrix}$$

$$F_{INS} = \begin{bmatrix} F_{11} & F_{12} & O_{3 \times 3} & O_{3 \times 3} & O_{3 \times 3} \\ F_{21} & F_{22} & F_{23} & F_{24} & O_{3 \times 3} \\ F_{31} & F_{32} & F_{33} & O_{3 \times 3} & F_{35} \\ O_{3 \times 3} & O_{3 \times 3} & O_{3 \times 3} & O_{3 \times 3} & O_{3 \times 3} \\ O_{3 \times 3} & O_{3 \times 3} & O_{3 \times 3} & O_{3 \times 3} & O_{3 \times 3} \end{bmatrix}$$

$$F_{11} = \begin{bmatrix} \frac{R_{mm}\rho_E}{R_m+h} & 0 & \frac{\rho_E}{R_m+h} \\ \frac{\rho_N}{\cos L} \left(\tan L - \frac{R_u}{R_i+h} \right) & 0 & \frac{\rho_N \sec L}{R_i+h} \\ 0 & 0 & 0 \end{bmatrix}$$

$$F_{12} = \begin{bmatrix} \frac{1}{R_m+h} & 0 & 0 \\ 0 & \frac{\sec L}{R_i+h} & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$F_{31} = \begin{bmatrix} \Omega_D - \frac{\rho_N R_u}{R_i+h} & 0 & \frac{-\rho_N}{R_i+h} \\ -\rho_E \frac{R_{mm}}{R_m+h} & 0 & \frac{-\rho_E}{R_m+h} \\ -\Omega_N - \rho_N \sec^2 L - \frac{\rho_D R_u}{R_i+h} & 0 & \frac{-\rho_D}{R_i+h} \end{bmatrix}$$

$$F_{32} = \begin{bmatrix} 0 & \frac{1}{R_i+h} & 0 \\ -\frac{1}{R_m+h} & 0 & 0 \\ 0 & -\frac{\tan L}{R_i+h} & 0 \end{bmatrix}$$

$$F_{21} = \begin{bmatrix} \frac{R_{mm}\rho_E}{R_m+h} v_D - (\rho_N \sec^2 L + 2\Omega_N) - \rho_N \rho_D R_u & 0 & \frac{\rho_E}{R_m+h} v_D - \rho_N \rho_D \\ (2\Omega_N + \rho_N \sec^2 L + \rho_D \frac{R_u}{R_i+h}) v_N - (\frac{\rho_N R_u}{R_i+h} - 2\Omega_D) & 0 & \frac{\rho_D}{R_i+h} v_N - \frac{\rho_N}{R_i+h} v_D \\ \rho_N^2 R_u + \rho_E^2 R_{mm} - 2\Omega_D v_E & 0 & \rho_N^2 + \rho_E^2 \end{bmatrix}$$

$$F_{24} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 + q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_0 q_2 + q_1 q_3) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_0 q_1 + q_2 q_3) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}$$

$$F_{33} = \begin{bmatrix} 0 & \Omega_D + \rho_D & -\rho_E \\ -\Omega_D - \rho_D & 0 & \Omega_N + \rho_N \\ \rho_E & -\Omega_N - \rho_N & 0 \end{bmatrix}$$

$$F_{35} = -C_b^n$$

$$F_{22} = \begin{bmatrix} \frac{v_D}{R_m+h} & 2\rho_D + 2\Omega_D & -\rho_E \\ -2\Omega_D - \rho_D & \frac{v_N \tan L + v_D}{R_i+h} & 2\Omega_N + \rho_N \\ 2\rho_E & -2\Omega_N - \rho_N & 0 \end{bmatrix}$$

$$F_{23} = \begin{bmatrix} 0 & -f_D & f_E \\ f_D & 0 & -f_N \\ -f_E & f_N & 0 \end{bmatrix}$$

여기서 시스템 행렬 F를 구성하는 성분들은 다음과 같이 정의된다.

- R_m = 지구 곡률의 종방향 반지름
- R_i = 지구 곡률의 횡방향 반지름
- R_{mm} = R_m 의 미분값
- R_u = R_i 의 미분값
- ρ_{NED} = Craft Rate

- Llh = 경도, 위도, 고도
- v_{NED} = 속도(NED좌표계)
- Ω_{NED} = 지구 회전을
- q = 쿼터니언