

비행안전분석을 위한 낙하분산영역 예측 기법에 대한 연구

최규성*, 심형석**, 고정환***, 정의승****

A Study on the Prediction Technique of Impact Dispersion Area for Flight Safety Analysis

Kyu-Sung Choi*, Hyung-Seok Sim**, Jeong-Hwan Ko***, Eui-Seung Chung****

Abstract

Flight safety analyses concerned with Launch Vehicle are performed to measure the risk to the people, ship and aircraft using impact point and impact dispersion area of debris generated by on-trajectory failures and malfunction turns. Predictions of impact point and impact dispersion area are essential for launch vehicle's flight safety analysis. Usually, impact dispersion area can be estimated in using Monte-Carlo simulation. However, Monte-Carlo method requires more several hundreds of iterative calculations which requires quite some time to produce impact dispersion area. Herein, we check the possibility of applying JU(Julier Uhlmann) transformation and Taguchi method instead of Monte-Carlo method and we propose a best method in terms of computational time to produce impact dispersion area by comparing the results of the three methods.

초 록

우주발사체의 비행안전분석은 정상 및 비정상 비행으로 인해 발생하는 파편의 낙하점 및 낙하분산영역을 예측하여 인명, 선박 그리고 항공기에 미치는 영향을 분석하게 된다. 낙하점 및 낙하분산영역 예측은 우주발사체의 비행안전분석에 필수 요소이다. 특히, 낙하분산영역은 몬테카를로 시뮬레이션을 적용하여 예측될 수 있다. 이럴 경우, 수백회 이상의 반복 계산이 요구되는 몬테카를로 방법은 낙하분산영역을 산출하는데 많은 시간이 소요된다. 본 논문에서는 몬테카를로 시뮬레이션을 대체할 수 있는 방안으로 JU 변환과 다구치 방법을 적용해보고, 세가지 방안의 결과를 비교하여 낙하분산영역 계산을 위해 적합한 방법을 제시한다.

키워드 : 비행안전분석(Flight Safety Analysis), 몬테카를로 시뮬레이션(Monte-Carlo Simulation), 다구치 방법(Taguchi Method), 무향 칼만필터(Unscented Kalman Filter), 낙하점(Impac Point), 낙하분산영역(Impact Dispersion Area), 발사체(Launch Vehicle)

접수일(2014년 9월 5일), 수정일(1차 : 10월 20일), 게재 확정일(2014년 11월 1일)

* 발사체품질보증팀/kschoi@kari.re.kr

** 발사체품질보증팀/aeroshs@kari.re.kr

*** 발사체품질보증팀/jko@kari.re.kr

**** 발사체임무보증실/ces@kari.re.kr

1. 서 론

우주발사체의 비행안전분석은 정상 및 비정상 비행(조기연소종료, 비행중단시스템 작동, 폭발 및 파괴등) 으로 인해 발생하는 낙하물 또는 파편들의 낙하점 및 낙하분산영역을 계산하여 인명, 선박 그리고 항공기 등에 미치는 영향을 분석하는 과정이다. 이중 낙하분산영역 산출은 수치적으로 일련의 난수를 반복적으로 발생시키는 몬테카를로 시뮬레이션을 주로 사용하고 있다. 몬테카를로 시뮬레이션 결과가 확률적으로 의미 있기 위해서는 다량의 시뮬레이션을 수행해야 하고, 결과적으로는 낙하분산영역계산의 소요시간에 많은 영향을 미치고 있어 반복 계산되는 몬테카를로 시뮬레이션의 시행횟수를 줄이는 방안이 요구되고 있다. 이에 따라, 몬테카를로 시뮬레이션을 대체할 수 있는 방안으로 본 논문에서는 JU(Julier-Uhlmann) 변환 기법과 다구치 방법을 적용해 보았다.

JU 변환 기법은 Unscented 칼만 필터의 핵심 알고리즘으로서 비선형 함수를 변환하는 과정에서 자코비안(Jacobian)을 사용하여 1차원 선형화하거나 무작위로 많은 샘플을 사용하는 몬테카를로 시뮬레이션 방법과 달리 샘플간의 가중치를 설정하여 적은수의 샘플로 가우시안 랜덤 변수를 통계적으로 계산한다. 위의 기법은 비선형 함수를 선형화하는 과정에서 고차항을 고려하지 않음으로서 나타나는 문제점 및 다량의 난수 발생을 통한 몬테카를로 시뮬레이션 방법으로 인한 계산 시간이 오래 걸리는 문제점을 해결할 수 있는 방안으로 알려져 있다. 한편 다구치 방법은 제품을 설계할 때, 그 제품의 사용환경을 고려하여 어떠한 조건의 환경에서도 제품이 목적하는 성능을 일정하게 발휘할 수 있는 즉 가장 작은 편차로 성능을 나타낼 수 있도록 강건하게 설계하는 기법이다. 이 과정에서 최적화된 실험 계획법을 선택하여 적용하게 된다. 실험계획법은 완전요인설계, 부분요인설계, 직교배열표로 구성되어 있지만 다구치 방법은 직교배열표를 사용하여 실험횟수를 줄여 시간을 최소화 할 수 있다.

본 논문에서는 JU 변환 기법과 다구치 방법을 기술하고 각 방법의 결과와 몬테카를로 시뮬레이션의 결과를 비교 분석하여 낙하분산영역 계산을 위해 적합한 방법을 제시하였다.

2. JU(Julier Uhlmann) 변환 기법

2.1 JU 변환

JU(Julier Uhlmann) 변환 기법은 비선형 함수를 변환하는 과정에서 Jacobian을 사용하여 1차원 선형화하거나 무작위로 많은 샘플을 사용하는 몬테카를로 시뮬레이션 방법과 달리 샘플간의 가중치를 설정하여 적은수의 샘플로 가우시안 랜덤 변수를 통계적으로 계산한다. JU 변환은 비선형 함수 변환에서 랜덤 변수의 통계를 계산하는 방법으로, 근사적인 확률 분포의 원리를 적용하고 있다. L 차원의 랜덤 변수 x 를 일반적인 비선형 함수에 의해 전파된다고 할 때 아래와 같이 정의할 수 있다.

$$y = f(x), \quad x \in R^L \quad (1)$$

여기서 임의의 x 평균은 \bar{x} 이고, x 의 공분산을 P_x 라 가정한다. JU 변환을 이용하여 y 의 통계치(평균, 공분산)를 계산하기 위한 절차를 수식으로 표현하면 아래와 같다[1].

$$\begin{aligned} \chi_0 &= \bar{x} & W_0 &= \kappa / (L + \kappa) & i &= 0 \\ \chi_i &= \bar{x} + (\sqrt{(L + \kappa)P_x})_i & W_i &= 1 / \{2(L + \kappa)\} & i &= 1, \dots, L \\ \chi_i &= \bar{x} - (\sqrt{(L + \kappa)P_x})_i & W_i &= 1 / \{2(L + \kappa)\} & i &= L + 1, \dots, 2L \end{aligned} \quad (2)$$

여기서 κ 는 크기조정 매개변수(scaled parameter)이고, $(\sqrt{(L + \kappa)P_x})_i$ 는 $(L + \kappa)P_x$ 의 행렬 제곱근 값의 i 번째 행 또는 열이다. W_i 는 i 번째 시그마 포인트의 가중치이다. 콜레스키 분해(Cholesky decomposition)는 행렬 제곱근 값을 수치적으로 계산하는데 효과적이므로 JU 변환중 행렬 제곱근에 사용된다. 각각의 시그마 포인트는 비선형 함수를 통해 아래 식으로 전개된다[1].

$$\psi_i = f(\chi_i) \quad i = 0, \dots, 2L \quad (3)$$

그리고 예측된 y 의 평균과 공분산은 아래와 같이 계산된다.

$$\begin{aligned} \bar{y} &= \sum_{i=0}^{2L} W_i \psi_i \\ P_y &= \sum_{i=0}^{2L} W_i (\psi_i - \bar{y})(\psi_i - \bar{y})^T \end{aligned} \quad (4)$$

2.2 Scaled JU 변환

위 JU 변환에서 사용된 시그마 포인트 결정 방식은 상태공간 L 차원의 증가로 인해, 모든 시그마 포인트의 반경이 증가하게 된다. 비선형의 경향이 큰 함수일 경우, 이 문제의 해법을 찾는 것은 매우 어렵게 된다. 이러한 문제를 해결하기 위해 κ 값의 결정으로 시그마 포인트의 사전 분포 \bar{x} 로부터 조정을 해야 한다. i 번째 시그마 포인트와 \bar{x} 사이의 거리, $|\chi_i - \bar{x}|$ 는 $\sqrt{(L+\kappa)}$ 에 비례한다. $\kappa=0$ 일 경우 거리는 \sqrt{L} 에 비례하고, $\kappa>0$ 일 경우는 시그마 포인트가 \bar{x} 로부터 멀어지고, $\kappa<0$ 일 경우는 \bar{x} 로부터 가까워진다. 특별한 경우로 $\kappa=3-L$ 로 설정했다면, x 의 차원에서 L 에 의한 영향이 없어지게 될 것이다. 그러나 $\kappa=3-L<0$ 이고, $W_0<0$ 이면 계산된 공분산이 음반정치(non-positive semidefinite) 상태가 된다. 위의 문제를 해결하기 위해 크기 조정 JU 변환 방법이 제안되었다[1].

크기 조정 JU 변환은 시그마 포인트를 구하는 과정을 다음과 같은 식으로 교체하였다.

$$\chi'_i = \chi_0 + \alpha(\chi_i - \chi_0) \quad i = 0, \dots, 2L \quad (5)$$

여기서 α 는 양의 크기조정 매개변수로서 고차항의 영향을 최소화하기 위해 임의의 작은 값 ($0 < \alpha < 1$)을 사용한다. 이 형식은 공분산을 구할 때 음반정치가 될 수 있는 공분산을 제외시

켜, 시그마 포인트의 크기를 조절 할 수 있다. 보조 랜덤 변수중의 하나인 α 는 비선형 모델 방정식(1)에 적용시키면 아래와 같은 식으로 전과 된다.

$$z = \dot{g}(x) = \frac{g[\bar{x} + \alpha(x - \bar{x})] - g(\bar{x})}{\alpha^2} + g(\bar{x}) \quad (6)$$

위 식(6)을 이용하여 계산된 \bar{z} 와 P_z 의 테일러 급수 전개식은 식(4)에서 구한 \bar{y} 와 P_y 으로 급수 전개 시킨 식과 2차항까지 정확하게 일치하고, 고차항 이상은 α 값을 조정하여 정확성을 향상시킬 수 있다. 따라서 크기조정 JU 변환은 JU 변환에 보조 랜덤 변수를 적용하게 된다. 앞서 식(2)을 이용하여 구한 시그마 포인트의 집합 $S = \{W, \chi; i = 0, \dots, 2L\}$ 은 아래의 식을 적용하여 새로운 시그마 포인트 집합 $S' = \{W', \chi'; i = 0, \dots, 2L\}$ 으로 변환된다.

$$\begin{aligned} \chi'_i &= \chi_0 + \alpha(\chi_i - \chi_0) \\ W'_i &= \begin{cases} \frac{W_0}{\alpha^2} + \left(1 - \frac{1}{\alpha^2}\right) & i = 0 \\ \frac{W_i}{\alpha^2} & i = 1, \dots, 2L \end{cases} \end{aligned} \quad (7)$$

α 는 새로운 시그마 포인트의 크기조정 매개변수이며, 아래의 식을 이용하여 시그마 포인트를 구하는 식을 단순화 한다.

$$\lambda = \alpha^2(L + \kappa) - L \quad (8)$$

시그마 포인트를 구하는 식은 다음과 같다.

$$\begin{aligned} \chi_0 &= \bar{x} \\ \chi_i &= \bar{x} + \left(\sqrt{(L+\lambda)P_x}\right)_i \quad i = 1, \dots, L \\ \chi_i &= \bar{x} - \left(\sqrt{(L+\lambda)P_x}\right)_i \quad i = L+1, \dots, 2L \\ W_0^{(m)} &= \lambda / (L + \lambda) \\ W_0^{(c)} &= \lambda / (L + \lambda) + (1 - \alpha^2 + \beta) \\ W_i^{(m)} &= W_i^{(c)} = 1 / \{2(L + \lambda)\} \quad i = 1, \dots, 2L \end{aligned} \quad (9)$$

위식을 이용하여 $2L+1$ 개의 시그마 포인트 집합 $S = \{W, \chi', i = 0, \dots, 2L\}$ 를 구하고 비선형 변환을 통해 각각의 시그마 포인트를 다음 단계로 전파하여 평균 \bar{y} 와 공분산 P_y 를 산출한다.

3. 다구치 방법

다구치 방법은 제품을 설계시 그 제품의 사용환경을 고려하여 어떠한 조건의 환경에서도 제품이 목적하는 성능을 일정하게 발휘할 수 있는 즉 가장 작은 편차로 성능을 나타낼 수 있도록 강건하게 설계하는 기법이다[9]. 이 과정에서 최적화된 실험 계획법을 선택하여 적용하게 된다. 실험계획법은 완전요인설계, 부분요인설계, 직교배열표로 구성되어 있지만 다구치 방법은 직교배열표를 사용하여 실험횟수를 줄여 시간 및 경비를 최소화하게 된다[7].

3.1 실험계획법

- 완전요인 설계(Full factorial design)

요인 실험은 통상 k^n 으로 나타낸다. 이는 인자의 수가 n 개이고 각 인자가 실험을 할 때 취하는 값(수준)이 k 라는 것을 의미한다. 일반적으로 수준이 2 또는 3인 경우가 가장 많이 사용된다. 완전요인배치는 모든 인자간 수준의 조합에서 실험이 이루어진다. 그러므로 k^n 번의 실험이 실시되어야 한다. 예로, 아래 그림은 인자수가 2개이고 3수준의 인자 배치도를 보여주고 있다.

No.	A	B	C
1	1	1	1
2	1	1	2
3	1	2	1
4	1	2	2
5	2	1	1
6	2	1	2
7	2	2	1
8	2	2	2

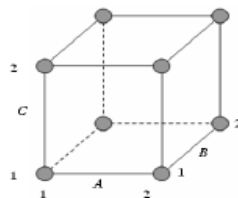


그림 1. 완전요인 배치도

- 부분요인설계
완전요인설계는 인자수가 많아지면 실험횟수

가 너무 많아지므로 의미가 적은 고차의 상관관계 사이의 정보를 희생시키고 중요한 주효과와 특정한 2인자와의 교호작용(상호작용)만 추출할 수 있도록 실험횟수를 줄인 방법이다. 예로, 그림 2는 인자수가 2개이고 3수준의 인자 배치도를 보여주고 있지만 배치도가 불균형한 형상을 하고 있다.

No.	A	B	C
1	1	1	1
2	1	1	2
3	1	2	2
4	2	2	2

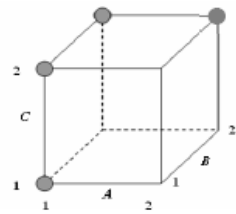


그림 2. 부분요인 배치도

- 직교배열표

다구치(Taguchi)가 개발한 방법으로 부분요인설계의 한 방법이다. 아래 3수준의 인자 배치도를 보면 각 3차원면에 균일하게 배치하고 있음을 알 수 있다.

No.	A	B	C
1	1	1	1
2	1	2	2
3	2	1	2
4	2	2	1

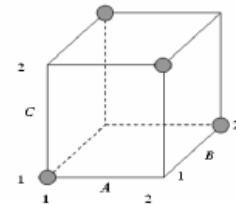


그림 3. 직교배열 배치도

3.2 직교배열법

실험계획법의 일종인 직교배열표를 이용하여 반복횟수를 최소로 하는 방법이다. 직교배열표는 부분요인설계에 사용되는 매트릭스 형태로 구성되어 있으며 변수의 영향을 효율적으로 결정할 수 있다. 다구치 방법은 매트릭스 실험을 위한 기준 직교배열을 사용하고 있다. 기본적으로 인자에 대한 수준을 2 또는 3수준으로 정의하고 있다.

- 직교표의 효용
특성치에 영향을 미치리라 여겨지는 설계변

수(제어인자)의 수가 많고 제한된 자원(실험시료, 시간, 비용등)으로 인자 수준의 모든 조합에 대해 실험을 수행하기 어려울 때 적은 실험 횟수로써 주효과와 선택된 교호 작용에 대한 정보를 얻기 위해 사용된다.

- 직교표의 표시법

직교표는 $L_N(b^k)$ 로 표기한다. 여기서, L : 라틴 방격법(Latin Squares), N : 행의 수(반복이 없는 경우에는 총실험 횟수), k : 열의 수, b : 열의 수준수이다.

- 직교표와 완전요인배치의 실험횟수 비교

실험방법에 따른 인자의 수준별 실험횟수를 비교하면 아래와 같다.

표 1. 실험방법과 수준별 실험횟수

2수준			3수준		
인자수	직교표	Factorial	인자수	직교표	Factorial
3	4	8	4	9	81
7	8	128	13	27	1594323
11	12	2048	40	81	∞
15	16	32768			
19	20	524288			

위와같이, 완전인자 배치의 경우 인자수가 증가할 경우 기하 급수적으로 증가하게 되어 실제에 적용하는 것은 불가능하다. 이에 대한 대안으로 직교표를 활용하면 만족스러운 결과를 얻을 수 있다.

3.3 분산분석

특성치의 총변동, 즉 총제공합을 요인 효과에 의한 변동들과 오차에 의한 변동으로 나누어 요인의 영향에 대한 유의성을 판정하는 통계적 방법이다. 요인효과는 인자가 주된 효과를 나타내는 주효과와 기술적으로 보아서 있을 것 같은 2개 이상의 인자가 상호작용을 하는 교호작용으로 구분된다[8].

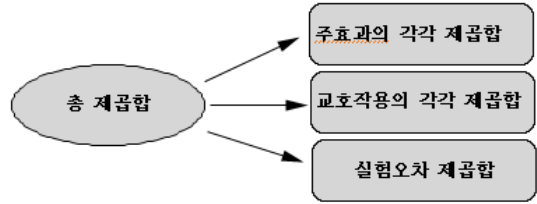


그림 4. 분산분석의 제공합

위의 제공합을 실험과 관련된 인자마다의 제공합으로 분해하여, 오차에 비해 특히 영향을 주는 요인이 무엇인가를 찾아내는 분석 방법이다. 분산분석을 통해 몬테카를로 시뮬레이션과 비교해서 적은 실험수로 특성치의 분산 분포 결과 및 각 제어인자에 대한 민감도를 동시에 계산할 수 있다는 장점을 가지고 있다.

4. 낙하분산영역에 대한 JU 변환, 다구치와 몬테카를로 시뮬레이션의 결과

비행체의 비정상 상태에서 발생한 파편의 낙하점 추정은 공력이 작용하지 않는다는 가정하에 계산하게 된다. 공력 고려시 위치 및 속도의 편차외에 비행체의 질량, 공력계수, 바람등의 불확실성 영향을 고려하게 되면 변수가 증가하여 계산량이 기하급수적으로 늘어나게 된다. 보다 간단히 셋방식의 결과 비교를 위해, 본 논문에서는 공력을 포함하지 않은 낙하점 모델을 대상으로 초기값인 위치 및 속도의 변수를 고려해 보았다. 현재 사용되고 있는 비행안전분석 소프트웨어에 변수의 예측치를 정확하게 예측할 수 없어 확정 모델(deterministic model)이 아닌 통계모델(stochastic model)을 사용하고 있는데, 수치적으로 일련의 난수를 반복적으로 발생해서 시뮬레이션을 하여 결과값을 얻고 분산 분석을 통해 평균 및 표준편차를 산출하게 된다.

그러나 많은 반복계산이 필요하며 변수가 많아짐에 따라 시뮬레이션 수행횟수도 비례하여 증가하게 된다. 많은 실험수를 줄이기 위해, 품질공학에 적용되었던 다구치 방법과 JU 변환 기법을

적용하여 결과를 비교하였다.

4.1 낙하점 계산 모델 및 초기 변수

진공낙하점 계산 모델은 비행중 추력이 없고 진공상태에서 운동하는 케플러 운동 방정식을 적용하여 속도와 위치를 초기값으로 낙하점을 예측한다. 비행체가 대기권 밖에서의 움직임을 케플러 운동에 적용하여 나타내면 다음과 같은 그림으로 표현된다.

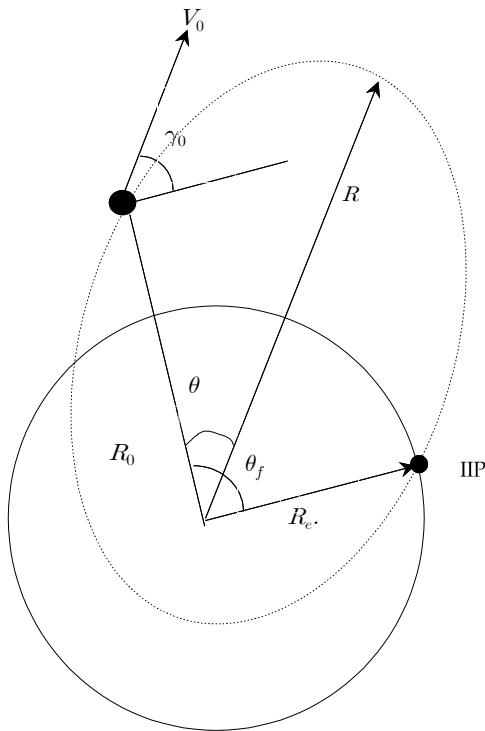


그림 5. 케플러 궤도

- , R_0 : 비행체가 현재 위치에서 지구 중심과의 거리,
- V_0, γ_0 : 비행체의 초기 속도 및 비행 경로각,
- θ_f : 비행체가 낙하점에 도달시 잔여비행각,
- R_e : 적도지점에서 지구 중심과의 거리,
- IIP : 공력을 고려하지 않은 순간낙하점

나로호(KSLV-I) 정상궤적상에서 비행시간 1초

단위로 비행실패가 발생한다고 가정하여 진공낙하점 계산 모델을 적용하여 그 시점에서의 낙하점 및 분산영역을 산출할 수 있다. 낙하점 및 낙하영역을 예측하는 주요 변수는 비행체의 위치, 속도, 질량, 공력등이 있다. 그러나 위에서 언급된 두 방법을 비교하기 위해서 KSLV-I의 nominal 궤적 중 비행시간 200초 시점의 상태벡터(위치,속도)를 불확실성 변수로 설정하였다. 상태벡터의 좌표계는 지구고정좌표계를 기준으로 하고 있으며 초기 위치, 속도 및 각 성분의 편차는 아래와 같이 임의로 가정하였다.

표 2. t=200초의 상태벡터의 평균 및 분산

변수	평균(μ)	분산(σ^2)
위치(X)	-3373.4(km)	10,000(m)
위치(Y)	4343.0(km)	10,000(m)
위치(Z)	3516.2(km)	10,000(m)
속도(Vx)	-2371.3(m/s)	200(m/s)
속도(Vy)	2164.1(m/s)	200(m/s)
속도(Vz)	-2239.4(m/s)	200(m/s)

4.2 JUT, 다구치 및 MCS 결과 비교

일반적으로 몬테카를로 시뮬레이션(MCS)을 적용하여 수치적으로 일련의 난수를 반복적으로 발생해서 시뮬레이션을 하여 결과값을 얻고 분산 분석을 통해 평균 및 분산영역을 산출하게 된다.

몬테카를로 시뮬레이션은 통계모델(stochastic model)을 사용하고 있어 많은 반복 계산이 필요하며 초기 불확실성 변수가 많아짐에 따라 시뮬레이션 수행횟수도 비례하여 증가하는 경향이 있다. 몬테카를로 시뮬레이션과 달리 JUT와 다구치 방법의 실험횟수는 상대적으로 적으며 초기 불확실성 인자수에 따른 JUT와 다구치의 실험횟수를 비교하면 아래와 같다.

표 3. JUT와 다구치 방법의 인자에 따른
최소실험횟수 비교

인자수	JUT	Taguchi (2수준)	Taguchi (3수준)
1	3	4	9
2	5	4	9
3	7	4	9
4	9	8	9
5	11	8	27
6	13	8	27
7	15	8	27
8	17	12	27
9	19	12	27
10	21	12	27

불확실성 인자수 6으로 하는 JUT 및 다구치 방법과 MCS의 실험수행횟수를 100회, 1,000회 및 1,000,000회로 설정한 결과를 비교하였다.

Case 1 : MCS(100회)

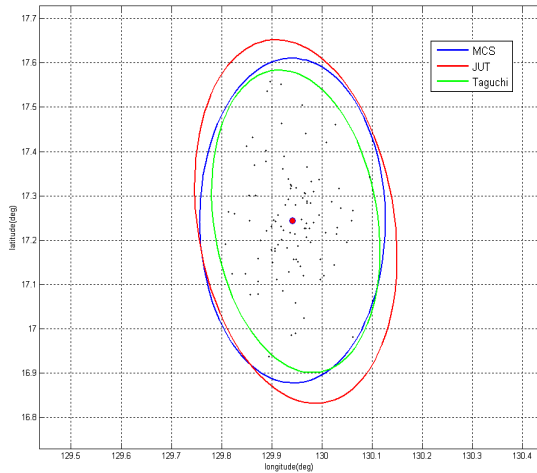


그림 6. MCS(100회), JUT 및 Taguchi의 분산영역
비교

Case 2 : MCS(1,000회)

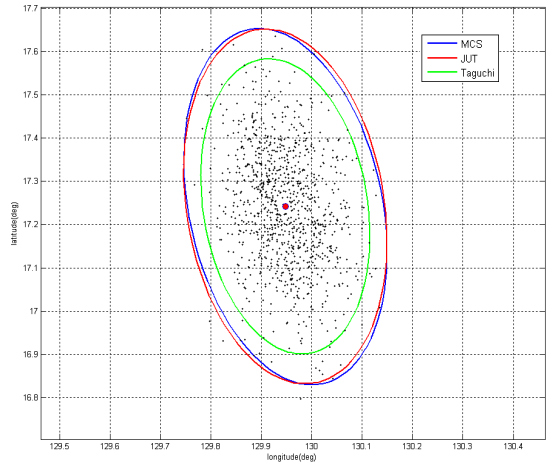


그림 7. MCS(1,000회), JUT 및 Taguchi의 분산영역
비교

Case 3 : MCS(1,000,000회)

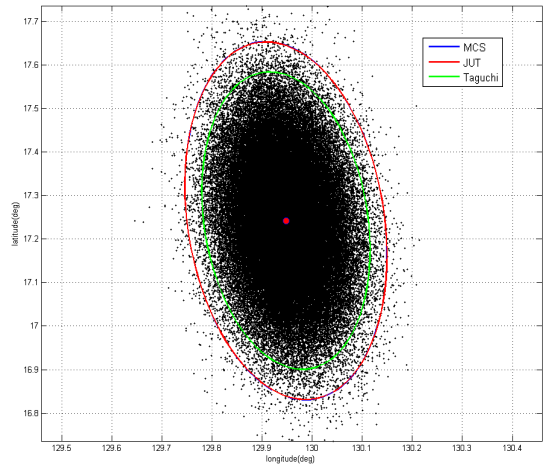


그림 8. MCS(1,000,000회), JUT 및 Taguchi의
분산영역 비교

MCS, JUT 및 다구치 방법으로 수행한 결과의 평균낙하점은 거의 동일하며 낙하분산영역을 MCS(1,000,000회)를 기준으로 낙하분산영역(장반경, 단반경, 방위각: 북쪽을 기준으로 장반경과 이루는 시계방향각)에 대해 각각의 실험방법에

대한 오차를 비교하면 아래와 같다.

표 4. MCS, JUT 및 다구치 방법에 따른 낙하분산영역

	낙하분산영역		
	장반경 (km)	단반경 (km)	방위각 (deg)
MCS (1,000,000회)	45.94 (-)	20.87 (-)	173.4 (-)
MCS(100회)	40.73 (11.34%)	19.65 (5.85%)	179.79 (3.68%)
MCS (1,000회)	46.24 (0.65%)	20.47 (1.92%)	171.40 (1.15%)
JUT (13회)	45.85 (0.2%)	20.88 (0.05%)	173.39 (0.005%)
Taguchi (27회)	38.15 (16.96%)	17.37 (16.77%)	173.39 (0.005%)

MCS(1,000,000회)를 기준으로 할 때 MCS(100회)의 장단반경 오차는 각각 11.34%, 5.85%, MCS(1,000회)의 장단반경 오차는 각각 0.65%, 1.92%, JUT의 장단반경 오차는 각각 0.2%, 0.05%이며 Taguchi의 장단반경 오차는 각각 16.96%, 16.77%이다.

5. 결 론

본 논문에서는 파편의 낙하분산영역 계산을 수행하는 방법들을 살펴보았다. MCS(1,000,000회)의 낙하분산영역을 기준으로 할 때 MCS의 실험횟수가 증가하면 기준 낙하분산영역과의 오차가 줄어들고 있으며 JUT는 오차가 거의 없을 정도로 정확하게 일치하고 있다. 다구치 방법은 인자에 따른 영향을 분석할 수 있는 장점이 있으나 장단반경 오차는 각각 16.96%, 16.77%로 기준값의 낙하분산영역과는 다소 차이가 있다. 결과적으로, 낙하분산영역 계산에 사용되는 MCS(실험횟수:100회) 방법을 대체하여 JU 변환 기법을 적용할 경우, 최소 1/5배의 실험횟수가 감소하게 될 것으로 예상되며 낙하분산영역 계산 시간의

소요량을 줄이는데 크게 기여할 것으로 판단된다. 향후 JU 변환 기법을 우주발사체 비행안전분석시스템 개발에 적용할 예정이다.

참 고 문 헌

1. R. van der Merwe , "Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models", In Proc. of Workshop on Advances in Machine Learning 2003.
2. L. Angrisani , M. D'Aupizzo , R. Schiano Lo Moriello, "Unscented Transform: A Powerful Tool for Measurement Uncertainty Evaluation", IEEE, Vol. 55, No. 3, JUNE 2006.
3. S. Julier , J. Uhlmann, "A General Method for Approximating Nonlinear Transformations of Probability Distributions", 1996.
4. S. Julier , J. Uhlmann, "Unscented Filtering and Nonlinear Estimation", Proceedings of IEEE, Vol. 92, No. 3, Aug. 2004.
5. Jack B. Lyle, "Use of The Julier-Uhlmann Transform for Rapid Computation of Debris Impact Dispersions", 2nd IAASS Conference "Space Safety in a Global World", 2007.
6. Taguchi, G., "Introduction to Quality Engineering", Asia Productivity Organization, Tokyo, 1986
7. F.Mistree, U. Lautenschlager, S.O. Erikstad, J.K. Allen, "Simulation Reduction Using the Taguchi Method", NASA Contractor Report 4542, 1993.
8. 염봉진, 변재현, 이승훈, 김성준, "실험계획 및 분석: 다구치 방법과 직교표의 활용", 한국과학기술원 산학협동 공개강좌, 2010
9. Taguchi, G., "Introduction to Quality Engineering", Asia Productivity Organization, Tokyo, 1986