MDO에서 CO의 고찰과 해결방안에 관한 연구

박창규
광주대학교 AI자동차학과
e-mail:ckpark@gwangju.ac.kr

A Study on the Consideration and Settlement of Collaborative Optimization Approach in the MDO

Chang-Kyu Park
Dept. of AI Automotive Engineering, Gwangju University

요 약
본 논문은 Multidisciplinary Design Optimization(MDO)에서 최근에 활발하게 논의 되고 있는 협동최적화 기법(CO)에서 MDO문제를 CO로 적용하는데 있어서 본래의 최적설계 문제의 수치상의 특성을 파괴하여 여러 가지 수치적인 문제점이 발생한다. 따라서, 본 논문에서는 MDO의 다양한 접근방식들을 정리 한 후 CO가 가지는 문제점들을 나열하여 그에 대한 하나의 해결방안을 소개하고자 한다.

1. 서론

최근에 대규모의 구조 및 시스템 설계에서 서로 영향을 주는 다양한 분야의 원리들을 동시에 고려하는 다분야통합최적설계(multidisciplinary design optimization, MDO)기법이 사용되고 있다. 고전적 최적설계기법(all-in-one design approach) 하에서는 대부분의 변수들이 다수의 공학적 현상에 연관되어 있으므로 최종적으로 얻어진 설계 해의 의미를 판정하기 어려울 뿐 아니라, 변수가 많은 대규모의 설계문제에서는 모든 변수가 최적 설계값에 도달하기 전에 수렴하는 경우가 발생한다. 이러한 최적화상의 문제를 효율적으로 해결하기 위해 수많은 설계변수와 구속함수를 수반하고 여러 가지 공학적 현상이 발생하는 설계 문제를 효율적으로 해석하기 위해 가장 발전된 개념의 MDO 접근방식인 협동최적화(Collaborative Optimization, CO)기법이 적용되고 있다.

그러나, MDO문제를 CO로 적용하는데 있어서 본래의 최적설계 문제의 수치상의 특성을 파괴하여 여러 가지 수치적인 문제점이 발생한다.[Alexandrov,2000]

따라서, 본 논문에서는 MDO의 다양한 접근방식들을 정리 한 후 CO가 가지는 문제점들을 나열하여 그에 대한 하나의 해결방안을 소개하고자 한다.

2. MDO기법들의 비교

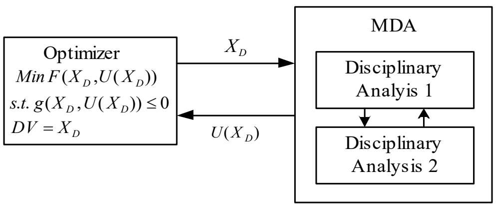
2.1 MDF(Multidisciplinary Feasible)

MDF 문제 구성은 MDO문제에 대한 해를 찾기 위한 일반적이며 전통적인 최적화 기법이다. MDF는 모든 하부레벨의 유용영역(feasible region)을 만족하는 설계를 하나의 프로세스의 계산으로 산출한다. 표 1 및 그림 1에 기술된 MDF 문제 구성에서 설계변수 X_D는 시스템에 대한 연성 변수이며 입력변수 X_D 및 출력변수 $U(X_D)$를 고정점 반복법(fixed-point iteration)에 의한 다분야통합해석(multidisciplinary analysis, MDA)을 통하여 목적함수 $F(X_D, U(X_D))$ 및 제약조건 $g(X_D, U(X_D))$을 계산하게 된다.

MDA에서는 모든 하부레벨의 해석모듈의 연성을 해결하기 위해서 반복계산을 통한 수렴해를 찾게 된다. 일반적으로 뉴우튼 방법(Newton Method)등이 MDA에 사용된다. 만약 본 문제를 해결하기 위해 도함수 기반방법(gradient-based method)이 사용된다면 MDA는 개별적인 몇 개의 점 뿐만 아니라 모든 점에서 전미분(derivative) 과정이 수행되어야하므로다분야유용설계(multidisciplinary feasibility)를 얻기 위해서는 계산비용이 많이 드는 단점이 발생하게 된다.

[표 1] Detail evaluation of new Top Side

Minimize	$F(X_D, U(X_D))$
Subject to	$g(X_D, U(X_D)) \le 0$
and bounds on design variable, X_D	



[그림 1] MDF Architecture

2.2 IDF(Individual Discipline Feasible)

IDF 문제 구성(formulation)은 MDO에서 계산비용의 주범인 MDA을 피할 수 있는 하나의 접근 방법이다. IDF는 개개의 분야의 유용설계만을 포함하는 반면에 다분야통합 유용설계(multidisciplinary feasibility) 및 분야간 연성변수를 제어할수 있는 적합성(compatibility)을 이끌어 내기 위한 최적화법이다. IDF에서 해석 변수는 공통 또는 연성을 나타내며, 해석되는 분야는 설계 변수와 상응되는 것으로 취급되고 각 하부레벨의 설계변수와는 구별된다. 요약하여 설명하자면 MFD에서 필요한 MDA과정을 버리고 시스템 레벨에서 이들을 관리하는 공통변수를 설정하여서 각 하부레벨의 유용설계 영역을 탐색하는 방식이다. 표 2와 그림 2에서 X_D는 설계변수의 집합이고 X_μ는 분야간 연성변수이다. C는 분야간 상수로 취급된다. 분야간의 적합성을 도모하기 위한 목적으로 J_i를 사용하는데 다음과 같다.

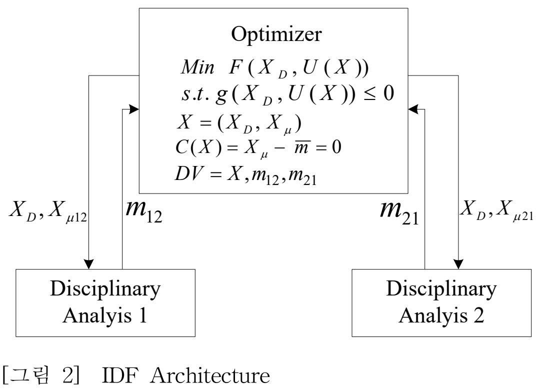
$$J_i = C_j^2 \le 0.0001 \qquad (1)$$

where, $j = 1, \cdots, number\ of\ disciplinary$

$U(X)$를 계산하는 것은 다분야 간 변수 X를 동시에 얻기 위해 개개의 분야 해석을 독립적으로 실행한다는 것이 중요하다. 그러므로 해석된 계산은 동시에 실행된다. 동시에 실행된다는 의미는 절차식의 과정이 아니라 독립적이고 평행한 계산이 가능하다는 의미이다. 일반적으로 다른 기종의 컴퓨터와 각각의 OS 환경하에서 독자적인 해석 패키지를 활용할 수 있다는 차원에서 IDF는 현실에 가장 부합될 수 있는 접근 방법임이라 본다.

[표 2] IDF Formulation

Minimize $F(X_D, U(X_D))$
with respect to $X = (X_D, X_\mu)$
s.t. $g(X_D, U(X_D)) \le 0$
$C(X) = X_\mu - \overline{m} = 0$
and bounds on design variable, X



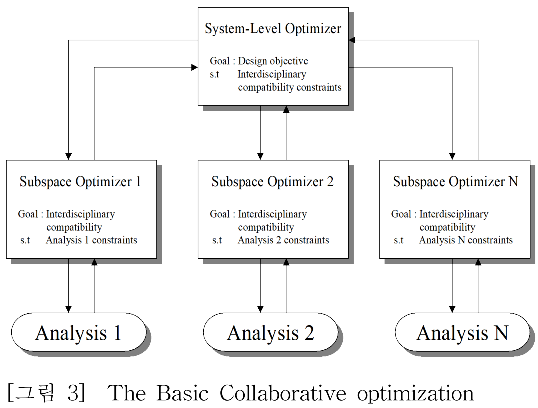
[그림 2] IDF Architecture

2.3 CO(Collaborative Optimization)

CO 구성은 MDO 에 대한 2계층 수직구조 기법(two-level hierarchical scheme)이며 시스템 레벨은 시스템 목적함수 F 를 최소화 하면서 분야간 공존 제약조건(interdisciplinary compatible constraint = discrepancy function) J^*를 만족하기 위한 다분야간 설계변수(또는 시스템 레벨 목표치, z)를 최적화 한다. 각각의 하부시스템 최적화(subsystem optimizer)는 하부시스템 제약조건 g_j를 만족하는 하부시스템 설계변수 x_i와 하부시스템 해석결과인 y_j간에 시스템 레벨 변수 z를 최소제곱(least square)의 형태로 최소화 한다. 이러한 시스템 레벨 설계변수 z는 하부시스템 문제 안에서 결정되도록 고려되어진다. 하부시스템이 하나 이상인 경우, 분야별 설계변수 x_{sj}와 공통 설계변수 x_j간에 구별이 되어야한다. 함축적인 목적으로 이러한 분야내부 간 공존 제약조건 J는 등가 제약조건 $J = 0.0$에 상반되게 비등가 제약조건 $J \le 0.0001$로 구성하며 J는 다음과 같이 정의된다. 여기서, $Z = \{Z^s, Z^c\}$이며 Z^s는 시스템 설계변수이고 Z^c는 시스템 연성변수를 나타낸다.

$$J_i = |X_j - Z_j^s|^2 + |Y_j - Z_j^c|^2 \qquad (2)$$

CO 구성은 공통 연성(interdisciplinary variable) x_j 보다 분야별 변수(disciplinary variable) x_{sj} 가 많을 때 사용하기에 이상적이다. 다른 말로 표현하면, CO 구성은 분야 개별적으로 큰 차원을 갖고, 설계문제간의 연성이 적은 설계문제를 풀기에 적절한 기법이다. 공통연성 변수의 증가는 시스템 차원에서의 설계변수의 증가를 의미하며 이는 시스템과 하위레벨간의 정확한 매칭을 저해하는 큰 요인이 되면 수렴시키기 위해서 상당한 계산비용이 요구된다. 하지만 분야간의 자율성을 최대한 보장한다는 목표로 구성된 CO에 있어서 각 분야 별로의 복잡성의 증가는 시스템 레벨 차원에서 보았을 때, 전자보다는 덜 치명적임을 알 수 있다.



[그림 3] The Basic Collaborative optimization Architecture

여러 가지 요소들의 상대적인 중요성을 평가할 때 AHP 기법은 평가의 상대적인 속성을 최대한 살릴 수 있는 일대일 비교를 기본

으로 하고 있다. 즉 모든 가능한 일대일 조합을 제시하고 각각에 대한 평가를 내리도록 유도함으로써 간접 비교에서 올 수 있는 부정확함을 최대한 배제하고, 좀더 신중하고 정확한 판단을 내리도록 한다. 또한 그 속성들의 수가 많을 경우 그룹을 지어 계층을 형성함으로써 그 평가의 체계화를 유도할 수 있다.

QFD에서 고객의 요구에 대한 비중을 결정할 때 AHP 기법을 이용함으로써 평가의 정확화, 체계화를 꾀할 수 있다. 특히 규모가 크고 복잡한 대상물의 경우 함께 고려해야할 기능의 수는 상당히 많아지게 된다. 이때 이들의 비중을 한번에 결정하는 방법은 부정확하고 일관성 없는 평가를 유도하기 쉽다. 계층을 형성하고 높은 단계에서부터 차례대로 비교 평가함으로써 일관적이고 체계적인 평가를 할 수 있다. 물론 이러한 비교 평가 이전에 관계자들 간에 충분한 토론을 거쳐 전체적인 계층 구조를 형성하는 작업도 매우 중요하다.

3. Penalty Function Method를 통한 수렴성개선

분야간 연성이 존재하는 MDA 문제(All-in-one or MDF)를 CO로 구성(formulation)하게 되면 구속조건의 비선형과 불연속 유용영역 왜곡으로 인해 해의 수렴성이 떨어지며 이로 인해 초기치 값의 변화에 매우 민감하게 된다. CO에 대한 수치적인 문제점은 다음과 같이 두 가지로 요약된다.

● 시스템 레벨 문제 구성에 관한 쿤-터커 (Kuhn-Tucker) 조건을 만족하지 못한다. 즉, 라그랑지 승수(Lagrange multiplier)가 시스템 레벨 등가 제약조건(equality constraint)에서 존재하지 않는다(Srinivas, 1998).

● CO로 구성된 시스템 레벨 최적화 문제가 원형인 MDF의 문제 구성을 왜곡하여 필요 없는 과도의 비선형성과 심한 불연속성을 띠게 된다.

이러한 문제점을 개선하기 위해 Lin[2002]은 CO의 시스템 레벨(system level)에 대해 목적함수를 penalty function method를 적용하여 최적해의 결과를 증가시켰으며, 보다 효율적인 강건설계 해를 보장해 주었다. CO는 system level에서 Kuhn-Tucker 필요조건을 만족하지 못한다. 따라서 최적해에 대한 정칙점(stationary point)이 존재하지 못하게 되며 그러한 이유로 인해 초기치에 매우 민감한 결과를 보이는 것이 사실이다. 그러나 CO의 시스템 레벨을 penalty function을 도입하여 비제약 최소화문제(unconstrained minimization problem)으로 구성하게 되면 그러한 문제점을 비교적 완화해주게 된다.

4. 수치 예제

CO를 다분야통합 최적설계 적용시 발생하는 문제점을 설명하기 위해 표 3과 같은, 구속조건(하나의 discipline으로 간주) 간에 연성이 존재하는 수치 예제의 최적화 문제를 생각하자. 그림 4에는 제약조건(constraint) 및 목적함수(objective) f 값에 대한 유용설계영역(feasible region)을 나타내었으며 제약조건과 목적함수가 교차하는 최적값을 기하학적으로도 쉽게 찾아낼 수 있다. 결국 전통적인 비선형 최적화기법(NLP)으로 쉽게 접근할 수 있는 문제의 형태를 갖추고 있음을 확인하였다. 다만 MDA 자체가 가지는 문제점으로 최적화 모듈에 할당하는 설계점에 따라 Newton Method에서 수렴된 결과를 주지 않는 경우가 있는데 이런 경우, 때때로 최적해에 수렴하지 못하고 발산되는 결과가 발견된다. 하지만 본 경우는 MDF 내부의 MDA의 문제이며 전체적인 시스템 레벨 문제 구성은 간단하다.

[표 3] Numerical Example Formulation

minimize $f = 100 - x_1^2 + x_2^2$

s.t. $g_1 = (0.2x_1^2 - 3y_2 - p_1^2 + 25)/10 \le 0$

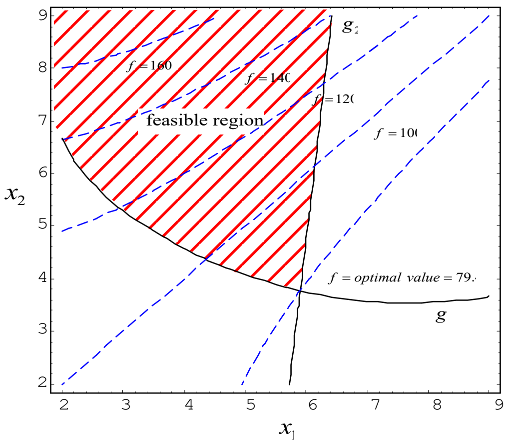
$g_2 = -(x_2 - y_1 - 3p_2 + 30)/20 \le 0$

$2 \le x_1 \le 9, \quad 2 \le x_2 \le 9$

where

$y_1 = x_1^2 - 0.2y_2 - 2p_1$

$y_2 = x_2 + \sqrt{y_1} + p_2^2$

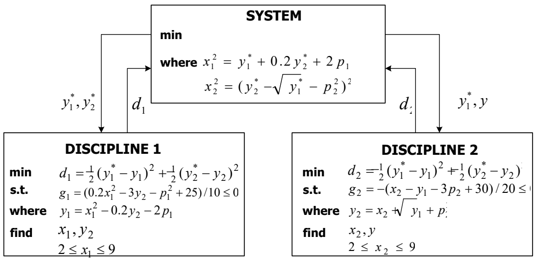


[그림 4] Feasible Region

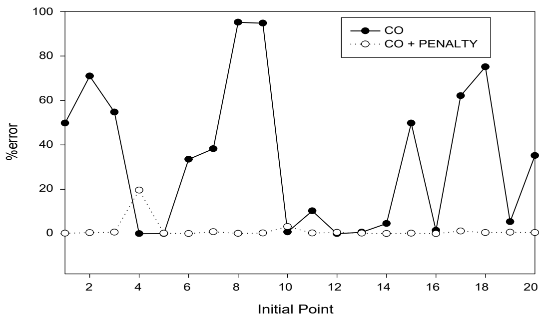
5. 결론

CO의 수치적인 문제점을 개선하기 위해 그림 3의 예제에 대하여 system level에 Penalty function method을 적용한 정식화를 그림 5에 나타내었다. 그림 6에는 초기치 값을 바꿔가면서 (20번) 해의 오차(% error)를 나타내었다. 그림 6에서 알 수 있는 바와 같이 CO에 비해 CO + Penalty function method을 적용한 경우(Table 4)이 최적해를 좀더 일관성을 유지하면서 Near-Optimum에 접근함을 알수 있다. CO에서 문제시 되고 있는 Discrepancy 구속조건을 Penalty Term으로 묶어서 비제약 구속조건 최적화 문제로 전환다는 점은 CO에서 '점원'으로 발생하는 조기 수렴문제를 점원마다에 차별성을 두게 되는 효과가

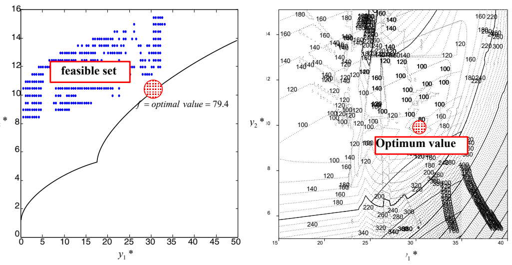
발생하여 비교적 우수한 해를 산출하고 있음을 확인할 수 있다. 이와 같이 CO + Penalty function method를 적용한 방법이 종래의 CO에 비해 안정된 해를 찾아주게 된다.



[그림 5] CO + Penalty Formulation



[그림 6] Results of Initial point variation



(a) CO (b) CO+penalty

[그림 7] Comparisons of feasible region

그림 5는 본 문제에 대한 CO 정식화이다. 여기에서 구속조건에 해당하는 식을 면밀하게 살펴보면 설계 유용영역이 점원으로 구성되어 있음을 알 수 있다. 결국 설계 영역 안에 유용영역의 존재가 이산화되어 존재함을 예상할 수 있는데, 설계변수는 각 하부레벨에 하달하는 공통연성 변수이며 하부레벨에서는 모든 설계변수에 관해서 만족되지 않는 것이 일반적이므로 그림 6의 (b)에서 볼 수 있듯이 전혀 연속성이 없는 점들의 집합이 유용설계 집합을 이루고 있음을 알 수 있다. 결국 이러한 CO의 내재된 문제점이 초기치의 변화에 따른, 일괄되지 못한 해를 도출하게 되는 큰 요인으로 작용하는 듯하다.

하부레벨의 문제 성향에 따라 본 유용 집합의 변화는 다양하게 이루질 수 있지만 본 수치 예제에서 조차 아무런 경향이 없는 형태를 띠는 것으로 미루어, 일반 공학문제에서도 비선형성이 더욱 심해질 수 있음에 주목해야 한다.

[표 4] Results of Optimization

		f	%error
All-in-one(MDF)		79.4118	
CO	Case 1	294.5253	49.78
	Case 2	113.0204	70.94
	Case 3	104.5587	54.75
	…		
	Case18	115.9899	75.11
	Case19	93.7131	5.43
	Case20	209.0511	35.19
CO +Penalty	Case 1	79.3480	0.22
	Case 2	79.6317	0.49
	Case 3	80.5414	0.71
	…		
	Case18	80.0916	0.51
	Case19	80.7188	0.67
	Case20	79.6951	0.51
Exact Sol.		79.4118	-

(※) $\%error = \dfrac{\sqrt{(y_1 - y_1*)^2 + (y_2 - y_2*)^2}}{\sqrt{y_1*^2 + y_2*^2}} \times 100(\%)$

NLP로 접근하기 어려운 문제를 풀어가기 위해서는 Evolutionary Optimization 혹은 Simulated Annealing와 같은 전역최적화 기법을 이용할 수 있다는 가능성은 있지만, 시스템 레벨에서의 반복계산의 증가는 중첩되어 있는 하부레벨의 최적화를 고려해 보건데 실현가능성이 없으며 이와 같은 전역 최적화 기법을 사용하였을 때 CO 문제구성에 필수적인 비선형 등가 구속조건도 다루기에 용이하지 않은 문제점을 안고 있다. 결국 계산비용과 해의 정확도 측면에서 CO의 문제구성을 적절하게 구현할 만한 최적화 기법은 존재하지 않는다고 본다.

참고문헌

[1] Alexandrov, N, M. and Lewis, R. M. " Analytical and Computational Aspects of Collaborative Optimization", NASA/TM-2000-21-1-4; AIAA J., vol. 40, pp.301-309, 2000.

[2] Srinivas Kodiyalam. "Evaluation of Methods for Multidisciplinary Design Optimization(MDO), Phase I", Paper No. NASA/CR-1998-208716, 1998.

[3] JiGuan G. Lin, and Linsys, "Examining and Improving Collaborative Optimization Approach to Multidisciplinary Design", AIAA paper 2002-5503, 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 2002.