

## 자동미분을 이용한 분리시스템동시최적화기법의 개선

### Improved Concurrent Subspace Optimization Using Automatic Differentiation

이종수\*                      박창규\*\*  
Lee, Jongsoo                Park, Changkue

---

#### ABSTRACT

The paper describes the study of concurrent subspace optimization(CSSO) for coupled multidisciplinary design optimization (MDO) techniques in mechanical systems. This method is a solution to large scale coupled multidisciplinary system, wherein the original problem is decomposed into a set of smaller, more tractable subproblems. Key elements in CSSO are consisted of global sensitivity equation(GSE), subspace optimization (SSO), optimum sensitivity analysis(OSA), and coordination optimization problem(COP) so as to inquiry valanced design solutions finally. Automatic differentiation has an ability to provide a robust sensitivity solution, and have shown the numerical numerical effectiveness over finite difference schemes wherein the perturbed step size in design variable is required. The present paper will develop the automatic differentiation based concurrent subspace optimization(AD-CSSO) in MDO. An automatic differentiation tool in FORTRAN(ADIFOR) will be employed to evaluate sensitivities. The use of exact function derivatives in GSE, OSA and COP makes possible to enhance the numerical accuracy during the iterative design process. The paper discusses how much influence on final optimal design compared with traditional all-in-one approach, finite difference based CSSO and AD-CSSO applying coupled design variables.

---

#### 1. 서 론

전통적인 최적화방법은 목적함수, 설계변수 및 구속조건 등을 정의하는 최적화의 수학적 표현을 하나의 단위(all-in-one)로 묶어 해석용 CAE 소프트웨어와 연결하여 최적화를 수행하는데, 다분야통합설계최적화(multidisciplinary design optimization, MDO)에서는 대부분의 설계변수들이 다수의 공학적 현상과 연관되어 있으므로 최종적으로 얻어진 최적설계의 의미를 판정하기 어려울 뿐 아니라, 대규모의 최적설계문제인 경우, 모든 설계변수가 최적설계에 도달하기 전에 수렴하는 경우가 발생한다. 이러한 MDO문제를 효율적으로 해결하기 위한 설계방법론으로서 분리시스템동시최적화기법(concurrent subspace optimization, CSSO)[1]이 적용되는데, 이 방법은 한 단위의 대규모 설계문제를 여러 개의 하부시스템으로 분리하여 독립적으로 설계를 수행하고 최종적으로 통합된 최적설계를 찾는 방법이다. MDO문제의 전체 설계시스템은 설계변수의 구성 및 연관된 공학적 현상에 따라 수직적(hierarchical) 또는 비수직적(nonhierarchical)시스템 등, 여러 형태의 하부 시스템으로 구성되는데 대부분의 경우 수직적연성(hierarchical coupling)된 시스템을 구성하기가 매우 어려우므로 대개는 비수직적으로 하부 시스템을 구성하는 것이 일반적이다.

CSSO는 전역민감도(global sensitivity equation, GSE)[2]를 기반으로 하부시스템최적화(subspace

---

\* 연세대학교 기계전자공학부 교수

\*\* 연세대학교 기계공학과 석사과정

optimization, SSO), 최적민감도해석(optimum sensitivity analysis, OSA), 중재통합최적화문제(coordination optimization problem, COP)로 구성되며 이러한 여러단계 중간계산과정에 1차 미분(1st order derivative)을 구해야한다. 설계 문제의 규모가 작거나 수식적으로 표현 가능한 경우에는 직접 미분을 취하여 1차 미분에 대한 엄밀해를 얻을 수 있으나, 유한 요소 등으로 계산되는 대규모 구조해석의 경우에는 수식적으로 정확한 함수의 표현이 불가능하므로 대부분의 경우, 유한 차분법 (finite difference scheme)에 의해 계산한다. 전방, 후방 및 중앙 차분법에 의한 미분 함수의 계산은, 설계 변수 또는 입력 변수의 변화량 (perturbed step size,  $\Delta x_i$ )에 따라 미분의 결과 및 최종적인 설계 해가 좌우된다. 예를 들어, 설계 변수의 변화량을 작게 선택한 경우에는 설계 최적화 과정의 반복 시간이 길어지게 되고, 변화량을 크게 선정하는 경우에는 오차가 많은 엄밀 해와 다른 결과를 생성하게 된다. 이러한 문제는 설계 파라미터에 대해 비선형적인 반응 함수에 대해 두드러지게 나타나고 수많은 반복 과정을 거치는 동안, 수치적 오차가 누적되는 결과를 가져온다. 설계 변수 변화량에 의해 결과가 좌우되는 유한 차분형 민감도 기법의 문제를 해결하기 위해서는 반응 함수를 구성하는 각 편미분 성분들의 특성을 고려한 미분 기법이 필요하게 되는데, 자동 미분 (automatic differentiation)은 각 독립 변수와 종속 변수의 관계를 이용하는 연쇄 법칙 (chain rule)을 근거로 신뢰성 있는 미분의 해를 제시하는 자동 미분 계산 기법이다.

본 논문에서는 하부시스템이 강한 연성을 갖는 경우에 비수직적 구조로써 하부시스템을 분리한 후 CSSO에서 사용된 1차 미분을 계산하는데 있어서 대표적인 자동 미분 프로그램인 ADIFOR (Automatic Differentiation In Fortran)[3]를 이용하여 수행한 결과와 유한차분법을 이용한 결과를 비교하였으며 또한, 고전적 최적설계기법(all-in-one design approach)에 의한 설계해를 비교하여 최종해에 얼마나 영향을 주게 되는지를 고찰 하였다.

## 2. 분리시스템동시최적화기법(concurrent subspace optimization, CSSO)

분리시스템동시최적화기법은 처음에 설계변수  $X_i$  와 교환계수(trade-off coefficient,  $t_k^*$ ) 를 초기화한 다음 시스템 해석이 수행되며 전역민감도(global sensitivity equation, GSE)를 기반으로 각 하부 시스템간의 연성된 설계변수 및 반응 함수의 결과사이의 연관관계를 파악한 후 CSSO의 발산을 피하기 위해 영향계수(effectiveness coefficient)를 구하여 가장 큰 변화를 갖는 설계변수를 할당한다. 이후 책임계수(responsibility coefficient,  $r_k^*$ )를 초기화 한 후 SSO를 수행하여 설계변수  $X_{ii}$ , 목적함수  $F$ 와 구속함수의 누계(cumulative measure of constraints, CMC)를 개선(update)한다.

SSO에서 수행된 연성계수( $r_k^*$ ,  $t_k^*$ )에 의해 새로운 구속최소점(constrained minimum point)이 정의 되므로 CMC와 F의 미분을 OSA에 의하여 구한다. OSA이후 여러개의 하부 시스템으로 분리된 시스템을 전체로 통합하여 연성계수를 최소화 하기위한 COP 수행하며 COP후에 연성계수를 개선한 후 수렴할때까지 반복한다.

### 2.1 전역 민감도(global sensitivity equation, GSE)

대규모의 다분야통합설계최적화 문제는 여러 개의 하부시스템으로 구성할 수 있으며, 이들은 서로 연성의 관계를 갖는다. 예를 들어, 전체 시스템이 두 개의 하부 시스템, A와 B로 이루어진 경우 다음과 같이 표현할 수 있다.

$$A[(X_A, X_B), Y_A] = 0 \quad B[(X_B, X_A), Y_B] = 0 \quad (1)$$

$$Y_A = Y_A(X_A, Y_B) \quad Y_B = Y_B(X_B, Y_A) \quad (2)$$

여기서  $X_A$  및  $X_B$ 는 각 하부 시스템에서 사용하는 독립된 설계 변수이고, 식(2)에서는 각 하부 시스템으로부터의 결과 함수  $Y_A$ 와  $Y_B$ 가 각각, 다른 하부 시스템의 해석 결과인  $Y_B$  및  $Y_A$ 와 서로 연성되어 있음을 보여준다. 이러한 하부 시스템간의 연성을 고려하기 위해 각 설계 변수에 대한 결과 함수의 민감도를 계산하는데, 식(2)에 연쇄 법칙을 적용하면 각 하부 시스템 A와 B에서의 전미분의 결과를 다음과 같이 얻을 수 있다.

$$\begin{bmatrix} I & -\frac{\partial Y_A}{\partial Y_B} \\ -\frac{\partial Y_B}{\partial Y_A} & I \end{bmatrix} \begin{bmatrix} \frac{dY_A}{dX_A} \\ \frac{dY_B}{dX_A} \end{bmatrix} = \begin{bmatrix} \frac{dY_A}{dX_A} \\ 0 \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} I & -\frac{\partial Y_A}{\partial Y_B} \\ -\frac{\partial Y_B}{\partial Y_A} & I \end{bmatrix} \begin{bmatrix} \frac{dY_A}{dX_B} \\ \frac{dY_B}{dX_B} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{dY_B}{dX_B} \end{bmatrix} \quad (4)$$

그 결과  $\frac{dY_A}{dX_A}$ ,  $\frac{dY_A}{dX_B}$ ,  $\frac{dY_B}{dX_A}$  및  $\frac{dY_B}{dX_B}$  등의 전미분의 값을 얻을 수 있는데 이러한 방법을 전역 민감도 방정식 (global sensitivity equation, GSE)이라 하며 이러한 민감도의 결과를 이용하여 각 하부 시스템간의 연성된 설계 변수 및 반응 함수의 결과사이의 연관관계를 파악할 수 있다.

## 2.2 영향계수(effectiveness coefficients)

각 하부 시스템에서 다량의 구속조건은 각각 다음과 같이 하나의 CMC에 의해 표현된다.

$$C = \frac{1}{\rho} \ln \left[ \sum_{j=1}^m \exp(\rho \cdot g_j) \right] \quad (5)$$

여기서,  $m$ 은 구속조건의 개수이며  $\rho$ 는 사용자가 정의한 상수이며  $g_j$ 는 부등호 구속조건 (inequality constraints)이다. CMC를 설계변수  $X_i$ 에 대하여 미분하면 식(6)과 같고 영향계수,  $e_{ij}$ 는 설계변수에 대한 구속조건과 목적함수의 변화량을 나타내며(식(7)), 모든 설계 구속조건에 대한 설계변수의 전체의 영향도(effectiveness)를 동시에 결정하기 위하여 식(7)을 하나의 CMC로 쓰기 위해 영향계수는 단 하나의 첨자로 식(8)과 같이 쓸 수 있다.

$$\frac{dC}{dX_i} = \left[ \sum_{j=1}^m \exp(\rho \cdot g_j) \right]^{-1} \left[ \sum_{j=1}^m \left\{ \frac{dg_j}{dX_i} \exp(\rho \cdot g_j) \right\} \right] \quad (6)$$

$$e_{ij} = \frac{dg_j/dX_i}{dF/dX_i} \quad (7)$$

$$e_i = \frac{dC/dX_i}{dF/dX_i} \quad (8)$$

영향계수를 구한후 CSSO의 발산을 피하기 위해 각 하부시스템에서 가장 큰 변화를 갖는 설계변수의 할당이 이루어진다. 예를 들어, 전체 시스템이 두 개의 하부 시스템, A와 B로 이루어진 경우 어떤 설계변수  $X_i$ 에 대한 영향계수의 값이 A보다 B의 하부 시스템이 작을 경우 그 설계변수는 B에 할당된다. 이것은 영향계수가 크면  $dC/dF$ 가 크게 되므로 구속조건이 목적함수에 위배되는 것이 커지게 되는 것을 의미한다.

## 2.3 책임계수 초기화

설계변수 할당후에  $r_k^p$ 의 초기화는 처음에 식(9)와 같이 k-번째 하부 시스템에 관련된 설계변수에 대한 p-번째 CMC의 미분을 구한다.

$$C_i^{pk} = \frac{dC^p}{dX_i^k} \quad i=1, \dots, nxk \quad (9)$$

여기서,  $nxk$ 는 k-번째 하부 시스템에서 설계변수의 총 개수이다. 다량의 하부 시스템의 CMC 민감도 행

렬은 수학적으로 다음 식(10)과 같이 자코비안(Jacobian)으로 표현된다.

$$J = \begin{matrix} & \begin{matrix} p=1 & \dots & p=\xi \end{matrix} \\ \begin{matrix} k=1 \\ \vdots \\ \vdots \\ k=\xi \end{matrix} & \begin{bmatrix} C_i^{p1} & \dots & C_i^{p\xi} \\ \vdots & \dots & \vdots \\ C_{mi}^{p1} & \dots & C_{mi}^{p\xi} \\ \vdots & \dots & \vdots \\ C_i^{p\xi} & \dots & C_i^{p\xi} \\ \vdots & \dots & \vdots \\ C_{mi}^{p\xi} & \dots & C_{mi}^{p\xi} \end{bmatrix} \end{matrix} \quad (10)$$

식(10)에서  $nx_k$ 는  $k$ 번째 하부 시스템에서 설계변수의 수이며  $\xi$ 는 하부 시스템의 전체 수이고  $mx_\xi$ 는  $\xi$ 번째 하부 시스템의 설계변수의 수이다. 변수  $v_k^p$ 는 각 하부 시스템에 대한 민감도의 최대 절대치의 항으로 정의되는데 다음식(11)과 같이 표현된다.

$$v_k^p = \frac{\max_i (|C_i^{pk}|)}{\max_k [\max_i (|C_i^{pk}|)]} \quad (11)$$

또한,  $v_k^p$ 를 비율(scaling)로 나타내면  $r_k^p$ 가 되며 다음과 같이 쓸 수 있다.

$$r_k^p = \left( \frac{1}{\sum_k v_k^p} \right) v_k^p \quad p=1, \xi \quad (12)$$

#### 2.4 하부시스템최적화(subspace optimization, SSO)

책임계수 초기화한 후 각 하부 시스템은 분리된 하부시스템최적화(subspace optimization, SSO)가 수행되는데 설계문제의 수식적 표현은 다음과 같다.

Minimize  $F(X^k)$

$$\text{subject to} \quad C^p \leq C^{p0} [s^p(1-r_k^p) + (1-s^p)t_k^p] \quad p=1, \text{NSS} \quad (13)$$

$$X_k^L \leq X_k \leq X_k^U$$

여기서, NSS는 하부 시스템의 갯수이며  $C^p$ 는  $p$ 번째 하부 시스템에 존재하는 CMC이며  $s^p$ ,  $r_k^p$ 와  $t_k^p$ 는 시스템 전체를 여러개의 하부 시스템으로 분리함으로써 나타나는 상호영향을 나타내는 계수이다.

책임계수  $r_k^p$ 는  $p$ 번째 SSO에서 CMC의 위배성을 줄이기 위하여  $k$ 번째 SSO에 할당된 책임(responsibility)을 나타낸다. 즉,  $r_k^p$ 는 만약 하나의 하부 시스템에서 구속조건이 위배되었다고 하면 그 위배된 것을 혼자 책임지는 것이 아니라 다른 하부 시스템이 조금씩 분배하여 책임을 나누게 된다. 그러므로 혼자 100%의 책임을 지는 것이 아니라 여러개로 분리된 각 하부 시스템이 100%의 책임을 조금씩 나누어 갖게 된다.

교환계수  $t_k^p$ 는  $p$ -번째 SSO에서 목적함수를 줄이기 위하여 구속조건이 위배 되었을 때 나머지  $k$ -번째 SSO에서 구속조건을 원래보다 조금 더 만족하여 전체에 대하여 구속조건이 상쇄되도록한다. 즉, 하나의 하부 시스템에서 구속조건이 커지면 나머지 하부 시스템은 작아지게 되어 결국은 전체의 합은 0이 된다.

$s^p$ 는 SSO에서  $r_k^p$ 와  $t_k^p$ 계수중에서 한가지만을 구속조건으로 표현하기 위한 스위치(switch)파라메타 이다.  $s^p$ 는 0또는 1이 되어  $r_k^p$ 와  $t_k^p$ 중에서 한가지만을 선택하도록 하게 하며 SSO후에 설계변수, 목적함수와 구속조건(CMC)이 개선(update)된다.

2.5 최적민감도 해석(optimum sensitivity analysis, OSA)

SSO에서 구속함수가  $r_k^p$  및  $t_k^p$ 로 형성되어 있기 때문에 새로운 최적점이 이러한 변수에 의존하게 된다. 그러므로 최적민감도 해석(optimum sensitivity analysis, OSA)이 이루어지는데 최적민감도 해석은 라그랑지 승수(Lagrange multiplier)에 의존하게 되기 때문에 Kuhn-Tucker condition의 항으로 정의되며 수학적 표현은 다음과 같다.

$$\begin{aligned} \nabla F(X^*) + \sum_{j=1}^m \lambda_j \nabla g_j(X^*) + \sum_{k=1}^n \lambda_{k+m} \nabla h_k(X^*) &= 0 \\ \lambda_j g_j(X^*) &= 0 \quad j=1, m \quad \text{and} \quad \lambda_j \geq 0 \end{aligned} \quad (14)$$

여기서  $\lambda_j$ 와  $\lambda_{k+m}$ 은 등호 구속조건(equality constraint)과 부등호 구속조건(inequality constraint)에 관계된 라그랑지 승수이며  $X^*$ 는 구속된 최적점(constrained optimum)에서 최적설계(optimum design)를 나타낸다. 하부 시스템에서 부등호 구속조건만이 존재하고 등호 구속조건은 없고 또한 부등호 구속조건은 하나의 CMC로 표현되므로 식(14)는 다음 식(15)와 같이 나타낼 수 있으며 시스템 전체 하부 시스템에 대하여 표현하면 식(16)과 같다.

$$\frac{dF}{dX_k^*} + \lambda_k \frac{dC}{dX_k^*} = 0 \quad (15)$$

$$\frac{dF}{dX_k^*} + \sum_{p=1}^s \lambda_k^p \frac{dC^p}{dX_k^*} = 0 \quad (16)$$

식(16)에 의한 라그랑지 승수(Lagrange multiplier)는 다음식(17)과 같고

$$[\lambda_k] = - \left[ \left[ \frac{dC}{dX_k^*} \right]^T \left[ \frac{dC^p}{dX_k^*} \right]^T \right]^{-1} \left[ \frac{dC}{dX_k^*} \right]^T \left[ \frac{dF}{dX_k^*} \right] \quad (17)$$

여기서,  $[\lambda_k] = [\lambda_k^1, \lambda_k^2, \dots, \lambda_k^s]^T$ ,  $[dC/dX_k^*] = [dC^1/dX_k^*, dC^2/dX_k^*, \dots, dC^s/dX_k^*]^T$ 이다.

라그랑지 승수(Lagrange multiplier)를 구한후 목적함수 F의 최적민감도(optimum sensitivity)는 식(18)과 같은데 여기서  $z_i$ 는  $r_k^p$  또는  $t_k^p$ 를 나타내는 변수이다.

$$\frac{dF}{dz_i} = \sum_p \lambda_k^p \frac{dC^p}{dz_i} \quad (18)$$

2.6 중재통합최적화문제(coordination optimization problem, COP)

식(18)에서 구한 목적함수의 미분은 시스템 전체에 대한 연성변수  $r_k^p, t_k^p$ 을 통합화하여 최소화 되도록 중재통합최적화문제(coordination optimization problem, COP)를 구성하는데 다음과 같다.

$$\begin{aligned} \text{Minimize} & \quad F(r_k^p, t_k^p) \\ \text{subject to} & \quad \sum_k r_k^p = 1 \\ & \quad \sum_k t_k^p = 0 \\ & \quad 0 \leq r_k^p \leq 1 \\ & \quad r_{kL}^p \leq r_k^p \leq r_{kU}^p \quad \text{and} \quad t_{kL}^p \leq t_k^p \leq t_{kU}^p \end{aligned} \quad (19)$$

중재통합최적화(COP)과정에서 계수를 개선(update)한 후에, 전과정이 수렴조건이 될 때까지 반복되는데 이러한 CSSO의 과정을 도식적으로 나타내면 Fig. 1과 같다.

### 3. 자동미분

설계 최적화 분야에서 민감도 (sensitivity)는 주어진 설계 변수 (design variable)의 변화에 대한 목적 함수 또는 구속 함수와 같은 반응 함수 (response function)의 변화량으로 정의되며, 이러한 민감도는 수학적으로 구배 (gradient) 또는 미분 (derivative)의 형태로 표현된다. 공학적 문제의 경향 분석을 위해 민감도 해석은 중요한 도구로 활용될 뿐 아니라, 미분에 근거한 근사 해법을 이용하여 설계를 수행할 경우, 방대한 계산 시간에 의한 설계 비용을 절감할 수 있는 급속 재설계 (rapid redesign)의 장점을 제공한다. 민감도 해석은 다양한 방법으로 전개할 수 있는데, 설계 문제의 규모가 작거나 수식적으로 표현 가능한 경우에는 직접 미분을 취하여 엄밀해를 얻을 수 있으나, 유한 요소 등으로 계산되는 대규모 구조해석의 경우에는 수식적으로 정확한 함수의 표현이 불가능하므로 대부분의 경우, 유한 차분법 (finite difference scheme)에 의해 민감도를 계산한다.

자동 미분은 편미분의 관계를 컴퓨터 프로그램에 의해 직접 미분하여 최종적인 전미분 (total derivative)의 해를 계산하므로 유한 차분법에서 요구되는 변수 변화량에 무관하고 기계 오차 (machine precision)의 범위 한도 내에서 엄밀 해와 일치된 결과를 생성할 수 있다. 자동 미분의 방법은 수학적 이론[4,5]에서 출발하여 다양한 기법이 개발되었으며, 전산 해석 및 설계 문제에 적용할 수 있는 대표적인 자동 미분 프로그램인 ADIFOR (Automatic Differentiation In Fortran)를 이용하여 설계 민감도 문제 및 최적화에 적용되고 있다.

ADIFOR는 Fortran 언어로 작성된 수치해석 프로그램 (source code)이 1차 미분, 또는 구배의 값을 계산할 수 있도록 자동 변환시켜주는 프로그램으로써, 구조 설계를 위한 대규모 최적화 문제뿐만 아니라, 전산 유체 역학에 의한 공력 설계 (aerodynamic design) 문제[6,7]에도 다양하게 적용되고 있다. 수많은 반복 계산 과정을 요구하는 최적 설계 문제에 있어서 유한 차분 기법에 의한 민감도 계산 시간을 줄이고 신뢰성 있는 해를 얻기 위해 설계 최적화 프로그램에 자동 미분 기법을 통합시키고 있다. 자동 미분 프로그램인 ADIFOR를 이용하여, 컬럼의 단면 크기에 대한 좌굴 하중의 민감도 및 아음속 날개 표면의 형상 변화에 대한 양력 계수의 민감도를 유한 차분법 또는 해석적 방법 (analytic method)에 의한 결과와 비교하였고 이를 Fig.3에 나타내었다. 유한 차분에 의한 결과는 설계 변수의 변화량 (perturbed step size)에 따라 다르게 나타나지만, ADIFOR는 일정한 허용 오차 범위 내에서 해석적 방법에 의한 결과에 일치하는 경향을 갖는 신뢰성 있는 결과를 생성한다.

#### 3.1 ADIFOR에 의한 분리시스템동시최적화기법

앞 절에서 설명한 전역민감도를 기반으로 하는 분리시스템동시최적화기법을 수행하는데 있어서 미분을 구하기 위해서는 유한 차분법(finite difference scheme)에 의해 계산된다. 그러나 유한 차분법은 입력변수의 변화량에 의해 결과가 좌우되며 최적화방법에서는 수많은 반복계산을 요구하므로 입력변수의 변화량을 아무리 작게 하여도 수치적 오차가 누적된다. 따라서 정확한 해의 경향을 위해서는 ADIFOR에 의한 미분을 수행하는 것이 바람직하다.

Fig.1에 분리시스템동시최적화기법에 대한 흐름도(flowchart)를 나타내었는데 ADIFOR를 이용한 흐름도는 Fig.2와 같은데 ADIFOR는 전역민감도 계산(식3, 식4), 영향계수(식8),  $r_k^*$  초기화과정(식(9)), OSA(식(17))과 COP를 구성하기 위한 목적함수의 미분(식(18))에 사용되었다.

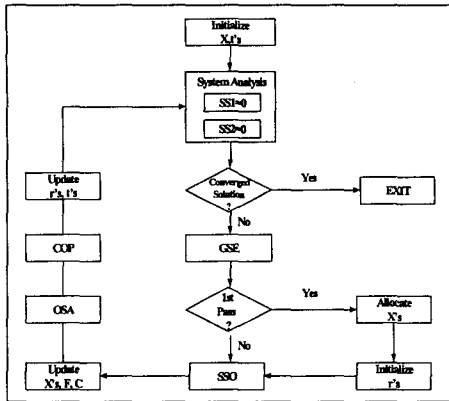


Fig1. Flowchart for CSSO method

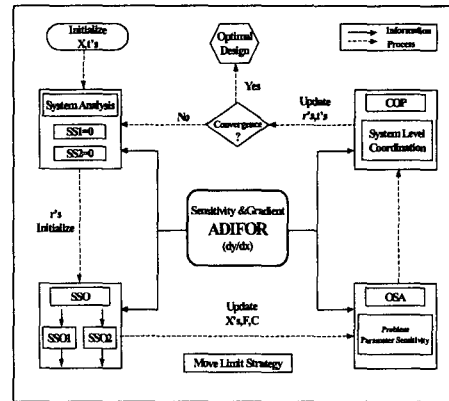


Fig2. CSSO using ADIFOR

### 6. 수차예제 (Speed Reducer)

본 예제에서는 Fig.3(model), Fig.4에 나타난 speed reducer의 기어 휠(gear wheel)과 전달축(transmission shaft)의 전체 체적의 최소값을 찾는 문제인데 이 문제는 기어(gear)와 축(shaft)으로 시스템이 연성되어 있기 때문에 기어와 축으로 2개의 하부시스템으로 분리하여 최적화를 수행하는 분리시스템동시최적화기법에 ADIFOR를 적용한 방법, FDM을 이용한 방법과 all-in-one과의 결과를 비교하였다. 이문제는 비수직적 (nonhierarchical)구조로 2개의 하부 시스템(기어와 축)으로 분리되었으며 각 하부 시스템간의 연성(coupling)은 Fig.6에 나타내었다.

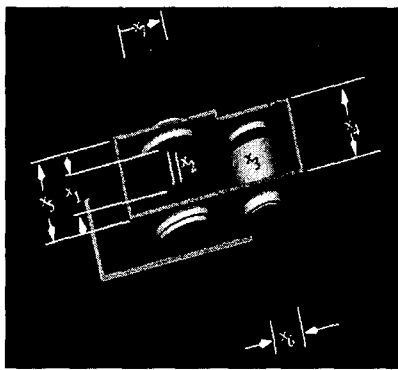


Fig3. Speed reducer model[8]

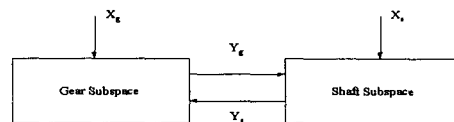


Fig4. Couplings between the gear and shaft subspaces

speed reducer는 7개의 설계변수와 11개의 구속조건으로 구성되며 최적화 문제의 수식적 표현은 다음과 같다.[9,10]

$$\begin{aligned} \text{Minimize } f(x) = & 0.7854x_1x_2^2(3.333x_3^2 + 14.9334x_3 - 43.0932) \\ & - 1.5079(x_6^2 + x_7^2)x_1 + 7.4769(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2) \end{aligned} \quad (20)$$

subject to

$$\begin{aligned}
 g_1 &= \frac{[x_1 x_2^2 x_3]}{27.0} - 1.0 & g_6 &= \frac{[x_2 x_3 x_6^4]}{[1.925 x_4^3]} - 1.0 \\
 g_2 &= \frac{[x_1 x_2^2 x_3^2]}{397.5} - 1.0 & g_7 &= \frac{[x_1 x_3 x_4^4]}{[1.925 x_5^3]} - 1.0 \\
 g_3 &= \frac{x_1}{[5x_2]} - 1.0 & g_8 &= \frac{[1100 \cdot B_1]}{A_1} - 1.0 \\
 g_4 &= \frac{[12.0 x_2]}{x_1} - 1.0 & g_9 &= \frac{[850 \cdot B_2]}{A_2} - 1.0 \\
 g_5 &= \frac{40.0}{[x_2 x_3]} - 1.0 & g_{10} &= \frac{x_4}{[1.5x_6 + 1.9]} - 1.0 \\
 & & g_{11} &= \frac{x_5}{[1.5x_7 + 1.9]} - 1.0 \\
 A_1 &= \left[ \left( \frac{745 \cdot x_4}{x_2 x_3} \right)^2 + 16.91 \times 10^6 \right]^{0.5} & B_1 &= 0.1x_6^3 \\
 A_2 &= \left[ \left( \frac{745 \cdot x_5}{x_2 x_3} \right)^2 + 157.5 \times 10^6 \right]^{0.5} & B_2 &= 0.1x_7^3
 \end{aligned} \tag{21}$$

Fig4.에 나타낸 기어(gear)와 축(shaft)로 2개의 비수직적 하부시스템으로 분리한 각각의 입력변수와 출력변수는 식(22)와 같다.

$$\begin{aligned}
 \{x_g\}^T &= \{x_1, x_2, x_3\} \\
 \{y_g\}^T &= \{f, g_1, g_2, g_3, g_4, g_5\} \\
 \{x_s\}^T &= \{x_4, x_5, x_6, x_7\} \\
 \{y_s\}^T &= \{f, g_6, g_7, g_8, g_9, g_{10}, g_{11}\}
 \end{aligned} \tag{22}$$

본 논문에서 사용한 Speed Reducer를 2개의 하부시스템으로 각각 기어, 축 하부시스템으로 분리하여 고전적 최적기법(all-in-one)과 시스템동시최적화기법을 비교하여 최적화를 수행하였고 동시최적화기법은 유한차분법(전방 차분법)을 이용한 방법과 ADIFOR를 이용한 3가지 방법에 대한 목적함수의 수렴과정을 Fig5.에 도식적으로 나타내었고 Tab.1에는 목적함수와 설계변수의 데이터 결과를 표로 나타내었는데 CSSO에 사용된 최적화( SSO, COP )코드는 CONMIN[11]을 사용하였으며 이 결과에서 알 수 있듯이 all-in-one과 유한차분법(전방차분법)을 이용한 분리시스템동시최적화기법에서 목적함수의 최적값에 상당한 차이가 났으며 또한 전방차분법을 이용한 분리시스템동시최적화기법보다는 ADIFOR를 이용한 방법이 훨씬 빠른 최적결과를 나타냈다. 이결과는 분리시스템동시최적화기법에 유한차분법을 이용한 방법은 반복적인 계산에 의한 수치누적에 의한 결과이며 다량의 설계 변수와 비선형함수, 시스템간에 강한 연성을 갖는 함수인 경우엔 이러한 결과의 오차가 커지게 됨으로써 정확한 해의 경향을 판단하기 어려워지게 하는 원인이 된다.

Tab.2 는 분리시스템동시최적화기법에 ADIFOR를 적용한 4번의 cycle과정에서의 구속조건(G(1),G(2))과  $r_k^s$ ,  $l_k^s$ 의 변화를 나타내었는데 앞에서 설명한 바와 첫 번째 0 cycle은  $r_k^s$ 의 초기화에 의한 값이고 두 번째부터 최적설계값으로 수렴하기까지의 결과를 살펴보면 구속조건의 값이 0보다 적은 경우에는 위배되지 않으므로  $l_k^s$ 의 값이 사용되었고(이때는  $s^p = 0$ ) 구속조건의 값이 0보다 커지게 되고 구속조건이 0보다 크면  $r_k^s$ (이때는  $s^p = 1$ )가 사용된다.

Fig.6에는 설계변수의 초기값(Initial point)에 대한 영향도(effectiveness)를 나타내었는데 여기서 알 수 있듯이 각 설계변수의 값이 상한(upper bound)과 하한(lower bound) 사이에 위치한다. 이것은 설계변수가 큰 변화가 없음을 나타내며 연성된 전체시스템을 비수직적(nonhierarchical) 구조로 분리하여 각각의 하부시스템을 최적화하여도 상한과 하한의 경계를 벗어나는 설계변수가 없으므로 앞에서 설명한 설계변수 할당을 이해



도 지장이 없음을 의미하며 effectiveness의 상한과 하한을 구하는 과정은 다음과 같이 식(23),(24),(25),(26)의 과정에 의하는데 여기서  $N$ 은 설계변수의 총개수 이며  $\bar{e}$ 는 평균 영향도,  $\sigma(e)$ 는 표준편차(standard deviation),  $e^u, e^l$ 은 각각 상한,하한 영향도를 나타낸다.

$$\bar{e} = \frac{1}{N} \sum_{i=1}^N e_i \quad (23)$$

$$\sigma(e) = \left[ \frac{1}{N-1} \sum_{i=1}^N (e_i - \bar{e})^2 \right]^{\frac{1}{2}} \quad (24)$$

$$e^u = \bar{e} + \sigma(e) \quad (25)$$

$$e^l = \bar{e} - \sigma(e) \quad (26)$$

Fig.7에는7개의 설계변수중에서 설계변수1,2,3에 대한 이동제한(move limit,  $\mu$ )의 % 분포를 4번의 cycle에 대하여 ADIFOR를이용한 분리시스템동시최적화기법에 대한 결과를 나타내었다(식(25)). 이 결과에서  $\mu$ 의 상한과 하한은 각각 90%, 10%로 주었는데 만약 하한의 값을 너무 적게 주면 계산량이 많아지게 되어 최적설계에 수렴하기 위해 반복과정이 많아지게 되고 상한을 너무 높게 주게 되면 선형화(linearization)를 할 수 없게 되어 결국 최적민감도해석(OSA)에서 활성구속조건(active constraint)이 비활성(inactive)으로 바뀌게 되는 결과를 초래하게 된다.

$$\mu \% = \frac{(e - e^l)}{2\sigma(e)} (\mu^u - \mu^l) + \mu^l \quad (27)$$

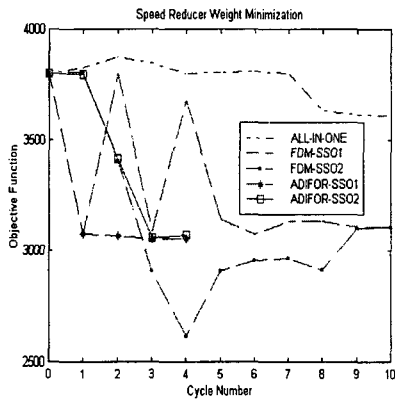


Fig5. CSSO(FDM-ADIFOR) & all-in-one convergence history

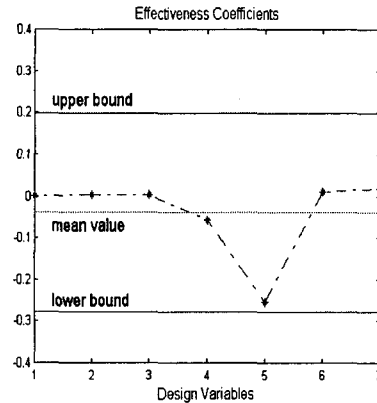


Fig6. Effectiveness space for initial point

Table1. Optimal design results

Initial	Final			
	all-in-one	CSSO		
		FDM	ADIFOR	
OBJ	3798.57	3609.62	3106.57	3053.86
X(1)	2.8	2.6	2.7976	2.7042
X(2)	0.7	0.6	0.60	0.601
X(3)	25.	28.	24.996	24.997
X(4)	7.9	7.5336	7.8993	7.8982
X(5)	7.6	7.9421	7.3	7.5497
X(6)	3.0	2.9	3.0637	3.0076
X(7)	5.0	5.0	5.0834	5.0041

Table.2 Coefficient and constraint values for 4 optimization cycles.(ADIFOR)

Cycle	Active Coefficients(ADIFOR-CSSO)					
	G(1)	G(2)	R11	R12	T11	T12
0	-2.37	-2.14	0.474	0.526	-	-
1	0.14	0.14	0.475	0.476	-	-
2	-0.17	-0.15	-	-	-1.0	1.0
3	-0.05	-0.03	-	-	-1.0	1.0
4	-0.001	-0.001	-	-	-1.0	1.0

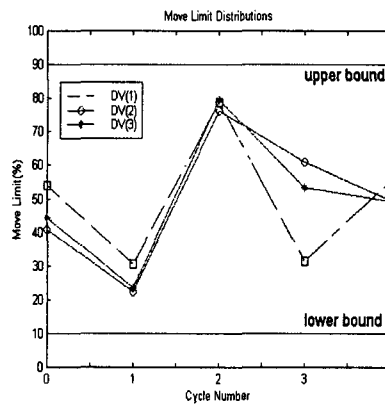


Fig7. Move limit distributions for dv(1), dv(2), dv(3)

## 7. 결론

본 논문을 통해 MDO의 효율적인 설계방법론으로 제시된 분리시스템동시최적화기법을 제시하고 all-in-one의 결과와 비교 하였다. 대부분의 MDO문제는 수많은 설계변수를 포함하며, 설계의 요구조건들도 명확하게 표현할 수 없는 경우가 생기게 된다. 따라서 이러한 MDO에서 정확한 해의 경향을 판단하는데 분리시스템동시최적화기법을 적용하는 것이 바람직하다. 또한, 분리시스템동시최적화기법을 통해 민감도 해석 시 발생하는 수치적 오차의 한계를 극복하고 신뢰성 있는 설계결과를 얻기 위해 자동 미분의 개념을 이용한 ADIFOR를 사용하여 해석했다. 분리시스템동시최적화기법은 최적민감도해석(OSA)에서 활성 구속조건(active constraint)이 비활성(inactive)으로 바뀌지 못하도록 하기위해서 하부시스템최적화(SSO)와 중재통합최적화문제(COP)에서 근사화(linearization)를 해야되는데 근사화는 각 설계변수에 move limit을 주어서 수행해야하는데 만약 move limit이 작으면 많은 계산량이 필요하며 move limit이 너무 클 경우에는 구속조건이 inactive로 바뀌는 결과를 초래 할 수 있는 단점이 있다. 또한 한가지는 계산과정이 너무 많아 CSSO를 수행하는데 여러 계산과정을 거쳐야 한다는 것이다. 그러나 이 방법은 여러개의 하부 시스템으로 분리하여 최적화를 수행함으로써 all-in-one보다 더 정확한 해를 찾게되며 한단계 더 나아가 분리시스템동시최적화기법에 ADIFOR를 이용한 방법은 유한차분법에 의한 방법보다 더 최종해를 개선할 수 있는 방법을 제시하게 되었다.

## 후기

이 연구는 한국과학재단 지정 최적설계신기술 연구센터(Center of Innovative Design Optimization Technology, iDOT)의 연구비 지원으로 수행되었습니다.

## 참고문헌

- [1] Bloebaum, C. L., Fomal and Heuristic System Decomposition Methods in Multidisciplinary Synthesis, Ph.D Dissertation in Aerospace Engineering, the University of Florida, Gainseville, FL, 1991.
- [2] Hajela, P., Bloebaum, C. L., and Sobieski, J., "Application of Global Sensitivity Equations in Multidisciplinary Aircraft Synthesis," the AIAA Journal of Aircraft, Vol. 27, No. 12, pp. 1002-1010, December, 1990.
- [3] ADIFOR 2.0, User's Manual(Revision D), Rice University, Texas, 1998.
- [4] Iri, M., "Automatic Computation of Partial Derivatives and Rounding Error Estimates with Applications to Large-Scale Systems of Nonlinear Equations," Journal of Computers and Applied Mathematics, Vol. 24, pp. 365-392, 1988.
- [5] Rall, L. B., "Automatic Differentiation: Techniques and Applications," Lecture Notes in Computer Science No. 120, Springer, New York, 1981.
- [6] Bischof, C., Corliss, G., Green, L., Griewank, A., Haigler, K., Newman, P., "Automatic Differentiation of Advanced CFD Codes for Multidisciplinary Design," Computing Systems in Engineering, Vol. 3, No. 6, pp. 625-637, 1992.
- [7] Kuethe, A. M., Chow, C.-Y., Foundations of Aerodynamics - Bases of Aerodynamic Design, John Wiley & Sons, 1986.
- [8] NASA TEST SUITE, Multidisciplinary Optimization Branch(MDOB), NASA Langley Research Center, Hampton, VA, 1999.
- [9] Mehmet, A. A., Domain Decomposition in Multidisciplinary Design: Role of Artificial Neural Networks and Intelligent Agents, Ph.D. Dissertation in Mechanical Engineering, Rensselaer Polytechnic Institute, Troy, NY, 1997.
- [10] T. W. Lee, "Weight Minimization of a Speed Reducer", ASME paper, No.77-DET-163, 1977.
- [11] Vanderplaats, G. N., "CONMIN-A FORTRAN Program for Constrained Function Minimization --User's Manual", NASA TM X-62282, 1973.