

외부 최적화 모듈과 ADAMS/Car의 접속 및 자동화를 위한 GUI 제어 기술 개발

Development of GUI Control Technology for Interfacing and Automation of an External Optimization module with ADAMS/Car

최병렬, 주민식, 최동훈(한양대학교)

Byung-Lyul Choi, Min-Sik Chu and Dong-Hoon Choi (Hanyang University)

여민수, 김만식(ATT R&D)

ABSTRACT

This research develops Graphic User Interface(GUI) control technology for performing the optimization using external optimization module and ADAMS/Car. To control ADAMS/Car's GUI, Window Scripting Host(WSH), which is a scripting engine and is presented by Microsoft, is used. Macpherson type suspension optimization is performed for interfacing an external optimization module and ADAMS/Car. This GUI control technology can be efficiently applied in the optimization using an external optimization module or the parameter study needed iterative analyses

1. 서론

현재 실제 제품설계과정에서 최적화 기법의 도입이 증가 되고 있는 경향을 보이고, 이에 대한 관심도 또한 증폭되고 있는 현실이다. 기존의 설계 과정은 설계자의 직관이나 경험을 바탕으로 반복적인 시행착오를 필요로 하여 상당한 시간을 요하는 과정이었다. 최적화 기법을 도입함으로써 인해 상당 부분의 설계 과정의 자동화를 이룰 수 있게 되었고, 아울러 설계 기간의 단축을 통하여 제작비 절감을 유도 할 수 있다. 이러한 추세에 발맞춰 컴퓨터 시뮬레이션을 이용한 상용 해석 소프트웨어에도 최적화 모듈이 내포되는 경향을 보이고 있다. 상용 소프트웨어에 내포된 최적화 모듈은 대개 국부 최적해를 찾아주는 국부 최적화 모듈이 대부분이며, 제공되어지는 최적화 알고리즘 외에는 사용할 수 없는 단점이 있다. 일반적으로 최적화 문제에 따라서 효율적인 알고리즘은 달라질 수 있으며, 특히 긴 해석 시간을 요하는 최적화 문제의 경우, 반복적인 해석을 요하는 일반적인 최적화 알고리즘은 적합하지 않다. 따라서 최적화 문제에 맞게 효율적인 해를 찾기 위해서 다양한 접근 방식을 제공하는 설계 프레임워크가 등장하기 시작했다. 설계 프레임워크는 최적화 기법을 포함한 다양한 설계

기법들을 포함하고 있으며, 기존의 상용 해석 소프트웨어와 접속하여 설계 결과를 제공하는 역할을 수행한다. 상용 소프트웨어와의 접속을 위한 일반적인 방법은 입출력 파일을 통해서 최적화에 필요한 설계변수와 성능지수를 전달하는 동시에 설계 프레임워크에서 호출 가능한 실행파일을 통해 해석 소프트웨어를 실행시키는 방식을 택한다. 하지만 대개의 해석 소프트웨어의 경우, 외부에서부터 설계 정보를 변경하여 반복적으로 해석기를 수행할 수 있는 기능은 제공하고 있지 않다. 특히 ADAMS/Car와 같이 GUI를 통해서만 설계 정보를 변경할 수 있는 소프트웨어의 경우, 최적화 모듈과의 접속이 가장 큰 어려움으로 나타났다. 이에 본 연구에서는 GUI를 외부에서 제어함으로써 설계 프레임워크와의 접속함과 동시에 ADAMS/Car의 현가장치 템플릿을 이용하여 현가장치의 최적설계를 수행하고자 한다. 승차감 및 운전특성 향상을 설계 고려사항으로 선정하였으며, 현가 장치의 특성에 주로 영향을 미치는 일반적인 자동차 기구학적 특성인 캐스터 앵글, 캠버 앵글, 토우 앵글, 엔타이 다이브, 롤 센터 하이트 등의 성능을 최적화하고자 한다.

2. 외부에서 GUI를 통한 ADAMS/Car의 제어

ADAMS/Car의 경우, 입출력파일과 실행파일만을 가지고 접속하기란 상당히 번거롭고, ADAMS/Car의 내부모듈에 대한 이해를 요하기 때문에 일반적으로 GUI 환경을 통해서 설계 변경을 하게 된다. 따라서 반복적인 호출을 요하는 최적화 모듈과의 접속을 이루기 위해서 새로운 방식의 접근이 필요하다. 본 연구에서는 스크립트를 이용하여 GUI를 제어하는 방식에 대해 기술하고자 한다.

스크립트의 대표적인 예로 배치파일을 들 수 있다. 일반적으로 사용되는 배치파일은 컴파일러 없이 인터프리터를 이용하여 여러 개의 연산이나 실행을 미리 기술하여 배치파일을 실행함으로써 미리 기술한 연산이나 실행을 수행하는 기능을 한다. 하지만 이 배치 파일의 인터프리터는 간단한 제어 요소들은 인식할 수 있으나 강력한 스크립팅 환경을 제공하지는 못한다. 마이크로소프트사에서 제공하는 Window Scripting Host(WSH)는 VBScript, JScript, 또는 다른 스크립트 언어로 작성된 스크립트 파일을 해석하기 위한 런타임 엔진처럼 작동하는 환경이며 Window 운영체제 표준의 일부이다. 따라서 WSH를 이용하여 거의 대부분의 윈도우 기반의 소프트웨어를 제어할 수 있게 되며, 본 연구에서는 Window 버전의 ADAMS/Car를 사용하였고, 스크립트에서는 현재 활성화된 소프트웨어에 키보드의 키 입력에 대해 기술된다. 따라서 ADAMS/Car의 GUI에서 단축키를 이용하여 설정 및 해석 방식을 스크립트로 기술할 수 있다. Fig.1은 GUI를 제어하기 위한 visual basic script 파일을 보여주고 있다.

```

set wshshell = wscript.createObject("wscript.shell")
wshshell.run "set /a? For=1"
wscript.sleep 1000
wshshell.execute "ADAMS/Car 13.0.0"
wscript.sleep 1000
wshshell.execute "mp"
wscript.sleep 100
wshshell.execute "mp"
wscript.sleep 100
wshshell.execute "g"
wscript.sleep 100
wshshell.execute "a"
wscript.sleep 100
wshshell.execute "C:\private.cad\assemblies.tb1\from_suspension_steering.asy"
wscript.sleep 100
wshshell.execute "(TAB)"
wshshell.execute "-"
wscript.sleep 10000
wshshell.execute "ba"
wscript.sleep 100
wshshell.execute "s"
wscript.sleep 100
wshshell.execute "p"
wscript.sleep 100
wshshell.execute "date/time"
wscript.sleep 100
wshshell.execute "(TAB)"
wscript.sleep 100
wshshell.execute "100"
wscript.sleep 100
wshshell.execute "(TAB)"
wscript.sleep 100
wshshell.execute "-100"
wscript.sleep 100
wshshell.execute "(TAB 4)"
wscript.sleep 100
wshshell.execute "-"
wscript.sleep 100
wshshell.execute "mp"
wscript.sleep 100
wshshell.execute "mp"
wscript.sleep 100
wshshell.execute "a"
wscript.sleep 100
wshshell.execute "-"
wscript.sleep 100
    
```

Fig. 1 Script for WSH

3. 최적화 모듈과의 접속

본 연구에서는 EMDIOS를 사용하여 최적화를 수행하였다. EMDIOS는 iDOT 센터에서 개발된 MDO(Multidisciplinary Design Optimization) 프레임워크이며, 최적화 모듈인 ADS, DOT와 근사 최적화 기법인 PQRSM등이 탑재되어 있으며, 추가적인 최적화 모듈의 탑재가 용이한 장점을 가지고 있다. Fig.2에서는 EMDIOS를 보여 주고 있다.

본 연구에서는 ADAMS/Car의 ASCII형식의 입출력 파일을 이용하여 설계변수의 수정 및 성능지수의 추출을 수행하며, ADAMS/Car의 설정 및 해석에 관해 기술한 스크립트를 실행시켜 EMDIOS와의 접속을 구성한다. Fig.3은 EMDIOS와 ADAMS/Car의 접속을 도식적으로 표현하고 있다.

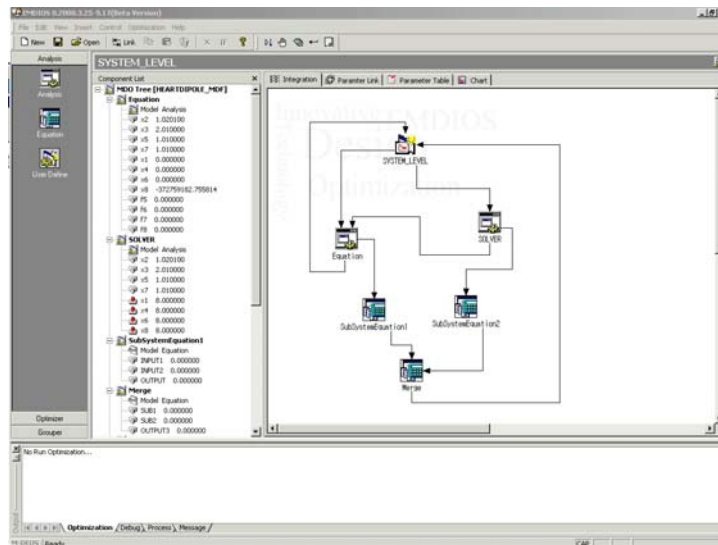


Fig.2 EMDIOS

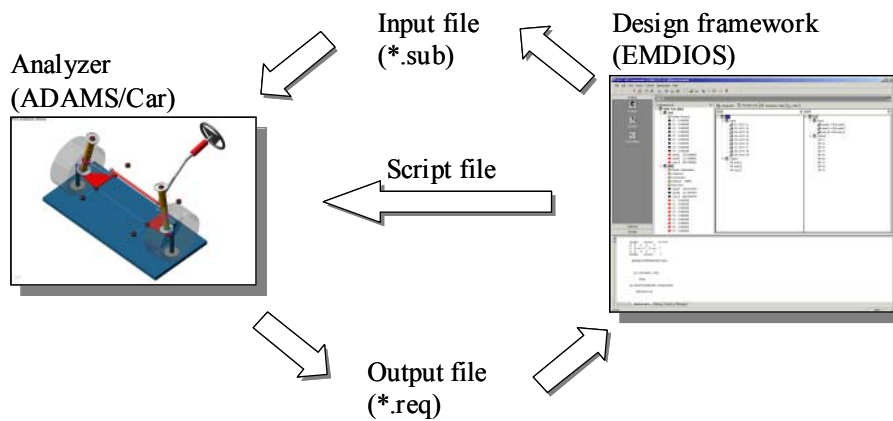


Fig. 3 Optimization procedure

본 연구에서는 ADAMS/Car를 이용하여 맥퍼슨형의 현가장치의 기구학적인 해석을 수행하여 최적화를 수행하고자 하며, 이를 위해서 설계변수로는 현가장치 모델의 하드포인트 위치를 설계변수로 설정하였으며, 목적함수와 구속조건을 구성하기 위하여 현가장치의 성능지수인 캐스터각, 캠버각, 토우각, 엔티-다이브, 롤-센터하이트를 추출하고자 한다. 따라서 ADAMS/Car에서의 현가장치 모델의 하드포인트 위치정보를 담고 있는 sub 파일을 파싱하여 설계변수를 수정하며, 성능지수의 추출을 위해서는 ADAMS/Car에서 제공하는 req 파일을 파싱한다. 여기서 파싱 과정은 미리 기술한 인자의 위치정보를 이용하여 해석이 수행될 때 마다 변경되는 인자의 값을 검출해 내는 과정이다. ADAMS/Car의 설정 및 해석과정은 VBScript를 이용하여 기술하였다. 따라서 EMDIOS에서는 최적화 과정 중 반복적인 해석을 위하여 확장자가 vbs 파일(VBScript 파일)을 마치 해석 프로그램인 것처럼 호출한다. Fig.4는 EMDIOS를 이용하여 ADAMS/Car를 실행하는 것을 보여 주고 있다.

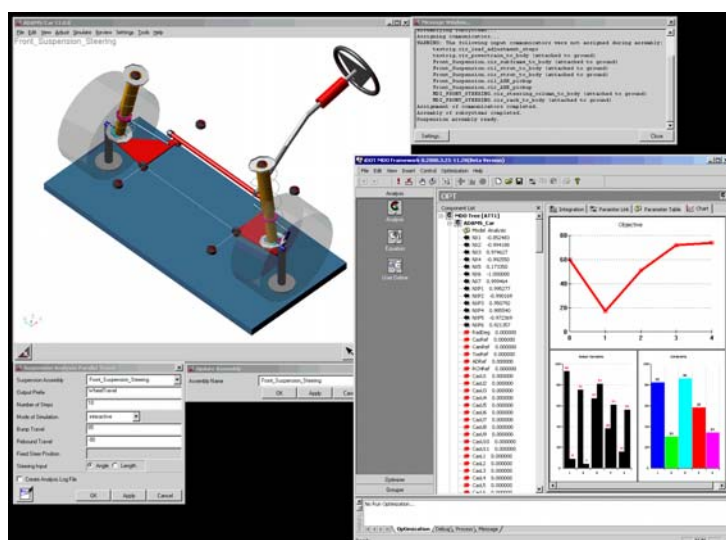


Fig.4 Execution of ADAMS/Car using script

4. ADAMS/Car를 이용한 예제

본 연구에서는 맥퍼슨 현가장치의 기구학적 특성 향상을 위한 최적설계를 통해서 EMDIOS와 ADAMS/Car의 접속을 수행하였다. 설계 요구 사항으로는 현가장치의 기구학적인 특성들에 대해서 parallel analysis를 통하여 얻은 결과값이 목표값에 근접하도록 하는 것이다.

4.1 최적설계 정식화

설계변수는 현가장치의 하드포인트 x, y, z 좌표(13개)로 Fig.5는 설계변수로 선정된 하드포인트의 위치를 보여주고 있으며, Table 1은 설계변수의 상/하한치와 초기치를 보여주고 있다.

목적함수는 현가장치 성능의 계산된 값과 목표값과의 차이를 최소화 하는 문제로 정의한다. 평가해야 할 성능지수가 여러 개 이므로 다중 목적함수로 구성하여야 한다. 다중 목적함수를 처리하는 방법으로 가중치 최소-최대법(weighted Min-Max method)을 사용하였다. 목적함수와 구속조건을 나타내는 수식은 식 (1)과 같다. 구속조건은 성능지수의 상/하한치 내에 포함되도록 하는 부등식 구속조건들로 구성된다.

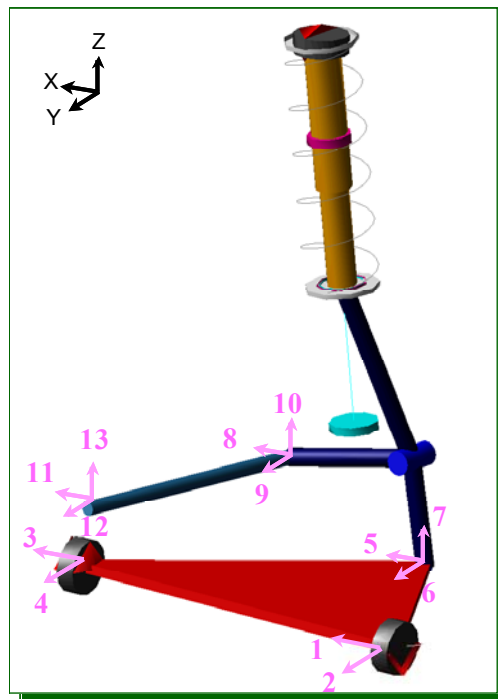


Fig. 5 Design variables

Table 1 Design variables (tabular data)

No		Configuration/point Description	Design variable setting			
			Lower bound	Initial point	Upper bound	
1	1	Lower wishbone front	X	-50.0	-50.0	0.0
	2	body mount	Y	380.0	385.0	385.0
2	3	Lower wishbone rear	X	210.0	230.0	240.0
	4	body mount	Y	380.0	385.0	385.0
3	5	Lower wishbone ball joint	X	-7.7	2.3	12.3
	6		Y	625.0	675.0	675.0
	7		Z	190.0	200.0	200.0
4	8	Tie rod outer ball joint	X	99.0	119.0	119.0
	9		Y	627.0	647.0	657.0
	10		Z	284.0	294.0	304.0
5	11	Tie rod inner ball joint	X	116.0	136.0	136.0
	12		Y	320.0	340.0	360.0
	13		Z	280.0	300.0	320.0

$$\begin{aligned}
 obj &= \sum_{i=1}^{N_{Char}} w_i \left(\max(\Delta R_1^i, \Delta R_2^i, \dots, \Delta R_{N_{Step}}^i) \right) \\
 g_i^H &= \max\left(\left(R_j / R_j^H\right) - 1\right)_i \\
 g_i^L &= \max\left(1 - \left(R_j / R_j^L\right)\right)_i \\
 \text{where } i &= 1..N_{Char}, \\
 j &= 1..N_{Step}, \\
 \Delta R_j^i &= \left(R_j^i - R_j^{i*}\right)^2
 \end{aligned} \tag{1}$$

4.2 최적화 수행 결과

본 연구에서 사용된 최적화 기법은 DOT에 포함된 Modified Method of Feasible Directions(MMFD)와 Sequential Quadratic Programming(SQP) 그리고, 자체 개발된 Progressive Quadratic Response Surface Modeling(PQRSM)을 사용하였다.

Table 2는 설계변수의 변화를 보여 주고 있다. Fig. 5를 참고로 설계변수는 총 13개로 선정 되었으며, 설계변수 1, 6, 8번에 대해서는 변화가 없는 것을 알 수 있다. 이는 최적화 시작단계부터 설계변수의 상하한치에서 시작하여 목적함수를 줄이거나 구속조건을 만족시키는 방

Table 2 Change of design variables after optimization

	Initial	Final (MMFD)	Final (SQP)	Final (PQRSM)
1	-50.0	-50.0	-50.0	-50.0
2	-385.0	-385.0	-381.8	-384.5
3	230.0	230.1	240.0	230.1
4	-385.0	-382.5	-380.0	-384.9
5	2.3	2.3	1.3	1.2
6	-675.0	-674.9	-675.0	-675.0
7	200.0	199.7	200.0	200.0
8	119.0	119.0	119.0	119.0
9	-647.0	-645.0	-657.0	-645.1
10	294.0	289.6	284.0	294.4
11	136.0	136.0	135.4	136.0
12	-340.0	-341.2	-354.6	-343.2
13	300.0	295.7	290.0	299.5

향으로의 설계변수 변화를 제한하고 있기 때문으로 판단된다. 따라서 보다 나은 최적화 진행을 위해서는 이러한 설계변수를 제거하거나 설계변수의 상하한치를 재설정하는 것이 바람직하다고 판단된다. 설계변수의 4, 13번의 경우 가장 많은 변화를 보이고 있으나 전반적으로 많은 변화를 보이고 있지는 않다. 이는 평가하고 있는 성능지수 가운데 토우각이 설계변수의 변동량에 대해 상당히 많은 변화를 보이며, 이로 인해 구속조건의 위배를 초래하게 되므로 토우각에 대한 구속조건의 만족을 위해서 설계변수의 변화량이 줄어드는 것으로 판단된다.

Fig.6은 목적함수의 변화를 각 반복회수마다 보여 주고 있으며, 최적설계 결과가 목적함수를 감소시키는 방향으로 성공적으로 진행되고 있음을 나타낸다. 국부최적화 방법인 MMFD와 SQP에 대해서는 각 최적화 반복회수에 따라 표시하였으며, 근사 최적화 기법인 PQRSM은 근사화 회수는 RSM call 회수에 따라 표시하였다. MMFD의 경우, 해석회수는 87번, 최적화 반복회수는 5번 수행되었으며, 목적함수는 약 20%정도의 감소를 보였다. SQP의 경우, 해석회수는 46번, 최적화 반복회수는 3번 수행되었으며, 목적함수는 약 31%의 감소를 보였다. PQRSM은 해석회수는 163번, RSM call 회수는 6번 수행되었으나 근사화 조건을 만족하여 채택된 것은 3번으로 나타났다. 목적함수는 약 18%의 감소를 보였다. 위의 결과를 토대로 볼 때, SQP가 본 문제에 대해서 가장 효율적인 알고리즘으로 판단된다.

Fig.7은 구속조건의 변화를 보여 주고 있으며, 초기 설계부터 구속조건을 모두 만족한 상태에서 최적화가 진행 되었으며, 세가지 방법 모두 구속 조건을 만족시키고 있다.

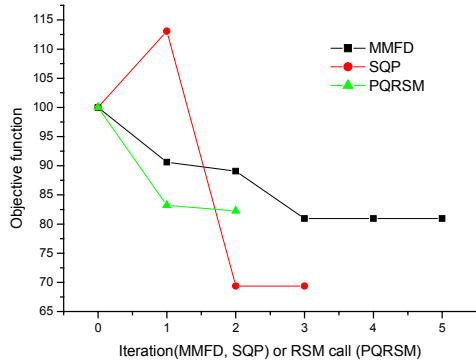


Fig.6 Convergence history

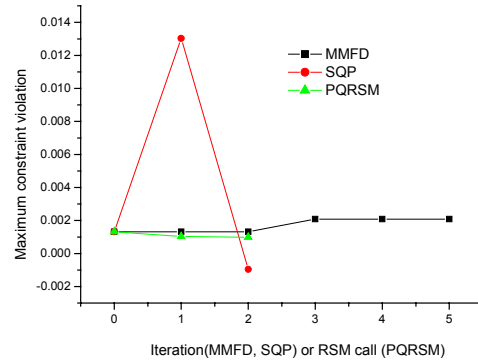


Fig.7 Maximum constraint violation

Fig.8은 토우각에 대한 변화를 보여준다. 초기 설계에서는 jounce(wheel travel : 0~90mm)되는 부분에서 목표값과의 차이가 크지만 최적화 수행 결과가 목표값에 상당히 근접해 있음을 알 수 있다. Fig.9는 캐스터 각에 대한 변화를 보여준다. 최적 설계 결과가 MMFD를 제외한 SQP, PQRSM의 경우, 초기 설계 보다 전반적으로 위쪽으로 이동하여 목표값에 근접하고 있음을 알 수 있다. Fig.10은 앤티-다이브의 변화를 보여 주고 있다. 초기 설계에서는 rebound(wheel travel -85 ~ 0mm) 시 목표값과 많은 차이를 보이고 있지만, MMFD나 PQRSM의 경우, 그 지점에서의 차이를 줄이고 있음을 알 수 있다. 하지만 jounce 시 초기 설계보다 목표값과의 차이가 증가되었음을 알 수 있다. 이는 앤티-다이브 곡선의 변화가 평행 이동하는 경향이 아니라 가운데 점을 중심으로 회전하는 경향을 보이므로 인해 목표값과의 최대값만을 감소시키는 최적화 수식으로는 전체적인 차이를 줄이는데 한계가 있음을 알려준다. Fig.11은 롤-센터하이트의 변화를 보여 주고 있다. 롤-센터하이트의 경우, 초기 설계가 주어진 상한치에 근접해 있어 설계 변화에 많은 제한이 있어 별다른 변화를 보이고 있지 않다. Fig.12는 캠버각에 대한 변화를 보여 주고 있다. 캠버각의 경우, 별다른 변화를 보여 주고 있지 않다. 이는 캠버각에 많은 영향을 미치는 설계변수인 설계변수 2번(Lower wishbone front body mount, Y좌표)과 4번(Lower wishbone rear body mount, Y좌표)의 변화가 토우각에도 상당한 영향을 미치기 때문에 두 성능지수의 상충효과로 토우각의 변화는 크지만 상대적으로 캠버각의 변화는 미미한 것으로 판단된다.

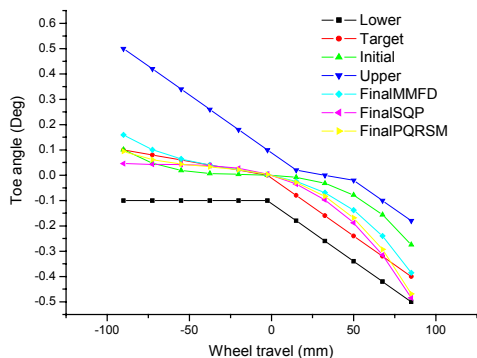


Fig.8 Toe angle
(Optimal result)

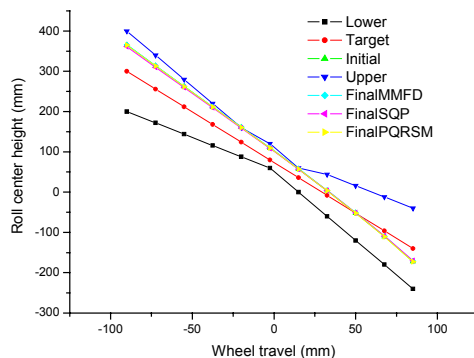


Fig.11 Roll center height
(Optimal result)

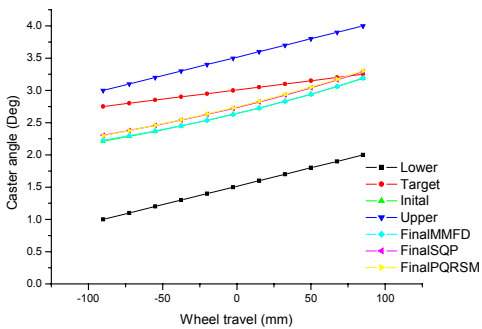


Fig.9 Caster angle
(Optimal result)

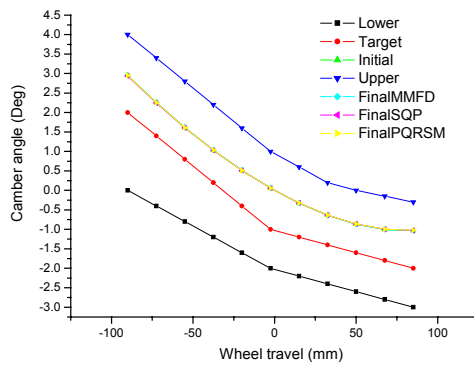


Fig.12 Camber angle
(Optimal result)

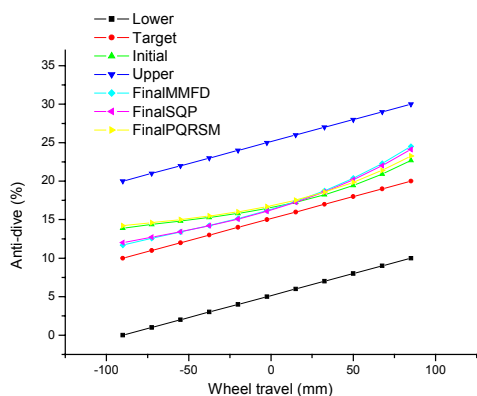


Fig.10 Anti-dive
(Optimal result)

5. 결론

본 연구에서는 MDO 프레임웍인 EMDIOS로 ADAMS/Car의 현가장치 해석모듈을 이용한 최적설계를 수행하였다. 이를 위해 EMDIOS와 ADAMS/Car와의 접속을 수행하였으며, 이 과정에서 WSH를 사용하여 ADAMS/Car의 실행 및 설정 변경 등을 이용하였다. 최적화 방법으로는 국부 최적화 기법인 MMFD와 SQP, 그리고 근사 최적화 방법인 PQRS를 사용하였고, 그 결과를 비교 분석하였다.

후기

본 연구는 한국과학재단 지정 최적설계신기술연구센터의 연구비 지원으로 수행되었습니다.

Reference

- [1] ADAMS 11.0 User manual
- [2] D. -I. J. Reimpell, D. -I. H. Stoll, 1996, *The Automotive Chassis: Engineering Principles*, Society of Automotive Engineers, Inc.
- [3] Dino Esposito, 1999, *Professional Visual C++ Windows Shell Programming*, Wrox
- [4] Herbert E. Ellinger, Richard B. Hathaway, 1989, *Automotive Suspension and Steering Theory and Service*, Prentice hall, Inc.
- [5] DOT Users Manul, version 5.0, Vanderplaats Research & Development, Inc.
- [6] 홍경진, 2001, “반응표면 모델링을 이용한 신뢰영역 관리 순차적 근사 최적화”, 박사학위 논문