

11장 기호 수학





1. 기호개체와 기호식
2. 기호식의 형태 변환
3. 대수방정식의 풀이
4. 미분
5. 적분
6. 상미분방정식의 풀이
7. 기호식의 그래프 그리기
8. 기호식에 의한 수치 계산
9. MATLAB 응용 예제

Symbolic Math의 특징

특징

- ❖ MATLAB을 가지고 이루어진 모든 수학연산은 수치연산
 - ▶ 연산은 수, 수치 값으로 할당 된 변수들이 포함된 수치 식으로 수행
 - ▶ $1/4 = 0.2500$, $1/3 = 0.3333$
- ❖ 수학, 과학, 공학에서 많은 응용들은 기호연산(Symbolic Operation)을 요구
 - ▶ 기호연산은 기호변수들을 포함하는 식으로 이루어짐
 - ▶ 예로 방정식, 미분, 적분이 있음
- ❖ 기호연산은 Symbolic Math Toolbox가 설치되어 있을 때 MATLAB에서 수행 할 수 있음
 - ▶ Symbolic Math Toolbox는 기호연산의 수행에 사용되는 MATLAB 함수들의 집합체
- ❖ 기호연산을 위한 출발점은 기호개체(Symbolic Object)

Symbolic Math의 장점

장점

- ❖ 문제를 간단하게 만들 수 있음
- ❖ 어떤 수치에서 오류가 있는지 정확한 솔루션을 제공
- ❖ 수치 해석이 가능한지 가능하지 않은지 판단 할 수 있음
- ❖ 문제와 그 해결책을 쉽게 이해 할 수 있음



11.1

기호개체와 기호식

기호개체와 기호식

특징

- ❖ 기호개체는 아직 할당되지 않은 변수, 수, 혹은 기호변수와 수들로 만들어진 기호식
- ❖ 기호식은 한 개 이상의 기호개체를 포함한 수학적
 - ▶ 입력할 때 기호식은 일반적인 수치적인 표현 식과 같이 보임
 - ▶ 기호식은 기호개체들을 포함하고 있으므로, 기호연산으로 수행

기호개체와 기호식

11.1.1 기호개체의 생성

- ❖ 기호개체는 변수 혹은 수
- ❖ 기호개체는 `sym`이나 `syms` 명령어를 이용하여 생성
 - ▶ 한 개 의 기호개체는 `sym`, 그 이상의 기호개체는 `syms`

```
Object_name = sym('string')
```

- ❖ 기호개체인 문자열(string)이 `object_name`에 할당
- ❖ 문자열이 될 수 있는 것
 - ▶ 한 개의 글자 또는 공백 없는 여러 글자들의 조합 ('a', 'x', 'yad')
 - ▶ 글자로 시작하는 공백 없는 글자들의 조합 ('xh12', 'r2d2')
 - ▶ 한 개의 수 ('15', '4')

기호개체와 기호식

11.1.1 기호개체의 생성

- ❖ 문자열이 한 개의 글자, 여러 글자들의 조합, 또는 글자들과 숫자들 조합에서는 기호개체는 기호 변수
- ❖ 필수적인 것은 아니지만 개체 이름은 문자열과 같게 하는 것이 편리
 - ▶ a, bb, x는 다음에 의해 기호변수로 정의

```
>> a = sym('a')
```

```
a =  
a
```

```
>> bb = sym('bb')
```

```
bb =  
bb
```

```
>> x = sym('x');
```

기호개체 a를 생성하여 a에 할당

기호개체의 출력은 들여쓰기가 되지 않음

세미콜론이 명령어 끝에 입력되므로
기호변수 x를 생성하지만 출력 하지 않음

기호개체와 기호식

11.1.1 기호개체의 생성

- ❖ 기호개체의 이름을 변수 이름과 다르게 할 수도 있음
 - ▶ 기호개체 gamma는 개체 이름을 다르게 설정

```
>> g = sym('gamma')
```

```
g =  
gamma
```

기호개체는 gamma이며 개체의 이름은 g

기호개체와 기호식

11.1.1 기호개체의 생성

- ❖ 기호개체는 수일 수도 있음
- ❖ 수는 문자열로 입력할 필요가 없음
 - ▶ 5와 7로 기호개체를 생성하고 기호개체들을 변수 c와 d에 할당

```
>> c = sym(5)
```

```
c =  
5
```

```
>> d = sym(7)
```

```
d =  
7
```

수 5로부터 기호개체를 생성하여 c에
할당

기호개체의 출력은 들여쓰기 되지 않음

기호개체와 기호식

11.1.1 기호개체의 생성

- ❖ 기호개체를 생성하고 세미콜론을 명령어 끝에 붙이지 않으면, MATLAB은 다음 두 줄에 걸쳐 개체의 이름과 개체 자체를 출력
- ❖ 기호개체의 출력은 해당 줄의 첫째 칸부터 시작되며 수치변수의 출력과는 달리 들여쓰기가 되지 않음
 - ▶ 수치변수 출력

```
>> e = 13
```

```
e =  
13
```

13이 e(수치변수)에 할당

수치변수의 값은 들여쓰기로 출력

기호개체와 기호식

11.1.1 기호개체의 생성

- ❖ 여러 개의 기호변수들은 다음 형식의 `syms` 명령어를 사용하여 한 개의 명령어로 생성 될 수 있음

```
syms variable_name variable_name variable_name
```

- ❖ 명령어는 기호변수와 동일한 이름을 가지는 기호개체를 생성
 - ▶ 세 변수 `y`, `z`, `d`를 하나의 명령어로 기호변수로 생성

```
>> syms y z d  
>> y
```

```
y =  
y
```

`syms` 명령어로 생성된 변수들은 자동으로 출력 되지 않음

변수 이름을 입력하면 변수가 생성되었음을 볼 수 있음

기호개체와 기호식

11.1.2 기호식의 생성

- ❖ 기호식은 기호변수들에 의해 작성된 수학식
- ❖ 기호변수들이 생성되면 이 변수들을 이용하여 기호식을 생성
- ❖ 기호식은 기호개체
 - ▶ 기호식을 생성하기 위한 형식

```
>> syms a b c x y  
>> f = a*x^2+b*x+c
```

```
f =  
a*x^2+b*x+c
```

a, b, c, x, y 를 기호변수로 정의
기호식 $ax^2 + bx + c$ 를 생성 f에 할당

기호식의 출력은 들여쓰기가 되지 않음

기호개체와 기호식

11.1.2 기호식의 생성

- ❖ 실행 가능한 수학연산(덧셈, 뺄셈, 곱셈, 나눗셈)이 포함된 기호식이 입력되면, MATLAB은 식의 생성과 함께 연산을 수행

```
>> g = 2*a/3 + 4*a/7 - 6.5*x +  
      4*5/3 - 1.5
```

```
g =  
(26*a)/21 - (37*x)/6 + 31/6
```

$\frac{2a}{3} + \frac{4a}{7} - 6.5x + \frac{x}{3} + 4.\frac{5}{3} - 1.5$ 가 입력

$\frac{26a}{21} - \frac{37x}{6} + \frac{31}{6}$ 이 출력

모든 계산이 수치적인 근사치 없이 정확
하게 수행된 점에 유의

기호개체와 기호식

11.1.2 기호식의 생성

❖ 정확한 계산과 근사적인 계산 사이의 차이를 비교

▶ 같은 수학연산을 한 번은 기호변수로 하고 한 번은 수치변수로 수행

```
>> a = sym(3); b = sym(5);  
>> e = b/a + sqrt(2)
```

```
e =  
2^(1/2) + 5/3
```

```
>> c = 3; d = 5;  
>> f = d/c + sqrt(2)
```

```
f =  
3.0809
```

a와 b를 각각 기호 3과 5로 정의
a와 b가 포함된 식을 생성
e의 정확한 값이 기호개체로 출력

c와 d를 각각 수치 3과 5로 정의
c와 d가 포함된 식을 생성
f의 근사 값이 수로 출력 (들여쓰기)

기호개체와 기호식

11.1.2 기호식의 생성

- ❖ 생성되는 식은 기호개체와 수치변수 모두를 포함 할 수 있음
 - ▶ 식이 하나의 기호개체(혹은 여러 개)를 포함하게 되면, 모든 수학연산이 정확하게 수행

>> $g = d/a + \text{sqrt}(2)$

$g =$
 $2^{(1/2)} + 5/3$

식에서 c 가 a 로 바뀌면, 처음 예제와 같이
결과가 정확하게 나옴

기호개체와 기호식

11.1.3 findsym 명령어와 기본 설정 기호변수

- ❖ 기호개체는 변수 혹은 수 일수 있음

`findsym(S)` 혹은 `findsym(S, n)`

- ❖ `findsym(S)` 명령어는 알파벳 순서대로 기호식 S에 있는 모든 기호변수들의 이름을 출력
- ❖ `findsym(S, n)` 명령어는 기본 설정된 순서에 따라 기호식 S에 있는 n개의 기호변수를 출력
 - ▶ 한 글자로 이루어진 기호변수의 경우, 기본 설정 순서는 x에서 시작하여 x에 가까운 글자 순을 따름
 - ▶ x로부터 똑같이 떨어져 있는 두 글자의 경우, 알파벳 순서에 따라 x 뒤에 오는 글자가 우선 (w보다 y가, v보다 z가 우선)
 - ▶ 기호식에서 기본 설정된 기호변수는 기본 설정 순서의 첫 번째 변수

기호개체와 기호식

11.1.3 findsym 명령어와 기본 설정 기호변수

❖ 식 S에서 기본으로 설정된 기호변수는 명령어 `findsym(S, 1)`의 실행에 의해 확인 할 수 있음

```
>> sym x h w y d t
>> S = h*x^2 + d*y^2 + t*w^2
S =
h*x^2 + d*y^2 + t*w^2

>> findsym(S)
ans =
d, h, t, w, x, y

>> findsym(S, 5)
ans =
x, y, w, t, h

>> findsym(S, 1)
ans =
x
```

x, h, w, d, t를 기호변수로 정의
기호식 S를 생성

`findsym(S)` 명령어를 이용
기호변수들이 알파벳 순서대로 출력

`findsym(S, n)` 명령어를 이용 (n = 5)
5개의 기호변수들이 기본 설정 순서대로
출력

`findsym(S, n)` 명령어를 이용 (n = 1)
기본 설정된 기호변수가 출력



11.2

기호식의 형태 변환

기호식의 형태 변환

특징

- ❖ 기호식은 사용자에게 의해 생성되거나 기호연산의 결과로 MATLAB에 의해 생성
- ❖ MATLAB에 의해 생성된 식은 가장 간단한 형태나 사용자가 선호하는 형태가 아닐 수도 있음
- ❖ 기존 기호식의 형태는 같은 지수(power)의 항끼리 모으고, 곱을 전개하고 인수분해하며, 수학 및 삼각함수 항등식과 다른 많은 연산들을 이용하여 바꿀 수 있음
- ❖ 하위 절에서 기존 기호식의 형태를 바꾸는 데 사용 할 수 있는 여러 명령어들에 대해 설명

기호식의 형태 변환

11.2.1 collect, expand, factor 명령어

- ❖ collect, expand, factor 명령어는 각각 자신의 이름이 암시하는 수학 연산을 수행하는데 사용
- ❖ collect 명령어
 - ▶ 같은 지수의 변수를 가지는 기호식에서 항들을 모음
 - ▶ 새 수식에서 항들은 지수의 내림차순으로 식을 돌려줌

`collect(S)` 혹은 `collect(S, variable_name)`

- ❖ `collect(S)` 형식은 기호식에 오직 한 개의 기호변수가 있을 때 효과적
- ❖ 식이 두 개 이상의 변수를 가지고 있으면, MATLAB은 먼저 한 변수의 항들을 모은 다음, 두 번째 변수의 항들을 모으고 계속해서 나머지 변수들에 대해 진행
- ❖ 변수들의 순서는 MATLAB에 의해 결정

기호식의 형태 변환

11.2.1 collect, expand, factor 명령어

- ❖ 정확 collect(S, variable_name) 형식의 명령어를 사용하여 첫 번째 변수를 지정 할 수 있음

```
>> syms x y  
>> S = (x^2 + x - exp(x)) + (x + 3)
```

```
S =  
(x + 3) * (x - exp(x) + x^2)
```

```
>> F = collect(S)
```

```
F =  
X^3 + 4 * x^2 + (3 - exp(x)) * x -  
3 * exp(x)
```

x와 y를 기호변수로 정의
기호식 $(x + 3)(x - e^x + x^2)$ 생성, S 할당

collect 명령어를 이용
MATLAB의 식
 $x^3 + 4x^2 + (3 - e^x)x - 3e^x$ 을 돌려줌

기호식의 형태 변환

11.2.1 collect, expand, factor 명령어

```
>> T = (2 * x^2 + y^2) * (x + y^2 + 3)
```

```
T =
```

```
(2 * x^2 + y^2) * (y^2 + x + 3)
```

```
>> G = collect(T)
```

```
G =
```

```
2 * x^3 + (2 * y^2 + 6) * x^2 +  
y^2 * x + y^2 * (y^2 + 3)
```

```
>> H = collect(T, y)
```

```
H =
```

```
y^4 + (2 * x^2 + x + 3) * y^2 + 2 * x^2  
* (x + 3)
```

기호식 $T (2x^2 + y^2)(y^2 + x + 3)$ 을 생성

collect(T) 명령어를 이용
MATLAB이 식

$2x^3 + (2y^2 + 6)x^2 + y^2x + y^2(y^2 + 3)$
을 돌려줌

collect(T, y) 명령어를 이용
MATLAB이 식

$y^4 + (2x^2 + x + 3)y^2 + 2x^2(x + 3)$
을 돌려줌

❖ collect(T)를 사용, 출력 식은 x의 내림차순에서 쓰였지만, collect(T, y)를 사용, 출력 식은 y의 내림차순에서 쓰였다는 점에 유의

기호식의 형태 변환

11.2.1 collect, expand, factor 명령어

❖ expand 명령어

- ▶ 덧셈(항들 중에서 최소한 한 개 이상의 항이 사용)이 포함되어 항들의 곱셈을 수행
- ▶ 덧셈이 포함된 해당 항들을 전개하기 위해 삼각함수 항등식과 지수 및 로그법칙을 이용

expand(S)

```
>> sym a x y
>> S = (x + 5) * (x - a) * (x + 4)
S =
-(a - x) * (x + 4) * (x + 5)
>> T = expand(S)
T =
20 * x - 20 * a - 9 * a * x * a *
x^2 + 9 * x^2 + x^3
>> expand(sin(x - y))
ans =
cos(y) * sin(x) - cos(x) * sin(y)
```

a, x, y를 기호변수로 정의
기호식 $(x + 5)(x - a)(x + 4)$ 를 생성
S에 할당

expand 명령어를 이용
MATLAB이 식
 $20x - 20a - 9ax - ax^2 + 9x^2 + x^3$
를 돌려줌

expand 명령어를 이용 $\sin(x - y)$ 전개
MATLAB이 삼각항등식을 이용하여 전개

기호식의 형태 변환

11.2.1 collect, expand, factor 명령어

❖ factor 명령어

▶ 더 낮은 차수의 다항식들의 곱으로 다항식을 변환

factor(S)

```
>> sym x
>> S = x^3 + 4 * x^2 - 11 * x - 30
S =
x^3 + 4 * x^2 - 11 * x - 30
>> factor(S)
ans=
(x + 5) * (x - 3) * (x + 2)
```

X를 기호변수로 정의
기 호 식 $x^3 + 4x^2 - 11x - 30$ 을 생 성
S를 할당

factor 명령어를 이용

MATLAB이 식
 $(x + 5)(x - 3)(x + 2)$
을 돌려줌

기호식의 형태 변환

11.2.2 simplify와 simple 명령어

- ❖ simplify와 simple 명령어는 모두 식의 형태를 간단하게 하기 위한 일반적인 도구
- ❖ simplify 명령어는 내장된 단순화 공식을 이용하여 원래 식보다 더 간단한 형태의 식을 만듦
- ❖ simple 명령어는 최소 개수의 글자들을 가진 식의 형태를 만들도록 프로그래밍됨
- ❖ 최소 글자 수를 가진 형태가 가장 간단하다는 보장은 없지만, 실제로 많이 쓰임

기호식의 형태 변환

11.2.2 simplify와 simple 명령어

❖ simplify 명령어

- ▶ simplify 명령어는 수학연산(덧셈, 곱셈, 분수 법칙, 거듭제곱, 로그 등)과 함수, 삼각함수 등을 이용하여 더 간단한 식의 형태로 만들어짐

simplify(S)

```
>> syms x y
>> S = (x^2 + 5 * x + 6) / (x + 2)
S =
(x^2 + 5 * x + 6) / (x + 2)
>> SA = simplify(S)
SA =
x + 3
>> simplify((x + y) / (1/x + 1/y))
ans =
x * y
```

x, y를 기호변수로 정의
기호식 $(x^2 + 5x + 6)/(x + 2)$ 를 생성
S에 할당

S의 간소화를 위해 simplify 명령어 이용

MATLAB이 $(x + 3)$ 으로 간단하게 함
 $(x + y) / \left(\frac{1}{x} + \frac{1}{y}\right)$ 을 간단하게 함

MATLAB이 xy 로 식을 간단하게 함

기호식의 형태 변환

11.2.2 simplify와 simple 명령어

❖ simple 명령어

- ▶ simple 명령어는 글자 수를 최소로 하는 식의 형태, 가장 간단한 식
- ▶ 세 가지 형식을 가짐

$F = \text{simple}(S)$

❖ S의 가장 짧은 형태가 F에 할당

$\text{simple}(S)$

❖ 모든 단순화 결과들이 출력, 가장 짧은 것은 ans에 할당

$[F \text{ how}] = \text{simple}(S)$

❖ S의 가장 짧은 형태는 F에 할당, 단순화 방법의 이름(문자열)은 how에 할당

기호식의 형태 변환

11.2.2 simplify와 simple 명령어

```
>> syms x  
>> S = (x^3 - 4 * x^2 + 16 * x) /  
        (x^3 + 64)
```

```
S =  
(x^3 - 4 * x^2 + 16 * x) / (x^3 +  
64)
```

```
>> F = simple(S)
```

```
F =  
x / (x + 4)
```

```
>> [G how] = simple(S)
```

```
G =  
x / (x + 4)  
how =  
simplify
```

x를 기호변수로 정의

기호식 $\frac{x^3-4x^2+16x}{x^3+64}$ 를 생성, S에 할당

F = simple(S) 명령어를 사용하여 간략화

S의 가장 간단한 형태인 $x / (x + 4)$ 가
F에 할당

[G how] = simple(S) 명령어를 사용

S의 가장 간단한 형태인 $x / (x + 4)$ 가
G에 할당

단어 simplify가 G에 할당
가장 짧은 형태가 simplify 명령어로
얻어졌음을 의미

기호식의 형태 변환

11.2.2 simplify와 simple 명령어

- ❖ `simple(S)` 형식의 명령어 이용은 결과의 출력이 길기 때문에 여기서는 설명되지 않음
- ❖ MATLAB은 10 가지의 다른 단순화 결과를 출력하며 이 중에서 가장 짧은 형태를 `ans`에 할당
- ❖ 이 명령어를 실행하고 결과 출력을 조사해보기 바람

기호식의 형태 변환

11.2.3 pretty 명령어

❖ pretty 명령어

- ▶ pretty 명령어는 일반적으로 수학식을 쓰는 형태와 닮은 형태로 기호식을 출력

pretty(S)

```
>> syms a b c x  
>> S = sqrt(a * x^2 + b * x + c)
```

```
S =  
(a * x^2 + b * x + c)
```

```
>> pretty(S)
```

```
 2    1 / 2  
(a x + b x + c)
```

a, b, c, x를 기호변수로 정의
기호식 $\sqrt{ax^2 + bx + c}$ 를 생성, S에 할당

pretty 명령어가 기호식을 수학형식으로
출력



11.3

대수방적식의 풀이

대수방정식의 풀이

11.3 단일방정식의 풀이

- ❖ 단일 대수 방정식은 단일 변수에 대해 풀 수 있고, 연립방정식은 solve 함수를 가진 여러 변수에 대해 풀 수 있음
- ❖ 단일 방정식의 풀이
 - ▶ 대수방정식은 한 개 이상의 기호변수를 가질 수 있음
 - ▶ 방정식이 한 개의 변수를 가지면, 해는 수치 값을 가짐
 - ▶ 방정식이 여러 개의 기호변수를 갖는 경우, 임의의 변수에 대한 해를 나머지 변수들에 의해 구할 수 있음
 - ▶ 해는 solve 명령어를 이용하여 구함
- ❖ solve 명령어의 기본 형식

`h = solve(eq)` 혹은 `h = solve(eq, var)`

대수방정식의 풀이

11.3 단일방정식의 풀이

- ❖ eq는 이미 생성된 기호식의 이름이거나 직접 입력할 식, 이전에 생성된 기호식 S를 인자 eq 대신 입력하거나 = 부호가 포함되지 않은 식을 eq 대신 입력하면, MATLAB은 식 $eq = 0$ 을 품
- ❖ $f(x) = g(x)$ 형태의 식은 eq에 대한 문자열로서 식(=부호를 포함한)을 입력하여 풀 수 있음
- ❖ 사용자가 solve(eq)로 입력하면, 해는 변수 ans에 할당
- ❖ 식이 두 개 이상의 해를 갖는 경우, 출력 h는 각 원소가 해인 기호 열 벡터가 됨
- ❖ 벡터의 원소는 기호개체이며, 기호개체 배열이 출력될 때 각 행은 대괄호([]) 안에 표시

대수방정식의 풀이

11.3 단일방정식의 풀이

❖ solve 명령어의 사용 예

```
>> syms a b x y z
>> f = solve(exp(2 * z) - 5
h =
Log(5) / 2

>> S = x^2 - x - 6
S =
x^2 - x - 6

>> k = solve(S)
k =
-2
3
```

a, b, x, y, z를 기호변수로 정의
solve 명령어를 이용 $e^{2x} - 5 = 0$ 을 계산

해가 h에 할당

기호식 $x^2 - x - 6$ 을 생성, S에 할당

solve(S) 명령어를 이용 $x^2 - x - 6$ 을 계산

식은 두 개의 해를 가짐
해는 기호개체를 가진 열벡터인 k에 할당

대수방정식의 풀이

11.3 단일방정식의 풀이

```
>> solve('cos(2 * y) + 3 * sin(y) = 2')
```

```
ans =
```

```
pi / 2
```

```
pi / 6
```

```
(5 * pi) / 6
```

```
>> T = a * x^2 + 5 * b * x + 20
```

```
T =
```

```
a * x^2 + 5 * b * x + 20
```

```
>> solve(T)
```

```
ans =
```

```
-(5 * b + 5^(1 / 2) * (5 * b^2 - 16 *  
a)^(1 / 2)) / (2 * a)
```

```
-(5 * b - 5^(1 / 2) * (5 * b^2 - 16 *  
a)^(1 / 2)) / (2 * a)
```

```
>> M = solve(T, a)
```

```
M =
```

```
-(5 * b * x + 20) / x^2
```

solve 명령어를 이용하여 식 $\cos(2y) + 3\sin(y) = 2$ 풀음

(식은 명령어에 문자열로 입력)

해가 ans에 할당

기호식 $ax^2+5bx+20$ 을 생성, T에 할당
solve(S) 명령어를 이용, T=0을 계산

기본 설정된 변수인 x에 대해 식 T=0을
계산

solve(eq, var) 명령어를 이용하여 T=0을
계산

변수 a에 대해 식 T=0을 계산

대수방정식의 풀이

11.3 단일방정식의 풀이

- ❖ 식에서 변수들을 기호개체로 먼저 생성하지 않고 solve 명령어를 이용하여 식을 문자열로 입력하는 것이 가능
- ❖ 식이 둘 이상의 변수를 가져서 변수들이 해에 포함된다면, 이 변수들은 독립 기호개체로서 존재하지 않음

```
>> ts = solve('4 * t * h^2 + 20 * t  
- 5 * g')
```

```
ts =  
(5 * g) / (4 * h^2 + 20)
```

식 $4th^2 + 20t - 5g$ 가 solve 명령어 입력

변수 t, h, g를 solve 명령어에 식을 입력하기 전에 기호변수로 생성하지 않음

MATLAB이 t에 대해 $4th^2 + 20t - 5g = 0$ 을 계산

대수방정식의 풀이

11.3 단일방정식의 풀이

❖ 식은 다른 변수에 대해 또한 풀 수도 있음

```
>> gs = solve('4 * t * h^2 + 20 * t  
- 5 * g', 'g')
```

```
gs =  
(4 * t * h^2) / 5 + 4 * t
```

앞에 방법과 동일하게 수행

대수방정식의 풀이

11.3 연립방정식의 풀이

- ❖ solve 명령어를 이용하여 연립방정식을 풀 수도 있음
- ❖ 식과 변수의 개수가 같으면, 수치 해를 얻음
- ❖ 변수의 개수가 식의 개수보다 많으면, 원하는 변수에 대한 해는 나머지 변수들로 표현된 기호 해가 됨
- ❖ 연립방정식은 한 개 또는 여러 개의 해를 가질 수 있음
 - ▶ 한 개의 해를 가진다면, 연립방정식이 풀리게 하는 각 변수는 한 개의 수치 값을 가짐
 - ▶ 둘 이상의 해를 갖는다면, 각 변수들은 여러 개의 값을 가질 수 있음
- ❖ solve 명령어의 기본 형식

output = solve(eq1, eq2, ..., eqn)

혹은

output = solve(eq1, eq2, ..., eqn, var1, var2, ..., varn)

대수방정식의 풀이

11.3 연립방정식의 풀이

- ❖ eq1, eq2, ..., eqn은 해를 구할 식으로 각 인자는 이전에 생성된 기호식의 이름이거나 문자열로 입력할 식이 될 수 있음
- ❖ 이전에 생성된 기호식 S가 입력될 때, 방정식은 $S = 0$, =부호가 포함되지 않은 문자열이 입력될 때, 방정식은 `expression = 0`, 부호가 포함된 식은 문자열로 입력되어야 함
- ❖ 첫 번째 형식에서, 방정식의 개수 n이 방정식에서 변수의 개수와 같으면, MATLAB은 모든 변수에 대해 수치 해를 줌, 변수의 개수가 방정식의 개수 n보다 많으면, MATLAB은 n개의 변수들의 해를 나머지 변수들에 의해 나타냄
- ❖ 해가 구해지는 변수들은 기본 설정된 순서에 따라 MATLAB에 의해 선택, 변수의 개수가 방정식의 개수 n보다 많으면, 사용자는 연립방정식을 어떤 변수들에 대해 풀 것인지 선택 할 수 있음

대수방정식의 풀이

11.3 연립방정식의 풀이

- ❖ solve 명령어에서 연립방정식에 대한 해인 output은 두 개의 형태를 가질 수 있음
 - ▶ 셀 배열, 구조체
- ❖ 셀 배열은 각 원소들이 하나의 배열 일 수 있는 형태의 배열
- ❖ 구조체는 필드라 불리는 원소들을 텍스트 필드 지정자로 주소 정을 하는 배열로, 구조체의 필드들은 서로 다른 크기와 종류의 배열들 일 수 있음
- ❖ 셀 배열이 solve 명령어의 출력에 사용될 때, 명령어의 형식 (방정식이 세 개인 경우)

```
[varA, varB, varC] = solve(eq1, eq2, eq3)
```

대수방정식의 풀이

11.3 연립방정식의 풀이

- ❖ 명령어가 실행되면, 해가 변수 `varA`, `varB`, `varC`에 할당되고 변수들이 할당된 해와 함께 출력
- ❖ 각 변수는 연립방정식이 한 개 혹은 여러 개의 해를 가지는지에 따라 한 개 또는 여러 개의 값을 열벡터로 가짐
- ❖ 사용자는 `varA`, `varB`, `varC`에 대해 임의의 이름을 선택할 수 있음
- ❖ MATLAB은 방정식에서 변수들에 대한 해를 알파벳 순서대로 할당, 예를 들면 풀려는 식의 변수가 `x`, `u`, `t` 라면 `t`에 대한 해는 `varA`에 `u`에 대한 해는 `varB`에, `x`에 대한 해는 `varC`에 할당

대수방정식의 풀이

11.3 연립방정식의 풀이

❖ 셀 배열이 출력에 사용되는 경우 solve 명령어를 사용하는 방법

```
>> syms x y t  
>> S = 10 * x + 12 * y + 16 * t;  
>> [xt yt] = solve(S, '5 * x - y =  
13 * t')
```

```
xt =  
2 * t  
yt =  
-3 * t
```

x, y, t를 기호변수로 정의
식 $10x + 12y + 16t$ 를 S에 할당

solve 명령어를 이용 연립방정식
 $10x + 12y + 16t = 0, 5x - y = 13t$ 을
계산

이름이 xt와 yt인 두 셀을 가진 셀 배열의
출력

x와 y에 대한 해가 각각 xt와 yt에 할당

x, y가 기본 설정순서의 첫 두 변수이므로
MATLAB은 연립방정식의 해를 t에 의해
풀었음을 유의

대수방정식의 풀이

11.3 연립방정식의 풀이

- ❖ solve 명령어의 두 번째 형식을 이용하여 연립방정식의 y 와 t 에 대한 해를 x 에 의해 구하는 예

```
>> [tx yx] = solve(S, '5 * x - y =  
13 * t', y, t)
```

```
tx =  
x / 2  
yx =  
-(3 * x) / 2
```

연립방정식을 풀 변수 y 와 t 가 입력

연립방정식에서 구한 변수들의 해는 알파벳 순서대로 출력변수에 할당

첫 번째 셀은 t 에 대한 해를, 두 번째 셀은 y 에 대한 해를 가짐

대수방정식의 풀이

11.3 연립방정식의 풀이

- ❖ 구조체가 solve 명령어의 출력에 사용될 때, 명령어의 형식 (방정식이 세 개인 경우)

```
AN = solve(eq1, eq2, eq3)
```

- ❖ AN은 구조체의 이름을 나타냄
- ❖ 명령어가 실행되면 해는 AN에 할당, MATLAB은 구조체의 이름과 구조체의 필드의 이름을 출력하고, 이것들은 방정식의 해를 어떤 변수에 대해 구했는지를 나타내는 그 변수들의 이름을 나타냄
- ❖ 각 필드의 크기와 종류는 필드 이름 옆에 출력
- ❖ 필드의 내용(변수에 대한 해)를 출력하기 위해서는 사용자는 필드 주소를 입력해야 함

대수방정식의 풀이

11.3 연립방정식의 풀이

❖ 이전 예에서 다룬 연립방정식을 구조체가 출력이 되도록 하여 다시 푸는 방법

```
>> syms x y t
>> S = 10 * x + 12 * y + 16 * t;
>> AN = solve(S, '5 * x - y = 13 * t')
AN =
    x : [1x1 sym]
    y : [1x1 sym]

>> AN.x
ans =
    2 * t

>> AN.y
ans =
   -3 * t
```

solve 명령어를 이용하여 연립방정식
 $10x + 12y + 16t = 0, 5x - y = 13t$ 를
계산

MATLAB이 구조체 AN의 이름과 구조체
의 필드 x와 y의 이름을 출력
(필드의 이름은 식의 해를 어떤 변수에 대
래 구했는지를 나타내는 변수 이름)

필드 x의 주소를 입력
필드의 내용(x에 대한 해) 출력

필드 y의 주소를 입력
필드의 내용(y에 대한 해) 출력

대수방정식의 풀이

11.3 연립방정식의 풀이

❖ 예제 11.1 원과 직선의 교차

- ▶ 반지름이 R 이고 중심이 점 $(2, 4)$ 인 $x - y$ 평면상의 원의 방정식은 $(x - 2)^2 + (y - 4)^2 = R^2$ 이다. 평면상의 직선 식은 $y = \frac{x}{2} + 1$ 이다. 직선과 원이 교차하는 지점의 좌표를 R 의 함수로 구하라.

대수방정식의 풀이

11.3 연립방정식의 풀이

❖ 예제 11.1 원과 직선의 교차 문제 풀이

```
>> syms x y R  
>> [xc, yc] = solve('(x - 2)^2 + (y - 4)^2 = R^2', 'y = x / 2 + 1')
```

```
xc =  
((4 * R^2) / 5 - 64 / 25)^(1 / 2) +  
14 / 5  
14 / 5 - ((4 * R^2) / 5 - 64 /  
25)^(1 / 2)  
yc =  
((4 * R^2) / 5 - 64 / 25)^(1 / 2) /  
2 + 12 / 5  
12 / 5 - ((4 * R^2) / 5 - 64 /  
25)^(1 / 2) / 2
```

두 방정식이 solve 명령어에 입력

셀 배열 출력

두 셀의 이름이 xc와 yc인 셀 배열로 출력
각 셀은 기호 열벡터로서 두 개의 해를
포함

대수방정식의 풀이

11.3 연립방정식의 풀이

❖ 예제 11.1 원과 직선의 교차 문제 풀이

```
>> COORD = solve('(x - 2)^2 +  
    (y - 4)^2 = R^2', 'y = x / 2 +  
    1')
```

COORD =

x : [2x1 sym]

y : [2x1 sym]

```
>> COORD.x
```

ans =

$((4 * R^2) / 5 - 64 / 25)^{(1 / 2)} +$
 $14 / 5$

$14 / 5 - ((4 * R^2) / 5 - 64 /$
 $25)^{(1 / 2)}$

두 번째 해는 구조체로 출력

대수방정식의 풀이

11.3 연립방정식의 풀이

❖ 예제 11.1 원과 직선의 교차 문제 풀이

```
>> COORD.y
```

```
ans =  
((4 * R^2) / 5 - 64 / 25)^(1 / 2) /  
2 + 12 / 5  
12 / 5 - ((4 * R^2) / 5 - 64 /  
25)^(1 / 2) / 2
```

구조체 출력

이름이 COORD이고 두 개의 필드 x, y를 가진 구조체로 출력

각 필드는 2x1의 기호벡터

필드 x의 주소를 입력
필드의 내용(x에 대한 해) 출력

필드 y의 주소를 입력
필드의 내용(y에 대한 해) 출력



11.4
미분

미분

11.4 미분

- ❖ 기호 미분은 diff 명령어로 수행 될 수 있음

diff(S) 혹은 diff(S, var)

- ❖ S는 이미 생성된 기호식의 이름 혹은 S 대신 입력될 수 있는 식이 될 수 있음
- ❖ diff(S) 명령어에서 식에 한 개의 기호변수가 포함된다면, 미분은 그 변수에 대해 수행, 식에 둘 이상의 변수가 포함되어 있으면, 미분은 기본 설정된 기호변수에 대해 수행
- ❖ diff(S, var) 명령어(여러 개의 기호변수가 포함된 식의 미분에 사용)에서 미분은 변수 var에 대해 수행
- ❖ 2차 또는 고차(n차) 미분은 diff(S, n) 혹은 diff(S, var) 명령어로 구함

미분

11.4 미분

❖ diff(S)와 diff(S, var)에 대한 예

```
>> syms x y t
>> S = exp(x^4)
>> diff(S)
ans =
4 * x^3 * exp(x^4)

>> diff((1 - 4 * x)^3)
ans =
-12 * (1 - 4 * x)^2

>> R = 5 * y^2 * cos(3 * t);
>> diff(R)
ans =
10 * y * cos(3 * t)
```

x, y, t 를 기호변수로 정의

식 e^{x^4} 을 S 에 할당

diff(S) 명령어를 이용하여 S 를 미분
답 $4x^3e^{x^4}$ 이 출력

Diff(S) 명령어를 이용하여 $(1 - 4x)^3$ 을
미분
답 $-12(1 - 4x)^2$ 이 출력

식 $5y^2\cos(3t)$ 를 R 에 할당

식 $5y^2\cos(3t)$ 를 R 에 할당
diff(R) 명령어를 사용하여 R 을 미분

MATLAB이 y (기본설정 기호변수)에 대해 R 을
미분
답 $10y\cos(3t)$

미분

11.4 미분

❖ diff(S)와 diff(S, var)에 대한 예

```
>> diff(R, t)
ans =
-15 * y^2 * sin(3 * t)
```

```
>> diff(S, 2)
ans =
12 * x^2 * exp(x^4) + 16 * x^6 *
exp(x^4)
```

diff(R, t) 명령어를 이용하여 t에 대해 R을 미분
답 $-15y^2\sin(3t)$ 가 출력

diff(S, 2)를 이용하여 S의 2차 미분을 얻음
답 $12x^2e^{x^4} + 16x^6e^{x^4}$ 이 출력

❖ 미분 할 식에 포함된 변수들을 기호개체로 먼저 생성하지 않고 식을 문자열로 직접 입력하여 diff 명령어를 사용 할 수 있음

❖ 그러나 미분 된 식에 있는 변수들은 독립적인 기호개체로 존재하지 않음



11.5

적분

적분

11.5 적분

- ❖ 기호 적분은 int 명령어로 수행 될 수 있음

int(S) 혹은 int(S, var)

- ❖ S는 이미 생성된 기호식의 이름 혹은 S 대신 입력될 수 있는 식이 될 수 있음
- ❖ int(S) 명령어에서 식에 한 개의 기호변수가 포함된다면, 적분은 그 변수에 대해 수행, 식에 둘 이상의 변수가 포함되어 있으면, 적분은 기본 설정된 기호변수에 대해 수행
- ❖ int(S, var) 명령어(여러 개의 기호변수를 가진 식의 적분에 사용)에서 적분은 변수 var에 대해 수행

적분

11.5 적분

❖ `int(S)`와 `int(S, var)`에 대한 예

```
>> syms x y t
>> S = 2 * cos(x) - 6 * x
>> int(S)
ans =
2 * sin(x) - 3 * x ^2

>> int(x * sin(x))
ans =
sin(x) - x * cos(x)

>> R = 5 * y^2 * cos(4 * t);
>> int(R)
ans =
(5 * y ^ cos(4 * t))/3

>> int(R, t)
ans =
(5 * y ^ 2 * sin(4 * t))/3
```

x, y, t 를 기호변수로 정의
식 $2\cos(x) - 6x$ 를 S 에 할당
`int(S)` 명령어를 이용하여 S 를 적분

답 $2\sin(x) - 3x^2$ 이 출력

`int(S)` 명령어를 이용하여 $x\sin(x)$ 를 적분
답 $\sin(x) - x\cos(x)$ 가 출력

식 $5y^2\cos(4t)$ 를 R 에 할당
`int(R)` 명령어를 사용하여 R 을 미분
MATLAB이 기본 설정된 변수 y 에 대해 R
을 적분하고, 답 $5y^2\cos(4t) / 3$ 을 출력

`int(R,t)` 명령어를 이용하여 t 에 대해 R 을
미분
답 $5y^2\sin(4t) / 4$ 가 출력

적분

11.5 적분

❖ 정적분

▶ 정적분을 위한 명령어의 형식은 다음과 같다.

```
int(S, a, b) 또는 int(S, var, a, b)
```

▶ 정적분 $\int_0^{\pi} (\sin y - 5y^2) dy$ 를 MATLAB으로 구하면 다음과 같다.

```
>> syms y  
>> int(sin(y) - 5*y^2, 0, pi)
```

```
ans =  
2 - (5 * pi ^ 3) / 3
```

적분

11.5 적분

- ❖ 적분할 식에 포함된 변수들로 기호개체를 먼저 생성하지 않고 직접 문자열에 식을 입력하여 `int` 명령어를 사용할 수도 있다. 그러나 적분된 식에 포함된 변수들은 독립적인 기호개체로 존재하지는 않는다.
- ❖ 적분은 가끔 어려울 수 있다. 닫힌 형태(closed form)의 해가 존재하지 않을 수도 있고, MATLAB도 찾지 못할 수도 있다. 이런 경우가 발생하면 MATLAB은 `int(S)` 와 메시지 'Explicit integral could not be found'를 보여준다.



11.6

상미분방정식의 풀이

상미분방정식의 풀이

11.6 상미분방정식의 풀이

- ❖ 11.6절의 목적은 미분방정식을 풀기 위한 MATLAB 사용 방법을 보임
- ❖ 상미분방정식(ODE)은 dsolve 명령어를 이용하여 기호형태로 구함
- ❖ dsolve 명령어는 단일 방정식이나 연립방정식을 푸는 데 사용됨
- ❖ 1차 ODE
 - ▶ 종속변수의 미분을 포함하는 식
 - ▶ 시간 t 가 독립변수이고 y 가 종속변수라면, 방정식은 다음과 같다.

$$\frac{dy}{dt} = f(t, y)$$

상미분방정식의 풀이

11.6 상미분방정식의 풀이

❖ 2차 ODE

- ▶ 종속변수(1차 미분도 포함 가능)의 2차 미분을 포함하는 식
- ▶ 시간 t 가 독립변수이고 y 가 종속변수라면, 방정식은 다음과 같다.

$$\frac{dy}{dt} = f\left(t, y, \frac{dy}{dt}\right)$$

- ▶ 해는 식을 만족하는 함수 $y=f(t)$ 임
- ▶ 일반해 또는 특수해 일 수 있음
- ▶ 일반해는 상수를 포함
- ▶ 특수해에서 상수는 해가 특정 초기조건이나 경계조건을 만족할 수 있도록 특정 수치 값을 구함
- ▶ 명령어 `dsolve`를 이용하여 일반해를 구할 수 있으며, 초기조건이나 경계조건이 주어지면 특수해를 구할 수 있음

상미분방정식의 풀이

11.6 상미분방정식의 풀이

❖ 일반해

- ▶ 일반해를 얻기 위한 dsolve 명령어의 형식은 다음과 같다.

```
dsolve('eq') 또는 dsolve('eq', 'var')
```

- ▶ eq는 풀고자 하는 방정식이며, 문자열(변수들이 기호개체일지라도)로 입력되어야 함
- ▶ 방정식의 변수들을 미리 기호개체로 생성할 필요는 없음
(기호개체가 생성되지 않는다면, 해에 포함된 변수들은 기호개체가 되지 않음)
- ▶ D를 제외한 어떤 글자(소문자 또는 대문자)도 종속변수로 사용 가능
- ▶ dsolve('eq') 명령어에서 독립 변수는 MATLAB에서 t(기본설정)로 가정
- ▶ dsolve('eq', 'var') 명령어에서 var(문자열) 대신 독립변수로 입력하여 정의 가능

상미분방정식의 풀이

11.6 상미분방정식의 풀이

- ▶ 식을 입력할 때 글자 D는 미분을 나타냄
- ▶ y 가 종속변수이고 t 가 독립변수이면, $Dy = \frac{dy}{dt}$ 를 나타냄
- ▶ 식 $\frac{dy}{dt} + 3y = 100$ 은 ' $Dy + 3*y = 100$ '으로 입력
- ▶ 2차 미분은 D2, 3차 미분은 D3 등과 같이 입력
- ▶ 식 $\frac{d^2y}{dt^2} + 3\frac{dy}{dt} + 5y = \sin(t)$ 은 ' $D2Y + 3*Dy + 5*y = \sin(t)$ '로 입력
- ▶ dsolve 명령어에 입력되는 ODE 방정식의 변수들은 미리 생성된 기호 변수일 필요 없음
- ▶ MATLAB은 해에서 적분상수로 C1, C2, C3 등을 사용

상미분방정식의 풀이

11.6 상미분방정식의 풀이

- ▶ 1차 ODE인 $\frac{dy}{dt} = 4t + 2y$ 의 일반해는 다음과 같이 얻어짐

```
>> dsolve('Dy=4*t+2*y')  
ans =  
C1 * exp(2*t) - 2*t - 1
```

답 $y = C_1 e^{2t} - 2t - 1$ 이 출력

- ▶ 2차 ODE인 식 $\frac{d^2x}{dt^2} + 2\frac{dx}{dt} + x = 0$

```
>> dsolve('D2x+2*Dx+x=0')  
ans =  
C1 / exp(t) - (C2*t) / exp(t)
```

답 $x = C_1 e^{-t} + C_2 t e^{-t}$ 이 출력

상미분방정식의 풀이

11.6 상미분방정식의 풀이

- ▶ 다음 예는 종속변수와 독립변수뿐만 아니라 기호변수가 포함되는 미분 방정식들의 해를 설명한다.

```
>> dsolve('Ds=a*x^2')
```

```
ans =
```

```
a*t*x^2 + C1
```

```
>> dsolve('Ds=a*x^2', 'x')
```

```
ans =
```

```
(a*x^3)/3 + C1
```

```
>> dsolve('Ds=a*x^2', 'a')
```

```
ans =
```

```
(a^2*x^2)/2 + C2
```

독립 변수는 t(기본 설정)

MATLAB이 방정식 $\frac{ds}{dt} = ax^2$ 을 품
해 $s = ax^{2t} + C_1$ 이 출력

독립 변수를 x 로 정의

MATLAB이 방정식 $\frac{ds}{dx} = ax^2$ 을 품
해 $s = \frac{1}{3}ax^3 + C_1$ 이 출력

독립 변수를 a 로 정의

MATLAB이 방정식 $\frac{ds}{da} = ax^2$ 을 품
해 $s = \frac{1}{2}a^2x^2 + C_1$ 이 출력

상미분방정식의 풀이

11.6 상미분방정식의 풀이

❖ 특수해

- ▶ ODE의 특수해는 경계조건(혹은 초기 조건)이 주어지면 구할 수 있음
- ▶ 1차 방정식은 한 개의 조건이 필요하고, 2차 방정식은 두 개의 조건이 필요하며, n 차 방정식 역시 같은 조건으로 됨
- ▶ 특수해를 구하기 위한 `dsolve` 명령어의 형식은 다음과 같다.

1차 ODE : `dsolve('eq', 'cond1', 'var')`

고차 ODE : `dsolve('eq', 'cond1', 'cond2', ... , 'var')`

- ▶ 고차 미분방정식을 풀기 위해, 추가적인 경계조건들은 명령어에 입력되어야 함
- ▶ 조건의 개수가 방정식의 차수의 수보다 작으면, MATLAB은 적분 상수 (C_1, C_2, C_3 등)를 포함한 해를 돌려줌

상미분방정식의 풀이

11.6 상미분방정식의 풀이

- ▶ 경계 조건은 다음과 같이 문자열로 입력된다.

수학 형식

$$y(a) = A$$

$$y'(a) = A$$

$$y''(a) = A$$

MATLAB 형식

$$'y(a) = A'$$

$$'Dy(a) = A'$$

$$'D2y(a) = A'$$

- ▶ 인자 'var'은 옵션이며, 식에서 독립 변수를 정의하는데 사용된다. 입력이 없다면, 기본 설정값은 t이다.
- ▶ 초기 조건이 $y(0)=5$ 인 1차 ODE $\frac{dy}{dt} + 4y = 60$ 은 MATLAB에 의해 다음과 같이 풀 수 있다.

```
>> dsolve('Dy+4*y=60', 'y(0)=5')
```

```
ans =
```

```
15 - 10/exp(4*t)
```

답 $y = 15 - (10 / e^{4t})$ 이 출력

상미분방정식의 풀이

11.6 상미분방정식의 풀이

- ▶ 초기조건이 $y(0)=1$ 과 $\left.\frac{dy}{dt}\right|_{t=0} = 0$ 인 2차 ODE $\frac{d^2y}{dt^2} + 2\frac{dy}{dt} + 2y = 0$ 의 해를 MATLAB에 의해 구하면 다음과 같다.

```
>> dsolve('D2y-2*Dy+2*y=0', 'y(0)=1',  
          'Dy(0)=0')
```

```
ans =
```

```
exp(t)*cos(t)-exp(t)*sin(t)
```

```
>> factor(ans)
```

```
ans =
```

```
exp(t)*(cos(t)-sin(t))
```

답 $y = e^t \cos(t) - e^t \sin(t)$ 이 출력

답은 factor 명령어를 가지고 간단하게
정리

정리한 답 $y = e^t (\cos(t) - \sin(t))$ 이 출력

- ▶ 미분방정식의 풀이에 대한 부가적인 예는 예제 11-5에서 보여준다.
- ▶ MATLAB이 해를 찾지 못하면, 빈 기호개체와 메시지 'Warning: explicit solution could not be found'를 돌려준다.



11.7

기호식의 그래프 그리기

기호식의 그래프 그리기

11.7 기호식의 그래프 그리기

- ❖ 많은 경우 기호식을 그래프로 그릴 필요가 있다.
- ❖ 기호식의 그래프를 그리는 것은 `ezplot` 명령어로 쉽게 할 수 있다.
- ❖ 한 개의 변수 `var`이 포함된 기호식 `S`에 대해, MATLAB은 이 기호식을 함수 $S(\text{var})$ 로 고려하며, 이 명령어는 `var`에 대한 $S(\text{var})$ 의 그래프를 생성한다.
- ❖ 두 기호변수 `var1`과 `var2`를 포함한 기호식에 대해, MATLAB은 식을 $S(\text{var1}, \text{var2})=0$ 의 형태를 가진 함수로 고려하며, 이 명령어는 나머지 변수에 대한 한의 변수의 그래프를 생성한다.

기호식의 그래프 그리기

11.7 기호식의 그래프 그리기

❖ 한 개 또는 두 개의 변수가 포함된 기호식 S를 그리기 위해서, ezplot 명령어의 형식은 다음과 같다.

ezplot(S) 혹은 ezplot(S, [min, max]) 혹은
ezplot(S, [xmin, xmax, ymin, ymax])

독립변수의 정의역

종속변수의 정의역

- ▶ S는 그래프로 나타낼 기호식이다. 이전에 생성된 기호식의 이름이거나 S대신 입력될 수 있는 식이다.
- ▶ 그래프로 그릴 식에 포함된 변수들을 기호개체로 먼저 생성하지 않고 문자열로 식을 직접 입력하는 것도 가능하다.
- ▶ S가 한 개의 기호변수를 가지면, 가로좌표(수평축)에 var(독립변수)의 값과 세로 좌표(수직축)에 S(var)의 값을 기용하는 var에 대한 S(var)의 그래프가 생성된다.

기호식의 그래프 그리기

11.7 기호식의 그래프 그리기

- ▶ 기호식 S 가 두 개의 기호변수 $var1$, $var2$ 를 가지면, 식은 $S(var1, var2)=0$ 의 형태를 가진 함수로 가정한다.
- ▶ MATLAB은 나머지 변수에 대한 하나의 변수의 그래프를 생성한다.
- ▶ 알파벳 순서로 먼저인 변수가 독립변수로 취급된다.
- ▶ S 의 변수가 x 와 y 라면, x 는 독립변수이고 가로좌표에 그려지고, y 는 세로 좌표에 그려진 종속변수이다.
- ▶ S 의 변수가 u 와 v 라면, u 는 독립변수이고 v 는 종속변수이다.
- ▶ `ezplot(S)` 명령어에서 S 가 하나의 변수($S(var)$)를 가진다면, 그래프는 정의역 $-2\pi < var < 2\pi$ (기본 설정)에 대해 그려지며 범위는 MATLAB에 의해 정해진다.

기호식의 그래프 그리기

11.7 기호식의 그래프 그리기

- ▶ S 가 두 개의 변수($S(\text{var1}, \text{var2})$)를 가진다면, 그래프는 $-2\pi < \text{var1} < 2\pi$ 과 $-2\pi < \text{var2} < 2\pi$ 에 대해 그려진다.
- ▶ `ezplot(S, [min,max])` 명령어에서, 독립변수에 대한 정의역은 `min`과 `max`에 의해 $\text{min} < \text{var} < \text{max}$ 로 정의되며 범위는 MATLAB에 의해 정해진다.
- ▶ `ezplot(S, [xmin, xmax, ymin, ymax])` 명령어에서, 독립변수에 대한 정의역은 `xmin`과 `xmax`에 의해 정의되며 종속변수의 정의역은 `ymin`과 `ymax`에 의해 정의된다.
- ▶ `ezplot` 명령어는 매개변수 형태로 주어지는 함수의 그래프 출력에도 사용될 수 있다. 이 경우 두 기호식 $S1$ 과 $S2$ 가 명령어에 포함되며, 각 기호식은 같은 기호변수(독립 매개변수)에 의해 표현된다.

기호식의 그래프 그리기

11.7 기호식의 그래프 그리기

- ▶ $x=x(t)$ 와 $y=y(t)$ 가 주어질 때 x 에 대한 y 의 그래프를 그리는 경우, `ezplot` 명령어의 형식은 다음과 같다. 독립 매개변수의 정의역

`ezplot(S1, S2)` 혹은 `ezplot(S1, S2, [min, max])`

- ▶ $S1$ 과 $S2$ 는 독립 매개변수인 동일한 한 개의 기호변수를 포함한 기호식이다. $S1$ 과 $S2$ 는 이전에 생성된 기호식의 이름 또는 입력될 수 있는 식이다.
- ▶ 명령어는 $S1(var)$ 에 대한 $S2(var)$ 의 그래프를 생성한다. 명령어에 먼저 입력되는 기호식(위의 정의에서 $S1$)은 수평축에 사용되며, 두 번째로 입력되는 식(위의 정의에서 $S2$)은 수직축에 사용된다.

기호식의 그래프 그리기

11.7 기호식의 그래프 그리기

- ▶ `ezplot(S1, S2)` 명령어에서 독립변수에 대한 정의역은 $0 < var2 < 2\pi$ (기본 설정)이다.
- ▶ `ezplot(S1, S2, [min, max])` 명령어에서 독립변수에 대한 정의역은 min과 max에 의해 $min < var < max$ 로 정의된다.
- ▶ $x=x(t)$ 와 $y=y(t)$ 가 주어질 때 x 에 대한 y 의 그래프를 그리는 경우, `ezplot` 명령어의 형식은 다음과 같다.

❖ 추가 설명

- ▶ 그래프가 일단 생성되면, `plot`이나 `fplot` 형식으로 생성된 그래프와 같은 방식으로 그래프 형식을 지정할 수 있다. 이것은 두 가지 방법: 명령어를 이용하는 방법과 그래프 편집기(5.4절 참조)를 이용하는 방법이 있다. 그래프가 생성될 때, 그래프에 사용된 식은 그래프 상단에 자동으로 표시된다. MATLAB은 2차원 극좌표 그래프와 3차원 그래프를 위한 추가적인 그래프 함수들을 가지고 있다.

기호식의 그래프 그리기

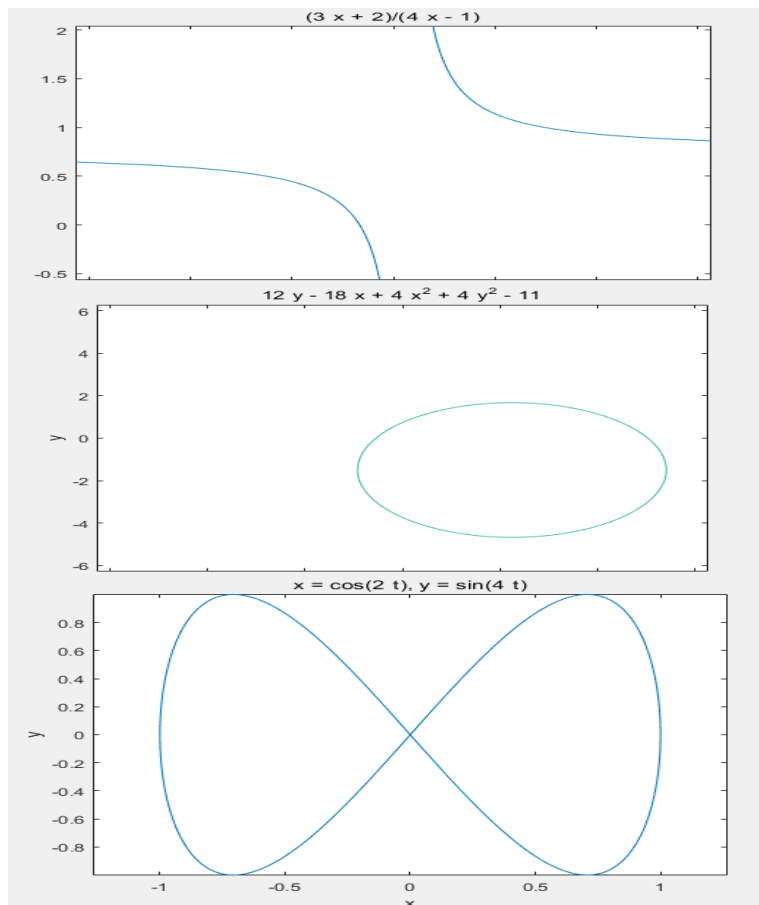
11.7 기호식의 그래프 그리기

▶ ezplot 명령어의 사용에 대한 몇 가지 예

```
>> syms x
>> S = (3*x+2)/(4*x-1)
S = (3*x+2)/(4*x-1)
>> ezplot(S)

>> syms x y
>> S = 4*x^2-18*x+4*y^2+12*y-11
S = 4*x^2-18*x+4*y^2+12*y-11
>> ezplot(S)

>> syms t
>> x = cos(2*t)
x = cos(2*t)
>> y = sin(4*t)
y = sin(4*t)
>> ezplot(x,y)
```





11.8

기호식에 의한 수치 계산

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

- ❖ 기호식이 사용자에게 의해 혹은 MATLAB의 기호연산 결과에 의해 생성 되면, 기호 변수에 대해 수를 대입할 필요와 식의 수치값을 계산할 필요가 있을 수 있다.
- ❖ 이것은 subs 명령어를 이용하여 이루어질 수 있다.
- ❖ sub 명령어는 여러 가지 형식을 가지고 있으며 여러 다른 방법에서 사용 될 수 있다.
- ❖ 다음은 사용하기 쉽고 대부분의 응용에 적당한 여러 형식들을 기술한다.

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

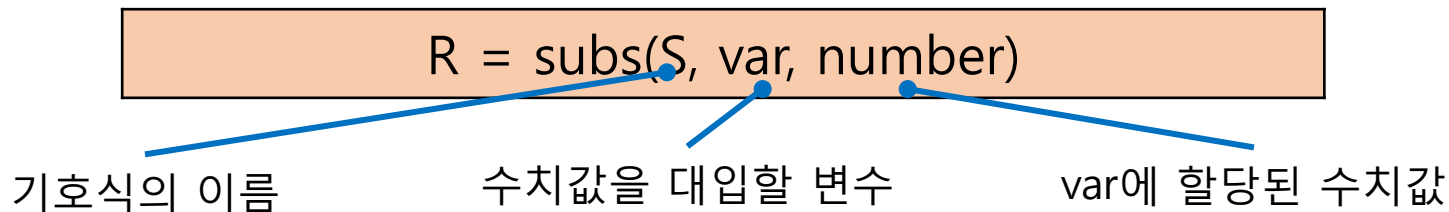
- ❖ 한 가지 형식에서는 수치값을 대입할 변수(또는 변수들)와 수치값 자체를 sub 명령어 안에 입력한다.
- ❖ 다른 형식에서 각 변수를 별도의 명령어에서 수치값으로 할당하고, 그 다음 해당 변수를 식에 대입한다.
- ❖ 변수와 변수의 값을 명령어 안에 입력하는 subs 명령어를 먼저 나타낸다.
- ❖ 두 가지 경우가 제시되는데, 첫 번째는 하나의 수치값(혹은 값들)을 한 개의 기호변수에 대입하고, 두 번째는 수치값들을 둘 이상의 기호변수에 대입한다.

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

❖ 단일 기호변수에 대한 단일 수치값의 대입

- ▶ 기호식이 한 개 이상의 기호변수를 가질 때, 수치값(또는 값들)을 하나의 기호변수에 대입할 수 있다.
- ▶ 이 경우 subs 명령어는 다음 형식을 갖는다



- ▶ number는 한 개의 수(스칼라)일 수도 있고 많은 원소들을 가진 배열(벡터 또는 행렬)일 수도 있다.
- ▶ S의 값이 number의 각 값에 대해 계산되며, 결과가 number(스칼라, 벡터 또는 행렬)와 같은 크기를 갖는 R에 할당된다.

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

- ▶ S가 하나의 변수를 가지면, 출력 R은 수치 값이다. S가 여러 개의 변수를 가지며 그 중 한 개의 변수에만 수치 값이 대입되면, 출력 R은 기호식이다.
- ▶ 하나의 기호변수가 포함된 식을 가진 예는 다음과 같다.

```
>> syms x
>> S = 0.8*x^3 + 4*exp(0.5*x)
S =
4*exp(x/2) + (4*x^3)/5
>> SD = diff(S)
SD =
2*exp(x/2) + (12*x^2)/5
>> subs(SD, x, 2)
ans =
15.0366
>> SDU = subs(SD, x, [2: 0.5: 4])
SDU =
15.0366 21.9807 30.5634
40.9092 53.1781
```

x를 기호변수로 정의
식 $0.8x^3 + 4e^{(0.5x)}$ 를 S에 할당

diff(S) 명령어를 이용하여 S를 미분
답 $2e^{x/2} + 12x^2/5$ 이 SD에 할당

subs 명령어를 이용하여 SD에 x=2를 대입
SD의 값이 출력
subs 명령어를 이용하여 SD에 x=[2, 2.5,
3, 3.5, 4] (벡터)를 대입
x의 각 값에 대한 SD의 값들(SDU에
할당)이 벡터로 출력

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

- ▶ 위의 예에서, 기호식의 수치값이 계산될 때, 답이 수치값(들여쓰기로 표시)임에 유의하라.
- ▶ 여러 개의 기호변수를 가지는 식에서 하나의 기호변수에 대해 수치값을 대입하는 예는 다음과 같다.

```
>> syms a g t v
>> Y = v^2*exp(a*t)/g
Y =
v^2*exp(a*t)/g

>> subs(Y, t, 2)
ans =
v^2*exp(2*a)/g
>> Yt = subs(Y, t, [2:4])
Yt =
[ v^2*exp(2*a)/g, v^2*exp(3*a)/g,
  v^2*exp(4*a)/g]
```

a, g, t, v를 기호변수로 정의
기호식 $v^2 e^{(at)}/g$ 를 생성하고 Y에 할당

subs 명령어를 이용하여 SD에 $t = 2$ 를
대입
답 $v^2 e^{(2a)}/g$ 이 출력

subs 명령어를 이용하여 Y에 $t=[2, 3, 4]$ (벡터)를 대입
답은 t의 각 값에 대한 기호식의 원소들을
가진 벡터

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

❖ 둘 이상의 변수에 대한 수치값의 대입

- ▶ 수치값(또는 값들)은 여러 개의 기호변수들이 기호식에 포함되어 있다면 둘 이상의 기호변수에 대입될 수 있다.
- ▶ 이 경우 subs 명령어는 다음과 같은 형식을 갖는다(두 변수에 대해 나타내지만, 더 많은 변수에 대해서도 같은 형식으로 사용할 수 있다).

```
R = subs(S, {var1, var2}, {number1, number2})
```

기호식의 이름

수치값을 대입할 변수

var1, var2에 할당된 수치값

- ▶ 변수 var1과 var2는 수치값이 대입될 식 S의 변수들이다. 변수들은 셀 배열(중괄호 { } 안)로 입력한다. 셀 배열은 각 셀이 수치 배열 혹은 텍스트의 배열이 될 수 있는 셀들의 배열이다.

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

- ▶ 변수들에 대입할 수인 `number1`, `number2`는 또한 셀 배열(중괄호 `{ }` 안)로 입력될 수 있다. 수는 스칼라, 벡터, 또는 행렬일 수 있다. 수치 셀 배열(`number1`)의 첫 번째 셀은 변수 셀 배열(`var1`)의 첫 번째 셀 변수에 대입되며, 나머지도 같은 식으로 대입된다.
- ▶ 변수에 대입되는 모든 수가 스칼라이면, 결과는 한 개의 수이거나 한 개의 식(일부 변수가 기호변수로 남아있는 경우)이 될 것이다.
- ▶ 적어도 한 개의 변수에 대해 대입할 수치가 배열이면, 수학 연산은 원소별로 수행되며 결과는 수치배열 또는 기호식들의 배열이 된다. 식 `S`가 원소별 표기법으로 입력되지 않을지라도 계산은 원소별로 수행됨이 강조되어야 한다. 이것은 또한 다른 변수들에 대입되는 배열들도 모두 같은 배열 크기를 가져야 함을 의미한다.

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

- ▶ 어떤 변수들에게는 같은 크기의 배열들을 대입하고, 다른 변수들에게는 스칼라를 대입하는 것이 가능하다. 이 경우에 원소별 연산을 수행하기 위해, MATLAB은 배열 결과로 출력하기 위해 스칼라로 확장(모든 원소가 1인 배열에 스칼라를 곱함)한다.
- ▶ 둘 이상의 변수에 수치값을 대입하는 것은 다음 예에서 설명한다.

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

```
>> syms a b c e x
>> S=a*x^e+b*x+c
S=
a*x^e+b*x+c
>> subs(S, {a,b,c,e,x}, {5,4,-20,2,3})
ans=
37
>> T=subs(S, {a,b,c}, {6,5,7})
T=
5*x+6*x^e+7
>> R=subs(S, {b,c,e}, {[2 4 6],9,[1 3 5]})
R=
[2*x+a*x+9, a*x^3+4*x+9,
a*x^5+6*x+9]
>> W=subs(S, {a,b,c,e,x}, {[4 2 0],[2 4 6],[2 2 2],[1 3 5],[3 2 1]})
W=
20 26 8
```

a, b, c, e, x를 기호변수로 정의
기호식 $ax^e + bx + c$ 를 생성하고 S에 할당
S의 모든 기호변수에 대해 스칼라를 대입
셀 배열, 셀 배열
S의 값이 출력

S의 기호변수 a, b, c에 대해 스칼라를
대입
결과는 변수 x와 e가 포함된 기호식
식 S에서 c에는 스칼라를, b와 e에는
벡터를 대입
결과는 기호식들의 벡터

S의 모든 변수들에 대해 벡터를 대입
결과는 수치값들의 벡터

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

- ▶ 기호식에서 수치값을 기호변수에 대입하기 위한 두 번째 방법은 먼저 수치값을 변수에 할당한 다음 subs 명령어를 이용하는 것이다.
- ▶ 이 방법에서 일단 기호식이 존재한다면, 변수들을 수치값으로 할당한다. 그 다음 subs 명령어는 다음과 같은 형식을 사용한다.

$$R = \text{subs}(S)$$

기호식의 이름

- ▶ 기호변수가 일단 수치변수로 재정의된다면 기호변수로 더 이상 사용될 수 없다.

기호식에 의한 수치 계산

11.8 기호식에 의한 수치 계산

▶ 이 방법은 다음 예에서 설명된다.

```
>> syms A c m x y
>> S=A*cos(m*x)+c*y
S=
c*y+A*cos(m*x)
>> A=10; m=0.5; c=3;
>> subs(S)
ans=
3*y+10*cos(x/2)

>> x=linspace(0, 2*pi, 4);
>> T=subs(S)
T=
[3*y+10, 3*y+5, 3*y-5, 3*y-10]
```

A, c, m, x, y를 기호변수로 정의
기호식 $A\cos(mx) + cy$ 를 생성하고 S에
할당

수치값들을 변수 A, m, c에 할당
식 S를 가지고 subs 명령어를 이용

변수 A, m, c의 수치값들이 식 S에 대입
수치 값들(벡터)을 변수 x에 할당
식 S를 가지고 subs 명령어를 이용
변수 A, m, c, x의 수치값들을 대입. 결과는
기호식들의 벡터



11.9

MATLAB 응용 예제

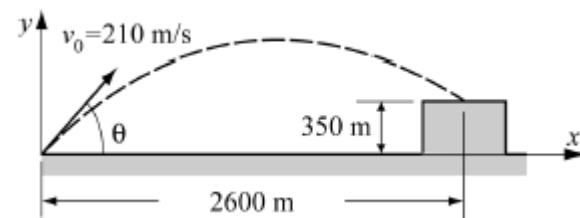
MATLAB 응용 예제

11.9 MATLAB 응용 예제

❖ 예제 11.2 발사체의 발사각

▶ 설명

- 발사체는 각 θ 에서 210 m/s의 속도로 발사된다.
- 발사체의 예정 목표물은 2600m 떨어진 거리에 있으며 발사지점보다 350m 높은 지점에 있다.



▶ 문제

- (a) 발사체가 목표물을 맞히게 될 각 θ 를 구하기 위해 풀어야 할 방정식을 유도하라.
- (b) MATLAB을 이용하여 (a)에서 유도된 방정식을 풀어라.
- (c) (b)에서 구한 각에 대해, ezplot 명령어를 이용하여 발사체의 궤적을 그래프로 그려라.

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- (a) 발사체의 운동은 수평 성분과 수직 성분을 고려하여 해석될 수 있다. 초기 속도 v_0 는 다음과 같이 수평 성분과 수직 성분으로 분해된다.

$$v_{0x} = v_0 \cos(\theta) \text{ 또는 } v_{0y} = v_0 \sin(\theta)$$

- 수평 방향에서, 속도는 일정하고 시간함수에 대한 발사체의 위치는 다음 식으로 주어진다.

$$x = v_{0x}t$$

- 발사체가 목표물까지 이동하는 수평거리에 $x = 2600\text{m}$ 를 대입하고 v_{0x} 에 $210\cos(\theta)$ 를 대입한 후, t 에 대해 풀면 다음과 같다.

$$t = \frac{2600}{210\cos(\theta)}$$

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 수직 방향에서, 발사체의 위치는 다음 식으로 주어진다.

$$y = v_{0y}t - \frac{1}{2}gt^2$$

- 목표물의 수직좌표에 $y = 360\text{m}$ 를 대입하고 v_{0x} , $g=9.81$ 과 위에서 구한 t 에 $210\cos(\theta)$ 를 대입하면, 다음 식을 얻을 수 있다.

$$350 = 210 \sin(\theta) \frac{2600}{210 \cos(\theta)} - \frac{1}{2} 9.81 \left(\frac{2600}{210 \cos(\theta)} \right)^2$$

또는

$$350 = \frac{2600 \sqrt{1 - \cos^2(\theta)}}{\cos(\theta)} - \frac{1}{2} 9.81 \left(\frac{2600}{210 \cos(\theta)} \right)^2$$

- 이 방정식의 해는 발사체가 발사되어야 할 각 θ 를 준다.

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- (b) solve명령어(명령어 창에서)를 이용하여 얻은 (a)에서 유도된 식의 해는 다음과 같다.

```
>> syms th
Angle = solve('2600*sqrt(1-cos(th)^2)/cos(th)-
0.5*9.81*(2600/(210*cos(th)))^2=350')
Angle=
1.245354497237416168313813580656
0.45925280703207121277786452037279
-0.45925280703207121277786452037279
-1.245354497237416168313580656
>> Angle1 = Angle(1)*180/pi

Angle1 =
224.16380950273491029648644451808/pi
>> Angle1=double(Angle1)
Angle1 = 71.3536
>> Angle2= Angle(2)*180/pi
Angle2 =
82.665505265772818300015613667102/pi
>> Angle2= double(Angle2)
Angle2 = 26.3132
```

MATLAB은 네 개의 해를 출력. 양의 두 해가 문제에
관련됨.
Angle의 첫 번째 원소의 해를 라디안에서 각도로
변환

MATLAB은 π 에 의한 기호개체로 답을 출력

double 명령어를 이용하여 Angle1의 수치값을 구함.

Angle의 두 번째 요소에 있는 해를 라디안에서 각도
로 전환

MATLAB은 π 에 의한 기호개체로 답을 출력
double명령어를 이용하여 Angle2의 수치값을 구함.

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- (c) (b)의 해는 두 개의 가능한 각이 있으며 따라서 두 궤적이 있음을 보여준다. 궤적의 그래프를 그리기 위해서, 발사체의 x, y 좌표는 다음과 같이 t (매개변수 형태)에 의해 주어진다.

$$x = v_0 \cos(\theta)t \text{ 또는 } y = v_0 \sin(\theta)t - \frac{1}{2}gt^2$$

- t 에 대한 정의역은 $t=0$ 에서 $t = \frac{2600}{210\cos(\theta)}$ 까지이다.
- 이 식들은 `ezplot` 명령어를 사용하여 스크립트 파일로 작성된 다음 프로그램에서 보여진 그래프를 그릴 수 있다.

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

```
xmax=2600; v0=210; g=9.81;  
theta1=1.24535; theta2=0.45925;  
t1= xmax/(v0*cos(theta1));  
t2= xmax/(v0*cos(theta2));  
syms t  
X1=v0*cos(theta1)*t;  
X2=v0*cos(theta2)*t;  
Y1=v0*sin(theta1)*t-0.5*g*t^2;  
Y2=v0*sin(theta2)*t-0.5*g*t^2;  
ezplot(X1,Y1,[0,t1])  
hold on  
ezplot(X2,Y2,[0,t2])  
hold off
```

(b)의 두 해를 θ_1 과 θ_2 에 할당

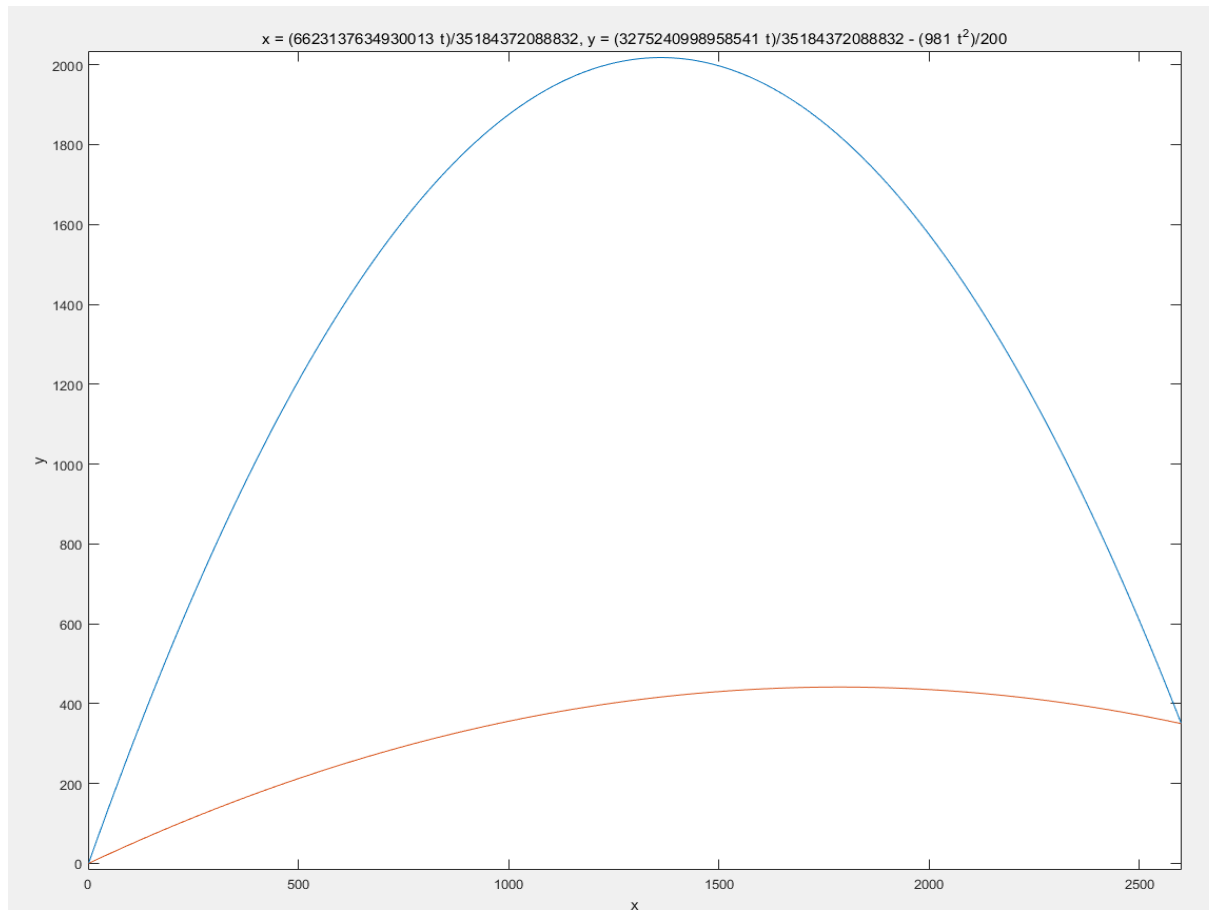
첫 번째 궤적의 그래프를 그림
두 번째 궤적의 그래프를 그림

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 이 프로그램이 실행되면, 다음 그래프가 그림 창에 생성된다.



MATLAB 응용 예제

11.9 MATLAB 응용 예제

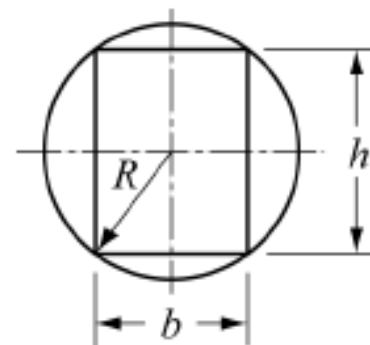
❖ 예제 11.3 빔의 굽힘 저항

▶ 설명

- 폭이 b 이고 높이가 h 인 직사각형 빔의 굽힘저항은 $I = \frac{1}{12}bh^3$ 으로 정의되는 빔의 관성모멘트 I 에 비례한다.
- 직사각형 빔을 반지름 R 의 원통형 통나무로부터 잘라내려고 한다.

▶ 문제

- 빔의 I 가 최대가 되는 R 에 대한 b 와 h 의 함수로 구하라.



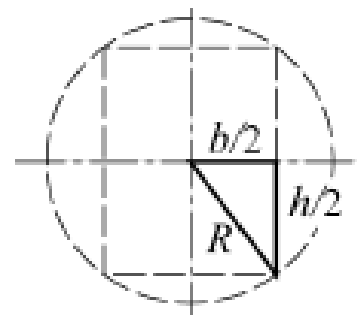
MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 다음 단계에 따라 문제를 푼다.

1. R, h, b 사이의 관계식을 작성하라.
2. h 에 대한 I 의 식을 유도하라.
3. h 에 대한 I 의 미분을 해라.
4. 미분을 0과 같다고 놓고 h 에 대해 풀어라
5. 해당하는 b 를 구하라.



- 첫 번째 단계는 그림의 삼각형으로부터 구할 수 있다. R, h, b 사이의 관계는 피타고라스 정리에 의해 $\left(\frac{b}{2}\right)^2 + \left(\frac{h}{2}\right)^2 = R^2$ 을 주어진 다. b 에 대해 이 식을 풀면 $b = \sqrt{4R^2 - h^2}$
- 나머지 단계들은 다음과 같이 MATLAB을 이용하여 이루어진다.

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

```
>> syms b h R
>> b=sqrt(4*R^2-h^2);
>> l=b*h^3/12
l =
(h^3*(4*R^2-h^2)^(1/2))/12
>> ID=diff(l,h)
ID =
(h^2*(4*R^2-h^2)^(1/2))/4-
h^4/(12*(4*R^2-h^2)^(1/2))
>> hs=solve(ID, h)
hs =
0
3^(1/2)*R
-3^(1/2)*R
>> bs=subs(b, hs(2))
bs =
(R^2)^(1/2)
```

b에 대한 기호식을 생성
단계 2: l에 대한 기호식을 생성
MATLAB이 l식에 b를 대입

단계 3: diff(R) 명령어를 이용하여 h에
대한 l를 미분

l의 미분이 출력
단계 4: solve 명령어를 이용하여 h에 대해
식 ID=0을 푼다. 답을 hs에 할당
MATLAB이 세 개의 해를 출력
0이 아닌 양의 해 $\sqrt{3}R$ 이 문제와 관련됨.
단계 5: subs명령어를 이용하여 b에 대한
식에서 h에 대한 해를 대입하여 b를 구함
b에 대한 답이 출력(답은 R이지만
MATLAB은 $(R^2)^{1/2}$ 로 표시)

MATLAB 응용 예제

11.9 MATLAB 응용 예제

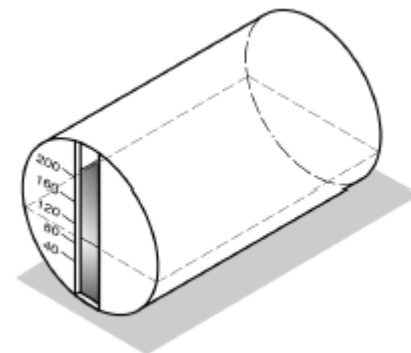
❖ 예제 11.4 탱크의 연료 레벨

▶ 설명

- 그림의 수평 원통탱크는 연료를 저장하도록 사용된다.
- 탱크는 지름 6m, 길이 8m를 가지고 있다.
- 탱크의 연료량은 탱크 앞면의 좁은 수직 유리창을 통해 본 연료레벨로 추정할 수 있다.
- 유리창 옆에 표시된 눈금은 1000 리터 단위로 40, 60, 80, 120, 160에 해당하는 연료의 레벨을 나타낸다.

▶ 문제

- 눈금 선들의 수직 위치(지면으로부터 측정된)를 구하라



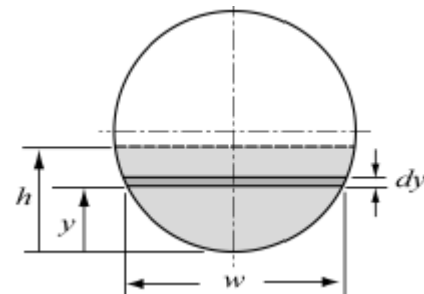
MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 연료의 레벨과 체적 사이의 관계는 정적분의 형태로 쓸 수 있다.
- 일단 적분을 수행하면, 식은 연료의 높이에 의한 체적에 대해 얻어진다.
- 특정 체적에 해당하는 높이는 높이에 대한 식을 풀면 구할 수 있다.
- 연료의 체적 V 는 연료의 횡단면(음영 부분)의 면적 A 와 탱크의 길이 L 을 곱하여 구할 수 있다. 단면적은 다음 적분으로 계산할 수 있다.

$$V = AL = L \int_0^h w dy$$



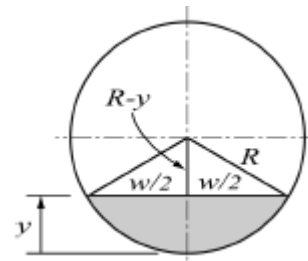
MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 연료 윗면의 폭 w 를 y 의 함수로 쓸 수 있다. 오른쪽 그림의 삼각형으로부터, 변수 y , w , R 의 관계식은 다음과 같다.

$$\left(\frac{w}{2}\right)^2 + (R - y)^2 = R^2$$



- W 에 대해 위 식을 풀면 다음 식으로 주어진다.

$$w = 2\sqrt{R^2 - (R - y)^2}$$

- 결과는 h 에 대한 체적 V 를 준다.
- 주어진 V 에 대한 h 의 값은 h 에 대한 식을 풀어서 구한다.
- 이 문제에서는 1000 리터 단위로 40, 60, 80, 120, 160의 체적에 대한 h 의 값을 구해야 한다.

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 해는 다음 MATLAB 프로그램(스크립트 파일)에 주어진다.

```
R3; L=8;  
syms w y h  
w=2*sqrt(R^2-(R-y)^2)  
S = L*w  
V = int(S, y, 0, h)
```

```
Vscale=[40: 40: 200]  
for i=1:5  
Veq=V-Vscale(i);  
h_ans(i)=solve(Veq);  
end  
  
h_scale=double(h_ans)
```

w에 대한 기호식을 생성
적분할 기호식을 생성
int 명령어를 이용하여 0에서 h까지 S를
적분. 결과는 h의 함수에 대한 V를 돌려줌
눈금의 V 값들을 벡터로 생성

루프를 반복할 때마다 V의 각 값에 대해
h를 푼다.
풀어야 할 h에 대한 식을 생성
solve 명령어를 이용하여 h에 대해 푼다.
h_ans는 벡터 Vscale에서 V의 값에
해당하는 h의 값들을 가진 벡터(수를 가진
기호)
double 명령어를 이용하여 벡터 h_ans의
원소들에 대한 수치값을 구함

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 스크립트 파일이 실행되면, 문장 끝에 세미콜론이 붙어있지 않은 명령어의 결과가 출력. 명령 창 화면은 다음과 같다.

```
>>w =  
2*(9-(y-3)^2)^(1/2)  
S =  
16*(9-(y-3)^2)^(1/2)  
V =  
36*pi+72*asin(h/3-1)+8*(9-(h-  
3)^2)^(1/2)*(h-3)  
  
Vscale =  
40 80 120 160 200  
h_scale =  
1.3972 2.3042 3.1439 3.9957  
4.9608
```

w에 대한 기호식이 출력
S는 적분될 식
적분 결과로 출력: h의 함수에 대한 V

눈금의 V 값들이 출력
눈금 선의 위치가 출력

- 단위: 위의 해에서 길이의 단위는 m이며, 이것은 체적 단위로 m^3 에 대응한다. ($1m^3=1,000L$).

MATLAB 응용 예제

11.9 MATLAB 응용 예제

❖ 예제 11.5 인체에서의 약물의 양

▶ 설명

- 인체에 존재하는 약물의 양 M 은 약물이 인체에 의해 소비되는 비율과 약물이 인체에 투입되는 비율에 의존하며, 여기서 약물이 소비되는 비율은 인체에 남아있는 양에 비례한다.
- M 에 대한 미분방정식은 다음과 같다.

$$\frac{dM}{dt} = -kM + p$$

- 여기서 k 는 비례상수이고, p 는 약물이 인체에 주입되는 비율이다.

▶ 문제

- (a) 눈금 선들의 수직 위치(지면으로부터 측정된)를 구하라
- (b) 환자가 병원에 입원하여 약물이 시간당 50mg의 비율로 투입된다(초기에 환자의 신체에는 약물이 없다). 시간의 함수에 대한 M 의 식을 유도하라.
- (c) 처음 24시간 동안에 대해 시간의 함수에 대한 M 을 그래프로 그려라.

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- (a) 비례상수는 새로운 약물의 투여 없이 인체에서 약물이 소비되는 경우를 고려하여 구할 수 있다. 이 경우 미분방정식은 다음과 같다.

$$\frac{dM}{dt} = -kM$$

- 위 식은 $t=0$ 에서 $M = M_0$ 인 초기조건으로 풀 수 있다.

```
>> syms M M0 k t  
>> Mt=dsolve('DM=-k*M', 'M(0)=M0')  
Mt =  
M0/exp(k*t)
```

dsolve 명령어를 이용하여 $\frac{dM}{dt} = -kM$ 을 푼다.

- 해는 다음과 같이 시간함수에 대한 M 을 준다.

$$M(t) = \frac{M_0}{e^{kt}}$$

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 3시간의 반감기는 $t=3$ 시간에서 $M(t) = \frac{1}{2}M_0$ 임을 의미한다. 이 정보를 위의 해에 대입하면 $0.5 = \frac{1}{e^{3k}}$ 이며, 상수 k 는 이 식을 풀어 구할 수 있다.

```
ks=solve('0.5=1/exp(k*3)')  
ks =  
.23104906018664843647241070715273
```

solve명령어를 이용하여 $0.5 = e^{-3k}$ 를 푼다

- 이 문제에 대한 M 의 미분방정식은 다음과 같다.

$$\frac{dM}{dt} = -kM + p$$

- 상수 k 는 (a)로부터 구해지고, $p = 50\text{mg/h}$ 는 주어진다.

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- 초기조건은 초기에 환자의 신체에 약물이 없다는 것이며, 혹은 $t = 0$ 에서 $M = 0$ 이라는 것이다.
- MATLAB에 의한 이 식의 해는 다음과 같다.

```
>> syms p  
>> Mtb=dsolve('DM=-k*M+p', 'M(0)=0')  
Mtb =  
(p-p/exp(k*t))/k)
```

dsolve 명령어를 이용하여 $\frac{dM}{dt} = -kM + p$ 를
푼다

- (c) $0 \leq t \leq 24$ 에 대해 시간의 함수에 대한 Mtb의 그래프는 ezplot 명령어를 이용하여 그려진다.

```
>>pgiven=50;  
>>Mtt=subs(Mtb, {p,k}, {pgiven, ks})  
Mtt =  
216.404-216.404/exp(0.231049*t)  
>>ezplot(Mtt,[0,24])
```

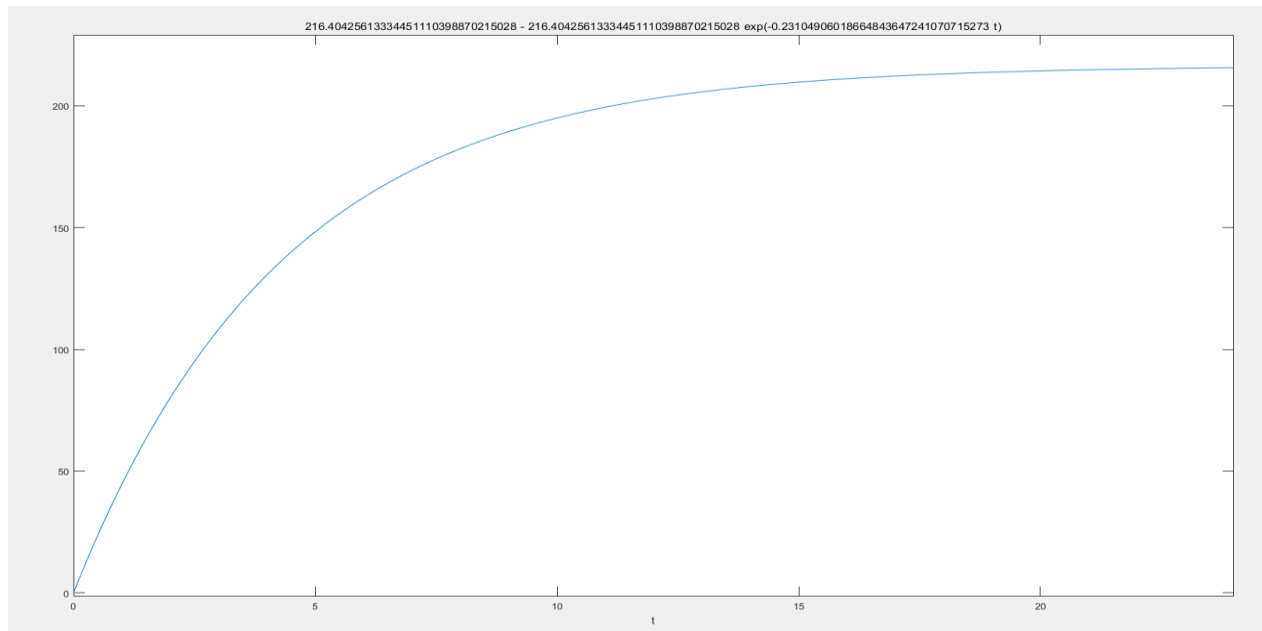
수치값을 p와 k에 대입

MATLAB 응용 예제

11.9 MATLAB 응용 예제

▶ 풀이

- MATLAB에 의해 생성된 이전 식의 실제 출력(Mtt= ...)은 위에 보여진 것보다 훨씬 많은 소수점 을 가지고 있다.
- 지면관계상 페이지에 맞도록 소수점 숫자들을 줄여서 표시하였다.
- 생성된 그래프는 다음과 같다.





Thank you!

11 장 끝