Table of Contents

```
clc, clear, close all
load 'constants.mat'
```
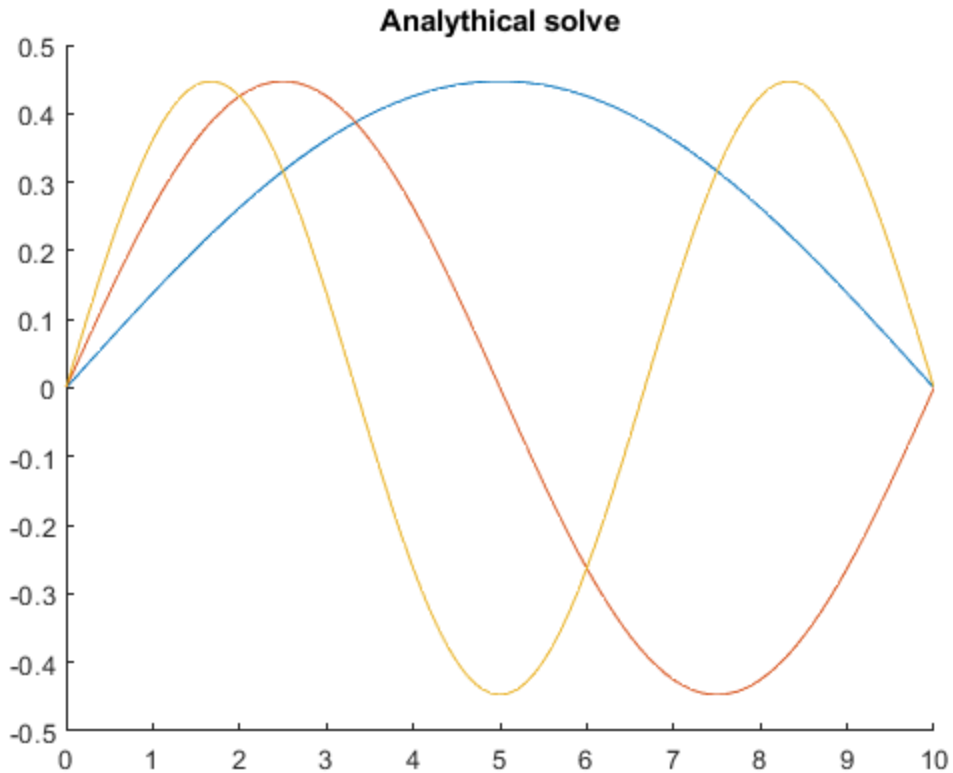
Handle-functions

```
Energy = @ (n, L, meff) (hbar * pi * n / (L * 1e-9))^2 / (2 * meff * m0) *
 J2eV * 1e3;
```

Ex.1 Analythical solve

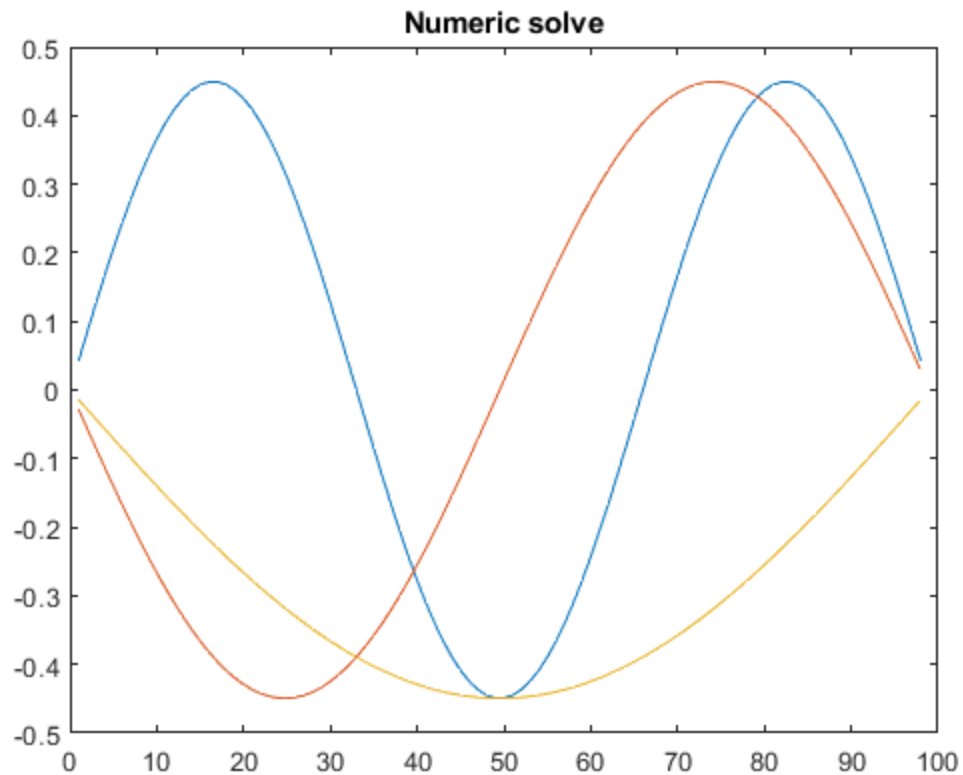In ex.1 we need to compare analythical and numeric solves for n = 1, 2, 3 by plotting.

```
figure('Name', 'Analythical solve')
hold on
L = 10;
x = 0:0.1:L;
h = 1/10;
for n = 1:3
    Psin = sqrt(2 / L) * (sin(pi * n * x / L));
    plot(x, Psin)
end
title('Analythical solve')
hold off
```

Analythical solve

Ex.1 Numeric solve

```matlab
figure('Name', 'Numeric solve')
N = 98;
I = eye(N, N);
T = I;
for i = 1:N
    for j = 1:N
        if i == j
            T(i, j) = -2;
            continue
        end
        if i == j - 1 || i == j + 1
            T(i, j) = 1;
        end
    end
end
[psi, D] = eig(T);
psi = psi ./ sqrt(h);
disp(sum(psi(:,N-2).^2)*h) % Make sure numbers are correct (need to = 1)
plot(1:N, psi(:,N-2), 1:N, psi(:,N-1), 1:N, psi(:,N))
title('Numeric solve')

    1.0000
```
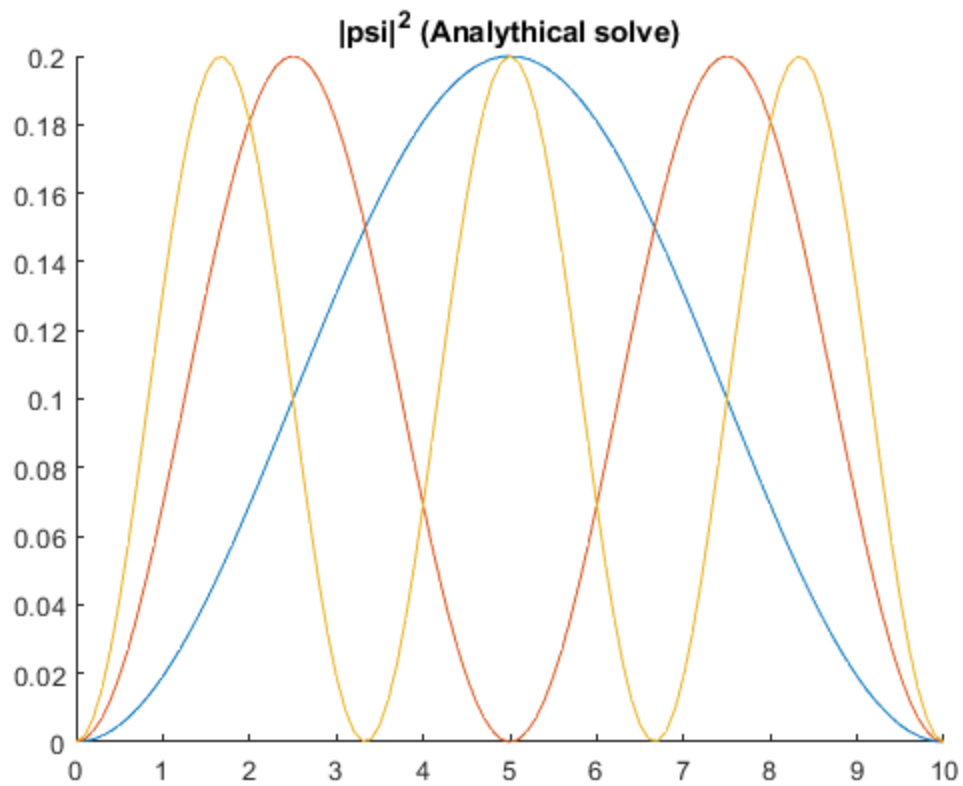
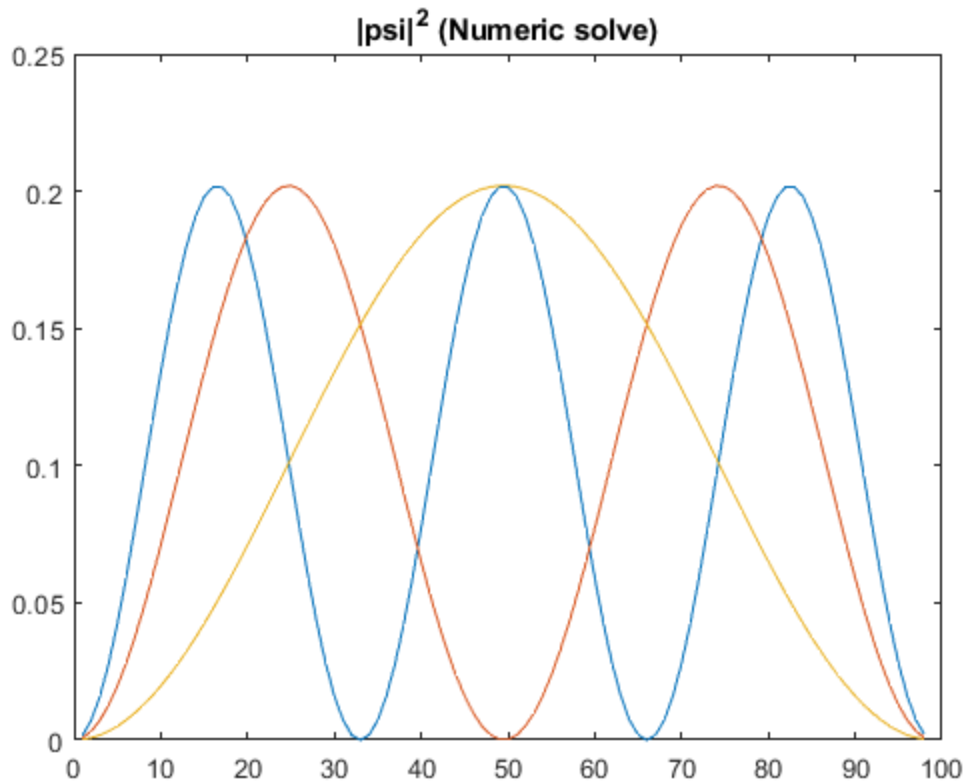**Numeric solve**

Ex.2 abs(psi)^2 (Analythical solve)

In ex.2 we need the same thing we had in ex.1 but for abs(psi)^2.

```matlab
figure('Name', 'abs(psi)^2 (Analythical solve)')
hold on
for n = 1:3
    Psin = sqrt(2 / L) * (sin(pi * n * x / L));
    plot(x, abs(Psin).^2)
end
title('|psi|^2 (Analythical solve)')
hold off
```

|psi|² (Analythical solve)

Ex.2 abs(psi)^2 (Numeric solve)

```matlab
figure('Name', 'abs(psi)^2 (Numeric solve)')
abPsi = abs(psi).^2;
plot(1:N, abPsi(:,N-2), 1:N, abPsi(:,N-1), 1:N, abPsi(:,N))
title('|psi|^2 (Numeric solve)')
```

$|psi|^2$ (Numeric solve)

Ex.3 E~n An. and Num. solves

In ex.3 we need to compare analythical and numeric connection between E and n by plotting. Also we need to try high n values (n = 10, 50, 90).

```
figure('Name', 'E~n Num. solves (n = 1...5)')
E = hbar^2 * diag(D) / (h^2 * 2 * m0);
E = wrev(E);
hold on
for n = 1:5
    x = [0 10];
    y = [E(n)/E(1); E(n)/E(1)];
    text(11, E(n)/E(1), strcat('n=', num2str(n)))
    plot(x, y)
    xlim([0 12]);
end
plot([10 10], get(gca, 'Ylim'), 'k');
title('E~n Num. solves (n = 1...5)')
hold off

figure('Name', 'E~n Num. solves (n = 10, 50, 90)')
hold on
for n = [10 50 90]
    x = [0 10];
    y = [E(n)/E(1); E(n)/E(1)];
    text(11, E(n)/E(1), strcat('n=', num2str(n)))
```

```matlab
    plot(x, y)
    xlim([0 12]);
end
plot([10 10], get(gca, 'Ylim'), 'k');
title('E~n Num. solves (n = 10, 50, 90)')
hold off

figure('Name', 'E~n An. solves (n = 1...5)')
hold on
for n = 1:5
    E = Energy(n, 10, 0.07);
    E = E/Energy(1, 10, 0.07);
    x = [0 10];
    y = [E E];
    text(11, E, strcat('n=', num2str(n)))
    plot(x, y)
    xlim([0 12]);
end
plot([10 10], get(gca, 'Ylim'), 'k');
title('E~n An. solves (n = 1...5)')
hold off

figure('Name', 'E~n An. solves (n = 10, 50, 90)')
hold on
for n = [10 50 90]
    E = Energy(n, 10, 0.07);
    E = E/Energy(1, 10, 0.07);
    x = [0 10];
    y = [E E];
    text(11, E, strcat('n=', num2str(n)))
    plot(x, y)
    xlim([0 12]);
end
plot([10 10], get(gca, 'Ylim'), 'k');
title('E~n An. solves (n = 10, 50, 90)')
hold off

% Another variant. All numeric and analythical energy levels.

figure('Name', 'E~n Num. and An. solves')
En = hbar^2 * diag(D) / (h^2 * 2 * m0);
for n = 1:98
    Ea(n) = Energy(n, 98, 0.07);
end
En = En./En(98);
En = wrev(En);
Ea = Ea./Ea(1);
hold on
plot(1:98, En, 1:98, Ea)
legend('Numeric', 'Analythical')
title('E~n Num. and An. solves')
hold off
```
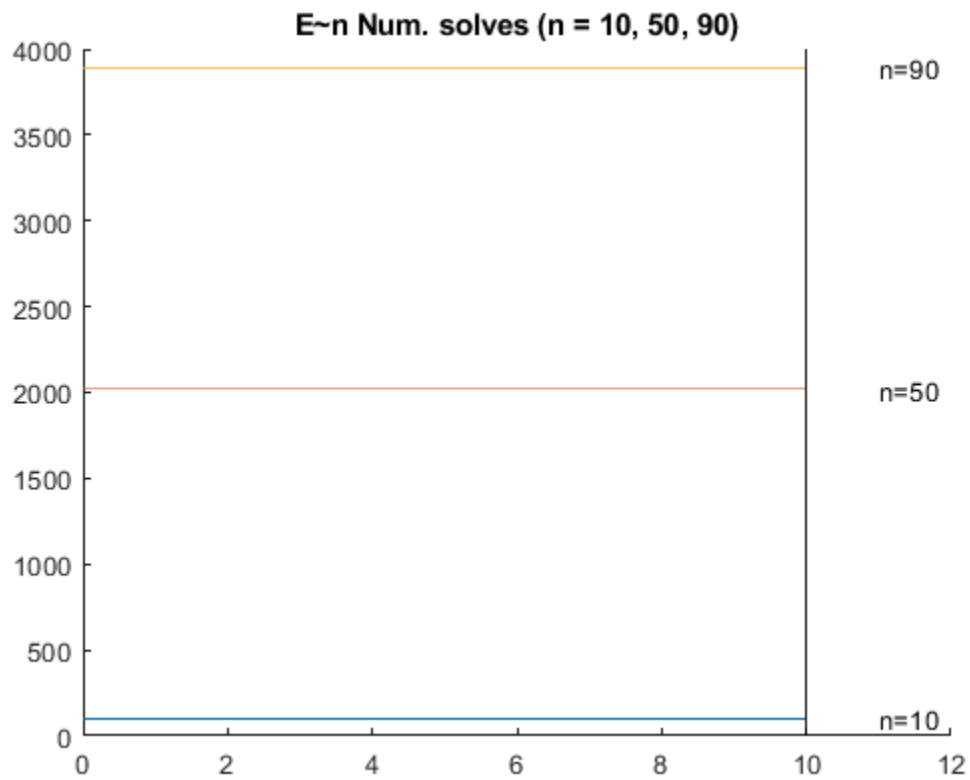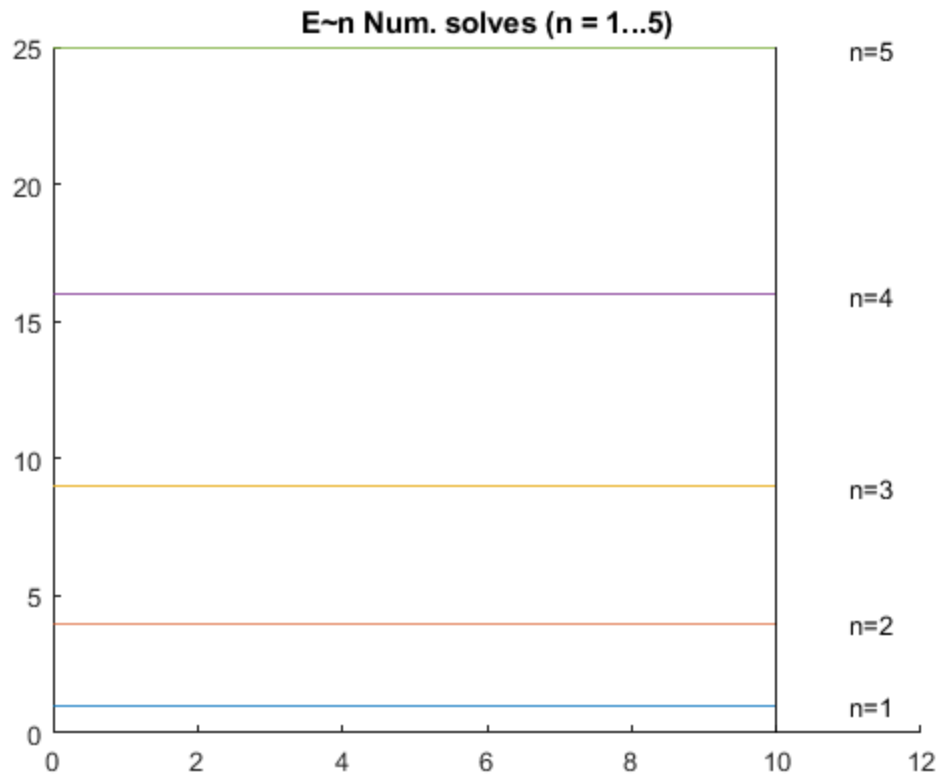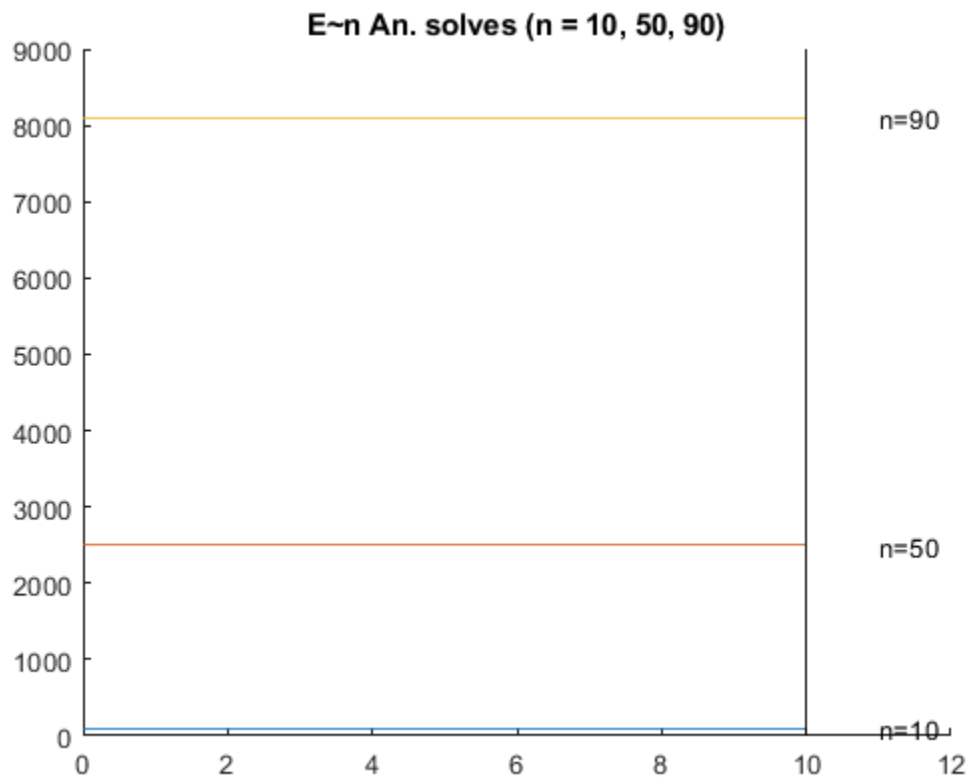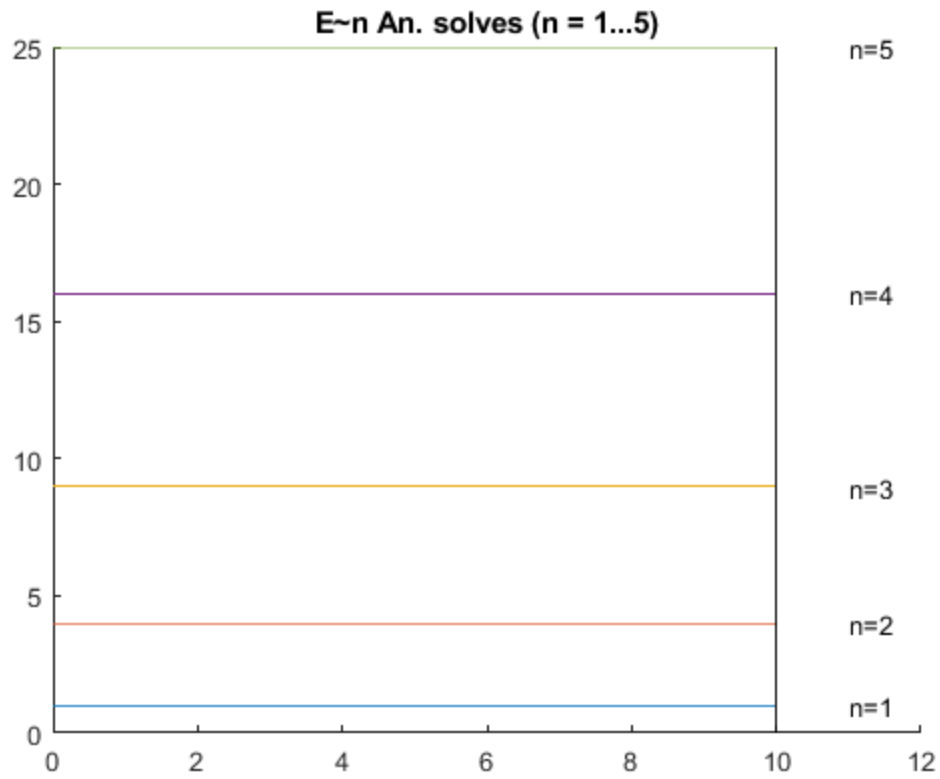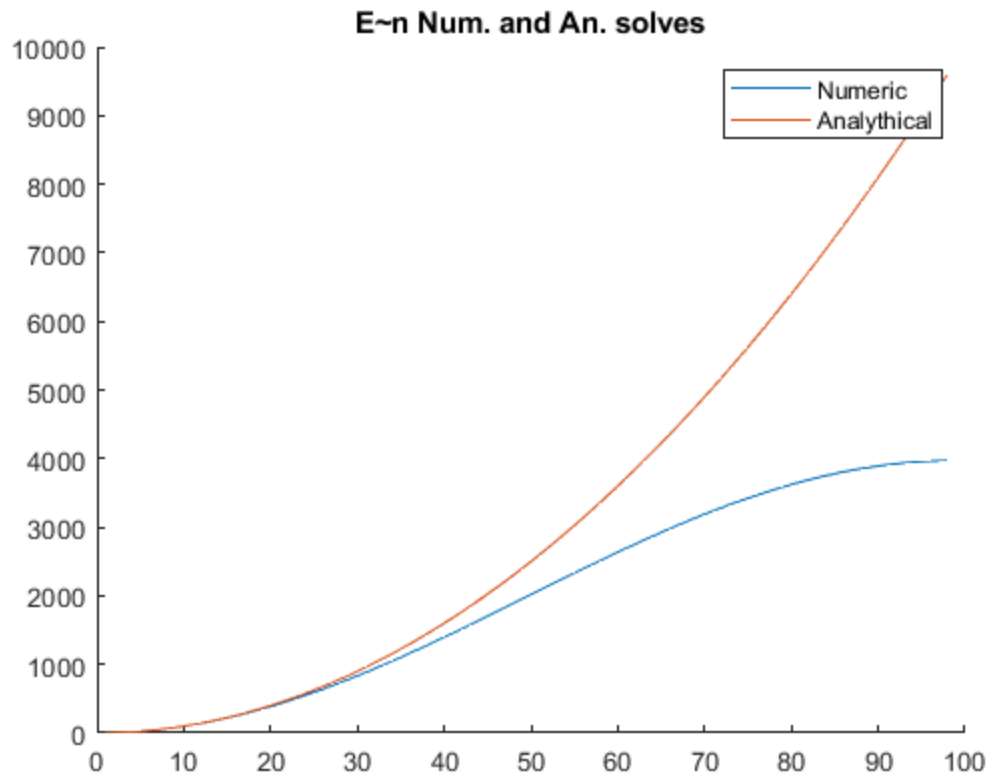
E~n Num. solves (n = 1...5)

n=5

n=4

n=3

n=2

n=1

E~n Num. solves (n = 10, 50, 90)

n=90

n=50

n=10

E~n An. solves (n = 1...5)

E~n An. solves (n = 10, 50, 90)

E~n Num. and An. solves

Ex. 4 An. and Num. solves for n >> 10

In ex.4 we need the same thing we had in ex.1 but for n >> 10 (for example, the highest n we had in task).

```
figure('Name', 'Analythical solve n >> 10')
L = 10;
x = 0:0.1:L;
h = 1/10;
n = 99;
Psin = sqrt(2 / L) * (sin(pi * n * x / L));
plot(x, Psin)
title('Analythical solve n >> 10')

figure('Name', 'Numeric solve n >> 10')
plot(1:N, psi(:,1))
title('Numeric solve n >> 10')
```

Analythical solve n >> 10
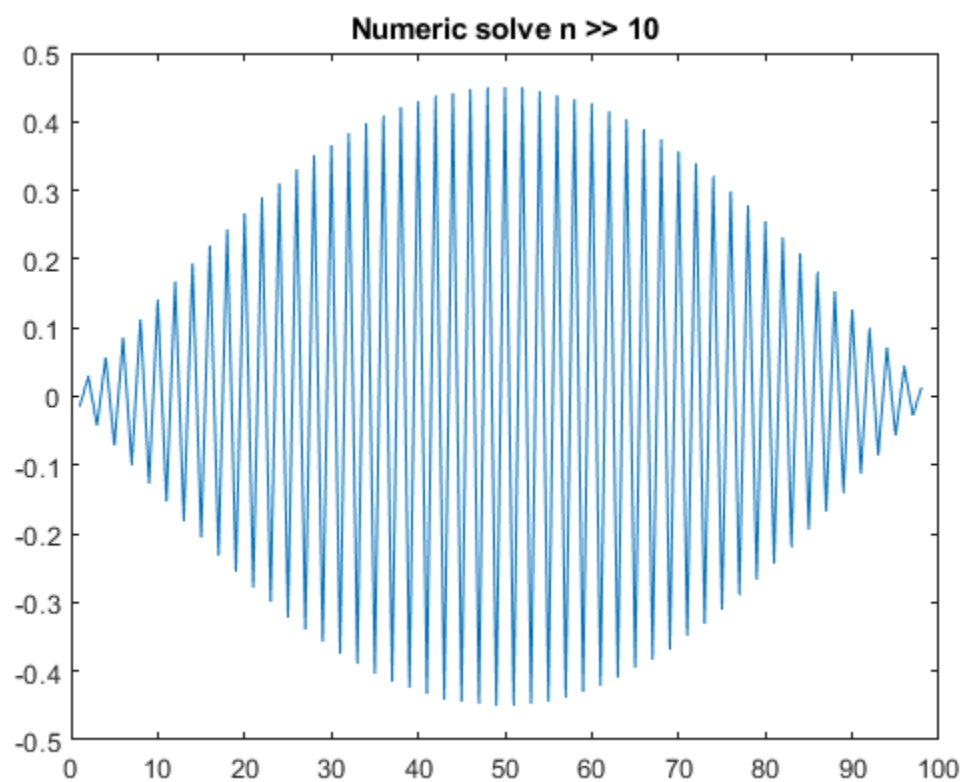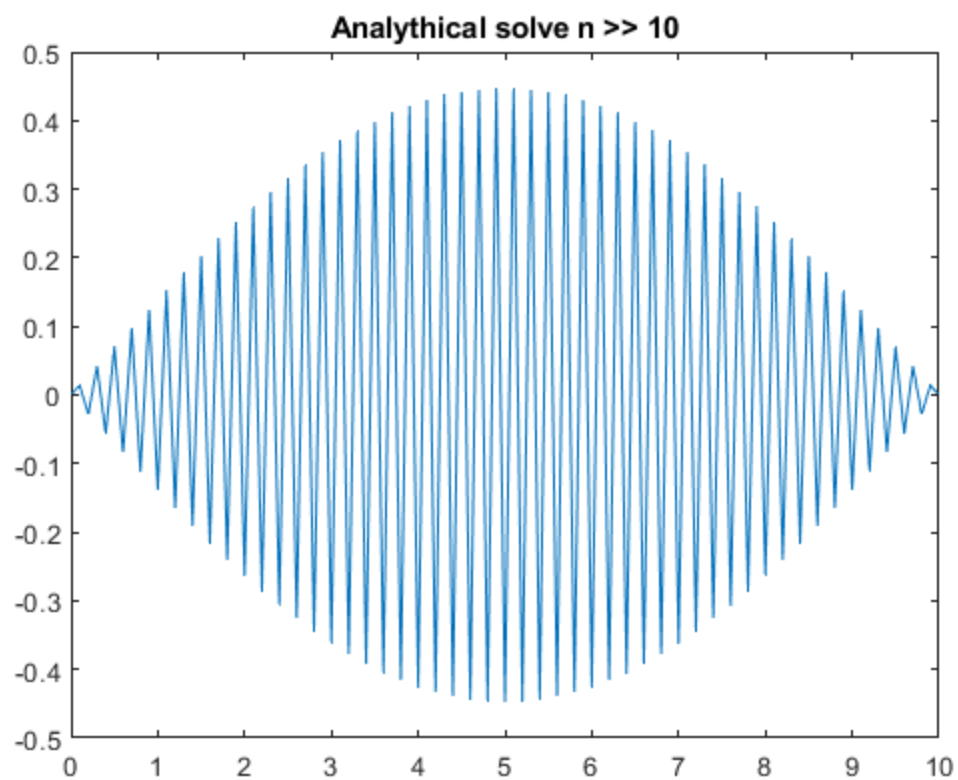
Numeric solve n >> 10

Ex. 5 (Advanced) En ~ n by numeric solve

In ex.5 we need to find out dependence of numeric En from n by plotting.
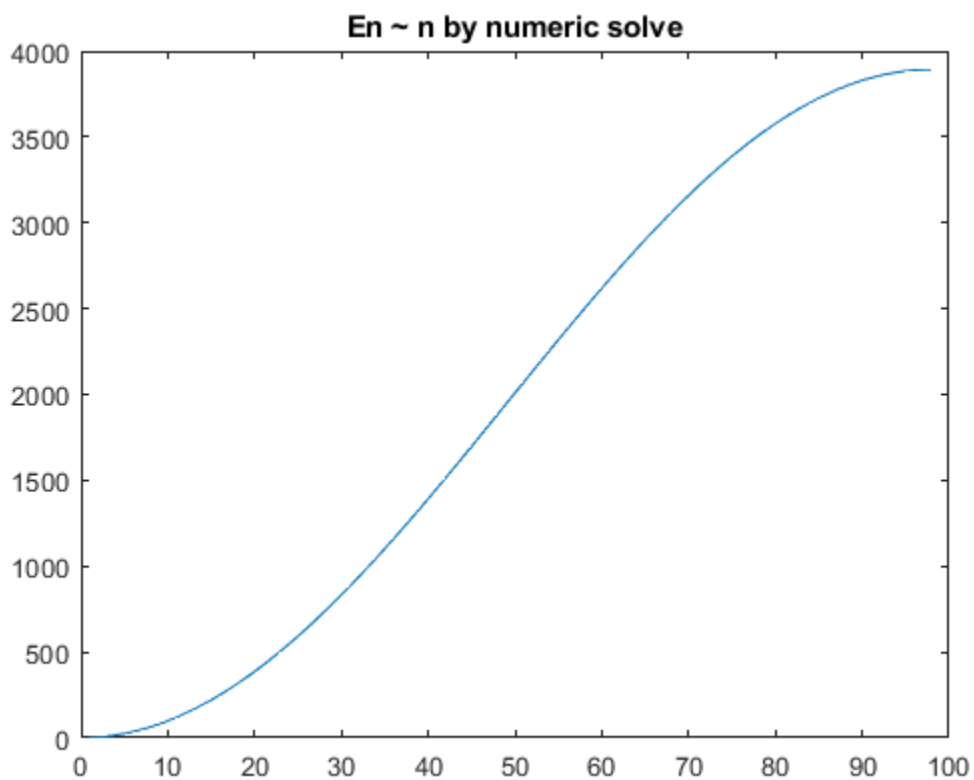
```
figure('Name', 'En ~ n by numeric solve')
n = 1:98;
h = 1;
t0 = hbar^2 / (2 * 0.07 * m0 * h);
En = 2 * t0 * (1 - cos(n * pi * h / 98));
plot(n, En./En(1))
title('En ~ n by numeric solve')

datetime(clock)


ans =

  datetime

   09-Nov-2023 22:30:55
```


Published with MATLAB® R2023a