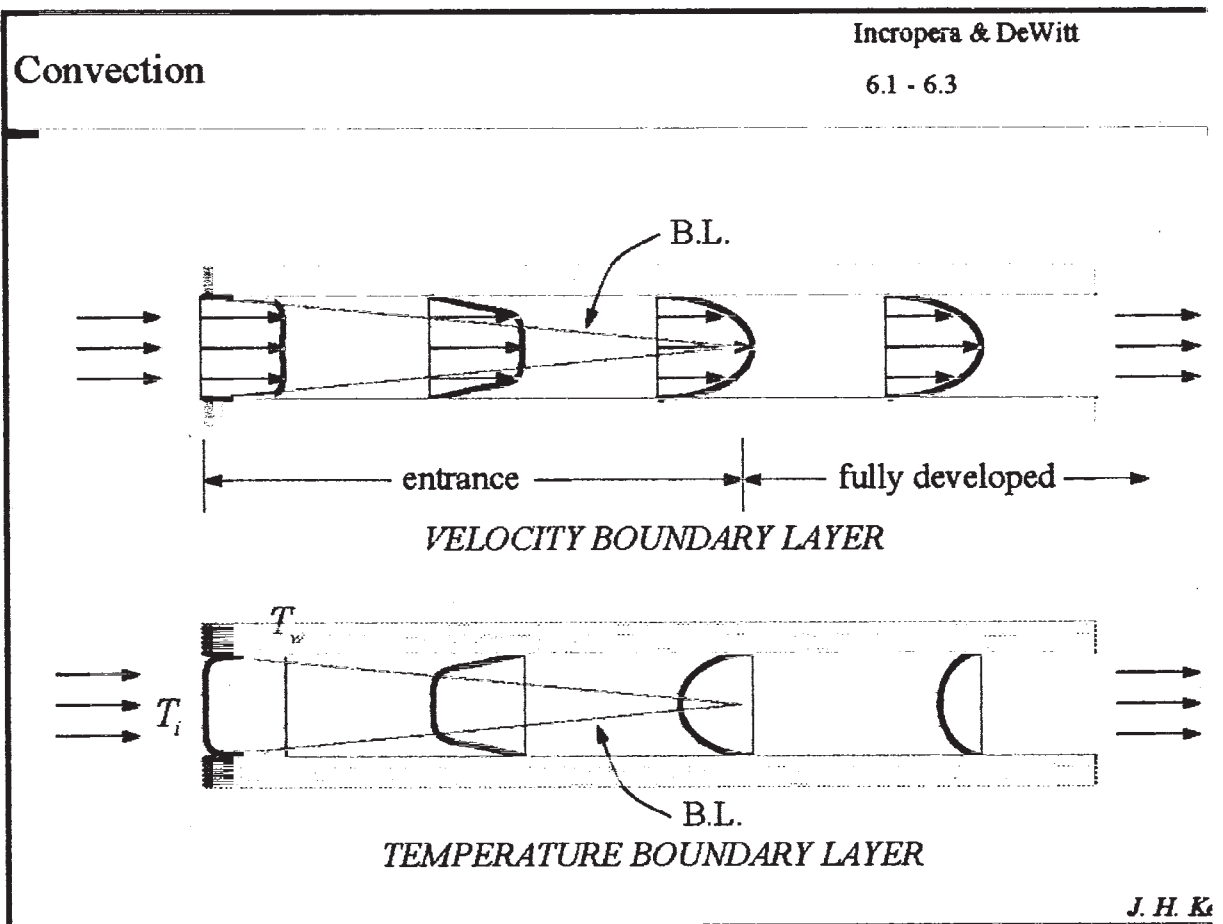


## Seconda Tesina

Si verifichi mediante integrazione numerica che  $Nu=3.66$  per la regione dinamica e termica completamente sviluppata (eqs. 7.140 e 7.141 del testo). Si adotti il metodo iterativo suggerito nella nota di pag. 278 partendo con  $\theta(r)=1$

- **Calcolo del Nusselt nella zona completamente sviluppata con il metodo delle approssimazioni successive**
- **Metodo alternativo per la determinazione del Nusselt nella zona completamente sviluppata: Soluzione di Bhatti**
- **Studio del condotto con temperatura alla parete imposta, nella zona completamente sviluppata dinamicamente ma non termicamente: Il Problema di Graetz**



L'equazione dell'energia, particolarizzata per questo esercizio e adimensionalizzata è:

i)  $r u \theta(r) = (r\theta')_r$  con l'aggiunta delle condizioni al contorno

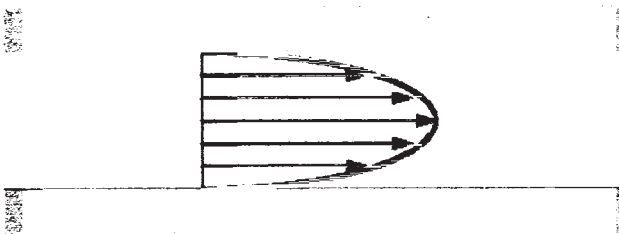
$\theta'(0)=0$  (condizione di simmetria)

$\theta(1)=0$  (temperatura costante alla parete)

con l'ipotesi di considerare un'opportuna adimensionalizzazione del campo di temperatura che renda possibile la dipendenza di  $\theta$  dalla sola coordinata  $r$

$\theta(r) = \frac{T - T_w}{T_i - T_w}$ . Facciamo l'ipotesi di trovarci nella zona completamente sviluppata sia dinamicamente che termicamente

$u = u(r)$  profilo parabolico di Poiseuille



a queste equazioni ne va aggiunta un'altra per la determinazione del Nu incognito:

ii)  $Nu = \frac{dq}{(t_w - t_b)} = -2\theta'(1)$

che risultata un'equazione nell'incognita Nu perchè  $\theta = \theta(r, Nu)$

L'equazione i) si presta ad essere integrata con il metodo delle "approssimazioni successive" che consiste nel fornire un valore di tentativo per la soluzione  $\theta(r)$  da sostituire al primo membro dell i) e poi nella successiva integrazione finchè il Nusselt ottenuto resta praticamente costante (dalla equ ii))

$$\text{ii) } Nu = \frac{q_w}{(t_w - t_b)} = -2\theta'(1)$$

che risultata un'equazione nell'incognita Nu perchè  $\theta = \theta(r, Nu)$

L'equazione i) si presta ad essere integrata con il metodo delle "approssimazioni successive" che consiste nel fornire un valore di tentativo per la soluzione  $\theta(r)$  da sostituire al primo membro dell i) e poi nella successiva integrazione finchè il Nusselt ottenuto resta praticamente costante (dalla equ ii)

$$\theta_0[r_] = 1;$$

$$u[r_] := (2 - 2 r^2)$$

$$V_1[r_] = \int_0^r r u[r] 1 Nu dr$$

Calcoliamo i primi integrali per notare la particolarità

$$Nu \left( r^2 - \frac{r^4}{2} \right)$$

$$\theta_1[r_] =$$

$$\text{Expand}[x[r] /. \text{Flatten}[\text{Simplify}[\text{DSolve}[\{x'[r] = V_1[r] \frac{1}{r}, x[1] = 0\}, x[r], r]]]]$$

$$-\frac{3 Nu}{8} + \frac{Nu r^2}{2} - \frac{Nu r^4}{8}$$

$$\text{Solve}[Nu == -2 \theta_1'[1], Nu]$$

$$V_2[r_] = \text{Expand}\left[\int_0^r r u[r] \theta_1[r] Nu dr\right]$$

$$-\frac{3}{8} Nu^2 r^2 + \frac{7 Nu^2 r^4}{16} - \frac{5 Nu^2 r^6}{24} + \frac{Nu^2 r^8}{32}$$

$$\theta_2[r_] =$$

$$\text{Expand}[x[r] /. \text{Flatten}[\text{Simplify}[\text{DSolve}[\{x'[r] = V_2[r] \frac{1}{r}, x[1] = 0\}, x[r], r]]]]$$

$$\frac{251 Nu^2}{2304} - \frac{3 Nu^2 r^2}{16} + \frac{7 Nu^2 r^4}{64} - \frac{5 Nu^2 r^6}{144} + \frac{Nu^2 r^8}{256}$$

si nota che la  $\theta$  ha una forma del tipo  $\theta[r] =$

$\sum_{n=0}^{\infty} C_{2n} r^{2n}$  dove si considerano solo i termini pari (2n) perchè le iterazioni che servono devono essere quelle con polinomi a esponente pari, che soddisfano il vincolo di simmetria a cui è sottoposto il profilo di temperatura

$$Nu[t_] = \text{Flatten}[N[\text{Solve}[Nu == -2 \theta_2'[1], Nu]]]$$

$$\{Nu \rightarrow 0., Nu \rightarrow 4.36364\}$$

Il ciclo per determinare i NUSSELT locali è quindi:

```

For[i = 1, i ≤ 80, If[(Mod[i, 2] != 0), Print[i, " ", Nu[i]]];
  Vi[x_] = Expand[ $\int_0^x r u[r] \theta_{i-1}[r] Nu dr$ ];  $\theta_i[x_] =$ 
  Expand[x[r] /. Flatten[Simplify[DSolve[{x'[r] = Vi[r]  $\frac{1}{r}$ , x[1] = 0}, x[r], r]]]];
  Nu[t_] = Flatten[FindRoot[Nu = -2  $\theta_i'$ [1], {Nu, 4.36}]]; i++]

1  Nu[1]
3  {Nu → 4.36364}
5  {Nu → 3.90768}
7  {Nu → 3.80572}
9  {Nu → 3.76257}
11 {Nu → 3.7388}
13 {Nu → 3.72376}
15 {Nu → 3.71338}
17 {Nu → 3.70578}
19 {Nu → 3.69998}
21 {Nu → 3.69541}
23 {Nu → 3.69172}
25 {Nu → 3.68867}
27 {Nu → 3.68611}
29 {Nu → 3.68393}
31 {Nu → 3.68205}
33 {Nu → 3.68042}
35 {Nu → 3.67898}
37 {Nu → 3.67771}
39 {Nu → 3.67657}
41 {Nu → 3.67556}
43 {Nu → 3.67464}
45 {Nu → 3.67381}
47 {Nu → 3.67305}
49 {Nu → 3.67236}
51 {Nu → 3.67172}
53 {Nu → 3.67113}
55 {Nu → 3.67059}
57 {Nu → 3.67009}
59 {Nu → 3.66962}

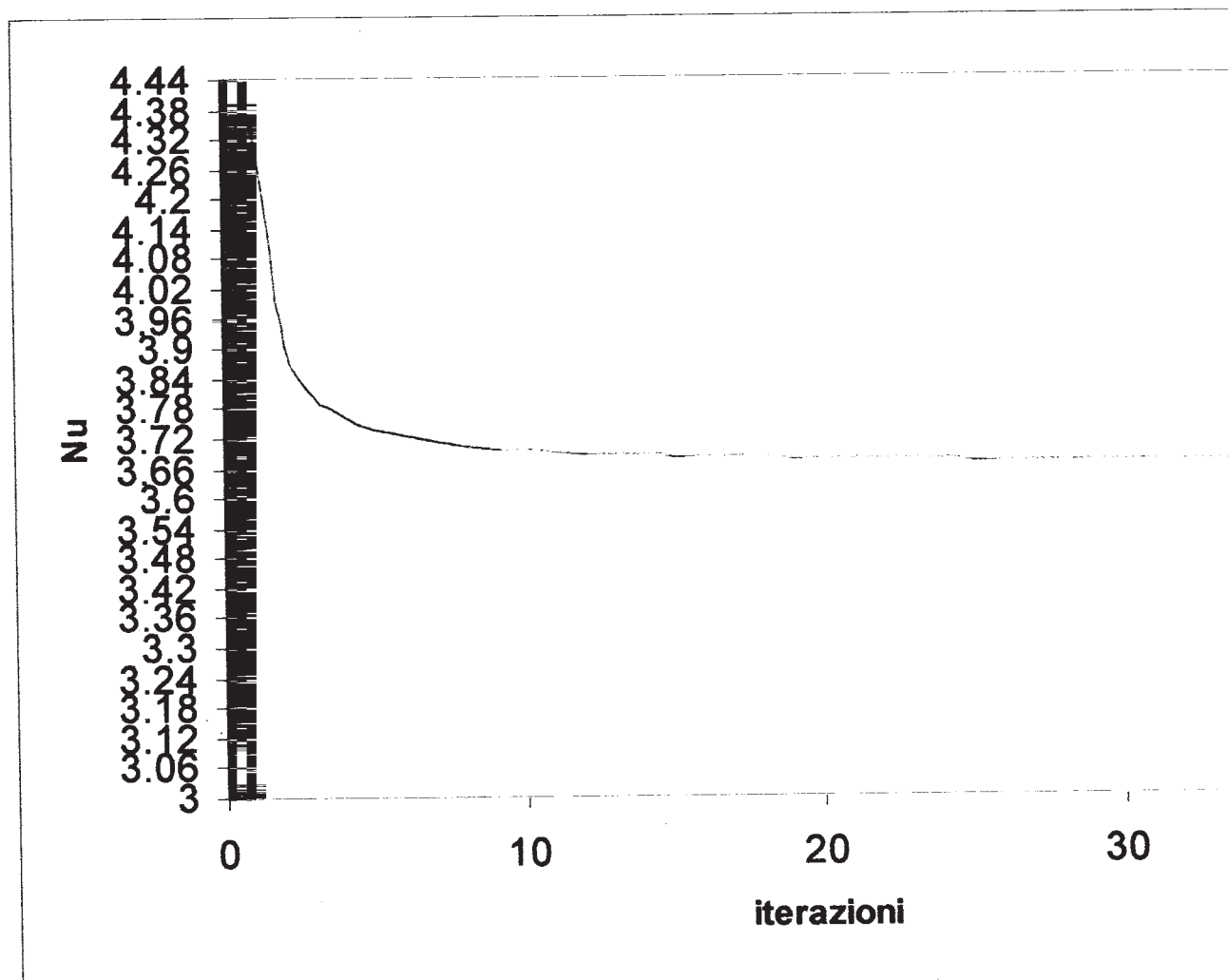
```

```

61 {Nu → 3.66919}
63 {Nu → 3.66878}
65 {Nu → 3.6684}
67 {Nu → 3.66804}
69 {Nu → 3.6677}
71 {Nu → 3.66739}
73 {Nu → 3.6671}
75 {Nu → 3.66689}
77 {Nu → 3.66692}

```

Che graficati, in funzione del numero di iterazioni



L'equazione dell'energia particolarizzata per il condotto è

$$\frac{1}{R} \frac{\partial \left( R \frac{\partial T}{\partial R} \right)}{\partial R} = \frac{U}{a} \frac{T_0 - T}{T_0 - T_m} \frac{dT_m}{dx}$$

dove  $T_0$  è la temperatura alla parete,  $T_m$  è la temperatura media ( $T_{\text{bulk}}$ ), alla cui equ. vanno aggiunte le condizioni al contorno:  $T = T_0$  ad  $R = R_0$

e  $\frac{\partial T}{\partial r} = 0$  ad  $R=0$

Il profilo parabolico di velocità dimensionale è  $U=2V(1-\frac{R^2}{R_0^2})$  con  $V$  velocità media nel condotto

La soluzione che prevede Bhatti è nella forma

$$\theta[r] = \frac{T_0 - T}{T_0 - T_m} = \sum_{n=0}^{\infty} C_{2n} \left( \frac{R}{R_0} \right)^{2n}$$

questa soluzione, nella forma è uguale a quella trovata col metodo delle approssimazioni successive

con  $C_0 = 1$ ,  $C_2 = -\frac{\lambda_0^2}{4} = -1.828397$

Ora procediamo al calcolo delle costanti tenendo conto che le prime 2,  $C_0$  e  $C_2$  ci sono date

$$C_0 = 1$$

$$1$$

$$\lambda_0 = 2.704364$$

$$2.70436$$

$$C_2 = -\frac{\lambda_0^2}{4}$$

$$-1.8284$$

La formula generale per le  $C_{2n}$  è

$$C_{2n} = \frac{Nu}{(\sqrt{2n})^2} (C_{2n-4} - C_{2n-2})$$

$$C_4 = \frac{Nu}{2(2)^2} (C_0 - C_2)$$

$$0.35355 Nu$$

$$C_6 = \text{Expand} \left[ \frac{Nu}{2(3)^2} (C_2 - C_4) \right]$$

$$-0.101578 Nu - 0.0196416 Nu^2$$

$$C_8 = \text{Expand}\left[\frac{\text{Nu}}{2 (4)^2} (C_4 - C_6)\right]$$

$$0.0142227 \text{Nu}^2 + 0.000613801 \text{Nu}^3$$

$$C_{10} = \text{Expand}\left[\frac{\text{Nu}}{2 (5)^2} (C_6 - C_8)\right]$$

$$-0.00203155 \text{Nu}^2 - 0.000677287 \text{Nu}^3 - 0.000012276 \text{Nu}^4$$

$$C_{12} = \text{Expand}\left[\frac{\text{Nu}}{2 (6)^2} (C_8 - C_{10})\right]$$

$$0.000225754 \text{Nu}^3 + 0.0000179318 \text{Nu}^4 + 1.705 \times 10^{-7} \text{Nu}^5$$

$$C_{14} = \text{Expand}\left[\frac{\text{Nu}}{2 (7)^2} (C_{10} - C_{12})\right]$$

$$-0.0000207301 \text{Nu}^3 - 9.2147 \times 10^{-6} \text{Nu}^4 - 3.08243 \times 10^{-7} \text{Nu}^5 - 1.7398 \times 10^{-9} \text{Nu}^6$$

Ora imponiamo l'equazione, che vale sempre per il caso di temperatura imposta alla parete,

$$\theta(1)=\theta(1,\text{Nu})==0$$

$$\text{FindRoot}[C_0 + C_2 + C_4 + C_6 + C_8 + C_{10} + C_{12} + C_{14} == 0, \{\text{Nu}, 3\}]$$

$$\{\text{Nu} \rightarrow 3.65889\}$$

abbiamo trovato un valore accettabile con sole 14 costanti

## Sturm-Liouville Boundary Value Problem

A differential equation defined on the interval  $a \leq x \leq b$  having the form of

$$\frac{d}{dx} \left[ p(x) \frac{dy}{dx} \right] + [q(x) + \lambda r(x)] y = 0$$

and the boundary conditions

$$\begin{cases} a_1 y(a) + a_2 y'(a) = 0 \\ b_1 y(b) + b_2 y'(b) = 0 \end{cases}$$

is called as **Sturm-Liouville boundary value problem** or **Sturm-Liouville system**, where  $p(x) > 0$ ,  $q(x)$ ,  $r(x) > 0$  are given functions;  $a_1$ ,  $a_2$ ,  $b_1$ ,  $b_2$  are given constants; and the eigenvalue  $\lambda$  is an unspecified parameter.

## Orthogonality and General Fourier Series

The non-trivial (non-zero) solutions  $\Phi_n(x)$ ,  $n = 1, 2, 3, \dots$ , of the Sturm-Liouville boundary value problem only exist at certain  $\lambda_n$ ,  $n = 1, 2, 3, \dots$ .  $\lambda_n$  is called *eigenvalue* and  $\Phi_n(x)$  is the *eigenfunction*.

The eigenvalues of a Sturm-Liouville boundary value problem are *non-negative real numbers*. In addition, the associated eigenfunctions  $\Phi_n(x)$  are *orthogonal* to each other with respect to the weighting function  $r(x)$ ,

$$\int_a^b r(x) \Phi_m(x) \Phi_n(x) dx = 0 \text{ if } m \neq n, m, n = 1, 2, 3, \dots$$

The complete set of the solutions  $\{\Phi_n(x) \mid a \leq x \leq b, n = 1, 2, 3, \dots\}$  forms a *complete orthogonal set* of functions defined on the interval  $a \leq x \leq b$ . Therefore, a piecewise continuous function  $f(x)$  can be expressed in terms of  $\Phi_n(x)$ ,  $n = 1, 2, 3, \dots$ , such that

$$\sum_{n=1}^{\infty} c_n \Phi_n(x) = \begin{cases} f(x) & \text{where } f(x) \text{ is continuous} \\ \frac{f(x^-) + f(x^+)}{2} & \text{at discontinuous points} \end{cases}$$

where

$$c_n = \frac{\int_a^b r(x) f(x) \Phi_n(x) dx}{\int_a^b r(x) \Phi_n(x) \Phi_n(x) dx}$$

The **completeness** allows us to *express any piecewise continuous function* in terms of these eigenfunctions while the **orthogonality** makes the expression *unique* and *compact* (no redundant terms). In addition, it can be shown that the orthogonal series is the *best* series available, i.e., each additional term fine tunes but not



overhauls the sum of the existing terms. These properties generalize the conventional [Fourier series](#)  $\sin \lambda_n x$  and  $\cos \lambda_n x$  to any complete orthogonal series  $\Phi_n(x)$ . The series is hence called the **generalized Fourier series**. The method of forming solutions by the general Fourier series is called the method of **eigenfunction expansion** which is an important technique in solving partial differential equations.

Examples of generalized Fourier series can be found in [Bessel functions](#), [Legendre polynomials](#), and other orthogonal polynomials such as [Laguerre polynomials](#), [Hermite polynomials](#), and [Chebyshev polynomials](#). Each of these polynomials represents a complete orthogonal set in different coordinates or circumstances and can be considered as a [special case of the Sturm-Liouville boundary value problem](#).

# Problema di Graetz

Il problema che ci apprestiamo a risolvere riguarda un condotto soggetto a temperatura imposta alla parete, del quale analizziamo la zona dinamicamente sviluppata in cui abbiamo profilo parabolico di velocità secondo la legge di Poiseuille, ma non termicamente sviluppato. In questo caso il campo di temperatura è funzione delle due coordinate  $x$  ed  $r$  nel condotto.

L'equazione dell'energia nella forma dimensionale è:

$$i) \frac{\partial^2 T}{\partial R^2} + \frac{1}{R} \frac{\partial T}{\partial R} = \frac{U}{a} \frac{\partial T}{\partial X} - \frac{\partial^2 T}{\partial X^2}$$

Questa equazione con le opportune adimensionalizzazioni

$$x = \frac{X}{R_o \text{RePr}}; r = \frac{R}{R_o}; u = \frac{U}{U_m}; \theta = \theta(x,r) = \frac{T_o - T}{T_o - T_e}$$

con  $R_o$  raggio del condotto,  $U_m$  velocità media,  $T_o$  temperatura alla parete,  $T_e$  temperatura all'ingresso,

$$U = 2 U_m \left(1 - \frac{R^2}{R_o^2}\right); u = u(r) = \frac{U}{U_m}$$

L'equazione i) adimensionalizzata è:

$$ii) \frac{\partial^2 \theta}{\partial r^2} + \frac{1}{r} \frac{\partial \theta}{\partial r} = \frac{u}{2} \frac{\partial \theta}{\partial x} - \frac{1}{(\text{RePr})^2} \frac{\partial^2 \theta}{\partial x^2}$$

Visto che il RePr assume valori molto elevati nel caso di fluidi come acqua e aria, si può trascurare l'ultimo termine dell'equ. ii) e riscriverla nella seguente forma:

$$iii) \frac{\partial^2 \theta}{\partial r^2} + \frac{1}{r} \frac{\partial \theta}{\partial r} = (1 - r^2) \frac{\partial \theta}{\partial x}$$

$$\frac{(r\theta_r)_r}{r} = (1 - r^2) \theta_x$$

Quest'ultima equazione è corredata dalle seguenti condizioni al contorno:

$$4i) \theta(0,r)=1 \quad \text{la temperatura all'ingresso (x=0) è } T = T_e$$

$$5i) \theta(x,1)=0 \quad T = T_o$$

$$6i) \theta_r(x,0) = 0 \quad \text{condizione di simmetria del profilo di Temperatura}$$

L'equazione ii) è lineare e omogenea e può essere risolta col metodo di separazione delle variabili:  $\theta = R(r) X(x)$

Sostituendo questa equazione nella ii) e "separando" le variabili  $R$  e  $X$  otteniamo:

$$R' X + r X R'' = r(1 - r^2) X' R$$

dividendo per  $RX$

$$\frac{R' + r R''}{r(1 - r^2) R} = \frac{X'}{X} = -\lambda^2$$

da questa ultima imposizione ricavo le due equazioni

$$7i) X' + \lambda^2 X = 0$$

$$8i) R''(r) + \frac{1}{r} R'(r) + \lambda^2 R(r) (1 - r^2) = 0$$

Le 4,5,6i) modificate, escludendo le soluzioni banali  $X(x)=0$  e  $R(r)=0$  diventano:

$$9i) X(0)R(r)=1 \quad \text{condizione iniziale}$$

10i)  $R(1)=0$

11i)  $R'(0)=0$

La soluzione della 7i) considerando la c.c. accoppiata 9i) è:  $X(x) = B \text{EXP}(-\lambda^2 x)$  dove B costante indeterminata causa 9i)

Ora procediamo al calcolo dei  $\lambda$

**Quit**

$\lambda = \text{Table}[i, \{i, 1, 12\}]$

$k = \text{Table}[i, \{i, 1, 12\}]$

{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}

{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}

Utilizziamo l'algoritmo di bisezione perchè ricerchiamo gli zeri di una funzione, cioè dobbiamo soddisfare la c.c.  $R(1)=0$ .

Nel ciclo ci calcoliamo la  $R(r)$  imponendo gli autovalori che mi permettano di gestire un intervallo attorno alla  $R(r)$ ,  $\{a,b\}$  e di valore medio

$c = \frac{a+b}{2}$ . Per esempio, per  $\lambda = \lambda_1$  ho  $a=R(1)$ , per  $\lambda = \lambda_2$  ho  $b=R(1)$ , poi per  $\lambda = \lambda_m = \frac{\lambda_1 + \lambda_2}{2}$  ho  $c=R(1)$ , per trovare il primo autovalore faccio restringere sempre di più questo intervallo  $a,b$  controllando il segno ( $a*c$ ) fino a che non arrivo ad avere  $\lambda_1 = \lambda_2$  il che significa che  $\lambda_m = 0$ , ho trovato la radice (autovalore  $\lambda$ ) che cercavo.

Questo ragionamento è giusto perchè la  $R(1)$  è una funzione che varia con i  $\lambda$  che io inserisco nell'istruzione  $\text{NDSolve}[\{R''[r] + R'[r]/r + (\lambda)^2 R[r] (1 - r^2) == 0, R[0.0001] == 1, R'[0.0001] == 0\}, R, \{r, 0.0001, 1.1\}]$

Al variare di  $\lambda$  inserito in questa istruzione ho valori di  $R(1)$  diversi quindi posso immaginare che  $R(1)$  sia una funzione di  $\lambda$  come in figura:

```

j = 1; L1 = -4; L2 = 0;
While[j ≤ 12,
  λ1 = L1 + 4; λ2 = L2 + 4; err = 1; i = 0;
  L1 = λ1; L2 = λ2;
  (*ogni volta i λ1 e λ2 iniziali sono quelli dell' iterazione precedente + 4*)
  While[err > 0.0001 && i < 100,
    i = i + 1; R = .;
    sol = NDSolve[{ R''[r] + R'[r] / r + (λ1)^2 R[r] (1 - r^2) == 0,
  R[0.0001] == 1, R'[0.0001] == 0}, R, {r, 0.0001, 1.1}]; R = R /. sol[[1]];
    a = R[1]; (*il valore della funzione per λ=λ1*)
    R = .;
    sol = NDSolve[{ R''[r] + R'[r] / r + (λ2)^2 R[r] (1 - r^2) == 0,
  R[0.0001] == 1, R'[0.0001] == 0}, R, {r, 0.0001, 1.1}]; R = R /. sol[[1]];
    b = R[1]; (*il valore della funzione per λ=λ2*)
    λm = (λ1 + λ2) / 2;
    R = .;
    sol = NDSolve[{ R''[r] + R'[r] / r + (λm)^2 R[r] (1 - r^2) == 0,
  R[0.0001] == 1, R'[0.0001] == 0}, R, {r, 0.0001, 1.1}]; R = R /. sol[[1]];
    c = R[1]; (*il valore della funzione per λ=λm*)
    err = Abs[c]; (*calcolo dell' errore, che va a zero*)
    If[a * c > 0, λ1 = λm, λ2 = λm];];
  (*se il segno della funzione in λ1
  è lo stesso che in λm allora il nuovo λ1=
  λm oppure il nuovo sarà λ2=λm*)
  λ[[j]] = λm; λ[[j]] = N[λ[[j]]]; j = j + 1

```

General::spell1 : Possible spelling error: new symbol name "λm" is similar to existing symbol "λ".

err

0.0000555384

λ

{2.70435, 6.6792, 10.6733, 14.6709, 18.6699,  
22.6689, 26.6689, 30.668, 34.668, 38.668, 42.668, 46.668}

Per controllare la esattezza di questi autovalori li confrontiamo con quelli che ci suggerisce il libro **Convective Heat and Mass, di Kays-Crawford**

che ci assicura una formula per i  $\lambda_n = 4n + \frac{8}{3}$

```

λ = Table[λn = 4 n + 2.705, {n, 0, 12}]
(*questi sono i λ secondo una formula sul libro*)
{2.705, 6.705, 10.705, 14.705, 18.705, 22.705,
26.705, 30.705, 34.705, 38.705, 42.705, 46.705, 50.705}

```

Il Campo di temperatura adimensionale è:

0j)  $\theta(x,r) = \sum_{n=0}^{\infty} C_n R_n(r)$   $X(x) = \sum_{n=0}^{\infty} C_n R_n(r) E^{-\lambda_n^2 x}$  considerando incluse nelle  $C_n$  le costanti non trovate della  $X(x)$

Per la ricerca dei  $\lambda$  abbiamo usato il metodo di bisezione sfruttando l'istruzione del *Mathematica* : NDSolve  
L'equazione agli Autovalori e alle Autofunzioni del problema studiato con la Separazione delle variabili è:

$$j) \quad R''(r) + \frac{1}{r} R'(r) + \lambda^2 R(r) (1 - r^2) = 0$$

Alla quale vanno aggiunte le condizioni al contorno

jj)  $R'(0)=0$  condizione di simmetria

jjj)  $R(1)=0$  la temperatura dimensionale deve essere costante alla parete

NDSolve trova le  $R_n$  autofunzioni con i vincoli imposti dalla jj) e da una nuova condizione:

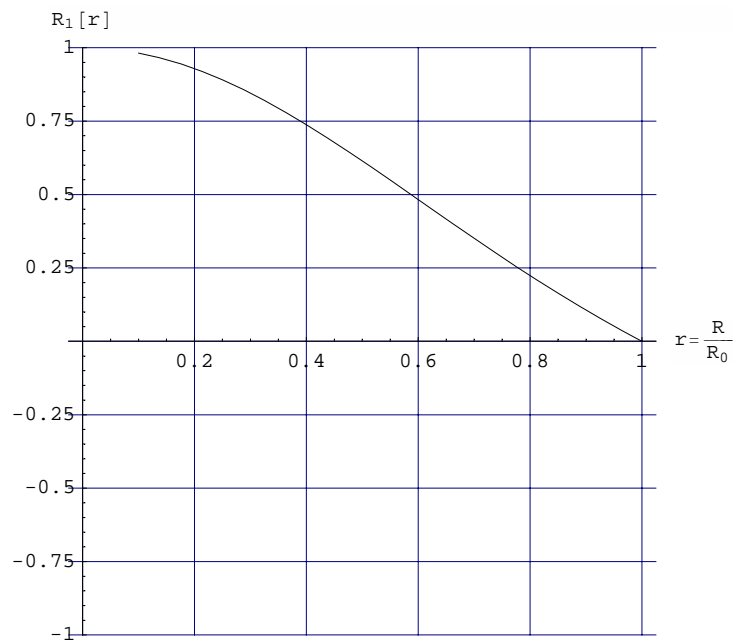
4j)  $R(0)=1$

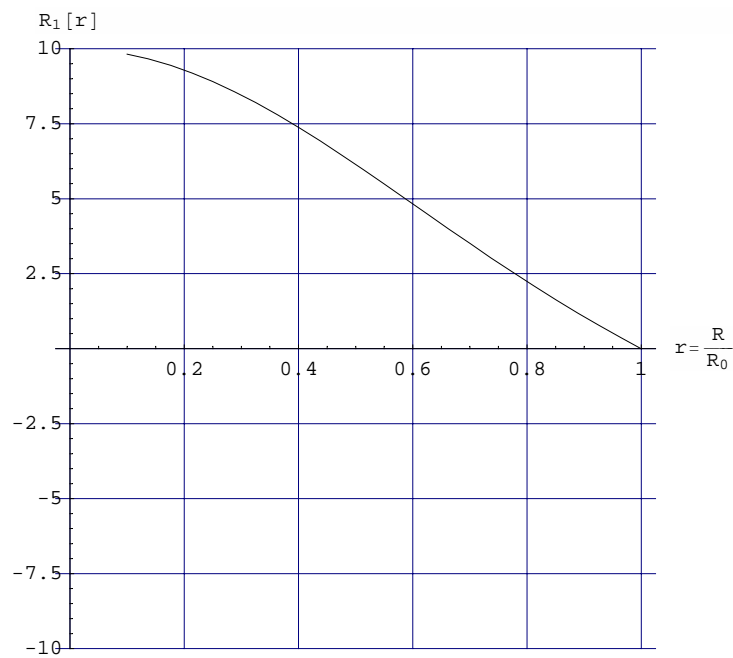
Il  $\lambda_n$  "esatto" per la  $R_n$  sarà quello che verifichi  $R_n(1)=0$

Dobbiamo imporre la 4i) perchè l'istruzione NDSolve risolve numericamente un'equazione richiedendo necessariamente una condizione iniziale che le jj) e jjj) non potevano darci.

Inoltre la 4j) ha un suo significato, nel senso che è strettamente relazionata alla SHAPE dell'autofunzione.

Come si può notare dai due grafici seguenti, abbiamo plottato la stessa autofunzione per due condizioni iniziali diverse: la prima con  $R(0)=1$ , la seconda con  $R(0)=10$ . La forma delle autofunzioni è esattamente la stessa, però sono cambiate le  $C_n$ ; invece le  $R_n(r)$  non cambiano.





Per  $R(0)=1$  l'equazione da soddisfare è la **condizione iniziale**  $\sum_{n=0}^{\infty} C_n R_n(0)=1$ , però se scrivevo  $R(0)=10$ , l'equ. era  $\sum_{n=0}^{\infty} B_n R_n(0)=10$ . Ciò significa che le costanti cambiano perchè sono le responsabili di un fattore di scala, invece le  $R_n$  sono sempre le stesse.

Per semplicità di calcolo abbiamo scelto che le autofunzioni partissero da 1 ad  $r=0$

Una volta conosciuti i  $\lambda$  autovalori ci calcoliamo le rispettive autofunzioni con il ciclo seguente

```
i = .
```

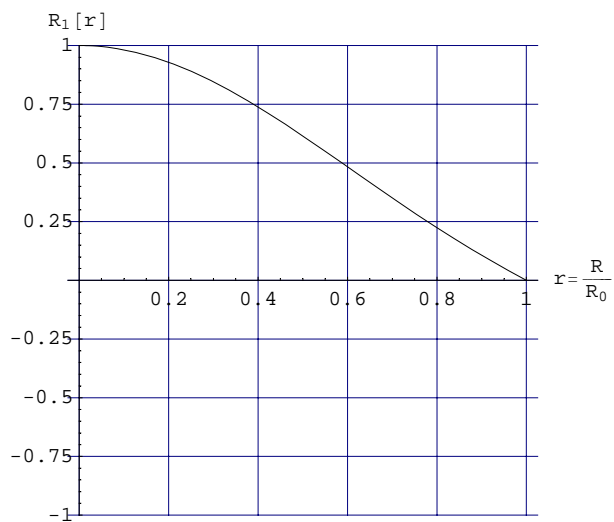
```
For[i = 1, i ≤ 12,
```

```
  sol_i = NDSolve[{ R_i''[r] + R_i'[r] / r + λ[[i]]^2 R_i[r] (1 - r^2) == 0, R_i[0.0001] == 1,
    R_i'[0.0001] == 0}, R_i, {r, 0.001, 1.1}]; R_i = R_i /. sol_i[[1]]; i++]
```

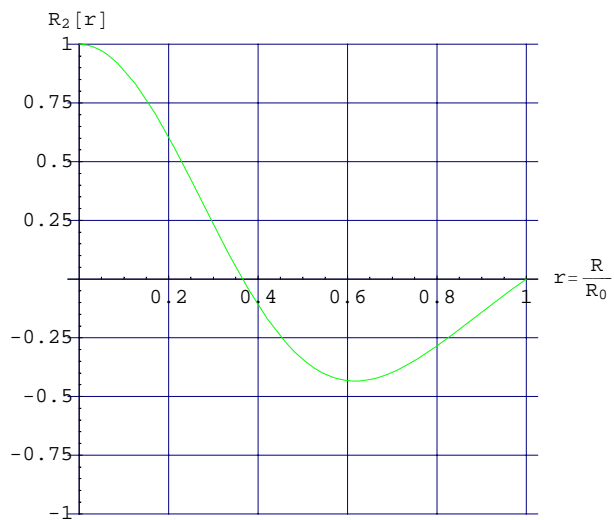
```
R_3[0.5]
```

```
-0.142346
```

```
p1 = Plot[R1[r], {r, 0.001, 1}, PlotRange → {-1, 1},  
  AspectRatio → 1, GridLines → Automatic, AxesLabel → {"r =  $\frac{R}{R_0}$ ", "R1[r]"}];
```



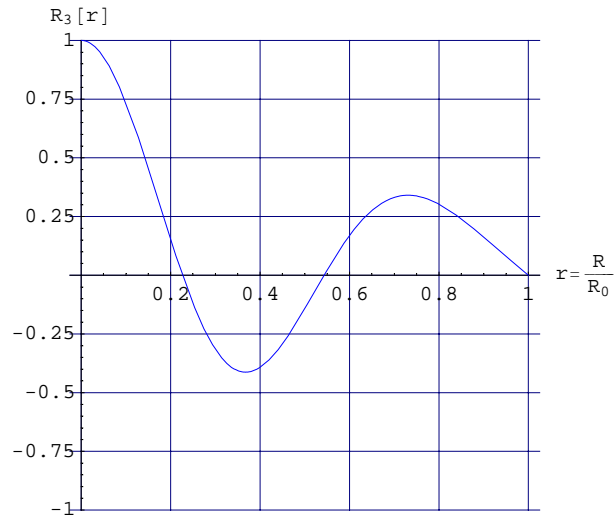
```
p2 = Plot[Evaluate[R2[r]], {r, 0.001, 1},  
  PlotRange → {-1, 1}, AspectRatio → 1, GridLines → Automatic,  
  PlotStyle → RGBColor[0, 1, 0], AxesLabel → {"r =  $\frac{R}{R_0}$ ", "R2[r]"}];
```



```

p3 = Plot[Evaluate[R3[r]], {r, 0.001, 1},
  PlotRange → {-1, 1}, AspectRatio → 1, GridLines → Automatic,
  PlotStyle → RGBColor[0, 0, 1], AxesLabel → {"r =  $\frac{R}{R_0}$ ", "R3[r]"}];

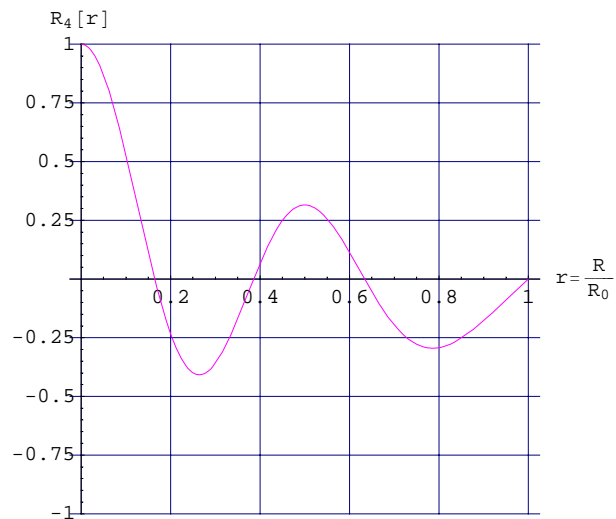
```



```

p4 = Plot[Evaluate[R4[r]], {r, 0.001, 1},
  PlotRange → {-1, 1}, AspectRatio → 1, GridLines → Automatic,
  PlotStyle → RGBColor[1, 0, 1], AxesLabel → {"r =  $\frac{R}{R_0}$ ", "R4[r]"}];

```

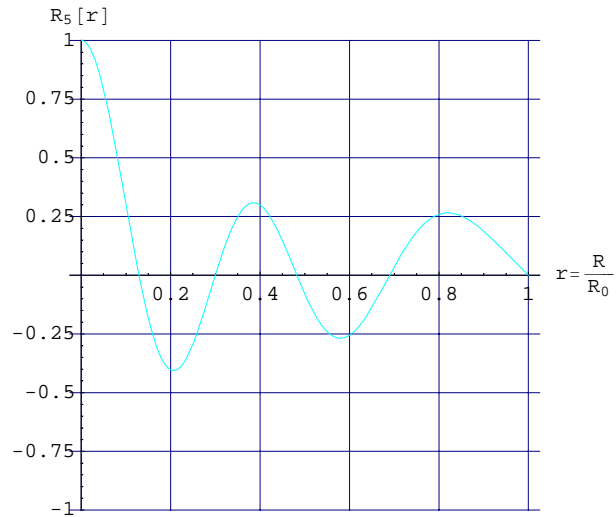




```

p5 = Plot[Evaluate[R5[r]], {r, 0.001, 1},
  PlotRange -> {-1, 1}, AspectRatio -> 1, GridLines -> Automatic,
  PlotStyle -> RGBColor[0, 1, 1], AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R5[r]"}];

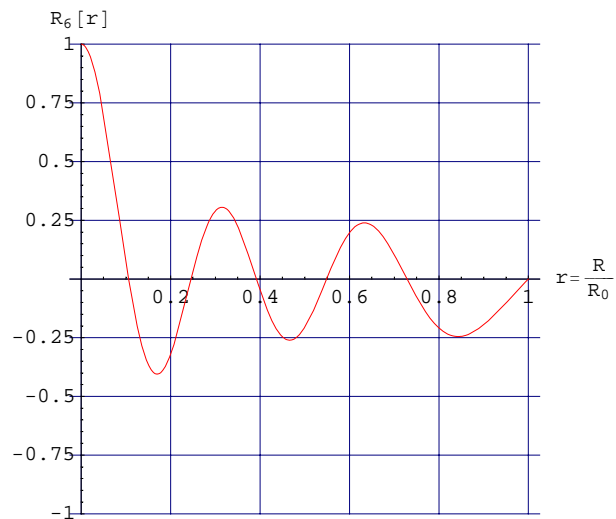
```



```

p6 = Plot[Evaluate[R6[r]], {r, 0.001, 1},
  PlotRange -> {-1, 1}, AspectRatio -> 1, GridLines -> Automatic,
  PlotStyle -> RGBColor[1, 0, 0], AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R6[r]"}];

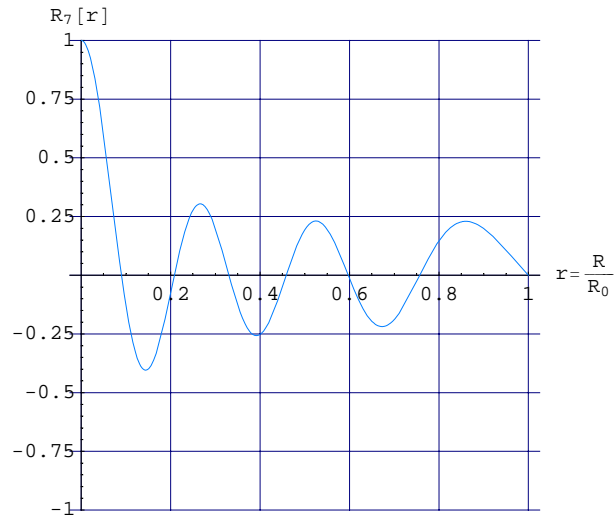
```



```

p7 = Plot[Evaluate[R7[r]], {r, 0.001, 1},
  PlotRange -> {-1, 1}, AspectRatio -> 1, GridLines -> Automatic,
  PlotStyle -> RGBColor[0, 0.5, 1], AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R7[r]"}];

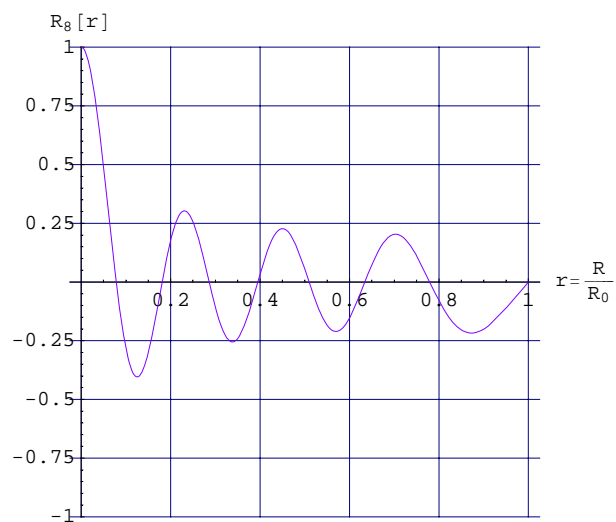
```



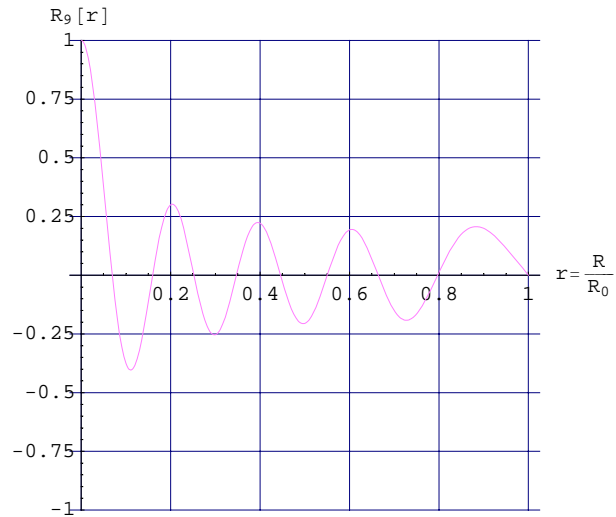
```

p8 = Plot[Evaluate[R8[r]], {r, 0.001, 1},
  PlotRange -> {-1, 1}, AspectRatio -> 1, GridLines -> Automatic,
  PlotStyle -> RGBColor[0.5, 0, 1], AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R8[r]"}];

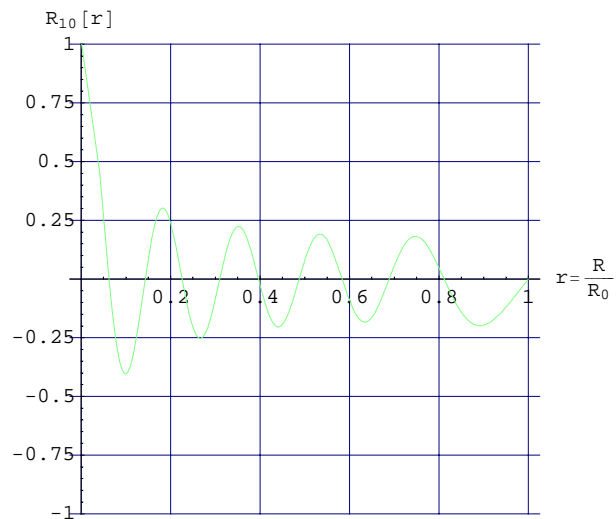
```



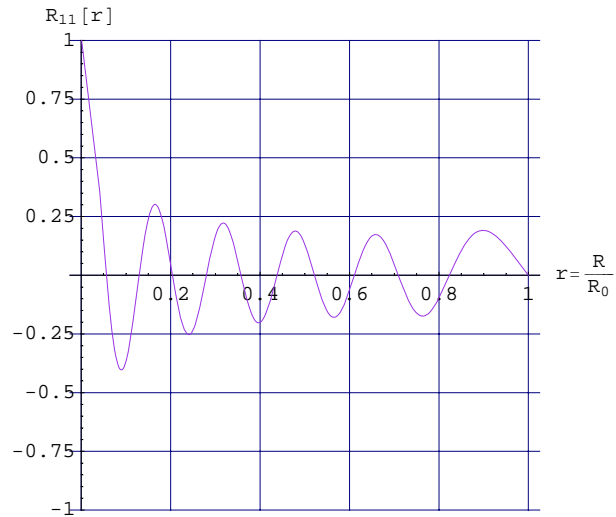
```
p9 = Plot[Evaluate[R9[r]], {r, 0.001, 1},  
PlotRange -> {-1, 1}, AspectRatio -> 1, GridLines -> Automatic,  
PlotStyle -> RGBColor[1, 0.5, 1], AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R9[r]"}];
```



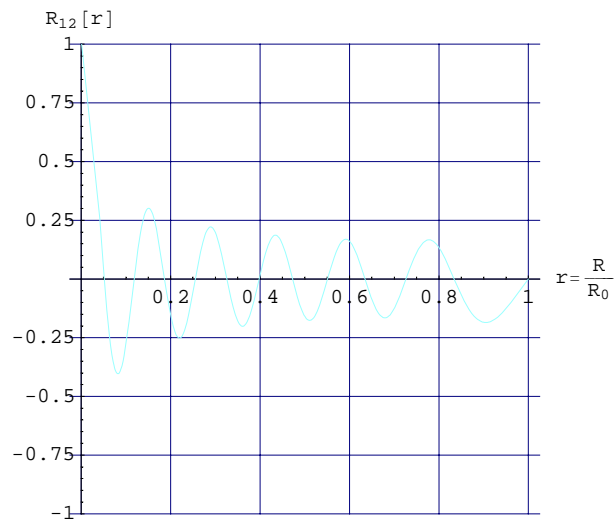
```
p10 = Plot[Evaluate[R10[r]], {r, 0.001, 1},  
PlotRange -> {-1, 1}, AspectRatio -> 1, GridLines -> Automatic,  
PlotStyle -> RGBColor[0.5, 1, 0.5], AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R10[r]"}];
```



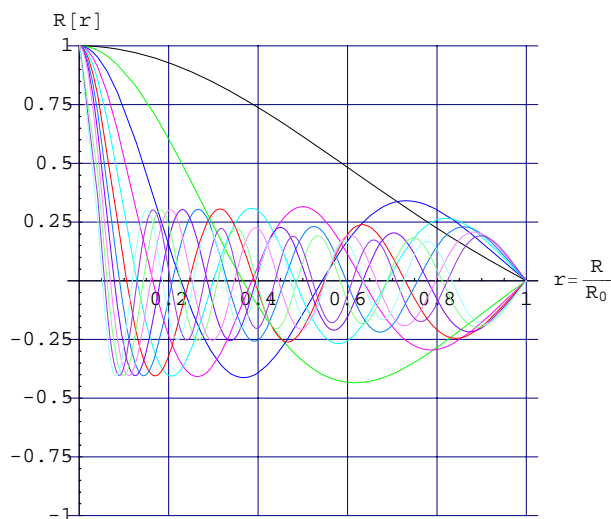
```
p11 = Plot[Evaluate[R11[r]], {r, 0.001, 1},  
  PlotRange -> {-1, 1}, AspectRatio -> 1, GridLines -> Automatic,  
  PlotStyle -> RGBColor[0.6, 0.2, 0.9], AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R11[r]"}];
```



```
p12 = Plot[Evaluate[R12[r]], {r, 0.001, 1},  
  PlotRange -> {-1, 1}, AspectRatio -> 1, GridLines -> Automatic,  
  PlotStyle -> RGBColor[0.6, 1, 1], AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R12[r]"}];
```



```
Show[p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, AxesLabel -> {"r =  $\frac{R}{R_0}$ ", "R[r]"}];
```



Per cercare le costanti del problema agli autovalori di Graetz (l'equazione 8i) si deve introdurre la teoria del Problema ai valori al contorno di Sturm-Liouville la quale ci permetterà di trovare la funzione peso del proble agli autovalori 8i). Un'equazione differenziale definita sull'intervallo  $a \leq x \leq b$  avente la forma:

$$i) \quad \frac{d}{dx} \left[ p(x) \frac{dy}{dx} \right] + [q(x) + \lambda r(x)] y = 0$$

e le condizioni al contorno

$$ii) \quad \begin{cases} a_1 y(a) + a_2 y'(a) = 0 \\ b_1 y(b) + b_2 y'(b) = 0 \end{cases}$$

Si può notare che la i) per  $p(x) = 1, r(x)=1, q(x)=0$   $\frac{d^2 y}{dx^2} + \lambda y = 0$  che è la formula che consente di trovare autovalori e autovettori per la lastra piana

i)+ii) è chiamato SISTEMA di Sturm-Liouville

con  $p(x) > 0$  necessariamente,  $r(x)$  FUNZIONE PESO,  $a_1, a_2$  e  $b_1, b_2$  sono delle costanti note,  $\lambda$  autovalori incogniti

Le soluzioni di questo problema sono autofunzioni  $\phi_n(x)$  legate a  $\lambda_n$  autovalori

I  $\lambda_n$  sono numeri reali positivi

Le  $\phi_n(x)$  rispettano la condizione di ORTOGONALITA' per cui devono verificare la condizione:

$$\int_a^b r(x) \phi_n(x) \phi_m(x) dx = 0 \quad \text{se } m \neq n \text{ e con}$$

$m, n = 1, 2, 3, \dots$

La i) opportunamente modificata con  $p(x)=x; q(x)=0$  diventa

$$I) \quad y'(x) + x y''(x) + \lambda r(x) y$$

che risulta identica alla nostra  $R''(r) + \frac{1}{r} R'(r) + \lambda^2 R(r)(1-r^2) = 0$

$$\text{I) } rR'' + R' + \lambda^2 R r (1-r^2) = 0 \quad \text{con la funzione peso pari a } "r(1-r^2)"$$

Le I) hanno la stessa forma

verifichiamo che il peso è  $r(1-r^2)$  facendo  $\int_0^1 R_i R_j r(1-r^2) dx = 0$

```
NIntegrate[R2[r] R3[r] r (1 - r2), {r, 0.001, 1}]
```

1.23853 × 10<sup>-6</sup>

Dall'equazione  $\theta(x,r) = \sum_{n=0}^{\infty} C_n R_n(r) X(x)$  particolarizzata per la condizione iniziale  $X(0)R(r)=1$  abbiamo

III)  $\sum_{n=0}^{\infty} C_n R_n(r) = 1$  della quale le incognite ormai sono le  $C_n$

Per la ricerca delle costanti sfruttiamo la condizione di ortogonalità delle autofunzioni quindi moltiplichiamo per una generica  $R_m(r)$  e integriamo tra 0 e 1 in dr.

$$B(r) = r(1-r^2) \quad \text{funzione peso}$$

$$\int_0^1 B(r) \sum_{n=0}^{\infty} C_n R_n(r) R_m(r) dr = \int_0^1 B(r) R_m(r) 1 dr$$

del primo membro ho  $\infty$  addendi dei quali sopravvive solo quello con  $m=n$

$$C_m = \frac{\int_0^1 B(r) R_m(r) 1 dr}{\int_0^1 B(r) R_m^2(r) dr}$$

```
c1 = NIntegrate[1 r (1 - r2) R1[r], {r, 0.0001, 1}] / NIntegrate[R1[r] R1[r] r (1 - r2), {r, 0.0001, 1}]
```

1.47643

Quindi ci calcoliamo le costanti

Le costanti  $C_n$  per comodità le chiamiamo  $K_n$

```
j = 1; While[j ≤ 12, k[j] = NIntegrate[1 r (1 - r2) Rj[r], {r, 0.001, 1}] / NIntegrate[Rj[r] Rj[r] r (1 - r2), {r, 0.001, 1}]; j = j + 1]
```

```
k
```

```
{1.47644, -0.806184, 0.588748, -0.475863, 0.405006, -0.355789,  
0.319162, -0.290764, 0.267815, -0.249145, 0.233168, -0.219813}
```

```
k[5]
```

0.405029

```
λ[4]
```

14.6709

Length[ $\lambda$ ]

12

Length[k]

12

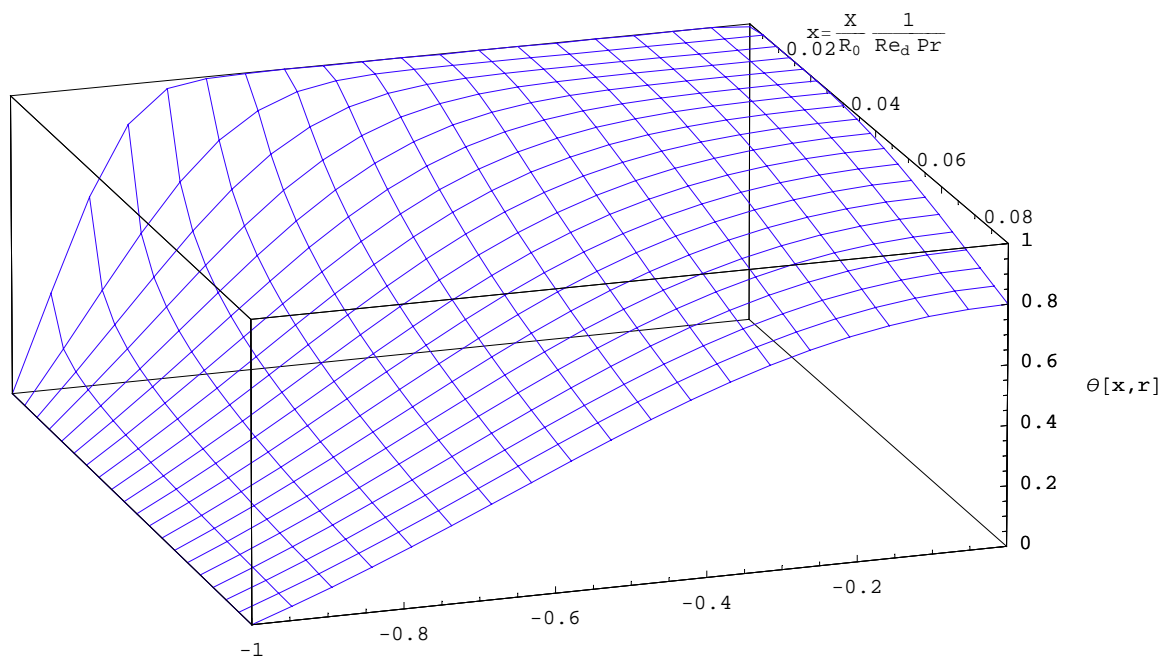
$$\theta[x_, r_] := \sum_{n=1}^{12} k[[n]] R_n[r] E^{-(\lambda[[n]])^2 x}$$

$$\theta_1[x_, r_] := \sum_{n=1}^{12} k[[n]] R_n[-r] E^{-(\lambda[[n]])^2 x}$$

<<Graphics`Graphics3D`

Nel plottare teniamo conto che  $r = \frac{R}{R_0} = \frac{2}{2} \frac{R}{R_0} = \frac{d}{d_0}$

```
P1 = Plot3D[ $\theta_1[x, r]$ , {x, 0.001, 0.08}, {r, -1, -0.001}, PlotRange -> {0, 1},
  Axes -> True, AxesLabel -> {"x =  $\frac{x}{R_0} \frac{1}{Re_d \setminus Pr}$ ", r =  $\frac{R}{R_0}$  ""}, " $\theta[x, r]$ "},
  Boxed -> True, LightSources -> {{{1.3, -2.4, 2.}, RGBColor[1, 0, 0]},
  {{2.7, 0., 2.}, RGBColor[0, 1, 0]}, {{-2.3, -1.4, 2.}, RGBColor[1, 1, 1]}},
  ViewPoint -> {30, -10, 10}, Mesh -> True, PlotPoints -> 20,
  MeshStyle -> {Hue[1.7]}, HiddenSurface -> False];
```

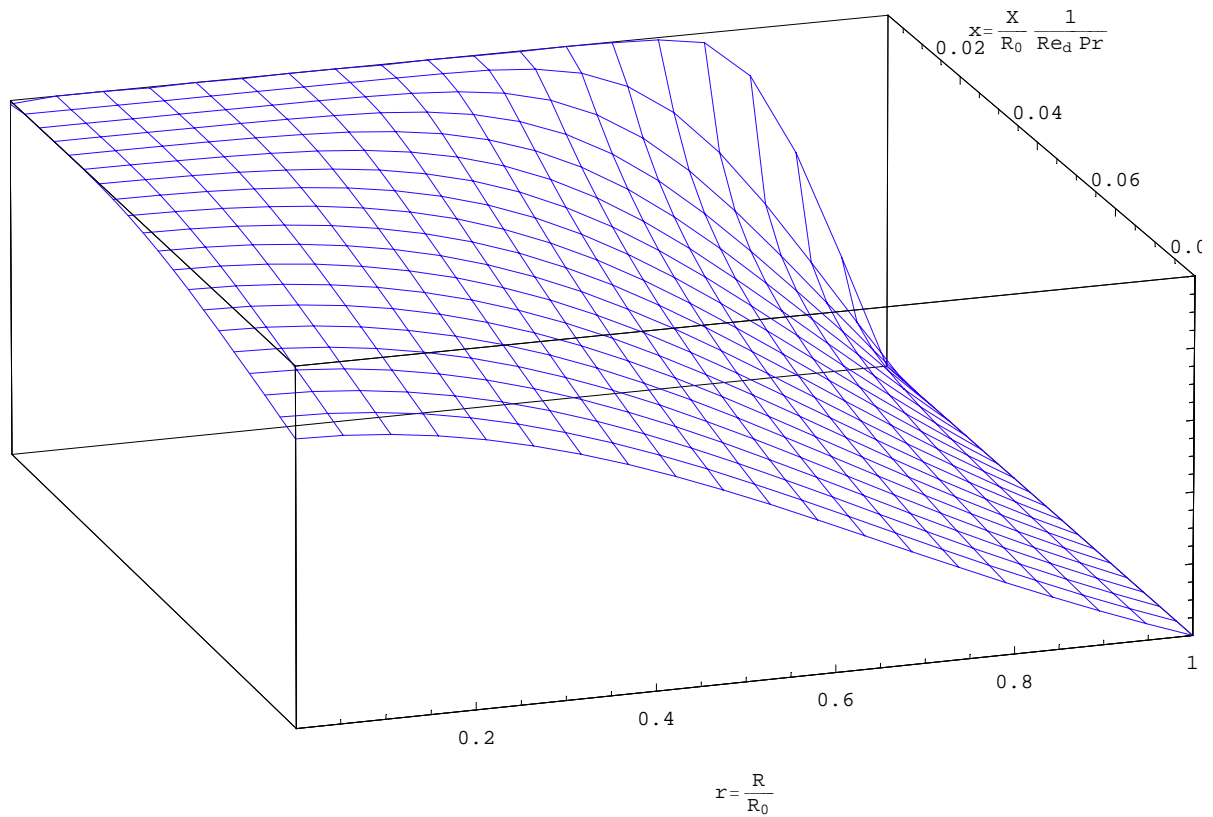


```
InterpolatingFunction[{{0.0001, 1.1}}, <>]  
(InterpolatingFunction[{{0.0001, 1.1}}, <>])_0
```

```

P2 = Plot3D[ $\theta[x, r]$ , {x, 0.001, 0.08}, {r, 0.001, 1}, PlotRange -> {0, 1},
  Axes -> True, AxesLabel -> {"x =  $\frac{x}{R_0} \frac{1}{Re_d \ Pr}$ ", "r =  $\frac{R}{R_0}$ ", " $\theta[x, r]$ "},
  Boxed -> True, LightSources -> {{1.3, -2.4, 2.}, RGBColor[1, 0, 0]},
  {{2.7, 0., 2.}, RGBColor[0, 1, 0]}, {{-2.3, -1.4, 2.}, RGBColor[1, 1, 1]}},
  ViewPoint -> {30, -10, 10}, Mesh -> True, PlotPoints -> 20,
  MeshStyle -> {Hue[1.7]}, HiddenSurface -> False];

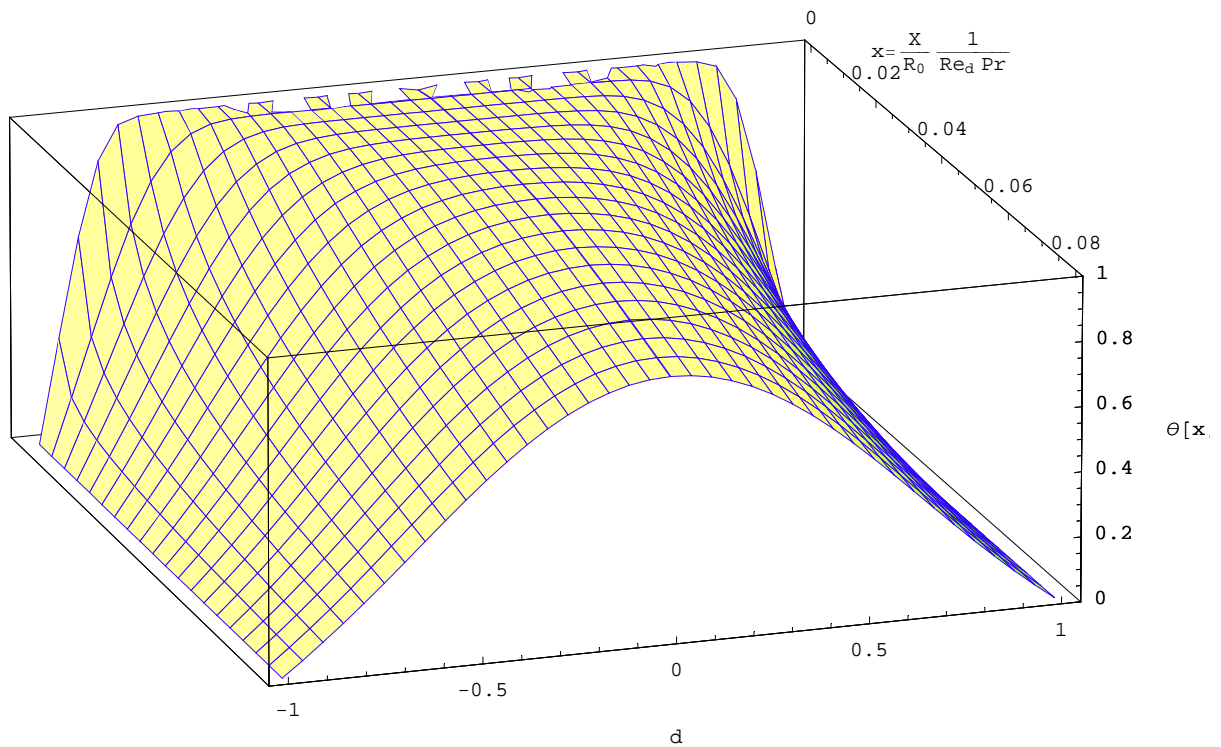
```



IL diagramma della temperatura adimensionale sarà:



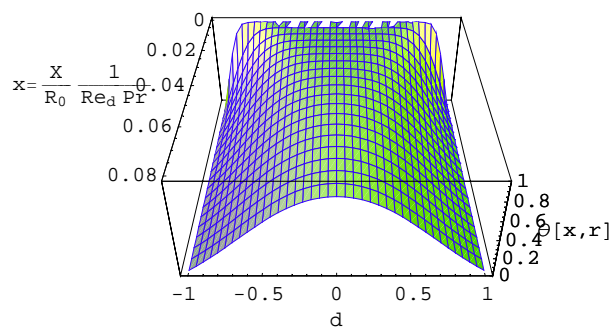
```
P3 = Show[P2, P1, AxesLabel -> {"x =  $\frac{x}{R_0} \frac{1}{Re_d \ Pr}$ ", "d", " $\theta[x,r]$ "}]
```

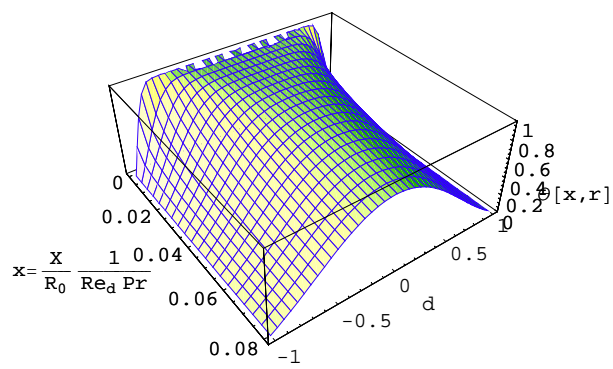
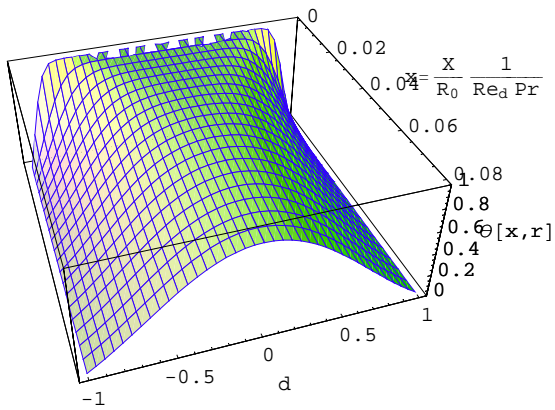


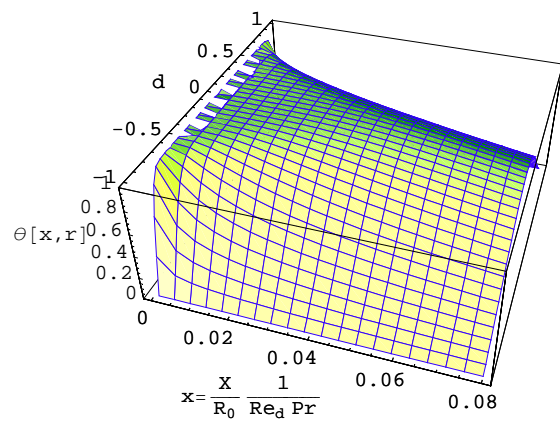
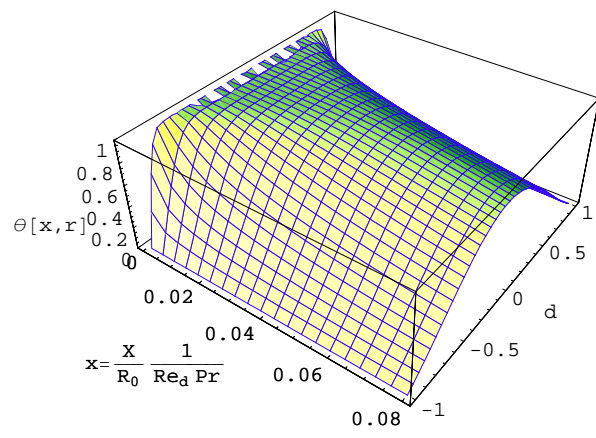
- Graphics3D -

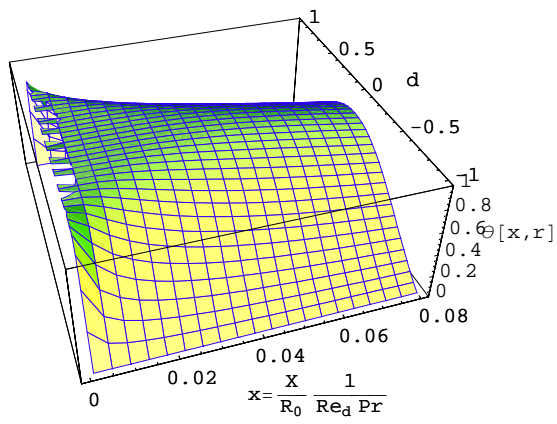
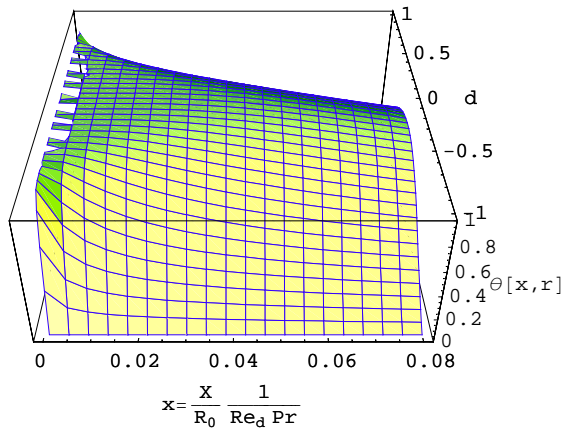
<<Graphics`Animation`

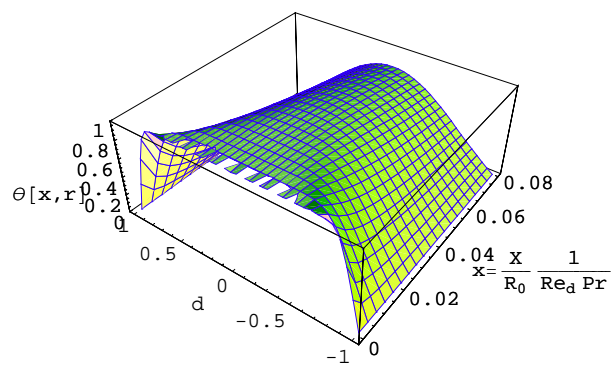
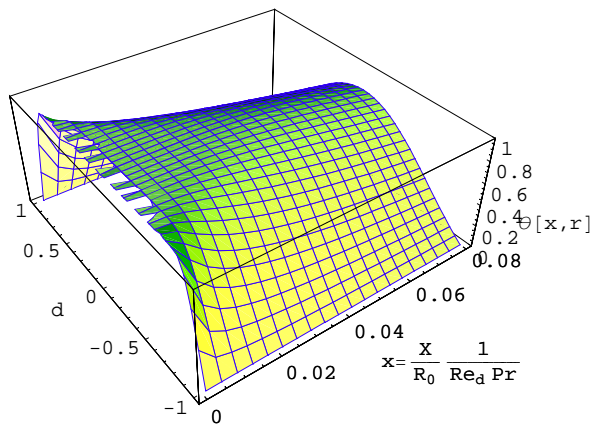
```
SpinShow[P3, Frames -> 10, SpinRange -> {0 Degree, 180 Degree}]
```

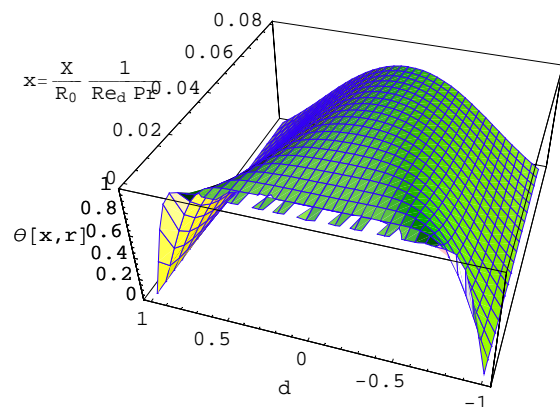












Per il calcolo del Nusselt ci servono le costanti  $G_n = -\frac{1}{2} K_n R_n'(1)$

$$G = \text{Table}\left[G_n = -\frac{1}{2} k[[n]] R_n'[1], \{n, 0, 12\}\right]$$

```
{0.72686, 0.537082, 0.459527, 0.413382, 0.381523, 0.357656,
0.338825, 0.323422, 0.310486, 0.299399, 0.289743, 0.281224}
```

Il Nusselt locale è

$$\text{Nux}[x_] := \frac{\sum_{n=1}^{12} G[[n]] E^{-(\lambda[[n]])^2 x}}{2 \sum_{n=1}^{12} \left(\frac{G[[n]]}{(\lambda[[n]])^2}\right) E^{-(\lambda[[n]])^2 x}}$$

**Nux[100]**

```
3.658512500000
```

**Nux[1]**

```
3.65851
```

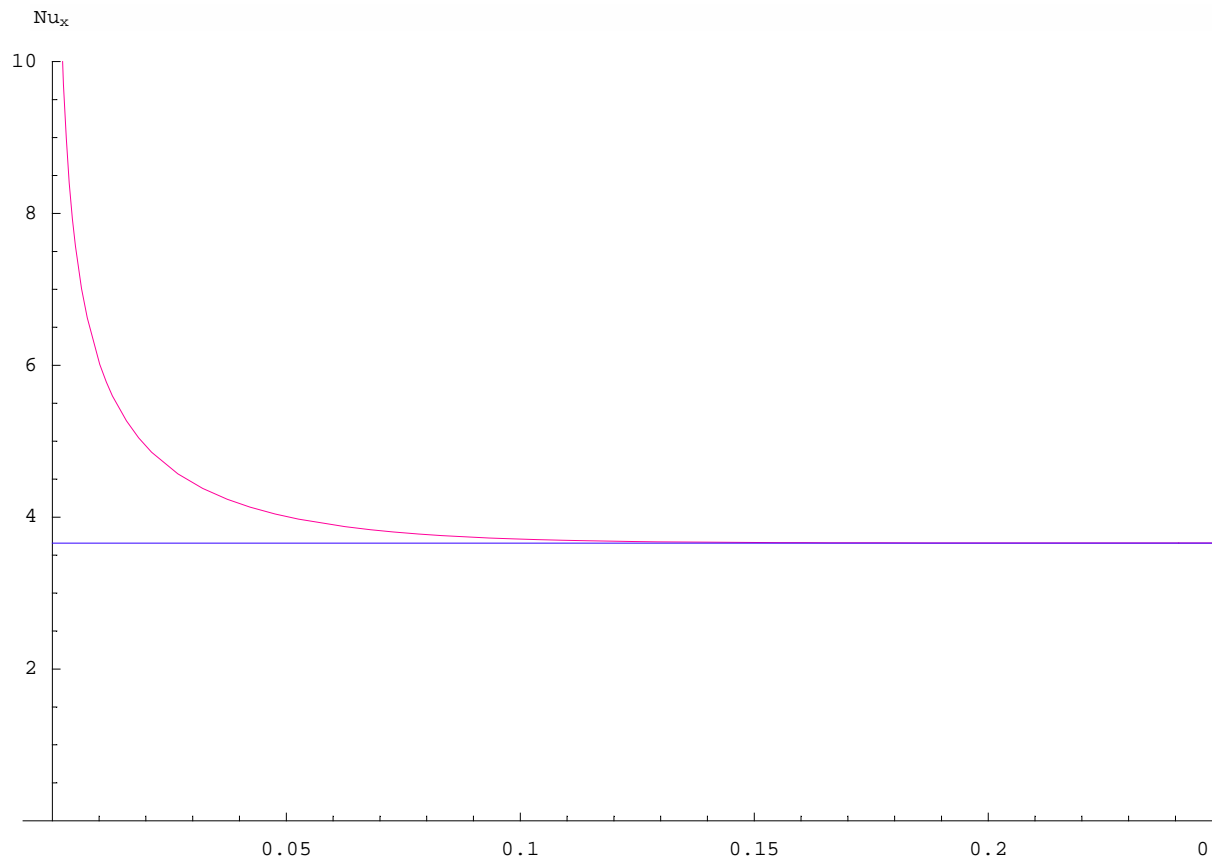
**Nux[0.5]**

```
3.65851
```

**Nux[0.1]**

```
3.7109
```

```
b1 = Plot[{Nux[x], 3.65851}, {x, 0, 0.25}, PlotRange -> {0, 10},
  AxesLabel -> {"x =  $\frac{x}{R_0} \frac{1}{Re_d Pr}$ ", "Nu_x"}, PlotStyle -> {Hue[0.9], Hue[1.7], Hue[0.7]}]
```



- Graphics -

Il Nusselt medio ha invece la forma

$$\text{Num}[x_] := \frac{1}{2x} \text{Log} \left[ \frac{1}{8 \sum_{n=1}^{12} \left( \frac{G[[n]]}{(\lambda[[n]])^2} \right) E^{-(\lambda[[n]])^2 x}} \right]$$

**Num[100]**

3.659661418215964

**Num[1]**

3.7734

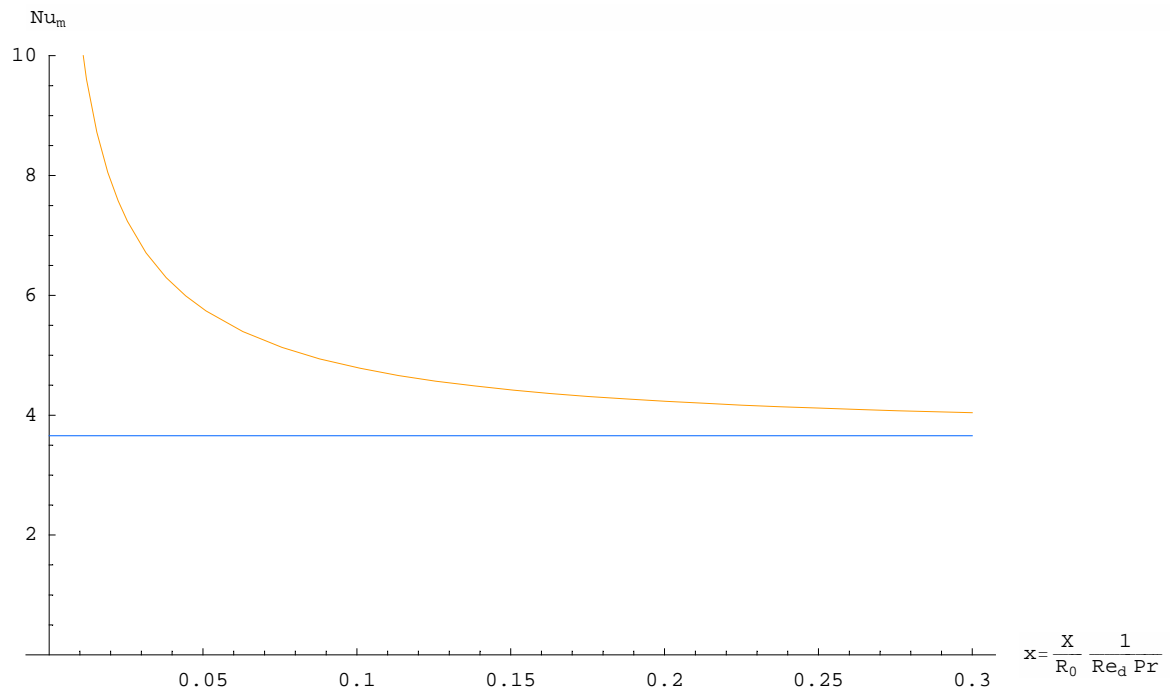
**Num[0.5]**

3.8883

**Num[0.1]**

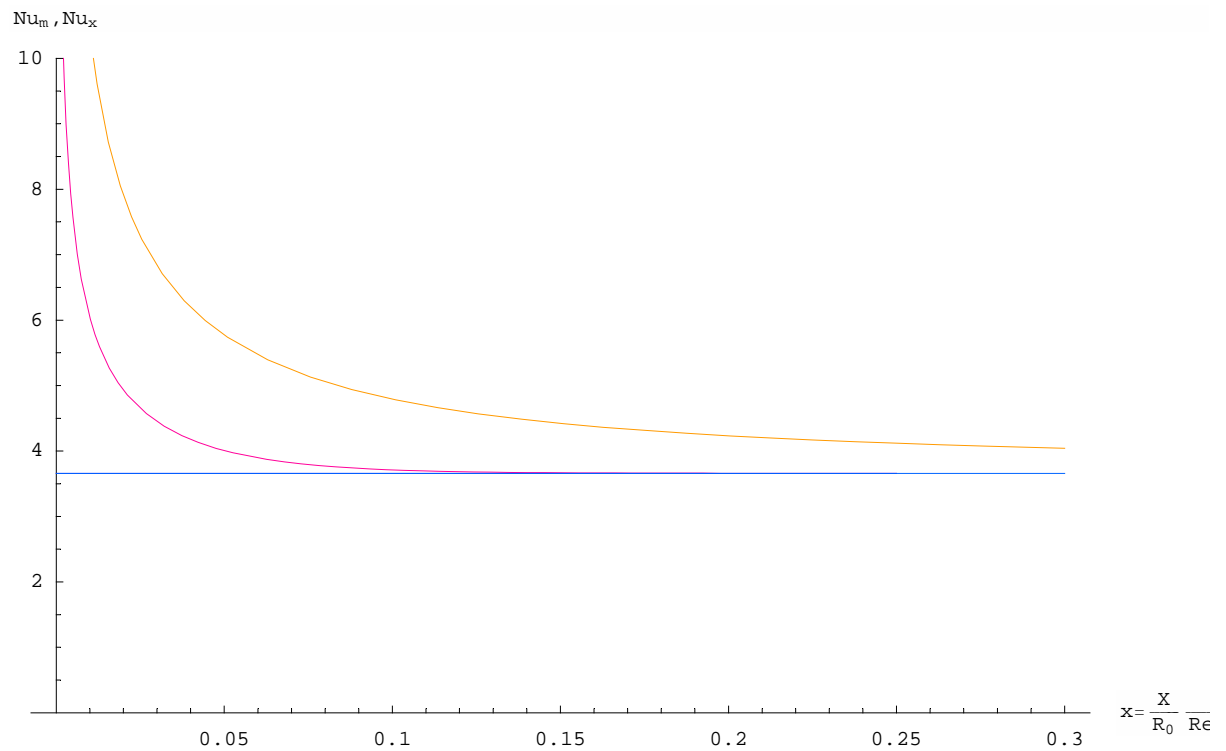
4.7935

```
b2 = Plot[{Num[x], 3.65851}, {x, 0, 0.3}, PlotRange -> {0, 10},
  AxesLabel -> {"x="  $\frac{x}{R_0} \frac{1}{Re_d \ Pr}$  ", "Num"}, PlotStyle -> {Hue[0.1], Hue[0.6], Hue[0.7]}]
```



- Graphics -

```
Show[b1, b2, AxesLabel -> {"x="  $\frac{x}{R_0} \frac{1}{Re_d \ Pr}$  ", "Num, Nux"}]
```



- Graphics -



Per il calcolo del Nusselt, in generale, si usano formule approssimate, una di queste è:

$$\text{Num}_a [x\_ ] := 3.6568 + \frac{0.0499}{\frac{x}{2}}$$

Questa formula è valida per  $x \geq 0.05$ . Si può notare che per  $x \rightarrow \infty$  essa ci da  $\text{Nu}=3.66$

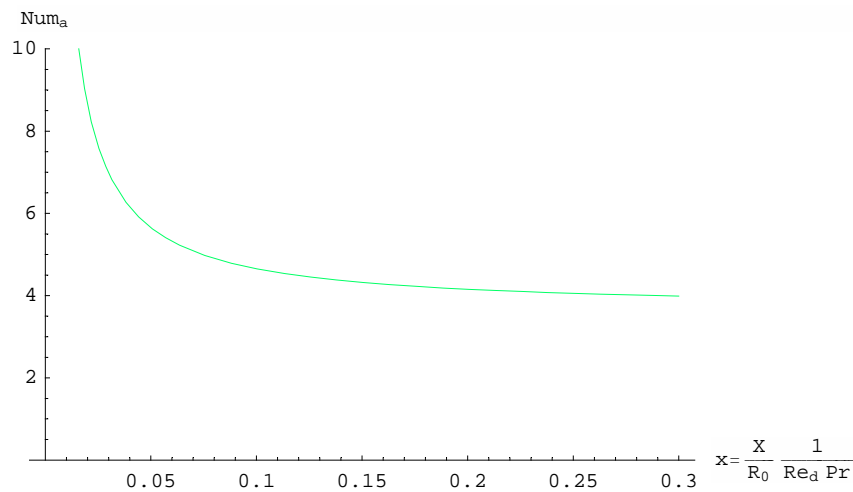
Il  $\text{Num}_a(x)$  è la soluzione asintotica della relazione che segue per  $x \geq 0.05$

Questa relazione approssimata riproduce il valore esatto con un errore del 1% nel range  $0 < x < \infty$  è:

$$\text{Num}_e [x\_ ] := \frac{3.657}{\text{Tanh}\left[2.264 \left(\frac{x}{2}\right)^{\frac{1}{3}} + 1.7 \left(\frac{x}{2}\right)^{\frac{2}{3}}\right]} + \frac{0.0499}{\frac{x}{2}} \text{Tanh}\left[\frac{x}{2}\right]$$

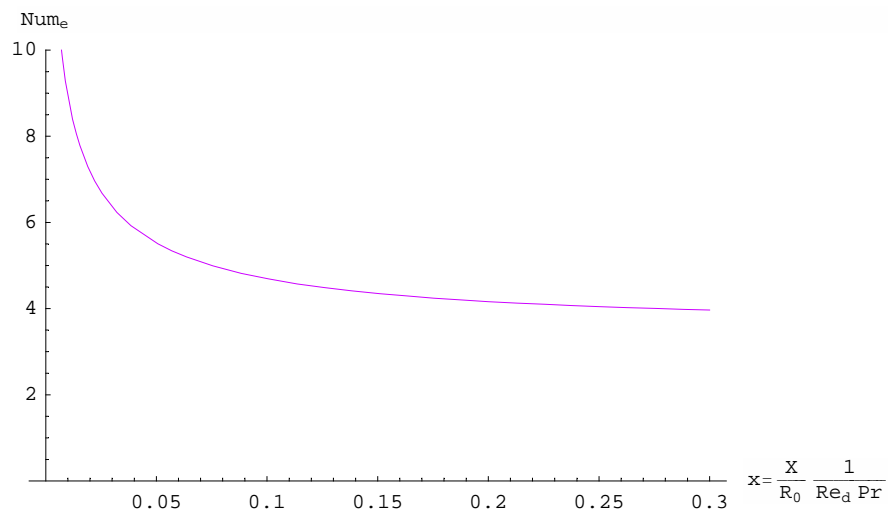
con  $x = \frac{X}{d \text{Pe}}$

```
b3 = Plot[{Num_a[x]}, {x, 0, 0.3}, PlotRange -> {0, 10},
  AxesLabel -> {"x = \frac{X}{R_0} \frac{1}{\text{Re}_d \ \text{Pr}}", "Num_a"}, PlotStyle -> {Hue[0.4], Hue[0.5], Hue[0.8]}]
```



- Graphics -

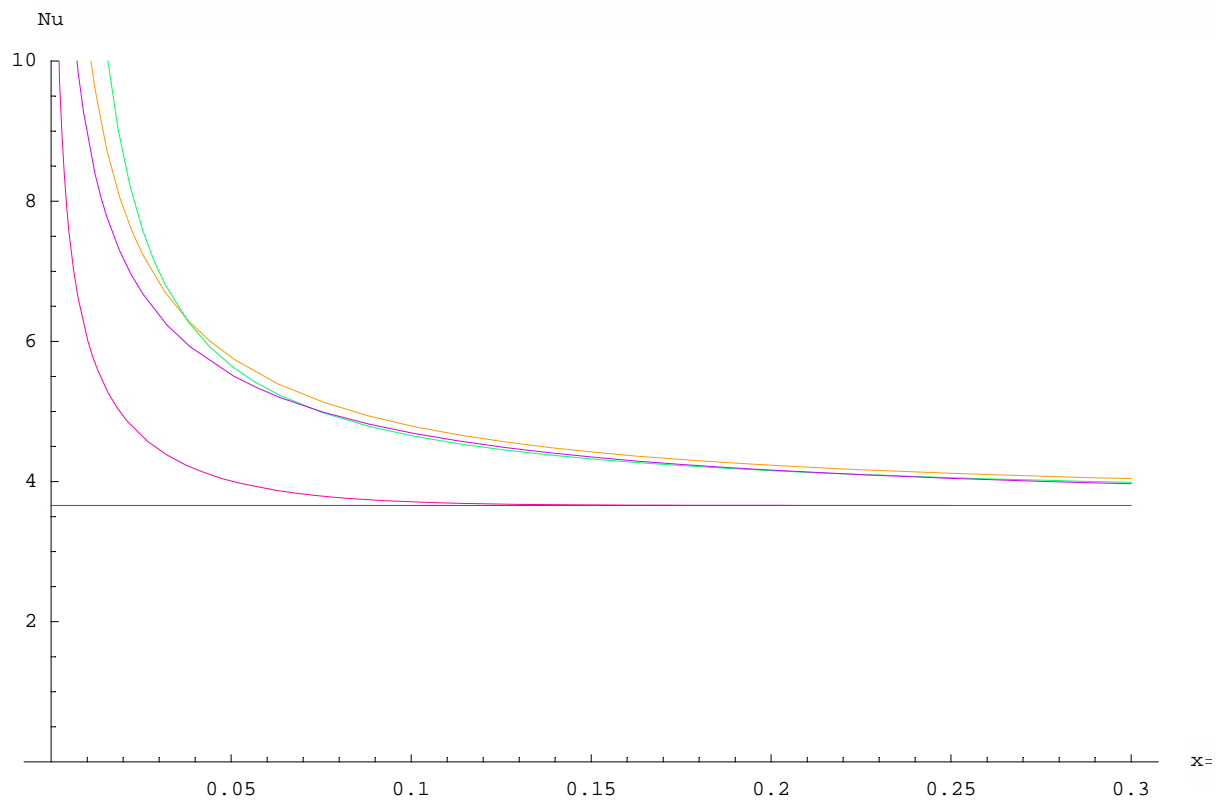
```
b4 = Plot[{Num_e[x]}, {x, 0, 0.3}, PlotRange -> {0, 10},
  AxesLabel -> {"x =  $\frac{x}{R_0} \frac{1}{Re_d Pr}$ ", "Num_e"}, PlotStyle -> {Hue[0.8], Hue[0.1], Hue[0.2]}]
```



- Graphics -

Ora plottiamo tutti i  $Nu_m$  per rapportarli a quello esatt locale in rosso

```
Show[b1, b2, b3, b4, AxesLabel -> {"x =  $\frac{x}{R_0} \frac{1}{Re_d \setminus Pr}$ ", "Nu"}]
```



- Graphics -

