Computação Gráfica

Aula 9.2:
Desenhando Superfícies Curvas Bicúbicas com Diferenças Adiante

Prof. Dr. rer.nat. Aldo von Wangenheim

Departamento de Informática e Estatística - INE/CTC/UFSC

Conceitos de Desenho de Superfícies Bicúbicas

- **Superfícies bicúbicas paramétricas podem ser desenhadas utilizando-se diferenças adiante (forward differences) da mesma maneira que curvas cúbicas**
- **Para isso vamos ter de aplicar forward differences de maneira bem específica a cada retalho da superfície.**
- **Maior desafio: Precisamos determinar, de maneira independente, as *condições iniciais* para cada curva indvidual que vai compor a superfície.**

Relembrando: Conceitos Básicos de Superfícies Bicúbicas

$$Q(s,t) = S \cdot M \cdot \begin{bmatrix} \mathbf{g}_{11} & \mathbf{g}_{12} & \mathbf{g}_{13} & \mathbf{g}_{14} \\ \mathbf{g}_{21} & \mathbf{g}_{22} & \mathbf{g}_{23} & \mathbf{g}_{24} \\ \mathbf{g}_{31} & \mathbf{g}_{32} & \mathbf{g}_{33} & \mathbf{g}_{34} \\ \mathbf{g}_{41} & \mathbf{g}_{42} & \mathbf{g}_{43} & \mathbf{g}_{44} \end{bmatrix} \cdot M^T \cdot T^T$$

Forma geral de uma Bicúbica ☞

Ou, representando de forma mais compacta:

$$Q(s,t) = S \cdot M \cdot \mathbf{G} \cdot M^T \cdot T^T, \quad 0 \le s,t \le 1$$

Relembrando: Conceitos Básicos de Superfícies Bicúbicas

- **Se reescrevermos a eq. anterior de forma separada para cada coordenada x,y e z, teremos a forma geral de uma superfície bicúbica:**

$$x(s,t) = S \cdot M \cdot \mathbf{G_x} \cdot M^T \cdot T^T$$

$$y(s,t) = S \cdot M \cdot \mathbf{G_y} \cdot M^T \cdot T^T, \quad 0 \leq s,t \leq 1$$

$$z(s,t) = S \cdot M \cdot \mathbf{G_z} \cdot M^T \cdot T^T$$

Relembrando: Forward Differences para Curvas Cúbicas

- Forward differences são calculadas como somas de derivadas.
- A fórmula 4.38 nos dá as condições iniciais de uma curva, que representam a taxa de variação quando $t = 0$:

$$
\begin{aligned}
f_0 &= d \\
\Delta f_0 &= a\delta^3 + b\delta^2 + c\delta \\
\Delta^2 f_0 &= 6a\delta^3 + 2b\delta^2 \\
\Delta^3 f_0 &= 6a\delta^3
\end{aligned}
\qquad \textbf{(EQ. 4.38)}
$$

Relembrando: Forward Differences para Curvas Cúbicas

- **A fórmula 4.39 reescreve as condições iniciais como uma multiplicação matricial entre os coeficientes C e a matriz E(δ_t), que representa a taxa de variação quando t = 0:**

$$D = \begin{bmatrix} f_0 \\ \Delta f_0 \\ \Delta^2 f_0 \\ \Delta^3 f_0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ \delta^3 & \delta^2 & \delta & 0 \\ 6\delta^3 & 2\delta^2 & 0 & 0 \\ 6\delta^3 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = E(\delta) \cdot C \qquad \text{(EQ. 4.39)}$$

Relembrando: Forward Differences para Curvas Cúbicas

- **A matriz E(δ) temos: São as derivadas para t= 0.**
- **Temos de achar a matriz dos coeficientes C. Calculamos através de C = M . G:**

$$x = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \cdot \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} p_{0x} \\ p_{1x} \\ p_{2x} \\ p_{3x} \end{bmatrix}$$

$$x = \mathbf{t} \cdot \mathbf{B}_{Bez} \cdot \mathbf{g}_x$$

$$x = \mathbf{t} \cdot \mathbf{c}_x$$

Forward Differences para Superfícies

- **Fórmula geral (ex. Bézier):**

$$\boxed{x(s,t) = \mathbf{s} \cdot \mathbf{B}_{Bez} \cdot \mathbf{G}_x \cdot \mathbf{B}_{Bez}^T \cdot \mathbf{t}^T}$$

$$\mathbf{s} = \begin{bmatrix} s^3 & s^2 & s & 1 \end{bmatrix}$$
$$\mathbf{t} = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}$$

Forward Differences para Superfícies

- Podemos reescrever (ex. Bézier):

$$\mathbf{x}(s,t) = \begin{bmatrix} \mathbf{s} \cdot \mathbf{C}_x \cdot \mathbf{t}^T \\ \mathbf{s} \cdot \mathbf{C}_y \cdot \mathbf{t}^T \\ \mathbf{s} \cdot \mathbf{C}_z \cdot \mathbf{t}^T \end{bmatrix}$$

- Com isso calculamos os coeficientes da superfície!

$$\boxed{\mathbf{C}_x = \mathbf{B}_{Bez} \cdot \mathbf{G}_x \cdot \mathbf{B}_{Bez}^T}$$

$$\mathbf{s} = \begin{bmatrix} s^3 & s^2 & s & 1 \end{bmatrix}$$

$$\mathbf{t} = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}$$

Forward Differences para Superfícies

- O resto é conhecido e constante para um retalho de superfície (ex. Bézier):
  - Obs.: Hermite não é simétrica!
- Agora só precisamos calcular as condições iniciais!
- Vamos necessitar de um equivalente ao vetor D

$$\mathbf{B}_{Bez} = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} = \mathbf{B}_{Bez}^{T}$$

$$\mathbf{G}_x = \begin{bmatrix} p_{0x} & p_{4x} & p_{8x} & p_{12x} \\ p_{1x} & p_{5x} & p_{9x} & p_{13x} \\ p_{2x} & p_{6x} & p_{10x} & p_{14x} \\ p_{3x} & p_{7x} & p_{11x} & p_{15x} \end{bmatrix}$$

Forward Differences para Superfícies

- **Para calcular as condições iniciais de uma superfície necessitamos das matrizes dos $E(\delta_t)$ e dos $E(\delta_s)$!**
- **Para superfície: $DD(s,t) = E(\delta_s) \cdot C \cdot E(\delta_t)^T$**
- **Se usarmos o mesmo passo e $\delta_s = \delta_t$:**

$$E(\delta s) = \begin{bmatrix} 0 & 0 & 0 & 1 \\ \delta^3 & \delta^2 & \delta & 0 \\ 6\delta^3 & 2\delta^2 & 0 & 0 \\ 6\delta^3 & 0 & 0 & 0 \end{bmatrix} = E(\delta t)$$

7.8. Forward Differences para Superfícies

- Na fórmula:

$$DD(s,t) = E(\delta_s) \cdot C \cdot E(\delta_t)^T$$

- DD(s,t) será um matriz 4x4 e não mais um vetor coluna, como era D(t).
- Vamos chamar DD(s,t) de *matriz das condições iniciais*.
- Ela é a chave para usar Forward Differences com superfícies bicúbicas!

Forward Differences para Superfícies

- **A *matriz das condições iniciais DD(s,t)* pode ser interpetada assim:**
  - **1ª linha contém as condições iniciais para f(0, t)**
  - **1ª coluna contém as condições iniciais para f(s,0)**

$$\begin{bmatrix} f(0,0) & \Delta_t f(0,0) & {\Delta_t}^2 f(0,0) & {\Delta_t}^3 f(0,0) \\ \Delta_s f(0,0) & & & \\ {\Delta_s}^2 f(0,0) & & & \\ {\Delta_s}^3 f(0,0) & & & \end{bmatrix}$$

Forward Differences para Superfícies

Para usar a *matriz das condições iniciais DD(s,t)* para calcular a superfície:

1. **Desenhamos a superfície curva-a-curva. Usamos a primeira linha de DD(s,t) para prover as condições iniciais da primeira curva em *t* para o algoritmo já visto:**
   `DesenhaCurvaFwdDiff( n, x, Δx, Δ²x, Δ³x, y, Δy, Δ²y, Δ³y, z, Δz, Δ²z, Δ³z)`
2. **Atualizamos DD(s,t) para gerar as condições iniciais da próxima curva em t ao longo de s: somamos as linhas de cima para baixo.**
3. **Desenhamos a próxima curva, até completar n_s curvas.**
4. **Reinicializamos a matriz DD(s,t) e transpomos, para a primeira linha agora representar as condições iniciais em s.**
5. **Repetimos os passos 1,2 e 3, até completar n_t curvas.**

Forward Differences para Superfícies

Como atualizamos DD(s,t) por linhas para gerar as condições iniciais da próxima curva?

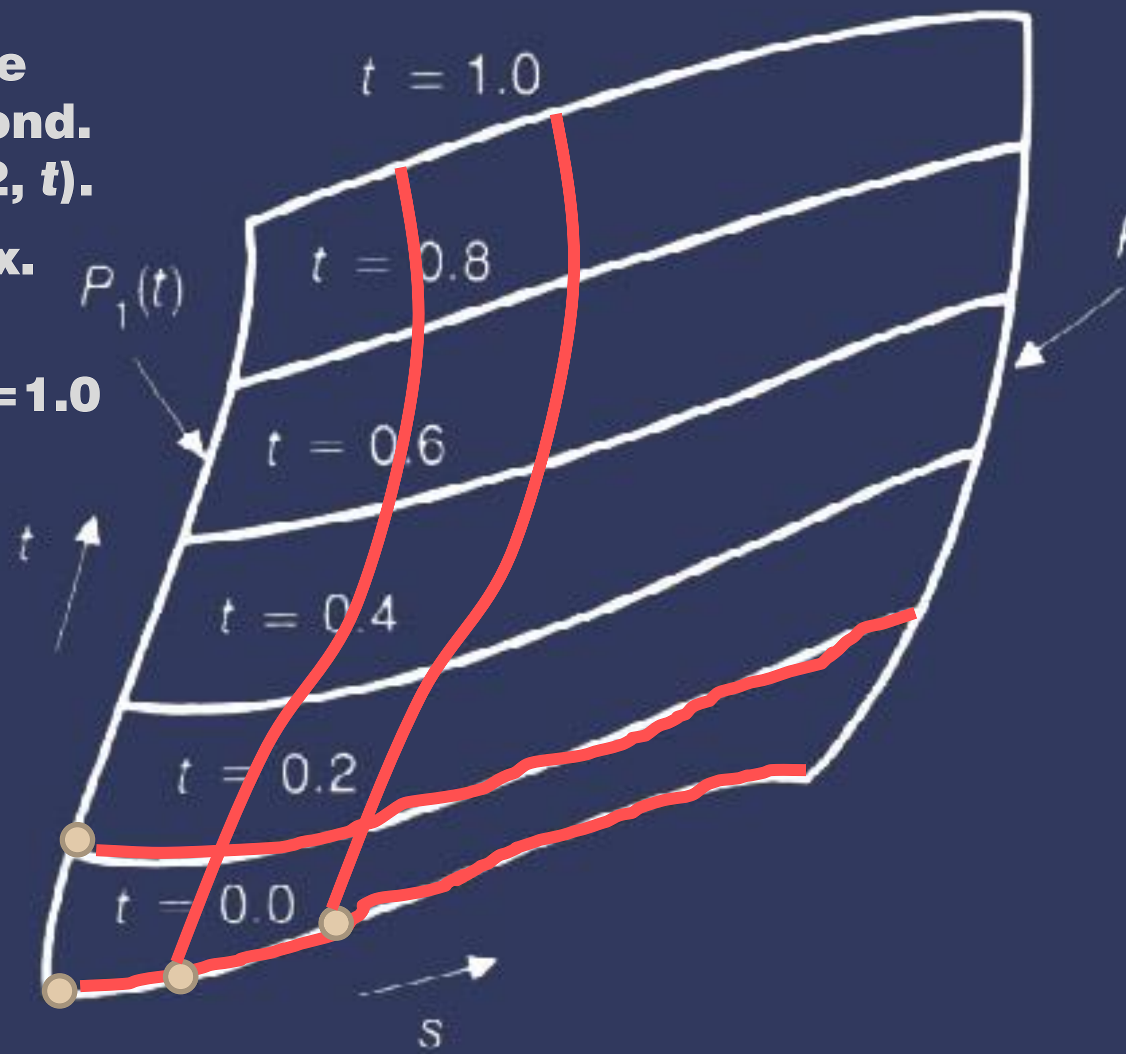
linha$_1$ <- linha$_1$ + linha$_2$

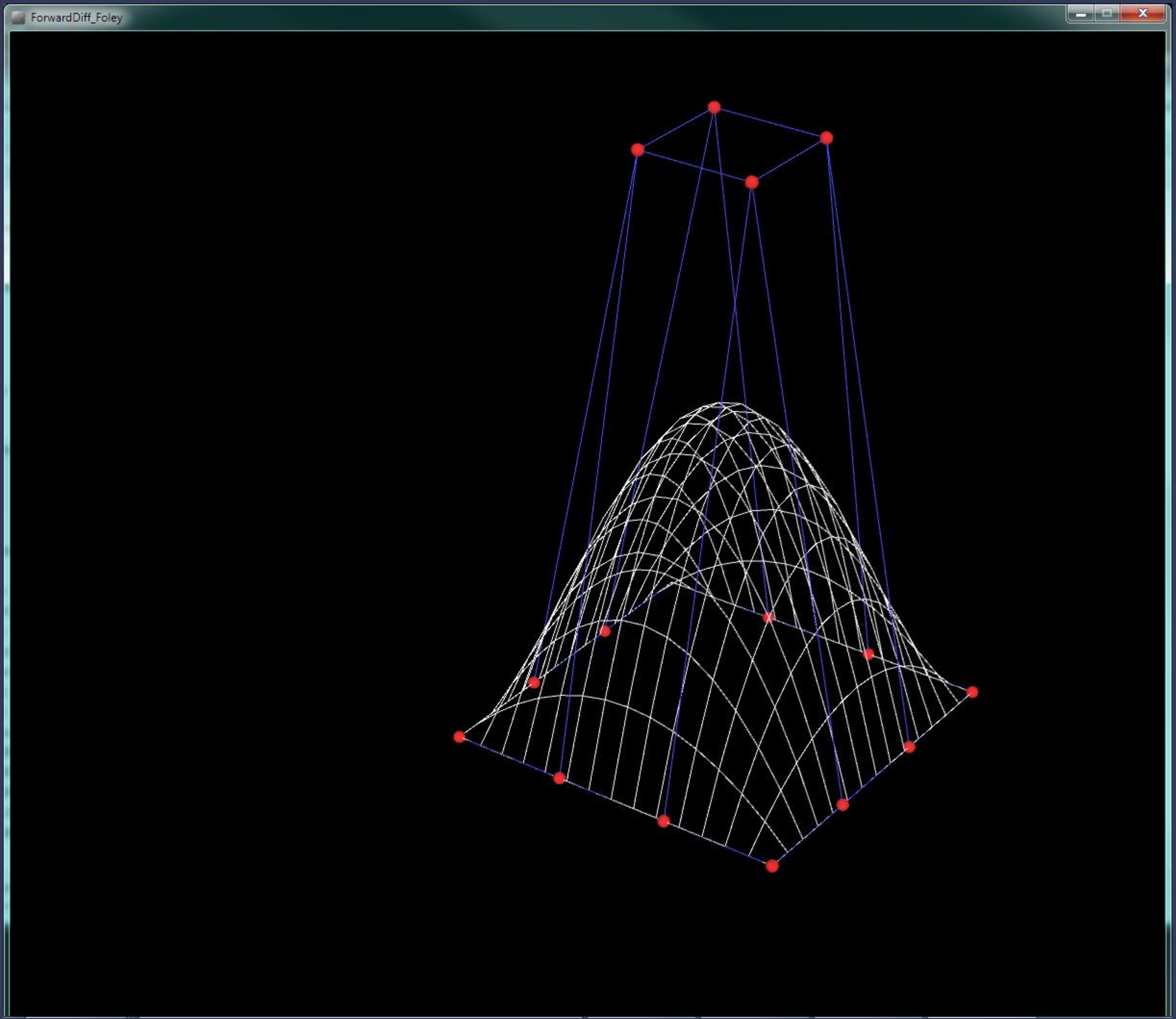
linha$_2$ <- linha$_2$ + linha$_3$

linha$_3$ <- linha$_3$ + linha$_4$

$$
\begin{bmatrix}
f(0,0) & \Delta_t f(0,0) & {\Delta_t}^2 f(0,0) & {\Delta_t}^3 f(0,0) \\
\Delta_s f(0,0) & & & \\
{\Delta_s}^2 f(0,0) & & & \\
{\Delta_s}^3 f(0,0) & & &
\end{bmatrix}
$$

1. **Cond. inic. para t em s = 0: 1ª linha DD(s,t)**
2. **Executa FwdDiff para curva em f(0,*t*)**
3. **Atualiza DD(*s*,*t*) através de soma de linhas p/gerar cond. iniciais p/próx. curva f(0.2, *t*).**
4. **Executa FwdDiff para próx. curva em f(*s*,*t*)**
5. **Retorna ao passo 3 até s=1.0**
6. **Reinicializa DD(s,t) e transpõe**
7. **Repete passos 3 e 4 até t=1**




ForwardDiff_Foley

Algoritmo Forward Differences para Superfícies

Inicializações

1. Calcule os coeficientes:

   $Cx = M_{método} \cdot G_x \cdot M^T_{método}$

   $Cy = M_{método} \cdot G_y \cdot M^T_{método}$

   $Cz = M_{método} \cdot G_z \cdot M^T_{método}$

2. Calcule os deltas para n_i passos em t e s:

   $\delta_s = 1/(n_s - 1)$

   $\delta_t = 1/(n_t - 1)$

3. Gere as matrizes $E\delta_s$ e $E\delta_t$ e transponha $E\delta_t$

$$E(\delta s) = \begin{bmatrix} 0 & 0 & 0 & 1 \\ \delta^3 & \delta^2 & \delta & 0 \\ 6\delta^3 & 2\delta^2 & 0 & 0 \\ 6\delta^3 & 0 & 0 & 0 \end{bmatrix} = E(\delta t)$$

Algoritmo Forward Differences para Superfícies

Inicializações

1. Calcule os coeficientes:

   $Cx = M_{método} \cdot G_x \cdot M^T_{método}$

   $Cy = M_{método} \cdot G_y \cdot M^T_{método}$

   $Cz = M_{método} \cdot G_z \cdot M^T_{método}$

2. Calcule os deltas para n_i passos em t e s:

   $\delta_s = 1/(n_s - 1)$

   $\delta_t = 1/(n_t - 1)$

3. Gere as matrizes $E\delta_s$ e $E\delta_t$ e transponha $E\delta_t$

$$E(\delta s) = \begin{bmatrix} 0 & 0 & 0 & 1 \\ \delta^3 & \delta^2 & \delta & 0 \\ 6\delta^3 & 2\delta^2 & 0 & 0 \\ 6\delta^3 & 0 & 0 & 0 \end{bmatrix} = E(\delta t)$$

Algoritmo Forward Differences para Superfícies

Inicializações

1. Calcule os coeficientes:

   $Cx = M_{método} \cdot G_x \cdot M^T_{método}$

   $Cy = M_{método} \cdot G_y \cdot M^T_{método}$

   $Cz = M_{método} \cdot G_z \cdot M^T_{método}$

2. Calcule os deltas para n_i passos em t e s:

   $\delta_s = 1/(n_s - 1)$

   $\delta_t = 1/(n_t - 1)$

3. Gere as matrizes $E\delta_s$ e $E\delta_t$ e transponha $E\delta_t$

$$E(\delta s) = \begin{bmatrix} 0 & 0 & 0 & 1 \\ \delta^3 & \delta^2 & \delta & 0 \\ 6\delta^3 & 2\delta^2 & 0 & 0 \\ 6\delta^3 & 0 & 0 & 0 \end{bmatrix} = E(\delta t)$$

Algoritmo Forward Differences para Superfícies

DesenhaSuperficieFwdDiff (n_s, n_t, C_x, C_y, C_z, $E\delta_s$, $E\delta_t$)

4. Calcule as condições iniciais:
   $DD_x = E\delta_s \cdot C_x \cdot E\delta_t^T$
   $DD_y = E\delta_s \cdot C_y \cdot E\delta_t^T$
   $DD_z = E\delta_s \cdot C_z \cdot E\delta_t^T$
5. Desenhe a família de curvas em t:
   Para i de 1 até n_s faça
   1. DesenhaCurvaFwdDiff (n_t, 1ª linha de DD_x, DD_y, DD_z)
   2. Atualize DD_x, DD_y, DD_z através de soma de linhas p/gerar cond.iniciais p/próx. Curva.
6. Recalcule as cond.inic. DD_x, DD_y e DD_z
7. Transponha DD_x, DD_y e DD_z
8. Desenhe a família de curvas em s:
   Para i de 1 até n_t faça
   1. DesenhaCurvaFwdDiff(n_s, 1ª linha de DD_x^T, DD_y^T, DD_z^T)
   2. Atualize DD_x^T, DD_y^T, DD_z^T através de soma de linhas

Algoritmo Forward Differences para Superfícies

DesenhaSuperficieFwdDiff (n_s, n_t, C_x, C_y, C_z, $E\delta_s$, $E\delta_t$)

4. Calcule as condições iniciais:

   $DD_x = E\delta_s \cdot C_x \cdot E\delta_t^T$

   $DD_y = E\delta_s \cdot C_y \cdot E\delta_t^T$

   $DD_z = E\delta_s \cdot C_z \cdot E\delta_t^T$

5. Desenhe a família de curvas em t:

   Para i de 1 até n_s faça

   1. DesenhaCurvaFwdDiff (n_t, 1ª linha de DD_x, DD_y, DD_z)
   2. Atualize DD_x, DD_y, DD_z através de soma de linhas p/gerar cond.iniciais p/próx. Curva.

6. Recalcule as cond.inic. DD_x, DD_y e DD_z
7. Transponha DD_x, DD_y e DD_z
8. Desenhe a família de curvas em s:

   Para i de 1 até n_t faça

   1. DesenhaCurvaFwdDiff(n_s, 1ª linha de DD_x^T, DD_y^T, DD_z^T)
   2. Atualize DD_x^T, DD_y^T, DD_z^T através de soma de linhas

Algoritmo Forward Differences para Superfícies

DesenhaSuperficieFwdDiff (n_s, n_t, C_x, C_y, C_z, $E\delta_s$, $E\delta_t$)

4. Calcule as condições iniciais:

   $DD_x = E\delta_s \cdot C_x \cdot E\delta_t^T$

   $DD_y = E\delta_s \cdot C_y \cdot E\delta_t^T$

   $DD_z = E\delta_s \cdot C_z \cdot E\delta_t^T$

5. Desenhe a família de curvas em t:

   Para i de 1 até n_s faça

   1. DesenhaCurvaFwdDiff (n_t, 1ª linha de DD_x, DD_y, DD_z)
   2. Atualize DD_x, DD_y, DD_z através de soma de linhas p/gerar cond.iniciais p/próx. Curva.

6. Recalcule as cond.inic. DD_x, DD_y e DD_z
7. Transponha DD_x, DD_y e DD_z
8. Desenhe a família de curvas em s:

   Para i de 1 até n_t faça

   1. DesenhaCurvaFwdDiff(n_s, 1ª linha de DD_x^T, DD_y^T, DD_z^T)
   2. Atualize DD_x^T, DD_y^T, DD_z^T através de soma de linhas

Algoritmo Forward Differences para Superfícies

DesenhaSuperficieFwdDiff (n_s, n_t, C_x, C_y, C_z, $E\delta_s$, $E\delta_t$)

4. Calcule as condições iniciais:
   $DD_x = E\delta_s \cdot C_x \cdot E\delta_t^T$
   $DD_y = E\delta_s \cdot C_y \cdot E\delta_t^T$
   $DD_z = E\delta_s \cdot C_z \cdot E\delta_t^T$
5. Desenhe a família de curvas em t:
   Para i de 1 até n_s faça
   1. DesenhaCurvaFwdDiff (n_t, 1ª linha de DD_x, DD_y, DD_z)
   2. Atualize DD_x, DD_y, DD_z através de soma de linhas p/gerar cond.iniciais p/próx. Curva.
6. Recalcule as cond.inic. DD_x, DD_y e DD_z
7. Transponha DD_x, DD_y e DD_z
8. Desenhe a família de curvas em s:
   Para i de 1 até n_t faça
   1. DesenhaCurvaFwdDiff(n_s, 1ª linha de DD_x^T, DD_y^T, DD_z^T)
   2. Atualize DD_x^T, DD_y^T, DD_z^T através de soma de linhas

Algoritmo Forward Differences para Superfícies

DesenhaSuperficieFwdDiff (n_s, n_t, C_x, C_y, C_z, $E\delta_s$, $E\delta_t$)

4. Calcule as condições iniciais:
   $DD_x = E\delta_s \cdot C_x \cdot E\delta_t^T$
   $DD_y = E\delta_s \cdot C_y \cdot E\delta_t^T$
   $DD_z = E\delta_s \cdot C_z \cdot E\delta_t^T$
5. Desenhe a família de curvas em t:
   Para i de 1 até n_s faça
   1. DesenhaCurvaFwdDiff (n_t, 1ª linha de DD_x, DD_y, DD_z)
   2. Atualize DD_x, DD_y, DD_z através de soma de linhas p/gerar cond.iniciais p/próx. Curva.
6. Recalcule as cond.inic. DD_x, DD_y e DD_z
7. Transponha DD_x, DD_y e DD_z
8. Desenhe a família de curvas em s:
   Para i de 1 até n_t faça
   1. DesenhaCurvaFwdDiff(n_s, 1ª linha de DD_x^T, DD_y^T, DD_z^T)
   2. Atualize DD_x^T, DD_y^T, DD_z^T através de soma de linhas

Algoritmo Forward Differences para Superfícies

DesenhaSuperficieFwdDiff (n_s, n_t, C_x, C_y, C_z, $E\delta_s$, $E\delta_t$)

4. Calcule as condições iniciais:

   $DD_x = E\delta_s \cdot C_x \cdot E\delta_t^T$

   $DD_y = E\delta_s \cdot C_y \cdot E\delta_t^T$

   $DD_z = E\delta_s \cdot C_z \cdot E\delta_t^T$

5. Desenhe a família de curvas em t:

   Para i de 1 até n_s faça

   1. DesenhaCurvaFwdDiff (n_t, 1ª linha de DD_x, DD_y, DD_z)
   2. Atualize DD_x, DD_y, DD_z através de soma de linhas p/gerar cond.iniciais p/próx. Curva.

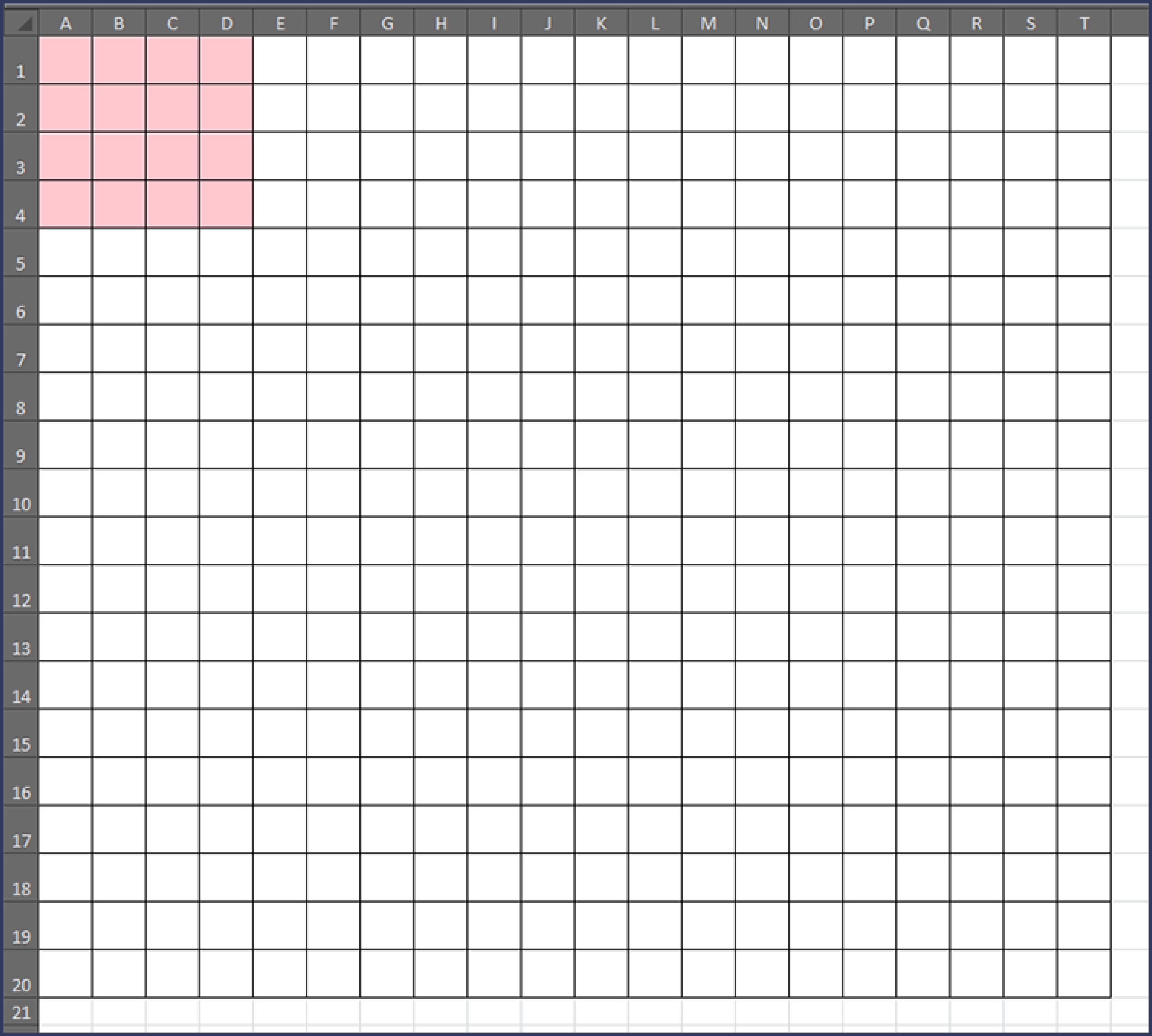
6. Recalcule as cond.inic. DD_x, DD_y e DD_z
7. Transponha DD_x, DD_y e DD_z
8. Desenhe a família de curvas em s:

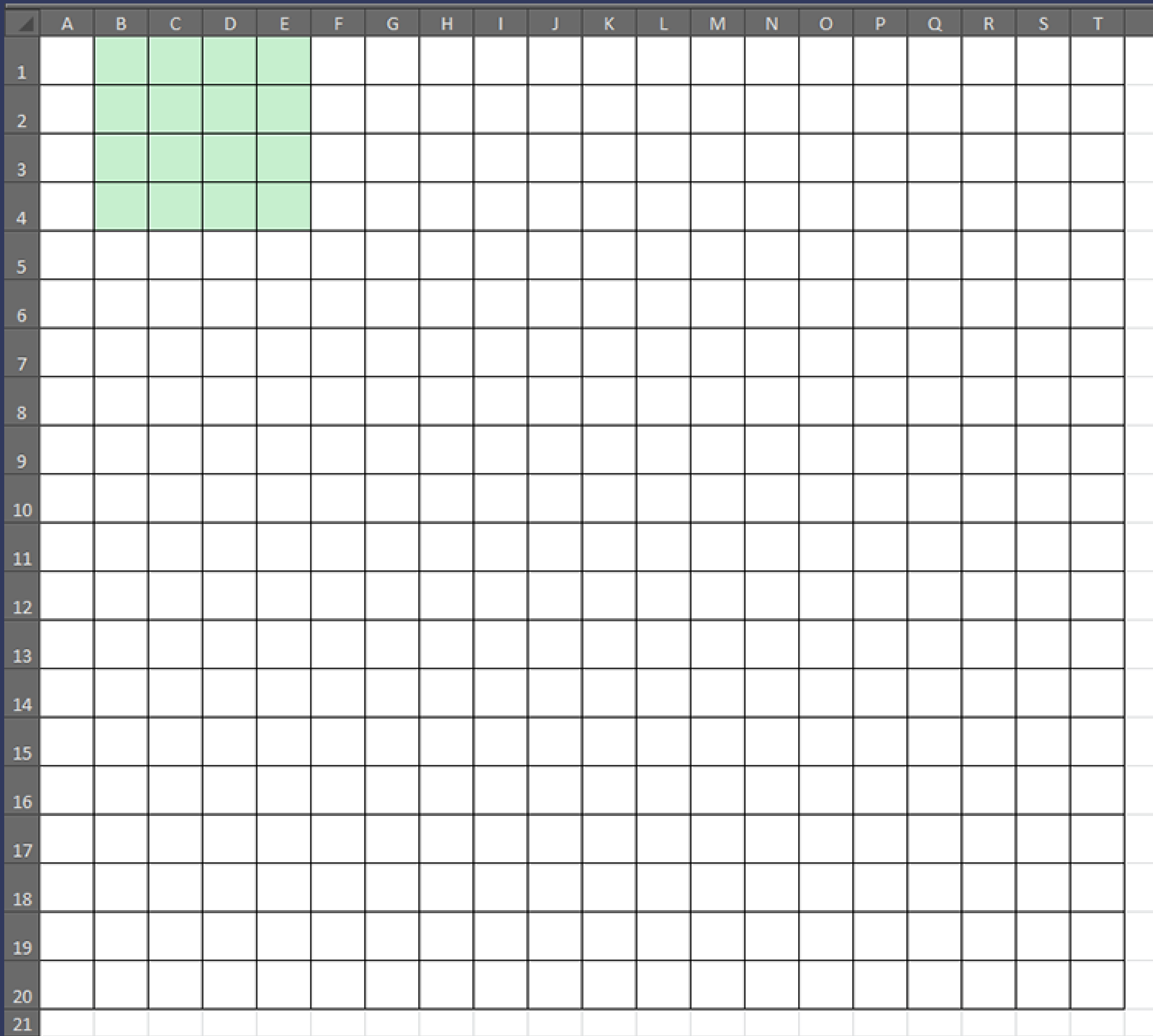
   Para i de 1 até n_t faça

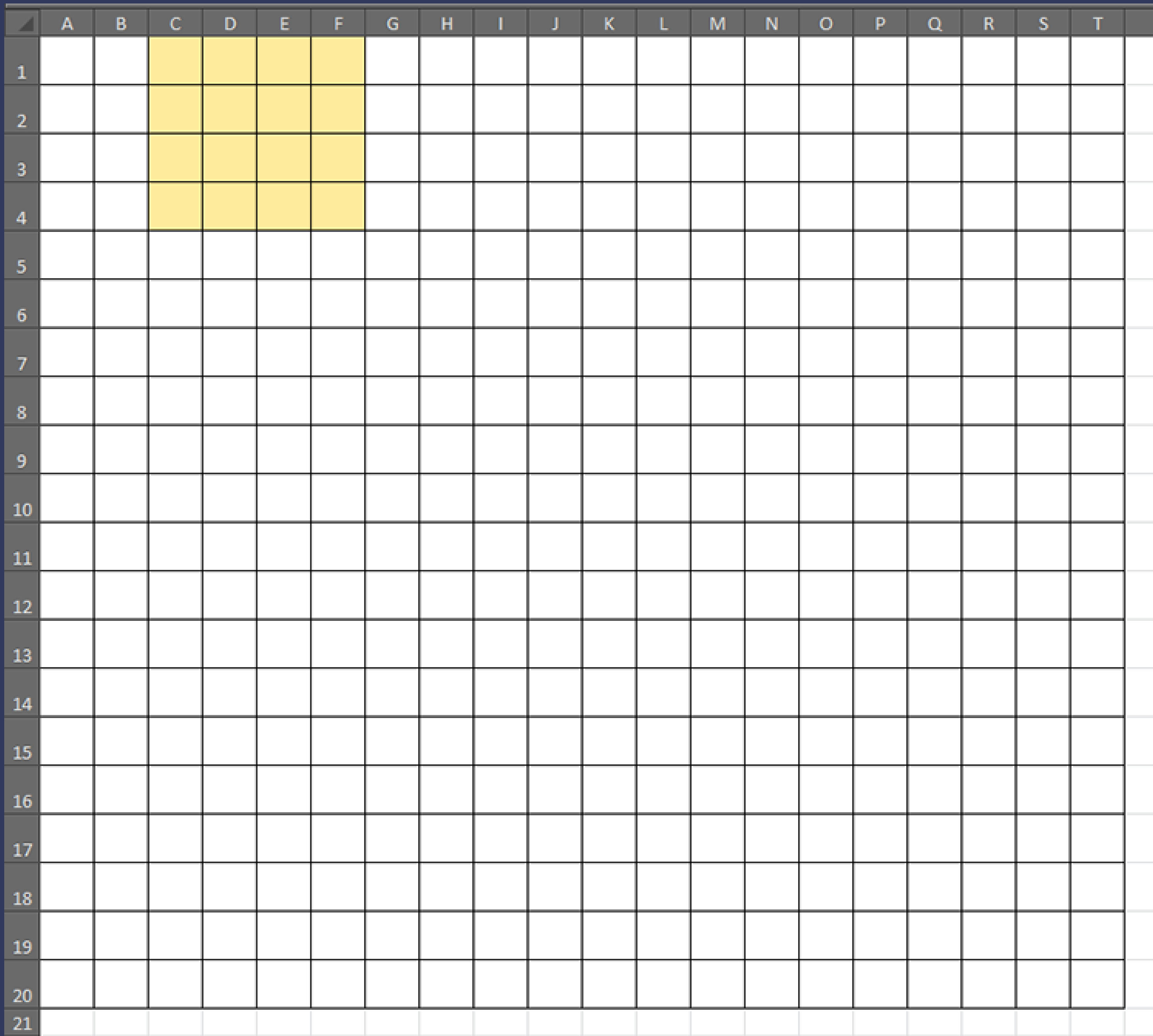
   1. DesenhaCurvaFwdDiff(n_s, 1ª linha de DD_x^T, DD_y^T, DD_z^T)
   2. Atualize DD_x^T, DD_y^T, DD_z^T através de soma de linhas

Trabalho

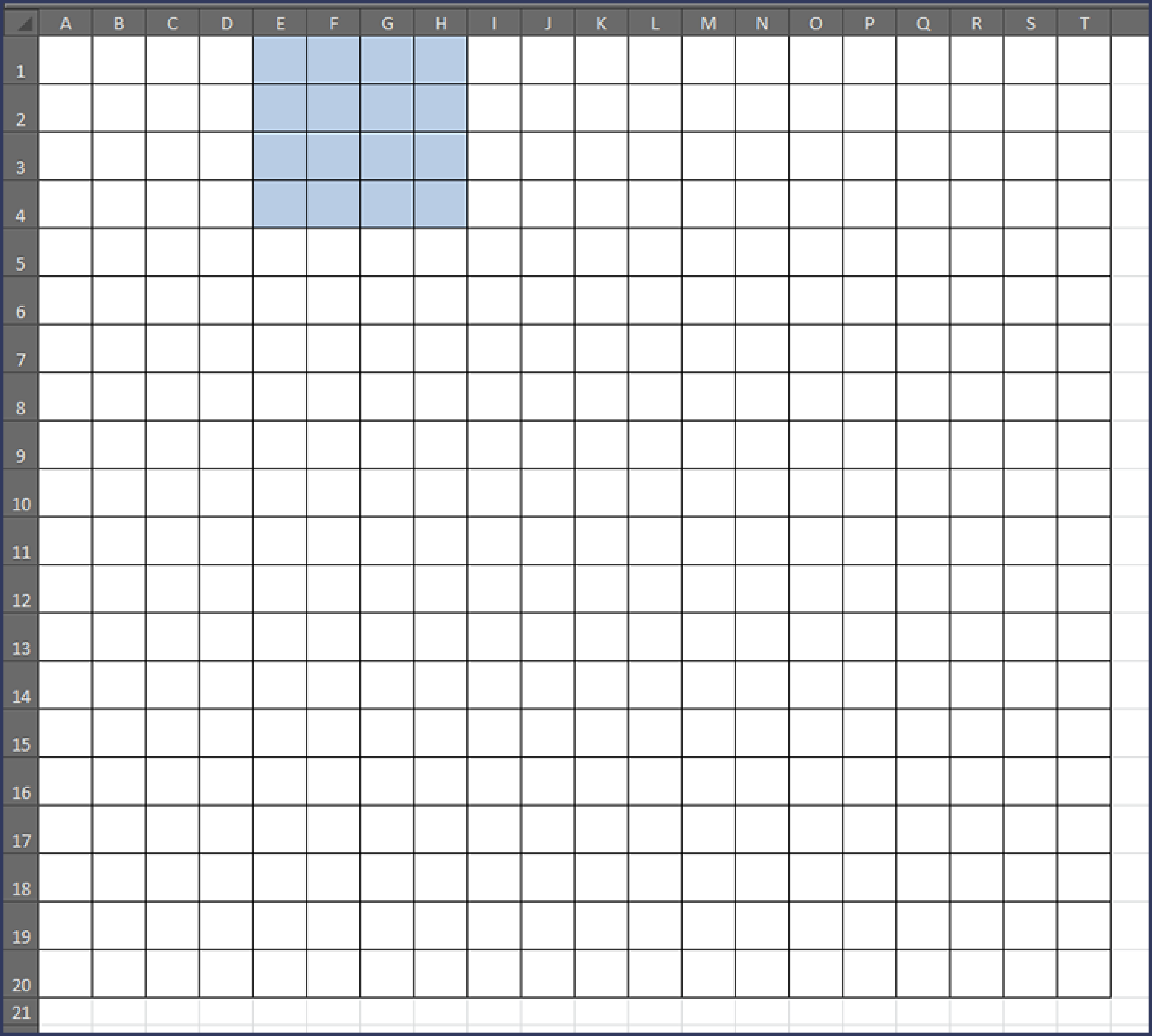
- Implemente Forward Differences para Superfícies Bicúbicas B-Spline
- Os pontos de controle deverão poder ser fornecidos por uma matriz de 4x4 até 20x20
  - Implemente uma rotina de percurso da matriz de pontos de controle para plotar cada sub-matriz 4x4

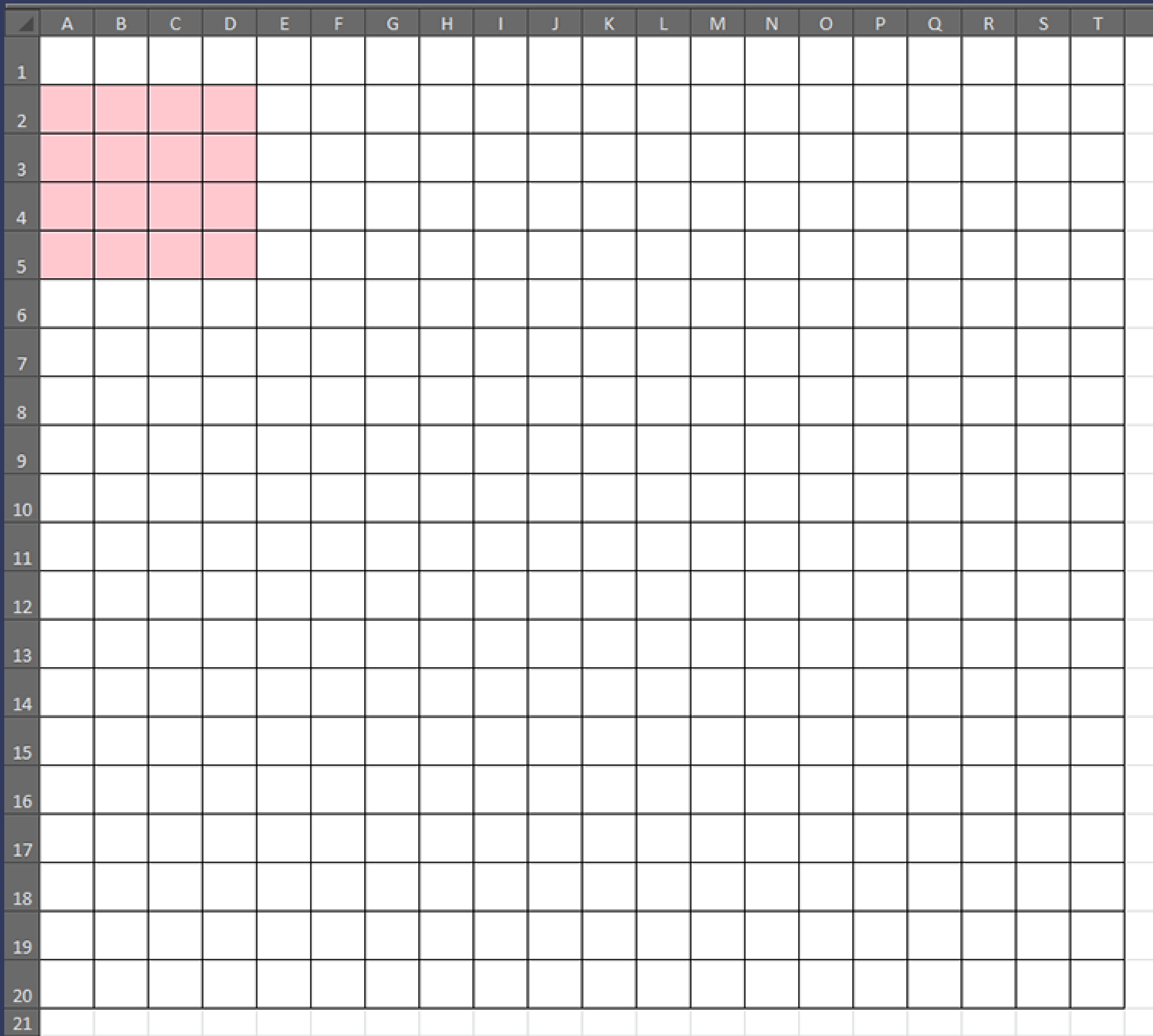

A B C D E F G H I J K L M N O P Q R S T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21


A B C D E F G H I J K L M N O P Q R S T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

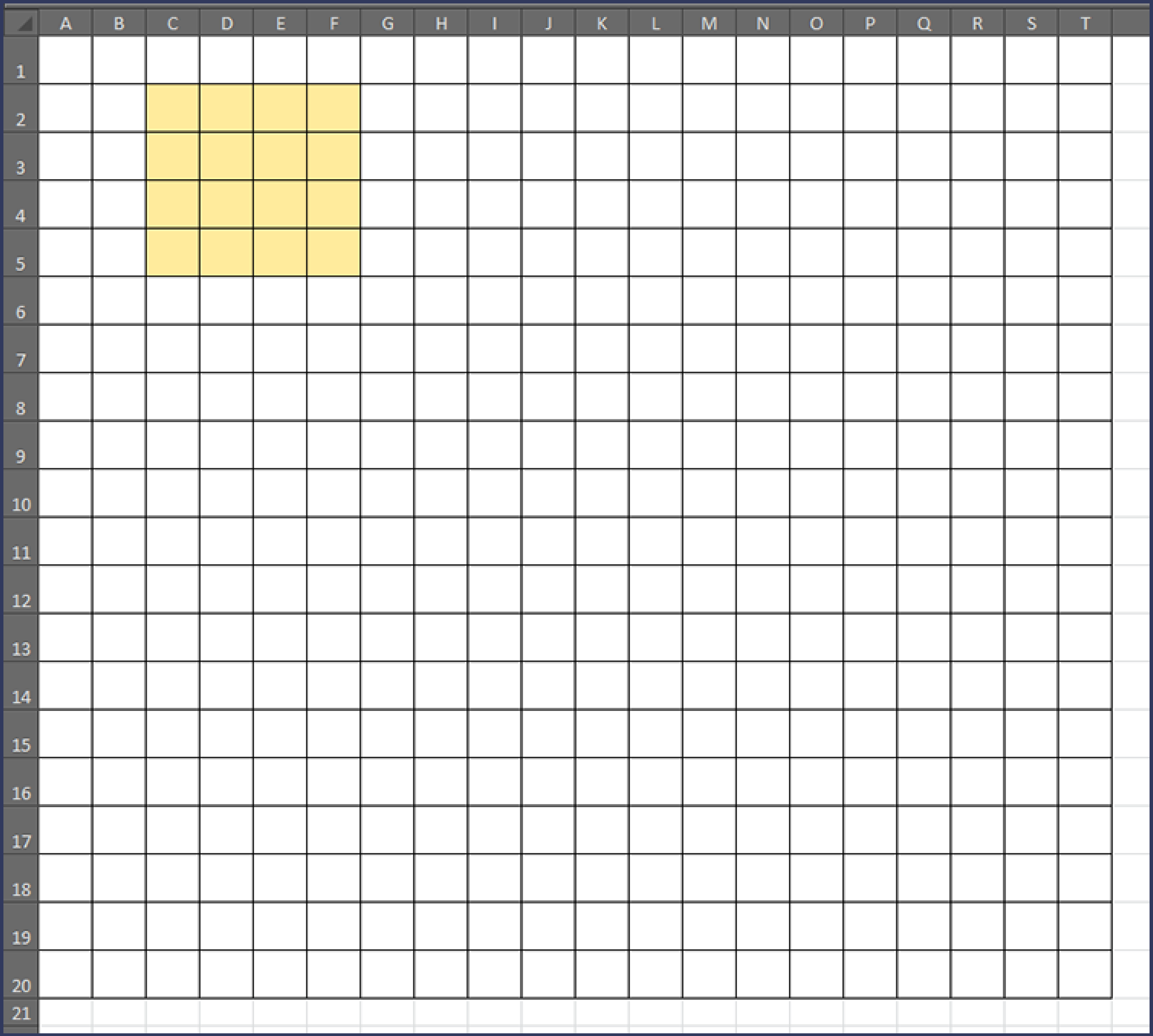

A B C D E F G H I J K L M N O P Q R S T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

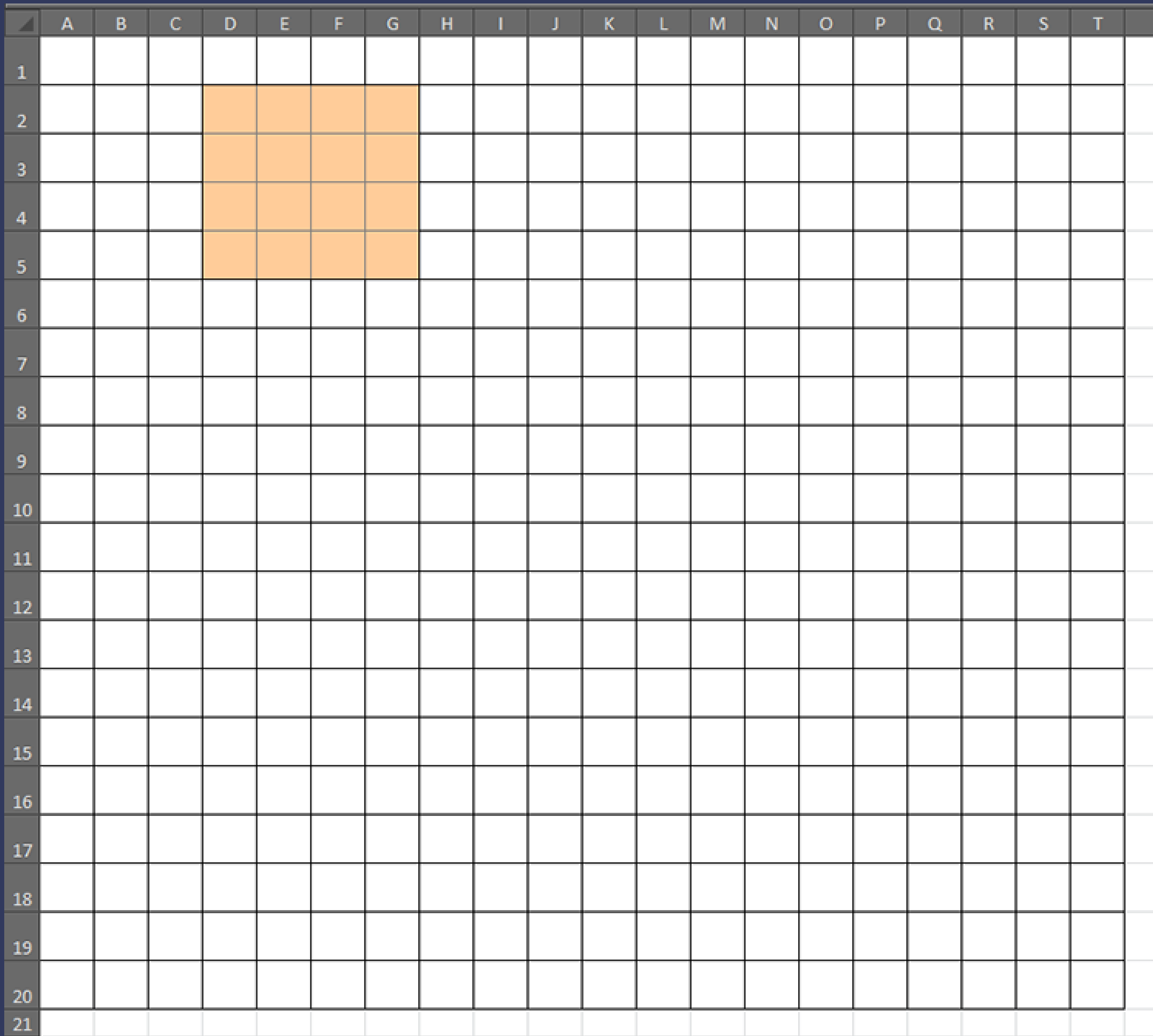
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1																				
2																				
3																				
4																				
5																				
6																				
7																				
8																				
9																				
10																				
11																				
12																				
13																				
14																				
15																				
16																				
17																				
18																				
19																				
20																				
21																				

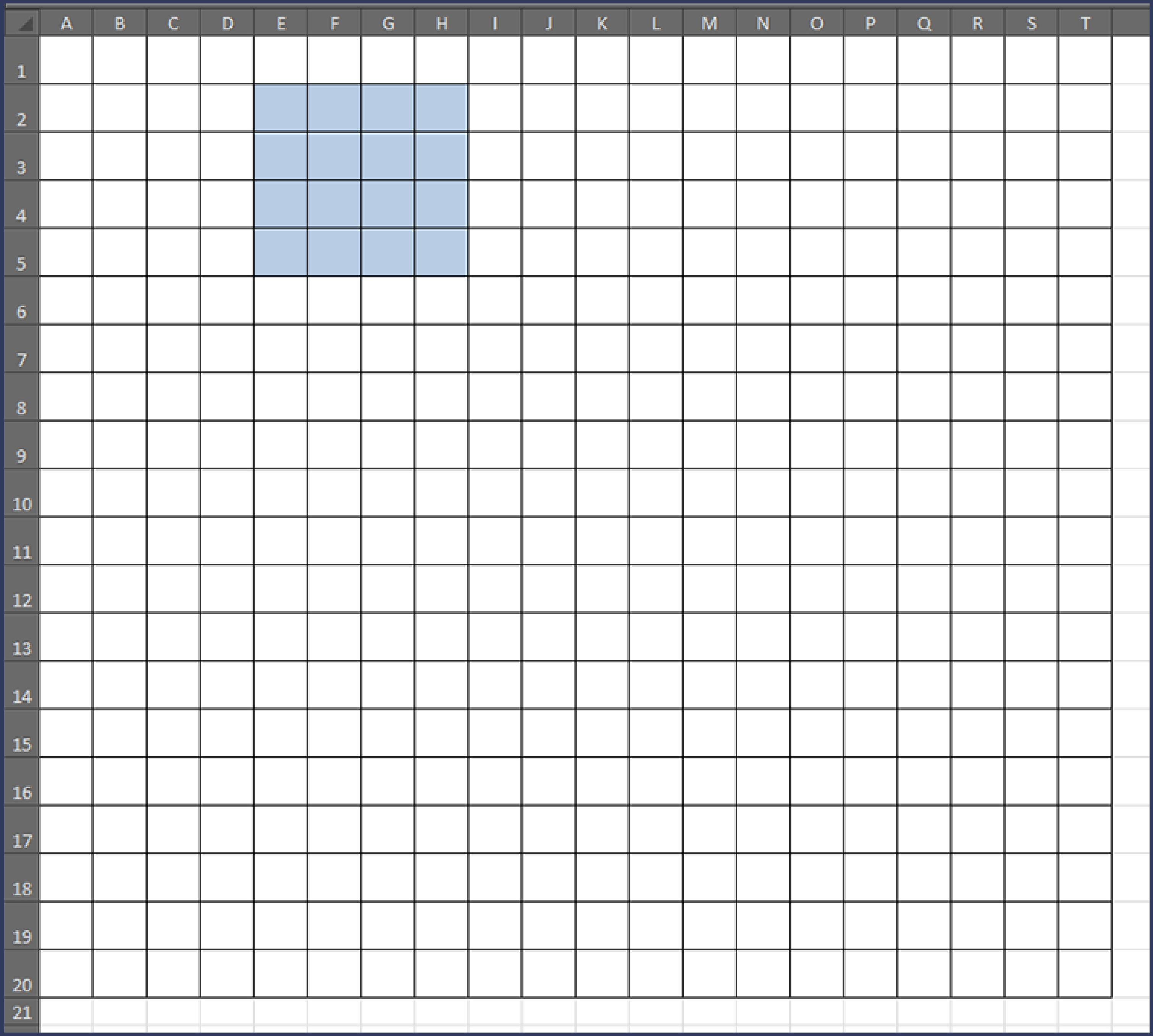

A B C D E F G H I J K L M N O P Q R S T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21


A B C D E F G H I J K L M N O P Q R S T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1																				
2																				
3																				
4																				
5																				
6																				
7																				
8																				
9																				
10																				
11																				
12																				
13																				
14																				
15																				
16																				
17																				
18																				
19																				
20																				
21																				


A B C D E F G H I J K L M N O P Q R S T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21


A B C D E F G H I J K L M N O P Q R S T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21


A B C D E F G H I J K L M N O P Q R S T
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

ARS ET SCIENTIA

UNIVERSIDADE FEDERAL DE SANTA CATARINA

UFSC