



Kontserbazio legeak dituzten deribatu partzialetako ekuazioetarako zenbakizko metodoak

Gratu Amaierako Lana
Matematikako Gratu

Jon Gabirondo Aranguren

Carlos Gorria Corres
Irakasleak zuzendutako lana

Leioa, 2019ko ekainaren 20a

Aurkibidea

Sarrera	v
1 Kontserbazio legeak eta Burgersen ekuazioa	1
1.1 Kontserbazio legeak	1
1 Kontserbazio legeen adibideak	2
1.2 Burgersen ekuazio biskositaterik gabe	3
1 Talka unea	5
2 Soluzio ahulak	6
3 Riemann-en problema	7
4 Entropia baldintzak	13
1.3 Burgersen ekuazioa biskositatearekin	14
2 Zenbakizko metodoak	17
2.1 Zenbakizko metodoen sarrera	17
2.2 Metodo kontserbakorrak	20
1 Diferentzia finituko metodo kontserbakor batzuk	24
2.3 Godunov-en metodoa	26
1 Denbora-tartearen aukeraketa zuzena.	30
3 Zenbakizko simulazioak	33
3.1 Diferentzia finituko metodoak	34
3.2 Godunov-en metodoa	38
4 Ondorioak eta hobetzeko aukerak	43
A Eranskina	45
Bibliografia	61

Sarrera

Lan honen helburua kontserbazio legeak dituzten deribatu partzialetako ekuazioak aztertu eta zenbakizko metodoak erabiliz ebaztea da. Graduan zehar ekuazio ez-linealak ez dira ia lantzen eta hortik dator motibazioa. Mota horretako ekuazioek zailtasun gehigarriak dituzte eta horiei aurre egiten ikasiko dugu. Lanaren hasieran kontserbazio legeen sarrera bat egingo da, segidan, Burgersen ekuazioa aurkeztuko da, lan guztian zehar kontserbazio legeen eredu izango dena, eta analitikoki ekuazioaren zenbait propietate garrantzitsu aztertuko dira. Ondoren, zenbakizko metodoen kontzeptu orokorren definizioak egingo dira eta zenbait zenbakizko eskema aurkeztuko dira Godunov-en metodoarekin bukatzeko. Lanaren amaieran, metodo horien zenbakizko simulazioak egingo dira bi problema zehatz ebazteko. Soluzioak eta grafikoak lortzeko *Mathematica* programa erabiliko da, grafikoak egin eta aztertzeko egokiena delako.

Lan hau egiteko graduan zehar jasotako jakintzak praktikan jarriko dira: lehenik eta behin, zenbakizko metodoei buruzko hiru ikasgaietan ikasitako kontzeptuak (Zenbakizko metodoak I, Zenbakizko metodoak II eta Numerical solution of differential equation); bestetik, zenbait integral ulertzeko eta moldatzeko kalkulu diferentziala eta baita ekuazio diferentzialetan ikasitako teknikak ere; eta azkenik, zuzenean eragiten ez duten baina baliagarriak egingo diren hainbat jakintza, hala nola, neurria, programazioa, fisika orokorra edo kurbak eta geometria.

1. kapitulua

Kontserbazio legeak eta Burgersen ekuazioa

Lehenengo kapituluan, hasieran, kontserbazio legeen sarrera bat egingo dugu: definitu egingo ditugu, zein propietate betetzen dituzten behatu eta adibide ezagunenak aurkeztu. Bigarren zatian, Burgersen ekuazioan zentratuko gara: karakteristikoaren metodoaren bidez ebatziko dugu, soluzio ahulak definituko ditugu, problema zehatzak ebatzi, eta azkenik, ekuazioa biskositatearekin ebatziko dugu.

1.1 Kontserbazio legeak

Fisika arloan, sistema isolatu batean zenbait propietate konstante mantentzen dira denbora zehar. Kontserbazio legeek frogatzen dute baldin eta sistema batek ez badu bere ingurunearekiko elkarrekin, orduan sistemaren zenbait aldagai mekaniko ezin direla aldatu: hala nola, energia, masa edota momentu lineala.

Definizioa 1.1.1. *Kontserbazio legea* denboran zehar kontserbatzen den aldagai bat edo gehiagoren eboluzioa deskribatzen duen deribatu partzialetakoa ekuazioa da.

Lege horien hasierako balioetako problemak ondoko forma dauka:

$$\begin{cases} \frac{\partial}{\partial t}u(x, t) + \frac{\partial}{\partial x}f(u(x, t)) = 0 \\ u(x, 0) = u_0(x) \end{cases} \quad (1.1)$$

non $u : \mathbf{R} \times \mathbf{R} \rightarrow \mathbf{R}^n$ egoera aldagaia deritzon, ezezaguna izango dena, eta $f : \mathbf{R}^n \rightarrow \mathbf{R}^n$ fluxu funtzioa. Lan honetan $n = 1$ kasua bakarrik aztertuko dugu. Egoera aldagaia kontserbatzen dela esaten da $u(x, t)$ funtzioaren balio totalak t -rekiko konstantea izan behar duelako, hau da,

$$\frac{d}{dt} \int_{-\infty}^{\infty} u(x, t) dx = 0 \quad (1.2)$$

Zenbaitetan, kontserbazio legeen forma integrala erabiltzen da. Berau lortzeko, ekuazioa x -rekiko integratu besterik ez dugu egin behar:

$$\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} f(u(x, t)) = 0$$

kontserbazio legea x -rekiko integratuko dugu x_1 -etik $x_2 > x_1$ -ra:

$$\frac{d}{dt} \int_{x_1}^{x_2} u(x, t) dx = f(u(x_1, t)) - f(u(x_2, t)). \quad (1.3)$$

Ekuazio honi kontserbazio legearen *forma integrala* deritzo.

1 Kontserbazio legeen adibideak

(i) Korteweg-de Vries (KdV) ekuazioa:

$$u_t + \alpha u u_x + \beta u_{xxx} = 0, \quad (1.4)$$

non α eta β konstanteak diren. Ekuazio hau kontserbazio legeen forman idatzi daiteke,

$$u_t + \left(\frac{\alpha}{2} u^2 + \beta u_{xx} \right)_x = 0.$$

(ii) Sine-Gordon ekuazioa:

$$\varphi_{tt} - \varphi_{xx} + \sin \varphi = 0. \quad (1.5)$$

Ekuazio horren bidez kontserbatzen den aldagai garrantzitsu bat ondorengo funtzioa da,

$$I(u, v, \rho) = \int_{-\infty}^{\infty} \left(\rho^2 \cos \frac{u+v}{2} + \frac{1}{4} \cos \frac{u-v}{2} \right) dx = K,$$

non u eta v (1.5) ekuazioaren bi soluzio diren Bäcklund* transformatuaren bidez erlazionatuta daudenak (ρ parametroaren bidez), [4] liburuan azaltzen den moduan.

(iii) Schrödinger-en ekuazio ez-lineala:

$$\begin{aligned} i\psi_t &= -\frac{1}{2} \Delta \psi + V(\mathbf{x})\psi + \beta |\psi|^{2\sigma} \psi, \quad \mathbf{x} \in \mathbf{R}^n, \quad t \geq 0, \\ \psi(\mathbf{x}, 0) &= \psi_0(\mathbf{x}), \quad \mathbf{x} \in \mathbf{R}^n, \end{aligned} \quad (1.7)$$

* u eta v Bäcklund transformatuaren bidez erlazionatuta daudela esaten da baldin eta

$$\begin{aligned} (\partial_t - \partial_x)(u-v) &= -4\rho \sin \frac{u+v}{2}, \quad \text{eta} \\ (\partial_t + \partial_x)(u+v) &= \frac{1}{\rho} \sin \frac{u-v}{2}. \end{aligned} \quad (1.6)$$

non σ konstante positibo bat den, $V(x)$ balio errealeko funtzio bat eta β beste konstante erreal bat. Kasu honetan, [1] liburuan agertzen den moduan, kontserbatzen den aldagai garrantzitsuenetako bat uhin funtzioaren normalizazioa da,

$$N(\psi(\cdot, t)) = \int_{\mathbf{R}^n} |\psi(\mathbf{x}, t)|^2 d\mathbf{x} \equiv N = N(\psi_0), \quad t \geq 0.$$

Kontserbatzen den beste aldagai bat energia da,

$$\begin{aligned} E(\psi(\cdot, t)) &= \int_{\mathbf{R}^n} \left(\frac{1}{2} |\nabla \psi(\mathbf{x}, t)|^2 + V(\mathbf{x}) |\psi(\mathbf{x}, t)|^2 + \frac{\beta}{\sigma + 1} |\psi(\mathbf{x}, t)|^{2\sigma+2} \right) d\mathbf{x} \\ &= E(\psi_0), \quad t \geq 0. \end{aligned}$$

1.2 Burgersen ekuazio biskositaterik gabe

Beste kontserbazio lege ezagunenetako bat Burgersen ekuazioa da. Lan honetan Burgersen ekuazioa erabiliko dugu kontserbazio legeetara zabaldu daitezkeen propietateak aztertzeko.

[2] artikuluan idatzita daude Burgersen ekuazioaren hastapenak. Harry Bateman[†] matematikari ingelesak idatzi zuen lehenengo aldiz deribatu partzialetako ekuazioa hasierako eta mugalde-baldintzekin 1915ean,

$$\begin{cases} \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}, & 0 < x < L, \quad t > 0, \\ u(x, 0) = u_0(x), & 0 < x < L, \\ u(0, t) = \psi_1(t), \quad u(L, t) = \phi_2(t), & t > 0, \end{cases} \quad (1.8)$$

non u, x, t eta ν abiadura, espazio- eta denbora-uneak eta biskositate zinatikoa diren hurrenez hurren. Urte batzuk beranduago, 1948an, Jan Burgers[‡] fisikari holandarrak ekuazioa erabili zuen fluidoaren mekanika arloan modelo matematiko bat eraikitzeko. Gizon honek egindako ekarpenagatik, ekuazioari Burgersen ekuazioa deritzo. Hopf-ek[§] eta Cole-ek[¶] aldagai-aldaketa baten bidez Burgersen ekuazioa beroaren ekuaziora eraman eta ebatzi zuten, bakoitzak bere aldetik. (1.9) aldagai-aldaketari *Hopf-Cole-en transformazioa* deritzo:

$$u(x, t) = -2\nu \frac{\varphi_x}{\varphi}. \quad (1.9)$$

Ohartu Burgersen ekuazioak eskuineko gairik gabe (1.1) kontserbazio legeen forma duela, $f(u) = \frac{1}{2}u^2$ hartuta:

$$u_t + \left(\frac{1}{2}u^2 \right)_x = 0 \quad (1.10)$$

[†] Harry Bateman, Manchester, 1882-1946.

[‡] Johannes Martinus Burgers, Arnhem, 1895-1981.

[§] Eberhard Hopf, Salzburgo, 1902-1983

[¶] Julian David Cole, Brooklyn, 1925-1999

Hortaz, lan honetan zehar askotan, ekuazioa soilik aztertu beharren $f(u)$ funtzio orokorraren arabera aztertuko dugu eta ondoren kasu partikularra aztertu.

Burgersen ekuazioa biskositaterik gabe aztertzen ari garela esaten da (1.8) ekuazioaren eskuineko gaia zero denean, $\nu = 0$ denean alegia. Problema osatzeko funtzioak hasierako unean duen balioa ezaguna izango da:

$$\begin{cases} u_t + uu_x = 0, \\ u(x, 0) = u_0(x). \end{cases} \quad (1.11)$$

Lehenik, soluzioa topatzeko karakteristikoaren metodoa erabiliko dugu. Ohartu (1.11) ekuazioak $u_t + cu_x = 0$, $c \in \mathbf{R}$ adbezio ekuazioaren itxura duela: c konstatearen ordeztu u funtzioa jarrita lortzen da, beraz, uhinaren abiadura puntu horretan duen balioaren arabera izango da. Karakteristikoan zehar x aldagaia t -ren menpe jar daitekeenez, metodoa aplikatuz

$$x'(t) = u(x(t), t) \quad (1.12)$$

lortzen da, eta beraz, t -rekiko integratuz,

$$x(t) = u(x(t), t) t + K, \quad (1.13)$$

non K konstante bat den. Horrez gain, $t = 0$ unean ekuazioa bete behar denez,

$$x(0) = u(x(0), 0) 0 + K = K \Rightarrow K = x(0) = x_0.$$

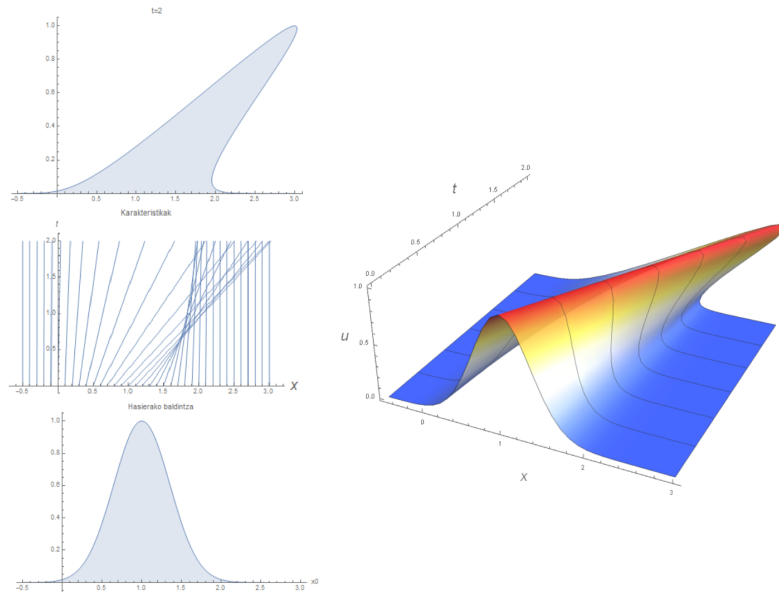
Bestetik, u funtzioa t -rekiko konstatea izango da karakteristiko horretan zehar:

$$\begin{aligned} \frac{d}{dt}u(x(t), t) &= \frac{\partial}{\partial x}u(x(t), t) x'(t) + \frac{\partial}{\partial t}u(x(t), t) \\ &= uu_x + u_t = 0. \end{aligned}$$

Orduan, kontuan hartuta u -ren balioa ez dela aldatzen karakteristikoan zehar, (1.13) zehaztuta geratzen da, $u(x(t), t) = u(x_0, 0)$ izango baita:

$$x(t) = u_0(x_0) t + x_0. \quad (1.14)$$

Ondorioz, karakteristikoak lerro zuzenak dira, hasierako baldintzaren menpe daudenak. Hasieran uhinaren balioa handia den puntuetan uhina azkar hedatuko da eta balio txikiagoa duen puntuetan, berriz, mantsoago. Hasierako baldintza punturen batean beherakorra bada, abiadura-diferentzia horregatik karakteristika elkartu egingo dira eta ekuazioak ez du soluzio bakarra izango (1.1. irudia). Are gehiago, karakteristika lehenengo aldiz mozten diren puntuan soluzioa eten egingo da eta uhina "lehertu" egingo dela esango dugu, talka sortuaz. Puntu horretan DPEak ez du soluziorik eta soluzio ahula eten egingo da.



1.1. irudia: Ezkerraldean, $t = 0$ eta $t = 2$ unean soluzioaren forma eta tarte horretako karakteristikak $[-0.5, 3.5]$ espazio-tartean: ohartu hasierako funtzioa beherakorra den tartetik ateratako karakteristikak moztu egiten direla. Eskuinaldean karakteristikoaren metodoaren bidezko soluzioa irudikatuta $[-0.5, 3.5] \times [0, 2]$ eremuan.

1 Talka unea

Maiz, mozketa hori zein denboran gertatuko den jakitea interesgarria izan daiteke. Hori kalkulatzeko, bi karakteristikoak berdindu besterik ez dugu egin behar, [6] liburuan azaltzen den bezala. Suposatuz $x_0 < x_1$ dela:

$$u_0(x_0)t + x_0 = u_0(x_1)t + x_1$$

berdintzatik t parametroa bakartuko dugu:

$$t = \frac{x_1 - x_0}{u_0(x_0) - u_0(x_1)} = \frac{\Delta x}{u_0(x_0) - u_0(x_0 + \Delta x)}.$$

Adierazpen hau $\Delta x \rightarrow 0$ eginda nahi genuen balioa lortuko dugu:

$$\lim_{\Delta x \rightarrow 0} t = -\frac{1}{u_0'(x_0)}.$$

t balio multzo hauetatik txikiena hartuko dugu, hau izango baita uhinak talka jasango duen unea, hortaz:

$$T_l = \min_x -\frac{1}{u_0'(x)}. \quad (1.15)$$

$t > T_l$ balioetarako soluzioak hiru balio hartuko ditu hainbat puntutan. Fisika alorrean zenbait kasutan zentzua izan dezake, olatu bat lehertzera doanean esaterako, baina beste zenbaitetan ezinezkoa da; adibidez, gas baten dentsitateak ezin ditu hiru balio izan.

Ezinezkoa da kasuetan, biskositatearen gaia (u_{xx}) sartuko dugu ekuazioan eta elementu horrek soluzioari ez dio utziko talka egoerara iristen: kapitulu honen bukaeran ikusiko dugun bezala, u_x handitzean u_{xx} are gehiago handituko da, eta beraz, karakteristikoak ez dira inoiz moztuko.

2 Soluzio ahulak

Ekuazio diferentzial bat ebazterako orduan espero ez dugun soluzioarekin topa gaitzke, balio bat baino gehiago hartzen dituelako edota oso zorrotza delako. Maiz, nahiz eta hasierako u_0 funtzioa C^1 klasekoa^{||} izan, soluzioa oso zorrotza bilakatzen da eta ez da deribagarria. Horrek ezin du gertatu, soluzioak ekuazio diferentziala bete behar duenez deribagarria izan behar delako. Arazo horretatik sortzen dira soluzio ahulak.

Ekuazioak eskatzen dituen deribagarritasun baldintzak betetzen dituen soluziorik ez dago. Ondorioz, gutxienez ekuazioaren soluzio hipotetikoaren ezaugarriak, hein handi batean, betetzen dituzten funtzioak bilatzen saiatuko gara. Funtzio mota horiei ekuazioaren *soluzio ahulak* deritzegu. Horretarako irizpide logiko batzuk finkatu behar dira, [3] erreferentzian azaltzen den moduan:

- C^∞ klaseko funtzio bat soluzio ahula da soluzio erregularra** bada.
- Funtzio ez-jarraitu bat soluzio ahula izan daiteke.
- Funtzio ez-jarraitu bat, ekuazioaren forma integrala betetzen badu bakarrik izan daiteke soluzio ahula.

Nahiz eta funtzio bat ez-jarraitua izan puntu batean, integratzen badugu funtzio jarraitu bat lortuko dugu. Arrazoi horregatik askotan ekuazioaren forma integrala erabiliko dugu. Soluzio ahula eratzeko, deribatu partzialak ekuazioa C^∞ klaseko *test funtzio* batez biderkatuko dugu, euskarri trinkoa duena, eta gero nahi beste aldiz integratu. Zatikako integrazioa eginez bilatzen ari garen u funtzioaren deribatuak desagertzen saiatuko gara, edo murrizten gutxienez.

Funtzio batek euskarri trinkoa edukitzeak esan nahi du funtzioaren balioa ez-nulua dela $(x, t) \in \mathbf{R} \times [0, +\infty)$ espazioko eremu trinko batean bakarrik.

^{||}Funtzio bat C^1 klasekoa dela esaten da bere deribatu partzial guztiak jarraituak badira. C^m klasekoa dela esango dugu bere n -garren deribatu partzialak jarraituak badira.

**Soluzio bat *erregularra* dela esaten da baldin eta soluzioak ekuazioak eskatzen dituen deribagarritasun mailak betetzen baditu.

Izan bedi ϕ euskarri trinkoa duen C^∞ klaseko funtzio bat. Orduan (1.1) ekuazioa ϕ -z biderkatuz eta zatika integratuz,

$$\begin{aligned} 0 &= \int_0^\infty \int_{-\infty}^\infty (u_t + (f(u))_x) \phi \, dx dt = \\ &= \left(\int_{-\infty}^\infty \phi u \, dx \right) \Big|_0^\infty - \int_0^\infty \int_{-\infty}^\infty \phi_t u + \phi_x f(u) \, dx dt \end{aligned} \quad (1.16)$$

lortzen dugu.

Badakigu (1.1) ekuazioa betetzen duten soluzio guztiek (1.3) ekuazioaren forma integrala betetzen dutela eta alderantzizkoa ez dela zertan bete. Hala ere, ustezko soluzioari edozein $\phi \in C^\infty$ funtzio biderkatuta ekuazio integrala betetze bada, jatorrizko (1.1) ekuazioaren soluzioak bete behar dituen ezaugarri gehienak betetzen dituela ondorioztatzen da.

Definizioa 1.2.1. (Soluzio ahula)

Izan bedi $u_t + (f(u))_x = 0$ kontserbazio legea. $u(x, t)$ kontserbazio legearen *soluzio ahula* izango da baldin eta edozein $\phi \in C^\infty$ funtzio euskarri trinkoduna hartuta,

$$\int_0^\infty \int_{-\infty}^\infty \phi_t u + \phi_x f(u) \, dx dt = - \int_{-\infty}^\infty \phi(x, 0) u(x, 0) \, dx \quad (1.17)$$

betetzen bada. ϕ funtzioari *test funtzioa* deritzo.

3 Riemann-en problema

Burgersen problema aztertzean interesgarria da talka osteko egoera definitzea. Beraz, Riemann-en problema ebatzi eta etengunean zer gertatzen den jakitea baliagarria gerta daiteke. Are gehiago, Godunov-en metodoa formulatzean funtsezko erabilera emango diogu Riemann-en problema etengabe erabiliko baitugu. Riemann-en problema, (1.11) Burgersen problema tarteka konstantea den hasierako baldintza batekin definitzen da:

$$u(x, 0) = \begin{cases} u_1 & x < x_0, \\ u_2 & x > x_0. \end{cases} \quad (1.18)$$

Riemann-en problema Burgersen ekuazioarentzat biskositaterik gabeko kasuan honela geratzen da:

$$\begin{cases} u_t + uu_x = 0, & x \in \mathbf{R}, t > 0, \\ u(x, 0) = u_0(x) = \begin{cases} u_1 & x < x_0 \\ u_2 & x > x_0 \end{cases} & x \in \mathbf{R}. \end{cases} \quad (1.19)$$

Problema sinplifikatzeko, $x_0 = 0$ hartuko dugu; traslazio batekin erraz lortuko genuke kasu orokorra. [6] liburuan azaltzen den moduan, bi egoera topa ditzakegu:

i) $u_1 > u_2$:

Kasu honetan problemak soluzio ahul bakarra du, *travelling wave* deritzona:

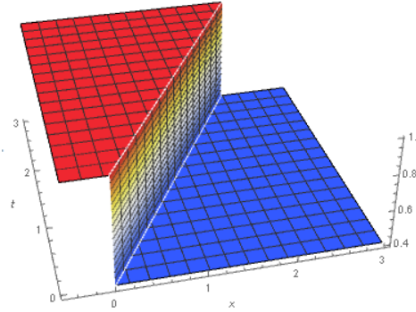
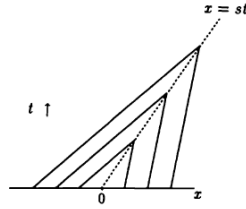
$$u(x, t) = u_0(x - x_0 - st),$$

edota, $x_0 = 0$ hartuta,

$$u(x, t) = \begin{cases} u_1 & x < st, \\ u_2 & x > st. \end{cases} \quad (1.20)$$

non $s = \frac{u_1 + u_2}{2}$ den (1.2. irudia).

Etengunearen inguruko karakteristikak



1.2. irudia: Ezkerraldean travelling wave-aren karakteristikak etengunearen inguruan, [6] liburutik aterata. Eskuinaldean travelling wave-a $[-0.5, 3] \times [0, 3]$ eremuan, $u_1 = 1$ eta $u_2 = 0.4$ izanik.

Froga dezagun $s = \frac{u_1 + u_2}{2}$ dela eta baita balio hori bakarra dela ere. Demagun M zenbaki handi bat dela, $M \gg st$, orduan badakigu (1.3) kontserbazio legearen forma integraletik,

$$\begin{aligned} \frac{d}{dt} \int_{-M}^M u(x, t) dx &= f(u(-M, t)) - f(u(M, t)) = \\ &= f(u_1) - f(u_2) = \frac{1}{2}u_1^2 - \frac{1}{2}u_2^2 = \\ &= \frac{1}{2}(u_1 + u_2)(u_1 - u_2). \end{aligned} \quad (1.21)$$

Bestetik, soluziotik badakigu t -ren arabera den integralaren balioa,

$$\int_{-M}^M u(x, t) dx = (M + st)u_1 + (M - st)u_2.$$

Adierazpen hau t -rekiko integratuz eta (1.21) lortutakoarekin berdinduz s balioa lortuko dugu:

$$\frac{d}{dt} \int_{-M}^M u(x, t) dx = s(u_1 - u_2).$$

Beraz,

$$s = \frac{u_1 + u_2}{2}. \quad (1.22)$$

Balio hau kontserbazio legeetara orokortu daiteke, *Rankine-Hugonioten baldintza* deritzona lortuaz:

$$f(u_1) - f(u_2) = s(u_1 - u_2). \quad (1.23)$$

Orain, froga dezagun (1.20) funtzioa (1.19) Riemann-en problemaren soluzioa ahula dela. Horretarako, gogoratuko dugu u soluzioa ahula izango dela baldin eta edozein $\phi \in C^\infty$ funtzio hartuta, euskarri trinko-duna,

$$\int_0^\infty \int_{-\infty}^\infty (\phi_t u + \phi_x f(u)) dx dt = - \int_{-\infty}^\infty \phi(x, 0) u(x, 0) dx \quad (1.24)$$

betetzen bada. Hasteko, $x = \frac{u_1 + u_2}{2} t$ zuzenak bereizten dituen bi eremutan u soluzioa konstante denez, integrala bi batugaietan banatuko dugu: aurrengoan soluzioak u_1 balioa hartzen duen eremua integratuko dugu, $x < \frac{u_1 + u_2}{2} t$ denean; eta bigarrean, u_2 hartzen duen eremua, $x > \frac{u_1 + u_2}{2} t$ (1.3. irudia). Beraz, integrala honela geratzen zaigu:

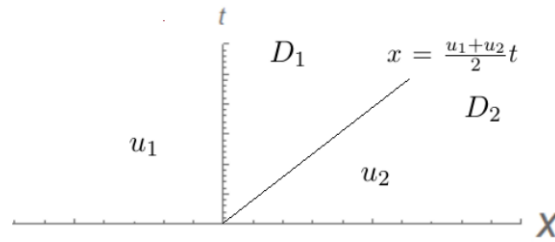
$$\int_0^\infty \int_{-\infty}^\infty \left(\phi_t u + \frac{1}{2} \phi_x u^2 \right) dx dt = \sum_{i=1,2} \iint_{D_i} \left(\phi_t u_i + \frac{1}{2} \phi_x u_i^2 \right) dx dt$$

non D_1 eta D_2 honela definitzen diren:

$$\begin{aligned} D_1 &= \left\{ (x, t) \in \mathbf{R} \times [0, \infty) \mid x < \frac{u_1 + u_2}{2} t \right\}, \\ D_2 &= \left\{ (x, t) \in \mathbf{R} \times [0, \infty) \mid x > \frac{u_1 + u_2}{2} t \right\}. \end{aligned} \quad (1.25)$$

Notazioa errazteko, batugai bakoitza bere aldetik garatuko dugu eta gero biak batuko ditugu, azken emaitza lortzeko.

$$\begin{aligned} (1) &= \iint_{D_1} \left(\phi_t u_1 + \frac{1}{2} \phi_x u_1^2 \right) dx dt = \int_0^\infty \int_{-\infty}^{\frac{u_1 + u_2}{2} t} \left(\phi_t u_1 + \frac{1}{2} \phi_x u_1^2 \right) dx dt, \\ (2) &= \iint_{D_2} \left(\phi_t u_2 + \frac{1}{2} \phi_x u_2^2 \right) dx dt = \int_0^\infty \int_{\frac{u_1 + u_2}{2} t}^\infty \left(\phi_t u_2 + \frac{1}{2} \phi_x u_2^2 \right) dx dt. \end{aligned}$$



1.3. irudia: D_i eremuan soluzioak u_i hartzen du $i = 1, 2$.

Integral hauek garatzeko orduan graduan zehar ikasitako hainbat teknika erabiliko ditugu:

- Baturaren integrala integralen batuketa da. Era berean, funtzio konstanteak integraletik kanpora atera ditzakegu:

$$\int (f + g) = \int f + \int g, \quad \int \lambda f = \lambda \int f.$$

- Barrow-en erregela: $\int_a^b f(x)dx = F(b) - F(a)$, non $F'(x) = f(x)$.
- Fubini-ren teorema: $\int_{A \times B} |f(x, y)| dx dy < \infty$ baldin bada, orduan,

$$\int_{A \times B} f(x, y) dx dy = \int_A \left(\int_B f(x, y) dy \right) dx = \int_B \left(\int_A f(x, y) dx \right) dy.$$

- Aldagai-aldaketa: $x = \frac{u_1+u_2}{2}t$ aldagai-aldaketa egingo dugu, eta beraz, $dx = \frac{u_1+u_2}{2} dt$ izango da.

Bestalde, kontuan hartu behar dugu ϕ euskarri trinkodunaenez funtzioa anulatu egiten dela infinituan: $\phi(x, t)|_{x=\infty}$ edo $t=\infty = 0$.

Teoria guzti hori kontuan hartuta, integralak garatuko ditugu:

$$\begin{aligned}
(1) &= u_1 \int_0^\infty \int_{-\infty}^{\frac{u_1+u_2}{2}t} \phi_t \, dx dt + \frac{u_1^2}{2} \int_0^\infty \phi \Big|_{-\infty}^{\frac{u_1+u_2}{2}t} \, dt \\
&= u_1 \int_{-\infty}^0 \int_0^\infty \phi_t \, dt dx + u_1 \int_0^\infty \int_{\frac{2}{u_1+u_2}x}^\infty \phi_t \, dt dx + \frac{u_1^2}{2} \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt \\
&= -u_1 \int_{-\infty}^0 \phi(x, 0) \, dx - u_1 \int_0^\infty \phi \left(x, \frac{2}{u_1+u_2}x \right) \, dx + \frac{u_1^2}{2} \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt \\
&\stackrel{x = \frac{u_1+u_2}{2}t}{=} -u_1 \int_{-\infty}^0 \phi(x, 0) \, dx + \left(-u_1 \frac{u_1+u_2}{2} + \frac{u_1^2}{2} \right) \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt \\
&= -u_1 \int_{-\infty}^0 \phi(x, 0) \, dx - \frac{u_1 u_2}{2} \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt.
\end{aligned}$$

$$\begin{aligned}
(2) &= u_2 \int_0^\infty \int_{\frac{u_1+u_2}{2}t}^\infty \phi_t \, dx dt + \frac{u_2^2}{2} \int_0^\infty \phi \Big|_{\frac{u_1+u_2}{2}t}^\infty \, dt \\
&= u_2 \int_0^\infty \int_0^{\frac{2}{u_1+u_2}x} \phi_t \, dt dx - \frac{u_2^2}{2} \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt \\
&= u_2 \int_0^\infty \phi \Big|_0^{\frac{2}{u_1+u_2}x} \, dx - \frac{u_2^2}{2} \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt \\
&= -u_2 \int_0^\infty \phi(x, 0) \, dx + u_2 \int_0^\infty \phi \left(x, \frac{2}{u_1+u_2}x \right) \, dx - \frac{u_2^2}{2} \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt \\
&\stackrel{x = \frac{u_1+u_2}{2}t}{=} -u_2 \int_0^\infty \phi(x, 0) \, dx + \left(u_2 \frac{u_1+u_2}{2} - \frac{u_2^2}{2} \right) \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt \\
&= -u_2 \int_0^\infty \phi(x, 0) \, dx + \frac{u_1 u_2}{2} \int_0^\infty \phi \left(\frac{u_1+u_2}{2}t, t \right) \, dt.
\end{aligned}$$

Orain, bi batugaiak batuta, (1) + (2), bakoitzaren bigarren gaia ezabatuko da eta emaitza lortuko dugu:

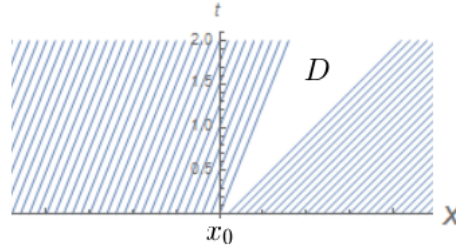
$$\begin{aligned}
\int_0^\infty \int_{-\infty}^\infty \phi_t u + \frac{1}{2} \phi_x u^2 \, dx dt &= (1) + (2) \\
&= -u_1 \int_{-\infty}^0 \phi(x, 0) \, dx - u_2 \int_0^\infty \phi(x, 0) \, dx \\
&= - \int_{-\infty}^\infty \phi(x, 0) u(x, 0) \, dx.
\end{aligned} \tag{1.26}$$

Frogatu dugu (1.20) funtzioa (1.19) Riemann-en problemaren soluzioa ahula dela.

ii) $u_1 < u_2$:

Bigarren kasu honetan, x_0 puntutik alde positiborantz karakteristikak beste aldera baino azkarrago doaz, eta beraz, ez dira elkar gurutzatuko. Ondorioz, D eremutik karakteristikak ez dira pasako:

$$D = \{(x, t) \in \mathbf{R} \times (0, \infty) \mid x_0 + u_1 t < x < x_0 + u_2 t\}. \quad (1.27)$$



1.4. irudia: Karakteristikak igarotzen ez diren D eremua.

Eremu horretan karakteristikak nolakoak diren definitu beharko dugu, betiere (1.23) Rankine-Hugonioten baldintza betetzen dutelarik. Problema soluzio ahul asko izango ditu, eremu horretan hartzen ditugun erabakien arabekoak. Gutxienez, horrelako bi soluzio ahul badaude: batetik, errarefakzio-talka, eta bestetik, errarefakzio-uhina.

Errarefakzio-talka (1.20) eta (1.22) adierazpenek definitzen dute:

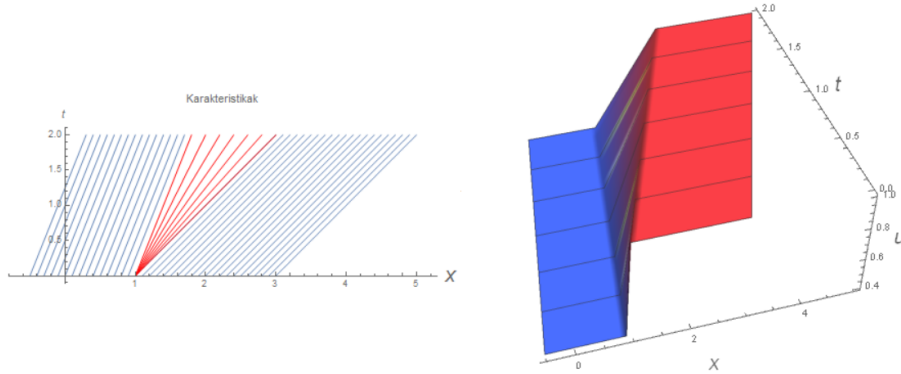
$$u(x, t) = \begin{cases} u_1 & x < st, \\ u_2 & x > st. \end{cases} \quad (1.28)$$

non $s = \frac{u_1 + u_2}{2}$ den. Soluzio ahul honek Rankine-Hugonioten baldintza betetzen badu ere, ez du geroago definituko dugun entropia baldintza betetzen, eta beraz, ez digu balioko.

Errarefakzio-uhinak, aldiz, entropia baldintza betetzen du. Kasu honetan, karakteristikak x_0 puntutik lerro zuzenetan abiatzen dira D eremuko puntu guztietara eta soluzioak ez du etengunerik izango (1.5. irudia):

$$u(x, t) = \begin{cases} u_1, & x < u_1 t, \\ \frac{x}{t}, & u_1 t < x < u_2 t, \\ u_2, & x > u_2 t. \end{cases} \quad (1.29)$$

Soluzioak ez du etengunerik eta hasierako baldintzako etengunea zabalitzen joango da denboran zehar. Godunov-en metodoa aplikatzerako orduan, soluzio hau erabiliko dugu $u_1 < u_2$ den kasuetan.



1.5. irudia: a) Errarefakzio-uhinarentzat karakteristikak $[-0.5, 3]$ tartetik abiatuta. Gorriz D eremuan definitu ditugunak. b) Errarefakzio-uhina $[0, 5]$ eremuan $x_0 = 1$, $u_1 = 0.4$ eta $u_2 = 1$ kasurako.

4 Entropia baldintzak

Ikusi berri dugun bezala, askotan soluzio ahula ez da bakarria, eta beraz, baldintza gehiago behar ditugu soluzio zuzena zehazteko. Biskositatedun ekuazioa biskosite-konstantea zerora eramanda lortzen den soluzioa izan behar da. Hala ere, askotan hau aztertzea ez da erraza.

[6] liburuak kontserbazio lege orokorretzat entropia baldintzen hiru bertsiio ematen dizkigu:

- **Entropia baldintza (I bertsiioa):**

(1.23) adierazpenaz definitzen den s abiaduraz hedatzen den etengune batek entropia baldintza betetzen du baldin eta

$$f'(u_1) > s > f'(u_2) \quad (1.30)$$

betetzen bada. Funtzioa konbexua^{††} baldin bada, Burgersena kasu, $f(u) = \frac{1}{2}u^2$ denez, $f'(u) = u$ eta (1.30) baldintza $u_1 > u_2$ desberdintzara sinplifikatzen da. Ondorioz, ii) atalean ezin dugu erabili denboran zehar mantentzen den errarefakzio-talka.

- **Entropia baldintza (II bertsiioa):** $u(x, t)$ entropia soluzioa izango da baldin eta etengune guztiek

$$\frac{f(u) - f(u_1)}{u - u_1} > s > \frac{f(u) - f(u_2)}{u - u_2} \quad (1.31)$$

^{††} *Funtzio konbexua:* funtzio erreala bat (a, b) tartean konbexua dela esango dugu tarteko edozein puntuetan zuzen ukitzaila puntu horretan funtzioaren azpian geratzen bada. Hau da,

$$\forall t \in [0, 1], f(ta + (1-t)b) \leq tf(a) + (1-t)f(b).$$

betetzen badute edozein $u \in (u_1, u_2)$ baliotarako.

• **Entropia baldintza (III bertsioa):**

$u(x, t)$ entropia soluzioa izango da baldin eta existitzen bada $E > 0$ konstante bat non edozein $a > 0$, $t > 0$ eta $x \in \mathbf{R}$ balioetarako,

$$\frac{u(x+a, t) - u(x, t)}{a} < \frac{E}{t}. \quad (1.32)$$

1.3 Burgersen ekuazioa biskositatearekin

Ekuazioaren forma orokorra ebatziko dugu atal honetan, (1.9) Hopf-Cole-en tranformazioa erabiliz [5] artikuluan azaltzen den moduan. Biskositatearen gaiak ez dio utziko soluzioari talka-egoerara iristen, u_x handitzen denean, u_{xx} are gehiago handituko baita. Hortaz, uhina egoera horretara iritsiko ez denez soluzio bakarra izango du, deribagarria izango dena, eta beraz, baliagarria dena.

Gogoratu, Burgersen ekuazioa biskositatearekin:

$$u_t + uu_x = \nu u_{xx}, \quad (1.33)$$

eta Hopf-Cole-en transformazioa:

$$u(x, t) = -2\nu \frac{\varphi_x}{\varphi}. \quad (1.9)$$

Problema ebazten hasteko, u_t , u_x eta u_{xx} elementuei (1.9) aldagai-aldaketa aplikatuko diegu:

$$\begin{aligned} u_t &= \frac{2\nu(\varphi_t \varphi_x - \varphi \varphi_{xt})}{\varphi^2}, \\ u_x &= \frac{2\nu(\varphi_x^2 - \varphi \varphi_{xx})}{\varphi^2}, \\ u_{xx} &= -\frac{2\nu(2\varphi_x^3 - 3\varphi \varphi_x \varphi_{xx} + \varphi^2 \varphi_{xxx})}{\varphi^3}. \end{aligned} \quad (1.34)$$

Adierazpen hauek (1.33) ekuazioan ordezkatzuz,

$$\frac{2\nu(\varphi_t \varphi_x - \varphi \varphi_{xt})}{\varphi^2} - \frac{4\nu^2(\varphi_x^3 - \varphi \varphi_x \varphi_{xx})}{\varphi^3} = -\frac{2\nu^2(2\varphi_x^3 - 3\varphi \varphi_x \varphi_{xx} + \varphi^2 \varphi_{xxx})}{\varphi^3}$$

lortuko dugu. Ekuazio berri hori φ^3 -z biderkatuko dugu:

$$2\nu\varphi\varphi_x\varphi_t - 2\nu\varphi^2\varphi_{xt} - 4\nu^2\varphi_x^3 + 4\nu^2\varphi\varphi_x\varphi_{xx} = -4\nu^2\varphi_x^3 + 6\nu^2\varphi\varphi_x\varphi_{xx} - 2\nu^2\varphi^2\varphi_{xxx}.$$

Ekuazioa are gehiago sinplifikatzen badugu,

$$\varphi_x\varphi_t - \varphi\varphi_{xt} - \nu\varphi_x\varphi_{xx} + \nu\varphi\varphi_{xxx} = 0$$

lortuko dugu, eta ondorioz,

$$\varphi_x(\varphi_t - \nu\varphi_{xx}) = \varphi(\varphi_t - \nu\varphi_{xx}).$$

Puntu honetan, $\frac{\varphi_x}{\varphi} = 1$ izateak u funtzio konstatea izatea behartuko luke (1.9) aldagai-aldaketa kontuan hartuta, eta beraz, ez da bilatzen ari garen soluzioa. Hortaz, φ -rentzat beroaren ekuazioa lortu dugu:

$$\varphi_t - \nu\varphi_{xx} = 0. \quad (1.35)$$

Hasierako baldintza kontuan hartuta problema osatuko dugu.

$$u(x, 0) = -2\nu \frac{\varphi_x(x, 0)}{\varphi(x, 0)} = u_0(x),$$

hau da,

$$\varphi(x, 0) = \exp\left(-\int_0^x \frac{u_0(y)}{2\nu} dy\right) = \varphi_0(x). \quad (1.36)$$

(1.35) eta (1.36) adierazpenak elkartuz, problema ongi definituta gertzen zaigu, $\nu > 0$ baterako:

$$\begin{cases} \varphi_t - \nu\varphi_{xx} = 0, & x \in \mathbf{R}, t > 0, \\ \varphi(x, 0) = \varphi_0(x), & x \in \mathbf{R}. \end{cases} \quad (1.37)$$

Problema hau ebazteko beroaren ekuazio homogeneoaren soluzioa gogora ekarriko dugu, $(-\infty, \infty)$ tartean:

$$\begin{cases} w_t - \nu w_{xx} = 0, & x \in \mathbf{R}, t > 0, \\ w(x, 0) = w_0(x), & x \in \mathbf{R}, \end{cases}$$

problemaren soluzioa ondorengoa da:

$$w(x, t) = \frac{1}{\sqrt{4\pi\nu t}} \int_{-\infty}^{\infty} \exp\left(-\frac{(x-y)^2}{4\nu t}\right) w_0(y) dy.$$

Adierazpen hau gure kasura ekarrita φ -ren adierazpena lortuko dugu:

$$\varphi(x, t) = \frac{1}{\sqrt{4\pi\nu t}} \int_{-\infty}^{\infty} \exp\left(-\frac{(x-y)^2}{4\nu t}\right) \varphi_0(y) dy, \quad (1.38)$$

eta baita bere deribatuarena ere,

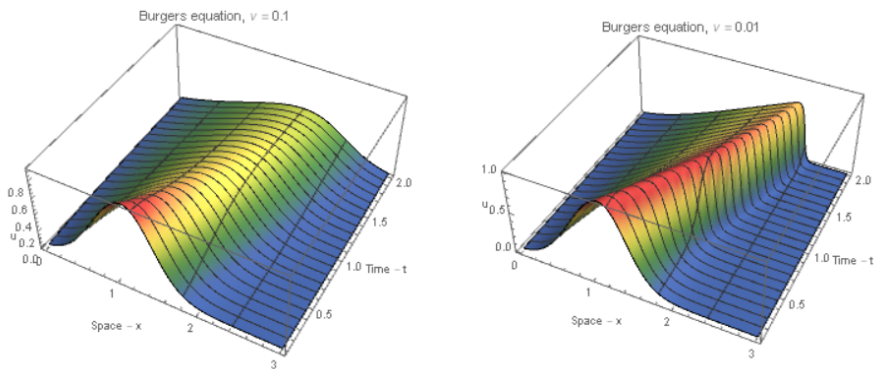
$$\varphi_x(x, t) = \frac{1}{\sqrt{4\pi\nu t}} \int_{-\infty}^{\infty} -\frac{(x-y)}{2\nu t} \exp\left(-\frac{(x-y)^2}{4\nu t}\right) \varphi_0(y) dy. \quad (1.39)$$

Beraz, berriz ere (1.9) aldagai-aldaketa erabiltzen badugu, ebatzi nahi genuen problemaren u soluzioa lortu dugu:

$$u(x, t) = \frac{\int_{-\infty}^{\infty} \frac{(x-y)}{t} \exp\left(-\frac{(x-y)^2}{4\nu t}\right) \varphi_0(y) dy}{\int_{-\infty}^{\infty} \exp\left(-\frac{(x-y)^2}{4\nu t}\right) \varphi_0(y) dy} \quad (1.40)$$

non φ_0 (1.36) adierazpenaren bidez definitzen den.

Ondorioz, (1.40) soluzioari esker ikusi dugu (X, T) planoko puntu bakoitzari $u(x, t)$ balio bat eta bakar bat dagokiola, ν biskositate-parametroaren araberakoa dena. Parametro hori zenbat eta handiagoa izan, soluzioa orduan eta leunagoa da; zerotik gertu dagoen balio bat ematen badiogu, berriz, soluzioak etengune forma hartuko du, betiere puntu bakoitzean balio bakar bat emanez (1.6. irudia). Entropia baldintza betetzen duen Burgesen ekuazio homogeneoaren soluzio ahula $\nu \rightarrow 0$ eramanda lortzen dugun soluzioaren berdina izan behar da.



1.6. irudia: Biskositatedun Burgesen ekuazioaren soluzioa $[0, 3] \times [0, 2]$ eremuan $u(x, 0) = \exp(-2(x - 1)^2)$ hasierako baldintzarekin. Ezkerraldean $\nu = 0.1$ hartuta eta eskuinaldean $\nu = 0.01$.

2. kapitulua

Zenbakizko metodoak

Ekuazio diferentzial asko, Burgersena esaterako, ez-linealak dira eta analitikoki soluzio zehatza lortzea oso zaila edo ezinezkoa izan daiteke. Zenbakizko metodoen helburua problemari soluzio hurbildua ematea da, ahalik eta zehaztasun handienarekin. Funtsezkoa izango da metodoaren egonkortasuna eta errorea aztertzea emaitzaren fidagarritasuna neurtzerako garaian.

Kapitulu honen hasieran zenbakizko metodoetako kontzeptu arrunt batzuk definituko ditugu, ondoren diferentzia finituetako metodo batzuk aztertuko ditugu, eta azkenik, Goduven metodoa garatuko dugu.

2.1 Zenbakizko metodoen sarrera

Zenbakizko metodoek duten ezaugarri nagusietako bat espazio finitu batean aztertu behar direla da. Horretarako, (X, T) plano diskretizatu beharko dugu.

Hasteko, X espazioa goitik eta behetik bornatu beharko dugu: kalkulatu nahi ditugun balioen arabera egingo dugu bornaketa hori. Kasu ohikoena ekuazioa X ardatzean $[0, L]$ tartean aztertzea izango da eta diskretizazioa honela geratuko zaigu:

$$0 = x_0 < x_1 < \dots < x_{M-1} < x_M = L \quad (2.1)$$

non $x_i = i\Delta x$ izango den $i = 0, 1, \dots, M$,

$$\Delta x = x_{i+1} - x_i = \frac{L}{M} \quad (2.2)$$

izanik. Espazioko posizioa deritzo i azpi-indizeari. Beste edozein $[a, b]$ tarte aztertu behar badugu translazio bat eginda lortuko dugu.

T aldagaia ere diskretizatu egingo dugu: nahi dugun Δt denbora-pausua hartuko dugu eta t^n notazioaz $n\Delta t$ adieraziko dugu. Aztertu nahi dugun denbora-tartearen arabera n -ren balioa handituko dugu. Hortaz, $(i\Delta x, n\Delta t) = (x_i, t^n)$ puntuek $[0, L] \times [0, T_{buk}]$ eremua puntu diskretuz ordezkaturiko dute.

Gure helburua puntu bakoitzean u funtzioak hartuko duen balio hurbildua kalkulatzeko izango da:

$$u_i^n \approx u(x_i, t^n). \quad (2.3)$$

Notazioa errazteko, denbora- eta espazio-tarteak h -ren eta k -ren bidez adieraziko ditugu:

$$\begin{aligned} \Delta x &= h, \\ \Delta t &= k. \end{aligned} \quad (2.4)$$

Zenbakizko metodoen formulazioaren gakoa ekuazio diferentzialean azaltzen diren u funtzio ezezagunaren deribatuak u -ren balioez ordezkatzean datza, eta ondoren, mugalde-baldintza egokiak kontuan hartuta, denbora-pausu bakoitzean hurrengo denbora-pausuko balioak kalkulatzeko.

Definizioa 2.1.1. (Metodo baten maila)

Metodo bat r mailakoa dela esango dugu, baldin eta $n+1$ -garren denbora-pausuko balioak kalkulatzeko r denbora-pausutako balioak behar badira.

Definizioa 2.1.2. (Metodo explizitua)

Metodo bat *explizitua* dela esango dugu, iterazio bakoitzean $n+1$ -garren denbora-pausuko balioak kalkulatzeko soilik aurreko denbora-pausuetako balioak behar badira.

Lan honetan ikusiko ditugun metodo guztiak explizituak izango dira.

Definizioa 2.1.3. (Metodo implizitua)

Metodoa *implizitua* dela esango dugu, iterazio bakoitzean $n+1$ -garren denbora-pausuko balioak kalkulatzeko $n+1$ -garren denbora-pausuko balioak behar baditugu. Kasu honetan, balio hori kalkulatzeko sistema lineal bat ebatzi beharko dugu.

Funtzioen deribatu partzialen hurbilketak lortzeko funtsezko teorema bat Taylorrena da:

Teorema 2.1.1. (Taylorren teorema)

Izan bedi $f : [a, b] \subseteq \mathbf{R} \rightarrow \mathbf{R}$ n -garren deribatu arte deribatu jarraituak dituen funtzio bat, $f^{(n)}$ deribagarria izanik. Orduan existitzen da $c \in [a, b]$ balio bat non

$$f(b) = f(a) + f'(a)(b-a) + \dots + \frac{f^{(n)}(a)}{n!}(b-a)^n + \frac{f^{(n+1)}(c)}{(n+1)!}(b-a)^{n+1}.$$

Beraz, u funtzioaren n -garren ordenako Taylorren polinomioa $x_0 + \Delta x$ puntuan x_0 -tik abiatuta ondokoa da:

$$u(x_0 + \Delta x, t) = u(x_0, t) + h u_x(x_0, t) + \dots + h^n \frac{u_{nx}(x_0, t)}{n!} + O(h^{n+1}). \quad (2.5)$$

Oro har, ezin dugu esan u soluzioa n -aldiz deribagarria denik, baina ekuazio diferentzian u_x azaltzen denez, soluzioa gutxienez x -rekiko deribagarria izan behar du. Hortaz, u_x terminoaren hurbilketa lortzeko, u_{i+1} puntuan (2.5) Taylorren polinomioa garatu eta bigarren batugaian moztuko dugu:

$$u|_{i+1} = u|_i + h u_x|_i + O(h). \quad (2.6)$$

Formula honetatik, u_x terminoaren *aurreranzko* hurbilketa lortuko dugu:

$$u_x|_i \approx \frac{u_{i+1} - u_i}{h} \quad (\text{aurreranzkoa}). \quad (2.7)$$

$O(h)$ ordeneko hurbilketa da. Taylorren polinomio u_{i-1} puntuan garatzen badugu u_i puntutik abiatuta, u_x -ren *atzeranzko* hurbilketa lortuko dugu:

$$u_x|_i \approx \frac{u_i - u_{i-1}}{h} \quad (\text{atzeranzkoa}). \quad (2.8)$$

Hurbilketa honen ordena ere $O(h)$ izango da. Eskemaren ordena hobetu nahi badugu, hurbilketa *zentrala* egitea komeniko zaigu:

$$u_x|_i \approx \frac{u_{i+1} - u_{i-1}}{2h} \quad (\text{zentrala}). \quad (2.9)$$

Taylorren polinomioa u_{i+1} eta u_{i-1} puntuetan garatzen badugu eta geo biak batu, hurbilketa zentrala lortuko dugu. Hurbilketa honen ordena $O(h^2)$ -koa izango da, eta beraz, hurbilketa hobea da.

Hurbilketa berdinak aplikatu genizkioke u_t gaiari ere. Ondorioz, n -garren denbora-pausuan, u_t honela hurbildu ahal izango dugu:

$$u_t|_n \approx \frac{u^{n+1} - u^n}{k} \quad (\text{aurreranzkoa}), \quad (2.10)$$

$$u_t|_n \approx \frac{u^n - u^{n-1}}{k} \quad (\text{atzeranzkoa}), \quad (2.11)$$

$$u_t|_n \approx \frac{u^{n+1} - u^{n-1}}{2k} \quad (\text{zentrala}). \quad (2.12)$$

Elementu bakoitza definituta, zenbakizko eskema definituko dugu. Demagun $u_t + uu_x = 0$ Burgers-en ekuazioa dugula Riemann-en hasierako baldintzarekin:

$$u_0(x) = \begin{cases} 1, & x < 0, \\ 0, & x > 0. \end{cases} \quad (2.13)$$

Denboran aurreranzko (2.10) eta espazioan atzeranzko (2.8) hurbilketak aplikatzen baditugu u_i^n puntuan, zenbakizko eskema honela geratuko zaigu:

$$\left(\frac{u_i^{n+1} - u_i^n}{k} \right) + u_i^n \left(\frac{u_i^n - u_{i-1}^n}{h} \right) = 0.$$

Beste modu batera idatzita, pausu bakoitzean egin beharreko iterazioa hau izango da:

$$u_i^{n+1} = u_i^n - \frac{k}{h} u_i^n (u_i^n - u_{i-1}^n). \quad (2.14)$$

Problemaren (2.13) hasierako baldintza kontuan hartzen badugu, $u_i^0 = 0$ izango da $i \geq 0$ guztietarako eta $u_i^0 = 1$, $i < 0$ guztietarako. Orduan, hurrengo pausuko balioak kalkulatzeko baditugu, aurreko pausuko balio berak lortuko ditugu;

$$u_j^1 = \begin{cases} 1 - \frac{k}{h} 1(1-1) = 1, & j < 0 \\ 0 - \frac{k}{h} 0(0 - u_{i-1}^0) = 0, & j \geq 0 \end{cases} = u_j^0. \quad (2.15)$$

Lehenengo denbora-pausuan funtzioaren balioa ez bada aldatzen, gainontzeko denbora-pausutan ere ez da aldatuko, eta beraz, $u_j^n = u_j^0$, $\forall n$. Soluzioa ez da zuzena, eta ondorioz, metodoa okerra dela esan dezakegu.

Adibide honek baieztatzen digu etenguneetan edozein metodok ez duela balio. Zenbakizko eskemek baldintza zehatz batzuk bete behar dituzte etengunea behar bezala hedatzeko denboran zehar. Kontserbazio legeak dituzten DPE ebazteko metodo kontserbakorrak erabili behar dira.

2.2 Metodo kontserbakorrak

Metodo kontserbakorrak definitu aurretik, $u_t + (f(u))_x = 0$ kontserbazio legeen esanahia gogoratuko dugu [3] artikuluan azaltzen den moduan. Hau da, $[x_1, x_2]$ espazio-tartean eta $[t_1, t_2]$ denbora-tartean egoera aldagaiaren balio totala,

$$\int_{x_1}^{x_2} u(x, t) dx,$$

mugaldeetako fluxu aldaketagatik bakarrik alda daiteke, hau da,

$$\int_{x_1}^{x_2} u(x, t_2) dx = \int_{x_1}^{x_2} u(x, t_1) dx + \int_{t_1}^{t_2} f(u(x_1, t)) dt - \int_{t_1}^{t_2} f(u(x_2, t)) dt. \quad (2.16)$$

Definizioa 2.2.1. (Zenbakizko metodo kontserbakorrak)

Zenbakizko eskema batek (1.1) kontserbazio legearentzat *forma kontserbakorra* duela esaten da baldin eta ondoko forma badauka:

$$u_i^{n+1} = u_i^n - \frac{k}{h} [F(u_{i-p}^n, \dots, u_{i+q}^n) - F(u_{i-p-1}^n, \dots, u_{i+q-1}^n)], \quad (2.17)$$

non F $p + q + 1$ argumentu dituen funtzio bat den. F funtzioari *zenbakizko fluxu funtzioa* deritzo. Kasurik sinpleena $p = 0$ eta $q = 1$ hartuta lortuko dugu; F bi aldagaietako funtzioa izango da, eta (2.17) honela geratuko zaigu:

$$u_i^{n+1} = u_i^n - \frac{k}{h} [F(u_i^n, u_{i+1}^n) - F(u_{i-1}^n, u_i^n)]. \quad (2.18)$$

Forma kontserbakorra duten zenbakizko metodoei *metodo kontserbakorrak* deritze.

Definizio hau berehalakoa izan daiteke (2.16) ekuaziotik u_i^n puntuz-puntuko balioaren hurbilketaren ordez, $[x_{i-1/2}, x_{i+1/2}] \times [t^n, t^{n+1})$ gelaxkaren batezbestekoaren hurbilketa bezala ikusten badugu, hau da,

$$u_i^n \approx \frac{1}{h} \int_{x_{i-1/2}}^{x_{i+1/2}} u(x, t^n) dx. \quad (2.19)$$

Kontserbazio legearen (2.16) forma integrala $[x_{i-1/2}, x_{i+1/2}] \times [t^n, t^{n+1})$ gelaxkan ondokoa da:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} u(x, t^{n+1}) dx = \int_{x_{i-1/2}}^{x_{i+1/2}} u(x, t^n) dx - \left[\int_{t^n}^{t^{n+1}} f(u(x_{i+1/2}, t)) dt - \int_{t^n}^{t^{n+1}} f(u(x_{i-1/2}, t)) dt \right]. \quad (2.20)$$

Ekuazio hau h espazio-tartez zatitzen badugu, (2.19) hurbilketa kontuan hartuta,

$$u_i^{n+1} = u_i^n - \frac{1}{h} \left[\int_{t^n}^{t^{n+1}} f(u(x_{i+1/2}, t)) dt - \int_{t^n}^{t^{n+1}} f(u(x_{i-1/2}, t)) dt \right] \quad (2.21)$$

lortuko dugu. Iterazio honek (2.18) metodo kontserbakorraren forma dauka, F fluxu-funtzioa $[t^n, t^{n+1})$ denbora-tartean f funtzioaren batez besteko balioa bada, hau da:

$$F(u_i^n, u_{i+1}^n) \approx \frac{1}{k} \int_{t^n}^{t^{n+1}} f(u(x_{i+1/2}, t)) dt. \quad (2.22)$$

Hurbilketa hau kontuan hartuta, Burgersen ekuazioa (1.1) kontserbazio legeen forman ikusten badugu $f(u) = \frac{1}{2}u^2$ hartuta, Burgersen ekuazioarentzat **upwind metodoa** lortuko dugu:

$$u_i^{n+1} = u_i^n - \frac{k}{h} \left[\frac{1}{2}(u_i^n)^2 - \frac{1}{2}(u_{i-1}^n)^2 \right]. \quad (2.23)$$

Definizioa 2.2.2. (Metodo trinkoa)

(2.17) metodo kontserbakorra (1.1) kontserbazio legearekiko *trinkoa* izango da baldin eta F zenbakizko fluxu funtzioa f kontserbazio legeko fluxu funtziora murrizten bada, fluxu konstantearen kasuan. Hau da, $u(x, t) \equiv \bar{u}$ bada,

$$F(\bar{u}, \dots, \bar{u}) = f(\bar{u}). \quad (2.24)$$

Are gehiago, baldin eta F -ren argumentuak \bar{u} balio konstante batera badoaz, orduan F funtzioak $f(\bar{u})$ baliora jo behar du:

$$\lim_{v_1, \dots, v_n \rightarrow \bar{u}} F(v_1, \dots, v_n) = f(\bar{u}). \quad (2.25)$$

Horretaz gain, F funtzioaren deribatuen jarraitasuna ere beharrezkoa da metodoa trinkoa izateko.

F zenbakizko fluxu funtzioa \bar{u} balioan *Lipschitz jarraitua* dela esango dugu existitzen bada $K = K(\bar{u}) \geq 0$ \bar{u} -ren menpekoa izan daitekeen konstante bat, non

$$|F(v_1, \dots, v_n) - f(\bar{u})| \leq K \max_{-p \leq i \leq q} (|v_i - \bar{u}|), \quad (2.26)$$

$|v_i - \bar{u}|$ nahiko txikiak izanik $i = -p, \dots, q$ guztietarako.

Beraz, metodo bat trinkoa izateko baldintzak hauek dira:

$$\begin{aligned} F(\bar{u}, \dots, \bar{u}) &= f(\bar{u}), \\ |F(v_1, \dots, v_n) - f(\bar{u})| &\leq K \max_{-p \leq i \leq q} (|v_i - \bar{u}|). \end{aligned} \quad (2.24), (2.26)$$

Burgersen ekuazioarentzat (2.23) upwind metodoa trinkoa dela berehalakoa da zenbakizko F fluxu funtzioa kontserbazio legearen f fluxu funtzio bera baita:

$$F(u_i^n, u_{i+1}^n) = f(u_i^n)$$

Metodo batek forma kontserbakorra izatearen eta trinkoa izatearen azterketa oso erabilgarria da hurrengo teoremari esker. Frogapena egiteko [6] liburuan oinarrituko gara.

Teorema 2.2.1. (*Lax-Wendroff-en teorema*)

Izan bitez $x_0 < \dots < x_M$ eta $0 = t^0 < \dots < t^R = T_{buk}$, $[x_0, x_M] \times [0, T_{buk}]$ espazioaren diskretizazioa, non $x_{i+1} - x_i = h$ eta $t^{n+1} - t^n = k$ diren, eta zenbakizko metodo trinko eta kontserbakorrek emandako deribatu partzialeko soluzioa hurbildu (x_i, t^n) puntuan u_i^n . Baldin eta existitzen bada $u(x, t)$ funtzio bat, non

$$\lim_{h, k \rightarrow 0} u_i^n = u(x_i, t^n) \quad \forall i = 0, 1, \dots, M, \quad \forall n = 0, \dots, R,$$

orduan $u(x, t)$ funtzioa kontserbazio legearen soluzio ahula da.

Froga. $u(x, t)$ funtzioa soluzio ahula bada, hurrengo ekuazioa bete behar du:

$$\int_0^\infty \int_\infty^\infty \phi_t u + \phi_x f(u) \, dx dt = - \int_{-\infty}^\infty \phi(x, 0) u(x, 0) \, dx. \quad (1.17)$$

Har dezagun $\phi \in C^\infty$ test funtzioa eta $\phi(x_i, t^n)$ (2.18) metodo kontserbakorren iterazio bakoitzarekin biderkatuko dugu:

$$\phi(x_i, t^n) u_i^{n+1} = \phi(x_i, t^n) u_i^n - \frac{k}{h} \phi(x_i, t^n) [F(u_i^n, u_{i-1}^n) - F(u_{i+1}^n, u_i^n)].$$

Orain, $n \geq 0$ iterazio guztiak batuz,

$$\sum_{n=0}^{\infty} \sum_{i=-\infty}^{\infty} \phi(x_i, t^n) (u_i^{n+1} - u_i^n) = -\frac{k}{h} \sum_{n=0}^{\infty} \sum_{i=-\infty}^{\infty} \phi(x_i, t^n) [F(u_i^n, u_{i+1}^n) - F(u_{i-1}^n, u_i^n)] \quad (2.27)$$

lortuko dugu. Ekuazio hau sinplifikatzeko, hurrengo identitatea hartuko dugu kontuan:

$$\sum_{j=0}^m a_j (b_{j+1} - b_j) = -a_0 b_0 + a_{m-1} b_m + \sum_{j=1}^m (a_{j-1} - a_j) b_j.$$

Batukaria infinituraino egiten ari garenez eta ϕ euskarri trinkoduna denez, infinituan, $x_{\pm\infty}$ edo t^∞ denean, hartuko duen balioa zero izango da. Identitate hori kontuan hartuz, (2.27) ekuazioa honela geratuko zaigu:

$$\begin{aligned} & -\sum_{i=-\infty}^{\infty} \phi(x_i, t^0) u_i^0 + \sum_{n=1}^{\infty} \sum_{i=-\infty}^{\infty} (\phi(x_i, t^{n-1}) - \phi(x_i, t^n)) u_i^n \\ & = \frac{-k}{h} \sum_{n=0}^{\infty} \sum_{i=-\infty}^{\infty} (\phi(x_i, t^n) - \phi(x_{i+1}, t^n)) F(u_i^n, u_{i+1}^n) \\ & = \frac{k}{h} \sum_{n=0}^{\infty} \sum_{i=-\infty}^{\infty} (\phi(x_{i+1}, t^n) - \phi(x_i, t^n)) F(u_i^n, u_{i+1}^n). \end{aligned}$$

Jarraian, ekuazio hau h -z biderkatuko dugu eta ekuazioa berregituratuko dugu, hasierako uneko balioak alde batera utziz:

$$\begin{aligned} & hk \left(\sum_{n=1}^{\infty} \sum_{i=-\infty}^{\infty} \frac{(\phi(x_i, t^n) - \phi(x_i, t^{n-1}))}{k} u_i^n \right. \\ & \left. + \sum_{n=0}^{\infty} \sum_{i=-\infty}^{\infty} \frac{(\phi(x_{i+1}, t^n) - \phi(x_i, t^n))}{h} F(u_i^n, u_{i+1}^n) \right) = -h \sum_{i=-\infty}^{\infty} \phi(x_i, t^0) u_i^0. \end{aligned} \quad (2.28)$$

$h, k \rightarrow 0$ eginik, (2.28) ekuazioko lehenengo gaia eta eskuinaldekoa (1.17) ekuazioko lehenengo eta eskuinaldeko batugaietara joaten dira, hurrenez hurren:

$$\begin{aligned} & hk \sum_{n=1}^{\infty} \sum_{i=-\infty}^{\infty} \frac{(\phi(x_i, t^n) - \phi(x_i, t^{n-1}))}{k} u_i^n \xrightarrow{h, k \rightarrow 0} \int_0^\infty \int_{-\infty}^\infty \phi_t u \, dx dt, \\ & -h \sum_{i=-\infty}^{\infty} \phi(x_i, t^0) u_i^0 \xrightarrow{h \rightarrow 0} - \int_{-\infty}^\infty \phi(x, 0) u(x, 0) \, dx. \end{aligned} \quad (2.29)$$

Bestetik, metodoa trinkoa denez, F Lipschitz jarraitua da u_i^n balioan,

$$|F(u_i^n, u_{i+1}^n) - f(u_i^n)| \leq K |u_{i+1}^n - u_i^n|,$$

non K Lipschitz konstantea den. Aurreko pausuan egin bezala, $h \rightarrow 0$ eramaten badugu, $|u_{i+1}^n - u_i^n| \rightarrow 0$ joango da, edo baliokidea dena,

$$F(u_i^n, u_{i+1}^n) \xrightarrow{h \rightarrow 0} f(u_i^n). \quad (2.30)$$

Beste gaiekin egin bezala, $h, k \rightarrow 0$ eginez eta (2.30) kontuan hartuz, falta zaigun gaia lortuko dugu:

$$hk \sum_{n=0}^{\infty} \sum_{i=-\infty}^{\infty} \frac{(\phi(x_{i+1}, t^n) - \phi(x_i, t^n))}{h} F(u_i^n, u_{i+1}^n) \xrightarrow{h, k \rightarrow 0} \int_0^{\infty} \int_{-\infty}^{\infty} \phi_x f(u) dx dt. \quad (2.31)$$

Azkenik, (2.29) eta (2.31) limiteak (2.28) ekuazioa ordezkatuz, soluzio ahulek bete behar duten (1.17) ekuazioa lortuko dugu. \square

1 Diferentzia finituko metodo kontserbakor batzuk

Atal honetan, $u_t + (f(u))_x = 0$ kontserbazio lege orokorrarentzat zenbait diferentzia finituko metodo kontserbakor aztertuko ditugu Burgersen ekuazioan aplikatuta. Eskema bakoitzaren logika behatuko dugu, forma kontserbakorra dutela frogatu eta baita metodo trinkoak direla ere.

Lax-Friedrichs

Zenbakizko eskema denboran (2.10) aurreranzko hurbilketa egitean u_i^n elemetuaren ordezkari espazioko ondoko posizioetako batezbestekoa hartzerakoan lortuko dugu. Fluxuari dagokionez ere, bi puntu horietako balioen batezbestekoa izango da.

$$u_i^{n+1} = \frac{1}{2}(u_{i-1}^n + u_{i+1}^n) - \frac{k}{2h}(f(u_{i+1}^n) - f(u_{i-1}^n)). \quad (2.32)$$

Maila bateko metodo explizitua da, $n + 1$ -garren pausuko balioak kalkulatzeko soilik n -garren pausuko balioak behar ditugulako.

Metodoak (2.18) forma kontserbakorra dauka ondoko fluxu funtzioarekin:

$$F(u_i, u_{i+1}) = \frac{h}{2k}(u_i - u_{i+1}) + \frac{1}{2}(f(u_i) + f(u_{i+1})). \quad (2.33)$$

Konsistentzia baldintzak betetzen dituela frogatuko dugu:

$$\begin{aligned} F(\bar{u}, \bar{u}) &= \frac{h}{2k}(\bar{u} - \bar{u}) + \frac{1}{2}(f(\bar{u}) + f(\bar{u})) = f(\bar{u}), \\ |F(v_1, v_2) - f(\bar{u})| &= \left| \frac{h}{2k}(v_1 - v_2) + \frac{1}{2}(f(v_1) + f(v_2)) - f(\bar{u}) \right| \\ &= \left| \frac{h}{2k}(v_1 - \bar{u} - (v_2 - \bar{u})) + \frac{1}{2}(f(v_1) - f(\bar{u}) + f(v_2) - f(\bar{u})) \right| \\ &\leq \left(\frac{h}{k} + K_f \right) \max(|v_1 - \bar{u}|, |v_2 - \bar{u}|), \end{aligned}$$

non K_f f fluxu funtzioarentzat Lipschitz konstantea den. Suposatuko dugu kontserbazio legearen fluxu funtzioa Lipschitz jarraitua dela. Erraz ikusten da Burgersen ekuazioak hori betetzen duela:

$$|f(u) - f(v)| = \frac{1}{2} |u^2 - v^2| \leq \underbrace{\frac{|u+v+\epsilon|}{2}}_{K_f} |u-v|, \quad \epsilon > 0 \quad (2.34)$$

Ondorioz, metodo kontserbakorra eta trinkoa denez, Lax-Wendroff-en teoremagatik, hasierako baldintza emanda, existitzen bada $u(x, t)$ funtzio bat non $\lim u_i^n = u(x_i, t^n)$ den edozein i, n balioetarako, orduan $u(x, t)$ funtzioa Burgersen ekuazioaren soluzio ahula izango da.

Lax-Wendroff

Lax-Wendroff-en metodoa $u_t + c u_x = 0$, $c \in \mathbf{R}$ adbezio-ekuazioarentzat ondokoa da:

$$u_i^{n+1} = u_i^n - \frac{kc}{2h}(u_{i+1}^n - u_{i-1}^n) + \frac{k^2 c^2}{2h^2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n). \quad (2.35)$$

Metodoa hau kontserbazio legeetara eramateko, bi aukera proposatu zituzten Richtmyer-ek eta MacCormac-ek:

- **Richtmyer**

$$u_i^{n+1} = u_i^n - \frac{k}{h}(f(u_{i+1/2}^{n+1/2}) - f(u_{i-1/2}^{n+1/2})), \quad (2.36)$$

non

$$u_{i+1/2}^{n+1/2} = \frac{1}{2}(u_i^n + u_{i+1}^n) - \frac{k}{2h}(f(u_{i+1}^n) - f(u_i^n)).$$

- **MacCormac**

$$u_i^{n+1} = \frac{1}{2}(u_i^n + u_i^*) - \frac{k}{2h}(f(u_i^*) - f(u_{i-1}^*)), \quad (2.37)$$

non

$$u_i^* = u_i^n - \frac{k}{h}(f(u_{i+1}^n) - f(u_i^n)).$$

Bi metodoak maila bateko metodo explizituak dira, $n + 1$ -garren pausuko balioak kalkulatzeko soilik n -garren pausuko balioak behar ditugulako. Bi metodoek forma kontserbakorra dute, ondoko zenbakizko fluxu funtzioak kontuan hartuz:

$$F_1(u_i^n, u_{i+1}^n) = f(u_{i+1/2}^{n+1/2}), \quad (2.38)$$

eta

$$F_2(u_i^n, u_{i+1}^n) = \frac{1}{2}(f(u_{i+1}^n) + f(u_i^*)), \quad (2.39)$$

hurrenez hurren. Azter dezagun bien trinkotasuna.

- Richtmyer:

$$\begin{aligned}
F_1(\bar{u}, \bar{u}) &= f\left(\frac{1}{2}(\bar{u} + \bar{u}) - \frac{k}{2h}(f(\bar{u}) - f(\bar{u}))\right) = f(\bar{u}), \\
|F_1(v_1, v_2) - f(\bar{u})| &= \left|f\left(\frac{1}{2}(v_1 + v_2) - \frac{k}{2h}(f(v_2) - f(v_1))\right) - f(\bar{u})\right| \\
&\stackrel{\text{(f Lipschitz)}}{\leq} K_{1f} \left|\frac{1}{2}(v_1 + v_2) - \frac{k}{2h}(f(v_2) - f(v_1)) - \bar{u}\right| \\
&\leq K_{1f} \left(\left|\frac{1}{2}(v_1 - \bar{u} + v_2 - \bar{u})\right| + \frac{k}{2h} |f(v_2) - f(\bar{u}) - (f(v_1) - f(\bar{u}))|\right) \\
&\stackrel{\text{(f Lipschitz)}}{\leq} K_{1f} \left(\max_{i=1,2} |v_i - \bar{u}| + \frac{k}{h} K_{2f} \max_{i=1,2} |v_i - \bar{u}|\right) \\
&= K_{1f} \left(1 + \frac{k}{h} K_{2f}\right) \max_{i=1,2} |v_i - \bar{u}|
\end{aligned}$$

Bi baldintzak betetzen direnez, Richtmyer-en metodoa trinkoa da.

- MacCormac:

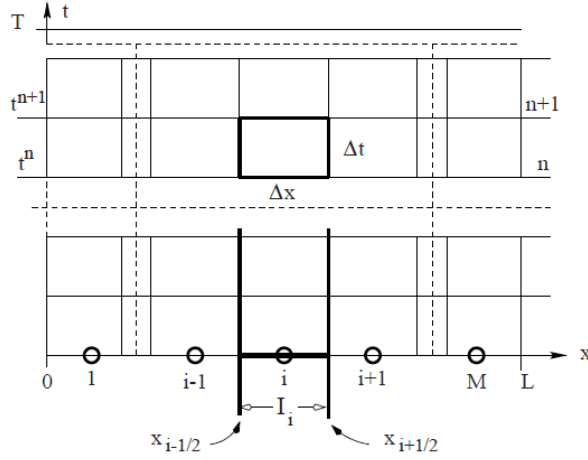
$$\begin{aligned}
F_2(\bar{u}, \bar{u}) &= \frac{1}{2} \left(f(\bar{u}) + f\left(\bar{u} - \frac{k}{h}(f(\bar{u}) - f(\bar{u}))\right) \right) = f(\bar{u}), \\
|F_2(v_1, v_2) - f(\bar{u})| &= \left| \frac{1}{2} \left(f(v_2) + f\left(v_1 - \frac{k}{h}(f(v_2) - f(v_1))\right) \right) - f(\bar{u}) \right| \\
&\leq \frac{1}{2} |f(v_2) - f(\bar{u})| + \frac{1}{2} \left| f\left(v_1 - \frac{k}{h}(f(v_2) - f(v_1))\right) - f(\bar{u}) \right| \\
&\stackrel{\text{(f Lipschitz)}}{\leq} \frac{K_{1f}}{2} |v_2 - \bar{u}| + \frac{K_{2f}}{2} \left| v_1 - \frac{k}{h}(f(v_2) - f(v_1)) - \bar{u} \right| \\
&\leq \frac{K_{1f}}{2} |v_2 - \bar{u}| + \frac{K_{2f}}{2} |v_1 - \bar{u}| + \frac{k}{2h} K_{2f} |f(v_2) - f(\bar{u}) + f(\bar{u}) - f(v_1)| \\
&\stackrel{\text{(f Lipschitz)}}{\leq} \frac{1}{2} \max_{i=1,2} K_{if} \max_{i=1,2} |v_i - \bar{u}| + \frac{k}{2h} K_{2f} K_{3f} \max_{i=1,2} |v_i - \bar{u}| \\
&\leq \frac{1}{2} \left(K + \frac{k}{h} K_{2f} K_{3f} \right) \max_{i=1,2} |v_i - \bar{u}|
\end{aligned}$$

non $K = \max_{i=1,2} K_{if}$ den. MacCormac-en eskemak bi baldintzak betetzen dituzenez, metodo trinkoa dela frogatu dugu.

2.3 Godunov-en metodoa

Godunov-en metodoa azaltzeko, lehenik eta behin $[0, L]$ espazio-tartearen diskretizazioa berdefinituko dugu. Espazio-tartea M tartetan zatituko dugu 2.1. irudian ikus daitekeen moduan,

$$x_{i-1/2} = (i-1)\Delta x \leq x \leq i\Delta x = x_{i+1/2}. \quad (2.40)$$



2.1. irudia: [7] liburuan azaltzen den bezala, $[0, L]$ tartean diskretizazioaren berdefinizioa.

Orain arteko notazioari jarraituz, problemaren zenbakizko soluzioa i espazio-posizioan eta t denbora-pausuan u_i^n -ren bidez adieraziko dugu. Demagun u_i^n ezaguna dela $i = 0, 1, \dots, M$ guztietarako, eta defini dezagun $\tilde{u}^n(x, t)$ $t^n \leq t \leq t^{n+1}$ tarterako. Hasteko, $t = t^n$ denboran $\tilde{u}^n(x, t)$ tarteka konstantea izango da,

$$\tilde{u}^n(x, t^n) = u_i^n, \quad x_{i-1/2} \leq x \leq x_{i+1/2}, \quad i = 1, \dots, M. \quad (2.41)$$

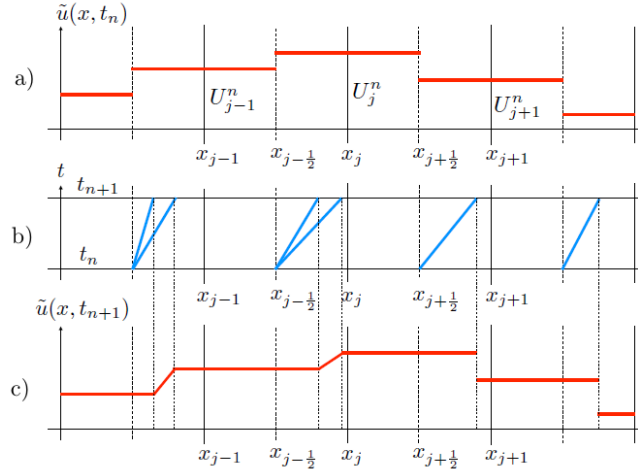
$\tilde{u}^n(x, t)$ Riemann-en problemaren bilduma baten soluzioa izango da $[t^n, t^{n+1}]$ tartean (2.2. irudia). Gelaxka bakoitzean 1. kapituluan ebatzi ditugun Riemann-en problemak ditugu: alde batetik, $u_i^n > u_{i+1}^n$ bada, travelling wave-aren bidez definituko dugu, etengunea mantenduz; eta bestetik, $u_i^n \leq u_{i+1}^n$ bada, errarefakzio-uhinak definituko dugu.

Hurrengo denbora-pausuko u_i^{n+1} zenbakizko hurbilketa $\tilde{u}^n(x, t^{n+1})$ funtzioaren batezbestekoaren bidez definituko dugu $x_{i-1/2} \leq x \leq x_{i+1/2}$ tartean,

$$u_i^{n+1} = \frac{1}{h} \int_{x_{i-1/2}}^{x_{i+1/2}} \tilde{u}^n(x, t^{n+1}) dx. \quad (2.42)$$

Metodoa programatzerako garaian batezbesteko hori kalkulatzeko kontserbazio legearen forma integrala erabiliko dugu, hau da:

$$\begin{aligned} \frac{1}{h} \int_{x_{i-1/2}}^{x_{i+1/2}} \tilde{u}^n(x, t^{n+1}) dx &= \frac{1}{h} \int_{x_{i-1/2}}^{x_{i+1/2}} \tilde{u}^n(x, t^n) dx \\ &\quad - \frac{1}{h} \left[\int_{t^n}^{t^{n+1}} f(\tilde{u}(x_{i+1/2}, t)) dt - \int_{t^n}^{t^{n+1}} f(\tilde{u}(x_{i-1/2}, t)) dt \right]. \end{aligned}$$



2.2. irudia: [3] artikuluan azaltzen da Godunov-en metodoa pausuz-pausu: a) $\tilde{u}^n(x, t^n)$ tarteka konstantea izango da $[x_{i-1/2}, x_{i+1/2}]$ tartean u_i^n balioa hartuz. b) Riemann-en problema ebazteko izango ditugun karakteristikak $[t^n, t^{n+1}]$ tartean. c) $\tilde{u}^n(x, t^{n+1})$ Riemann-en problemak ebazterakoan lortuko dugun soluzio bilduma. u_i^{n+1} lortzeko soluzio hori $[x_{i-1/2}, x_{i+1/2}]$ tartean integratuko dugu.

Edozein $i = 1, \dots, M$ hartuta, $\tilde{u}^n(x_{i+1/2}, t)$ funtzio konstateak dira $t \in [t^n, t^{n+1}]$ tartean, beraz, integraletik aterata berehala lortzen dugu iterazio bakoitzerako zenbakizko eskema. $\tilde{u}^n(x_{i+1/2}, t)$ balio konstante bakoitza u_i^n eta u_{i+1}^n balioen menpekora izango da, horien zeinuen menpekora hobeto esanda. Konstante hori idazteko $u^*(u_i^n, u_{i+1}^n)$ notazioa erabiliko dugu. Beraz, zenbakizko eskema honela geratuko zaigu:

$$u_i^{n+1} = u_i^n - \frac{k}{h}(F(u_i^n, u_{i+1}^n) - F(u_{i-1}^n, u_i^n)), \quad (2.43)$$

non

$$F(u_i^n, u_{i+1}^n) = f(u^*(u_i^n, u_{i+1}^n)).$$

Godunov-en metodoak forma kontserbakorra duela berehalakoa da, ikus daitekeen bezala. Eskema guztiz zehazteko $u^*(u_i^n, u_{i+1}^n)$ balio konstantea definitzea falta zaigu, bi egoera posible ditugularik gelaxka bakoitzean:

- $u_i^n \geq u_{i+1}^n$:

$$u^*(u_i^n, u_{i+1}^n) = \begin{cases} u_i^n, & (f(u_{i+1}^n) - f(u_i^n))/(u_{i+1}^n - u_i^n) > 0, \\ u_{i+1}^n, & (f(u_{i+1}^n) - f(u_i^n))/(u_{i+1}^n - u_i^n) \leq 0. \end{cases}$$

- $u_i^n < u_{i+1}^n$:

$$u^*(u_i^n, u_{i+1}^n) = \begin{cases} u_i^n, & u_i^n > 0, \\ u_s, & u_i^n \leq 0 \leq u_{i+1}^n, \\ u_{i+1}^n, & u_{i+1}^n, \end{cases}$$

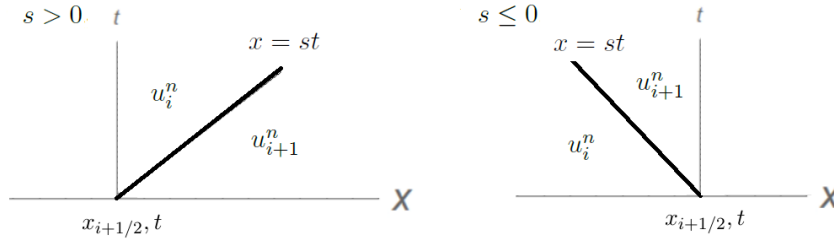
non $u_s f'(u_s) = 0$ betetzen duen balioa den.

Burgersen ekuazioa aztertu behar badugu, balio konstante hori ondokoa izango da:

- $u_i^n \geq u_{i+1}^n$:

$$u^*(u_i^n, u_{i+1}^n) = \begin{cases} u_i^n, & s > 0, \\ u_{i+1}^n, & s \leq 0, \end{cases}$$

non $s = \frac{u_i^n + u_{i+1}^n}{2}$ den. 2.3. irudian ikus daitekeen moduan, s positiboa den kasuan $\tilde{u}^n(x_{i+1/2}, t) = u_i^n$ izango da edozein $t \in [t^n, t^{n+1}]$, eta negatiboa den kasuan, berriz, u_{i+1}^n .

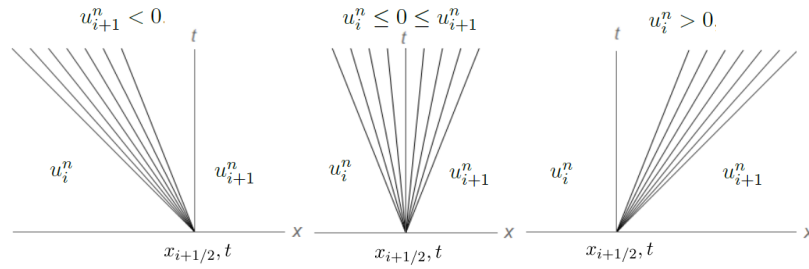


2.3. irudia: Travelling wave-aren karakteristikak s desberdinetarako: positiboa den kasua (ezker) eta negatiboa den kasua (eskuin).

- $u_i^n < u_{i+1}^n$:

$$u^*(u_i^n, u_{i+1}^n) = \begin{cases} u_i^n, & u_i^n > 0, \\ 0, & u_i^n \leq 0 \leq u_{i+1}^n, \\ u_{i+1}^n, & u_{i+1}^n < 0. \end{cases}$$

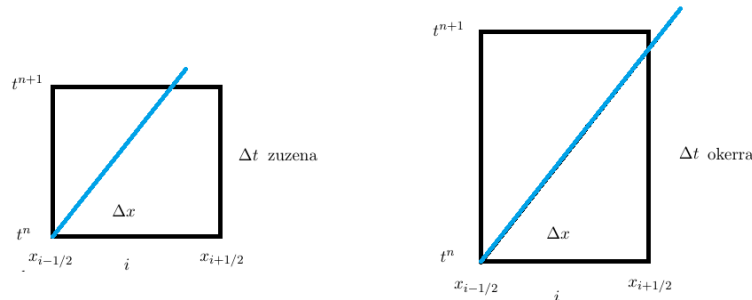
Kasu honetan, errarefakzio-uhinak izango ditugu, (1.29) adierazpenak definitzen duena (2.4. irudia): u_i^n eta u_{i+1}^n bi balioak negatiboak badira, hau da, $u_{i+1}^n < 0$, bigarren kapituluan definitu dugun (1.27) D eremua i -garren gelaxkan geldituko da eta $\tilde{u}^n(x_{i+1/2}, t) = u_{i+1}^n$ izango da (ezkerraldean). Biak positiboak badira, D eremua $i + 1$ -garren gelaxkan geldituko da eta $\tilde{u}^n(x_{i+1/2}, t) = u_i^n$ izango da (eskuinaldean). Erdiko kasua aztertzerakoan, u_s kalkulatzeko $f'(u) = u$ zerorekin berdintzean $u_s = 0$ lortzen dugu, eta beraz, $\tilde{u}^n(x_{i+1/2}, t) = 0$ lortzen dugu (erdian).



2.4. irudia: Errarefakzio uhinaren karakteristikak kasu desberdinetako: ezkerrean, $u_{i+1}^n < 0$; erdian, $u_i^n \leq 0 \leq u_{i+1}^n$; eta eskuinean, $u_i^n > 0$.

1 Denbora-tartearen aukeraketa zuzena.

Godunov-en metodoa ez da diferentzia finituetako metodo bat, iterazio bakoitzeko Δt desberdina delako. Iterazio bakoitzean Δt denbora-tarteak nahikoa txikia izan behar du Godunov-en metodoak zuzen funtzionatu dezan: $x_{i-1/2}$ puntutik abiatzen diren karakteristikak ezin dute gainditu Δx espazio-tartea (2.5. irudia).



2.5. irudia: i espazio-posizioko gelaxkako karakteristika, $[t^n, t^{n+1}]$ tartean. Δt zuzena hartu beharko dugu metodoa aplikatzeko: ezker aldean Δt okerra eta eskuinaldean Δt zuzena.

Adbekzio-ekuazio linealerako *Courant* zenbakia erabiltzen da Δt aukeratzeko:

$$\text{Courant zenbakia: } C = \frac{c \Delta t}{\Delta x}.$$

Courant zenbakiak behe- eta goi-borneak dituzenez, Δt denbora-pausua borrotatua egongo da Δx finko bakoitzarentzat:

$$0 \leq |C| \leq 1.$$

Burgersen ekuazioan, ordea, uhinaren abiadura ez da konstantea, eta beraz, irizpide hori aldatu beharko dugu. Denbora-pausu bakoitzean uhinaren

abiadura maximoa kalkulatu dugu, eta horrela ziurtatu dugu balio horretarako aukeratu dugun Δt tarteak gelaxka guztietarako balioko duela. Izan bedi S_{\max}^n uhinaren abiadura maximoa n -garren denbora-pausuan, orduan C_{CFL} Courant zenbaki maximoa,

$$C_{CFL} = \frac{S_{\max}^n \Delta t}{\Delta x} \quad (2.44)$$

izango da, non Courant-Friedrichs-Levy baldintza bete behar duen,

$$0 < C_{CFL} < 1. \quad (2.45)$$

Ondorioz, aurrena S_{\max}^n kalkulatu beharko dugu, hau da, gelaxka bakoitzeko karakteristiken abiadura maximoa.

$$S_{\max}^n = \max_{i=0, \dots, M} \{S_{i+1/2}^n\}, \text{ non, } S_{i+1/2}^n = \begin{cases} \frac{1}{2} |u_i^n + u_{i+1}^n| & (\text{Travelling wave}), \\ \max\{|u_i^n|, |u_{i+1}^n|\} & (\text{Errarefakzio-uhina}). \end{cases} \quad (2.46)$$

Behin S_{\max}^n kalkulatu daukagunean, Δx eta C_{CFL} Courant zenbakia finkatu ditugu, eta denbora-pausua zehaztu geratuko zaigu:

$$\Delta t = \frac{C_{CFL} \Delta x}{S_{\max}^n} \quad (2.47)$$

3. kapitulua

Zenbakizko simulazioak

Kapitulu honetan, *Mathematica* programaren bidez Burgersen ekuazioa bis-kositaterik gabe ebatziko dugu 2. kapituluan ikusi ditugun zenbakizko metodo kontserbakorrak erabilia. Metodo bakoitzarekin lortutako soluzio hurbildu-ak alderatuko ditugu eta ondorioak atera. Burgersen ekuazioa hasierako baldintzarekin:

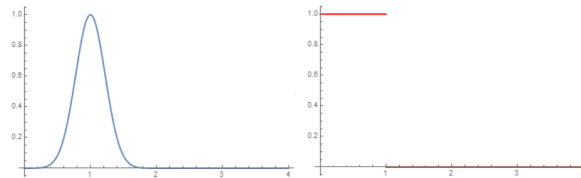
$$\begin{cases} u_t + uu_x = 0, & 0 \leq x \leq L, t \geq 0, \\ u(x, 0) = u_0(x), & 0 \leq x \leq L. \end{cases} \quad 1.11$$

Ezaguna da soluzioa hasierako baldintzaren menpe egongo dela, iterazio guztiak t^0 denbora-pausuko balioetatik eratortzen baitira. Hasierako bi baldintza aztertuko ditugu metodo bakoitzarekin: alde batetik, *kanpai* funtzioa edo funtzio *gaussianarra* $x = 1$ puntuan zentratua; eta bestetik, Riemann-en hasierako baldintza.

$$u_0(x) = \exp(-10(x-1)^2) \quad (\text{Kanpai funtzioa}), \quad (3.1)$$

eta

$$u_0(x) = \begin{cases} 1, & x < 1, \\ 0, & x \geq 1, \end{cases} \quad (\text{Riemann-en hasierako baldintza}). \quad (3.2)$$



(a) Kanpai funtzioa edo funtzio gaussianarra. (b) Riemann-en hasierako baldintza.

Lan honetan ez gara zentratuko mugaldea zer gertatzen den, hortaz, X ardatzean $L = 4$ hartuko dugu, nahikoa izango dena ikusteko soluzioak zernolako itxura hartzen duen. Tarteari dagokionez, $\Delta x = 0.04$ hartuko dugu oro har.

$x = 0$ eta $x = L$ zuzenetan ezarriko ditugun mugalde-baldintzak hasierako funtzioen arabera jarriko ditugu:

- Kanpai funtzioaren kasuan, bi mugetan balio nuluak ezarriko ditugu:

$$u(0, t) = u(L, t) = 0, \quad t \geq 0. \quad (3.3)$$

- Riemann-en problemaren kasuan, berriz,

$$u(0, t) = 1, \quad t \geq 0, \quad u(L, t) = \begin{cases} 1, & t > 6, \\ 0, & 0 \leq t \leq 6. \end{cases} \quad (3.4)$$

Hedatzen ari den etengunea $x = L$ zuzena $t = 6$ unean egitea espero baitugu, 1. kapituluaren ikusi dugun bezala.

Denborari dagokionez, gogoratu zenbakizko metodoetan muga bat ezarri behar dugula derrigor, nahi beste luzatu dezakeguna. Gure helburu nagusia etengunearekin zer gertatuko den ikustea da eta horren arabera ezarriko dugu denbora-muga. Kanpai funtzioaren kasuan $T_{buk} = 5$ ezartzearekin nahikoa izango dugu, eta Riemann-en problemaren kasuan $T_{buk} = 8$, etengunea espazio-mugara iritsi dadin.

3.1 Diferentzia finituko metodoak

Problema osatzeko parametro guztiak ezarrita, 2. kapituluaren aztertutako ditugun metodoak aplikatuko ditugu. Atal honetako metodoak diferentzia finitukoak dira, eta beraz, Δt denbora-tartea finkoa izango da: gure kasuan, $\Delta t = 0.02$ erabiliko dugu. Hasi gaitezen kanpai funtzioa aztertzen.

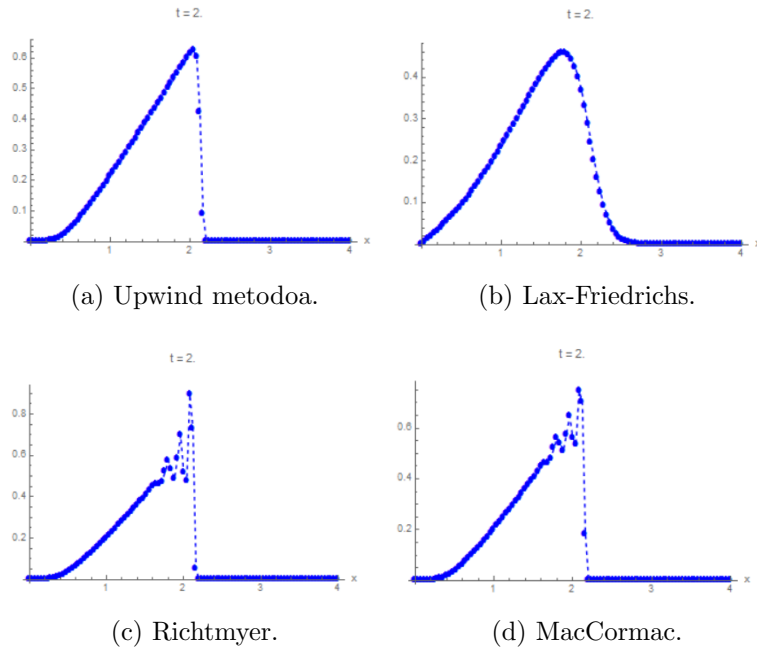
Hasteko, *Mathematica* programaren bidez karakteristikoaren metodoak ematen digun talka-unea kalkulatuko dugu:

$$T_l = 0.116582.$$

Oraingo kasua desberdina da, benetako talka-egoera bat ez delako emango, hau da, (x_i, t^n) puntu batean soluzioak ez ditu bi balio izango. Baina bai talka-egoera horretara hurbilduko dela.

(3.2. irudia) grafikan metodo bakoitzak $t = 2$ unean egiten duen hurbilketa agertzen da.

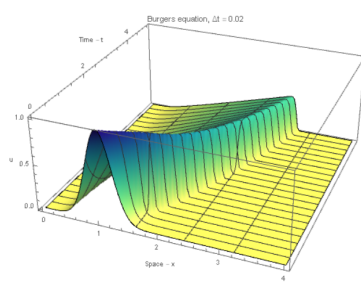
Grafika honi begiratuta ondorio batzuk atera ditzakegu: alde batetik, ikus daiteke nola Lax-Wendroff-etik eratorritako bi metodoak disipatiboak diren, eta beraz, emango dizkiguten soluzioak ez zaizkigu hain baliagarriak izango; bestalde, upwind metodoak etengunea ongi hurbiltzen du eta



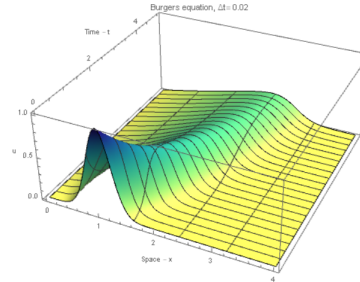
3.2. irudia: $t = 2$ unean soluzio hurbilduak.

funtzioaren altuera gutxitzen bazaio ere, mantentzen duela esan daiteke; azkenik, Lax-Friedrichs-en metodoari dagokionez, hasierako kanpai-funtzioa zabaldu egiten da, etengunea sortu beharrean.

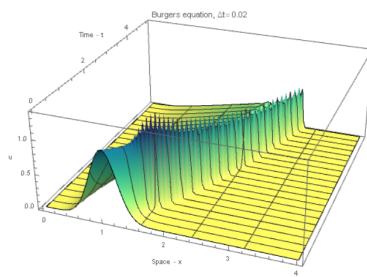
Ondorio horiek hobeto atera daitezke soluzio hurbildua $[0, 4] \times [0, 5]$ eremu osoan irudikatzen badugu (3.3. irudia): hasteko, upwind metodoak (a) eta Lax-Friedrichs-enak (b) ez dute disipasiziorik, beste biak etengunean zehar disipatiboak diren bitartean; bestetik, Lax-Friedrichs-en metodoa bakarrik ez da iristen talka-egoerara, gainontzekoetan etengunea denboran zehar hedatzen da; eta azkenik, metodo guztiek ematen duten soluzioaren balioa denbora zehar txikiagotzen eta zabaltzen doa, ondorioz metodo egonkorak direla esango dugu.



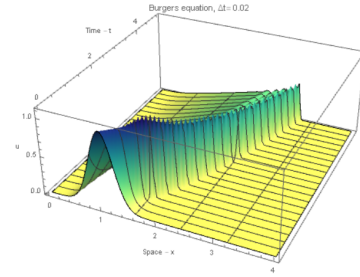
(a) Upwind metodoa.



(b) Lax-Friedrichs.



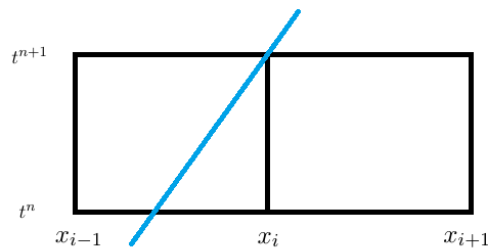
(c) Richtmyer.



(d) MacCormac.

3.3. irudia: (3.1) hasierako baldintzarako soluzio hurbildua $[0, 4] \times [0, 5]$ eremuan.

Upwind metodoak ematen digun soluzio hurbildua da entropia baldintza betetzen duen soluzio ahularen antza gehiena duena. Horren arrazoia da u_i^{n+1} puntuz-puntuko hurbilketa lortzeko u_{i-1}^n eta u_i^n balioak erabiltzen dituela, karakteristikoaren ondoan dauden biak. Gainontzeko metodoek, berriz, u_{i+1}^n balioa ere erabiltzen dute.

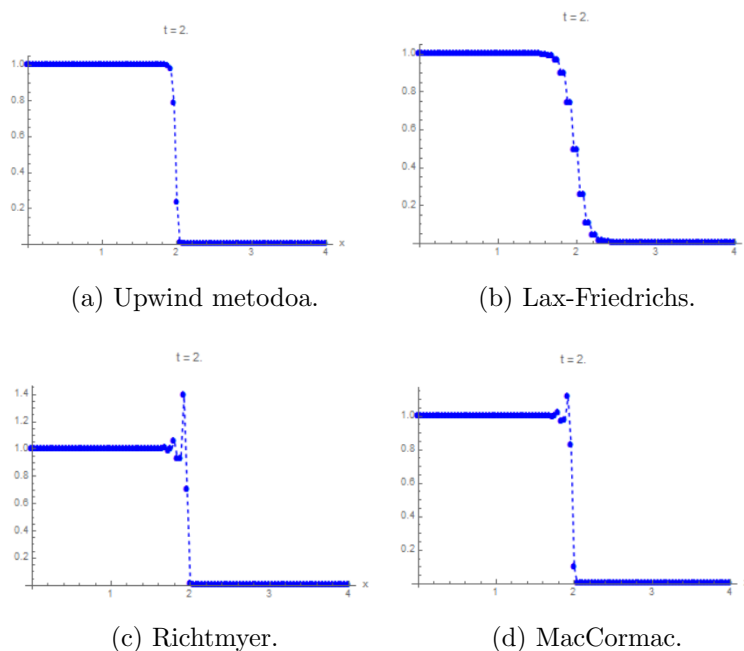


3.4. irudia: u_i^{n+1} puntutik pasatzen den karakteristika

Aztertu dezagun orain (3.2) hasierako baldintza duen Riemann-en problema. Kasu honetan, karakteristikoaren metodoaren bidez ebatziko bagenu

hasierako unetik karakteristikak mozten dira $x = 1$ ardatzean. Lehenengo kapitulutik, badakigu problemak soluzio ahul bakarra duela: (1.20) eta (1.22) adierazpenek definitzen duten travelling wave-a. Hortaz, zenbakizko hurbilketek soluzio horretara jo behar dute.

Aurreko problemarekin egin dugun bezala, $t = 2$ unean zenbakizko soluzioek hartzen dituzten balioak irudikatuko ditugu (3.5. irudia):

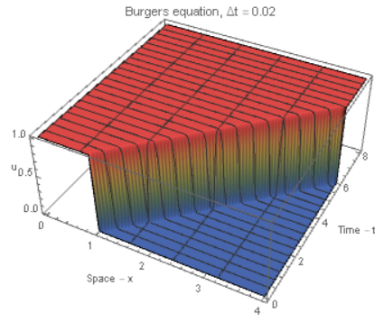


3.5. irudia: $t = 2$ unean soluzio hurbilduak.

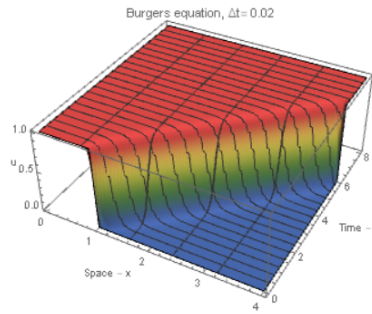
Nahiz eta soilik denbora-maila bateko balioak izan, ondorio batzuk atera ditzakegu grafikari erreparatuz. Kanpai funtzioarekin lotu ditugun ondorioak berretsi dira: Richtmyer-en eta MacCormac-en metodoak disipatiboak dira etengunean, bigarrenaren disipazioak txikiagoak izanik, ez dute etengunea egoki hurbiltzen eta problema honetan hori da bilatzen ari garena; horrez gain, upwind metodoak egoki hurbiltzen du etengunea, kontuan hartuta puntu batean ezin dituela 0-tik 1-erako balio guztiak hartu, soluzio ahulak betetzen duena; Lax-Friedrichs-en metodoari dagokionez, ez du disipaziorik jasaten baina etengunea behar baino gehiago zabaltzen da.

Lau metodoek $[0, 4] \times [0, 8]$ eremuan ematen dituzten zenbakizko soluzioak irudikatzen ditu hurrengo grafikak (3.6. irudia).

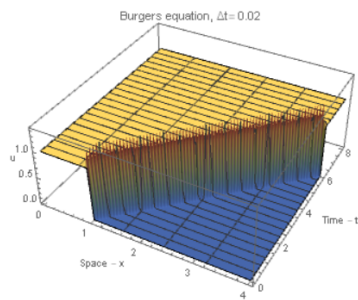
Ondorioak jada aipatuta dauzkagu: Koloreek erakusten digute Richtmyer-en metodoaren disipazioak etengunetik urruntzen direla, MacCormac-en metodoaren disipazioak baino gehiago, eta Lax-Friedrichs-en metodoak ez duela etengune garbi bat egiten baizik eta kurba bat.



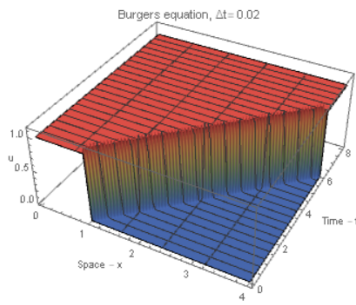
(a) Upwind metodoa.



(b) Lax-Friedrichs.



(c) Richtmyer.



(d) MacCormac.

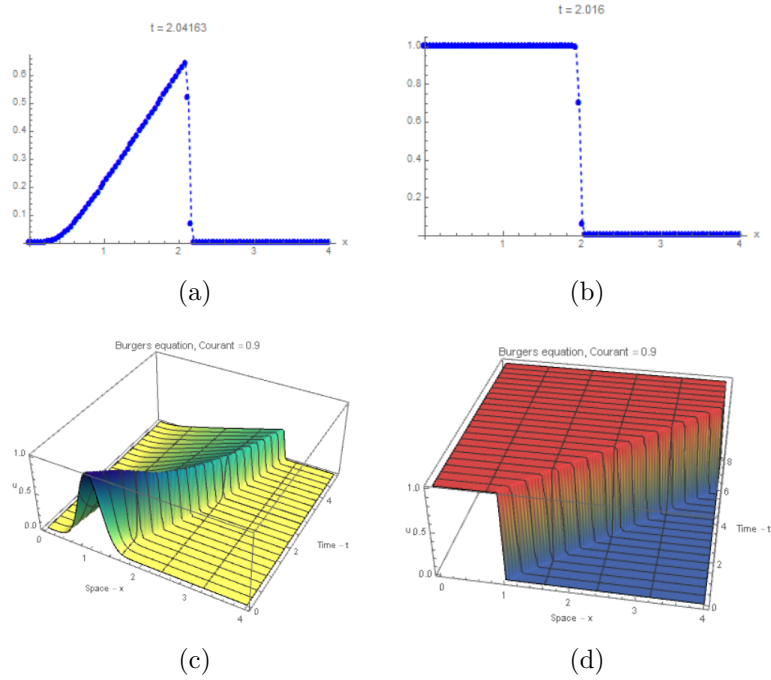
3.6. irudia: (3.2) hasierako baldintzarako soluzio hurbildua $[0, 4] \times [0, 8]$ eremuan.

3.2 Godunov-en metodoa

Lan honetan, zenbakizko metodoen ataleko helburu nagusia da Godunov-en metodoa azaltzea eta ulertzea. Erabat funtsezkoa da, beraz, zenbakizko simulazioak aztertzea; alde batetik, aurreko bi problemak ebartziko ditugu Godunov-en metodoaren bidez, Burgers-en ekuazioa biskositaterik gabe (3.1) eta (3.2) hasierako baldintzekin; eta bestetik, Courant zenbakiaren arabera Δt tarte desberdinak aztertuko ditugu.

Diferentzia finituko metodoena baino hurbilketa hobea ematen du Godunov-en metodoak (3.7. irudia); hasteko, etenguneak zuzen hurbilduta daude ((a) eta (b)), disipaziorik gabe eta gehiegi zabaldu gabe; bestetik, kanpai funtzioari dagokionez, aurreko metodoekin alderatuz altuera gutxien murrizten duena da; azkenik, ikusten dugu Δt egokiak erabiltzen dituela iterazio baktizean.

Denbora-kostuari dagokionez, Godunov-en metodoa diferentzia finituko metodoak baino azkarragoa dela esan dezakegu. Diferentzia finituko metodoak zehatzak izateko Δt denbora-pausu txikia izan behar da. Godunov-en metodoak,



3.7. irudia: Godunov-en metodoak emandako Burgersen ekuazioaren soluzio hurbilduak hasierako baldintza desberdinetarako: (3.2) kanpai funtzioa (ezker) eta (3.2) Riemann-en hasierako baldintza (eskuin). Goikaldean $t \approx 2$ unean, behekaldean eremu osoan.

aldiz, kalkulatzan ari den eremuaren arabera denbora-pausua handitu dezake, uhinaren balioa ia nulua den eremuan esaterako, eta horrek prozesua azkartzen du.

Riemann-en problemako balioak aldatzen baditugu, hau da,

$$u_0(x) = \begin{cases} 0, & x < 1, \\ 1, & x > 1, \end{cases} \quad (3.5)$$

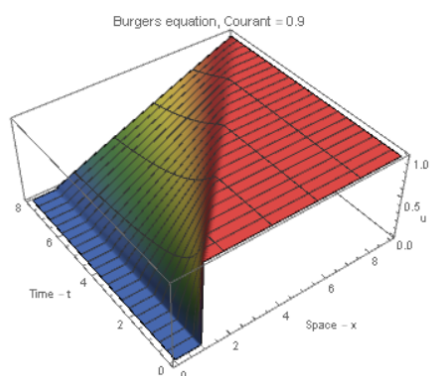
orduan errarefakzio-uhina lortzen dugula ezaguna da. Godunov-en metodoak kasu honetan ere egoki funtzionatzen du (3.8. irudia).

Courant zenbakiaren arabera.

Bukatzeko, metodoa Courant zenbakia desberdinetarako aztertuko dugu. Badakigu zenbakizko soluzioa problemaren soluzio ahulera zuzen hurbiltzeko CFL baldintza bete behar duela, hots,

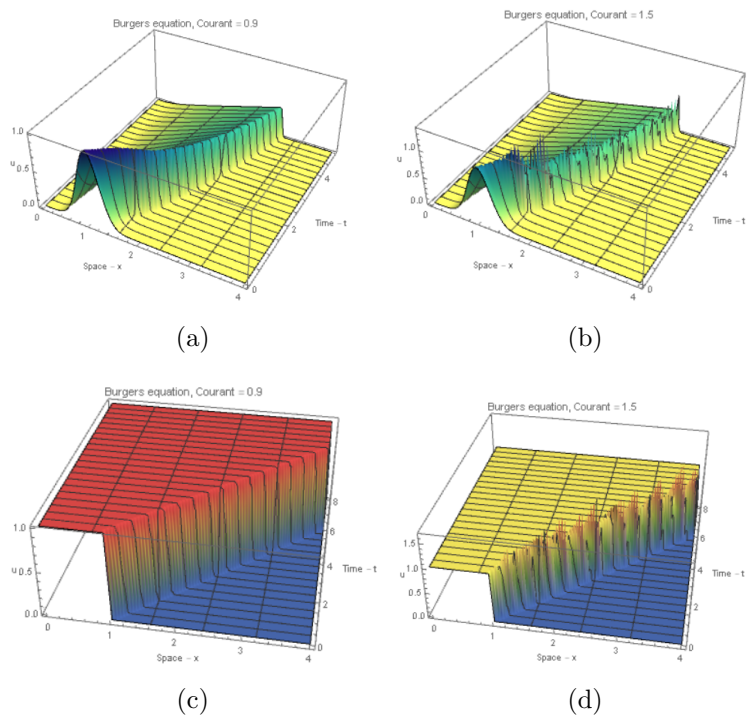
$$0 < C_{CFL} < 1.$$

Ohikoena $C_{CFL} = 0.9$ hartzea da, aurreko ariketak egiteko erabili duguna, eta ikusi dugu egokia dela. Konpara, dezagun emaitza $C_{CFL} > 1$ hartuta,



3.8. irudia: (3.5) hasierako baldintzarentzat zenbakizko soluzioa Godunov-en metodoaren bidez.

adibidez, $C_{CFL} = 1.5$. Bien arteko desberdintasuna nabaria da (3.9. irudia), Courant zenbakia borneratik kanpo hartzen badugu ((b) eta (d)), emaitza disipatiboa eta okerra lortuko dugu.



3.9. irudia: Godunov-en metodoak emandako Burgersen ekuazioaren soluzio hurbilduak hasierako baldintza desberdinetarako: (3.2) kanpai funtzioa (goian) eta (3.2) Riemann-en hasierako baldintza (behean). Ezker-raldean $C_{CFL} = 0.9$ Courant zenbaki egokia aukeratuta, eta eskuinaldean $C_{CFL} = 1.5$.

4. kapitulua

Ondorioak eta hobetzeko aukerak

Kontserbazio legeen azterketa analitikoa eta zenbakizkoa eginda, ondorio batzuk ateratzera iritsi gara. Hauek dira lan hau egiterakoan atera ditugun ondorio nagusiak:

- Soluzio ahulak beharrezkoak dira kontserbazio legeetan. Karakteristikoaren metodoaren bidez lortutako soluzioak ez digu balio, soluzioaren bakartasuna ez baita betetzen, eta beraz, existitzen ez den soluzio horrek beteko lituzkeen propietate gehienak betetzen dituen soluzio bat bilatuko dugu.
- Zenbakizko analisisian, Lax-Wendroff-en teorema funtsezkoa da metodoen konbergentzia aztertzeke. Ez dugu ahaztu behar metodoen bidez lortzen ditugun zenbakizko soluzioek problemaren soluzio ahulera jotzen dutela.
- Diferentzia finituko metodoetatik upwind metodoak bakarrik lortzen du soluzio hurbildu baliagarria. Gainera, metodo azkarra eta nahiko intuitiboa da kontserbazio legeen forma integraletik berehala eratortzen baita. Aztertu ditugun gainontzeko metodoek ez dute etengunea egoki hurbiltzen (Ritchmyer eta MacCormac) edo soluzioa gain behera etortzen da (Lax-Friedrichs).
- Godunov-en metodoa oso aproposa da Burgersen ekuazioarentzat: alde batetik, etengunea zehaztasunez definitzen duelako; eta bestetik, uhinak altuera ez duelako gehiegi murrizten.

Bestetik, lan honen jarraipena egiteko ondoko azterketa hauek egin ditzakegu:

- Alde batetik, metodoen bidez lortu ditugun soluzioak eta biskositate-dun ekuazioa $\nu \rightarrow 0$ eramanda lortzen dugun soluzioa berdinak direla frogatu dezakegu, egiazki soluzio ahulak lortu ditugula baieztatzeko.
- Bestetik, zenbakizko metodoen errorea teorikoki eta praktikoki aztertu dezakegu. Agian teorikoki mozketaren errorea eta metodoaren ordena kalkulatu zailegia da, baina puntuz-puntu errorea kalkulatu dezakegu. Metodo bakoitzaren erroreak konpara ditzakegu, begibistakoak diren ondorioak baieztatzeko.

A. eranskina

Mathematica programa

Kontserbazio legeak dituzten DPE Burgersen ekuazioa

Karakteristikoaren metodoa

```
(*Hasteko, aztertu nahi dugun hasierako baldintza definituko dugu:*)
f0[x_] := Exp[-(2 (x - 1))^2]; (*Kanpai funtzioa*)
(*f0[x_] := If[x < 0.1, 0.4, 1] Riemann-en hasierako baldintza*)

(*Karakteristikak irudikatuko ditugu (X,T) planoan,
teorian kalkulatu ditugun itxura dutenak: x=x0+f0[x0]t *)
Plot1 = ParametricPlot[Table[{x0 + f0[x0] t, t}, {x0, -.5, 3, 0.1}], {t, 0, 2},
  AxesLabel -> {Text[Style["x", Italic, 23]], Text[Style["t", Italic, 12]]},
  PlotLabel -> "Karakteristikak"]
(* D karakteristikoak definitu beharreko eremuako karakteristikak, Riemannen ii)
problemarako.
Plot2=ParametricPlot[Table[{lambda t,t},{lambda,.4,1,0.1}],{t,0,2},
  AxesLabel->{Text[Style["x",Italic,23]],Text[Style["t",Italic,12]]},PlotStyle->Red];
Show[Plot1,Plot2]*)

(*Karakteristikoaren metodoaren bidez kalkulaturako soluzioa irudikatzeko,
karakteristikoetan zehar badakigu f0[x0] balioa hartzen duela,
beraz, 3 dimentsioetara hedatuko dugu.*)
Plot3 =
  ListPlot3D[Table[Table[{x0 + f0[x0] t, t, f0[x0]}, {x0, -.5, 3, .1}], {t, 0, 2, 0.1}],
  ColorFunction -> "TemperatureMap",
  PlotStyle -> Directive[Opacity[0.9]], MeshFunctions -> {#2 &},
  Mesh -> 5, PlotRange -> All, Axes -> {True, True, True}, Boxed -> False, ImageSize -> 600,
  AxesLabel -> {Text[Style["x", Italic, 23]], Text[Style["t", Italic, 23]],
  Text[Style["u", Italic, 23]]}, PlotLabel -> "Burgersen ekuazioa"]

(*Hasierako baldintza irudikatuta*)
Plot4 = Plot[f0[x0], {x0, -.5, 3},
  Filling -> Bottom, AxesLabel -> {Text[Style["x", Italic, 23]],
  Text[Style["t", Italic, 23]], Text[Style["u", Italic, 23]]},
  PlotLabel -> "Hasierako baldintza"] (* t=2 denboran funtzioaren itxura*)
Plot5 = ListPlot[Table[{x0 + 2 f0[x0], f0[x0]}, {x0, -.5, 3, .01}],
  Filling -> Bottom, Joined -> True, AxesLabel -> {Text[Style["x", Italic, 23]],
  Text[Style["t", Italic, 23]], Text[Style["u", Italic, 23]]}, PlotLabel -> "t=2"]
```

Talka-unea

Kalkulatuko dugu olatua noiz lehertzen den:

```

f0[x_] := Exp[-(10 (x - 1))^2]; (*Hasierako unea*)
t[x1_] := f0'[x1];
T[x1_] := 1/t[x1]

Solve[T'[x] == 0, x] (*Maximoa kalkulatu dugu,
T[1/60 (60+sqrt(2))] negatiboa da, beraz, ez digu balio*)

a = N[T[1/20 (20 - sqrt(2))]] ;

ListPlot[Table[{x0 + f0[x0] a, f0[x0]}, {x0, .5, 1.5, .01}],
Joined -> True, AxesLabel -> {Text[Style["x", Italic, 23]],
Text[Style["t", Italic, 23]], Text[Style["u", Italic, 23]]},
PlotLabel -> "Talka-unea t=" <> ToString[a], PlotRange -> All, Filling -> Bottom]
(*Irudiaren bidez ikusten da talka-unea zuzen kalkulatu dugula*)

```

Errarefakzio-uhinak

```

x0 = 0;
u1 = .4;
u2 = 1; (*Riemann-en hasierako baldintzako parametroak definituko ditugu*)
f0[x_] := If[x < x0, u1, u2];
v[x_, t_] := Piecewise[{{u1, x <= u1 t}, {u2, x >= u2 t}, {x/t, x > u1 t && x < u2 t}}]
Plot1 = Plot3D[v[x, t], {x, -.5, 3},
{t, 0, 3}, ColorFunction -> "TemperatureMap", AxesLabel ->
{Text[Style["t", Italic, 23]], Text[Style["x", Italic, 23]]}, Boxed -> False]

```

Burgersen ekuazioa biskositearekin

Kasu honetan, soluzioa zehatza kalkulatu da. Beraz, eremua eta parametroak definituta ditugularik, puntu bakoitzari dagokion balioa esleituko diogu. Kalkulua pisutsua denez, eremua diskretizatu egingo dugu bilatzen ari garen zehaztasunaz. Biskosite-parametroaren arabera, bi aldiz ebatziko dugu problema, $\nu=0.1$ eta $\nu=0.01$ kasuak.

$\nu=0.1$

```

Clear["Global`*"]
x0 = 0; xL = 3;
M = 150; h = (xL - x0) / M;
t0 = 0.1; tf = 2;
k = 0.1;
F0[x_] := Exp[-(2 (x - 1))^2];
nu = 0.1;
phi0[x_] := Exp[-Integrate[F0[y] / (2 nu), {y, 0, x}]];

```

```

u1[x0_, t0_, m_, phi0_] := ulag[Function[m, phi0], x0, t0];
ulag[x0_, t0_, phi0_] :=
  Module[{x = x0, t = t0},
    NIntegrate[ ((x - m) / t) E^(- (x - m)^2 / (4 v t)) phi0[m], {m, -Infinity, Infinity}] /
    NIntegrate[phi0[m] E^(- (x - m)^2 / (4 v t)), {m, -Infinity, Infinity}]];
Data = Table[{x, 0.1, ulag[x, 0.1, phi0]}, {x, x0, xL, h}];
For[t = t0, t <= tf, t = t + k,
  Data = Join[Data, Table[{x, t, ulag[x, t, phi0]}, {x, x0, xL, h}]]]
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20},
  AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel -> "Burgers equation, v = " <> ToString[v],
  ColorFunction -> (ColorData["DarkRainbow"][#3] &)]

```

$v=0.01$

```

Clear["Global`*"]
x0 = 0; xL = 3;
M = 150; h = (xL - x0) / M;
t0 = 0.1; tf = 2;
k = 0.1;
F0[x_] := Exp[- (2 (x - 1))^2];
v = 0.01;
phi0[x_] := Exp[- Integrate[F0[y] / (2 v), {y, 0, x}]];

u1[x0_, t0_, m_, phi0_] := ulag[Function[m, phi0], x0, t0];
ulag[x0_, t0_, phi0_] :=
  Module[{x = x0, t = t0},
    NIntegrate[ ((x - m) / t) E^(- (x - m)^2 / (4 v t)) phi0[m], {m, -Infinity, Infinity}] /
    NIntegrate[phi0[m] E^(- (x - m)^2 / (4 v t)), {m, -Infinity, Infinity}]];
Data = Table[{x, 0.1, ulag[x, 0.1, phi0]}, {x, x0, xL, h}];
For[t = t0, t <= tf, t = t + k,
  Data = Join[Data, Table[{x, t, ulag[x, t, phi0]}, {x, x0, xL, h}]]]
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20},
  AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel -> "Burgers equation, v = " <> ToString[v],
  ColorFunction -> (ColorData["DarkRainbow"][#3] &)]

```

Zenbakizko metodoak

Metodo kontserbakor batzuk:

Guztietan programa berdintsua erabiliko dugu. Hasieran, problema zehazteko parametroak definituko ditugu: $[0, L]$ espazio-tartea, h espazio-pausua, $[0, t_{fin}]$ denbora-tartea, k denbora-pausua eta hasierako baldintza T_0 deituko dioguna. Gero, iterazio bakoitza definituko dugu, horretarako behar ditugun hasierako- eta mugalde-baldintzak aurrez definituz ($T[j, n]$ hasierako-baldintza eta $T[0, n]$ eta $T[M, n]$ mugalde-baldintzak). Zenbakizko eskema definituta, denbora zehatz batzuetan irudikatuko dugu soluzio hurbildua (adibidez, $t=0$, $t=2$ eta $t=5$) eta azkenik, 3 dimentsiotan eremu osoan: Horretarako, datuak taula batean sartuko ditugu, ListPlot3D aginduarekin irudikatu ahal izateko.

a) Upwind method

$$f_0(x) = \text{Exp}[-10(x-1)^2]$$

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 5;
k = .02;
tazkena = Ceiling[tbuk / k, 10];
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ", " Δx = ", N[h], " Δt = ",
k, " tazkena = ", tazkena, " Egiazko denbora-muga = ", N[tazkena k]]

T01[x_] := Exp[-10 (x - 1) ^ 2]
(*Plot[T01[X], {X, 0, 4}, PlotRange -> All, PlotStyle -> Thick] *)

T[0, n_] := 0;
T[M, n_] := 0;
T[j_, 0] := T01[x] /. x -> j h;
T[j_, n_] := T[j, n] = T[j, n - 1] - (k / (2 h)) ((T[j, n - 1]) ^ 2 - (T[j - 1, n - 1]) ^ 2);

nt0 = 0; nt1 = Round[2 / k]; nt2 = Round[5 / k];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined -> True,
PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
AxesLabel -> {"x", ""}, PlotLabel -> "t = " <> ToString[k nt0] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined -> True,
PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
AxesLabel -> {"x", ""}, PlotLabel -> "t = " <> ToString[k nt1] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined -> True,
PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
AxesLabel -> {"x", ""}, PlotLabel -> "t = " <> ToString[k nt2] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, n ≤ tazkena, n++, Data = Join[Data, Table[{h j, k n, T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
PlotLabel -> "Burgers equation, Δt = " <> ToString[k],
ColorFunction -> (ColorData[{"BlueGreenYellow", "Reverse"}][#3] &)]

```

$$f_0(x) = \text{If}[x < 1, 1, 0]$$

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 8;
k = .02;
tazkena = Ceiling[tbuk / k, 10];
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ", " Δx = ", h, " Δt = ",
k, " tazkena = ", tazkena, " Egiazko denbora-muga = ", N[tazkena k]]

```

```

T01[x_] := If[x < 1, 1, 0]
(*Plot[T01[X], {X, 0, 4}, PlotRange -> All, PlotStyle -> {Red, Thick}] *)

T[0, n_] := 1;
T[M, n_] := If[y < 6, 0, 1] /. y -> n k;
T[j_, 0] := T01[x] /. x -> j h;
T[j_, n_] := T[j, n] = T[j, n - 1] - (k / (2 h)) ((T[j, n - 1])^2 - (T[j - 1, n - 1])^2);

nt0 = 0; nt1 = Round[2 / k]; nt2 = Round[5 / k];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt0] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt1] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt2] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, n <= tazkena, n++, Data = Join[Data, Table[{h j, k n, T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel -> "Burgers equation, Δt = " <> ToString[k],
  ColorFunction -> (ColorData["DarkRainbow"][#3] &)]

```

b) Lax-Friedrichs

$$f_0(x) = \text{Exp}[-10(x-1)^2]$$

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 5;
k = .02;
tazkena = Ceiling[tbuk / k, 10];
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ", " Δx = ", h, " Δt = ",
  k, " tazkena = ", tazkena, " Egiazko denbora-muga = ", N[tazkena k]]

T01[x_] := Exp[-10 (x - 1)^2];
T[0, n_] := 0;
T[M, n_] := 0;
T[j_, 0] := T01[x] /. x -> j h;
T[j_, n_] := T[j, n] =
  (T[j + 1, n - 1] + T[j - 1, n - 1]) / 2 - (k / (2 h)) ((T[j + 1, n - 1])^2 - (T[j - 1, n - 1])^2) / 2;

```



```

nt0 = 0; nt1 = Round[2/k]; nt2 = Round[5/k];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt0] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt1] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt2] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, n ≤ tazkena, n++, Data = Join[Data, Table[{h j, k n, T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange → All, Mesh → {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel → "Burgers equation, Δt = " <> ToString[k],
  ColorFunction → (ColorData[{"BlueGreenYellow", "Reverse"}][#3] &)]

```

f0(x)=If[x<1,1,0]

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 8;
k = .02;
tazkena = Ceiling[tbuk/k, 10];
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ", " Δx = ", h, " Δt = ",
  k, " tazkena = ", tazkena, " Egiazko denbora-muga = ", N[tazkena k]]

T01[x_] := If[x < 1, 1, 0]
T[0, n_] := 1;
T[M, n_] := If[y < 6, 0, 1] /. y → n k;
T[j_, 0] := T01[x] /. x → j h;
T[j_, n_] := T[j, n] =
  (T[j + 1, n - 1] + T[j - 1, n - 1])/2 - (k / (2 h)) ((T[j + 1, n - 1])^2 - (T[j - 1, n - 1])^2)/2;

nt0 = 0; nt1 = Round[2/k]; nt2 = Round[5/k];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt0] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt1] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt2] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, n ≤ tazkena, n++, Data = Join[Data, Table[{h j, k n, T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange → All, Mesh → {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel → "Burgers equation, Δt = " <> ToString[k],
  ColorFunction → (ColorData["DarkRainbow"][#3] &)]

```

c) Ritchmyer

$$f_0(x) = \text{Exp}[-10(x-1)^2]$$

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 5;
k = .02;
tazkena = Ceiling[tbuk / k, 10];
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ", " Δx = ", h, " Δt = ",
      k, " tazkena = ", tazkena, " Egiazko denbora-muga = ", N[tazkena k]]

T01[x_] := Exp[-10 (x - 1) ^ 2];
T[0, n_] := 0;
T[M, n_] := 0;
T[j_, 0] := T01[x] /. x -> j h;
T[j_, n_] := T[j, n] = T[j, n - 1] - (k / h)
  (((T[j, n - 1] + T[j + 1, n - 1]) / 2 - (k / (2 h)) ((T[j + 1, n - 1]) ^ 2 - (T[j, n - 1]) ^ 2) / 2) ^ 2 -
   ((T[j - 1, n - 1] + T[j, n - 1]) / 2 -
    (k / (2 h)) ((T[j, n - 1]) ^ 2 - (T[j - 1, n - 1]) ^ 2) / 2) ^ 2) / 2;

nt0 = 0; nt1 = Round[2 / k]; nt2 = Round[5 / k];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt0] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt1] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt2] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, n ≤ tazkena, n++, Data = Join[Data, Table[{h j, k n, T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel -> "Burgers equation, Δt = " <> ToString[k],
  ColorFunction -> (ColorData[{"BlueGreenYellow", "Reverse"}][#3] &)]

```

$$f_0(x) = \text{If}[x < 1, 1, 0]$$

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 8;
k = .02;
tazkena = Ceiling[tbuk / k, 10];
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ", " Δx = ", h, " Δt = ",
      k, " tazkena = ", tazkena, " Egiazko denbora-muga = ", N[tazkena k]]

```

```

T01[x_] := If[x < 1, 1, 0];
T[0, n_] := 1;
T[M, n_] := If[y < 6, 0, 1] /. y -> n k;
T[j_, 0] := T01[x] /. x -> j h;
T[j_, n_] := T[j, n] = T[j, n - 1] - (k/h)
  (((T[j, n - 1] + T[j + 1, n - 1])/2 - (k/(2 h)) ((T[j + 1, n - 1])^2 - (T[j, n - 1])^2)/2)^2 -
  ((T[j - 1, n - 1] + T[j, n - 1])/2 -
  (k/(2 h)) ((T[j, n - 1])^2 - (T[j - 1, n - 1])^2)/2)^2)/2;

nt0 = 0; nt1 = Round[2/k]; nt2 = Round[5/k];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt0] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt1] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[k nt2] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, n <= tazkena, n++, Data = Join[Data, Table[{h j, k n, T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel -> "Burgers equation, Δt = " <> ToString[k],
  ColorFunction -> (ColorData["DarkRainbow"][#3] &)]

```

d) MacCormac

$$f_0(x) = \text{Exp}[-10(x-1)^2]$$

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 5;
k = .02;
tazkena = Ceiling[tbuk/k, 10];
Print["BURGERSsen EKUAZIOA: u_t + u u_xx = 0 ", " Δx = ", h, " Δt = ",
  k, " tazkena = ", tazkena, " Egiazko denbora-muga = ", N[tazkena k]]

T01[x_] := Exp[-10 (x - 1)^2];
T[0, n_] := 0;
T[M, n_] := 0;
T[j_, 0] := T01[x] /. x -> j h;
T[j_, n_] := T[j, n] = T[j, n - 1] - (k/(2 h)) ((T[j + 1, n - 1])^2 - (T[j, n - 1])^2)/2 -
  (k/(2 h)) ((T[j, n - 1] - (k/h) ((T[j + 1, n - 1])^2 - (T[j, n - 1])^2)/2)^2 -
  (T[j - 1, n - 1] - (k/h) ((T[j, n - 1])^2 - (T[j - 1, n - 1])^2)/2)^2)/2;

```

```

nt0 = 0; nt1 = Round[2/k]; nt2 = Round[5/k];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt0] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt1] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt2] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, n ≤ tazkena, n++, Data = Join[Data, Table[{h j, k n, T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange → All, Mesh → {4, 20}, AxesLabel → {"Space - x", "Time - t", "u"},
  PlotLabel → "Burgers equation, Δt = " <> ToString[k],
  ColorFunction → (ColorData[{"BlueGreenYellow", "Reverse"}][#3] &)]

```

f0(x)=If[x<1,1,0]

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 8;
k = .02;
tazkena = Ceiling[tbuk/k, 10];
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ", " Δx = ", h, " Δt = ",
  k, " tazkena = ", tazkena, " Egiazko denbora-muga = ", N[tazkena k]]

T01[x_] := If[x < 1, 1, 0];
T[0, n_] := 1;
T[M, n_] := If[y < 6, 0, 1] /. y → n k;
T[j_, 0] := T01[x] /. x → j h;
T[j_, n_] := T[j, n] = T[j, n - 1] - (k / (2 h)) ((T[j + 1, n - 1])^2 - (T[j, n - 1])^2) / 2 -
  (k / (2 h)) ((T[j, n - 1] - (k / h) ((T[j + 1, n - 1])^2 - (T[j, n - 1])^2) / 2)^2 -
  (T[j - 1, n - 1] - (k / h) ((T[j, n - 1])^2 - (T[j - 1, n - 1])^2) / 2)^2) / 2;

nt0 = 0; nt1 = Round[2/k]; nt2 = Round[5/k];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt0] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt1] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[k nt2] ]

```

```
Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, n ≤ tazkena, n++, Data = Join[Data, Table[{h j, kn, T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange → All, Mesh → {4, 20}, AxesLabel → {"Space - x", "Time - t", "u"},
  PlotLabel → "Burgers equation, Δt= " <> ToString[k],
  ColorFunction → (ColorData["DarkRainbow"][#3] &)]
```

Godunov-en metodoa

Godunov-en metodoak programa garatuago bat eskatzen du. Hasieran ezingo dugu k denbora-pausua zehaztu baina Courant zenbakia definitu beharko dugu. Ondoren, hasierako baldintzarekin batera teoriarik definitu dugun u^* funtzio konstantea kalkulatu dugu eta Burgersen ekuazioari dagokion $f(u)$ ere bai. Iterazio bakoitzean denbora-pausua definitu beharko dugu kn bektorean sartuko ditugunak. Horiek kalkulatzeko aurrez sn balio maximoak kalkulatu ditugu iterazio bakoitzean. Paraleloki joango gara $T[j,n]$ elementuak kalkulatu. Bukatzeko, diferentzia finituko metodoekin bezala une zehatz batzuetan irudikatuko dugu zenbakizko soluzioa eta baita eremu osoan ere.

Courant=0.9

$$f_0(x) = \text{Exp}[-10(x-1)^2]$$

```
Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 5;
Courant = .9;
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ",
  " Δx = ", h, " Courant = ", Courant]

T01[x_] := Exp[-10 (x - 1) ^ 2];
u1[u_, v_] := If[u ≥ v, If[(u + v) / 2 > 0, u, v], If[u > 0, u, If[v < 0, v, 0]]];
F[u_, v_] := ((u1[u, v]) ^ 2) / 2;

T[0, n_] := 0;
T[M, n_] := 0;
T[j_, 0] := T01[x] /. x → j h;
s[j_, n_] :=
  If[T[j, n] > T[j + 1, n], Abs[T[j, n] + T[j + 1, n]] / 2, Max[Abs[T[j, n]], Abs[T[j + 1, n]]]];
s0 = Max[Table[s[j, 0], {j, 0, M}]];
kn[0] := Courant * h / s0;
T[j_, n_] := T[j, n] =
  T[j, n - 1] - (kn[n - 1] / h) (F[T[j, n - 1], T[j + 1, n - 1]] - F[T[j - 1, n - 1], T[j, n - 1]]);
sn[n_] := Max[Table[s[j, n], {j, 0, M}]];
kn[n_] := kn[n] = Courant * h / sn[n];
tn[n_] := Sum[kn[i], {i, 0, n}];
```

```

nt0 = 0;
For[i = 0, tn[i] <= 2, i++, nt1 = i + 1];
For[i = 0, tn[i] <= 5, i++, nt2 = i + 1];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt0]] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt1]] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt2]] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, tn[n] <= tbuk, n++, Data = Join[Data, Table[{h j, tn[n], T[j, n]}, {j, 0, M}]]];
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel -> "Burgers equation, Courant = " <> ToString[Courant],
  ColorFunction -> (ColorData[{"BlueGreenYellow", "Reverse"}][#3] &)]

```

$f_0(x) = \text{If}[x < 1, 1, 0]$

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 8;
Courant = .9;
Print["BURGERSEN EKUAZIOA: u_t + u u_xx = 0 ",
  "   Δx = ", h, "   Courant = ", Courant]

T01[x_] := If[x < 1, 1, 0];
u1[u_, v_] := If[u ≥ v, If[(u + v) / 2 > 0, u, v], If[u > 0, u, If[v < 0, v, 0]]];
F[u_, v_] := ((u1[u, v]) ^ 2) / 2;

T[0, n_] := 1;
T[M, n_] := 0;
T[j_, 0] := T01[x] /. x -> j h;
s[j_, n_] :=
  If[T[j, n] > T[j + 1, n], Abs[T[j, n] + T[j + 1, n]] / 2, Max[Abs[T[j, n]], Abs[T[j + 1, n]]]];
s0 = Max[Table[s[j, 0], {j, 0, M}]];
kn[0] := Courant * h / s0;
T[j_, n_] := T[j, n] =
  T[j, n - 1] - (kn[n - 1] / h) (F[T[j, n - 1], T[j + 1, n - 1]] - F[T[j - 1, n - 1], T[j, n - 1]]);
sn[n_] := Max[Table[s[j, n], {j, 0, M}]];
kn[n_] := kn[n] = Courant * h / sn[n];
tn[n_] := Sum[kn[i], {i, 0, n}];

```

```

nt0 = 0;
For[i = 0, tn[i] < 2, i++, nt1 = i + 1];
For[i = 0, tn[i] < 5, i++, nt2 = i + 1];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt0]] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt1]] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt2]] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, tn[n] ≤ tbuk, n++, Data = Join[Data, Table[{h j, tn[n], T[j, n]}, {j, 0, M}]]];
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel -> "Burgers equation, Courant = " <> ToString[Courant],
  ColorFunction -> (ColorData["DarkRainbow"][#3] &)]

```

f0(x)=If[x<1,0,1]

```

Clear["Global`*"]
L = 9;
M = 100;
h = L / M;
tbuk = 8;
Courant = .9;
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ",
  " Δx = ", h, " Courant = ", Courant]

T01[x_] := If[x < 1, 0, 1];
u1[u_, v_] := If[u ≥ v, If[(u + v) / 2 > 0, u, v], If[u > 0, u, If[v < 0, v, 0]]];
F[u_, v_] := ((u1[u, v]) ^ 2) / 2;

T[0, n_] := 0;
T[M, n_] := 1;
T[j_, 0] := T01[x] /. x -> j h;
s[j_, n_] :=
  If[T[j, n] > T[j + 1, n], Abs[T[j, n] + T[j + 1, n]] / 2, Max[Abs[T[j, n]], Abs[T[j + 1, n]]]];
s0 = Max[Table[s[j, 0], {j, 0, M}]];
kn[0] := Courant * h / s0;
T[j_, n_] := T[j, n] =
  T[j, n - 1] - (kn[n - 1] / h) (F[T[j, n - 1], T[j + 1, n - 1]] - F[T[j - 1, n - 1], T[j, n - 1]]);
sn[n_] := Max[Table[s[j, n], {j, 0, M}]];
kn[n_] := kn[n] = Courant * h / sn[n];
tn[n_] := Sum[kn[i], {i, 0, n}];

```

```

nt0 = 0;
For[i = 0, tn[i] < 2, i++, nt1 = i + 1];
For[i = 0, tn[i] < 5, i++, nt2 = i + 1];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[tn[nt0]] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[tn[nt1]] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[tn[nt2]] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, tn[n] ≤ tbuk, n++, Data = Join[Data, Table[{h j, tn[n], T[j, n]}, {j, 0, M}]]];
ListPlot3D[Data, PlotRange → All, Mesh → {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel → "Burgers equation, Courant = " <> ToString[Courant],
  ColorFunction → (ColorData["DarkRainbow"][#3] &)]

```

Courant=1.5

$$f_0(x) = \text{Exp}[-10(x-1)^2]$$

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 5;
Courant = 1.5;
Print["BURGERSEN EKUAZIOA: u_t + u u_xx = 0 ",
  "   Δx = ", h, "   Courant = ", Courant]

T01[x_] := Exp[-10 (x - 1) ^ 2];
u1[u_, v_] := If[u ≥ v, If[(u + v) / 2 > 0, u, v], If[u > 0, u, If[v < 0, v, 0]]];
F[u_, v_] := ((u1[u, v]) ^ 2) / 2;

T[0, n_] := 0;
T[M, n_] := 0;
T[j_, 0] := T01[x] /. x → j h;
s[j_, n_] :=
  If[T[j, n] > T[j + 1, n], Abs[T[j, n] + T[j + 1, n]] / 2, Max[Abs[T[j, n]], Abs[T[j + 1, n]]]];
s0 = Max[Table[s[j, 0], {j, 0, M}]];
kn[0] := Courant * h / s0;
T[j_, n_] := T[j, n] =
  T[j, n - 1] - (kn[n - 1] / h) (F[T[j, n - 1], T[j + 1, n - 1]] - F[T[j - 1, n - 1], T[j, n - 1]]);
sn[n_] := Max[Table[s[j, n], {j, 0, M}]];
kn[n_] := kn[n] = Courant * h / sn[n];
tn[n_] := Sum[kn[i], {i, 0, n}];

```



```

nt0 = 0;
For[i = 0, tn[i] <= 2, i++, nt1 = i + 1];
For[i = 0, tn[i] <= 5, i++, nt2 = i + 1];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt0]] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt1]] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined -> True,
  PlotMarkers -> {Automatic, Small}, PlotStyle -> {Blue, Dashed}, PlotRange -> All,
  AxesLabel -> {"x", " "}, PlotLabel -> "t = " <> ToString[tn[nt2]] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, tn[n] <= tbuk, n++, Data = Join[Data, Table[{h j, tn[n], T[j, n]}, {j, 0, M}]]];
ListPlot3D[Data, PlotRange -> All, Mesh -> {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel -> "Burgers equation, Courant = " <> ToString[Courant],
  ColorFunction -> (ColorData[{"BlueGreenYellow", "Reverse"}][#3] &)]

```

f0(x)=If[x<1,1,0]

```

Clear["Global`*"]
L = 4;
M = 100;
h = L / M;
tbuk = 8;
Courant = 1.5;
Print["BURGERSen EKUAZIOA: u_t + u u_xx = 0 ",
  "   Δx = ", h, "   Courant = ", Courant]

T01[x_] := If[x < 1, 1, 0];
u1[u_, v_] := If[u ≥ v, If[(u + v) / 2 > 0, u, v], If[u > 0, u, If[v < 0, v, 0]]];
F[u_, v_] := ((u1[u, v]) ^ 2) / 2;

T[0, n_] := 1;
T[M, n_] := 0;
T[j_, 0] := T01[x] /. x -> j h;
s[j_, n_] :=
  If[T[j, n] > T[j + 1, n], Abs[T[j, n] + T[j + 1, n]] / 2, Max[Abs[T[j, n]], Abs[T[j + 1, n]]]];
s0 = Max[Table[s[j, 0], {j, 0, M}]];
kn[0] := Courant * h / s0;
T[j_, n_] := T[j, n] =
  T[j, n - 1] - (kn[n - 1] / h) (F[T[j, n - 1], T[j + 1, n - 1]] - F[T[j - 1, n - 1], T[j, n - 1]]);
sn[n_] := Max[Table[s[j, n], {j, 0, M}]];
kn[n_] := kn[n] = Courant * h / sn[n];
tn[n_] := Sum[kn[i], {i, 0, n}];

```

```

nt0 = 0;
For[i = 0, tn[i] < 2, i++, nt1 = i + 1];
For[i = 0, tn[i] < 5, i++, nt2 = i + 1];
ListPlot[Table[{h j, T[j, nt0]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[tn[nt0]] ]
ListPlot[Table[{h j, T[j, nt1]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[tn[nt1]] ]
ListPlot[Table[{h j, T[j, nt2]}, {j, 0, M}], Joined → True,
  PlotMarkers → {Automatic, Small}, PlotStyle → {Blue, Dashed}, PlotRange → All,
  AxesLabel → {"x", " "}, PlotLabel → "t = " <> ToString[tn[nt2]] ]

Data = Table[{h j, 0, T[j, 0]}, {j, 0, M}];
For[n = 0, tn[n] ≤ tbuk, n++, Data = Join[Data, Table[{h j, tn[n], T[j, n]}, {j, 0, M}]]]
ListPlot3D[Data, PlotRange → All, Mesh → {4, 20}, AxesLabel -> {"Space - x", "Time - t", "u"},
  PlotLabel → "Burgers equation, Courant = " <> ToString[Courant],
  ColorFunction → (ColorData["DarkRainbow"][#3] &)]

```

Bibliografia

- [1] BAO, W. (2007). The nonlinear Schrödinger equation and applications in Bose-Einstein condensation and plasma physics. *Dynamics in Models of Coarsening, Coagulation, Condensation and Quantization* (IMS Lecture Notes Series, World Scientific), 9, 141-240.
- [2] BONKILE, M. P., AWASTHI, A., LAKSHMI, C., MUKUNDAN, V., & ASWIN, C. S. (2018). A systematic literature review of Burgers' equation with recent advances. *Pramana*, (90)(6), (69).
- [3] CAMERON, M. (2011). Notes on the Burgers Equation. *University of Maryland*.
- [4] GEICKE, J. (1981). On the number of conservation laws in the sine-Gordon equation. *Physics Letters A*, 86(6-7), 333-336.
- [5] LANDAJUELA, M. (2011) Burgers equation. *BCAM Internship: Basque center for applied mathematics*.
- [6] LEVEQUE, R. J. (1992). *Numerical methods for conservation laws* (Vol. 132). Basel: Birkhäuser.
- [7] TORO, E. F. (2013). *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media
- [8] WHITHAM, G. B. (2011). *Linear and nonlinear waves* (Vol. 42). John Wiley & Sons, 2011.