



UNIVERSITAT DE BARCELONA

Treball de Fi de Grau

GRAU D'ENGINYERIA INFORMÀTICA

**Facultat de Matemàtiques i Informàtica
Universitat de Barcelona**

SIMULACIÓ MULTI-AGENT DE DISTRIBUCIÓ LOCAL D'ENERGIA ELÈCTRICA

Andreu Serra Llopart

Directora: Dra. Maite López Sánchez
Realitzat a: Departament de
Matemàtiques i Informàtica
Barcelona, 19 de juny de 2020

Índex

Resum.....	4
Resum.....	4
Resumen.....	4
Abstract.....	5
1. Introducció.....	6
1.1 Motivació.....	6
1.2 Objectius.....	7
1.3 Estructura de la memòria.....	7
2. Necessitats identificades.....	9
3. Decisió del framework de simulació multi-agent.....	10
3.1 Diferències entre netlogo i mesa.....	10
4. Disseny sistema multi agent.....	12
4.1 Entorn.....	12
4.1.1 Mapa consum.....	12
4.1.2 Mapa producció.....	13
4.2 Agents.....	14
4.2.1 Cases.....	14
4.2.2 Xarxa d'energia elèctrica.....	18
4.3 Relacions entre agents.....	18
4.3.1 Interacció casa i companyia elèctrica.....	18
4.3.2 Interacció entre dues cases.....	18
4.3.2.1 Negociació estàtica.....	19
4.3.2.2 Negociació lliure.....	21
4.4 Variables d'entrada i sortida.....	23
4.4.1 Variables d'entrada.....	23
4.4.2 Variables de sortida.....	25
5. Explicació del model i la seva implementació.....	29
5.1 Inicialització del model.....	29
5.1.1 Inicialització del mapa.....	29
5.1.2 Inicialització de les cases.....	30
5.1.2.1 Posicionament de les cases.....	30
5.1.2.2 Inicialització variables.....	32
5.1.2.3 Inicialització producció.....	33
5.2 Execució del model.....	35
5.2.1 Condicions meteorològiques.....	35
5.2.2 Calcular ingressos de les cases.....	36
5.2.3 Càlcul energia produïda.....	36
5.2.4 Compra electricitat.....	36
5.2.4.1 Negociació lliure.....	37

5.2.4.2	Negociació estàtica.....	38
5.2.4.3	Compra a la xarxa d'energia electrica.....	40
6.	Proves realitzades.....	42
6.1	Proves.....	42
6.1.1	Primera prova.....	42
6.1.2	Segona prova.....	45
6.1.3	Tercera prova.....	47
6.1.4	Quarta prova.....	49
6.1.5	Cinquena prova.....	51
6.1.6	Sisena prova.....	53
6.1.7	Setena prova.....	55
6.1.8	Vuitena prova.....	57
6.1.9	Novena prova.....	59
6.1.10	Desena prova.....	61
6.1.11	Onzena prova.....	63
6.1.12	Dotzena prova.....	65
6.2	Conclusió.....	66
7.	Conclusions.....	67
7.1	Conclusions.....	67
7.2	treball futur.....	67
8.	Bibliografia.....	68
9.	Annex.....	69
9.1	Configuració de l'entorn.....	69

Resum

Resum

L'objectiu d'aquest Treball de Fi de Grau és explorar l'impacte que tindria per a certes zones de Rubí l'ús d'energies renovables. Concretament, es vol observar l'impacte que tindria l'ús d'aquestes energies per a les famílies de Rubí en l'àmbit econòmic, es vol comprovar quin seria el millor sistema de distribució per al municipi i si hi ha zones de Rubí que es puguin autoabastir. Per a donar-li una perspectiva més realista al treball, ens vam posar en contacte amb tècnics de l'ajuntament de Rubí. Per a realitzar aquest treball s'ha dissenyat i implementat un model basat en agents que simula un mercat elèctric per a diferents zones de Rubí, en aquesta simulació tindrem dos agents diferents: les cases (simularan les diferents famílies que viuen a Rubí) i la companyia d'energia elèctrica. Aquests agents podran interactuar entre ells durant el transcurs de la simulació per tal d'abastir d'energia elèctrica totes les cases del municipi. Per a la implementació del model s'ha utilitzat tecnologies com netlogo, un framework que permet implementar models multi agents. Els resultats obtinguts amb el model són clars, l'autoproducció d'energia elèctrica ajuda a les famílies a reduir considerablement la factura de la llum, per altra banda, actualment és difícil que grups de cases puguin autoabastir-se, ja que no es pot acumular energia elèctrica i aquells dies de baixa producció s'ha d'acudir a la companyia d'energia elèctrica.

Resumen

El objetivo de este Trabajo de Fin de Grado es explorar el impacto que tendría para ciertas zonas de Rubí el uso de energías renovables. Concretamente, se quiere observar el impacto que tendría el uso de estas energías para las familias de Rubí en el ámbito económico, se quiere comprobar cuál sería el mejor sistema de distribución para el municipio y si hay zonas de Rubí que puedan autoabastecerse. Para darle una perspectiva más realista al trabajo, nos pusimos en contacto con técnicos del ayuntamiento de Rubí. Para realizar este trabajo se ha diseñado e implementado un modelo basado en agentes que simula un mercado eléctrico para diferentes zonas de Rubí, en esta simulación tendremos dos agentes diferentes: las casas (simularán las diferentes familias que viven en Rubí) y la compañía de energía eléctrica. Estos agentes podrán interactuar entre ellos durante el transcurso de la simulación con el fin de abastecer de energía eléctrica todas las casas del municipio. Para la implementación del modelo se ha utilizado tecnologías como NetLogo, un framework que permite implementar modelos multi agentes. Los resultados obtenidos con el modelo son claros, la autoproducción de energía eléctrica ayuda a las familias a reducir considerablemente la factura de la luz, por otra parte, actualmente es difícil que grupos de casas puedan autoabastecerse, ya que no se puede acumular energía eléctrica y aquellos días de baja producción se debe acudir a la compañía de energía eléctrica.

Abstract

The aim of this Final Degree Project is to explore the impact that the use of renewable energies would have for certain areas of Rubí. Concretely, we want to observe the impact that the use of these energies would have for the families of Rubí in the economic field, we want to check what would be the best distribution system for the municipality and if there are areas of Rubí that can be self-sufficient. To give a more realistic perspective to the work, we contacted with technicians from Rubí town hall. To carry out this work we have designed and implemented a model based on agents that simulates an electricity market for different areas of Rubí, in this simulation we will have two different agents: the houses (they will simulate the different families living in Rubí) and the electric power company. These agents will be able to interact with each other during the course of the simulation in order to supply electricity to all of the houses in the municipality. Technologies such as netlogo, a framework that allows multi-agent models to be implemented, have been used to implement the model. The results obtained with the model are clear, the self-production of electricity helps families to significantly reduce the electricity bill, on the other hand, it is currently difficult for groups of houses to be self-sufficient, as it is not possible accumulate electricity and those days of low production must go to the electricity company.

1. Introducció

1.1 Motivació

En un món on la tecnologia avança cada cop més i més ràpid, és molt important tenir en compte l'impacte que pot tenir en el medi ambient i a l'economia domèstica el fet d'aprofitar els recursos naturals per a la producció d'una energia elèctrica neta per a l'ús propi. Amb la possibilitat de derivar energia elèctrica seguint diferents procediments de negociació, si es donés el cas que hi hagués famílies que tinguessin sobreproducció, per tal d'aprofitar l'excedent d'energia elèctrica i no malbaratar-la.

Hi ha molts estudis que alerten de les conseqüències negatives que tindrà el canvi climàtic si no reduïm dràsticament la dependència dels combustibles fòssils i les emissions de gasos d'efecte hivernacle.

El 91% de l'energia utilitzada a Espanya prové de fonts no renovables com combustibles fòssils o energia nuclear. Espanya i 5 països més de la Unió Europea acumulen al voltant del 70% dels gasos d'efecte hivernacle del continent.

El sector energètic a causa de l'ús d'energies brutes (petroli, carbó i gas) és un dels majors contribuïdors a l'escalfament global, unes 90 empreses són responsables de quasi les dues terceres parts de les emissions mundials. A Espanya les grans elèctriques (Endesa, Iberdrola, Naturgy, EDP i Viesgo) segueixen generant bona part de la seva electricitat fent servir fonts no renovables, per això s'ha de treballar perquè aquest model insostenible canviï i s'acceleri la transició a un sistema energètic eficient i 100% renovable [1].

Per altra banda, a Espanya, el preu de la factura de la llum és un dels més alts d'Europa. Els consumidors domèstics espanyols han passat de pagar un preu per l'electricitat de 0.1124 euros per kWh en el primer semestre de 2008 a 0.1889 euros per kWh en la primera meitat de 2019. És a dir, entre 2008 i 2019 el preu de l'electricitat ha augmentat un 70%. La pujada més pronunciada es va produir en plena crisi econòmica, i és que entre 2008 i 2012 va superar els 0.17 euros per kWh [2].

Amb les energies renovables s'aconseguirà pal·liar els efectes del canvi climàtic i aconseguir una eficiència energètica que reduirà el cost de l'electricitat i que es traslladarà en mans de la ciutadania, perquè això es fes realitat, caldria anar introduint cada vegada més l'autoproducció a les nostres ciutats.

1.2 Objectius

Amb aquest treball es proposa com a objectius dissenyar i desenvolupar un model que simuli l'eficiència en l'ús d'energies renovables per a diverses famílies en una ciutat amb perfils de producció, consum i rendes diferents. Aquest model tindrà en compte que l'energia elèctrica autoproduïda podrà gastar-se per a consum propi i per a distribuir el restant a consumidors propers.

Per a donar-li una perspectiva realista al treball, ens vam posar en contacte amb treballadors de l'ajuntament de Rubí, des d'on s'està duent a terme la iniciativa Rubí Brilla [3]. L'objectiu del projecte Rubí Brilla és que la ciutat de Rubí es converteixi en un referent en l'eficiència energètica i en l'ús d'energies renovables en entorns industrials, comercials i domèstics com a factor de millora de la competitivitat i millora ambiental del territori. Des de l'ajuntament de Rubí, estan interessats a fer un estudi de quin impacte pot tenir a la ciutat de Rubí l'autoproducció i compartició d'energies renovables per a diverses cases, amb això es voldria comprovar si hi ha zones que es poguessin autoabastir i en cas que no fos així, estudiar el benefici que tindrien les famílies que fan ús de l'energia renovable en el preu de la factura de la llum.

Un altre dels objectius és estudiar com de vulnerables són cada un dels perfils de les famílies, en funció del seu consum d'energia, la seva renda i la producció que puguin tenir a les seves cases.

Un altre propòsit és estudiar les característiques (consum, preus de l'energia elèctrica...) que portarien a assolir una forma eficient de distribució per tal d'evitar que les famílies més vulnerables no caiguin en un estat de pobresa energètica.

En aquest treball ens centrarem únicament en l'energia elèctrica, no es tindran en compte altres energies com el gas.

1.3 Estructura de la memòria

Capítol 2: En primer lloc s'explica les necessitats que havia de cobrir el model i que ens van traslladar des de l'ajuntament de Rubí.

Capítol 3: Al capítol 3 es fa una petita explicació sobre que són els models basats en agents i s'explica el simulador multi agent que s'ha escollit per a implementar el model del treball.

Capítol 4: Al capítol 4 s'explica el desenvolupament que s'ha dut a terme en aquest treball, les característiques del model implementat, s'explica l'entorn del model, els agents, les interaccions entre els agents i quines variables es tenen en compte per a avaluar el funcionament del model.

Capítol 5: En aquest capítol hi trobem una explicació de la implementació del model.

Capítol 6: Al capítol 6 es realitzen diferents proves amb el model per a avaluar diferents aspectes sobre l'impacte que tindria l'ús d'energies renovables per a diferents entorns.

Capítol 7: Es valora el treball que s'ha fet, mirant si s'han assolit els objectius o no, i acabant anomenant feines futures per millorar i avançar en aquest treball.

Capítol 8: A la bibliografia es troba tots els enllaços de les pàgines que s'ha extret informació per aquest treball.

Capítol 9: a l'annex hi trobem una explicació de com configurar un entorn per tal que funcioni el model implementat.

2. Necessitats identificades

Hi ha una necessitat de nous estudis a causa de l'emergència climàtica, aquestes necessitats s'identifiquen amb l'energia, la mobilitat i les emissions d'un territori determinat.

Per a la realització d'aquest projecte he tingut l'oportunitat de parlar amb tècnics de l'ajuntament de Rubí, que estan desenvolupant el projecte Rubí Brilla [3] i que amb col·laboració amb empreses del sistema energètic, han fet possible el projecte del mapa de consum elèctric de Rubí.

Els ajuntaments són actors privilegiats, poden integrar, relacionar, explotar, extreure i finalment obrir a la societat i l'interès general coneixements molt afinats.

Des de l'ajuntament de Rubí, es volia elaborar un model predictiu per avaluar el grau d'assoliment de l'autoconsum en les diferents zones del municipi per tal d'aplicar les mesures necessàries per reduir la insostenibilitat del model actual, aportant més enllà del valor ambiental, valor econòmic i social. Valors que necessàriament hauran d'anar de la mà, ja que són inherents al canvi de model energètic, i també dissenyar sistemes de generació/acumulació distribuïts en el municipi.

3. Decisió del framework de simulació multi-agent

Un model basat en agents (MBA) és un tipus de model computacional que permet la simulació d'accions i interaccions d'individus autònoms dins d'un entorn, i permet determinar quins efectes produeixen en el conjunt del sistema. Els models simulen les operacions simultànies dels agents, en un intent de recrear i predir les accions de fenòmens complexos. Els agents actuen segons el que perceben com els seus interessos propis, com ara benefici econòmic, o estatus social, però el seu coneixement és limitat.

A l'hora d'implementar el projecte el primer repte va ser pensar amb quin framework de simulació multi-agent podia implementar el model. Durant la carrera a l'assignatura d'intel·ligència artificial distribuïda vaig implementar un model en netlogo. Investigant una mica vaig adonar-me que existeix un framework de simulació multi-agent programat en python: mesa. A partir d'aquests 2 frameworks a utilitzar he mirat quines eren les principals diferències entre ells i els avantatges i inconvenients de cadascun d'ells.

3.1 Diferències entre netlogo i mesa

Facilitat d'aprendre

Sense cap dubte netlogo és molt més fàcil d'aprendre que mesa, ja que netlogo utilitza un llenguatge de programació molt bàsic i molt fàcil d'entendre, netlogo també disposa d'una interfície que és molt fàcil d'entendre i que et permet començar a desenvolupar ràpidament entorns multi-agent i ve amb una extensa llibreria d'exemples, simulacions i documentació.

Velocitat

Mesa és un simulador multi-agent que et permet modelar entorns molt més complexos que netlogo, ja que el llenguatge de programació python és molt més potent que netlogo, tot i així, cal remarcar que netlogo és un simulador multi-agent que et permet modelar entorns força complexos amb un bon rendiment

Llibreries

En aquest punt s'ha de dir que python és un llenguatge de programació que té moltes llibreries per a qualsevol tipus d'aplicació, però netlogo té l'avantatge que també pot executar codi python i gairebé totes les llibreries de python són compatibles a netlogo

Conclusions

Vist tots aquests punts, per a la realització d'aquest treball s'utilitzarà netlogo. Netlogo és molt millor a l'hora d'introduir-se a programar models basats en agents donat la simplicitat en comparació a mesa i tenint en compte que amb netlogo també es poden aprofitar moltes de les llibreries de python, malgrat això, un cop es domina mesa, és un simulador que et permet programar models més complexos.

Característiques de netlogo:

Netlogo disposa d'una interfície fàcil d'aprendre, el llenguatge de programació que utilitza netlogo és una variant de logo, logo és un llenguatge de programació interpretat. Netlogo disposa d'una extensa llibreria d'exemples i de simulacions que et permeten aprendre ràpidament el llenguatge.

S'executa en una màquina virtual de java per tant és compatible en totes les principals plataformes (mac, windows i linux).

El principal inconvenient de netlogo és que netlogo és un sistema síncrona, això vol dir que les accions dels agents no s'executen de manera simultània sinó que primer s'executa les accions d'un agent, seguidament les d'un altre agent. Per a simular una execució asíncrona, l'ordre d'execució dels agents és aleatòria.

4. Disseny sistema multi agent

Donada l'alta complexitat de la tasca, he optat per a simplificar el problema i es proposaran els següents objectius i requisits:

1. Simularem un mercat elèctric per a diferents zones de Rubí, els objectius seran: comprovar si hi ha zones de Rubí que són potencialment autosuficients i quins són els millors mètodes d'intercanvi d'energia elèctrica entre els agents per tal d'aprofitar el màxim d'energia elèctrica possible.
2. El mercat es centrarà en l'energia elèctrica, no es tindrà en compte el consum de gas o altres energies.
3. Cada un dels ticks de netlogo simularà 1 dia. L'Energia elèctrica generada en un determinat tick no es podrà guardar per al següent. L'energia que sobri haurà de ser injectada a la xarxa, el preu de compra serà molt reduït.

4.1 Entorn

En el nostre model es deixarà escollir a l'usuari entre diferents zones de la ciutat de Rubí. Per a definir l'entorn del nostre model s'han utilitzat dos mapes extrets de dos estudis que s'han realitzat a la ciutat de Rubí [4] [5]. L'Entorn definirà el consum i la producció de les cases.

4.1.1 Mapa consum

El primer mapa que s'ha utilitzat és el mapa de consum [4], aquest mapa s'ha extret d'un estudi que pretén detectar els llocs de la ciutat on existeixen consums excessius, que poden ser causa de mals hàbits en l'ús de l'energia o males característiques dels habitatges i, al contrari, detectar consums registrats inferiors, que puguin denotar risc de pobresa. En aquest mapa podem diferenciar diferents tipus de zones que es distingeixen en diferents colors, aquests colors s'utilitzen per a determinar els consums.

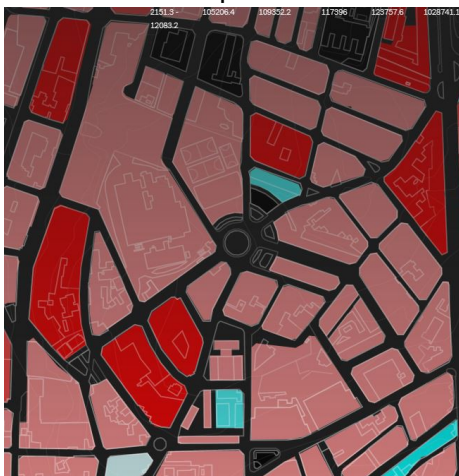


Figura 1



Figura 2

Els colors dels mapes de les figures: Figura 1 i Figura 2 poden ser:

- Color negre: aquestes seran caselles neutres, en aquestes caselles no hi haurà cases.
- Color blau: Les cases que hi hagi en aquestes cases seran cases amb un consum baix d'energia elèctrica.
- Color rosa: Les cases que hi hagi en aquestes cases seran cases amb un consum mitjà d'energia elèctrica.
- Color vermell: Les cases que hi hagi en aquestes cases seran cases amb un consum alt d'energia elèctrica.

4.1.2 Mapa producció

Un cop definit el consum de les cases, procedirem a definir la producció d'energia d'aquestes, per fer-ho s'extraurà una imatge de la mateixa zona de Rubí que en el mapa de consum d'una aplicació web [5] destinada a calcular de manera acurada els rendiments energètics i econòmics de les teulades de la zona de Rubí.



Figura 3



Figura 4

Els colors que utilitza els mapes de les figures: Figura 3 i Figura 4 són:

- Color groc: Seran cases que aprofiten tota l'energia del sol, aquestes cases seran considerades altes productores d'energia elèctrica.
- Color taronja: Aquestes cases aprofitaran el 50% de l'energia del sol.
- Color marró: Seran considerades baixes productores d'energia i aprofitaran el 20% de l'energia del sol.
- Si la casa està en un patch d'un altre color en el mapa de producció, serà considerada com a no productora. En aquest cas la seva producció serà 0.

4.2 Agents

En el nostre mercat s'han definit dos tipus d'agents, aquests agents interactuaran entre ells durant el transcurs de la simulació per tal d'incrementar les probabilitats de no finalitzar la simulació en risc de pobresa energètica.

4.2.1 Cases

Les cases seran els agents principals del mercat, el seu objectiu principal serà la seva supervivència. Tindran un consum i una producció d'energia elèctrica, uns ingressos en funció de la seva renda i un capital. Les cases que es quedin a 0 de capital seran considerades com a cases que no han estat capaces de fer front a la factura de la llum i per tant estan en risc de pobresa energètica.

L'objectiu de la implementació és comprovar com de vulnerables són cada un dels perfils de les cases. Per tal que el funcionament del mercat elèctric sigui versemblant s'han agafat els següents rangs de valors:

Taula de rang de valors

	Baix	Mig	Alt
Renda	900- 1300 €	2000- 2400 €	3000 - 4000 €
Consum	500 kWh	750 kWh	1000 kWh
Producció	20%	50%	100%

El valor de la producció fa referència al percentatge d'energia solar que és capaç d'absorbir una casa en funció de la inclinació de la coberta, el valor de producció també pot ser 0%.

El valor de renda mínima que s'ha escollit és el salari mínim interprofessional espanyol de 2019: 900 € [6].

El valor de renda mig, s'ha escollit, ja que està al voltant del salari mig espanyol: 2295 € [7].

Els valors del consum s'han extret de factures de la llum individuals de diferents habitatges.

Producció: no disposàvem de les dades i s'han posat aquests valors de referència.

S'han utilitzat les variables color, i tamany de les cases per a poder identificar visualment a la interfície els diferents perfils de cada una de les cases inicialitzades als mapes descrits anteriorment:

Per a identificar la producció d'una casa s'ha utilitzat la variable color:

- Color magenta: la casa no té producció d'energia elèctrica.
- Color blau: la casa té una baixa producció d'energia elèctrica.
- Color groc: la casa té una producció mitja d'electricitat.
- Color verd: la casa té una alta producció.

Per a identificar la variable consum s'ha utilitzat la intensitat del color:











- Colors clars: Signifiquen que les cases tenen un consum baix.
- Colors d'intensitat mitja: Les cases amb aquests colors tenen un consum mig.
- Colors foscos: Són cases amb consums alts.

S'han definit tres tamanyos diferents per a identificar els diferents tipus de renda:

- Tamany 3: Les cases amb aquest tamany són cases amb rendes baixes.
- Tamany 5: Són cases amb rendes mitjanes.
- Tamany 7: Són cases amb rendes altes.

Les cases en risc de pobresa energètica seran de color vermell i és i deixaran d'interactuar amb l'entorn i altres agents.

Llegenda dels perfils de cases

Consum	Renda	Producció	Representació al model
Baix	Baix	Zero	
mig	Baix	Zero	
Alt	Baix	Zero	
Baix	Mitja	Zero	
mig	Mitja	Zero	
Alt	Mitja	Zero	
Baix	Alta	Zero	
mig	Alta	Zero	
Alt	Alta	Zero	
Baix	Baix	Baix	
mig	Baix	Baix	

Alt	Baix	Baix	
Baix	Mitja	Baix	
mig	Mitja	Baix	
Alt	Mitja	Baix	
Baix	Alta	Baix	
mig	Alta	Baix	
Alt	Alta	Baix	
Baix	Baix	Mig	
mig	Baix	Mig	
Alt	Baix	Mig	
Baix	Mitja	Mig	
mig	Mitja	Mig	
Alt	Mitja	Mig	
Baix	Alta	Mig	
mig	Alta	Mig	
Alt	Alta	Mig	

Baix	Baix	Alt	
mig	Baix	Alt	
Alt	Baix	Alt	
Baix	Mitja	Alt	
mig	Mitja	Alt	
Alt	Mitja	Alt	
Baix	Alta	Alt	
mig	Alta	Alt	
Alt	Alta	Alt	

Producció d'energia elèctrica de les cases

Per a la producció d'energia elèctrica s'han tingut en compte diferents variables.

En primer lloc es tindrà en compte el perfil de la casa, com ja hem vist anteriorment hi haurà cases que no tindran plaques fotovoltaïques instal·lades a la coberta (cases no productores d'energia elèctrica) i hi haurà cases que sí, en el cas que la casa sigui productora d'energia elèctrica, el percentatge de producció vindrà determinat per la inclinació de la coberta de la casa.

També es tindrà en compte el fet que les condicions meteorològiques siguin favorables o no, per tant, la producció d'energia elèctrica podrà variar cada dia. La variable temps (condicions meteorològiques) podrà tenir 3 valors diferents, els dies de sol el factor temps serà 1, els dies de núvol 0.2 i els dies de pluja 0.

Finalment s'ha determinat que l'eficiència de les plaques fotovoltaïques instal·lades a totes les cases serà del 20%.

Fórmula: $E_{produccio} = \text{FactorProduccioCasa} * \text{EnergiaSolar} * \text{Eficiencia} * \text{temps}$

4.2.2 Xarxa d'energia elèctrica

La xarxa d'energia elèctrica és l'agent que s'encarrega de vendre energia elèctrica a les cases que no han pogut cobrir les seves necessitats amb l'energia que elles mateixes han produït o han comprat a altres cases.

També s'encarrega de comprar l'energia elèctrica que els hi ha sobrat a les cases que han produït més energia elèctrica de la que necessitaven i no han estat capaces de vendre-la a altres cases que la necessitessin.

4.3 Relacions entre agents

Els agents podran relacionar-se entre ells al llarg de la simulació. Es podran diferenciar dos tipus d'interaccions en funció de quins agents hi participin: entre dues cases o entre una casa i la companyia d'energia elèctrica.

4.3.1 Interacció casa i companyia elèctrica

A cada un dels ticks la companyia elèctrica proporcionarà l'electricitat necessària a les cases que no han produït suficient energia elèctrica per autoabastir-se i que no hagin estat capaces de comprar l'energia restant que necessitaven a altres cases, la factura de la llum es pagarà un cop cada 30 dies.

Aquelles cases que en determinats dies produeixin més energia de la necessària i no siguin capaces de vendre el sobrant a altres cases, li vendran a la companyia elèctrica, el preu de l'energia injectada a la xarxa elèctrica serà molt reduït i s'abonarà al mateix dia, aquest preu ve fixat pel preu de l'energia elèctrica injectada a la xarxa.

4.3.2 Interacció entre dues cases

Durant el transcurs de la simulació, si així ho escull l'usuari, les cases podran negociar entre elles, aquelles cases que han produït més energia de la que consumiran en un dia determinat, tindran l'opció de vendre el sobrant a altres cases. Aquelles cases que no hagin produït suficient energia tindran l'opció de compra'ls-hi l'energia elèctrica que els hi falta a altres cases. S'han definit dos algoritmes de negociació entre cases:

4.3.2.1 Negociació estàtica

L'algoritme de negociació estàtica està basat en el real decret de 5 d'abril de 2019 que regula l'autoconsum d'energia elèctrica [8], que estableix les següents normes per poder fer efectiu l'intercanvi d'energia elèctrica:

Es firmará un contracte on s'establirà la quantitat d'energia elèctrica que s'intercanvia, la quantitat d'energia elèctrica intercanviable serà sempre la mateixa durant el contracte, s'establirà també la durada del contracte. La compravenda d'energia només es podrà efectuar si el comprador i el productor no estan a més de 500 metres de distància i en cas que les dues cases estiguin en el mateix registre cadastral.

Per al model s'ha intentat seguir al màxim les normes esmentades anteriorment, però s'han fet algunes modificacions.

```
to setPreuElectricitatEstatic
ask cases with [death = false] [
  ifelse count my-links = 0 [
    ifelse energiaDisponible > consum [
      set venedor true
      set energiaVenda (energiaDisponible * (50 / 100))
    ]
    [
      set venedor false
      set preuCompra preuNegociacio
      set rentavilitat ((consum - energiaDisponible) * preuCompra)
    ]
  ]
  [
    if venedor = true [
      set energiaVenda (energiaDisponible * (50 / 100))
    ]
    if venedor = false[
      set preuCompra preuNegociacio
    ]
  ]
]
end
```

En primer lloc, es calcularà la producció d'electricitat de cada casa i aquelles cases que tinguin una baixa producció d'energia elèctrica i no cobreixin el seu consum i no tinguin un contracte actiu amb una altra casa realitzaran una oferta de compra d'energia que contindrà: la quantitat d'energia que compren i el preu a la qual la compren, aquest preu serà el preu de l'energia de negociació (preuNegociacio), per altra banda, aquelles cases amb sobre producció d'energia i no tinguin cap contracte actiu posaran la meitat de l'energia produïda a la venda. Aquelles cases que tinguin contractes en actiu actualitzaran per part del venedor, l'energia a la venda i per part de comprador el preu de compra d'aquesta energia.

```

to assignacioLinks [t]
  let millorCasa -1
  let millorRentavilitat 0
  ask cases with [death = false and venedor = true][

    if count my-links = 0 [
      ask other cases in-radius RadiVenta with [death = false and venedor = false] [
        if millorCasa = -1 or millorRentavilitat < rentavilitat and count my-links = 0
          [
            set millorCasa self
            set millorRentavilitat rentavilitat
          ]
        ]
      if millorCasa != -1 [
        create-conexio-from millorCasa
        ask conexio [who] of millorCasa who [
          set nTicks t
        ]
        set millorCasa -1
      ]
    ]
  ]
end

```

En segon lloc, les cases venedores d'energia elèctrica seleccionaran als seus compradors, per a l'elecció es tindrà en compte que el comprador d'energia estigui dins la distància màxima de negociació (radiVenta), aquesta distància la podrà modificar l'usuari, i la quantitat d'energia que vol comprar la casa (es vendrà l'energia elèctrica a la casa que més electricitat estigui interessada a comprar), un cop s'escull al comprador s'estableix la durada del contracte, aquesta durada serà un nombre predefinit de 4 dies, l'energia que es comercialitzarà durant la durada del contracte serà sempre el 50% de l'energia que ha produït la casa venedora.

```

to processaNegociacio

  ask conexions [

    ;; context del venedor
    ask end2 [
      set eComprada 0
      set eC 0
      ;;context del comprador
      ask other-end [
        set energiaDisponible ( energiaDisponible + [energiaVenta] of myself )
        set eComercialitzada ( eComercialitzada + energiaDisponible )
        set eC [energiaVenta] of myself
      ]

      set eComprada eC
      set capital ( capital - ( eC * preuCompra ))
      set preuComp preuCompra
    ]
    set energiaDisponible ( energiaDisponible - eComprada )
    set energiaVenta 0
    set capital ( capital + (eComprada * preuComp ) )

    set nTicks (nTicks - 1)
    if nTicks = 0
      [ die ]
    ]
  ]
end

```

Finalment s'efectuarà l'intercanvi d'energia entre les dues cases, aquest intercanvi implica que la casa venedora haurà de donar-li el 50% de la seva energia elèctrica a la casa compradora i

la casa compradora pagar-li aquesta energia en funció del preu que s'havia establert a l'inici. Un cop efectuat l'intercanvi d'energia, es comprovarà si el contracte ha finalitzat, en cas que hagi finalitzat les cases seran lliures de modificar-lo, en cas que no hagi finalitzat les cases estaran obligades a seguir amb les mateixes condicions de negociació al següent tick.

4.3.2.2 Negociació lliure

En aquest algoritme s'ha volgut simular un mercat d'energia elèctrica menys regulat, tot i que s'ha mantingut algunes de les regulacions anteriors.

```
to setPreuElectricitatLliure
  ask cases with [death = false] [

    if count my-links = 0 [

      ifelse energiaDisponible > consum [
        set venedor true
        set energiaVenta (energiaDisponible - consum)
      ]
      [
        set venedor false
        set preuCompra preuEnegociacio
        set rendibilitat ((consum - energiaDisponible) * preuCompra)
      ]
    ]
  ]
end
```

En aquest procediment de compravenda, igual que a l'anterior, un cop es calcula la producció de cada una de les cases, aquelles cases que tinguin una baixa producció d'energia elèctrica i no cobreixin el seu consum, realitzaran una oferta de compra d'energia que contindrà: l'energia total que compren i el preu al qual la compren i es definiran com a compradores d'energia elèctrica. Aquelles cases amb sobre producció es definiran com a venedores d'electricitat i posaran l'energia que els hi sobra a la venda tenint en compte que han de cobrir el seu propi consum.

```
to assignacioLinks [t]
  let millorCasa -1
  let millorRentavilitat 0
  ask cases with [death = false and venedor = true][

    if count my-links = 0 [
      ask other cases in-radius RadiVenta with [death = false and venedor = false] [
        if millorCasa = -1 or millorRentavilitat < rentavilitat and count my-links = 0
          [
            set millorCasa self
            set millorRentavilitat rentavilitat
          ]
      ]
    ]
    if millorCasa != -1 [
      create-conexio-from millorCasa
      ask conexio [who] of millorCasa who [
        set nTicks t
      ]
      set millorCasa -1
    ]
  ]
end
```

Les cases amb sobre producció d'energia elèctrica seran les que escolliran a qui venen aquesta energia elèctrica, aquesta elecció es farà tenint en compte de maximitzar la rendibilitat (aquelles que tinguin una millor relació quantitat/preu), el càlcul d'aquesta rendibilitat es farà multiplicant l'energia que la casa compra pel preu al qual compra l'electricitat. L'electricitat comercialitzada serà com a màxim l'energia sobrant del venedor tenint en compte el consum de la casa. El contracte entre les dues cases tindrà una durada d'un dia, això vol dir que a cada un dels ticks les cases venedores podran escollir un comprador diferent en funció de la rendibilitat que els hi ofereix cada un d'aquests, la restricció de la distància màxima es segueix mantenint en aquest algoritme, l'usuari serà el que escollirà la distància màxima de negociació.

```

to processaNegociacio

ask conexions [
;; context del venedor
ask end2 [
;;context del comprador
ask other-end [

    ifelse [energiaVenta] of myself < consum - energiaDisponible
    [
        set energiaDisponible ( energiaDisponible + ( [energiaVenta] of myself ))
        set eC [energiaVenta] of myself
    ]
    [
        set eC ( consum - energiaDisponible )
        set energiaDisponible consum
    ]

    ]
    set eComercialitzada ( eComercialitzada + energiaDisponible )
]
set eComercialitzada ( eComercialitzada + eC )

set eComprada eC
set capital ( capital - ( eC * preuCompra ))
set preuComp preuCompra

]
set energiaDisponible ( energiaDisponible - eComprada )
set energiaVenta 0
set capital ( capital + (eComprada * preuComp ) )
]
set nTicks (nTicks - 1)
if nTicks = 0
[ die ]
]
end

```

Finalment s'efectuarà l'intercanvi d'energia entre les dues cases, aquest intercanvi implica que la casa venedora haurà de donar-li la quantitat d'energia pactada a la casa compradora (el màxim que pot vendre la casa venedora o el màxim que pot comprar la casa compradora) i la casa compradora pagar-li aquesta energia en funció del preu que s'havia establert a l'inici. Un cop efectuat l'intercanvi d'energia, finalitzarà el contracte de compravenda d'energia elèctrica.

4.4 Variables d'entrada i sortida

4.4.1 Variables d'entrada

En el model l'usuari tindrà l'opció de modificar algunes de les variables per tal de poder simular diferents entorns, les variables d'entrada que l'usuari podrà modificar les podem dividir en quatre grups: variables referents a la inicialització del mapa, variables relacionades amb les cases, variables globals i variables destinades a regular les interaccions entre els agents.

Variables d'inicialització

Inicialitzacio



The image shows a user interface for setting initialisation variables. It consists of four distinct controls:

- A dropdown menu labeled 'inicialitzacio' with the selected option 'sequencial'.
- A slider control labeled 'numCases' with a red marker and the value '128' displayed on the right.
- A dropdown menu labeled 'zonaRubi' with the selected option 'central'.
- A dropdown menu labeled 'mapaProd' with the selected option 'estandar'.

El model té 4 variables referents a la inicialització del mapa. La variable zonaRubi et permet escollir la zona de Rubí que es vol simular i la variable mapaProd fa referència al mapa de producció que es tindrà en compte, si s'escull l'opció estàndard, el model inicialitzarà la producció de les cases en funció del mapa de producció explicat al capítol 4.1.2 Mapa producció, si s'escull una altra opció, totes les cases tindran el mateix perfil de producció, es podrà modificar per tal que les cases siguin totes altes, mitges o baixes productores.

La variable inicialització fa referència al posicionament que tindran les cases en el mapa, els valors poden ser seqüencial o aleatori, en el posicionament seqüencial les cases estaran posicionades de forma ordenada, en el posicionament aleatori les cases es posicionaran de manera aleatòria, només es podran inicialitzar cases en les zones que siguin de color vermell, rosa o blau.

La variable numCases indica el nombre de cases que s'inicialitzaran a la simulació, el límit del nombre de cases ve marcat pel nombre màxim de cases que hi caben al mapa.

Variables de les cases

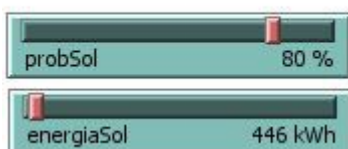
Variables cases



En el grup de variables que caracteritzen els agents casa hi trobem: la variable `initCapital` indica els diners estalviats que tindran les cases en el moment d'inicialitzar-se. Les últimes tres variables: `rendaBaixa`, `rendaMitja` i `rendaAlta` defineixen els ingressos que tindrà cada casa en funció dels seus perfils de renda, els valors que poden tenir aquestes tres variables està explicat a la Taula de rang de valors. En aquesta simulació s'ha tingut en compte només el consum energètic elèctric, per tant, a l'hora d'ingressar els diners de la renda, només es tindrà en compte el percentatge de la renda destinat a cobrir aquestes necessitats, la `rendaDestinadaElectricitat` indica el percentatge de renda que les cases destinaran a cobrir les seves necessitats energètiques. Des de l'ajuntament de Rubí ens van dir que de normal es destina el 4% de la renda a pagar les despeses de llum, al model aquesta variable s'ha limitat al 10%.

Variables globals

Variables globals



El model disposa de dos variables globals que serviran per a modificar les condicions meteorològiques, la variable `probSol` indica la probabilitat que faci sol i la variable `energiaSol` l'energia solar que arriba a la superfície de la terra un dia de sol.

Els valors d'aquestes variables poden variar bastant en funció de l'època de l'any, com es pot apreciar al següent enllaç [9] podem veure que la probabilitat que faci sol a Espanya a certes èpoques de l'any pot arribar al 76% i en d'altres al 40%. El valor de la variable de l'energia del sol variarà també en funció de l'època de l'any que volem simular i els metres quadrats de plaques fotovoltaïques que tindran instal·lades les cases, com es pot veure als següents enllaços [9] [10], a Espanya, arriba uns entre 7 kWh i 2 kWh d'energia solar a la superfície de la terra per metre quadrat depenent de l'època de l'any.

Variables d'interacció entre agents

Variables negociació

TipusNegociacio
Estatica

RadiVenta 100

preuExarxa 0.15 €

preuNegociacio 0.00 €

preuInjectada 0.20 €

Les variables de negociació es centraran en regular la negociació entre els agents, en primer lloc tenim el desplegable tipusNegociacio, aquest desplegable ens permetrà escollir el tipus de negociació que volem entre les cases: Estàtica, lliure o que no existeixi negociació d'energia elèctrica, explicat al capítol 4.3.2 Interacció entre dues cases. RadiVenta és la variable que regula la distància màxima a la qual les cases poden comercialitzar.

PreuExarxa serà el preu de l'energia que les cases compren a la xarxa d'energia elèctrica, preuNegociacio és el preu que les cases hauran de vendre la seva energia a altres cases i el preuInjectada serà el preu pel qual la xarxa d'energia elèctrica els hi comprarà l'energia sobrant a les cases.

4.4.2 Variables de sortida

Per a monitorar el funcionament del mercat controlarem les següents variables:

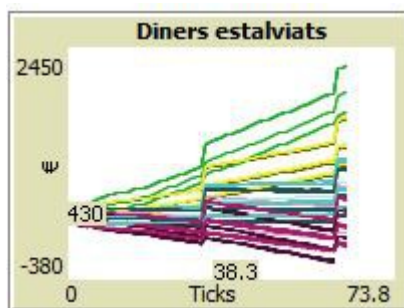
Inicialització

Cases amb renda baixa 0	Cases amb baix consum 0	Cases amb produccio 0 0
Cases amb renda mitja 0	Cases amb consum mig 0	Cases amb produccio baixa 0
Cases amb renda alta 0	Cases amb consum alt 0	Cases amb produccio mitja 0
		Cases amb produccio alta 0

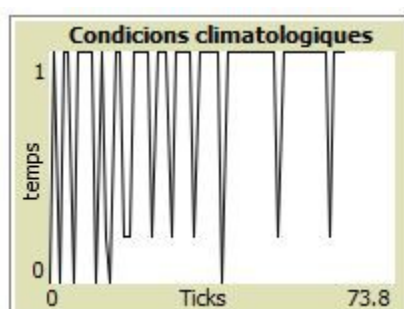
Cases amb diners 0

Cases amb diners 0 0		
Cases amb renda baixa 0	Cases amb baix consum 0	Cases amb produccio 0 0
Cases amb renda mitja 0	Cases amb consum mig 0	Cases amb produccio baixa 0
Cases amb renda alta 0	Cases amb consum alt 0	Cases amb produccio mitja 0
		Cases amb produccio alta 0

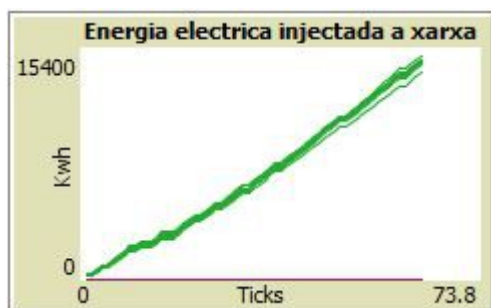
Primer de tot, disposem de dos grups de monitors, el grup de monitors que es mostren a la part esquerra de la figura ens mostrarà el nombre total de cases inicialitzades en cada un dels perfils, el grup de monitors que es mostren a la part dreta de la figura es centra només amb les cases que han arribat a 0 de capital, ens mostrarà de cada un dels perfils quantes cases han quedat en risc de pobresa energètica.



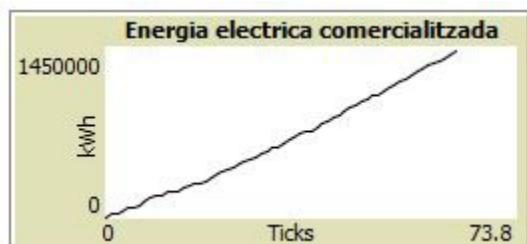
Aquesta gràfica ens mostrarà els diners estalviats i per tant acumulats al llarg dels dies de cada una de les cases, en aquesta gràfica cada casa estarà representada amb el seu color.



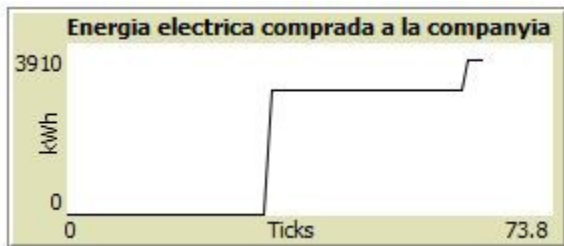
Aquesta gràfica ens mostra les condicions climatològiques a cada un dels ticks, si el valor és 1, vol dir que fa sol, són ticks d'alta producció d'energia, si el valor és 0.2, significa que fa núvol i per tant la producció serà baixa i si el valor és 0, la producció d'energia elèctrica serà 0.



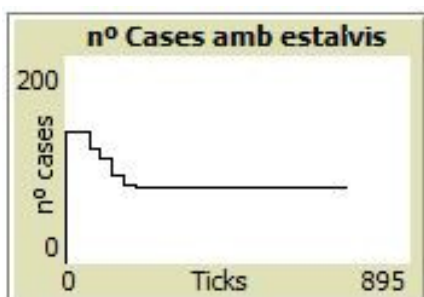
Aquesta gràfica ens mostra la quantitat d'energia elèctrica que ha injectat cada casa a la xarxa.



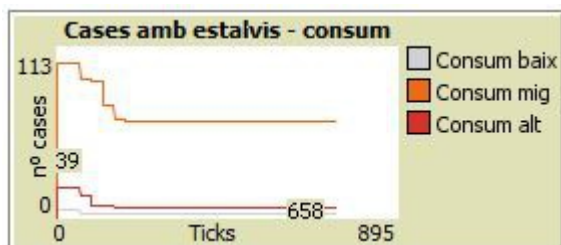
L'energia comercialitzada és l'energia total que han comercialitzat (s'ha realitzat una transacció de compravenda entre dues cases) entre totes les cases.



Amb aquesta gràfica podem veure la quantitat global d'energia elèctrica que les cases han comprat a la xarxa d'energia elèctrica al llarg de la simulació.



Mostrem també el nombre de cases que no estan en risc de pobresa energètica





Finalment, les últimes tres gràfiques mostren el nombre de cases que no estan en risc de pobresa energètica en funció del consum que tenen, la producció i la renda respectivament.

5. Explicació del model i la seva implementació

En primer lloc, cal inicialitzar el model, la inicialització del model implica diferents passos, en primer lloc inicialitzem l'entorn: explicat al capítol 4.1 Entorn. En segon lloc inicialitzarem les cases i en funció del posicionament que tinguin cada una de les cases definirem els diferents perfils d'aquestes, la renda, el consum i la producció, els agents cases estan explicats al capítol 4.2.1 Cases.

En segon lloc tindrem l'execució del model, l'execució del model consta de diferents etapes. En primer lloc, es calcularà el temps (condicions meteorològiques), les condicions meteorològiques afectaran la producció d'energia elèctrica que tindran les cases en un dia determinat. El segon pas serà calcular els ingressos de cada una de les cases, aquests ingressos es calcularan en funció de la renda de la casa. El tercer pas serà la generació d'energia elèctrica, en aquest pas es calcularà l'energia elèctrica que ha generat cada una de les cases. Finalment, a l'últim pas s'efectuarà la compra d'energia elèctrica, en aquest últim pas en primer lloc les cases intentaran comercialitzar entre elles seguint els procediments descrits al capítol: 4.3.2 Interacció entre dues cases, un cop finalitzada la compravenda d'energia elèctrica entre les cases, aquelles cases que encara no tinguin cobert el seu consum d'energia, interactuaran amb la xarxa d'energia elèctrica per a comprar el restant de l'energia que necessitin: 4.3.1 Interacció casa i companyia elèctrica.

5.1 Inicialització del model

5.1.1 Inicialització del mapa

```
to initMap
  py:setup py:python
  py:run "from skimage import io"
  py:run "import time"
  py:run "from skimage.transform import resize"

  py:set "x" zonaRubi
  py:run "x=x+'C.jpg'"

  py:run "mapF = io.imread(x)"
  py:run "mapF = resize(mapF, (400, 400))"
  py:run "io.imsave('mapFC.jpg', mapF)"

  import-drawing "mapFC.jpg"
  import-pcolors "mapFC.jpg"

  ask patches with [(pcolor > 132 and pcolor < 137) or (pcolor > 16 and pcolor < 19)] [ set pcolor pink]
  ask patches with [pcolor > 12 and pcolor < 16 ] [ set pcolor red]
  ask patches with [pcolor > 92 and pcolor < 97 ] [ set pcolor cyan]
  ask patches with [pcolor > 82 and pcolor < 87 ] [ set pcolor cyan]
  ask patches with [pcolor > 102 and pcolor < 107 ] [ set pcolor cyan]
end
```

En primer lloc importem la llibreria `skimage` de `python`, aquesta llibreria ens permetrà redimensionar la imatge base per tal que les mides de la imatge es corresponguin amb el tamany de l'output de `netlogo`.

Utilitzarem les funcions de `netlogo`:

- `Import-drawing` : Ens permet dibuixar la imatge tal qual està guardada en l'output de

netlogo.

- *Import-pcolor* : Ens permet dibuixar la imatge en l'output de netlogo utilitzant els colors de netlogo.

Un cop dibuixat el mapa amb els colors de netlogo transformem tots aquells patches que tenen un color dintre l'escala de vermells a vermell, tots aquells colors que estan dintre l'escala de blaus a cian i tots aquells que estan dintre l'escala de roses a rosa.

5.1.2 Inicialització de les cases

5.1.2.1 Posicionament de les cases

```
to initCases
  if inicialitzacio = "aleatoria" [
    create-cases numCases [
      posicioRandom
      initVars
    ]
  ]

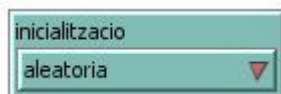
  if inicialitzacio = "seqüencial" [
    while [ posY < 40 ] [
      create-cases 1 [
        setxycasa
        posicioSeqüencial
        initVars
      ]
    ]
    let nCases count cases - numCases
    if nCases > 0
      [ ask n-of nCases cases [ die ] ]
  ]
end
```

En el nostre model de netlogo disposem de dos algorismes per tal d'inicialitzar les cases: la inicialització aleatòria i la seqüencial. En el posicionament seqüencial es vol reflectir una distribució més realista en el territori, mentre que en el posicionament aleatori, es vol simular un entorn més equilibrat pel que fa a perfils de consum, ja que en aquest posicionament tindrem un nombre semblant de cases a cada un dels diferents consums.

Amb la variable numCases limitarem el nombre màxim de cases que es poden inicialitzar al nostre entorn.



La selecció de l'algorisme d'inicialització es fa a través del següent desplegable:



Posicionament aleatori:

```
to posicioRandom
  let x random 3
  let tipus item x tipusConsum
  if tipus = "altConsum" [
    let patch_ one-of (patches with [pcolor = red])
    setxy [pxcor] of patch_ [pycor] of patch_
    set tipus_consum tipus
  ]
  if tipus = "consumMig" [
    let patch_ one-of (patches with [pcolor = pink])
    setxy [pxcor] of patch_ [pycor] of patch_
    set tipus_consum tipus
  ]
  if tipus = "baixConsum" [
    let patch_ one-of (patches with [pcolor = cyan])
    setxy [pxcor] of patch_ [pycor] of patch_
    set tipus_consum tipus
  ]
end
```

Primer de tot li assignem un consum a la casa, l'assignació es fa aleatòriament i tots tres consums tenen la mateixa probabilitat. Un cop se li ha assignat el consum, s'escollirà un patch aleatori del color corresponent al consum. Si el consum és baix, el patch que se li assignarà a la casa serà de color cian, si el consum és mig, se li assignarà un patch de color rosa i si el consum és alt de color vermell.

Posicionament seqüencial

```
to posicioSeqüencial
  let x 0
  while [ ( pcolor != red ) and ( pcolor != cyan ) and ( pcolor != pink ) ] [
    setxy posX posY
    set x ( x + 1 )
    set posX ( posX + 5 )
    if posX > 40 [
      set posX -40
      set posY ( posY + 5 )
    ]
    if posY > 40 [
      stop
    ]
  ]
  if pcolor = red
  [ set tipus_consum "altConsum" ]
  if pcolor = pink
  [ set tipus_consum "consumMig" ]
  if pcolor = cyan
  [ set tipus_consum "baixConsum" ]
end
```

A l'algoritme de posicionament seqüencial, es busca posicionar el màxim nombre de cases possible, aquestes cases hauran d'estar a una distància mínima de 5 patchs i hauran de posicionar-se en un patch de color vermell, cian o rosa. Un cop posicionada la casa se li assignarà un consum en funció del color del patch on estigui posicionada.

5.1.2.2 Inicialització variables

```
to initVars

  calculConsum
  set size 4
  set shape "house"

  set capital initCapital
  set energiaInjectada 0
  set energiaDisponible 0
  set consumMes 0

  let x random 3
  set tipus_renta item x tipusRenta
  set death false

  set produccio 0

end
```

Un cop posicionades les cases es cridarà a la funció `initVars`, aquesta funció s'encarrega d'inicialitzar totes les variables de l'agent casa:

- **consum:**

```
to calculConsum
  ask cases [
    if tipus_consum = "baixConsum" [
      set consum 500 / 30
    ]
    if tipus_consum = "consumMig" [
      set consum 750 / 30
    ]
    if tipus_consum = "altConsum" [
      set consum 1000 / 30
    ]
  ]
end
```

Per a assignar el valor de la variable `consum` cridem a la funció `calculConsum`, aquesta funció mirarà quin tipus de consum té la casa i li assignarà un valor en funció d'aquest `tipus_consum`, els valors són els següents:

	Baix	Mig	Alt
Consum del tot el mes	500 kWh	750 kWh	1000 kWh

- **shape:** s'assignarà una forma "house" a l'agent casa.
- **Capital:** Aquesta variable determinarà els diners inicials disponibles destinats a la compra d'energia elèctrica, el valor inicial d'aquesta variable vindrà definit per al slide:

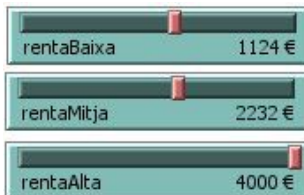


`InitCapital` és una de les variables d'entrada del model, explicada al capítol: `Variabes de les cases`

- **energiaInjectada:** és una variable per controlar l'energia total que les cases injecten a la xarxa, és l'energia elèctrica que les cases productores no han pogut aprofitar. El preu

de l'energia injectada ve donat per la variable `elinjectadaXarxa`, explicada al capítol: Variables d'interacció entre agents

- **energiaDisponible:** És la variable que utilitzarem per a guardar l'energia que s'ha produït en un tick, aquesta energia podrà ser utilitzada per a comercialitzar i per al consum propi.
- **consumMes:** Aquesta variable la utilitzarem per guardar la quantitat d'energia que cada una de les cases ha comprat a la xarxa a cada un dels ticks, passats 30 ticks, s'utilitzarà per a calcular el preu de la factura de l'electricitat.
- **tipus_renda:** Aquesta variable defineix el tipus de renda que té cada una de les cases, els tipus de renda són: renda alta, renda mitja i renda baixa. Els valors de cada una de les rendes ve definit pels següents slides. Aquestes són variables d'entrada del model i formen part del grup de Variables de les cases:



- **death:** És una variable que s'utilitza per controlar aquelles cases que no han estat solvents a l'hora de pagar la factura de la llum, si una casa arriba a 0 de diners (`capital = 0`), aquesta variable serà true i la casa deixarà d'interactuar amb l'entorn.
- **produccio:** utilitzem aquesta variable per saber l'energia elèctrica que produeix cada casa en un tick.

5.1.2.3 Inicialització producció

```
to initProd
  py:run "mapF = io.imread('mapProd.jpg')"
  py:run "mapF = resize(mapF, (1000, 1000))"
  py:run "io.imsave('mapF.jpg', mapF)"

  import-pcolors "mapF.jpg"

  ask patches with [ (pcolor > 11 and pcolor < 18) or (pcolor > 20 and pcolor < 22)] [ set pcolor red]
  ask patches with [ pcolor > 42 and pcolor < 48 ] [ set pcolor yellow]
  ask patches with [pcolor > 22 and pcolor < 28 ] [ set pcolor orange]

end

to setProd
  ask cases[
    if pcolor = red [
      set produccio 0.2
    ]
    if pcolor = orange [
      set produccio 0.5
    ]
    if pcolor = yellow [
      set produccio 1
    ]
  ]
end
```

Per a saber com d'eficient serà la producció de cada casa, s'ha agafat una fotografia de la

mateixa zona de Rubí que en el mapa de consum d'una l'aplicació web destinada a calcular els rendiments energètics i econòmics de les teulades de la zona de Rubí [2], s'ha importat la imatge amb la funció import-pcolors, tots els colors que tenen una tonalitat vermella, s'han passat a vermell, els colors amb tonalitat groga a groc i els colors amb tonalitat taronja a taronja. Un cop s'ha passat la imatge del mapa a l'output de netlogo (no visible als usuaris però els agents hi poden interactuar) s'ha mirat el color del patch on estava cada una de les cases i s'ha assignat un factor de producció, les cases que estan sobre d'un patch vermell podran aprofitar un màxim del 20% de l'energia del sol, les cases que estan sobre un patch taronja un 50% i les cases amb un patch groc el 100% de l'energia, les cases que no estan sobre un dels tres colors esmentats anteriorment seran considerades com a cases no productores i per tant la variable producció serà 0.

Identificació visual de les cases

```
to colorCases
  ask cases [
    if produccio = 0
      [ set color 128 ]
    if produccio = 0.2
      [ set color 88 ]
    if produccio = 0.5
      [ set color 48 ]
    if produccio = 1
      [ set color 67 ]

    if tipus_consum = "consumMig"
      [ set color ( color - 3 ) ]
    if tipus_consum = "altConsum"
      [ set color ( color - 6 ) ]

    if tipus_renta = "altaRenta"
      [ set size 7 ]
    if tipus_renta = "rentaMitja"
      [ set size 5 ]
    if tipus_renta = "rentaBaixa"
      [ set size 3 ]
  ]
end
```

Les variables colors i tamany de les cases s'utilitzaran per a identificar visualment els diferents perfils de les cases. Assignarem un color en funció de la producció, la foscor d'aquest color determinarà el consum de cada casa i el tamany la renda. Els valors són els següents:

Totes aquelles cases que tinguin un color dintre de l'escala de liles, tindrà producció 0, les cases amb colors blaus seran baixes productores d'energia elèctrica, les cases amb producció mitja seran de color groc i les que tinguin una producció alta seran de color verd.

La intensitat del color determinarà el consum de la casa, com més fosca sigui la casa més alt serà el seu consum. Finalment utilitzarem el tamany de la casa per determinar la renda, renda baixa tindrà un tamany 3, renda mitjana 5 i renda alta 7. Per a més informació consultar

Llegenda dels perfils de cases

5.2 Execució del model

```
to go

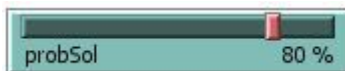
  calcularTemps
  generarIngresos
  generarEnergia
  compraElectricitat
  tick
  py:set "x" tempsTick
  py:run "time.sleep(x)"
end
```

5.2.1 Condicions meteorològiques

```
to calcularTemps
  let x random 100

  let y ((100 - probSol) / 2)

  if x < (probSol + (y * 2)) [
    set temps 0
  ]
  if x < (probSol + y) [
    set temps 0.2
  ]
  if x < probSol [
    set temps 1
  ]
end
```



En primer lloc calcularem les condicions meteorològiques, la variable temps fa referència a si les condicions meteorològiques són favorables o no per a la producció d'energia. Per calcular el temps s'utilitzarà la variable probSol, el valor d'aquesta variable ve donat pel slide probSol i fa referència a la probabilitat que faci sol. La variable temps podrà tenir 3 valors diferents, 1 si les condicions meteorològiques són bones per a la producció, 0.2 si aquestes condicions no són les més indicades i 0 si durant aquell tick no es podrà produir energia. La probabilitat que la variable temps sigui 1 vindrà donat pel valor de la variable probSol, la probabilitat que la variable sigui 0.2 o 0 es calcularà de la següent manera: $p(0.2) = p(0) = (1 - \text{probSol}) / 2$.

5.2.2 Calcular ingresos de les cases

```
to generarIngresos
  if ticks mod 30 = 0 or ticks = 0 [
    ask cases with [death = false] [
      if tipus_renta = "rentaBaixa" [
        set renta rentaBaixa
      ]
      if tipus_renta = "rentaMitja" [
        set renta rentaMitja
      ]
      if tipus_renta = "altaRenta" [
        set renta rentaAlta
      ]
      set capital (capital + (renta * ( rentaDestinadaElectricitat / 100 ) ) )
    ]
  ]
end
```



Cada 30 ticks (dies), les cases, rebran uns ingressos en funció de la seva renda. Ja que en aquest model ens fixem en el consum d'energia elèctrica, a la variable capital se li sumarà la part de la renda destinada a cobrir les despeses en electricitat, aquest percentatge ve definit per al slide rentaDestinadaElectricitat.

5.2.3 Càlcul energia produïda

```
to generarEnergia
  ask cases with [death = false] [
    set energiaDisponible (energiaDisponible + (produccio * (( energiaSol * 0.2 ) * temps)))
  ]
end
```



Per calcular l'energia que generen les cases a cada tick es tindran en compte diferents variables: l'energia del sol que arriba a les cobertes de les cases, la producció de la casa, les condicions meteorològiques (temps) i es tindrà en compte que les plaques fotovoltaïques que tenen instal·lades les cases absorbeixen un 20% de l'energia del sol.

5.2.4 Compra electricitat

La compra d'electricitat tindrà diverses fases, en primer lloc les cases negociaran entre elles per tal de distribuir l'energia sobrant que han produït i en segon lloc aquelles cases que no han cobert les seves necessitats compraran l'energia restant a la companyia elèctrica.

Com ja s'ha dit anteriorment hi ha dos algorismes de negociació: negociació estàtica i negociació lliure.

5.2.4.1 Negociació lliure

```
;;implementacio algoritme comerç lliure
to comerçLliure

;; en primer lloc establim si les cases som compradores o venedores d'electricitat i establim un preu
setPreuElectricitatLliure

;; en segon lloc, les cases compradores seleccionen les millors cases per a vendre electricitat
assignacioLinks 1

;;finalment processem la compra d'energia
processaNegociacio

end

to setPreuElectricitatLliure
ask cases with [death = false] [

  if count my-links = 0 [

    ifelse energiaDisponible > consum [
      set venedor true
      set energiaVenta (energiaDisponible - consum)
    ]
    [
      set venedor false
      set preuCompra preuNegociacio
      set rendibilitat ((consum - energiaDisponible) * preuCompra)
    ]
  ]
]
end
```

L'algoritme de negociació lliure es divideix en tres passos, en el primer pas cada casa decidirà si vol comprar energia o si la vol vendre, aquesta decisió es farà en funció de si les cases tenen sobre producció o no, en cas que la casa tingui sobre producció, posarà a la venda aquella energia que els hi sobra tenint en compte que ha de cobrir el seu consum, en cas que no tinguin suficient producció per cobrir el seu consum, les cases estaran interessades a comprar tota aquella energia elèctrica que necessitin.

```
to assignacioLinks [t]
let millorCasa -1
let millorRentavilitat 0
ask cases with [death = false and venedor = true][

  if count my-links = 0 [
    ask other cases in-radius RadiVenta with [death = false and venedor = false] [
      if millorCasa = -1 or millorRentavilitat < rentavilitat and count my-links = 0
      [
        set millorCasa self
        set millorRentavilitat rentavilitat
      ]
    ]
  ]
  if millorCasa != -1 [
    create-conexio-from millorCasa
    ask conexio [who] of millorCasa who [
      set nTicks t
    ]
    set millorCasa -1
  ]
]
end
```

En el segon pas, les cases compradores seleccionaran les cases a les quals els hi vendran l'energia elèctrica, aquesta elecció es farà calculant la rendibilitat que els hi donarà cada una de les cases compradores, aquella casa que els hi ofereixi una millor rendibilitat serà la que escolliran per vendre'ls-hi l'energia, el càlcul de la rendibilitat es farà:

rendibilitat= energiaCompra * preuEnergia.

Les cases tindran una distància màxima per a poder interaccionar amb altres cases, aquesta distància ve definida per la variable radiVenda, és una variable d'entrada del model, explicada al capítol Variables d'interacció entre agents.

```
to processaNegociacio
ask conexions [
;; context del venedor
ask end2 [
;;context del comprador
ask other-end [

    ifelse [energiaVenta] of myself < consum - energiaDisponible
    [
        set energiaDisponible ( energiaDisponible + ( [energiaVenta] of myself ))
        set eC [energiaVenta] of myself
    ]
    [
        set eC ( consum - energiaDisponible )
        set energiaDisponible consum
    ]

    ]
set eComercialitzada ( eComercialitzada + energiaDisponible )
]
set eComercialitzada ( eComercialitzada + eC )

set eComprada eC
set capital ( capital - ( eC * preuCompra ))
set preuComp preuCompra
]
set energiaDisponible ( energiaDisponible - eComprada )
set energiaVenta 0
set capital ( capital + (eComprada * preuComp ) )
]
set nTicks (nTicks - 1)
if nTicks = 0
[ die ]
]
end
```

Finalment es processarà la compra d'energia, en aquest pas es durà a terme l'intercanvi d'energia elèctrica entre les dues cases.

S'actualitzarà el capital de cada una de les cases i l'energia elèctrica de la qual disposen. El contracte entre les dues cases només durarà un dia, per tant, les cases repetiran el mateix procediment cada dia.

5.2.4.2 Negociació estàtica

```
;;implementacio algoritme comerç estatic
to comerçEstatic

    setPreuElectricitatEstatic

    assignacioLinks 4

    processaNegociacio

end
```

```

to setPreuElectricitatEstatic
ask cases with [death = false] [
  ifelse count my-links = 0 [
    ifelse energiaDisponible > consum [
      set venedor true
      set energiaVenta (energiaDisponible * (50 / 100))
    ]
    [
      set venedor false
      set preuCompra preuEnegociacio
      set rentavilitat ((consum - energiaDisponible) * preuCompra)
    ]
  ]
  [
    if venedor = true [
      set energiaVenta (energiaDisponible * (50 / 100))
    ]
    if venedor = false [
      set preuCompra preuEnegociacio
    ]
  ]
]
end

```

L'algoritme de negociació estàtica és molt semblant al de negociació lliure però hi ha més restriccions.

En el primer pas es determinarà si les cases seran compradores o venedores d'energia elèctrica, les cases venedores hauran de vendre sempre el 50% de l'energia produïda en aquell tick, encara que la necessitin per a elles i les cases compradores estaran obligades a comprar tota aquesta energia elèctrica encara que els hi suposi tenir excedent.

```

to assignaciolinks [t]
let millorCasa -1
let millorRentavilitat 0
ask cases with [death = false and venedor = true][

  if count my-links = 0 [
    ask other cases in-radius RadiVenta with [death = false and venedor = false] [
      if millorCasa = -1 or millorRentavilitat < rentavilitat and count my-links = 0
      [
        set millorCasa self
        set millorRentavilitat rentavilitat
      ]
    ]
  ]
  if millorCasa != -1 [
    create-conexio-from millorCasa
    ask conexio [who] of millorCasa who [
      set nTicks t
    ]
    set millorCasa -1
  ]
]
end

```

En el segon pas les cases compradores decidiran a quines cases els hi venen l'energia elèctrica, aquesta elecció es farà en funció de la quantitat d'energia elèctrica que comprin les cases, aquelles cases que necessitin una major quantitat d'energia elèctrica tindran prioritat. La distància màxima de negociació vindrà donada per la variable d'entrada radiVenta del grup de variables: Variables d'interacció entre agents. El contracte entre les dues cases tindrà una durada de quatre dies.

```

to processaNegociacio
  ask conexions [

  ;; context del venedor
  ask end2 [
    set eComprada 0
    set eC 0
    ;;context del comprador
    ask other-end [
      set energiaDisponible ( energiaDisponible + [energiaVenta] of myself )
      set eComercialitzada ( eComercialitzada + energiaDisponible )
      set eC [energiaVenta] of myself
    ]

    set eComprada eC
    set capital ( capital - ( eC * preuCompra ) )
    set preuComp preuCompra
  ]
  set energiaDisponible ( energiaDisponible - eComprada )
  set energiaVenta 0
  set capital ( capital + (eComprada * preuComp ) )

  set nTicks (nTicks - 1)
  if nTicks = 0
    [ die ]
  ]
end

```

Finalment es processarà la compra d'energia, en aquest pas es durà a terme l'intercanvi d'energia elèctrica i s'actualitzarà el capital i l'energia disponible de cada una de les cases. També es comprovarà si ja han passat els quatre dies de la durada del contracte, si el contracte fa quatre dies que es va acordar, les cases seran lliures de comercialitzar amb altres cases.

5.2.4.3 Compra a la xarxa d'energia electrica

```

to procesaCompra
  ask cases with [death = false] [
    ;;si l'energia disponible es menor al consum
    if energiaDisponible < consum [
      let x (consum - energiaDisponible)
      set consumMes (consumMes + (x * preuExarxa))
      set energiaDisponible 0
    ]
    if energiaDisponible >= consum [
      set energiaDisponible (energiaDisponible - consum)
    ]
    ;; s'injecta el restant d'energia a la companyia i es calcula la retribucio que obtindra la casa
    set energiaInjectada ( energiaDisponible + energiaInjectada )
    set capital ( capital + ( energiaDisponible * preuEinjectada))
    set energiaDisponible 0

    if ticks mod 30 = 0 [
      set capital (capital - consumMes)
      set eCompradaXarxa ( eCompradaXarxa + consumMes )
      if capital < 0 [
        set death true
        set color red
      ]
      set consumMes 0
    ]
  ]
end

```


Finalment, totes aquelles cases que no tinguin suficient energia per a cobrir les seves necessitats acudirán a la xarxa d'energia elèctrica, la factura es pagarà cada mes.

Aquelles cases que en un tick determinat tinguin un excés d'electricitat injectaran el sobrant a la xarxa rebent una compensació econòmica, el valor d'aquesta compensació econòmica vindrà donada per la variable `preuEinjectada`, del grup de: Variables d'interacció entre agents.

6. Proves realitzades

6.1 Proves

A continuació es mostraran diferents simulacions que s'han realitzat amb el model implementat, cada una de les proves es simularà 10 cops diferents amb 10 random-seed diferents, a netlogo la variable random-seed s'utilitza per a establir "l'aleatorietat", els valors de random-seed utilitzats seran: 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000 i 10000 per cada una de les proves.

6.1.1 Primera prova

Input

Inicialització

Input

Inicialització	Variables cases	Variables globals	Variables negociació
inicialitzacio sequencial	initCapital 200 €	probSol 70 %	TipusNegociacio Estatica
numCases 100	rentaDestinadaElectricitat 4.0 %	energiaSol 50 kWh	RadiVenta 20
Seleccio mapes	rentaBaixa 900 €		preuExarxa 0.15 €
zonaRubi sud	rentaMitja 2300 €		preuNegociacio 0.08 €
mapaProd estandar	rentaAlta 3500 €		preuInjectada 0.01 €

Els valors de renda per a la primera prova que s'han agafat pels diferents perfils de cases són els següents:

- Renda baixa: Les cases amb renda baixa uns ingressos de 900 €, el salari mínim interprofessional espanyol [6]
- Renda mitja: Per a les cases amb renda mitja s'ha agafat el salari mig espanyol, 2300 € [7]
- Renda alta: Per les cases amb renda alta s'ha agafat una renda de 3500 €

Com a variables globals s'han pres els següents valors:

S'ha agafat una probabilitat de sol del 70% i una energia solar de 50 kWh.

Si tenim en compte que a la superfície de la terra hi arriben 7 kWh/m² [9][10] les cases haurien de tenir instal·lades 7 m² de plaques fotovoltaïques.

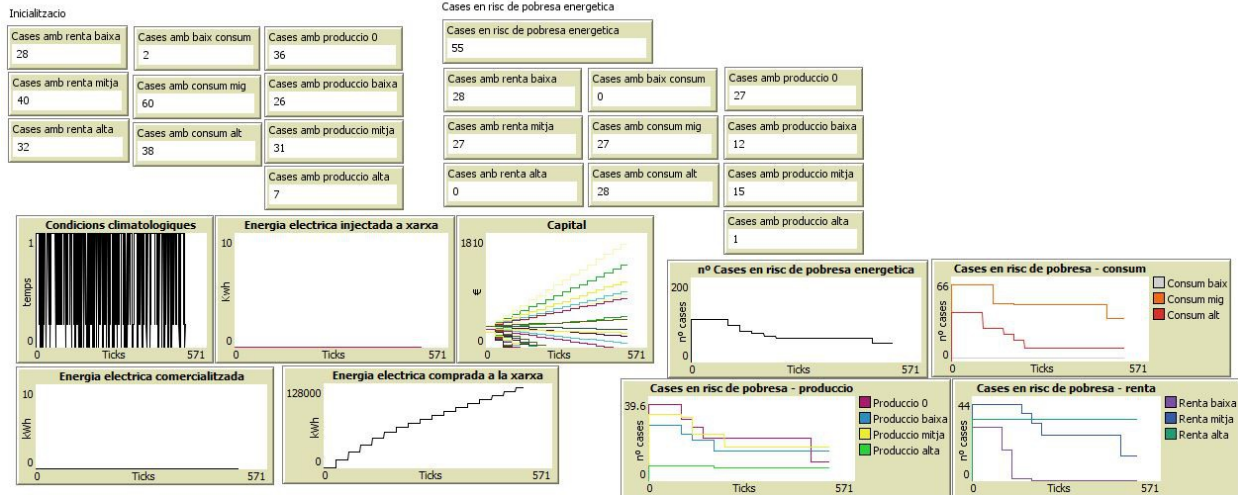
Pel que fa a les variables referents a les interaccions amb els agents s'han agafat els següents valors: El radi de venda serà de 20 patches, el preu de l'energia que les cases compren a la xarxa és 0.15 € el kWh, aquest valor és el cost actual del kWh a Espanya, el preu de l'energia injectada a la xarxa serà de 0.01€ kWh, aquest valor ens el van donar des de l'ajuntament de Rubí. I el preu de negociació entre les cases de 0.08 €. El tipus de negociació serà estàtica.

Output



Figura 5

Output



Mitjana de cases al llarg de les 10 simulacions

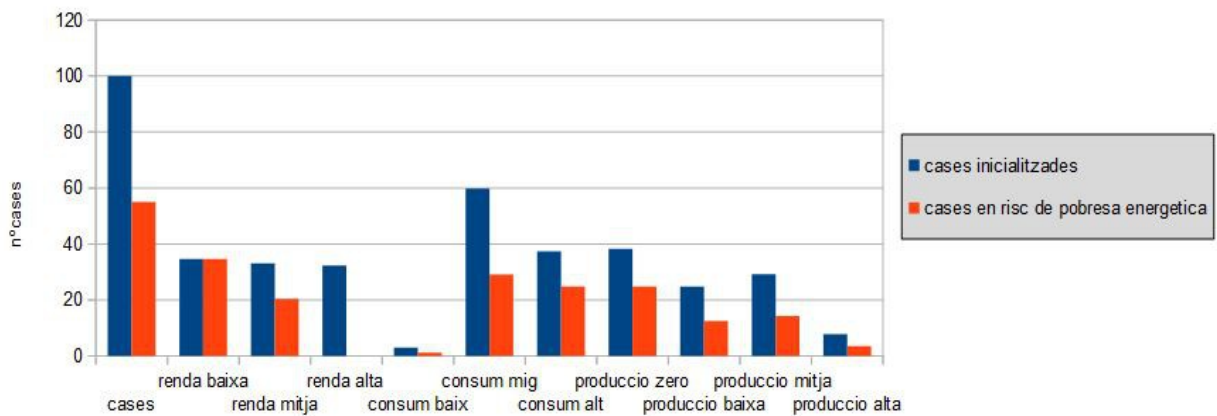


Figura 6

El gràfic de la Figura 6 ens mostra, per una banda les barres blaves la mitjana de cases inicialitzades en cada un dels diferents perfils i les barres vermelles ens mostren la mitjana de les cases que han finalitzat les simulacions en risc de pobresa energètica. Les dues primeres barres corresponen al nombre total de cases independentment del perfil que tinguin. Això es mantindrà pels gràfics de totes les proves.

En aquesta primera prova podem observar que tot i que la producció d'energia no és suficient per a abastir la demanda d'energia de les cases, aquelles cases que són productores d'energia ho tenen més fàcil per a fer front a la factura de la llum, ja que el nombre de cases amb producció alta que finalitzen les simulacions és més elevat que la resta.

També podem observar que en aquesta simulació, el nivell de renda de les cases té un pes molt important, totes aquelles cases que tenen un nivell de renda baix han acabat la simulació en risc de pobresa energètica, moltes de les cases amb una renda mitja també, en canvi, les cases amb renda alta han aconseguit totes elles fer front a la factura de la llum sense problemes.

També cal remarcar el pes que ha tingut el consum d'energia, com podem observar a mesura que va pujant el consum, el percentatge de cases en risc de pobresa energètica augmenta considerablement.

En aquesta simulació la producció no era suficient per a abastir les cases, ja que les cases no han interactuat entre elles, ja que l'energia total comercialitzada és 0, l'algoritme de negociació no ha tingut cap impacte.

6.1.2 Segona prova

Input

Inicialització

inicialitzacio
sequencial

numCases 100

Seleccio mapes

zonaRubi
sud

mapaProd
estandar

Input

Variables cases

initCapital 200 €

rentaDestinadaElectricitat 4.0 %

rentaBaixa 900 €

rentaMitja 2300 €

rentaAlta 3500 €

Variables globals

probSol 80 %

energiaSol 150 kWh

Variables negociacio

TipusNegociacio
Estatica

RadiVenta 20

preuExarxa 0.15 €

preuNegociacio 0.08 €

preuInjectada 0.01 €

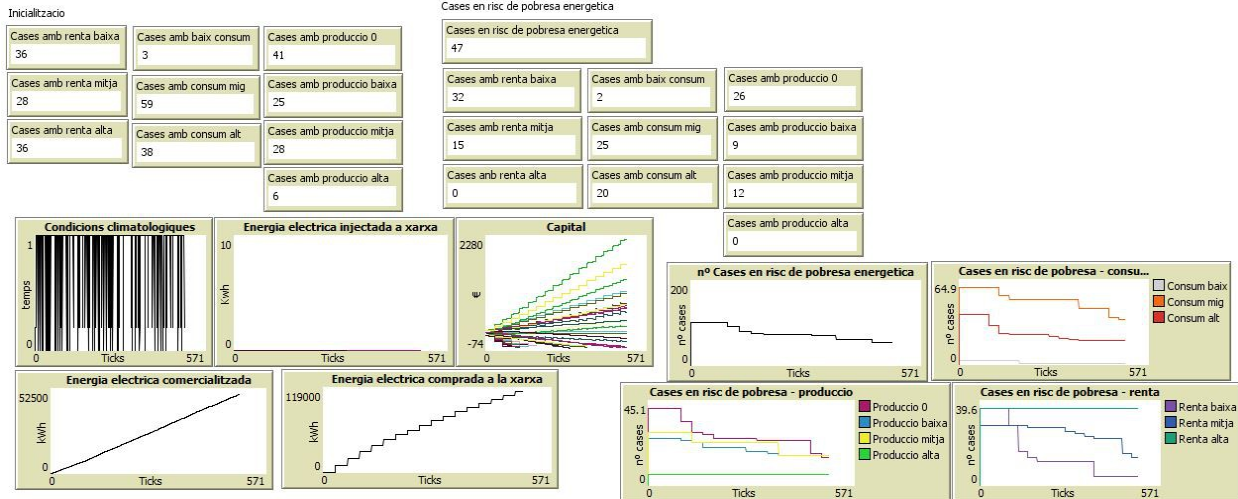
Per a la segona prova, s'han agafat les mateixes variables de renda que a la prova anterior i les mateixes variables de negociació, s'ha incrementat l'energia solar fins a 150 kWh i la probabilitat de sol també s'ha augmentat a 80%.

Output



Figura 7

Output



Mitjana de cases al llarg de les 10 simulacions

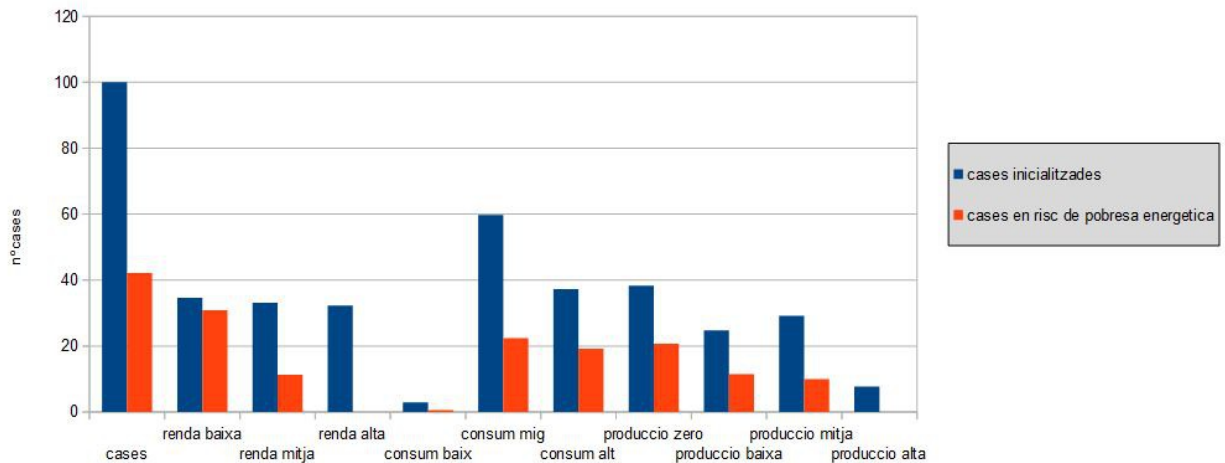


Figura 8

En aquesta segona prova, podem observar que el nombre total de cases en risc de pobresa energètica s'ha reduït (de 55 a 42) a causa de l'augment de la producció d'energia elèctrica. Podem veure que algunes de les cases amb renda baixa ja poden fer front a la factura de la llum, cosa que no passava a la simulació anterior, i com podem observar a la Figura 7, aquelles cases que tenen renda baixa i han finalitzat la simulació, són cases que tenen una producció alta o mitja d'energia elèctrica.

També es pot observar, que totes aquelles cases amb una producció alta han finalitzat totes les simulacions sense estar en risc de pobresa energètica.

6.1.3 Tercera prova

Input

Input

Inicialitzacio	Variables cases	Variables globals	Variables negociacio
inicialitzacio sequencial	initCapital 200 €	probSol 80 %	TipusNegociacio Estatica
numCases 100	rentaDestinadaElectricitat 4,0 %	energiaSol 150 kWh	RadiVenta 20
Seleccio mapes	rentaBaixa 1100 €		preuExarxa 0.15 €
zonaRubi sud	rentaMitja 2300 €		preuNegociacio 0.08 €
mapaProd estandar	rentaAlta 3500 €		preuEinjectada 0.01 €

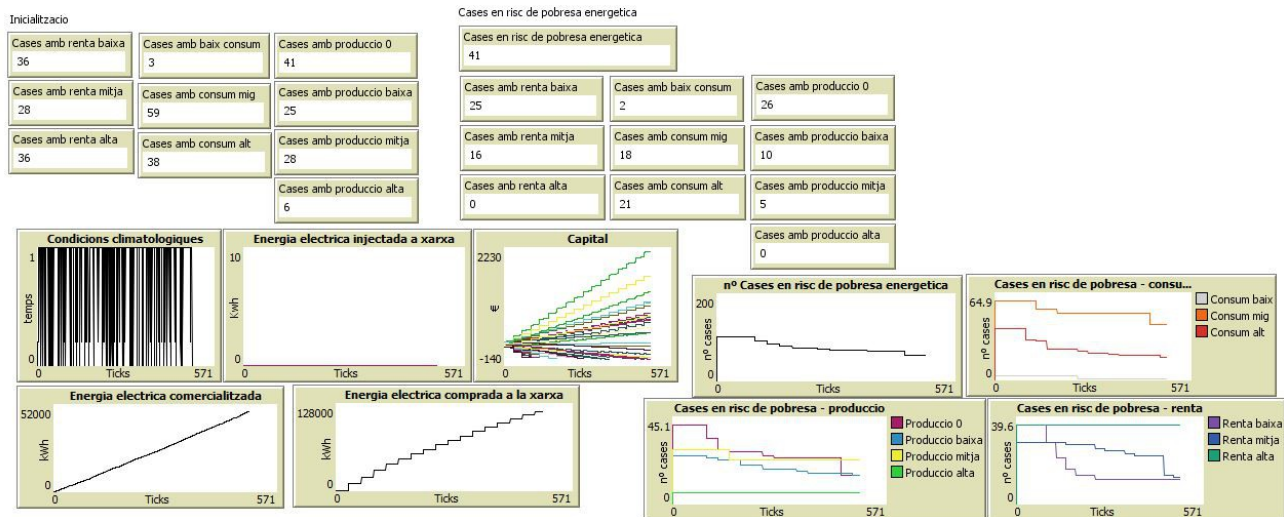
Per a la tercera simulació, s'han agafat les mateixes variables de negociació i de producció d'energia que en la prova anterior, s'ha incrementat la renda baixa a 1100 €.

Output



Figura 9

Output



Mitjana de cases al llarg de les 10 simulacions

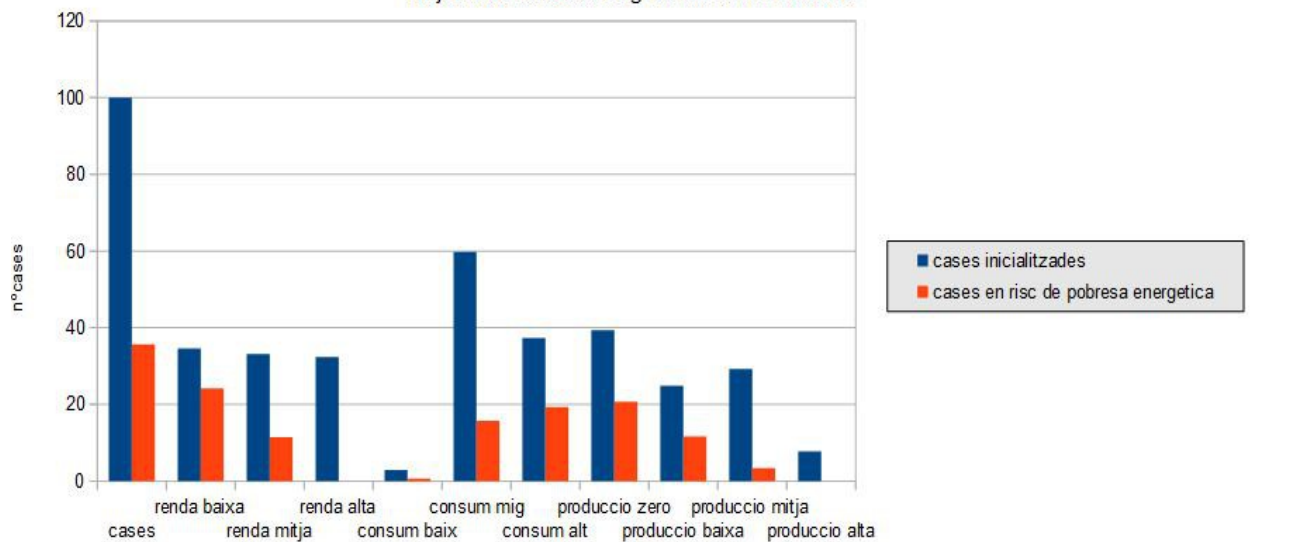


Figura 10

Podem observar que el nombre de cases en risc de pobresa energètica amb una renda baixa ha disminuït respecte a l'anterior simulació i la 6.1.1 Primera prova (de 100% a la primera prova al 70% en aquesta prova), tot i així, podem observar a la Figura 9, com les cases amb una renda baixa segueixen depenent de la seva pròpia producció, ja que les cases amb renda baixa que han pogut afrontar la factura de la llum tenien totes elles una producció alta o mitja d'energia elèctrica.

6.1.4 Quarta prova

Input

Input

Inicialització	Variables cases	Variables globals	Variables negociació
inicialització: sequencial	initCapital: 200 €	probSol: 80 %	TipusNegociació: Estatica
numCases: 100	rentaDestinadaElectricitat: 8.0 %	energiaSol: 150 kWh	Radiventa: 20
Seleccio mapes	rentaBaixa: 1100 €		preuExarxa: 0.15 €
zonaRubi: sud	rentaMitja: 2300 €		preuEnegociació: 0.08 €
mapaProd: estandar	rentaAlta: 3500 €		preuEinjectada: 0.01 €

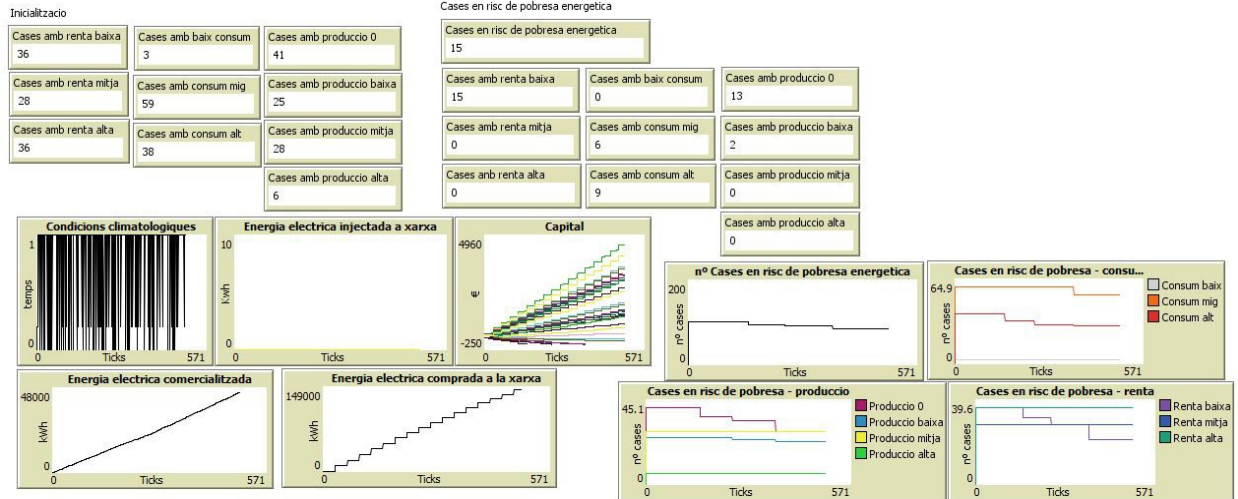
A la quarta prova s'ha augmentat el percentatge de renda que destinen les cases a pagar la factura de la llum, en aquest cas s'ha establert un 8%. S'han mantingut les variables de negociació i producció d'energia.

Output



Figura 11

Output



Mitjana de cases al llarg de les 10 simulacions

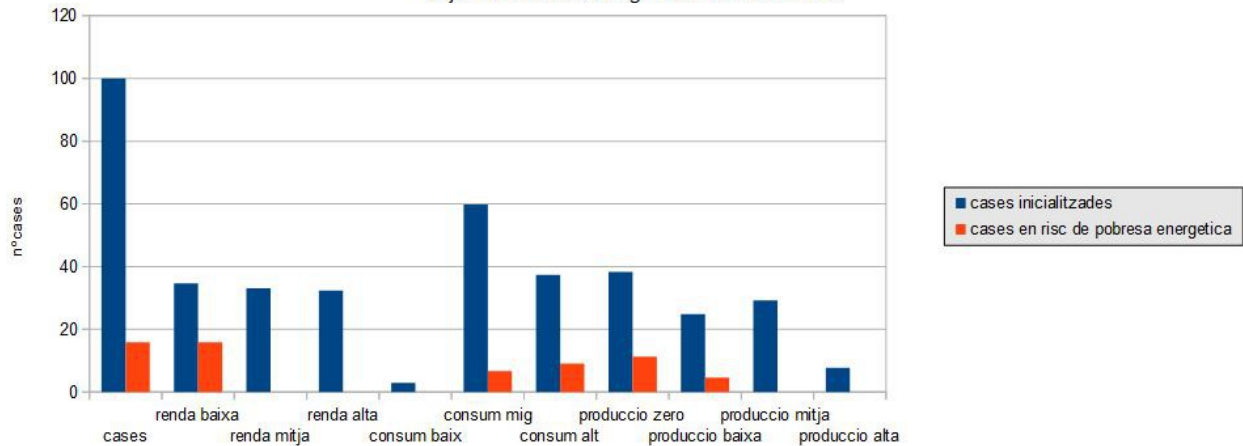


Figura 12

En aquesta simulació podem observar com el nombre total de cases que finalitzen les simulacions en risc de pobresa energètica ha disminuït considerablement respecte a la prova anterior (de 35 a 15) i que totes aquelles cases en risc de pobresa energètica, són aquelles cases amb perfils molt desfavorables. S'ha de donar una combinació de renda baixa, consums mig o alt i una producció d'energia elèctrica molt baixa.

6.1.5 Cinquena prova

Input

Input

Inicialització	Variables cases	Variables globals	Variables negociació
inicialització sequencial	initCapital 200 €	probSol 70 %	TipusNegociació Lliure
numCases 100	rentaDestinadaElectricitat 4.0 %	energiaSol 50 kWh	RadiVenda 20
Seleccio mapes	rentaBaixa 900 €		preuExarxa 0.15 €
zonaRubi sud	rentaMitja 2300 €		preuNegociació 0.08 €
mapaProd estandar	rentaAlta 3500 €		preuEinjectada 0.01 €

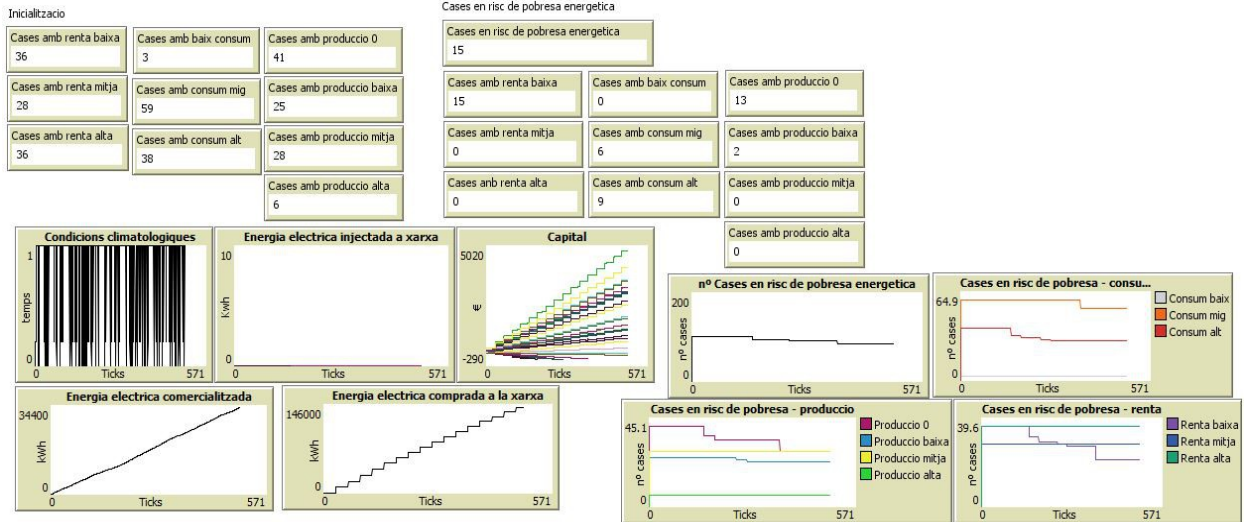
En aquesta prova es mantenen tots els valors de la prova anterior exceptuant l'algoritme de negociació, a la prova anterior s'utilitzava el procediment de negociació estàtica, en aquesta prova s'utilitzarà l'algoritme de negociació lliure.

Output



Figura 13

Output



Mitjana de cases al llarg de les 10 simulacions

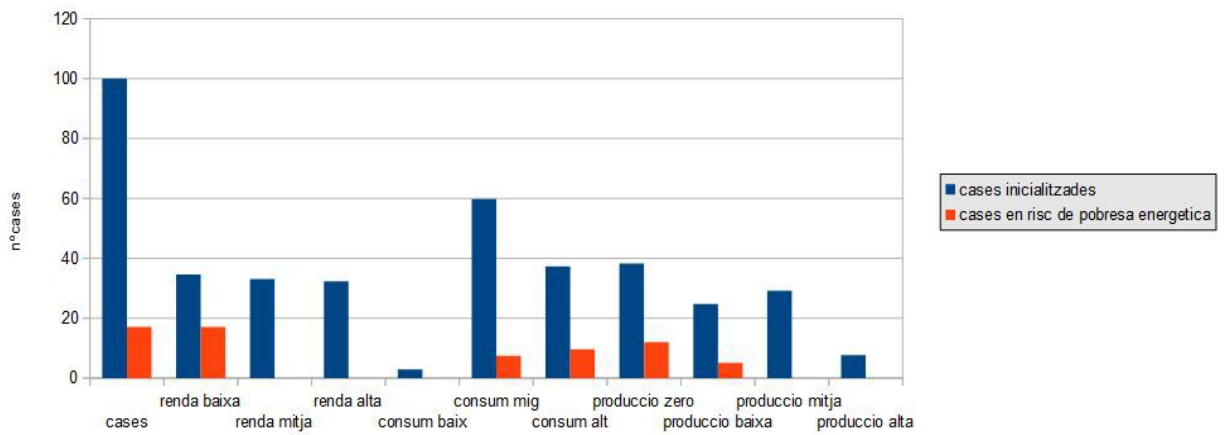


Figura 14

Podem observar com modificant l'algorisme de negociació no s'aprecia una diferència important en el nombre de cases en risc de pobresa energètica que hi ha al final de cada simulació. Els resultats d'aquesta 5a prova són molt semblants als resultats de la 6.1.4 Quarta prova.

6.1.6 Sisena prova

Input

Input

Inicialitzacio	Variables cases	Variables globals	Variables negociacio
inicialitzacio aleatoria	initCapital 200 €	probSol 80 %	TipusNegociacio Lliure
numCases 100	rentaDestinadaElectricitat 8.0 %	energiaSol 150 kWh	RadiVenta 20
Seleccio mapes	rentaBaixa 1100 €		preuExarxa 0.15 €
zonaRubi sud	rentaMitja 2300 €		preuEnegociacio 0.08 €
mapaProd estandar	rentaAlta 3500 €		preuEinjectada 0.01 €

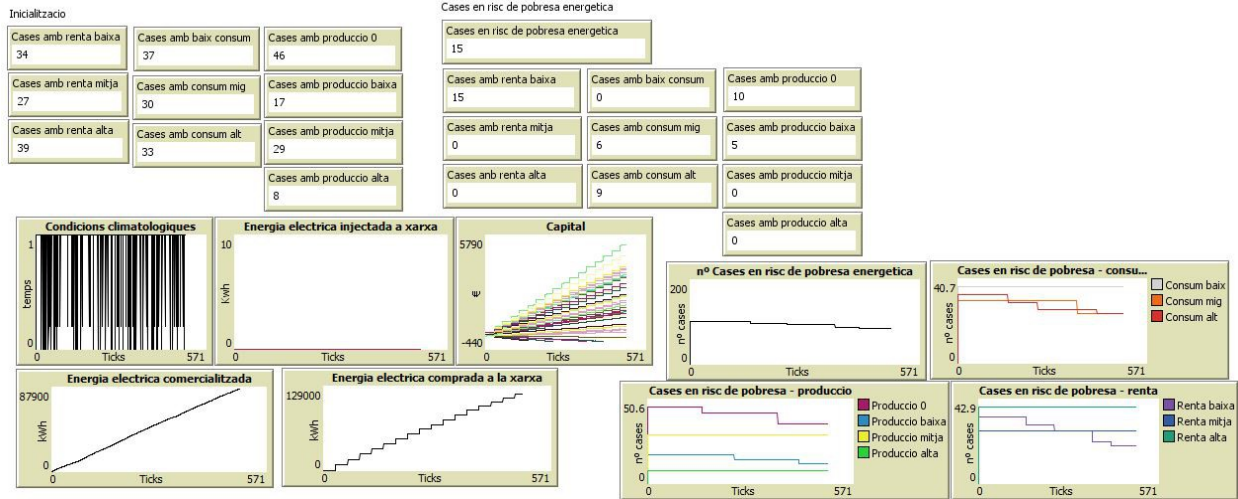
En aquesta prova es mantindran tots els valors de la 6.1.5 Cinquena prova però es modificarà la inicialització de les cases. En aquest cas la inicialització de les cases es farà de manera aleatòria.

Output



Figura 15

Output



Mitjana de cases al llarg de les 10 simulacions

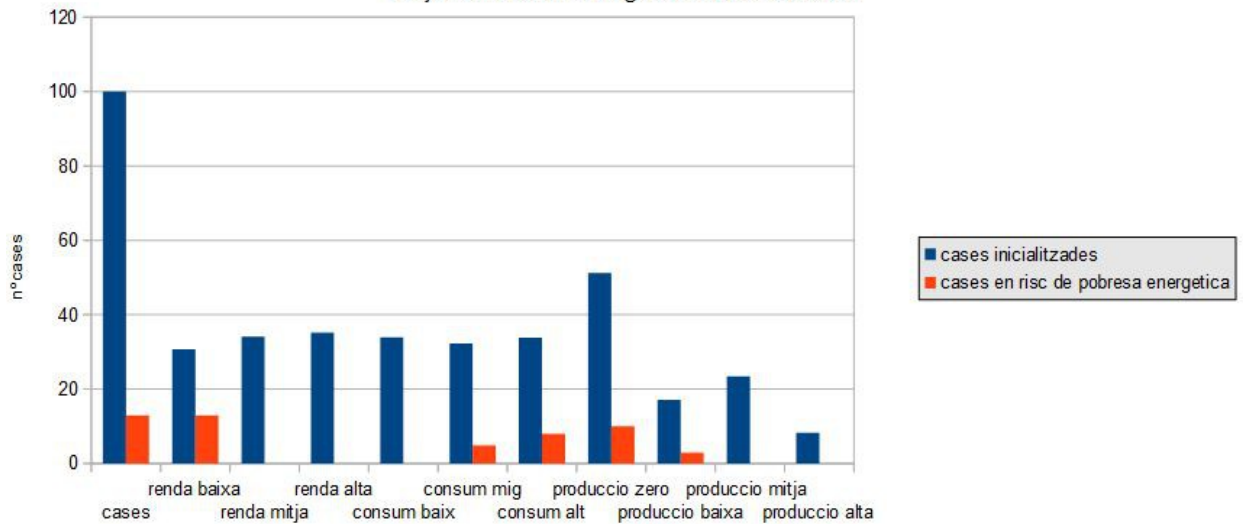


Figura 16

En aquest cas podem observar com el nombre total de cases en risc de pobresa ha disminuït una mica respecte a la simulació anterior (de 17 a 13). Això és degut al fet que amb el posicionament aleatori s'inicialitzen el mateix nombre de cases amb els 3 perfils de consum, mentre que en el posicionament seqüencial el nombre de cases amb consum baix era molt més reduït que les cases amb consum mig i consum alt.

6.1.7 Setena prova

Input

Input

<p>Inicialització</p> <p>inicialització sequencial</p> <p>numCases 100</p> <p>Seleccio mapes</p> <p>zonaRubi sud</p> <p>mapaProd altaProduccio</p>	<p>Variables cases</p> <p>initCapital 200 €</p> <p>rentaDestinadaElectricitat 8.0 %</p> <p>rentaBaixa 1100 €</p> <p>rentaMitja 2300 €</p> <p>rentaAlta 3500 €</p>	<p>Variables globals</p> <p>probSol 80 %</p> <p>energiaSol 150 kWh</p>	<p>Variables negociacio</p> <p>TipusNegociacio Lliure</p> <p>RadiVenta 20</p> <p>preuExarxa 0.15 €</p> <p>preuEnegociacio 0.08 €</p> <p>preuEinjectada 0.01 €</p>
--	---	--	---

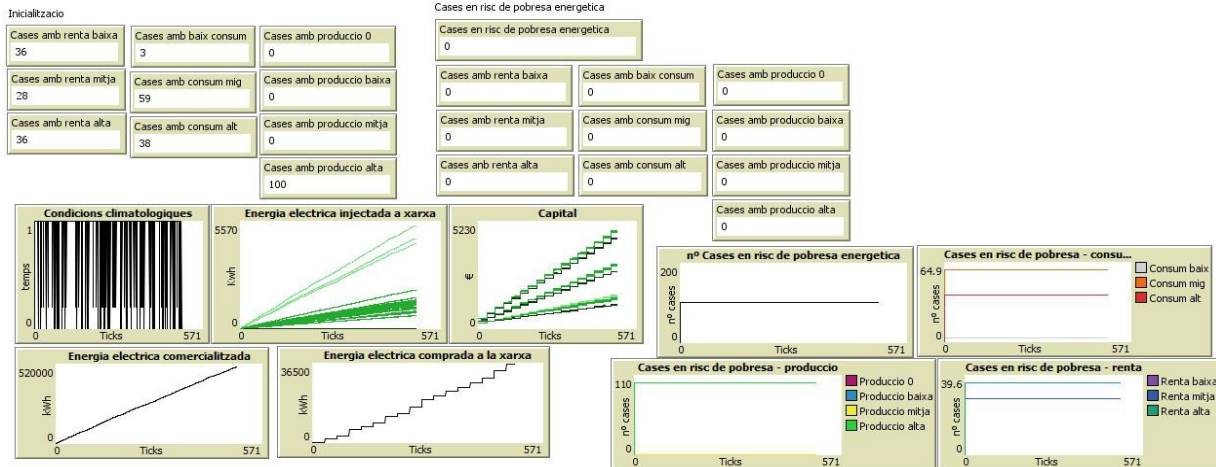
En aquesta prova agafarem els mateixos valors que a la 6.1.5 Cinquena prova, inicialització seqüencial, les mateixes variables de negociació i les mateixes variables referents a les condicions meteorològiques però modificarem la producció de les cases per tal que totes les cases siguin altes productores d'energia elèctrica.

Output



Figura 17

Output



Mitjana de cases al llarg de les 10 simulacions

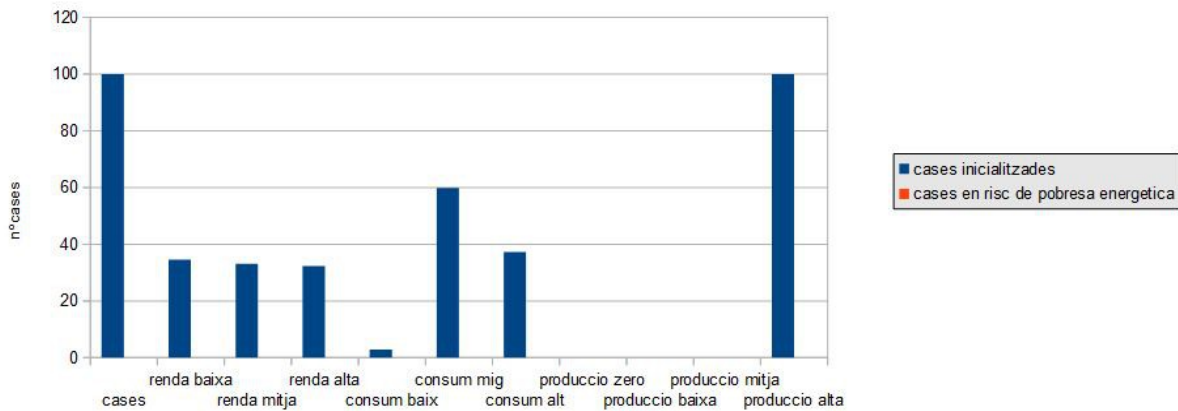


Figura 18

En aquesta prova podem observar com al llarg de les 10 simulacions cap de les cases, tingui el perfil que tingui, ha acabat en risc de pobresa energètica, podem veure doncs que la producció de les cases és una variable molt important a tenir en compte i les ajuda molt a reduir el preu de la factura de la llum.

Per altra banda, podem veure que tot i així, aquelles cases amb un consum elevat segueixen sense ser autosuficients, ja que no injecten energia a la xarxa.

6.1.8 Vuitena prova

Input

Input

Inicialització	Variables cases	Variables globals	Variables negociació
inicialització: sequencial	initCapital: 200 €	probSol: 60 %	TipusNegociació: Lliure
numCases: 100	rentaDestinadaElectricitat: 8.0 %	energiaSol: 80 kWh	RadVenda: 20
Seleccio mapes: zonaRubi: sud	rentaBaixa: 1100 €		preuExarxa: 0.15 €
mapaProd: altaProduccio	rentaMitja: 2300 €		preuNegociació: 0.08 €
	rentaAlta: 3500 €		preuEinjectada: 0.01 €

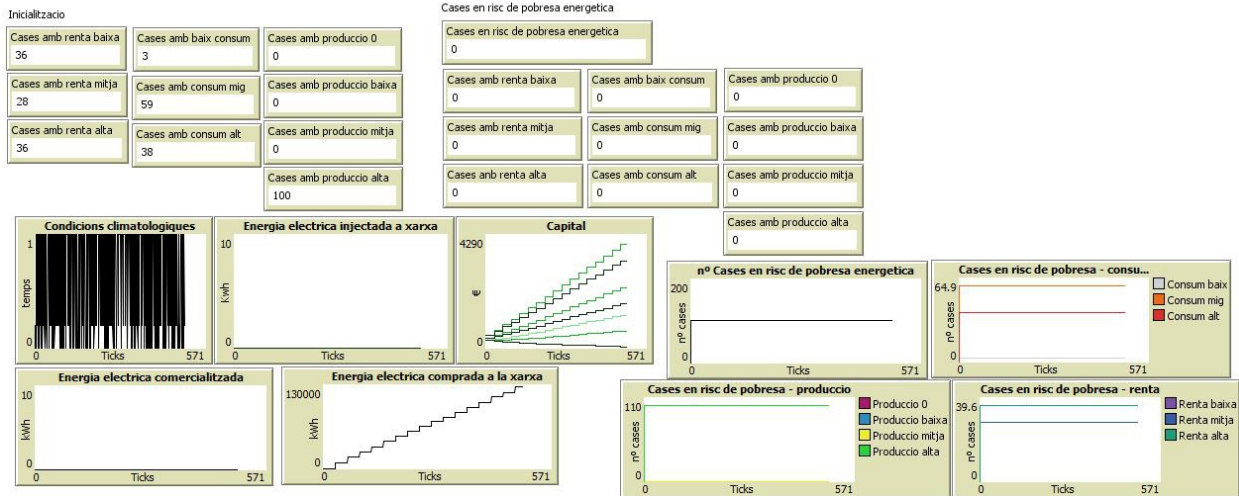
En aquesta simulació reduïm els valors referents a la producció d'energia elèctrica: la probabilitat de sol la baixarem a 60% i l'energia solar la baixarem a 80 kWh, d'aquesta manera compararem la diferència que hi ha en el fet que les cases puguin aprofitar menys l'energia solar, però sent totes altes productores.

Output



Figura 19

Output



Mitjana de cases al llarg de les 10 simulacions

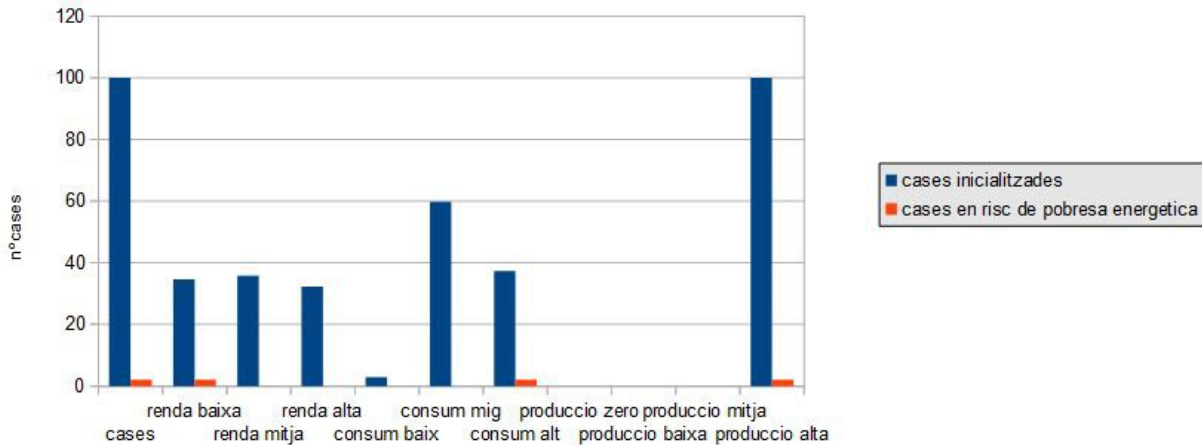


Figura 20

En aquesta prova, tot i reduir considerablement les variables que afecten la producció d'energia: la probabilitat de sol i l'energia del sol, podem veure com al llarg de les 10 simulacions molt poques cases un cop transcorreguts els cinc-cents dies de cada una de les deu simulacions han quedat en risc de pobresa energètica, totes les cases que han quedat en risc de pobresa energètica tenien perfils molt poc favorables, ja que tenen una renda baixa i un consum alt. La gràfica que ens mostra el capital de les cases es pot observar com hi ha certs perfils de cases que no són capaces d'estalviar diners al llarg del mes.

6.1.9 Novena prova

Input

Input

<p>Inicialització</p> <p>inicialització sequencial</p> <p>numCases 100</p> <p>Seleccio mapes</p> <p>zonaRubi central</p> <p>mapaProd estandar</p>	<p>Variables cases</p> <p>initCapital 200 €</p> <p>rentaDestinadaElectricitat 4,0 %</p> <p>rentaBaixa 900 €</p> <p>rentaMitja 2300 €</p> <p>rentaAlta 3500 €</p>	<p>Variables globals</p> <p>probSol 70 %</p> <p>energiaSol 50 kWh</p>	<p>Variables negociació</p> <p>TipusNegociació Estatica</p> <p>RadiVenta 20</p> <p>preuExarxa 0.15 €</p> <p>preuNegociació 0.08 €</p> <p>preuInjectada 0.01 €</p>
---	--	---	---

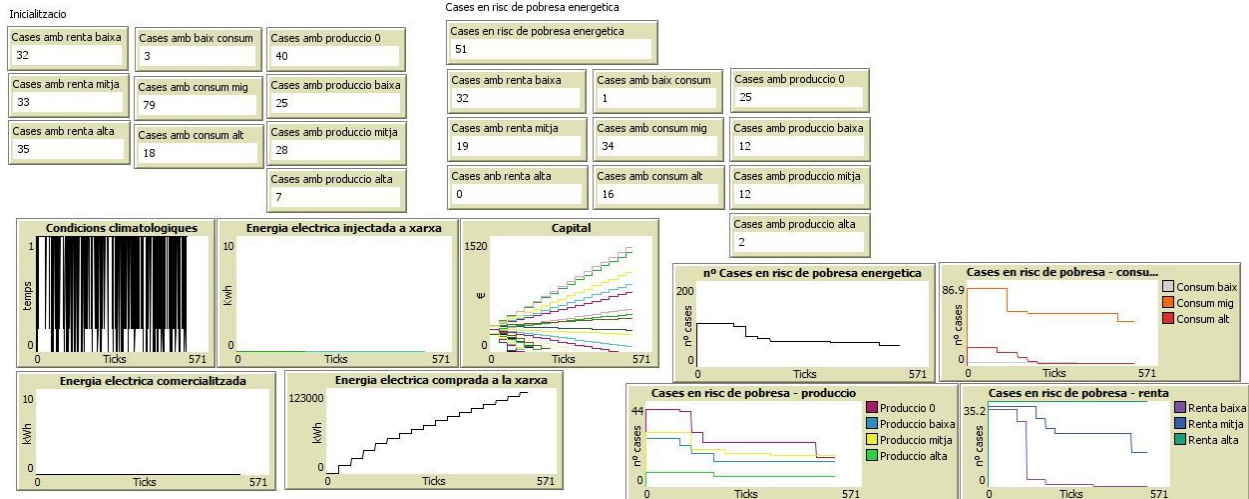
Per a les següents proves testejarem l'altre entorn de Rubí, agafarem els mateixos valors que a la 6.1.1 Primera prova però modificant el mapa d'inicialització.

Output



Figura 21

Output



Mitjana de cases al llarg de les 10 simulacions

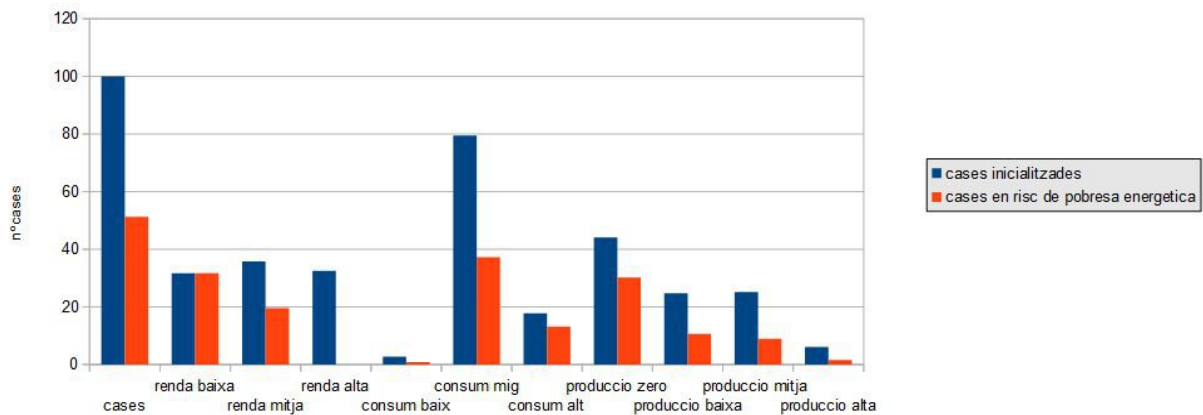


Figura 22

Podem observar, que tot i canviar el mapa de Rubí la inicialització de les cases segueix sent molt semblant a l'anterior, per tant hem obtingut resultats molt semblants a la 6.1.1 Primera prova: Totes les cases amb una renda baixa han finalitzat les simulacions en risc de pobresa energètica, com podem observar a la gràfica cases en risc de pobresa – producció i a la gràfica de la Figura 22, les cases amb més producció tendeixen a sobreviure més. Finalment, podem dir que l'algoritme de negociació no ha tingut efecte en aquesta prova, ja que l'energia elèctrica comercialitzada és 0.

6.1.10 Desena prova

Input

Inicialitzacio

inicialitzacio
sequencial

numCases 100

Seleccio mapes

zonaRubi
central

mapaProd
estandar

Input

Variables cases

initCapital 200 €

rentaDestinadaElectricitat 4.0 %

rentaBaixa 1300 €

rentaMitja 2300 €

rentaAlta 3500 €

Variables globals

probSol 70 %

energiaSol 50 kWh

Variables negociacio

TipusNegociacio
Estatica

RadiVenta 20

preuExarxa 0.15 €

preuNegociacio 0.08 €

preuInjectada 0.01 €

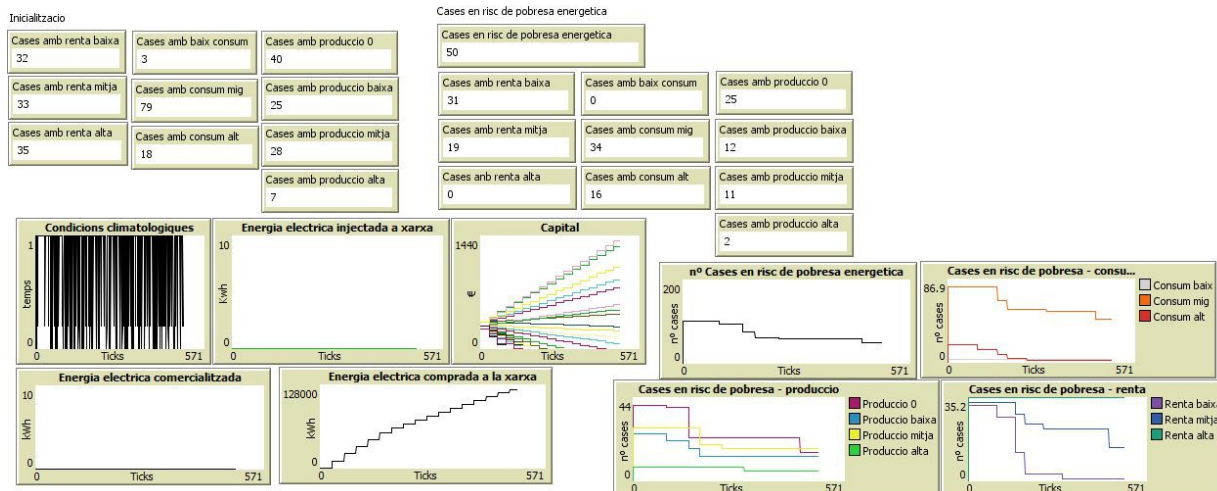
Per a la prova 10 augmentarem la variable de renda baixa a 1300 € i es mantindran la resta de variables

Output



Figura 23

Output



Mitjana de cases al llarg de les 10 simulacions

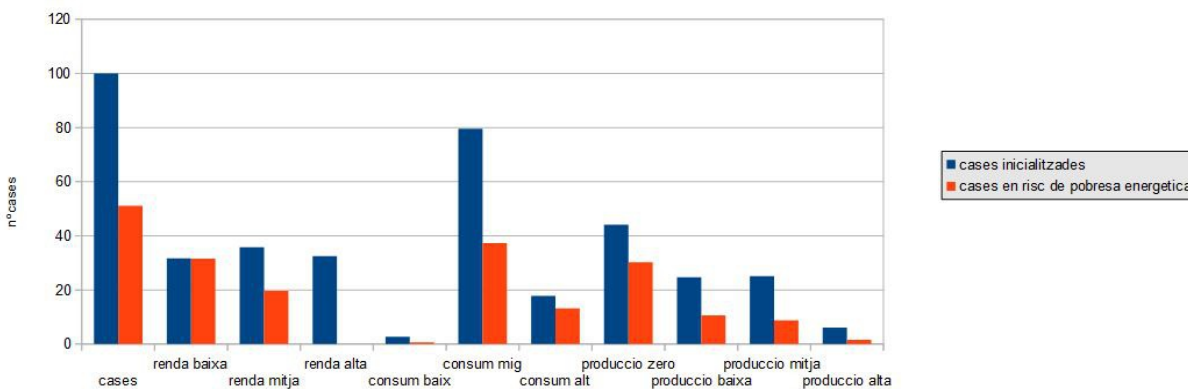


Figura 24

En aquesta prova tot i augmentar la renda baixa a 1300 €, podem observar uns resultats molt semblants a la 6.1.9 Novena prova, totes aquelles cases amb renda baixa han finalitzat les 10 simulacions en risc de pobresa energètica.

6.1.11 Onzena prova

Input

Input

<p>Inicialització</p> <p>inicialització sequencial</p> <p>numCases 100</p> <p>Seleccio mapes</p> <p>zonaRubi central</p> <p>mapaProd estandar</p>	<p>Variables cases</p> <p>initCapital 200 €</p> <p>rentaDestinadaElectricitat 4.0 %</p> <p>rentaBaixa 1300 €</p> <p>rentaMitja 2300 €</p> <p>rentaAlta 3500 €</p>	<p>Variables globals</p> <p>probSol 70 %</p> <p>energiaSol 200 kWh</p>	<p>Variables negociacio</p> <p>TipusNegociacio Estatica</p> <p>RadiVenta 40</p> <p>preuExarxa 0.15 €</p> <p>preuNegociacio 0.04 €</p> <p>preuInjectada 0.01 €</p>
---	---	--	---

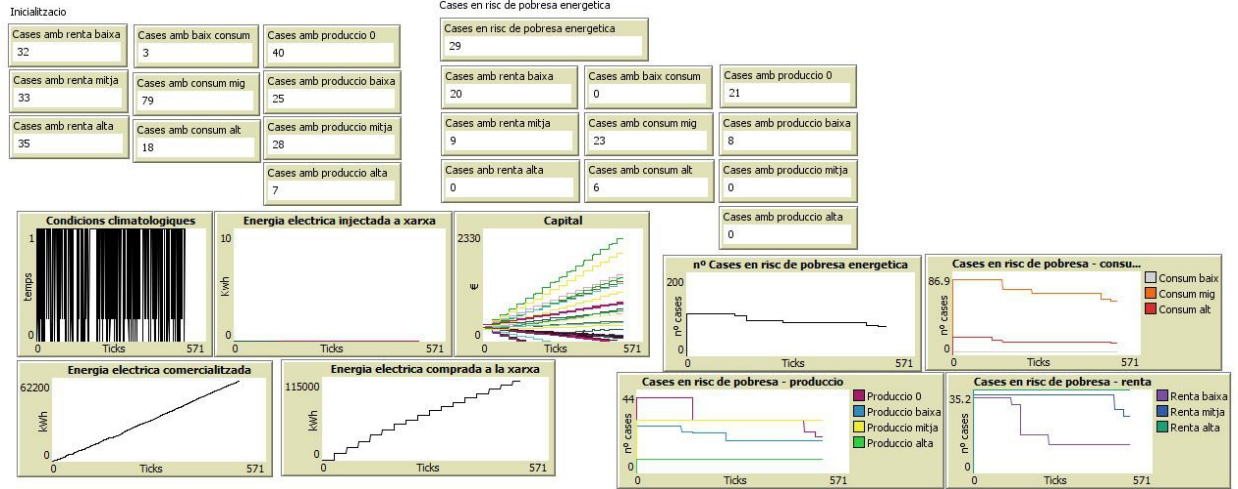
Per les següents dues proves es vol mirar l'eficiència de cadascun dels algorismes de negociació, per poder-ho provar s'augmentarà la producció d'energia de les cases considerablement per tal que les cases amb producció alta tinguin la capacitat de produir més energia de la que necessiten i així la comercialitzaran, també s'augmentarà el radi de venda a 40 patches per tal que les cases tinguin més llibertat a l'hora de vendre energia elèctrica i es reduirà el preu de l'energia que negociaran les cases a 0.04 €.

Output



Figura 25

Output



Mitjana de cases al llarg de les 10 simulacions

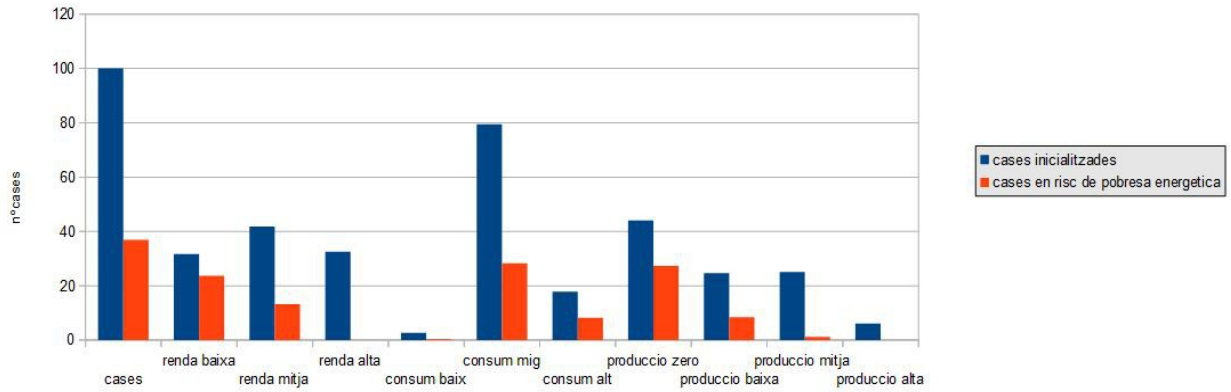


Figura 26

6.1.12 Dotzena prova

Input

Input

Inicialització	Variables cases	Variables globals	Variables negociació
inicialització sequencial	initCapital 200 €	probSol 70 %	TipusNegociació Lliure
numCases 100	rentaDestinadaElectricitat 4.0 %	energiaSol 200 kWh	RadiVenda 40
Seleccio mapes	rentaBaixa 1300 €		preuExarxa 0.15 €
zonaRubi central	rentaMitja 2300 €		preuEnegociació 0.04 €
mapaProd estandar	rentaAlta 3500 €		preuEinjectada 0.01 €

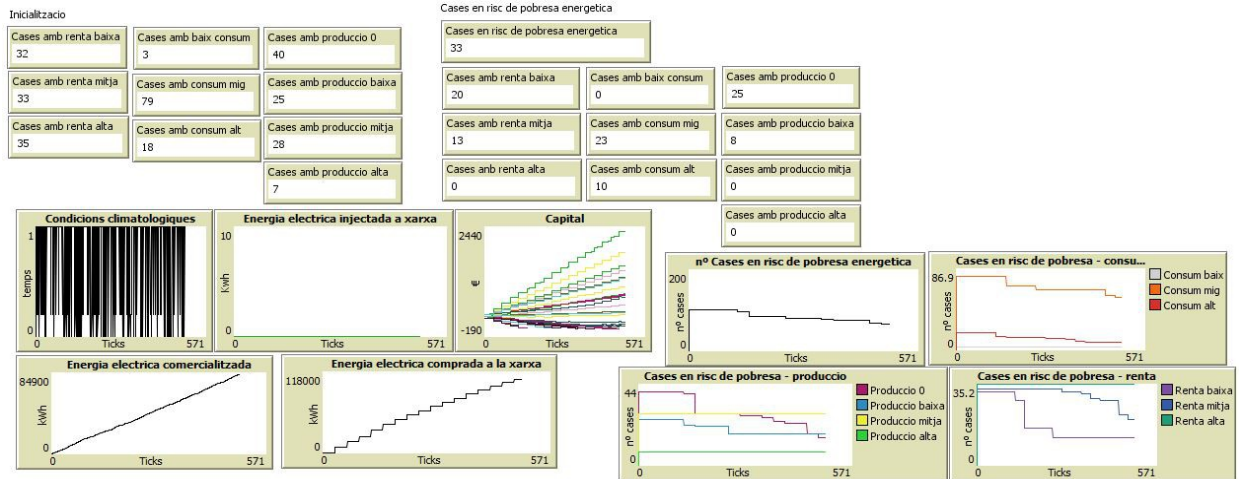
Per a la prova 12 es mantindran els valors de la 6.1.11 Onzena prova però modificant l'algorisme de negociació, en aquest cas s'utilitzarà l'algorisme de negociació lliure.

Output



Figura 27

Output



Mitjana de cases al llarg de les 10 simulacions

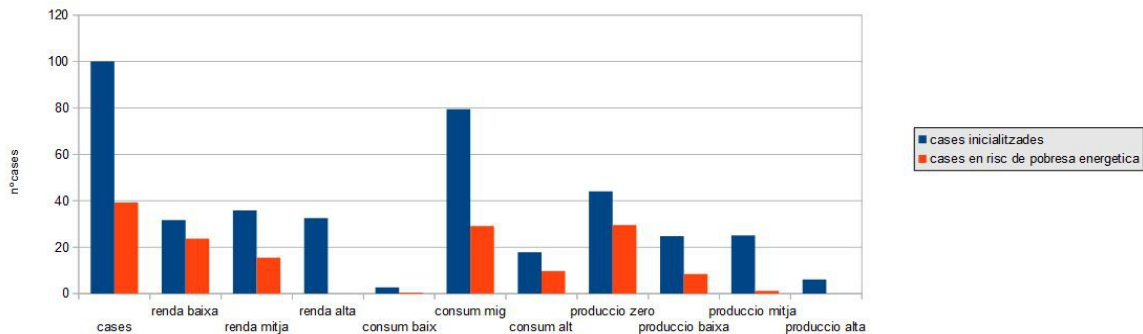


Figura 28

Podem observar que no hi ha una gran diferència si es modifica l'algorisme de negociació a l'hora de mirar el total de cases que acaben les simulacions en risc de pobresa energètica.

6.2 Conclusió

Després d'haver dut a terme les proves amb el model, es pot afirmar que tot i tenir una producció no molt elevada d'energia elèctrica, aquesta producció ajuda a les cases a mantenir-se al llarg del temps evitant el risc de pobresa energètica, dit això, cal esmentar que com hem pogut comprovar, la renda de les cases té un pes molt important i hem vist com les cases amb rendes baixes ho tenen força difícil per a sortir de la pobresa energètica.

També s'ha pogut comprovar com ara per ara es fa força difícil que una urbanització o ciutat pugui ser autosuficient, ja que l'energia produïda no es pot acumular i els dies que les condicions climatològiques no són favorables les cases han de recórrer a l'energia elèctrica que els hi proporciona la companyia elèctrica.

Finalment també s'ha vist com l'algorisme de negociació utilitzat per les cases a l'hora de comercialitzar l'energia elèctrica no és un element diferenciador a l'hora de comprovar el total de cases que queden en risc de pobresa energètica.

7. Conclusions

7.1 Conclusions

A la primera fase del treball, ens vam posar en contacte amb tècnics de l'ajuntament de Rubí que estan desenvolupant el projecte Rubí Brilla [2], per recollir els requisits del projecte. Aquests requisits consistien a incloure al model implementat totes aquelles dades que poguessin ser d'interès per a ells.

Un cop recollits els requisits, en la segona part del projecte, vaig decidir la tecnologia emprada per a la implementació del model. Crec que l'elecció de Netlogo ha estat molt positiva, ja que com he dit al 3. Decisió del framework de simulació multi-agent, netlogo és més senzill a l'hora d'utilitzar-se i fàcil d'aprendre.

A la tercera part, he dissenyat i implementat el model, un model que ens permet comprovar bona part dels objectius que ens vam plantejar des de l'inici del projecte.

Finalment, a l'última part del projecte, he fet una sèrie de proves amb el model implementat. Aquestes proves van orientades a conèixer amb més profunditat els aspectes plantejats a la recollida dels requisits.

7.2 treball futur

Un cop acabat aquest treball m'he adonat que hi ha moltes coses que es poden seguir implementant per a millorar aquest model.

En primer lloc, millorar els dos algorismes de negociació: l'algoritme de negociació estàtica i el de negociació lliure. Implementant algorismes d'aprenentatge automàtic per tal que les cases donessin millors preus a l'energia simulant d'aquesta manera un mercat més realista i més competitiu.

Un altre aspecte que es podria millorar del model és donar variables més realistes a les cases tant de renda com de consum. En el nostre model tenim 3 perfils diferents de renda i consum, es podria millorar introduint fluctuacions més realistes.

8. Bibliografia

[1] Greenpeace

<https://es.greenpeace.org/es/trabajamos-en/cambio-climatico/>

[2] Preu electricitat

<https://www.businessinsider.es/11-paises-europa-donde-factura-luz-cara-535867>

[3] Rubí Brilla

<https://www.rubi.cat/es/ayuntamiento/proyectos-estrategicos/rubibrilla>

[4] Mapa consum

<http://rubiennergia.300000kms.net/>

[5] Mapa producció

<https://visors.icgc.cat/sostenibilitat/#/visor/Rubi>

[6] salari mínim 2019

<http://www.salariominimo.es/2019.html>

[7] Salari mig

<https://datosmacro.expansion.com/mercado-laboral/salario-medio/espana#:~:text=El%20salario%20m%C3%A9dio%20en%20Espa%C3%B1a,c%C3%A1culo%20suponiendo%2012%20pagas%20anuales.>

[8] Real decret

<https://www.boe.es/boe/dias/2019/04/06/pdfs/BOE-A-2019-5089.pdf>

[9] Energia sol

<https://es.weatherspark.com/y/33741/Clima-promedio-en-Grado-Espa%C3%B1a-durante-todo-el-a%C3%B1o>

[10] Energia sol

<https://energeticafutura.com/blog/cuanta-energia-se-puede-extraer-de-la-radiacion-solar/>

9. Annex

9.1 Configuració de l'entorn

A continuació es farà una breu explicació de com s'ha de configurar un ordinador Windows per tal que funcioni correctament el model implementat per a aquest TFG.

Netlogo:

En primer lloc ens descarregarem la versió 6.X.X de netlogo, podem trobar l'instal·lador a la pàgina oficial de netlogo: <https://ccl.northwestern.edu/netlogo/download.shtml>

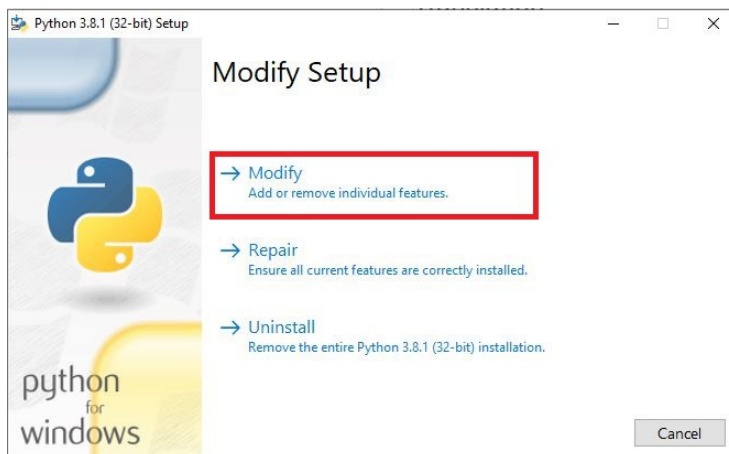
Configuració python 3 i netlogo

Un cop instal·lat el netlogo s'haurà de configurar per tal que pugui utilitzar l'extensió de python 3.

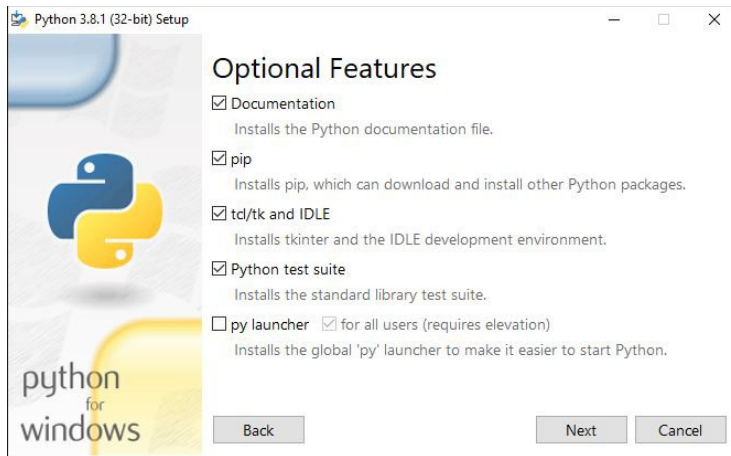
En primer lloc instal·larem python 3, que el podem trobar en el següent enllaç:

<https://www.python.org/downloads/>.

Un cop instal·lat python 3 s'haurà d'instal·lar la "feature" pip, per tal de fer-ho se li donarà doble clic a l'instal·lador de python i es seleccionarà l'opció de modificar la instal·lació.



Es seleccionaran les següents opcions:



Un cop instal·lat la “feature” pip de python, obrirem un terminal i escriurem la comanda:
pip install scikit-image per instal·lar les llibreries necessàries de python.

Finalment anirem al següent enllaç: <https://github.com/NetLogo/NetLogo/wiki/Extensions>, ens hem de descarregar l’extensió de python. Un cop descarregada descomprimem la carpeta a la següent ruta: *NetLogo 6.0.4\app\extensions*.