

# **Escola Universitària Politécnica de Mataró**

Centre adscrit a:



**UNIVERSITAT POLITÈCNICA  
DE CATALUNYA**

**Grau en Enginyeria Electrònica i Automàtica**

**Estudi sobre la generació dels polígons de Voronoi en la delimitació  
d'àrees d'influència en l'espai urbà.**

**Memòria**

**Isaac Flores**

**PONENT: Miquel Roca**

**PRIMAVERA 2014**



**TecnoCampus  
Mataró-Maresme**

## **Resum**

El present treball de final de grau tracta de donar solucions i establir noves eines d'anàlisi territorial per l'ajuntament de Mataró y per el Centre de Coneixement Urbà del Tecnocampus Mataró.

En aquest treball es crea una aplicació amb la qual es podran generar zones d'influència per els diversos centres proveïdors de servei en Mataró, així com Escoles, CAPS, parades de bus etc... Aquest sistema s'implementarà amb un mètode geomètric anomenat Voronoi, que permet calcular àrees d'influència amb una gran eficàcia i exactitud.

## **Resumen**

El presente Trabajo de final de grado trata de dar soluciones y establecer nuevas herramientas de análisis territorial para el ayuntamiento de Mataró y para el Centro de Conocimiento Urbano del Tecnocampus Mataró.

En este trabajo se crea una aplicación con la cual se podrán generar zonas de influencia para los diversos centros proveedores de servicio en Mataró tales como escuelas, CAPS, paradas de bus etc... Este sistema se implementará con un método geométrico llamado Voronoi, que permite calcular áreas de influencia con una gran eficacia y exactitud.

## **Abstract**

The present minor thesis seeks to provide solutions and implement new spatial analysis tools for Mataró city council and also for Centre de Coneixement Urbà from Tecnocampus Mataró.

In this project I have created an application which will be able to generate influence zones for several centers service providers in Mataró such as Schools, CAPS, Bus stops... This system will be implemented with a geometric method named Voronoi, which will enable to determinate influences zones in a really effective and accurate way.

## Índex.

Índex de figures.....	III
Índex de taules.....	VIII
Glossari de termes. ....	IX
1. Introducció.....	1
1.1. Objectius .....	2
1.1.1. Finalitat .....	2
1.1.2. Propòsit .....	2
1.1.3. Objecte .....	2
1.1.4. Abast .....	2
1.2. Planificació final del treball .....	2
1.3. Antecedents del projecte .....	4
2. Antecedents del CCU i del Geomedia .....	9
2.1. CCU .....	9
2.2. Introducció als SIG .....	11
2.3. GeoMedia Professional .....	12
2.4. Creació de zones d'influència.....	16
3. Voronoi i els seus mètodes.....	23
3.1. Introducció als polígons de Voronoi.....	23
3.2. Aplicacions .....	24
3.3 Algoritmes de generació .....	26
3.3.1. Bowyer-Watson.....	26
3.3.2. Fortune .....	28

## II

3.3.3 Lloyd .....	29
4. Mètodes externs per calcular Voronoi. ....	33
5. Explicació de la aplicació.....	45
5.1. Algoritme i codi inicial.....	45
5.2. Formulari .....	53
5.3. Mòduls.....	57
5.3.1. Mòdul Obtenció de coordenades .....	57
5.3.2. Modul Voronoi_code .....	59
6. Utilització de l'aplicació .....	68
6.1. Carregar l'aplicació al Geomedia.....	68
6.2. Exemples de la aplicació .....	71
6.2.1. Escoles bressol .....	72
6.2.2. CAPS .....	78
6.2.3. Parades de bus .....	82
6.2.4. Consultes .....	85
7. Conclusions .....	90
8. Referències .....	91

## Índex de figures.

Fig. 1.1 Localització geogràfica de Mataró .....	1
Fig. 1.2. Mapa tèrmic de la península ibèrica .....	4
Fig. 1.3. Mapa tèrmic dels Estats Units .....	5
Fig. 1.4. Exemple de diagrama de Voronoi .....	5
Fig. 1.5. Triangulació de Delaunay .....	6
Fig. 1.6. Applet java dels Poligons de voronoi .....	6
Fig.1.7. Software de la implementació dels polígons de Voronoi .....	7
Fig 1.8. Conca hidogràfica distribuïda segons regions de Voronoi .....	8
Fig. 2.1. Nova connexió.....	12
Fig. 2.2. Dades nova connexió.....	13
Fig. 2.3. Buscar cadastre12.....	13
Fig. 2.4. Afegir entrades a llegenda.....	14
Fig. 2.5. Visualització de "illes". .....	14
Fig. 2.6. Eina de rotació del mapa.....	14
Fig. 2.7. Sistema de coordenades del Workspace.....	15
Fig. 2.8. Paràmetres de projecció.....	15
Fig. 2.9. Finestra mòdul creació de zones de influència.....	17
Fig. 2.10. Zones d'influència circulars sobre els CAPS.....	18
Fig. 2.11. Agregació per obtenir els habitants.....	19
Fig. 2.12. Distància màxima de la escola al segment del graf.....	19
Fig. 2.13. Dades del exemple de graf sobre CAPS.....	20
Fig. 2.14. Zones d'influència graf sobre CAPS.....	21

Fig. 3.1. Exemple d'un diagrama de Voronoi .....	22
Fig. 3.2. 1er pas Bowyer-Watson .....	25
Fig. 3.3. 2n pas Bowyer-Watson .....	25
Fig. 3.4. 3r pas Bowyer-Watson .....	26
Fig. 3.5. 4t pas Bowyer-Watson .....	26
Fig. 3.6. 2n pas Bowyer-Watson .....	26
Fig. 3.7. 2n pas Bowyer-Watson .....	26
Fig. 3.8. simulació algoritme fortune .....	27
Fig. 3.9. Algoritme de fortune finalitzat.....	27
Fig. 3.10. Primera iteració Lloyd.....	31
Fig. 3.11. Segona iteració Lloyd.....	31
Fig. 3.12. Tercera iteració Lloyd .....	31
Fig. 3.13. Quinzena iteració Lloyd .....	31
Fig 4.1. Crear nova entitat de tipus area (1/4). .....	32
Fig 4.2. Crear nova entitat de tipus area (2/4). .....	33
Fig 4.3. Crear nova entitat de tipus area(3/4). .....	33
Fig 4.4. Crear nova entitat de tipus area(4/4). .....	34
Fig 4.5. Exportar capa en format shapefile.....	35
Fig 4.6. Selecció de coordenades geogràfiques.....	36
Fig 4.7. Les dues capes mostrades en pantalla. ....	36
Fig 4.8. "feature info tool".....	37
Fig 4.9. Opció polígons de Voronoi. ....	37
Fig 4.10. Selecció de límits. ....	38
Fig 4.11. Polígons de voronoi en Global Mapper.....	38

Fig 4.12. Allargar extrems polígons(1/4).....	39
Fig 4.13. Allargar extrems polígons(2/4).....	39
Fig 4.14. Allargar extrems polígons(3/4).....	39
Fig 4.15. Allargar extrems polígons(4/4).....	40
Fig 4.16. Exportar l'arxiu shapefile a globalMapper (1/4).....	41
Fig 4.17. Exportar l'arxiu shapefile a globalMapper (2/4).....	41
Fig 4.18. Exportar l'arxiu shapefile a globalMapper (3/4).....	41
Fig 4.19. Exportar l'arxiu shapefile a globalMapper (4/4).....	42
Fig. 4.20. Polígons de Voronoi en Qgis i Matlab. ....	43
Fig. 5.1. Exemple Voronoi amb 50 punts i coordenades aleatòries.....	48
Fig. 5.2. Exemple Voronoi amb 8 punts i coordenades aleatòries.....	49
Fig. 5.3. Exemple Voronoi amb 3 punts en les coordenades desitjades. ....	50
Fig. 5.4. Llistat de mòduls i formulari del projecte. ....	50
Fig. 5.5. Formulari de la aplicació creada.....	51
Fig. 5.6. Desplegable per seleccionar l'entitat puntual. ....	51
Fig. 5.7. Codi relacionat amb el desplegable de selecció d'entitat puntual.....	52
Fig. 5.8. Desplegable per seleccionar l'àrea delimitant.....	52
Fig. 5.9. Codi relacionat amb el desplegable de selecció d'àrea delimitant.....	52
Fig. 5.10. Desplegable per seleccionar la sortida de les línies dels polígons. ....	53
Fig. 5.11. Codi relacionat amb el desplegable del desplegable de selecció de sortida....	53
Fig. 5.12. Botó calcular Voronoi. ....	53
Fig. 5.13. Codi que va darrere del botó "Calcular Voronoi". ....	54
Fig. 5.14. Botó sortir. ....	54
Fig. 5.15. Codi que va darrere del botó sortir. ....	54

Fig. 5.16. Generar.dll.....	65
Fig. 6.1. Contingut del nou projecte. ....	66
Fig. 6.2. Contingut carpeta "scr". ....	66
Fig. 6.3. Contingut de la carpeta "bin". ....	67
Fig. 6.4. Instal·lació nou mòdul creat. ....	67
Fig. 6.5. Instal·lació satisfactòria del mòdul. ....	68
Fig. 6.6. Addició de les connexions necessàries.....	68
Fig. 6.7. Icona per defecte del nostre mòdul. ....	69
Fig. 6.8. Finestra mòdul.....	69
Fig. 6.9. Agregació de entrades de llegenda necessàries.....	70
Fig. 6.10. Escoles bressol i el terme municipal mostrats a la pantalla. ....	70
Fig. 6.11. Propietats de les escoles bressol.....	71
Fig. 6.12. Omplir desplegable del mòdul per escoles bressol.....	71
Fig. 6.13. Polígons de Voronoi sobre les escoles bressol.....	72
Fig. 6.14. Intersecció espacial entre línies i terme municipal.....	73
Fig. 6.15. Voronoi final sobre les escoles bressol mitjançant intersecció.....	73
Fig. 6.16. Procés dividir entitats (1/3). ....	73
Fig. 6.17. Procés dividir entitats (2/3). ....	74
Fig. 6.18. Procés dividir entitats (3/3). ....	74
Fig. 6.19. Voronoi resultant sobre escoles bressol mitjançant divisió d'entitats. ....	75
Fig. 6.20. Voronoi resultant sobre escoles amb la capa d'illes. ....	75
Fig. 6.21. Voronoi resultant sobre escoles amb la capa "ortofoto". ....	76
Fig. 6.22. Caps i terme municipal mostrats en pantalla.....	76
Fig. 6.23. Omplir desplegable del mòdul per escoles bressol.....	77



Fig. 6.24. Diagrama de Voronoi sobre CAPS.....	77
Fig. 6.25. Voronoi final sobre els CAPS mitjançant intersecció espacial .	78
Fig. 6.26. Voronoi final sobre els CAPS amb la capa d'illes .....	78
Fig. 6.27. Voronoi final amb la vista Aérea.....	79
Fig. 6.28. Parades de bus i terme municipal mostrats en pantalla. ....	80
Fig. 6.29. Omplir desplegable del mòdul per Parades de bus. ....	80
Fig. 6.30. Diagrama de Voronoi per Parades de bus. ....	81
Fig. 6.31. Intersecció espacial entre línies i terme municipal.....	81
Fig. 6.32. Polígons Voronoi resultants sobre parades de bus mitjançant intersecció espacial.....	82
Fig. 6.33. Polígons Voronoi resultants amb la capa d'illes.....	82
Fig. 6.34. Polígons Voronoi resultants amb la vista Aérea.....	83
Fig. 6.35. CEIPcon i centre urbà en pantalla. ....	84
Fig. 6.36. Consulta espacial. ....	84
Fig. 6.37. Nova consulta espacial representada a la llegenda.....	85
Fig. 6.38. selecció de la consulta creada al desplegable. ....	85
Fig. 6.39. Omplir desplegable del mòdul per escoles bressol.....	86
Fig. 6.40. Polígons de Voronoi per la consulta creada. ....	86
Fig. 6.41. Polígons de voronoi finals per la consulta creada mitjançant dividir entitats.	87

## **Índex de taules.**

Taula 1.1. Resum de tasques del projecte.....	3
---	---

## **Glossari de termes.**

SIG	Sistemes d'informació geogràfica
CAP	Centre de Autogestió Primària de salut
VB	Visual Basic 6.0
TFG	Treball Final de Grau
CCU	Centre de Coneixement Urbà
ESO	Educació Secundària Obligatoria
CEIPcon	Centres d'Educació Infantil i Primària Concertats
EUPMT	Escola Universitària Politècnica de Mataró
GTC	Graf de Tram de Carrers
PFC	Projecte Final de Carrera
UPM	Universitat Politècnica de Madrid
BBDD	Bases de Dades
ZI	Zones d'Influència
EB	Escoles Bressol

## 1. Introducció

Mataró és una ciutat catalana situada al litoral mediterrani. És la capital i la ciutat més gran de la comarca del Maresme. Es troba a uns 30 quilòmetres al nord de la ciutat de Barcelona.

Aquest municipi conté un total de 124.099 habitants, segons dades del 2013, que estan repartits per una extensió de 22,53 Km<sup>2</sup>, el que comporta una densitat de 5.508,17 habitants/m<sup>2</sup>.



*Fig. 1.1 Localització geogràfica de Mataró*

Segons aquestes dades podem afirmar que ens trobem en una ciutat amb un gran volum d'habitants en un territori bastant extens també. Per tant, l'estudi de la població i dels serveis que proporciona la ciutat es quelcom molt important per l'ajuntament de Mataró.

La administració pública té en la seva possessió diferents bases de dades on es guarda gran quantitat de documentació dels habitants de Mataró. Aquesta informació ha estat essencial a l'hora de donar forma l'actual projecte.

L'elaboració d'aquest projecte s'ha portat a terme gràcies a la col·laboració paral·lela de l'ajuntament de Mataró conjuntament amb el Centre de Coneixement Urbà, això m'ha ofert la possibilitat de disposar de tres convenis de cooperació educativa, repartits en dos anys de treball, per tal de endinsar-me en els sistemes d'informació geogràfica que han servit per familiaritzar-me amb l'entorn de treball i que consegüentment m'ha facilitat la realització del present projecte.

## **1.1. Objectius**

### **1.1.1. Finalitat**

Estudiar l'ús dels diagrames de Voronoi en la delimitació de zones d'influència a la ciutat de Mataró .

### **1.1.2. Propòsit**

Implementar un algoritme per la generació de polígons de Voronoi dins de l'entorn del SIG en que està treballant el CCU.

### **1.1.3. Objecte**

Estudiar diferents algoritmes i escollir-ne un per codificar una aplicació que treballi sobre l'entorn del programa GeoMedia Professional.

### **1.1.4. Abast**

Construir un mòdul en VB6 per l'entorn GeoMedia on s'implementi el mètode Voronoi per generar particions de l'espai urbà de Mataró i aplicar-ho a la la generació de Zones d'Influència de centres proveïdors de serveis com ara Centres Cívics, Escoles, CAPS etc. Tot i que també s'utilitzaran altres alternatives per fer-ho de forma indirecta com són Global Mapper i Qgis.

## **1.2. Planificació final del treball**

Finalment ha canviat una mica el format exposat en l'avantprojecte respecte a les parts del projecte i la durada d'aquestes. A última hora va haver-hi un problema amb l'algoritme amb el qual s'estava treballant, ja que utilitzava una tècnica que no es podia traslladar al Geomedia, vaig haver de demanar prorroga fins al juny, ja que vaig haver de començar de nou buscant un altre algoritme que funcionés de forma correcta i es pogués adaptar totalment al Geomedia, a pesar de tot una part de la feina ja estava feta, ja que moltes funcions i l'interface del mòdul ja estaven creats i han estat de gran utilitat de cara a la creació del mòdul final. A

més del coneixements adquirits en aquest primer quadrimestre sobre VB, ja que vaig començar quasi de 0.

Finalment les tasques del projecte han quedat de la següent manera:

Nom de la tasca	Durada	Principi	Final
Recerca de l'algorithm en VB	15 dies	1/02/2014	15/02/2014
Adaptar el codi VB	13 dies	15/02/2014	28/02/2014
Crear l'interface del nou mòdul	3 dies	5/03/2014	7/03/2014
Implementar el mòdul al Geomedia amb diverses funcions	46 dies	10/03/2014	25/04/2014
Proves i millores constants de la nova aplicació	21 dies	26/04/2014	15/05/2014
Redacció de la memòria	30 dies	1/05/2014	30/05/2014
Lliurament de la memòria	1 dia	2/06/2014	2/06/2014

*Taula 1.1. Resum de tasques del projecte.*

### 1.2.1. Planificació de la memòria

- En el segon capítol "antecedents del CCU i del Geomedia" es comenten en primera instància els treballs final de carrera que s'han realitzat a l'escola amb l'ajuda del CCU. A continuació es fa una breu introducció dels SIG i de com funciona el Geomedia.
- En el tercer capítol, es fa una breu introducció dels polígons de Voronoi i s'expliquen els diferents algoritmes que es poden utilitzar per generar-los.
- En el quart, s'explica una manera de calcular els polígons de forma indirecta utilitzant programes pont i s'esmenten dues més.
- En el cinquè, es comença a explicar el funcionament de l'aplicació des de el punt de vista del programador, comentant les parts del codi que es consideren més importants.

- En el sisè, s'explica l'aplicació des de un punt de vista més pràctic amb diferents exemples, de manera que sigui més fàcil d'entendre per part de l'usuari.
- Finalment s'esmenten les conclusions finals del projecte i les referències de enllaços webs i llibres utilitzats per realitzar el treball.

### 1.3. Antecedents del projecte

En la història recent s'han realitzat diversos projectes utilitzant els diagrames de Voronoi. A continuació podem veure alguns exemples de projectistes que han utilitzat aquesta tècnica. [1]

a) Representació gràfica de Superfícies mitjançant mapes tèrmics. (Projectista Ginés Rodríguez Gavilán. Universitat politècnica de Madrid)

L'objecte d'aquest treball d'aquest projectista va ser el desenvolupament d'una aplicació informàtica per la representació gràfica de Superfícies a partir d'un conjunt de punts de coordenades X,Y,Z utilitzant els següents mètodes d'interpolació:

- 1) Polígons de Voronoi: Mètode basat en la distància euclidiana i en la presunció lògica de que tots els punts propers a un punt mostrejat se semblaran a aquest.
- 2) Xarxes de triangles irregulars: la superfície real es pot aproximar a una superfície matemàtica discreta formada per Superfícies elementals planes, en aquest cas triangles, que es defineixen a partir de conjunts de punts donats.

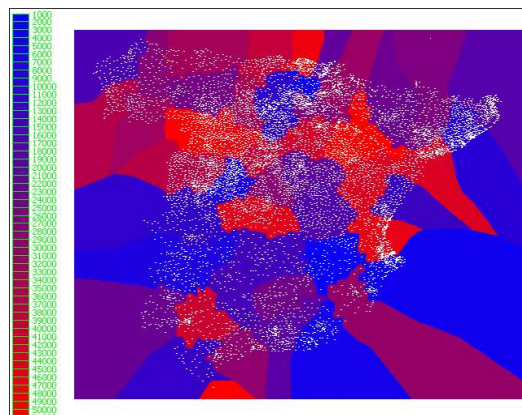


Fig. 1.2. Mapa tèrmic de la península ibèrica

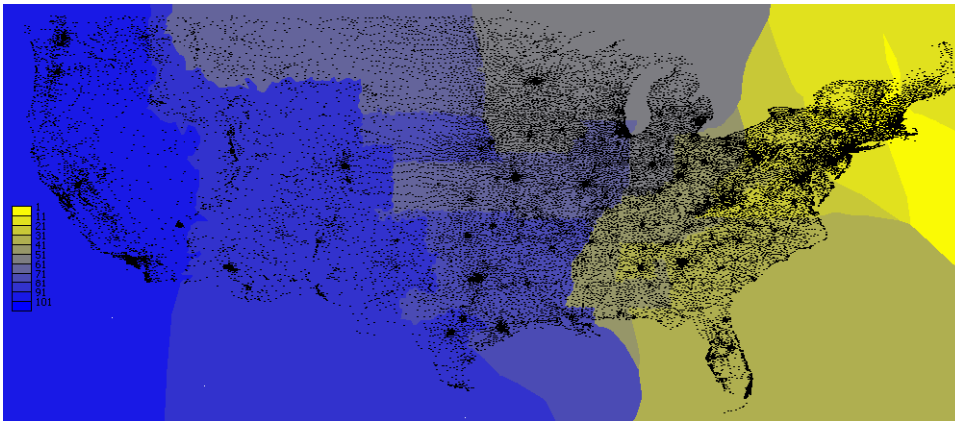


Fig. 1.3. Mapa tèrmic dels Estats Units

Aquests mètodes estan basats en dos estructures Fundamentals dins de la geometria computacional com son la “triangulació de Delaunay”, i la seva estructura dual, el “Diagrama de Voronoi”.

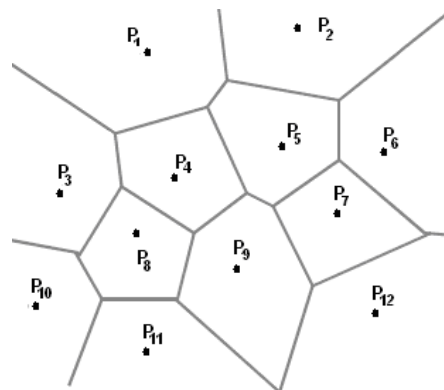


Fig. 1.4. Exemple de diagrama de Voronoi

Definim el diagrama de Voronoi del conjunt P com una subdivisió del pla en n regions, una per cada punt de P, amb la propietat de que un punt “q” pertany a la regió corresponent al punt “p” si la distancia euclidiana de “q” a “p” és menor que la distancia fins a qualsevol altre punt del conjunt P.

Creem aquest polígons entre si, traçant mediatris dels segments de unió. Les interseccions d’aquestes mediatris determinen una sèrie de polígons en un espai bidimensional al voltant d’un conjunt de punts de control, de manera que el perímetre dels polígons generat sigui equidistant als punts veïns i designant la seva àrea t’influència. (Triangulació de Delaunay)



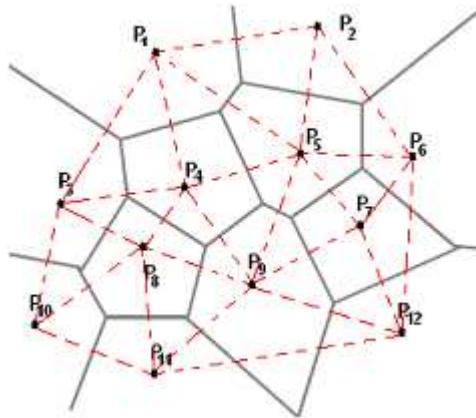


Fig. 1.5. Triangulació de Delaunay

b) Desenvolupament de Software per generar polígons de Voronoi

Tal mostra la figura 1.6, podem interactuar amb els polígons de Voronoi, en un applet Java creat per Manuel Mabellanes ( responsable del departament de matemàtiques aplicades de la UPM). [2]

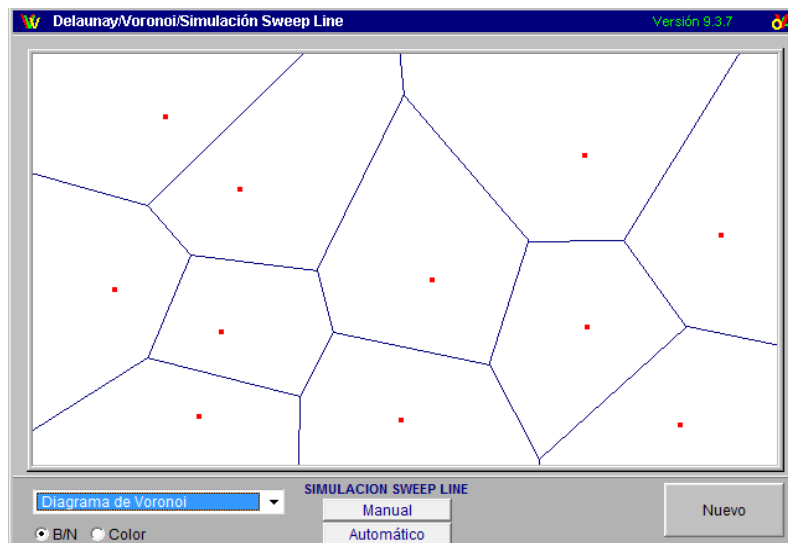
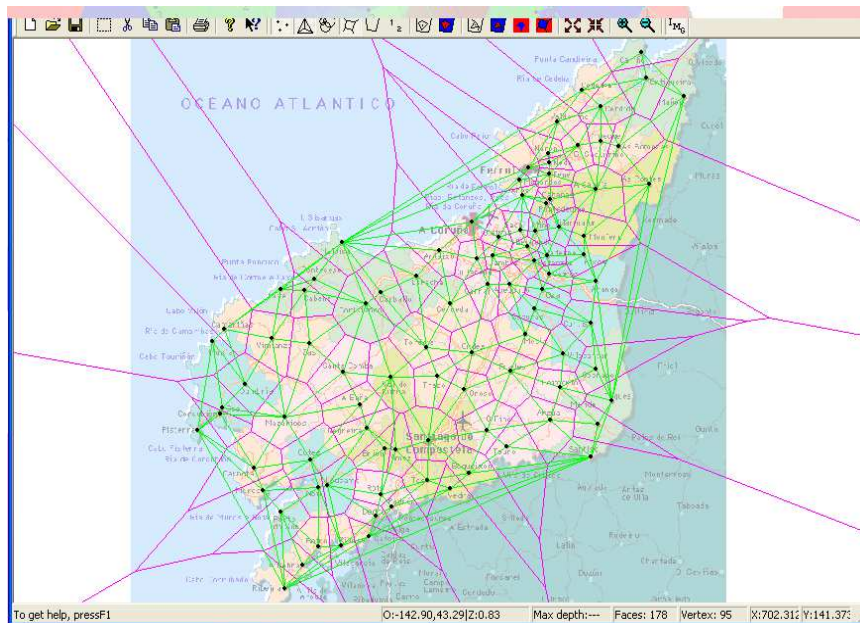


Fig. 1.6. Applet java dels Poligons de voronoi

També va crear un software en c++ per la implementació d'aquests polígons amb l'ajuda d'Alfredo De Las Vegas. En la figura 1.7 podem veure una mostra d'aquest software. [3]



*Fig.1.7. Software de la implementació dels polígons de Voronoi*

c) Competència matemàtica i digital: Aplicacions pràctiques dels conjunts de Voronoi realitzat per Martha Ivón Cárdenas i Daniel López. [4]

L'objectiu d'aquest projecte és iniciar als alumnes de l'ESO a l'estudi dels conjunts de Voronoi, mitjançant els quals els hi ajudi a desenvolupar la matemàtica computacional contextualitzada, a desenvolupar la representació gràfica i l'expressió simbòlica, a la utilització de diferents eines informàtiques, com a estratègia integradora i d'aprenentatge de diferents conceptes matemàtics que es treballen normalment en tres blocs separats (Geometria, Resolució de Problemes i Probabilitat i Estadística), i

també, a estudiar l'aplicació d'aquests diagrames a casos pràctics, tot desenvolupant competències bàsiques tals com: les competències comunicatives i lingüístiques, aprendre l'ús d'eines i aplicacions informàtiques, el tractament de la informació i competència digital, la competència matemàtica, la competència d'aprendre a aprendre, l'autonomia i iniciativa personal i el coneixement i interacció amb el món físic. Totes aquestes competències que cal assolir permetran que l'alumnat integri els

aprenentatges relacionats amb els conjunts de Voronoi i que s'hi pugui estudiar mitjançant aplicacions informàtiques la seva aplicació de manera efectiva en situacions i contextos reals.

Com a resultat, es pretén potenciar el treball cooperatiu com una metodologia que permet ensenyar i avaluar per competències.

Les activitats dissenyades en aquest projecte tenen una finalitat clara i amb un sentit i uns continguts significatius per al nostre alumnat, i on ells han de resoldre una situació diferent amb cada activitat que els farà actuar de manera conscient, crítica i responsable.

Aquest projecte està integrat dins d'un curs de Moodle de l'Institut. Els alumnes amb els quals s'ha realitzat l'activitat, tenen un nivell de 4t de l'ESO. Aquesta activitat es va portar a terme durant el segon trimestre, durant les sessions que formen part del bloc de Geometria.

#### d) Determinació de la precipitació mitjana a la conca. [5]

Les dimensions d'una conca hidrogràfica son molt variades i les precipitacions també varien en el temps i l'espai. Per prendre en consideració aquestes diversitats i conèixer el comportament de les pluges, per realitzar aquest important procés, es sol utilitzar el mètode dels polígons de Voronoi, com podem veure a la següent imatge.



Fig 1.8. Conca hidrogràfica distribuïda segons regions de Voronoi.

S'uneixen les estacions adjacents amb segments de recta, i després es construeixen bisectrius perpendiculars a cada segment, estenent-les fins que s'interceptin, formant polígons irregulars. Aquesta metodologia és objectiva i dona resultats satisfactoris si es consta de una xarxa adequada de pluviòmetres. No es recomanable en àrees muntanyoses, ja que els coeficients no reflecteixen de cap manera els efectes altitudinals, i tampoc es recomana la seva aplicació per derivar promitjos regionals en el cas de tempestes locals i intenses.

## 2. Antecedents del CCU i del Geomedia

### 2.1. CCU

El Centre de Coneixement Urbà (CCU-TCM) és una de les línies de R+D+i dins dels projectes de transferència de coneixement del Tecnocampus Mataró-Maresme. Aquest és un grup d'investigació especialitzat en l'anàlisi i la gestió urbana i l'acústica utilitzant els SIG. Té com a objectiu el desenvolupament del coneixement agregat en tots els àmbits de la gestió municipal i supramunicipal. Els seu responsable tècnic és el professor Miquel Roca i Cisa, el qual té una experiència dilatada en la seva direcció.

Aquest estudi forma part i utilitza totes les eines realitzades durant anys per el professor Miquel Roca i Cisa amb diferents tasques fetes en projectes final de carrera, estudis paral·lels, tasques de suport amb els serveis municipals de l'Ajuntament i realitzant programes o mòduls especialitzats en tasques d'estudi de la població.

Cal fer esment a les tasques realitzades per el CCU prèviament a aquest projecte, en especial als projectes final de carrera realitzats per estudiants amb el suport del CCU. En primer lloc, el projecte final de carrera de la Meritxell González Navarro a la primavera 2004, anomenat "*Utilització d'un sistema d'informació geogràfica per analitzar la població de Mataró*". En aquest primer projecte, es va situar el punt de partida dels mòduls utilitzant els complements necessaris del Visual Basic amb la finalitat de automatitzar tasques i introduir una línia de treball encara existent actualment. Posteriorment, el PFC d'en David Martin Perelló Mires a la tardor 2005, anomenat "*Aplicació pel Centre de Coneixement Urbà de Mataró*" apareix una revisió i ampliació de la feina feta per la Meritxell, en aquest treball s'incorpora el Mòdul *OperacionsGM* destinat a l'ús comú de tots els programadors i amb la utilitat de simplificar operacions que es realitzen amb certa periodicitat en els diferents projectes, aquesta eina s'utilitza actualment en el present projecte.

També es milloren els que fins el moment havien estat els mòduls per a la creació de la taula resum, la creació de zones d'influència circular i el mòdul de càlcul de proximitat d'entitats a partir de distàncies a partir de la seva àrea d'influència. Aquests mateixos mòduls també van ser estudiats en una vessant més general per la projectista Míriam del Castillo, que a la tardor

del 2006, va realitzar un PFC anomenat “Estudi dels objectes del GeoMedia Pro”. Aquest últim treball conté un clar desenvolupament de les feines fetes fins el moment. Aquell mateix any, el projectista Gerard Mas Martínez a la convocatòria de tardor, va realitzar el seu PFC titulat “Estudi de l'accessibilitat de la via pública de Mataró utilitzant un SIG” .

Aquests avanços van trobar un punt d'inflexió amb l'aparició del graf de Trams de Carrer (GTC) com a entitat per l'estudi de la població. El graf és el conjunt d'objectes anomenats nodes i una selecció d'entitats lineals que uneixen els nodes, anomenades trams.

A partir de la incorporació del GTC, el CCU va mostrar un salt qualitatiu alhora de fer els càlculs a partir dels segments, i va donar pas al TFC “Eines d'anàlisi territorial per l'ajuntament de Mataró” realitzat per l'Eric Belando a la primavera de 2011, on ens presenta un nou mòdul que permet calcular distàncies a peu entre entitats d'origen i de destí.

Un avenç molt important pel servei Web Map del CCU va arribar l'any 2009 on l'Albert Ruiz amb el seu treball final de carrera “Servei Web pel CCU amb GeoMedia WebMap 6.1” va realitzar un estudi detallat sobre el mètode generació d'entorns web mitjançant la família d'aplicacions GeoMedia d'Intergraph. Aquest estudi demostra que és viable el desenvolupament d'un Geoportal web en un entorn Intranet amb GeoMedia WebMap 6.1. A més, es van implementar operacions d'anàlisi pel CCU semblants a les actuals en escriptori per a GeoMedia, desenvolupades en projectes per la Meritxell González i en David Perelló.

En l'actualitat, amb el present projecte es vol tornar a entrar en el tema de les zones d'influència dels serveis que ofereix Mataró. Però utilitzant un altre mètode totalment diferent, ja que el mòdul fet fins aleshores oferia la possibilitat de calcular zones d'influència per els diferents serveis relacionant la quantitat de població amb les places que oferia el mateix servei.

En aquest cas es vol fer quelcom totalment oposat, i les zones d'influència es crearien exclusivament per distància euclidiana (distància en línia recta), i utilitzant el mètode de Voronoi aconseguiríem que totes les zones fossin exactes. D'aquesta forma podríem saber de forma precisa quins serveis ens queden més a prop de la zona en que vivim.

Seguidament es farà una introducció dels SIG (Sistemes d'informació geogràfica) i del Geomedia, per acabar amb un manual del mòdul de zones d'influència existents fins al moment en el CCU.

## **2.2. Introducció als SIG**

Els Sistemes de Informació Geogràfica (SIG)

Un sistema d'informació geogràfica (SIG, o GIS en el seu acrònim anglès) és un sistema informàtic capaç d'integrar, emmagatzemar, editar, analitzar, compartir i mostrar informació amb referències geogràfiques. De forma més genèrica, es tracta d'una eina a partir de la qual els usuaris poden fer consultes interactives (cerques definides per l'usuari), analitzar la informació espacial, i editar-ne les dades.

Qualsevol objecte d'un SIG està descrit fonamentalment per:

- La seva localització absoluta sobre la superfície terrestre.
- La descripció de la seva posició i forma geomètrica en dos o tres dimensions.
- Informació alfa numèrica que el qualifica i classifica.
- Per les relacions espacials amb els altres elements.

Els sistemes d'informació geogràfica poden utilitzar-se en la recerca, la gestió de recursos, la planificació de despeses, l'avaluació de l'impacta ambiental, els plans de desenvolupament, la cartografia o la planificació de rutes. Per exemple, pot permetre el càlcul dels temps de resposta en cas d'emergència a conseqüència d'un desastre natural, o bé per a detectar espais que calgui protegir de la contaminació.

Un sistema d'informació geogràfica pot convertir informació digital, que no es troba en forma de mapa, de manera que pugui ser tractada i visualitzada de forma integrada.

El SIG que utilitzarem per a l'estudi dels mòduls es el Intergraph GeoMedia Professional.

## 2.3. GeoMedia Professional

Abans de començar l'anàlisi de les dades cal verificar que el nostre programari està correctament instal·lat i per això caldrà comprovar que hem seguit satisfactòriament tots els passos de la instal·lació del GeoMedia de 'l'Annex I'. Un cop hem comprovat que tot és correcte, començarem introduint els primers passos per utilitzar el programa com a eina de consulta per a l'usuari.

El software dels sistemes d'informació geogràfica, utilitza un magatzem d'informació que el permet vincular unes coordenades amb un objecte, aquest magatzem s'anomena connexió, ja que és el vincle entre el GeoMedia i l'objecte que es necessari per atribuir a aquest qualsevol tipus de característica.

En el nostre cas, les connexions les farem mitjançant tres bases de dades en format Access que treballaran paral·lelament amb el GeoMedia. S'hauran de realitzar aquestes connexions prèviament abans de indicar qualsevol ordre al programa.

Obrim la finestra Almacen > Conexiones... Pitjar a 'Nueva'



Fig. 2.1. Nova connexió.

'Tipo de conexión' Access, omplir el nom de la connexió '**Cadastre**' i pitjar a 'Examinar'

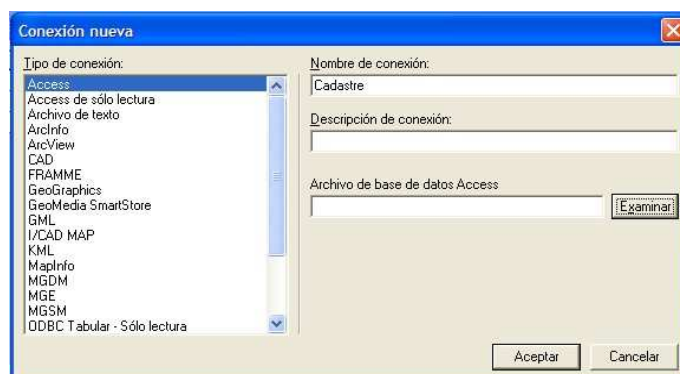


Fig. 2.2. Dades nova connexió.

Seleccionar 'Cadastr12.mdb', i pitjar 'Abrir' i accepteu aquesta nova connexió

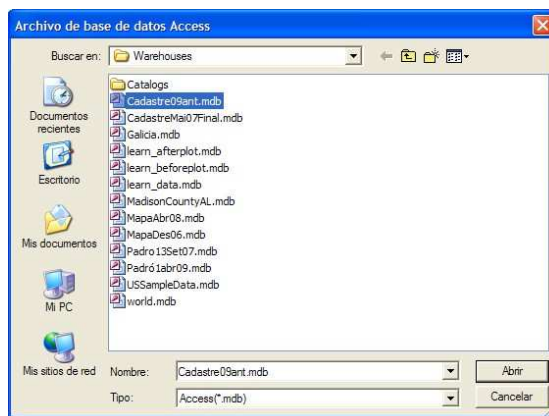


Fig. 2.3. Buscar cadastr12.

Hem realitzar aquesta operació dues vegades més, les connexions que haurem de formular són 'MapaMataro' i 'Padro'. I els arxius d'Access són 'MapaFeb12.mdb' i 'Padró1Mai12.mdb'.

**NOTA:** És molt important que alhora de anomenar les connexions respectem els mateixos caràcters utilitzats pel programador (com són les majúscules i la falta d'accents), de no fer-ho la connexió no tindrà cap tipus de servei.

Un cop realitzades les connexions, estem en condicions de llançar alguna consulta als mòduls, no obstant, comprovarem que les connexions s'hagin realitzat correctament i que els paràmetres que conté el GeoMedia són els òptims per al nostre tipus de consultes.

Obrim la finestra **Leyenda > Agregar Entradas a la leyenda...**, seleccionem MapaMataro > Illes i pitjar acceptar.



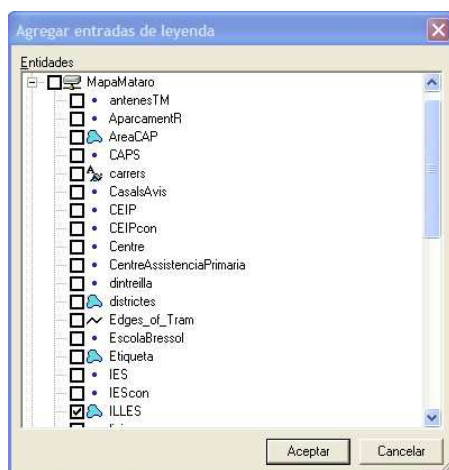


Fig. 2.4. Afegir entrades a llegenda.

Apareixeran les entitats gràfiques ‘illes’ que formen el Mapa de la ciutat de Mataró.



Fig. 2.5. Visualització de "illes".

El Mapa és possible que aparegui en una posició inclinada, poc atractiva alhora de treballar amb resultats, es possible rotar el Mapa amb una de les barres de utilitats que conté el GeoMedia, concretament la barra “col·locació i edició” i l’eina Rotar. Concretament, la rotació òptima és a 42°.

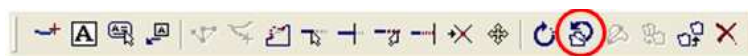


Fig. 2.6. Eina de rotació del mapa.

Finalment, abans d'introduir el funcionament dels mòduls, hem d'ajustar el sistema de coordenades sobre l'espai.

Obrim la finestra **Ver > Sistemas de coordenadas del GeoWorkspace**, seleccionem la pestanya '**Espacio de la proyección**', seleccionem sistema de projecció UTM i pitjar sobre '**Parámetros de proyección...**'

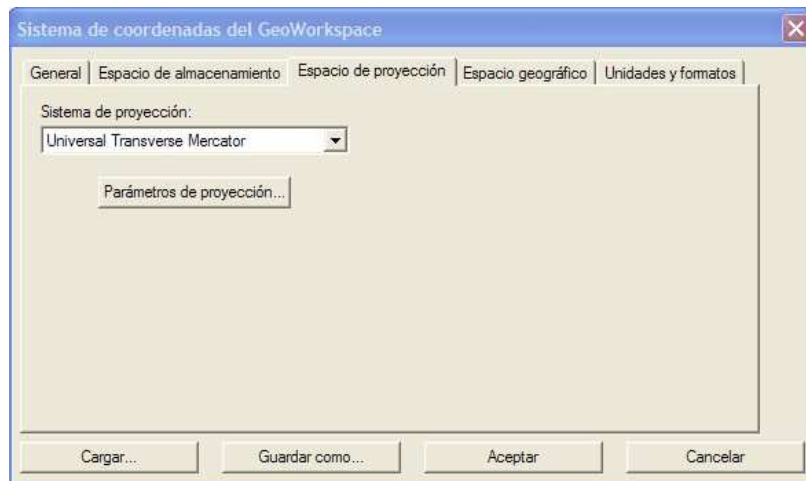


Fig. 2.7. Sistema de coordenadas del Workspace.

Substituir els paràmetres que té per defecte, per els de la següent imatge i pitjar 'Aceptar'.



Fig. 2.8. Parámetros de proyección.

Un cop introduïts els paràmetres, guardeu la configuració al GeoWorkspace.

## 2.4. Creació de zones d'influència.

El CCU fa servir el SIG com una eina per dur a terme anàlisis complexes on intervenen varies BBDD de la ciutat simultàniament. Per fer aquests anàlisis complexes hem de programar uns mòduls( extensions del Geomedia), ja que només amb el programa estàndard no és suficient. A continuació tenim un manual de com funciona l'únic mòdul per determinar zones d'influència creat anteriorment pel CCU.

Aquest mòdul necessita de una taula resum( veure capítol 2 del annex III) per realitzar la creació d'àrees d'influència, l'objectiu d'aquest apartat és presentar visualment una zona d'influència a sobre del mapa, la seva utilitat està lligada a una entitat del Mapa de Mataró que utilitzem per treballar, com per exemple els Instituts d'Educació Primària, els CAPS, Cassals, etc...

L'objectiu d'aquesta eina es basa en relacionar la quantitat de població d'un territori amb la capacitat que té aquest en donar suport a un servei.

Aquest mòdul realitza el càlcul de les Àrees d'Influència d'una entitat proveïdora de servei amb el camp **NPlaces**, que és tracte de l'aforament d'un centre per a realitzar un servei. A més a més de les ZI circulars també calcula les ZI a través del graf de trams de carrer (GTC) incloent els trams finals o parcials.

Perquè la informació que es presenti sigui compatible amb l'espai que treballem, haurem d'activar a la llegenda, la capa de Illes, i la localització puntual sobre la que es treballi.

Pel que fa al mètode de treball d'aquest mòdul, de mateixa manera que com l'anterior, podrà treballar tant per illes, com per parcel·les, cosa que haurem de tindre en compte alhora de generar una taula a partir de les mateixes entitats que treballem en aquesta segona consulta.

Seguidament, definirem la entitat on volem generar l'àrea d'influència, aquestes entitats es troben en el desplegable, MapaMataro. Generalment, treballarem amb entitats puntuals, no obstant hi ha altres entitats que estan definides al voltant d'una àrea. Hem de tindre en compte si treballem amb les entitats definides com a àrees, ja que les possibilitats de que la zona d'influència acapari més territori són més elevades.

Pel que fa al Radi sobre el càlcul de l'entorn, el definirem en funció del nombre de places i el territori que s'hi aproxima a l'entorn de l'entitat, també podem relacionar aquest radi, amb un nombre d'iteracions, amb un alt nombre d'iteracions, el programa tardarà més en calcular les possibilitats del radi de la ZI, però es una forma d'aproximar amb major exactitud l'àrea d'influència.

En la alguns de casos, la cobertura que s'estudia és del 100%, però també hi ha altres situacions, com per exemple l'estudi de la ZI de les Escoles de bressol, on encara que prèviament a la taula resum hem introduït uns marges de [0-2]anys, la cobertura per aquests serveis pot ser més baixa que la del 100%, per tant la podem modificar sempre que convingui. El programa també ens dona la possibilitat de aportar un número dels habitants no coberts.

El resultat de la creació de ZI el podem treure per una classe d'entitat o bé ens podem limitar a mostrar-lo en la llegenda.

Cada vegada que es modifiqui algun tipus de radi, haurem de pitjar 'recalcular Radi inicial', i cada vegada que volem introduir una segona o tercera consulta, es recomanable pitjar 'Esborrar la darrera consulta', amb la precaució que esborra la última entitat que s'ha introduït en llegenda.

Exemplifiquem aquesta operació per corroborar el seu funcionament:

Obrim la finestra **Mòduls CCU > Creació d'àrees d'influència(circular/graf)**. Utilitzant la continuació de l'exemple anterior, que vàrem crear una taula resum per habitants de [0-120]anys, comprovarem la zona de cobertura dels CAPS per a tota aquesta població.

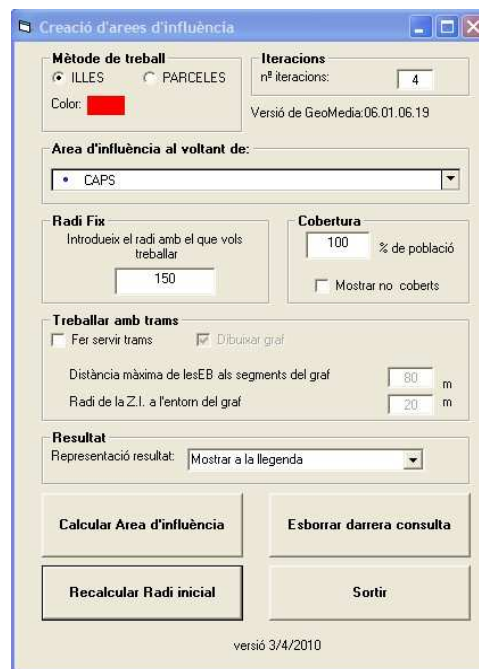


Fig. 2.9. Finestra mòdul creació de zones de influència.

Omplim la finestra amb les dades que apareixen a la fig. 2.9. Mètode de treball: Illes, n° iteracions:4, Al voltant de: CAPS, Radi: 150m, Cobertura 100%, pitgem 'Recalculer radi inicial', i pitgem 'Calcular Area d'influència'. El Resultat d'aquesta darrera acció, pot tardar uns quants segons, en funció de l'ordinador sobre el qual es faci l'operació. Seguidament, es llançarà una zona d'influència sobre el mapa de Mataró com podem veure a la següent imatge.



Fig. 2.10. Zones d'influència circulars sobre els CAPS.

Com podem observar, apareixen unes zones circulars amb diferents radis, aquesta raó es degut al nombre d'habitants que hi viu al voltant del CAP i també al n° de places que el CAP sosté.

Per entendre millor l'afectació de cadascuna de les entitats, podem estudiar de forma independent cadascun dels cassos. Si pitgem a sobre d'una de les zones d'influència, ens apareixerà un diàleg com el següent:

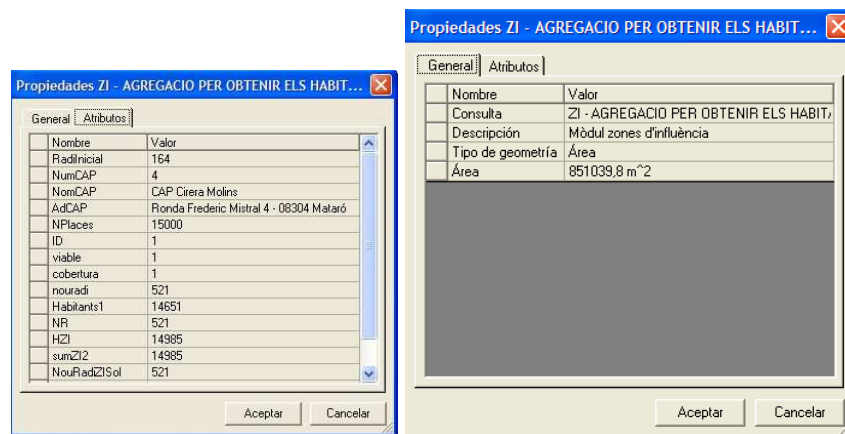


Fig. 2.11. Agregació per obtenir els habitants.

Aquest desplegable ens dona informació sobre la posició del l'entitat, l'àrea en la qual dona servei, etc..

Una altre manera de presentar el resultat de les ZI que permet aquest mòdul és la utilització dels trams, aquesta operació requereix d'un càlcul més complex, però alhora més visual.

Abans d'exemplificar-ho amb el resultat anterior, per poder fer una valoració entre els dos resultats, haurem d'entendre dos conceptes.

Primer, haurem d'introduir la distància màxima de les EB als segments del graf.

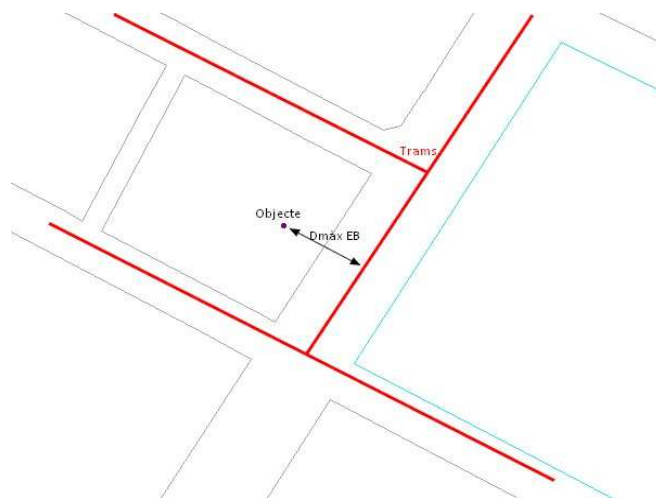


Fig. 2.12. Distància màxima de la escola al segment del graf.

Tal com es veu a la imatge la distància EB màxima que definim es la distància entre l'objecte entitat i el tram.

Aquesta distància no caldrà indicar-la en cas de no sigui necessari, la distància EB que el mòdul dóna per defecte són 80m i el radi de gruix de la ZI 20m, com podem observar a la següent figura. 2.12.

L'altre concepte és el Radi de la ZI a l'entorn del graf, que representa el gruix que li donem a les línees que representem del graf.

Exemplifiquem aquesta operació per corroborar el seu funcionament:

Obrim la finestra **Mòduls CCU > Creació d'àrees d'influència(circular/graf)**. Utilitzant la continuació de l'exemple anterior, que vàrem crear una taula resum per habitants de [0-120]anys, comprovarem la zona de cobertura dels CAPS per a tota aquesta població.

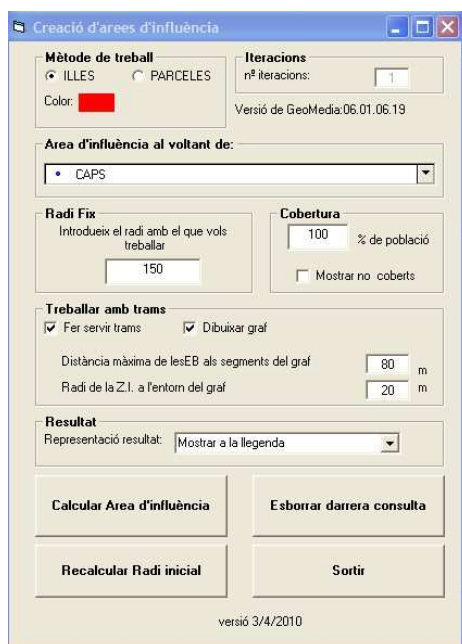


Fig. 2.13. Dades del exemple de graf sobre CAPS.

Omplim la finestra amb les dades que apareixen a l'anterior imatge, mètode de treball: Illes, nº iteracions:4, Al voltant de: CAPS, Radi: 150m, distància EB:80m Radi ZI:20m, Cobertura 100%, pitgem 'Recalcular radi inicial', i pitgem 'Calcular Area d'influència'. El Resultat d'aquest darrera acció, pot tardar uns quants segons, en funció de l'ordinador sobre el qual es faci l'operació. Seguidament, es llançarà una zona d'influència sobre el mapa de Mataró com podem veure a la següent imatge.

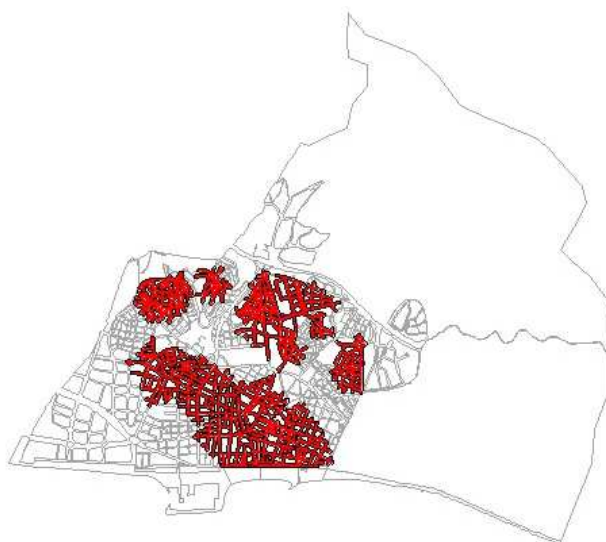


Fig. 2.14. Zones d'influència graf sobre CAPS



Aquesta vegada, a diferència del càlcul anterior, hem generat dos entrades a la llegenda, una el graf i l'altre la ZI CAPS, que es la que conté el radi indicat de 20metres.

Es important, ahora de registrar un resultat per calcular un àrea d'influència, com indiquem a la figura. 2.13., haurem de definir si aquest resultat es mostri a la llegenda o bé, es guardi en classe d'entitat.

### 3. Voronoi i els seus mètodes

#### 3.1. Introducció als polígons de Voronoi

Els diagrames de Voronoi nomenats en honor al matemàtic Georgy Voronoi, són una construcció geomètrica que permet construir una partició del pla euclidià. Aquests objectes també van ser estudiats pel meteoròleg nord-americà Alfred H. Thiessen del que prenen el nom alternatiu de polígons de Thiessen i pel matemàtic Gustav Lejeune Dirichlet del que prenen el nom de tessellació de Dirichlet.

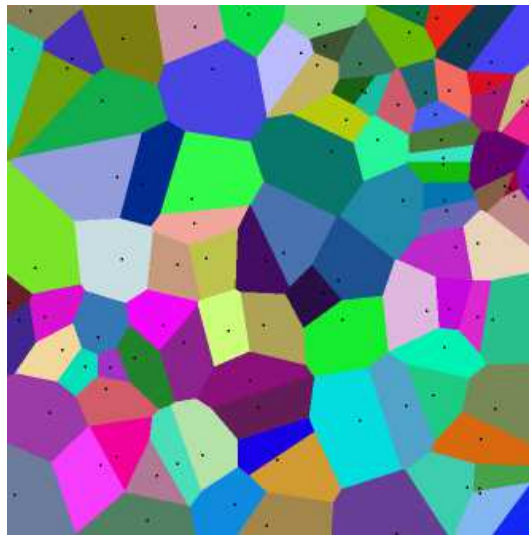


Fig. 3.1. Exemple d'un diagrama de Voronoi

Els *diagrames de Voronoi* són un dels mètodes d'interpolació més simples, basats en la distància euclidiana, sent especialment apropiada quan les dades són qualitatives. Es creen en unir els punts entre si, traçant les mediatris dels segments d'unió. Les interseccions d'aquestes mediatris determinen una sèrie de polígons en un espai bidimensional al voltant d'un conjunt de punts de control, de manera que dintre de cada polígon o regió la distància a un punt de control associat és sempre menor que a qualsevol altre punt de les altres regions.

$$l_2 = d[(a_1, a_2), (b_1, b_2)] = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2} \quad (3.1)$$

## 3.2. Aplicacions

- En epidemiologia , diagrames de Voronoi es poden utilitzar per correlacionar les fonts de les infeccions en les epidèmies . Una de les primeres aplicacions dels diagrames de Voronoi va ser implementat per John Snow per estudiar el brot de còlera de Broad Street 1854 a Soho , Anglaterra . Ell va demostrar la correlació entre àrees al mapa de Londres utilitzant una bomba d'aigua en particular, i les àrees amb la majoria de les morts pel brot .
- Una estructura de dades la ubicació del punt es pot construir a la part superior del diagrama de Voronoi per tal de respondre a les consultes de veïns més pròxims, en els que un vol trobar l'objecte que està més a prop d'un punt de consulta donada. Consultes de veïns més propers tenen nombroses aplicacions . Per exemple , un podria voler trobar l'hospital més proper , o l'objecte més similar a una base de dades . Una aplicació de grans dimensions és de quantificació vectorial , comunament utilitzat en la compressió de dades . Aquesta seria la funció utilitzada per el present projecte.
- En geometria , els diagrames de Voronoi es poden utilitzar per trobar el major cercle buit enmig d'un conjunt de punts , i en un polígon que tanca ; per exemple la construcció d'un nou supermercat a la mesura del possible de tots els existents , estirat en una determinada ciutat .
- Els diagrames de Voronoi juntament amb els diagrames de Voronoi més allunyat de punt s'utilitzen per algoritmes eficients per calcular la rodonesa d'un conjunt de punts .
- L'enfocament de Voronoi també es posa a bon ús en l'avaluació de la circularia / rodonesa mentre s'avalua el conjunt de dades a partir d'una màquina de mesura de coordenades .
- En la creació de xarxes , diagrames de Voronoi es poden utilitzar en les derivacions de la capacitat d'una xarxa sense fils .
- En climatologia , diagrames de Voronoi s'utilitzen per calcular la precipitació d'una àrea , sobre la base d'una sèrie de mesuraments de punt . En aquest ús , que es denominen generalment com polígons de Thiessen .

- En ecologia, els diagrames de Voronoi s'utilitzen per estudiar els patrons de creixement dels boscos i les copes dels arbres , i també poden ser útils en el desenvolupament de models de predicció dels incendis forestals .
- En la química computacional , cèl·lules de Voronoi definits per les posicions dels nuclis en una molècula s'utilitzen per calcular les càrregues atòmiques . Això es fa usant el mètode de densitat de deformació de Voronoi .
- En física de polímers , diagrames de Voronoi es poden utilitzar per a representar volums lliures de polímers .
- En la ciència de materials , microestructures policristal·lines en aliatges metàl·lics es representen comunament usant mosaics de Voronoi. En la física d'estat sòlid , la cel·la de Wigner - Seitz és la teselació de Voronoi d'un sòlid.
- En la mineria , els polígons de Voronoi s'utilitzen per estimar les reserves de materials valuosos , minerals o altres recursos . Sondatges exploratoris s'utilitzen com el conjunt de punts en els polígons de Voronoi .
- En gràfics per ordinador , els diagrames de Voronoi s'utilitzen per calcular els patrons de geometria trencament /fracturament 3D. També s'utilitza per generar processalment textures orgàniques.
- En la navegació de robots autònoms, els diagrames de Voronoi s'utilitzen per trobar rutes clares . Si els punts són obstacles , a continuació, les vores de la gràfica seran les rutes més allunyats dels obstacles.
- En l'aprenentatge de màquina, els diagrames de Voronoi s'utilitzen per fer classificacions 1- NN.

### 3.3 Algoritmes de generació

En aquest apartat s'estudiaran diversos algoritmes que permeten la generació dels Polígons de Voronoi, tot i que l'elegit per la implementació de la aplicació serà un altre comentat en el capítol 5 més senzill a l'hora de codificar.

#### 3.3.1. Bowyer-Watson

En la geometria computacional, l'algoritme Bowyer-Watson és un mètode per calcular la triangulació de Delaunay (es fa referència en el primer capítol, en l'apartat d'antecedents del projecte) d'un conjunt finit de punts en qualsevol nombre de dimensions. L'algoritme es pot utilitzar per obtenir un diagrama de Voronoi dels punts, que és la versió dual de la triangulació de Delaunay com és mostra a la figura 1.5 i s'explica al punt 1.3 del present projecte.

L'algoritme Bowyer-Watson és un algorisme incremental. Funciona mitjançant l'addició de punts, de un en un, utilitzant una triangulació de Delaunay vàlida del subconjunt de punts desitjats. Després de cada inserció, qualsevol triangle que en la seva circumferència inscrita contingui el nou punt és esborrat, deixant un forat poligonal en forma d'estrella, que és llavors re-triangulat utilitzant el nou punt.

L'algoritme es coneix de vegades simplement com l'algorisme Bowyer o l'algorisme de Watson. Adrian Bowyer i David Watson van inventar de forma independent l'un de l'altre, a la vegada, i cada publicar un article sobre ell en la mateixa edició de The Computer Journal.

Els passos per reproduir l'algoritme serien: [6]

1er pas: insertar un node en un super-triangle tancat    2n pas: insertar un segon node

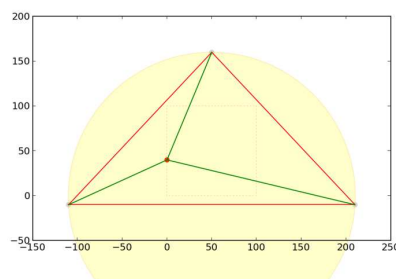


Fig. 3.2. 1er pas Bowyer-Watson

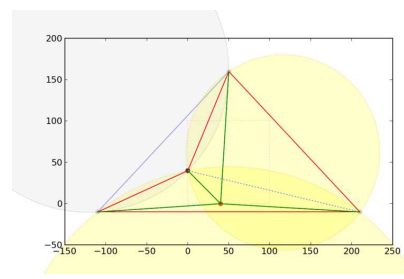


Fig. 3.3. 2n pas Bowyer-Watson

3r pas: insertar un tercer node

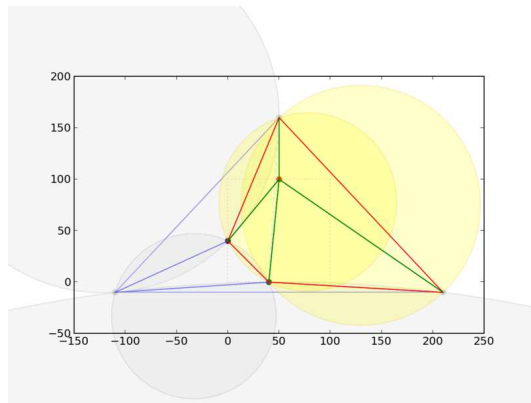


Fig. 3.4. 3r pas Bowyer-Watson

4t pas: insertar un quart node

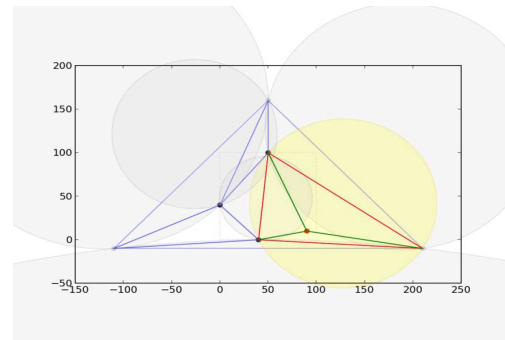


Fig. 3.5. 4t pas Bowyer-Watson

5è pas: insertar el cinquè i últim node

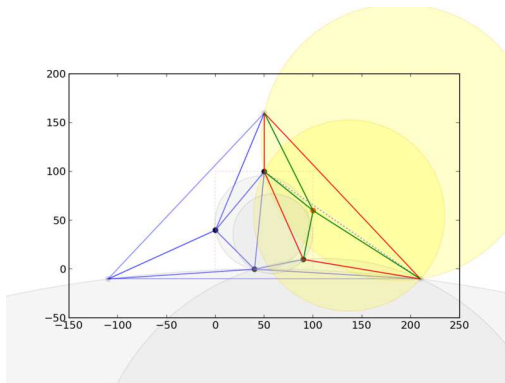


Fig. 3.6. 5è pas Bowyer-Watson

6è pas: eliminar les arestes del super triangle

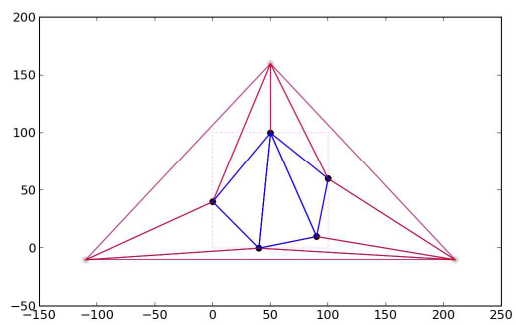


Fig. 3.7. 6è pas Bowyer-Watson

### 3.3.2. Fortune

L'algoritme de Fortune és un algoritme de línia d'escombrat per generar un diagrama de Voronoi a partir d'un conjunt de punts en un pla utilitzant  $O(n \log n)$  i  $O(n)$  en l'espai. Va ser publicat originalment per Steven Fortune el 1986 en el seu article "A sweepline algorithm for Voronoi diagrams." [7]

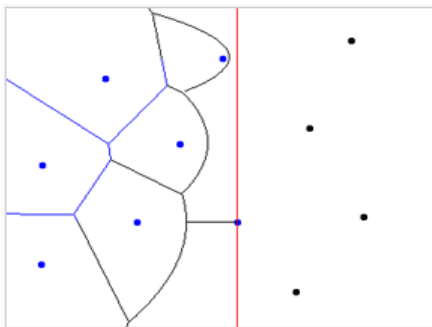


Fig. 3.8. simulació algoritme fortune

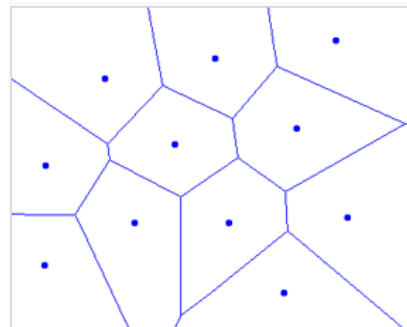


Fig. 3.9. Algoritme de fortune finalitzat

#### Descripció de l'algoritme

L'algorisme manté tant una línia d'escombrat i una línia de la platja, que totes dues es mouen a través del pla que l'algorisme progressa. La línia d'escombrat és una línia recta, que per convenció, suposem que és vertical i es mou d'esquerra a dreta en el pla. En qualsevol moment durant l'algoritme, els punts d'entrada a l'esquerra de la línia d'escombrat s'han incorporat en el diagrama de Voronoi, mentre que els punts a la dreta de la línia d'escombrat no s'han considerat encara. La línia de la platja no és una línia, sinó una corba complexa a l'esquerra de la línia d'escombrat compost per trossos de paràboles que divideix la porció del pla en el qual el diagrama de Voronoi pot ser conegut (3.2), independentment que altres punts poden ser a la dreta de la línia d'escombrat, des de la resta del pla. Per a cada punt de l'esquerra de la línia d'escombrat, es pot definir una paràbola dels punts equidistants d'aquest punt i de la línia d'escombrat; la línia de platja és el límit de la unió d'aquestes paràboles. Com la línia d'escombrat progressa, els vèrtexs de la línia de platja, en el qual dues paràboles creuen, tracen les vores del diagrama de Voronoi. La línia de la platja avança mantenint cada paràbola base exactament a mig camí entre els punts inicialment escombrats per la línia d'escombrat, i la nova posició de la línia d'escombrat, també anomenada sweep line.

L'algorisme manté com a estructures de dades d'un arbre de cerca binària que descriu l'estructura combinatòria de la línia de la platja, i una cua de prioritat llistat de possibles esdeveniments futurs que podrien canviar l'estructura de la línia de platja. Aquests esdeveniments inclouen l'addició d'una altra paràbola a la línia de platja (quan la línia d'escombrat creua un altre punt d'entrada) i l'eliminació d'una corba de la línia de platja (quan la línia d'escombrat és tangent a un cercle a través d'uns tres punts d'entrada les paràboles forma segments consecutius de la línia de platja). Cadascun d'aquests esdeveniments pot ser prioritzat per la coordenada  $x$  de la línia d'escombrat en el punt que el succés ocorre. L'algorisme en si llavors consisteix a retirar repetidament el proper esdeveniment de la cua de prioritat, la recerca dels canvis que l'esdeveniment provoca en la línia de platja, i l'actualització de les estructures de dades. Com que hi ha  $O(n)$  per processar els esdeveniments (cada un està associat amb alguna característica del diagrama de Voronoi) i  $O(\log n)$  temps per processar un esdeveniment (cadascun format per un nombre constant d'arbre de cerca binària i operacions de la cua de prioritat) on el temps total és  $O(n \log n)$ .

$$B_{j^i} = y = \frac{1}{2(p_{j^i} - l_y)} (x^2 - 2p_{j^i}x + p_{j^i}^2 + p_{j^i}^2 - l_y^2) \quad (3.2)$$

### 3.3.3 Lloyd

En ciències de la computació i enginyeria elèctrica, l'algorisme de Lloyd, també conegut com iteració Voronoi o relaxació, és un algoritme creat per Stuart P. Lloyd per trobar conjunts uniformement espaiats de punts en subconjunts d'espais euclidians, i particions d'aquests subconjunts en cèl·lules convexes de mida uniforme.

Aquest mètode troba repetitivament el centroide de cada conjunt de la partició, i després re-particions l'entrada d'acord a quin d'aquests centroides està més a prop. No obstant això, l'algorisme de Lloyd, es diferencia respecte altres mètodes semblants en que la seva entrada és una regió geomètrica contínua en lloc d'un conjunt discret de punts. Per tant, en tornar a la partició de l'entrada, l'algorisme de Lloyd utilitza diagrames de Voronoi i no simplement determinar el centre més proper a cada un d'un conjunt finit de punts com fan altres mètodes.

Encara que el algoritme pot aplicar més directament al pla euclidià, algoritmes similars també es poden aplicar als espais de dimensions superiors o als espais amb altres mètriques no



euclidianes . L'algorisme de Lloyd es pot utilitzar per construir aproximacions properes als diagrames de centroides de Voronoi, que poden ser utilitzat per a la quantificació , tramat , i puntejat . Altres aplicacions de l'algorisme de Lloyd inclouen el suavitzat de malles de triangles en el mètode d'elements finits .

### *Descripció de l'algorisme*

L'algorisme de Lloyd comença per una col·locació inicial d'un nombre  $k$  de llocs puntuals en el domini d'entrada . En les aplicacions de suavitzat de malla ; aquests serien els vèrtexs de la malla a ser suavitzada ; en altres aplicacions que es poden col·locar a l'atzar , o per la intersecció d'una malla triangular uniforme de la mida apropiada amb el domini d'entrada .

Després executa repetidament la següent etapa de relaxació :

- El diagrama de Voronoi dels espais  $k$  es calcula .
- Cada cèl·lula del diagrama de Voronoi s'integra i el centroide es calcula .
- Cada espai es mou llavors al centroide de la seva cel·la de Voronoi .

A causa dels algorismes de construcció del diagrama de Voronoi pot ser altament difícil , especialment per a les entrades de dimensió superior a dos , els passos de calcular aquest diagrama i la recerca dels centroides de les seves cèl·lules es poden aproximar per una discretització adequada en què , per a cada cel·la , el lloc més proper es determina , després de les quals el centre de gravetat per a la cel·la d' un lloc s'aproxima per la mitjana dels centres de les cel·les de la quadrícula que se li assignin . Alternativament , els mètodes de Monte Carlo es poden utilitzar , en el qual els punts de mostra a l'atzar es generen d'acord amb alguna distribució de probabilitat subjacent fixa , assignats al espai més proper , i es fa promig per aproximar el centroide per a cada espai.

### *Convergència*

Cada vegada que es realitza una etapa de relaxació, els punts es deixen en una lleugera distribució: punts estretament espaiats es mouen més lluny, i punts àmpliament espaiats es mouen més a prop entre ells. En una dimensió, aquest algorisme s'ha demostrat que convergeix a un diagrama centroidal de Voronoi.

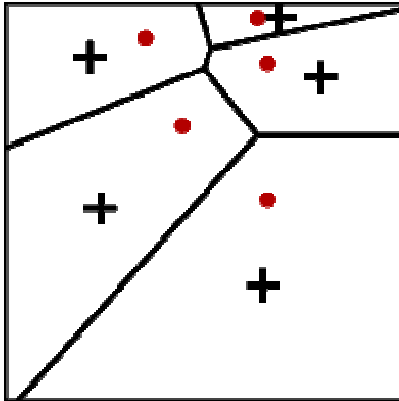
L'algoritme convergeix lentament o, a causa de les limitacions en la precisió numèrica, pot no convergir. Per tant, les aplicacions del món real de l'algorisme de Lloyd normalment s'aturen quan la distribució és "prou bo". Un criteri de terminació comú és aturar quan la distància màxima moguda per qualsevol lloc en una iteració cau per sota d'un llindar preestablert.

### *Aplicacions*

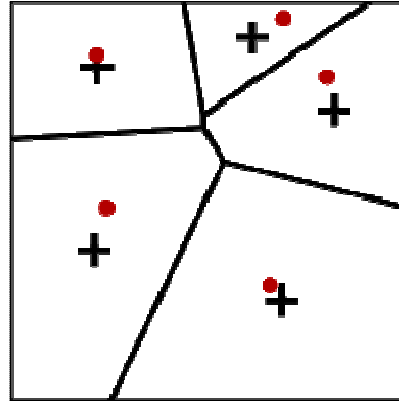
Mètode de Lloyd es va utilitzar originalment per quantificació escalar, però està clar que el mètode s'estén per quantificació vectorial també. Com a tal, s'utilitza àmpliament en les tècniques de compressió de dades en teoria de la informació. El mètode de Lloyd s'utilitza en gràfics per ordinador perquè la distribució resultant té característiques de soroll blau, el que significa que hi ha pocs components de baixa freqüència que poden ser interpretades com artefactes. És particularment ben adaptat per recollir les posicions de mostra per a la interpolació. L'algorisme de Lloyd també s'utilitza per generar dibuixos de punt en l'estil de punteig. En aquesta aplicació, els centroides poden ser ponderats d'acord amb una imatge de referència per produir il·lustracions puntejades que coincideixin amb una imatge d'entrada.

En el mètode d'elements finits, un domini d'entrada amb una geometria complexa es divideix en elements amb formes més simples; per exemple, els dominis de dues dimensions (ja siguin subconjunts del pla euclidià o superfícies en tres dimensions) són sovint dividits en triangles. És important per a la convergència dels mètodes d'elements finits que aquests elements es van formar així; en el cas de triangles, sovint es prefereixen elements que són triangles equilàters gairebé. L'algorisme de Lloyd es pot utilitzar per suavitzar una malla generada per algun altre algorisme, movent els seus vèrtexs i canviant el patró de connexió entre els seus elements per tal de produir triangles que són més pròxims a equilàters. Aquestes aplicacions utilitzen típicament un menor nombre d'iteracions de l'algorisme de Lloyd, aturant-se a la convergència, per tal de preservar altres característiques de la malla, com ara les diferències en la mida de l'element en diferents parts de la malla. En contrast amb un mètode diferent de suavitzat, suavitzat de Laplace (en el qual els vèrtexs de malla es mouen a la mitjana de les posicions dels seus veïns), l'algorisme de Lloyd pot canviar la topologia de la malla, el que porta als elements més quasi-equilàters, així com evitar els problemes amb embolics que poden sorgir amb el suavitzat de Laplace. No obstant això, la suavització de Laplace es pot aplicar de forma més general a malles amb elements no triangulars.

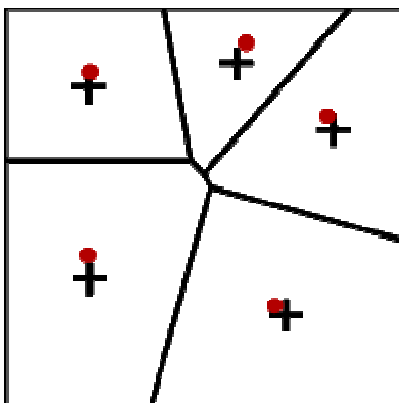
Exemple de l'algoritme de Lloyd's. Es mostra el diagrama de Voronoi dels punts actuals en cada iteració. Els signes positius marquen els centroides de les regions de Voronoi.



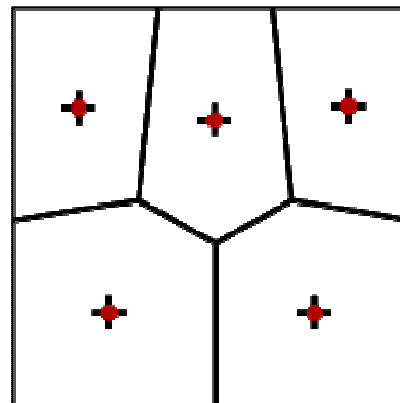
*Fig. 3.10. Primera iteració Lloyd*



*Fig. 3.11. Segona iteració Lloyd*



*Fig. 3.12. Tercera iteració Lloyd*



*Fig. 3.13. Quinzena iteració Lloyd*

En la última imatge, els punts estan molt a prop dels centroides de les regions de Voronoi. Per tant l'algoritme funciona de manera correcta i bastant exacta.

Com es pot apreciar en la figura 3.13, els punts estan totalment en el centre de cada regió de Voronoi inicial i han quedat modificades respecte als nous punts. Per tant, s'ha completat amb èxit el diagrama de Voronoi centroide.

## 4. Mètodes externs per calcular Voronoi.

Per això es necessita utilitzar un programa que tingui la opció de crear els polígons de Voronoi de forma directa. I que pugui treballar amb format .shape(format necessari per exportar al Geomedia). He trobat dos programes que compleixen aquests requisits, tot i que segur que n'hi ha més. Aquests dos programes són el Global Mapper i el Qgis i el matlab. Dintre d'aquests crec que el més senzill d'utilitzar és el Global Mapper.

A continuació en faré un petit tutorial de com crear els polígons de voronoi en el Geomedia amb l'ajuda del Global Mapper, utilitzant com a classe d'entitat els CAPS.

Per traçar els polígons de Voronoi, caldran dues capes, la capa A, que serà de tipus àrea, que serà responsable de marcar els límits, i la capa B, que seran els punts, els quals seran separats pels polígons.

En el següent exemple es mostra la realització completa de l'estudi dels polígons de Voronoi tenint com a punts els CAPS de Mataró, i com a límits, tot el perímetre de Mataró (terme municipal).

Pas 1: Aconseguir les dues capes.

Per aconseguir la primera capa, de tipus àrea, haurem de crear una nova entitat, en aquest cas en MapaMataro (Almacén->Definición de clase de entidad).

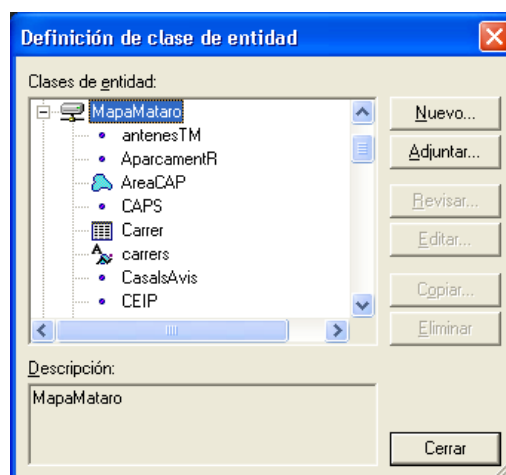


Fig 4.1. Crear nova entitat de tipus area (1/4).

A la finestra que s'obre, es marca la connexió on es vol crear la nova entitat, en aquest cas MapaMataro, i es prem "Nuevo"

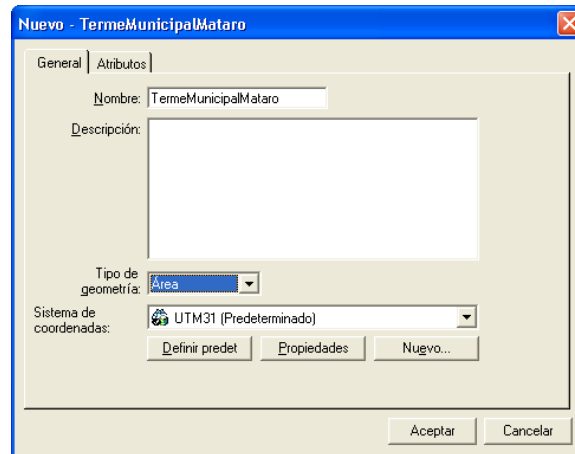


Fig 4.2. Crear nova entitat de tipus area (2/4).

Es posa el nom que es vulgui a la nova entitat, i se selecciona Tipus Àrea. El sistema de coordenades serà el que tinguem definit, no s'ha de modificar.

Un cop creada l'entitat, no tindrà cap registre. S'ha de fer una inserció (Insertar->Entidad)

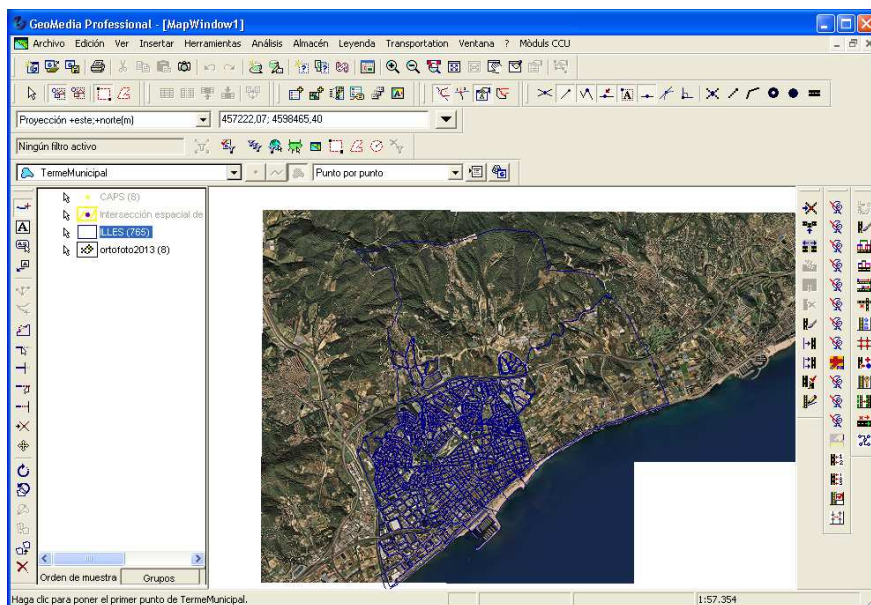


Fig 4.3. Crear nova entitat de tipus area(3/4).

Es selecciona l'entitat que s'acaba de crear, just a sobre de la llegenda. Aleshores el cursor passarà a ser una creu, amb la que punt per punt podem anar seleccionant tot el contorn del

terme municipal. Per fer més precisa la selecció del contorn, és recomanable visualitzar les illes, i anar marcant a sobre del seu contorn (la zona exterior).

Un cop marcat tot el perímetre, ja es tindria la primera capa, hauria de quedar així:

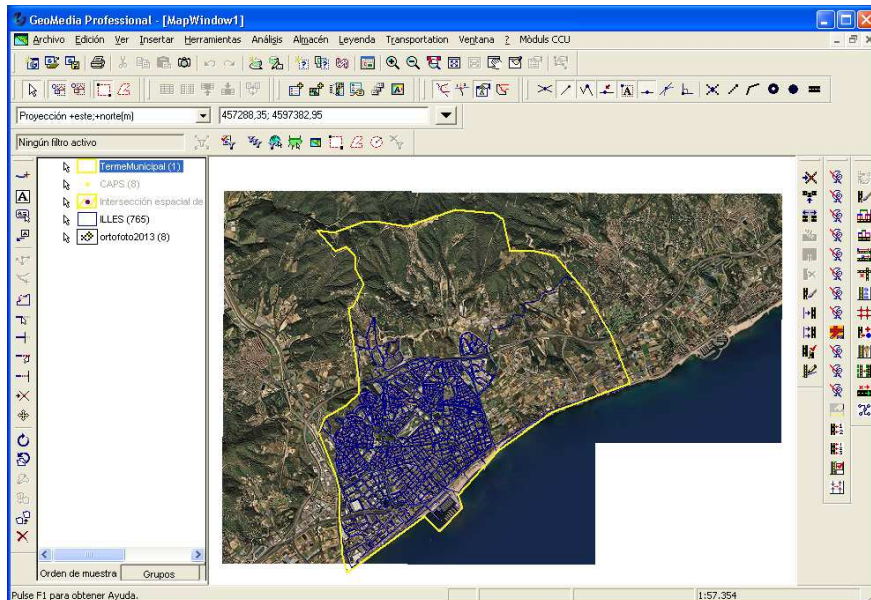


Fig 4.4. Crear nova entitat de tipus area(4/4).

La segona capa simplement és la capa CAPS.

Pas 2: Exportar les capes en format ShapeFile.

Per tal de poder treballar amb les capes en el Global Mapper, aquestes s'han d'exportar en format ShapeFile (Almacén->Exportar a->Archivo Shape...).

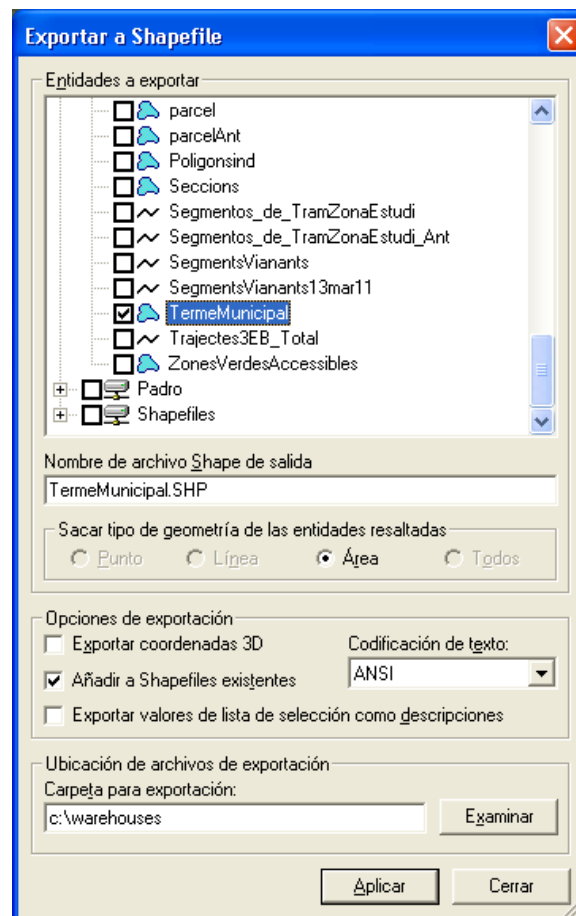


Fig 4.5. Exportar capa en format shapefile.

A continuació se selecciona el nom i es marca la casella de l'entitat que es vol exportar, s'introdueix un nom per al arxiu Shape, se selecciona la carpeta on es vol exportar, i es prem el botó aplicar. Ja es tenen els arxius Shape al a carpeta d'exportació triada.

Pas 3: Tractament al Global Mapper.

S'obre el Global Mapper i s'obren els arxius Shape que s'acaben de crear (file->open Data Files). Se seleccionen els dos arxius (prement la tecla Ctrl) i s'obren.

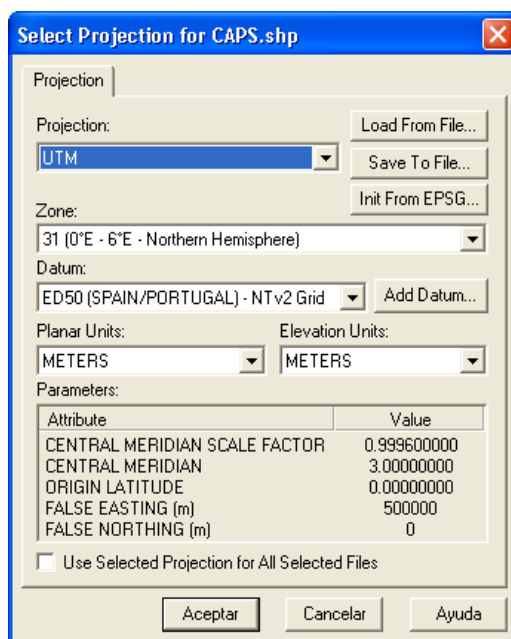


Fig 4.6. Selecció de coordenades geogràfiques.

Apareixerà una finestra com aquesta. S'ha de tenir en compte el sistema de coordenades, si es treballa amb la zona de Mataró, hauria de ser així. Aquesta finestra apareixerà per cada arxiu que obrim. Un cop acceptats, es mostraran les dues capes.

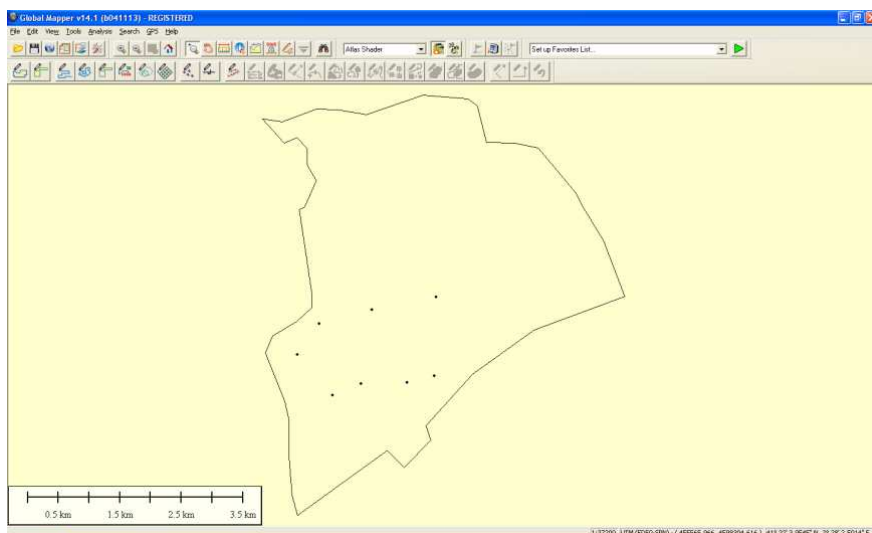


Fig 4.7. Les dues capes mostrades en pantalla.



Del menú superior, se selecciona l'eina "feature info tool", que servirà per seleccionar l'àrea amb la que es vol treballar, en aquest cas, el terme municipal.

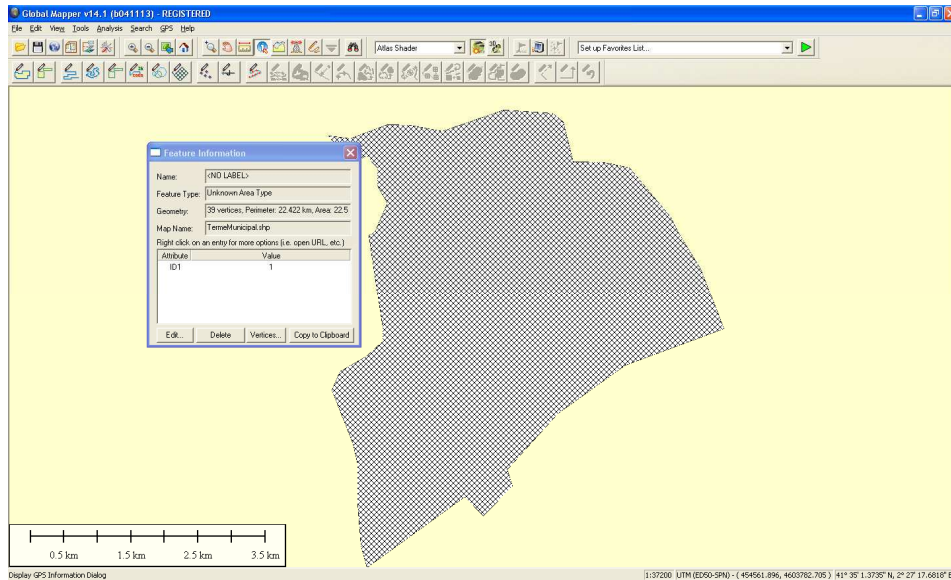


Fig 4.8. "feature info tool".

Un cop seleccionat, es tanca la finestra que apareix, i es va a: Analysis->Create Voronoi/Thiessen diagram from point features.

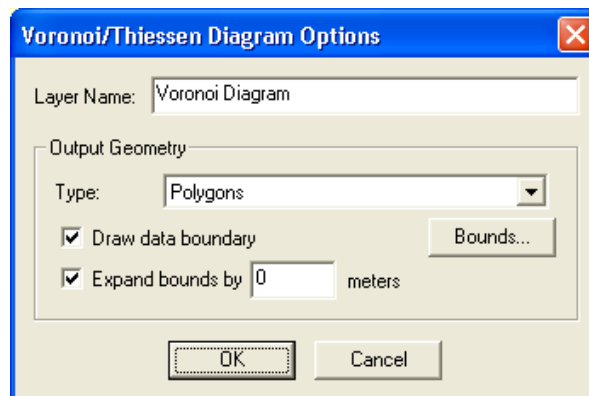


Fig 4.9. Opció polígons de Voronoi.

Es deixa tot per defecte, es prem "Bounds" per seleccionar els límits.

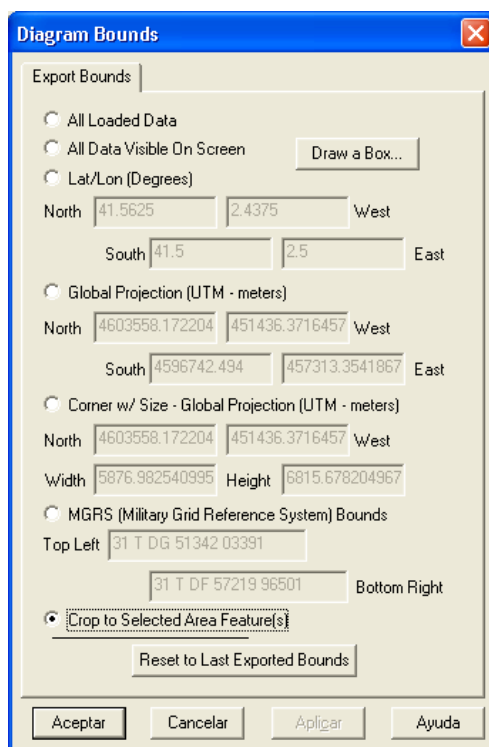


Fig 4.10. Selecció de límits.

Se selecciona l'última opció (Crop to Selected Area Feature), que configurarà els límits a l'àrea seleccionada. S'accepta i es prem "ok" a l'altra finestra d'abans.

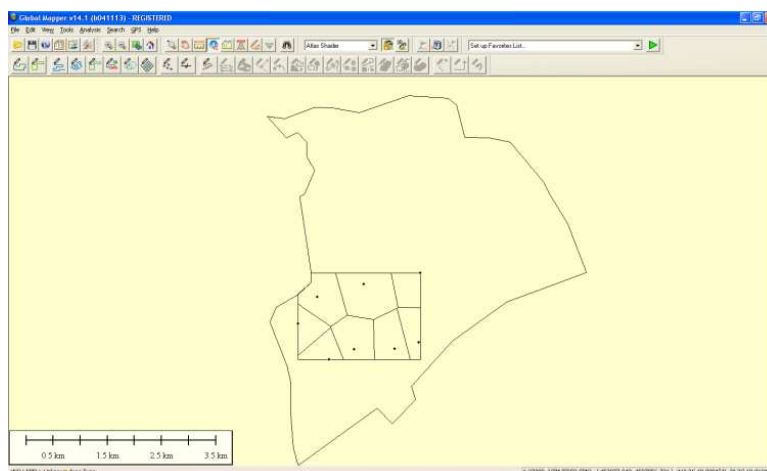


Fig 4.11. Polígons de voronoi en Global Mapper.

Ha de quedar alguna cosa semblant a aquesta, com es pot observar, els polígons no arriben a tocar els extrems del terme municipal, per allargar-los mirem la figura 4.12

Per tornar a fer-ho de forma correcta, es prem el botó “open control center” i apareixeran les capes, es tanca la capa Voronoi, i es tornen a fer els passos d’abans, amb un petit canvi, els límits s’expandiran 4000 metres.

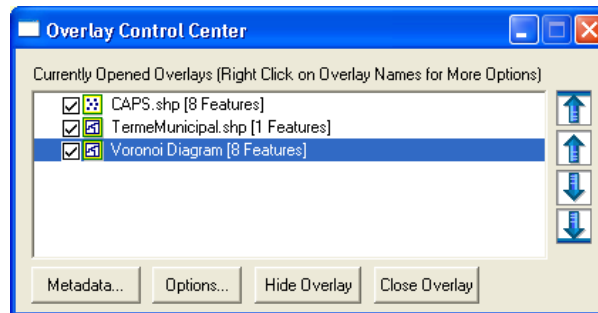


Fig 4.12. Allargar extrems polígons(1/4).

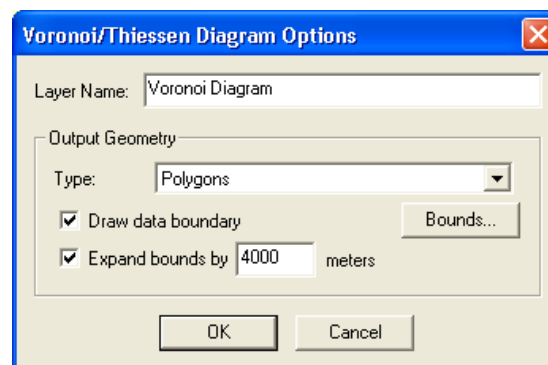


Fig 4.13. Allargar extrems polígons(2/4).

El mapa amb els polígons correctament situats quedaria d’aquesta forma.

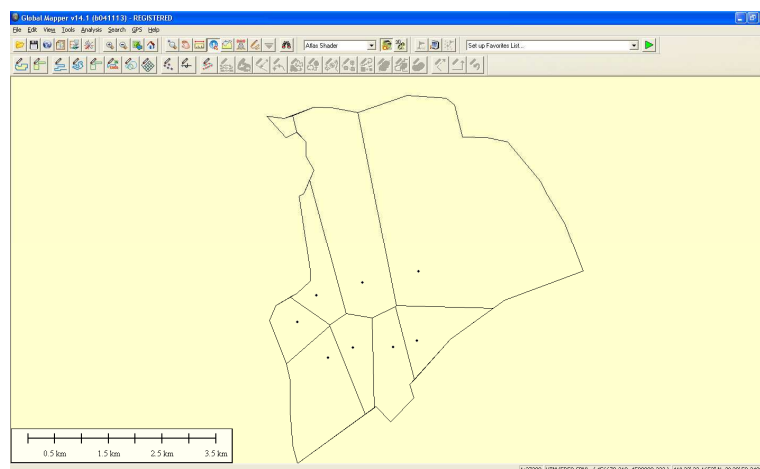


Fig 4.14. Allargar extrems polígons(3/4).

Per fer el centre urbà seria el mateix però agafant com a capa àrea només el centre (al Geomedia), on són totes les illes juntes, quedaria de la següent manera:

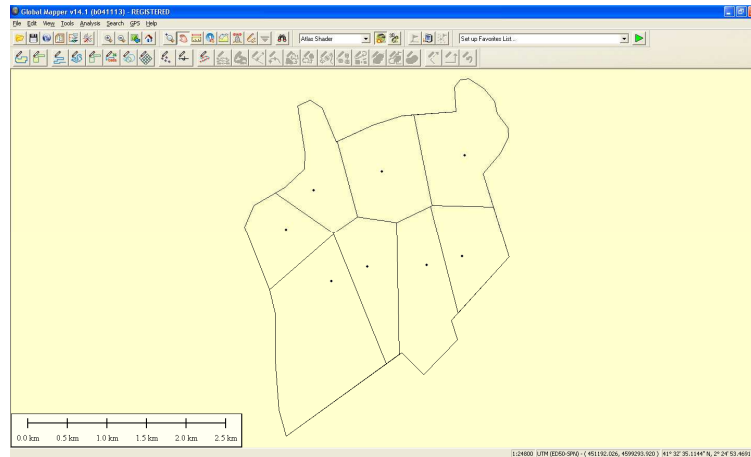


Fig 4.15. Allargar extrems polígons(4/4).

Per poder utilitzar els polígons creats amb el Global Mapper al Geomedia, s'han d'exportar com ShapeFile.

Pas 4: Exportar ShapeFiles desde el Global Mapper:

Seleccionar l'eina "digitizer tool". Seleccionar tot (Edit->Select all features with digitizer tool). Un cop seleccionades, botó dret a sobre->Attribute/Style functions->Add/Update the measure attributes of selected features.

File->Export->Export Vector format->Shapefile->ok->ok.

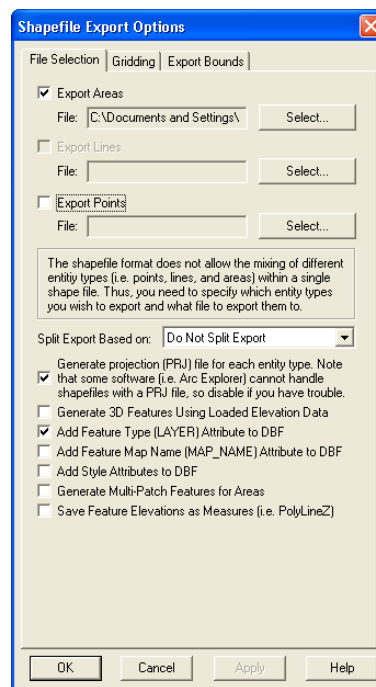


Fig 4.16. Exportar l'arxiu shapefile a globalMapper (1/4).

Quan es prem “OK” ja queden guardats a la ruta especificada.

Pas 5: Importar ShapeFiles desde Geomedia professional:

Almacén->Conexión nueva

- Tipo de conexión: ArcView.
- Nombre de conexión: ShapeFiles (o el que vulguem).
- Carpeta de trabajo ArcView: seleccionem la ruta de la carpeta on vam guardar la configuració dels polígons Voronoi.

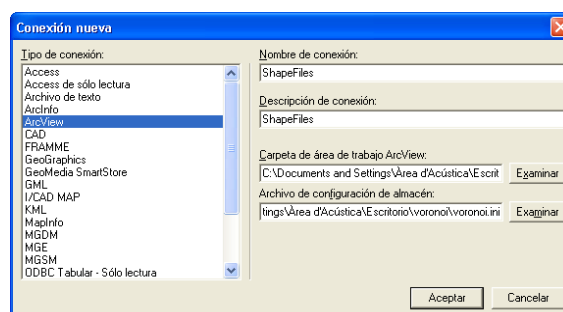


Fig 4.17. Exportar l'arxiu shapefile a globalMapper (2/4).

Leyenda->Agregar entradas de leyenda...

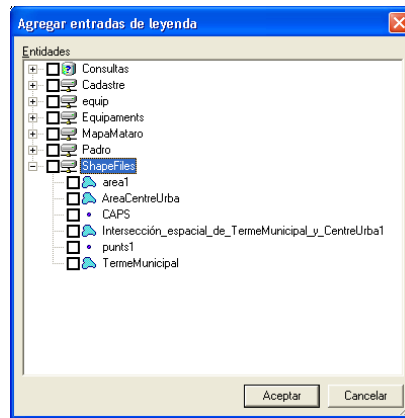


Fig 4.18. Exportar l'arxiu shapefile a globalMapper (3/4).

Es pot veure que ja s'han importat els arxius de la ruta especificada, i es poden mostrar al mapa. S'ha finalitzat el procés d'importació.

Si més tard s'afegeixen més arxius Shape a la carpeta on es van exportar els polígons, s'ha de tancar la connexió del Geomedia i tornar-la a obrir per tal de que reconegui aquests nous arxius.

Almacén->Conexiones->Seleccionem la connexió->Cerrar/abrir.

Així quedaria el polígon de Voronoi importat al geomedia, amb la capa ortofoto a sota:

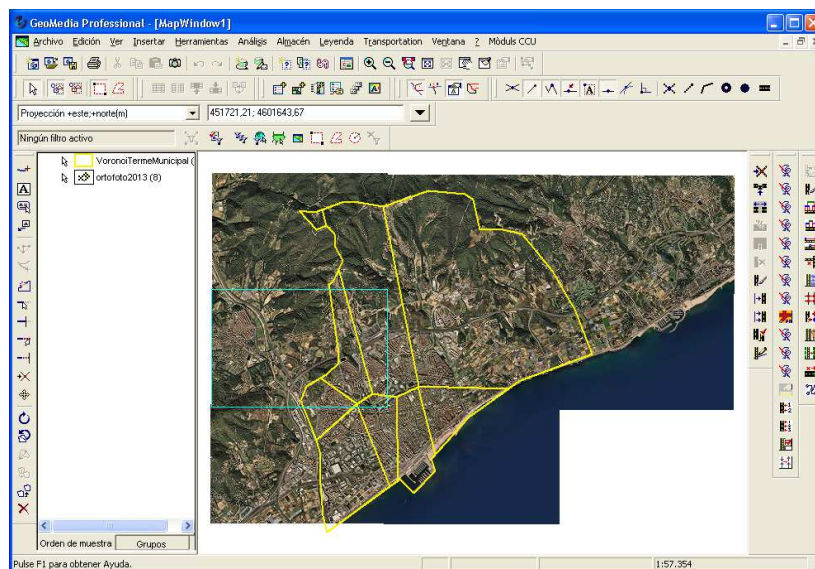


Fig 4.19. Exportar l'arxiu shapefile a globalMapper (4/4).

S'ha explicat amb el Global Mapper, però també es podria aconseguir de la mateixa forma utilitzant altres programes punt com el Qgis o Matlab i després exportant-los com a shape, com veiem a continuació en les següents captures.

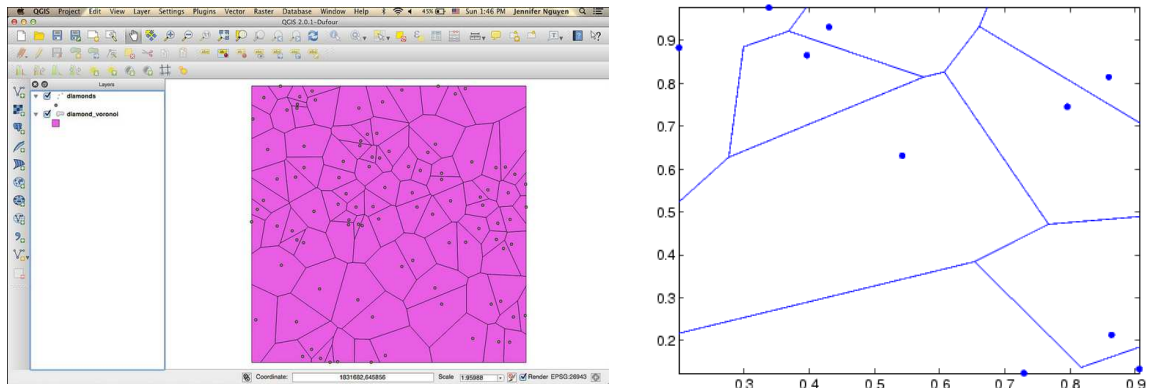


Fig. 4.20. Polígons de Voronoi en Qgis i Matlab.

## 5. Explicació de la aplicació

En aquest apartat es comentarà de forma extensa l'aplicació des de el punt de vista del programador. És a dir, es comentarà pas per pas totes les parts de les quals consta el codi de la aplicació en Visual Basic.

### 5.1. Algoritme i codi inicial

Primer de tot comentaré l'algoritme i el codi utilitzat per du a terme la nova aplicació, trobats a la pagina web de l'informàtic japonès Takashi Ohyam.[8]

*Algoritme*

$i=1, \dots, N-1$

$j=i+1, \dots, N$

        Considerem una bisectriu entre  $p(i)$  i  $p(j)$

$k=1, \dots, N$  excepte per "i" i "j"

            Considerem una bisectriu entre  $p(i)$  i  $p(k)$

            Calculem els punts d'intersecció entre les bisectrius  $(i,j)$  i  $(i,k)$

        next k

        Afegim els punts  $x=0$  i  $x=$  amplada de la pantalla de la bisectriu  $(i,j)$  als punts de les interseccions

        Classifiquem els punts de les interseccions en termes de coordenades x

$k=1, \dots$ , el nombre d'interval dels punts d'interseccions.

            Sigui C un punt mitjà de l' interval dels punts d'intersecció.

            Sigui d igual a  $d(c,p(i))$

$h=1, \dots, N$  excepte per "i" i "j"

                Sigui d' igual a  $d(c,p(h))$

                If  $d' < d$  llavors saltem de línia

            next h

        Si  $d' > d$  llavors es dibuixa l' interval de punts de la intersecció

        next k

    next j

next i



## Codi inicial

```

Private Sub Form_Load()
    Me.Show
    Dim Return_Api As Long

    Randomize ' per utilitzar valors al atzar alhora de posar
    ' les coordenades dels punts i el numero de punts totals
    Dim NNN As Integer           Dim cpx As Double
    Dim ax(100) As Double        Dim ys As Double
    Dim ay(100) As Double        Dim t As Double
    Dim ad(100) As Double        Dim memox As Double
    Dim kx(2000) As Double       Dim memoy As Double
    Dim ky(2000) As Double       Dim x0 As Double
    Dim kz(2000) As Double       Dim y0 As Double
    Dim di2 As Double            Dim xx As Double
    Dim di As Double             Dim yy As Double
    Dim cp2 As Double            Dim xa1 As Double
    Dim ys3 As Double            Dim ys3 As Double
    Dim t2 As Double             Dim t2 As Double
    Dim ya1 As Double            Dim ds As Double
    Dim yy2 As Double            Dim us As Double
    Dim di4 As Double            Dim yNNN As Double
    Dim di3 As Double            Dim y21 As Double
    Dim cp3 As Double            Dim sa0 As Double
    Dim cpx3 As Double           Dim sa5 As Double
    Dim sa1 As Double            Dim sa6 As Integer
    Dim sa2 As Double            Dim br As Integer
    Dim sa3 As Double            Dim br2 As Integer
    Dim sa4 As Double

    ForeColor = RGB(122 + Int(Rnd * 123), 222 + Int(Rnd * 23), 222 + Int(Rnd * 23))
    ' rnd= coeficient al atzar entre 0 i 1
    memox = Rnd * (frmVoro.Width / 15 - 380)
    memoy = 40 + Rnd * 150
    NNN = 4 + Int(96 * Rnd) ' NN=nombre de punts totals
    frmVoro.FontSize = 20
    frmVoro.Line (memox, memoy)-(memox, memoy)
    frmVoro.Print "N="; NNN
    For i = 1 To NNN 'Aquest bucle posa tots els punts en els quals es vol aplicar el voronoi
        ax(i - 1) = Rnd * frmVoro.Width / 15 ' coordenada x punt
        ay(i - 1) = Rnd * frmVoro.Height / 15 ' coordenada y punt
        ad(i - 1) = dis1(0, 0, ax(i - 1), ay(i - 1)) ' funció per treure el mòdul de les
        'dues coordenades
        frmVoro.Circle (ax(i - 1), ay(i - 1)), 3, ForeColor, , , 1
        'dibuixa els punts a la pantalla del form
    Next i
    Call hSort(NNN, ad, ax, ay)

```

```

For i = 1 To MNM - 1
  For j = i + 1 To MNM
    DoEvents
    '//Considerar la bisectriu (perpendicular al punt mig)entre "i" i "j"
    di2 = (ay(i - 1) - ay(j - 1)) / (ax(i - 1) - ax(j - 1)) 'slope of generators
    'i and j
    di = -1 / di2 'Declinar la bisectriu entre "i" i "j"
    cp2 = (ay(i - 1) + ay(j - 1)) / 2 ' Coordenada Y del punt mig entre "i" i "j"
    cpx = (ax(i - 1) + ax(j - 1)) / 2 ' Coordenada X del punt mig entre "i" i "j"
    ys = cp2 - cpx * di 'Interceptar la bisectriu ( perpendicular a punt mig)
    ' entre "i" i "j"
    '//Ara, y= di* x +ys es la bisectriu( perpendicular al punt mig)entre "i" i "j"
    '//Per tant, cada punt (x,y) en aquesta línia te una propietat.
    '//Distancia entre "i" i (x,y) és igual a la distancia entre "j" i (x,y).
    '//Llavors podem obtenir la regió V(a_i) = {x|d(x,a_i)<=d(x,a_j) per "j" i no
    'per "i", en aquest cas, j està fixat.
    '//L'equació de Voronoi es V(a_i) = {x|d(x,a_i)<=d(x,a_j) per tots els punts
    '"j" no "i". En aquest cas j no està fixat.
    '//Llavors, hem de comprovar els segments de línia (grup de punts) on aquesta
    'bisectriu es més propera que a qualsevol altre "j" (no fixada, aquesta "j" es
    'converteix en "k" i "u" més tard en el bucle).
    '// Fem aquest bucle de verificació- for(u=1;u<=N;u++) u<>i, u<>j
    '
    '//Ara, que són els segments de línia (grup de punts)?
    '// Volem comparar amb qualsevol altre "j", llavors els segments de línia son
    'obtinguts des de la intersecció entre aquesta bisectriu i la bisectriu entre
    '"i" i "k" (k no és i,j)
    t = dis1(ax(i - 1), ay(i - 1), ax(j - 1), ay(j - 1)) 'distancia entre "i" i "j"

    '//si intercepció>0 i <(la alçada de la pantalla) llavors el punt inicial de la
    'bisectriu es (0,ys)

    If ys > 0 And ys < frmVoro.Height / 15 Then 'frmVoro.Height= alçada del form
      x0 = 0
      y0 = ys
    Else 'ys<0 or ys> la alçada de la pantalla
      If di > 0 Then '// if slope>0 llavors el punt inicial és (-ys/di,0)
        x0 = -ys / di
        y0 = 0
      Else
        x0 = ((frmVoro.Height / 15) - ys) / di
        y0 = frmVoro.Height / 15
      End If
    End If
    yy = di * (frmVoro.Width / 15) + ys '// yy és la coordenada Y quan X= a la
    'emplada de la pantalla
    If yy > 0 And yy < frmVoro.Height / 15 Then '//if yy>0 and yy<La alçada de
    'la pantalla, llavors el punt final de la bisectriu ( perpendicular al punt
    'mig) entre "i" i "j" és (amplada de la pantalla,yy)
      xa1 = frmVoro.Width / 15
      ya1 = yy
    Else 'yy<0 or yy>la alçada de la pantalla
      If di > 0 Then '// slope>0
        xa1 = ((frmVoro.Height / 15) - ys) / di
        ya1 = frmVoro.Height / 15
      Else '//slope<0
        xa1 = -ys / di
        ya1 = 0
      End If
    End If
  End If
End For

```

```

'//Calcular la intersecció de les dues bisectrius (i,j) i (i,k)
'//Primera intersecció és el punt inicial de la bisectriu entre "i" i "j"

l = 1
kx(l - 1) = x0 '//punt inicial de la bisectriu entre "i" i "j"
ky(l - 1) = y0 '
kz(l - 1) = 2.1
sa2 = ax(j - 1) - ax(i - 1) '// diferència de x entre "i" i "j"
sa4 = ay(j - 1) - ay(i - 1) '// diferència de y entre "i" i "j"
br = 0

'//Calcula la intersecció de les dues bisectrius (i,j) i (i,k), k no és (i,j)
For k = 1 To MNM
  If k <> i And k <> j Then '//k no es "i" ni "j"
    di4 = (ay(i - 1) - ay(k - 1)) / (ax(i - 1) - ax(k - 1)) '// declina la línia connectant dos punts "i" i "k"
    di3 = -1 / di4 'Declina la bisectriu entre "i" i "k"
    cp3 = (ay(i - 1) + ay(k - 1)) / 2 '//coordenada Y del punt mig entre "i" i "k"
    cpx3 = (ax(i - 1) + ax(k - 1)) / 2 '//coordenada X del punt mig entre "i" i "k"
    ys3 = cp3 - cpx3 * di3 '// Intercepta la bisectriu ente "i" i "k"

    '//ara, y = di3 * x +ys3 es la bisectriu(perpendicular al punt mig) entre "i" i "k"(k no es i,j).

    t2 = dis1(ax(i - 1), ay(i - 1), ax(k - 1), ay(k - 1)) '//Distància entre "i" i "k"
    yNNN = di3 * x0 + ys3 '// Coordenada Y del punt inicial( aquest punt inicial es el punt inicial(coordnada X)
    'de la bisectriu entre "i" i "j") de la bisectriu entre "i" i "k"
    y21 = di3 * xa1 + ys3 '//Coordenada Y del punt inicial( aquest punt inicial es el punt inicial(coordnada X)
    'de la bisectriu entre "i" i "j") de la bisectriu entre "i" i "k"
    sa0 = y0 - yNNN '// la diferència entre les dues coordenades Y al punt inicial
    sa1 = ya1 - y21 '// la diferència entre les dues coordenades Y al punt final
    sa3 = ax(k - 1) - ax(i - 1) '//la diferència de coordenades X entre "i" i "k".
    sa5 = ay(k - 1) - ay(i - 1) '//la diferència de coordenades Y entre "i" i "k".

    If sa2 * sa3 > 0 And sa4 * sa5 > 0 Then '//Desde el punt de vista de i(establir el punt de "i" fins el punt d'origen,
    'que és, x_i=0, y_i=0), x_j*y_j>0 and x_k*y_k>0 then
      sa6 = 1
    Else
      sa6 = 0
    End If
    If sa0 * sa1 > 0 And t > t2 And sa6 = 1 Then '//Des de el punt de vista de "i", si "j" i k van en la mateixa direcció
    '( els signes de X i Y son iguals), i dos bisectrius (i,j) i (i,k) no es creuen dintre de la pantalla, i K es més proper
    'a i que a j
      br = 1 '//Llavors la bisectriu de (i,j) no apareix dintre de la pantalla
      Exit For 'br3 = 1 'break
    End If
    If sa0 * sa1 < 0 Or t < t2 Or sa6 < 0 Then '//Cas else
      If sa0 * sa1 < 0 Or t > t2 Then '//Hi ha una intersecció dintre de la pantalla i "j" es més a prop de "i" que de "k".
        l = l + 1
        kx(l - 1) = (ys3 - ys) / (di - di3)
        ky(l - 1) = di * kx(l - 1) + ys
        kz(l - 1) = 2.1
      End If
    End If
  End If ')//if(k!=i && k!=j)
Next k
If br = 0 Then
  l = l + 1 '// Establir el punt final com a intersecció.
  kx(l - 1) = xa1
  ky(l - 1) = ya1
  kz(l - 1) = 2.1

  '//Ara, en la bisectriu entre "i" i "j", y=di*x+ys, Hi ha "l" interseccions.

```

```

Call hSort(1, kx, ky, kz)

For k = 1 To 1 - 1 '//Es consideren els intervals entre dues interseccions
k2 = k + 1 '// k es el punt inicial del interval, k2 es el punt final de l-interval
xx = (kx(k - 1) + kx(k2 - 1)) / 2 '// Coordenada X del punt mig entre les dues interseccions
yy2 = di * xx + ys '// Coordenada Y del punt mig entre les dues interseccions
ds = dis1(xx, yy2, ax(i - 1), ay(i - 1)) '//distancia entre el punt mig i "i", es el mateix que la distancia
'entre el punt mig i "j".

'//si aquesta distancia, ds es més curta que qualsevol altra distancia fins a u(u no es i,j), aquest segment es OK
'//Equació Voronoi: V(a_i) = {x|d(x,a_i)<=d(x,a_j) per tots j no i}, en cas de que j no estigui fixada, per tots j no i
br2 = 0 '//si la distancia, ds és més curta que qualsevol altra distancia fins a u, br2 és manté 0.
For u = 1 To MNM '// u<>i, u<>j
If u <> i And u <> j Then '// si la distancia a u és mes petita f the distance to u és més petita que la distancia
' a j(i), llavors la bisectriu no deuria ser dibuixada -> break
us = dis1(xx, yy2, ax(u - 1), ay(u - 1)) '// distancia entre el punt mig i "u"
If us < ds Then
br2 = 1
Exit For
End If
End If
Next u
If br2 = 0 Then
frmVoro.Line (kx(k - 1), ky(k - 1)-(kx(k2 - 1), ky(k2 - 1)) ' funció que a partir dels punt finals et dibuixa les
'linies del diagrama de voronoi
Exit For 'break
End If
Next k
End If
Next j
Next i
cmdRetry.Enabled = True

```

- Subrutina cridada en el codi

```

Public Sub hSort(NN As Integer, s1() As Double, s2() As Double, s3() As Double)
Dim si As Integer
Dim kk As Integer
Dim kks As Integer
Dim ii As Integer
Dim jj As Integer
Dim mm As Integer
Dim b1 As Double
Dim b2 As Double
Dim b3 As Double
Dim c1 As Double
Dim c2 As Double
Dim c3 As Double
kks = Int(NN / 2)
For kk = kks To 1 Step -1
ii = kk
b1 = s1(ii - 1)
b2 = s2(ii - 1)
b3 = s3(ii - 1)
Do While 2 * ii <= NN
jj = 2 * ii
If jj + 1 <= NN Then
If s1(jj - 1) < s1(jj) Then
jj = jj + 1
End If
End If
If s1(jj - 1) <= b1 Then
Exit Do
End If

s1(ii - 1) = s1(jj - 1)
s2(ii - 1) = s2(jj - 1)
s3(ii - 1) = s3(jj - 1)
ii = jj
Loop
s1(ii - 1) = b1
s2(ii - 1) = b2
s3(ii - 1) = b3
Next kk

```

```

For nnn = NN - 1 To 1 Step -1
  c1 = s1(nnn)
  c2 = s2(nnn)
  c3 = s3(nnn)
  s1(nnn) = s1(0)
  s2(nnn) = s2(0)
  s3(nnn) = s3(0)
  ii = 1
  Do While 2 * ii <= nnn
    kk = 2 * ii
    If kk + 1 <= nnn Then
      If s1(kk - 1) <= s1(kk) Then
        kk = kk + 1
      End If
    End If
    If s1(kk - 1) <= c1 Then
      Exit Do
    End If
    s1(ii - 1) = s1(kk - 1)
    s2(ii - 1) = s2(kk - 1)
    s3(ii - 1) = s3(kk - 1)
    ii = kk

  Loop
  s1(ii - 1) = c1
  s2(ii - 1) = c2
  s3(ii - 1) = c3
Next nnn
End Sub

```

- Funció utilitzada en el codi

```

Private Function dis1(d1 As Double, d2 As Double,
  d3 As Double, d4 As Double) As Double
  dis1 = ((d3 - d1) ^ 2 + (d4 - d2) ^ 2) ^ 0.5
End Function

```

A partir d'aquest codi obtenim un programa que ens dibuixa al formulari un nombre aleatori de punts en unes coordenades també aleatòries com podem veure a continuació en les següents captures.

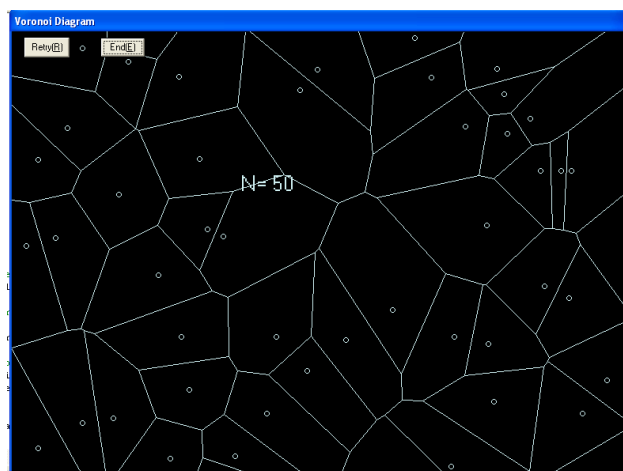


Fig. 5.1. Exemple Voronoi amb 50 punts i coordenades aleatòries

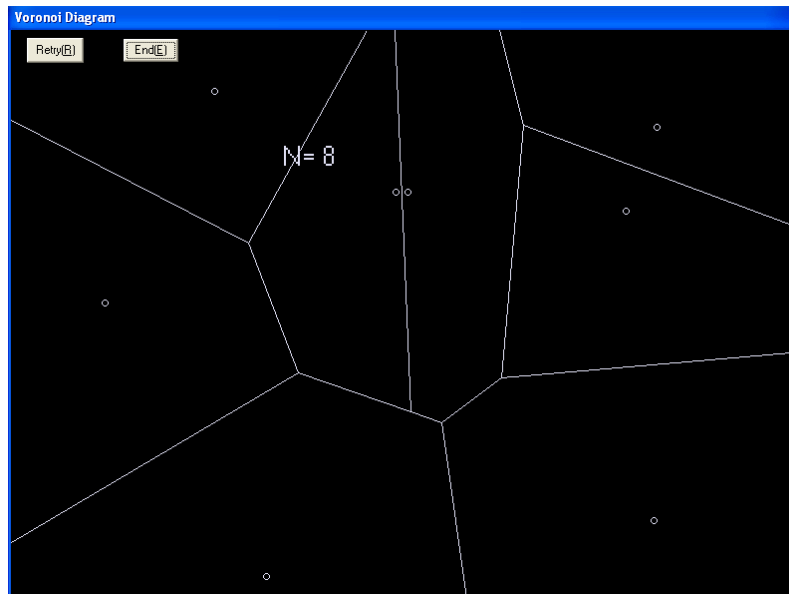


Fig. 5.2. Exemple Voronoi amb 8 punts i coordenades aleatòries

Per poder mostrar el número de punts que nosaltres vulguem simplement hauríem de modificar el paràmetre NNN del codi, i assignar-li el valor de punts que volem mostrar en pantalla.

Per el tema de les coordenades del punts haurem de crear dos vectors, un per les coordenades X i un altre per les Y de NNN posicions i anar recorrent les posicions en el bucle.

Com podem veure en el següent exemple farem el Voronoi de tres punts (NNN=3) amb les coordenades que es poden veure a continuació:

```
X(0) = 1439.96
Y(0) = 928
X(1) = 976.73
Y(1) = 4002.72
X(2) = 3308.92
Y(2) = 2576.08
```

I finalment se li assignen els vectors X i Y a les variables ax i ay, per poder dibuixar els punts amb la funció "circle" en les coordenades que hem introduït.

```
For i = 1 To NNN
  ax(i - 1) = X(i - 1)
  ay(i - 1) = Y(i - 1)
  ad(i - 1) = dis1(0, 0, ax(i - 1), ay(i - 1))
  frmVoro.Circle (ax(i - 1), ay(i - 1)), 10, ForeColor, , , 1
```

Seguint tot l'algoritme amb aquests nous canvis ens quedaria el diagrama de Voronoi de la següent forma.

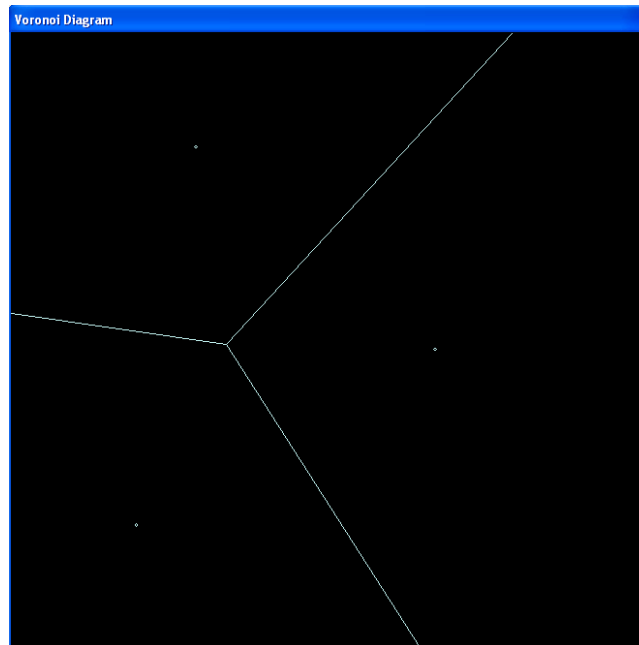


Fig. 5.3. Exemple Voronoi amb 3 punts en les coordenades desitjades.

A partir d'aquest codi he fet una sèrie de modificacions per arribar al resultat de l'aplicació implementada per el Geomedia Professional.

Seguidament és mostraran tots els canvis fets en el codi inicial i les funcions especials per el Geomedia afegides de cara a la present aplicació.

EL programa final ha constatat d'una sèrie de mòduls i un formulari.

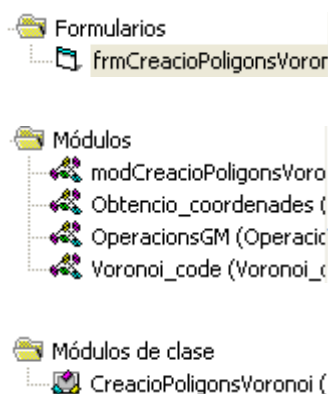
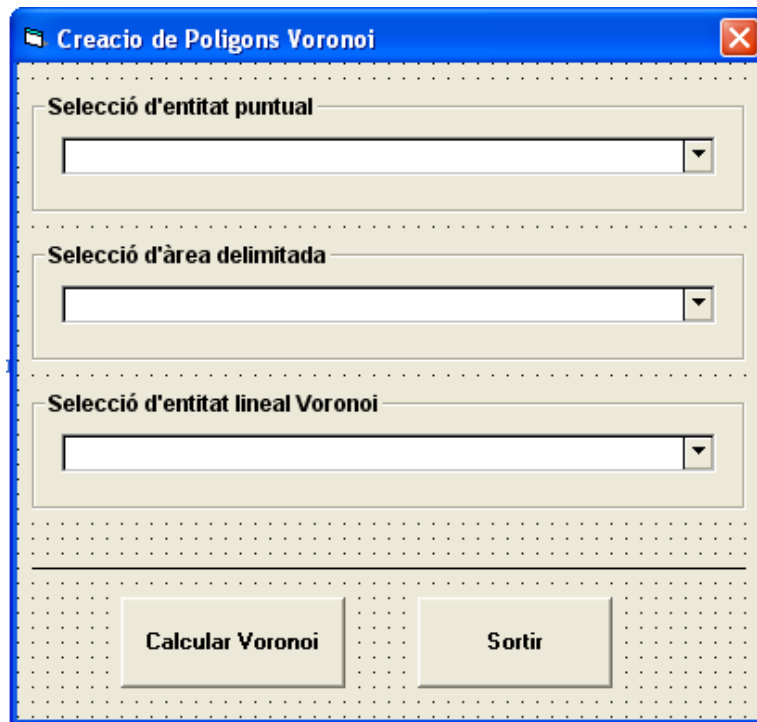


Fig. 5.4. Llistat de mòduls i formulari del projecte.

## 5.2. Formulari

Primer de tot s'ha creat el formulari, que serà la finestra que apareixerà una vegada pitgem per accedir al mòdul creat. El formulari el podem veure a continuació.

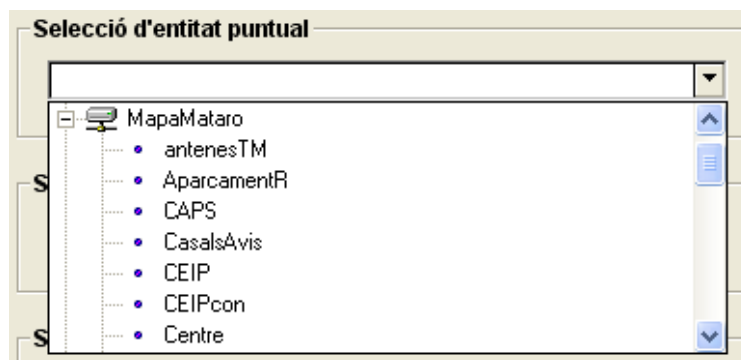


The image shows a dialog box titled "Creacio de Poligons Voronoi". It contains three dropdown menus for selection: "Selecció d'entitat puntual", "Selecció d'àrea delimitada", and "Selecció d'entitat lineal Voronoi". At the bottom, there are two buttons: "Calcular Voronoi" and "Sortir".

Fig. 5.5. Formulari de la aplicació creada.

Ara comentaré part per part els diferents desplegable i botons utilitzats en el formulari.

- En el primer desplegable hem de seleccionar la classe d'entitat puntual sobre el qual volem crear els polígons de Voronoi.



The image shows a dropdown menu titled "Selecció d'entitat puntual". It displays a tree view with "MapaMataro" expanded, showing a list of point entity classes: antenesTM, AparcamentR, CAPS, CasalsAvis, CEIP, CEIPcon, and Centre.

Fig. 5.6. Desplegable per seleccionar l'entitat puntual.



```
'especifiquem els magatzems que volem que es visualitzin en el
'combobox, i de cada magatzem les entitats que tenen geometria de punt
With entitats
    .GeometryTypesVisible = gmmtPoint 'les entitats punt
    .ConnectionsVisible = True 'visualitzar totes les connexions
    .ReadOnlyVisible = True 'no visualitzar les connexions de només lectura
    .QueriesVisible = True 'Visualitzar la cua de queries
    'no indicar la selecció amb el nom complert, que seri nomconnexió.nomentitat,
    'sino que nomes agafarà el nomentitat seleccionada.
    .FullName = False
End With
```

Fig. 5.7. Codi relacionat amb el desplegable de selecció d'entitat puntual.

- En el segon desplegable, haurem de seleccionar la classe d'entitat d'àrea que volem que limiti els polígons de Voronoi. En el nostre cas el terme municipal de Mataró.

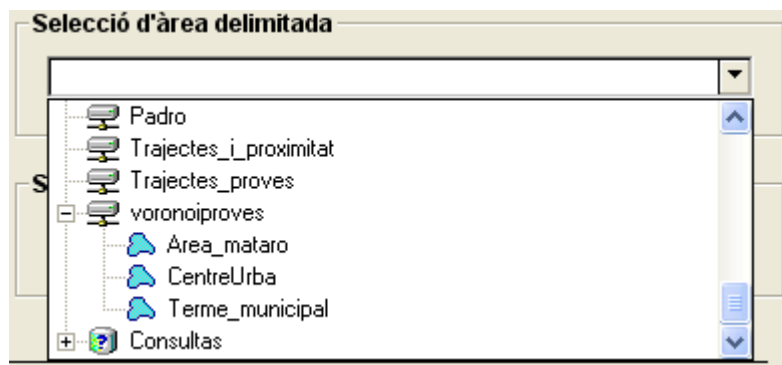


Fig. 5.8. Desplegable per seleccionar l'àrea delimitant.

```
'Fem el mateix per les entitats d'àrea
With areas
    .GeometryTypesVisible = gmmtAreal 'les entitats d'àrea
    .ConnectionsVisible = True 'visualitzar totes les connexions
    .ReadOnlyVisible = False 'no visulatzar les connexions de només lectura
    .QueriesVisible = True 'Visualitzar la cua de queries
    'no indicar la selecció amb el nom complert, que seri nomconnexió.nomentitat,
    'sino que nomes agafarà el nomentitat seleccionada.
    .FullName = False
End With
```

```
Sub areas_LeafClick()
    areas_con = areas.ConnectionName
    areas_select = areas.SelectedItem
End Sub
```

Fig. 5.9. Codi relacionat amb el desplegable de selecció d'àrea delimitant.

- Per últim, seleccionarem la classe d'entitat de línia de sortida. És a dir, on volem que vagin a parar les línies o segments que formaran els polígons de Voronoi finals.

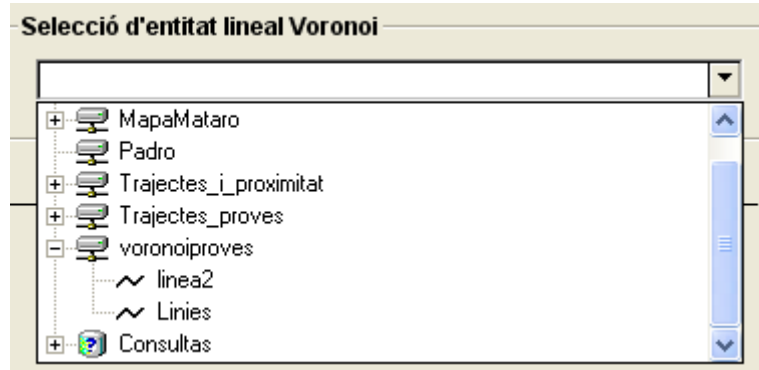


Fig. 5.10. Desplegable per seleccionar la sortida de les línies dels polígons.

```
With entitat_linia
  .GeometryTypesVisible = gmmntLinear 'les entitats punt
  .ConnectionsVisible = True 'visualitzar totes les connexions
  .ReadOnlyVisible = False 'no visualitzar les connexions de només lectura
  .QueriesVisible = True 'Visualitzar la cua de queries
  'no indicar la selecció amb el nom complet, que seri nomconnexió.nomentitat,
  'sino que nomes agafarà el nomentitat seleccionada.
  .FullName = False
End With

Sub entitat_linia_LeafClick()
  entitat_linia_con = entitat_linia.ConnectionName
  entitat_linia_select = entitat_linia.SelectedItem
End Sub
```

Fig. 5.11. Codi relacionat amb el desplegable del desplegable de selecció de sortida.

- Botó "calcular Voronoi"

Aquest botó el que ens farà serà primerament carregar les dades seleccionades als quadres de diàleg del formulari, utilitzant la subrutina "Carregar\_dades".

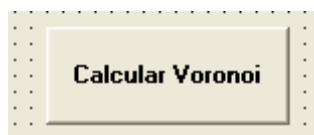


Fig. 5.12. Botó calcular Voronoi.

```

Sub Calcular_voronoi_Click()

Carregar_dades
voronoi_mapa areas_con, areas_select
End Sub

Sub Carregar_dades()

'Consulta com a registre d'entrada
'RS consultes
'Obtenim el recordset de l'entitat origen
Dim rs As GRecordset

On Error GoTo esConsulta:

GoTo fiEsConsulta:
esConsulta:
Dim qf As Object
Dim j As Integer
Set qf = gobjGeoApp.Document.QueryFolder.QuerySubFolders("Queries")
For j = 1 To CInt(qf.Queries.Count)
    If qf.Queries.Item(j).Name = entitats.SelectedItem Then
        Set rs = ferConsulta(qf.Queries.Item(j).Recordset, "")
    End If
Next j
fiEsConsulta:

'afegim parametre per la cua de querys
'cridem a la funció que generarà la consulta en funció de l'entitat i connexió que ha seleccionat l'usuari
Call Proximitat(rs, entitats.SelectedItem, entitats.ConnectionName)
'un cop generada la consulta tanquem el quadre de diàleg
Unload Me
End Sub

```

Fig. 5.13. Codi que va darrere del botó "Calcular Voronoi".

I finalment executarà la subrutina Voronoi\_mapa ( que conté l'algoritme ) i d'aquesta manera formarà els polígons desitjats. Aquesta subrutina serà comentada més endavant en l'explicació del mòdul voronoi\_code.

- Botó Sortir

El botó sortir, simplement seria per poder sortir de la aplicació en qualsevol moment.



Fig. 5.14. Botó sortir.

```

Sub Sortir_Click()
Unload Me
End Sub

```

Fig. 5.15. Codi que va darrere del botó sortir.

## 5.3. Mòduls

Pel que fa als mòduls, comentaré els dos mòduls principals ( “Obtencio de coordenades” i “voronoi\_code”) ja que la resta són creats de forma automàtica quan creem el Geomedia Comand Wizard, que seria el plug-in per tal de poder utilitzar l’aplicatiu VB en el geomedia (Explicat en l’annex II). Tot i que com es veurà més endavant també s’afegeixen algunes funcions necessàries en el mòdul “OperacionsGM”.

### 5.3.1. Mòdul Obtenció de coordenades

Aquest mòdul és essencial per poder passar al algoritme les coordenades de les diferents classes d’entitats puntuals o serveis sobre les quals haurà de crear les zones d’influència (polígons Voronoi).

El mòdul consta de la subrutina "proximitat", que li entren els paràmetres rs, entitats.SelectedItem i entitats.ConnectionName procedents de la subrutina "carregar\_dades" esmentada al formulari.

```
Public Sub Proximitat(rs As GRecordset, Ntaula As String, nomcon As String)

On Error GoTo err_proximitat:
    Dim OrigPipe As OriginatingPipe
    Dim con As Connection
    Dim RSEntitat As GRecordset
    Dim agregacio As AggregationPipe

    'Inici de la pb,c status-----
    Set GeoApp = GetObject(, "Geomedia.Application")
    GeoApp.BeginProgressBar
    GeoApp.BeginInterruptProcessing
    GeoApp.BeginWaitCursor
    GeoApp.SetProgressBarRange 1, 1000
```

- El primer pas per obtenir les coordenades és afegir a la taula de l’entitat puntual escollida per l’usuari les coordenades X Y d’aquesta. Això és codifica de la següent manera:

```

        If (rs Is Nothing) Then
On Error GoTo err_rs:
'Agafem la informació de l'entitat amb la que l'usuari vol treballar i que és la que
'ha escollit en el quadre de dialog(ComboBox).
Set con = gobjGeoApp.Document.Connections(nomcon)
con.CreateOriginatingPipe OrigPipe
OrigPipe.Table = Ntaula 'taula de la que agafem la informació
Set RSEntitat = OrigPipe.OutputRecordset 'guardem resultat
Else
err_rs:
'MsgBox ("3")
Set RSEntitat = rs
End If

'QRY002-Guardar una query

OperacionsGM.crearQuery RSEntitat, "MOD - Consulta de " & Ntaula, "-", "Queries", 0
'QRY002-Final
Set OrigPipe = Nothing

Set con = Nothing

'Definim l'atribut que calcularà la coordenada X de l'entitat puntual escollida
Dim objXep As FunctionalAttribute
GetGeometryFieldNameFromRecordset2 RSEntitat, CampGeometria_Origen

Set objXep = CreateObject("GeoMedia.FunctionalAttribute")
'expressió per calcular la coordenada X de l'entitat escollida
objXep.Expression = "X(Input." & CampGeometria_Origen & "; ProjectedMeas; Metro)"
objXep.FieldName = "Xep" 'nom que volem per l'atribut

'Definim l'atribut que calcularà la coordenada Y de l'entitat puntual escollida
Dim objYep As FunctionalAttribute
Set objYep = CreateObject("GeoMedia.FunctionalAttribute")
'expressió per calcular la coordenada Y de l'entitat escollida
objYep.Expression = "Y(Input." & CampGeometria_Origen & "; ProjectedMeas; Metro)"
objYep.FieldName = "Yep" 'nom que volem per l'atribut

GeoApp.SetProgressBarPosition 200
GeoApp.SetStatusBar CStr("Calculant x i y..."), 1

```

- Generem un recordset afegint a la taula de l'entitat puntual escollida els atributs funcionals que s'han definit anteriorment (Xep, Yep). Un recordset és una estructura utilitzada en programació que permet emmagatzemar informació des d'una taula d'una base de dades.

```

Dim objFAP As FunctionalAttributesPipe
Dim RSXYep As GRecordset
Set objFAP = CreateObject("GeoMedia.FunctionalAttributesPipe")
Set objFAP.InputRecordset = RSEntitat 'entitat a la que volem afegir els dos atributs
'definits anteriorment
'afegim els dos atributs a l'entitat amb el mètode append
objFAP.OutputFunctionalAttributes.Append objXep
objFAP.OutputFunctionalAttributes.Append objYep
Set RSXYep = objFAP.OutputRecordset 'guardem resultat

'QRY003-Guardar una query
'OperacionsGM.crearQuery RSXYep, "002 - X i Y de la consulta " & Ntaula, "-", "Queries"
'QRY003-Final
GeoApp.SetProgressBarPosition 300
GeoApp.SetStatusBar CStr("Afegint atributs..."), 1

```

- Passem el resultat de la consulta (recordset) a una array per les X i per les Y.

```

Dim PROVA As Variant
,
RSXYep.MoveLast
RSXYep.MoveFirst
ReDim X(RSXYep.RecordCount - 1)
ReDim Y(RSXYep.RecordCount - 1)

For kmm = 0 To RSXYep.RecordCount - 1
    X(kmm) = CDb1(RSXYep.GFields(RSXYep.GFields.Count - 2).Value)
    Y(kmm) = CDb1(RSXYep.GFields(RSXYep.GFields.Count - 1).Value)
    RSXYep.MoveNext
Next kmm

```

- Obtenim les coordenades en les variables X i Y, que seran utilitzades més endavant en el mòdul Voronoi\_code, de cara al algoritme.

```

Exit Sub

err_proximitat:
    GeoApp.EndInterruptProcessing
    GeoApp.EndProgressBar
    GeoApp.EndWaitCursor

    Set RSEntitats0 = Nothing
    Set RSPoblacio3 = Nothing
    Set RSAgSegon = Nothing
    Set RSProximitat = Nothing
    res = MsgBox(Err.Description)

End Sub

```

### 5.3.2. Modul Voronoi\_code

Aquest mòdul consta bàsicament del algoritme i de subrutines i funcions d'ajuda per que es puguin crear els polígons de Voronoi sobre les coordenades de les classes d'entitats puntuals obtingudes gràcies al mòdul anterior.

La subrutina principal del mòdul és "voronoi\_mapa" que podem veure comentada per part següent.

- Primerament, definim les variables i recordsets necessaris per la utilització del programa.

```

Sub voronoi_mapa(nomcon As String, Ntaula As String)
    Dim Recordset_linia As GRecordset
    Dim recordset_Area As GRecordset
    Dim conexion As Connection
    Dim origenPipe As OriginatingPipe

    Set conexion = gobjGeoApp.Document.Connections(nomcon)
    conexion.CreateOriginatingPipe origenPipe

    origenPipe.Table = Ntaula 'Taula de la que agafem l'informació

    Set recordset_Area = origenPipe.OutputRecordset

    Dim gPolygon As PolygonGeometry, GSS As GeometryStorageService
    Dim GeomOut, GeomObj As Object

    Set gPolygon = CreateObject("Geomedia.PolygonGeometry")
    Set GSS = CreateObject("Geomedia.GeometryStorageService")

    Dim nom_geo As String
    OperacionsGM.GetGeometryFieldNameFromRecordset2 recordset_Area, nom_geo
    GeomOut = recordset_Area.GFields(nom_geo).Value
    GSS.StorageToGeometry GeomOut, GeomObj
    Dim altura As Double
    Dim ample As Double
    Dim pos_x_High As Double
    Dim pos_y_High As Double
    Dim i As Integer

```

- Calculem els paràmetres ample i altura que seran utilitzats més endavant en l'algoritme.

```

    For i = 1 To GeomObj.Points.Count
        If i = 1 Then
            pos_X = GeomObj.Points(i).X
            pos_Y = GeomObj.Points(i).Y
            pos_x_High = GeomObj.Points(i).X
            pos_y_High = GeomObj.Points(i).Y
        End If
        If pos_X > GeomObj.Points(i).X Then
            pos_X = GeomObj.Points(i).X
        End If
        If pos_Y > GeomObj.Points(i).Y Then
            pos_Y = GeomObj.Points(i).Y
        End If
        If pos_x_High < GeomObj.Points(i).X Then
            pos_x_High = GeomObj.Points(i).X
        End If
        If pos_y_High < GeomObj.Points(i).Y Then
            pos_y_High = GeomObj.Points(i).Y
        End If
    Next i
    ample = pos_x_High - pos_X
    altura = pos_y_High - pos_Y

```

- Establim alguns recordsets necessaris i definim les noves variables utilitzades en l'algoritme. També cridem la funció "borrar\_entitat", que ens permetrà cada vegada que obrim el mòdul buidar la classe d'entitat de línia que he utilitzat per crear el Voronoi anterior.

```

Set conexion = Nothing
Set origenPipe = Nothing
Set recordset_Area = Nothing

Set conexion = gobjGeoApp.Document.Connections(entitat_linia_con)
conexion.CreateOriginatingPipe origenPipe
origenPipe.Table = entitat_linia_select 'Taula de la que agafem l'informació

Set Recordset_linia = origenPipe.OutputRecordset
borrar_entitat Recordset_linia, nomcon
Set Recordset_linia = Nothing
Set Recordset_linia = origenPipe.OutputRecordset

'Me.Show
Dim Return_Api As Long

Dim Height, Width, relacio As Double
Dim Forecolor As Double

Dim NNN As Integer
Dim ax(), ay(), ad(), kx(), ky(), kz() As Variant
Dim di2, di, cp2, cpx, ys, t, memox, memoy As Variant
Dim x0, y0, xx, yy, xa1, ya1, yy2 As Variant
Dim di4, di3, cp3, cpx3, ys3, t2, ds, us As Variant
Dim yNNN, y21, sa0, sa1, sa2, sa3, sa4, sa5 As Variant
Dim sa6, br, br2, u, k2 As Integer
Dim ret As Long
Dim j, k, l As Integer
Dim q1 As Long

                                xa1 = 0
                                ya1 = 0

                                Height = altura
                                Width = ample

                                relacio = 1

Sub borrar_entitat(deleteRs As GRecordset, nomcon As String)

Do While Not deleteRs.EOF
deleteRs.Delete
deleteRs.MoveNext
Loop

    gobjGeoApp.Document.Connections(nomcon).ReloadDatabase
    Dim le As PView.LegendEntry
    For Each le In gobjGeoApp.ActiveWindow.MapView.Legend.LegendEntries
        le.LoadData
    Next
    gobjGeoApp.ActiveWindow.MapView.Refresh True

End Sub

```

- Definim la variable NNN que determina el nombre d'entitats que te la classe d'entitat seleccionada. També a última hora es van haver d'augmentar el número de posicions



dels vectors utilitzats en l'algoritme, ja que en alguns casos on hi havien moltes entitats no acabava de completar l'algoritme per totes elles.

```

NNN = UBound(X) + 1
ReDim ax(NNN)
ReDim ay(NNN)
ReDim ad(NNN)
ReDim kx(NNN * 20)
ReDim ky(NNN * 20)
ReDim kz(NNN * 20)

```

- Comença l'algoritme amb el següent bucle, ens carrega en memòria totes les entitats en les coordenades corresponents, de cara a crear els polígons.

```

For i = 1 To NNN
    ax(i - 1) = X(i - 1) - pos_X
    ay(i - 1) = Y(i - 1) - pos_Y
    ad(i - 1) = dis1(0, 0, ax(i - 1), ay(i - 1))
Next i

```

- ad(i-1), rep el valor del mòdul de les coordenades X i Y de cada punt i l'utilitza alguns cops en l'algoritme

```

Private Function dis1(d1 As Double, d2 As Double,
    d3 As Double, d4 As Double) As Double
    dis1 = ((d3 - d1) ^ 2 + (d4 - d2) ^ 2) ^ 0.5
End Function

```

- Cridem la subrutina "hSort" passant-li els paràmetres NN, ad, ax, ay calculats anteriorment.

```

Public Sub hSort(NN As Integer, s1() As Double, s2() As Double, s3() As Double)
    Dim si As Integer
    Dim kk As Integer
    Dim kks As Integer
    Dim ii As Integer
    Dim jj As Integer
    Dim mm As Integer
    Dim b1 As Double
    Dim b2 As Double
    Dim b3 As Double
    Dim c1 As Double
    Dim c2 As Double
    Dim c3 As Double
    kks = Int(NN / 2)
    For kk = kks To 1 Step -1
        ii = kk
        b1 = s1(ii - 1)
        b2 = s2(ii - 1)
        b3 = s3(ii - 1)
        Do While 2 * ii <= NN
            jj = 2 * ii
            If jj + 1 <= NN Then
                If s1(jj - 1) < s1(jj) Then
                    jj = jj + 1
                End If
            End If
            If s1(jj - 1) <= b1 Then
                Exit Do
            End If
        Loop
    Next kk
End Sub

```

```

        s1(ii - 1) = s1(jj - 1)
        s2(ii - 1) = s2(jj - 1)
        s3(ii - 1) = s3(jj - 1)
        ii = jj
    Loop
    s1(ii - 1) = b1
    s2(ii - 1) = b2
    s3(ii - 1) = b3
Next kk

For mm = NN - 1 To 1 Step -1
    c1 = s1(mm)
    c2 = s2(mm)
    c3 = s3(mm)
    s1(mm) = s1(0)
    s2(mm) = s2(0)
    s3(mm) = s3(0)
    ii = 1
    Do While 2 * ii <= mm
        kk = 2 * ii
        If kk + 1 <= mm Then
            If s1(kk - 1) <= s1(kk) Then
                kk = kk + 1
            End If
        End If
        If s1(kk - 1) <= c1 Then
            Exit Do
        End If
        s1(ii - 1) = s1(kk - 1)
        s2(ii - 1) = s2(kk - 1)
        s3(ii - 1) = s3(kk - 1)
        ii = kk

        Loop
        s1(ii - 1) = c1
        s2(ii - 1) = c2
        s3(ii - 1) = c3
    Next mm
End Sub

```

- Es va executant el gruix de l'algoritme (explicat en l'apartat algoritme i codi inicial) .

```

For j = i + 1 To NNN
DoEvents
di2 = (ay(i - 1) - ay(j - 1)) / (ax(i - 1) - ax(j - 1))
di = -1 / di2
cp2 = (ay(i - 1) + ay(j - 1)) / 2
cpx = (ax(i - 1) + ax(j - 1)) / 2
ys = cp2 - cpx * di
t = dis1(ax(i - 1), ay(i - 1), ax(j - 1), ay(j - 1))
If ys > 0 And ys < Height / relacio Then
x0 = 0
y0 = ys
Else
If di > 0 Then
x0 = -ys / di
y0 = 0
Else
x0 = ((Height / relacio) - ys) / di
y0 = Height / relacio
End If
End If
yy = di * (Width / relacio) + ys
If yy > 0 And yy < Height / relacio Then
xa1 = Width / relacio
ya1 = yy
Else
If di > 0 Then
xa1 = ((Height / relacio) - ys) / di
ya1 = Height / relacio
Else
xa1 = -ys / di
ya1 = 0
End If
End If
End If

l = 1
kx(l - 1) = x0
ky(l - 1) = y0
kz(l - 1) = 2.1
sa2 = ax(j - 1) - ax(i - 1)
sa4 = ay(j - 1) - ay(i - 1)
br = 0
For k = 1 To NNN
If k <> i And k <> j Then
di4 = (ay(i - 1) - ay(k - 1)) / (ax(i - 1) - ax(k - 1))
di3 = -1 / di4
cp3 = (ay(i - 1) + ay(k - 1)) / 2
cpx3 = (ax(i - 1) + ax(k - 1)) / 2
ys3 = cp3 - cpx3 * di3
t2 = dis1(ax(i - 1), ay(i - 1), ax(k - 1), ay(k - 1))
yNMM = di3 * x0 + ys3
y21 = di3 * xa1 + ys3
sa0 = y0 - yNMM
sa1 = ya1 - y21
sa3 = ax(k - 1) - ax(i - 1)
sa5 = ay(k - 1) - ay(i - 1)
If sa2 * sa3 > 0 And sa4 * sa5 > 0 Then
sa6 = 1
Else
sa6 = 0
End If
If sa0 * sa1 > 0 And t > t2 And sa6 = 1 Then
br = 1
Exit For
End If
If sa0 * sa1 < 0 Or t < t2 Or sa6 <> 0 Then
If sa0 * sa1 < 0 Or t > t2 Then
l = l + 1
kx(l - 1) = (ys3 - ys) / (di - di3)
ky(l - 1) = di * kx(l - 1) + ys
kz(l - 1) = 2.1
End If
End If
End If
End If

```

```

Next k
If br = 0 Then
    l = l + 1
    kx(l - 1) = xa1
    ky(l - 1) = ya1
    kz(l - 1) = 2.1
    Call hSort(l, kx, ky, kz)
    For k = 1 To l - 1
        k2 = k + 1
        xx = (kx(k - 1) + kx(k2 - 1)) / 2
        yy2 = di * xx + ys
        ds = dis1(xx, yy2, ax(i - 1), ay(i - 1))
        br2 = 0
        For u = 1 To MNM
            If u <> i And u <> j Then
                us = dis1(xx, yy2, ax(u - 1), ay(u - 1))
                If us < ds Then
                    br2 = 1
                    Exit For
                End If
            End If
        Next u
    Next k
Next i

```

- Finalment amb la funció `Inserta_linia` es formen els segments dels polígons a partir d'un punt inicial i un punt final, on les coordenades dels punts serien respectivament  $x_0 = kx(k-1)$   $y_0 = Ky(k-1)$  i les finals  $x = kx(k2-1)$  i  $y = Ky(k2-1)$ .

```

If br2 = 0 Then
    Inserta_linia Recordset_linia, Cdbl(kx(k - 1)), Cdbl(ky(k - 1)), Cdbl(kx(k2 - 1)), Cdbl(ky(k2 - 1))
    Exit For 'break
End If
Next k
End If
Next j
Next i

```

Els segments resultants es guarden en un recorset anomenat "Recordset\_linia".

```

Sub Inserta_linia(AddRs As GRecordset, X_inici As Double, Y_inici As Double, X_final As Double, Y_final As Double)
    Dim gLine As LineGeometry, GSS As GeometryStorageService
    Dim GeomOut As Variant
    Set gLine = CreateObject("Geomedia.LineGeometry")
    Set GSS = CreateObject("Geomedia.GeometryStorageService")

    gLine.Start.X = X_inici + pos_X
    gLine.Start.Y = Y_inici + pos_Y
    gLine.Start.Z = 0
    gLine.End.X = X_final + pos_X
    gLine.End.Y = Y_final + pos_Y
    gLine.End.Z = 0

    GSS.GeometryToStorage gLine, GeomOut
    Dim d As Integer

    With AddRs
        .AddNew
        .GFields("Geometry").Value = GeomOut
        .Update
    End With
End Sub

```

- Per últim, haurem de mostrar el resultat aconseguit en el mapa i la llegenda

```
'Mostrar resultats en el mapa

gobjGeoApp.SetProgressBarPosition 600
gobjGeoApp.SetStatusBar CStr("Carregat llegenda"), 1
Dim objRLE As RecordLegendEntry
Set objRLE = New RecordLegendEntry
Obtencio_coordenades.GetGeometryFieldNameFromRecordset2 Recordset_linia, CampGeometria_Origen
With objRLE
    .GeometryFieldName = CampGeometria_Origen
    Set .Recordset = Recordset_linia
End With
```

- Seleccionem l'estil de la línia i el nom que li volem posar al resultat.

```
With objRLE
    Set .Style = New LinearStyle
    .Style.BackColor = RGB(255, 0, 0) 'Color vermell
    .Style.ForeColor = RGB(255, 0, 0)
    .Style.Width = 1
    .Title = "Voronoi"
End With
```

- Introduïm la entrada de la llegenda en la primera posició

```
Dim Entries As LegendEntries

Set Entries = gobjGeoApp.ActiveWindow.MapView.Legend.LegendEntries

gobjGeoApp.SetProgressBarPosition 900
gobjGeoApp.SetStatusBar CStr("Carregat la llegenda..."), 1

If Entries.Count = 0 Then
    Entries.Append objRLE
Else
    Entries.Append objRLE, 1
End If

gobjGeoApp.Document.Connections(nomcon).ReloadDatabase
Dim le As PView.LegendEntry
For Each le In gobjGeoApp.ActiveWindow.MapView.Legend.LegendEntries
    le.LoadData
Next

Set Entries = Nothing

gobjGeoApp.SetProgressBarPosition 1000
gobjGeoApp.SetStatusBar CStr("Procés finalitzat"), 1

'refresquem llegenda i finestra de mapa
gobjGeoApp.ActiveWindow.MapView.Fit
gobjGeoApp.ActiveWindow.MapView.Refresh True
gobjGeoApp.ActiveWindow.MapView.Legend.Fit
gobjGeoApp.ActiveWindow.MapView.Legend.Visible = True
gobjGeoApp.ActiveWindow.MapView.Legend.Refresh

gobjGeoApp.EndInterruptProcessing
gobjGeoApp.EndProgressBar
gobjGeoApp.EndWaitCursor

gobjGeoApp.ActiveWindow.MapView.Refresh True

End Sub
```

- Per compilar l'arxiu el programa creat, haurem d'anar a Archivo> generar nomprojecte.dll com podem veure en la següent captura d'imatge. En cas, que no doni cap error de compilació se'ns haurà creat un arxiu dll i estarà llest per carregar-ho al Geomedia.

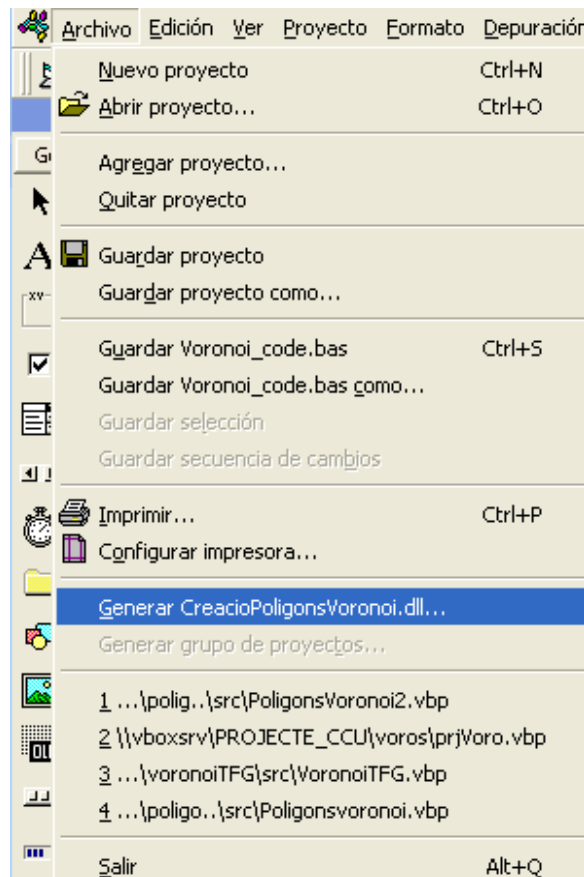


Fig. 5.16. Generar.dll.

Cada vegada que s'hagin fet canvis en el programa, s'haurà de crear una nova dll, de cara a que els canvis sorgeixin efecte en el Geomedia. Com també haurà d'estar tancat el Geomedia Professional m'entres és realitza aquest procés, ja que en cas contrari donarà error.

## 6. Utilització de l'aplicació

En l'anterior apartat s'ha explicat la aplicació creada des del punt de vista d'un programador, en aquest capítol s'explicarà com funciona des del punt de vista de l'usuari.

Es comentarà tot el procés que ha de fer l'usuari per poder utilitzar el mòdul de forma correcte i treure-li el millor partit possible. Per això també es realitzaran una sèrie d'exemples per que l'usuari tingui la màxima facilitat a l'hora d'entendre el funcionament.

### 6.1. Carregar l'aplicació al Geomedia

Primerament explicaré com carregar el mòdul VB al Geomedia, de tal forma que pugui ser utilitzat per l'usuari.

- Quan utilitzem el geomedia command wizard , plug-in que serveix per implementar el mòdul visual basic per que pugui ser utilitzar en l'entorn del geomedia professional, se'ns creen dues carpetes.

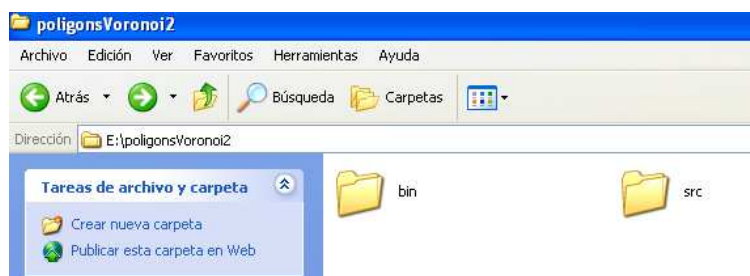


Fig. 6.1. Contingut del nou projecte.

- La carpeta "src" conté tots els arxius del projecte (mòduls, formularis etc..)

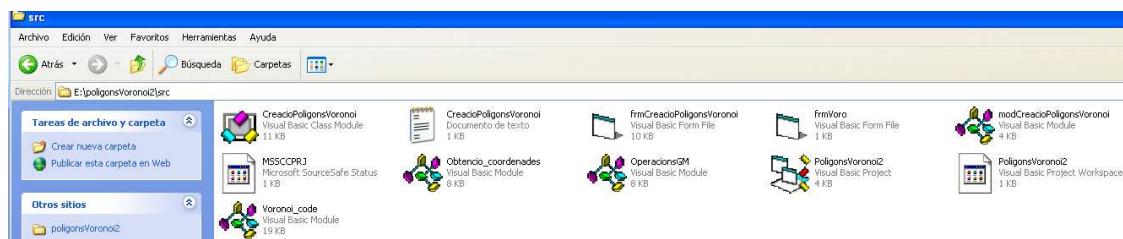


Fig. 6.2. Contingut carpeta "src".

- En la carpeta "bin" trobem l'arxiu .dll que es el que s'haurà d'instal·lar dintre del Geomedia.

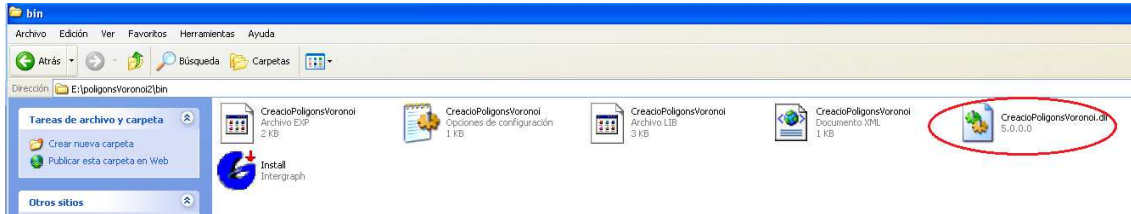


Fig. 6.3. Contingut de la carpeta "bin".

- Per instal·lar-ho utilitzem l'icona "install" d'intergraph, com podem veure a la següent imatge. Seleccionem la ruta de la carpeta "bin" del nostre projecte i pitgem "OK".

Aquest procés tan sols s'haurà de fet un primer cop, ja que una vegada instal·lat el mòdul una vegada, per qualsevol canvi en el mòdul tan sols s'haurà de generar una nova dll, amb el procés explicat anteriorment al final del capítol anterior.

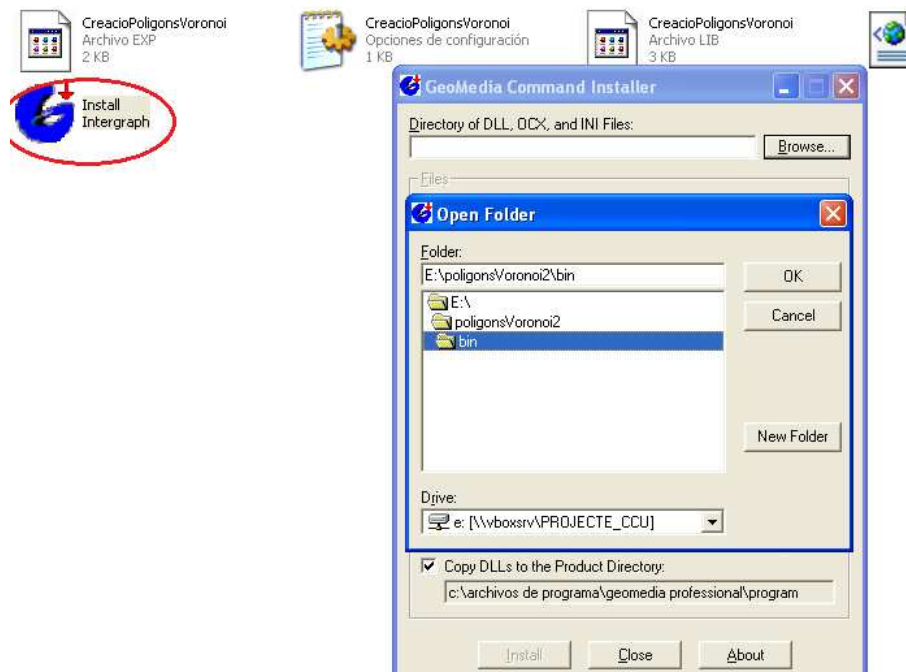


Fig. 6.4. Instal·lació nou mòdul creat.

- Finalment, s'hi la instal·lació ha estat un èxit, ens hauria de sortir un missatge com el següent.



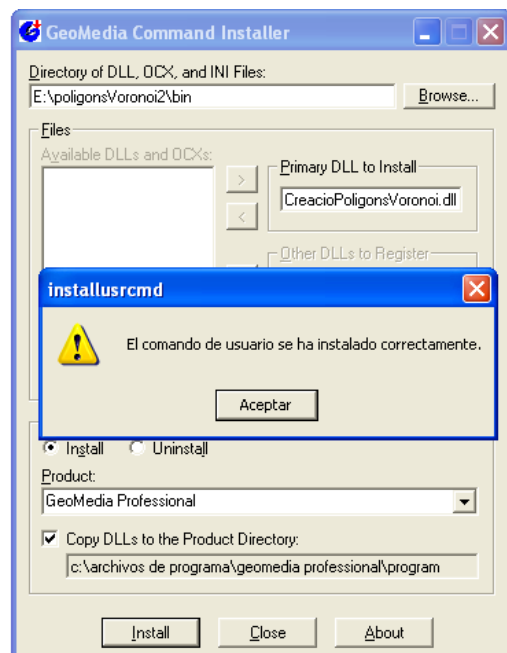


Fig. 6.5. Instal·lació satisfactòria del mòdul.

- Una vegada estem dintre del interface del Geomedia Professional abans de poder provar el nostre mòdul, haurem de fer una mínim una connexió amb una base de dades que contingui classes d'entitat puntual de línia i d'àrea. ja que les tres son necessàries de cara al funcionament del aplicatiu. Per fer la connexió seguim el mateix procés explicat en el capítol 2, apartat Geomedia Professional.

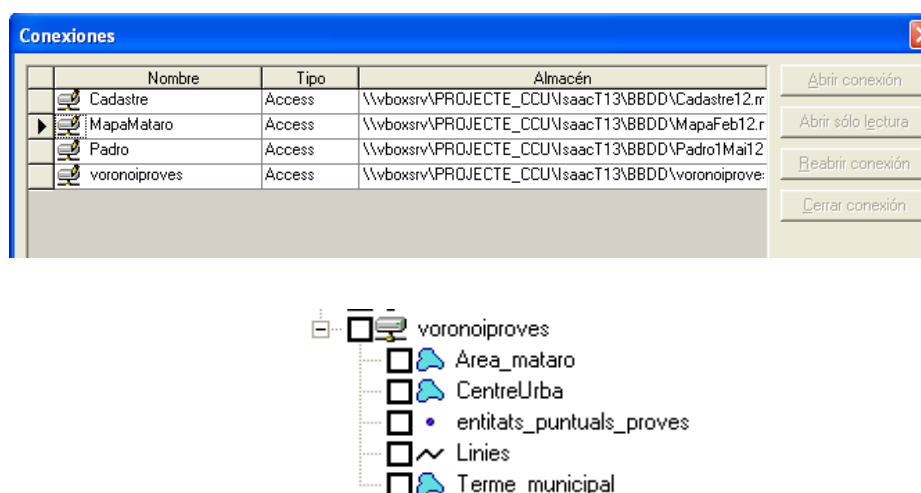


Fig. 6.6. Addició de les connexions necessàries.

- Una vegada fetes les connexions necessàries haurem de pitjar sobre un icona que ens apareixerà per defecte com el de la següent imatge. (Si volem tenir el mòdul a un dels menús, s'ha de seguir el procés explicat a L'annex 1).

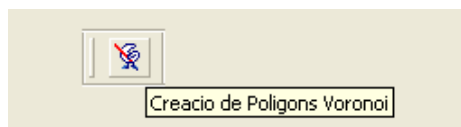


Fig. 6.7. Icona per defecte del nostre mòdul.

- Que ens portarà a la finestra del nostre mòdul, preparat per ser utilitzat.

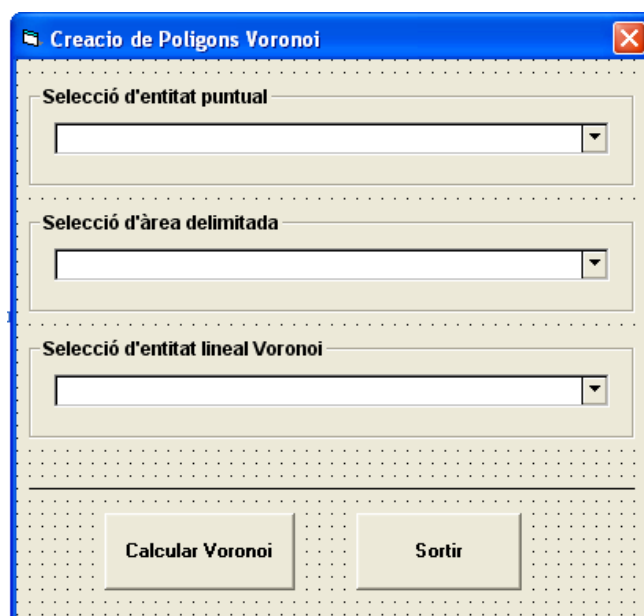


Fig. 6.8. Finestra mòdul.

## 6.2. Exemples de la aplicació

En aquest capítol s'explicarà amb exemples el funcionament de la aplicació creada en l'entorn del Geomedia Professional.

És realitzaran 4 exemples amb centres proveïdors de serveis (escoles bressol, CAPS, centres de formació primària, Parades de bus) i un especial amb una consulta. Aquests exemples estaran documentats pas per pas i aportant diferents captures per mostrar els diferents punts de vista.

### 6.2.1. Escoles bressol

Per poder visualitzar el resultat dels polígons de Voronoi de la millor manera possible, haurem de tenir activades a la llegenda un mínim de dos classes d'entitat. Per una banda, l'area que limitarà el Voronoi, en aquest cas, el terme municipal de Mataró i per l'altra la classe d'entitat puntual sobre la que volem crear les nostres zones d'influència, en aquest exemple, seran les escoles bressol. Per això anem a *Leyenda*>*Agregar entradas de leyenda* i seleccionem aquestes en els desplegable de les connexions i pitgem el botó "aceptar".

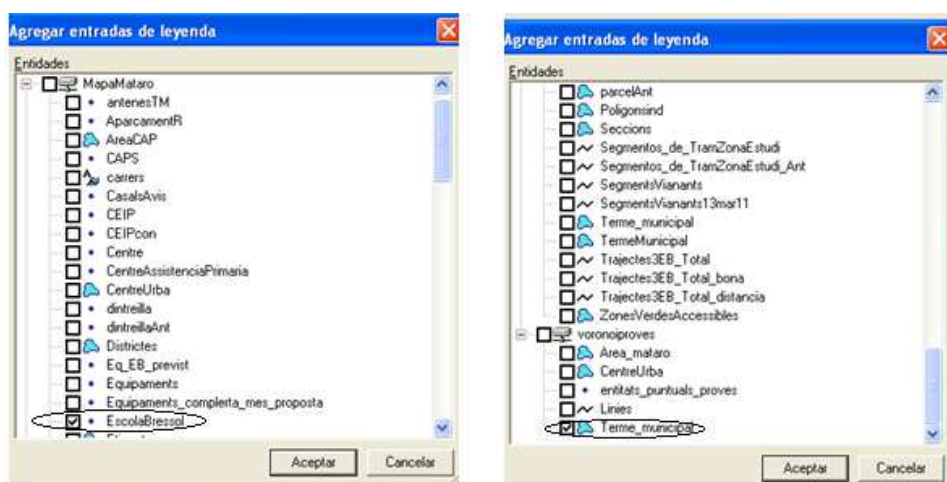


Fig. 6.9. Agregació de entrades de llegenda necessàries.

Com podem veure en les imatges, ens apareixeran les escoles bressol situades en el terme municipal.

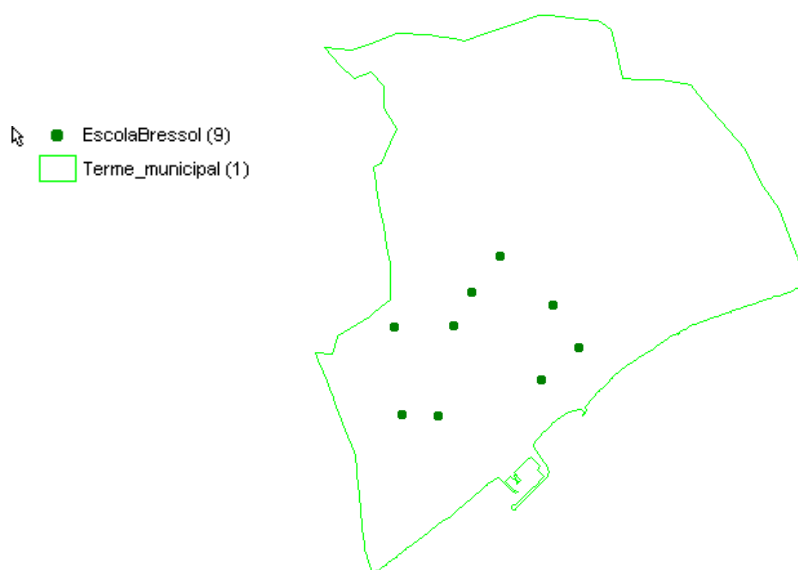


Fig. 6.10. Escoles bressol i el terme municipal mostrats a la pantalla.

Si volem saber més detalls sobre les escoles, simplement pitgem a sobre d'una i ens apareixerà un quadre de diàleg com el següent on ens donarà propietats com el nom de l'escola o el numero de places entre d'altres.

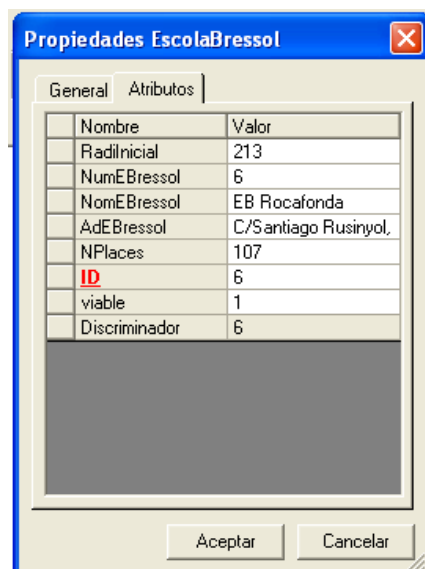


Fig. 6.11. Propietats de les escoles bressol.

Una vegada tenim les classes d'entitat seleccionades a la llegenda, anem al mòdul i seleccionem en els desplegable "EscolaBressol" com a entitat puntual, "Terme\_municipal" com àrea delimitada i "Linies" com a sortida lineal per guardar el resultat del diagrama de Voronoi. Finalment pitgem el botó Calcular Voronoi.

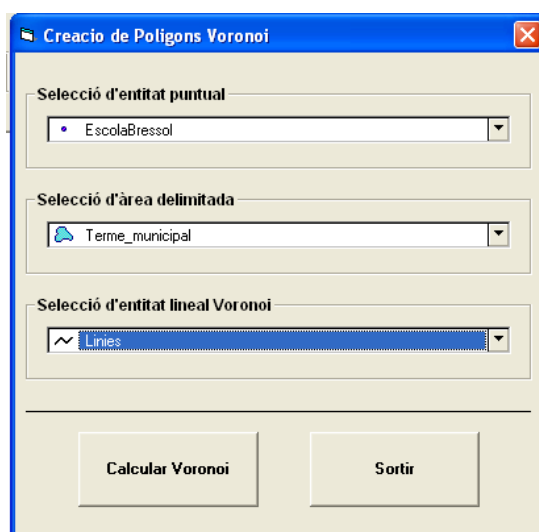


Fig. 6.12. Omplir desplegable del mòdul per escoles bressol.

Com podem veure obtenim el diagrama Voronoi perfecte sobre les escoles bressol. El diagrama per aquest cas constarà de 17 línies guardades a la classe d'entitat "linies".

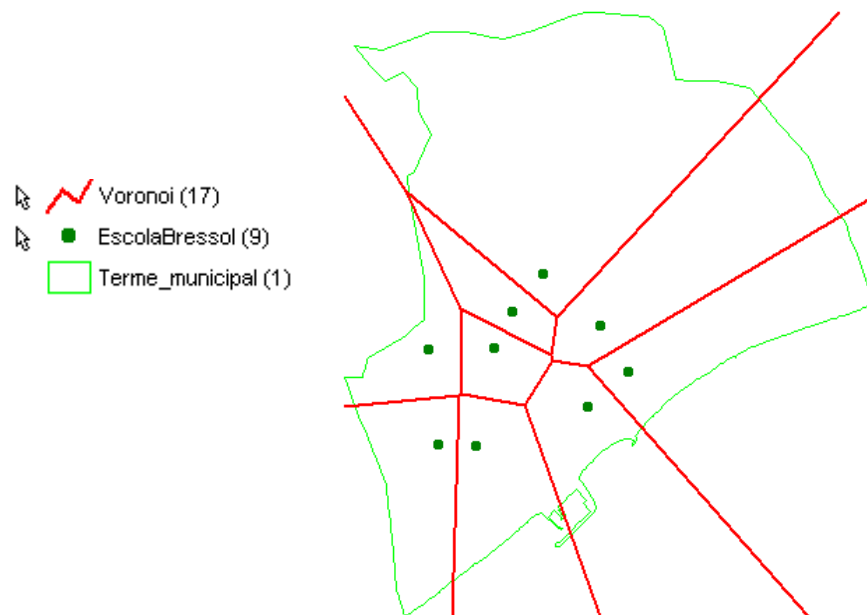


Fig. 6.13. Polígons de Voronoi sobre les escoles bressol.

Com es pot comprovar, les línies no acaben en el límit del terme municipal, això es deu, ja que l'algoritme de Voronoi sempre es base en un quadrat o rectangle per arribar al punt final de les línies. Per tant en aquest cas calcula un quadrat imaginari amb els punts més alts de l'amplada i l'alçada. Per poder arreglar aquest problema, tenim dos formes de fer-ho des de el Geomedia:

1. Fent una intersecció espacial:

Anem a Anàlisis > Intersecció espacial .Una vegada dintre haurem de seleccionar les classes d'entitat de línia i terme municipal i deixar per defecte la opció " es toquen". Escrivim el nom final de la consulta resultant i pitgem "acceptar".

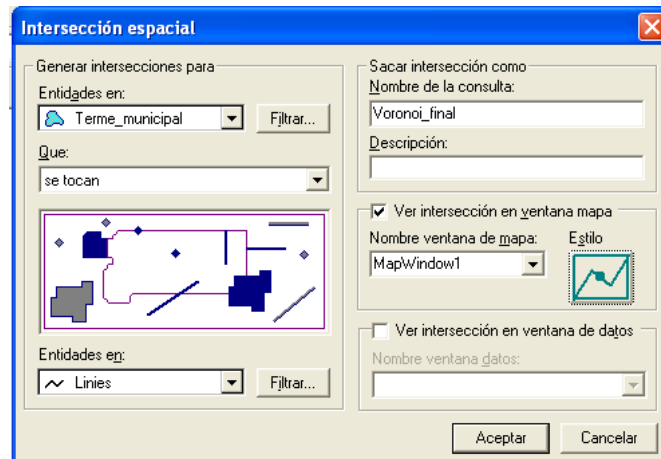


Fig. 6.14. Intersecció espacial entre línies i terme municipal.

Amb aquesta intersecció aconseguirem el resultat desitjat, com podem veure a la següent captura. El resultat de la consulta com podem veure a la llegenda es una barreja entre línies i àrees.

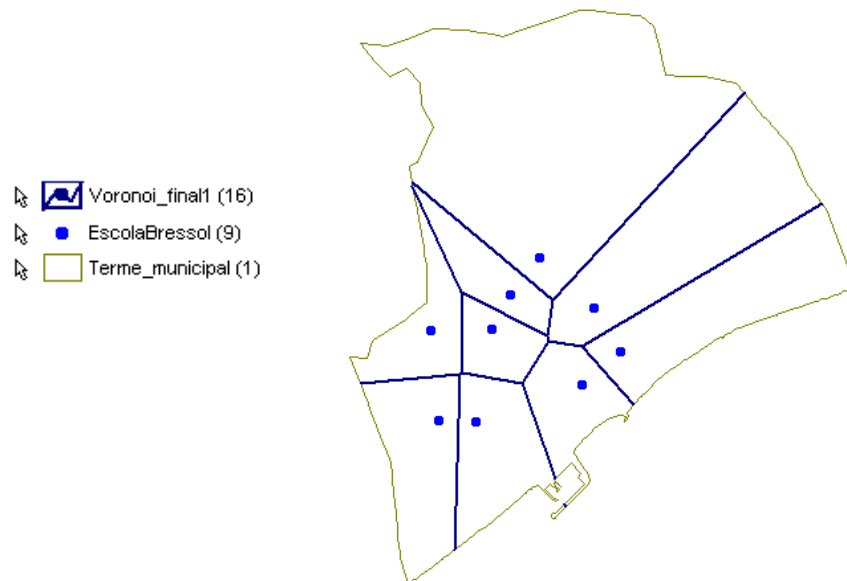


Fig. 6.15. Voronoi final sobre les escoles bressol mitjançant intersecció.

2. Amb la opció dividir entidades.

Pitgem el boto "dividir entidades".

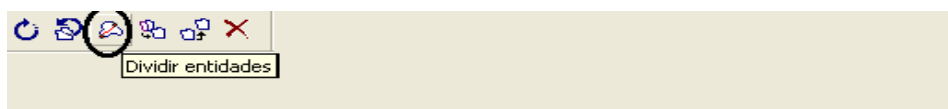


Fig. 6.16. Procés dividir entitats (1/3).

Marquem el cursor i seleccionem el mapa amb les línies amb un quadrat.



Fig. 6.17. Procés dividir entitats (2/3).

Pitgem amb el botó dret a la pantalla i seleccionem la opció "realizar division". Llavors, et va dividint el terme municipal en seccions de una en una.

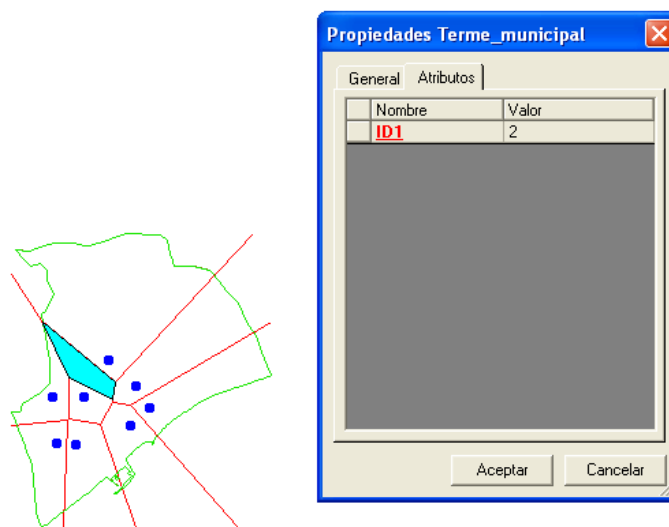


Fig. 6.18. Procés dividir entitats (3/3).

Fins arribar al resultat final on tenim el Terme municipal dividit en regions de Voronoi.

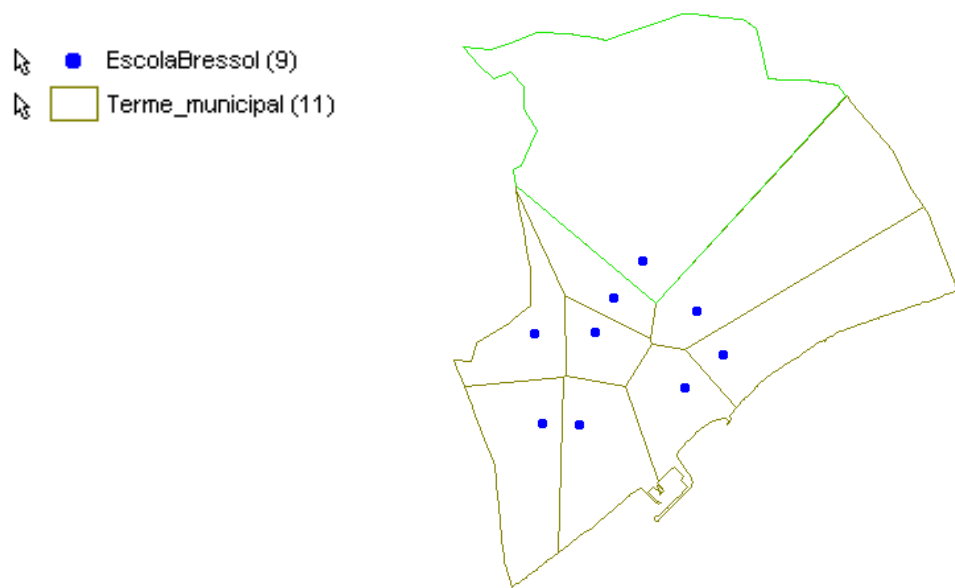


Fig. 6.19. Voronoi resultant sobre escoles bressol mitjançant divisió d'entitats.

Llavors amb aquest mètode obtenim com a resultat final àrees. Seleccionant a la llegenda la classe d'entitat "illes" podríem veure les regions d'una forma més visual.

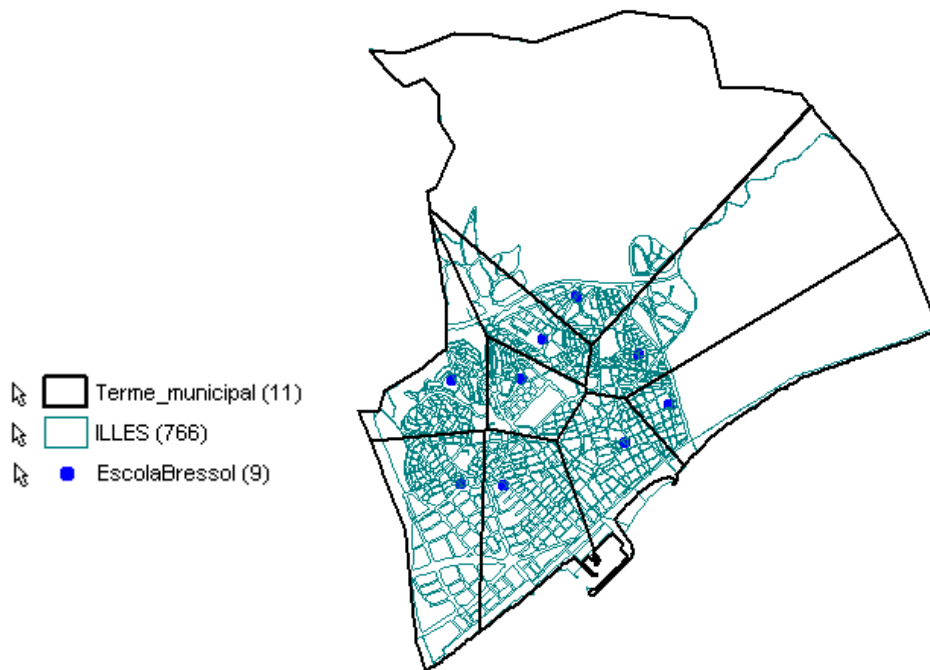


Fig. 6.20. Voronoi resultant sobre escoles amb la capa d'illes.

Per últim activant la capa de la "ortofoto2013" a la llegenda podem visualitzar els polígons des de una vista Aèria similar a la de Google Maps.





Fig. 6.21. Voronoi resultant sobre escoles amb la capa "ortofoto".

## 6.2.2. CAPS

En aquest exemple crearem les regions de Voronoi al voltant dels CAPS de Mataró.

Primer de tot, com hem fet en l'exemple anterior agreguem a la llegenda la classe d'entitat puntual (CAPS) i l'àrea delimitant (terme\_municipal). Com podem comprovar tenim 8 CAPS tal i com ens mostra la llegenda.

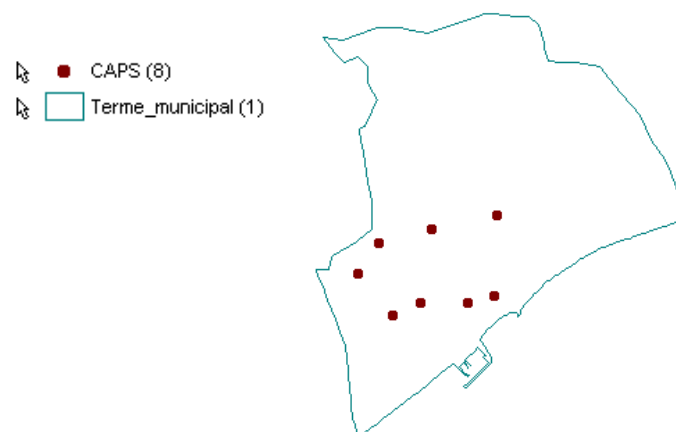


Fig. 6.22. Caps i terme municipal mostrats en pantalla.

Obrim el mòdul i com hem fet en l'exemple anterior, seleccionem l'entitat puntual, (CAPS en aquest cas), el terme municipal com a àrea delimitada i per últim "línies" com a classe d'entitat de sortida per guardar el resultat. Pitem Calcular Voronoi.

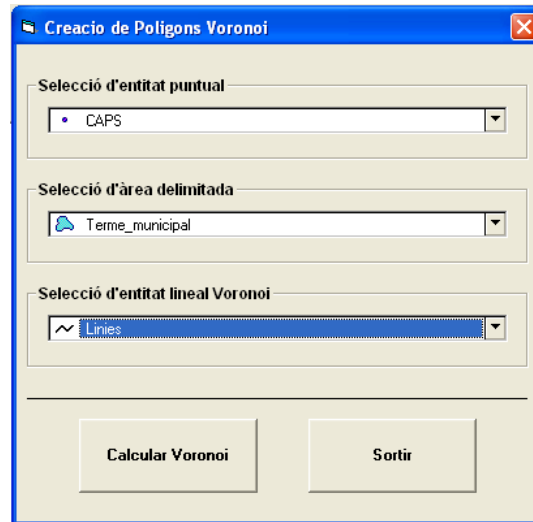


Fig. 6.23. Omplir desplegable del mòdul per escoles bressol.

Obtenim el digrama de Voronoi resultant pels CAPS, que conté en aquest cas 14 línies.

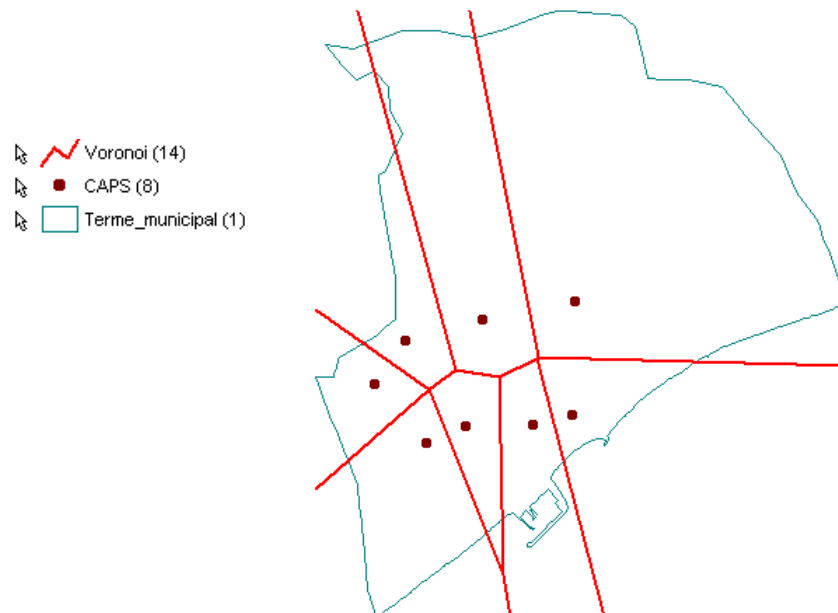


Fig. 6.24. Diagrama de Voronoi sobre CAPS.

Llavors, utilitzant el segon sistema explicat anteriorment (Escoles bressol) per separar el Terme municipal en les regions de Voronoi obtindríem la següent imatge.

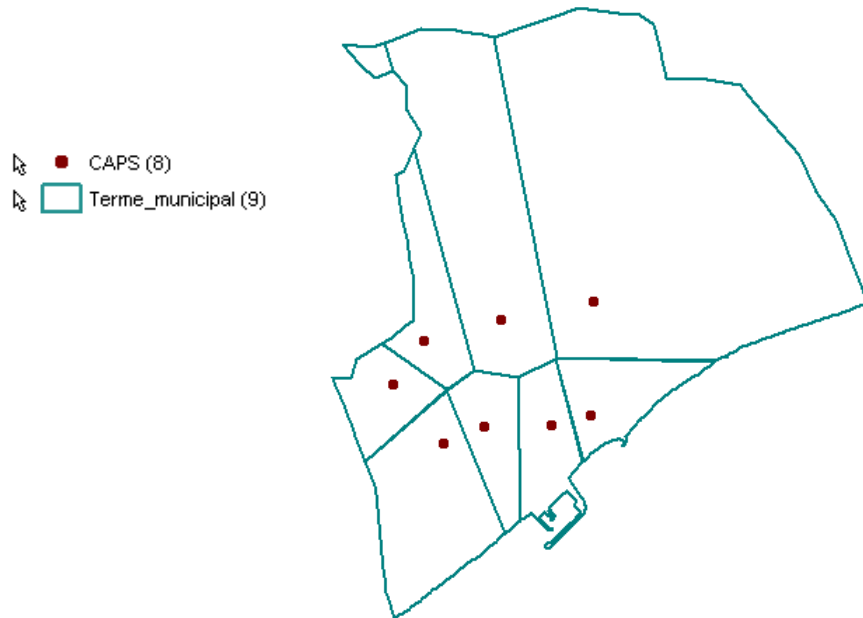


Fig. 6.25. Voronoi final sobre els CAPS mitjançant intersecció espacial .

Agregant la classe d'entitat illes a la llegenda tindríem la següent vista.

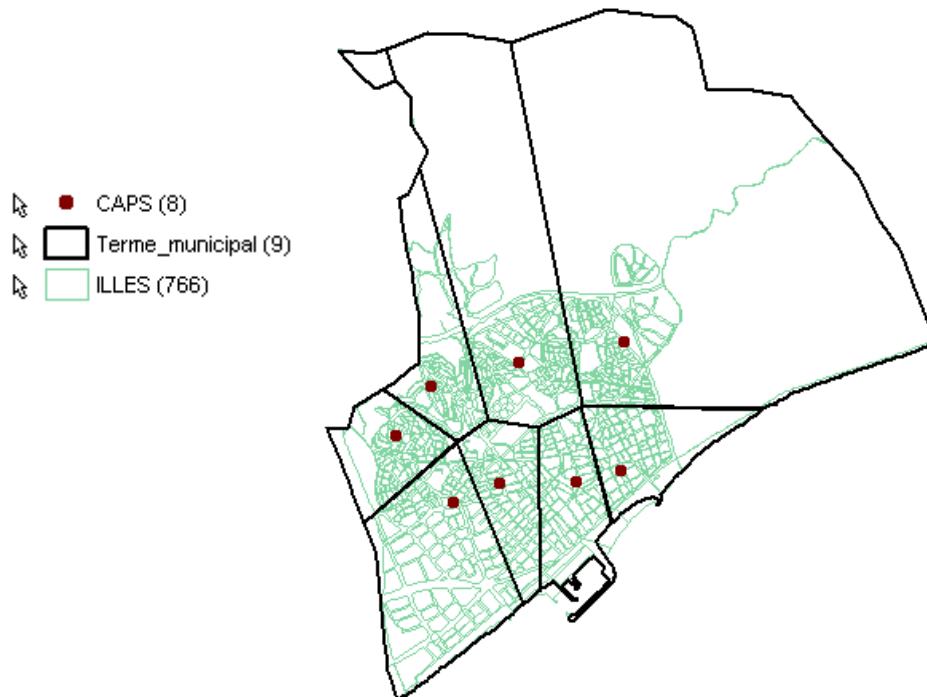
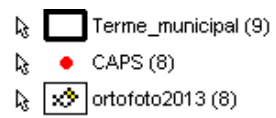


Fig. 6.26. Voronoi final sobre els CAPS amb la capa d'illes .

Finalment amb activant la Ortofoto com en l'exemple anterior, tindríem la vista des de dalt.



*Fig. 6.27. Voronoi final amb la vista Aérea.*

### 6.2.3. Parades de bus

Com als altres dos exemples, agreguem a la llegenda les parades de bus en aquest cas i el terme municipal. Com es pot comprovar a la següent imatge hi ha un total de 144 parades de bus a Mataró.

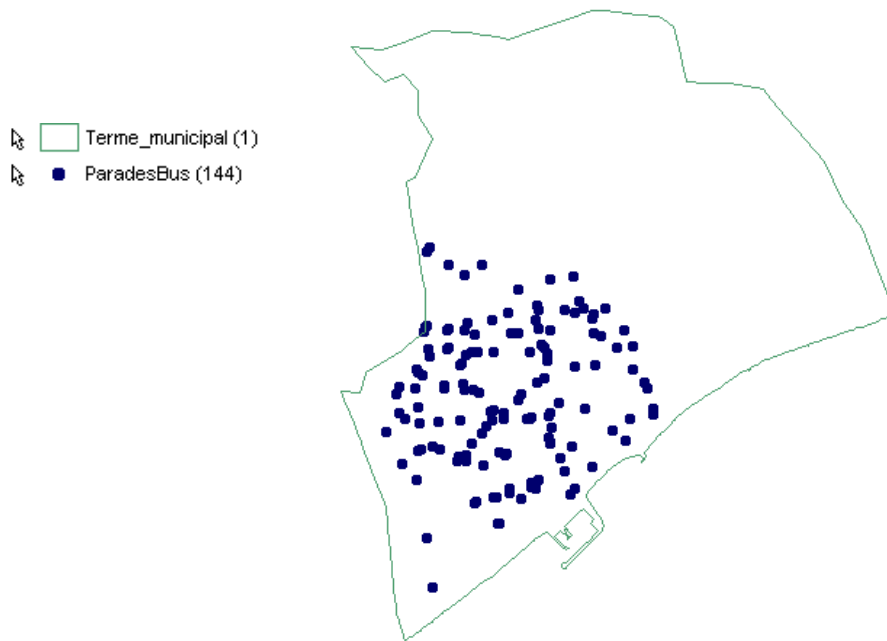


Fig. 6.28. Parades de bus i terme municipal mostrats en pantalla.

Seleccionem ParadesBus, Terme\_municipal i Línies en els desplegable del nostre mòdul.

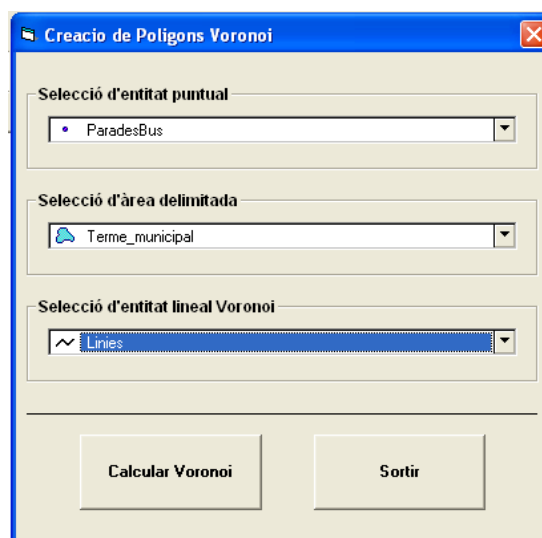


Fig. 6.29. Omplir desplegable del mòdul per Parades de bus.

Una vegada pitgem al botó Calcular Voronoi, obtindrem un resultat com el següent, abans però haurem d'esperar uns segons ja que al haver-hi 144 parades de bus, el programa necessita més temps per executar l'algoritme. Per tant no serà de forma immediata com en els altres dos exemples. Com podem veure a la imatge s'han necessitat 408 línies per completar el diagrama de Voronoi.

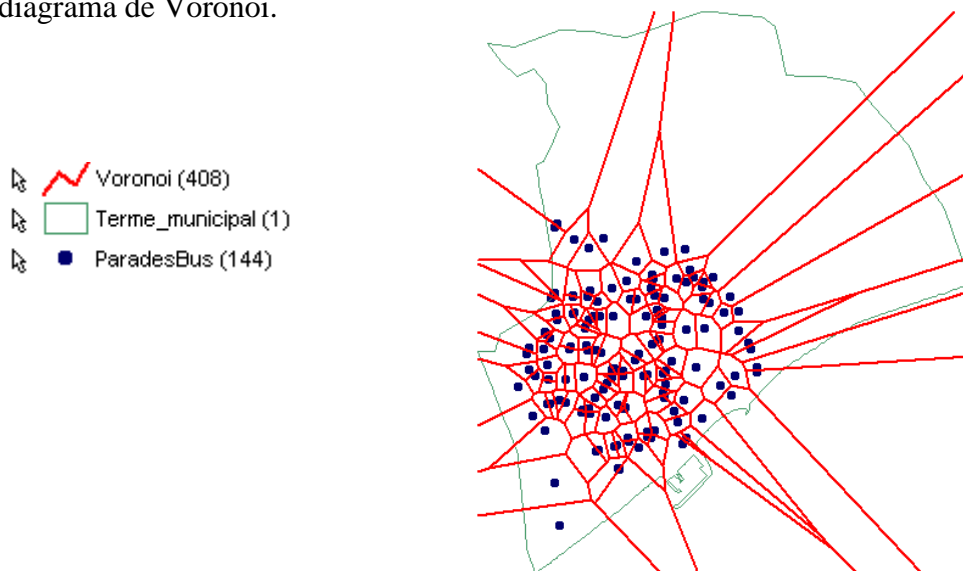


Fig. 6.30. Diagrama de Voronoi per Parades de bus.

En aquest exemple utilitzem el primer mètode de intersecció espacial per arribar al resultat final desitjat. Anem a Anàlisis > Intersecció espacial i omplim els camps de la mateixa manera que en l'exemple de les escoles bressol.

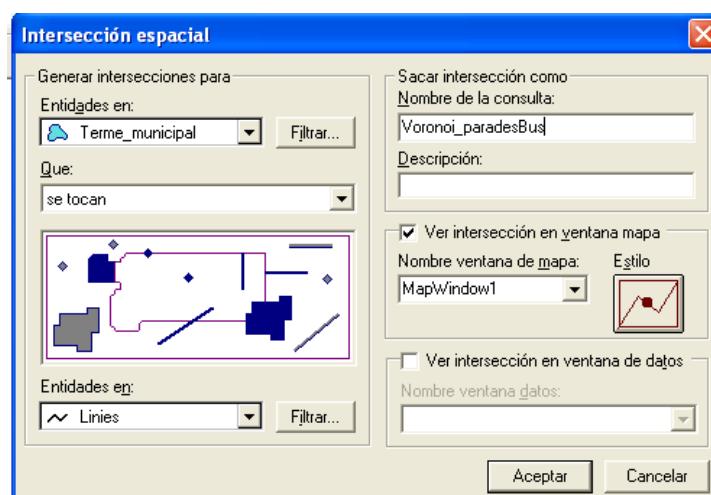


Fig. 6.31. Intersecció espacial entre línies i terme municipal.

Al premer el botó Acceptar obtindrem el Voronoi final de les parades de bus.

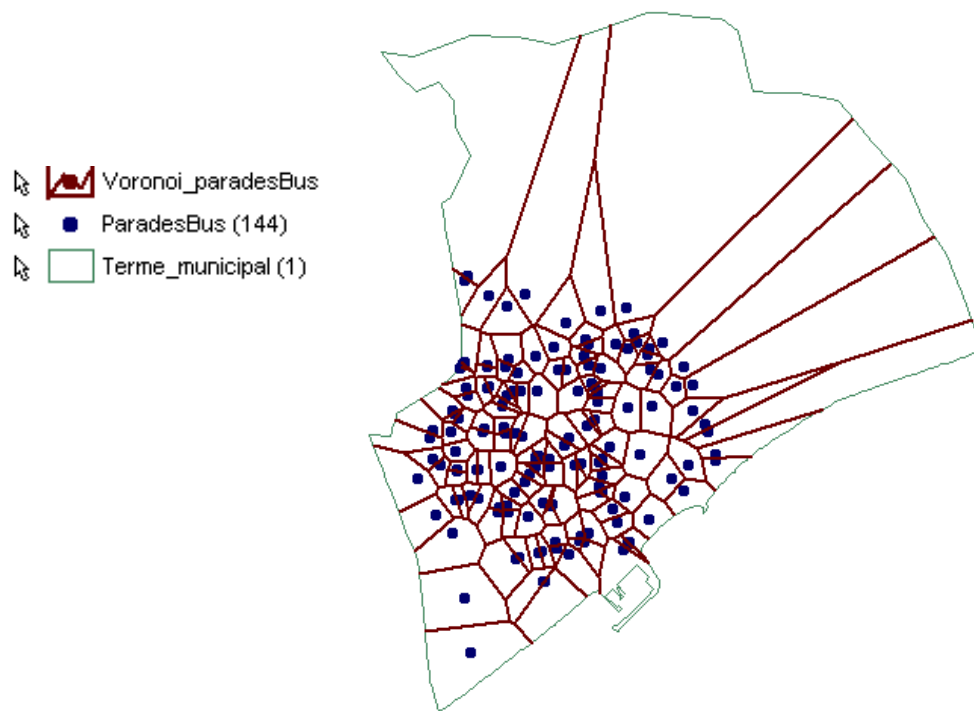


Fig. 6.32. Polígons Voronoi resultants sobre parades de bus mitjançant intersecció espacial.

Si afegim les illes a la llegenda, l'aspecte ens quedaria de la següent forma.

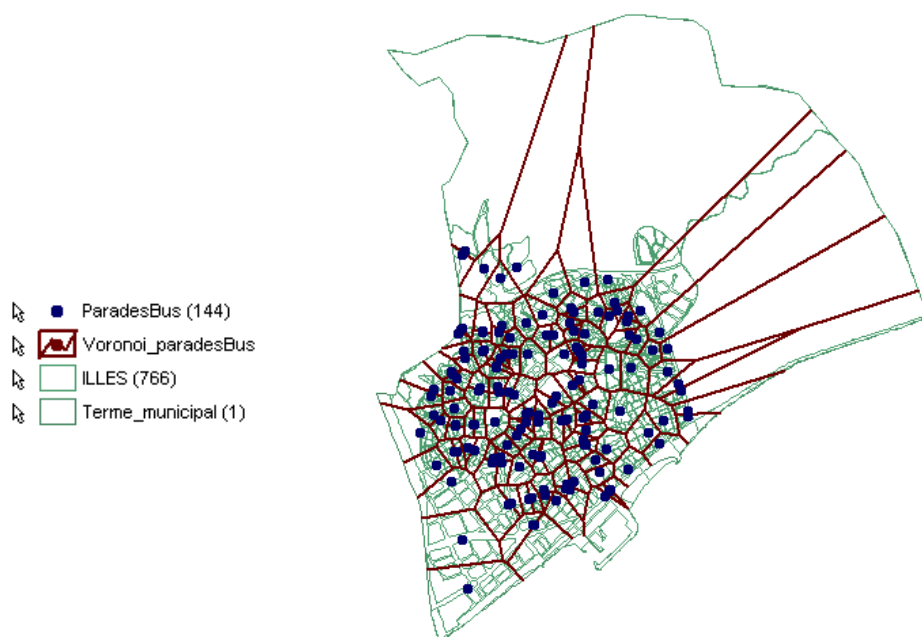
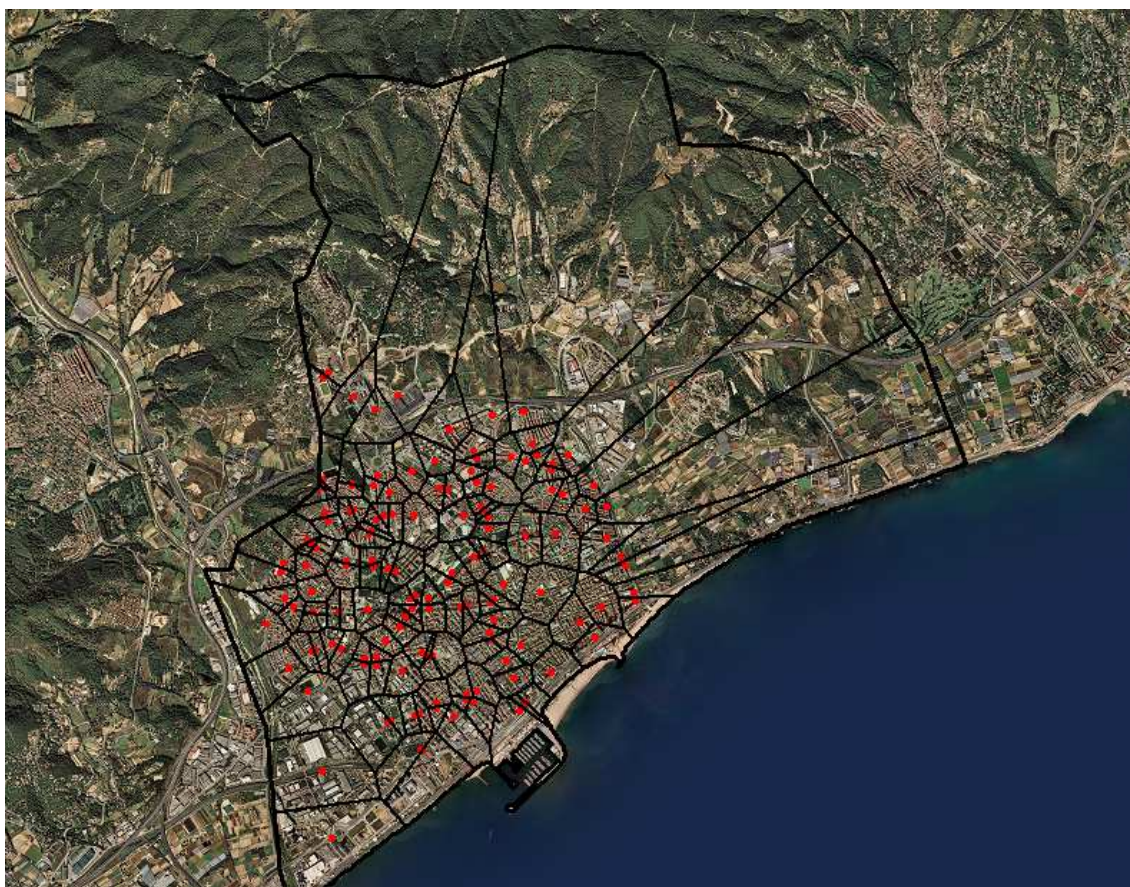


Fig. 6.33. Polígons Voronoi resultants amb la capa d'illes.

Apliquem la ortofoto per visualitzar el resultat de la millor forma possible.



*Fig. 6.34. Polígons Voronoi resultants amb la vista Aèrea.*

#### **6.2.4. Consultes**

Aquest mòdul no només es pot utilitzar per classe d'entitats que estiguin a la base de dades connectada. Aquesta aplicació com la majoria de les creades en el CCU també pot utilitzar consultes que estiguin en memòria. La particularitat de la consulta es que pots seleccionar d'una mateixa classe d'entitat les entitats que tu vols, i després executar el mòdul sobre aquella selecció feta.

Si per exemple volem fer el Voronoi dels CEIPcon en el centre Urbà de Mataró com és mostra en la fotografia, hi ha un CEIPcon que quedaria fora d'aquest centre urbà. El problema vindria, ja que al executar el mòdul ho faria per totes les entitats, inclús la que queda fora del territori, per tant una consulta ens pot permetre eliminar aquesta entitat en memòria sense modificar la base de dades i poder realitzar el diagrama de Voronoi de forma satisfactòria.



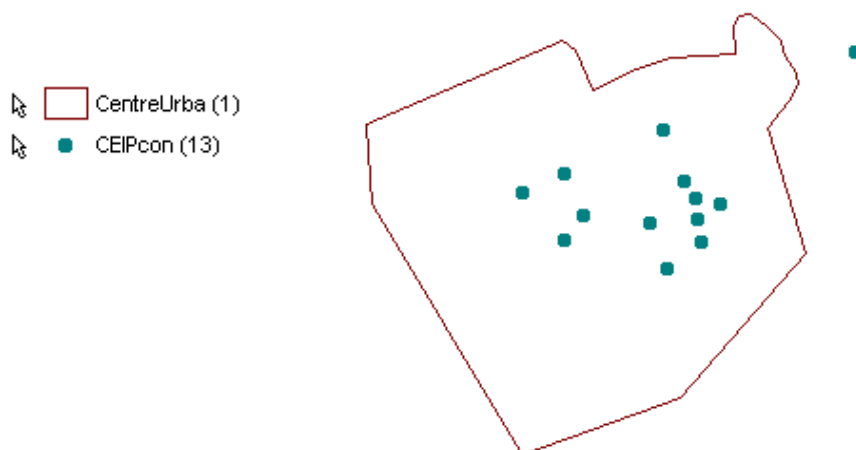


Fig. 6.35. CEIPcon i centre urbà en pantalla.

Per crear la consulta desitjada, anem a Anàlisis>consulta espacial. Seleccionem CEIPcon en el desplegable seleccionar entitats en: . Després haurem de seleccionar la opció "estan contenides en" en el segon desplegable. I Per últim seleccionarem Centre Urba en el tercer desplegable. Finalment introduïrem un nom per la consulta de sortida i pitjarem "Acceptar".

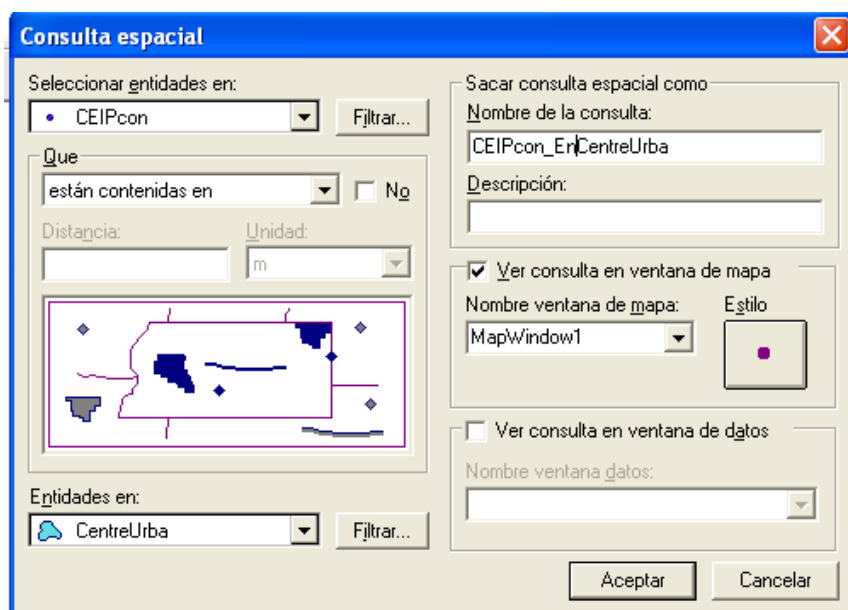


Fig. 6.36. Consulta espacial.

Per tant aquesta consulta el que ens farà serà seleccionar totes les entitats de CEIPcon que estiguin dintre del centre urbà.

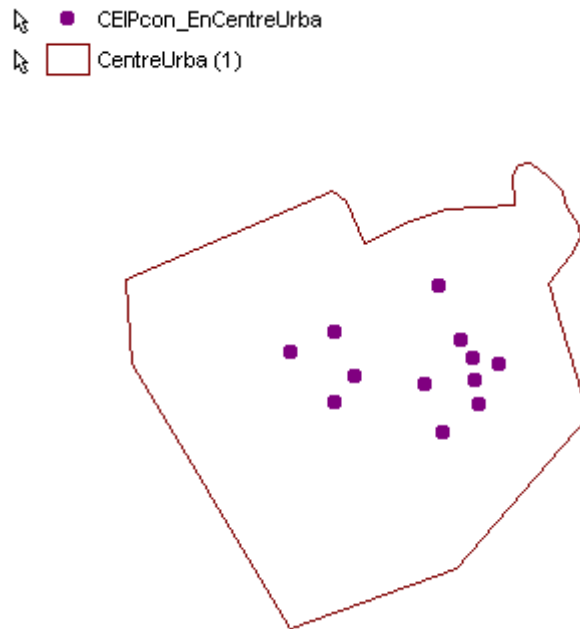


Fig. 6.37. Nova consulta espacial representada a la llegenda.

Ara ja podem utilitzar el mòdul per la consulta desitjada. Simplement accedim al mòdul i en el desplegable de selecció d'entitat puntual, seleccionem la consulta creada en el sub-desplegable de consultes.

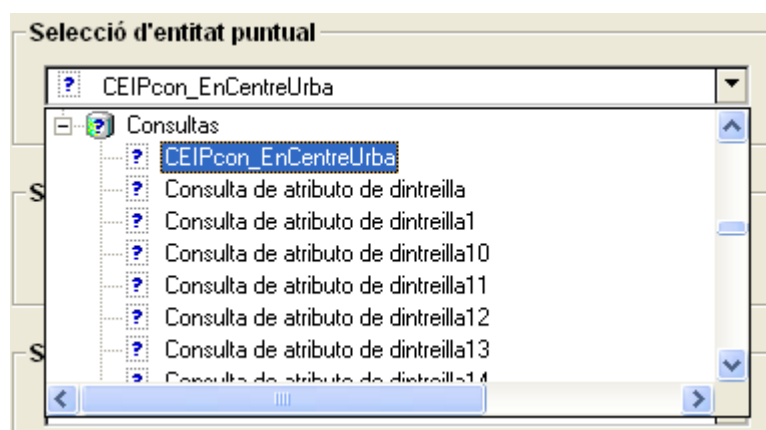


Fig. 6.38. selecció de la consulta creada al desplegable.

Com en exemples anteriors, seleccionem l'àrea delimitada en aquest cas seria el centre urbà i la classe d'entitat de tipus línia de sortida.

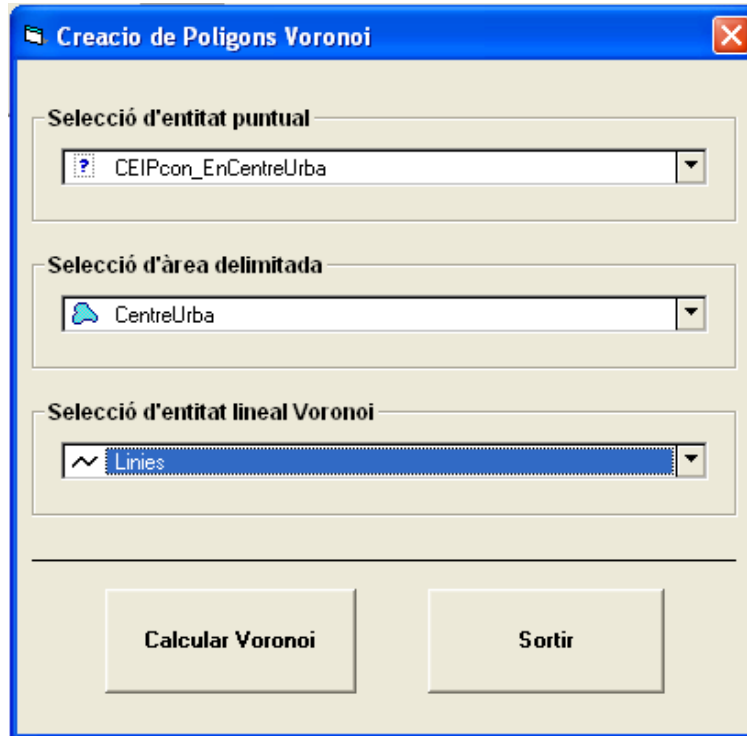


Fig. 6.39. Omplir desplegable del mòdul per escoles bressol.

Obtenim el Voronoi de forma perfecte sense tenir en compte la entitat exclosa.

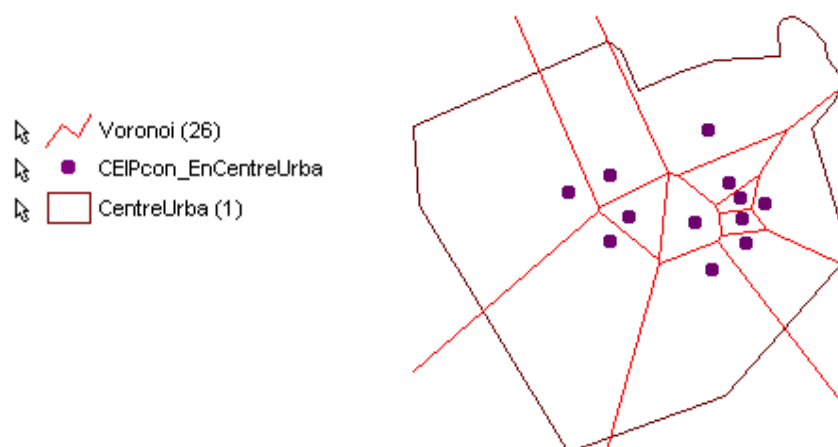


Fig. 6.40. Polígons de Voronoi per la consulta creada.

Utilitzem la opció "dividir entidades" per obtenir el digrama final.

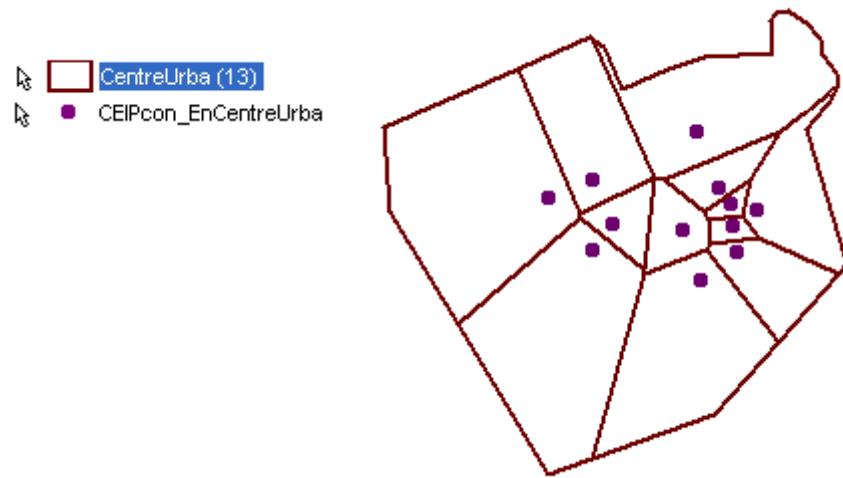


Fig. 6.41. Polígons de voronoi finals per la consulta creada mitjançant dividir entitats.

## 7. Conclusions

Com a conclusió final d'aquest TFG s'obté un estudi en profunditat de les diverses opcions de crear zones d'influència per centres proveïdors de serveis a Mataró utilitzant el mètode del diagrama de Voronoi.

L'objectiu inicial, crear una nova aplicació que permetés calcular els polígons de Voronoi directament des de el Geomedia Professional, ha estat complert de forma satisfactòria com s'ha pogut comprovar en l'anterior capítol. Per tant, El CCU i l'ajuntament de Mataró tindran al seu abast un altre mòdul capaç de generar zones d'influències, a part del ja comentat anteriorment en el capítol 2 (mòdul de creació de zones d'influència circulars/Graf).

Queda pendent en el futur alguna possible millora del actual mòdul, ja que interessaria poder automatitzar la última part, de la intersecció de les línies de Voronoi amb el terme municipal. S'han ofert dos possibilitats per fer-ho utilitzant eines del Geomedia (capítol 6), tot i que es podria intentar fer-ho tot de forma codificada, de tal manera, que una vegada pitgessim el botó de Calcular Voronoi ens mostrés que les línies del diagrama de Voronoi finalitzen el seu recorregut justament en el terme municipal.

En resum, el treball ha estat realment profitós no només per completar el objectiu final, sinó també per el procés d'aprenentatge diari en termes de programació, quelcom que no havia tractat molt a fons en la carrera fins ara i que hem podrà ser de gran ajuda en possibles projectes futurs.

## 8. Referències

[1] Ginés Rodríguez Gavilán, " Representación gráfica de superficies mediante mapas térmicos". Treball final de carrera, 2009.

[http://oa.upm.es/1800/1/PFC\\_GINES\\_RODRIGUEZ\\_GAVILAN.pdf](http://oa.upm.es/1800/1/PFC_GINES_RODRIGUEZ_GAVILAN.pdf)

[2] Manuel Abellanas, "applet Voronoi (java script)".

<http://www.dma.fi.upm.es/mabellanas/voronoi/applet/voronoi-jar.html>

[3] Manuel Abellanas i Alfredo de Las Vegas, "Depthlaunay".

<http://www.dma.fi.upm.es/mabellanas/delonedepth/>

[4] Martha Ivón i Daniel López, "Competència matemàtica i digital: Aplicacions pràctiques dels conjunts de Voronoi". Jornades TIC i Educació- Curs 2012-2013.

[http://web2.udg.edu/ice/jornades\\_tic2/pdf/S%20Cardenas,MI%20Voronoi.pdf](http://web2.udg.edu/ice/jornades_tic2/pdf/S%20Cardenas,MI%20Voronoi.pdf)

[5] Miguel Pineda, "Métodos para el cálculo de precipitaciones en una cuenca". Publicació del Institut universitari politècnic "Santiago Mariño", 7/06/2013.

<http://www.slideshare.net/MIGUELPR1/metodos-calculo-media-en-cuenca-22579168>

[6] Mark de Berg i Marc van Kreveld, "Computational Geometry: Algorithms and Applications". Springer Verlag, 2000.

[7] Francisco Rivero Mendoza, "Geometría Computacional". Publicació de la Universitat de "Los Andes", Gener 2010.

[http://www.ciens.ula.ve/matematica/publicaciones/libros/por\\_profesor/lico/geometria\\_computacional.pdf](http://www.ciens.ula.ve/matematica/publicaciones/libros/por_profesor/lico/geometria_computacional.pdf)

[8] Takashi Ohyam, creador de algoritme Voronoi utilitzat en el projecte. Pagina web oficial del autor, Juny 2010.

<http://www.nirarebakun.com/voro/evoro.html>

[9] GeoMedia, Manual d'usuari del GeoMedia Professional 6.0.

[10] GeoMedia, Referència d'objectes del GeoMedia Transportation Manager.

[11] [www.intergraph.es](http://www.intergraph.es) , Pàgina web de Intergraph.