

REDES AUTOORGANIZATIVAS

1. Introducción a la Autoorganización.
2. Regla de Hebb.
3. Aprendizaje Competitivo.
 - 3.1. Características.
 - 3.2. Ventajas y Limitaciones.
3. Modelo de Kohonen. (Clustering)
4. Mapas Topológicos de Características Autoorganizados de Kohonen.
 - 4.1. Funcionalidad.
 - 4.2. Algoritmo de entrenamiento.
 - 4.3. Propiedades de los mapas topológicos.
 - 4.4. LVQ.
 - 4.5. Aplicaciones de los Mapas Autoorganizados.
 - 4.6. Comparación con técnicas de crecimiento de células.

INTRODUCCIÓN

- Consiste en la modificación repetida de conexiones en respuesta a patrones de activación y siguiendo unas reglas preestablecidas, hasta el desarrollo final de la estructura o sistema.
- La organización en las redes neuronales tiene lugar en 2 niveles diferentes, los cuales están íntimamente interconectados en forma de lazo:

Actividad: Ciertos patrones de actividad son producidos por la estructura en respuesta a señales de entrada.

Conectividad: Los Pesos de las diferentes interconexiones, son modificados en respuesta a señales producidas por los PE. (Concepto de ***Plasticidad***).

PRINCIPIOS:

Principio 1: Modificaciones en los Pesos tienden a *auto-amplificarse*.

El proceso de auto-amplificación está relacionado con el hecho de que las modificaciones de los Pesos están basadas en señales variables localmente.

Principio 2: Limitaciones debido a la *competitividad* establecen la selección de grupos de interconexiones fuertes, a expensas de otras.

Principio 3: Modificaciones en los Pesos tienden a una cooperación entre todos ellos.

Contexto de las Redes Autoorganizativas.

- Existen problemas donde el conjunto de entrenamiento está formado por un conjunto de patrones de entrada y donde las salidas deseadas no están disponibles.
- En estos casos la información relevante debe de ser localizada en los propios patrones de entrada (redundancia).

Problemas tales como:

- **Clustering**: Los datos de entrada pueden ser agrupados en “clusters” y el sistema al procesar los datos debe de encontrar los centros de esos clusters.
- **Cuantización de Vectores**: Este problema ocurre cuando en un espacio continuo tiene que ser discretizado. La entrada al sistema son vectores n-dimensionales y la salida es una representación discreta del espacio de entradas.
- **Reducción de Dimensionalidad**: Los datos de entrada deben de ser agrupados en un subespacio con una dimensionalidad más baja que la dimensionalidad de los datos.
- **Extracción de Características**: El sistema tiene que extraer características de los datos de entrada (supone casi siempre una reducción de la dimensionalidad).

REGLA DE HEBB (Matlab y Neural Nets.Capt 8)

- Esta regla fue introducida por Donald Hebb, apareciendo en su libro titulado "**La Organización del Conocimiento**" en 1949.

La ley nos dice: "*Si un elemento de procesado simple recibe una entrada de otro elemento de procesado y ambos están activos, el peso correspondiente a su conexión debería reforzarse (aumentar)*".

Las **características** en que se basa esta regla son:

Asociatividad.
Autoorganización.

- Trata de asociar patrones de salida a patrones de entrada y la estructura se organiza a sí misma.
- El sistema se puede entender como un decodificador y un codificador.
- Los pesos en estructuras que utilizan para su aprendizaje este tipo de ley, se pueden considerar como la memoria necesaria para sintetizar estas relaciones asociativas.

La Regla de Hebb se puede expresar matemáticamente de la forma siguiente:

$$\nabla W_{ij} = lr * y_i * x_j$$

W_{ij} : conexión entre elemento i y j. (Salida de j entra en i)

y_i : salida neurona i. x_j : salida neurona j

lr : representa la velocidad de aprendizaje (0,1).

ALGORITMO:

1. Inicializar los Pesos de la red a valores pequeños aleatorios.
2. Para cada patrón obtener la salida y modificar los pesos.
3. Repetir el paso 2 hasta que los pesos se estabilicen. Para un número grande de pasos, los pesos W_{ji} converge a la componente i del autovector asociado con la componente j del autovalor de la matriz de correlación de la matriz de entrada.

$$M \text{ (Matriz de Correlación)} = Y * X$$

CARACTERÍSTICAS:

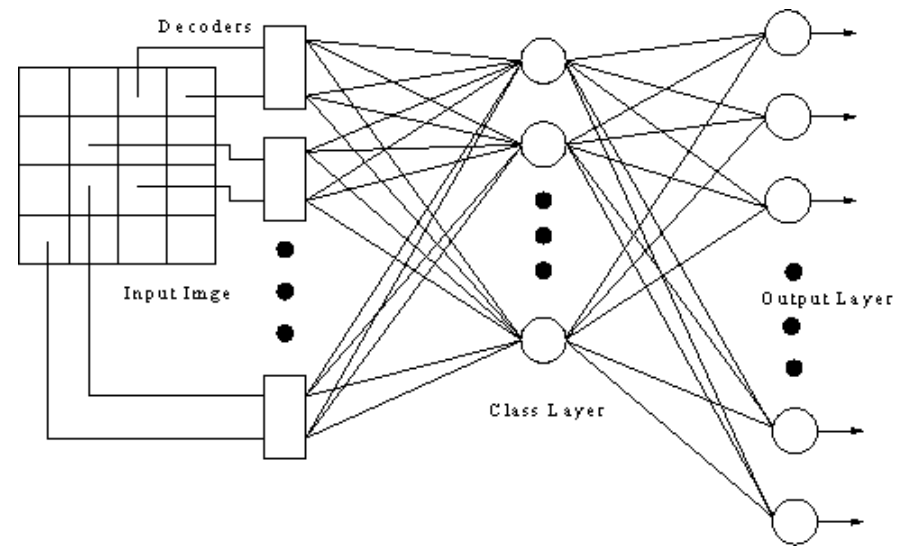
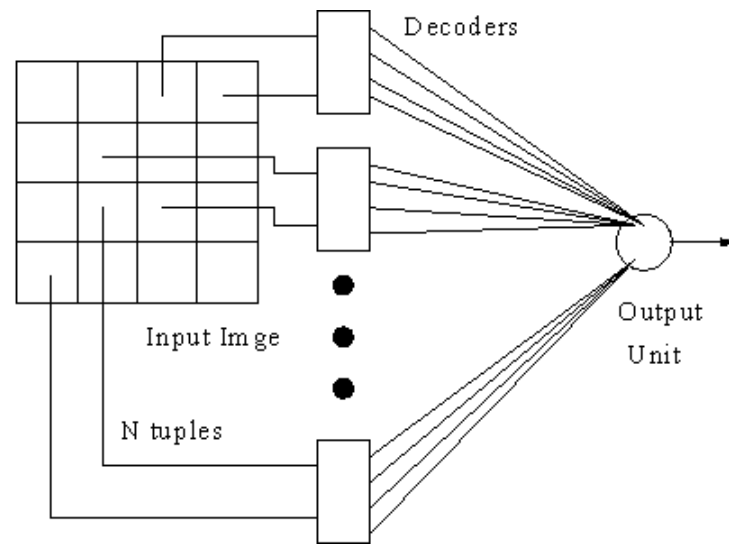
- No existe límite para la matriz de pesos.
- Integración de un *Término de Decaimiento*.

$$\nabla W_{ij} = (lr * y_i * x_j) - (dr * W_{ij})$$

dr: especifica cuanto debe decaer un peso durante un ciclo de entrenamiento. Normalmente es un valor comprendido entre los valores 0 y 1. Si el valor se aproxima a 1, la red tendrá pesos pequeños, pero recordará poco de lo que aprendió en ciclos previos.

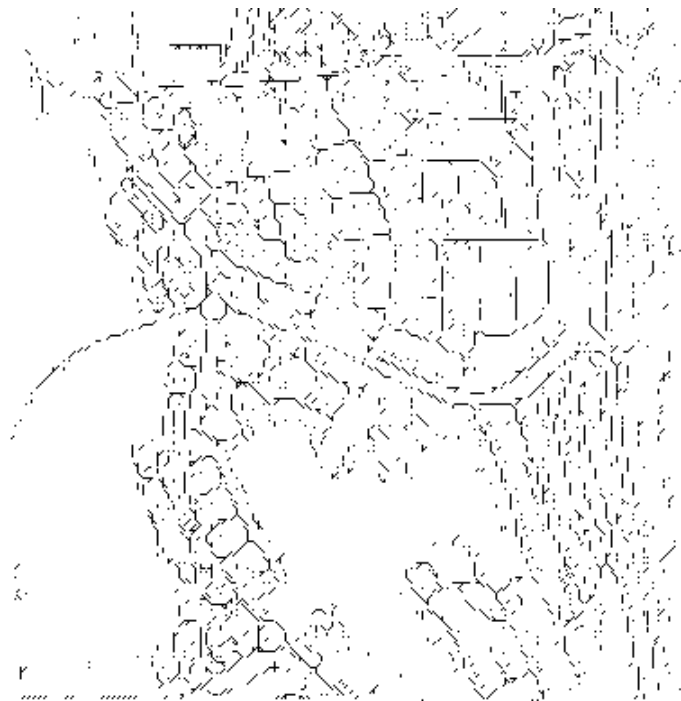
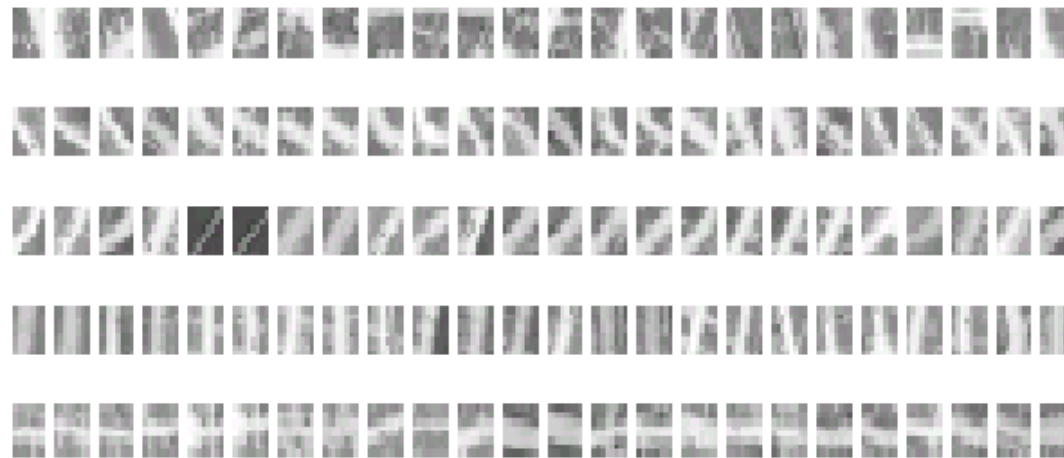
Esta modificación la realizó **Oja** para tratar de estabilizar la regla de Hebb, conociéndose a partir de entonces como *REGLA DE OJA*.

Ejemplo



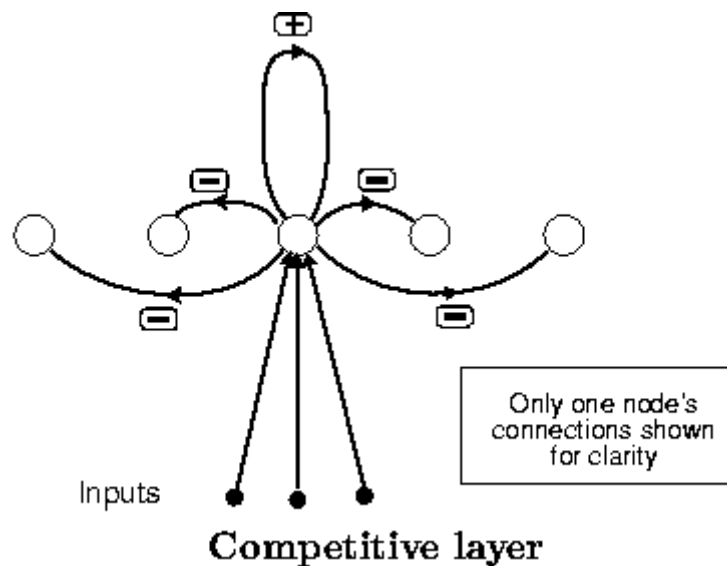






APRENDIZAJE COMPETITIVO

- Esta idea se basa en la existencia de una cierta **competitividad** entre los PEs de una cierta capa por la oportunidad de entrenarse (aprender).
- Esto, se refiere a que, el PE que produce la salida mayor se le considera Ganador, y tiene la capacidad de inhibir a los otros PEs (no presentan activación: salida nula). Todo ello conlleva que solamente los pesos del PE ganador podrán ser ajustados.

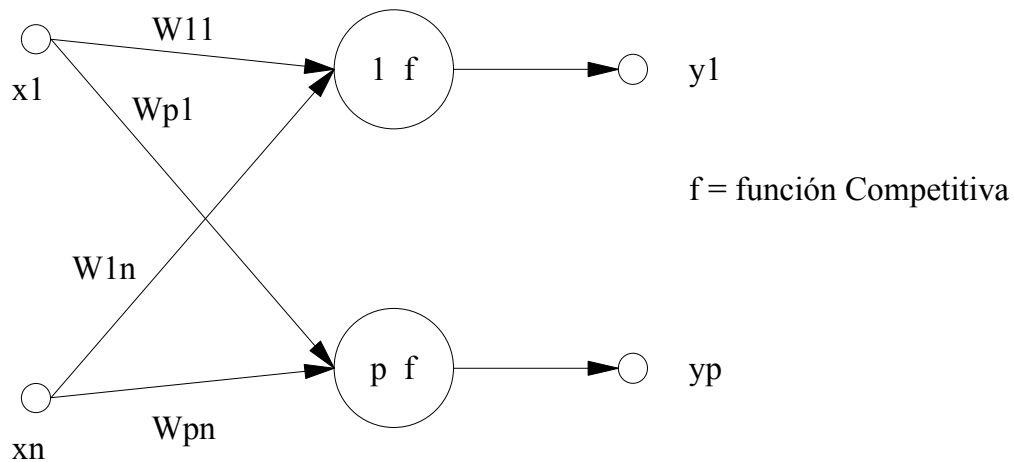


Cada nodo recibe las mismas entradas de una capa previa y entre los nodos de la capa existen conexiones laterales: excitadora (sobre si mismo), inhibidoras (sobre todos los demás nodos de la capa).

MODELO DE KOHONEN. (Clustering)

Matlab, Neural Neurocomputing capt. 5

- Teuvo Kohonen, profesor de la Facultad de Ciencias de la Información (Universidad de Helsinki), trabajó extensivamente, en lo que se denomina *Memorias Asociativas* y en modelos para actividad neurobiológica.
- En líneas generales, las redes entrenadas mediante esta regla se caracterizan por diferentes factores:



Estructura Competitiva

- Asocian vectores de entrada a patrones de salida.
- El aprendizaje es Sin Supervisar.
- Las estructuras de las redes las forman solamente 2 capas.

Se determina la similitud entre los PE y el vector de entrada mediante la formula:

$$d_j = \sum_{i=1}^n (x_i(t) - w_{ji}(t))^2$$

Una vez que se determina cual es el PE ganador (supongamos "k") se aplica la función de transferencia de tipo competitiva:

$$y_j = 0 \quad \forall j \neq k$$

$$y_j = 1 \quad j = k$$

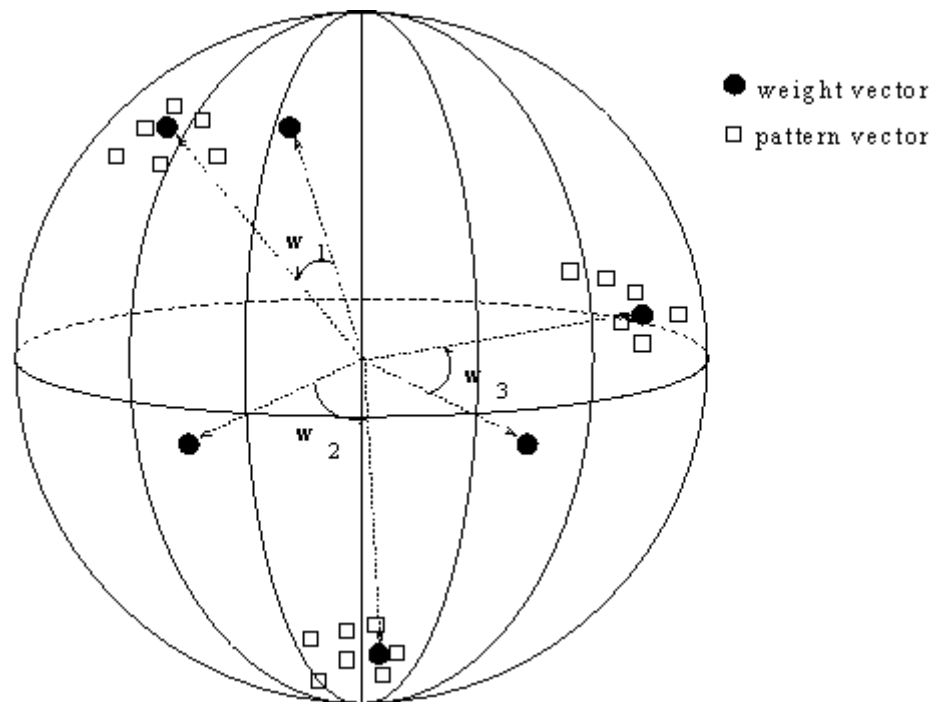
Una vez seleccionado el PE ganador se modifican sus pesos mediante la siguiente regla:

$$w_{ki}(t+1) = w_{ki}(t) + \mu(t)(x_i(t) - w_{ki}(t))$$

- En algunos casos, la velocidad de aprendizaje, suele *ser dependiente del tiempo y normalmente decrece en cada paso*.

- **CARACTERÍSTICAS**

- La velocidad de aprendizaje suele disminuir con el tiempo, hasta que toma un valor próximo a 0 en cuyo caso el aprendizaje finaliza.



Se trata de asociar cada PE de la capa de salida a un grupo de vectores de entrada, con una cierta similitud, generando de tal manera *clases o clusters*. Así, los pesos de los Pes, son interpretados como los centros de los clusters.

Si existen más PE en la capa de salida que clases de patrones de entrada, pueden suceder dos cosas:

- quedan PE inactivos, sin asociar a ninguna clase
- una clase tiene asociados más de un PE.

- Normalmente los patrones de entrada se *Normalizan*. Razón:

El PE ganador se determina calculando una medida de similaridad entre entrada y peso. Dicha similaridad se calcula empleando, normalmente, *la Distancia Euclídea* y ésta compara magnitudes y orientación espacial.

Patrones y Pesos Normalizados.

- Este hecho supone incrementar la velocidad de aprendizaje ya que, existe menos variabilidad en el espacio de pesos.

Por ejemplo: $(1,1,1,1);(4,4,4,4)$ son idénticos---- $(1/2,1/2,1/2,1/2)$.

Una *Limitación de las redes competitivas* es que algunas neuronas pueden no ser entrenadas. En otras palabras, pueden existir vectores de pesos muy distanciados de las entradas, con lo cual, nunca ganarán. Importantísimo en estos sistemas es la inicialización de pesos.

Demostración de que el aprendizaje competitivo sigue la idea de la Regla Delta.

La función error para el patrón p es:

$$E^p = \frac{1}{2} \sum_j (w_{kj} - x_j^p)^2$$

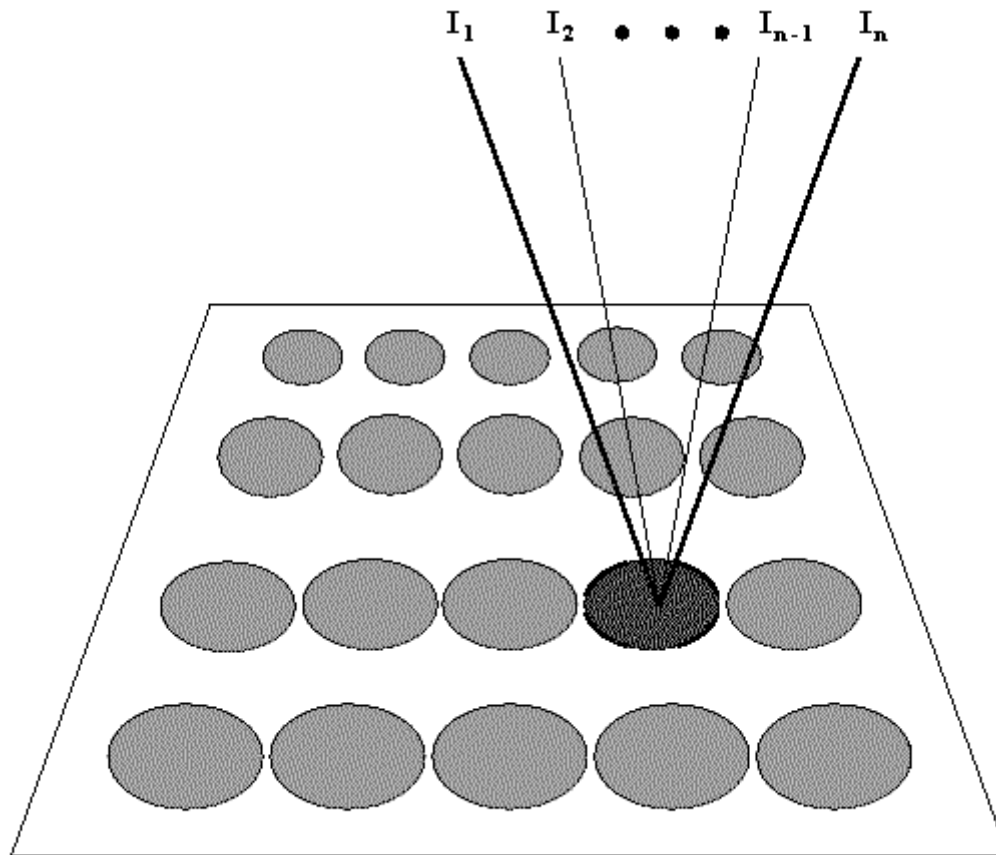
$$\Delta_p w_{ij} = -\gamma \frac{\partial E^p}{\partial w_{ij}}$$

$$\frac{\partial E^p}{\partial w_{ij}} = \begin{cases} w_{ij} - x_j^p & \text{i ganadora} \\ 0 & \text{en otro caso} \end{cases}$$

$$\Delta_p w_{ij} = -\gamma (w_{ij} - x_j^p) = \gamma (x_j^p - w_{ij})$$

MAPAS AUTOORGANIZATIVOS (TOPOLÓGICOS)

- La idea de la *Autoorganización* partió de los diferentes trabajos realizados por Kohonen sobre la región cerebral auditiva.
 - En dicha región, es posible distinguir una ordenación espacial de las neuronas, la cual refleja la respuesta en frecuencia del sistema auditivo. Bajas frecuencias generan respuestas en un extremo de la región y altas frecuencias en el extremo opuesto.
- Ha sido postulado que el cerebro utiliza un *mapeado* espacial para modelar estructuras de datos complejas. Kohonen ha utilizado dicha idea para crear estructuras neuronales utilizando aprendizaje competitivo por una parte y por la otra tratando de generar una *relación espacial o topológica* entre PE (más bien sus pesos), a partir del entrenamiento y siempre en función de los vectores de entrada.



- Estructura formada por dos capas.
- Las neuronas en la capa de salida se ordenan en espacios n-dimensionales.
- Los pesos de las unidades de salida se adaptan de forma que el orden presente en el espacio de entradas persiste en la salida.
- La idea del algoritmo de aprendizaje consiste *en organizar los PEs en el espacio de salida, en regiones (locales)* que actuarán como clasificadores de los datos de entrada.
- El mapa topográfico es organizado de manera autónoma mediante un proceso cíclico de comparación de patrones de entrada (entrenamiento) y vectores de pesos.

- **MÉTODO:**

1- Inicialización de la Red:

Pesos a valores pequeños.

Poner el radio de la región de interés grande.

2- Presentar un vector de entrada. Calcular las distancias para cada PE.

3- Seleccionar el PE ganador (mínima distancia), activando dicho elemento y desactivando los demás.

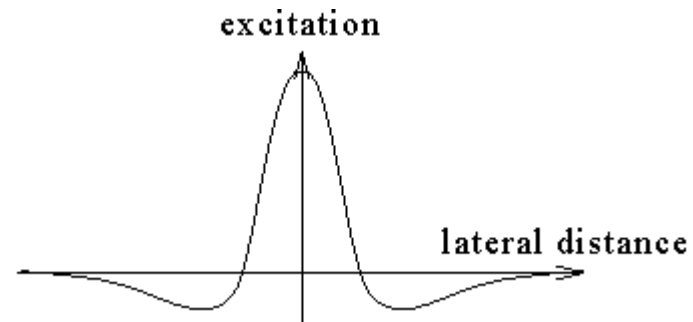
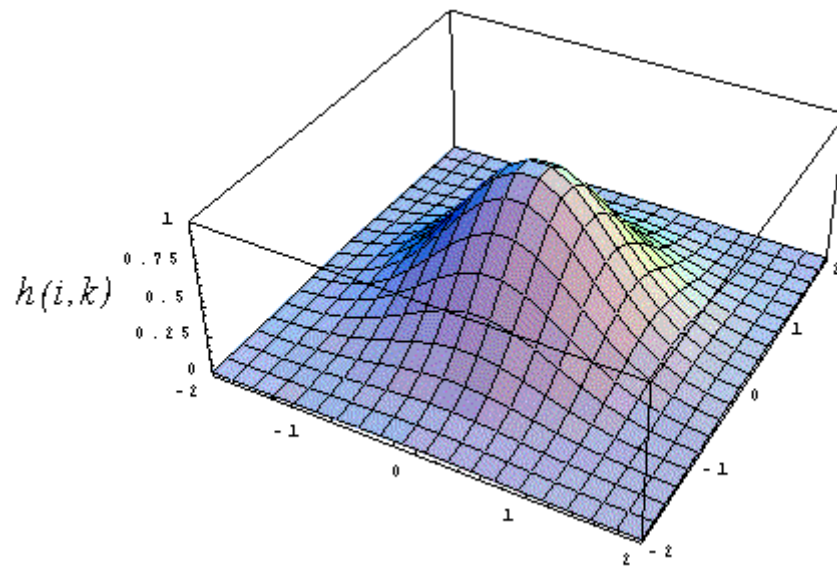
5- Modificar los pesos: Se modifican los pesos del PE ganador y de todos aquellos PEs que se encuentren dentro de la región de interés centrada en el PE ganador.

$$w_i(t+1) = w_i(t) + \gamma h(i,k) (x(t) - w_i(t)) \quad \forall i \in S$$

$h(i,k)$ - función dependiente de la distancia entre unidades.

S- Región de Interés considerada.

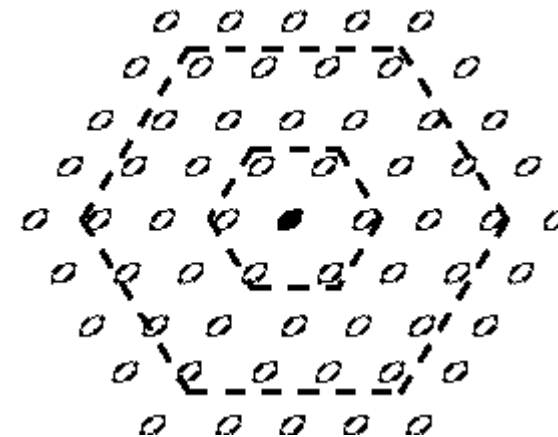
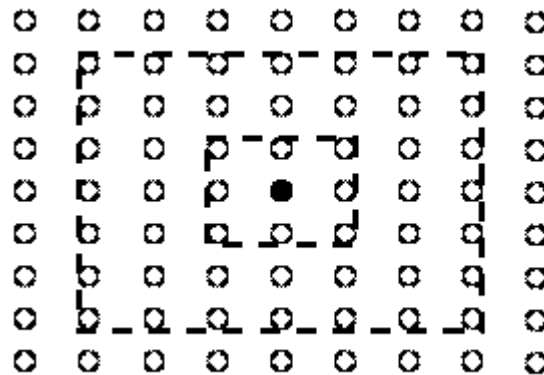
6- volver al paso 2.

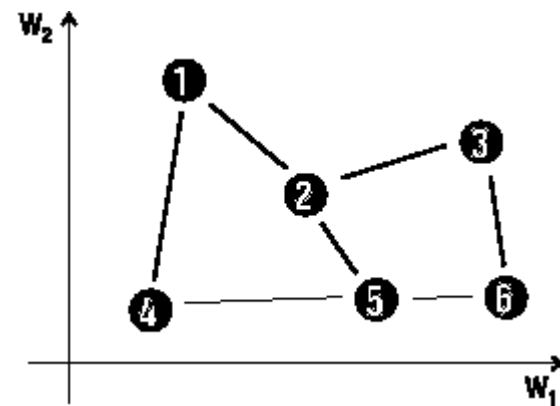
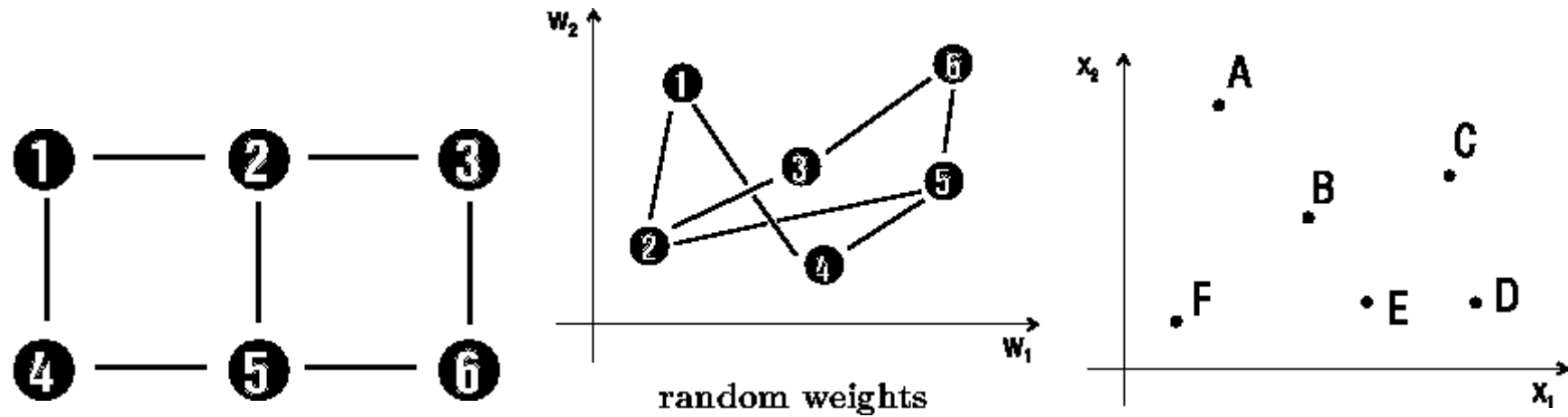


Tipos de $h(i,k)$:

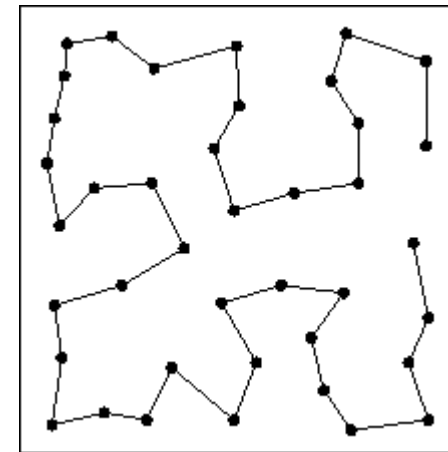
Gausiana y Sombrero Mejicano.

Tipos de Regiones de Interés.





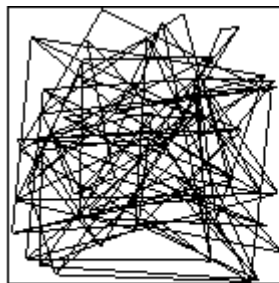
(a)



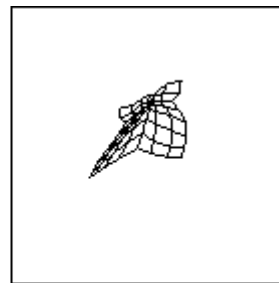
(b)

Espacio de Entrada Bidimensional: a) Espacio de Salida Bidimensional. b) Espacio de salida Unidimensional

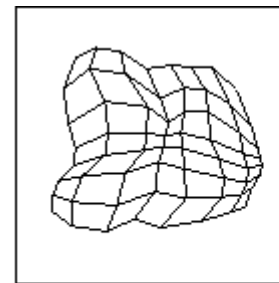
Espacio de entrada bidimensional con patrones uniformemente distribuidos y espacio de salida bidimensional. Evolución de los pesos asignados a cada neurona.



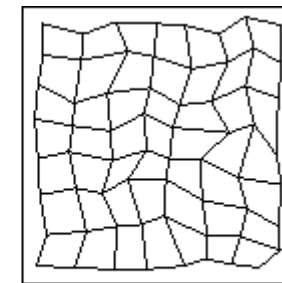
Iteration 0



Iteration 200

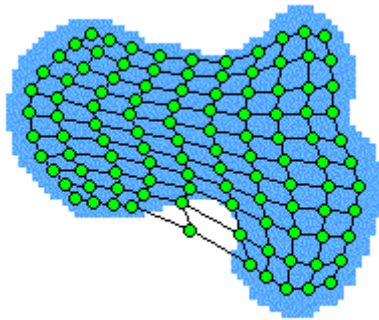


Iteration 600



Iteration 1900

PROPIEDADES DE LOS MAPAS AUTOORGANIZATIVOS



Propiedad 1: Aproximación del Espacio de Entrada.

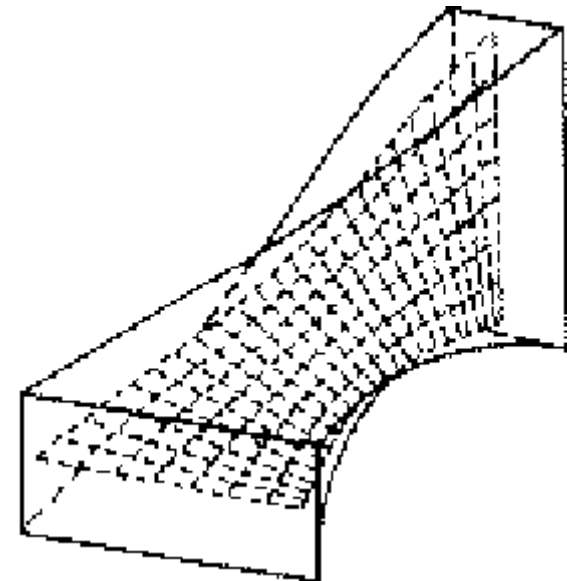
Los Mapas de características, representados por el conjunto de vectores de pesos, nos dan una buena aproximación al espacio de entrada.

Propiedad 2: Ordenación Topológica.

Los Mapas son ordenados topológicamente en el sentido de que la localización espacial de una neurona corresponde a un dominio particular o conjunto de patrones de entrada.

Propiedad 3: Densidad de Muestreo.

Los Mapas autoorganizativos reflejan variaciones en las estadísticas de la distribución de entrada: regiones en el espacio de entrada, cuyos vectores tienen altas probabilidades de ocurrencia, son mapeados en largos dominios del espacio de salida y entonces con mejor resolución que regiones de las cuales sus patrones ejemplo tienen pocas probabilidades de ocurrencia.

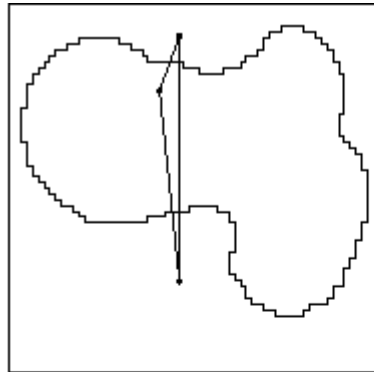


CARACTERÍSTICAS:

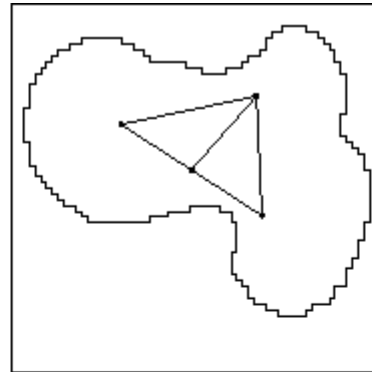
- Con respecto a pesos: lo mismo que para el aprendizaje competitivo.
- Los nodos pertenecientes a la región de interés del ganador son también modificados. La razón de este hecho es que lo que se trata de conseguir es que la red cree regiones que respondan a valores muy próximos al del vector de entrenamiento. Los PE que se encuentren próximos al ganador tendrán un alineamiento similar y sobre el ciclo de entrenamiento esto será una ventaja en la representación de la clase para dicha entrada. Como consecuencia de que son similares espacialmente a vectores de entrenamiento, serán correctamente clasificados incluso aunque éstos no formen parte del conjunto de entrenamiento. Esto demuestra la *Generalidad* de esta estructura.
- La región de interés normalmente, se reduce de manera lineal dependiendo del nº de ciclos del entrenamiento:
- Las dos ideas centrales en las que se basa Kohonen para desarrollar sus estructuras usando aprendizaje *autoorganizativo* y competitivo son: "*El proceso de adaptación de pesos y el concepto de geometría topológica de PEs*".

APLICACIONES:

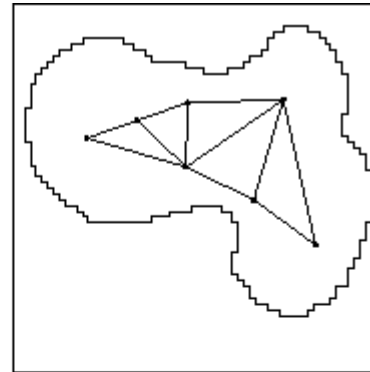
Crecimiento de Células



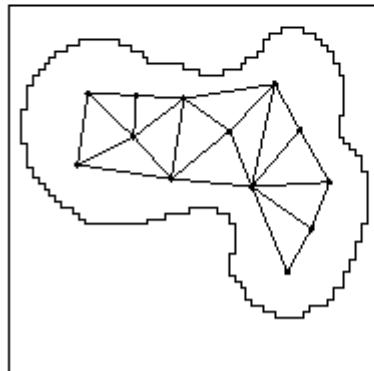
a) 0 signals



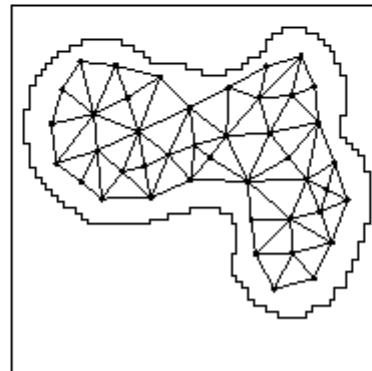
b) 100 signals



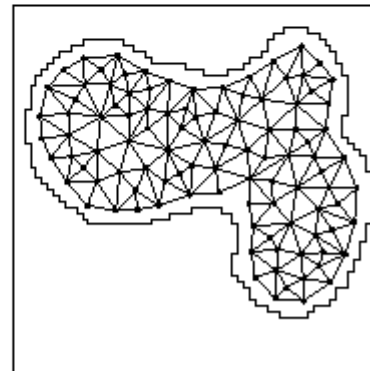
c) 400 signals



d) 1000 signals

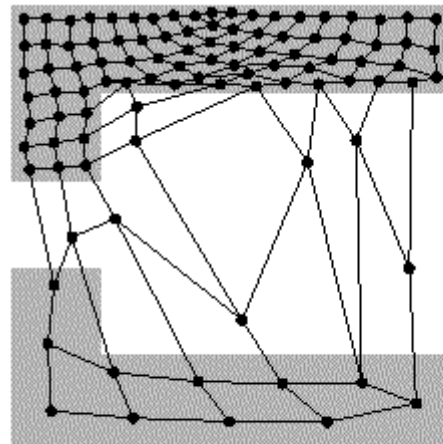


e) 4000 signals

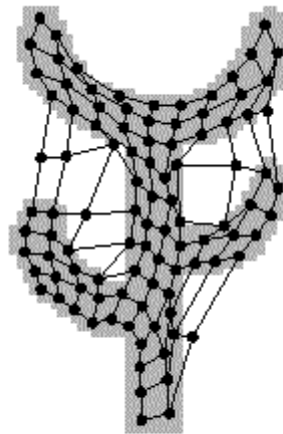


f) 10000 signals

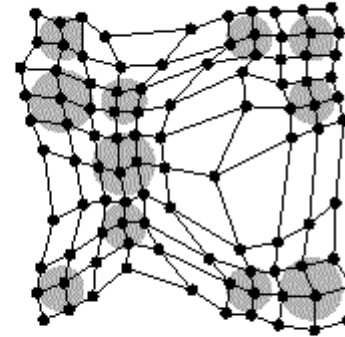
- Misma filosofía que los mapas de Kohonen.
- Si se cumplen ciertas condiciones con respecto a distancias entre vecinos se introducen nuevas células.



a)

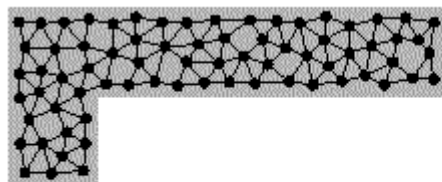


b)

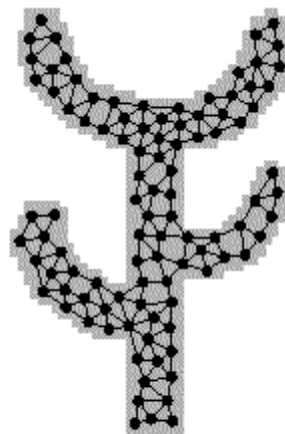


c)

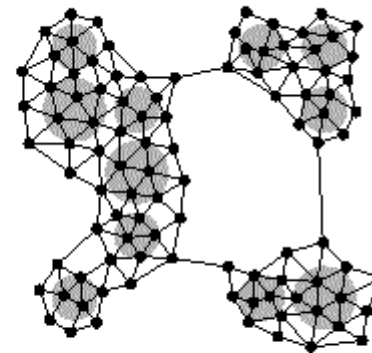
Mapa de Kohonen



a)

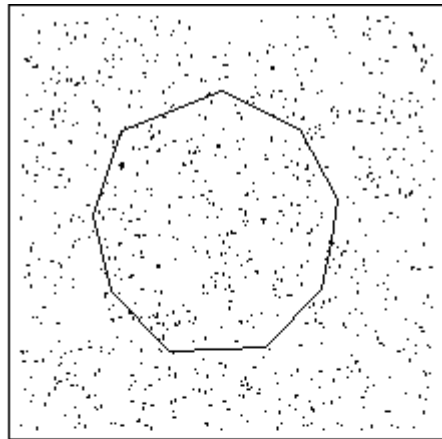


b)

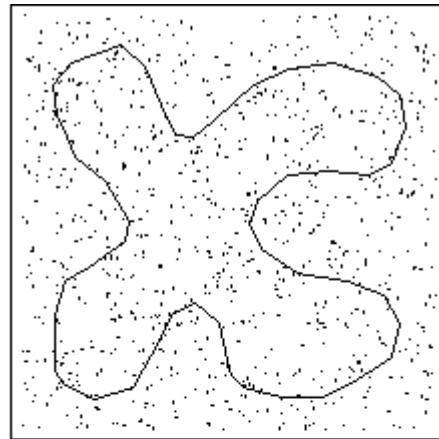


c)

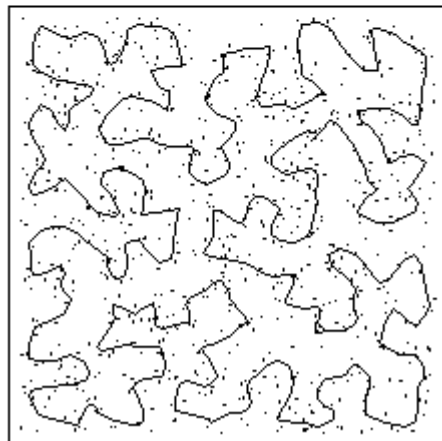
Crecimiento



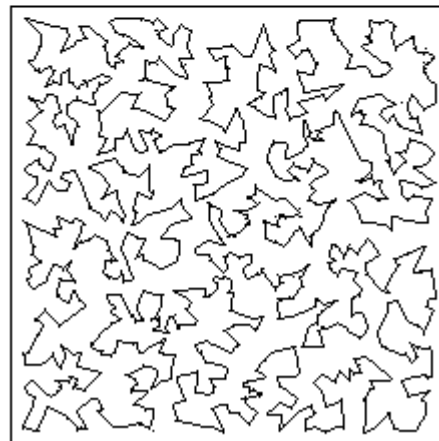
10 cells



50 cells



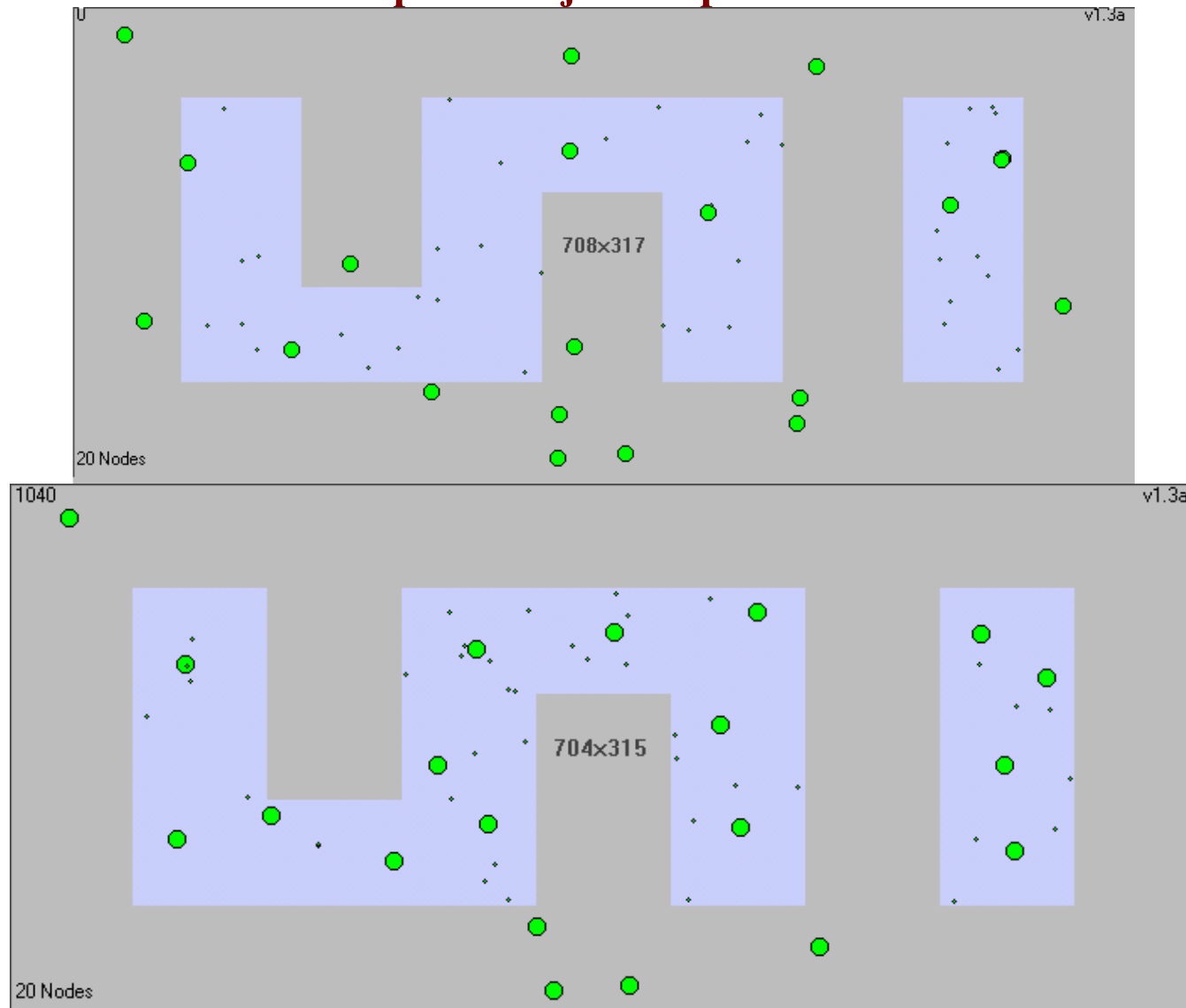
500 cells

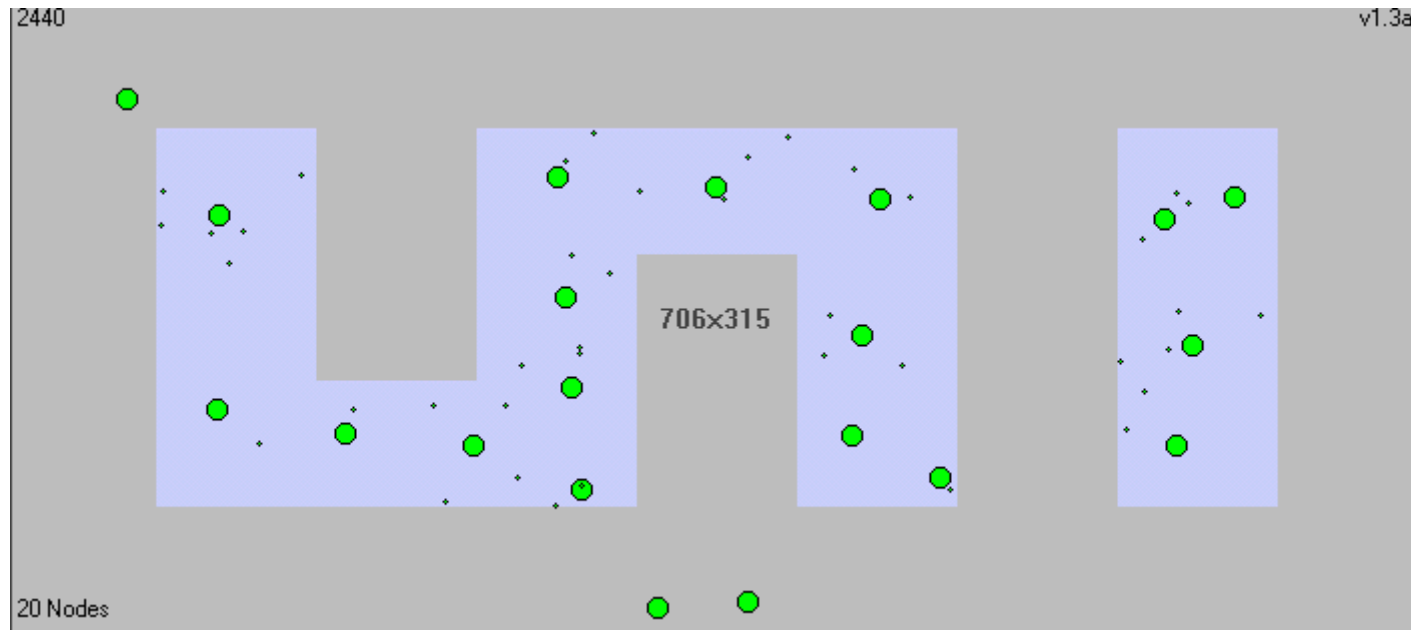


2000 cells

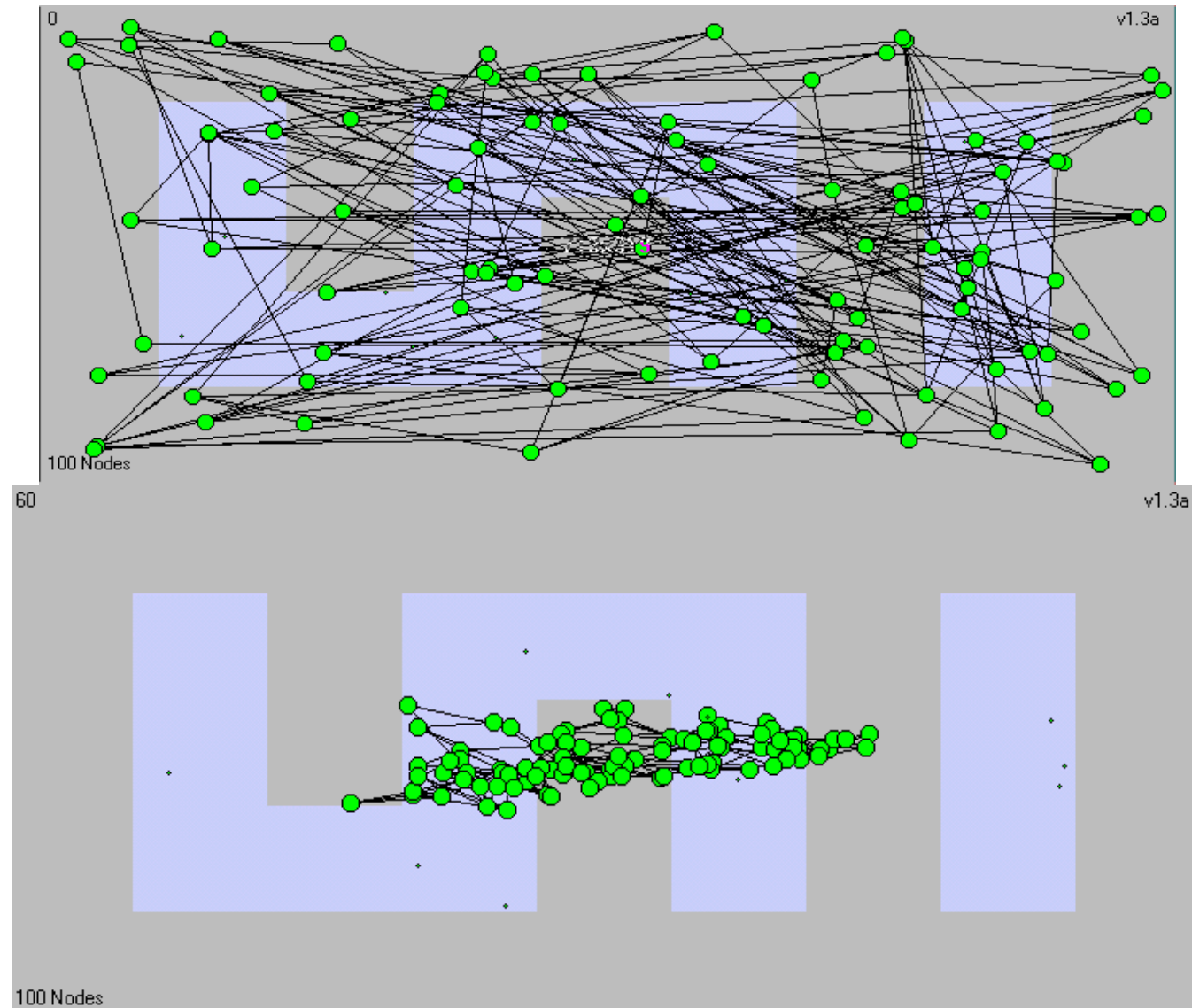
Problema de TSP mediante crecimiento de Células.

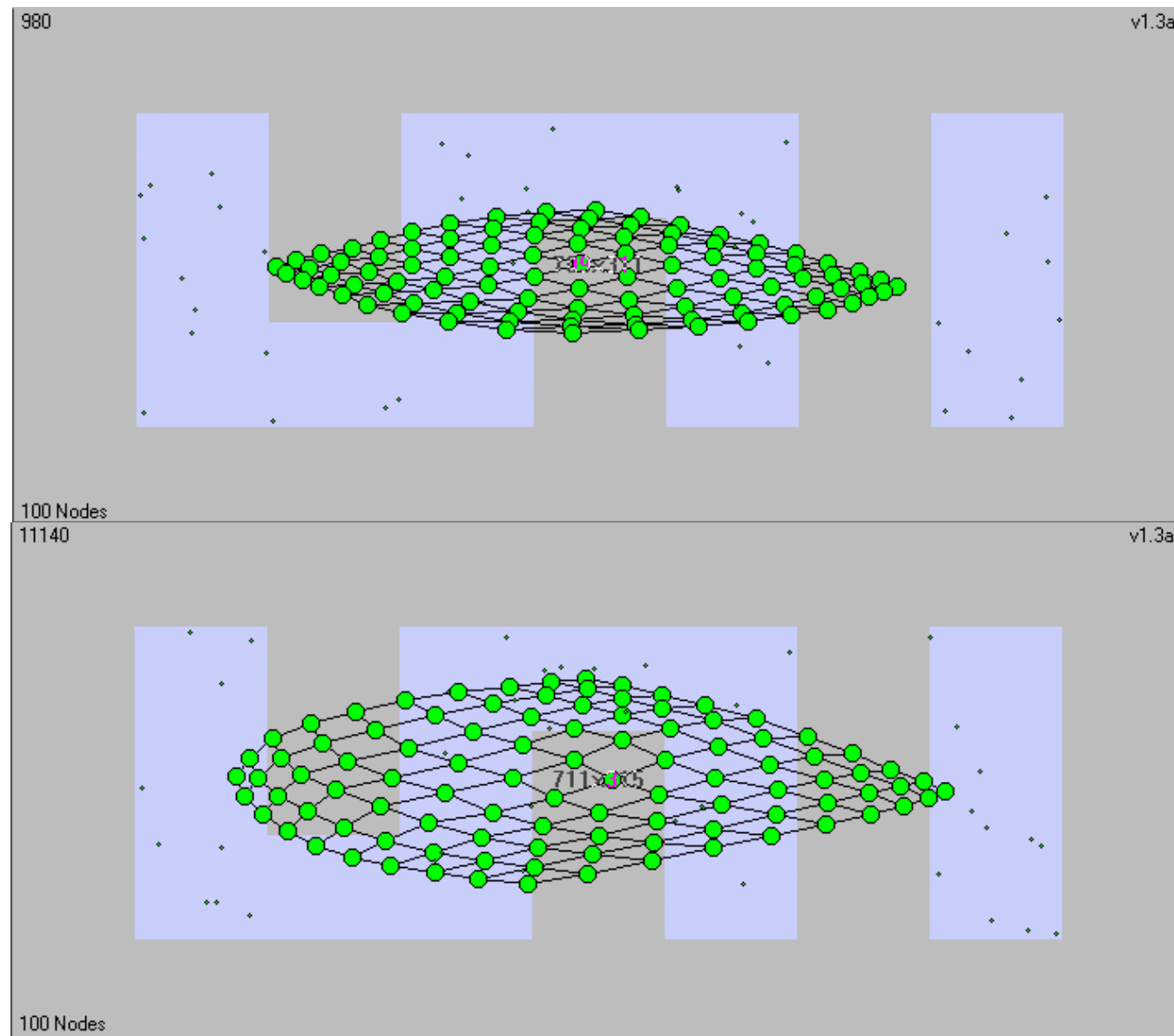
Aprendizaje Competitivo

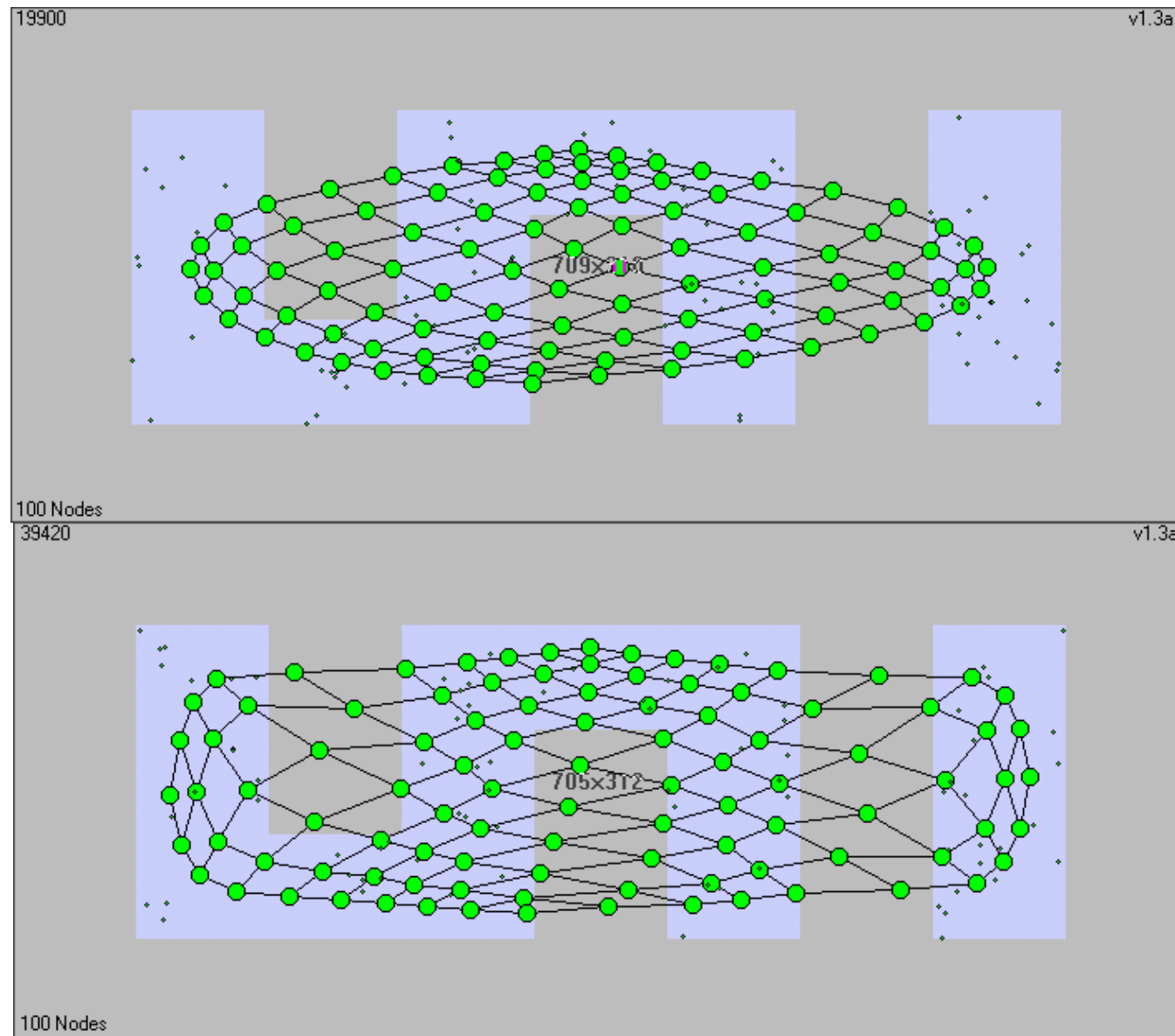




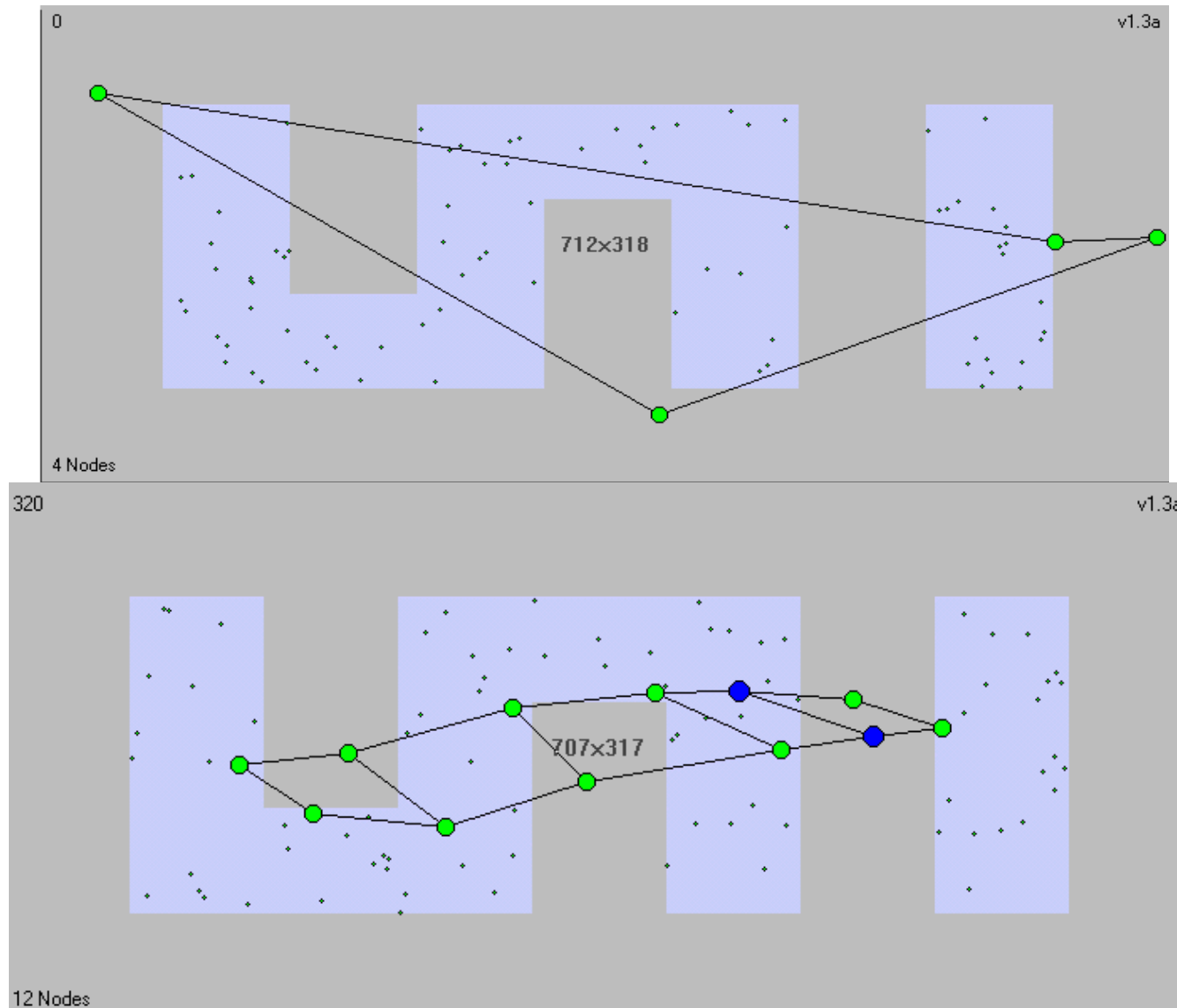
Red de Kohonen

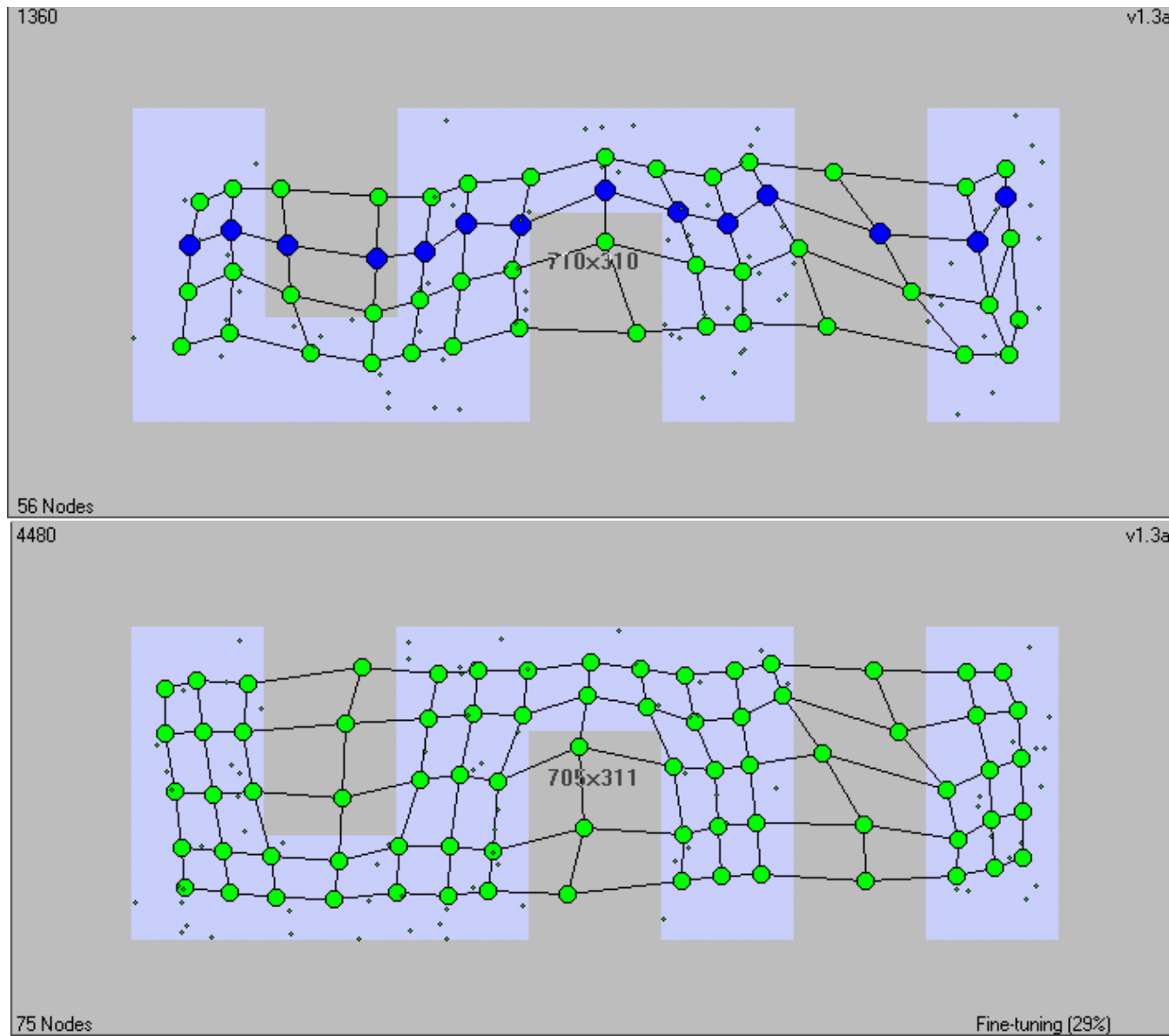




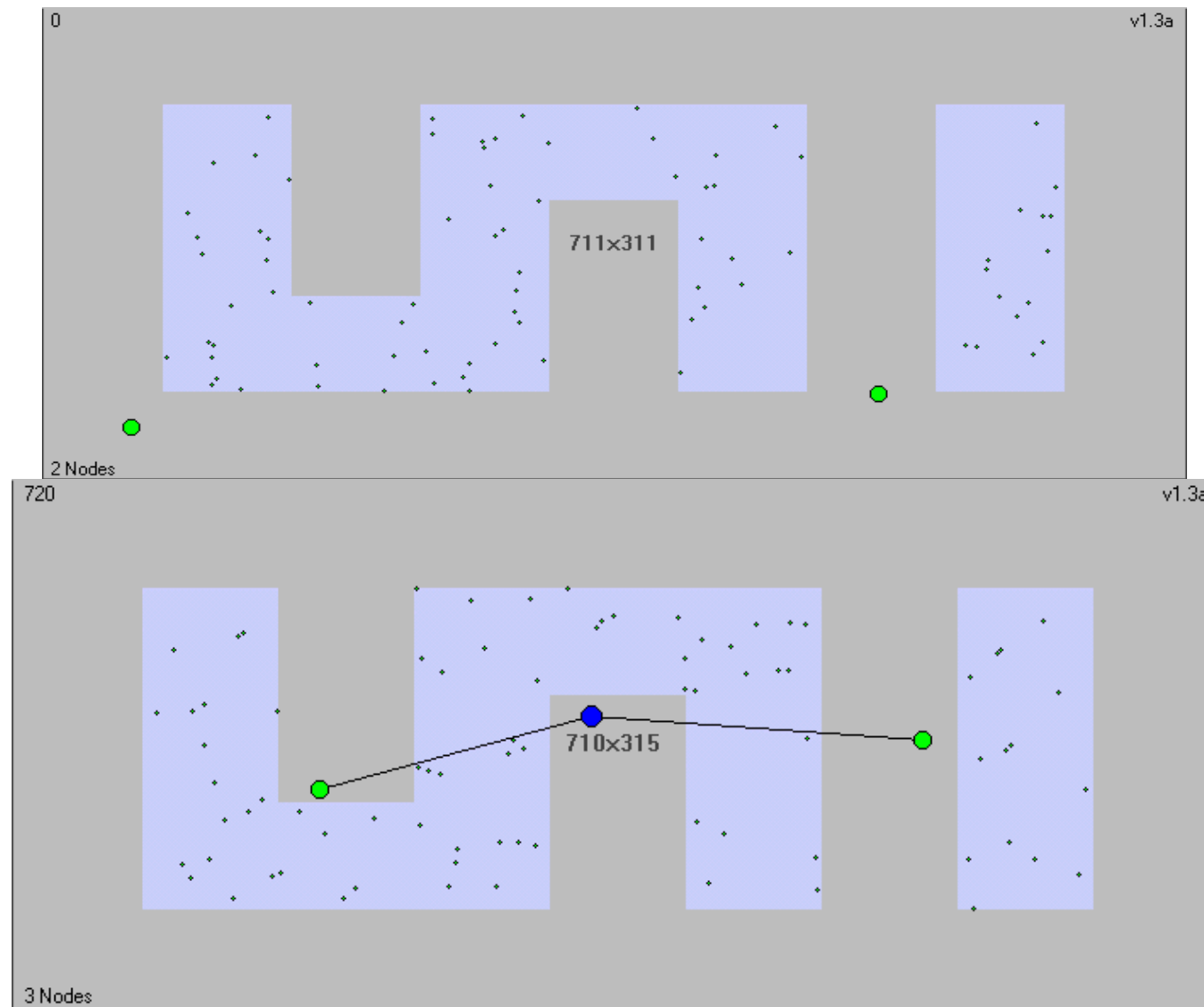


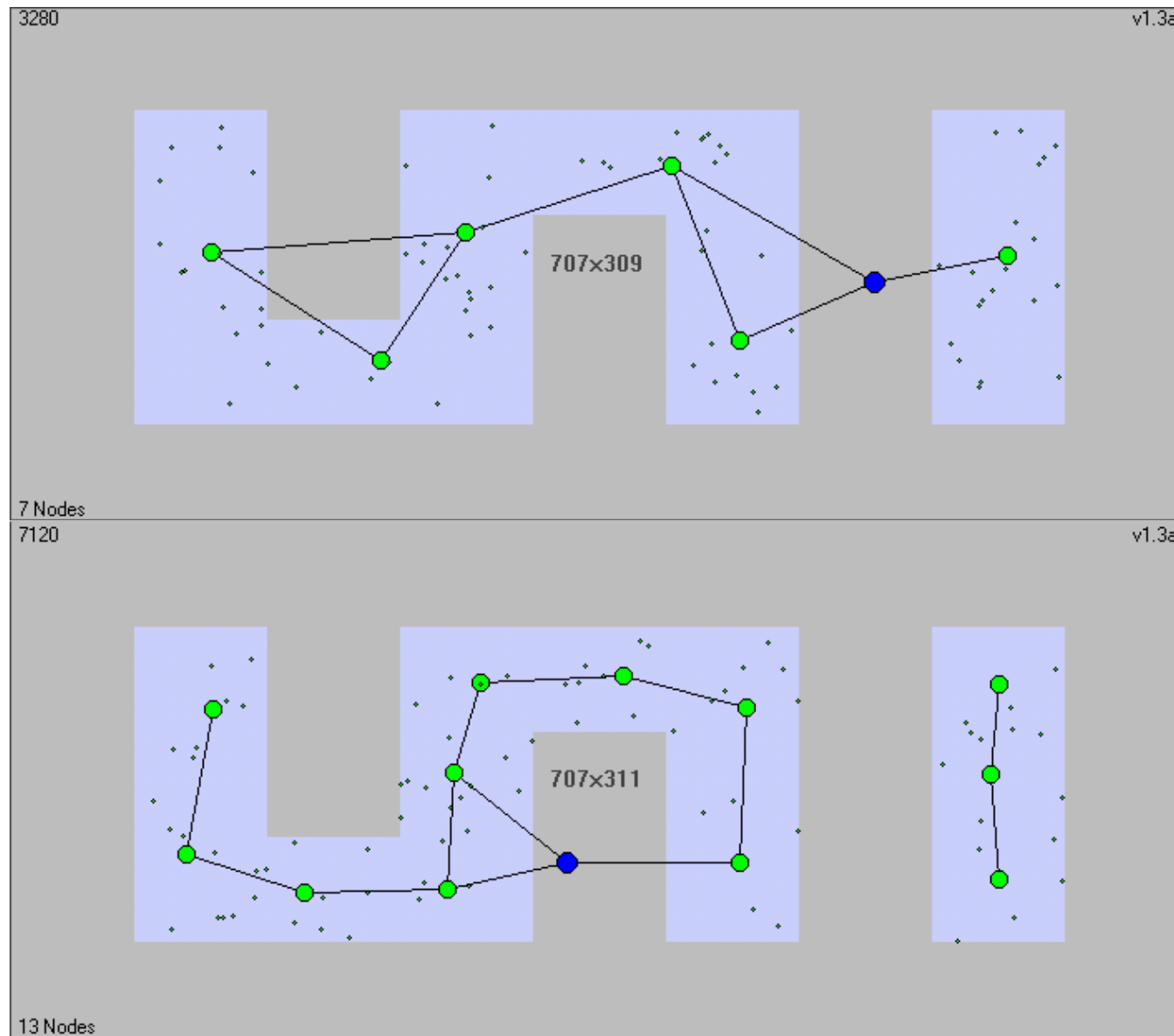
Crecimiento de Mallas

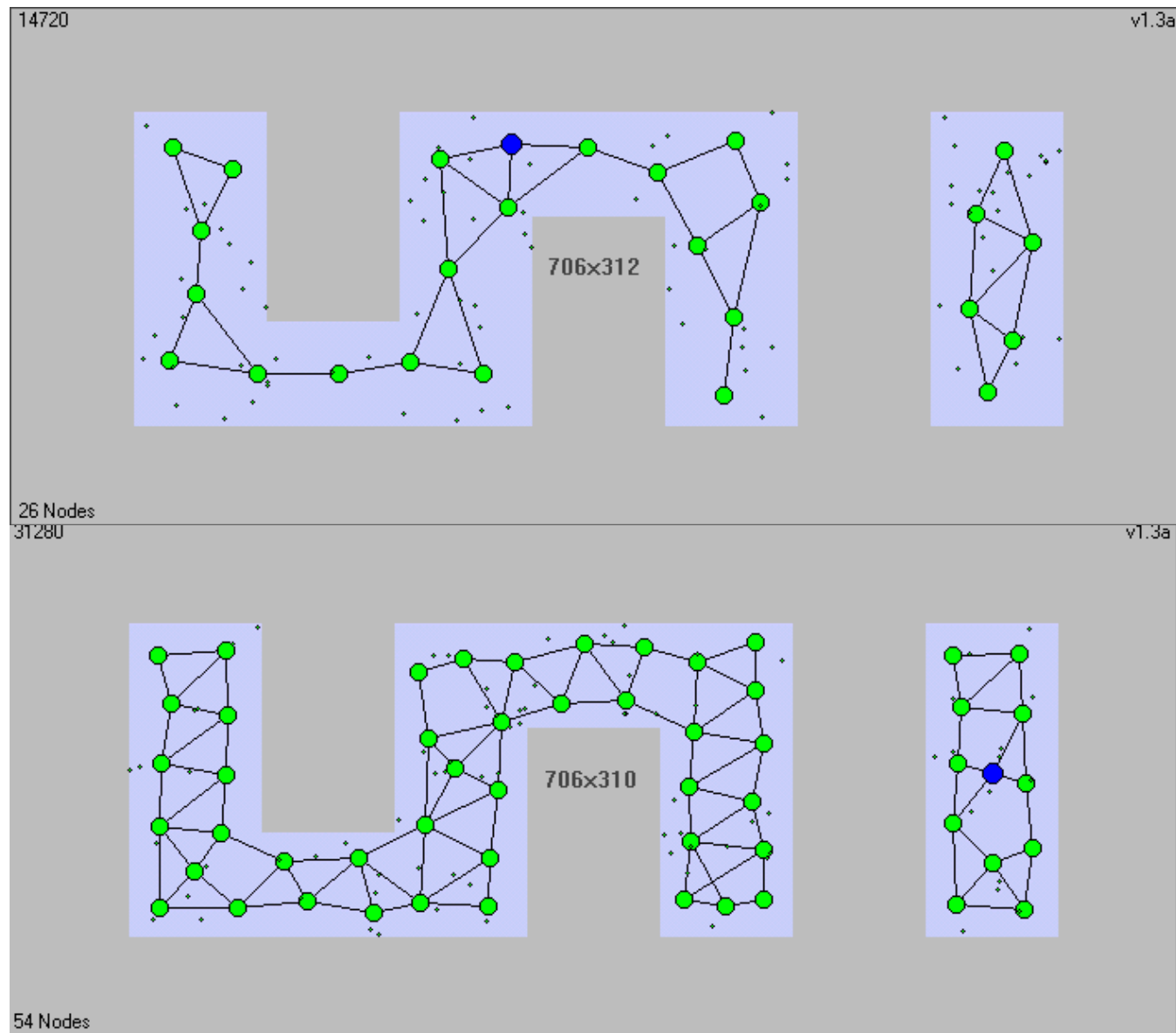




Crecimiento de Células







Segmentación de Imágenes

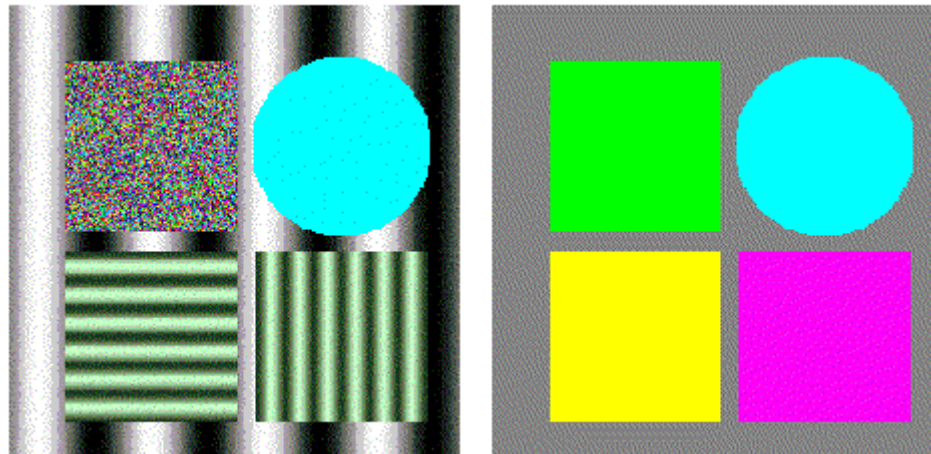


Imagen Original

Imagen Segmentada Ideal

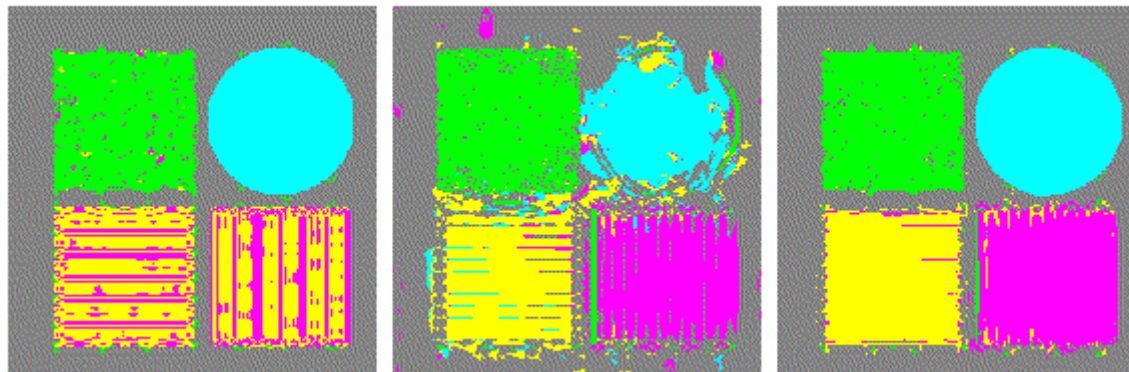
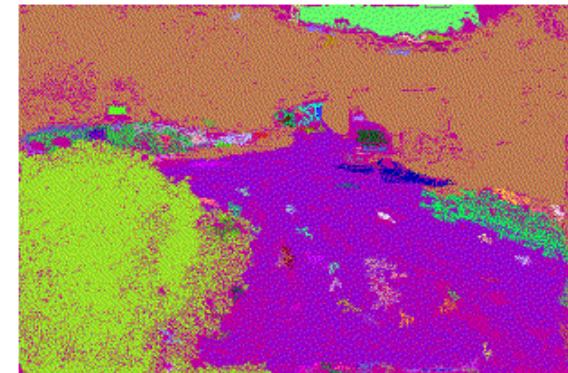
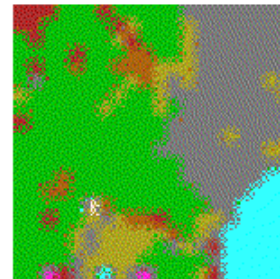
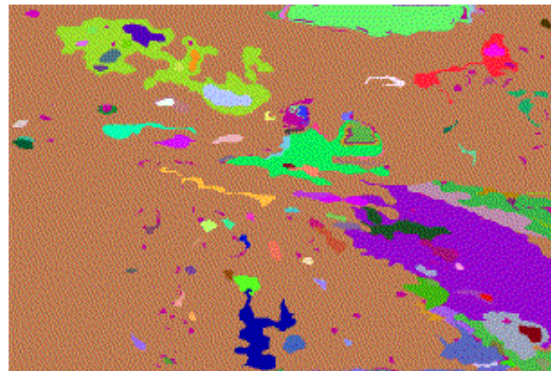
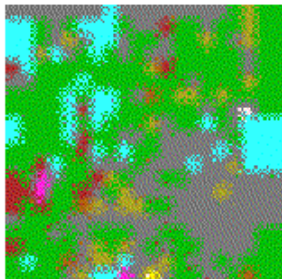
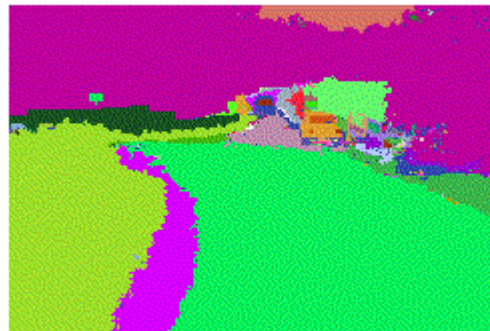


Imagen segmentada (Color)

Imagen segmentada (Textura)

Imagen Segmentada (Color y Textura)

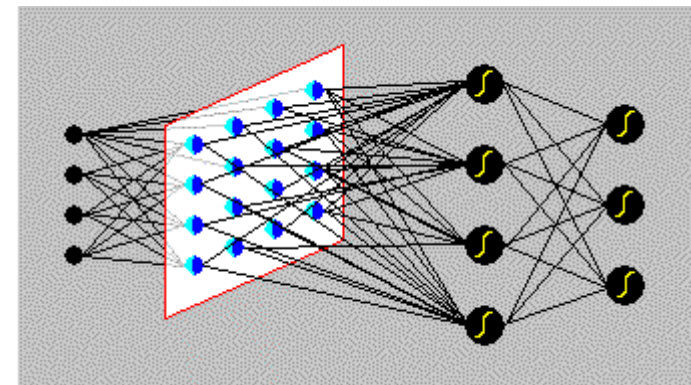
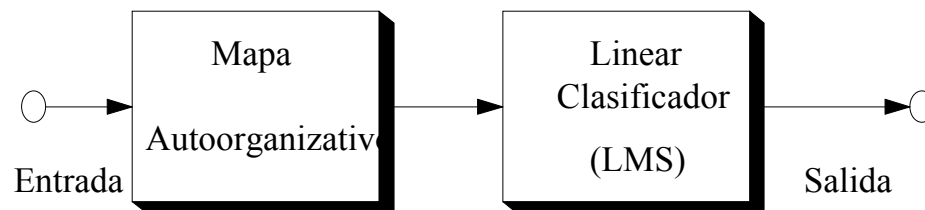


Segmentación Color

Segmentación Color y Textura

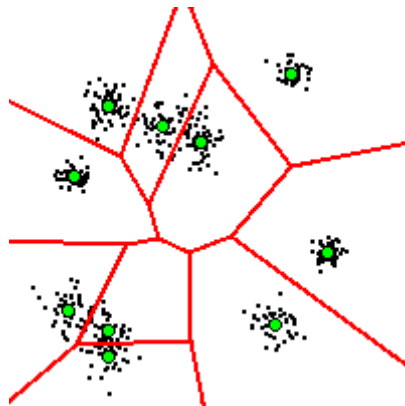
Clasificador de Mapa de Características. Learning Vector Quantization (LVQ)

- En problemas de clasificación de patrones, el requerimiento consiste en clasificar un conjunto de patrones de entrada en un número finito de clases. En estos problemas, una de las tareas más importantes consiste en delimitar las condiciones de separación de clases.
- Hasta el momento se han descrito métodos tanto en aprendizaje supervisado como en aprendizaje sin supervisar que trataban de resolver este problema. Diferentes estudios han llegado a demostrar que en ciertos casos un *método Híbrido* soluciona mejor este problema. La solución pasa por usar una combinación de un Mapa de características y un clasificador lineal con aprendizaje supervisado (Lippmann, 1989).



LVQ. Vector Quantization.

- **Vector Quantization** es una técnica por la cual el espacio de entradas es dividido en un número determinado de regiones y para cada una de ellas es definido un vector que la caracteriza. Un espacio continuo tiene que ser discretizado. La entrada al sistema son vectores n-dimensionales y la salida es una representación discreta del espacio de entradas.



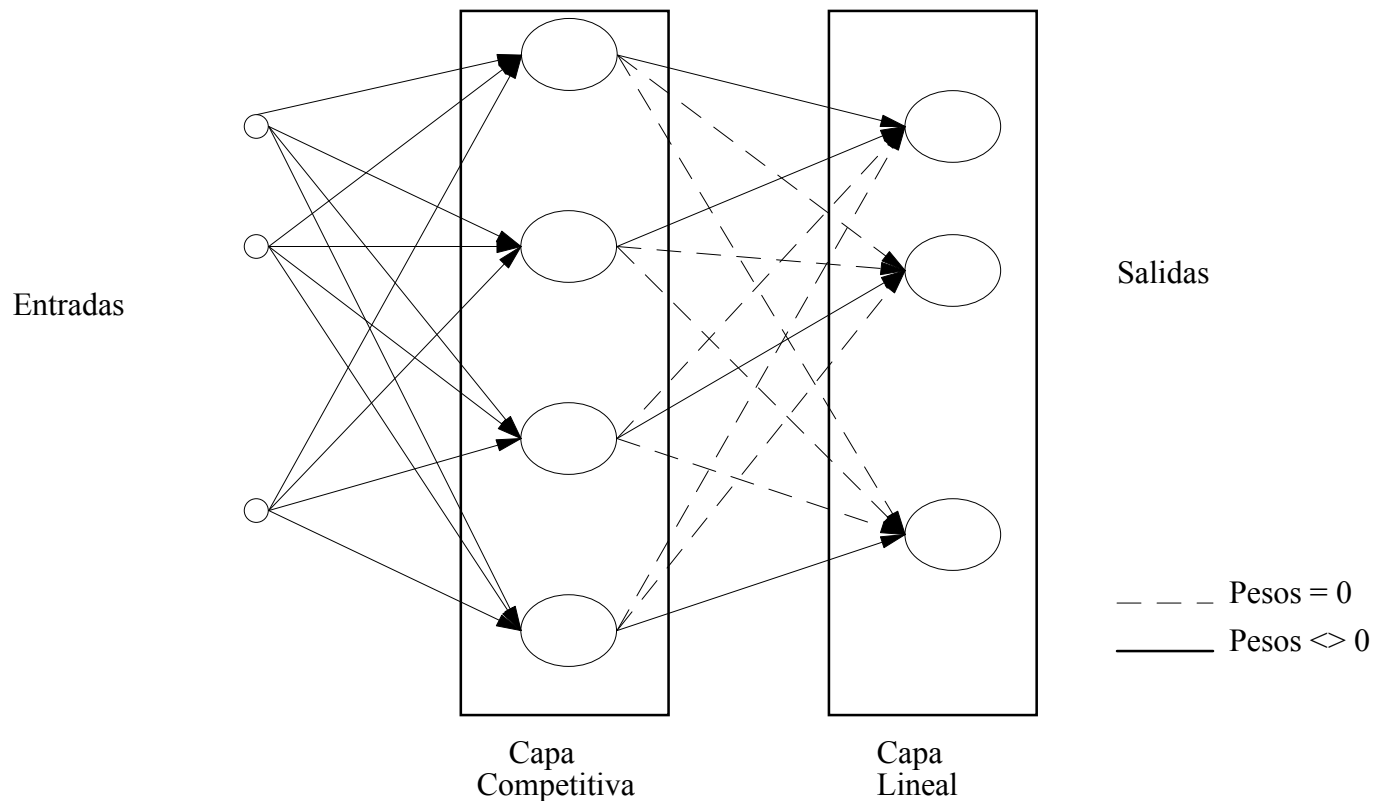
Partición de un espacio continuo en 10 valores discretos

- Cuando al sistema se le presenta un nuevo vector de entrada, primero se le asigna una determinada región y después es representado por el vector característico de dicha región.

Esta técnica es frecuentemente utilizada en transmisión de datos, utilizando una versión codificada del vector de representación en lugar de los patrones de entrada.

- Cuando la medida de similaridad que se utiliza para asignar a un patrón de entrada a una determinada región es la Distancia Euclídea, el quantizer es denominado de **Voronoi**. Este, divide el espacio de entradas en **Celdas de Voronoi**, y cada celda es representada por uno de los vectores de reconstrucción W_i . La i -ésima celda contiene aquellos puntos del espacio de entradas que están más cercanos al vector W_i que a cualquier otro W_j .
- La regla de aprendizaje competitivo basada en la distancia Euclídea puede ser utilizada para posicionar un conjunto de vectores de representación (Kohonen 1989).
 - Este algoritmo de aprendizaje competitivo puede ser visto como un método aproximado para computar dichos vectores de manera sin supervisar.

Kohonen, diseñó versiones supervisadas de este método, **Learning Vector Quantization (LVQ)**, para problemas de **clasificación adaptiva de patrones**. La información de clases se utiliza para afinar los vectores de representación: Mejora en la clasificación de regiones.



ESTRUCTURA QUE SIMULA LVQ

OPERACIÓN: LVQ es una técnica de aprendizaje supervisado. Un vector de entrada X es tomado aleatoriamente del espacio de entradas. Si las clases del vector X y la del W asociado coinciden, entonces el vector W es movido en la dirección de X . Por la otra parte, si las clases son diferentes, el vector W es alejado de X .

Las reglas del cambio de los pesos asociados a los PE que forman la región a la cual el Mapa Autoorganizativo da como ganadora son las siguientes:

$$\text{Si } \zeta_w = \zeta_x$$

$$w(t+1) = w(t) + \mu(t)(x - w(t))$$

$$\text{Si } \zeta_w \neq \zeta_x$$

$$w(t+1) = w(t) - \mu(t)(x - w(t))$$

Es deseable que el factor de ganancia $\mu(t)$ decrezca con el tiempo.

Los pesos de la capa lineal:

- Se inicializan a valores (1 y 0) antes del aprendizaje (Usual).
- Se modifican utilizando la Regla Delta.