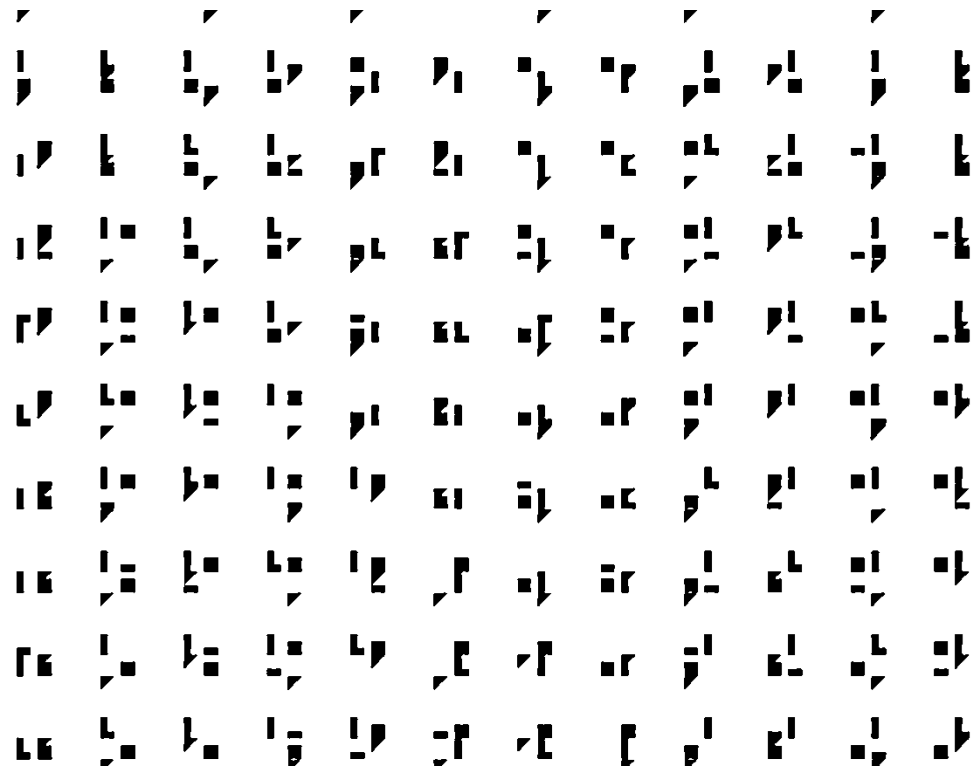


Flatwriter:

representación de una realidad cambiante

a través de la obra de Yona Friedman



*A mis abuelos,
J e I.*

RESUMEN

“Hacia una arquitectura científica” de Yona Friedman nace para intentar reconciliar al arquitecto con la sociedad a través de la democratización de la arquitectura y el no-paternalismo del arquitecto. Publicado en 1971, el texto recoge la propuesta del Flatwriter, además demuestra una fascinación por los ordenadores y el mundo de la lógica y las matemáticas, intentando recoger sus teorías en un método científico; un sentimiento que parece estar creciendo en la actualidad con el auge de la arquitectura paramétrica.

A través del análisis del Flatwriter de Yona Friedman se pretende una revisión de estas corrientes de pensamiento desarrolladas en los 70 con la intención de saber si estas teorías de empoderamiento del usuario, en su día consideradas utópicas, tienen cabida con los métodos gráficos y proyectuales que hoy día se encuentran a nuestra disposición.

Conociendo el potencial de las herramientas informáticas en el panorama actual se plantea el uso de Rhinoceros+Grasshopper 3D para la creación de un repertorio, el trabajo se apoya en la teoría de grafos y matrices de adyacencia desarrolladas por Friedman y en herramientas de Diseño Generativo, para que cualquier usuario pueda elegir libremente buscando además, nuevas perspectivas sobre el papel del arquitecto en la sociedad.

PALABRAS CLAVE: *Friedman; método científico; arquitectura paramétrica; Flatwriter; utopías.*

ABSTRACT

Yona Friedman's "Towards a scientific architecture" was born to try to reconcile the architect with society through the democratisation of architecture and the non-paternalism of the architect. Published in 1971, the text includes the Flatwriter's proposal and demonstrates a fascination with computers and the world of logic and mathematics, trying to gather in his theories a scientific method; a feeling that seems to be growing nowadays with the rise of parametric architecture.

Through the analysis of Yona Friedman's Flatwriter, a review of these currents of thought developed in the 1970s is attempted with the intention of finding out whether these theories of empowering the user, once considered utopian, have a place with the graphic and design methods that are available to us today.

Knowing the potential of computer tools in the current panorama, we propose the use of Rhinoceros+Grasshopper 3D for the creation of a repertoire, based on the theory of graphs and adjacency matrices developed by Friedman and using Generative Design tools, so that any user can "choose freely" and moreover seeking for new perspectives on the role of the architect in society.

KEY WORDS: *Friedman; scientific method; parametric architecture; Flatwriter; utopias.*

1. Introducción	
<i>Objetivos</i>	7
<i>Estado de la cuestión</i>	8
<i>Metodología</i>	11
2. Contexto	
<i>Crisis del papel del arquitecto</i>	13
<i>Teorías del GEAM</i>	15
3.El Flatwriter de utopía a realidad	
<i>Fundamentos del método de Yona Friedman</i>	19
<i>El Flatwriter: un ejemplo de puesta en práctica del repertorio</i>	23
<i>The YONA system: una realidad programable</i>	26
4. Estrategia	
<i>Secuencia para el desarrollo del sistema</i>	31
5. Desarrollo	
<i>Creación y representación del grafo</i>	35
<i>Obtención de matrices a partir del grafo</i>	38
<i>Puntos de destino: elección de la infraestructura</i>	40
<i>Creación de un repertorio completo</i>	42
<i>Elección y filtrado de combinatorias</i>	44
<i>Definición</i>	50
6. Discusión de resultados	
<i>Evaluación del sistema y posibles mejoras</i>	55
7. Conclusiones	
<i>Vigencia de las teorías de Friedman en la actualidad</i>	59
8. Referencias	
<i>Bibliografía</i>	63
<i>Créditos de la figura</i>	66
9. Anexo: definiciones	

"I am not a utopist, but a realist. My realism is not of the same breed as that of that of the mainstream architects. That is my main discovery."

Yona Friedman

1. INTRODUCCIÓN

1. Objetivos

1. Indagar sobre las iniciativas teóricas del último cuarto del siglo XX dirigidas a un nuevo modelo de urbanismo basado en la autodeterminación del habitante. En particular, sobre el trabajo teórico de Yona Friedman y el desarrollo práctico de su propuesta denominada el Flatwriter, así como en el método científico que se sigue para su desarrollo.
2. Proponer una revisión sobre el papel de los arquitectos en el diseño más allá de la customización en masa, interpretando a través del Flatwriter diversas teorías de participación ciudadana y emancipación del usuario desarrolladas a partir de los años 70.
3. Evaluar la validez del método científico de Friedman en la actualidad. Para ello, se usarán métodos de representación gráfica y programación visual actuales como Rhinoceros 3d y Grasshopper 3d.



Fig. 1.1.

Fig. 1.1. Yona Friedman nació en 1923 en Budapest (Hungria) y vive y trabaja en París (Francia). Se forma como arquitecto y salta a la fama con su manifiesto “*L’architecture Mobile*” y su idea de un enfoque diferente del crecimiento urbano con la *Ville Spatiale* de 1956. (www.yonafriedman.nl)

2. Estado de la cuestión

Las obras del arquitecto húngaro-francés Yona Friedman han sido ampliamente estudiadas y referenciadas por gran parte de la comunidad científica y arquitectónica. Durante el último cuarto del siglo XX sus dibujos y manifiestos fueron ampliamente publicados por la prensa especializada del momento y han inspirado a numerosos arquitectos como Schulze-Fielitz, Kenzo Tange, Kurokawa, Archigram, Safdie, Bofill, Mühlestein y muchos otros.

A pesar de esto, los fuertes conceptos teóricos desarrollados por Yona Friedman no tuvieron ese mismo grado de repercusión y fueron en parte ignorados u olvidados en estas aportaciones, por lo que la fuente más amplia de la que disponemos actualmente es la de los propios escritos de Yona Friedman. Para ello, nos basamos principalmente en su libro “*Hacia una arquitectura científica*” publicado en 1971 y en las teorías predecesoras desarrolladas también por Friedman y otros grupos de trabajo en los que se encontraba.

Sus primeras publicaciones y manifiestos teóricos como “*L’architecture mobile*” (1956), o los publicados por el GEAM “*Groupe d’étude d’architecture mobile*” (1958) y más adelante, el GIAP, “*Groupe International d’Architecture Prospective*” vienen recogidos en la antología de Conrads, el editor de la revista “*Bauwelt*” junto otros de los programas y manifiestos más importantes de la época. La recopilación de Conrads, “*Programs and manifestoes on 20th-century Architecture*”, recoge los manifiestos sobre arquitectura móvil, y acaba la publicación con un texto de 1962 “*The ten principle of the spacial city by Yona Friedman*”. En una perspectiva más actual encontramos dos libros clave en los que participó el propio Friedman y que constituyen una antología de sus textos y del desarrollo de maquetas, dibujos e ideas: “*Pro domo*” publicado por Actar Publishers en 2006 y “*Yona Friedman: The dilution of architecture*” publicado por Park Books en 2015.

Este último libro contiene además la tesis más reciente sobre la carrera e ideas de Friedman: “*The erratic universe of Yona Friedman*”, de Manuel Orazi, PhD de la escuela de estudios avanzados en Venecia (SSAV). El trabajo

de Orazi constituye un recorrido de los procesos, escritos y manifiestos del arquitecto húngaro-francés desde sus primeros años, hasta el desarrollo de sus teorías sobre sociedad y tecnología, también analizadas para este trabajo. Los escritos de Yona Friedman y la tesis de Orazi se publican conjuntamente en 2015 y se materializa antes de la publicación en una exposición de las más recientes y completas de su obra “*Genesis of a Vision*”, presentada en el EPFL (Lausanne, Suiza) en 2012.

Las preocupaciones sociales de Friedman se recogen también en otro de las publicaciones conjuntas de Orazi y Friedman: “*Architecture with the people, by the people, for the people*” (2011) editado por María Inés Rodríguez.

Las líneas de investigación de Manuel Orazi son de gran importancia, pues establecen un marco histórico y social con perspectiva histórica que nos permite comprender el momento en el que surgieron estas teorías, de igual manera que no olvida la marcada relación con las teorías de Friedman y la tecnología, donde se explica además su teoría de grafos y las relaciones de Friedman con el Instituto Tecnológico de Massachusetts (MIT).

Siguiendo con la investigación sobre la tecnología del momento y la teoría de grafos y programación, destacan diversos estudios como los de Theodora Vardouli para su Tesis “*Graphing Theory: New Mathematics, Design, and the Participatory Turn*” para el MIT. Estas líneas de investigación nos permiten un acercamiento a los experimentos desarrollados por el MIT durante el periodo de los 70-75. Este hecho nos permite estudiar el momento en el que las teorías y experimentos de participación y diseño asistido para los usuarios comenzaba a implementarse, permitiéndonos indagar más sobre estas intenciones y metodologías pioneras, sin toda la carga histórica que contienen actualmente las teorías sobre programación y basándonos en el contexto histórico del Flatwriter.

Los estudios de Vardouli se basan en la recopilación y estudio de experimentos como “*Architecture-by-yourself: An Experiment with Computer Graphics for House Design*” desarrollado por Weinzapfel y Negroponte, que desarrollaron este sistema con la participación de Yona Friedman. Además de otros estudios predecesores desarrollados por el Media Lab como: “*IMAGE: an interactive computer system for multi-constrained spatial synthesis*” uniendo intenciones y reacciones recogidas de estos experimentos en las publicaciones del MIT Press durante estos años, con especial cabida de “*From Soft Architecture Machines*”, también de Negroponte.

Los documentos originales de estos experimentos del MIT y otros recogidos en la bibliografía han sido consultados y estudiados para el desarrollo de este trabajo por el valor de ser pioneros en su campo dándonos una visión específica de los resultados de experimentos predecesores al planteado en este trabajo.

Centrándonos en el **estado técnico de la cuestión**, que también abordada este trabajo, nos basamos en distintos intentos de adaptar las ideas de Friedman a un contexto más actual, experimentos presentados en conferencias y otros medios de divulgación científica y tecnológica como “*Inspired by the Atmosphere. See the Invisible*”, “*Pervasive Collaboration and Tangible Complexity in Realtime Architecture*”. Aunque no hemos encontrado relación directa con una obra concreta o el método desarrollado por Friedman en “*Hacia una arquitectura científica*”, sí encontramos similitudes en la estética pretendida en las publicaciones. Aunque el concepto estético se aleja de lo que pretende este trabajo, las divulgaciones de IMAGINI? o ECADEE nos han llevado también a la investigación “*Designing with Space Syntax*” en la que los ingenieros holandeses Pirouz Nourian, Samaneh Rezvani y Sevil Sariyildiz desarrollan SYNTACTIC, un plugin para Grasshopper sobre la teoría de grafos que será de utilidad para el ejercicio planteado.

Tampoco se ha encontrado ninguna aproximación actual al Flatwriter pese a seguir un método fácilmente programable, aunque sí contamos con algunos ejemplos de intentos actuales de programación del urbanismo con matrices binarias, como los llevados a cabo por Reinhard König y Christian Bauriedel en 2004 en la Universidad Técnica de Kaiserslautern basadas en el algoritmo del Cellular Automata (CA), implementando nuevas reglas para la simulación urbana que presentan en su base teórica y matricial una similitud a este trabajo.

Siguiendo estas dos vertientes para la documentación, teórica y técnica, y dada la carencia que hay en las líneas de investigación de programación y emancipación del usuario con los medios informáticos actuales, con lo planteado en el método desarrollo en “*Hacia una arquitectura científica*”, este trabajo pretende sin olvidar todos los aportes previos a la cuestión a tratar, responder algunas de las preguntas surgidas de una realidad actual sobre el Flatwriter. Y con ello, la posibilidad de creación un sistema de elección libre e indeterminado que permita la participación ciudadana de una manera más activa y mejorar el proceso de información entre el arquitecto y el usuario a través del Diseño Generativo.

3. Metodología

De entrada este trabajo se plantea como una revisión con los métodos actuales de representación y programación sobre el método desarrollado por Yona Friedman en su libro *“Hacia una arquitectura científica”*.

La primera fase de este trabajo supone el análisis y la documentación de las teorías y manifiestos desarrollados por Friedman a través de las diversas bases expuestas en el estado de la cuestión, adentrándonos de esta manera en la creación del Flatwriter que llega a elaborar únicamente de manera teórica.

La simplicidad o limitación en la elección para la creación de este repertorio es la que permite que sea fácilmente extrapolable y programable, como se ha planteado en esta investigación. Por ello, la segunda fase consiste en la búsqueda y elección de posibles herramientas y métodos de expresión gráfica actuales para el desarrollo de un modelo simplificado del Flatwriter, elaborando una metodología que nos permita seguir los pasos explicados por Friedman en su libro. Finalmente nos decantamos por softwares de diseño CAD y BIM ampliamente utilizados por arquitectos en la actualidad, y cada vez más ligados a herramientas de programación como Dynamo para Revit Architecture o, Grasshopper 3D ® para Rhinoceros 3D.

En la tercera y última fase del trabajo, se elaborará un modelo simplificado del Flatwriter a través de herramientas gráficas de Diseño Paramétrico y Diseño Generativo en auge en el ámbito de la Expresión Gráfica con los que se propondrá la creación de un repertorio y un sistema de elección, abierto a la utilización de usuarios no expertos, a través de lo explicado por Friedman.

Por la extensión de este trabajo y el tiempo disponible, se postergará la optimización del sistema a investigaciones posteriores si los resultados de la actual son satisfactorios. Sí que se realizará una discusión y evaluación de los resultados obtenidos en esta investigación, así como otras consideraciones finales.

2. CONTEXTO

1.Crisis del papel del arquitecto

La profesión de arquitecto, desde la perspectiva social, ha estado en constante cambio desde el origen de la disciplina. Uno de esos cambios clave ocurre en el año 56, tras la celebración del último Congreso Internacional de Arquitectura Moderna (CIAM X) celebrado en Dubrovnik.

Las diversas crisis que acontecen a este congreso exigen una respuesta a los problemas sociales por parte de la arquitectura y los arquitectos, algo que, pese a las buenas intenciones promulgadas en la carta de Atenas (1943), aún estaba muy lejos de encontrar solución, especialmente tras las alarmantes críticas hacia los complejos edificatorios proyectados en la época.

La polarización entre el Movimiento moderno y la sociedad se produce casi de manera inevitable, y a medida que aumenta el descontento social, jóvenes arquitectos ajenos a la primera generación fundadora del CIAM, pretenden aportar nuevas teorías en Dubrovnik, alejándose así de las bases establecidas por sus predecesores. Entre ellos Yona Friedman, expone su posición personal en Dubrovnik, basándose en su tesis para la Universidad de Haifa (Israel):

LA REVOLUCIÓN DE LOS ASENTAMIENTOS.

La determinación del usuario hace que los asentamientos funcionen.

La determinación del usuario es mejor que la planificación gubernamental

La determinación del usuario resuelve problemas que los gobiernos no pueden resolver.

(Orazi M, Friedman Y., 2016,379)

Estos eslóganes expuestos en Dubrovnik son una propuesta de reforma integral del ejercicio de la arquitectura, que pone el foco en el usuario y el papel de la autodeterminación. Un año más tarde, en 1957, Friedman empieza a trabajar sobre estas bases en una teoría, que según sus palabras, permita emancipar al habitante del avasallamiento del arquitecto.

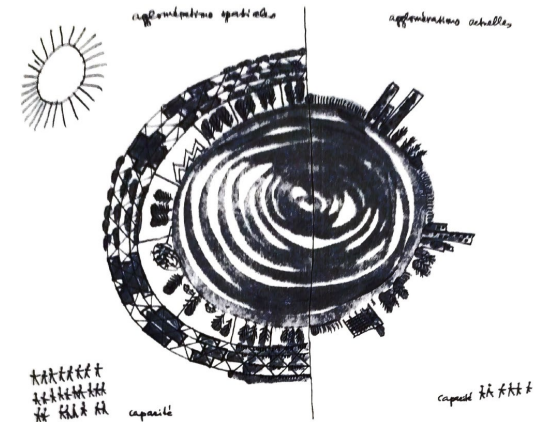


Fig. 2.1.

Fig. 2.1. Yona Friedman: ratio de capacidad respecto la ciudad espacial y la ciudad actual, 1958



Fig. 2.2

1. **Urbanismo móvil:** término desarrollado a través de arquitectura móvil, definida en 1957 por Friedman en "*L'architecture mobile*" (1956) " como cualquier solución que permitiera a los usuarios tomar una decisión directa y transformar ellos mismos directamente su entorno en el momento que decidan revisar su decisión anterior.

2. **Urbanismo espacial:** término definido en 1962 por Yona Friedman en "*The ten principles of Space town planning*" (Conrads, 1971, 1983) en el que Friedman enuncia los principios para la planificación de un nuevo tipo de ciudad que permita la construcción de barrios tanto yuxtapuestos como superpuestos.

3. **Infraestructura:** término acuñado por Friedman que hace referencia a marcos estructurales vacíos que pueden ser utilizados de diversas maneras y que contiene en su interior todas las instalaciones necesarias para que fuera habitable y garantizará un sistema indeterminado de libre elección dentro de la misma

Fig. 2.2. Pictograma de Yona Friedman publicado en el artículo "*Soft Machines*" de Nicholas Negroponte.

Junto a él, otros arquitectos de la época sintieron este mismo distanciamiento de su profesión con la sociedad e intentaron adoptar una actitud científica hacia el diseño, intentando buscar así una solución a los numerosos problemas sociales a los que la arquitectura del momento no conseguía dar respuesta. Simultáneamente, empieza a crecer la fascinación por el mundo de la informática, los análisis matemáticos y los intentos de establecer unas nuevas bases del diseño buscando una arquitectura menos arbitraria.

Para Friedman el problema es que la profesión del arquitecto responde a estándares anticuados. Esta manera de entender la arquitectura era efectiva cuando el arquitecto disponía de tiempo suficiente para conocer los gustos y necesidades de su cliente y podía actuar en consecuencia a estos, como traductor de ideas, materializando las propias decisiones de sus clientes. En este nuevo contexto social, el arquitecto no es capaz de conocer las necesidades de cada uno de sus clientes (por la urgencia de la edificación, el elevado número de habitantes para los que se construye, etc.). Por este motivo nace la necesidad, para Friedman, de elaborar un método científico, pues la misión del arquitecto debería ser facilitar al usuario que elija, libre y democráticamente, las decisiones que tengan que ver con su modo de vida.

La función del nuevo arquitecto será pues, la construcción objetiva de un repertorio, unas advertencias y una infraestructura como explica en su libro "*Hacia una arquitectura científica*", tratando de evitar así el paternalismo del arquitecto.

Para dotar de mayor validez y autonomía a estas ideas, será necesario primero el desarrollo de nuevos términos que apoyen sus teorías tales como '**urbanismo móvil**'¹, '**urbanismo espacial**'²; y nuevas técnicas como la '**infraestructura espacial**'³, que empezará a desarrollar conceptualmente con el GEAM en 1958, buscando siempre un acercamiento democrático a la arquitectura. Friedman pretende redefinir el concepto de arquitecto y habitante, alejándose del concepto de "hombre promedio" impuesto por el Movimiento moderno y las estadísticas para lograr una "*arquitectura con la gente, por la gente y para la gente*". (Friedman, 2011, 15)

2. Teorías del GEAM

Tras la división producida en el CIAM de 1956, Friedman comienza a publicar copias de su manifiesto “*L’architecture mobile*”, donde ensayaba sus primeras ideas sobre arquitectura móvil. Asimismo, realiza numerosos viajes y conferencias por Europa para divulgar sus ideas y buscar apoyos.

En el prefacio de la segunda publicación de “*L’architecture mobile*”, anuncia la creación del “*Groupe d’étude d’architecture mobile*” (GEAM, 1958-1962) junto a Frei Otto, Gunther Günsel y otros arquitectos e ingenieros de prestigio.

En 1960, el grupo formalizó sus lemas en un manifiesto titulado “*Programme for a mobile architecture*”. Donde se exponían las dificultades creadas por el urbanismo del movimiento Moderno y lo que suponía directa e indirectamente para los habitantes de las ciudades. Entre los principales problemas identificados se encontraban una rigidez excesiva de la edificación, un crecimiento de la población imposible de planificar y el elevado precio de la vivienda.

Cada uno de los integrantes del GEAM debía realizar una propuesta individual indagando en el parámetro de la movilidad. La propuesta de Friedman para esta serie de propuestas fue La Ville Spatiale⁴, donde proponía un ‘urbanismo móvil’, que permitía la construcción de grandes unidades con el más alto grado de flexibilidad, sin limitar las libertades individuales y lo más importante, que fueran capaces de asumir variaciones importantes con rapidez y sencillez técnica.

“Proponemos un urbanismo móvil que busque técnicas que permitan la construcción de grandes unidades dentro de la infinita flexibilidad que se requiere; técnicas que permitan la provisión de suministros (agua, energía, eliminación de aguas residuales) capaces de una rápida alteración y reutilización, técnicas que utilicen elementos baratos, sencillos de montar, fáciles de transportar, reutilizables”. (Orazi M, Friedman Y., 2016, 54)

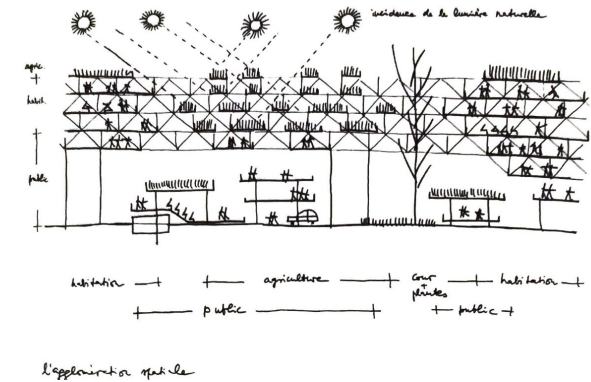


Fig. 2.3

4. Ville Spatiale: Una Ville Spatiale es un esqueleto estructural multicapa (rejilla) sobre pilotes que puede ajustarse de forma flexible cuando se desee. La estructura se apoya en columnas situadas a un intervalo de 40-60 metros y que albergan los accesos y las redes de instalaciones. La base de la retícula es un módulo de 6x6 metros que puede albergar todo tipo de funciones. (<http://www.yonafriedman.nl/>)

Fig. 2.3 Conferencia en Ámsterdam del GEAM, mayo de 1962.

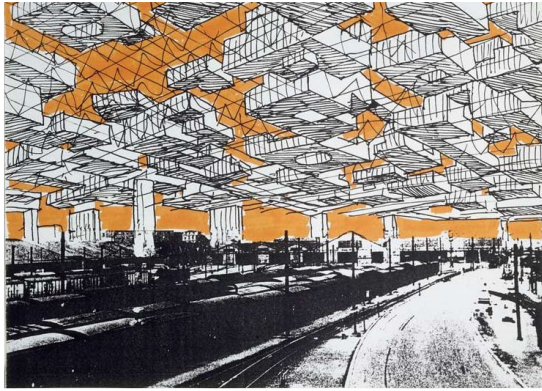


Fig. 2.4

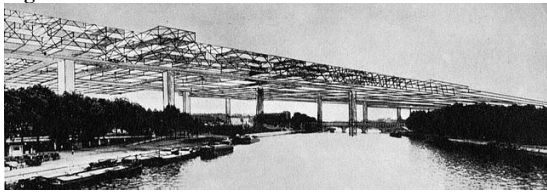


Fig. 2.5

5. Hardware. Friedman define con este término la infraestructura material: “Podemos establecer la manera de establecer cambios si decimos; todo lo que está dentro de la infraestructura (y por tanto todo el proceso de abrir o colocar espacios) debe ser reversible, el hardware debe ser cambiabile-solo la infraestructura puede ser rígida y colocada en el sitio” (Friedman, 1971, 66)

1. Hardware: conjunto de aparatos de una computadora (<https://dle.rae.es/>)

6. Software: se refiere a las partes abstractas que no se pueden contabilizar, la infraestructura inmaterial del proceso o la puesta en práctica de un sistema.

1. Software: Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora. (<https://dle.rae.es/>)

Fig. 2.4 Fotomontaje de Yona Friedman, la *Ville Spatiale* sobre la terminal de Paris, 1959-65

Fig. 2.5 Fotomontaje de Yona Friedman, infraestructura espacial sobre el Sena con el ponte de Bercy, Paris.

Friedman buscaba esta flexibilidad a través de lo que define como ‘**urbanismo espacial**’ que “permite la agrupación de barrios tanto yuxtapuestos como superpuestos” (Conrads, 1971, 183), adentrándose tras la disolución del GEAM en conceptos y terminologías correspondientes al nuevo tipo de urbanismo propuesto:

“Podemos establecer la manera de establecer cambios si decimos; todo lo que está dentro de la infraestructura (y por tanto todo el proceso de abrir o colocar espacios) debe ser reversible, el hardware debe ser cambiabile-solo la infraestructura puede ser rígida y colocada en el sitio” (Friedman, 1971, 66)

Como vemos, estás ideas sobre flexibilidad y arquitectura móvil, son las que llevan a Friedman al desarrollo de lo que él define como un método científico. Sus teorías proclaman a través de la lógica el empoderamiento del usuario frente al sistema del movimiento Moderno.

En el siguiente capítulo definiremos conceptos como ‘**hardware**’⁵, ‘**software**’⁶ o ‘**infraestructura**’, contextualizados dentro del método científico de Friedman, lo que finalmente nos permitirá la elaboración de un modelo simplificado siguiendo estos principios, sin olvidar el objetivo del método: permitir a los propios usuarios la elección dentro de un sistema indeterminado sin la supervisión paternalista del arquitecto.

3. EL FLATWRITER

1. Fundamentos del método de Yona Friedman

En su búsqueda de un nuevo rol para el arquitecto, Friedman desarrolla un método científico que explica en su libro *“Hacia una arquitectura científica”* (1971), un sistema objetivo que se desarrolla de forma secuencial describiendo las fases para la construcción de un repertorio que permitirá al futuro usuario diseñar y ubicar su apartamento dentro de una infraestructura ya dada. El método se apoya en la lógica y las matemáticas para enunciar primero 3 ‘**axiomas**’⁷ en un campo de validez fijado previamente: *todo espacio habitado por un ser humano*. Así se definen las actividades del arquitecto y del urbanista en 3 puntos:

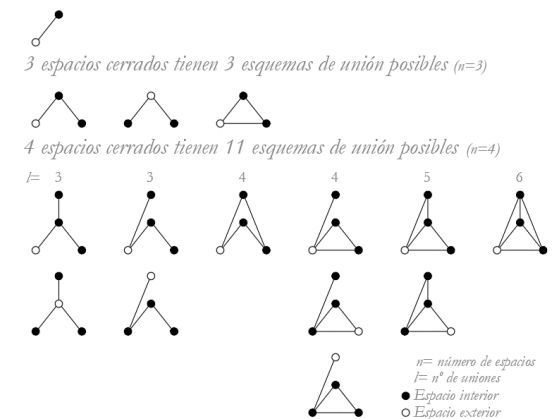
- 1- Producción de un espacio cerrado en el espacio preexistente.
- 2- De cada espacio cerrado hay al menos un camino hacia los otros espacios cerrados.
- 3- Hay al menos dos diferentes categorías de espacios cerrados. (Friedman, 1971, 44)

Una vez establecida esta axiomática se obtienen las 3 reglas del ‘**mapeado**’⁸ (mapping), (el código que hace posible transcribir cada operación o elemento en un sistema de representación), convirtiendo así los axiomas en una representación de las elecciones de los usuarios:

- 1- Un \bullet representa un espacio cerrado
- 2- Una — representa un acceso
- 3- Una etiqueta representa una diferencia. (Friedman, 1971, 44) **Fig. 3.1**

El ‘**mapeado**’ consistirá según estas reglas en puntos (*espacios cerrados*), conectados por líneas representando un acceso en los que ningún punto quedará desconectado, esta representación recibe el nombre de ‘**grafo**’⁹ (graph). Las ‘**etiquetas**’¹⁰ definen especialidades o diferencias, en este caso dos: *espacio abierto* y *espacio cerrado* (aunque podrían existir más etiquetas con otros distintivos); y el número de espacios cerrados se representa

Las reglas del mapping permiten construir las listas combinatorias completas de todas las soluciones posibles (sin omisión)
2 espacios cerrados solo tienen una unión posible ($n=2$)



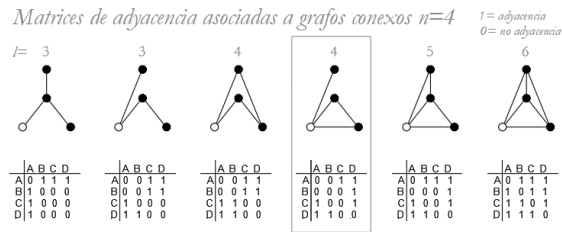
Solución: 11 combinatorias posibles para un grafo $n=4$ con 3 espacios interiores y 1 espacio exterior

Fig. 3.1

7. **Axioma:** 1. Proposición tan clara y evidente que se admite sin demostración. 2. Cada uno de los principios indemostrables sobre los que, por medio de un razonamiento deductivo, se construye una teoría. (<https://dle.rae.es/>)

8. **Mapeado:** 1. Localizar y representar gráficamente la distribución relativa de las partes de un todo; como los genes en los cromosomas. 3. Trasladar a un mapa sistemas o estructuras conceptuales. (<https://dle.rae.es/>)

Fig. 3.1 Esquema de las reglas de mapeado (mapping).



Distintas formas de representar el grafo escogido

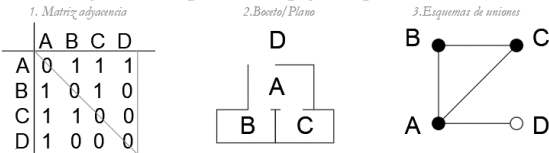


Fig. 3.2

9. **Grafo:** 2. Diagrama que representa mediante puntos y líneas las relaciones entre pares de elementos y que se usa para resolver problemas lógicos, topológicos y de cálculo combinatorio. (<https://dle.rae.es/>)

10. **Etiqueta:** representa una especialización, una diferencia “la etiqueta en su capacidad de signo de diferenciación, puede indicar categorías totalmente heterogéneas (...) en las cuales un arquitecto está llamado a trabajar” (Friedman, 1971, p.50)

11. **Puntos de cruce.** Punto o lugar de paso de un lado a otro. (<https://www.lexico.com/>)

12. **Topología:** trata las propiedades de las figuras atendiendo a la adyacencia (continuidad) de cada uno de los espacios relacionados, con independencia de su posición en el grafo, tamaño o forma. Dos figuras son topológicamente distintas si una no se puede convertir en otra sin distorsión de sus uniones. (como si se tratase de una estructura tridimensional)

Fig. 3.2. Matrices de adyacencia generadas para cada grafo con esquemas de uniones y sus representaciones, redibujado de “Hacia una arquitectura científica”

por un valor n , de manera que $n=1,2,3,4,\dots$, aunque a medida que aumenta el número de espacios cerrados, lo hace también la complejidad del ‘**grafo**’, aumentando así las posibles relaciones entre espacios, y a partir de un valor $n=4$, no podemos asegurar que el ‘**grafo**’ sea plano.

Un ‘**grafo**’ no plano, es un grafo complejo, en el que no se puede asegurar que se produzcan ‘**puntos de cruce**’¹¹ (crossover) entre las líneas que representan los accesos. A partir de $n>4$, no podemos asegurar que el ‘**grafo**’ sea plano pues podría darse un cruce entre los accesos que acabaría materializándose en dos espacios cerrados que no tuvieran una relación directa, algo que debería resolverse en la práctica a través de un pasillo o una escalera y que no está valorado en las reglas establecidas previamente.

Para simplificar el método en un primer momento Friedman establece en su teoría de grafos estas reglas para grafos con $n<5$, es decir grafos planos para la construcción de las viviendas. Más adelante en el capítulo 4 se desarrolla: “Los grafos planos completos con $n\geq 7$: ‘cruces’ y ‘conflictos’ potenciales” (Friedman, 1971, 77) donde Friedman propone cómo solucionar estos puntos conflictivos. Además, establece la relación por la que se indicaría el número máximo de uniones que puede soportar un grafo plano a través de la expresión: $L=3(n-2)$; donde L representa el número máximo de uniones, por lo que en un ‘**grafo**’ $n=4$ el número máximo de uniones posibles es de 6, esta misma relación nos demuestra que no es posible en un ‘**grafo**’ $n=5$ que todas las habitaciones estén relacionadas directamente entre sí, pues esta operación requería 10 uniones siendo $L=9$ para $n=5$. Fig. 3.3

Resumiendo, el ‘**mapeado**’ de esta axiomática por grafos conexos y etiquetados permitirá definir una lista combinatoria con todas las uniones posibles para $n<5$, además estas relaciones entre espacios podrán ser representadas por matrices de adyacencia (matrices binarias donde 1 representará conexión directa entre los espacios cerrados y 0 no conexión), lo que permitiría la utilización directa del ordenador, algo significativo para el desarrollo de este trabajo. Fig. 3.2

Tenemos fijados por tanto, un máximo de 4 espacios y 2 etiquetas, lo que da un resultado de 11 grafos, ‘**topológicamente distintos**’¹² y 11 matrices que serían la representación matemática de cada uno de los grafos, lo que constituye nuestro ‘**repertorio**’¹³ como una lista completa y exhaustiva de todas las relaciones posibles.

El rigor del método de Friedman se va poniendo de manifiesto durante todo el desarrollo del método de construcción del ‘**repertorio**’ que contiene todas las posibilidades de elección. De esa manera, el usuario será consciente de lo que implica cada una de sus elecciones gracias a lo intuitivo del proceso y, a una serie de advertencias que se irán desarrollando a partir de este punto.

La primera de esta serie de advertencia es la ‘**matriz de caminos**’¹⁴, generada a partir de la matriz primitiva de adyacencia con sencillas operaciones aritméticas y que a su vez generará la ‘**matriz de esfuerzos**’¹⁵, permitiendo al futuro usuario conocer las posiciones ventajosas de cada uno de los espacios cerrados y el esfuerzo que le supondría según sus hábitos, para finalmente calcular el ‘**valor de esfuerzo local**’¹⁶ en una matriz a modo de sistema de advertencia. Consciente de la repercusión de sus decisiones iniciales, el usuario tiene la posibilidad de retroceder a pasos previos y corregir sus decisiones iniciales. **Fig. 3.4**

Siguiendo este mismo sistema de advertencia crea un ‘**diagrama de esfuerzos**’¹⁷ a nivel global de la ciudad, que se genera uniendo los valores resultantes en ‘líneas de isoesfuerzos’ y constituyendo el ‘**sistema de advertencia**’¹⁸ a nivel urbanístico, conociéndose así las repercusiones causadas por las decisiones de cada usuario para los demás habitantes de la Ville Spatiale. **Fig. 3.5**

Es necesario recalcar en este punto que todas las fórmulas y matrices presentadas en este trabajo son de gran simplicidad, permitiendo su cálculo y representación sin la necesidad de un ordenador, pero elaboradas dentro de un sistema que no deja nada a la intuición, con una lógica y precisión matemática que hacen de este sistema idóneo para su representación con medios informáticos como demostraremos más adelante elaborándolas a través del método explicado en nuestro caso práctico.

Una vez resuelto como se construye el ‘**repertorio**’ debemos abordar el problema del ‘**hardware**’ (infraestructura material), capaz de admitir “cualquier organización de espacios cerrados correspondientes a una elección particular, sea cual fuese esta” (Friedman, 1971, 59) que se planteará siguiendo la lógica desarrollada hasta este punto, a modo de resumen:

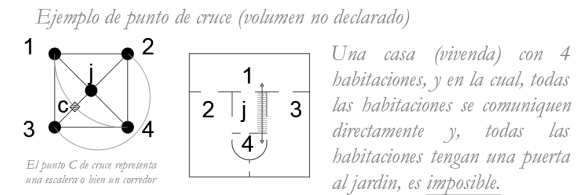


Fig. 3.3

Distintas formas de representar el grafo escogido

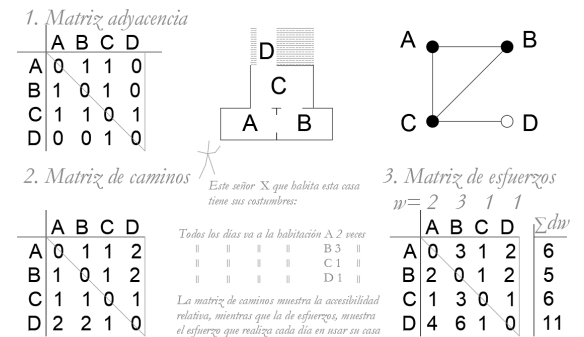


Fig. 3.4

13. **Repertorio**: “lista completa de los posibles esquemas de uniones, o conjuntos, (...) definidos por n-grafos planos conexos y etiquetados” (Friedman, 1971, 50)

14. **Matriz de caminos**: muestra la accesibilidad relativa entre los distintos espacios de un grafo suponiendo que la longitud entre los espacios es la misma. Cuando las distancias son consecutivas, se representará con una unidad, si para acceder a ese espacio hay que atravesar otro, con un dos, y así sucesivamente.

15. **Matriz de esfuerzos**: muestra el esfuerzo relativo a la accesibilidad y utilización de cada espacio. se obtiene al multiplicar la distancia d (representada en la matriz de caminos) por la frecuencia de utilización w (elegida por el usuario)

16. **Valor de esfuerzo local**: mide el grado de privilegio de situación de un espacio cerrado perteneciente al interior de un esquema de conjunto utilizado de una manera particular y concreta. Se obtiene del sumatorio los términos de la fila de matriz de la matriz de esfuerzos ($d * w$)

Fig. 3.3. Matriz de adyacencia, de caminos y de esfuerzos.

Fig. 3.4. Representación de un crossover.

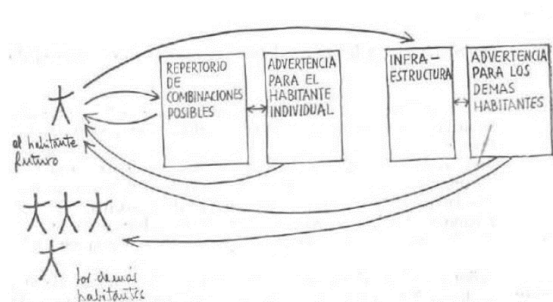
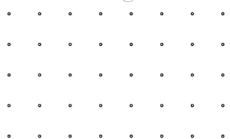


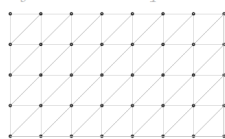
Fig. 3.5.

Tipos de infraestructura

Infraestructura troglodita



Infraestructura esqueleto



Matrices de adyacencia asociada a cada infraestructura

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

0	1	0	0	1	1	0	0	1	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	1	1	0	0	0	0
1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	1	1	0	1	0	1	0	1	0	0	0	0	1	0	0
0	0	1	1	0	1	0	1	0	0	0	0	0	0	1	1

Fig. 3.6.

17. **Diagrama de esfuerzos:** advierte del nivel de eficiencia del conjunto a nivel urbano. Se obtiene de los valores de las 'líneas de isoesfuerzos'.

18. **Sistema de advertencias:** permite informar al futuro usuario y al resto de habitantes de la Ville Spatiale sobre las consecuencias implícitas que puede acarrear la toma de ciertas tanto a su modo personal de utilización, como para el resto de habitantes

Fig. 3.5. Proceso de información y advertencias del Flatwriter

Fig. 3.6. Tipos de infraestructura y sus matrices.

- 1) Tomar un conjunto de n puntos
- 2) Hacer la lista combinatoria de todas las uniones planas, transformando estos n puntos en una figura única y conexa.
- 3) Construir sobre cada una de estas uniones la lista combinatoria de las m etiquetas posibles (m =número de etiquetas) (Friedman, 1971, 59)

Por tanto, si consideramos el **'hardware'** como un conjunto de puntos (espacios cerrados) conectados a través de líneas (conexiones entre los espacios cerrados), se puede representar este sistema con las reglas enunciadas con anterioridad. De hecho, siguiendo la lógica podemos optar a dos opciones para su construcción; como un **'grafo'** no relacionado (puntos sin relaciones), o como un **'grafo'** completo (con los puntos unidos al máximo, grafo saturado), denominándose para Friedman, **'infraestructura'** *troglodita*, o *esqueleto* respectivamente. Fig. 3.6

"La infraestructura es el único elemento material fijo de la ciudad: todos los demás elementos del 'hardware' (elementos 'injertados' en la infraestructura, techos, suelos, muros, cerramientos, equipamientos, etc.) así como los elementos no materiales (movimientos, climas, aspectos visuales, etc.), son intercambiables, transformables, desplazables: son en efecto "móviles" y pueden adaptarse a cualquier modo de uso en particular". (Friedman, 1971, 133)

Friedman define por tanto el **'hardware'** como toda la infraestructura material, partes de *arquitectura móvil* incluidas, y la **'infraestructura'** únicamente como la *parte fija* incluyendo las instalaciones necesarias en ella. El diseño de la **'infraestructura'** es otra de las tareas en el nuevo rol del arquitecto, pues la neutralidad de esta garantizará una ciudad indeterminada, en lo que Friedman y el grupo GEAM vienen definiendo como **'urbanismo móvil'** desde 1957.

El **'software'**, o infraestructura inmaterial del proceso, es la puesta en práctica del **'repertorio'**: el Flatwriter. Este se diseña siguiendo los pasos enunciados, y permite a los usuarios diseñar su propio apartamento en la Ville Spatiale, siendo capaz de acoger todas las elecciones y futuros cambios que ellos mismos escojan, con libertad de elección sin la supervisión paternalista del arquitecto, tratando de democratizar así la arquitectura para cada uno de los futuros usuarios y el resto de habitantes de la ciudad.

2. El Flatwriter: un ejemplo de puesta en práctica del repertorio

Tras explicar la construcción del **‘repertorio’**, Friedman presenta en el capítulo 3 el *“Flatwriter, un ejemplo de puesta en práctica del repertorio”* (Friedman, 1971,68). Friedman planeaba presentar este **‘software’** en la Feria Mundial de Osaka de 1970, algo que, por diversos motivos no llegó suceder. No sería hasta 1971 cuando se publica su teoría ya desarrollada, primero en la revista especializada *“Progressive Architecture”* (marzo 1971) bajo el título *“The Flatwriter: choice by computer”* y más tarde en su libro *“Hacia una arquitectura científica”* (1971) desarrollando una teoría completa para la creación del **‘software’** analizada en el punto anterior del presente trabajo.

El propio Friedman describe sus intenciones para la muestra de Osaka en una entrevista concedida en 1969:

“Yo llamo a esto una máquina de elegir. En la práctica, decimos que hay un número increíble de posibilidades para los apartamentos. Se pueden hacer millones y millones. Y si hago todo un repertorio, lo llamaré el menú de esta máquina, pues se muestran todas las posibilidades, de las que cualquiera puede elegir cualquier cosa, la que quiera. Habría sido un libro enorme, sólo lleno con pequeños planos. En lugar de eso, hago algo parecido a una máquina de escribir. Es decir, que, en lugar del libro, hice la herramienta que imprime el libro” (ORTE, 1969)

Como explica Friedman, la propuesta para Osaka era la construcción de una máquina¹⁹ con 53 teclas que mostrarían las distintas configuraciones posibles, así como las formas y la disposición del mobiliario. Con estas 53 teclas se podrían elegir, según Friedman, de 3 a 5 millones de apartamentos diferentes, de manera que cada asistente pudiera imprimir su propio apartamento fruto exclusivamente de sus deseos y elecciones. Después de elegir la configuración de su apartamento, los usuarios irían a una segunda máquina en la que elegirían la ubicación de este dentro de la ciudad espacial, comprobando al mismo tiempo, que cumple en todo momento con las condiciones básicas que le debe a sus vecinos (acceso, luz, ventilación, etc.), y

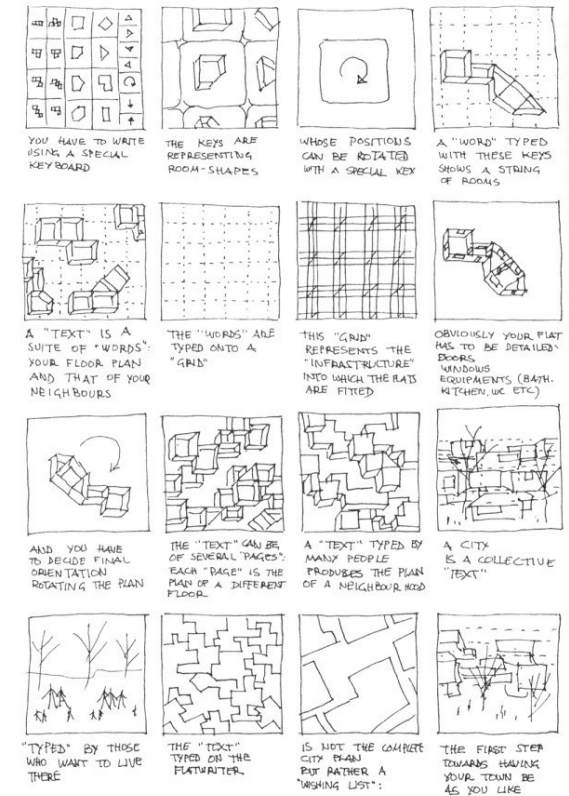


Fig.3.7

19. Según el propio Friedman: “una adaptación de la máquina de escribir IBM Selectric, (...), la que lleva los caracteres en relieve en una bola que parece una pelota de golf...” (Friedman en Anon., 2006).

Fig.3.7 Pasos seguidos para el diseño de una vivienda y su inserción en la Ville Spatiale a través del Flatwriter

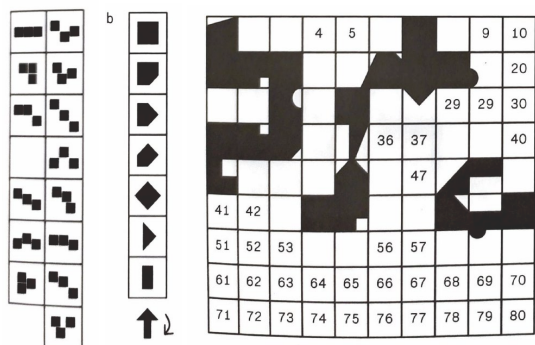


Fig.3.8.

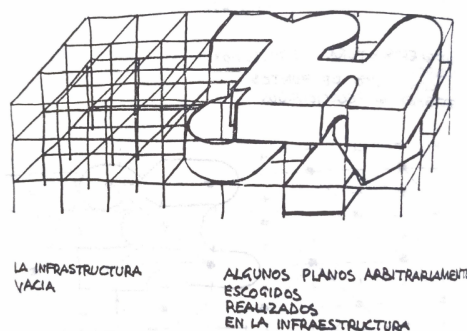


Fig.3.9.

Fig.3.8. Algunas de las 53 de las que constaría la máquina del Flatwriter junto a la imagen de la planta dónde se pretende insertar el apartamento.

Fig.3.9. Infraestructura esqueleto de una Ville Spatiale con algunos planos arbitrarios en la infraestructura a modo de ejemplo inserción del apartamento en la Ville Spatiale.

mostrando el resultado de la ciudad espacial tras esta inserción en la pantalla para conformidad del resto de habitantes.

En otras palabras, esta máquina ideada por Friedman contenía un **‘reportorio’** de varios millones de apartamentos y sabía elaborar instrucciones o avisos sobre las consecuencias que implicaban las elecciones de cada usuario, tanto para él mismo como para el resto de los habitantes de la ciudad espacial siguiendo el **‘sistema de advertencia’** explicado con anterioridad. El proceso implicaba simultáneamente el uso de una **‘infraestructura’**: “cualquier esqueleto estructural vacío de varias plantas” (Pro domo, 2006, 130), diseñada permitiendo la implementación de cualquier apartamento y su modificación o corrección, dividiéndose en dos pasos principales, diseño del apartamento y su inserción en la Ville Spatiale.

Aunque la máquina nunca se llegó a construir, Friedman publica un año más tarde, en 1971, los pasos necesarios para la elaboración del **‘software’**. Para el usuario, el procedimiento se divide en 8 etapas: de la etapa 1 a la 5 se planifica el diseño del apartamento y de la 6 a la 8 la elección de la ubicación. Friedman ejemplifica el proceso a través del Sr. Smith, un futuro usuario de la Ville Spatiale que utiliza el Flatwriter para diseñar su nuevo apartamento de 3 estancias. Éste comienza escogiendo las formas y orientaciones de los espacios que desea incluir así como sus accesos y otras características diferenciales, **‘etiquetas’**, para que en la segunda etapa la máquina sin contradicción ni crítica imprima las preferencias del usuario en un plano y en la tercera, el precio de la vivienda escogida

Continuando con la cuarta etapa, el Sr. Smith llega a otro teclado: el teclado de frecuencias, donde el Flatwriter le pide que ingrese la frecuencia con la visita cada habitación (número de veces al día) para calcular en la quinta etapa el esfuerzo que esto supondría para el Sr. Smith por si desea cambiar la disposición de las habitaciones. Lo que en el método se explica como **‘matriz de caminos’** y **‘matriz de esfuerzos’**

En la sexta etapa el Flatwriter reproducirá sobre una pantalla la **‘infraestructura’** vacía con cada uno de los vacíos numerado y los contornos de los apartamentos ya existentes de la Ville Spatiale, de esta forma el Sr. Smith puede ubicar su apartamento eligiendo el número de posición en un teclado.

El Flatwriter calcula en la séptima etapa los efectos de esta elección para sus vecinos, y si las elecciones del Sr Smith no son perjudiciales para el resto proyectaría la nueva silueta de la planta con la elección del Sr. Smith (en el caso de que lo fuesen, le pediría que modificara su elección).

La octava y última etapa es el cálculo del efecto global, lo que definimos anteriormente como **‘diagrama de esfuerzos’** de toda la **‘infraestructura’**, que permitirá a los nuevos usuarios de la Ville Spatiale estar informados sobre los cambios que implica la llegada de un nuevo habitante (el ruido, la calma, el valor comercial, la accesibilidad, etc.) contenidas todas ellas en el parámetro de esfuerzos.

Friedman anuncia que el Flatwriter es “la aplicación de un nuevo proceso de información entre el usuario y el objeto, permite una decisión individual casi ilimitada, así como posibilidad directa de corregir sus propios errores, sin la intervención de profesionales intermediarios” (Friedman, 1971, 76). A pesar del entusiasmo con la idea, nunca llegó a realizarse de la manera prevista para la feria de Osaka, pero Friedman nunca perdió el contacto con ella, diseñando numerosos sistemas de participación a lo largo de los años y situando al usuario en el foco de las corrientes arquitectónicas otra vez.



Fig.3.10.

Fig.3.9 Dibujo de portada de la primera publicación del Flatwriter en “*Progressive Architecture*”, en marzo de 1971

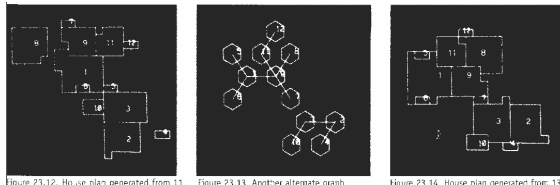


Fig.3.11.



Fig.3.12.

Fig.3.11. Primeros estudios de Negroponte sobre diseño participativo en arquitectura.

Fig.3.12. Pictograma de Yona Friedman, “no plan” sobre la necesidad de improvisación en la arquitectura.

3. Una realidad programable: The YONA system

Por las intenciones de este trabajo es importante entender que, para Friedman, como ha recalcado en diversas ocasiones “la misma operación realizada por el Flatwriter puede realizarse, con sellos, piezas en un tablero, o con un ordenador, pues, la idea se concibe como una herramienta de asistencia para diseños participativos independientemente del soporte tecnológico utilizado” (Anon., 2006). Sin embargo, el método de Friedman cuenta con rigor científico y matemático y, además, plantea un sistema de matricial, por lo que no cabe duda de que la intención era, al menos en un primer momento, permitir su posible adaptación informática. Esta adaptación encajaría con sus teorías sobre el nuevo papel del arquitecto y la democratización de la arquitectura, pues convertiría a los ordenadores en mediadores entre el usuario y el objeto deseado: su propia vivienda.

Como apunta Theodora Vardouli en su tesis, Friedman contribuyó a la sistematización de la filosofía del empoderamiento del usuario, convirtiendo estos conceptos en teorías y métodos, y una vez se sentarán las bases de determinados avances informáticos, estos podían convertirse en una herramienta que facilitara la participación y el empoderamiento de usuarios no expertos.

Un intento de esta democratización por la que abogaba Friedman la encontramos en “*Architecture-by-yourself*”, un experimento llevado a cabo por Nicholas Negroponte y Guy Winzafel (MIT) y el propio Yona Friedman. Donde los dos investigadores del MIT y Friedman proponían una aplicación sobre gráficos de ordenador titulada “*Machine Recognition and Inference Making in Computer Aids to Design*”. En el experimento desarrollan “*The YONA system*”, donde los principios de autodeterminación del usuario sin una guía paternalista del arquitecto se llevan a cabo aplicando las teorías de ‘grafos’ y las reglas del ‘mapeado’ de Yona Friedman.

“La investigación se enfrenta al delicado equilibrio entre la perspicacia y el paternalismo hacia un usuario que es, en última instancia, quien corre el riesgo. El documento presenta un estudio de caso, siguiendo la filosofía subyacente del arquitecto francés Yona Friedman, y una aplicación informática, llamada YONA. Como parte de una investigación más amplia, *Machine Recognition and Inference Making in Computer*, postulamos nuevos desarrollos que ejercitarán las técnicas de entrada y visualización en un entorno implacable, *implacable* en el sentido de que el usuario es exigente, el problema es difícil y el producto es un diseño personal y único” (Negroponte & Weinzapfel, 1976, 74)

El experimento se estructuró en 6 pasos, los tres primeros tienen que ver con la teoría de grafos desde la creación y conexión de espacios hasta la organización del ‘**grafo**’ permitiendo su modificación o rectificación en cada momento. En el siguiente se le permitirá al usuario mover los espacios cambiando su orientación y la relación entre otras habitaciones y en los pasos 5 y 6 se procede a la elaboración de un boceto y tras los cambios que el usuario considere oportunos, la impresión de este. Sus intenciones acerca del papel del usuario quedan reflejadas en todo el documento: “vemos estos usuarios como diseñadores, los cuales, al contrario de los arquitectos, no están entrenados o practican esta habilidad.” (Negroponte & Weinzapfel, 1976, 74)

El experimento realizado por el MIT no cuenta ni con la ‘**infraestructura**’ de la Ville-Spatiale ni otras características del ‘**hardware**’ ideado por Friedman, sin embargo, supone la prueba de que las teorías de Friedman son programables y no están encerradas en una realidad utópica. Además, los resultados obtenidos de este experimento representan un campo de prueba y error previo sobre la programación de estas teorías.

Podemos considerar el experimento como la verificación de que es posible llevar el método a la práctica. Lo que abre la puerta a la revisión de estos pensamientos sociales y arquitectónicos más que como un pensamiento utópico, como una realidad actual, por el interés social que sigue presentando. En el desarrollo de este ‘**software**’ están patentes muchas de las teorías explicadas. Por tanto, el desarrollo de YONA, prueba que las intenciones, características formales, y desarrollos informáticos que tuvo el proceso ideado por Friedman lo hacen apropiado para su programación, y por tanto, para su revisión en un contexto actual.



Fig.3.13.

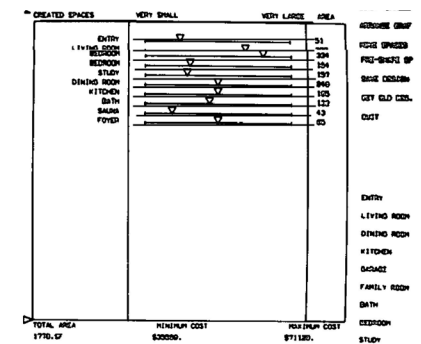


Fig.3.14.

Fig.3.12. Usuario eligiendo la configuración del grafo que ha diseñado en una pantalla táctil en el experimento “*Architecture-by-yourself*”

Fig.3.13. Pantalla del software “*The YONA system*” en la que se muestra la barra deslizadora para las elecciones de espacios y áreas de estos.

Fig.3.15.



Fig.3.16.

Pasos 1 y 2 del método, proceso de construcción del repertorio, tarea del arquitecto y urbanista.

Pasos 3 y 4 del método, proceso de representación y toma de decisiones por parte del futuro usuario.

Paso 5 del método, sistema de advertencia, y posibilidad de modificar las elecciones (posibilidad de retroceso)

1. AXIOMAS Y REGLAS DEL MAPPING

- 1- Producción de un espacio cerrado en el espacio preexistente. Un punto representa un espacio cerrado
- 2- De cada espacio cerrado hay al menos un camino hacia los otros espacios cerrados. Una línea representa un acceso.
- 3- Hay al menos dos diferentes categorías de espacios cerrados. Una etiqueta representa una diferencia

2. MAPEADO

Las reglas del mapping permiten construir las listas combinatorias completas de todas las soluciones posibles (sin omisión)

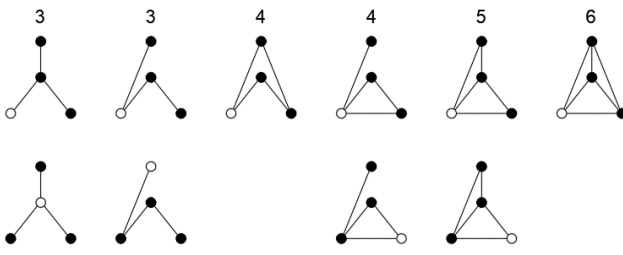
2 espacios cerrados solo tienen una unión posible



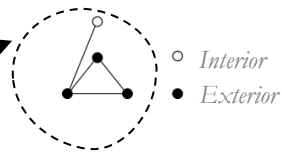
3 espacios cerrados tienen 3 esquemas de uniones posibles



4 espacios cerrados tienen 11 esquemas de uniones posibles

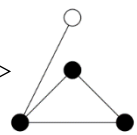


Elección del usuario dentro del repertorio



3. REPRESENTACIÓN DEL GRAFO

Grafos conexos (n=4),

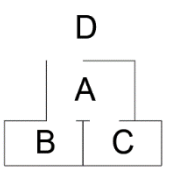


Matriz de adyacencia

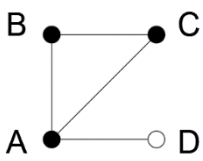
	A	B	C	D
A	0	1	1	1
B	1	0	1	0
C	1	1	0	0
D	1	0	0	0

1= adyacencia
0= no-adyacencia

Posible plano



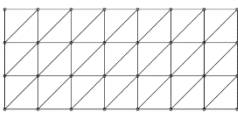
Otras representaciones



5. CORRECCIÓN DE ELECCIONES

4. ELECCIÓN DE UNA INFRAESTRUCTURA

Se debe asegurar una infraestructura que garantice la elección libre y democrática a todos los usuarios.



0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

0	1	0	0	1	1	0	0	1	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	1	1	0	0	0	0	0
1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	1	1	0	1	0	1	0	1	0	0	0	0	1	0	0
0	0	1	1	0	1	0	1	0	0	0	0	0	1	0	0

Representación del grafo y sus etiquetas (formas, disposición, etc.) en la infraestructura dada (sistema indeterminado)

Insertión de la elección del usuario en el punto elegido por él de la Ville Spatile.

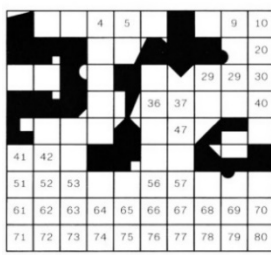


Fig.3.16. Recopilatorio de 3.1 Fundamentos del método de Yona Friedman

4. ESTRATEGIA

1. Secuencia para el desarrollo del sistema

Para probar la vigencia del método de Friedman en la actualidad se lleva a cabo una investigación con métodos de Expresión Gráfica actuales a través de herramientas como el Diseño Paramétrico y el Diseño Generativo.

El desafío de lo planteado no reside ni en la optimización de los resultados, ni la definición automática del modelo, si no en la elaboración de un **‘repertorio’** de figuras y la generación de un modelo que permita la libre elección dentro de un sistema indeterminado. Para ello nos basamos en una simplificación del método desarrollado por Friedman. Esto nos permite la creación de un repertorio a través de matrices de adyacencia donde la solución son todas las combinaciones posibles de los distintos espacios.

Para el procedimiento explicado se va a utilizar como herramienta gráfica Rhinoceros ® (Robert McNeel & Associate) y como herramienta de programación visual Grasshopper 3D ® desarrollado por David Rutten. Grasshopper 3D es un editor de algoritmos incluido en Rhinoceros 3D, que no requiere conocimientos avanzados de programación ni de scripting²⁰ como sucede en otros lenguajes de programación como Rhino.Scrip o Rhino.Phyton. Por ello, su uso se ha popularizado entre arquitectos y otros profesionales que no están familiarizados con la programación y que gracias a este lenguaje de programación visual pueden crear algoritmos sin la necesidad de escribir código, Grasshopper permite además la incorporación de ‘plugins’ de otros desarrolladores con el correspondiente script ya desarrollado. **Fig. 4.1.**

La definición de Grasshopper contiene distintos parámetros y componentes con los que se escribe el código. Este código se desarrolla con carácter lineal, lo que implica que un retroceso en el desarrollo del algoritmo inhabilitaría la definición. Esto condiciona el orden de la secuencia explicada a continuación que buscará la determinación de unos parámetros iniciales o, **‘axiomas’**, para la construcción del repertorio, acercándose este carácter lineal sin retroceso, a lo definido por Friedman, y permitiendo correcciones y variaciones de parámetros con respuesta instantánea del sistema.

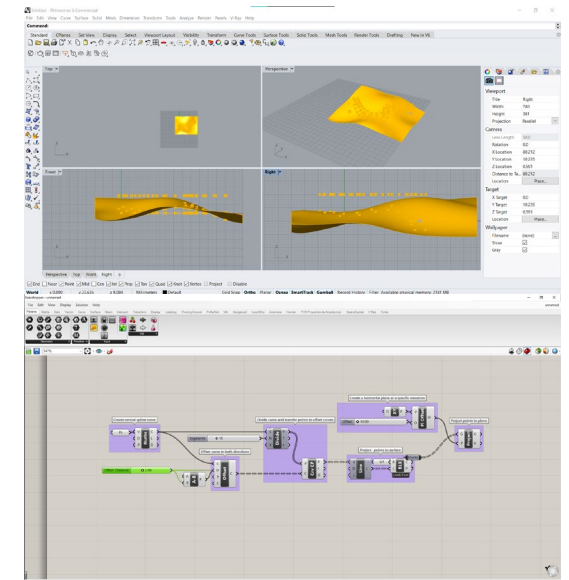


Fig. 4.1.

Fig. 4.1.. Pantalla superior, interfaz y entorno gráfico de Rhinoceros, inferior, Grasshopper 3D. Construcción de polisuperficie de distinta alturas.

20. Un lenguaje de scripting o script es un lenguaje de programación para un sistema en tiempo de ejecución que automatiza la ejecución de tareas que, de otro modo, serían realizadas individualmente por un operador humano. Los lenguajes de script suelen interpretarse en tiempo de ejecución en lugar de compilarse

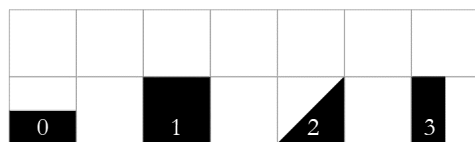


Fig. 4.2.

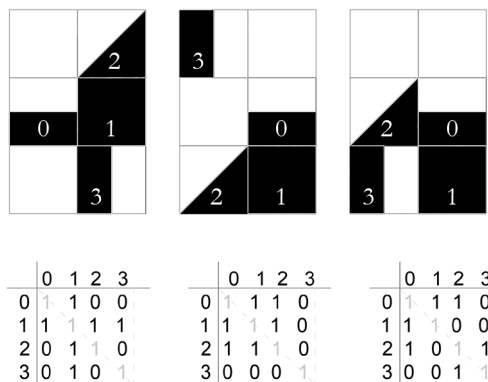


Fig. 4.3.

Fig. 4.2. Las 4 figuras escogidas (polilíneas cerradas), en una retícula cuadrada y numeradas con sus respectivos índices (0, 1, 2, 3)

Fig. 4.3. Muestra de posibles relaciones entre las piezas sin rotación de estas; debajo, la matriz binaria que define la adyacencia entre las piezas, cuya diagonal principal es 1 por las consideraciones tomadas en el trabajo.

La **secuencia** para desarrollar en Grasshopper para el modelo simplificado es la siguiente:

- 1º Selección de formas geométricas simples como sustitutos de las estancias o habitaciones.
- 2º Creación de todas las combinaciones posibles entre estas piezas a través del Diseño Generativo.
- 3º Selección de la distribución de piezas deseada a través del plugin *Space Syntax* (y su conversión a la matriz de adyacencia deseada, a partir de ahora, matriz ‘target’).
- 4º Cada una de las combinaciones lleva asociada una matriz de adyacencia, por lo que a través de la comparación de matrices de adyacencia filtramos las posibles combinaciones que cumplan las condiciones de adyacencia deseadas (matriz ‘target’).

Con objeto de simplificar, el experimento se ha realizado sólo con 4 figuras (estancias) que pueden encontrarse en 6 puntos de posición distintos sin variar su orientación con lo que se obtiene una combinatoria de 1295 grupos con las 4 figuras en diferentes posiciones. Cada grupo o combinación de piezas tendrá su propia matriz de adyacencia.

Esto nos permite, a través de la matriz ‘target’, filtrar de todas las combinaciones creadas, las posiciones cuya matriz de adyacencia (relaciones entre las piezas) coincidan con las deseadas, marcando así dentro del repertorio, los grupos de piezas o “apartamentos” que son **iguales topológicamente**¹² a la que deseamos conseguir. Es decir, distintas variaciones, pero con la misma distribución (adyacencia entre piezas).

Para este experimento bastará con que dos formas se toquen en un punto para cumplir la condición de adyacencia. Cada figura tiene un índice (0, 1, 2, 3), Fig. 4.2. La matriz obtenida es una matriz binaria que indica (para cada índice), con un 1 si dos de las figuras se tocan (adyacencia) y con un signo 0 si no existe esta relación (no-adyacencia). La matriz es siempre simétrica respecto a su diagonal principal y se establece por tanto, que las propias figuras tienen adyacencias con ellas mismas, siendo por ello la diagonal principal 1 en todas las matrices desarrolladas en este sistema. Fig. 4.3.

5. DESARROLLO

1. Creación y representación del Grafo

SYNTACTIC (Designing with *Space Syntax*) es un plugin para Grasshopper diseñado por Pirouz Nourian y Samaneh Rezvani, ingenieros de la Universidad de Delft. Esta herramienta recoge una síntesis de numerosos métodos de configuraciones espaciales, análisis de redes y establece un cuerpo metodológico sobre la representación de teorías de grafos y análisis de configuración espacial muy próximo al experimento ya mencionado, en “*Architecture-by-yourself*” de Nicholas Negroponte y Guy Winzapel (MIT) en colaboración con Yona Friedman, nombrado como YONA.

La herramienta SYNTACTIC nos permite elaborar un análisis de sintaxis espacial en tiempo real en la pantalla del ordenador a través de unos pasos sencillos e interactivo, también nos permite la visualización del ‘**grafo**’ y de un diagrama de burbujas interactivo, ambos muestran los cambios a tiempo real. Para demostrar la utilidad de esta herramienta y la relación con nuestro caso de estudio realizamos la siguiente comparativa.

El proceso de SINTATIC para el usuario es el siguiente:

1. Dibujamos una serie de puntos arbitrarios. Un punto representa un espacio.
2. Unimos cada uno de los puntos con líneas. Las líneas representan accesos (adyacencias).
3. Cada punto lleva asociado “dos etiquetas espaciales”, una lista con sus respectivos nombres, y otra con el área. Estas diferencias son las etiquetas del ‘**grafo**’. Fig. 5.1.

La metodología explicada lleva unas reglas, ‘**axiomas**’, asociadas a un proceso de ‘**mapeado**’ (representación gráfica de estas reglas), al igual que como vimos anteriormente en el método de Friedman:

- 1- Producción de un espacio cerrado en el espacio preexistente. Un • representa un espacio cerrado.
- 2- De cada espacio cerrado hay al menos un camino hacia los otros. Una — representa un acceso.
- 3- Hay al menos dos diferentes categorías de espacios cerrados. Una etiqueta representa una diferencia.

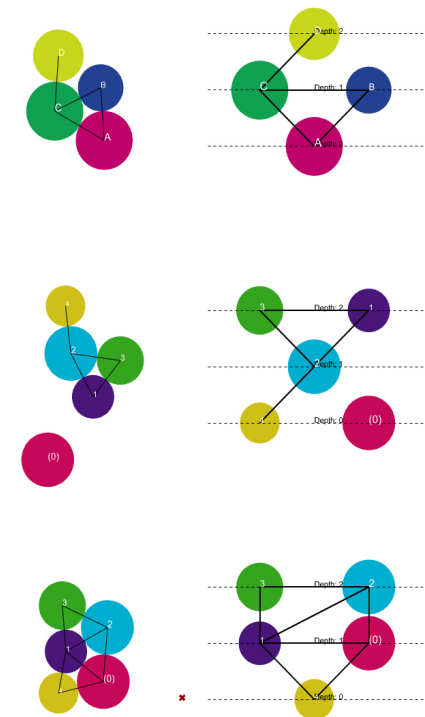


Fig. 5.1.

Fig. 5.1. Grafos dibujando puntos (espacios) y sus uniones con la herramienta Space Syntax



Fig. 5.2.



Fig. 5.3.

Fig. 5.2. Distintas formas de mostrar el mismo grafo a través de las configuraciones de *Space Syntax*.

Fig. 5.3. Representación de un grafo cuyas conexiones son mayores a lo establecido por Friedman $L=3(n-2)$, por tanto un grafo no plano, donde se produciría un punto de cruce.

Fig. 5.4. Creación de todos los grafos posibles para unir 4 espacios, 3 interiores y uno exterior. Arriba, según el método de Friedman (*blanco= espacio exterior*). Abajo, mismas conexiones realizadas con Space Syntax (*amarillo= espacio exterior*)

Fig. 5.5. Recreación del experimento "*Architecture by yourself*" llevado a cabo por el MIT en 1973 con la herramienta *Space Syntax*. Arriba, serie de imágenes del proceso llevado a cabo en el experimento. Abajo, proceso en *Space Syntax*.

Esta herramienta nos permite una representación directa de lo ya expuesto por Friedman, el problema para la implementación en la presente investigación es, como veremos a continuación, la obtención de la matriz de adyacencia del '**grafo**' dibujado con la herramienta SYNTACTIC. Hay que destacar, que también recoge grandes similitudes con el experimento planteado por el MIT, pues los pasos, grados de desarrollo, e intenciones que recogen resultan muy parecidos. Fig.5.4. y 5.5.

Con los antecedentes de *Space Syntax*, obtenemos la configuración espacial a través de las decisiones del usuario y lo representa en forma de '**grafo**', dando además, un listado de las relaciones de cada espacio con los demás, lo que constituirá el primer paso que llevará a la creación del repertorio.

Comparativa 1

Arriba, dibujos de grafos por el método de Yona Friedman para la obtención del repertorio; abajo, grafos topológicamente iguales a los de Friedman representados con Space Syntax.

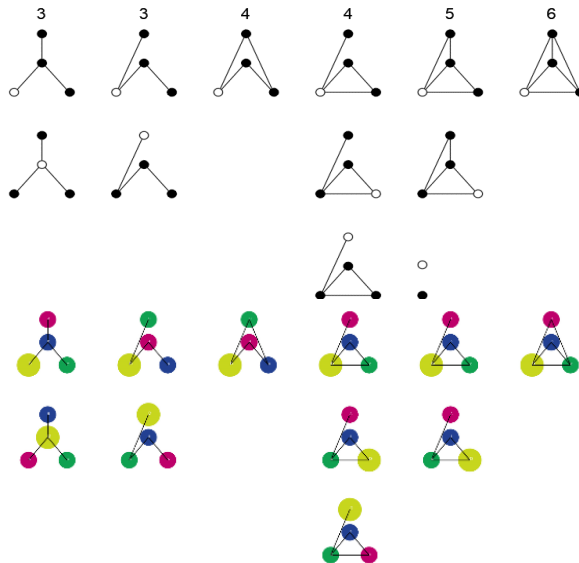


Fig 5.4.

Comparativa 2

Arriba, imágenes presentadas en “Architecture-by-yourself” (MIT) abajo, recreación de los pasos gracias al componente de Diagrama de Burbujas de Space Syntax.

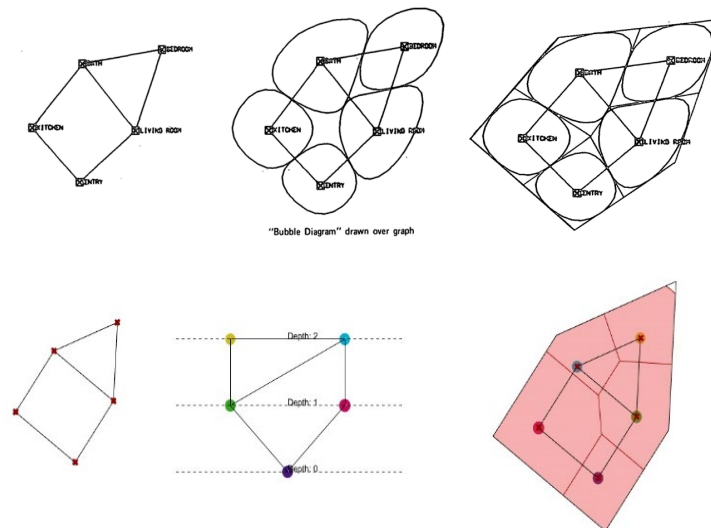


Fig 5.5.

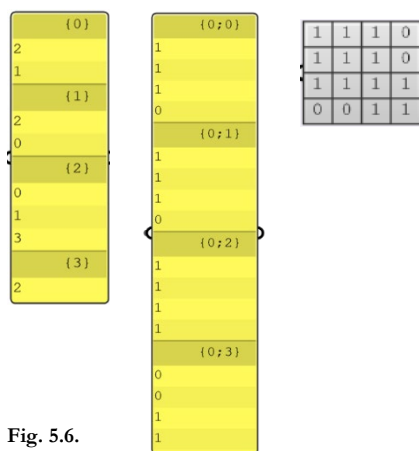


Fig. 5.6.

Fig. 5.6. A la izquierda, lista de conexiones obtenidas del grafo de *Space Syntax*; a la derecha, lista de adyacencias obtenida a través del método de conversión elaborado en este trabajo.

2. Obtención de matrices a partir del grafo

Tomando como base inicial *Space Syntax* nos proponemos la obtención de la matriz de adyacencia, y a partir de ella, de la ‘matriz de caminos’ y la ‘matriz de esfuerzos’ para seguir con coherencia el planteamiento desarrollado por Friedman.

Nos encontramos con el problema de que a pesar de que *Space Syntax* sí realiza distintos análisis espaciales, tanto de conectividad como de transformación del ‘grafo’, no existe una forma directa de obtener la matriz de adyacencia que explica Friedman, sin la que no podría plantearse una aplicación de esta herramienta para nuestro ejemplo práctico, pues es la matriz de adyacencia es una de las reglas fundamentales en la que se basa el proceso posterior.

No obtener esta matriz de adyacencia a través de *Space Syntax* nos generaría un problema de conectividad con el resto de componentes utilizados en nuestro procedimiento, lo que limitaría significativamente la investigación.

Para solucionar este problema se desarrolla un método de conversión de una de las salidas (*outputs*) del ‘grafo’. Desde la lista generada en el *Diagrama de conexiones G*, se propone una estrategia matemáticamente sencilla; la lista mencionada está elaborada de forma que en la sub-lista de cada elemento {0, 1, 2, 3} aparece el índice de cada elemento con el que tiene adyacencia (por ejemplo el elemento {0} tiene adyacencia con 2 y 1 (Fig. 5.6.)), así que la estrategia planteada para la obtención de nuestra matriz de adyacencia es contar el número de veces que cada elemento se da en cada lista. Tras esto, se genera otra lista secuencial para la adyacencia de cada uno de los elementos con él mismo (como ya se planteaba en las estrategias de esta investigación), que se sumará a la lista obtenida, obteniendo así la matriz de adyacencia (binaria y simétrica) desde el ‘grafo’.

La ‘**matriz de caminos**’, que nos indica los espacios que hay recorrer de una figura a otra ($1 = \text{conexión directa}$, $2 = \text{atravesar 1 espacio}$, $3 = \text{atravesar de 2 espacios} \dots$), se obtiene del componente *Integration Analysis* de *Space Syntax*, y la ‘**matriz de esfuerzos**’ se obtiene de multiplicar w (la frecuencia elegida por el propio usuario) y, realizando un sumatorio de esta, también podemos obtener el ‘**valor de esfuerzo local**’ ($d \cdot w$) de cada una de los espacios según las costumbres de nuestro usuario (número de veces al día con la que frecuenta cada habitación), proceso que se ha conseguido sistematizar por completo dentro del algoritmo, gracias al desarrollo de dos componentes para lo propuesto en esta investigación. Fig. 5.7.

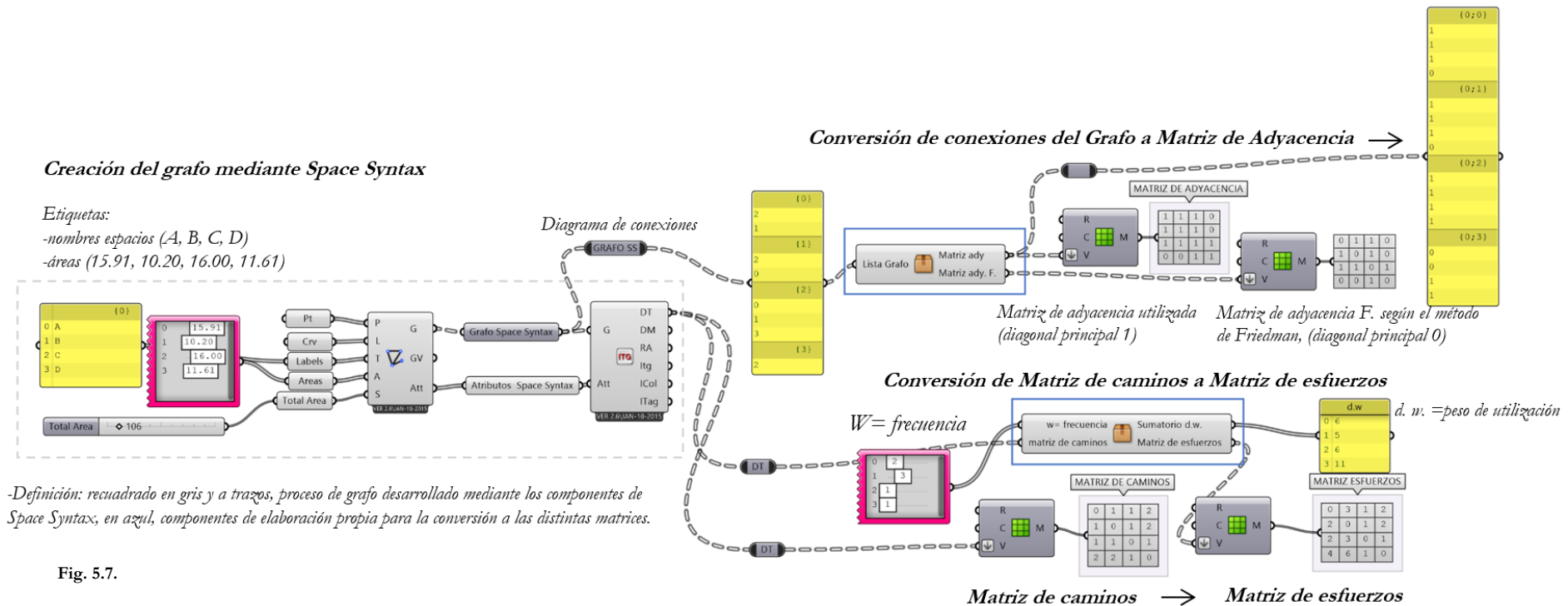


Fig. 5.7.

Fig. 5.7. Obtención de la matriz de adyacencia, a través de métodos de conversión elaborados.

3. Puntos de destino: elección de la infraestructura

La elección de lo que ya hemos definido como ‘**infraestructura**’ condicionará la disposición en la que se pueden encontrar nuestras piezas. Para una elección adecuada que nos permita obtener resultados significativos, se han tenido en cuenta los criterios establecidos por Friedman para su ciudad espacial. Por ello, finalmente se opta por una ‘**infraestructura**’ de celdas cuadrada con 6 puntos de posiciones, una retícula (*grid*) de extensión $x=1, y=2$. **Fig. 5.8.**

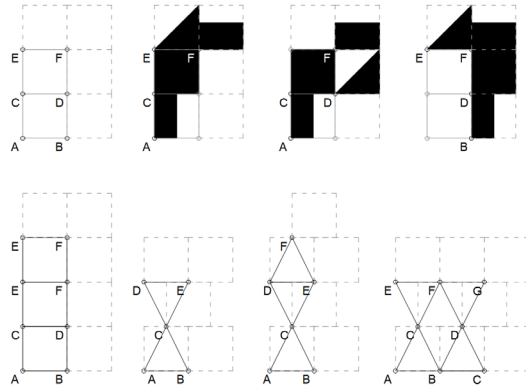


Fig.5.8.

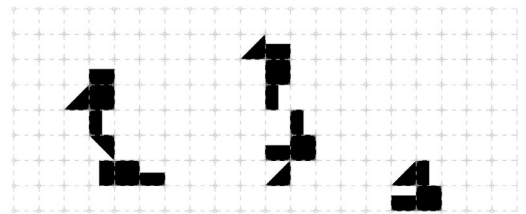


Fig.5.9.

Fig.5.8. Infraestructura elegida de celdas cuadradas, 6 puntos de posición, (A, B, C, D, E, F), correspondientes a una retícula cuadrada de extensión $x=1, y=2$, donde se pueden encontrar los vértices de las figuras.

Fig.5.7. Otros posibles puntos de posición para una infraestructura de celdas cuadradas como la empleada.

Para su determinación, se experimenta con distintas ‘**infraestructuras**’ basándonos en los distintos puntos de posición en los que se pueden encontrar nuestras piezas. En un primer momento se realiza el experimento con 4 puntos de posición (A, B, C, D). Para obtener resultados más significativos aumentamos los puntos de posición de las figuras de 4 a 6, pues cuantos más puntos de posición mayor será el número de combinaciones y por lo tanto los resultados de entre los que elegir. **Fig. 5.8.**

Este aumento de números de resultados implica una ralentización del algoritmo generado, debido al elevado número de posibilidades que el ordenador debe realizar. Ejemplarizando esto con números, en una combinación en la que 4 piezas pueden encontrarse en 4 puntos de posición distintos, obtenemos la siguiente estructura de datos: $\{0;255\}$ ($N=4$), lo que quiere decir que tenemos 255 grupos de 4 figuras cada grupo, un total de 1024 valores. Sin embargo, si deseamos obtener la combinatoria de las mismas cuatro figuras en 6 posiciones distintas, la cifra ascendería a 5.1834 valores con la estructura de datos $\{0;1295\}$ ($N=4$). Lo mismo sucede al aumentar el número de piezas.

Esto nos ayuda a comprender que aunque teóricamente las posibilidades de combinaciones pueden ser mucho más significativas que en lo planteado en este experimento, la potencia del ordenador juega un papel clave en la decisión del número de figuras y posiciones.

Aunque finalmente, por la concordancia con el método de Friedman, se decide dejar los resultados del experimento en función de una ‘**infraestructura**’ de celdas cuadradas, en la que los vértices de los cuadraros son cada uno de los puntos de posición en los que se puede encontrar la figura. No deja de sorprender las posibilidades y optimización que ofrece el cambio de este parámetro, por ejemplo si optamos por que los puntos de posición sean los vértices de una retícula triangular o los centroides de una retícula hexagonal, permitiendo con estas variaciones, infinidad de posibles resultados. Fig.5.8., 5.9. y 5.10

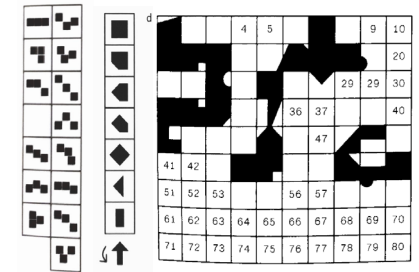


Fig.5.10.

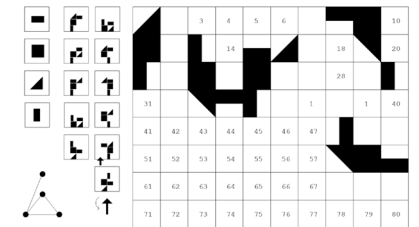


Fig.5.11.

Fig.5.10. Dibujo de Yona Friedman del Flatwriter.

Fig.5.11. Simulación, con los resultados obtenidos del experimento de lo planteado por Yona Friedman.

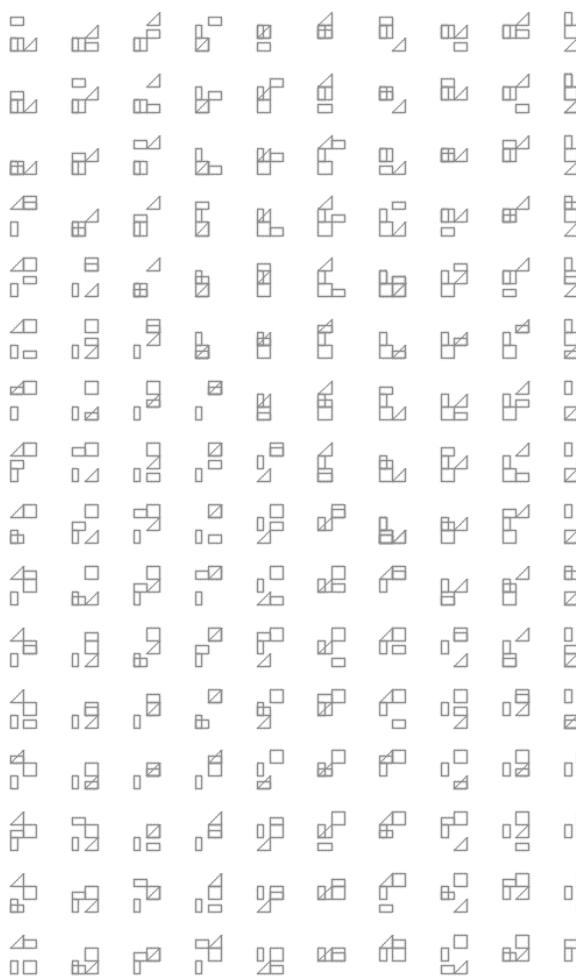


Fig. 5.12.

Fig. 5.12. Selección de parte del repertorio con las combinaciones de cada una de las cuatro figuras (0, 1, 2, 3) en los seis puntos de posición (A, B, C, D, E, F)

4. Creación de un repertorio completo

Teniendo clara la posición de nuestras piezas (**‘infraestructura’** elegida), el siguiente objetivo es elaborar una estrategia intermedia con la que, como explica Friedman, podamos elaborar un **‘repertorio’** de todas las combinaciones posibles entre los espacios.

La estrategia para elaborar este **‘repertorio’** es la siguiente: seleccionamos 4 figuras (formas simples creadas a través de polilíneas cerradas), cada una de estas figuras lleva asociado un índice (0, 1, 2, 3) Fig. 5.13., y se analiza la colisión de cada una de las figuras con el grupo completo²¹(las 4 figuras de índice (0, 1, 2, 3)). El resultado de colisionar nuestras figuras con esta estrategia es una lista de valores booleanos (Boolean Values)²². Por tanto, *true* si existe adyacencia con otra de las piezas de la lista (y con ella misma) y *false* si no existe dicha condición de adyacencia. Expresando el resultado en un valor numérico, *true*=1, *false*=0. (Como la lista mostrada en la Fig. 5.6.). De aquí se obtiene otra matriz de adyacencia, que representa las relaciones entre los espacios a través de matrices binarias y simétricas respecto a su diagonal principal tal y como la describe Friedman, con la única diferencia de que la diagonal principal es 1.

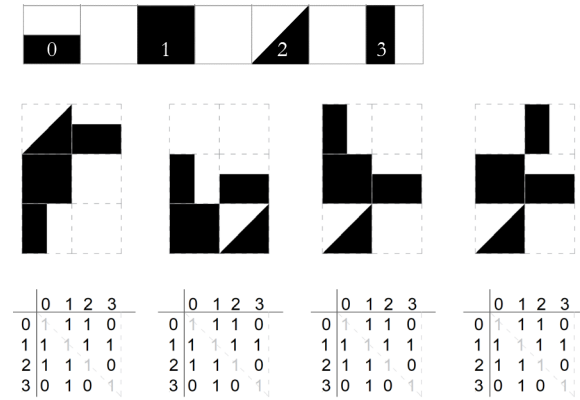
De esta manera podemos crear un repertorio con todas las combinaciones posibles y evaluar en cada grupo de 4 figuras su adyacencia, lo que más adelante nos permitirá elegir de entre todas las soluciones que creemos las que sean iguales a nuestra matriz ‘target’. Con este procedimiento conseguimos la construcción de un **‘repertorio’** de figuras construido directamente por el ordenador dependiendo de la **‘infraestructura’** (puntos de posición) y las formas elegidas.

Friedman elabora en su método un **‘repertorio’** de matrices: para un grafo $n=4$, obtiene 11 **‘grafos’** **‘topológicamente distintos’** con sus 11 matrices correspondientes. Lo que propone el método creado es sin embargo, la creación de un **‘repertorio’** de figuras que incluya todas las tipologías posibles (posiciones de unas piezas respecto a otras), por lo que las soluciones obtenidas son significativamente mayores. En

concreto, con los parámetros elegidos para el ejemplo desarrollado, 1295 posibles combinaciones de las 4 piezas (posibilidades que se filtrarán más adelante para seleccionar las que cumplan determinadas condiciones).

Los 11 ‘**grafos**’ correspondientes a todas las opciones ‘**topológicamente distintos**’ de unión de 4 figuras, serán los creados directamente por el usuario a través de lo explicado en 5.1 *Creación del grafo*. A partir de estos, el ordenador filtrará todas las posibilidades de posición con esta adyacencia dentro del ‘**repertorio**’ ya generado.

De esta forma, tenemos 1.295 matrices binarias de 4x4, donde 1 representará *conexión directa* entre los espacios cerrados y 0 *no conexión*, dependiendo de la posición de cada una respecto a las demás, lo que constituirá nuestro ‘**repertorio**’ completo (de todas las posibilidades de posición de las 4 piezas en los 6 puntos de posición). Todas las combinatorias generadas se muestra una retícula bidimensional, dónde cada una de las combinaciones posibles aparece en una celda de la retícula, permitiendo visualizar todas las opciones generadas al mismo tiempo, así cómo los cambios en la elección o en el ‘**grafo**’ que decida el usuario en cada momento de su proceso de diseño.



5.13.

5.13. Ejemplo de distintas combinatorias del repertorio con la misma matriz de adyacencia. (0=rectángulo, 1=cuadrado, 2=triángulo, 3=pilar) Estas figuras son topológicamente iguales (mismas conexiones entre las piezas) sin embargo representan una variedad, 4 tipos de tipologías para la misma adyacencia.

21. Por esta condición de nuestro modelo de evaluación asumimos en la investigación las figuras tienen adyacencia con ellas mismas.

22. Un valor booleano representa un valor de verdad; es decir, TRUE o FALSE

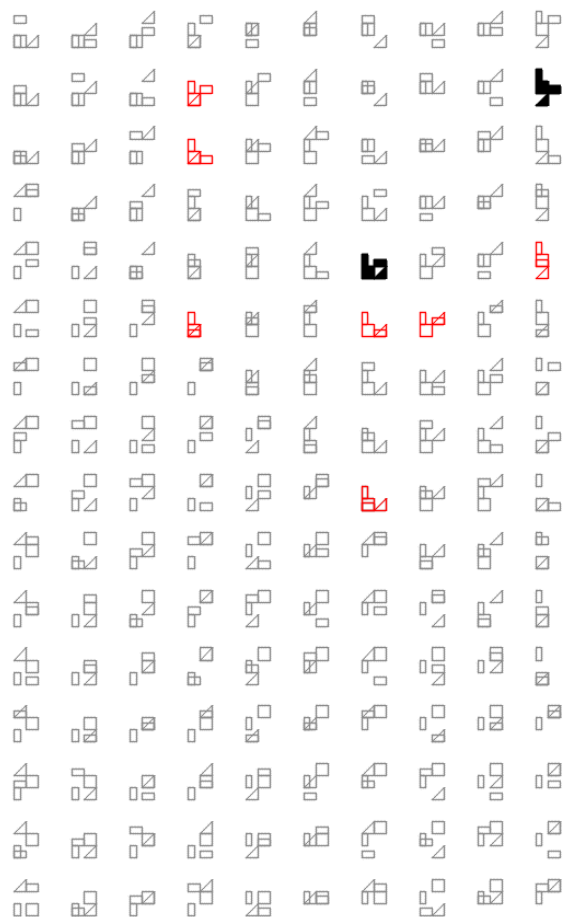


Fig.5.14.

Fig.5.14. Selección de parte del repertorio con las combinaciones de cada una de las cuatro figuras, resaltadas en rojo, combinatorias coincidentes con la matriz de adyacencia, pero que no cumplen la condición de área, en negro las combinatorias que cumplen ambas condiciones

5. Elección y filtrado de combinatorias

Una vez tenemos la adyacencia entre espacios deseada (tras la creación del ‘**grafo**’) y un ‘**repertorio**’ con todas las combinaciones dependiendo de las distintas posiciones en la que se pueden encontrar las piezas, (resultado del análisis de colisiones en cada una de las combinaciones de nuestro ‘**repertorio**’), debemos poder elegir (filtrar), de entre todas las combinaciones, la deseada.

Para ello, recurrimos otra vez a un método lógico y objetivo que nos permita en primer lugar, seleccionar las matrices matemáticamente iguales a nuestra matriz ‘target’, y después, eliminar de estas combinaciones seleccionadas aquellas en las que dos piezas se superpongan (por encontrarse en el mismo punto de posición).

Este proceso de selección responde a la metodología del Diseño Generativo, pues generamos todas las soluciones posibles, para más adelante seleccionar las que cumplan ciertos requisitos. Los requisitos para la selección en nuestro caso son dos: el primero, la igualdad a nuestra matriz de adyacencia ‘target’, y el segundo, la no superposición de piezas. Por lo tanto, nuestros dos filtrados se elaboran del siguiente modo:

1. Como disponemos de todo el ‘**repertorio**’ ya generado y evaluado (con todas las matrices de adyacencia generadas), tenemos en primer lugar que procesar todos los datos, de manera que, consigamos extraer de cada grupo de 4 figuras, su matriz de adyacencia 4x4. Una vez que tengamos estos datos, le restaremos a cada una de las matrices nuestra matriz ‘target’. Las matrices resultantes cuyo resultado sea 0 serán las que cumplan esta primera condición.
2. Para evitar que dentro de nuestra elección se representen las combinaciones en las que las piezas se superpongan, el método consistirá en hallar el sumatorio del área de nuestras 4 figuras por separado. Las combinaciones cuya área sea menor al área obtenida serán descartadas también de nuestra selección.

De esta manera el futuro usuario podrá evaluar de entre las distintas combinaciones mostradas, y elegir de entre ellas la que más satisfaga sus expectativas. Respondiendo así, a las intenciones de Yona Friedman y

siguiendo su método de construcción del ‘repertorio’, de la manera explicada en el ejemplo del Flatwriter: “Este ejemplo muestra el método que proponemos para establecer un repertorio al servicio del cliente del arquitecto o del urbanista. El cliente (al que más arriba hemos llamado futuro usuario, dispondrá realmente de plena libertad para escoger cualquier punto posible, en vez de tener que seguir obligatoriamente las preferencias de tal o cual persona (arquitecto o urbanista)” (Friedman, 1971, 49)

Las intenciones del ejemplo planteado aquí, pretenden responder al mismo tiempo, a las corriente de empoderamiento del usuario no experto, por la que tanto abogaba Friedman, definiendo las tareas del nuevo arquitecto y urbanista como: “Asegurar la constitución de un ‘stock’ de todas las posibilidades de conjuntos que un cliente es susceptible de elegir (se trata de asegurar el ‘hardware’, es decir ‘la infraestructura’), de tal forma que cuando el cliente haya hecho su elección, tenga la posibilidad de cambiarla o modificarla (y por consiguiente de modificar el consiguiente ‘hardware’) cuando lo desee.” (Friedman, 1971, 49)

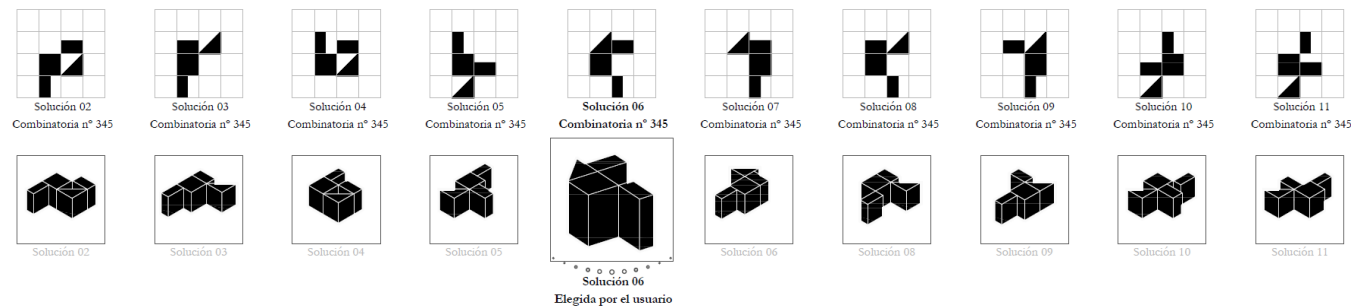


Fig.5.15.

Todas las opciones se encuentran dentro del repertorio. La tarea del usuario será dibujar el grafo con la disposición que desea, y el algoritmo filtrará las combinaciones que cumplan dicha condición mostrando el número de resultados, dentro de esas posibilidades el usuario podrá escoger la que más se adapte a sus necesidades.

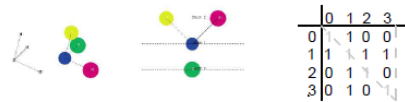
01. Resultados del grafo y matriz de adyacencia presentados:



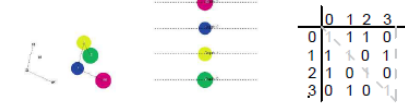
02. Resultados del grafo y matriz de adyacencia presentados:



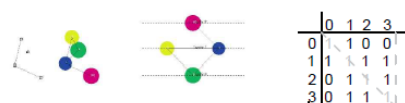
03. Resultados del grafo y matriz de adyacencia presentados:



04. Resultados del grafo y matriz de adyacencia presentados:



05. Resultados del grafo y matriz de adyacencia presentados:



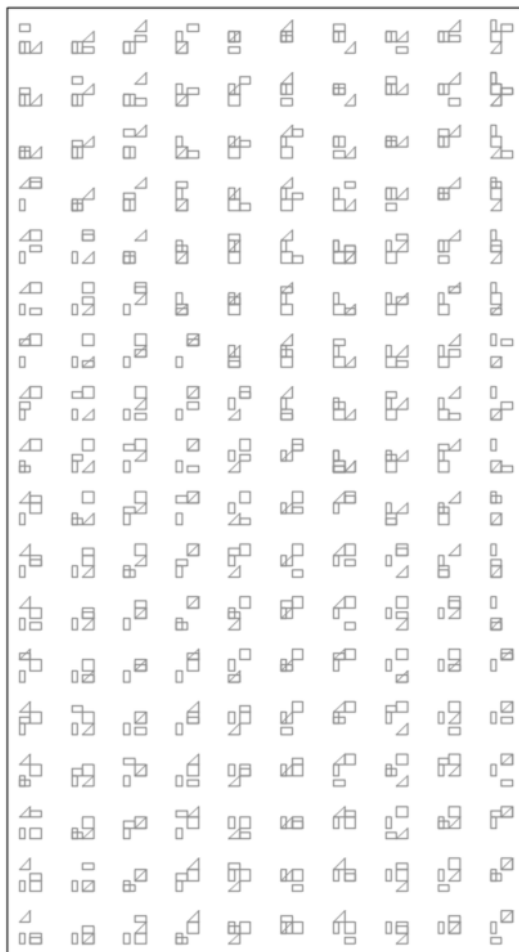
Legenda de índice y de colores de las piezas:



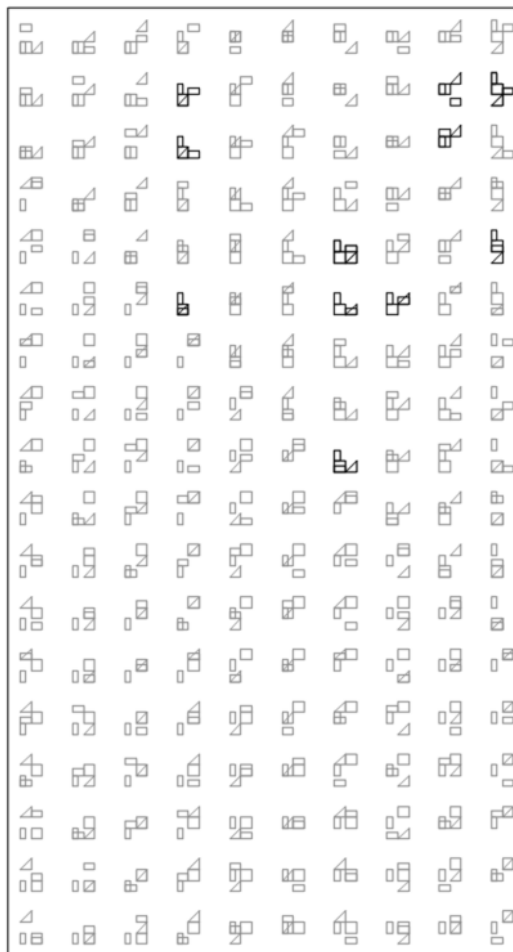
Fig.5.16.

Paso 1.

Se muestran todas las combinaciones creadas a partir de 4 piezas y 6 posiciones.

**Paso 2.**

En negro las combinaciones con adyacencia común a la matriz objetivo.

**Paso 3.**

Sombreadas las combinatorias filtradas del paso 2 dónde ninguna figura se superpone.

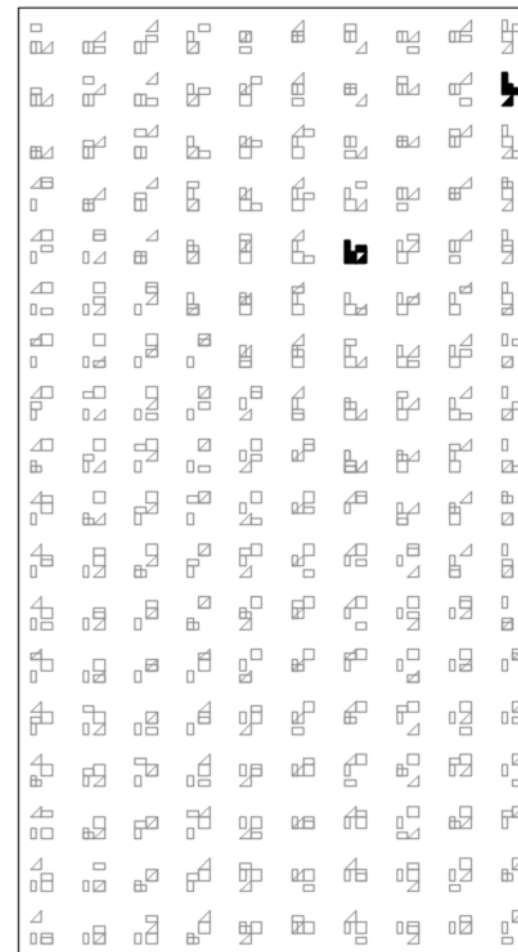


Fig.5.17.

En negro figuras seleccionadas

Fig.5.18.

Muestra del repertorio

Figuras coincidentes a nuestra matriz 'target' en negro. 620 combinaciones mostradas de un total de 1295 (4 piezas, 6 posiciones)

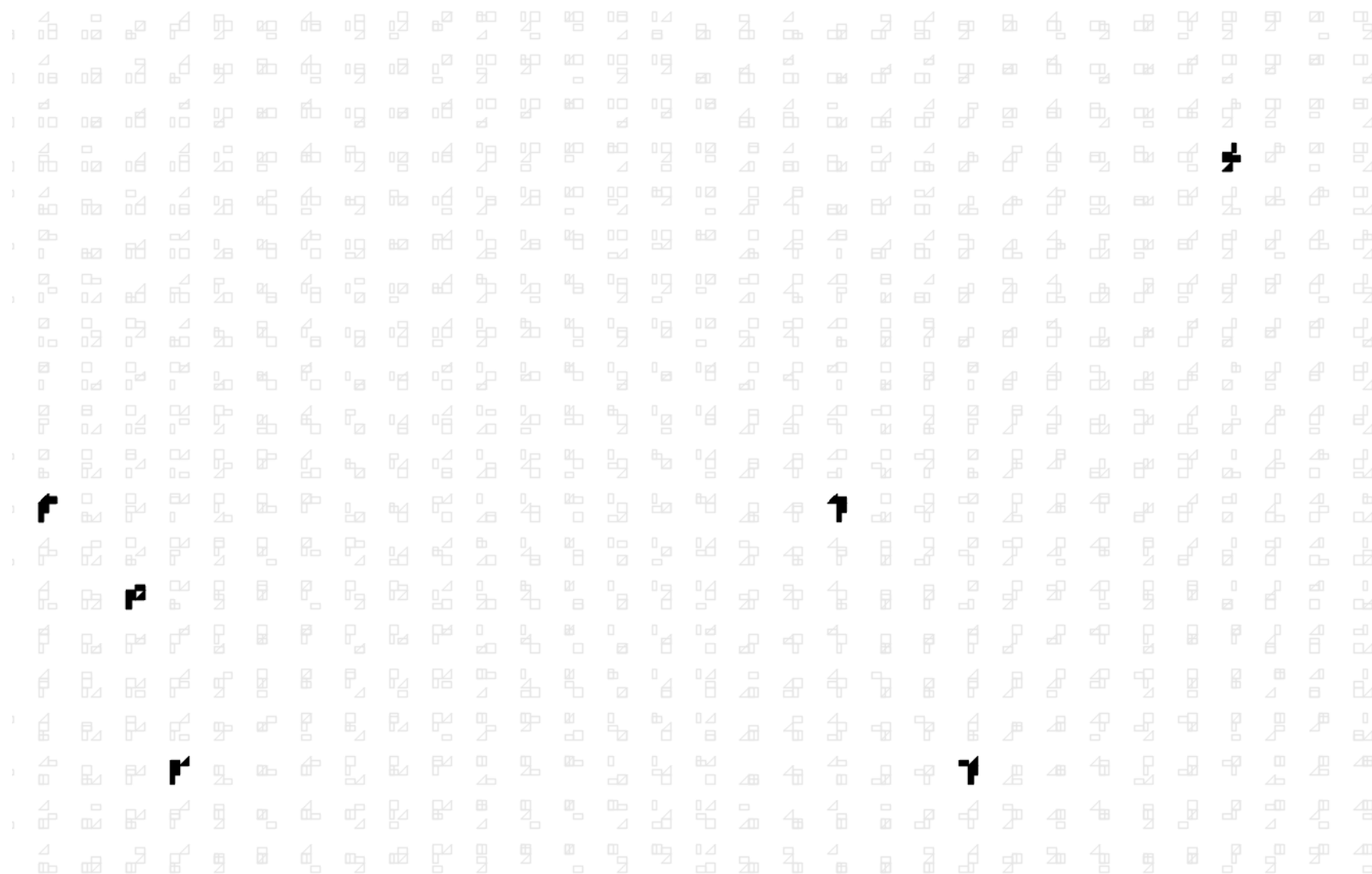
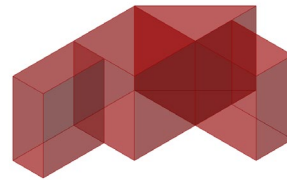
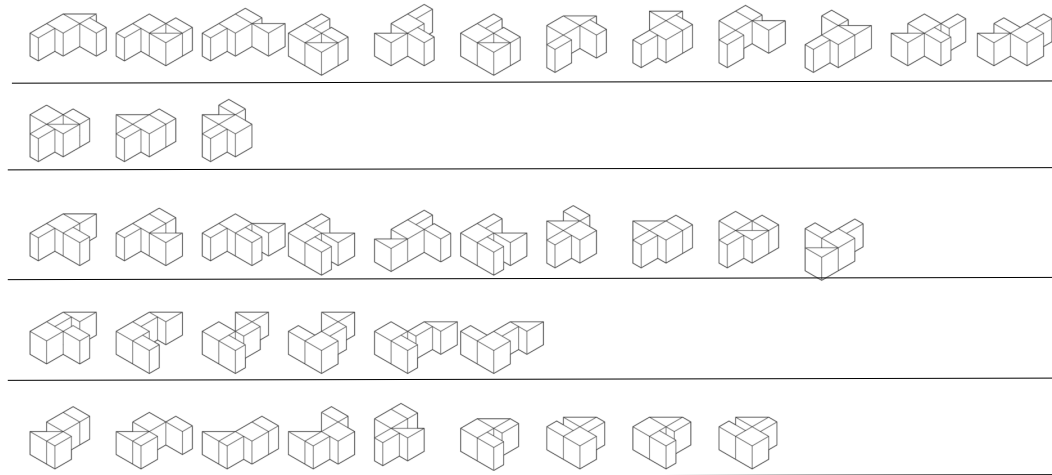


Fig.5.19.

Elección del usuario dentro del repertorio

El futuro usuario puede elegir entre un 'stock' de figuras dependiendo del grafo que dibuje en el ordenador



Distribución elegida

Fig.5.20.

Inserción en la Ville Spatiale

El futuro usuario elige el punto desea ubicar su vivienda

1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21

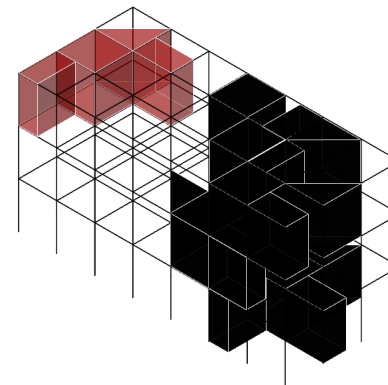
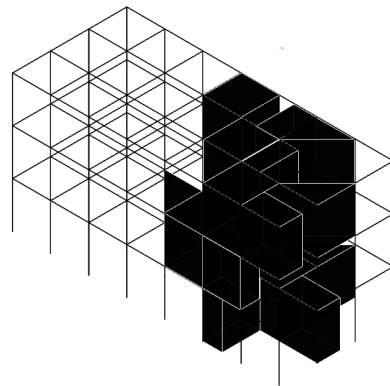
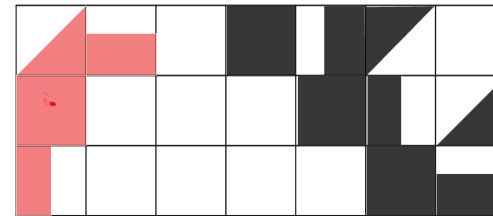
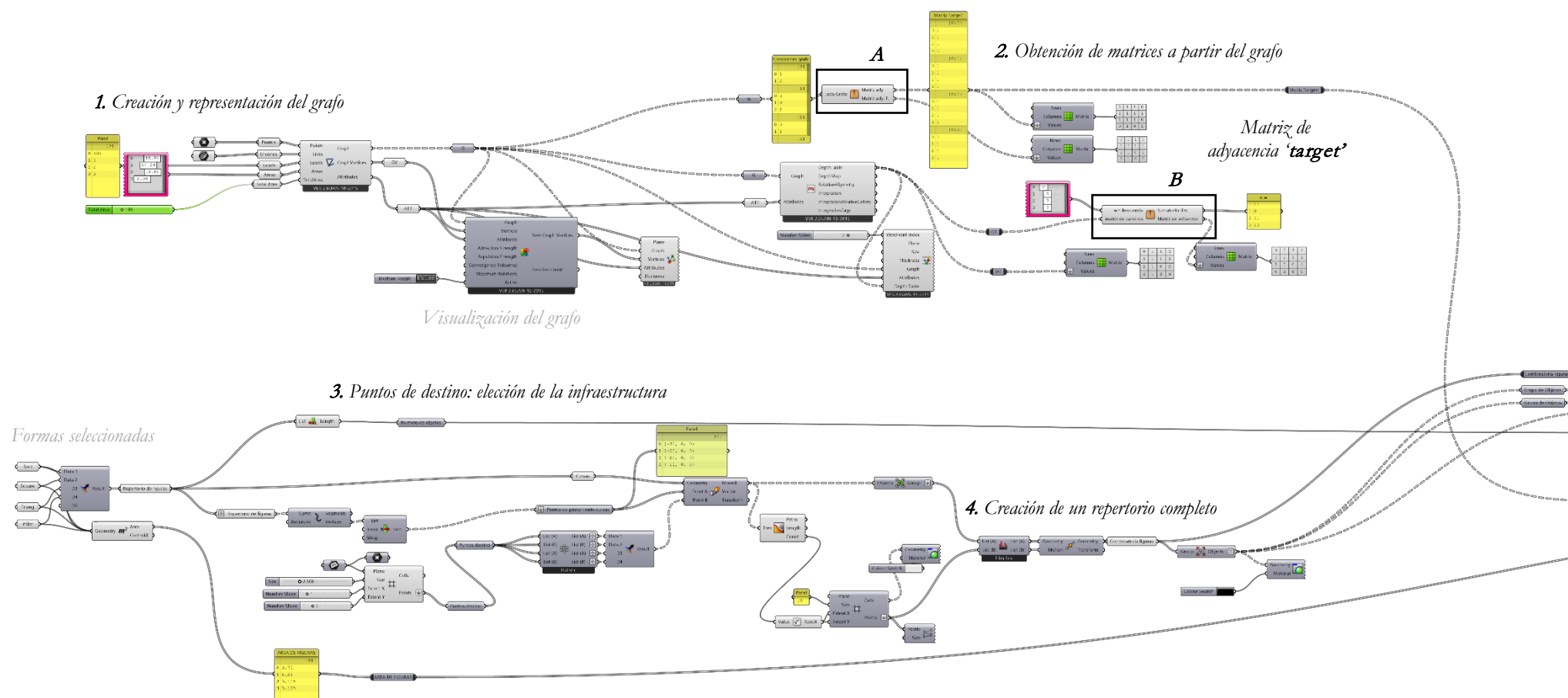


Fig.5.21.

6. Definición.



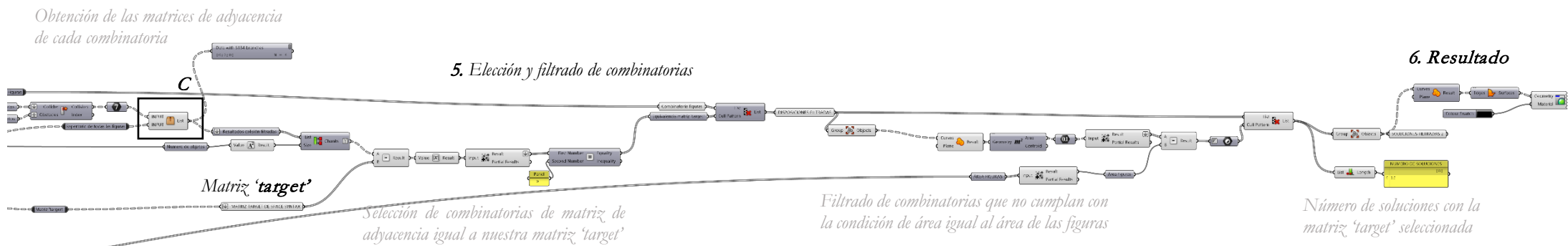


Fig.5.22.

Recuadrados en negro. Componentes de elaboración propia desarrollados por el autor para el sistema planteado

A. Conversión del grafo a Matriz de Adyacencia

B. Desarrollo de la matriz de esfuerzos y peso de utilización

C. Sistema para el filtrado de colisiones en grupos

6. DISCUSIÓN DE RESULTADOS

1. Evaluación del sistema

En las pruebas realizadas hasta el momento, se utiliza el Diseño Generativo para la obtención del ‘repertorio’ a través de todas las combinaciones posibles con las formas elegidas. Posteriormente, las soluciones filtradas que cumplen las condiciones marcadas pasarían a ser redibujadas más en detalle, con la inclusión de mobiliario, puertas, etc. Se podría definir en mayor detalle incluso, su inserción en la ‘infraestructura’ definida por Friedman o, en cualquier otra trama urbana definida previamente.

Con los resultados obtenidos, podemos razonar que es posible la consecución del objetivo propuesto por Friedman, mediante la programación paramétrica actual, aunque esta experiencia es susceptible de ser refinada. En primer lugar, cabría solucionar el hecho de que en una adyacencia real de formas arquitectónicas, basta con un solo punto de contacto para cumplir la condición de adyacencia de las piezas, sino con un tramo o lado compartido. Otra cuestión para optimizar es que el elevado número de valores generados y la potencia de cálculo que esto requiere. Por lo que al aumentar el número de posiciones y de piezas se ralentiza significativamente el cálculo del algoritmo.

Nº de figuras	4 posiciones	6 posiciones	8 posiciones
4	{0;255} (N=4), 1.024 valores, Tiempo de cálculo: 24.6 s*	{0;1295} (N=4), 5.184 valores Tiempo de cálculo: 640.4 s*	{0;4095} (N=4), 16.384 valores -
5	- -	{0;7775} (N=5), 38.880 valores -	{0;32767} (N=5), 163.840 valores -

*En azul los valores favorables para el cálculo hasta el momento

En términos generales, los resultados muestran que se puede plantear esta estrategia como base para una investigación más extensa. Dados los resultados significativos obtenidos con las 4 figuras y 6 posiciones, y los métodos de obtención de matrices y conversiones, como el de la obtención de la matriz de adyacencia a través del grafo generado por *Space Syntax*, podemos calificar los resultados obtenidos como satisfactorios, aunque conviene formular ciertas mejoras para que el sistema funcione de manera más eficiente.

El hecho de aumentar el número de figuras a 5 (o más) como vemos dispararía el número de valores calculados, si a esto a demás le sumamos el hecho de que las piezas giren (por ejemplo, 4 posiciones de giro 90°, por seguir con la definición de nuestra infraestructura cuadrada), la cantidad de datos que debería calcular un ordenador convencional sería desmesurada.

Por lo tanto, se plantearía en primer lugar que la condición adyacencia entre figuras se diera cuando estas compartieran un tramo (longitud) previamente definida en nuestros parámetros, para en segundo lugar, optimizar el algoritmo para la obtención de resultados en menor tiempo, lo que nos permitiría seguir el experimento con más piezas de mayor complejidad y más puntos de posición (por lo tanto, mayor número de disposiciones y de resultados para cada ‘**grafo**’ dibujado).

Si con una optimización del sistema presentado no fuera suficiente, podría plantearse el desarrollo del sistema con un Método Evolutivo. Método que se descartó en un primer momento, pues no nos permitiría la creación de un repertorio como el desarrollo, y no cumpliría en principio con la elaboración “completa y exhaustiva” por no mostrar todas las posibles soluciones, pero que permitiría mayor rapidez en el desarrollo planteado.

7. CONCLUSIONES

1. Vigencia de las teorías de Friedman

El Flatwriter nace para intentar un acercamiento a la democratización de la arquitectura. Por ello, pese a que este trabajo aborda la posibilidad de implementar el método de creación de un repertorio desarrollado en “Hacia una arquitectura científica” con las herramientas de expresión gráfica actuales, no debe olvidarse en ningún momento ni el espíritu crítico, ni el compromiso político y social del que nacen todas las ideas y manifiestos elaborados por Friedman durante más de 60 años.

La elección del tema viene marcada por el intento de Friedman de reconciliar a la sociedad con el papel del arquitecto, situando al usuario en el foco de las decisiones. El descontento social respecto a la arquitectura que se construye en ese periodo es algo que podría considerarse reflejado, por diversos motivos, en la sociedad actual y por lo que cabría plantear una revisión del papel del arquitecto.

El rigor con el que Friedman expone sus ideas es clave para asegurar el éxito de su implementación, algo que se ha conseguido satisfactoriamente siguiendo los pasos y la lógica para conseguir elaborar un repertorio ‘completo’ y ‘exhaustivo’, que contiene todas las posibles combinaciones utilizando como herramienta de expresión gráfica Rhinoceros+Grasshopper 3D.

Tras el desarrollado del modelo propuesto en este trabajo, no resulta arriesgado plantear que las ideas renovadoras y visionarias de Friedman, patentes en el Flatwriter continúan generando nuevas perspectivas e inquietudes aún 50 años después, lo que nos lleva a plantearnos su utopía en el panorama actual.

Los problemas a los que se enfrenta la sociedad actual y la vigencia de las teorías de Friedman, así como la necesidad de nuevos métodos de comunicación arquitecto-cliente y el desarrollo de herramientas de participación y emancipación del usuario, con la intención de crear una arquitectura más justa, permiten la revisión de estas corrientes de pensamiento con la intención de aportar otro enfoque de cómo estas herramientas podrían ayudar en la toma de decisiones del cliente, tratando de socializar la arquitectura y haciéndola más accesible a usuarios no expertos. Además, sugiere la expansión del ámbito de la arquitectura

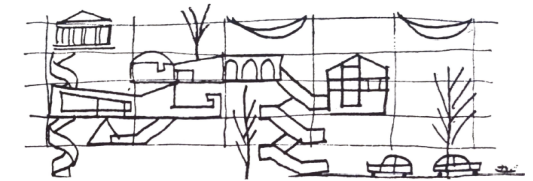


Fig.7.1.

a nuevos campos, más allá del diseño y a la construcción, algo que parece estar dándose ya, por ejemplo, con el auge de las arquitecturas paramétricas y la programación en la arquitectura.

El éxito del método es, a mi juicio, más que la creación de un sistema indeterminado vinculado a las decisiones del usuario, la elaboración de un sistema ‘complejo’ explicado de forma muy sencilla y visual, un mensaje capaz de llegar a todo el mundo, un modo de democratizar la arquitectura a través de la información, haciéndola accesible a toda la sociedad. El diseñar junto al cliente y para el cliente, debe ser además de una responsabilidad, una oportunidad, y sin embargo, resulta algo poco recurrente en nuestra profesión.

Por otro lado, analizando desde el punto de vista arquitectónico, el desarrollo del repertorio y la elección de parámetros, resulta inevitable juzgar cierta ingenuidad en su simplicidad, pero no debemos olvidar que es esa misma simplicidad la que permite que sea fácilmente extrapolable y programable. En la medida que se cuente con mayor potencia de cálculo, se podría introducir mayor complejidad a este modelo, por ejemplo, aumentando el número de piezas (estancias), incrementando las posiciones posibles y orientación, o variando e incrementando el número de formas escogidas en un primer momento.

Pese a estas posibles futuras mejoras, sí puede concluirse que es posible la programación del método de Friedman (y por lo tanto también de su ejemplo práctico, el Flatwriter). Aunque se han observado ciertas limitaciones en la herramienta gráfica escogida Grasshopper 3D, (respecto a otras herramientas de programación como Rhino.Scrip o Rhino.Phyton), se considera un acierto el método elegido pues gracias a Grasshopper, se ha podido programar el repertorio de principio a fin, con lo explicado en *“Hacia una arquitectura científica”*, permitiendo además una visualización de cada uno de los pasos y cierta autonomía que podría permitir en un futuro la utilización de este algoritmo con usuarios no expertos.

Otra conclusión implícita en el modelo desarrollado, es la demostración de la facilidad de implementación de matrices binarias en la programación arquitectónica, algo sobre lo que se está investigando en otros experimentos como el realizado por Reinhard König y Christian Bauriedel de la Universidad Técnica de Kaiserslautern. Sin embargo, con los antecedentes de Space Syntax y el desarrollo de un método de conversión para la construcción de matrices de adyacencia binarias, podrían abrirse nuevas líneas de investigación para la sintaxis de espacios arquitectónicos, incluso para su programación por la sencillez que plantea a nivel matemático el manejo de datos desarrollados en este trabajo.

Tras este desarrollo cabe plantearse, aunque sea de manera superflua cómo sería la aplicación del método llevado al extremo en la vida real, y las aportaciones que esto tendría para la participación de usuarios no expertos en la toma de decisiones y el rol de los arquitectos. Además con el desarrollo de internet y las experiencias de diseño colaborativo, se van incrementando en los últimos años con plataformas como ArcBazar cofundada por Imdat As, un sitio web que permite la competencia de diseños en línea para que los propios clientes elijan entre las propuestas presentadas (Crosbie, M. J., 2018). Lo que representa un claro ejemplo de la implementación en plataformas web, que con la ayuda de técnicas de Inteligencia Artificial podría dar lugar a un acercamiento a la democratización propuesta por Friedman con ejemplos como el planteado en este trabajo.

8. REFERENCIAS

Bibliografía

Ameijde, J. v., 2019. The Architecture Machine Revisited. *SPOOL*, 6(1), pp. 17-35.

Bianic, D., 2015. *Empowering Users through Design: Interdisciplinary Studies and Combined Approaches for Technological Products and Services*. London: Springer International Publishing.

Duarte, J. P., 2004. Towards the mass customization of housing: the grammar of Siza's houses at Malagueira. *Environment and Planning B*, 32(6), pp. 347 - 380.

Conrads, U., 1971, *Programs and manifestoes on 20th-century architecture*. Cambridge (MA), London: MIT Press.

Friedamn, Y., 1971. Choise by computer. *Progressive Architecture*, March, Issue 3, pp. 98-101.

Friedman, Y., 1971. *Hacia una arquitectura científica*. Madrid: Alianza editorial.

Friedman, Y., 2001. *Interview with Yona Friedman* [Interview] (January 2001).

Friedman, Y., 2005. *A conversation with Yona Friedman* [Interview] 2005.

Friedman, Y., 2006. *Pro Domo*. Sevilla: Actar Publishers.

Friedman, Y., 2016. *Interview with Yona Friedman: "Imagine, Having Improvised Volumes 'Floating' In Space, Like Balloons"* [Interview] (June 2016).

Friedman, Y. & Orazi, M., 2015. *The Diluation of Architecture*. Zurich: Park Books, Archizoom-EPFL.

Friedman, Y. & Rodriguez, M. I., 2011. *Arquitectura con la gente, por la gente, para la gente..* Barcelona: Actar.

Grasser, A. & P. A. & H. U., 2020. *Pervasive Collaboration and Tangible Complexity in Realtime Architecture*. Berlin, eCAADe (Education and Research in Computer Aided Architectural Design in Europe, pp. 393-400.

Negroponte, N., 1975. *Soft Architecture Machines*. Cambridge (MA), London: MIT Press.

Negroponte, N., & Weinzapfel, G., 1976. Architecture-by-yourself: An Experiment with Computar Graphics for House Design. *Computer Graphics*, Volume 10, pp. 74-78.

Nourian, P., 2016. *Configraphics: Graph Theoretical Methods for Design and Analysis of Spatial Configurations*. Nº1 ed. Delft: TU Delft.

Nourian, P., Rezvani, S. & Sariyildiz, S., 2013. *A syntactic architectural design: integrating real-time space syntax analysis in architectural design process*. Seul, Sejong University.

Reinhard, K., & Christian, B., 2004, Computergenerierte Stadtstrukturen (TU Karlsruhe, Technischen Universität Kaiserslautern)

Rossi, D. & C. g. R. & M. G. & R. M., 2017. *Inspired by the Atmosphere. See the Invisible..* Brixen, IMMAGINI? Conference 2017, p. no. 9: 906.

Tedeschi, A., 2015. *AAD Algorithms-Aided Design: parametric strategies using grasshopper*. Brienza: Le Penseur.

Vardouli, T., 2012. *Design-for-Empowerment-for-Design: Computational Structures for Design Democratization*. (MIT, Massachusetts Institute of Technology).

Vardouli, T., n.d. *"Architecture-by-yourself": Early studies in computer-aided participatory design*. (MIT, Massachusetts Institute of Technology).

Weinzapfel, G., Johnson, T. & John, P., 1971. IMAGE: an interactive computer system for multi-constrained spatial synthesis. *DAC '71: Proceedings of the 8th Design Automation Workshop*, pp. 101-108.

Websites:

-Friedman, Y., Anon., 2006. *Utopies Réalisables*. [Online]
Available at: <http://yonafriedman.blogspot.com/2006/11/during-this-first-meeting-we-intervied.html>
[Accessed 1 05 2021].

Bono Cremades, J. A, 2016. *Diseño Arquitectónico Integramente Asistido por Ordenador*. [Online]
Available at: <https://cargocollective.com/jbono/Diseno-Arquitectonico-Integramente-Asistido-por-Ordenador-Trabajo>
[Accessed 03 05 2021].

Crosbie, M. J., 2018. *Doom or Bloom: What Will Artificial Intelligence Mean for Architecture?*. Common Edge. [Online]
Available at: <https://commonedge.org/doom-or-bloom-what-will-artificial-intelligence-mean-for-architecture/>
[Accessed 29 05 2021].

Pertigkiozoglou, E., 2014. Architecture Machine Group MIT [Online]
Available at: <https://medium.com/designscience/1976-852a377855fe>
[Accessed 08 04 2021].

Pertigkiozoglou, E., 2014. 1960, *Christopher Alexander* [Online]
Available at: <https://medium.com/designscience/1976-852a377855fe>
[Accessed 08 04 2021].

Pertigkiozoglou, E., 2014. 1960, *Nicholas Negroponte and Architecture Machine Group MIT* [Online]
Available at: <https://medium.com/designscience/1976-852a377855fe>
[Accessed 08 04 2021].

ORTF, 1969. *Yona Friedman regarding the machine that invents flats*. [Online]
Available at: <https://fresques.ina.fr/europe-des-cultures-en/fiche-media/Europe00061/yonafriedman-regarding-the-machine-that-invents-flats.html>
[Accessed 8 04 2021].

Vlissingen, H. F. v., n.d. *Yona Friedman*. [Online]
Available at: <http://www.yonafriedman.com/>
[Accessed 28 02 2021].

Créditos de las imágenes

Fig. 1.1 Friedman, Y. & Orazi, M. (2015.) *The Dilution of Architecture*.

Fig. 2.1. Friedman, Y. (2011), *Arquitectura con la gente, por la gente, para la gente*.

Fig. 2.2. Negroponte, N. (1975) *Soft Architecture Machines*.

Fig. 2.3. Friedman, Y. & Orazi, M. (2015.) *The Dilution of Architecture*.

Fig. 2.4. Friedman, Y. & Orazi, M. (2015.) *The Dilution of Architecture*.

Fig. 2.5. Friedman, Y. & Orazi, M. (2015.) *The Dilution of Architecture*.

Fig. 3.1. Elaboración propia. Redibujado de *Hacia una arquitectura científica*.

Fig.3.2. Elaboración propia. Redibujado de *Hacia una arquitectura científica*.

Fig.3.3. Elaboración propia. Redibujado de *Hacia una arquitectura científica*.

Fig.3.4. Elaboración propia. Redibujado de *Hacia una arquitectura científica*.

Fig.3.5. Friedman, Y. (1971), *Hacia una arquitectura científica*.

Fig.3.6. Elaboración propia. Redibujado de *Hacia una arquitectura científica*.

Fig.3.7. Friedman, Y. (2011), *Pro domo*.

Fig.3.8. Friedman, Y. (2011), *Pro domo*.

Fig.3.9. Friedman, Y. (1971), *Hacia una arquitectura científica*.

Fig.3.10. Friedman, Y. (1971) *Chose by computer. Progressive Architecture*.

Fig.3.11. Negroponte, N. (1975) *Soft Architecture Machines*.

Fig.3.12. Friedman, Y. & Orazi, M. (2015.) *The Dilution of Architecture*.

Fig.3.13. Negroponte, N. & Weinzaepfel, G. (1976) *Architecture-by-yourself: An Experiment with Computer Graphics for House Design*.

Fig.3.14. Negroponte, N. & Weinzaepfel, G. (1976) *Architecture-by-yourself: An Experiment with Computer Graphics for House Design*.

Fig. 3.15. Elaboración propia. Redibujado de *Soft Architectures*.

Fig. 3.16. Elaboración propia.

Fig. 4.1. Elaboración propia.

Fig. 4.2. Elaboración propia.

Fig. 4.3. Elaboración propia.

Fig. 5.1. Elaboración propia.

Fig. 5.2. Elaboración propia.

Fig. 5.3. Elaboración propia.

Fig. 5.4. Elaboración propia.

Fig. 5.5. Arriba, Negroponte, N. (1975) *Soft Architecture Machines*.

Abajo, elaboración propia

Fig. 5.6. Elaboración propia.

Fig. 5.7. Elaboración propia.

Fig. 5.8. Elaboración propia.

Fig. 5.9. Elaboración propia.

Fig. 5.10. Friedman, Y. (2011), *Pro domo*.

Fig. 5.11. Elaboración propia.

Fig. 5.12. Elaboración propia.

Fig. 5.13. Elaboración propia.

Fig. 5.14. Elaboración propia.

Fig. 5.15. Elaboración propia.

Fig. 5.16. Elaboración propia.

Fig. 5.17. Elaboración propia.

Fig. 5.18. Elaboración propia.

Fig. 5.19. Elaboración propia.

Fig. 5.20. Elaboración propia.

Fig. 5.21. Elaboración propia.

Fig. 5.22. Elaboración propia.

Fig. 7.1. Friedman, Y. (1971), *Hacia una arquitectura científica*.

Imagen portada. Elaboración propia

Imagen nota biográfica.

9. ANEXO: DEFINICIONES

Términos utilizados

1. Urbanismo móvil: término desarrollado a través de arquitectura móvil, definida en 1957 por Friedman en “*L’architecture mobile*” (1956) como cualquier solución que permitiera a los usuarios tomar una decisión directa y transformar ellos mismos directamente su entorno en el momento que decidan revisar su decisión anterior.

2. Urbanismo espacial: término definido en 1962 por Yona Friedman en “*Ten principles of Space town planning*” (Conrads, 1971, 1983) en el que Friedman enuncia los principios para la planificación de un nuevo tipo de ciudad que permita la construcción de barrios tanto juxtapuestos como superpuestos.

3. Infraestructura: término acuñado por Friedman que hace referencia a marcos estructurales vacíos que pueden ser utilizados de diversas maneras y que contiene en su interior todas las instalaciones necesarias para que fuera habitable y garantizará un sistema indeterminado de libre elección dentro de la misma

4. Ville Spatiale: Una Ville Spatiale es un esqueleto estructural multicapa (rejilla) sobre pilotes que puede ajustarse de forma flexible cuando se desee. La estructura se apoya en columnas situadas a un intervalo de 40-60 metros y que albergan los accesos y las redes de instalaciones. La base de la retícula es un módulo de 6×6 metros que puede albergar todo tipo de funciones. (<http://www.yonafriedman.nl/>)

5. Hardware. Friedman define con este término la infraestructura material: “todo lo que está dentro de la infraestructura (y por tanto todo el proceso de abrir o colocar espacios) debe ser reversible, el hardware debe ser cambiabile-solo la infraestructura puede ser rígida y colocada en el sitio”(Friedman, 1971, 66)

1. Hardware: conjunto de aparatos de una computadora (<https://dle.rae.es/>)

6. Software: se refiere a las partes abstractas que no se pueden contabilizar, la infraestructura inmaterial del proceso o la puesta en práctica de un sistema.

1. Software: Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora. (<https://dle.rae.es/>)

7. Axioma: 1. Proposición tan clara y evidente que se admite sin demostración.

2. Cada uno de los principios indemostrables sobre los que, por medio de un razonamiento deductivo, se construye una teoría. (<https://dle.rae.es/>)

8. Mapeado: 1. Localizar y representar gráficamente la distribución relativa de las partes de un todo; como los genes en los cromosomas.

3. Trasladar a un mapa sistemas o estructuras conceptuales. (<https://dle.rae.es/>)

9. Grafo: 2. Diagrama que representa mediante puntos y líneas las relaciones entre pares de elementos y que se usa para resolver problemas lógicos, topológicos y de cálculo combinatorio. (<https://dle.rae.es/>)

10. Etiqueta: representa una especialización, una diferencia “la etiqueta en su capacidad de signo de diferenciación, puede indicar categorías totalmente heterogéneas (..) en las cuales un arquitecto está llamado a trabajar” (Friedman, 1971, p.50)

11. Puntos de cruce. Punto o lugar de paso de un lado a otro. (<https://www.lexico.com/>)

12. Topología: trata las propiedades de las figuras atendiendo a la adyacencia (continuidad) de cada uno de los espacios relacionados, con independencia de su posición en el grafo, tamaño o forma. Dos figuras son topológicamente distintas si una no se puede convertir en otra sin distorsión de sus uniones. (como si se tratase de una estructura tridimensional).

13. Repertorio: “lista completa de los posibles esquemas de uniones, o conjuntos, (...) definidos por n-grafos planos conexos y etiquetados” (Friedman, 1971, 50)

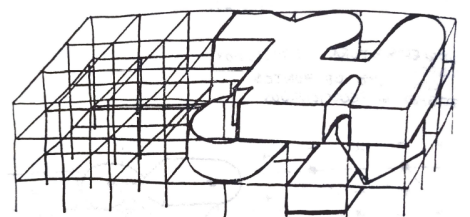
14. Matriz de caminos: muestra la accesibilidad relativa entre los distintos espacios de un grafo suponiendo que la longitud entre los espacios es la misma. Cuando las distancias son consecutivas, se representará con una unidad, si para acceder a ese espacio hay que atravesar otro, con un dos, y así sucesivamente.

15. Matriz de esfuerzos: muestra el esfuerzo relativo a la accesibilidad y utilización de cada espacio. se obtiene al multiplicar la distancia d (representada en la matriz de caminos) por la frecuencia de utilización w (elegida por el usuario)

16. Valor de esfuerzo local: mide el grado de privilegio de situación de un Sespacio cerrado perteneciente al interior de un esquema de conjunto utilizado de una manera particular y concreta. Se obtiene del sumatorio los términos de la fila de matriz de la matriz de esfuerzos ($d * w$)

17. Diagrama de esfuerzos: advierte del nivel de eficiencia del conjunto a nivel urbano. Se obtiene de los valores de las líneas de isoesfuerzos.

18. Sistema de advertencias: permite informar al futuro usuario y al resto de habitantes de la Ville Spatiale sobre las consecuencias implícitas que puede acarrear la toma de ciertas tanto a su modo personal de utilización, como para el resto de habitantes.



LA INFRAESTRUCTURA
VACIA

ALGUNOS PLANOS ARBITRARIAMENTE
ESCOGIDOS
REALIZADOS
EN LA INFRAESTRUCTURA

