



Instituto Politécnico Nacional

ESCUELA SUPERIOR DE CÓMPUTO

DINÁMICA ESPACIAL DE LA EVOLUCIÓN DEL COVID-19 EN MÉXICO

Autor:

Juan Carlos García Medina

PROFESOR:

Dr. Genaro Juárez Martínez

MATERIA:

Sistemas Complejos

, 20 de junio de 2020



Índice general

1. Introducción	1
1.1. COVID-19	1
2. Marco teórico	2
2.1. Autómatas celulares en Grafos	2
2.2. El modelo matemático SIR	3
3. Desarrollo	4
4. Resultados	9
Bibliografía	13

Capítulo 1

Introducción

Los coronavirus son una extensa familia de virus que pueden causar enfermedades tanto en animales como en humanos. En los humanos, se sabe que varios coronavirus causan enfermedades respiratorias que pueden ir desde el resfriado común hasta enfermedades más graves como el síndrome respiratorio de oriente medio (MERS) y el síndrome respiratorio agudo severo (SRAS). El coronavirus que se ha descubierto más recientemente causa la enfermedad COVID-19.

1.1. COVID-19

La COVID-19 es una enfermedad infecciosa causada por el coronavirus que se ha descubierto más recientemente. Tanto este nuevo virus como la enfermedad que provoca eran desconocidos antes de que estallara el brote de Wuhan (China) en diciembre de 2019. Actualmente la COVID-19 es una pandemia que afecta a muchos países de todo el mundo.

Una persona puede contraer COVID-19 por contacto con otra persona que esté infectada por el virus. La enfermedad se propaga principalmente de persona a persona a través de las gotículas que salen despedidas de la nariz o la boca de una persona infectada al toser, estornudar o hablar. Estas gotículas son relativamente pesadas, no llegan muy lejos y caen rápidamente al suelo. Una persona puede contraer la COVID-19 si inhala las gotículas procedentes de una persona infectada por el virus. Por eso es importante mantenerse al menos a un metro de distancia de los demás. Estas gotículas pueden caer sobre los objetos y superficies que rodean a la persona, como mesas, pomos y barandillas, de modo que otras personas pueden infectarse si tocan esos objetos o superficies y luego se tocan los ojos, la nariz o la boca. [1]

Entender el comportamiento de la pandemia permite a las entidades de salud mundiales y a los gobiernos a tomar decisiones que permitan reducir resultados no deseables como el colapso de hospitales o la muerte masiva simultánea de seres humanos, para ello se han desarrollado diversos modelos tales como simuladores de la propagación del virus y para el desarrollo de este proyecto se propone un autómata celular basado en teoría de grafos utilizando un modelo SIR que se explica en la siguiente sección.

Capítulo 2

Marco teórico

2.1. Autómatas celulares en Grafos

Un grafo G es un par (V, E) donde $V = \{v_1, v_2, v_3, \dots, v_n\}$ es un conjunto ordenado no vacío de elementos llamados nodos (o vértices) y E es una familia finita de pares de elementos de V llamadas aristas. Dos nodos del grafo, $v_i, v_j \in V$ se denominan adyacentes (o vecinos) si existe una arista en E de la forma (v_i, v_j) . Se consideran grafos no dirigidos i. e. $(v_i, v_j) = (v_j, v_i) \in E$. Un grafo G es llamado simple si no hay dos aristas de G con el mismo final y no tiene ciclos i. e. las aristas que inician y terminan en el mismo nodo. [2]

Si $V = \{v_1, \dots, v_n\}$, la matriz de adyacencia de G es la matriz de $n \times n$, $A = (a_{ij})$, donde:

$$a_{ij} = \begin{cases} 1 & \text{si } (v_i, v_j) \in E \\ 0 & \text{si } (v_i, v_j) \notin E \end{cases} \quad (2.1)$$

Para este proyecto se utilizaron grafos no dirigidos por lo que la matriz de adyacencia es simétrica. La vecindad de un nodo $v \in V$, N_v es el conjunto de todos los nodos de G que son adyacentes a v , es decir, $N_v = \{u \in V \mid (v, u) \in E\}$. El grado de un nodo v , d_v , es el número de sus vecinos. Un autómata celular en un grafo no dirigido $G = (V, E)$ es una 4-tupla $A = (V, S, N, f)$. El conjunto V define el espacio de células de el CA tal que cada nodo representa una célula del autómata celular. S es el conjunto finito de estados que pueden ser asumidos por los nodos en cada paso de tiempo. El estado del nodo V en un paso de tiempo t está denotado por $s_v^t \in S$ y cambia de acuerdo a la función de transición local f . N es la función de vecindad la cuál asigna a cada nodo su vecindad, es decir:

$$N : V \longrightarrow 2^V \quad (2.2)$$

$$v_i \mapsto N(v_i) = N_{v_i} = \{v_{i_1}, v_{i_2}, \dots, v_{i_{d_v}}\} \quad (2.3)$$

Finalmente la función de transición local f calcula el estado de cada nodo en un tiempo particular $t + 1$ de los estados de sus vecinos en el paso de tiempo previo t , es decir:

$$s_v^{t+1} = f(s_{v_{i_1}}^t, s_{v_{i_2}}^t, \dots, s_{v_{d_v}}^t) \quad (2.4)$$

donde $N_v = \{v_{i_1}, v_{i_2}, \dots, v_{i_n}\}$.

2.2. El modelo matemático SIR

En el modelo matemático epidemiológico SIR la población se divide en tres clases: Los susceptibles a la enfermedad, los infectados y los que se han recuperado y son inmunes a la enfermedad [3]. Más aún, la población se localiza en centros de ciudad que representan los nodos del grafo G . Si hay un tipo de conexión de transporte entre dos de estas ciudades, los nodos asociados se conectan mediante una arista. Se asume lo siguiente:

1. La población de cada nodo permanece constante en el tiempo, es decir, sin nacimientos ni muertes se toman en cuenta. Más aún, la distribución de la población es no homogénea, donde P_u es el número de individuos de el nodo $u \in V$ y $P = \max\{P_u, u \in V\}$.
2. La transmisión de la enfermedad es a través de contacto directo físico.
3. La población es capaz de moverse de su nodo a otro y regresar al nodo origen en cada paso del tiempo.

En un grafo de autómatas celulares, el estado del nodo $u \in V$ en el tiempo t es la tripleta $s_u^t = (S_u^t, I_u^t, R_u^t) \in Q \times Q \times Q$ donde $S_u^t \in [0, 1]$ que representa la fracción de individuos susceptibles del nodo u en el tiempo t , $I_u^t \in [0, 1]$ que representa la fracción de individuos infectados del nodo u en el tiempo t y $R_u^t \in [0, 1]$ que representa la fracción de individuos recuperados del nodo u en el tiempo t [4]. Consecuentemente, la función de transición del CA tiene la siguiente forma:

$$s_u^t = f(s_{v_1}^{t-1}, \dots, s_{v_{d_u}}^{t-1}) = (S_u^t, I_u^t, R_u^t) \quad (2.5)$$

El terreno donde se propaga la epidemia se modela como un grafo ponderado donde cada nodo representa una ciudad o un pueblo y el arco entre dos nodos representa la conexión entre las ciudades correspondientes. En este sentido, el factor de conexión entre los nodos u y v es el peso asociado a la arista $(u, v) \in E$ y es denotado por W_{uv} . Depende de la capacidad de transporte.

$$W_{uv} = \frac{h_{uv}}{\max\{h_{xy} \forall x, y \in V\}} \in [0, 1] \quad (2.6)$$

donde h_{uv} es la cantidad total de población que se mueve de u a v durante un paso de tiempo.

Capítulo 3

Desarrollo

Tomando la teoría de autómatas sobre grafos computacionales como base en el desarrollo se consideran las entidades federativas del país como nodos del grafo y para formar la lista de adyacencia se considera la adyacencia geográfica, es decir, para el caso de Michoacán, se consideran los estados con los que colinda: Colima, Jalisco, Guanajuato, Estado de México y Guerrero, para hacerlo web se desarrolló en el lenguaje javascript y para mostrar un mapa interactivo de los estados se utilizó una API de google llamada Google Charts.

El grafo se forma a partir de la lista de aristas y con esta se construye la lista de adyacencia, esto para tener en todo momento la información de manera directa de los vecinos de un nodo v , el identificador del nodo se mapea para que cada estado tenga uno único.

A pesar de que el modelo que se considera en [4] considera a cada nodo como un grupo de personas, de momento no se consideran datos reales sobre la población de cada entidad, sin embargo es algo que se puede y debe considerar para modelar correctamente la propagación de la pandemia.

Click aquí para ver una versión actualizada del proyecto.

```
1 class State {
2     constructor(I, S, R) {
3         this.I = I;
4         this.S = S;
5         this.R = R;
6     }
7 };
8
9 class Node {
10     constructor(index, s, p) {
11         this.index = index;
12         this.s = s;
13     }
14 }
```

```

14 };
15 //edgeList = {(Weight, Node, Node)};
16 class Graph {
17     constructor(edgeList, n) {
18         this.edgeList = edgeList;
19         this.adjList = {};
20         V = new map();
21         for(var i = 0; i < edgeList.length; i++) {
22             u = edgeList[i][1][0];
23             v = edgeList[i][1][1];
24             w = edgeList[i][0];
25             V.set(u.index, u);
26             V.set(v.index, v);
27             this.adjList[u.index].push([v.index, w]);
28             this.adjList[v.index].push([u.index, w]);
29         }
30     }
31     getNeighbours(u) {
32
33         return this.adjList[u];
34     }
35     getP() {
36         P = 0
37         for (const [key, value] of this.V.entries()) {
38             P = max(P, this.adjList[key].p); // console.log(key, value);
39         }
40         return P;
41     }
42 };
43 class CellularAutomaton {
44     //G Graph class
45     constructor(G) {
46         this.G = G;
47         this.V = G.V;
48

```

```
49     this.N = new map();  
50     for(var i = 0; i < this.V.length; i++) {  
51         N.set(i, this.adjList[i]);
```

```
1 states = [  
2     ['State', 'Casos'],  
3     ['Baja California', 50000],  
4     ['Sonora', 50000],  
5     ['Chihuahua', 50000],  
6     ['Coahuila', 50000],  
7     ['Nuevo León', 50000],  
8     ['Tamaulipas', 50000],  
9     ['Sinaloa', 50000],  
10    ['Nayarit', 50000],  
11    ['Durango', 0],  
12    ['Zacatecas', 400],  
13    ['Jalisco', 30000],  
14    ['Colima', 30000],  
15    ['Tlaxcala', 30000],  
16    ['Aguascalientes', 30000],  
17    ['Zacatecas', 30000],  
18    ['San Luis Potosí', 30000],  
19    ['Puebla', 400],  
20    ['Guanajuato', 400],  
21    ['Querétaro', 400],  
22    ['Hidalgo', 400],  
23    ['Morelos', 400],  
24    ['Estado de México', 400],  
25    ['Distrito Federal', 400],  
26    ['Michoacán', 0],  
27    ['Baja California Sur', 200],  
28    ['Guerrero', 500],  
29    ['Oaxaca', 10000],  
30    ['Veracruz', 10000],  
31    ['Tabasco', 10000],  
32    ['Campeche', 500],  
33    ['Chiapas', 200],  
34    ['Quintana Roo', 500],  
35    ['Yucatán', 500]
```

```
36 ];
37
38 class StatesDictionary {
39     constructor() {
40         this.dict = {};
41         for(var i = 1; i < states.length; i++) {
42             this.dict[states[i][0]] = i-1;
43         }
44     }
45     indexState(name) {
46         return this.dict[name];
47     }
48 };
49 statesDict = new StatesDictionary();
```

Capítulo 4

Resultados

Como se mencionó anteriormente, no se consideraron datos reales sobre la población por entidades y específicamente fueron valores aleatorios cercanos al promedio por entidad, esto debido a la falta de tiempo para extraer los datos de una fuente confiable y que como primer paso se quería experimentar de esta manera ya que al ser un modelo SIR, este no toma en cuenta muertes ni nacimientos lo cual aunque en conjunto parece estar lejos de un modelo real es un buen inicio y que una vez que se corrobore un comportamiento adecuado con estos datos se pueda crecer el modelo con datos reales y al agregar más variables.

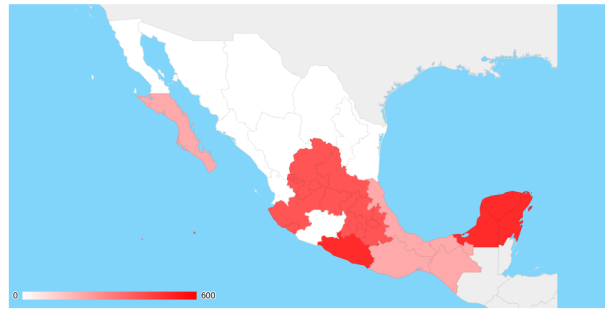


Figura 4.1: Avance del virus extendido en el sur del país

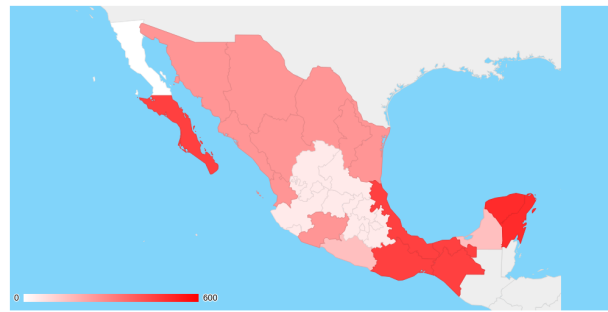


Figura 4.2: El virus prácticamente ha alcanzado todo el territorio



Figura 4.3: En otro tiempo se empieza a concentrar en el bajío del país



Figura 4.4: La aparición de casos nuevos disminuye y en algunos estados no se presentan nuevos casos.

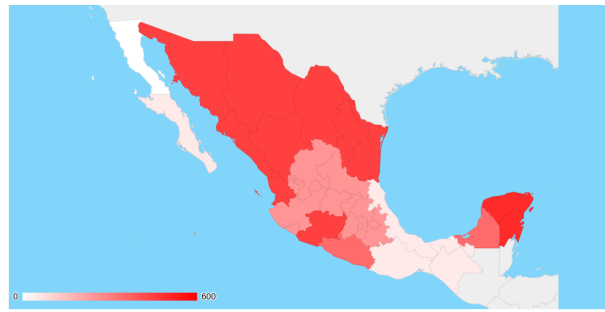


Figura 4.5: Hay una aparente uniformidad de nuevos casos en el país.



Figura 4.6: El virus aún no se controla y tiene un patrón de crecimiento agresivo en el norte lo que indica un rebrote.

Ver Autómata en línea

Para modelar el virus de manera adecuada y tener herramientas sólidas de comparación es necesario trabajar en lo siguiente.

- Cargar datos reales de la población en el país
- Extender el paradigma a otros modelos como SIRS o SEIR
- Elegir los factores que mejor se ajusten a la propagación de la pandemia real
- Generar gráficas en tiempo real para comparar los casos confirmados tanto acumulados como nuevos por día con las gráficas que presentan las autoridades.

Bibliografía

- [1] “Preguntas y respuestas sobre la enfermedad por coronavirus (covid-19).” <https://www.who.int/es/emergencies/diseases/novel-coronavirus-2019/advice-for-public/q-a-coronaviruses>, 2020. 1
- [2] “Graph theory.” https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-fall-2010/readings/MIT6_042JF10_chap05.pdf, 2010. 2
- [3] M. Batista, “Estimation of the final size of the coronavirus epidemic by the sir model,” 02 2020. 3
- [4] M. J. F. Martínez, E. G. Merino, E. G. Sánchez, J. E. G. Sánchez, A. M. del Rey, and G. R. Sánchez, “A graph cellular automata model to study the spreading of an infectious disease,” in *Advances in Artificial Intelligence* (I. Batyrshin and M. González Mendoza, eds.), (Berlin, Heidelberg), pp. 458–468, Springer Berlin Heidelberg, 2013. 3, 4