SIMULACIÓN PARA EL ACCESO A LA PAZ COMO UN BIEN PÚBLICO PURO A LA LUZ DE LA TEORÍA DE JUEGOS

AUTORES
MARÍA CATALINA SAAVEDRA LLOREDA
SANTIAGO MOSQUERA DAZA

DIRECTORES DEL PROYECTO
NATALIA GONZÁLEZ GÓMEZ
ANDRÉS FELIPE MUÑOZ GÓMEZ

UNIVERSIDAD ICESI
FACULTAD DE CIENCIAS ADMINISTRATIVAS Y ECONÓMICAS
ECONOMÍA Y NEGOCIOS INTERNACIONALES
SANTIAGO DE CALI
2017

Índice general

Agradecimientos

Un especial agradecimiento a Natalia González y Andrés Muñoz por su respaldo académico.

Dedicado con cariño a Natalia González por su excelente labor como maestra.

María Catalina S.Ll y Santiago M.D

Resumen/Abstract

Resumen

El presente trabajo busca analizar la provisión de la paz como un bien público puro, utilizando como herramienta teórica la teoría de juegos y como herramienta metodológica la programación. Como bien público puro, la paz no estará exenta de la presencia de free riders o polizones, los cuales son agentes que buscan disfrutar el consumo del bien en cuestión sin realizar ningún tipo de contribución. Sin embargo, se plantean diversas estrategias materializadas en forma de castigo, que permitirán llevar a que se dé la cooperación entre los agentes.

Palabras claves: Teoría de juegos, Bien público, Paz, free rider, Simulación.

Abstract

The present work seeks to analyze the provision of peace as a pure public good, using the game theory as theoretical tool and programming in R as methodological tool. As a pure public good, peace is not exempt from the presence of free riders, who are the agents who seek to enjoy the consumption of the good in question without making any contribution. However, various strategies materialize in the form of punishment, will lead to cooperation between agents.

Key words: Game theory, Public Good, Peace, Free rider, Simulation.

Introducción

Antes de comenzar, es indispensable especificar el concepto de paz que usaremos a lo largo del proyecto. Definida en sentido positivo representa un estado, tanto a nivel social como personal, en el cual las partes de una unidad logran alcanzar el equilibrio. Por otro lado, el derecho internacional define un estado de paz como aquel en el cual los conflictos se han resuelto por una vía alternativa a la violencia y en consecuencia se denomina ”paz” al acuerdo o tratado que pone fin al conflicto.

Bajo este orden de ideas, se parte del hecho de considerar a la paz como un consenso que se da entre los actores enmarcados en el contrato social. Sin embargo, es indispensable desligarse de aspectos que sesgan el concepto, tales como la ”ausencia de guerra”, debido a que, en ocasiones, dicho consenso se materializa posteriormente a un proceso bélico. De esta forma, entiéndase la paz no sólo como acuerdo, equilibrio, estabilidad y concilio, sino también como aquél fin donde las partes gozan de mejores garantías de tal forma que las mismas no posean en sus deseos la discordia del bienestar ajeno dado el propio. Ahora bien, para efectos prácticos, la aplicación de la paz como bien público la definiremos como aquel estado en el que se mejoran las oportunidades para todos los individuos de una sociedad, en temas relacionados con la educación, la salud, la seguridad, la calidad de vida y el acceso a oportunidades en general. Por lo tanto, el costo de contribuir está determinado precisamente por los requerimientos, bien sea de transporte, de materiales, de construcción de instalaciones, entre otros.

Ahora bien, ya habiéndose mencionado el concepto de paz, se toma a éste como bien público, por sus dos propiedades de ser no excluyente y no rival; lo anterior, debido a que se somete a una situación bajo la cual distintos agentes acceden a él, ya sea como polizones o como contribuyentes, creando un escenario de decisiones estratégicas donde las interacciones entre los individuos determinarán el nivel de las contribuciones y donde las reglas de juego influyen en la asignación eficiente del bien según los aportes de los agentes en cada una de las etapas jugadas.

El fenómeno se modela en un juego repetido con información completa pero imperfecta, donde inicialmente interactúan dos grupos representativos de ciudadanos de manera simultánea, quienes deciden cuánto aportar dado un costo de provisión por hacerlo y su nivel de paciencia. En el periodo siguiente, los jugadores observan las decisiones de la etapa anterior y el pago obtenido por dicha decisión, para elegir un nuevo nivel de contribución y así de manera repetida. Posteriormente, se realizarán nuevas simulaciones cada una con diferente cantidad de grupos de ciudadanos, esto, con el objetivo de acercar el análisis a la realidad, como también estudiar el comportamiento del equilibrio del juego cuando aumentan los participantes.

Es de rescatar que para todo agente, las estrategias del juego planteado son *Contribuir* o *No contribuir*, con lo que bajo un terreno competitivo, se esperaría un dilema del prisionero en el cual el equilibrio representa la alternativa menos favorable de forma conjunta, aun cuando la alternativa de cooperar es la más beneficioso a nivel social. Dicha situación podrá ser superada mediante la negociación y la aplicación de castigos en las etapas superiores del juego en su versión dinámica. En este punto, analizar estrategias desoladoras o Trigger Strategies resulta conveniente para el análisis. Por último, asumiendo el comportamiento de contribución de los agentes, las funciones de pago de los mismos serán determinadas a partir de sus contribuciones y los costos incurridos para ello, los cuales representan las variables exógenas del modelo.

Metodología

Iniciar una simulación requiere de una idea previa hacia aquello, a lo cual, se quiere llegar. Bajo este criterio, los objetivos fueron los siguientes:

- **Objetivo General**: Simular en R un problema *ex-ante* de acceso a la paz como un bien público puro y evidenciar lo que la Teoría de Juegos predice acerca de este.
- **Objetivos Específicos**:
  - Analizar los Equilibrios de Nash (E.N) con agentes homogéneos y heterogéneos.
  - Evidenciar los diferentes equilibrios obtenidos acorde a las estrategias llevadas por los jugadores.
  - Proponer estrategias para lograr el óptimo social.

De manera general, la metodología constó de recopilación bibliográfica, modelo, programación y resultados, ámbitos que serán explicados en breve.

Recopilación Bibliográfica

En este ámbito, se realizó una exhaustiva revisión bibliográfica en materia de Teoría de Juegos, Economía Política y Programación en R. Entre aquella revisión se halla el texto guía de Sebastian Kranz *Interactively Solving Repeated Games: A Toolbox*, paper cumbre de la ejecución en programación del presente trabajo, dado su respaldo en la construcción de los códigos que se encontraban disponibles en su trabajo y en GitHub. Asimismo, se realizó la lectura de autores como (Abreu,1988), quien dio un componente teórico adicional al propuesto por (Gibbons,1992) en su texto *Un Primer Curso de Teoría de Juegos*, además de (Downs, 1957) cuyo gran aporte fue en materia de Economía Política, e incluso (Friedman, 2014) cuyo paper fue una introducción a lo que es una simulación bajo el Modelo de Hotelling, el cual no se hallará en el presente escrito, pero que tuvo gran incidencia en cuanto a la decisión del tema a tratar y de cómo abordarlo.

Modelo y Programación

Paralelo a la revisión bibliográfica, se llevaba a cabo el acercamiento a los programas R y LaTeX mediante talleres guiados por los tutores, los cuales fueron vitales en el proceso de aprendizaje y asimilación de ambos softwares y del cómo emplearlos para la construcción de la idea a investigar.

Una vez cubiertos dichos talleres, el aprendizaje en programación se llevaba a cabo de forma independiente bajo el texto guía ya mencionado, aunque con el objetivo de construir el modelo, el cual se tratará de una forma más profunda en el siguiente capítulo. No obstante, un breve adelanto podrá denotar que se llevó a cabo la construcción de las funciones de utilidad de los agentes y los supuestos del modelo, además de la forma en la que se codificarían.

De hecho, los paquetes empleados en R fueron *devtools, glpkAPI,lattice, Rcpp, rgmpl, skUtils, slam, stringr,repgame, restorepoint y rowmins.* Para el proceso de instalación de cada uno de ellos, se respaldó en el texto guía y en la plataforma `https://github.com`, adicional a las guías propias del lenguaje de programación R.

Resultados

Una vez cubiertos los pasos anteriores, se recolectó la información obtenida por la simulación y se realizó un análisis detallado de la misma. Estos podrán observarse en Anexos, donde yace el Script del trabajo y podrá percatarse con detalle el motivo de los códigos y la forma en que se abordó el problema, además de lo que se dispone en el capítulo Resultados.

Modelo y Dinámica

MODELO

Suponga una Economía compuesta por dos tipos de agentes, algunos que cooperando reciben un beneficio y otros que deciden no contribuir y aun así desean recibir beneficios (en adelante, denominado *polizón* o *free rider*). Las estrategias para cada agente son, por ende, contribuir o no hacerlo para la paz como un bien público. En este punto conviene denotar la siguiente información:

- Por simplicidad, la economía se divide en grupos homogéneos de personas, los cuales tienen como unidad en común las estrategias y las funciones de utilidad. Entre los n-grupos puede existir la misma función de utilidad[1], en cuyo caso habrá homogeneidad entre los grupos, o en caso contrario, cuando se difiera, se tratará de heterogeneidad entre los grupos en cuestión.

  Es importante manifestar que la homogeneidad o no **entre** los grupos se determina a través de la forma funcional bajo criterios por parámetros, como la importancia que tiene el beneficio por la paz a la hora de contribuir para un determinado grupo de personas. Se va a suponer que aquellos que reciban mayor utilidad a la construcción del bien público son los más afectados por la guerra.

- Los agentes pueden diferir o no en los costos de contribución. Dichos costos son aquello a lo cual los agentes deben renunciar para contribuir, o simplemente la erogación por contribución. Por lo tanto, dichos costos contienen, por ejemplo, el IVA de los productos adquiridos para construir escuelas, mejorar la educación y garantizar paz; o también el coste de transporte y otros que se perciben al momento de mejorar el sistema agrícola y en general los asuntos de tierras como mecanismos que sirvan como políticas de distribución que lleven a incentivar la paz.

- Lo anterior implica el supuesto pertinente de la endogenización de los procesos o mecanismos para acceder a la paz vía contribuciones sin la intervención de algún planificador central. Por lo tanto, la decisión de maximizar el bienestar social depende únicamente de los agentes que interactúen entre sí y decidan los montos que contribuirán para financiar el bien público.

  Por consiguiente, la dinámica de los agentes es la que determina los equi-

[1] Dichas funciones cumplen las condiciones de Inada y difieren entre polizones y no polizones, las cuales se construirán más adelante.

librios que los individuos alcancen acorde a las estrategias que los mismos decidan llevar a cabo. La siguiente sección tratará sobre ello.

DEFINICIÓN Y DINÁMICA DEL JUEGO

Sobre las funciones de utilidad de los agentes

Para efectos de análisis, suscita el interés por conocer las decisiones de los agentes con base a las reglas del juego y a sus funciones de pagos. Se considerarán dos perfiles de funciones, a saber, una en la cual se puede ser polizón y otra en la cual no hay la oportunidad para serlo. Bajo este contexto, las funciones son las siguientes:

- Con posibilidad de Free-rider

$$g_i^f = \frac{\sum_{i=1}^{n} x_i}{n} - k_i x_i \tag{1}$$

- Sin posibilidad de Free-rider

$$g_i^{\neg f} = \sum_{i=1}^{n} ln(x_i)^n - k_i x_i \tag{2}$$

donde g_i representa la utilidad del agente i, mientras x_i y k_i son las contribuciones y el costo de hacerlo, respectivamente.

Cabe destacar, que el encuentro que se da entre los agentes para acceder al bien público es antes de la creación del mismo. Es decir, es *ex-ante* a la provisión. Lo anterior permite que la función que describe la no posibilidad de Free-rider sea tal, que al no decidirse una provisión del agente i, el agente j obtenga una pérdida por el costo incurrido de decidir contribuir. O, ante un suceso extremo, en el cual ambos no decidan contribuir, la utilidad de ambos es nula. Con todo, se sabe que la función de utilidad $u_i^{\neg f}$ ante el caso de una no contribución de alguno de los agentes, o ambos, será una tal que $u_i^{\neg f} \leq 0$.

Lo previamente dicho es necesario ya que la naturaleza de la función (2) viola los principios de un bien público, es decir, exclusión y no rivalidad.

Juego en dos etapas

Juego en dos etapas con información completa pero imperfecta, donde el aspecto de información imperfecta radica en el hecho de que los jugadores toman las decisiones de manera simultánea en cada etapa:

$$G = \langle N, X_i, g_i \rangle$$

donde $X_i = (x_1, x_2, ... x_N)$ representa el set de acciones de los agentes y N el número de jugadores.

Ahora bien, para el juego de 2 grupos de ciudadanos en dos etapas:

1. El grupo de ciudadanos 1 y 2 escogen simultáneamente las acciones x_1 y x_2 de los conjuntos de acciones factibles X_1 y X_2, respectivamente, las cuales son contribuciones.

2. Nuevamente los grupos 1 y 2 observan el resultado de la primera etapa, (x_1, x_2), y toman simultáneamente la decisión sobre las acciones x_1 y x_2 de los conjuntos factibles X_1 y X_2, respectivamente.

3. Los conjuntos factibles X_1 y X_2 son iguales y están dados por las estrategias de Cooperar y No Cooperar, que en este contexto sería ”proveer el bien público” y ”no proveer” a través de las contribuciones x_i.

4. Las ganancias son $g_i(x_1, x_2)|\forall_i i = 1, 2$.

5. Es decir que para cada resultado factible (x_1, x_2) de la primea etapa, el juego de la segunda tiene un único equilibrio de Nash denominado por $(x_1^*(x_1, x_2), x_2^*(x_1, x_2))$.

6. Resultado esperado: El E.N se dará en (No cooperar, No cooperar) lo cual representa un dilema del prisionero.

Juego Dinámico Infinito

$$G = \langle \infty, \langle N, X_i, g_i, \delta \rangle_t \rangle$$

Para el juego dinámico de dos jugadores con estrategia del disparador, δ representa la Tasa de descuento y g_i, traído a valor presente, como $VP(g_i)$:

Así que se jugará ”Cooperar” en la primera etapa. En la t-ésima etapa, si el resultado de todas las t-1 etapas anteriores ha sido (Cooperar, Cooperar) entonces jugar ”Cooperar” en caso contrario, jugar ”No Cooperar” donde $G(\infty)$ representa el juego repetido infinitamente y $G(\infty, \delta)$ es el juego en su versión infinita con la versión formalizada expresada previamente.

Resultado esperado: Si ambos jugadores adoptan la trigger strategy, el resultado del juego repetido infinitamente será (Cooperar, Cooperar) en cada etapa.

Tipos o alternativas de Trigger strategies

GRIM TRIGGER

El castigo continúa de forma indefinida luego del desvío de uno de los jugadores. La dinámica es la siguiente:

1. En el primer período, el jugador i comienza con una estrategia cooperativa o de aporte para el bien público paz.

2. En caso que en el resto de las etapas uno de los agentes se desvíe de la estrategia, es decir, que haya un agente j que decida no aportar para el bien en cuestión, el agente i, quien primerio decidió jugar de forma cooperativa, decidirá cortar su contribución y seguirá sin aportar por el resto del juego.

Éste se representa de la siguiente manera:

$$\begin{aligned} s_i(x_1,...,x_T) = C\, si\, x_t &\neq (\neg C, \neg C)\forall_t : t \in [1,T] \\ s_i(x_1,...,x_T) &= \neg C\, en\ caso\ contrario. \end{aligned}$$

STICK & CARROT

A diferencia de las estrategias ”Trigger”,usando la estrategia Stick & Carrot, una desviación es seguida de una fase de castigo que dura un número finito de periodos:

- Stick: Implica castigar al jugador que se desvía por un tiempo suficiente.

- Carrot: Proporciona a cada castigador un incentivo para castigar, prometiéndole una ”carrot” una vez que la fase de castigo se ha producido.

DESARROLLO DEL JUEGO

Función de pagos para el juego con dos agentes:

$$g_i = \frac{x_i + x_j}{2} - k_i x_i$$

donde i y j son los agentes

Matriz de pagos:

$$\begin{array}{lcc} & cooperar & no\ cooperar \\ \begin{array}{l} cooperar \\ \\ no\ cooperar \end{array} & \multicolumn{2}{c}{\left(\begin{array}{cc} \frac{x_i+x_j}{2} - k_i x_i \,, \frac{x_i+x_j}{2} - k_j x_j & \frac{x_i}{2} - k_i x_i \,, \frac{x_i}{2} \\ \\ \frac{x_j}{2} \,, \frac{x_j}{2} - k_j x_j & 0 \;,\; 0 \end{array} \right)} \end{array}$$

Juego en dos etapas:

- Etapa 1: Los jugadores eligen simultaneamente sus cantidades a contribuir.
- Etapa 2: Los jugadores reciben los pagos correspondientes a la cantidad provista del bien público.

$$MAX_{x_i}\ g_i = \frac{x_i + x_j}{2} - k_i x_i$$

CPO:

$$\frac{1}{2} - k_i = 0$$

$$k_i = \frac{1}{2}$$

Reemplazando en la función de pagos:

$$\hat{g}_i = \frac{x_i + x_j}{2} - \frac{1}{2} x_i$$

$$\hat{g}_i = \frac{x_j}{2} \rightarrow \hat{x}_i = 0$$

La simetría en las funciones de pagos implica que:

$$\hat{g}_j = \frac{x_i + x_j}{2} - \frac{1}{2}x_j$$
$$\hat{g}_j = \frac{x_i}{2} \rightarrow \hat{x}_j = 0$$

En consecuencia, en el juego de dos etapas es óptimo para los agentes no contribuir para la provisión del bien público, pues actúan como free riders esperando que sea el otro agente quien contribuya en su totalidad. Bajo este contexto, como ambos agentes piensan de la misma manera se obtiene un resultado final con una provisión nula de paz en la sociedad.

Lo anterior, permite concluir que en el juego de etapa es imposible lograr la cooperación para conseguir que ambos agentes esten mejor con una provisión positiva del bien público; es necesario analizar el juego infinito para determinar los mecanismos que conllevan a que se dé la cooperación y las condiciones para que dicha estrategia se mantenga.

Generalizado para n agentes

$$g_i = \frac{\sum_{i=1}^{n} x_i}{n} - k_i x_i$$

$$MAX_{x_i}\ g_i = \frac{(x_i + \sum_{s=1}^{n-1} x_s)}{n} - k_i x_i$$

CPO:

$$\frac{1}{n} - k_i = 0$$
$$k_i = \frac{1}{n}$$

Reemplazando en la función de pagos

$$\hat{g}_i = \frac{1}{n}x_i + \frac{\sum_{s=1}^{n-1} x_s}{n} - \frac{1}{n}x_i$$

$$\hat{g}_i = \frac{\sum_{s=1}^{n-1} x_s}{n}$$

Juego Dinámico infinito

Para solucionar el juego en su versión dinámica se debe realizar un procedimiento distinto al del juego estático de dos etapas, en el cual, lo importante es el encuentro entre las funciones de mejor respuesta de los jugadores. En el juego dinámico infinito se debe proponer una estratégia y ensayar si dicha estratégia corresponde a un equilibrio de Nash en subjuegos perfecto.

Comenzaremos con la estrategia Grim Trigger o del ”disparador”, descrita con anterioridad, donde se define a δ como el factor al cual los individuos descuentan sus ganancias futuras, o dicho de otra manera, la paciencia que tienen los jugadores de obtener ganancias futuras a costa de ganacias presentes.

Bajo este lineamiento, de acuerdo con la matriz de pagos, la estrategia Grim Trigger es óptima si cooperar es más rentable que desviarse, es decir, si contribuir positivamente para la provisión del bien público es más rentable que actuar como free rider en alguno de los periodos, estrategia que conllevaría a que en lo que resta del juego se llegue al equilibrio de Nash del juego de etapa, el cual implica provisiones nulas del bien público bajo ausencia de cooperación.

En conclusión no desviarse o cooperar es más rentable para el jugador i si:

$$\frac{x_j}{2} + 0\delta + 0(\delta)^2 + 0(\delta)^3 + ... + 0(\delta)^T \; < \; V$$

Donde V es la ganancia de cooperar y de seguirlo haciendo en el futuro y despejando corresponde a:

$$V = \frac{x_i + x_j}{2} - k_i x_i + \delta V$$
$$V = \frac{x_i + x_j - 2k_i x_i}{2(1-\delta)}$$

Entonces

$$\frac{x_j}{2} \; < \; \frac{x_i + x_j - 2k_i x_i}{2(1-\delta)}$$

$$x_j - \delta x_j \; < \; x_i + x_j - 2k_i x_i$$

$$\frac{2k_i x_i - x_i}{x_j} \; < \; \delta$$

Donde el valor δ hallado corresponde al mínimo nivel de paciencia necesario para que el agente i decida cooperar y la estratétgia Grim Trigger sea racional. Como ambos individuos son igulaes, dicho δ es el mismo para ambos jugadores

SIN POSIBILIDAD DE HACER FREE RIDING

Si bien la propia definición de un bien público, como un bien no rival y no excluyente, trae ineludiblemente la aparición de agentes que se comportan como free riders o polizones, es factible introducir la posibilidad que los jugadores se vean limitados en su totalidad a comportarse como free riders si se supone una etapa inicial en la que el bien público no ha sido construido y se desea generarlo en la economía.

Para el caso de estudio, se tratará una etapa inicial en la cual, tras un periodo de conflicto tanto social como político, se busca financiar cualquier tipo de proyecto cuya finalidad sea materializar la paz. Ya podrá percatarse, por ende, que puede ser una ley o regla social que advierta a los agentes a contribuir para el bien público paz que se construirá. Bajo este contexto, dichas contribuciones podrían ser, a manera de ejemplo, inversiones para educación y su fomento, mejoras en la agricultura y el ámbito de tierras que han sido raíz de diversos problemas ante cuestiones guerrilleras y campesinos inconformes.

Dicho lo anterior, la **restricción de no polizón** será impuesta dentro de la naturaleza de la función de utilidad con la ayuda del logarítmo, el cual no permite valores nulos en las contribuciones. Por lo tanto, la función de pagos estará dada por la siguiente ecuación:

$$g_i = \sum_{i=1}^{n} log_b(x_i^n) - k_i x_i = \sum_{i=1}^{n} \frac{ln(x_i^n)}{ln(b)} - k_i x_i = \sum_{i=1}^{n} \frac{nln(x_i)}{ln(b)} - k_i x_i$$

Tomando al cociente de la cantidad de agentes y el logaritmo de la base como una constante, queda de la siguiente forma:

$$g_i = \Phi \sum_{i=1}^{n} ln(x_i) - k_i x_i \; con \; \Phi = \Phi(n,b) = \frac{n}{ln(b)}$$

La lógica del juego de dos etapas es la misma que en el caso anterior: En una primera etapa los jugadores eligen simultaneamente las cantidades x de contribución que maximizan sus funciones de pagos. Posteriormente, los agentes reciben los pagos correspondientes a la cantidad total provista del bien público en la economía en la segunda etapa del juego.

$$MAX_{x_i}\ g_i = \Phi \sum_{i=1}^{n} ln(x_i) - k_i x_i \quad = \Phi\left[ln(x_i) + \textstyle\sum_{s=1}^{n-1} ln(x_s)\right] - k_i x_i$$

CPO

$$\frac{\Phi}{x_i} - k_i = 0 \ \rightarrow \ \hat{x}_i = \frac{\Phi}{k_i}$$

Remplazando en la función de pagos

$$\hat{g}_i = \Phi\left[ln(\hat{x}_i) + \sum_{s=1}^{n-1} ln(x_s)\right] - k_i\hat{x}_i \quad = \quad \Phi\left[ln\left(\frac{\Phi}{k_i}\right) + \sum_{s=1}^{n-1} ln(x_s)\right] - k_i\left(\frac{\Phi}{k_i}\right)$$

$$\hat{g}_i = \Phi\left[ln(\Phi) - ln(k_i) + \sum_{s=1}^{n-1} ln(x_s)\right] - \Phi \quad = \quad \Phi\left[ln(\Phi) - ln(k_i) + \sum_{s=1}^{n-1} ln(x_s) - 1\right]$$

Para la construcción de la Matriz de pagos, se deben realizar los cálculos pertinentes para que se muestren las no contribuciones. Bajo este contexto se presentan los siguientes escenarios:

- *El agente i decide no cooperar*:

$$\begin{aligned}
\lim_{x_i \to 0} g_i &= \lim_{x_i \to 0} \Phi \sum_{i=1}^{n} ln(x_i) - \lim_{x_i \to 0} k_i x_i \\
\lim_{x_i \to 0} g_i &= \Phi\left[\lim_{x_i \to 0} \sum_{j=1}^{n-1} ln(x_j) + \lim_{x_i \to 0} ln(x_i)\right] - \lim_{x_i \to 0} k_i x_i \\
\lim_{x_i \to 0} g_i &= \Phi \sum_{j=1}^{n-1} ln(x_j) + \lim_{x_i \to 0} ln(x_i) \\
\lim_{x_i \to 0} g_i &= \Phi \sum_{j=1}^{n-1} ln(x_j) + \mathscr{L} = -\infty \\
Entonces &\quad (g_i \rightarrow -\infty)\,(x_i \rightarrow 0)
\end{aligned}$$

- *Ningún agente desea cooperar* Ante esta situación se puede acudir a la siguiente notación de tal forma que haya dos tipos de agentes, el agente i los que no son el agente i:

$$\lim_{x_i \to 0} \lim_{x_{\neg i} \to 0} g_i = \Phi \left[\lim_{x_i \to 0} ln(x_i) + \lim_{x_{\neg i} \to 0} ln(x_{\neg i}) \right] - \lim_{x_i \to 0} k_i x_i$$
$$Entonces\, (g_i \to -\infty)\, (x_i \wedge x_{\neg i} \to 0)$$

¿Cuál será el conjunto de contribuciones de los agentes cuando existe una restricción de no free-rider?

Dado que la restricción se encuentra activa de dos formas, se asumirá, por tanto, que las mismas partirán desde 1, es decir $x \in [1, 10]$. Dichas maneras son las siguientes:

1. *Endógena:* Por la forma funcional, la cual es logarítmica y no puede tomar valores iguales a 0. Esa es la razón por la cual se calcula con límites.
2. *Exógena:* Dado que todos deben contribuir, se parte que, como mínimo, los agentes deben aportar 1.

Por lo tanto, la función de pagos bajo los escenarios son las que siguen:

- *El agente i decide no cooperar*

$$\bar{g}_i = \Phi \left[\sum_{i=1}^{n-1} ln(x_{\neg i}) + ln(x_i = 1) \right] - k_i(x_i = 1) = \Phi \sum_{i=1}^{n-1} ln(x_{\neg i}) - k_i$$

- *Ningún agente desea cooperar*

$$\tilde{g}_i(x_i = 1) = \Phi \sum_{i=1}^{n} ln(x_i) - k_i x_i = -k_i$$

Matriz de pagos: Por lo tanto, la matriz de pagos para los agentes es la siguiente, donde $\neg C$ es no Cooperar y C lo contrario:

$$\begin{array}{cc} & \begin{array}{cc} C & \neg C \end{array} \\ \begin{array}{c} C \\ \neg C \end{array} & \begin{pmatrix} g_i\,,\, g_{\neg i} & \Phi \sum_{i=1}^{n-1} ln(x_i) - k_i x_i, \Phi \sum_{i=1}^{n-1} ln(x_i) - k_{\neg i} \\ \Phi \sum_{\neg i=1}^{n-1} ln(x_{\neg i}) - k_i, \Phi \sum_{\neg i=1}^{n-1} ln(x_{\neg i}) - k_{\neg i} x_{\neg i} & -k_i \;\;,\;\; -k_{\neg i} \end{pmatrix} \end{array}$$

Juego Dinámico infinito Como se especificó en la sección de Juego dinámico infinito, cuando existe la posibilidad de ser polizón, no desviarse o cooperar es más rentable para el jugador i si:

$$\Phi \sum_{\neg i=1}^{n-1} ln(x_{\neg i}) - k_i \sum_{k=1}^{T} \delta^k < V$$

Donde V es la ganancia de cooperar y de seguirlo haciendo en el futuro y despejando corresponde a:

$$V = \Phi \textstyle\sum_{i=1}^{n} ln(x_i) - k_i x_i + \delta V$$

$$\Rightarrow V = \tfrac{\Phi \sum_{i=1}^{n} ln(x_i) - k_i x_i}{1-\delta}$$

Por ende:

$$\Phi \sum_{\neg i=1}^{n-1} ln(x_{\neg i}) - k_i \sum_{k=1}^{T} \delta^k < \frac{\Phi \sum_{i=1}^{n} ln(x_i) - k_i x_i}{1-\delta}$$

Dada la complejidad que implica hallar el factor de descuento que cumpla con dicha condición, se deja al lector la certeza que dicho delta puede resolverse bajo algoritmos complejos llevados a cabo por el software de programación que se emplea para la simulación. Lo que sí es pertinente es representarlo de la siguiente forma:

$$\Phi \textstyle\sum_{\neg i=1}^{n-1} ln(x_{\neg i}) - k_i \sum_{k=1}^{T} \delta^k < \tfrac{g_i}{1-\delta}$$

$$\Phi \textstyle\sum_{\neg i=1}^{n-1} ln(x_{\neg i}) - k_i \sum_{k=1}^{T} \delta^k - \delta \left[\Phi \sum_{\neg i=1}^{n-1} ln(x_{\neg i}) - k_i \sum_{k=1}^{T} \delta^k\right] < g_i$$

$$\Phi \textstyle\sum_{\neg i=1}^{n-1} ln(x_{\neg i}) - k_i \sum_{k=1}^{T} \delta^k - \delta\Phi \sum_{\neg i=1}^{n-1} ln(x_{\neg i}) + k_i \sum_{k=1}^{T} \delta^{k+1} - g_i < 0$$

Además, se debe tener en cuenta que el proceso algorítmico se desarrolla por etapas, las cuales estarán representadas con T. Bajo este contexto, la dinámica de la simulación hallará un vector de δ tal que cumplan la desigualdad, descartando, desde luego, los valores de δ que hagan parte del conjunto de los números complejos.

HETEROGENEIDAD CON POLIZONES[2]

Ante la presencia de heterogeneidad de agentes, se representará una situación en la cual el agente i tenga un nivel de importancia mayor por la contribución para la paz. La forma en la que se hará dicho proceso es tal que la utilidad de i reciba α veces la utilidad con respecto al otro agente con $\alpha > 1$.

Por simplicidad, además por efectos prácticos, dado que el script posee el caso para dos agentes, se hará de dicha forma, con los agentes 1 y 2. Las formas funcionales son las que siguen:

$$\begin{aligned} g_1 &= \alpha\left[\frac{x_1+x_2}{2} - k_1x_1\right] \\ g_2 &= \frac{x_1+x_2}{2} - k_2x_2 \end{aligned}$$

Por lo tanto, acorde a las Condiciones de Primer Orden, el óptimo se alcanza cuando se cumplen las siguientes condiciones:

$$\frac{\alpha}{2} = k_1 \;\wedge\; \frac{1}{2} = k_2$$

Reemplazando los óptimos:

$$\begin{aligned} \hat{g_1} &= \alpha\left[\frac{x_1+x_2}{2} - \frac{\alpha}{2}x_1\right] \\ \hat{g_2} &= \frac{x_1+x_2}{2} - \frac{1}{2}x_2 = \frac{x_1}{2} \end{aligned}$$

Podrá verse además que $\hat{g_1}$ alcanza su máximo cuando $\alpha = \frac{x_1+x_2}{2x_1}$, es decir que su máximo valor es:

$$\bar{\hat{g_1}} = \frac{x_1+x_2}{2x_1}\left[\frac{x_1+x_2}{2} - \frac{\frac{x_1+x_2}{2x_1}}{2}x_1\right] = \frac{(x_1+x_2)^2}{8x_1}$$

Lo cual es significativo dada la importancia que el agente 1 le atribuye a la paz implica que este no pueda contribuir 0, pese a que exista la posibilidad de ser un free-rider. Esto tiene sentido puesto que un grupo de individuos que haya sido

[2]Se omite el escenario cuando no existe la posibilidad de free-rider, ya que existe una gran similitud en las conclusiones, partiendo de la intuición. Esto porque la restricción de no polizón obliga a todos a contribuir, asunto que no es interesante para la evaluación pues no permite observar qué tan importante es la paz para el agente si tiene la oportunidad de contribuir más de lo que debe o incluso contribuir pudiendo no hacerlo; situación contraria cuando existe la restricción, pues queriendo poco o más la paz con respecto a otros agentes, tendrá que contribuir.

afectado cruelmente por la guerra, deseará la paz y gustaría llegar a ella de cualquier forma, partiendo del hecho que desee hacerlo, motivo por el cual no tendrá incentivos a aportar nada. En el proceso de simulación, se fijó a $\alpha = 2$ para observar la intuición que yace en el material disponible en el anexo y en el capítulo de resultados.

Resultados

Lo que sigue son las gráficas tomadas de la simulación que sirven para ilustrar los resultados. Antes de entrar en materia conviene advertir cierta información que se aplicará en la interpretación de las gráficas.

- Los costos simétricos tratados, en todas las simulaciones, equivalen al 75 % de las contribuciones realizadas, por lo que $k_i = 0{,}75$ cuando se está en el escenario de costos simétricos.

- En cuanto a los costos asimétricos en estática comparativa se tiene lo siguiente:

  - Cuando se trabaja con dos agentes, los mismos equivalen a $k_1 = 0{,}6$ *y* $k_2 = 0{,}75$.
  - Cuando se simula con tres agentes, la decisión estándar fue la siguiente:

$$\begin{aligned} k_1 &= \tilde{k} \\ k_2 &= k_1 + \frac{1}{6} \\ k_3 &= k_2 + \frac{1}{6} \end{aligned}$$

  Donde $\tilde{k}$ tomaba distintos valores, aunque por cuestiones de estandarización, se simulón con $\tilde{k} = \frac{2}{6} = \frac{1}{3}$

- Las gráficas de nivel o *levelplots* trabajan con diferentes secuencias de los costos. Es decir que se vectoriza una secuencia de cambios en k_i a lo largo de su conjunto acotado bajo una tasa de incremento constante a manera de progresión aritmética.

  Bajo este punto es importante manifestar que también poseen dos escenarios: simétricos y asimétricos. Los primeros son a una misma tasa de cambio bajo un punto inicial el común, mientras los segundos parten desde el mismo valor, pero a diferente tasa. [3]

- **En lo referido a la forma:**Las gráficas a continuación poseen entre dos y tres ejes. Bajo el primer escenario, el eje de la ordenada representa la Utilidad Máxima (Ue) y las curvas alcanzarán diferentes niveles de la misma acorde a la estrategia de los agentes, mientras que en la abscisa yace el factor de descuento (Discount Factor), el cual es el δ del Modelo.

  De otro lado, ante una gráfica de tres ejes, la distribución cambia dada la curva de nivel. En el eje vertical está el factor de descuento o δ (delta), en el eje horizontal inferior los costos k y en la parte superior la Utilidad Máxima (Ue).

Ya mencionado lo anterior, se prosigue con la evaluación de resultados.

CASO PARA DOS JUGADORES

Con Polizones (CP)

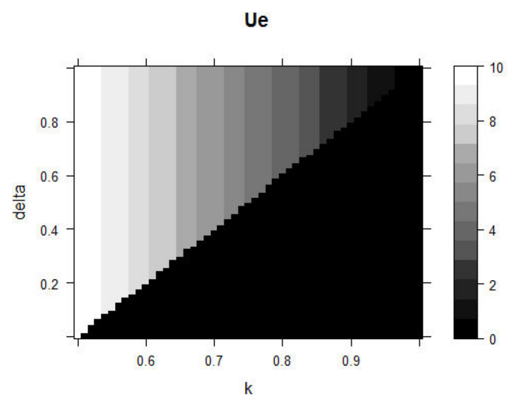

Figura 1: Levelplot Costos Simétricos-CP

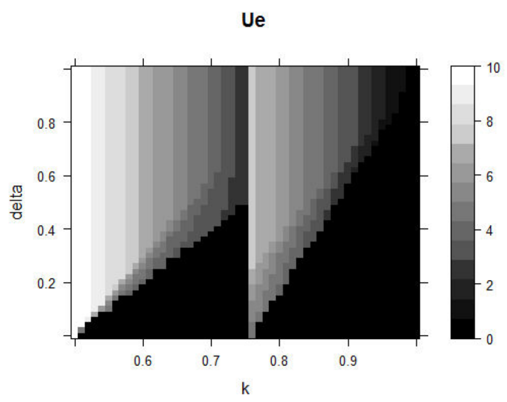

Figura 2: Levelplot Costos asimétricos-CP

Tal como se mencionó anteriormente, los *levelplots* comprenden el componenete dinámico de los cambios de las variables acorde un parámetro específico que

[3] Para mayor información sobre este criterio de cambio en los costos, remítase al capítulo Anexos donde se encuentra el Script.

varía. Podrá notarse, por lo tanto, que ambas gráficas comparan los cambios en el factor de descuento y Ue cuando los costos cambian.

La Figura 1 muestra a un escenario de costos simétricos con Polizones, es decir, un contexto en donde los agentes pueden contribuir nada para la paz, pese a que los mismos tengan la misma tasa de cambio en los costos. Bajo este contexto, se observa que, como es de esperar, la Ue tiene una relación inversa con k, mientras que el factor de descuento tiene una relación directa con los costos.

Dicho lo anterior, la lógica de fondo radica en lo que representa el factor de descuento[4] con respecto a los costos, es decir, siendo delta una función que depende inversamente de r, el cual es costo de oportunidad. Así pues, si los costos suben, el delta necesario se incrementa dado un nivel mayor de paciencia requerido para seguir en la dinámica de contribución, lo que a su vez requiere un r más bajo, es decir, un costo de oportunidad menor que justifique la continuidad de la contribución para la paz y no desviarse hacia otro proyecto en el cual el agente i derive mayor utilidad.

En cuanto a la Figura 2, la intuición es la misma entre las variables, con la salvedad de la situación que altera la relación entre los costos y la utilidad. Bajo este contexto se puede observar que hay un nivel de costo igual a 0.75 que divide la figura en la mitad. Esto se debe a la diferencia que hay entre los costos, pues son asimétricos en la tasa de cambio. Es decir, uno de los agentes posee una mayor tasa de incremento, la cual llega hasta el punto en que el beneficio obtenido por la contribución se iguale al costo de hacerla[5].

Puesto que los agentes tienen la posibilidad de ser polizones y no lo hacen en la medida que van jugando, se encontró que el agente quien tenga crecimiento ralentizado en sus costos, contribuirá hasta que su utilidad también sea nula, pese a que el otro agente no pueda contribuir. Este es el motivo por el cual mayores valores de costos a 0.75 vuelve a iniciarse una nueva relación entre el factor de descuento que genera una utilidad esperada mayor que la inmediatamente anterior, aunque menor a la inicial. Esto se observa en la escala que yace a la derecha de la Figura 2, donde la utilidad percibida a costos nulos es 10, comparada a la menor a 9 que se observa a costos iguales a 0.75.

Ahora bien, este resultado no es fortuito, dado que si se observa con detalle, hay dos niveles de costos que generarán una misma utilidad para los agentes. Con esto se quiere decir que hay un costo menor a 0.75 que garantiza que ambos agentes contribuyan y reciban la misma utilidad que percibirían a un costo mayor de

[4] $\delta = \frac{1}{1+r}$

[5] Lo que es igual a decir $g_i = 0$

0.75 en el cual un sólo agente puede seguir contribuyendo, por lo que podría tratarse de un equilibrio, no necesariamente el de Nash, en el cual ambos agentes enuentren un punto de indiferencia entre sus costos y su utilidad debido a la diferencia en la tasa de costos. Sin embargo, para verlo con más detalle, se sugiere ir a los Anexos en el Script al cual se refiere.

Sin Polizones (SP)

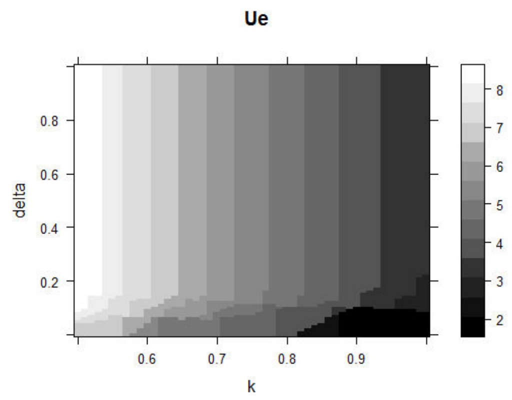

Figura 3: Levelplot Costos Simétricos-SP

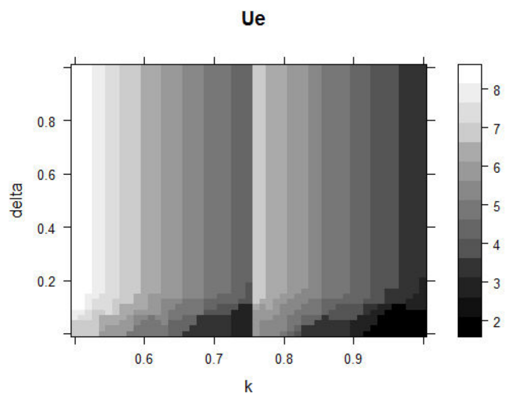

Figura 4: Levelplot Costos asimétricos-SP

La lógica empleada en las primeras dos Figuras es compatible con este par. La relación que hay entre la utilidad máxima y los costos se conserva, al igual que la relación que hay entre el nivel de paciencia y los costos, aunque de una forma más difusa. Vale la pena aclarar que aquello que subyace al comportamiento de estas gráficas es la forma funcional, puesto que el logarítmo restringe valores nulos de contribución, por lo que los agentes deben contribuir.

Bajo este contexto, la Figura 1 presenta una situación en la que el agente, quiera o no, deba contribuir pese a los incrementos que se vayan presentando en los costos. Dado el sentido de obligatoriedad que pueda tener el agente, hay incrementos en los costos que no tienen mucha incidencia sobre la paciencia, lo que se traduce en un escenario donde los agentes, pese a que observen reducciones en los costos de oportunidad en la medida que se incrementen los costos de contribuir, no se sienten incentivados (o no pueden) renunciar al hecho de seguir contribuyendo hasta cierta etapa en la cual se llegue a un Equilibrio de Nash. Esto considerando una situación ajena a la estrategia del gatillo.

En cuanto a la Figura 4, se evidencia que la gráfica se divide en el nivel de costos donde el agente que tiene una mayor tasa de crecimiento en los costos percibe una utilidad nula, y corresponde, igualmente, a un valor aproximado a 0.75. Un ejercicio que ayuda a observar la dualidad que puede albergar un nivel determinado de paciencia es observar que, a un mismo nivel de delta, hay por lo menos dos puntos donde se percibe la misma utilidad a diferentes niveles de costos, conclusión similar a la que se llegó con base a la Figura 2.

CP vs. SP

Óptima

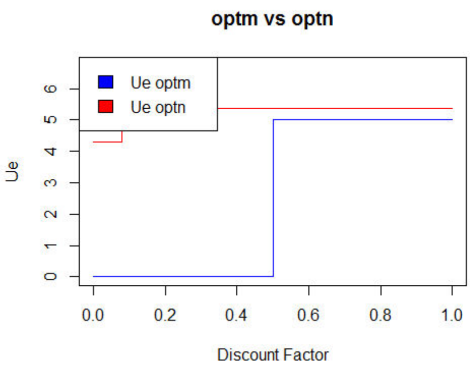

Figura 5: CP vs SP con Costos Simétricos-OPT

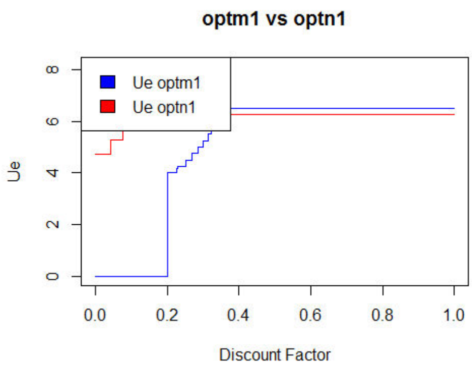

Figura 6: CP vs SP con Costos Asimétricos-OPT

De acuerdo con las figuras 5 y 6, la incidencia que tienen los polizones o su imposibilidad en el equilibrio es radical. Bajo una estrategia óptima puede verse en la Figura 5 que la Utilidad máxima obtenida sin polizón es superior cuando se permite. Esto tiene sentido si se piensa que, dada una obligación en la contribución para agentes con costos simétricos, el bienestar social es mayor si todos deben contribuir.

No obstante, dicho sentido parece desvanecerse cuando se presenta un caso más cercano a la realidad, en la cual hay asimetría en costos, cuya gráfica es la Figura 6. En ella se puede ver que bajo un valor de delta entre 02 y 0.4, la curva de utilidad máxima obtenida con polizones (azul) supera a aquella cuando existe la restricción de No polizón. Es trasfondo de este resultado se sostiene en la diferencia entre los costos y la obligación a contribuir, pues en la medida que todos deban contribuir, incluso si sus costos van en detrimento de los agentes, el bienestar social es inferior a cuando la posibilidad de ser polizón es libre, lo que desde otra perspectiva implica la libertad de contribuir hasta el nivel que se desee. Por lo tanto, el factor de descuento donde se da el cruce es el nivel de paciencia en el cual los agentes que contribuían, dada la prohibición de free-rider, comienzan a percibir una desutilidad **relativa** frente a la situación en la que se contribuye o no libremente.

Grim Trigger (GT)

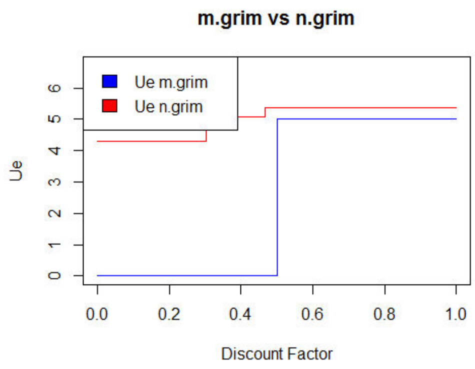

Figura 7: CP vs SP con Costos Simétricos-GT

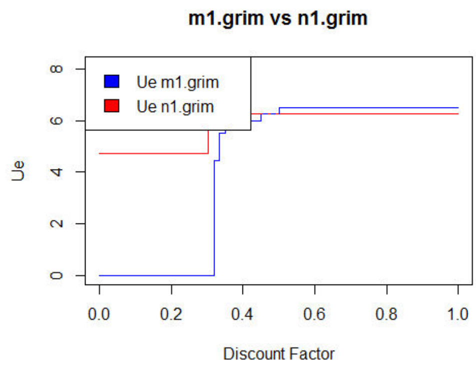

Figura 8: CP vs SP con Costos Asimétricos-GT

Como se puede ver en la Figura 7, suponiendo simetría en los costos de contribución, la imposición a contribuir sumado a la amenaza del castigo (línea roja), conlleva a que la utilidad social sea mayor con respecto a una suceso donde se permitan polizones (línea azul), debido a que en la última situación existen incentivos a no contribuir para valores de paciencia inferiores a 0.5.

Ahora bien, bajo la presencia de costos asimétricos en la Figura 8 se puede observar que, si bien el nivel de paciencia necesario para que se dé la contribución en el caso de polizones es el mismo que antes (0,5 como se puede observar), la utilidad máxima ahora es mayor cuando se permiten polizones paralelo al caso adverso, debido a que aquel individuo que tenga unos costos de contribución inferiores, aportará al bien público porque esto le genera utilidad y no porque existe una "ley" que lo obliga a hacerlo.

CASO PARA N JUGADORES

Con Polizones (CP)

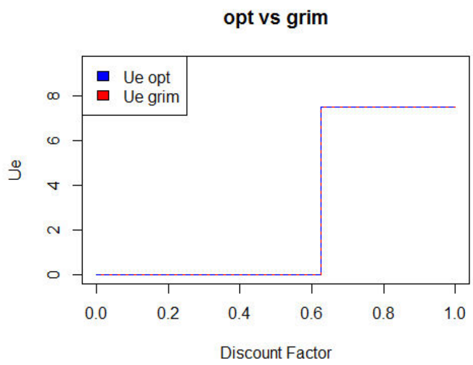

Figura 9: OPT vs GT con Costos Simétricos de 0.75 y 3 agentes-CP

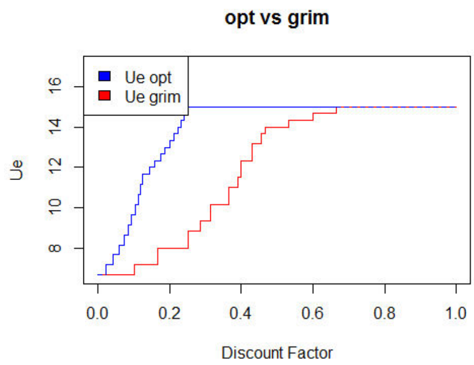

Figura 10: OPT vs GT con Costos Asimétricos $(2/6, 3/6, 4/6)$ con 3 agentes-CP

Como puede observarse en ambas gráficas, el comportamiento con tres agentes, es decir, tres grupos homogéneos dentro de la sociedad que debaten si se contribuye para la paz, es similar cuando hay dos agentes. De hecho, el comportamiento es la misma, resultado que podrá corroborarse con el código disponible en el Script. Se puede ver que bajo costos sim'étricos, en la Figura 9, la utilidad máxima percibida es la misma en ambos escenarios, eso se debe al hecho a la valoración que los mismos individuos le dan a la paz y a la no diferencia que hay entre aplicar la estrategia del gatillo cuando se cuenta con costos simétricos.

Por otro lado, cuando hay asimetría en los costos, en la figura 10, la utilidad máxima de optimal es superior a la obtenida en la estrategia GT cuando el delta se encuentra entre poco más de 0 y un poco menos de 0,65. Esto evidencia una situación en la cual, obligar a los agentes a contribuir por medio de un castigo, como lo hace la estrategia del gatillo, puede llegar a ser ineficiente en la medida que se va jugando cada etapa con asimetría en costos, dado que el agente que alcance su utilidad nula se verá obligado a seguir con su estrategia pese a que no lo desee estrictamente dado el castigo del gatillo.

Sin Polizones (SP)

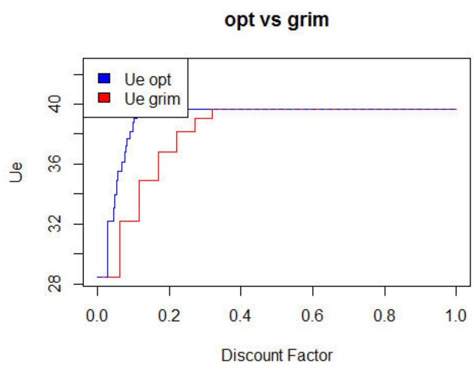

Figura 11: OPT vs GT con Costos Simétricos de 0.75 y 3 agentes-SP

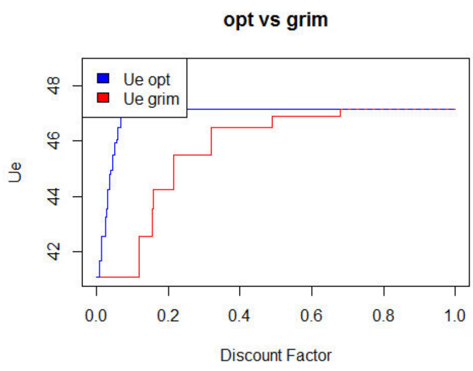

Figura 12: OPT vs GT con Costos Asimétricos $(2/6, 3/6, 4/6)$ con 3 agentes-SP

Como se puede observar en las figuras 10 y 11, obligar a contribuir, no únicamente por el castigo impuesto por el GT, puede llegar a determinar los resultados del juego para el resto de las etapas de tal forma que la Ue optimal sea superior o igual a la percibida bajo la estrategia GT. En la Figura 10 se detalla que la utilidad máxima obtenida en Optimal supera a la de GT de forma estricta cuando se encuentra el factor de descuento entre un poco más de 0 y menos de 0.25, lo que implica que a mayores valores de paciencia, se reducen los incentivos a seguir contribuyendo por decisión propia de los agentes, lo que se manifiesta en menores costos de oportunidad que confrontan a los agentes en cuanto a la decisión de aportar un monto superior o inferior al otro, aunque jamás igual a 0.

Por otro lado, la Figura 11 ilustra que ante asimetría en los costos, los agentes pueden tener un nivel mayor de paciencia para seguir contribuyendo de más por voluntad, dado que la Utilidad máxima es superior a la percibida con la estrategia GT. La intuición capturada por medio de este gráfico radica en que se requiere una mayor tasa de descuento que haga a los ciudadanos indiferentes entre aportar voluntariamente o de forma obligada, puesto que a mayores valores de delta, menor es la tasa r que mide el costo de oportunidad de contribuir, pese a que la restricción de no polizón y la forma funcional no permitan que se desvíe y por obligación deban aportar un valor mayor a 1, dato disponible en el Script de Anexos.

STICK & CARROT (SC)

Con Polizones (CP)

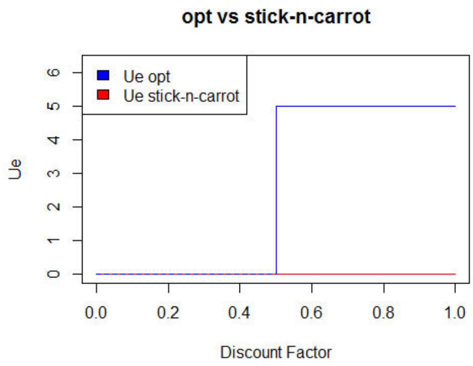

Figura 13: OPT vs SC con Costos Simétricos de 0.75 y 2 agentes-Stick & Carrot- CP

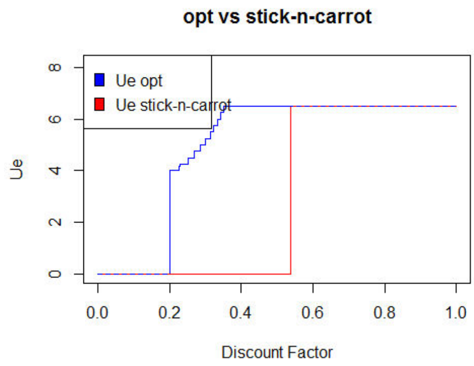

Figura 14: OPT vs GT con Costos Asimétricos con 2 agentes-Stick & Carrot - CP

Sin Polizones (SP)

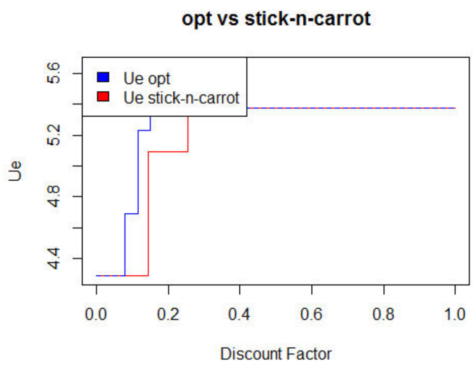

Figura 15: OPT vs SC con Costos Simétricos de 0.75 y 2 agentes-Stick & Carrot- SP

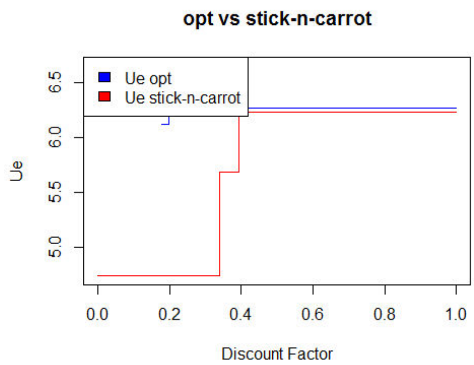

Figura 16: OPT vs GT con Costos Asimétricos con 2 agentes-Stick & Carrot - SP

La Figura 13, demuestra que la estrategia SC no es muy efectiva ante presencia de costos simétricos y polizones, pues arroja para todo factor de descuento una utilidad conjunta nula; contrario al caso en el cual existen costos simétricos, situación en la cual la estrategia SC se comporta muy similar a la estrategia Grim Trigger representada en la Figura 7, aunque con un nivel de paciencia un poco superior para lograr la cooperación.

Lo anterior tiene sentido si se recuerda que SC es una estrategia con un castigo mucho más suave que el de la estrategia Trigger. Por otro lado, cuando existe la restricción a ser polizón (Ver Figura 15 y Figura 16) la utilidad alcanzada mediante el uso de una estrategia Stick and Carrot es superior que aquella cuando el comportamiento estratégico de los jugadores, conduce al surgimiento de polizones. Es incluso superior a la alcanzada mediante una estrategia Grim Trigger cuando se obliga contribuir. Lo anterior, puede denotar una efectividad de la estrategia Stick and Carrot de regresar a los agendas de nuevo a la senda de la cooperación, situación en la cual predomina "la zanahoria" por sobre el "garrote". En consecuencia, esta situación demuestra que el castigo impuesto por una estrategia GT puede ser muy severo ante un contexto en el que de todas maneras los agentes están obligados a contribuir.

CASO CON AGENTES HETEROGÉNEOS

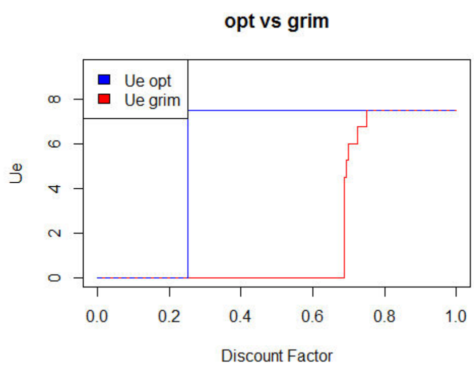

Figura 17: OPT vs GT con Costos Simétricos-Agentes Heterogéneos

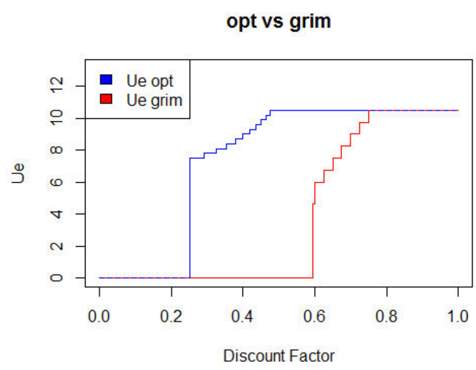

Figura 18: OPT vs GT con Costos Asimétricos - Agentes Heterogéneos

En este contexto hay una heterogeneidad entre los agentes (o grupos) a raíz de la importancia que el agente i tiene sobre la decisión de aportar para el bien público, lo que en otras palabras es la mayor percepción de utilidad ante el acceso a la paz. Esta situación evidencia una en la cual agentes afectados directamente por la guerra (y que desean la paz) interactúan con otros que fueron afectados de forma indirecta y cuyo deseo de paz puede ser menor a la del primer grupo o agente.

La Figura 17 muestra cuando los agentes están sometidos a los mismos costos, cuya particularidad merece recalcar, dado que ambas curvas (la OPT y la GT) no son las mismas, puesto que la importancia que le atribuye el agente 1 es mayor, por lo que la Utilidad máxima (Ue) es mayor, pues comprende la utilidad total de los implicados. Lo importante de recalcar es la brecha entre la tasa de descuento en la cual la utilidad máxima OPT supera a la obtenida bajo Grim Trigger, dado que se encuentra entre poco más de 0.25 y menos de 0.8, los cuales son valores, un poco más extremos, que indican una mayor disponibilidad a la contribución para ciertos valores de costos de oportunidad que no estén comprendidos a deltas mayores de 0.7 y menores de 0.25, dado que sus r asociados son 0.428 y 3 respectivamente.

Por otro lado, la Figura 18 muestra el escenario con costos asim'étricos, con la particularidad del rango de delta que comprende los valores en los que la utilidad máxima percibida por optimal supere a la de GT (es decir, la curva azul superior a la roja), puesto que esta es menor, situación no percibida en las gráficas anteriores, excepto en Stick & Carrot. Lo que se resalta de este suceso es que bajo costos asimétricos, el agente que valora más la paz seguirá contribuyendo más, motivo por el cual la Ue es mayor que en la Figura 17.[6] De hecho, se requiere un mayor costo de oportunidad que haga a los individuos indiferentes en la contribución bajo ambas estrategias (OPT y GT), los cuales son los deltas que garantizan que ambas curvas sean las mismas. Por ende, de ambas gráficas se desprende una idea intuitiva: a mayor importancia que se obtenga ante la paz, el bienestar social es mayor ante mayores incrementos de las contribuciones, sea bajo costos simétricos o asimétricos, hasta que se alcance un nivel de paciencia que haga indiferentes a los agentes contribuir por decisión propia o por el castigo de Grim Trigger.

[6]Note que la Figura 17 comprende valores de Ue entre 0 y 10, mientras la Figura 18 posee un rango que llega hasta 14 la Ue.

Conclusiones

- Para todos los casos, excepto para el que se desarrolla con costos simétricos y posibilidad de polizones, la estrategia "optimal", la cual conduce a un castigo de un periodo si un agente se ha desviado, conduce a la cooperación para valores de paciencia inferiores que aquellos necesarios tanto en la estrategia Grim Trigger como Stick and Carrot. Lo anterior, puede denotar que dicho mecanismo es el más eficiente para garantizar la cooperación. Sin embargo, en el punto de la máxima cooperación, ambas estrategias alcanzan la misma utilidad máxima conjunto para ambos grupos de agentes.

- La diferencia en los costos de contribución, factor que afecta la función de pagos de un agente repercutiendo indirectamente en su valoración, siempre conduce a niveles de utilidad más altos que en el caso de costos simétricos. Lo anterior, debido a que aquel agente con los costos inferiores está más dispuesto a contribuir para cualquier nivel de paciencia e incluso cuando el otro agente está actuando como un polizón.

- La estrategia "Stick and Carrot" no es muy eficaz cuando se realiza la simulación con costos simétricos, pues para cualquier nivel de paciencia, la utilidad conjunta es nula (Ver Figura 13). Todo parece indicar, que para este caso en particular el garrote predomina sobre la zanahoria, el cual parece inexistente para salir del periodo de castigo. Sin embargo, para el caso de agentes con diferencia en los costos de contribución, existe un factor de descuento (aproximadamente 0.9) para el cual la "zanahoria" o premio es lo suficientemente atractivo como para garantizar la cooperación(Ver Figura 14). En definitiva, la estrategia Stick and Carrot, requiere un nivel de paciencia muy elevado, dada la suavidad de su castigo en comparación con una estrategia Grim Trigger, para garantizar la cooperación y lograr la máxima provisión del bien público.

- A manera general, el aumento en la cantidad de agentes, en este caso grupo de ciudadanos, requiere de tasas de paciencia superiores para lograr la cooperación (Ver Figura 9).

- *Con Polizones*:

  - A medida que el costo de contribuir aumenta, el nivel de paciencia necesario para que haya contribución es mayor, debido a la necesidad de individuos menos elásticos ante cambios en los costos(Ver Figura 1). De hecho si los costos tienden a cero, el individuo i tiende a ser perfectamente elástico ante cambios en k (A medida que el costo de contribución aumenta el nivel de paciencia necesario para que haya contribución es mayor. Dado que se necesitan individuos que no reaccionen tanto ante incrementos en los costos)

- Cuando hay costos asimétricos por la diferencia en la tasa de cambio de los mismos, el individuo cuya tasa de cambio de costos sea menor seguirá contribuyendo una vez el otro agente no pueda contribuir más dada la desutilidad por contribución(Ver Figura 2). (dadas diferentes tasas de cambio en los costos, los individuos seguirán contribuyendo hasta el punto en el que su utilidad percibida sea nula, lo que ocurrirá primero en el que agente con una tasa de crecimiento mayor de sus costos)

- *Sin Polizones*

  - Ante una restricción de no polizón y costos simétricos los individuos no son tan sensibles hasta cambios en los costos porque no se les permite actuar como polizones (evadir la contribución), salvo en costos superiores a 0,8 como proporción de la contribución, caso en el que los agentes se vuelven más sensibles ante dichos cambios (ver Figura 3).
  - Cuando no es posible dejar de contribuir, la utilidad máxima de los agentes se ve afectada en la medida que haya tasas de interés que incrementen la desutilidad por contribuir, lo que se manifiesta en una menor utilidad máxima esperada bajo la estrategia GT. Esto tiene sentido en la medida que obligar a contribuir a aquellos que no lo desean en un momento que evalúa trade-offs con respecto a ello, influirá negativamente en su utilidad percibida. Esto sucede tanto para agentes con costos simétricos, como asimétricos, con la salvedad que estos últimos requieren niveles de paciencia mayores para lograr la máxima cooperación, y en consecuencia, la máxima utilidad.
  - A diferencia de la utilidad máxima percibida cuando la restricción de no polizón esta inactiva con diferentes tasas de cambio en los costos, ante un escenario con una contribución obligatoria, la utilidad máxima percibida es mayor manteniendo los mismos rangos del factor de descuento y de los costos. Esto es una idea tentativa ante la viabilidad de una ley que obligue a contribuir y más aún cuando se está en un contexto de materialización de la paz en la cual no todos los agentes tiene las preferencias por ella, incluso se podría decir que hay personas que devenguen utilidad por la ausencia de la paz, caso no representado en el presente trabajo.

Anexos

```
################# SCRIPT PDG
################# MARÍA CATALINA SAAVEDRA LLOREDA
################# SANTIAGO MOSQUERA DAZA
#################
################# SIMULACIÓN PARA EL ACCESO A LA PAZ COMO UN BIEN
################# PÚBLICO PURO A LA LUZ DE LA TEORÍA DE JUEGOS
#################
################# AÑO 2017
#################

#El presente Script es realizado bajo la versión R-3.3.1. Los paquetes que
#requieren instalarse son devtools, glpkAPI,lattice, Rcpp, rgmpl, skUtils,
#slam, stringr,repgame, restorepoint y rowmins. Adicionalmente, se
#emplea stargazer para exportar tablas a Latex. El Script presente es
#realizado con la ayuda de los indicios de programación para simulación
#de juegos repetidos de Sebastian Kranz en su publicación "Interactively
#solving repeated games". Todos los códigos son explicados en Script con
#la mayor brevedad y simpleza posible. En caso de presentarse cuestiones
#más allá de las que comprende  este archivo, sugerimos que se remita al
#texto de Kranz o su plataforma en GitHub
# https://github.com/skranz/repgame

#Ahora bien, la función de Bienes públicos es una función de bienes
#públicos que depende de los elementos X y ki, donde X son las
#contribuciones y  ki son los costos de contribución para el agente i=1,2.


############################################################
########## C A S O  P A R A  D O S  J U G A D O R E S
###########################################################

############################################################
C O M P O N E N T E  O P T.V S. G R I M - T R I G G E R  (GT)
############################################################

############################################################
############## C O N  P O L I Z O N E S (CP)
```

```
############################################################

##### COSTOS SIMÉTRICOS (CS)
public.goods.game = function(X, k1,k2=k1) {
store.objects("public.goods.game")
x1m = matrix(X,NROW(X),NROW(X),byrow=FALSE)
x2m = matrix(X,NROW(X),NROW(X),byrow=TRUE)
g1 = (x1m+x2m)/2 - k1*x1m
g2 = (x1m+x2m)/2 - k2*x2m

#La definición de las funciones de pago, gi, corresponden a un escenario
#donde hay posibilidad de ser free-rider. La idea es observar si, ante la
#presencia o no de polizones, hay una diferencia en el E.N

name = paste("Public Goods Game k1=", k1, " k2=", k2, sep="")
m = init.game(n=2,g1=g1,g2=g2,name=name, lab.ai=X)

#En esta línea podemos decidir si se desea asimetría o no. Esto se haría
# simplemente agregando dentro de m=init.game(, symmetric= T o F).
# Para efectos prácticos, no decidimos agregarlo.

#En últimas, se requiere simetría para aplicar la estrategia Stick-Carrot,
#que más adelante se tratarán bajo ciertas modificaciones. Luego se
# soluciona el juego m para arrojar la gráfica. Aquí sólo aparecerán las
#filas donde hay equilibrios óptimos.

m = solve.game(m, keep.only.opt.rows=T)
plot(m)
return(m)
}
#Podemos garantizar que las matrices de contribuciones de los agentes
# sean las matrices transpuestas del resto.  Esto significa que el agente 1
#  tiene una matriz de contribuciones transpuesta al del agente 2, para el
#caso de  dos jugadores o contribuyentes para el bien público.

restore.objects("public.goods.game")
x1m= matrix(X,NROW(X),NROW(X),byrow=!TRUE)
x2m = t(x1m)
g1 = (x1m+x2m)/2 - k1*x1m
g2 = (x1m+x2m)/2 - k2*x2m
```

```
#Se inicia la simulación con los parámetros de interés. Para el caso
#presente, se comienza con un set de contribuciones que van desde 0
#hasta 10. Cabe destacar que las contribuciones pueden tener la mayor
#cota que se quiera imaginar. Para efectos más simples, se ha tomado
#como conjunto acotado el espacio de contribuciones que se
#comprendan dentro del intervalo [0,10], en otras palabras, desde no
#contribuir hasta hacerlo con 10. El primer caso se hará para costos
#simétricos. A saber, 0.75, pese a que este puede ser cualquier valor
#siempre que está entre [0.5,1].

#Gráfica 1. PGG-Opt CS-CP
m = public.goods.game(X=0:10,k1=0.75,k2=0.75)
m$opt.mat
stargazer(m$opt.mat)

###Análisis de estática comparativa. ¿Qué sucede ante incrementos de
#costos de forma secuencial a una diferencia de 0,01 desde 0,5 hasta 1.
#Así que se vectorizaría cada argumento de los parámetros de costo. Esto
#simularía el proceso iterativo de un loop que aplique constantemente  la
#función public.goods.game() para cada valor de ki. m.list almacenaría
#toda estas iteraciones. De esta forma si se desea conocer la solución en
#vigésimo quinto juego se escribe m.list[[25]] y arrojaría todos los datos
#alusivos a dicho equilibrio.

pg.games = Vectorize(public.goods.game,
vectorize.args = c("k1","k2"),SIMPLIFY=FALSE)
k.seq = seq(0.5,1,by = 0.01)
m.list = pg.games(X=0:10,k1=k.seq,k2=k.seq)

### No obstante, observar juego por juego no tiene mayor efecto
#práctico que observar el comportamiento general de los valores de las
#variables de interés. De esta forma, se genera una Gráfica de niveles de
#los máximos pagos conjuntos de equilibrio en función del factor de
#descuento, delta, y el costo marginal de producción, k.

#Gráfica 2.  LevelPlot Ue vs k vs delta en Opt CS-CP

mat = levelplot.payoff.compstat(m.list, par = k.seq,
xvar = "k", yvar = "delta", payoff.var="Ue",
delta = seq(0,1,by=0.01), col.scheme = "grey")
### ESTRATEGIA GT.
```

```
###Dado lo anterior, aplicaremos el efecto que tiene en el juego la
# decisión de aplicar una estrategia de gatillo. Tomando como referencia
#a m, se aplica entonces la solución para la  estrategia en cuestión.

m.grim = solve.game(set.to.grim.trigger(m))
m.grim$opt.mat
stargazer(m.grim$opt.mat)
m$opt.mat

###¿Cuál es la diferencia entre el resultado de m y m.grim en cuanto a Ue
# y delta? Para responder a dicha pregunta se ejecuta el siguiente
#comando que permite observar cómo cambia la máxima utilidad
#conjunta de los agentes acorde al valor de delta y a la estrategia jugada.

#Gráfica 3. Opt vs GT  CS-CP
plot.compare.models(m, m.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")


##### COSTOS ASIMÉTRICOS (CA)

#Gráfica 4. PGG-Opt CA-CP
m1 = public.goods.game(X=0:10,k1=0.6,k2=0.75)
m1$opt.mat
stargazer(m1$opt.mat)
m1.grim = solve.game(set.to.grim.trigger(m1))
m1.grim$opt.mat
stargazer(m1.grim$opt.mat)

###Como se observa, se ha decidido llevar a cabo la solución del juego
#por ambas estrategias para poder compararlas con lo anterior.
#Gráfica 5. Opt vs GT CA-CP
plot.compare.models(m1, m1.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")

#### *La Gráfica comparativa entre las estrategias, cambiando
#únicamente la naturaleza de los costos.*

###Para este caso, ver cómo se comporta la relación entre los niveles de
#los máximos pagos conjuntos de equilibrio en función  de delta y k, se
#hace se vectoriza y posteriormente se gráfica para ver dicho
```

```
#comportamiento. Tal como en el caso anterior. No obstante, dado que
#hay asimetría de costos, se puede asumir que los costos del agente 1
#aumentan más rápido que los del 2. De esta forma el código queda de
#la siguiente forma:

pg.games = Vectorize(public.goods.game,
vectorize.args = c("k1","k2"),SIMPLIFY=FALSE)
k.seq1 = seq(0.5,1,by = 0.02)
k.seq2 = seq(0.5,1,by = 0.01)

#Gráfica 6. LevelPlot Ue vs k vs delta en Opt CA-CP
m.list = pg.games(X=0:10,k1=k.seq1,k2=k.seq2)
mat = levelplot.payoff.compstat(m.list, par = k.seq2,
xvar = "k", yvar = "delta", payoff.var="Ue",
delta = seq(0,1,by=0.02), col.scheme = "grey")

##Aquí se debe rescatar que se ha decidido que el elemento "par=k.seq"
#de la configuración de la Gráfica quede como "par=k.seq2", el cual es la
#secuencia más lenta. Esto se prefiere así porque al optar por el más
#rápido, la Gráfica no puede llevarse a cabo por la diferencia en el
#número de filas.

#################################
################################# S I N  P O L I Z O N E S (SP)
#################################

#### COSTOS SIMÉTRICOS (CS)
public.goods.game = function(X, k1,k2=k1) {
store.objects("public.goods.game")
x1m = matrix(X,NROW(X),NROW(X),byrow=FALSE)
x2m = matrix(X,NROW(X),NROW(X),byrow=TRUE)
g1 = 2*log(x1m)+2*log(x2m) - k1*x1m
g2 = 2*log(x1m)+2*log(x2m) - k2*x2m

#La definición de las funciones de pago, gi, corresponden a un escenario
#donde no hay posibilidad de ser free-rider. Aquí cabe recalcar que las
#funciones de pago dependen de la suma ponderada de elasticidades de
#las contribuciones menos el costo de llevarlas a cabo.

name = paste("Public Goods Game k1=", k1, " k2=", k2, sep="")
n = init.game(n=2,g1=g1,g2=g2,name=name, lab.ai=X)
```

```
#En esta línea podemos decidir si se desea asimetría o no. Esto se haría
#simplemente agregando dentro de m=init.game(, symmetric= T o F).
#Para efectos prácticos, no decidimos agregarlo. En últimas, se requiere
#simetría para aplicar la estrategia Stick-Carrot, que más adelante se
#trataría bajo ciertas modificaciones. Luego se soluciona el juego n para
#arrojar la Gráfica. Aquí sólo aparecerían las filas donde hay equilibrios
#óptimos.

n = solve.game(n, keep.only.opt.rows=T)
plot(n)
return(n)
}
# Se hace el mismo proceso de transponer las contribuciones entre los
# agentes y almacenar las funciones de pago sin oportunidad de polizón.

restore.objects("public.goods.game")
x1m= matrix(X,NROW(X),NROW(X),byrow=!TRUE)
x2m = t(x1m)
g1 = 2*log(x1m)+2*log(x2m) - k1*x1m
g2 = 2*log(x1m)+2*log(x2m) - k2*x2m

#Se inicia la simulación con los parámetros de interés. Para el caso
#presente, se comienza con un set de contribuciones que van desde 0
#hasta 10 como en el caso anterior.

#Cabe destacar que las contribuciones pueden tener la mayor cota que se
#quiera imaginar. Para efectos más simples, se ha tomado como conjunto
#acotado el espacio de contribuciones que se comprendan dentro del
#intervalo (0,10], recordando que no hay oportunidad para comportarse
#como polizón, es decir, los agentes deben contribuir. El primer caso se
# haría para costos simétricos. A saber, 0.75, como en el ejemplo anterior
#para que sea comparable.

#Cabe resaltar que la función logarítmica no permite contribuciones
#nulas, he ahí la restricción de no free-rider.
#Dado que el conjunto de contribuciones escogido ha sido (0,10], puede
#que un agente decida aportar muy poco, sin que esto no implique que
#decida no aportar. Así se forma el siguiente escenario con un conjunto
#de contribuciones desde 10^(-70)  hasta 10.
```

```
#Gráfica 7. PGG-Opt CS-SP con contribuciones pequeñas
n = public.goods.game(X=10^(-70):10,
k1=0.75,k2=0.75)

#Pero ante efectos prácticos, los aportes son unitarios, así que se
# compara con el siguiente escenario:
#Gráfica 8. PGG-Opt CS-SP con Contribuciones unitarias

n = public.goods.game(X=1:10,k1=0.75,k2=0.75)
n$opt.mat
stargazer(n$opt.mat)

#Bajo esta situación, cabe preguntarse sobre el comportamiento de la
#grafica del óptimo cuando se compara Ue vs V. Así que vectorizamos el
#argumento de las contribuciones y formamos una secuencia con costos
#constantes iguales a 0.75

pg.games = Vectorize(public.goods.game,
vectorize.args = "X",SIMPLIFY=FALSE)
x.seq = seq(0.1,1,by = 0.018)
x.list = pg.games(X=x.seq,k1=0.75, k2=0.75)

#Gráfica 9. Levelplot entre Ue vs delta vs X con x como secuencia entre
# (0.1, 1) en 0.018
mat = levelplot.payoff.compstat(m.list, par = x.seq,
xvar = "x", yvar = "delta", payoff.var="Ue",
delta = seq(0,1,by=0.02), col.scheme = "grey")

###Análisis de estática comparativa. ¿Qué sucede ante incrementos de
#costos de forma secuencial a una diferencia de 0,01 desde 0,5 hasta 1.
#Así que se vectorizaría cada argumento de los  parámetros de costo.
#Esto simularía el proceso iterativo de un loop que aplique
# constantemente la función public.goods.game() para cada valor de ki.
# m.list almacenaría toda estas iteraciones. De esta forma si se desea
#conocer la solución en vigésimo quinto juego se escribe n.list[[25]] y
#arrojaría todos los datos alusivos a dicho equilibrio.

pg.games = Vectorize(public.goods.game,
vectorize.args = c("k1","k2"),SIMPLIFY=FALSE)
k.seq = seq(0.5,1,by = 0.01)
n.list = pg.games(X=1:10,k1=k.seq,k2=k.seq)
```

```
### Se genera una Gráfica de niveles de los máximos pagos conjuntos de
# equilibrio en función del factor de descuento, delta, y el costo marginal
#de producción, k.

#Gráfica 10.  LevelPlot Ue vs k vs delta en Opt CS-SP
mat = levelplot.payoff.compstat(n.list, par = k.seq,
xvar = "k", yvar = "delta", payoff.var="Ue",
delta = seq(0,1,by=0.01), col.scheme = "grey")

### ESTRATEGIA GT.

###Dado lo anterior, aplicaremos el efecto que tiene en el juego la
#decisión de aplicar una estrategia de gatillo. Tomando como referencia
# a m, se aplica entonces la solución para la estrategia en cuestión.

n.grim = solve.game(set.to.grim.trigger(n))
n.grim$opt.mat
stargazer(n.grim$opt.mat)
n$opt.mat
stargazer(n$opt.mat)

###¿Cuál es la diferencia entre el resultado de m y m.grim en cuanto a Ue
# y delta? Para responder a dicha pregunta se ejecuta el siguiente
#comando que permite observar cómo cambia la máxima utilidad
#conjunta de los agentes acorde al valor de delta y a la estrategia jugada.

#Gráfica 11. Opt vs GT  CS-SP
plot.compare.models(n, n.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")

##### COSTOS ASIMÉTRICOS (CA)

#Gráfica 12 PGG- Opt CA-SP
n1 = public.goods.game(X=1:10,k1=0.6,k2=0.75)
n1.grim = solve.game(set.to.grim.trigger(n1))

###Como se observa, se ha decidido llevar a cabo la solución del juego
#por ambas estrategias para poder compararlas con lo anterior, cuando
#hay asimetría de costos, los resultados son los siguientes.
```

```
n1$opt.mat
stargazer(n1$opt.mat)
n1.grim$opt.mat
stargazer(n1.grim$opt.mat)

#Gráfica 13 Opt vs GT CA-SP
plot.compare.models(n1, n1.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")

#### *La Gráfica comparativa entre las estrategias, cambiando
#únicamente la naturaleza de los costos.*

###Para este caso, ver cómo se comporta la relación entre los niveles de
#los máximos pagos conjuntos de equilibrio en función  de delta y k, se
# hace se vectoriza y posteriormente se gráfica para ver dicho
#comportamiento.
###Tal como en el caso anterior. No obstante, dado que hay asimetría de
# costos, se puede asumir que los costos del agente 1 aumentan más
#rápido que los del 2. De esta forma el código queda de la siguiente forma:

pg.games = Vectorize(public.goods.game,
vectorize.args = c("k1","k2"),SIMPLIFY=FALSE)
k.seq1 = seq(0.5,1,by = 0.02)
k.seq2 = seq(0.5,1,by = 0.01)

n1.list = pg.games(X=1:10,k1=k.seq1,k2=k.seq2)

#Gráfica 14 Levelplot Ue vs k vs delta Opt CA-SP
mat = levelplot.payoff.compstat(n1.list, par =k.seq2,
xvar = "k", yvar = "delta", payoff.var="Ue",
delta = seq(0,1,by=0.02), col.scheme = "grey")

###Aquí se debe rescatar que se ha decidido que el elemento "par=k.seq"
# de la configuración de la Gráfica quede como "par=k.seq2", el cual es la
#secuencia más lenta. Esto se prefiere así porque al optar por el más
#rápido, la Gráfica no puede llevarse a cabo por la diferencia en el número
# de filas.


##########################
########################## C O M P A R A C I Ó N  E N T R E  CP  Y  SP
```

```
##########################

# Escenario 1: Óptimos con costos simétricos entre la posibilidad de
# polizones (CP) y sin ella (SP).
## Gráfica 15. CP[Opt, CS] vs SP[Opt, CS]
cmn=plot.compare.models(m, n, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "optm", m2.name = "optn")

# Escenario 2: Óptimos con costos asimétricos entre la posibilidad de
# polizones (CP) y sin ella (SP).
## Gráfica 16. CP[Opt, CA] vs SP[Opt, CA]
cm1n1=plot.compare.models(m1, n1, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "optm1", m2.name = "optn1")

# Escenario 3: GT con costos simétricos entre la posibilidad de polizones
# (CP) y sin ella (SP).

## Gráfica 17. CP[GT, CS] vs SP[GT, CS]
cmgng=plot.compare.models(m.grim, n.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "m.grim", m2.name = "n.grim")

# Escenario 4: GT con costos asimétricos entre la posibilidad de polizones
# (CP) y sin ella (SP).
## Gráfica 18. CP[GT, CA] vs SP[GT, CA]
cm1gn1g=plot.compare.models(m1.grim, n1.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "m1.grim", m2.name = "n1.grim")

############################################################
############################################################

#¿Qué tanto cambia el escenario sin polizones (SP) cuando x:(0,10]=
# x:[1,10] pasa a ser
# x: (0,10]= x:[10^{-t}, 10] donde t es un número lo suficientemente
# grande para que las contribuciones mínimas tiendan a cero, es decir,
# tiendan a no aportar sin que violen la condición de no polizón.
# Sea t=100, las contribuciones van desde 10^{-100} hasta 10. Así, el
# escenario SP es el siguiente, definiendo nuevamente la función con la
#simulación "q":
public.goods.game = function(X, k1,k2=k1) {
store.objects("public.goods.game")
x1m = matrix(X,NROW(X),NROW(X),byrow=FALSE)
```

```
x2m = matrix(X,NROW(X),NROW(X),byrow=TRUE)
g1 = 2*log(x1m)+2*log(x2m) - k1*x1m
g2 = 2*log(x1m)+2*log(x2m) - k2*x2m
name = paste("Public Goods Game k1=", k1, " k2=", k2, sep="")
q = init.game(n=2,g1=g1,g2=g2,name=name, lab.ai=X)
q = solve.game(q, keep.only.opt.rows=T)
plot(q)
return(q)}
#Llevando a cabo la simulación:
q = public.goods.game(X=10^(-100):10,k1=0.75,k2=0.75)
n = public.goods.game(X=1:10,k1=0.75,k2=0.75)
#¿Qué tanto difieren en UE y V ambas situaciones?
## Gráfica 19. SP[n, q]
cmn=plot.compare.models(q, n, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "optq", m2.name = "optn")
#En esta Gráfica puede verse que la Utilidad percibida no difiere en
# absoluto si el conjunto acotado de las contribuciones mínimas tiende a
# lo nulo, cuya prueba radica en la comparación entre las contribuciones
# mínimas como 1 vs la cota mínima de 10^{-100}.


#########################################################################
#################
################# C A S O  P A R A  N-  J U G A D O R E S
#################
#########################################################################

#########################################################################
##################
################## C O N  P O L I Z O N E S (CP)
##################
#########################################################################

#La función sin posibilidad de polizones es la siguiente:

pg.game = function(n,X,k=rep(((1+1/n)/2),n)) {
#pg.game es una función que retorna una matriz de pagos dada una
#matriz de acción x.mat, en la cual cada fila corresponde a un perfil de
#acción.
g.fun = function(x.mat) {
g = matrix(0,NROW(x.mat),n)
```

```
xsum = rowSums(x.mat)
#Es la forma funcional sin polizones, la cual es el promedio de las
# contribuciones menos el costo individual de aportar
for (i in 1:n) {
g[,i] = xsum / n - k[i]*x.mat[,i]
}
#Garantiza que el promedio de los valores de las filas de
# las contribuciones por cada jugador se le descuente el costo
# correspondiente del jugador i para brindar el aporte.
g
}
name=paste(n,"Player Public Goods Game")
m = init.game(n=n,g.fun=g.fun,action.val = X,
name=name, lab.ai=round(X,2))
m=solve.game(m)
plot(m)
m
}
#g.fun es la función que reemplaza init.game para el caso de más
# jugadores,  junto con action.Val, que es un vector por jugador, es decir,
#el conjunto contribuciones de los jugadores dado un sólo sujeto, puede
#ser fila o columna. Lo que hace action.val es asociar la acción de cada
# jugador a su contribución. Lo que hace g.fun es tomar el vector de
#perfiles de acción (matriz de contribuciones) y regresa el valor
#correspondiente para la función de pagos de cada etapa.

###COSTOS SIMÉTRICOS Y n=3 (CS, n3)

# Escenario ki=0
#Gráfica 20. (CS.0, n3)
s.0=pg.game(n=3, X=0:10, k=c(0, 0, 0))

#Escenario ki=0.25
#Gráfica 21. (CS.25, n3)
s.25=pg.game(n=3, X=0:10, k=c(0.25, 0.25, 0.25))

#Escenario ki=0.5
#Gráfica 22. (CS.5, n3)
s.5=pg.game(n=3, X=0:10, k=c(0.5, 0.5, 0.5))

#Escenario ki=0.75
```

```
#Gráfica 23. (CS.75, n3)
s.75=pg.game(n=3, X=0:10, k=c(0.75, 0.75, 0.75))

#Escenario Ki=1
#Gráfica 24. (CS, n3)
s.1=pg.game(n=3, X=0:10, k=c(1, 1, 1))

#ESTRATEGIA GT

#Para efectos prácticos, y siguiendo la línea de costos que hemos venido
#tratando, asumiremos que se emplea la estrategia GT en el Escenario
# ki=0.75.
s.75.grim=solve.game(set.to.grim.trigger(s.75))
s.75.grim$opt.mat
s.75$opt.mat
stargazer(s.75.grim$opt.mat)
stargazer(s.75$opt.mat)
#### Comparación

#Gráfica 25. Opt.vs.GT CP-CS n3
scomp=plot.compare.models(s.75, s.75.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")


###COSTOS ASIMÉTRICOS Y n=3 (CA, n3)
#K1=0, k2=k1+(1/6), k3=k2+(1/6)
#Gráfica 26. (CA1, n3)
sa.1=pg.game(n=3, X=0:10, k=c(0, 1/6, 2/6))

#K1=1/6, k2=k1+(1/6), k3=k2+(1/6)
#Gráfica 27. (CA2, n3)
sa.2=pg.game(n=3, X=0:10, k=c(1/6, 2/6, 3/6))

#K1=2/6, k2=k1+(1/6), k3=k2+(1/6)
#Gráfica 28. (CA3, n3)
sa.3=pg.game(n=3, X=0:10, k=c(2/6, 3/6, 4/6))

#K1=3/6, k2=k1+(1/6), k3=k2+(1/6)
#Gráfica 29. (CA4, n3)
sa.4=pg.game(n=3, X=0:10, k=c(3/6, 4/6, 5/6))
```

```
#ESTRATEGIA GT

#Tomamos el escenario sa.3 para jugar la estrategia GT:

sa.3.grim=solve.game(set.to.grim.trigger(sa.3))
sa.3.grim$opt.mat
sa.3$opt.mat
stargazer(sa.3.grim$opt.mat)
stargazer(sa.3$opt.mat)
#### Comparación

#Gráfica 30. Opt.vs.GT CP-CA n3
sa3comp=plot.compare.models(sa.3, sa.3.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")

##########################################################################
################## S I N  P O L I Z O N E S (SP)
##########################################################################

#Al igual que el caso de 3 jugadores con oportunidad de polizones, la
#función se define de la siguiente forma conservando gran parte de la
#estructura del código, teniendo como únicas diferencias el cambio
#de nombres y la reestructuración de las funciones de pagos.
pg.gamesp = function(n,X,k=rep(((1+1/n)/2),n)) {
g.fun = function(x.mat) {
g = matrix(0,NROW(x.mat),n)
xsumlog = rowSums(n*(log(x.mat)))
for (i in 1:n) {
g[,i] = xsumlog - k[i]*x.mat[,i]
}
g
}
name=paste(n,"Player Public Goods Game SP")
ms = init.game(n=n,g.fun=g.fun,action.val = X,
name=name, lab.ai=round(X,2))
ms=solve.game(ms)
plot(ms)
ms
}
###COSTOS SIMÉTRICOS Y n=3 (CS, n3)
```

```
# Escenario ki=0
#Gráfica 31 (CS.0, n3) SP
sp.0=pg.gamesp(n=3, X=1:10, k=c(0, 0, 0))

#Escenario ki=0.25
#Gráfica 32. (CS.25, n3) SP
sp.25=pg.gamesp(n=3, X=1:10, k=c(0.25, 0.25, 0.25))

#Escenario ki=0.5
#Gráfica 33. (CS.5, n3) SP
sp.5=pg.gamesp(n=3, X=1:10, k=c(0.5, 0.5, 0.5))

#Escenario ki=0.75
#Gráfica 34. (CS.75, n3) SP
sp.75=pg.gamesp(n=3, X=1:10, k=c(0.75, 0.75, 0.75))

#Escenario Ki=1
#Gráfica 35. (CS, n3) SP
sp.1=pg.gamesp(n=3, X=1:10, k=c(1, 1, 1))

#ESTRATEGIA GT

#Para efectos prácticos, y siguiendo la línea de costos que hemos venido
#tratando, asumiremos que se emplea la estrategia GT en el Escenario
# ki=0.75.
sp.75.grim=solve.game(set.to.grim.trigger(sp.75))
sp.75.grim$opt.mat
sp.75$opt.mat
stargazer(sp.75.grim$opt.mat)
stargazer(sp.75$opt.mat)


#### Comparación

#Gráfica 36. Opt.vs.GT CP-CS n3 SP
spcomp=plot.compare.models(sp.75, sp.75.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")


###COSTOS ASIMÉTRICOS Y n=3 (CA, n3) SP
#K1=0, k2=k1+(1/6), k3=k2+(1/6)
```

```
#Gráfica 37. (CA1, n3) SP
spa.1=pg.gamesp(n=3, X=1:10, k=c(0, 1/6, 2/6))

#K1=1/6, k2=k1+(1/6), k3=k2+(1/6)
#Gráfica 38. (CA2, n3) SP
spa.2=pg.gamesp(n=3, X=1:10, k=c(1/6, 2/6, 3/6))

#K1=2/6, k2=k1+(1/6), k3=k2+(1/6)
#Gráfica 39. (CA3, n3) SP
spa.3=pg.gamesp(n=3, X=1:10, k=c(2/6, 3/6, 4/6))

#K1=3/6, k2=k1+(1/6), k3=k2+(1/6)
#Gráfica 40. (CA4, n3) SP
spa.4=pg.gamesp(n=3, X=1:10, k=c(3/6, 4/6, 5/6))

#ESTRATEGIA GT

#Tomamos el escenario spa.3 para jugar la estrategia GT:

spa.3.grim=solve.game(set.to.grim.trigger(spa.3))
spa.3.grim$opt.mat
spa.3$opt.mat
stargazer(spa.3.grim$opt.mat)
stargazer(spa.3$opt.mat)
#### Comparación

#Gráfica 41. Opt.vs.GT CP-CA n3 SP
spa3comp=plot.compare.models(spa.3, spa.3.grim, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")


########################################################################
########################################################################
########################################################################
################# C O M P O N E N T E  S T I C K  A N D  C A R R O T
#################
########################################################################
########################################################################
########################################################################
########################################################################
```

```
#Este sección trata sobre la Estrategia Stick and Carrot aplicada en el
#acceso al bien público paz. Es menester anticiparse al hecho que no se
#realizarían todas las simulaciones anteriores con esta estrategia por
#efectos prácticos. Tomaremos dos escenarios: Con polizones (CP) y sin
#polizones (SP). En cada escenario hay dos casos: Costos simétricos y
#costos asimétricos. Antes de ello, se deben llevar a cabo unos cambios
#en las funciones, puesto que las contribuciones deben ser simétricas.

## Cabe destacar lo prácticos que seremos a la hora del análisis, motivo
#por el cual escogeremos toda $x \in [0,10]$  con posibilidad de polizones,
#mientras que toda $x \in [1,10]$ sería para la condición sin polizones,
#espacios de contribuciones que previamente se han venido trabajando.
#Igualmente observaremos cómo difiere en el óptimo aplicar la estrategia
# Stick and Carrot con costos simétricos iguales a 0.75 y costos asimétricos
#con 0.6 y 0.75 para los agentes 1 y 2. Es pertinente manifestar que se
#tratan sólo dos jugadores, simulando una sociedad cuya población se
# divide en dos tipos de grupos acorde a sus costos.

## Podemos pensar esto como una sociedad con Clase Baja y Clase Alta.
#Para el caso de n=3, los efectos no cambian mucho en el análisis, puesto
#que por simplicidad se opta por n=2. Se deja abierto para el lector en
#caso que desee correr el modelo con n=3.
#########################################################################
#####################
##################### C O N  P O L I Z O N E S (CP)
#####################
##########################################################################

#Dos jugadores
public.goods.game = function(X, k1,k2=k1) {
store.objects("public.goods.game")
x1m = matrix(X,NROW(X),NROW(X),byrow=FALSE)
x2m = matrix(X,NROW(X),NROW(X),byrow=TRUE)
g1 = (x1m+x2m)/2 - k1*x1m
g2 = (x1m+x2m)/2 - k2*x2m
name = paste("Public Goods Game k1=", k1, " k2=", k2, sep="")
s = init.game(n=2,g1=g1,g2=g2,name=name, lab.ai=X, symmetric = T)
#En esta línea se simetriza introduciendo symetric=TRUE
s = solve.game(s, keep.only.opt.rows=T)
plot(s)
```

```
return(s)
}
restore.objects("public.goods.game")
x1m= matrix(X,NROW(X),NROW(X),byrow=!TRUE)
x2m = t(x1m)
g1 = (x1m+x2m)/2 - k1*x1m
g2 = (x1m+x2m)/2 - k2*x2m

#Costos simétricos
sncs=public.goods.game(X=0:10, k1=0.75, k2=0.75)
#Stick and Carrot
stick.cs.sp=solve.game(set.to.Abreu.stick.carrot(sncs))
stick.cs.sp$opt.mat
stargazer(stick.cs.sp$opt.mat)
#Comparación entre Opt y Stick and Carrot con costos simétricos
#Gráfica 42. OPT vs SnC. CS-CP
snc.opt.comp.cs=plot.compare.models(sncs,stick.cs.sp, xvar="delta",
yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "stick-n-carrot")


#Costos asimétricos
snca=public.goods.game(X=0:10, k1=0.6, k2=0.75)
#Stick and Carrot
stick.ca.sp=solve.game(set.to.Abreu.stick.carrot(snca))
stick.ca.sp$opt.mat
stargazer(stick.ca.sp$opt.mat)
#Comparación entre Opt y Stick and Carrot con costos simétricos
#Gráfica 43. OPT vs SnC CA-CP
snc.opt.comp.ca=plot.compare.models(snca,stick.ca.sp, xvar="delta",
yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "stick-n-carrot")

#########################################################################
###################
################### S I N  P O L I Z O N E S (SP)
###################
#########################################################################

public.goods.game = function(X, k1,k2=k1) {
store.objects("public.goods.game")
```

```
x1m = matrix(X,NROW(X),NROW(X),byrow=FALSE)
x2m = matrix(X,NROW(X),NROW(X),byrow=TRUE)
g1 = 2*log(x1m)+2*log(x2m) - k1*x1m
g2 = 2*log(x1m)+2*log(x2m) - k2*x2m
name = paste("Public Goods Game k1=", k1, " k2=", k2, sep="")
r = init.game(n=2,g1=g1,g2=g2,name=name, lab.ai=X, symmetric=T )
#Simetrizamos.
r = solve.game(r, keep.only.opt.rows=T)
plot(r)
return(r)}
restore.objects("public.goods.game")
x1m= matrix(X,NROW(X),NROW(X),byrow=!TRUE)
x2m = t(x1m)
g1 = 2*log(x1m)+2*log(x2m) - k1*x1m
g2 = 2*log(x1m)+2*log(x2m) - k2*x2m

#Costos Simétricos
srcs=public.goods.game(X=1:10, k1=0.75, k2=0.75)
#Stick And Carrot.
sc.sr.cs=solve.game(set.to.Abreu.stick.carrot(srcs))
sc.sr.cs$opt.mat
stargazer(sc.sr.cs$opt.mat)
#Comparación entre Opt y Stick and Carrot con costos simétricos
#Gráfica 43.OPT vs SnC CS-SP
sc.opt.comp.cs=plot.compare.models(srcs,sc.sr.cs, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "stick-n-carrot")


#Costos Asimétricos (CA)
sr=public.goods.game(X=1:10, k1=0.6, k2=0.75)
#Stick And Carrot.
sc.sr.ca=solve.game(set.to.Abreu.stick.carrot(sr))
sc.sr.ca$opt.mat
stargazer(sc.sr.ca$opt.mat)

#Comparación entre Opt y Stick and Carrot con costos simétricos
#Gráfica 44. OPT vs SnC CA-SP
sc.opt.comp.ca=plot.compare.models(sr,sc.sr.ca, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "stick-n-carrot")

##########################################################################
```

```
##############
############## C A S O  C O N  A G E N T E S  H E T E R O G É N E O S
##############
##########################################################################

# En esta sección nos preguntamos ¿Qué pasa cuando los agentes no son
# homogéneos? es decir, cuando los agentes no tienen las mismas
#funciones de pago o de utilidad. El trasfondo es el siguiente: Ya se sabe
# que se ha trabajado con agentes que diferían en sus costos o no, además
# de su comportamiento ante la contribución, lo que en otras palabras es
#haber trabajado con polizones y sin polizones, entre ellos con costos
#simétricos y asimétricos. La cuestión ahora es trabajar, caeteris paribus,
#los cambios de equilibrios ante una aplicación diferente en la función de
# los agentes.

#La lógica interna que alberga la situación, planteada en el contexto del
#bien público paz, se halla en el hecho de diferentes efectos de la guerra
#sobre las decisiones de los individuos. El deseo de la paz por parte de
#los principales afectados por la guerra no es el mismo que poseen los que
# no son afectados de forma directa. El supuesto es que los agentes que
#fueron directamente afectados por la guerra perciban el doble de
#utilidad que aquellos que no fueron afectados directamente por lo
#bélico. Puede pensarse, de esta forma que un campesino y su familia,
#desplazados por la violencia, reciben más felicidad por el hecho que l
#guerra culmine con respecto a una familia citadina, que en el diario
#vivir, perciben la guerra vía medios de comunicación. Sea "g1" el pago
#recibido por contribución de los agentes directamente afectados por la
#guerra, y sea "g2" los pagos de aquellos que han sido afectados de forma
#indirecta por la guerra.

## Se sabe que g1=2g2. Se asume que la contribución no es obligatoria,
#por más mínimo que sea, (posibilidad de polizones)  y habrá dos tipos de
#situaciones: Costos simétricos (los costos de realizar la contribución son
# los mismos sin importar la situación ante la guerra) o asimétricos (los
#costos de la contribución son distintos).
#   Se asume que el costo de contribución, ante simetría, corresponde a
#3/4 de la contribución. Y ante costos asimétricos, se asumiría que la
#diferencia radicaría entre 3/5 y 3/4. Los escenarios serían 2  (los primeros
#que se han trabajado: E.N y estrategia GT).

## LA FUNCIÓN:
```

```
public.goods.game = function(X, k1,k2=k1) {
store.objects("public.goods.game")
x1m = matrix(X,NROW(X),NROW(X),byrow=FALSE)
x2m = matrix(X,NROW(X),NROW(X),byrow=TRUE)
g1 = 2*((x1m+x2m)/2 - k1*x1m)
g2 = (x1m+x2m)/2 - k2*x2m
name = paste("Public Goods Game k1=", k1, " k2=", k2, sep="")
h = init.game(n=2,g1=g1,g2=g2,name=name, lab.ai=X, symmetric = T)
h = solve.game(h, keep.only.opt.rows=T)
plot(h)
return(h)
}
restore.objects("public.goods.game")
x1m= matrix(X,NROW(X),NROW(X),byrow=!TRUE)
x2m = t(x1m)
g1 = 2*((x1m+x2m)/2 - k1*x1m)
g2 = (x1m+x2m)/2 - k2*x2m

## EL JUEGO:
### Costos Simétricos
#OPT
h = public.goods.game(X=0:10,k1=0.75,k2=0.75)
h$opt.mat
stargazer(h$opt.mat)
#GT
hgt=solve.game(set.to.grim.trigger(h))
hgt$opt.mat
stargazer(hgt$opt.mat)

#Gráfica 45. OPT vs GT CS-CP heterogéneo
comph=plot.compare.models(h, hgt, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")


### Costos Asimétricos
#OPT
hs = public.goods.game(X=0:10,k1=0.6,k2=0.75)
hs$opt.mat
stargazer(hs$opt.mat)
#GT
```

```
hsgt=solve.game(set.to.grim.trigger(hs))
hsgt$opt.mat
stargazer(hsgt$opt.mat)
#Gráfica 46. OPT vs GT CA-CP heterogéneo
comphs=plot.compare.models(hs, hsgt, xvar="delta", yvar="Ue",
legend.pos = "topleft", m1.name = "opt", m2.name = "grim")

#######################################################################
#################
################# F     I     N
#################
#######################################################################
```

Bibliografía

- Abreu, D. (2 de March de 1988). On the Theory of Infinitely Repeated Games with Discounting. ECONOMETRICA, Journal of the Econometric Society, 56(2), 383-396. Obtenido de http://www.jstor.org/stable/1911077

- Kranz, S. (2012 de March de 27). Interactively Solving Repeated Games: A Toolbox. Obtenido de https://www.uni-ulm.de/fileadmin/website_uni_ulm/mawi.inst.160/pdf_dokumente/mitarbeiter/kranz/Interactively_Solving_Repeated_Games.pdf

- Downs, A. (1957). An Economic Theory of Political Action in a Democracy. The Journal of Political Economy.

- Friedman, C. K. (2014). Continuous Differentiation: Hotelling Revists the Lab.

- Kranz, S. (03 de November de 2014). Github. Obtenido de Skranz-repgame https://github.com/skranz/repgame/blob/master/inst/examples_part1.r

- Gibbons, R. (1992). Un Primer Curso de Teoría de Juegos. United States: Antony Bosch.