

# TP

July 4, 2022

## 1 Difusión sobre el Product Space

En este trabajo implementaremos un modelo de difusión sobre el Product Space, haciendo uso del formalismo EB-DEVS. Utilizaremos los mismos datos que en los trabajos originales (extraídos de Harvard Dataverse) y exploraremos qué ocurre con distintas dificultades de difusión (valores de  $\Omega$ ) y simularemos tanto con criterio de retroalimentación global como sin éste.

### 1.1 Introducción

#### 1.1.1 Introducción al modelo Product Space

Diremos que un país  $c$  exporta un producto  $p$  cuando  $RCA_{c,p} = \frac{\$ \text{ de } p \text{ exportados por } c}{\$ \text{ promedio de exportaciones de } p} > 1$ .

Para simplificar la notación  $x_p^c = 1$  si  $RCA_{c,p} > 1$  (y 0 sino).

Luego se define la similitud entre productos de la siguiente manera:

$$\varphi_{i,j} = \min\{P(RCA_i|RCA_j), P(RCA_j|RCA_i)\} = \sum_c x_i^c x_j^c \times \min\left\{\frac{1}{\sum_c x_i^c}, \frac{1}{\sum_c x_j^c}\right\}$$

con  $\$ P(RCA\_i, RCA\_j) = \frac{\sum_c x_i^c x_j^c}{\sum_c x_i^c \sum_c x_j^c} ; / ; \# \text{países } \$$ .

Notar que la matrix que resulta  $\Phi = \{\varphi_{i,j}\}_{i,j}$  depende del  $x$  de todos los países y productos.

#### 1.1.2 Introducción de variable temporal y progreso de la difusión

Si bien el Product Space observa versiones estáticas de las variables mencionadas arribas, nosotros buscamos agregarle tiempo y progreso. La dimensión temporal resulta natural arregarla a  $x_{p,t}^c$  para todo  $p$  y  $c$ . Esto a su vez deriva las versiones temporales de  $\varphi_{i,j}^t$  y  $\Phi^t$ .

Para el concepto de progreso establecemos ciertas reglas:

- Sea  $\Omega$  el umbral de complejidad que se debe cruzar para desarrollar un producto.
- Si  $x_{p,t}^c = 0$  entonces la única manera de que el producto sea exportado es si  $\Omega < \max_{i/x_{i,t}=1} \varphi_{p,i}$ . Resultando en  $x_{p,t+1}^c = 1$ .
- Si para  $t$   $x_{p,t}^c = 1$  entonces  $\forall t' > t$   $x_{p,t'}^c = 1$ . O sea, una vez exportado un bien por un país, este siempre lo exporta por el resto de la simulación.

#### Simplificaciones cuantísticas

- Llamaremos  $\vec{d}_p^c$  al vector que representa las distancias entre un país  $c$  y un producto  $p$ , cuya definición es  $\vec{d}_p^c = \max_{j/x_j^c=1} \varphi_{p,j}$ .

- O de forma matricial  $\vec{d}^c = \max \text{fila}(\Phi X^c)$  con

$$X^c = \begin{bmatrix} x_1^c & 0 & \dots & 0 \\ 0 & x_2^c & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & x_n^c \end{bmatrix}$$

Notar que  $\vec{d}_p^{c,t}$  también depende del tiempo.

### 1.1.3 Variantes: retroalimentación vs local

Como mencionamos, simularemos dos variantes distintas, una con retroalimentación de los países y sus descubrimientos al resto del mundo y otra en la que el desarrollo se da de forma aislada en cada país y la complejidad de producir no cambia en función del tiempo.

- Variante global (o con retroalimentación): se basa principalmente en actualizar  $\Phi^{t+1}$  con los datos de  $x_{p,t}^c$  y luego que  $\vec{d}_p^{c,t+1}$  utilice  $\Phi^{t+1}$ .

Esta es la versión más sensata desde el punto de vista de la realidad, ya que suponemos que los avances y desarrollos que se dan en el mundo globalizado en que vivimos afectan la complejidad de producir. Pero como no removemos productos de las canastas de exportaciones, equivaldría a creer que la complejidad expresada en  $\Phi$  siempre se reduce, cosa que por crisis, faltantes, pérdidas de competitividad, etc. es un poco naïve.

- Variante local (o sin retroalimentación): se basa en hacer que  $\Phi$  no dependa del tiempo, solamente se tiene en cuenta la  $\Phi$  inicial, o sean las relaciones entre los países y sus exportaciones en el instante inicial. Luego cada país progresa mediante la actualización de  $X_t^c$ . De esta forma los avances son aislados entre los países y productos.

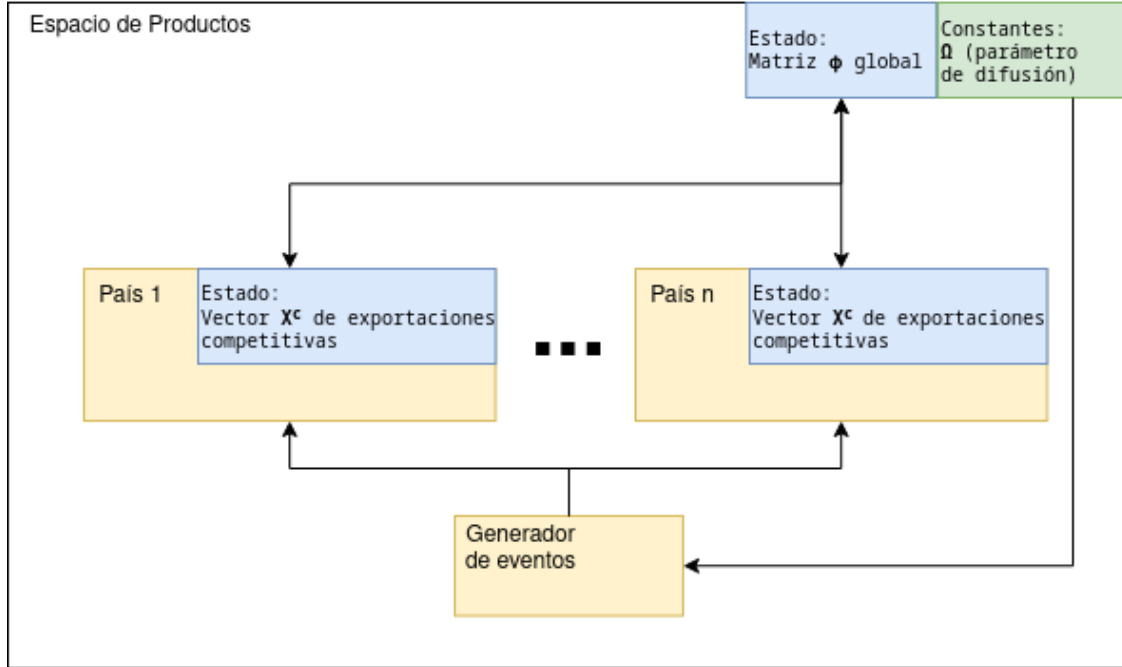
En el trabajo intentaremos ver si estos métodos tienen diferencias y en caso de que las tuvieran por qué surgen.

## 1.2 Modelo Conceptual

Contamos con dos tipos de modelos atómicos:

- Los países, que tienen noción de sus exportaciones competitivas y realizan el proceso de difusión local.
- El generador de eventos, que una vez por unidad de tiempo manda un mensaje a cada país para que difunda. En este mensaje, informa a los países del valor de  $\Omega$ , que toma del acoplado.

Ambos atómicos se sitúan en el acoplado de Espacio de Productos, que tiene conocimiento de la matriz  $\Phi$  (macro estado). La  $\Phi$  se actualizará con los cambios en el micro estado de los países si se usa el modelo con retroalimentación.



### 1.3 Simulación

Implementamos el modelo usando EB-DEVS, lo que permite que el acoplado de Espacio de Productos tenga el macro estado de la simulación, y que este sea accesible por los atómicos para afectar su funcionamiento. Además, en el caso con retroalimentación, permite que el acoplado actualice su estado con los cambios en el micro estado de los atómicos.

#### 1.3.1 Generación de datos

Utilizamos los datos de comercio internacional [publicados por el Growth Lab de la Universidad de Harvard](#), en formato SITC Rev. 2 con códigos de producto de cuatro dígitos.

De este dataset, tomamos el RCA (que viene precalculado) para cada producto en un país y año, y a partir de estos datos generamos las matrices  $X$  de cada año.

#### 1.3.2 Cómo simular

Para correr la simulación, se requiere Python3 con EB-DEVS y numpy instalado. Se puede correr el script `main.py` con los siguientes argumentos:

```
--duration, -d: cantidad de interacciones de difusión a realizar
--big-omega, -O: valor del parámetro de difusión (por defecto, 0.55)
--metrics-folder, -m: carpeta o directorio en el que guardar las métricas
--phi-matrix-update, -u: booleano que define si se utiliza el modelo con retroalimentación (por defecto, True)
--X-matrices-file, -f: datos de entrada, en formato pkl, que contiene las matrices  $X$  (por defecto, None)
--year, -y: año de origen de la simulación, define con qué matriz  $X$  se empieza (por defecto, 2015)
```

El simulador genera dos archivos de salida con métricas:

- `omega.csv`: máximo, mínimo y promedio del vector  $\vec{d}^{c,t}$  de un país en cada generación/iteración

- `exports.csv`: cantidad de productos exportados por un país en cada generación/iteración

## 1.4 Análisis

Simulamos estas combinaciones

### Carga de datos

```
[9]: import pandas as pd
import plotly.express as px
import numpy as np
from plotly.subplots import make_subplots
import plotly.graph_objects as go
import plotly.offline as py
py.init_notebook_mode(connected=False)
# import plotly.io as pio
# pio.renderers.default = 'pdf'

UNINTRESTING_OMEGAS = (0.1, 0.2, 0.3, 0.4, 0.6, 0.65, 0.8, 0.9)
INTRESTING_OMEGAS = (0.5, 0.55)

## CARGA DE DATOS PARA MEDIR DIFERENCIAS ENRTRE CRITERIOS
exports_diff_dfs = []
exports_global_dfs = []
exports_local_dfs = []
for omega in list(UNINTRESTING_OMEGAS) + list(INTRESTING_OMEGAS):
    exports_global_df = pd.read_csv(f"experimentacion/metrics-true-{omega}/
↳exports.csv")
    exports_global_df["omega"] = omega
    exports_global_df.set_index(["country", "generation"], inplace=True)
    exports_global_dfs.append(exports_global_df)

    exports_local_df = pd.read_csv(f"experimentacion/metrics-false-{omega}/
↳exports.csv")
    exports_local_df["omega"] = omega
    exports_local_df.set_index(["country", "generation"], inplace=True)
    exports_local_dfs.append(exports_local_df)

    exports_diff_df = exports_global_df - exports_local_df
    exports_diff_df["omega"] = omega

    exports_diff_dfs.append(exports_diff_df)

exports_diff_df = pd.concat(exports_diff_dfs)
exports_diff_df.sort_values("omega", inplace=True)
```

```

exports_global_df = pd.concat(exports_global_dfs)
exports_global_df.sort_values("omega", inplace=True)

exports_local_df = pd.concat(exports_local_dfs)
exports_local_df.sort_values("omega", inplace=True)

```

#### 1.4.1 Cantidad de generaciones necesarias para alcanzar estabilidad

Como veremos a continuación, si bien hicimos 20 generaciones de difusión, con 10 ya sería suficiente. En muchos de los gráficos subsiguientes solo se muestran hasta 10 generaciones por motivos de claridad ya que los comportamientos no cambiarían en las generaciones mayores.

```

[10]: paises = ["ARG", "BRA", "CHL", "AUS"]
      # graficar las dos curvas
      # TODO: fijar los paises
      paises = set(paises + [country for country, _ in np.random.
        ↪choice(exports_local_df.index, 6 - len(paises))])

      # Criterio local
      df = exports_local_df.loc[sorted(paises)].reset_index().
        ↪sort_values(["country", "generation", "omega"])
      fig = px.line(df,
        x="generation",
        y="count_exports",
        color="omega",
        facet_col="country",
        facet_col_wrap=2,
        height=600,
        title="Exportaciones en función de la generación por país y omega
        ↪para criterio local",
        labels={"count_exports": "Exportaciones", "generation": "
        ↪Generación", "country": "País"})
      fig.for_each_annotation(lambda a: a.update(text=a.text.replace("=", " = ")))
      fig.add_vline(x=10, annotation_text="Fin sugerido de la simulación",
        ↪annotation_position="bottom right")
      fig.add_hline(y=df.count_exports.max(), annotation_text="Máximas
        ↪exportaciones", opacity=.25)
      fig.show()

      # Criterio global
      df = exports_global_df.loc[sorted(paises)].reset_index().
        ↪sort_values(["country", "generation", "omega"])
      fig = px.line(df,
        x="generation",
        y="count_exports",
        color="omega",
        facet_col="country",

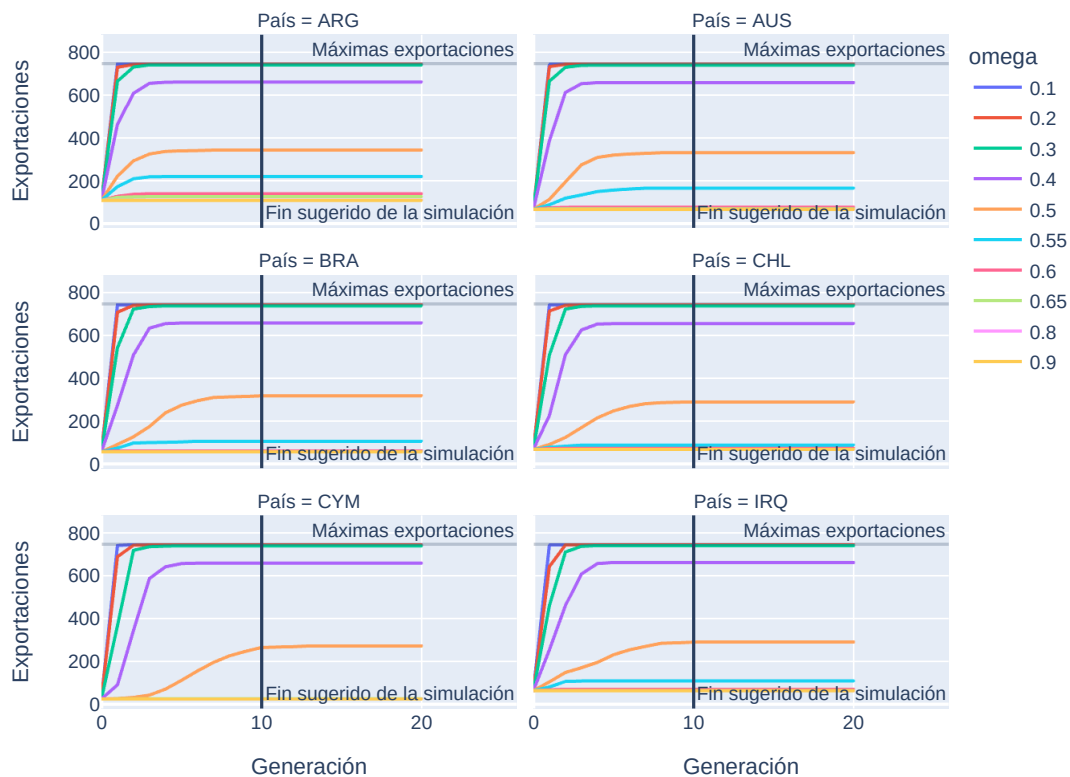
```

```

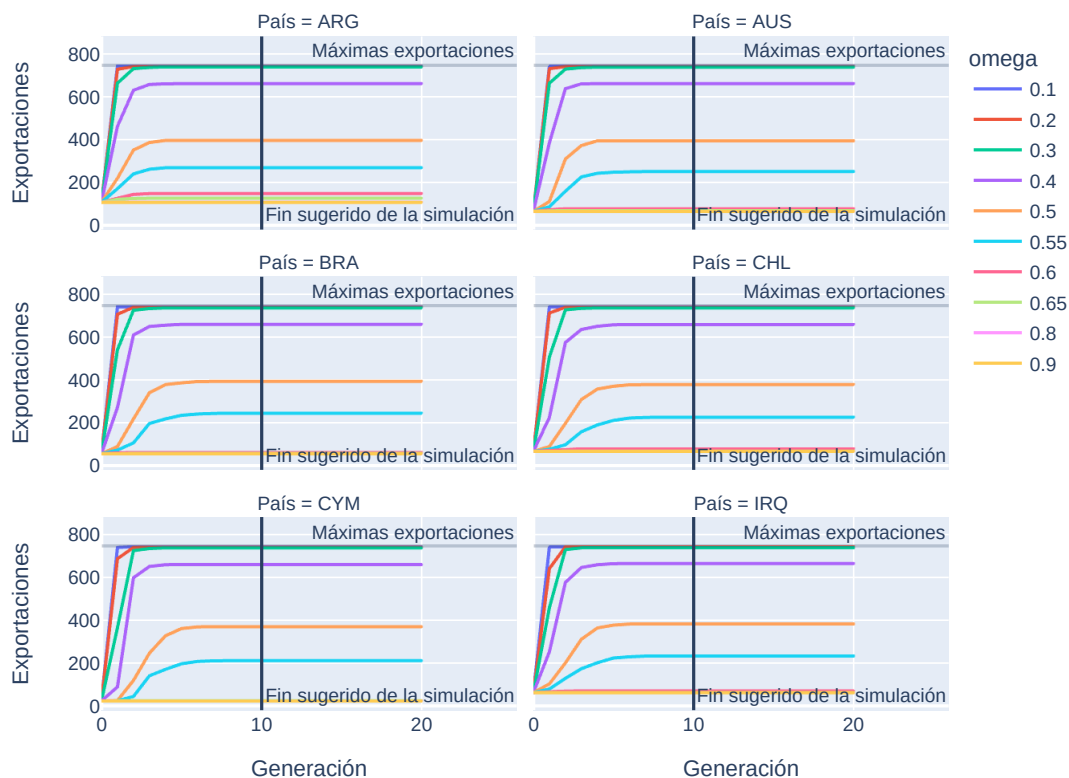
facet_col_wrap=2,
height=600,
title="Exportaciones en función de la generación por país y omega_
↳para criterio global",
labels={"count_exports": "Exportaciones", "generation": "
↳"Generación", "country": "País"})
fig.for_each_annotation(lambda a: a.update(text=a.text.replace("=", " = ")))
fig.add_vline(x=10, annotation_text="Fin sugerido de la simulación",
↳annotation_position="bottom right")
fig.add_hline(y=df.count_exports.max(), annotation_text="Máximas_
↳exportaciones", opacity=.25)

```

Exportaciones en función de la generación por país y omega para criterio local



## Exportaciones en función de la generación por país y omega para criterio global



Como podemos observar, luego del criterio de corte en la décima generación, no ocurre nada de interés.

Por lo tanto, las gráficas posteriores tendrán solo 10 generaciones, también es un criterio útil para simular menos generaciones y obtener resultados similares.

Ya que si bien el costo del simulador actualmente es más bien numérico (y por lo tanto se simula “rápido”), al hacerle modificaciones en el futuro en el que se le agrega inteligencia o dificultad para exportar esto podría cambiar y ejecutar menos generaciones resultar ventajoso.

```
[11]: # corte de dataframes

def filter_large_generations(index):
    return [(country, generation) for country, generation in index if
    ↪ generation > 10]

exports_diff_df.drop(index=filter_large_generations(exports_diff_df.index),
    ↪ inplace=True)
```

```

exports_global_df.drop(index=filter_large_generations(exports_global_df.index),
↳inplace=True)
exports_local_df.drop(index=filter_large_generations(exports_local_df.index),
↳inplace=True)

```

#### 1.4.2 Diferencias de exportaciones por omega y criterio de simulación

```

[12]: exports_global_df_copy = exports_global_df.copy()
exports_global_df_copy["local"] = False
exports_local_df_copy = exports_local_df.copy()
exports_local_df_copy["local"] = True
exports_df = pd.concat([exports_local_df_copy, exports_global_df_copy])

```

Veamos que ocurre para los países que acabamos de analizar en función de cada criterio de simulación, en línea punteada las exportaciones para las simulaciones globales y en línea común las exportaciones para las simulaciones sin actualización de  $\Phi$ .

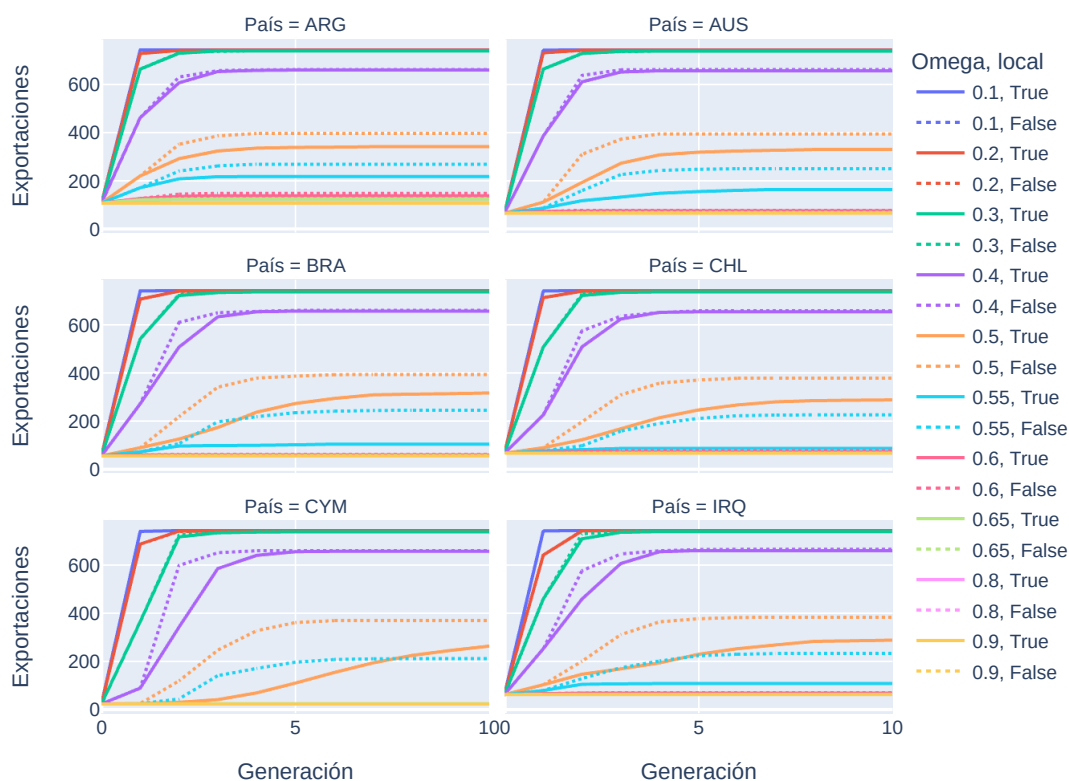
```

[13]: df = exports_df.loc[sorted(países)].reset_index().
↳sort_values(["country", "generation", "omega"])
fig = px.line(
    df,
    x="generation",
    y="count_exports",
    color="omega",
    line_dash="local",
    facet_col="country",
    facet_col_wrap=2,
    height=600,
    title="Exportaciones por país y criterio de simulación",
    labels={"count_exports": "Exportaciones", "generation": "Generación",
↳"country": "País", "omega": "Omega"},
)
fig.for_each_annotation(lambda a: a.update(text=a.text.replace("=", " = ")))
fig.show()

```



## Exportaciones por país y criterio de simulación



Se observan claras diferencias entre las para cada  $\Omega$ , en algunos casos ( $\Omega$  medios) la ventaja en exportaciones para las simulaciones globales son muy significativas, mientras que para aquellos umbrales muy altos o muy bajos (muy complejo o muy simple), no parece haber diferencia. Recordar que a medida avanza la simulación, dado que es un proceso de difusión, se aproximan asintotas sobre las metricas observadas

Para tener una visión más clara, veamos la diferencia absoluta entre las exportaciones por tipo de simulación.

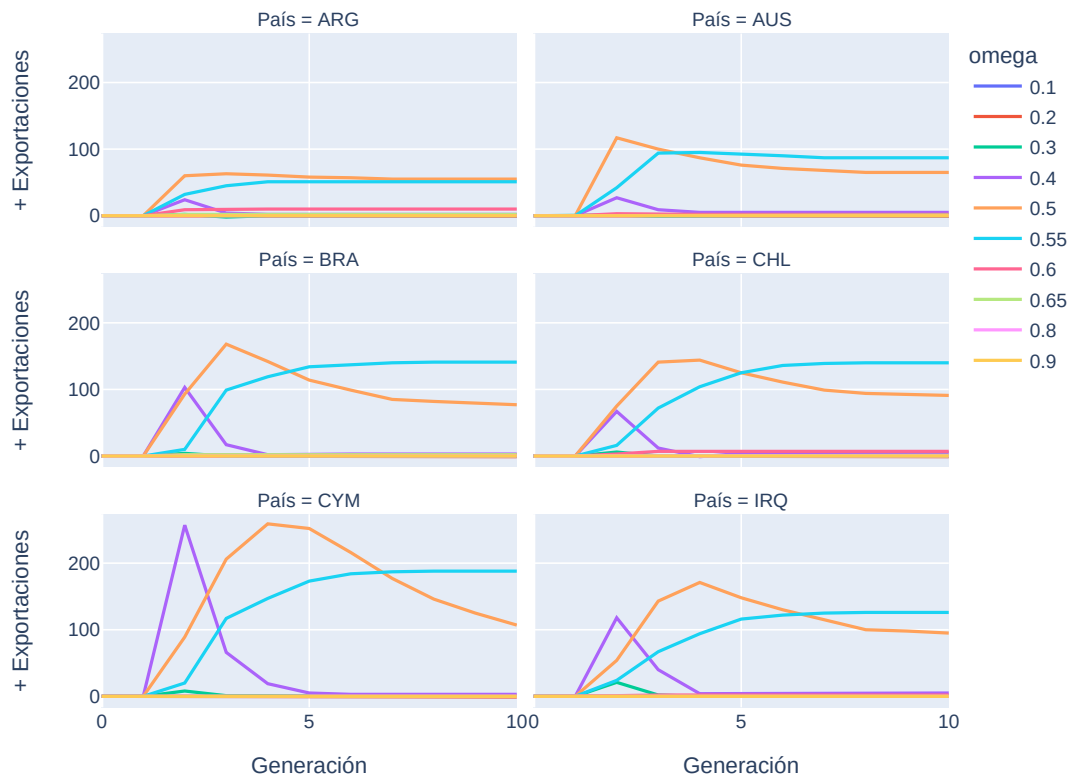
```
[14]: df = exports_diff_df.loc[sorted(países)].reset_index().
      ↪ sort_values(["country", "generation", "omega"])
fig = px.line(df,
              x="generation",
              y="count_exports",
              color="omega",
              facet_col="country",
              facet_col_wrap=2,
              height=600,
```

```

        title="Cantidad de exportaciones extra del criterio global respecto
↪del local",
        labels={"count_exports": "+ Exportaciones", "generation": "
↪Generación", "country": "País"}
    )
fig.for_each_annotation(lambda a: a.update(text=a.text.replace("=", " = ")))
fig.show()

```

Cantidad de exportaciones extra del criterio global respecto del local



Se observa que, para todos los países, la diferencia se estabiliza en las últimas generaciones, por lo que tomaremos las comparativas con la menor divergencia (la última generación disponible).

Veamos con mayor detalle como es la distribución según  $\Omega$  (umbral de “descubrimiento”).

```

[15]: max_diff_df = (
    exports_diff_df
    .reset_index()
    .set_index(["generation", "country", "omega"])
    .loc[max(exports_diff_df.index.get_level_values(1))]

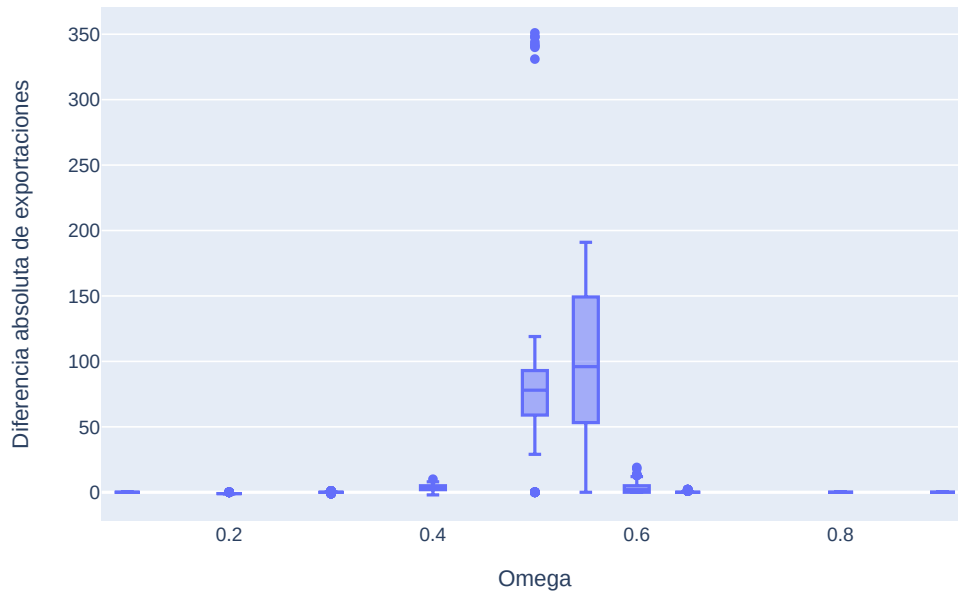
```

```

        .max(axis=1)
    )
    max_diff_df.rename("difference", inplace=True)
    fig = px.box(
        max_diff_df.reset_index(),
        x="omega",
        y="difference",
        title=f"Distribucion de la ventaja del criterio global respecto al criterio_
        ↪ local según omega para todos los países (ultima generacion)",
        labels={"difference": "Diferencia absoluta de exportaciones", "omega": "
        ↪ Omega"})
    fig.show()

```

Distribucion de la ventaja del criterio global respecto al criterio local según omega para t



Claramente los más interesantes de analizar son los valores de  $\Omega = 0.5$  y  $\Omega = 0.55$ . Esto se condice con la idea de que a valores muy pequeños de  $\Omega$  el descubrimiento es fácil y a valores muy elevados es muy difícil, por lo que las alternativas globales y locales pecan en los mismos aspectos.

La elección de  $\Omega$  resultaría más fuerte en la simulación que actualizar  $\Phi$ .

### Local vs global para $\Omega$ poco interesante

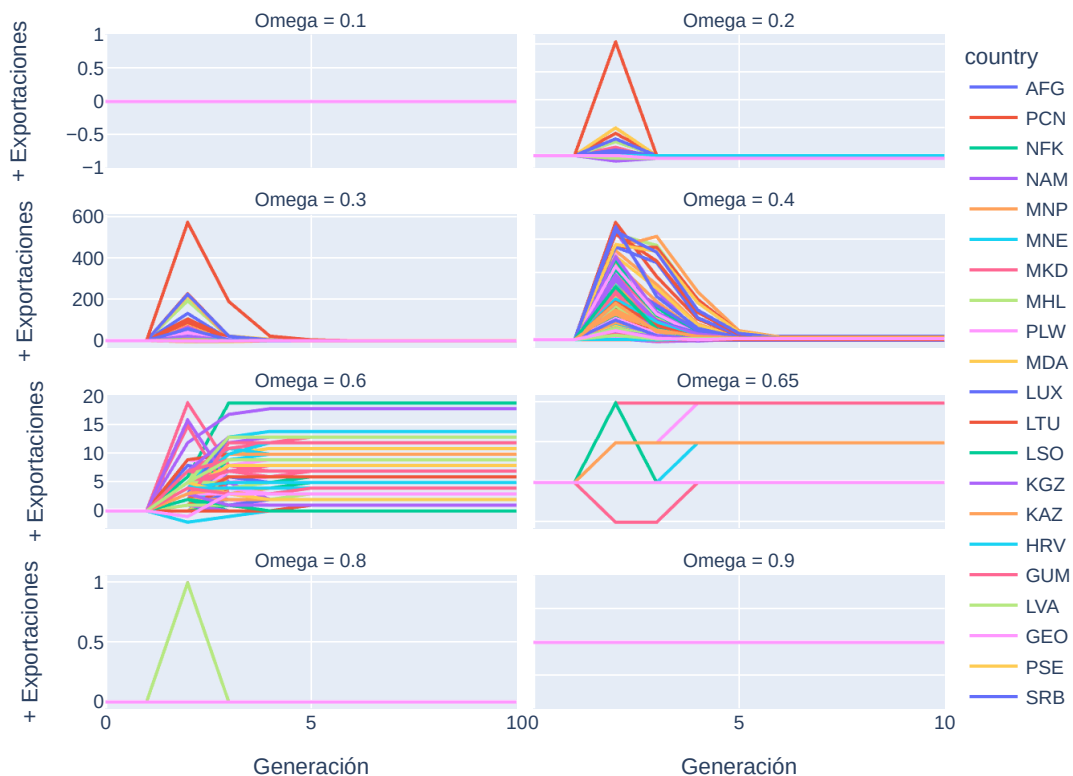
[16]: *# diferencia total en la cantidad de exportaciones de los omegas*

```

df_ = exports_diff_df[exports_diff_df.omega.isin(UNINTRESTING_OMEGAS)].
↳reset_index().sort_values(["omega", "generation"])
fig = px.line(df_,
              color="country",
              x="generation",
              y="count_exports",
              facet_col="omega",
              facet_col_wrap=2,
              height=600,
              title="Cantidad de exportaciones extra del criterio global respecto del_
↳local para todos los países",
              labels={"count_exports": "+ Exportaciones", "omega": "Omega",
↳"generation": "Generación"})
fig.for_each_annotation(lambda a: a.update(text=a.text.replace("=", " = ")))
fig.update_yaxes(matches=None)
fig.update_xaxes(matches=None)
fig.show()

```

Cantidad de exportaciones extra del criterio global respecto del local para todos los países



Aclaración: los ejes  $y$  no mantienen escala.

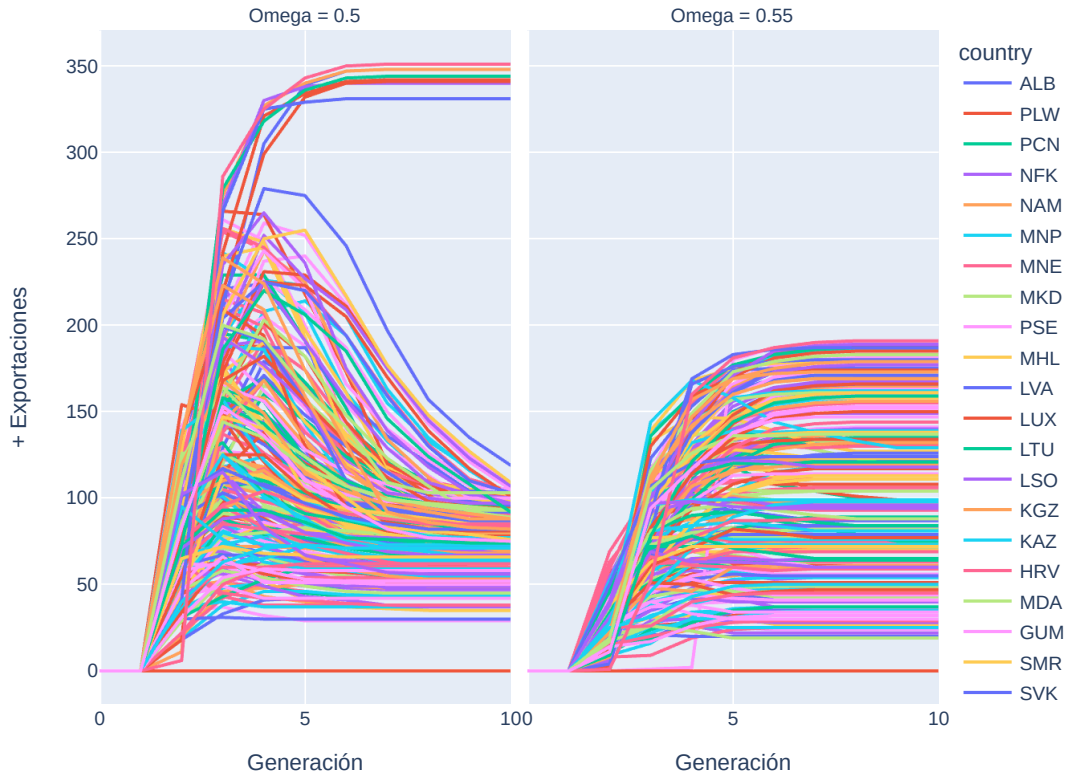
Como vemos, no hay cambios de interés fuera de las primeras generaciones para ninguno de los  $\Omega$  elegidos, por lo que para  $\Omega = 0.1, 0.2, 0.3, 0.4, 0.6, 0.65, 0.8, 0.9$  concluimos que las versiones locales y globales son indistinguibles.

Notar que para  $\Omega = 0.6$  la máxima diferencia es de 20 productos en la última generación, respecto de una canasta total de 700.

### Local vs global para $\Omega$ interesante

```
[17]: df_ = exports_diff_df[exports_diff_df.omega.isin(INTRESTING_OMEGAS)].
      ↪reset_index().sort_values(["omega", "generation"])
fig = px.line(df_,
              color="country",
              x="generation",
              y="count_exports",
              facet_col="omega",
              facet_col_wrap=2,
              height=600,
              title="Cantidad de exportaciones extra del criterio global respecto del
      ↪local para todos los países",
              labels={"count_exports": "+ Exportaciones", "omega": "Omega",
      ↪"generation": "Generación"})
fig.for_each_annotation(lambda a: a.update(text=a.text.replace("=", " = ")))
fig.update_xaxes(matches=None)
fig.show()
```

Cantidad de exportaciones extra del criterio global respecto del local para todos los países



$\Omega = 0.5$  y  $0.55$  si muestran comportamientos significativamente distintos, por lo que no podemos concluir que la version local y la global tengan indistinguibles. Como lo que esta siendo graficado es global - local y la diferencia es positiva y mayor a 20 productos en la mayoria de los casos, interpretamos que la version global facilita la difusión.

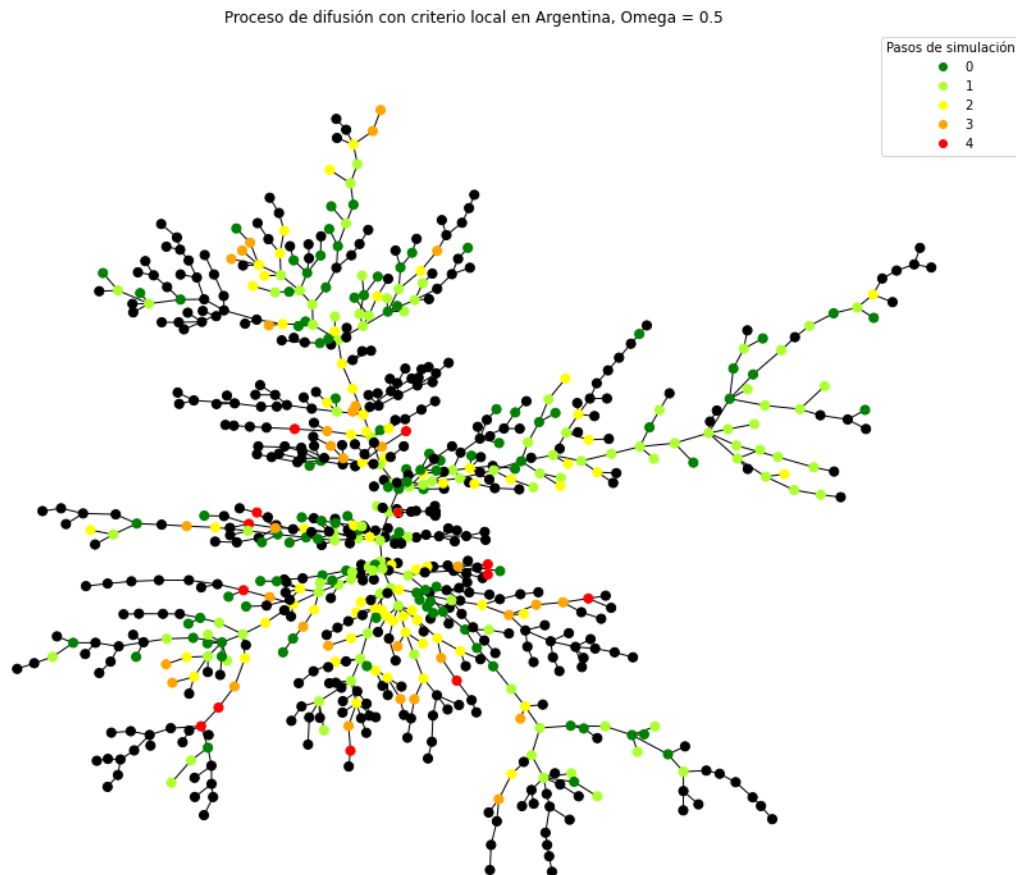
Explicamos el comportamiento indistinguible de los casos clasificados como  $\Omega$  poco interesante ya que para estos valores de  $\Omega$  la difusión suele alcanzar máximos por ser muy fácil difundir o caso contrario, demasiado difícil, entonces la retroalimentacion de  $\Phi$  no aporta el suficiente peso como para contrarrestar los efectos negativos en el desarrollo de nuevos productos.

Para terminar, mostraremos las diferencias en los productos desarrollados con ambos modelos. Graficamos el árbol generador máximo del espacio de productos mundial previo a la simulación, y coloreamos los nodos que son desarrollados en distinta cantidad de iteraciones de simulación. Usamos como ejemplo Argentina, con  $\Omega = 0.5$  y un máximo de 5 generaciones.

```
[18]: from metrics.product_space_graph import ProductSpaceGraph
graph_local = ProductSpaceGraph("experimentacion/full-local/exports_history.
    ↪pk1")
```

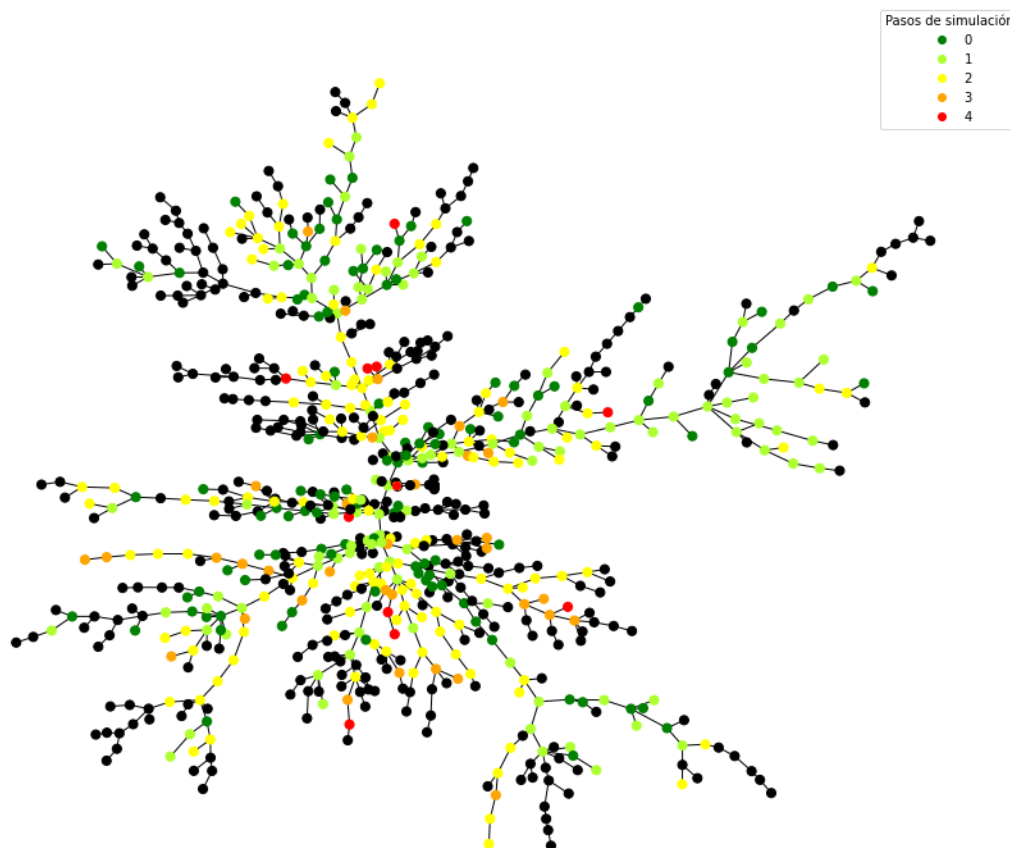
```
graph_global = ProductSpaceGraph("experimentacion/full-global/exports_history.  
↪pk1")
```

```
[19]: graph_local.plot("Proceso de difusión con criterio local en Argentina, Omega = 0.5  
↪0.5")
```



```
[20]: graph_global.plot("Proceso de difusión con criterio global en Argentina, Omega = 0.5")
```

Proceso de difusión con criterio global en Argentina,  $\Omega = 0.5$



Podemos observar que varios productos sólo se desarrollan en el modelo global. Además, otros productos se desarrollan en los dos modelos, pero requiere menos iteraciones en el global. Esto concuerda con lo que esperábamos a partir de lo observado en las figuras anteriores.

## 1.5 Conclusiones

En primer lugar corresponde marcar una diferencia muy importante en nuestros hallazgos y los papers originales: para  $\Omega = 0.5$  y  $0.55$  el modelo global y local muestran diferencias muy marcadas. Contrario a lo que se afirma en la bibliografía donde se asegura que son indistinguibles. No vemos ningún motivo particular para que nuestras conclusiones sean tan distintas, consideramos que el modelo de difusión replica fielmente al original (en la medida de la información disponible). Cabe destacar que esta comparación fue posible gracias a la posibilidad de modelar la retroalimentación entre micro y macro estado mediante EB-DEVS.

La herramienta desarrollada permite estudiar distintas métricas de redes, incluyendo las del trabajo original. Esperamos que esto permita realizar futuros trabajos más detallados, enfocados en países y/o productos particulares, para estudiar posibles estrategias de desarrollo.