

AUTOMATAS

1. Encuentre la partición correspondiente a la relación R y construya el grafo dirigido de la máquina cociente que corresponde a la máquina de Moore, cuya tabla de transición de estados se muestra a continuación:

	a	b	c	d	e	f	g	
0	a	g	f	a	a	g	g	$s_0 = a$
1	c	d	e	d	d	f	c	$T = \{d, e\}$

Figure 1:

Aqui el codigo en Python:

```
import networkx as nx
import matplotlib.pyplot as plt

transitions = {
    'a': {'0': 'a', '1': 'c'},
    'b': {'0': 'g', '1': 'd'},
    'c': {'0': 'f', '1': 'e'},
    'd': {'0': 'a', '1': 'd'},
    'e': {'0': 'a', '1': 'd'},
    'f': {'0': 'g', '1': 'f'},
    'g': {'0': 'g', '1': 'c'}
}

final_states = {'d', 'e'}

partitions = {
    'a': 'A', 'g': 'A', # [a, g]
    'b': 'B', 'f': 'B', # [b, f]
    'c': 'C',          # [c]
    'd': 'D', 'e': 'D'  # [d, e]
}

# Crear el grafo dirigido
G = nx.DiGraph()

# Agregar nodos del grafo (particiones)
G.add_nodes_from(['A', 'B', 'C', 'D'])

# Agregar transiciones del grafo cociente
for state, partition in partitions.items():
    for input_symbol, target_state in transitions[state].items():
        target_partition = partitions[target_state]
        G.add_edge(partition, target_partition, label=input_symbol)
```

```
# Dibujar el grafo
pos = nx.spring_layout(G)
labels = nx.get_edge_attributes(G, 'label')
nx.draw(G, pos, with_labels=True, node_size=2000, node_color='lightblue', font_size=16,
font_weight='bold')
nx.draw_networkx_edge_labels(G, pos, edge_labels=labels, font_size=16)
plt.title('Grafo de la Mquina Cociente')
plt.show()
```

Resultado Grafico del Código

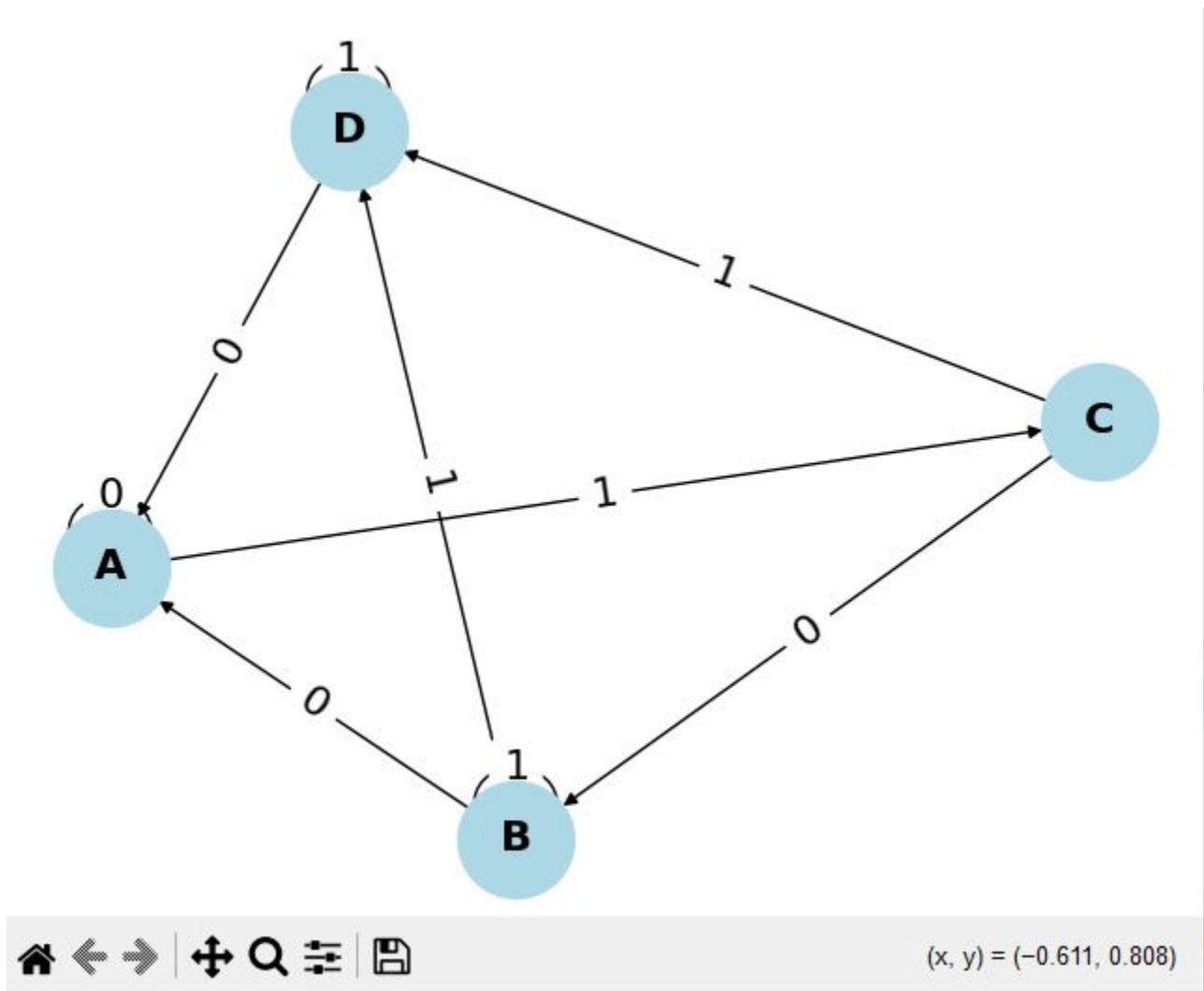


Figure 2: Resultado del grafo dirigido