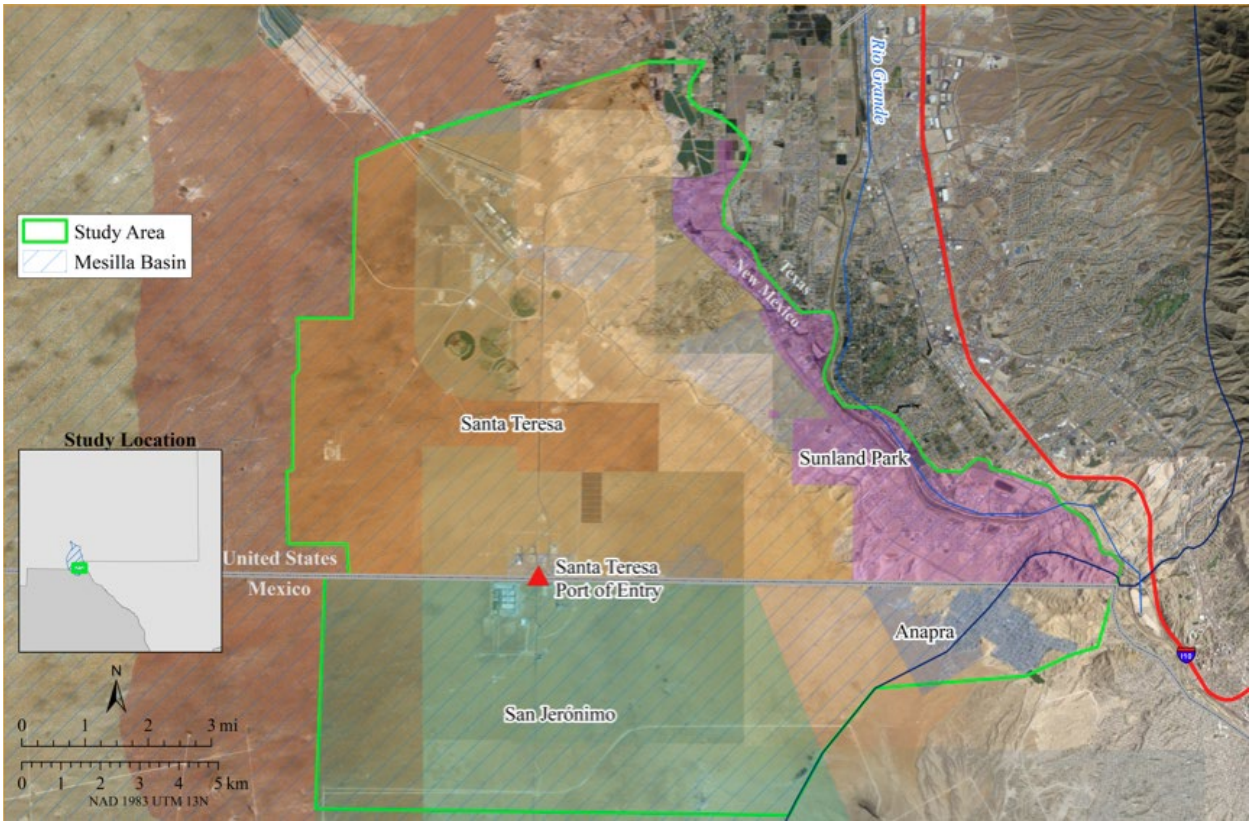
Supplementary Materials

1. Model Access

The model can be accessed online at the following address:
https://exchange.iseesystems.com/models/player/ashleyatkins/delays

2. Map of Study Site



A map of the study site, as it appeared in Page et al., "A Dynamic Hydro-Socio-Technical Policy Analysis of Transboundary Desalination Development," *Journal of Environmental Accounting and Management* 7, no. 1 (2019).

3. Model Documentation

		Equation	Properties	Units
	Top-Level Model:			
	Demand:			
○	consumption_demand	A.smooth_conservation_effect_on_use + B.consumption_demand		acre*ft/yr
○	withdrawal_demand	A.water_demand + B.withdrawal_demand		acre*ft/yr
	Desalination_Capacity:			
▭	Desalination_Capacity(t)	Desalination_Capacity(t - dt) + (desalination_capacity_development - desalination_plant_decay) * dt	INIT Desalination_Capacity = 0	acre*ft/yr
⁂	desalination_capacity_development	DELAY3(indicated_desalination_development, desalination_development_delay)		acre*ft/yr/Years
⁂	desalination_plant_decay	Desalination_Capacity / desalination_plant_life		acre*ft/yr/Years
○	cap_utilization_p	1		1
○	demand_gap	A.reported_demand_gap + B.reported_demand_gap		acre*ft/yr
○	desalination_capacity_utilization	MIN(1, 2 - 2 / ((MAX(0,demand_gap) / MAX(1, Desalination_Capacity))^cap_utilization_p + 1))		1
○	desalination_development_delay	15		yr
○	desalination_plant_life	10		yr
○	desired_addition_to_desal_capacity	(1 - strategic_thinking_weight) * demand_gap + strategic_thinking_weight * predicted_demand_gap		acre*ft/yr
○	gap_closure_time	1		yr
○	indicated_desalination_development	perceived_desalination_gap / gap_closure_time		acre*ft/yr/Years
◎	perceived_desalination_gap	SMTH1(desired_addition_to_desal_capacity, policy_perception_delay)		acre*ft/yr
○	policy_perception_delay	2		yr
○	predicted_demand_gap	FORCST(A.reported_demand_gap + B.reported_demand_gap, shortage_averaging_time, desalination_development_delay)		acre*ft/yr
○	shortage_averaging_time	10		yr
○	strategic_thinking_weight	0.5		1
	Water:			
▭	brackish_water(t)	brackish_water(t - dt) + (brackish_water_intrusion - desalination_rate) * dt	INIT brackish_water = 54e6	acre*ft
▭	Desalinated_Water(t)	Desalinated_Water(t - dt) + (desalination_rate - desal_water_withdrawal) * dt	INIT Desalinated_Water = 0	acre*ft
▭	Freshwater(t)	Freshwater(t - dt) + (fresh_water_inflow - freshwater_withdrawal - brackish_water_intrusion) * dt	INIT Freshwater = 6e5	acre*ft
▭	withdrawn_water(t)	withdrawn_water(t - dt) + (freshwater_withdrawal + desal_water_withdrawal - water_consumption) * dt	INIT withdrawn_water = 25000	acre*ft
⁂	brackish_water_intrusion	Freshwater * fr_fresh_water_intruded		acre*ft/yr
⁂	desal_water_withdrawal	(Demand.withdrawal_demand - freshwater_withdrawal) * desal_availability		acre*ft/yr
⁂	desalination_rate	Desalination_Capacity.Desalination_Capacity * Desalination_Capacity.desalination_capacity_utilization * brackish_water_availability		acre*ft/yr

	fresh_water_inflow	781		acre*ft/yr
	freshwater_withdrawal	Demand.withdrawal_demand * freshwater_availability		acre*ft/yr
	water_consumption	Demand.consumption_demand * stored_water_availability		acre*ft/yr
	brackish_avail_p[1]	.1		1
	brackish_avail_p[2]	.5		
	brackish_water_availability	(LN(MIN(1, brackish_water / INIT(brackish_water)) + 1 - brackish_avail_p[1]) / LN(2 - brackish_avail_p[1]))^brackish_avail_p[2]		1
	consumption_threshold	20000		acre*ft/yr
	desal_avail_p	1		1
	desal_availability	(2 - 2 / (MIN(1, (Desalinated_Water / desal_coverage_time) / MAX(1, Demand.withdrawal_demand - freshwater_withdrawal)) + 1))^desal_avail_p		1
	desal_coverage_time	1		yr
	fr_fresh_water_intruded	max_intrusion_fr - max_intrusion_fr / MAX(1, freshwater_withdrawal / consumption_threshold)^intrusion_p		1/yr
	freshwater_availability	(LN(MIN(1, Freshwater / INIT(Freshwater)) + 1 - fw_avail_p[1]) / LN(2 - fw_avail_p[1]))^fw_avail_p[2]		1
	fw_avail_p[1]	.1		1
	fw_avail_p[2]	.5		
	intrusion_p	1		1
	max_intrusion_fr	.5		1/yr
	stored_water_avail_p[1]	.1		1
	stored_water_avail_p[2]	.5		
	stored_water_availability	(LN(MIN(1, withdrawn_water / INIT(withdrawn_water)) + 1 - stored_water_avail_p[1]) / LN(2 - stored_water_avail_p[1]))^stored_water_avail_p[2]		1
	A:			
	normal_demand_gap(t)	normal_demand_gap(t - dt) + (normal_demand_gap_change) * dt	INIT normal_demand_gap = 0	acre*ft/yr
	normal_demand_gap_change	(indicated_normal_demand_gap - normal_demand_gap)		acre*ft/yr/Years
	conservation_effect	perceived_water_demand_gap / normal_demand_gap		1
	demand_gap	water_demand - water_use		acre*ft/yr
	indicated_normal_demand_gap	perceived_water_demand_gap		acre*ft/yr
	perceived_water_demand_gap	SMTH1(reported_demand_gap, perception_delay, 0)		acre*ft/yr
	perception_delay	2		yr
	reported_demand_gap	DELAY(demand_gap, reporting_delay, 0)		acre*ft/yr
	reporting_delay	1		yr
	smooth_conservation_effect_on_demand	conservation_effect		1
	smooth_conservation_effect_on_use	conservation_effect		acre*ft/yr
	water_demand	smooth_conservation_effect_on_demand		acre*ft/yr
	water_use	smooth_conservation_effect_on_use		acre*ft/yr
	B:			
	normal_demand_gap(t)	normal_demand_gap(t - dt) + (normal_demand_gap_change) * dt	INIT normal_demand_gap	acre*ft/yr

			= 0	
	normal_demand_gap_change	(indicated_normal_demand_gap - normal_demand_gap) / time_to_adjust_anchor		acre*ft/yr/Years
	conservation_effect	MAX(max_conservation, 1 - conservation_p1 * MAX(0, perceived_water_demand_gap / MAX(1, normal_demand_gap) - 1))		1
	conservation_effect_delay	3		yr
	conservation_p1	.5		1
	conservation_p2	.5		yr
	consumption	consumption_demand * Water.stored_water_availability		acre*ft/yr
	consumption_demand	SMTH1(normal_consumption_demand * conservation_effect, conservation_effect_delay)		acre*ft/yr
	demand_gap	withdrawal_demand - consumption		acre*ft/yr
	indicated_normal_demand_gap	perceived_water_demand_gap * (1 + normal_demand_gap_bias)		acre*ft/yr
	max_conservation	0.5		1
	normal_consumption_demand	19457.4*.2		acre*ft/yr
	normal_demand_gap_bias	0		1
	normal_withdrawal_demand	19457.4*.2		acre*ft/yr
	perceived_water_demand_gap	SMTH1(reported_demand_gap, perception_delay, 0)		acre*ft/yr
	perception_delay	2		yr
	reported_demand_gap	DELAY(demand_gap, reporting_delay, 0)		acre*ft/yr
	reporting_delay	1		yr
	smooth_conservation_effect	SMTH1(conservation_effect, smooth_delay)		1
	smooth_delay	5		yr
	time_to_adjust_anchor	10		yr
	withdrawal_demand	normal_withdrawal_demand * smooth_conservation_effect		acre*ft/yr

Run Specs	
Start Time	1
Stop Time	50
DT	1/10
Fractional DT	True
Save Interval	0.1
Sim Duration	1.4999978
Time Units	Years
Pause Interval	0
Integration Method	Euler
Keep all variable results	True
Run By	Run
Calculate loop dominance information	True
Exhaustive Search Threshold	1000

Array Dimension	Indexed by	Elements

two	Number	2

User Macro	Output
TRND(<INPUT>, <time to perceive trend>, <time to establish reference condition>, [<ITRND>])	TRND

		Equation	Properties	Units
	Perceived_Present_Condition_of_Input(t)	Perceived_Present_Condition_of_Input(t - dt) + ("dPPC/dt") * dt	INIT Perceived_Present_Condition_of_Input = INPUT / (1 + time_to_perceive_present_condition * ITRND)	
	Reference_Condition_of_Input(t)	Reference_Condition_of_Input(t - dt) + ("dRC/dt") * dt	INIT Reference_Condition_of_Input = Perceived_Present_Condition_of_Input / (1 + time_to_establish_reference_condition * ITRND)	
	TRND(t)	TRND(t - dt) + ("dTRND/dt") * dt	INIT TRND = ITRND	1/yr
	"dPPC/dt"	(INPUT - Perceived_Present_Condition_of_Input) / time_to_perceive_present_condition		
	"dRC/dt"	(Perceived_Present_Condition_of_Input - Reference_Condition_of_Input) / time_to_establish_reference_condition		
	"dTRND/dt"	(reference_trend_in_input - TRND) / time_to_perceive_trend		1/yr/Months
	INPUT	0		
	ITRND	0		1/yr
	reference_trend_in_input	(Perceived_Present_Condition_of_Input - Reference_Condition_of_Input) / MAX(1e-6, Reference_Condition_of_Input * time_to_establish_reference_condition)		
	time_to_establish_reference_condition	1		yr
	time_to_perceive_present_condition	0.2 * time_to_establish_reference_condition		yr
	time_to_perceive_trend	1		yr