Appendix C: Learning algorithms compared in terms of efficiency, equity and sustainability, and proportion of agents using each of the three strategies (i.e. HM=Hydraulic Mission; EF=Efficiency First; PR=Policy Reform) by the end of the simulation.

	Efficiency	Equity (Variance)	Sustainability	Proportion of agents using strategy
Learning1[a]	Total value: High; Rand/m3: Medium; Sectoral value: High	Allocation: Med-high Unmet demand: Low	Wta[h]: High PEMC[i]: High %Transformed: High	HM: 0.30 EF: 0.33 PR: 0.36
Learning2[b]	Total value: Low; Rand/m3: Medium Sectoral value: Low	Allocation: Low Unmet demand: Low	Wta: Low PEMC: Low %Transformed: Low	HM: 0.34 EF: 0.26 PR: 0.40
Learning3[c]	Total value: High; Rand/m3: Medium; Sectoral value: High	Allocation: High Unmet demand: Low	Wta: High PEMC: High %Transformed: High	HM: 0.30 EF: 0.40 PR: 0.30
Learning4[d]	Total value: Medium; Rand/m3: Med-high; Sectoral value: Med-low	Allocation: Low Unmet demand: Low	Wta: Low PEMC: Low %Transformed: Low	HM: 0.46 EF: 0.30 PR: 0.24
Learning5[e]	Total value: High; Rand/m3: Med; Sectoral value: High	Allocation: High Unmet demand: Low	Wta: High PEMC: High %Transformed: High	HM: 0.29 EF: 0.32 PR: 0.39
Learning6[f]	Total value: High; Rand/m3: Med; Sectoral value: Med-high	Allocation: High Unmet demand: Low	Wta: High PEMC: High %Transformed: High	HM: 0.33 EF: 0.34 PR: 0.33
Learning8[g]	Total value: Med; Rand/m3: High; Sectoral value: Low	Allocation: Low Unmet demand: Low	Wta: Low PEMC: Low %Transformed: Low	HM: 0.47 EF: 0.27 PR: 0.27

[a] Learning algorithm: allocateCollectiveLearningMeanUnmetDemand (threshold = 1.0). Finds mean unmet demand for each strategy and uses strategy that achieved minimum.
[b] Learning algorithm: allocateCollectiveLearningMaxAllocation (threshold = 1.0). Finds water unit that achieved maximum allocation in last timestep and uses that strategy.
[c] Learning algorithm: allocateCollectiveLearningMinUnmetDemand (threshold = 0.5). Finds water unit that achieved minimum unmet demand in last timestep and uses that strategy.
[d] Learning algorithm: allocateCollectiveLearningMaxAllocation (threshold = 0.5). Finds water unit that achieved maximum allocation in last timestep and uses that strategy.
[e] Learning algorithm: allocateCollectiveLearningMeanUnmetDemand (threshold = 0.5). Finds mean unmet demand for each strategy and uses strategy that achieved minimum.
[f] Learning algorithm: allocateCollectiveLearningProportionSatisfied; successThreshold:= 0.25 based on surveying proportion of agents who satisfied all demand (i.e. unmet demand = 0) for each strategy for all water units in CMA.
[g] Learning algorithm: allocateCollectiveLearningMaxAllocation (threshold = 0.75). Finds water unit that achieved maximum allocation in last timestep and uses that strategy. Also restricted export and transfer to water units with same strategy (note that transfer was not used in all other simulations)
[h] Withdrawal-to-availability ratio
[i] Present Ecological Management Class