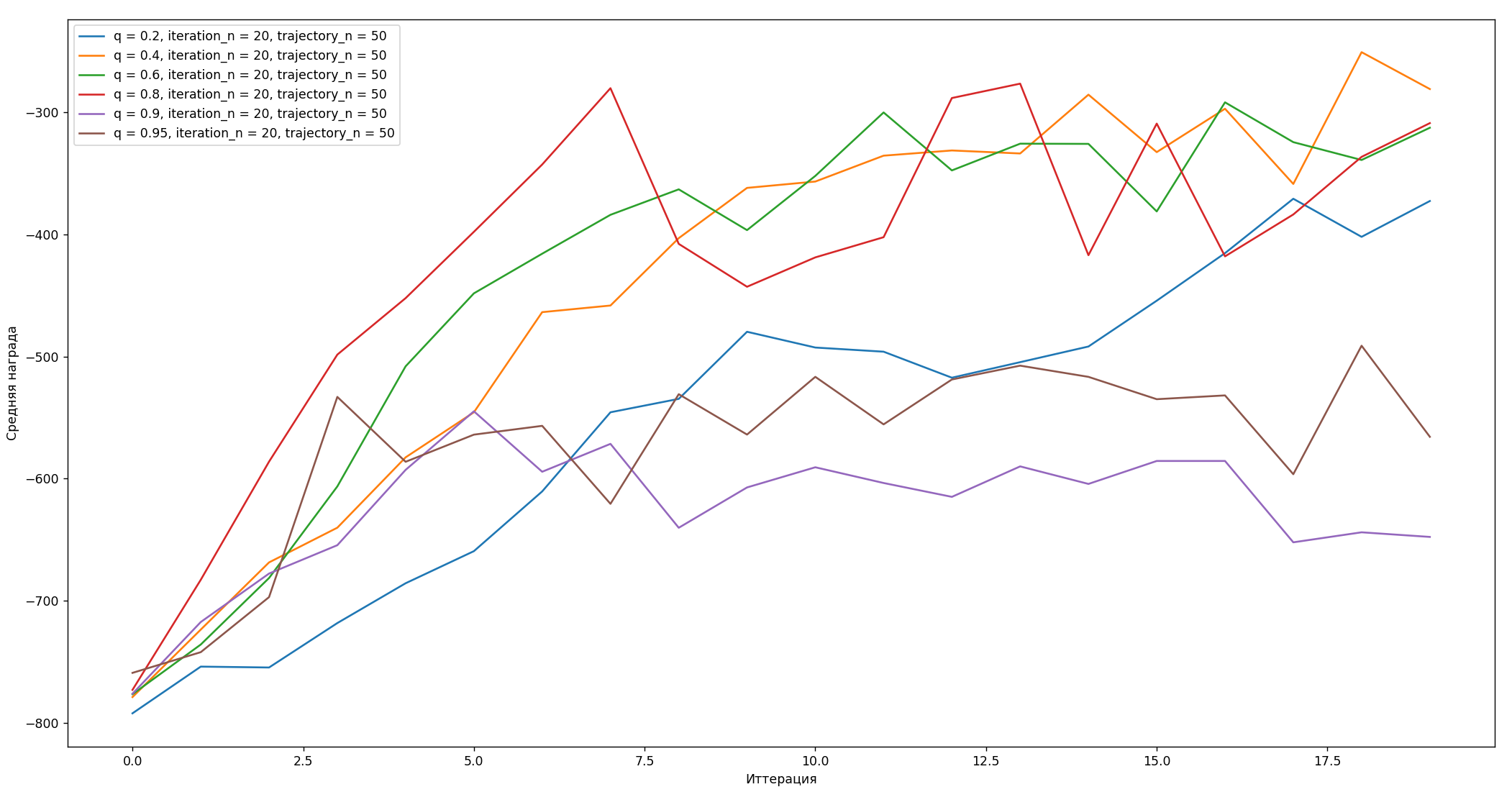
**Трапер Максим. СПБГУ. Магистратура 1 курс «Искусственный интеллект и наука о данных»**

**Задание №1:** пользуясь алгоритмом Кросс-Энтропии обучить агента решать задачу Taxi-v3 из Gym. Исследовать гиперпараметры алгоритма и выбрать лучшие.

Ввиду стохастичности начальных условий среды (местоположения такси и пассажира), обучение агента задаче происходит гораздо труднее.

**Эксперимент №1**: попробуем различные значения квантиля при количестве итераций = 20 и количестве траекторий = 50.

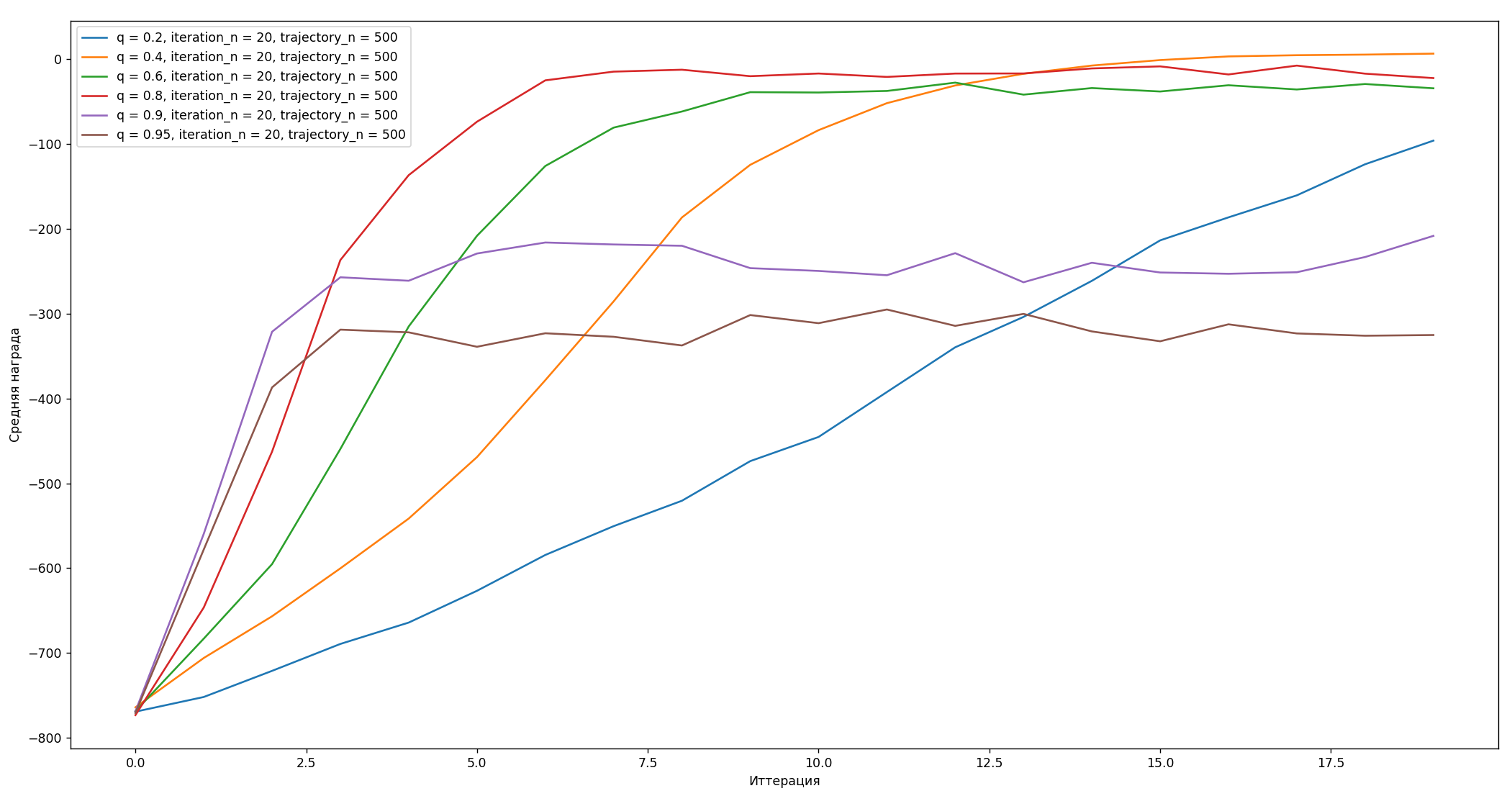
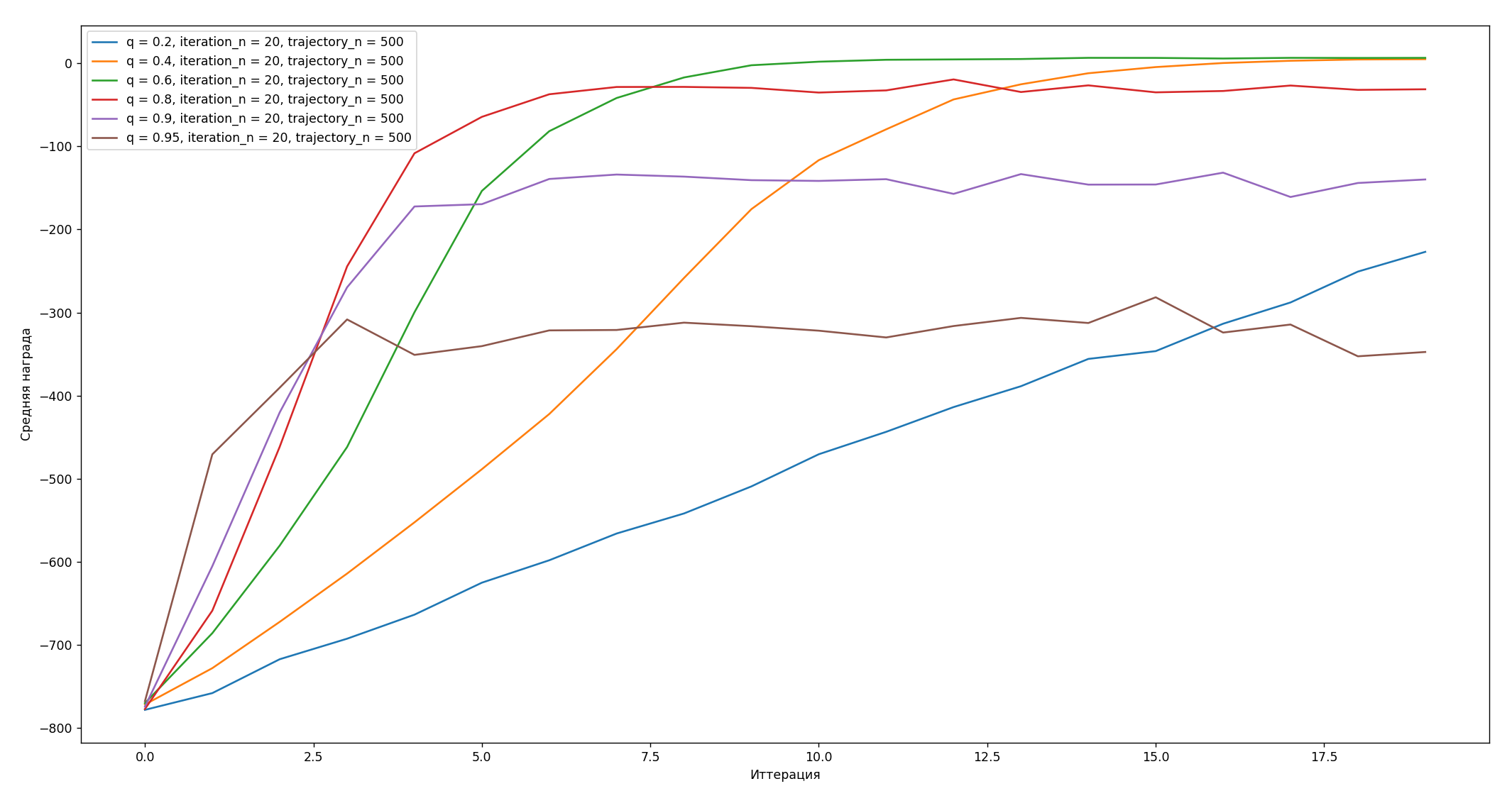




Обучившись 2 раза, можно сделать вывод, что при прочих равных, высокие значения квантиля показывают в среднем способность агента к обучению хуже, чем при более низких значениях квантиля.

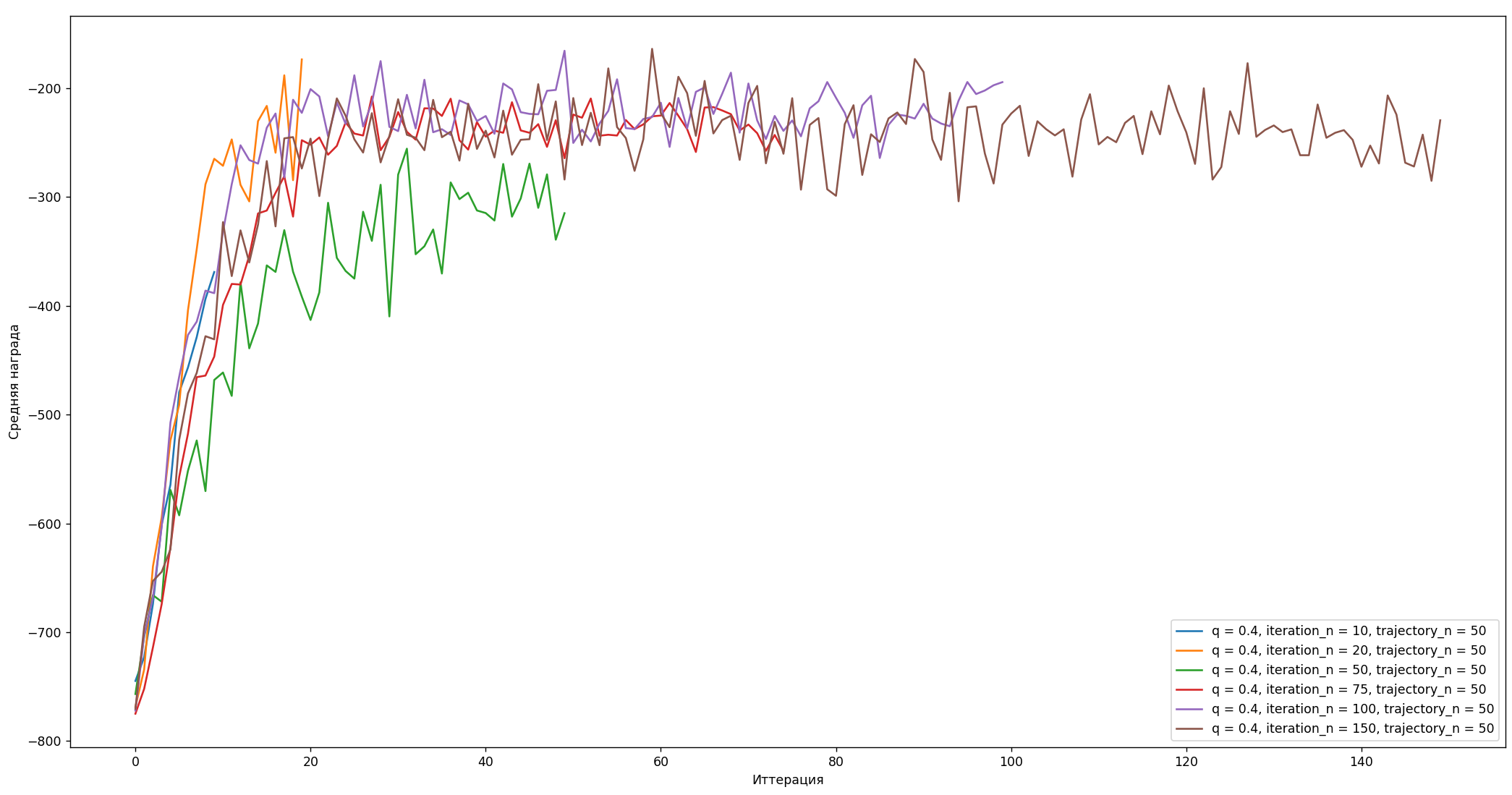
Попробуем также проитерироваться по квантилям, но увеличим количество сэмплируемых траекторий в 10 раз.

**Эксперимент №2:** попробуем различные значения квантиля при количестве иттераций = 20 и количестве траекторий = 500.



Вывод тот же, что и в прошлом эксперименте. Но можно также отметить, что при 10 кратном повышении количества сэмплируемых траекторий, в ряде случаев агент наконец-то смог обучиться действовать почти всегда эффективно (о чём свидетельствует средняя награда 0).

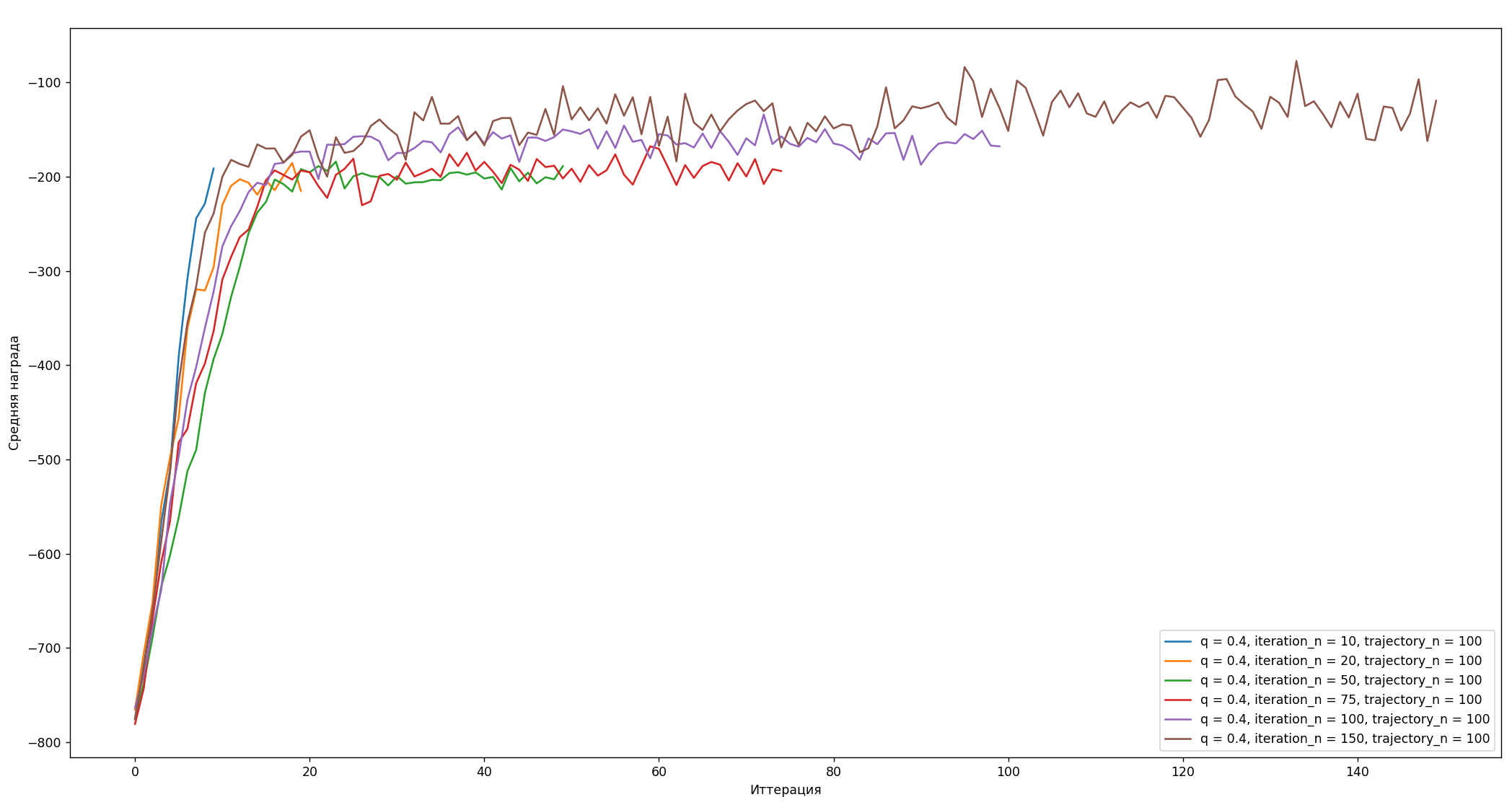
**Эксперимент №3:** попробуем различные значения количества итераций при значении квантиля = 0.4 и количестве траекторий = 50.



Из этого графика видно, что количество итераций вцелом не очень сильно влияет на обучение агента. Примерно после 20-30 итерации значение средней награды было похоже больше на случайное блуждание.

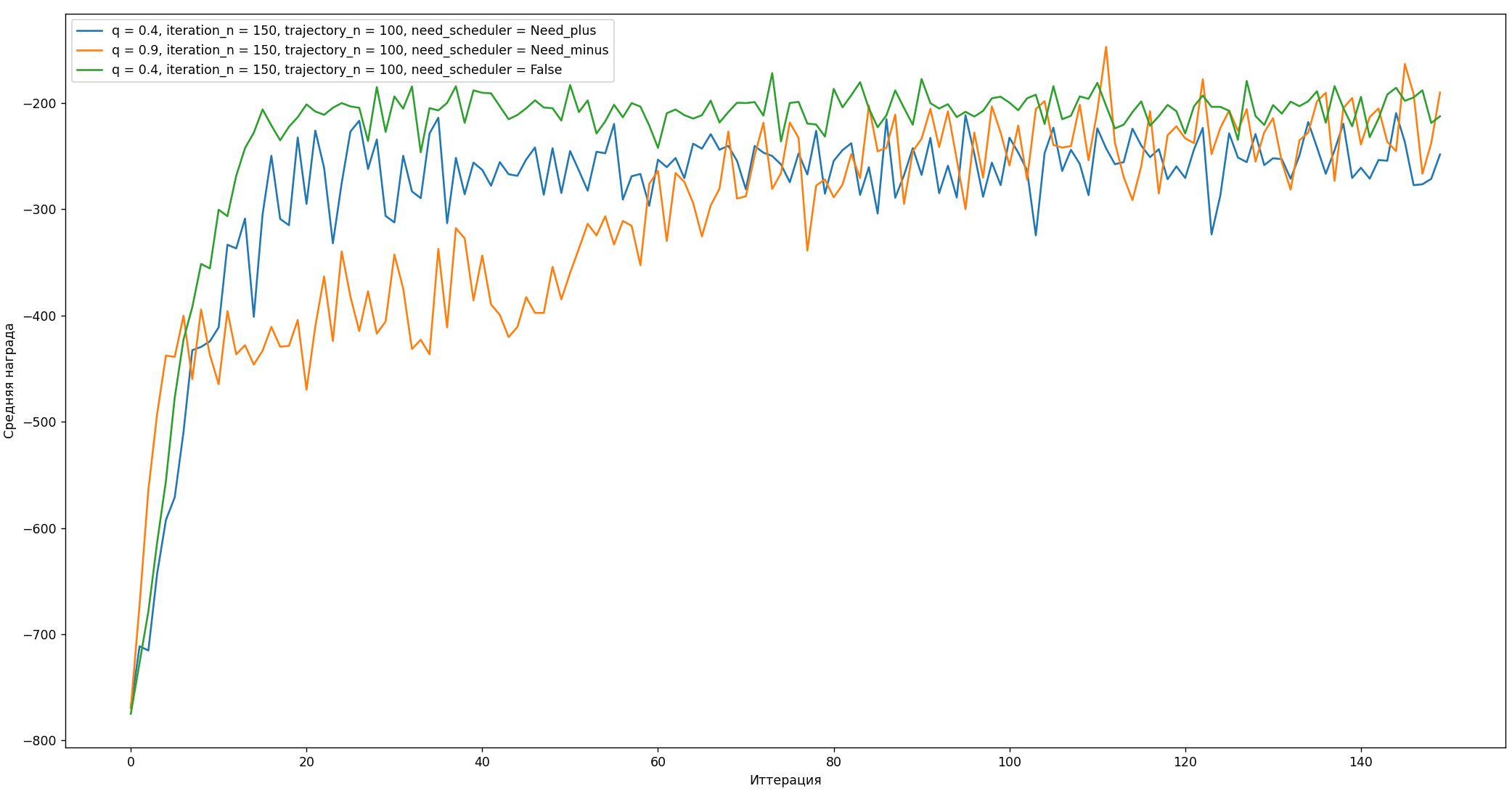
Ещё надо отметить, что квантиль = 0.4 ещё раз оправдал доверие к себе, так как при нём в среднем агент обучается лучше, чем при других (см. эксперимент №1).

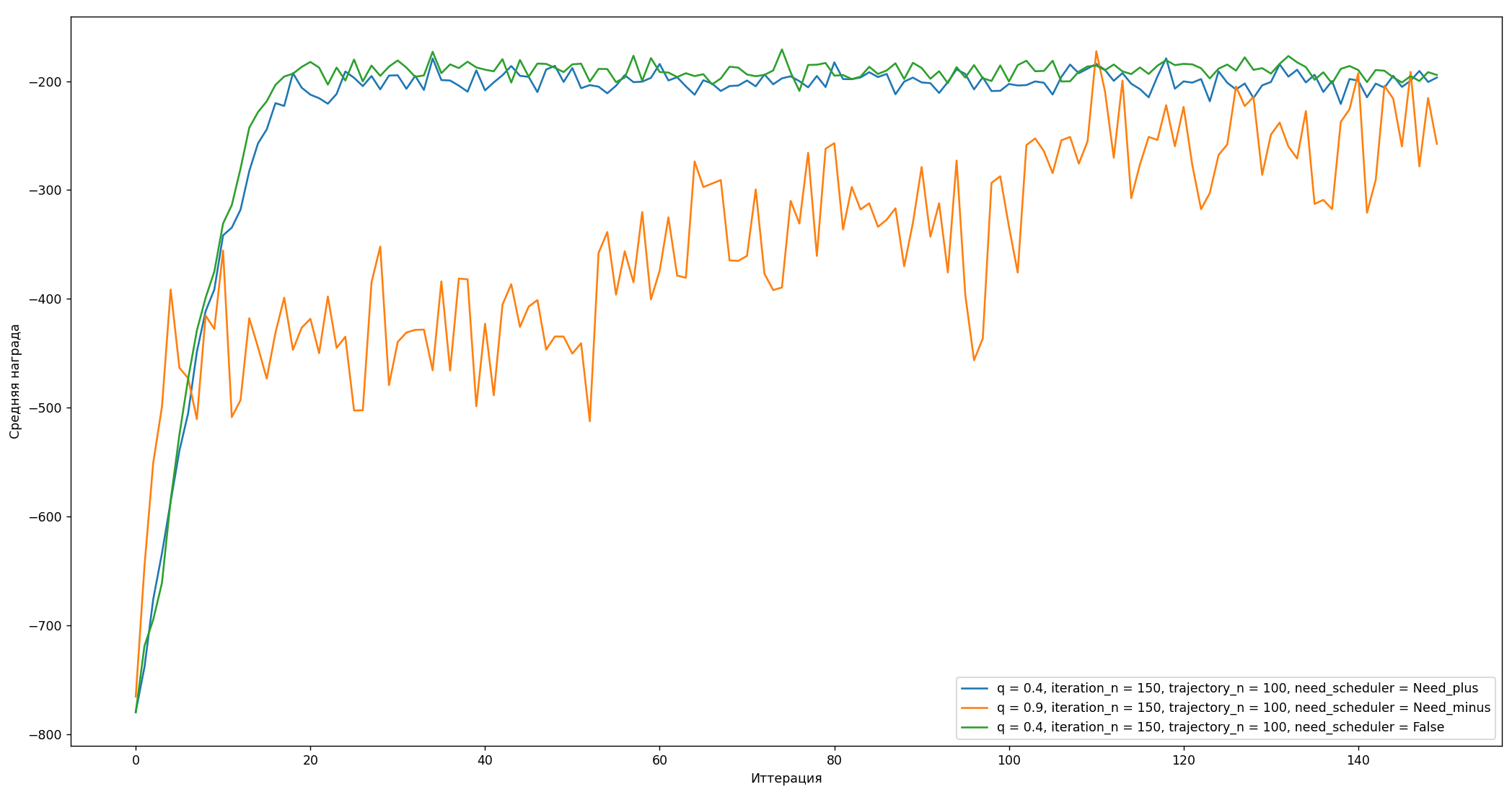
**Эксперимент №4:** попробуем различные значения количества иттераций при значении квантиля = 0.4 и количестве траекторий = 100.



Качество обучение повысилось, но эффективность обучение кратно падает вновь после 20-30 итераций.

**Эксперимент №5:** попробуем ко всем условиям эксперимента №4 добавить ещё одно – применять или не применять «планировщик» (scheduler) квантиля. В зависимости от выбранного режима, каждые 50 итераций q будет увеличиваться/уменьшаться на 0.2.

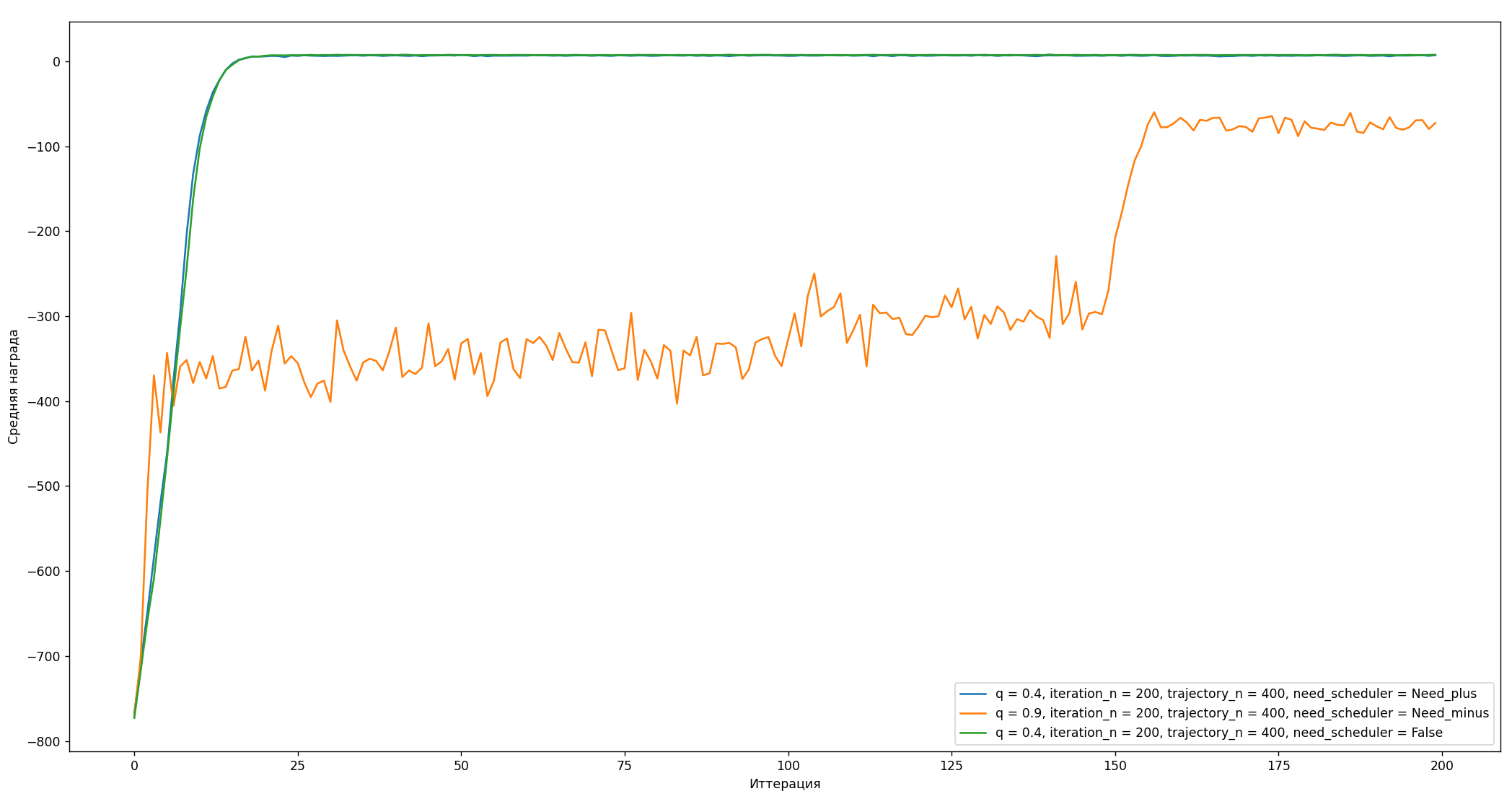




Эксперимент не оправдался.

В случае со стратегией «минус\_квантиль» была логика такая, что на начальных этапах большинство траекторий несут минимум полезной информации, поэтому их можно отбрасывать, но с ростом кол-ва итераций, траектории всё больше и больше этой информации накапливают и нужно больше брать информации из них путём понижения порога квантиля.

**Эксперимент №6**: повышаем количество сэмплируемых траекторий до 400. Итерируемся по различным вариантам планировщика. Количество итераций = 200.

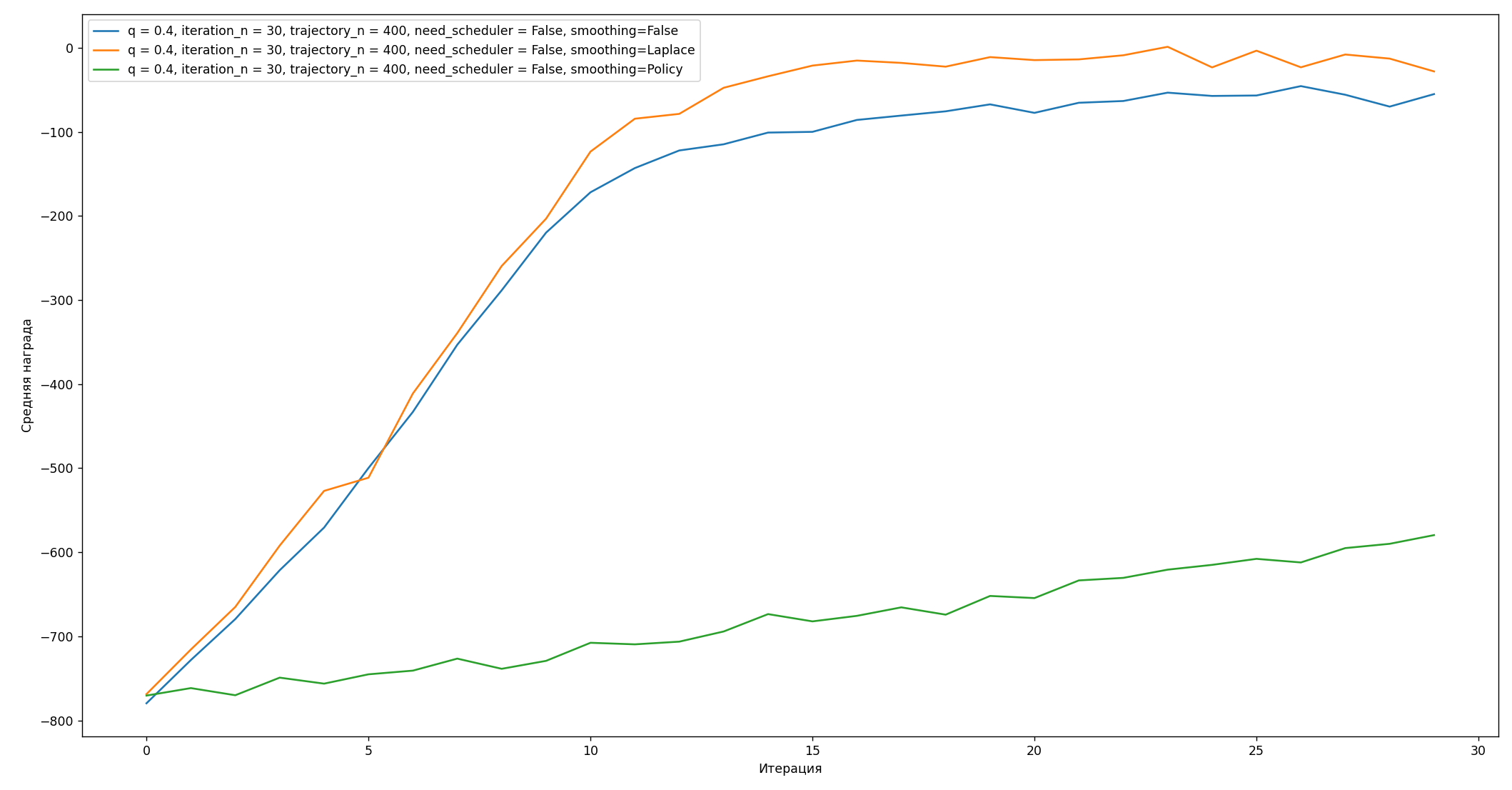


На данном этапе уже можно сделать однозначный вывод, что гиперпараметр trajectory\_n – основной, отвечающий за эффективность обучения агента. К слову, стратегия планировщика «начинаем с 0.4 и +0.2 каждые 50 итераций» показала себя так же эффективно, как политика без планировщика. Если подумать, если планировщик “Need\_plus” правда эффективен в обучении, то ещё с его использованием понижается время работы, так как на поздних итерациях в процессе обучения приходится перебирать меньше элитных траекторий.

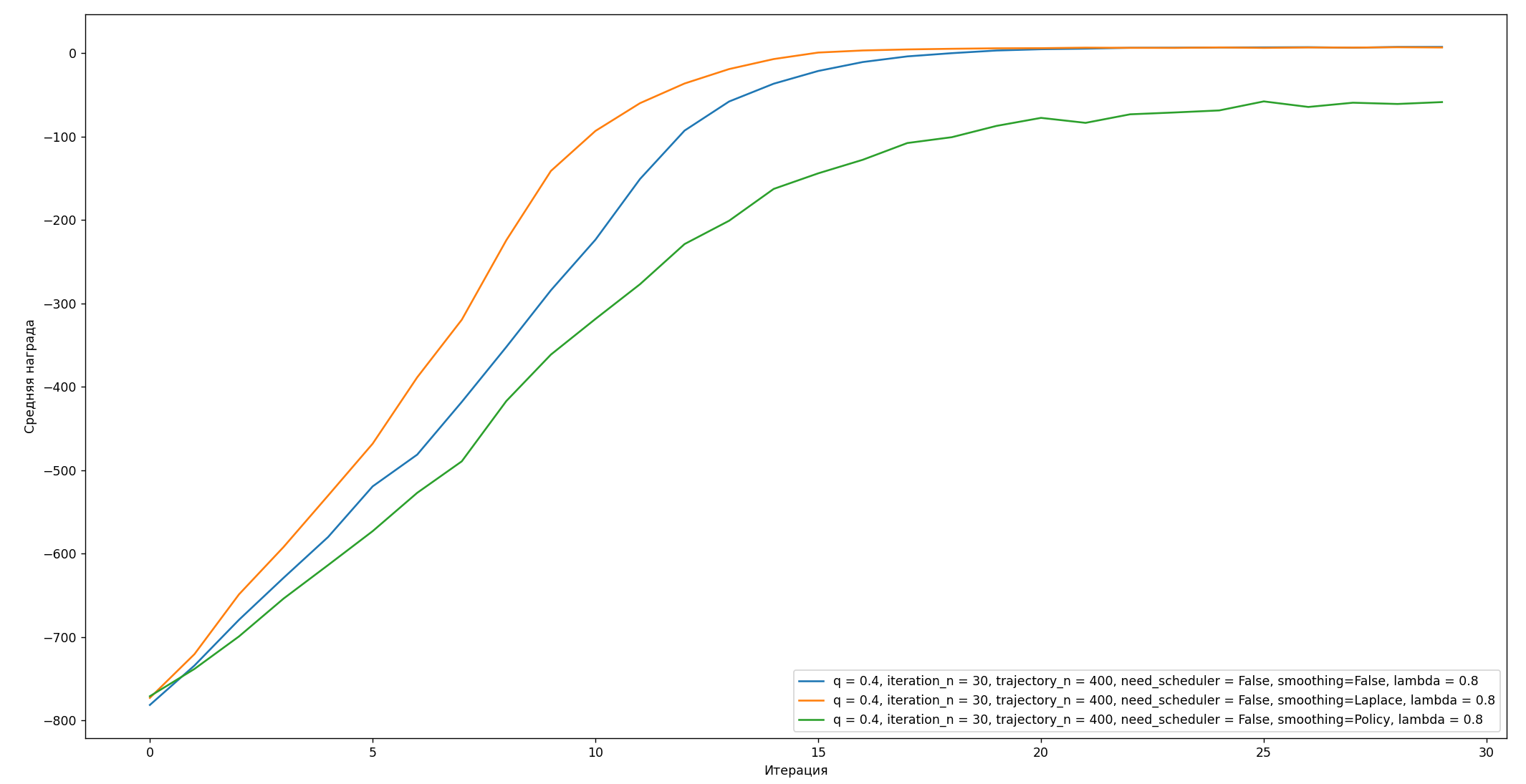
В общем, на основе проведённых экспериментов, для следующих заданий буду использовать набор гиперпараметров {q: 0.4, iteration\_n = 30, trajectory\_n = 400}, как основной.

**Задание №2:** реализовать алгоритм Кросс-Энтропии с двумя типами сглаживания, указанными в лекции 1. При выбранных в пункте 1 гиперпараметров сравнить их результаты с результатами алгоритма без сглаживания.

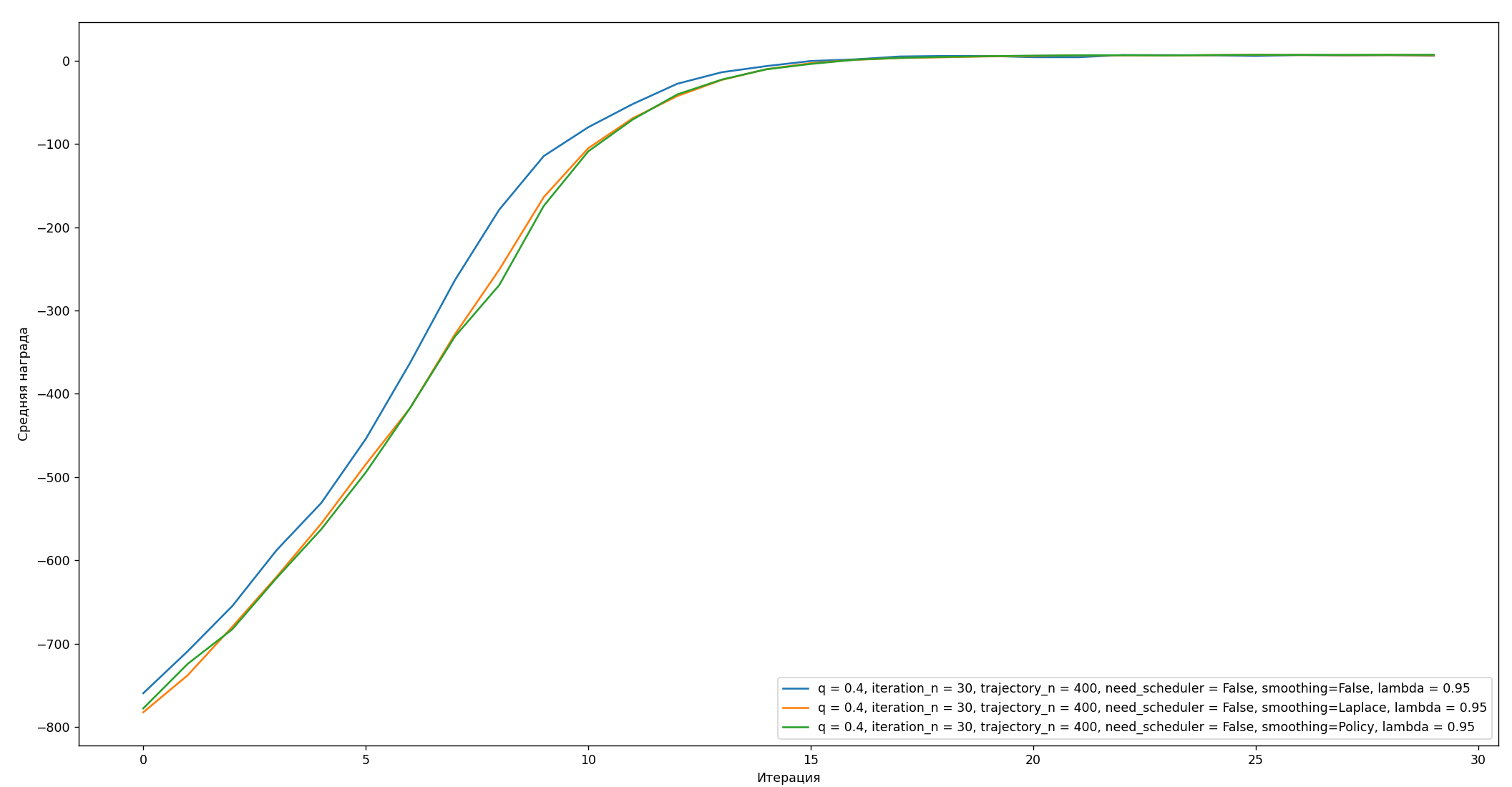
λ = 0.1



λ = 0.8



λ = 0.95



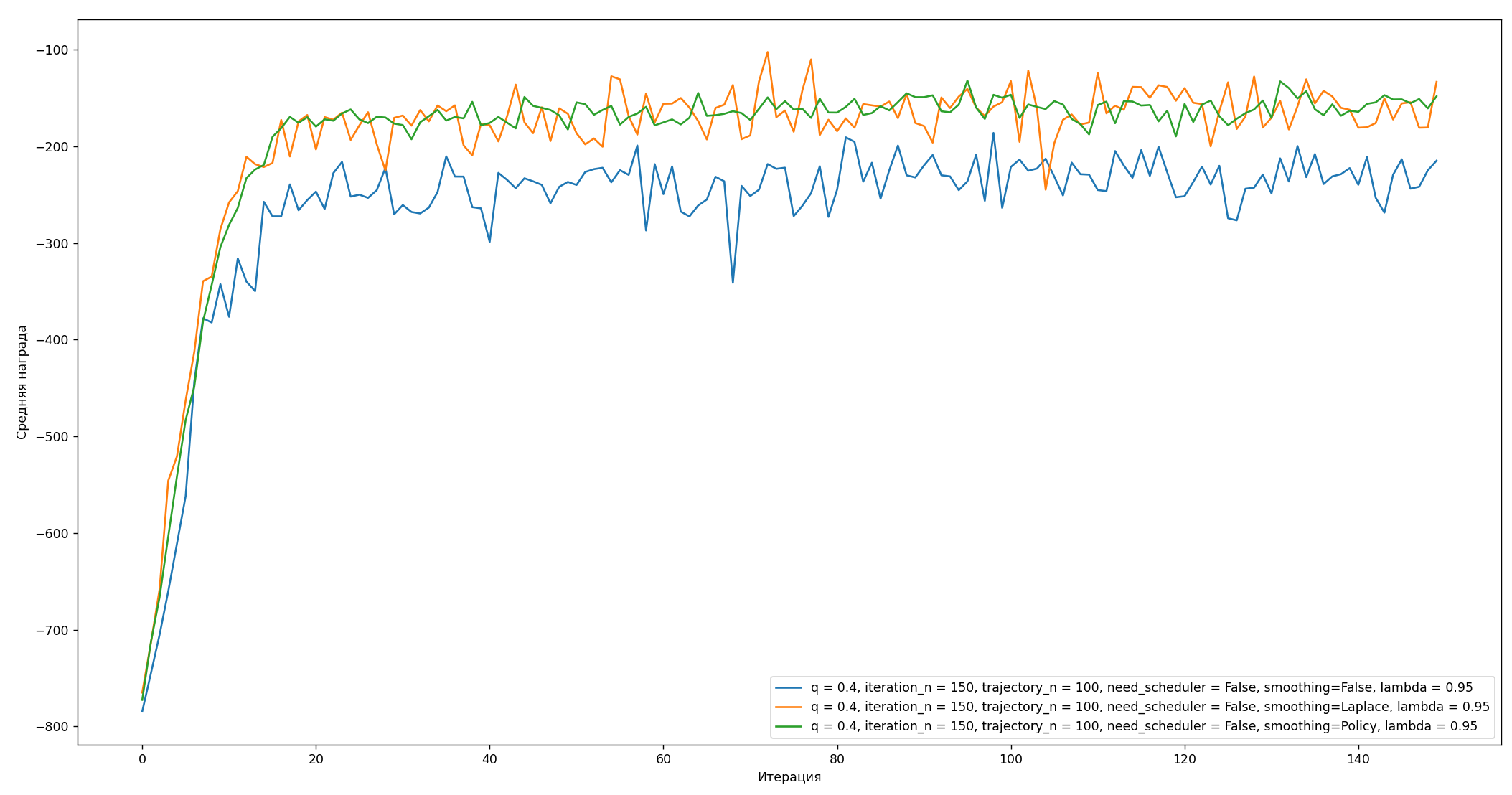
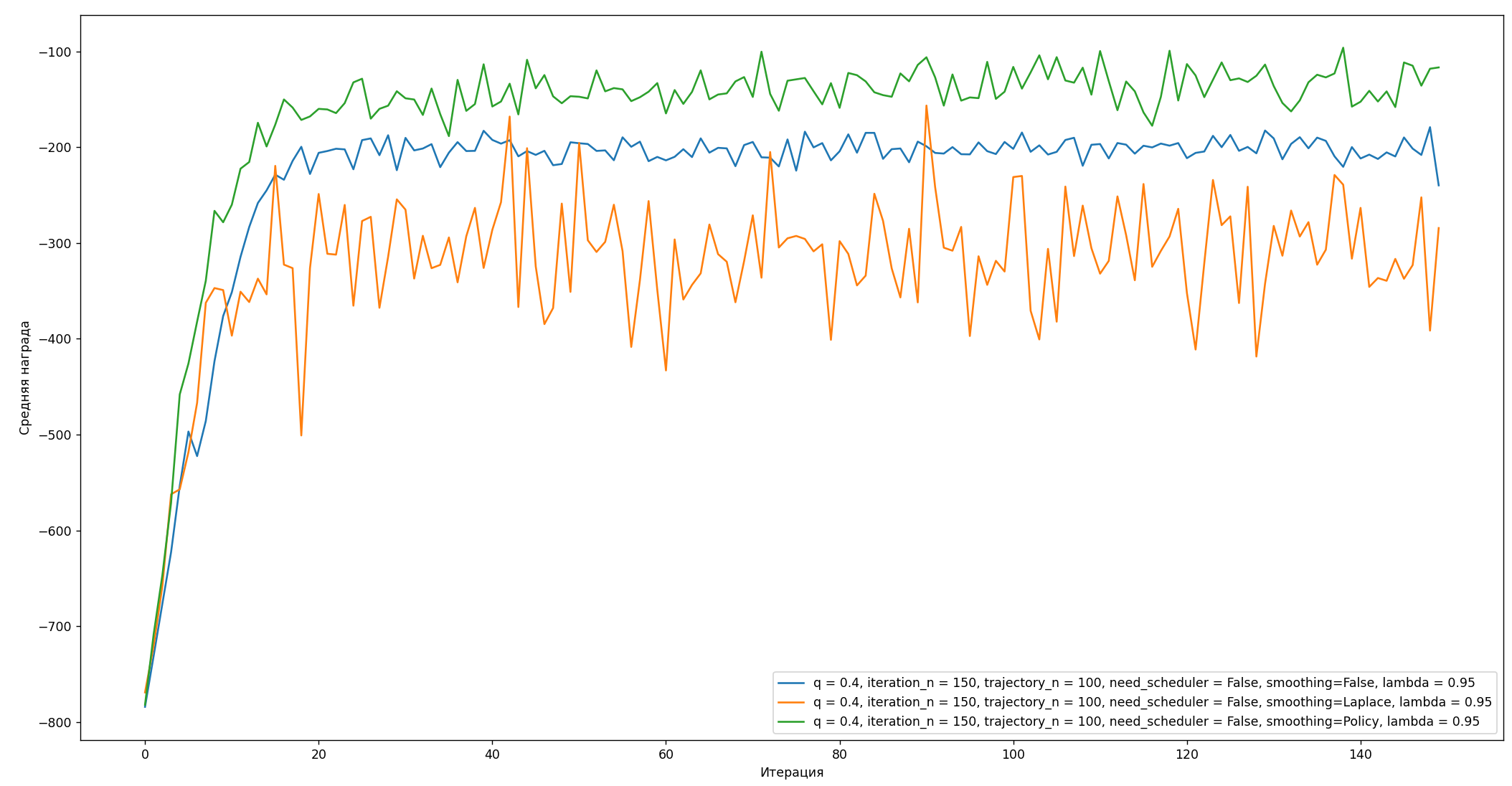
В 3 рассмотренных случаях сглаживание Лапласа работало лучше, чем сглаживание политика (экспоненциальное сглаживание) и в двух из трех случаев даже лучше, чем обучение без сглаживания.

Чтобы “Policy smoothing” был эффективен, лямбда должна быть около 1. Лишь при такая стратегия показала сходный результат с другими вариантами.

Но все агенты обучаются до эффективной стратегии движения примерно за одинаковое количество итераций. Всё ввиду больших значений гиперпараметров, что помогает быстро обучаться агенту, но не изучить работу различных механизмов сглаживания.

Попробую один раз обучить агентов на всех типах сглаживания не по данных, которые были выбраны на 1 шаге.

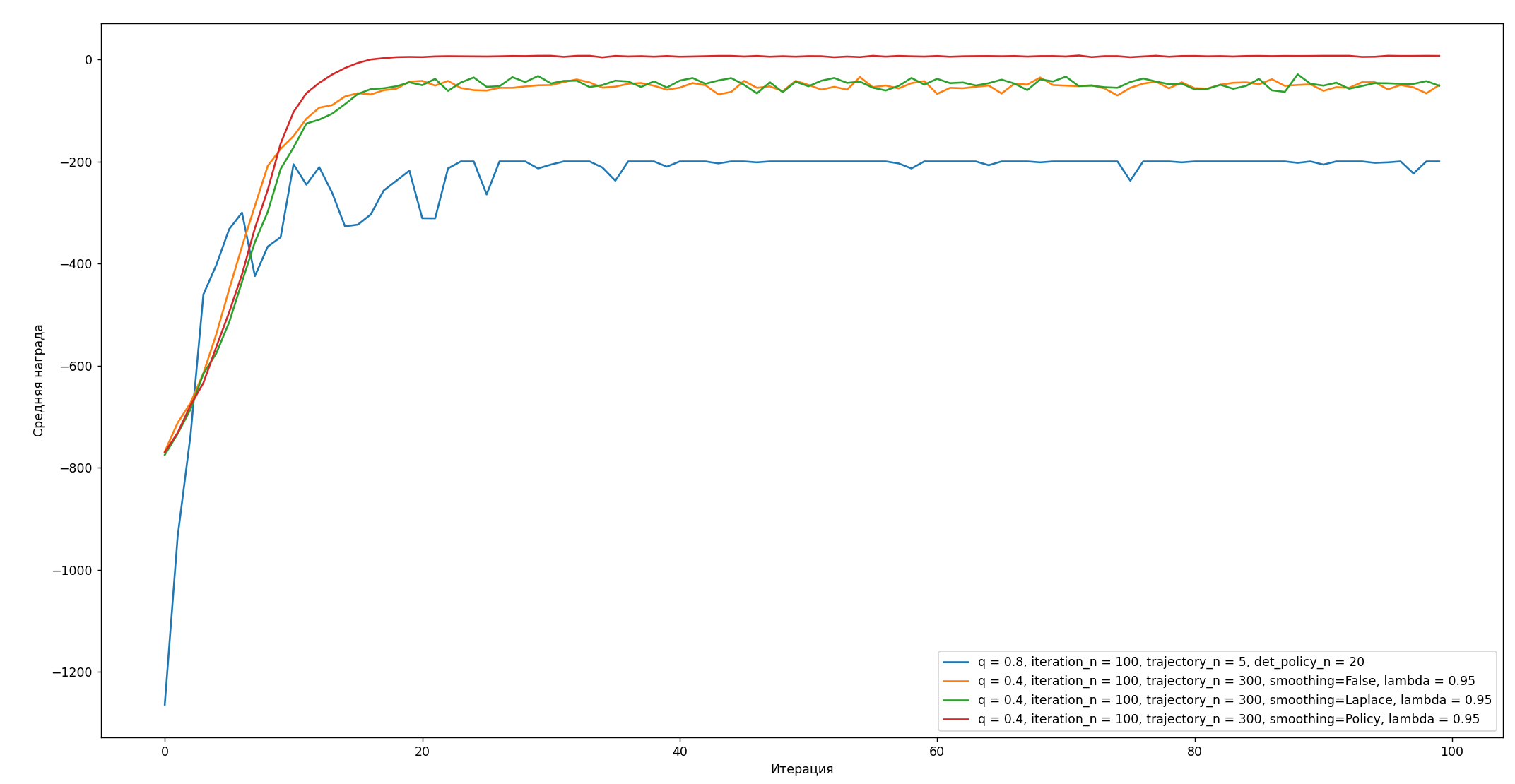
**Эксперимент**: итерируемся по видам сглаживания. Количество итераций = 200. Количество траекторий = 100. q = 0.4.

В случае этого эксперимента “policy smoothing” показало лучше всего, особенно на фоне остальных. Поэтому нельзя однозначно сказать, как вид сглаживания лучше. Но оно точно может помочь. Хотя во втором эксперименте агент с «Лапласовским» сглаживанием обучился хуже всего.

**Задание №3:** реализовать модификацию алгоритм Кросс-Энтропии для стохастических сред, указанную в лекции 1. Сравнить ее результат с алгоритмами из пунктов 1 и 2.

Алгоритм был реализован в соответствии с лекцией, но почему-то, проведя множество итераций с различными гиперпараметрами, максимальная награда = -200.



Для агентов из задания №1 и №2 взяты гиперпараметрами, выбранными как лучшие в предыдущих заданиях.

Для агента, построенного на сэмплировании из стохастической политики – деметрминированных, выбраны такие параметры, как: q = 0.8, кол-во итераций = 100, количество траекторий = 5, количество сэмплируемый траекторий = 20 (даже при 3000 выше -200 не поднималось).

По моему мнению, этот метод вряд ли особо пригоден для использования в реальных задачах, так как количество состояний может стремиться к бесконечности, так и количество действий. А с ростом их количества всё труднее и труднее будет оптимизировать обучение (по крайней мере, кратно растёт вычислительная сложность).

**Общий вывод по работе**: метод кросс-энтропии неплохо работает на «простых» задачах с малой размерностью пространств состояний и действий. Он легко вычисляем и легко интерпретируем. Это хорошо. Методики сглаживания тоже дополнительно помогли достичь хорошего результата. А вот метод сэплирования детерминированных политик требует более глубоко изучения, чтобы понять его реальную применимость.