0.1 Алгоритм

С учетом вышесказанного можно построить следующий алгоритм. Алгоритмы 1, 2 демонстрируют обратный и прямой проход для получения оптимальной поправки. Алгоритм 3 демонстрирует общий алгоритм построения управления.

Продемонстрируем работу алгоритма для классической задачи перехода в целевое состояние x^{final} без фазовых ограничений. Для этого будем использовать следующие компоненты функции цены:

$$q^{\text{final}}(x) = \left\|x - x^{\text{final}}\right\|^2, \qquad q(x) \equiv 0. \tag{0.1}$$

Начальным референсным управлением выберем

$$\bar{u}^k = 0,\ k = \overline{1, N}. \tag{0.2}$$

Алгоритм 1: Обратный проход

function *BackwardPass($\bar{u}$, $\bar{x}$)*
begin
  $S_{N+1}, v^{N+1} \leftarrow$ (**??**)
  for $k \leftarrow N$ **to** 1 **do**
    $S_k, v^k \leftarrow$ (**??**)
  end
  return *S, v*
end

Алгоритм 2: Прямой проход

```
function ForwardPass(ū, x̄, J_prev)
begin
    η ← 1
    do
        δx^0 ← 0
        for k ← 1 to N do
            δu^k, δx^{k+1} ← (??), (??)
            u^k ← ū^k + δu^k
        end
        J ← (??)
        η ← γη
    while (J_prev − J) / (J_δ(0) − J_δ(δu)) ∉ [ξ_1, ξ_2]
    return u, J
end
```

Алгоритм 3: Синтез управления

```
function Synthesis(ū)
begin
    J ← (??)
    do
        J_prev ← J
        x̄ ← (??)
        S, v ← BackwardPass(ū, x̄)
        u, J ← ForwardPass(S, v, J_prev)
    while |J − J_prev| ⩾ ε
    return u
end
```

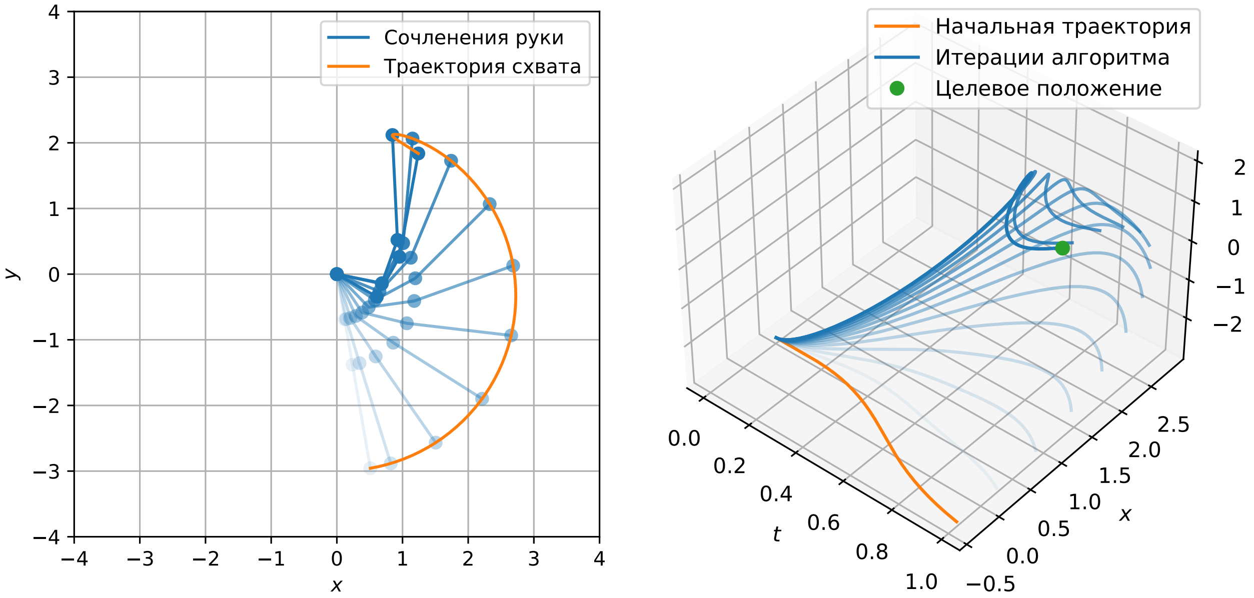

Рис. 1: Решение задачи перехода в целевое состояние (0.1) с начальным референсным управлением (0.2). Слева: поведение системы при полученном управлении. Справа: траектории схвата на каждой итерации алгоритма, более ранние итерации показаны бледнее. Алгоритм сошелся на 14 итерации.