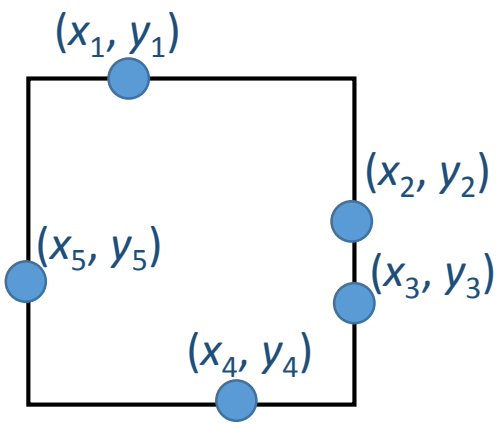
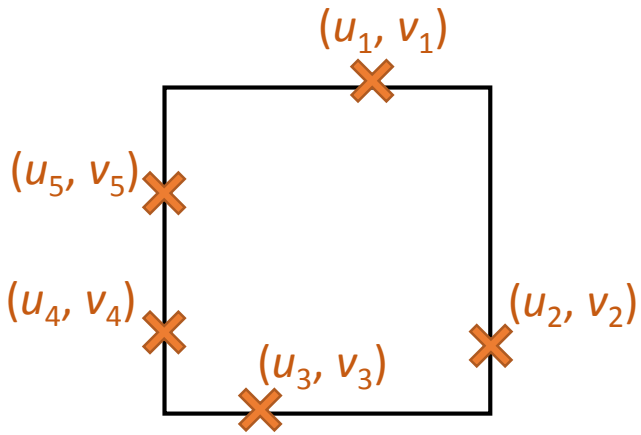
ICPC Asia::Tokyo 2014 Problem K – L_∞ Jumps

K: L_∞ Jumps – Solution (1/4)

- Sort the given points (x_1, y_1), …, (x_n, y_n) in clockwise order. Call them base vectors.
- Suppose we fix jump vectors to (u_1, v_1), …, (u_n, v_n). $(u_1+\ldots+u_n = s, v_1+\ldots+v_n = t)$
- What is the optimal assignment between base vectors and jump vectors? → **Greedy is the best**



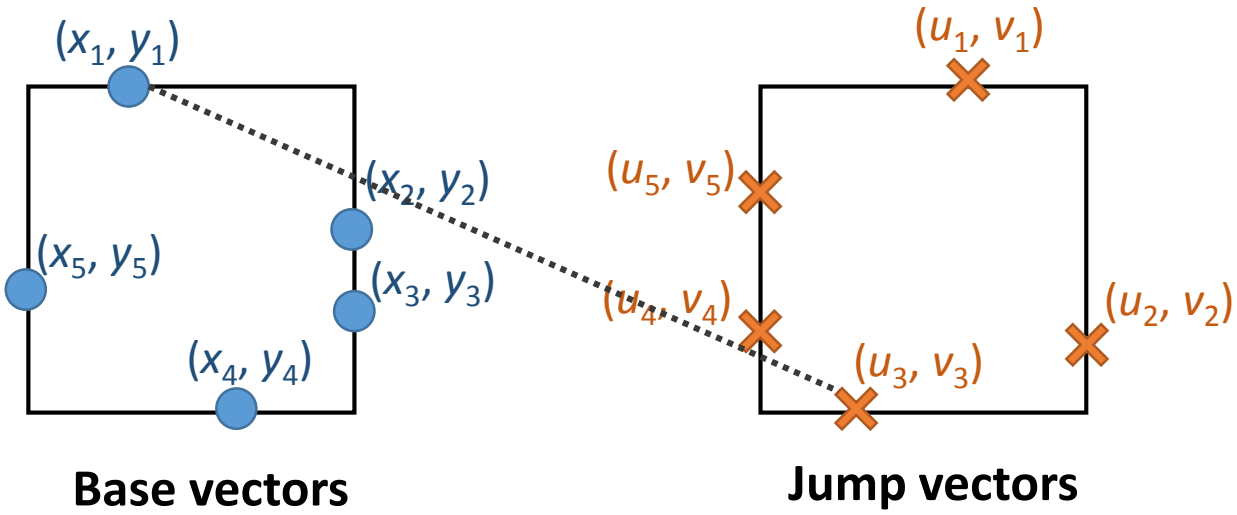
Base vectors



Jump vectors

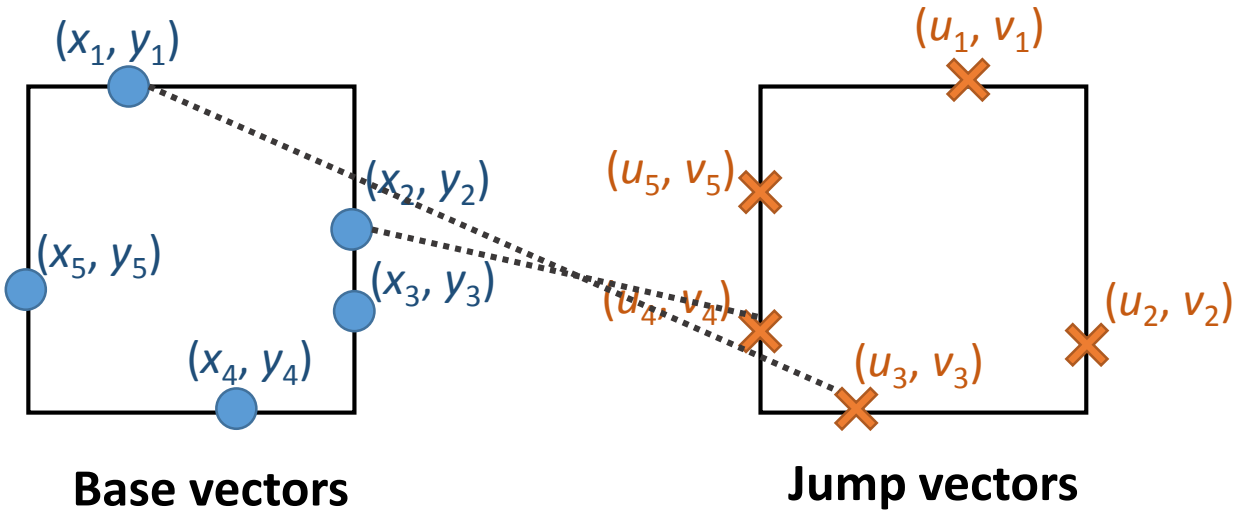
K: L_∞ Jumps – Solution (1/4)

- Sort the given points (x_1, y_1), …, (x_n, y_n) in clockwise order. Call them base vectors.
- Suppose we fix jump vectors to (u_1, v_1), …, (u_n, v_n). $(u_1+\ldots+u_n = s,\ v_1+\ldots+v_n = t)$
- What is the optimal assignment between base vectors and jump vectors? → **Greedy is the best**



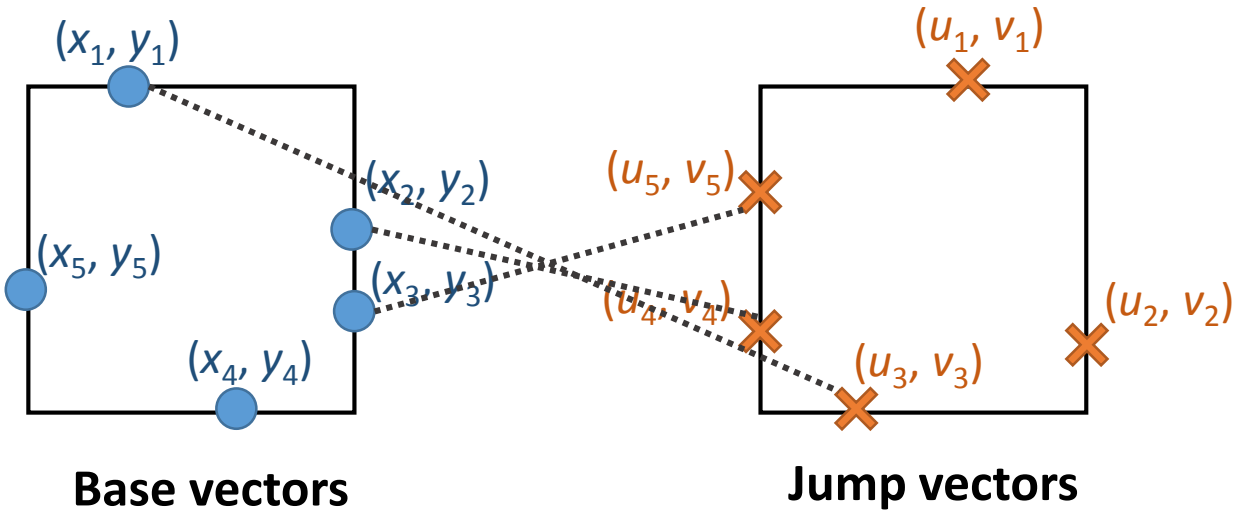
K: L_∞ Jumps – Solution (1/4)

- Sort the given points (x_1, y_1), …, (x_n, y_n) in clockwise order. Call them base vectors.
- Suppose we fix jump vectors to (u_1, v_1), …, (u_n, v_n). ($u_1+\ldots+u_n = s$, $v_1+\ldots+v_n = t$)
- What is the optimal assignment between base vectors and jump vectors? → **Greedy is the best**



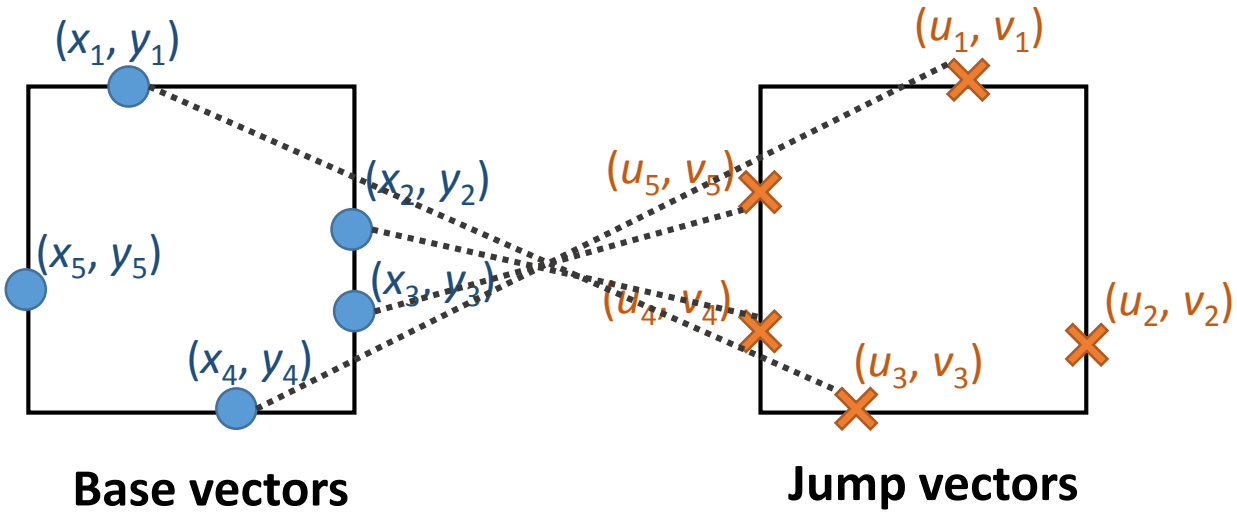
K: L_∞ Jumps – Solution (1/4)

- Sort the given points (x_1, y_1), …, (x_n, y_n) in clockwise order. Call them base vectors.
- Suppose we fix jump vectors to (u_1, v_1), …, (u_n, v_n). ($u_1+\ldots+u_n = s$, $v_1+\ldots+v_n = t$)
- What is the optimal assignment between base vectors and jump vectors? → **Greedy is the best**



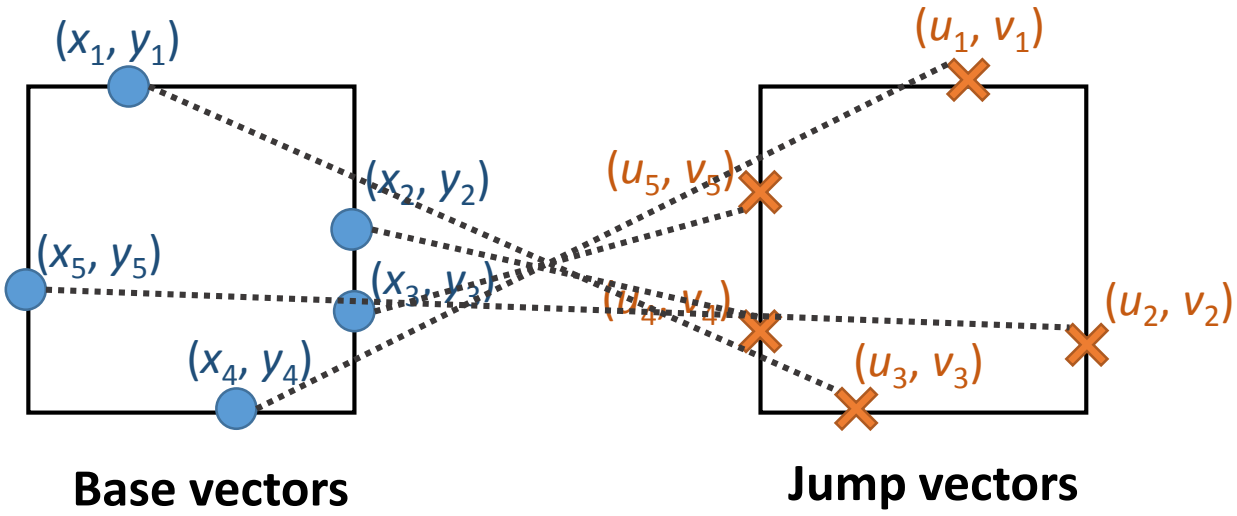
K: L_∞ Jumps – Solution (1/4)

- Sort the given points (x_1, y_1), …, (x_n, y_n) in clockwise order. Call them base vectors.
- Suppose we fix jump vectors to (u_1, v_1), …, (u_n, v_n). $(u_1+\ldots+u_n = s,\ v_1+\ldots+v_n = t)$
- What is the optimal assignment between base vectors and jump vectors? → **Greedy is the best**



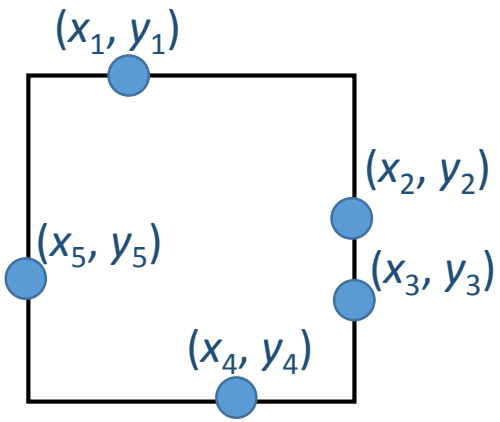
K: L_∞ Jumps – Solution (1/4)

- Sort the given points (x_1, y_1), …, (x_n, y_n) in clockwise order. Call them base vectors.
- Suppose we fix jump vectors to (u_1, v_1), …, (u_n, v_n). $(u_1+\ldots+u_n = s,\ v_1+\ldots+v_n = t)$
- What is the optimal assignment between base vectors and jump vectors? → **Greedy is the best**

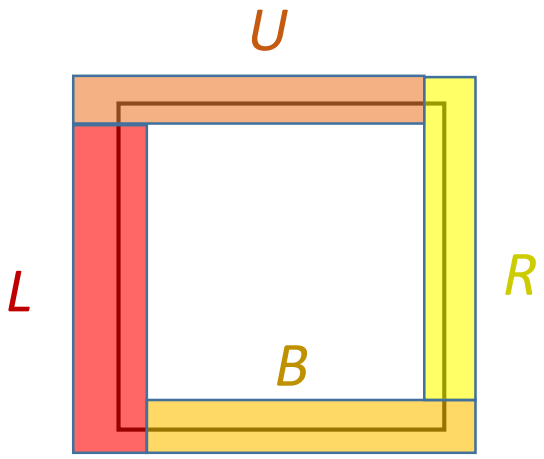


K: L_∞ Jumps – Solution (2/4)

- But how to determine jump vectors?
- Let's fix the count of jump vectors in (upper/right/bottom/left) part of the square edge.
  - U := count in upper
    R := count in right
    B := count in bottom
    L := count in left



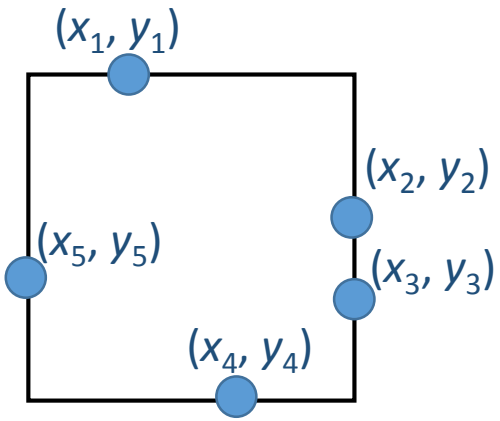
Base vectors



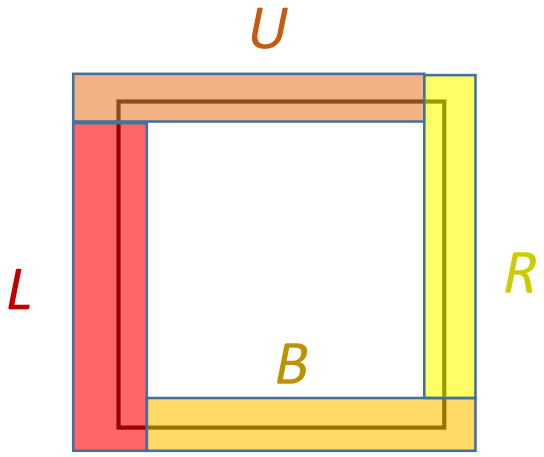
Jump vectors

K: L_∞ Jumps – Solution (3/4)

- Let (p_1, d), …, (p_U, d) := jump vectors in upper,
  $(q_1, -d)$, …, (q_B, d) := jump vectors in bottom.
  - Since U,R,B,L is fixed,
    $$p_1+\ldots+p_U+q_1+\ldots+q_B - Ld + Rd = s$$
    must be satisfied.
- Since greedy assignment is the best, we can compute optimal jump vectors (if we fix offset.)



Base vectors

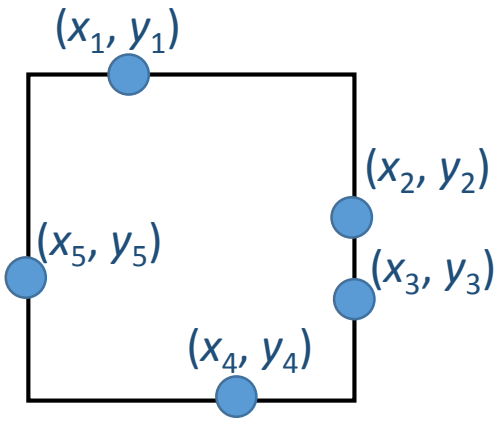


Jump vectors

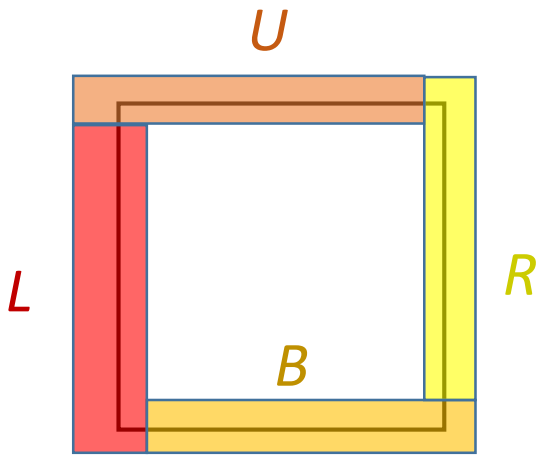
K: L_∞ Jumps – Solution (4/4)

- Complexity?
  - Fix U,R,B,L: Since $U+R+B+L=n$, there are $O(n^3)$ combins.
  - Fix offset for greedy assignment : $O(n)$ ways.
  - Compute the cost for jump vectors : $O(n)$ time.

  - $O(n^5)$ time in total.



Base vectors



Jump vectors

K: L_∞ Jumps – Summary

No submission…