The audio for the last 6 mins of the video lecture did not get recorded. Here is what I explained in those last minutes:

- We use dynamic programming to compute $P_Y(x,y)$ for each label subset Y. In the previous slide, we saw that the two inner summations of the RHS of P(x,y) was equal to the polynomial $P_Y(x,y)$.
- Now, the outer summation of the RHS of P(x,y) is over all subsets X and Y is [k]\X. Therefore, the outer summation is essentially over all subsets Y.
- Thus, once we evaluate all $P_Y(x,y)$ for all label subsets Y, we can evaluate P(x,y).
- Time taken is polynomial for evaluating each $P_Y(x,y)$ and there are 2^k possible Y's. So total time to evaluate P(x,y) is 2^k poly(n).
- Last slide: Thus, we obtain a randomized FPT algorithm for k-path, running in 2^k poly(n) time. The randomization is due to application of the Schwartz-Zippel Lemma and the bulk of the running time is because of evaluating the Labeled Walk polynomial (using weighted Inclusion-Exclusion).
- In this part, we have looked at 2 important algebraic tool, Inclusion-Exclusion and identically zero polynomial testing, in order to design FPT algorithms.
- Algebraic tools like these are often used in Parameterized Complexity to design algorithms. In particular, in recent times many state of the art algorithms are based on algebraic techniques.