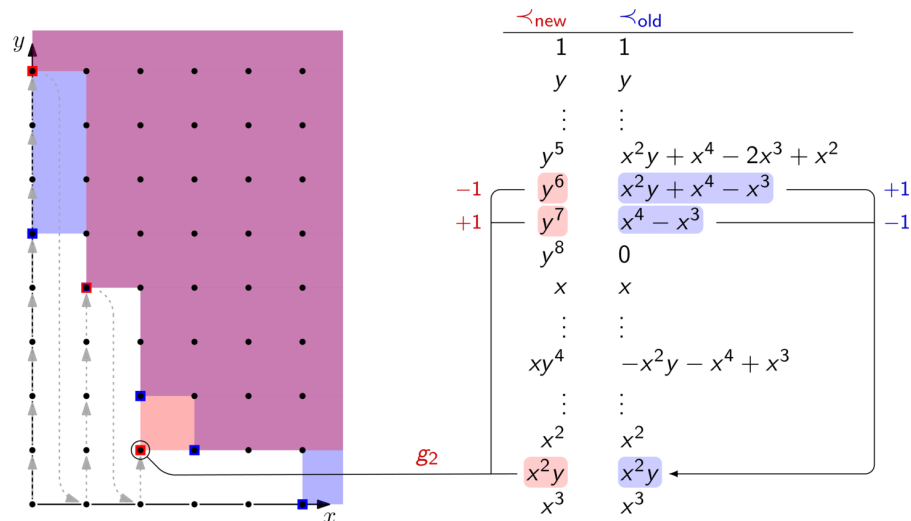


# Converting Gröbner bases with FGLM

by Jan Ferdinand Sauer Oct 22, 2020 0 Comment

Have you ever computed a Gröbner basis in some monomial order  $\prec_0$  only to then realize that you *actually* wanted it in another order  $\prec_1$ ? I know, happens *all* the time... But fret not, FGLM can convert between the two orders, and you don't have to start your computation from scratch.



[Here's a recording](#) where I'm explaining how FGLM works. If you'd like to look through the slides at your own pace, [here they are](#). You can also find the two Gröbner bases used in the example as well as pseudocode for FGLM in the slides, in case you want to retrace the steps yourself.



If you are interested in not only *how* but also *why* FGLM works, have a look at [the original publication](#). Section 5 and 6 give a comprehensive overview of FGLM's complexity, which can be summarized as  $O(n \cdot D^3)$  where  $D$  is the dimension of quotient ring  $k[x_0, \dots, x_{n-1}] / \langle G \rangle$  as a  $k$ -vector-space. More intuitively,  $D$  is the number of monomials in the staircase of a Gröbner basis, which are those dots in the non-blue region in this post's first picture.

## FGLM – pseudocode

**Input:**  $\prec_{\text{new}}, G_{\text{old}}, \prec_{\text{old}}$   
**Output:**  $G_{\text{new}}$   
dict =  $\emptyset$   
 $G_{\text{new}} = \emptyset$   
next\_monoms = {1}  
**while** next\_monoms  $\neq \emptyset$  **do**  
    monom =  $\min_{\prec_{\text{new}}} \{\text{next\_monoms}\}$

```
else
    dict = dict ∪ (monom : reduced_monom)
    next_monoms = next_monoms ∪ {xi · monom | i ∈ {0, ..., n - 1}}
return Gnew
```



**Jan Ferdinand Sauer**

Website: <https://asdm.gmbh>

## Leave a Reply

---

Logged in as [Jan Ferdinand Sauer](#) [Logout](#)

**Comment**

Submit