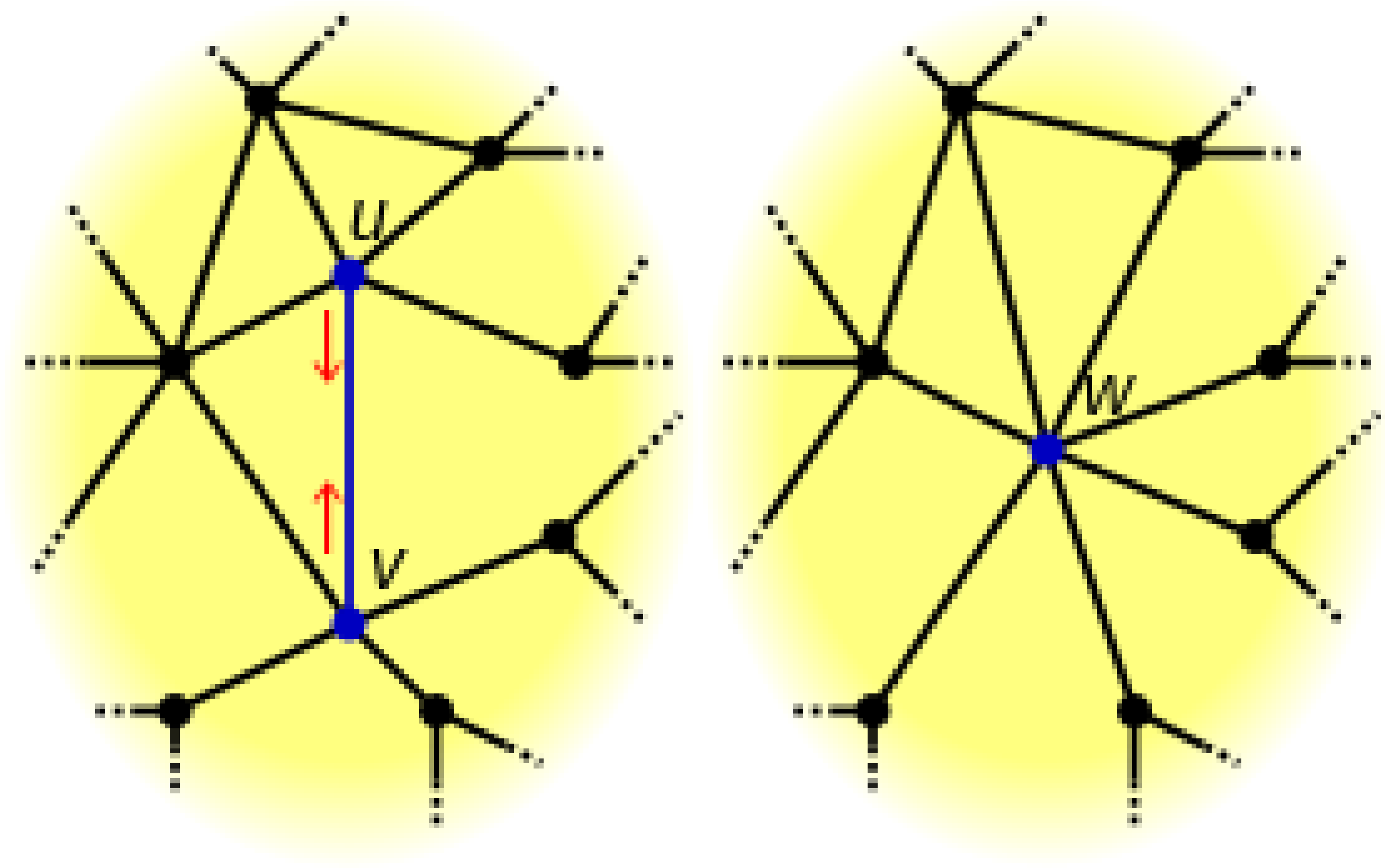
Stellingen

- Een graaf is planair dan en slechts dan als hij niet ofwel K_5 of $K_{3,3}$ als *minor* heeft
- De minor *H* uit *G* is een graaf die uit *G* gemaakt kan worden door de volgende operaties:
  - Een edge verwijderen
  - Een vertex verwijderen
  - *Edge contraction*

Edge Contraction



Stellingen

- Een graaf is planair dan en slechts dan als al zijn *biconnected* componenten planair zijn
  - *Biconnected* betekend dat er geen enkele vertex kan worden weggehaald zodat de graaf niet meer *connected* is.

Stellingen

- Voor een planaire graaf met $v >= 3$ geldt:
  $e < 3v - 6$
- Voor een planaire graaf met $v >= 3$ zonder cycels van lengte 3 geldt: $e < 2v - 4$
- Voor *v* het aantal vertices en *e* het aantal edges

Algoritme

- Allereerst wordt de graaf gesplits in zijn *biconnected* componenten
- Voor ieder component wordt bepaald of het K_5 of $K_{3,3}$ als *minor* bevat
- Dit doen we door alle mogelijke *minors* af te gaan en te kijken of deze *isomorph* zijn aan K_5 of $K_{3,3}$
- O(2^n)

Algoritme

- We kunnen het algorithme nog optimalizeren:
- Wanneer de vergelijkingen $e < 3v - 6$ of $e < 2v - 4$ niet gelden voor een graaf, is de graaf niet planair
- Dus heeft de graaf K_5 of $K_{3,3}$ als *minor*
- Wanneer we dus een *minor* van G vinden waarvoor de vergelijkingen niet gelden, is G ook niet planair

Algoritme

- Dus gaan we eerst de *minors* proberen te maken uit G die zoveel mogelijk edges hebben met zo min mogelijk vertices
- Dus we proberen *e* te maximaliseren, en *v* te minimaliseren
- Is alleen sneller voor niet-planaire grafen, voor planaire grafen moeten nog steeds alle *minors* doorlopen worden