Implementation (sources: http://drorbn.net/icerm23/ap). I like it most when the implementation matches the math perfectly. We failed here.

```
Once[<< KnotTheory`];
```

```
Loading KnotTheory` version
  of February 2, 2020, 10:53:45.2097.
Read more at http://katlas.org/wiki/KnotTheory.
```

Utilities. The step function, algebraic numbers, canonical forms.

```
θ[x_] /; NumericQ[x] := UnitStep[x]
```

```
ω2[v_][p_] := Module[{q = Expand[p], n, c},
  If[q === 0, 0,
   c = Coefficient[q, ω, n = Exponent[q, ω]];
   c v^n + ω2[v][q - c (ω + ω^-1)^n]]];
```

```
sign[ℰ_] := Module[{n, d, v, p, rs, e, k},
 {n, d} = NumeratorDenominator[ℰ];
 {n, d} /= ω^(Exponent[n,ω]/2+Exponent[n,ω,Min]/2);
 p = Factor[ω2[v]@n * ω2[v]@d /. v → 4 u^2 - 2];
 rs = Solve[p == 0, u, Reals];
 If[rs === {}, Sign[p /. u → 0],
  rs = Union@(u /. rs);
  Sign[(-1)^(e=Exponent[p,u]) Coefficient[p, u, e]] + Sum[
    k = 0;
    While[(d = RootReduce[∂_{u,++k} p /. u → r]) == 0];
    If[EvenQ[k], 0, 2 Sign[d]] * θ[u - r],
    {r, rs}]
 ]
]
```

```
SetAttributes[B, Orderless];
CF[b_B] := RotateLeft[#, First@Ordering[#] - 1] & /@
  DeleteCases[b, {}]
```

```
CF[ℰ_] := Module[{γs = Union@Cases[ℰ, γ_ | γ̄_, ∞]},
 Total[CoefficientRules[ℰ, γs] /.
   (ps_ → c_) :> Factor[c] × Times @@ γs^ps]]
```

```
CF[{}] = {};
CF[𝒞_List] :=
 Module[{γs = Union@Cases[𝒞, γ_, ∞], γ},
  CF /@ DeleteCases[0][
    RowReduce[Table[∂_γ r, {r, 𝒞}, {γ, γs}]].γs] ]
```

```
(ℰ_)* := ℰ /. {γ̄ → γ, γ → γ̄, ω → ω^-1, c_Complex :> c*};
r_Rule^+ := {r, r*}
```

```
RulesOf[γ_i_ + rest_.] := (γ_i → -rest)^+;
CF[PQ[𝒞_, q_]] := Module[{n𝒞 = CF[𝒞]},
  PQ[n𝒞, CF[q /. Union @@ RulesOf /@ n𝒞]] ]
```

```
CF[Σ_b_[σ_, pq_]] := Σ_CF[b][σ, CF[pq]]
```

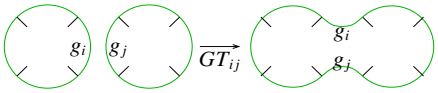
Pretty-Printing.

```
Format[Σ_b_B[σ_, PQ[𝒞_, q_]]] := Module[{γs},
  γs = γ_# & /@ Join @@ b;
  Column[{TraditionalForm@σ,
    TableForm[Join[
      Prepend[""] /@ Table[TraditionalForm[∂_c r],
        {r, 𝒞}, {c, γs}],
      {Prepend[""][
        Join @@
          (b /. {l_, m___, r_} :>
             {DisplayForm@RowBox[{"(", l}],
              m, DisplayForm@RowBox[{r, ")"}]}]) /.
         i_Integer :> γ_i ]},
      MapThread[Prepend,
       {Table[TraditionalForm[∂_{r,c} q], {r, γs*},
         {c, γs}], γs*}]
     ], TableAlignments → Center]
   }, Center] ];
```

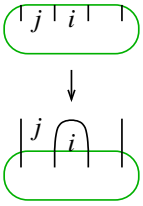
The Face-Centric Core.

```
Σ_b1_[σ1_, PQ[𝒞1_, q1_]] ⊕ Σ_b2_[σ2_, PQ[𝒞2_, q2_]] ^:=
  CF@Σ_Join[b1,b2][σ1 + σ2, PQ[𝒞1 ⋃ 𝒞2, q1 + q2]];
```

GT for Gap Touch:




```
GT_i_,j_@Σ_B[{li___,i_,ri___},{lj___,j_,rj___},bs___][σ_,
   PQ[𝒞_, q_]] :=
 CF@Σ_B[{ri,li,j,rj,lj,i},bs][σ, PQ[𝒞 ⋃ {γ_i - γ_j}, q]]
```



cor·don (kôr'dn) *n.*

THE FREE DICTIONARY BY FARLEX

1. A line of people, military posts, or ships stationed around an area to enclose or guard it: *a police cordon.*

2. A rope, line, tape, or similar border stretched around an area, usually by the police, indicating that access is restricted.

$$\begin{matrix} s \\ i \end{matrix}\begin{pmatrix} 0 & \phi\, C_{\text{rest}} \\ \bar{\phi}^T & \lambda \quad \theta \\ \bar{C}^T_{\text{rest}} & \bar{\theta}^T A_{\text{rest}} \end{pmatrix} \to \begin{cases} \exists p\, \phi_p \neq 0 & \text{use } \phi_p \text{ to kill its row and column, drop a } \binom{01}{10} \text{ summand} \\ \phi = 0, \lambda \neq 0 & \text{use } \lambda \text{ to kill } \theta, \text{ let } s \mathrel{+}= \operatorname{sign}(\lambda) \\ \phi = 0, \lambda = 0 & \text{append } \theta \text{ to } C_{\text{rest}}. \end{cases}$$

```
Cordon_i_@Σ_B[{li___,i_,ri___},bs___][σ_, PQ[𝒞_, q_]] :=
 Module[{ϕ = ∂_γ_i 𝒞, λ = ∂_γ̄_i,γ_i q, nσ = σ, n𝒞, nq, p},
  {p} = FirstPosition[(# =!= 0) & /@ ϕ, True, {0}];
  {n𝒞, nq} = Which[
    p > 0, {𝒞, q} /. (γ_i → -𝒞[[p]] / ϕ[[p]])^+ /. (γ_i → 0)^+,
    λ =!= 0, (nσ += sign[λ];
     {𝒞, q /. (γ_i → -(∂_γ̄_i q) / λ)^+ /. (γ_i → 0)^+}),
    λ === 0, {𝒞 ⋃ {∂_γ̄_i q}, q /. (γ_i → 0)^+}];
  CF@Σ_B[Most@{ri,li},bs][nσ,
    PQ[n𝒞, nq] /. (γ_Last@{ri,li} → γ_First@{ri,li})^+] ]
```

Video and more at http://www.math.toronto.edu/~drorbn/Talks/ICERM-2305/