

Pensieve header: The Splash Program of Figure 3.1, computing  $\$z\_0\$$ .

```
SetDirectory["C:\\drorbn\\AcademicPensieve\\Projects\\PPSA\\PPSA-170103"];
```

## Splash Demo

Program0

```
CF[ $\omega\_.$   $\mathbb{E}[Q\_]$ ] := Simplify[ $\omega$ ]  $\mathbb{E}$ [Simplify[ $Q$ ]];
 $\mathbb{E}$  /:  $\mathbb{E}[Q1\_]$   $\mathbb{E}[Q2\_]$  := CF@ $\mathbb{E}[Q1 + Q2]$ ;
 $N_{u_i c_j \rightarrow k\_}$ [ $\omega\_.$   $\mathbb{E}[Q\_]$ ] := CF[ $\omega$   $\mathbb{E}[e^{-\gamma} \beta u_k + \gamma c_k + (Q / . c_j | u_i \rightarrow \theta)]$ ] /. { $\gamma \rightarrow \partial_{c_j} Q$ ,  $\beta \rightarrow \partial_{u_i} Q$ };
 $N_{w_i c_j \rightarrow k\_}$ [ $\omega\_.$   $\mathbb{E}[Q\_]$ ] := CF[ $\omega$   $\mathbb{E}[e^{\gamma} \alpha w_k + \gamma c_k + (Q / . c_j | w_i \rightarrow \theta)]$ ] /. { $\gamma \rightarrow \partial_{c_j} Q$ ,  $\alpha \rightarrow \partial_{w_i} Q$ };
 $N_{w_i u_j \rightarrow k\_}$ [ $\omega\_.$   $\mathbb{E}[Q\_]$ ] := CF[ $\nu$   $\omega$   $\mathbb{E}[-b \nu \alpha \beta + \nu \beta u_k + \nu \delta u_k w_k + \nu \alpha w_k + (Q / . w_i | u_j \rightarrow \theta)]$ ] /.  $\nu \rightarrow (1 + b \delta)^{-1}$  /.
  { $\alpha \rightarrow \partial_{w_i} Q / . u_j \rightarrow \theta$ ,  $\beta \rightarrow \partial_{u_j} Q / . w_i \rightarrow \theta$ ,  $\delta \rightarrow \partial_{w_i, u_j} Q$ };
 $m_{i_, j\_ \rightarrow k\_}$ [ $\omega\_.$   $\mathbb{E}[Q\_]$ ] := CF[Module[{ $x$ }, ( $\omega$   $\mathbb{E}[Q]$  //  $N_{w_i c_j \rightarrow x}$  //  $N_{u_i c_x \rightarrow x}$  //  $N_{w_x u_j \rightarrow x}$ ) /. { $c_i \rightarrow c_k$ ,  $w_j \rightarrow w_k$ ,  $y_{-x} \rightarrow y_k$ }]]]
```

Data0

```
 $R_{i_, j\_}^+$  :=  $\mathbb{E}[b c_j + b^{-1} (e^b - 1) u_i w_j]$ ;
 $R_{i_, j\_}^-$  :=  $\mathbb{E}[-b c_j + b^{-1} (e^{-b} - 1) u_i w_j]$ ;
```

Run0

```
 $z = R_{12,1}^- R_{2,7}^- R_{8,3}^- R_{4,11}^- R_{16,5}^+ R_{6,13}^+ R_{14,9}^+ R_{10,15}^+$ ;
Do[ $z = z$  //  $m_{1,n \rightarrow 1}$ , { $n$ , 2, 16}];
CF@ $z$  /.  $\omega\_.$   $\mathbb{E}[Q\_]$   $\Rightarrow \omega^{-1}$  /.  $b \rightarrow \text{Log}[t]$ 
```

Run0

$$\frac{1 - 4t + 8t^2 - 11t^3 + 8t^4 - 4t^5 + t^6}{t^3}$$

## Exporting the above as PDF files

The below is adapted from pensieve://2016-04/GaussGassner/GaussGassnerDemo.nb.

```
ConditionalExport[fname_String, rest___] := Module[{temp, exists},
  temp = "ConditionalExportTemporary" <> "." <> FileExtension[fname];
  exists = FileExistsQ[fname];
  Export[temp, rest];
  If[exists && FileByteCount[fname] === FileByteCount[temp],
    DeleteFile[temp],
    (* else *) Print["Exporting " <> fname <> "..."];
    If[exists, DeleteFile[fname]];
    RenameFile[temp, fname]
  ];
  fname
]

SetOptions[$FrontEndSession, PrintingStyleEnvironment -> "Working"];
TagProperties[_] := {};
TagProperties["ct-def"] = {PageWidth -> 5/0.66};
Options[CellExport] = {
  PageWidth -> 3.2/0.66 * 4.2/3.8, CellFilter -> Identity, ExportDirectory -> "Snips",
  ExportBaseFilename -> Automatic, ExportFormat -> ".pdf", ExportOptions -> {}, Split -> False
};
CellExport[tag_String, opts___Rule] := CellExport[
  NotebookGet[EvaluationNotebook[]],
  tag, opts
];
CellExport[nb_Notebook, tag_String] := CellExport[nb, tag, TagProperties[tag]];
CellExport[nb_Notebook, tag_String, OptionsPattern[]] := Module[
  {cells, cell, filename, format},
```

```

filename = FileNameJoin[{
  OptionValue[ExportDirectory] /. Automatic → Directory[],
  OptionValue[ExportBaseFilename] /. Automatic → tag
}];
format = OptionValue[ExportFormat];
cells = OptionValue[CellFilter][Cases[
  nb, c_Cell /; FreeQ[List@@c, Cell] && !FreeQ[c, CellTags → tag],
  Infinity
]];
If[!OptionValue[Split],
  If[Length[cells] ≥ 1,
    If[Length[cells] == 1,
      cells = Join[First[cells],
        Cell[PageWidth → 1.2 × 72 OptionValue[PageWidth], Background → {White, Opacity[0]}]],
      cells = Cell[CellGroup[cells], PageWidth → 72 OptionValue[PageWidth]]
    ];
  ConditionalExport[
    filename <> format, cells,
    ImageResolution → 300,
    OptionValue[ExportOptions]
  ]
],
k = 0;
Table[
  ++k;
  ConditionalExport[
    filename <> "-" <> ToString[k] <> format,
    Append[cell, PageWidth → 72 OptionValue[PageWidth]],
    ImageResolution → 300,
    OptionValue[ExportOptions]
  ],
  {cell, cells}
]
];

```

```

ExportCells := (
  nb = NotebookGet[EvaluationNotebook[]];
  tags = Cases[nb, (CellTags → tag_String) ⇒ tag, Infinity] // Union;
  Print[tags];
  CellExport /@ tags;
  Print["Done."]
);

```

### ExportCells

```
{Data0, Program0, Run0}
```

```
Exporting Snips\Data0.pdf...
```

```
Exporting Snips\Program0.pdf...
```

```
Exporting Snips\Run0.pdf...
```

```
Done.
```