Proof of Equivalence

We show by rewriting that the proposed algorithm returns a result that is equivalent to the result created by the definition of the generalised group-by operator:

$$_{GC(t_1,t_2)}\mathcal{G}_{p_1 \mapsto F_1, p_2 \mapsto F_2, \ldots p_n \mapsto F_n} R$$

The algorithm to compute the generalised group-by using vanilla relational algebra operators is as follows.

$\mathit{result} = \emptyset$
for $t \in \mathit{Pop}(R)$:
  $R' = \sigma_{GC(t,\cdot)} R$
  if $R' = \{t\}$:
   $\{q_1, \ldots, q_m\} = \{p_1, \ldots, p_n\} \cup \mathit{Schema}(R)$
   $\{s_1, \ldots, s_k\} = \{p_1, \ldots, p_n\} - \mathit{Schema}(R)$
   $\mathit{result} = \mathit{result} \cup \pi_{q_1,\ldots,q_m} R' \times \{s_1 \mapsto \perp, \ldots, s_k \mapsto \perp\}$
  else:
   $\mathit{result} = \mathit{result} \cup \{\{p_i \mapsto F_i(R') \mid i \in \{1, \ldots, n\}\}\}$
return result

This can be rewritten by distributing the for statement over the if statement as:

$\mathit{result} = \emptyset$
for $t \in \mathit{Pop}(R)$:
  $R' = \sigma_{GC(t,\cdot)} R$
  if $R' = \{t\}$:
   $\{q_1, \ldots, q_m\} = \{p_1, \ldots, p_n\} \cup \mathit{Schema}(R)$
   $\{s_1, \ldots, s_k\} = \{p_1, \ldots, p_n\} - \mathit{Schema}(R)$
   $\mathit{result} = \mathit{result} \cup \pi_{q_1,\ldots,q_m} R' \times \{s_1 \mapsto \perp, \ldots, s_k \mapsto \perp\}$
for $t \in \mathit{Pop}(R)$:
  $R' = \sigma_{GC(t,\cdot)} R$
  if $R' \neq \{t\}$:
   $\mathit{result} = \mathit{result} \cup \{\{p_i \mapsto F_i(R') \mid i \in \{1, \ldots, n\}\}\}$
return result

This can be rewritten by rewriting the relational algebraic expression as:

$\mathit{result} = \emptyset$
for $t \in \mathit{Pop}(R)$:
  $R' = \sigma_{GC(t,\cdot)} R$
  if $R' = \{t\}$:
   $\mathit{result} = \mathit{result} \cup \{p_i \mapsto (t[p_i] \text{ if } p_i \in \mathit{Schema}(R) \text{ else } \perp) \mid$
          $i \in \{1, 2, \ldots, n\}\}$
for $t \in \mathit{Pop}(R)$:
  $R' = \sigma_{GC(t,\cdot)} R$

if $R' \neq \{t\}$:
$\mathit{result} = \mathit{result} \cup \{\{p_i \mapsto F_i(R') \mid i \in \{1, \ldots, n\}\}\}$
return *result*

This can be rewritten by turning the for and if statements into set comprehensions as:

$$\begin{aligned}
&\mathit{result} = \{p_i \mapsto (t[p_i] \text{ if } p_i \in \mathit{Schema}(R) \text{ else } \perp) \mid \\
&\qquad i \in \{1, 2, \ldots, n\} \wedge t \in \mathit{Pop}(R) \wedge \sigma_{GC(t,\cdot)} R = \{t\}\} \\
&\mathit{result} = \mathit{result} \cup \big\{\{p_i \mapsto F_i(R') \mid i \in \{1, \ldots, n\}\} \bigm| \\
&\qquad\qquad t \in \mathit{Pop}(R) \wedge R' = \sigma_{GC(t,\cdot)} R \wedge R' \neq \{t\}\big\}
\end{aligned}$$

return *result*

This can be rewritten by rewriting the selection statements as:

$$\begin{aligned}
&\mathit{result} = \{p_i \mapsto (t[p_i] \text{ if } p_i \in \mathit{Schema}(R) \text{ else } \perp) \mid \\
&\qquad i \in \{1, 2, \ldots, n\} \wedge t \in \mathit{Pop}(R) \wedge \\
&\qquad \nexists t' \in \mathit{Pop}(R) \colon t' \neq t \wedge GC(t, t')\} \\
&\mathit{result} = \mathit{result} \cup \big\{\{p_i \mapsto F_i(R') \mid i \in \{1, \ldots, n\}\} \bigm| \\
&\qquad\qquad t \in \mathit{Pop}(R) \wedge R' \subseteq R \wedge R' \neq \{t\} \wedge \forall v \in R' \colon GC(t, v) \wedge \\
&\qquad\qquad \nexists z \in R \backslash R' \colon \exists u' \in R' \colon GC(u', z)\big\}
\end{aligned}$$

return *result*

It is easy to see that this is equivalent to the definition of the population of the generalised group-by statement and that it has the schema $\{p_1, p_2, \ldots, p_n\}$.