Program Analysis (CO470/97128/97146)
From Live Variable Analysis to Parity Analysis

Herbert Wiklicky

Department of Computing
Imperial College London

herbert@doc.ic.ac.uk
h.wiklicky@imperial.ac.uk

Autumn 2020
Live Variable Analysis

A variable is *live* at the exit from a label if there exists a path from the label to a use of the variable that does not re-define the variable. The *Live Variables Analysis* will determine:

*For each program point, which variables may be live at the exit from the point.*
Live Variable Analysis

A variable is *live* at the exit from a label if there exists a path from the label to a use of the variable that does not re-define the variable. The *Live Variables Analysis* will determine:

*For each program point, which variables may be live at the exit from the point.*

This analysis might be used as the basis for *Dead Code Elimination*. If the variable is not live at the exit from a label then, if the elementary block is an assignment to the variable, the elementary block can be eliminated.
Parity Analysis

A variable has *even* or *odd parity* at a label if we can guarantee that its value is *even* (e) or *odd* (o) for any execution of this label (not necessarily the same actual value). The *Parity Analysis* will determine:

*For each program point, what is the parity of each variable.*
A variable has even or odd parity at a label if we can guarantee that its value is even (e) or odd (o) for any execution of this label (not necessarily the same actual value). The Parity Analysis will determine:

For each program point, what is the parity of each variable.

This analysis might be used as the basis for ... (saving a bit?).
LV Analysis: Property Space

\[ \text{kill}_{LV} : \text{Block}_\star \rightarrow \mathcal{P}(\text{Var}_\star) \]
LV Analysis: Property Space

\[
\text{kill}_{LV} : \text{Block}_\star \rightarrow \mathcal{P}(\text{Var}_\star)
\]

\[
\text{gen}_{LV} : \text{Block}_\star \rightarrow \mathcal{P}(\text{Var}_\star)
\]

Important fact: Information we are interested in is in \( \mathcal{P}(\text{Var}_\star) \).
LV Analysis: Property Space

\[ \text{kill}_{LV} : \text{Block}_\star \rightarrow \mathcal{P}(\text{Var}_\star) \]

\[ \text{gen}_{LV} : \text{Block}_\star \rightarrow \mathcal{P}(\text{Var}_\star) \]

\[ \text{LV}_{entry} : \text{Lab}_\star \rightarrow \mathcal{P}(\text{Var}_\star) \]

Important fact: Information we are interested in is in \( \mathcal{P}(\text{Var}_\star) \).
LV Analysis: Property Space

\[ \text{kill}_{LV} : \text{Block}_\ast \rightarrow \mathcal{P}(\text{Var}_\ast) \]

\[ \text{gen}_{LV} : \text{Block}_\ast \rightarrow \mathcal{P}(\text{Var}_\ast) \]

\[ \text{LV}_{entry} : \text{Lab}_\ast \rightarrow \mathcal{P}(\text{Var}_\ast) \]

\[ \text{LV}_{exit} : \text{Lab}_\ast \rightarrow \mathcal{P}(\text{Var}_\ast) \]

Important fact: Information we are interested in is in \( \mathcal{P}(\text{Var}_\ast) \).
LV Analysis: Property Space

\[ \text{kill}_{LV} : \text{Block}_* \rightarrow \mathcal{P}(\text{Var}_*) \]

\[ \text{gen}_{LV} : \text{Block}_* \rightarrow \mathcal{P}(\text{Var}_*) \]

\[ \text{LV}_{\text{entry}} : \text{Lab}_* \rightarrow \mathcal{P}(\text{Var}_*) \]

\[ \text{LV}_{\text{exit}} : \text{Lab}_* \rightarrow \mathcal{P}(\text{Var}_*) \]

Important fact: Information we are interested in is in \( \mathcal{P}(\text{Var}_*) \).
LV Equations and Transfer Functions

\[ \text{LV}_{\text{exit}}(\ell) = \begin{cases} \emptyset, & \text{if } \ell \in \text{final}(S_\star) \\ \bigcup \{ \text{LV}_{\text{entry}}(\ell') \mid (\ell', \ell) \in \text{flow}^R(S_\star) \} , & \text{otherwise} \end{cases} \]

\[ \text{LV}_{\text{entry}}(\ell) = (\text{LV}_{\text{exit}}(\ell) \setminus \text{kill}_{\text{LV}}([B]_\ell)) \cup \text{gen}_{\text{LV}}([B]_\ell) \]

where \([B]_\ell \in \text{blocks}(S_\star)\)

with

\[ \text{kill}_{\text{LV}}([x := a]_\ell) = \{ x \} \]
\[ \text{kill}_{\text{LV}}([\text{skip}]_\ell) = \emptyset \]
\[ \text{kill}_{\text{LV}}([b]_\ell) = \emptyset \]

\[ \text{gen}_{\text{LV}}([x := a]_\ell) = \text{FV}(a) \]
\[ \text{gen}_{\text{LV}}([\text{skip}]_\ell) = \emptyset \]
\[ \text{gen}_{\text{LV}}([b]_\ell) = \text{FV}(b) \]
Parity Information

The LV Analysis associates to labels/blocks some information, concretely the set of live variables, i.e. a set in $\mathcal{P(Var_\star)}$. 
Parity Information

The LV Analysis associates to lables/blocks some information, concretely the set of live variables, i.e. a set in $\mathcal{P}(\text{Var}^\star)$. This is modified by local \textit{transfer functions} at each block (using $\text{kill}_{LV}$ and $\text{gen}_{LV}$) and \textit{collected} globally according to \textit{flow}.

Questions: How to modify parity information locally and how to combine it, e.g. maybe $\{ (x, e), (x, o), (y, e) \} \cup \{ (x, e), (y, e) \}$. 
Parity Information

The $LV$ Analysis associates to labels/blocks some information, concretely the set of live variables, i.e. a set in $\mathcal{P}(\mathbf{Var}_\star)$. This is modified by local transfer functions at each block (using $\text{kill}_{LV}$ and $\text{gen}_{LV}$) and collected globally according to flow.

For Parity we have identify the abstract properties to work with.
Parity Information

The LV Analysis associates to labels/blocks some information, concretely the set of live variables, i.e. a set in $\mathcal{P}(\text{Var}_*)$.

This is modified by local transfer functions at each block (using $\text{kill}_{LV}$ and $\text{gen}_{LV}$) and collected globally according to flow.

For Parity we have identify the abstract properties to work with.

- Sets in $\mathcal{P}(\text{Var}_* \times \{e, o\})$ or maybe $\mathcal{P}(\text{Var}_* \times \{e, o, ?\})$, e.g. 
  \{(x, e), (x, o), (y, e)\} \equiv \{(x, ?), (y, e)\}. 
Parity Information

The \( LV \) Analysis associates to labels/blocks some information, concretely the set of live variables, i.e. a set in \( \mathcal{P}(\text{Var}_\star) \).

This is modified by local \emph{transfer functions} at each block (using \( \text{kill}_{LV} \) and \( \text{gen}_{LV} \)) and \emph{collected} globally according to \emph{flow}.

For Parity we have identify the abstract properties to work with.

- Sets in \( \mathcal{P}(\text{Var}_\star \times \{e, o\}) \) or maybe \( \mathcal{P}(\text{Var}_\star \times \{e, o, ?\}) \), e.g. \( \{(x, e), (x, o), (y, e)\} \equiv \{(x, ?), (y, e)\} \).

- Functions in \( \text{Var}_\star \rightarrow \{e, o\} \) or better \( \text{Var}_\star \rightarrow \{e, o, ?\} \). e.g. \( \{x \mapsto ?, y \mapsto e\} \).

Questions: How to modify parity information locally and how to combine it, e.g. maybe \( \{(x, e), (x, o), (y, e)\} \cup \{(x, e), (y, e)\} \).
Parity Information

The LV Analysis associates to labels/blocks some information, concretely the set of live variables, i.e. a set in $\mathcal{P}(\text{Var}_*)$.

This is modified by local \textit{transfer functions} at each block (using kill$_{LV}$ and gen$_{LV}$) and \textit{collected} globally according to \textit{flow}.

For Parity we have identify the abstract properties to work with.

- Sets in $\mathcal{P}(\text{Var}_* \times \{e, o\})$ or maybe $\mathcal{P}(\text{Var}_* \times \{e, o, ?\})$, e.g. $\{(x, e), (x, o), (y, e)\} \equiv \{(x, ?), (y, e)\}$.

- Functions in $\text{Var}_* \rightarrow \{e, o\}$ or better $\text{Var}_* \rightarrow \{e, o, ?\}$. e.g. $\{x \mapsto ?, y \mapsto e\}$.

- represented as value tables, e.g. $\{x \mapsto ?, y \mapsto e\} = \begin{array}{c|c}
  x & y \\
  \hline
  ? & e
\end{array}$.
Parity Information

The LV Analysis associates to labels/blocks some information, concretely the set of live variables, i.e. a set in $\mathcal{P}(\text{Var}_*)$.

This is modified by local transfer functions at each block (using $\text{kill}_{LV}$ and $\text{gen}_{LV}$) and collected globally according to flow.

For Parity we have identify the abstract properties to work with.

- Sets in $\mathcal{P}(\text{Var}_* \times \{e, o\})$ or maybe $\mathcal{P}(\text{Var}_* \times \{e, o, ?\})$, e.g. \{(x, e), (x, o), (y, e)\} $\equiv$ \{(x, ?), (y, e)\}.
- Functions in $\text{Var}_* \rightarrow \{e, o\}$ or better $\text{Var}_* \rightarrow \{e, o, ?\}$. e.g. \{x $\mapsto$ ?, y $\mapsto$ e\}.
- Represented as value tables, e.g. \{x $\mapsto$ ?, y $\mapsto$ e\} = \[
\begin{array}{c|c}
 x & y \\
\hline
 ? & e \\
\end{array}
\]

Questions: How to modify parity information locally and how to combine it, e.g. maybe \{(x, e), (x, o), (y, e)\} $\cup$ \{(x, e), (y, e)\}. 