Towards a Cost Metric for Nearest Neighbor Constraints in Reversible Circuits

Abhoy Kole[1], Kamalika Datta[2], Indranil Sengupta[1], and Robert Wille[3,4]

[1] Dept. of Computer Science & Engineering, Indian Inst. of Techn., Kharagpur, India
[2] Dept. of Computer Science & Engineering, Nat. Inst. of Techn., Shillong, India
[3] Institute of Computer Science, University of Bremen, 28359 Bremen, Germany
[4] Cyber-Physical Systems, DFKI GmbH, 28359 Bremen, Germany
abhoy.kole@yahoo.com, kdatta@nitm.ac.in, isg@iitkgp.ac.in, rwille@informatik.uni-bremen.de

Abstract. This work in progress report proposes a new metric for estimating nearest neighbor cost at the reversible circuit level. This is in contrast to existing literature where nearest neighbor constraints are usually considered at the quantum circuit level. In order to define the metric, investigations on a state-of-the-art reversible to quantum mapping scheme have been conducted. From the retrieved information, a proper estimation to be used as a cost metric has been obtained. Using the metric, it becomes possible for the first time to optimize a *reversible* circuit with respect to nearest neighbor constraints.

Keywords: Quantum cost, nearest neighbor cost, quantum circuit, reversible circuit

1 Introduction

Motivated by the promises of quantum computation [6] researchers started to investigate how to efficiently synthesize quantum circuits. This eventually established a design flow for quantum circuits representing Boolean components which (1) realizes the desired functionality as a reversible circuit (using methods e.g. proposed in [4, 11, 13, 12, 7]) and (2) maps the resulting circuit in its respective technological quantum circuit description (using mapping schemes as e.g. proposed in [1, 3, 5, 14]).

However, while this design flow leads to proper results, it does not consider certain technological constraints. In particular, so-called *nearest neighbor constraints* are not considered by this flow, although many important quantum computing technologies heavily rely on them. In order to satisfy these constraints, it has to be ensured that computations are only performed between adjacent (i.e. nearest neighbor) signals.

A major problem is thereby that methods considering nearest neighbor constraints are usually applicable at the quantum circuit level only (see e.g. [9, 10, 16]). This is mainly caused by the absence of proper cost metrics which could be applied at the reversible logic level. In the established design flow sketched above, the handling of nearest neighbor constraints is indeed considered as another separate (third) design step which is applied not until the technology mapping of the second step has been completed.

In this work in progress, we aim for overcoming this drawback and for allowing nearest neighbor optimization at the reversible logic level – the abstraction level in which the actual synthesis is performed. For this purpose, we propose a cost metric which, for the first time, can be used to evaluate nearest neighbor constraints for reversible circuits rather than quantum circuits. In order to define the metric, investigations on a state-of-the-art reversible to quantum mapping scheme have been conducted. From the retrieved information, a proper estimation to be used as a cost metric has been obtained.

2 A Nearest Neighbor Cost Metric for the Reversible Logic Level

In general, the *Nearest Neighbor Costs* (NNC) for a quantum gate circuit are defined as the number of SWAP gates needed to make it nearest neighbor compliant. Thus far, the various works that have been reported to make a circuit nearest neighbor compliant target quantum circuits only and are unable to provide any cost estimate e.g. for synthesis at the reversible logic level[1]. In this work, we propose a cost metric which serves this purpose. To this end, we investigate a state-of-the-art reversible to quantum mapping scheme and derive systematic information to be utilized in order to formulate an NNC metric for the reversible logic level. In this section, the underlying reversible to quantum mapping scheme is reviewed first. Afterwards, we summarize our analyzes and, eventually, present the resulting metric.

2.1 Mapping of Reversible to Quantum Circuits

Our investigations are based on the mapping scheme as introduced by Miller et al. in [5]. The general idea is to partition a Toffoli gate $g(C;t)$ with a set of control lines C into a cascade of smaller gates including subsets C_1, C_2 with $C = C_1 \cup C_2$ and $C_1 \cap C_2 = \emptyset$. For this purpose, a so-called *ancilla line* (denoted by a) with $a \notin C$ and $a \neq t$ is additionally applied. More precisely,

$$\begin{aligned} T(C;t) = V(a;t)T(C_1;a)V^\dagger(a;t)T(C_2;a) \\ V(a;t)T(C_1;a)V^\dagger(a;t)T(C_2;a) \end{aligned} \tag{1}$$

is applied where $T(C;t)$ denotes an MCT gate, $V(a;t)$ denotes a V-gate, and $V^\dagger(a;t)$ denotes a $V^\dagger$-gate with the respective control and target lines. Fig. 1 illustrates the resulting structure.

This partitioning is repeated until only Toffoli gates with two control lines result. While decomposing C_1 in the respective iterations, some of the control lines in C_2 can be used as ancilla. However, further ancilla lines may be required to decompose C_2. Those can be chosen as follows:

[1] Note that the authors of [2] proposed a solution to achieve adjacency of Toffoli gates. But as discussed in the previous section, this is not sufficient to also ensure nearest neighbor compliance at the quantum logic level. In [15], nearest neighbor compliance at the reversible circuit level was investigated. But here a special model (based on multi-level quantum systems) has been assumed.

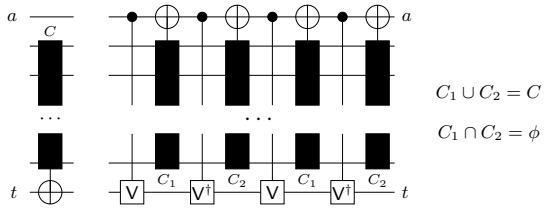

Fig. 1. Reversible to quantum mapping scheme

a) *First choice:* Use the target line t (this is possible only one time).
b) *Second choice:* Use any other free line a' with $a' \notin C_1 \cup C_2 \cup \{t\}$.
c) *Third choice:* Use any of the control lines in C_1 (this results in higher quantum cost compared to options (a) and (b)).

After all iterations have been completed, a circuit results which is composed of either V- and $V^\dagger$-gates (which are already quantum gates) or Toffoli gates with at most two control lines. These Toffoli gates are eventually mapped to quantum gate cascades as shown in Fig. 2. Overall, the functionality of the original Toffoli gate has been realized as a quantum circuit.

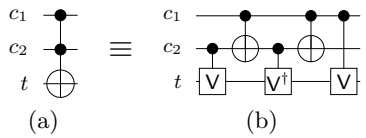

Fig. 2. Toffoli gate and its equivalent NCV cascade

In [5], further simplifications are conducted which allow for reducing the number of quantum gates in the resulting cascade by a so-called line labeling procedure. However, considering those simplifications would make the derivation an NNC cost metric significantly harder. Hence, they have been omitted in our investigations. Initial evaluations show that this has an acceptable effect on the precision of the proposed metric.

2.2 Investigations and Resulting Cost Metric

The mapping scheme reviewed above provides the basis of our investigations towards an NNC metric for the reversible logic level. According to Eqn. 1, each MCT gate $T(C;t)$ is mapped into four quantum gates $V(a;t)$ with the same control and target lines[2], two identical MCT gates $T(C_1;a)$, and another two

[2] For simplicity, V and $V^\dagger$-gates are used interchangeably in the following.

identical MCT gates $T(C_2; a)$. Hence, the NNC resulting from the mapping from the gate $T(C; t)$ has to be the sum of the NNCs resulting from each of these gates. This leads to:

$$NNC(T(C;t)) = 4 * NNC(V(a;t)) + 2 * NNC(T(C_1;a)) + 2 * NNC(T(C_2;a)), \quad (2)$$

The NNC of the quantum gates can directly be determined by considering the distance between the control and the target line. It is usually assumed that two SWAP gates are required in order to decrease this distance by one [8] – (one SWAP gate for moving the control and the target line together; another to restore the original order). Hence, assuming a numerical encoding of the control and target lines from the topmost line to the undermost line, the NNC of the quantum gates is $NNC(V(a;t)) = 2(|a-t|-1)$.

The NNC of the respective $T(C_i; a)$ gates can be computed by recursively applying Eqn. 2 together with the following base conditions:

a) $|C_i| = 1$:

$$NNC(T(\{c\};t)) = 2(|c-t|-1)$$

That is, similar to the quantum gates, the NNC is determined by considering the distance between the control and target lines.

b) $|C_i| = 2$:

$$\begin{aligned} NNC(T(\{c_1,c_2\};t)) &= 4(|c_1-c_2|-1) \\ &+ \min\{4(|c_1-t|-1), 4(|c_2-t|-1)\} \\ &+ \max\{2(|c_1-t|-1), 2(|c_2-t|-1)\} \end{aligned}$$

Here, the mapping of a Toffoli gate $T(\{c_1, c_2\}; t)$ into a a cascade of five quantum gates (shown in Fig. 2) is considered. The NNC value of the two controlled-NOT gates can be estimated as $4(|c_1-c_2|-1)$ – this is reflected in the first term. Similarly, the second term indicates the NNC value for the two controlled-V and controlled-$V^\dagger$ gates. Finally, the third term contributes to the NNC due to the controlled-V or controlled-$V^\dagger$ from the cascade. For the latter two terms, the respective configuration of the Toffoli gate $T(\{c_1, c_2\}; t)$ with respect to its control and target lines has to be taken into account. Fig. 3 shows the two possibilities. This motivates the respective application of the min/max-values.

All these observations eventually result in the following cost metric for nearest neighbor costs to be applied at the reversible logic level:

Definition 1. *Given a reversible circuit $G = g_1 g_2 \dots g_{|G|}$ composed of multiple control Toffoli gates. The* Nearest Neighbor Costs *(NNC) of G is defined as the sum of the NNCs of its gates, i.e. $NNC(G) = NNC(g_1) + NNC(g_2) + \dots + NNC(g_{|G|})$. The NNC of a gate g_i is defined as the result of the linear-time algorithm given in Fig. 4.*

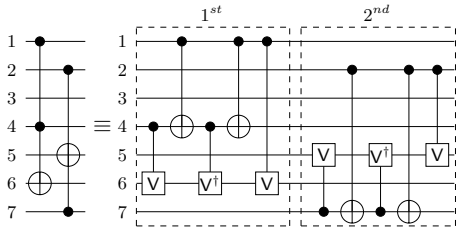

Fig. 3. Toffoli gates and their respective decomposed netlist

3 Conclusions and Future Work

In this work in progress report, we proposed a cost metric that, for the first time, allows for the consideration of nearest neighbor constraints at the reversible circuit level. Thus far, corresponding optimizations could usually be applied after technology mapping only, i.e. rather late in the design process. By investigating a state-of-the-art reversible to quantum mapping scheme, we were able to derive a proper approximation which, eventually, allows those considerations e.g. directly during the synthesis of the reversible circuit. Evaluations on the accuracy as well on the applicability of the proposed metric are left for future work.

4 Acknowledgements

This work has been supported by the Department of Science and Technology (DST) and the German Academic Exchange Service (DAAD).

References

1. Barenco, A., Bennett, H.H., Cleve, R., DiVinchenzo, D.P., Margolus, N., Shor, P., Sleator, T., Smolin, J.A., Weinfurter, H.: Elementary gates for quantum computation. Physical Review A (Atomic, Molecular, and Optical Physics) 52(5), 3457–3467 (1995)
2. Chakrabarti, A., Sur-Kolay, S., Chaudhury, A.: Linear nearest neighbor synthesis of reversible circuits by graph partitioning. CoRR (2011)
3. Maslov, D., Young, C., Dueck, G.W., Miller, D.M.: Quantum circuit simplification using templates. In: Design, Automation and Test in Europe. pp. 1208–1213 (2005)
4. Miller, D.M., Maslov, D., Dueck, G.W.: A transformation based algorithm for reversible logic synthesis. In: Design Automation Conference. pp. 318–323 (2003)
5. Miller, D.M., Wille, R., Sasanian, Z.: Elementary quantum gate realizations for multiple-control Toffolli gates. In: Intl. Symposium on Multi-valued Logic (ISMVL). pp. 217–222 (May 2011)
6. Nielsen, M., Chuang, I.: Quantum Computation and Quantum Information. Cambridge Univ. Press (2000)
7. Saeedi, M., Sedighi, M., Zamani, M.S.: A novel synthesis algorithm for reversible circuits. In: Intl. Conference on CAD. pp. 65–68. IEEE (2007)
8. Saeedi, M., Wille, R., Drechsler, R.: Synthesis of quantum circuits for linear nearest neighbor architectures. Quant. Info. Proc. 10(3), 355–377 (2011)

Algorithm NNC (C, t, a)

Inputs: MCT gate with a numerical encoding of the set of control lines C, the target line t, and the ancilla line a
Outputs: NNC of that gate

begin
  if $(|C| = 1)$
    then $NNC = 2(|c - t| - 1)$;
  else if $(|C| = 2)$
    then $NNC = 4(|c - t| - 1)$
      $+ \min \{4(|c_1 - t| - 1), 4(|c_2 - t| - 1)\}$
      $+ \max \{2(|c_1 - t| - 1), 2(|c_2 - t| - 1)\}$;
    else
      begin
        Split C into C_1 and C_2 such that $C_1 \cup C_2 = C$
          and $C_1 \cap C_2 = \phi$;
        $NNC = 4 * NNC(a, t, \phi) + 2 * NNC(C_1, a, a_1)$
          $+2 * NNC(C_2, a, a_2)$;
        // a_1, a_2 are selected ancilla lines for the mapping
      end
  return NNC;
end

Fig. 4. Algorithm determining the NNC of a reversible gate

9. Shafaei, A., Saeedi, M., Pedram, M.: Optimization of quantum circuits for interaction distance in linear nearest neighbor architectures. In: Design Automation Conf. pp. 41–46 (2013)
10. Shafaei, A., Saeedi, M., Pedram, M.: Qubit placement to minimize the communication overhead in circuits mapped to 2D quantum architectures. In: ASP Design Automation Conf. pp. 495–500 (2014)
11. Shende, V.V., Shende, A.K., Markov, I.L., Hayes, J.P.: Synthesis of reversible logic circuits. IEEE Trans. on CAD of Integrated Circuits and Systems 22(6), 710–722 (June 2003)
12. Soeken, M., Wille, R., Hilken, C., Przigoda, N., Drechsler, R.: Synthesis of reversible circuits with minimal lines for large functions. In: Asia and South Pacific Design Automation Conference. pp. 85–92 (2012)
13. Wille, R., Drechsler, R.: BDD-based synthesis of reversible logic for large functions. In: Design Automation Conference. pp. 270–275 (2009)
14. Wille, R., Soeken, M., Otterstedt, C., Drechsler, R.: Improving the mapping of reversible circuits to quantum circuits using multiple target lines. In: ASP Design Automation Conf. pp. 85–92 (2013)
15. Wille, R., Lye, A., Drechsler, R.: Considering nearest neighbor constraints of quantum circuits at the reversible circuit level. Quantum Information Processing 13(2), 185–199 (2014)
16. Wille, R., Lye, A., Drechsler, R.: Exact reordering of circuit lines for nearest neighbor quantum architectures. IEEE Trans. on CAD 33(12), 1818–1831 (2014)