Typed Graph Theory
Extending graphs with type systems

Rodrigo C. O. Rocha[1]

Abstract

In this paper, we propose *typed graph theory*, a generalisation of graph theory by extending graphs with type systems. Type theory, as a study of type systems, was originally developed as a formal system in logics. In the proposed typed graph theory, every vertex has a type and operations are restricted to vertices of a certain type. We revisit core concepts in graph theory, where new interesting properties emerge due to the proposed extension with type systems.

Keywords: graph theory, type theory, type systems, typed graphs
2010 MSC: 05C60, 05C15, 05C69, 05C05

1. Introduction

In this paper, we extend the concept of a graph with type systems, called *typed graphs*. In particular, we revisit several well-known graph properties and operations where new interesting properties emerge due to the proposed extension with type systems. The proposed extension to graph theory with type systems, that we call *typed graph theory*, is a generalisation of graph theory, and we assure to keep the generalisation consistent throughout the paper.

Type systems are subject of study in type theory and programming languages. In a system of type theory, each term has a type and operations are restricted to terms of a certain type [1, 2]. Type theory was initially proposed by Bertrand Russel [1], and further developed by others such as Whitehead and Hilbert, in order to address paradoxes in formal logics and rewriting systems. Afterwards, Alonzo Church [2] developed simply typed

[1]School of Informatics, University of Edinburgh, UK. Email:r.rocha@ed.ac.uk

λ-calculus which is a typed formalism based on λ-calculus and simple type theory. In Section 3 we define a substitution rule with a similar concept to the one used by Church [2].

2. Typed graphs

In this section, we define typed graphs as undirected graphs extended with type systems, however, these concepts can be easily extended to directed graphs.

Definition 2.1. *Let T be a non-empty set of types and let L be a set of labels*[2]*. We say that $G = (V, E, T)$ is a typed graph if (V, E) is a graph with vertex-set $V \subseteq L \times T$ and edge-set $E \subseteq V \times V$.*

We will denote vertices $(v_1, \tau_1) \in V$ by $v_1{:}\tau_1$ and edges by $\{v_1{:}\tau_1, v_2{:}\tau_2\}$. Notice that two distinct vertices $v_1{:}\tau_1, v_2{:}\tau_2 \in V$ have $v_1 \neq v_2$ but their types τ_1 and τ_2 can be the same. A homogeneous graph G of type τ is a typed graph $G = (V, E, T)$ where $T = \{\tau\}$ and therefore all vertices in V have type τ. Similarly, a *singleton* graph of type τ is a graph defined as $G = (\{v{:}\tau\}, \emptyset, \{\tau\})$. Graphs in classic graph theory are equivalent to homogeneous typed graphs and thus just special cases of typed graphs.

Definition 2.2 (Subgraph)**.** *Let $G_1 = (V_1, E_1, T_1)$ be a typed graph. $G_2 = (V_2, E_2, T_2)$ is a typed subgraph of G_1, denoted by $G_2 \subseteq G_1$, if and only if $V_2 \subseteq V_1$, $E_2 \subseteq E_1$ and $T_2 \subseteq T_1$.*

Similar to the definitions of vertex-induced subgraph and edge-induced subgraph, we also define *type-induced subgraph*, see Definition 2.3.

Definition 2.3 (Type-induced subgraph)**.** *Let $G = (V, E, T)$ be a typed graph and $T' \subseteq T$. Define the type-induced subgraph of G, denoted by $G[T']$, as the typed subgraph (V', E', T') of G such that $V' = \{v{:}\tau \in V | \tau \in T'\}$ and $E' = \{\{v_1{:}\tau_1, v_2{:}\tau_2\} \in E | v_1{:}\tau_1, v_2{:}\tau_2 \in V'\}$.*

Definition 2.4 (Isomorphism)**.** *Let $G_1 = (V_1, E_1, T_1)$ and $G_2 = (V_2, E_2, T_2)$ be typed graphs, with $V_1 \subseteq L_1 \times T_1$ and $V_2 \subseteq L_2 \times T_2$. The two typed graphs G_1 and G_2 are isomorphic, denoted $G_1 \simeq G_2$, if and only if there are bijections $\phi : L_1 \to L_2$ and $\psi : T_1 \to T_2$ such that $V_2 = \{\phi(v_i){:}\psi(\tau_i) | v_i{:}\tau_i \in V_1\}$ and $\forall \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E_1, \{\phi(v_i){:}\psi(\tau_i), \phi(v_j){:}\psi(\tau_j)\} \in E_2$.*

[2] Usually we can consider that L is the set of natural number $\mathbf{N}$.

Whilst both vertex and edge deletion are type-independent operations, vertex and edge contractions are type-dependent operations in typed graphs. As we formally define below, vertex contraction is well-defined only for pairs of vertices of the same type and, similarly, edge contraction is well-defined only for edges with endpoints of the same type.

Definition 2.5 (Vertex contraction)**.** *Let $G = (V, E, T)$ be a typed graph and $v_i{:}\tau, v_j{:}\tau \in V$ are vertices of the same type τ. Let ϕ be a function which maps every vertex in $V \setminus \{v_i{:}\tau, v_j{:}\tau\}$ to itself, and otherwise, maps it to a new vertex $w{:}\tau$. The contraction of vertices $\{v_i{:}\tau, v_j{:}\tau\}$ results in a new typed graph $G' = (V', E', T)$, where $V' = (V \setminus \{v_i{:}\tau, v_j{:}\tau\}) \cup \{w{:}\tau\}$ and E' is defined with a correspondence to E such that for every $v \in V$, $v' = \phi(v) \in V'$ is incident to an edge $e' \in E'$ if and only if, the corresponding edge, $e \in E$ is incident to v in G.*

Definition 2.6 (Edge contraction)**.** *Let $G = (V, E, T)$ be a typed graph containing an edge $\varepsilon = \{v_i{:}\tau, v_j{:}\tau\}$, with vertices of the same type τ. Let ϕ be a function which maps every vertex in $V \setminus \{v_i{:}\tau, v_j{:}\tau\}$ to itself, and otherwise, maps it to a new vertex $w : \tau$. The contraction of ε results in a new typed graph $G' = (V', E', T)$, where $V' = (V \setminus \{v_i{:}\tau, v_j{:}\tau\}) \cup \{w{:}\tau\}$ and E' is defined with a correspondence to $E \setminus \{\varepsilon\}$ such that for every $v \in V$, $v' = \phi(v) \in V'$ is incident to an edge $e' \in E'$ if and only if, the corresponding edge, $e \in E \setminus \{\varepsilon\}$ is incident to v in G.*

Definition 2.7 (Reduced normal form)**.** *A typed graph is in the reduced normal form if and only if for any given edge $\{v_i : \tau_i, v_j : \tau_j\} \in E$, we have $\tau_i \neq \tau_j$.*

If a typed graph is not in the reduced normal form, its reduced normal form can be obtained by repeatedly contracting edges with endpoints of the same type, until no further edge contraction is possible. Corollary 2.12 shows that the reduced normal form of a typed graph is unique. Denote by $\Re_G$ the reduced normal form of a typed graph G. Theorem 2.8, that follows immediately from the definition, states that if G is in the reduced normal form then $\Re_G \simeq G$ and no further reduction is possible.

Theorem 2.8. *If a typed graph is in the reduced normal form, no reduction by edge contration is possible.*

Proof. Follows immediately from the definition that a graph in the reduced normal form has no edge with both endpoints of the same type. □

Theorem 2.9. *If G is a typed graph in the reduced normal form and H is a subgraph of G, then H is in the reduced normal form.*

Proof. Suppose H is not in the reduced normal form, then there is an edge $\{v_i : \tau_i, v_j : \tau_j\}$ in the edge-set of H, with $\tau_i = \tau_j$, which is a contradiction because $\{v_i : \tau_i, v_j : \tau_j\}$ is also in the edge-set of G, which is in the reduced normal form. $\square$

Definition 2.10 (Homogeneous connected components)**.** *Let $G = (V, E, T)$ be a typed graph. Define a relation $\sim$ on V as follows: for all $v_i{:}\tau_i, v_j{:}\tau_j \in V$, define $v_i{:}\tau_i \sim v_j{:}\tau_j$ if and only if $\tau_i = \tau_j$ and there is at least one homogeneous path from $v_i{:}\tau_i$ to $v_j{:}\tau_j$, where all vertices in the path have the same type τ_i. The relation $\sim$ is an equivalence relation. The subgraphs of G induced by the equivalence classes of $\sim$ are called homogeneous connected components of G.*

Theorem 2.11. *Let $G = (V, E, T)$ be a typed graph and S be the set of homogeneous connected components of G. Let $G' = (V', E', T)$ be the graph consisting of the homogeneous connected components of G as vertices, i.e. $V' = \{H : \tau | H \in S \wedge T(H) = \{\tau\}\}$, and E' is the set of edges for which $\{H_1{:}\tau_1, H_2{:}\tau_2\} \in E'$ implies that $H_1, H_2 \in S$, with $H_1 \neq H_2$, and $\exists\{v_1{:}\tau_1, v_2{:}\tau_2\} \in E$ with $v_1{:}\tau_1 \in V(H_1)$ and $v_2{:}\tau_2 \in V(H_2)$. Therefore G' is isomorphic to $\Re_G$.*

Proof. Let $H \in S$. We have that the reduced normal form of H is a *singleton* typed graph, i.e., $\Re_H \simeq (\{w{:}\tau\}, \emptyset, \{\tau\})$, since, by definition, H is a connected graph and $\forall v_1{:}\tau_1, v_2{:}\tau_2 \in V(H)$, $\tau_1 = \tau_2$. Therefore $\Re_H$ contracts all edges of H, resulting in the *singleton* typed graph, regardless of the order the edges are contracted. The resulting contracted *singleton* typed graph corresponds to the vertex $H{:}\tau$ in the typed graph G'. However, edges of G with endpoints of different types are not contracted by $\Re_G$ and by definition they also belong to different homogeneous connected components of G, therefore belonging to both G' and $\Re_G$. $\square$

Corollary 2.12 (The reduced normal form uniqueness)**.** *If G' and G'' are reduced normal forms of a typed graph G, $G' \simeq G''$.*

Proof. Follows immediately from Theorem 2.11 and also from the fact that the set of equivalence classes of an equivalence relation is uniquely defined. $\square$

Definition 2.13 (Reduced form equivalence)**.** *Let $G_1 = (V_1, E_1, T_1)$ and $G_2 = (V_2, E_2, T_2)$ be typed graphs. If $\Re_{G_1}$ and $\Re_{G_2}$ are isomorphic, then G_1 and G_2 are said to be reduced form equivalent, denoted by $G_1 \equiv G_2$.*

Notice that the reduced form equivalence satisfies reflexivity, symmetry and transitivity. We denote by $[G]_\equiv$ the equivalence class of a graph G regarding the *reduced form equivalence* relation. Clearly $|[G]_\equiv|$ is infinite, since each vertex in the reduced normal form could be a result of reducing any connected typed graph with all vertices having the same type as the reduced vertex. We provide further analysis of the *reduced form equivalence* class in Section 3.

3. Subgraph substitution rule

In this section, we define a substitution rule that maps from one typed graph to another *reduced form equivalent* typed graph. We first define two predicates that will be used for defining the subgraph substitution rule.

Let $G = (V, E, T)$ be a typed graph and $v{:}\tau \in V$. The first predicate, $P_P(S, v{:}\tau)$, defines a set S that is a valid superset of the complete partitioning of $N(v{:}\tau)$, the set of vertices adjacent to $v{:}\tau$.

$$\begin{aligned} P_P(S, v{:}\tau) :& [\forall S' \in S, S' \neq \emptyset \implies \nexists S'' \neq S' \in S, S' \cap S'' \neq \emptyset] \wedge \\ & [\forall v'{:}\tau' \in N(v{:}\tau), \exists S' \in S, v'{:}\tau' \in S']. \end{aligned}$$

In other words, $P_P(S, v{:}\tau)$ is true for a given set S if and only if $N_P \subseteq S$, such that N_P is a partition of $N(v{:}\tau)$, i.e. N_P contains a collection of mutually disjoint non-empty sets whose union is $N(v{:}\tau)$.

The second predicate, $P_C(V', E', \tau)$, is true if and only if $(V', E', \{\tau\})$ is an homogeneous connected component.

$$P_C(V', E', \tau) : \forall S'{:}\tau \in V', \forall S''{:}\tau \in V', S'{:}\tau \sim S''{:}\tau$$

where $\sim$ is the relation presented in Definition 2.10.

Let $\mathcal{G}(v{:}\tau)$ such that

$$\mathcal{G}(v{:}\tau) = \{(V', E', \{\tau\} | V' = \{S'{:}\tau | S' \in S\} \wedge P_P(S, v{:}\tau) \wedge P_C(V', E', \tau)\}.$$

$\mathcal{G}(v : \tau)$ defines the set of all typed graphs $G' \in \mathcal{G}(v : \tau)$ such that $\Re_{G'}$ is isomorphic to the *singleton* graph $(\{v{:}\tau\}, \emptyset, \{\tau\})$, i.e., $\mathcal{G}(v{:}\tau)$ is isomorphically equivalent to the set of all connected homogeneous typed graphs with type-set $\{\tau\}$.

Definition 3.1 (Vertex substitution)**.** *Let $G = (V, E, T)$ be a typed graph and $v{:}\tau \in V$. Let $H \in \mathcal{G}(v{:}\tau)$.*

$$\mathcal{S}_H^{v:\tau}|G = (V', E', T)$$

such that $V' = (V \setminus \{v{:}\tau\}) \cup V(H)$ and $E' = (E \setminus E'') \cup E(H) \cup E'''$ with $E'' = \{\{v{:}\tau, v'{:}\tau'\}|v'{:}\tau' \in N(v{:}\tau)\}$ and $E''' = \{\{S'{:}\tau, v'{:}\tau'\}|v'{:}\tau' \in N(v{:}\tau) \wedge S'{:}\tau \in V(H)\}$. $V(H)$ is the vertex-set of H and $E(H)$ is the edge-set of H.

Theorem 3.2. *Let $G = (V, E, T)$ be a typed graph and $v:\tau \in V$. For all $H \in \mathcal{G}(v{:}\tau)$ we have that $\mathcal{S}_H^{v:\tau}|G \equiv G$.*

Theorem 3.2 follows directly from the definition of *vertex substitution.* $\mathcal{S}_H^{v:\tau}|G$ substitutes a vertex $v{:}\tau$ by a connected typed graph that is *reduced form equivalent* to the *singleton* graph $(\{v{:}\tau\}, \emptyset, \{\tau\})$.

Theorem 3.3. *Let G be a typed graph and $\Re_G = (V, E, T)$ be its reduced normal form, where $V = \{v_1{:}\tau_1, v_2{:}\tau_2, \ldots, v_n{:}\tau_n\}, n \in \mathbf{N}$. Let K be the set*

$$K = \{\mathcal{S}_{H_1}^{v_1:\tau_1}|\mathcal{S}_{H_2}^{v_2:\tau_2}|\ldots|\mathcal{S}_{H_n}^{v_n:\tau_n}|\Re_G \quad | \ H_i \in \mathcal{G}(v_i{:}\tau_i), v_i{:}\tau_i \in V\}$$

where $\mathcal{S}_{H_1}^{v_1:\tau_1}|\mathcal{S}_{H_2}^{v_2:\tau_2}|\cdots|\mathcal{S}_{H_n}^{v_n:\tau_n}|\Re_G$ represents the consecutive application of the substitution rule, i.e., $\mathcal{S}_{H_1}^{v_1:\tau_1}|\left(\mathcal{S}_{H_2}^{v_2:\tau_2}|\left(\cdots|\left(\mathcal{S}_{H_n}^{v_n:\tau_n}|\Re_G\right)\cdots\right)\right)$. Therefore,

$$(\forall G' \in K, \exists G'' \in [G]_\equiv, G' \simeq G'') \wedge (\forall G' \in [G]_\equiv, \exists G'' \in K, G' \simeq G'')$$

Proof. It follows directly from the definition of the substitution rule and the definition of $\mathcal{G}(v_i{:}\tau_i), v_i{:}\tau_i \in V$, since $\mathcal{G}(v_i{:}\tau_i)$ is isomorphically equivalent to the set of all connected typed graphs with type-set $\{\tau_i\}$. By substituting all vertices $v_i{:}\tau_i$ in the reduced graph of G, by any other H_i of the same type, we can produce typed graphs isomorphic to any graph in the equivalence class $[G]_\equiv$. □

Proposition 3.4 analyses the size of a restricted subset of this equivalence class, which is itself finite if and only if the typed graph in reduced normal form is also finite.

Proposition 3.4. *Let $G = (V, E, T)$ be a typed graph and $v{:}\tau \in V$. Consider a restricted form of the first predicate that accepts only sets that is itself a complete partitioning of $N(v{:}\tau)$, i.e., $P_P^*(S, v{:}\tau) : P_P(S, v{:}\tau) \wedge [\forall S' \in S, S' \subseteq N(v{:}\tau)]$. In other words, $P_P(S, v{:}\tau)$ is true for a given set S if and only if S*

itself is a partition of $N(v{:}\tau)$, i.e. S contains a collection of mutually disjoint non-empty sets whose union is $N(v{:}\tau)$. Consider also a restricted subset of $\mathcal{G}(v{:}\tau)$, namely, $\mathcal{G}^*(v{:}\tau)$, such that

$$\mathcal{G}^*(v{:}\tau) = \{(V', E', \{\tau\} | V' = \{S'{:}\tau | S' \in S\} \wedge P_P^*(S, v{:}\tau) \wedge P_C(V', E', \tau)\}.$$

Finally, consider $K \subseteq [G]_\equiv$ such that

$$K = \{\mathcal{S}_{H_1}^{v_1:\tau_1} | \mathcal{S}_{H_2}^{v_2:\tau_2} | \dots | \mathcal{S}_{H_n}^{v_n:\tau_n} | G \quad | \quad H_i \in \mathcal{G}^*(v_i{:}\tau_i), v_i{:}\tau_i \in V\}$$

Therefore

$$\prod_{v:\tau \in V} \binom{d(v{:}\tau)}{2} \leq |K| < \prod_{v:\tau \in V} \binom{d(v{:}\tau)}{2} 2^{\binom{d(v:\tau)}{2}}$$

Proof. Let $d' = d(v{:}\tau)$. Notice that $\{S | P_P^*(S, v{:}\tau)\}$ is the set of all valid partitions of the set $N(v{:}\tau)$ into exactly d' partitions, where some of the partitions may receive no element of $N(v{:}\tau)$. We can partition $N(v{:}\tau)$ in a total of $\binom{d'}{2}$ different ways. Thus $|\{S | P_P^*(S, v{:}\tau)\}| = \binom{d'}{2}$.

For any valid vertex-set $V' = \{S'{:}\tau | S' \in S\}$, with $P_P^*(S, v{:}\tau)$ true, the set $\{E' | P_C(V', E', \tau)\}$ represents the set of all edge-sets such that the resulting graph is connected. Therefore, $|\{E' | P_C(V', E', \tau)\}| < 2^{\binom{d'}{2}}$, which also follows that $\binom{d'}{2} \leq |\mathcal{G}^*(v{:}\tau)| < \binom{d'}{2} 2^{\binom{d'}{2}}$.

If we repeat this process for all vertices in V we can easily conclude that

$$\prod_{v:\tau \in V} \binom{d(v{:}\tau)}{2} \leq |K| < \prod_{v:\tau \in V} \binom{d(v{:}\tau)}{2} 2^{\binom{d(v:\tau)}{2}}$$

$\square$

We have defined and studied a vertex substitution rule. We can also generalise this substitution rule as a subgraph substitution rule. Let $G' = (V', E', \{\tau\})$, $\tau \in T$, be a connected homogeneous subgraph of type τ of G, i.e. $G' \subseteq G$ and G' have all vertices of type τ. Let $N(G')$ be the set $\{v{:}\tau \in V \setminus V' | \{v{:}\tau, v'{:}\tau'\} \in E \wedge v'{:}\tau' \in V'\}$. We first consider another predicate, $P_P(S, G', \tau)$, that defines a set S that is a valid superset of the complete partitioning of $N(G')$.

$$\begin{aligned} P_P(S, G') :& [\forall S' \in S, S' \neq \emptyset \wedge \nexists S'' \neq S' \in S, S' \cap S'' \neq \emptyset] \wedge \\ & [\forall v'{:}\tau' \in N(G'), \exists S' \in S, v'{:}\tau' \in S']. \end{aligned}$$

Let $\mathcal{G}(G')$ such that

$$\mathcal{G}(G') = \{(V', E', \{\tau\})|V' = \{S'{:}\tau|S' \in S\} \wedge P_P(S, G') \wedge P_C(V', E', \tau)\}.$$

$\mathcal{G}(G')$ defines the set of all typed graphs $G'' \in \mathcal{G}(v{:}\tau)$ such that G'' is *reduced form equivalent* to G'. The set $\mathcal{G}(G')$ also represents the set of all connected typed graphs with type-set $\{\tau\}$.

Definition 3.5 (Subgraph substitution)**.** *Let $G = (V, E, T)$ be a typed graph and G' be a homogeneous connected subgraph of G, where all vertices of G' have the same type $\tau \in T$. Let $H \in \mathcal{G}(G')$.*

$$\mathcal{S}_H^{G'}|G = (V', E', T)$$

such that $V' = (V \setminus V(G')) \cup V(H)$ and $E' = ((E \setminus E(G')) \setminus E'') \cup E(H) \cup E'''$ where $E'' = \{\{v{:}\tau, v'{:}\tau'\} \in E | v{:}\tau \in V \setminus V(G') \wedge v'{:}\tau' \in V(G')\}$ and $E''' = \{\{S'{:}\tau', v{:}\tau\}|v{:}\tau \in N(G') \wedge S'{:}\tau' \in V(H) \wedge v{:}\tau \in S'\}$.

Theorem 3.6. *Let $G = (V, E, T)$ be a typed graph and G', be a connected subgraph of G, with all vertices of G' having the same type $\tau \in T$. For all $H \in \mathcal{G}(G')$ we have that $\mathcal{S}_H^{G'}|G \equiv G$.*

Theorem 3.6 follows directly from the definition of *subgraph substitution.* $\mathcal{S}_H^{G'}|G \equiv G$ substitutes a connected subgraph G' with all vertices having the same type $v : \tau$, i.e. G' is *reduced form equivalent* to the *singleton* graph $(\{v{:}\tau\}, \emptyset, \{\tau\})$, by another connected typed graph that is also *reduced form equivalent* to the *singleton* graph with type τ.

4. Vertex colouring

Vertex colouring is a labelling of the vertices of a graph given some restrictions. In this section, we provide two definitions for vertex colouring in typed graphs: (i) *type-restrictive vertex colouring* is a labelling of the vertices such that no two adjacent vertices or no two vertices of different types can have the same colour; (ii) *type-permissive vertex colouring* is a labelling of the vertices such that no two adjacent vertices of the same type can have the same colour.

Definition 4.1 (Type-restrictive vertex colouring)**.** *Let $\kappa_R : V \to \mathbf{N}$ be the type-restrictive vertex colouring of a typed graph $G = (V, E, T)$. For $v_i{:}\tau_i, v_j{:}\tau_j \in V$, if $\tau_i \neq \tau_j$ or $\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$ then $\kappa_R(v_i{:}\tau_i) \neq \kappa_R(v_j{:}\tau_j)$.*

Definition 4.2 (Type-permissive vertex colouring)**.** *Let $\kappa_P : V \to \mathbf{N}$ be the type-permissive vertex colouring of a typed graph $G = (V, E, T)$. For $v_i{:}\tau_i, v_j{:}\tau_j \in V$, if $\tau_i = \tau_j$ and $\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$ then $\kappa_P(v_i{:}\tau_i) \neq \kappa_P(v_j{:}\tau_j)$.*

If the type-set has size one, both typed vertex colouring definitions simplifies to the classic definition of vertex colouring. Theorem 4.3 shows that both typed vertex colouring definitions are equivalent when the typed graph has type-set of size one.

Theorem 4.3. *Let $G = (V, E, T)$ be a typed graph. If $|T| = 1$, then the set of type-restrictive vertex colouring equals the set of type-permissive vertex colouring.*

Proof. Consider the two predicates $P_E(v_i{:}\tau_i, v_j{:}\tau_j) : \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$ and $P_T(v_i{:}\tau_i, v_j{:}\tau_j) : (\tau_i = \tau_j)$. Let $\kappa : V \to \mathbf{N}$. Define the following two predicates:

$$P_R(\kappa)\colon \forall v{:}\tau \in V, \nexists v'{:}\tau' \neq v{:}\tau \in V, \kappa(v{:}\tau) = \kappa(v'{:}\tau') \wedge (\overline{P_T(v{:}\tau, v'{:}\tau')} \vee P_E(v{:}\tau, v'{:}\tau'))$$

$$P_P(\kappa)\colon \forall v{:}\tau \in V, \nexists v'{:}\tau' \neq v{:}\tau \in V, \kappa(v{:}\tau) = \kappa(v'{:}\tau') \wedge P_T(v{:}\tau, v'{:}\tau') \wedge P_E(v{:}\tau, v'{:}\tau')$$

Notice that $\kappa : V \to \mathbf{N}$ is a type-restrictive vertex colouring of G if and only if $P_R(\kappa)$ is true and that $\kappa : V \to \mathbf{N}$ is a type-permissive vertex colouring of G if and only if $P_P(\kappa)$ is true. Since all vertices have the same type, $P_T(v{:}\tau, v'{:}\tau')$ is always true for all $v{:}\tau, v'{:}\tau' \in V$. Therefore both predicates simplify to

$$P_K(\kappa) : \forall v{:}\tau \in V, \nexists v'{:}\tau' \neq v{:}\tau \in V, (\kappa(v{:}\tau) = \kappa(v'{:}\tau')) \wedge P_E(v{:}\tau, v'{:}\tau')$$

□

Notice also that the predicate $P_K(\kappa : V \to \mathbf{N})$ is equivalent to the classic definition of vertex colouring.

We call *type-restrictive chromatic number* the minimum number of colours by a type-restrictive vertex colouring, denoted by $\chi_R(G)$. Similarly, we call *type-permissive chromatic number* the minimum number of colours by a type-permissive vertex colouring, denoted by $\chi_P(G)$.

Theorem 4.4. *Let $G = (V, E, T)$ be a typed graph in the reduced normal form. The type-restrictive chromatic number of G is $\chi_R(G) = |T|$.*

Proof. By definition, we have that $\chi_R(G) \geq |T|$, since no two vertices of different types can have the same colour. However, $\nexists\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E, \tau_i = \tau_j$, and thus all vertices of the same type can have the same colour, since they are all independent. Therefore, $\chi_R(G) = |T|$. $\square$

Theorem 4.5. *Let $G = (V, E, T)$ be a typed graph in the reduced normal form. The type-permissive chromatic number of G is $\chi_P(G) = 1$.*

Proof. By definition, $\forall\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E, \tau_i \neq \tau_j$. Therefore, by definition, there is a type-permissive vertex colouring $\kappa_P : V \to \mathbf{N}$ such that $\forall v_i{:}\tau_i, v_j{:}\tau_j \in V, \kappa_P(v_i{:}\tau_i) = \kappa_P(v_j{:}\tau_j)$, since $\tau_i \neq \tau_j$. Hence, $\chi_P(G) = 1$. $\square$

Theorem 4.6. *In a type-restrictive vertex colouring, whenever two vertices can have the same colour, they can also have the same colour in a type-permissive vertex colouring, but not vice versa. Formally: Let $G = (V, E, T)$ be a typed graph. Then*

$$\begin{aligned} &\forall \kappa_R : V \to \mathbf{N}, \forall v{:}\tau, v'{:}\tau' \in V, \kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau') \\ &\qquad \Longrightarrow \exists \kappa_P : V \to \mathbf{N}, \kappa_P(v{:}\tau) = \kappa_P(v'{:}\tau') \end{aligned}$$

and

$$\begin{aligned} &\forall \kappa_P : V \to \mathbf{N}, \forall v{:}\tau, v'{:}\tau' \in V, \kappa_P(v{:}\tau) = \kappa_P(v'{:}\tau') \\ &\qquad \not\Longrightarrow \exists \kappa_R : V \to \mathbf{N}, \kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau') \end{aligned}$$

such that $\kappa_R : V \to \mathbf{N}$ is a type-restrictive vertex colouring and $\kappa_P : V \to \mathbf{N}$ is a type-permissive vertex colouring.

Proof. Consider again the two predicates $P_E(v_i{:}\tau_i, v_j{:}\tau_j) : \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$ and $P_T(v_i{:}\tau_i, v_j{:}\tau_j) : (\tau_i = \tau_j)$. Let us prove the first statement. By definition, $\kappa_R(v:\tau) = \kappa_R(v':\tau')$ implies that $P_T(v:\tau, v':\tau') \land \overline{P_E(v{:}\tau, v'{:}\tau')}$. Because $\overline{P_E(v{:}\tau, v'{:}\tau')}$ is true, i.e. $\{v : \tau, v' : \tau'\} \notin E$, by definition it follows that $\exists \kappa_R : V \to \mathbf{N}, \kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau')$.

Now, let us prove the second statement by contradiction. Suppose

$$\begin{aligned} &\forall \kappa_P : V \to \mathbf{N}, \forall v{:}\tau, v'{:}\tau' \in V, \kappa_P(v{:}\tau) = \kappa_P(v'{:}\tau') \\ &\qquad \Longrightarrow \exists \kappa_R : V \to \mathbf{N}, \kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau'). \end{aligned}$$

By definition, $\kappa_P(v{:}\tau) = \kappa_P(v'{:}\tau')$ implies that $\overline{P_T(v{:}\tau, v'{:}\tau')} \lor \overline{P_E(v{:}\tau, v'{:}\tau')}$. Therefore, either $\overline{P_T(v{:}\tau, v'{:}\tau')}$ is true, or $\overline{P_E(v{:}\tau, v'{:}\tau')}$ is true, or both are true. However, if $\overline{P_T(v{:}\tau, v'{:}\tau')}$ is true, then $v{:}\tau$ and $v'{:}\tau'$ have different types, i.e., $\tau \neq \tau'$. Therefore, by definition, $\nexists \kappa_R : V \to \mathbf{N}, \kappa_R(v:\tau) = \kappa_R(v':\tau')$, which is a contradiction. $\square$

Theorem 4.6 suggests that type-restrictive vertex colouring is more prohibitive for repeating colours than type-permissive vertex colouring. Corollary 4.7 follows directly from Theorem 4.6, because we can always find a type-permissive vertex colouring that repeats at least as many colours as the type-restrictive vertex colouring.

Corollary 4.7. *Let $G = (V, E, T)$ be a typed graph. Thus $\chi_P(G) \leq \chi_R(G)$.*

Theorem 4.8. *Let $G = (V, E, T)$ be a typed graph, with $|T| = k$, and S be the set of homogeneous connected components of G. For each $\tau_i \in T$, let $H_i = \bigcup\{S' \in S | T(S') = \{\tau_i\}\}$, i.e. H_i is the union of the homogeneous connected components of G with all vertices having the same type τ_i. Let $\kappa_1, \kappa_2, \dots, \kappa_k : V \to \mathbf{N}$, such that $\kappa_i : V \to \mathbf{N}$ is a type-restrictive vertex colouring for the subgraph H_i, for all $i \in [1, k]$. If $\mathrm{Im}\ \kappa_i \cap \mathrm{Im}\ \kappa_j = \emptyset$, then $\bigcup_{i=1}^{k} \kappa_i$ is a type-restrictive vertex colouring of G.*

Proof. Let us prove by contradiction. Suppose $\kappa_R = \bigcup_{i=1}^{k} \kappa_i$ is not a type-restrictive vertex colouring. Therefore $\exists v{:}\tau, v'{:}\tau' \in V$ such that $\kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau') \wedge (\tau \neq \tau' \vee \{v{:}\tau, v'{:}\tau'\} \in E)$. Let us analyse $\kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau') \wedge \tau \neq \tau'$ and $\kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau') \wedge \{v{:}\tau, v'{:}\tau'\} \in E$ in two separate cases and show that they both never occur.

Because $\tau \neq \tau'$ implies that $v{:}\tau$ and $v'{:}\tau'$ belong two different subgraphs, resulting from the union of homogeneous connected components of the same type, and therefore $\kappa_R(v{:}\tau) \neq \kappa_R(v'{:}\tau')$ since κ_i have disjoint image sets for different types. Therefore $\kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau') \wedge \tau \neq \tau'$ is not possible.

For the second case, we have shown that $\tau \neq \tau'$ contradicts $\kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau')$, then we consider that $\tau = \tau'$. However, $\tau = \tau' \wedge \{v{:}\tau, v'{:}\tau'\} \in E$ implies that $v : \tau$ and $v' : \tau'$ are in the same subgraph, resulting from the union of homogeneous connected components of the same type, and are also adjacent vertices. Which contradicts the assumption that k_i are valid type-restrictive vertex colouring for the subgraph H_i, for all $i \in [1, k]$. Therefore, $\kappa_R(v{:}\tau) = \kappa_R(v'{:}\tau') \wedge \{v{:}\tau, v'{:}\tau'\} \in E$ is also not possible.

Since both cases are not possible, we have a contradiction. □

Corollary 4.9. *Let $G = (V, E, T)$ be a typed graph, with $|T| = k$, and S be the set of homogeneous connected components of G. For each $\tau_i \in T$, let $H_i = \bigcup\{S' \in S | T(S') = \{\tau_i\}\}$, i.e. H_i is the union of the homogeneous connected components of G with all vertices having the same type τ_i. Let $\kappa_1, \kappa_2, \dots, \kappa_k : V \to \mathbf{N}$, such that $\kappa_i : V \to \mathbf{N}$ is a minimum type-restrictive* *vertex colouring for the subgraph H_i, for all $i \in [1,k]$. If* $\text{Im } \kappa_i \cap \text{Im } \kappa_j = \emptyset$, *then* $\bigcup_{i=1}^{k} \kappa_i$ *is a minimum type-restrictive vertex colouring of* G.

Proof. Suppose $\bigcup_{i=1}^{k} \kappa_i$ is not a minimum type-restrictive vertex colouring of G. Then there is a minimum type-restrictive vertex colouring $k : V \to \mathbf{N}$, such that $|\text{Im}\kappa| < |\text{Im} \bigcup_{i=1}^{k} \kappa_i|$. By definition, if we denote by $\text{Im}[\tau]\kappa$ the image of the colouring function κ for the vertices of type τ, i.e. $\text{Im}[\tau]\kappa = \{\kappa(v\!:\!\tau) | v\!:\!\tau \in V[\tau]\}$, then we know that $\tau_i, \tau_j \in T, \tau_i \neq \tau_j$ implies that $(\text{Im}[\tau_i]\kappa) \cap (\text{Im}[\tau_j]\kappa)$, because no two vertices of different types can have the same colour. Therefore,

$$
\begin{aligned}
|\text{Im}\kappa| &< |\text{Im} \bigcup_{i=1}^{k} \kappa_i| \\
|\bigcup_{i=1}^{k} (\text{Im}[\tau_i]\kappa)| &< |\text{Im} \bigcup_{i=1}^{k} \kappa_i| \\
\sum_{i=1}^{k} |\text{Im}[\tau_i]\kappa| &< \sum_{i=1}^{k} |\text{Im}\kappa_i|
\end{aligned}
$$

and thus $\exists \tau_i \in T$ such that $|\text{Im}[\tau_i]\kappa < \text{Im}\kappa_i$, which is a contradiction, since $\kappa_i : V \to \mathbf{N}$ is a minimum type-restrictive vertex colouring for the subgraph H_i, for all $i \in [1,k]$. $\square$

In this section, we studied two definitions of vertex colouring. Although these definitions are consistent with the classic definition of vertex colouring for homogeneous graphs, there is no reason to not consider similar definitions with the opposite type-based restriction, which we call *negative* definitions. In the remaining of this section, we study the *negative* definitions of both vertex colouring definitions presented above. These *negative* vertex colouring definitions do not simplify to the classic definition of vertex colouring if we consider homogeneous graphs.

Definition 4.10 (Negative type-restrictive vertex colouring)**.** *Let* $\overline{\kappa}_R : V \to \mathbf{N}$ *be the negative type-restrictive vertex colouring of a typed graph* $G = (V, E, T)$. *For* $v_i : \tau_i, v_j : \tau_j \in V$, *if* $\tau_i = \tau_j$ *or* $\{v_i : \tau_i, v_j : \tau_j\} \in E$ *then* $\overline{\kappa}_R(v_i\!:\!\tau_i) \neq \overline{\kappa}_R(v_j\!:\!\tau_j)$.

Definition 4.11 (Negative type-permissive vertex colouring)**.** *Let* $\overline{\kappa}_P : V \to \mathbf{N}$ *be the negative type-permissive vertex colouring of a typed graph* $G =$

(V, E, T). *For* $v_i : \tau_i, v_j : \tau_j \in V$, *if* $\tau_i \neq \tau_j$ *and* $\{v_i : \tau_i, v_j : \tau_j\} \in E$ *then* $\overline{\kappa}_P(v_i{:}\tau_i) \neq \overline{\kappa}_P(v_j{:}\tau_j)$.

Theorem 4.12 shows that both negative vertex colouring definitions are equivalent when the typed graph has all vertices with an unique type.

Theorem 4.12. *Let* $G = (V, E, T)$ *be a typed graph. If* $|T| = |V|$, *i.e.,* $\forall v{:}\tau \in V, \nexists v'{:}\tau' \in V$ *such that* $v{:}\tau \neq v'{:}\tau' \wedge \tau = \tau'$, *then the set of negative type-restrictive vertex colouring equals the set of negative type-permissive vertex colouring.*

Proof. Consider the two predicates $P_E(v_i : \tau_i, v_j : \tau_j) : \{v_i : \tau_i, v_j : \tau_j\} \in E$ and $P_T(v_i : \tau_i, v_j : \tau_j) : (\tau_i = \tau_j)$. Let $\overline{\kappa} : V \to \mathbf{N}$. Define the following two predicates:

$$P_{\overline{R}}(\overline{\kappa}) : \forall v{:}\tau \in V, \nexists v'{:}\tau' \neq v{:}\tau \in V, \overline{\kappa}(v{:}\tau) = \overline{\kappa}(v'{:}\tau') \wedge (P_T(v{:}\tau, v'{:}\tau') \vee P_E(v{:}\tau, v'{:}\tau'))$$

$$P_{\overline{P}}(\overline{\kappa}) : \forall v{:}\tau \in V, \nexists v'{:}\tau' \neq v{:}\tau \in V, \overline{\kappa}(v{:}\tau) = \overline{\kappa}(v'{:}\tau') \wedge \overline{P_T(v{:}\tau, v'{:}\tau')} \wedge P_E(v{:}\tau, v'{:}\tau')$$

Notice that $\overline{\kappa} : V \to \mathbf{N}$ is a negative type-restrictive vertex colouring of G if and only if $P_{\overline{R}}(\overline{\kappa})$ is true and that $\overline{\kappa} : V \to \mathbf{N}$ is a negative type-permissive vertex colouring of G if and only if $P_{\overline{P}}(\overline{\kappa})$ is true. Since all vertices have the unique types, $\overline{P_T(v{:}\tau, v'{:}\tau')}$ is always true for all $v : \tau, v' : \tau' \in V$. Therefore both predicates simplify to

$$P_K(\overline{\kappa}) : \forall v{:}\tau \in V, \nexists v'{:}\tau' \neq v{:}\tau \in V, (\overline{\kappa}(v{:}\tau) = \overline{\kappa}(v'{:}\tau')) \wedge P_E(v{:}\tau, v'{:}\tau')$$

□

5. Special subsets of vertices

In this section we study special subsets of vertices in the light of typed graph theory. First we define, prove individual properties and also relations between vertex cover and independent sets. Second, we analyse dominating sets in typed graphs.

5.1. Vertex cover and independent set

Definition 5.1 (Type-restrictive vertex cover)**.** *Let* $G = (V, E, T)$ *be a typed graph.* $S \subseteq V$ *is a type-restrictive vertex cover if and only if* $\forall \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$, $(\tau_i = \tau_j \implies (v_i{:}\tau_i \in S \vee v_j{:}\tau_j \in S) \wedge (\tau_i \neq \tau_j \implies (v_i{:}\tau_i \in S \wedge v_j{:}\tau_j \in S)$.

A type-restrictive vertex cover must contain at least one vertex of each edge or both if they have different types.

Definition 5.2 (Type-permissive vertex cover)**.** *Let* $G = (V, E, T)$ *be a typed graph.* $S \subseteq V$ *is a type-permissive vertex cover if and only if* $\forall\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$, $\tau_i \neq \tau_j \vee (v_i{:}\tau_i \in S \vee v_j{:}\tau_j \in S)$.

A type-permissive vertex cover must contain at least one vertex of each edge with both endpoints of the same type.

Theorem 5.3. *Let* $G = (V, E, T)$ *be a homogeneous typed graph, i.e.* $|T| = 1$. *Therefore,* $S \subseteq V$ *is a type-restrictive vertex cover if and only if* S *is also a type-permissive vertex cover.*

Proof. Since all vertices have the same type, $\forall v_i{:}\tau_i, v_j{:}\tau_j \in V, \tau_i = \tau_j$, then the definitions of both type-restrictive vertex cover and type-permissive vertex cover can be simplified for the homogeneous typed graph G. In particular, $S \subseteq V$ is a type-restrictive vertex cover of G if and only if $\forall\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$, $\tau_i = \tau_j \implies (v_i{:}\tau_i \in S \vee v_j{:}\tau_j \in S)$, because $\tau_i \neq \tau_j \implies (v_i{:}\tau_i \in S \wedge v_j{:}\tau_j \in S)$ is always true. Similarly, $S \subseteq V$ is a type-permissive vertex cover of G if and only if $\forall\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$, $(v_i{:}\tau_i \in S \vee v_j{:}\tau_j \in S)$, because $\tau_i \neq \tau_j$ is always false. Therefore, both definitions are equivalent for the simplified scenario of homogeneous typed graphs. $\square$

Theorem 5.4. *Let* $G = (V, E, T)$ *be a connected typed graph in the reduced normal form. Therefore,* $S \subseteq V$ *is a type-restrictive vertex cover if and only if* $S = V$.

Proof. By definition, $\forall\{v_i : \tau_i, v_j : \tau_j\} \in E, \tau_i \neq \tau_j$ and there is no isolated vertex in G. Therefore, a type-restrictive vertex cover of G must contain both endpoints of every edge in G. Hence, V is the only type-restrictive vertex cover of G. $\square$

Theorem 5.5. *Let* $G = (V, E, T)$ *be a connected typed graph in the reduced normal form. Therefore,* $\emptyset$ *is a type-permissive vertex cover of* G.

Proof. Let us prove by contradiction. Suppose $\emptyset$ is not a type-permissive vertex cover of G. Then, $\exists\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$ such that $\tau_i = \tau_j \wedge v_i{:}\tau_i \notin \emptyset \wedge v_j{:}\tau_j \notin \emptyset$, which is a contradiction because,$\nexists\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E, \tau_i = \tau_j$. $\square$

Definition 5.6 (Type-restrictive independent set)**.** *Let $G = (V, E, T)$ be a typed graph. $S \subseteq V$ is a type-restrictive independent set if and only if $\forall v_i{:}\tau_i, v_j{:}\tau_j \in S$, $v_i{:}\tau_i \neq v_j{:}\tau_j \implies (v_j{:}\tau_j \notin N(v_i{:}\tau_i) \wedge \tau_i = \tau_j)$.*

A type-restrictive independent set contains only independent (i.e. non-adjacent) vertices of the same type.

Definition 5.7 (Type-permissive independent set)**.** *Let $G = (V, E, T)$ be a typed graph. $S \subseteq V$ is a type-permissive independent set if and only if $\forall v_i{:}\tau_i, v_j{:}\tau_j \in S$, $v_i{:}\tau_i \neq v_j{:}\tau_j \implies (v_j{:}\tau_j \notin N(v_i{:}\tau_i) \vee \tau_i \neq \tau_j)$.*

Theorem 5.8. *Let $G = (V, E, T)$ be a homogeneous typed graph, i.e. $|T| = 1$. Therefore, $S \subseteq V$ is a type-restrictive independent set if and only if S is also a type-permissive independent set.*

Proof. Since all vertices have the same type, the definitions of both type-restrictive independent set and type-permissive independent set can be simplified for the homogeneous typed graph G. In particular, $S \subseteq V$ is a type-restrictive independent set of G if and only if $\forall v_i{:}\tau_i, v_j{:}\tau_j \in S$, $v_i{:}\tau_i \neq v_j{:}\tau_j \implies v_j{:}\tau_j \in N(v_i{:}\tau_i)$. because $\tau_i = \tau_j$ is always true. Similarly, $S \subseteq V$ is a type-permissive independent set of G if and only if $\forall v_i{:}\tau_i, v_j{:}\tau_j \in S$, $v_i{:}\tau_i \neq v_j{:}\tau_j \implies v_j{:}\tau_j \in N(v_i{:}\tau_i)$. because $\tau_i \neq \tau_j$ is always false. Therefore, both definitions are equivalent for the simplified scenario of homogeneous typed graphs. □

Theorem 5.9. *Let $G = (V, E, T)$ be a typed graph in the reduced normal form. Let $\tau \in T$ and $V[\tau] = \{v_i{:}\tau_i \in V | \tau_i = \tau\}$. Therefore, $V[\tau]$ is a type-restrictive independent set of G.*

Proof. By definition, $\forall \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E, \tau_i \neq \tau_j$. Therefore, $\nexists v_i{:}\tau_i, v_j{:}\tau_j \in V$ such that $\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j$. Hence, $V[\tau]$ is a set of independent vertices of the same type, i.e. $V[\tau]$ is a type-restrictive independent set of G. □

Theorem 5.10. *Let $G = (V, E, T)$ be a typed graph in the reduced normal form. Therefore, V is a type-permissive independent set of G.*

Proof. By definition, $\forall \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E, \tau_i \neq \tau_j$. Suppose V is not a type-permissive independent set. Thus $\exists v_i{:}\tau_i, v_j{:}\tau_j \in V, \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j$, which is a contradiction. Hence, V is a type-permissive independent set of G. □

For a typed graph G, we define $\overline{G}$ as the complement graph with the types of vertices preserved. The following two theorems show relations between typed independent sets and the complement graph.

Theorem 5.11. *Let $G = (V, E, T)$ be a typed graph and $S \subset V$ be a type-restrictive independent set of G. Thus, $\overline{G}[S]$ is a complete homogeneous typed graph, i.e., every pair of vertices in $\overline{G}[S]$ are adjacent and of the same type.*

Proof. Let us prove by contradiction. Suppose $\overline{G}[S] = (S, E', T')$ is not a complete homogeneous typed graph. Then, $\exists v_i{:}\tau_i, v_j{:}\tau_j \in S, \{v_i{:}\tau_i, v_j{:}\tau_j\} \notin E' \vee \tau_i \neq \tau_j$.

First, $\{v_i : \tau_i, v_j : \tau_j\} \notin E'$ implies that $\{v_i : \tau_i, v_j : \tau_j\} \in E$, which is a contradiction since $v_i{:}\tau_i, v_j{:}\tau_j \in S$ and S contains no adjacent vertices of G.

Second, $\tau_i \neq \tau_j$ is also a contradiction, again, because $v_i{:}\tau_i, v_j{:}\tau_j \in S$ and S contains only vertices of the same type. □

Theorem 5.12. *Let $G = (V, E, T)$ be a typed graph and $S \subset V$ be a type-permissive independent set of G. All type-induced subgraphs of $\overline{G}[S]$ are complete homogeneous typed graph, i.e., $\forall \tau \in T$, $(\overline{G}[S])[\tau]$ is a complete homogeneous typed graph.*

Proof. Let us prove by contradiction. Suppose $(\overline{G}[S])[\tau] = (V', E', \{\tau\})$, for any $\tau \in T$, is not a complete homogeneous typed graph. Then, $\exists v_i{:}\tau, v_j{:}\tau \in V', \{v_i{:}\tau, v_j{:}\tau\} \notin E'$. However, $\{v_i{:}\tau, v_j{:}\tau\} \notin E'$ implies that $\{v_i{:}\tau, v_j{:}\tau\} \in E$, which is a contradiction since $v_i : \tau, v_j : \tau \in S$ and S contains no adjacent vertices of the same type in G. □

Theorem 5.13. *Let $G = (V, E, T)$ be a typed graph. If $S \subseteq V$ is a type-restrictive vertex cover then $V \setminus S$ is a type-permissive independent set.*

Proof. Let us prove by contradiction. Suppose $V \setminus S$ is not a type-permissive independent set. Thus, $\exists v_i{:}\tau_i, v_j{:}\tau_j \in V \setminus S$ such that $v_i{:}\tau_i \neq v_j{:}\tau_j \wedge \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j$, which is a contradiction, because we assumed that $v_i{:}\tau_i, v_j{:}\tau_j \in V \setminus S$, but we know that $\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j$ implies that $v_i{:}\tau_i \in S \vee v_j{:}\tau_j \in S$, since S is a type-restrictive vertex cover. □

Theorem 5.14. *Let $G = (V, E, T)$ be a typed graph. $S \subseteq V$ is a type-permissive independent set if and only if $V \setminus S$ is a type-permissive vertex cover.*

Proof. We prove in two steps.

($\Longrightarrow$) Let us prove by contradiction. Suppose $V \setminus S$ is not a type-permissive vertex cover. Thus, $\exists\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$ such that $\tau_i = \tau_j \wedge v_i{:}\tau_i \notin V \setminus S \wedge v_j{:}\tau_j \notin V \setminus S$, i.e., $\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j \wedge v_i{:}\tau_i \in S \wedge v_j{:}\tau_j \in S$, which is a contradiction since S is a type-permissive independent set.

($\Longleftarrow$) Let us prove by contradiction. Suppose $V \setminus S$ is not a type-permissive independent set. Thus, $\exists v_i{:}\tau_i, v_j{:}\tau_j \in V \setminus S$ such that $v_i{:}\tau_i \neq v_j{:}\tau_j \wedge \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j$, i.e., for $v_i{:}\tau_i \neq v_j{:}\tau_j$, $v_i{:}\tau_i, v_j{:}\tau_j \notin S \wedge \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j$, which is a contradiction since $\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j$ implies that $v_i{:}\tau_i \in S \vee v_j{:}\tau_j \in S$. $\square$

Theorem 5.15. *Let $G = (V, E, T)$ be a typed graph. If $S \subseteq V$ is a type-restrictive independent set then $V \setminus S$ is a type-permissive vertex cover.*

Proof. Let us prove by contradiction. Suppose $V \setminus S$ is not a type-permissive vertex cover. Thus, $\exists\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E$ such that $\tau_i = \tau_j \wedge v_i{:}\tau_i \notin V \setminus S \wedge v_j{:}\tau_j \notin V \setminus S$, i.e., $\{v_i{:}\tau_i, v_j{:}\tau_j\} \in E \wedge \tau_i = \tau_j \wedge v_i{:}\tau_i \in S \wedge v_j{:}\tau_j \in S$, which is a contradiction since S is a type-restrictive independent set, and therefore $v_i{:}\tau_i, v_j{:}\tau_j \in S \implies \{v_i{:}\tau_i, v_j{:}\tau_j\} \notin E \wedge \tau_i = \tau_j$. $\square$

5.2. Dominating set

Definition 5.16 (Dominating set)**.** *Let $G = (V, E, T)$ be a typed graph. $S \subseteq V$ is a dominating set if and only if $\forall v_i{:}\tau_i \in V, v_i{:}\tau_i \in S \vee (\exists v_j{:}\tau_j \in S, v_j{:}\tau_j \in N(v_i{:}\tau_i) \wedge \tau_i = \tau_j)$.*

If G is a homogeneous typed graph then Definition 5.16 of dominating set is equivalent to the classic untyped definition of dominating set.

Theorem 5.17. *If $G = (V, E, T)$ is a typed graph in the reduced normal form, then V is the only dominating set of G.*

Proof. Because G is in the reduced normal form, by definition, $\forall v_i{:}\tau_i \in V, \nexists v_j{:}\tau_j \in S, v_j{:}\tau_j \in N(v_i{:}\tau_i) \wedge \tau_i = \tau_j$. Therefore, every vertex $v_i{:}\tau_i \in V$ must be in the dominating set. $\square$

Theorem 5.18. *Let G be a typed graph. $\forall \tau \in T$, S_τ is a dominating set of $G[\tau]$ if and only if $\cup_{\tau \in T} S_\tau$ is a dominating set of G.*

Proof. ($\Longrightarrow$) Suppose $\forall \tau \in T$, S_τ is a dominating set of $G[\tau]$. Hence, $\forall v_i{:}\tau \in V$ it holds that either $v_i{:}\tau \in S_\tau$ or there is $v_j{:}\tau \in S$ such that $v_j{:}\tau \in N(v_i{:}\tau)$. Therefore, $\cup_{\tau \in T} S_\tau$ is a dominating set of G.

($\Longleftarrow$) Suppose S is a dominating set of G. Let $S_\tau = \{v_i{:}\tau_i \in S | \tau_i = \tau\}$. Thus, for all vertex $v_i{:}\tau \in V$ of type τ, it holds that either $v_i{:}\tau \in S$, and then $v_i{:}\tau \in S_\tau$, or there is $v_j{:}\tau \in S$, i.e. $v_j{:}\tau \in S_\tau$, such that $v_j{:}\tau \in N(v_i{:}\tau)$. Therefore, S_τ is a dominating set of $G[\tau]$. □

Corollary 5.19. *Let G be a typed graph and S be the minimum dominating set of G. Thus, $|S|$ is greater than or equal to the number of homogeneous connected components of G.*

Definition 5.20 (Negative dominating set)**.** *Let $G = (V, E, T)$ be a typed graph. $S \subseteq V$ is a dominating set if and only if $\forall v_i{:}\tau_i \in V, v_i{:}\tau_i \in S \vee (\exists v_j{:}\tau_j \in S, v_j{:}\tau_j \in N(v_i{:}\tau_i) \wedge \tau_i \neq \tau_j)$.*

If G is a typed graph in the reduced normal form then Definition 5.20 of negative dominating set is equivalent to the classic untyped definition of dominating set.

Theorem 5.21. *If $G = (V, E, T)$ is a homogeneous typed graph, then V is the only negative dominating set of G.*

Proof. Because G is a homogeneous typed graph, by definition, $\forall v_i : \tau_i \in V, \nexists v_j{:}\tau_j \in S, v_j{:}\tau_j \in N(v_i{:}\tau_i) \wedge \tau_i \neq \tau_j$. Therefore, every vertex $v_i{:}\tau_i \in V$ must be in the dominating set. □

6. Binary operations

Let $G_1 = (V_1, E_1, T_1)$ and $G_2 = (V_2, E_2, T_2)$ be typed graphs.

Definition 6.1 (Graph union)**.** $G_1 \cup G_2 = (V_1 \cup V_2, E_1 \cup E_2, T_1 \cup T_2)$

Definition 6.2 (Graph intersection)**.** $G_1 \cap G_2 = (V_1 \cap V_2, E_1 \cap E_2, T_1 \cap T_2)$

Definition 6.3 (Join)**.** *If $V_1 \cap V_2 = \emptyset$ then we define $G_1 \bowtie G_2 = (V_1 \cup V_2, E_1 \cup E_2 \cup E', T_1 \cup T_2)$ with E' connecting vertices of the same type, i.e., $E' = \{\{v_1{:}\tau_1, v_2{:}\tau_2\} | v_1{:}\tau_1 \in V_1 \wedge v_2{:}\tau_2 \in V_2 \wedge \tau_1 = \tau_2\}$*

Definition 6.4 (Negative join)**.** *If $V_1 \cap V_2 = \emptyset$ then we define $G_1 \overline{\bowtie} G_2 = (V_1 \cup V_2, E_1 \cup E_2 \cup E', T_1 \cup T_2)$ with E' connecting vertices of different types, i.e., $E' = \{\{v_1{:}\tau_1, v_2{:}\tau_2\} | v_1{:}\tau_1 \in V_1 \wedge v_2{:}\tau_2 \in V_2 \wedge \tau_1 \neq \tau_2\}$*

Proposition 6.5. *Let G_1,G_2 be typed graphs and $H = \Re_{G_1} \overline{\bowtie} \Re_{G_2}$. $H \simeq \Re_H$.*

Proof. Let $\Re_{G_1} = (V_1, E_1, T_1)$ and $\Re_{G_2} = (V_2, E_2, T_2)$. By definition we have $H = (V_1 \cup V_2, E_1 \cup E_2 \cup E', T_1 \cup T_2)$ with $E' = \{\{v_1{:}\tau_1, v_2{:}\tau_2\} | v_1{:}\tau_1 \in V_1 \wedge v_2{:}\tau_2 \in V_2 \wedge \tau_1 \neq \tau_2\}$. Since, by definition, $\forall E_i \in \{E_1, E_2, E'\}, \nexists \{v_i{:}\tau_i, v_j{:}\tau_j\} \in E_i$ where $\tau_i = \tau_j$, therefore H is in the reduced normal form. □

Proposition 6.6. *Let $G_1 = (V_1, E_1, T)$ and $G_2 = (V_2, E_2, T)$ be typed graphs with the same type-set and $H = G_1 \bowtie G_2$. For all $\tau \in T$, $\Re_H[\tau]$ is a singleton graph of type τ.*

Proof. By definition, $\forall v_1{:}\tau_1 \in V_1$ and $\forall v_2{:}\tau_2 \in V_2$, if $\tau_1 = \tau_2$ then $v_1{:}\tau_1$ and $v_2{:}\tau_2$ are adjacent in H. Therefore, $H[\tau]$ is a single homogeneous connected component in H. As proved in Theorem 2.11, the reduced normal form of a homogeneous connected component of type τ is a singleton graph of type τ. □

7. Properties of typed trees

In this section we study some properties of typed trees. Typed trees are special cases of typed graphs. A typed graph $G = (V, E, T)$ is a typed tree if (V, E) is a tree.

Proposition 7.1. *Let $G = (V, E, T)$ be a perfect typed n-ary tree, i.e. every non-leaf vertex contains exactly n children, of height h and $|T| = k$. Considering that the types are randomly distributed amongst the vertices with a uniform probability. Let $B(r)$ be the set of all binary r-tuples $(b_1, b_2, \dots, b_r)$, where $b_i \in \{0, 1\}$ for all b_i. Therefore, the expected height of $\Re_G$ is*

$$h \left(1 - \sum_{(x_0, x_1, \dots, x_{h-1}) \in B(h)} \prod_{i=0}^{h-1} x_i (\frac{1}{k})^{n^{i+1}} \right)$$

Proof. Given any vertex $v{:}\tau$, the probability that all its n children have the same type τ is $(\frac{1}{k})^n$.

A reduction happens in a level l of the typed tree if and only if all n^l vertices in level l have all their respective children with the same type. Thus, the reduction happens in a level l with probability

$$P\{R_l\} = \left(\left(\frac{1}{k} \right)^n \right)^{n^l} = \left(\frac{1}{k} \right)^{n^{l+1}}$$

Considering the function

$$p(x_0, x_1, \ldots, x_{h-1}) = \prod_{i=0}^{h-1} x_i P\{R_i\}$$

where $x_i \in \{0, 1\}$ for all x_i.

Therefore, the expected height of $\Re_G$ is

$$h\left(1 - \sum_{\mathbf{x} \in B(h)} p(\mathbf{x})\right)$$

where $p(\mathbf{x}) = p(x_0, x_1, \ldots, x_{h-1})$ for $\mathbf{x} = (x_0, x_1, \ldots, x_{h-1}) \in B(h)$. $\square$

Proposition 7.2. *Let $G = (V, E, T)$ be a typed tree in the reduced normal form. Using de definition of $\mathcal{G}^*(v{:}\tau)$, $v{:}\tau \in V$, presented in Proposition 3.4, consider $K \subseteq [G]_\equiv$ such that*

$$K \subseteq \{\mathcal{S}_{H_1}^{v_1:\tau_1} | \mathcal{S}_{H_2}^{v_2:\tau_2} | \ldots | \mathcal{S}_{H_n}^{v_n:\tau_n} | G \quad | \quad H_i \in \mathcal{G}^*(v_i{:}\tau_i), v_i{:}\tau_i \in V\}$$

where K contains only typed trees. Therefore

$$\prod_{v:\tau \in V} \binom{d(v{:}\tau)}{2} \leq |K| \leq \prod_{v:\tau \in V} \binom{d(v{:}\tau)}{2} d(v : \tau)^{d(v:\tau)-2}$$

Proof. Let $d' = d(v{:}\tau)$. Again, notice that $\{S | P_P^*(S, v{:}\tau)\}$ is the set of all valid partitions of the set $N(v{:}\tau)$ into exactly d' partitions, where some of the partitions may receive no element of $N(v{:}\tau)$. We can partition $N(v{:}\tau)$ in a total of $\binom{d'}{2}$ different ways. Thus $|\{S | P_P^*(S, v{:}\tau)\}| = \binom{d'}{2}$.

For any valid vertex-set $V' = \{S'{:}\tau | S' \in S\}$, with $P_P^*(S, v{:}\tau)$ true, consider the subset $E'' \subseteq \{E' | P_C(V', E', \tau)\}$ that represents the set of all edge-sets such that the corresponding graph is a tree. Since $|V'| \leq d'$, by using Cayley's formula [3], the number of trees on d' vertices is $d'^{d'-2}$, i.e., $|E''| \leq d'^{d'-2}$.

If we repeat this process for all vertices in V, considering only graphs that represent trees, we conclude that

$$\prod_{v:\tau \in V} \binom{d(v{:}\tau)}{2} \leq |K| \leq \prod_{v:\tau \in V} \binom{d(v{:}\tau)}{2} d(v : \tau)^{d(v:\tau)-2}$$

$\square$

References

[1] B. Russell, Mathematical logic as based on the theory of types, American Journal of Mathematics 30 (3) (1908) 222–262.

[2] A. Church, A formulation of the simple theory of types, Journal of Symbolic Logic 5 (2) (1940) 56–68.

[3] A. Cayley, A theorem on trees, Quart. J. Pure Appl. Math. 13 (23) (1889) 376–378, Collected Mathematical Papers Vol. 13, Cambridge University Press 1897, 26-28.