Reasoning on Company Takeovers during the COVID-19 Crisis with Knowledge Graphs*

Luigi Bellomarini[1], Marco Benedetti[1], Stefano Ceri[2], Andrea Gentili[1], Rosario Laurendi[1], Davide Magnanimi[1,2], Markus Nissl[3], and Emanuel Sallinger[3,4]

[1] Banca d'Italia
[2] Politecnico di Milano
[3] TU Wien
[4] University of Oxford

Abstract. When some country takes a disproportionate hit by a large-scale turmoil—just like Italy did during the COVID-19 pandemics—the share prices of its companies plunge. Suddenly, it becomes feasible to attempt foreign takeovers of national assets, including those of strategic interest. To avert this risk, the Government can veto transactions by summoning the so-called "Golden Powers". Or, it can work to proactively identify structural weaknesses in the control or shareholding chains of key companies, in order to reinforce them without resorting to special powers. Sometimes, vulnerabilities and attacks hide in plain sight due to how complex and intertwined the network of mutual company shareholding is. In this work, we show how to leverage *Knowledge Graphs* (KGs) as a representation and reasoning framework to analyze both reactive and proactive measures against takeover attempts, however intricate the setting where they take place. We formally characterize a set of reasoning tasks that define when and if to employ Golden Powers, plus others that aim at pinpointing companies prone to attacks. These criteria are exercised on the real network of all Italian companies, built for the occasion. A rich set of experiments is provided, including on several large synthetic instances, to prove the robustness of our method.

Keywords: Knowledge Graphs · Reasoning · Company Takeovers.

1 Introduction

The COVID-19 outbreak has had an immense impact on our society. Besides the critical health crisis, it has become clear that preventing, or at least managing, its large-scale economic effects will become critical as well. The work we present here deals with reasoning about company takeovers. To this end, we employ *Knowledge Graphs* (KGs) as a representation and reasoning framework: Our approach analyzes automatically a large graph of knowledge (representing the

*The views and opinions expressed in this paper are those of the authors and do not necessarily reflect the official policy or position of Banca d'Italia.

entire set of Italian companies and their mutual financial relationships) in search for ongoing or potential hostile takeovers over companies of strategic national interest. Specific reactive or proactive defence measures by the Government, possibly involving the use of the so-called "*Golden Powers*" (GPs), are automatically produced as a result of our analysis. This initiative forms one pillar of a flagship project launched by the Central Bank of Italy in reaction to the COVID-19 crisis,[5] and is in line with a call to action by the President of the European Commission[6] on counteracting hostile takeovers of strategic companies.

It is essential that the ownership of — or, more importantly, the control upon — companies deemed of *strategic relevance* (e.g., in the energy, military, transport, telecommunications sectors) remains in the hands of trusted shareholders. Yet, with the COVID-19 outbreak, companies stretched by massive shutdowns and production plunge are subject to an abnormal number of *hostile takeover attempts* because, in conditions of market turbulence, attackers try to take advantage of lowered share prices. A hostile takeover consists in gaining the control of a target company against the will of its management. *Company control* can be gained directly by acquiring the majority of the target company shares, or indirectly, by gaining control over a set of companies that jointly own the majority of the shares of the target. In real-world company networks, such indirect undertakings with very long and intertwined control chains are regularly present.

Multiple countries have historically resorted to legal frameworks to protect strategic companies against foreign takeovers [16]. Italy is a relevant example: Being one of the countries most struck by the COVID emergency, it carried out a careful application of the so-called *Golden Powers* [19], that is, the possibility for the central Government to veto individual acquisition transactions (e.g., in terms of shares of stocks) that would cause strategic assets to fall victim to takeovers. Likewise, the Government can intervene to secure companies by acquiring or increasing its participation in the strategic firms (technically, *investment beef-up*) via publicly controlled intermediaries.

Golden Power Settings. Unfortunately, an effective application of the mentioned legal frameworks (and of GPs in particular) is by no means trivial. How can we tell whether a transaction is a takeover attempt? Will a transaction lead to a takeover? What is the minimum amount of share that must shift to public control in order to protect a company? And, how to protect against coordinated, *collusive*, transactions aiming at a takeover? These problems lend themselves to be addressed by a *declarative and fully explainable* approach, and encoded as reasoning tasks on the KG of the Italian companies, built and maintained by the Bank of Italy [2]. The technical challenges are significant: Dealing with indirect control chains requires a Knowledge Representation and Reasoning (KRR) language that supports *recursion and creation of new values*. Indeed, the problems underlying the application of GPs are hardly addressed by traditional data management technology, where support for recursion is absent or laborious. Moreover, the massive amount of domain knowledge available makes resorting to a pure graph database impractical, as it leads to either proliferation of over-

[5] `https://kg19.bankit.art`

[6] `https://trade.ec.europa.eu/doclib/press/index.cfm?id=2124`

complicated non-scalable queries or to a substantial impossibility of representing the necessary queries within poorly expressive host languages. Finally, machine learning or network analysis approaches would lack explainability.

KG19. This work leverages our experience in state-of-the-art reasoning in Vadalog KGs [6,7] to study the impact of the outbreak on the Italian company network under various perspectives and provide policymakers, analysts, and economists, with actionable AI tools and data to support businesses and lessen the economic impact of COVID-19. Although we focus on the Italian case, our initiative aims at providing methodologies and tools that are valid in general, independently of the specific country and crisis situation. A wider picture of the lockdown impact on the company network can be found in a recent report [5].

Contribution and Overview. We present the first results of the application of rule-based reasoning on KGs to aid decision making about the application of Golden Powers to contrast hostile takeovers. In particular, this paper contributes:

- The main references to Vadalog (Section 2) and a compact formalization of the *company control problem* (Section 3) as background material.
- A formal characterization and discussion of a set of *reasoning tasks about Golden Powers* (Section 4), modeling the possibility of different Governments to intervene on transactions that may underlie takeover attempts.
- A discussion of the application of our techniques to the real data from the KG of the Italian companies, with an evaluation of the soundness of the approach for real takeover patterns. We also study the scalability of the approach on real data as well as on synthetic instances of the relevant GP reasoning tasks and of the company control problem in the Vadalog System (Section 5).

Related work is discussed in Section 6, while Section 7 concludes the paper.

2 Vadalog Knowledge Graphs

The formalization of the company control problem and the GP settings in this paper are encoded in Vadalog, a language from the Datalog$^\pm$ family [9,13]; experimental evaluations are run in the Vadalog System. Datalog$^\pm$ generalizes Datalog with existential quantification in the rule conclusion. A *rule* is a first-order sentence of the form $\forall\bar{x}\forall\bar{y}(\varphi(\bar{x},\bar{y}) \rightarrow \exists\bar{z}\,\psi(\bar{x},\bar{z}))$, where φ (the *body*) and ψ (the *head*) are conjunctions of atoms. For brevity, we omit universal quantifiers and denote conjunction by comma. As usual in this context, the semantics of a set of rules is defined by the well-known chase procedure. The core of Vadalog is based on *Warded Datalog*$^\pm$ [6], a syntactic restriction to Datalog$^\pm$ that guarantees decidability and tractability in the presence of recursion and existential quantification. In terms of expressive power, Warded Datalog$^\pm$ captures full Datalog and `OWL 2` direct semantics entailment regime for `OWL 2 QL`. The language underpinnings are exploited by the reasoner to allow for efficient execution of reasoning tasks [7]. Vadalog augments Warded Datalog$^\pm$ with supplementary features such as aggregation, algebraic operations, and stratified negation.

Vadalog supports *monotonic aggregations*, whose full details can be found in [7]. However, a simpler form of aggregation, which suffices to our ends, is

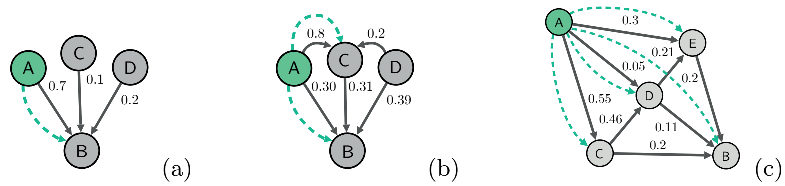


Fig. 1: Sample ownership graphs where A controls B. Nodes are entities; solid edges are direct ownerships; dashed edges are control relationships.

based on *stratified semantics*, where the basic idea for our case is very simple: An aggregation function (e.g., *sum*, in Section 4) is computed only when its input operands are completely known. All our use cases admit such simplification.

3 Company Control Problem: A Deductive Approach

Underlying the study of hostile takeovers is the notion of company control. It concerns decision power, i.e., when a company can direct the decisions of another company by controlling the vote majority via the majority of the shares. Let us consider an ownership graph, i.e., the directed graph where nodes are shareholders and edges represent share ownership.

Along the lines of existing formulations from the logic and database literature [10], we see the company control problem as follows.
A company x controls *a company* y, *if: (i)* x *directly owns more than* 50% *of* y*; or, (ii)* x *controls a set of companies that jointly (i.e., summing their shares), and possibly together with* x*, own more than* 50% *of* y.

Figure 1 shows basic cases of company control. In Figure 1(a), the first clause applies: Company A directly controls company B. In Figure 1(b), the second clause applies: Company A does not directly control B; nevertheless, it has an 80% share on company C and hence it controls C. Thus, the total share of company B that A controls rises to 61%, which is the sum of the direct 30% ownership of A on B and the 31% ownership of the controlled company C on B. In the end, A controls B. Figure 1(c) demonstrates a more convoluted form of control: An entity A can control a company B anywhere in the graph, given that A can indirectly control the majority of the shares of B even if no direct control exists between A and B or even between any intermediate company and B.

Company control can be formulated as a Vadalog reasoning task:

$$Company(x) \rightarrow Control(x, x) \tag{1}$$

$$Control(x, y), Own(y, z, w), v = sum(w), v > 0.5 \rightarrow Control(x, z) \tag{2}$$

Assuming that every company has control on itself (Rule 1),[7] we inductively define control of x on z by summing the shares of z owned by any company y over all companies y controlled by x (Rule 2). The presence of cycles in the ownership graphs, a common case indeed, is irrelevant for control purposes.

[7] This formalization of the base case is slightly different from the natural definition but commonly assumed in the literature as it is more compact and formally equivalent.

4 Reasoning on Golden Power

In this section we elaborate on a set of KG-based applications revolving around the use of GPs. The framework we develop covers five fundamental concerns raised by business stakeholders, related to: Decision and policy making, advice to be given to companies, and proactive actions to be taken. We present a number of *core reasoning tasks* providing insights on: 1. *detecting* cases of transactions hiding possible takeover attempts; 2. *suggesting* limits within which GPs may be exercised; 3. giving options for *proactively protecting* companies from takeover attempts. Figure 2 summarizes the scenarios under consideration. Each column describes one scenario, specifying its goal, general setting, business question, and resulting insight. At the bottom of each column, we report one example.

At the core of our cases, there is the Company Control setting from Section 3. Companies are assigned different roles: *trusted* (e.g., public companies or Governmental bodies, pink in the figure), *attacker* (e.g., a company out of the national border in the figure, i.e., incorporated or organized under the law of another country), or *target* (e.g., the strategic company to be protected, green); all the others are assumed to be *neutral* (gray).

We perform both *reactive* analysis, checking whether specific variations to the graph generated by *candidate transactions* (acquisition of shares) culminate in unwanted takeovers, and *proactive* analysis, detecting structural vulnerabilities and possible countermeasures, independently of any possibly ongoing attack.

In the following paragraphs, we introduce the VADALOG formulations of the five cases. Each one is an extension of Company Control, so Rules 1 and 2 from Section 3 are assumed to be inherited by all the formulations. Attackers, target, and trusted companies are respectively denoted by atoms V, T, and P.

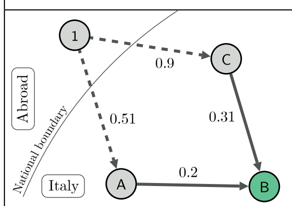
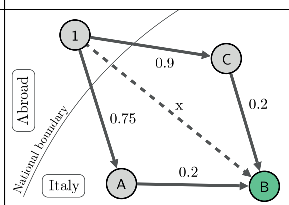
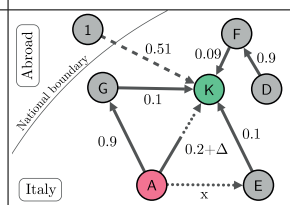
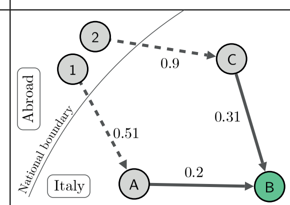
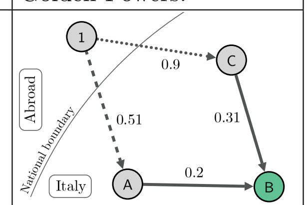
Encoding these criteria as rules in an expressive and scalable declarative framework such as VADALOG is of the essence here because while we have analyzed the key reasoning patterns in this work, more emerge on a daily basis during the interactions with business experts: Changes that would radically impact a procedural approach (requiring a substantial rewrite) are just a minor amendment away when the domain knowledge is captured declaratively. Not only does VADALOG allow us to quickly test and deploy new criteria, but the amount of time we spend to get sure we are on the same page as our business colleagues — i.e., to convince them that the implementation is actually computing what we have agreed on paper — dramatically shortens.

Golden Power Check. We show how to detect individual transactions that cause some target company to be taken over. This is a case where Golden Power may be an option to exercise. We call this problem: *Golden Power Check.*

Example. Let us consider the example shown at the bottom of Figure 2(1). We first consider the setting. Company 1 is in the set of attackers, e.g., potentially attacking companies under investigation (forming the set V in the definition shown in Figure 2), while the colored node B is in the set of target companies (forming the set T in our definition shown in Figure 2).

Candidate transactions are shown using dashed edges. Our first candidate transaction is t_1, where an ownership of 51% of A is acquired by 1. The second candidate transaction is t_2 where an ownership of 90% of C is acquired by B. Let

Fig. 2: Golden Power Settings

(1) Golden Power **Check**	(2) Golden Power **Limit**	(3) Golden Power **Protection**	(4) **Collusion** Golden Power **Check**	(5) **Cautious** Golden Power **Check**
Analysis: Reactive	**Analysis**: Proactive	**Analysis**: Proactive	**Analysis**: Proactive	**Analysis**: Reactive
Goal: Checking whether an acquisition (of shares, stocks, etc.) causes any target company to become controlled by an attacking company. **Setting**: Let T be a set of target companies, V a set of attacking companies, and N a set of neutral companies. Let t be a transaction (e.g., an offer issued by a company x to buy an amount s of shares of a company y), with $x \in V$, $y \in T \cup N$. **Question**: Does t cause any company in V to gain control of any company in T? **Insight**: If the answer is YES, consider the possibility to block t via Golden Powers.	**Goal**: Computing the maximum amount of share a company x can buy of a company y without controlling any company in a set T. **Setting**: Let T be a set of target companies and V be a set of attacking companies. Let t be a transaction (e.g., an offer issued by a company x to buy an amount s of shares of a company y), with $x \in V$, $y \in T$. **Question**: What is the maximum value s_{max} for s such that there are no companies in V that gain control over any company in T? **Insight**: Transactions that acquire (much) less than s_{max} shares of y do not require the application of Golden Powers immediately to protect y.	**Goal**: Computing the share increment needed by trusted companies to prevent hostile takeovers. **Setting**: Let T be a set of target companies and V be a set of attacking companies. Let P be a set of trusted companies (such that P is disjoint from V and T). **Question**: Which acquisitions of shares of companies in T by companies in P guarantee that no set of transactions t (from x to y, with $x \in V$, $y \in T$) allows any company in V to gain control over one in T? **Insight**: Consider the possibility to (temporarily) buy shares of T via P as per the answer to the above question to prevent takeovers (Golden Powers are not needed).	**Goal**: Checking whether an acquisition (of shares, stocks, etc.) causes any target company to be possibly controlled by a set of attacking companies acting in collusion. **Setting**: Let T be a set of target companies, V be a set of attacking companies, and N a set of neutral companies. Let t be a transaction (e.g., an offer issued by a company x to buy an amount s of shares of a company y), with $x \in V$, $y \in T \cup N$. **Question**: Does t cause V to gain joint control of any company in T? **Insight**: If the answer is YES, consider the possibility to block t via Golden Powers.	**Goal**: Checking whether an acquisition (of shares, stocks, etc.) causes any target company to become possibly controlled by an attacking company for which shareholding information is incomplete. **Setting**: Let T be a set of target companies, V be a set of attacking companies, and N a set of neutral companies. Let t be a transaction (e.g., an offer issued by a company x to buy an amount s of shares of a company y), with $x \in V$, $y \in T \cup N$. **Question**: Assuming that any unassigned share of y is in fact owned by some $v \in V$, does t allow v to gain control of y? **Insight**: If the answer is YES, consider the possibility to block t via Golden Powers.
				

us first consider transaction t_1 as our transaction of interest. This would give 1 control of A, and hence a 20% ownership of B. So far, the total ownership of target company B by company 1 is thus 20% with no need to block t_1.

Now assume that transaction t_1 was processed (i.e., it becomes a solid line), and consider transaction t_2, where an ownership of 90% of C is obtained by 1. This would give 1 control of C and hence 31% ownership of B. Together with the ownership of 20% of target company B that 1 already holds, it now has 51% ownership of company B and thus controls it. Transaction t_2 must be blocked using Golden Power if the target company B should not come under the control of 1. Finally, we remark that had transaction t_2 come before t_1, it would have been fine to process t_2 and block t_1. This concludes our example.

This line of reasoning can be formalized as a Vadalog reasoning task by extending Company Control with the following rules:

$$V(x), \neg V(y), Tx(x, y, w) \rightarrow Own(x, y, w) \qquad (1)$$

$$V(x), T(y), Control(x, y) \rightarrow GPCheck(x, y) \qquad (2)$$

Rule 1 defines that, for the purpose of our analysis, we consider transaction Tx to be virtually applied, i.e., leading to actual ownership even if it has not taken place. Then, Rule 2 captures our goal by computing all companies in V that control at least one company in T. If *GPCheck* is empty, there is no reason to use Golden Powers. In case it is non-empty, it gives a list of the companies which are possible subject of takeovers caused by the single acquisition of share Tx.

Golden Power Limit. The second relevant problem is to *advise* companies about what transactions are allowed to take place (without requiring the use of GPs to prevent a takeover). We call this problem *Golden Power Limit*. The definition and an example are given in Figure 2(2). A full explanation of this (and following) examples can be found in the appendix,[8] while the definition can be formulated with the following Vadalog rules:

$$Control(x, y), Own(y, z, w), v = sum(w) \rightarrow PControl(x, z, v) \qquad (1)$$

$$V(x), T(y), PControl(x, y, v), v < 0.5 \rightarrow GPLimit(x, y, 0.5 - v) \qquad (2)$$

$$Tx(x, y, w), GPLimit(x, y, h), w < h \rightarrow Allowed(x, y, w) \qquad (3)$$

Rule 1 captures the case of partial control, i.e., the sum of shares does not necessarily lead to control. Rule 2 evaluates the threshold $w = 0.5 - v$, representing the limit of shares of y that company x can buy without obtaining control. In Rule 3 the transaction Tx is validated against the limit computed in Rule 2.

Golden Power Protection. The use of Golden Power comes with political and economic consequences, so in many cases it may be desirable to proactively prevent takeovers. The third problem is about *preventing* the use of Golden Powers from becoming a necessity. The definition and an example are given in Figure 2(3). A Vadalog formulation is the following:

$$Control(x, y), Own(y, z, w), v = sum(w) \rightarrow PControl(x, z, v) \qquad (1)$$

$$P(x), T(y), PControl(x, y, v), v < 0.5 \rightarrow Prot(x, y, 0.5 - v) \qquad (2)$$

[8] `https://bit.ly/2WOYzZZ`

We use the same approach as in *Golden Power Limit*. Value v, computed by Rule 2, quantifies the additional direct share ("beef-up") that a trusted company x needs to secure in order to gain control over a target company y.

Collusion Golden Power Check. We now formulate the *Golden Power Check* problem under the possibility of collusive attackers, which we call *Collusion Golden Power Check*. In general, there is a range of possibilities in defining what types of "collusion" we address, with choices that need to be validated at the business level. The definition and an example are given in Figure 2(4).

The definition of Figure 2 can be formulated as the following VADALOG reasoning task, which extends the definition we have seen before:

$$V(x),\ V(y) \rightarrow \mathit{Control}(x, y) \tag{1}$$

$$V(x), \neg V(y),\ \mathit{Tx}(x, y, w) \rightarrow \mathit{Own}(x, y, w) \tag{2}$$

$$V(x),\ T(y),\ \mathit{Control}(x, y) \rightarrow \mathit{CGPCheck}(x, y) \tag{3}$$

We have already discussed Rules 2 and 3 when considering *Golden Power Check*. Rule 1 captures the intuition that companies in V may be acting collusively, i.e., surreptitiously exerting control over one another.

Cautious Golden Power Check. We now consider the case of missing or unknown information. So far, in our examples we have assumed that if only a partial percentage of the ownership of a company is represented, the remaining shares are not relevant. However, in a *cautious* scenario, we may want to presume that the most unfavorable conditions hold, namely that the non-represented shares are already in the hands of the attackers. This pessimistic assumption can be combined with any of our base scenarios; here we introduce *Cautious Golden Power Check*, in Figure 2(5). The following VADALOG rules formalize our concept of caution:

$$\neg V(x), \neg V(y), \mathit{Own}(x, y, v), w = \mathit{sum}(v) \rightarrow \mathit{Assigned}(y, w) \tag{1}$$

$$\mathit{Assigned}(y, w), w < 1 \rightarrow \exists z\ \mathit{Company}(z), V(z), \mathit{Own}(z, y, 1 - w) \tag{2}$$

$$V(x), \neg V(y),\ \mathit{Tx}(x, y, w), v = \mathit{sum}(w) \rightarrow \mathit{Own}(x, y, v) \tag{3}$$

$$V(x),\ T(y),\ \mathit{Control}(x, y) \rightarrow \mathit{GPCCheck}(x, y) \tag{4}$$

We compute the total amount of currently (un)assigned shares, for each company, with Rule 1. Then, by Rule 2 we depict the worst case scenario where if a non-attacker company (neutral or target) has unassigned shares, there exists an attacker retaining all of them. We assume this is true when reasoning on transactions (Rule 3) and while performing the control check (Rule 4).

5 Experimental Evaluation

We applied our technique to the ownership graph of the Italian companies [5] to detect potential hostile takeovers. First, we evaluate the soundness of the approach on the real KG of Italian companies, then we show its scalability in both real and *synthetic* scenarios. We employed a cloud-hosted memory-optimized virtual machine with 16 cores and 128 GB RAM on an Intel Xeon architecture.

Evaluating Approach Soundness. We validated our VADALOG rules for *Company Control* by comparing our results for the Italian KG with a sample of 382k control instances from Orbis, a commercial source. This check resulted in a very good 99.7% overlap; the remaining 0.3% was manually traced to mismatches in the data. We also applied our takeover VADALOG rules (Section 4) to the Italian KG. We grouped companies by sector of economic activity (as identified by the "`ATECO`" code, derived from the "NACEv2" European classification) selecting the 5 largest clusters. For each cluster, we ran our proactive criteria and identified the potential takeovers that are avoidable with Golden Powers as well as the needed countermeasures. We observed a maximum of 365 avoidable takeovers for a very relevant economic sector. Results and execution times are in Figures 4(a-b).

Evaluating Scalability. Given the large scale of the KG and the sophisticated reasoning tasks at hand, we need to verify that our technique scales nicely on larger and more "complex" cases. A first reassuring (purely theoretical) consideration is that all our rules can be syntactically formulated in terms of the VADALOG core, *Warded Datalog*$^{\pm}$, a logic fragment where reasoning is `PTIME` in data complexity [6]. This bodes well for the asymptotic behavior, but says nothing on the absolute running times of the real cases. To assess the situation, we ran the reasoning tasks for the five GP cases of Section 4 in a total of 4 test scenarios, working on the *real-world* Italian KG and on 10 *synthetic* KGs, in Figure 4(e-h). We also ran specific tasks for Company Control in 2 scenarios over the *real-world* (not in the figure) and 9 *synthetic* KGs, in Figure 4(c-d). Each run (311 in total) has been repeated 5 times averaging the execution times.

Datasets (KGs). For the real-world scenarios, we considered the Italian KG. For the synthetic scenarios, we generated 12 *synthetic scale-free* graphs [3,15] (whose details are in Figure 3) with an adaptation of the Barabási–Albert model [3] for directed scale-free networks. The linking probability has been tuned to adjust graph density and the shares have been generated by sampling from a Beta distribution with parameters fitted from the real-world graph. Each company has been assigned a role among *trusted*, *attacker*, *target*, *neutral*, sampling from a multinomial distribution fitting the real-world case. Synthetic KGs are named in the `N{n}D{d}` format, where n denotes the number of nodes (in millions) and d is the average non-zero out-degree. To cover specific scenarios, we generated an additional series of `N7D1U{u}` graphs, where u denotes the number of companies (in $100k$ multiples) having unallocated shares (for $u = 1$, the suffix `U1` is omitted).

Discussion. Absolute times of Company Control are very satisfactory, e.g., with 292 secs for `Real`/`N7D1` and 1195 secs in `N21D1`. A polynomial behaviour is confirmed by the trend in Figure 4(c). The trend diverges for `N28D1` as the reasoner hits memory limits. Increasing density (Figure 4(d)) highlights the transition from the average to the worst case for a baseline setting with fixed number of nodes and shows its dependence on the out-degree. For higher densities the system accumulates more and longer control chains, with higher elapsed times.

In comparable graphs (e.g., `N7D1` vs `Real`), all the GP scenarios are faster than pure Company Control (Figure 4(e)). Proactive cases curb input cardinal-

Graph	Nodes	Edges	SCCs	WCCs	Average non-zero in degree	Average non-zero out degree	Average nodes in WCC	Max in degree	Max out degree	Largest WCC (nodes)	Nodes w/ unallocated shares
`Real`	7.0M	6.3M	7.0M	1.4M	2.43	1.37	5.1	3.2k	1.5k	1.7M	191k
`N7D1`	7.0M	5.3M	7.0M	1.72M	1.59	1.02	4.07	1.0k	4	1.6M	114k
`N9D1`	9.0M	7.0M	9.0M	2.0M	1.55	1.06	4.53	526	5	2.1M	152k
`N11D1`	11.0M	8.6M	11.0M	2.4M	1.57	1.06	4.63	544	5	2.7M	189k
`N14D1`	14.0M	11.0M	14.0M	3.0M	1.58	1.06	4.66	531	5	3.6M	240k
`N21D1`	21.0M	16.5M	21.0M	4.5M	1.58	1.06	4.66	540	5	5.3M	361k
`N28D1`	28.0M	21.7M	28.0M	6.3M	1.53	1.06	4.41	533	6	5.7M	466k
`N7D2`	7.0M	8.4M	7.0M	1.1M	1.72	2.02	6.40	1040	6	1.6M	177k
`N7D3`	7.0M	10.6M	7.0M	0.8M	1.95	2.94	8.79	1057	7	1.9M	216k
`N7D4`	7.0M	14.3M	7.0M	0.5M	2.48	3.9	14.62	1055	8	2.4M	433k
`N7D1U2`	7.0M	5.5M	7.0M	1.5M	1.58	1.06	4.66	532	5	1.8M	207k
`N7D1U3`	7.0M	5.5M	7.0M	1.5M	1.58	1.06	4.66	531	5	1.8M	294k
`N7D1U4`	7.0M	5.5M	7.0M	1.5M	1.58	1.06	4.66	532	5	1.8M	381k

Fig. 3: Characteristics of the Knowledge Graphs used in our experiments.

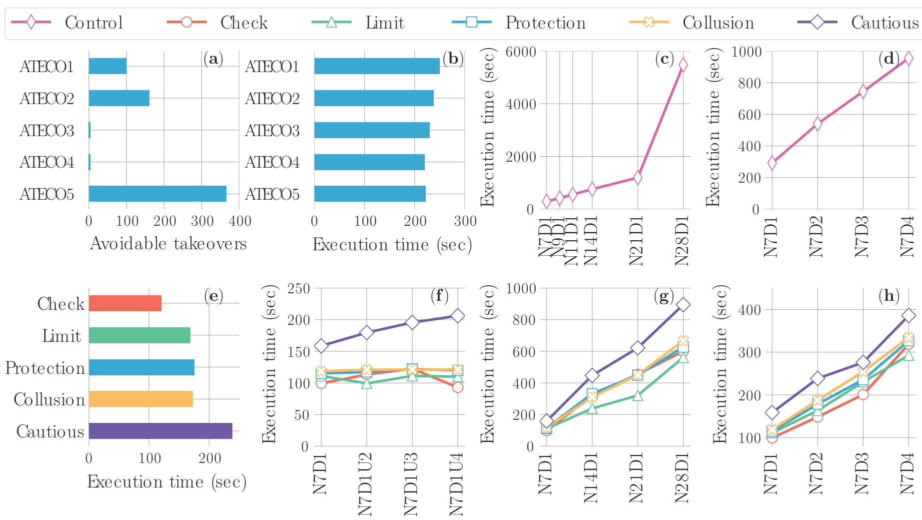


Fig. 4: Reasoning times for *Company Control* (c-d) and *Golden Power* (a-b, e-h) with *real* (a-b,e) and *synthetic* (c-d, f-h) data.

ity by filtering on the considered companies (V and T predicates); reactive cases produce smaller output than Company Control: Although Limit and Protection consider all partial controls, they concentrate on control of attackers resp. trusted companies over targets. While Collusion artificially adds new Control facts, it does not add edges and therefore simplifies into an attackers-to-target constrained version of Company Control. Cautious is the more demanding case: Reasoning generates at most $|T \cup N|$ companies and edges, depending on the number of nodes having unassigned shares. In the worst case, this doubles the graph size, so it does not affect problem complexity. In fact, Figure 4(f) shows a polynomial growth of the elapsed time of Cautious caused by the number of companies having unassigned shares, which affects the other cases only marginally.

Our cases exhibit good scalability with number of nodes (Figure 4(g)), with a polynomial trend consistently less steep than pure Company Control. The trend of Cautious is slightly steeper than the other cases, as a result of the addition of new attackers (due to existential Rule 2 for the Cautious case in Section 4), which amplify the growth in node count. Interestingly, our cases show good scalability w.r.t. graph density (Figure 4(h)), and the time difference between the cases keeps stable when nearing to the worst case. This is coherent as growth in density does not over-activate the mentioned existential rule.

6 Related Work

The reasoning tasks we have presented exemplify how graph database technology has insufficient expressive power for our goals (e.g., for the joint need for recursion and value creation). Indeed, *Regular Path Query languages* [8], at the basis of common property graphs query approaches [1], lack support for full recursion and ontological reasoning, leading to a proliferation of diverse cases to handle, an unbearable effort in production applications. On the other hand, Datalog with a mild form of negation and tractable existential quantification is a good yardstick for the expressive power needed in takeover analysis.

In graph processing, *partial evaluation techniques* [11] are a related procedural approach that has shown good results for reachability queries; also, specific algorithmic solutions to compute company ownership have been proposed [20]. Yet, in our experience none of these approaches conveys at the same time explainability, scalability, and adaptability as a Vadalog declarative solution does.

Under the economic angle, many approaches study hostile takeovers at macro level [18], or concentrate on reactive techniques [21]. Relevant works try to predict takeovers by matching company characteristics [17]. The corporate economics community has been studying the related subproblem of company control at various levels [4,12,14], privileging matrix-based formulation with computational limitations and non-explainable answers.

The AI and database communities have shown theoretical interest in problems related to company graphs [10], but no approaches have applied reasoning or KGs to the hostile takeover phenomenon. We have recently adopted the company control problem as a reference use case for novel KRR formalisms and addressed it in the context of enterprise company KGs [2,6].

7 Conclusion and Future Work

Driven by the ongoing effort to prevent strategic companies from becoming the target of hostile takeovers, in this work we recognize the power of reasoning on KGs and contribute a set of foundational use cases and formalisations to detect takeover attempts; we also suggest countermeasures and avoidance strategies in a fully explainable fashion. To the best of our knowledge, this is the first use of reasoning techniques in the takeover setting.

We tested all proposed reasoning tasks in the Vadalog System with both real-world and synthetic data. We focused on clarity, by showing only the key rules governing each use case, while leaving out for brevity the large number of auxiliary rules necessary for data extraction, cleaning, integration, etc.

The formalisation we have presented in this work is at the basis of our future work, which concerns the construction of a set of more sophisticated reasoning-based indicators, meant to build automatically a fully explainable ranking of the most vulnerable national companies.

Acknowledgements. E. Sallinger acknowledges the support of the WWTF (Vienna Science and Technology Fund) grant VRG18-013, the EPSRC grant EP/M025268/1, and the EU Horizon 2020 grant 809965.

References

1. Angles, R.: The property graph database model. In: AMW (2018)
2. Atzeni, P., Bellomarini, L., Iezzi, M., Sallinger, E., Vlad, A.: Weaving enterprise knowledge graphs: The case of company ownership graphs. In: EDBT. pp. 555–566. OpenProceedings.org (2020)
3. Barabasi, A.L., Albert, R.: Emergence of scaling in random networks. Science (New York, N.Y.) 286, 509–12 (11 1999)
4. Barca, F., Becht, M.: The control of corporate europe. Oxford University Press, European Corporate Governance Network (2001)
5. Bellomarini, L., Benedetti, M., Gentili, A., Laurendi, R., Magnanimi, D., Muci, A., Sallinger, E.: COVID-19 and company knowledge graphs: Assessing golden powers and economic impact of selective lockdown via AI reasoning. CoRR abs/2004.10119 (2020)
6. Bellomarini, L., Gottlob, G., Pieris, A., Sallinger, E.: Swift logic for big data and knowledge graphs. In: IJCAI (2017)
7. Bellomarini, L., Sallinger, E., Gottlob, G.: The vadalog system: Datalog-based reasoning for knowledge graphs. PVLDB 11(9), 975–987 (2018)
8. Bonifati, A., Fletcher, G.H.L., Voigt, H., Yakovets, N.: Querying Graphs. Synthesis Lectures on Data Management, Morgan & Claypool (2018)
9. Calì, A., Gottlob, G., Lukasiewicz, T., Pieris, A.: Datalog+/-: A family of languages for ontology querying. In: Datalog Reloaded (2011), `http://dx.doi.org/10.1007/978-3-642-24206-9_20`
10. Ceri, S., Gottlob, G., Tanca, L.: Logic programming and databases. Springer (2012)
11. Fan, W., Wang, X., Wu, Y.: Performance guarantees for distributed reachability queries. Proceedings of the VLDB Endowment 5(11), 1304–1316 (2012)
12. Glattfelder, J.B., Battiston, S.: Backbone of complex networks of corporations: The flow of control. Physical Review E 80(3), 036104 (2009)
13. Gottlob, G., Lukasiewicz, T., Pieris, A.: Datalog+/-: Questions and answers. In: KR (2014)
14. Granovetter, M.: Coase revisited: Business groups in the modern economy. Industrial and corporate change 4(1), 93–130 (1995)
15. Hidalgo, C., Barabási, A.: Scale-free networks. Scholarpedia 3(1), 1716 (2008)
16. Kinnear, M., Shan, W.: The Legal Protection of Foreign Investment: A Comparative Study (with a Foreword by Meg Kinnear, Secretary-General of the ICSID). Bloomsbury Publishing (2012), `https://books.google.it/books?id=RyvcBAAAQBAJ`
17. Powell, R.: Takeover prediction models and portfolio strategies: A multinomial approach. Multinational Finance Journal 8(1/2), 35–72 (2004)
18. Puziak, M., Martyniuk, M.: Defensive strategies against hostile takeovers. In: Journal of International Studies (2012)
19. Reuters: Italy to defend strategic companies from foreign takeovers. `https://reut.rs/3g84ReL` (2020), [Online; accessed 20-May-2020]
20. Romei, A., Ruggieri, S., Turini, F.: The layered structure of company share networks. In: 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA). pp. 1–10. IEEE (2015)
21. Schoenberg, R., Thornton, D.: The Impact of Bid Defences in Hostile Acquisitions. European Management Journal 24(2-3), 142–150 (2006)