



3-Party Covenant Financing of ‘Semi-Regulated’ Pumped Hydro Assets

EPRG Working Paper 2405

Cambridge Working Paper in Economics 2425

Paul Simshauser and Nicholas Gohdes

Abstract

All credible scenarios of a decarbonising Australian power system with high levels of renewables rely on a portfolio of flexible, dispatchable storage and firming assets. Given our current understanding of costs and prices, such portfolios are thought to include short-duration batteries, intermediate-duration pumped hydro, with gas turbines providing the last line of defence. The stochastic intermittency of wind, the synchronicity of rooftop and utility-scale solar PV, and stubbornly inelastic aggregate final demand – at least over the medium term – only serve to underscore this point. Wind and solar output need to be moved through space (networks) and time (storage). The storage asset class with by far the highest energy density, pumped hydro, appears to be facing structurally high capital costs with most recent estimates given via high profile projects under development (viz. Snowy 2.0, Borumba) being ~\$5000-\$6000/kW in real terms. Yet under-development of pumped hydro will result in sharply rising renewable curtailment rates and a greater reliance on gas turbines – with the latter likely to be intractable. In this article, we focus on material reductions in the carrying cost of capital-intensive, ultra-long-lived pumped hydro assets through a 3-Party Covenant (3PC) financing structure between governments, the consumer base and plant investors. Our modelling suggests this reduces the annual capital costs and imputed cost of storage during operations by more than 35%. Our 3PC model is orchestrated through a semi-regulated business model and issuance of 10-year Commonwealth Government Bonds with a zero ‘credit spread’ – all of which are critical to minimise costs to consumers.

Keywords pumped hydro, energy storage, energy-only markets

JEL Classification D52, D53, G12, L94 and Q40

Contact

Publication May, 2024

Financial Support