Online supplement for "Combining polyhedral approaches and stochastic dual dynamic integer programming for solving the uncapacitated lot-sizing problem under uncertainty"

Franco Quezada
Sorbonne Université, LIP6, franco.quezada@lip6.fr,
Universidad de Santiago de Chile, LDSPS, franco.quezada@usach.cl,

Céline Gicquel
Université Paris Saclay, LISN, celine.gicquel@lri.fr,

Safia Kedad-Sidhoum
CNAM, CEDRIC, safia.kedad_sidhoum@cnam.fr,

This online supplement provides some complementary knowledge to the work presented in the paper "Combining polyhedral approaches and stochastic dual dynamic integer programming for solving the uncapacitated lot-sizing problem under uncertainty". First, it recalls two classes of valid inequalities used to strengthen the mixed-integer linear programming formulation of the stochastic uncapacitated lot-sizing problem. Second, a small numerical example is provided. This one illustrates how alternative strengthened Benders' cuts can be generated by using alternative mixed-integer linear programming formulations for the sub-problems. It also shows that these alternative strengthened Benders' cuts do not necessarily dominate each other. Third, the proposed extSDDiP algorithm is described in detail. Finally, extensive numerical results showing the individual impact of each of the three main elements of the proposed extSDDiP algorithm are provided.
Note that, unless otherwise noted, all section, page, paragraph, equation, table, and figure numbers of this online supplement refer to the main paper.

Key words: Multi-stage stochastic programming, stochastic lot-sizing, stochastic dual dynamic integer programming, node aggregation, valid inequalities, partial decomposition.

Appendix A: Strengthening of the extensive MILP formulation

This section recalls previously published results on the polyhedral structure of the extensive formulation (1)-(4) and their use to strengthen its linear relaxation.

Formulation (1)-(4) can be strengthened by applying valid inequalities known for the deterministic ULS to each path of the scenario tree.

Proposition 1. *Guan et al. (2006b).*
Given $\ell \in \mathcal{V}$ and $S \subseteq \mathcal{P}(1,\ell)$, the following (ℓ,S) inequality is valid for problem (1)-(4):

$$s^0 + \sum_{n\in S} x^n + \sum_{n\in \bar{S}} d^{n\ell} y^n \geq d^{1\ell} \tag{31}$$

with $d^{n\ell} = \sum_{m\in\mathcal{P}(n,\ell)} d^m$ and $\bar{S} = \mathcal{P}(1,\ell)\setminus S$.

For the deterministic ULS, the (ℓ,S) inequalities provide a full description of the convex hull of the feasible space. It is however not the case for the SULS. Guan et al. (2009) thus proposed to further strengthen formulation (1)-(4) by exploiting its tree structure.

Proposition 2. *Guan et al. (2009).*
Given a subset $\mathcal{O} = \{n_1,...,n_O\} \subseteq \mathcal{V}$ which is partially ordered such that $0 = d^{1n_0} \leq d^{1n_1} \leq ... \leq d^{1n_O}$ and a subset $S_{\mathcal{O}}$ of nodes belonging to $\cup_{o=1...O}\mathcal{P}(1,n_O)$, the following inequality is valid for problem (1)-(4):

$$s^0 + \sum_{n\in S_{\mathcal{O}}} x^n + \sum_{n\in \bar{S}_{\mathcal{O}}} \Delta_n(\mathcal{O}) y^n \geq d^{1n_O} \tag{32}$$

with $\Delta_n(\mathcal{O}) = \sum_{m_o\in\mathcal{O}\cap\mathcal{V}(n)}(d^{1m_o} - d^{1m_{o-1}})$.

Note that inequalities (31) can be seen as a special case of inequalities (32) in which $\mathcal{O}$ comprises a single node ℓ. Guan et al. (2006a) showed that a particular case of inequalities (32) suffices to fully describe the convex hull of the two-period SULS. However, this is not the case anymore when more than two planning periods are involved in the problem.

The number of valid inequalities (31) and (32) is too large to allow adding all of them a priori to formulation (1)-(4). Hence, a cutting-plane generation strategy is needed to add only a subset of these valid inequalities into the formulation. Consequently, the corresponding separation problems must be solved in order to identify which inequalities should be incorporated in the formulation. For inequalities (31), the separation problem can be solved exactly by the polynomial time algorithm proposed by Barany et al. (1984). For inequalities (32), the complexity status of the separation problem remains unknown but Guan et al. (2009) proposed a heuristic separation algorithm.

Appendix B: Generation of strengthened Benders' cuts using alternative MILP formulations of the sub-problems

This section provides additional insights about the generation of additional strengthened Benders' cuts using alternative MILP formulations of the sub-problems. We present a numerical example illustrating the fact that there is no dominance between the strengthened Benders' cuts generated while using different MILP formulations of the SULS. To do this, we use formulation (21)-(25) rather than formulation (9)-(17). It namely has a single continuous state variable at each node, which facilitates the graphic representation of the approximation of the expected cost-to-go function.

Example 1. Consider the following scenario tree involving $\mathcal{S} = 4$ stages and $R^{\sigma} = 3$ realizations at stages 2 to 4. Each realization is described by its stage σ and its realization index $r \in \{1,2,3\}$. The structure of the scenario tree is provided in Figure 1 and the corresponding values of the uncertain parameters are provided in Table 5.

(σ, r)	d	f	g	h
(1,1)	87	934	10	0
(2,1)	69	1182	20	10
(2,2)	38	585	13	2
(2,3)	73	1259	11	1
(3,1)	7	1743	18	5
(3,2)	86	956	20	6
(3,3)	23	108	12	10
(4,1)	14	643	3	6
(4,2)	11	1583	9	0
(4,3)	91	1074	13	10

Table 5 Numerical values for Example 1

Let us consider a partial decomposition involving $\Gamma = 2$ macro-stages with $\mathcal{S}(1) = \{1,2\}$ and $\mathcal{S}(2) = \{3,4\}$. At macro-stage $\gamma = 1$, we have a single realization $\mathcal{X}^{1,1} = \{(1,1),(2,1),(2,2),(2,3)\}$. At macro-stage $\gamma = 2$, we have 3 realizations: $\mathcal{X}^{2,1} = \{(3,1),(4,1),(4,2),(4,3)\}$, $\mathcal{X}^{2,2} = \{(3,2),(4,1),(4,2),(4,3)\}$, $\mathcal{X}^{2,3} = \{(3,3),(4,1),(4,2),(4,3)\}$.

At the first iteration of Algorithm 1, as $\psi_1^1 \equiv 0$, the feasible solution obtained at the end of the forward step is such that the leaving inventory at nodes $(2,1)$, $(2,2)$ and $(2,3)$ is equal to 0. The strengthened Benders' cuts generated during the backward step of the first iteration of Algorithm 1 to approximate $\mathcal{Q}^1(s^{\ell})$ are provided below:

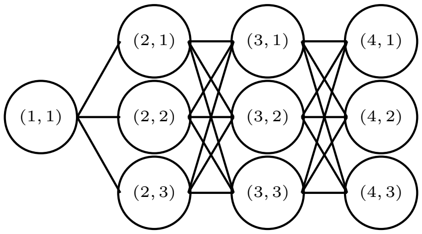

Figure 1 Scenario tree structure for Example 1

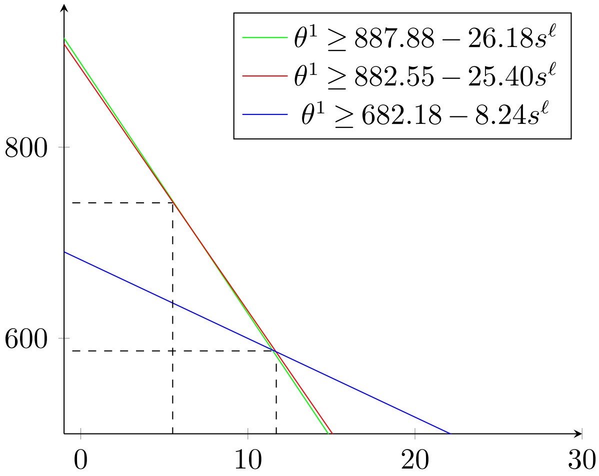

Figure 2 **Illustration for Example 1.**

- by solving the LP relaxation of sub-problems $\tilde{P}_1^2(0,0,\mathcal{X}^{2,r},IF(\emptyset))$, $r \in \{1,2,3\}$:

$$\theta^1 \geq 682.18 - 8.24s^\ell \tag{33}$$

- by solving the LP relaxation of sub-problems $\tilde{P}_1^2(0,0,\mathcal{X}^{2,r},IF(\phi_1^{2,r}))$, $r \in \{1,2,3\}$, in which $\phi_1^{2,r}$ contains only path inequalities (31):

$$\theta^1 \geq 882.55 - 25.40s^\ell \tag{34}$$

- by solving the LP relaxation of sub-problems $\tilde{P}_1^2(0,0,\mathcal{X}^{2,r},IF(\phi_1^{2,r}))$, $r \in \{1,2,3\}$, in which $\phi_1^{2,r}$ contains both path inequalities (31) and tree inequalities (32):

$$\theta^1 \geq 887.88 - 26.18s^\ell \tag{35}$$

Figure 1 provides a graphic representation of the three generated cuts. We observe that there is no dominance amongst them, i.e. none of the cuts seems to provide a better approximation of the expected cost-to-go function for all possible values of the leaving inventory level s^ℓ. More specifically, when s^ℓ lies in the interval $[1,7]$, the best approximation is obtained by the cut generated using formulation $IF(\phi_i^{\gamma,r})$ strengthened by inequalities (31) and (32). When s^ℓ lies in $[7,12]$, the best approximation is obtained by the cut generated using formulation $IF(\phi_i^{\gamma,r})$ strengthened only by inequalities (31). Finally, when s^ℓ is greater than 12, the best approximation is obtained by the cut generated using formulation $IF(\emptyset)$. This shows the practical interest of generating strengthened Benders'cuts based on alternative MILP formulation of the sub-problems.

Appendix C: Detailed description of the proposed extSDDiP

Algorithm 1: Strengthening sub-problems algorithm

```
if $\lambda = 0$ then
    $F = IF(\emptyset))$
else if $\lambda = 1$ then
    Run the cutting plane procedure to generate path inequalities (31)
    Add the generated inequalities to $\phi_i^{\gamma,r}$ to get $\phi_{i+1}^{\gamma,r}$
    $F = IF(\phi_i^{\gamma,r})$
else if $\lambda = 2$ then
    Run the cutting plane procedure to generate tree inequalities (32)
    Add the generated inequalities to $\phi_i^{\gamma,r}$ to get $\phi_{i+1}^{\gamma,r}$
    $F = IF(\phi_i^{\gamma,r})$
```

Algorithm 2: Approximate extSDDiP algorithm

```
while no stopping criterion is satisfied do
    Randomly select $K$ scenarios $\Omega_i = \{\omega_i^1, ..., \omega_i^K\}$
    for $k = 1, ..., K$ do
        for $\gamma = 1, ..., \Gamma$ do
            Solve $\tilde{P}_i^\gamma(s_i^m, \tilde{\psi}_i^\gamma, \mathcal{X}^{\gamma, r_i^{k,\gamma}}, IF(\phi_i^{\gamma,r}))$ for $m = \omega_i^k \cap \mathcal{V}^{t'(\gamma-1)}$
            Record $s_i^\ell$ for $\ell = \omega_i^k \cap \mathfrak{L}(\gamma, r_i^{k,\gamma})$
        end
    end
    for $\gamma = \Gamma - 1, ..., 1$ do
        for $k = 1, ..., K$ do
            for $r \in \mathcal{R}^{\gamma+1}$ do
                Let $m = \omega_i^k \cap \mathcal{V}^{t'(\gamma)}$
                Run Algorithm 1
                Solve the Linear relaxation of $\tilde{P}_i^{\gamma+1}(s_i^m, \tilde{\psi}_{i+1}^{\gamma+1}, \mathcal{X}^{\gamma+1,r}, F(\phi_{i+1}^{\gamma,r}))$ and collect the
                  coefficients of the strengthened Benders' cut
                Solve the Lagrangian relaxation of $\tilde{P}_i^{\gamma+1}(s_i^m, \tilde{\psi}_{i+1}^{\gamma+1}, \mathcal{X}^{\gamma+1,r}, F(\phi_{i+1}^{\gamma,r}))$ and collect the
                  constant value of the strengthened Benders' cut
            end
            Add the generated cut to $\tilde{\psi}_i^\gamma$ to get $\tilde{\psi}_{i+1}^\gamma$
        end
        if a criterion for Algorithm 1 is satisfied then
            $\lambda \leftarrow \lambda + 1$
        end
        $LB \leftarrow \tilde{Q}_{i+1}^{1,1}(0)$
        $i \leftarrow i + 1$
    end
end
```

Algorithm 3: extSDDiP algorithm

```
Initialize $LB \leftarrow -\infty, UB \leftarrow +\infty, i \leftarrow 1, \lambda \leftarrow 0$
for $\gamma = \Gamma - 1, ..., 1$ do
    for $r \in \mathcal{R}^{\gamma+1}$ do
        Initialize $\phi^{\gamma,r} \leftarrow \emptyset$
    end
end
Run Algorithm 2
while no stopping criterion is satisfied do
    Randomly select $K$ scenarios $\Omega_i = \{\omega_i^1, ..., \omega_i^K\}$
    for $k = 1, ..., K$ do
        for $\gamma = 1, ..., \Gamma$ do
            Solve $\hat{P}_i^{\gamma}(u_i^m, \psi_i^{\gamma}, \mathcal{X}^{\gamma, r_i^{k,\gamma}}, IF(\phi_i^{\gamma,r}))$ for $m = \omega_i^k \cap \mathcal{V}^{t'(\gamma-1)}$.
            Record $u_i^{\ell}$ for $\ell = \omega_i^k \cap \mathfrak{L}(\gamma, r_i^{k,\gamma})$
        end
        $\upsilon^k \leftarrow \sum_{n \in \omega_i^k} (f^n y_i^n + h^n s_i^n + g^n x_i^n)$
    end
    $\hat{\mu} \leftarrow \sum_{k=1}^K \upsilon^k$ and $\hat{\chi}^2 \leftarrow \frac{1}{K-1} \sum_{k=1}^K (\upsilon^k - \hat{\mu})^2$
    $UB \leftarrow \hat{\mu} + z_{\alpha/2} \frac{\hat{\chi}}{\sqrt{K}}$
    for $\gamma = \Gamma - 1, ..., 1$ do
        for $k = 1, ..., K$ do
            for $r \in \mathcal{R}^{\gamma+1}$ do
                Let $m = \omega_i^k \cap \mathcal{V}^{t'(\gamma)}$
                Run Algorithm 1
                Solve the Linear relaxation of $\hat{P}_i^{\gamma+1}(u_i^m, \psi_{i+1}^{\gamma+1}, \mathcal{X}^{\gamma+1,r}, F(\phi_{i+1}^{\gamma,r}))$ and collect the
                  coefficients of the strengthened Benders' cut
                Solve the Lagrangian relaxation of $\hat{P}_i^{\gamma+1}(u_i^m, \psi_{i+1}^{\gamma+1}, \mathcal{X}^{\gamma+1,r}, F(\phi_{i+1}^{\gamma,r}))$ and collect the
                  constant value of the strengthened Benders' cut
                Solve $\hat{P}_i^{\gamma+1}(u_i^m, \psi_{i+1}^{\gamma+1}, \mathcal{X}^{\gamma+1,r}, F(\phi_{i+1}^{\gamma,r}))$ and collect the coefficients of the integer
                  optimality cut
                -Solve the Lagrangian dual problem and collect the coefficients of the Lagrangian cut
            end
            Add the 3 generated cuts to $\psi_i^{\gamma}$ to get $\psi_{i+1}^{\gamma}$
        end
        if a criterion for Algorithm 1 is satisfied then
            $\lambda \leftarrow \lambda + 1$
        end
        $LB \leftarrow \hat{Q}_{i+1}^{1,1}(0)$
        $i \leftarrow i + 1$
    end
end
```

Appendix D: Computational assessment of the various components of the extSDDiP algorithm

This section provides the results of additional computational experiments carried out in order to assess the individual impact of each of the three main elements of algorithm extSDDiP, namely the partial decomposition of the stochastic problem into smaller stochastic sub-problems, the introduction of an initial phase in which the state variables are kept continuous and the exploitation of alternative MILP formulations of the stochastic sub-problems to generate additional strengthened Benders's cuts, on its overall performance.

Tables 6-9 provide the results obtained while running extSDDiP-I, i.e. running only the approximate sub-tree based algorithm. Tables 10-13 provide the results obtained while running extSDDiP-II, i.e. running only Algorithm 1 without an initial phase based on the approximate sub-tree based algorithm.

Table 6 Performance of the approximate algorithm at solving instances from Set 1 ($\Sigma = 4, b = 1$) of the SULS problem

R	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{L}(1)\rvert$	G	Method	Gap	Time (s)	# ite	# VI
10	1111	1000	1	SDDiP	9.59	1,100.96	135	0
			2	extSDDiP-I-0	9.44	7.42	16	0
				extSDDiP-I-1	6.55	12.56	33	92
				extSDDiP-I-2	5.61	21.26	64	678
			4	CPX	1.36	1590	-	2642
20	8420	8000	1	SDDiP	15.62	1,074.37	77	0
			2	extSDDiP-I-0	8.28	25.40	18	0
				extSDDiP-I-1	5.25	36.50	35	377
				extSDDiP-I-2	4.43	75.93	86	5,545
			4	CPX	4.67	1,801	-	3224

Table 7 Performance of the approximate algorithm at solving instances from Set 2 ($\Sigma = 6, b = 1$) of the SULS problem

R	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{L}(1)\rvert$	G	Method	Gap	Time (s)	# ite	# VI
10	111111	100000	1	SDDiP	21.49	1,241.18	80	0
			2	extSDDiP-I-0	13.91	23.22	23	0
				extSDDiP-I-1	8.93	44.99	54	286
				extSDDiP-I-2	6.73	104.13	101	1,563
			3	extSDDiP-I-0	7.66	146.98	18	0
				extSDDiP-I-1	4.85	288.06	38	1,886
				extSDDiP-I-2	4.56	787.07	71	20,829
			6	CPX	19.28	1801	-	0
20	3.36×10^6	3.2×10^6	1	SDDiP	28.77	1,214.26	46	0
			2	extSDDiP-I-0	15.47	71.33	30	0
				extSDDiP-I-1	8.88	198.51	67	1,132
				extSDDiP-I-2	7.95	621.07	137	11,762
			3	extSDDiP-I-0	10.67	1,349.91	8	0
				extSDDiP-I-1	10.75	1,369.64	7	0
				extSDDiP-I-2	11.81	1,386.40	7	0
			6	CPX	94.24	1,801	-	0

Table 8 Performance of the approximate algorithm at solving instances from Set 3 ($\Sigma = 8, R = 5$) of the SULS

b	$\|\mathcal{V}\|$	$\|\mathcal{L}(1)\|$	G	Method	Gap	Time (s)	# ite	# VI
2	195312	78125	1	SDDiP	20.08	1662.00	91	0
			2	extSDDiP-I-0	15.16	40.83	21	0
				extSDDiP-I-1	6.61	55.23	52	432
				extSDDiP-I-2	6.30	76.85	84	1,002
			4	extSDDiP-I-0	7.04	676.63	13	0
				extSDDiP-I-1	3.49	1,011.61	28	6,008
				extSDDiP-I-2	3.12	1,285.02	32	7,975
			8	CPX	43.81	1,802.40	-	0
5	488280	78125	1	SDDiP	13.14	2075.09	70	0
			2	extSDDiP-I-0	13.73	113.42	21	0
				extSDDiP-I-1	2.56	130.48	49	2,196
				extSDDiP-I-2	2.58	163.25	65	3,746
			4	extSDDiP-I-0	6.39	1,179.24	12	0
				extSDDiP-I-1	3.12	1,739.40	18	19,857
				extSDDiP-I-2	2.89	1,765.80	19	21,455
			8	CPX	71.04	1,803.99	-	0

Table 9 Performance of the approximate algorithm at solving instances from Set 4 ($\Sigma = 12, R = 3, b = 1$) of the SULS

$\|\mathcal{V}\|$	$\|\mathcal{L}(1)\|$	G	Method	Gap	Time (s)	# ite	# VI
265720	177147	1	SDDiP	29.12	1,716.06	100	0
		2	extSDDiP-I-0	24.01	73.32	20	0
			extSDDiP-I-1	11.15	88.51	53	86
			extSDDiP-I-2	10.32	104.50	81	137
		3	extSDDiP-I-0	15.78	23.95	19	0
			extSDDiP-I-1	7.87	36.14	47	235
			extSDDiP-I-2	6.43	59.33	84	775
		4	extSDDiP-I-0	12.82	74.63	17	0
			extSDDiP-I-1	4.84	99.06	38	627
			extSDDiP-I-2	4.80	196.46	64	2,535
		6	extSDDiP-I-0	7.72	1,009.62	10	0
			extSDDiP-I-1	4.25	1,275.45	18	3,323
			extSDDiP-I-2	4.11	1,361.15	23	4,747
		12	CPX	53.78	1,803.29	-	0

Table 10 Performance of Algorithm 1 at solving instances from Set 1 ($\Sigma = 4, b = 1$) of the SULS problem

R	$\|\mathcal{V}\|$	$\|\mathcal{L}(1)\|$	G	Method	Gap	Time (s)	# ite	# VI
10	1111	1000	1	SDDiP	9.59	1,100.96	135	0
			2	extSDDiP-II-0	4.86	951.89	74	0
				extSDDiP-II-1	3.00	908.93	77	879
				extSDDiP-II-2	3.16	928.67	72	875
			4	CPX	1.36	1590	-	2642
20	8420	8000	1	SDDiP	15.62	1,074.37	77	0
				extSDDiP-II-0	5.54	971.92	42	0
				extSDDiP-II-1	3.58	961.21	41	5,075
				extSDDiP-II-2	3.70	958.20	41	5,068
			4	CPX	4.67	1,801	-	3224

Table 11 Performance of Algorithm 1 at solving instances from Set 2 ($\Sigma = 6, b = 1$) of the SULS problem

R	$\|\mathcal{V}\|$	$\|\mathcal{L}(1)\|$	G	Method	Gap	Time (s)	# ite	# VI
10	111111	100000	1	SDDiP	21.49	1,241.18	80	0
			2	extSDDiP-II-0	12.42	1,182.40	25	0
				extSDDiP-II-1	9.21	1,187.84	25	1,400
				extSDDiP-II-2	8.84	1,176.92	25	1,389
			3	extSDDiP-II-0	7.68	1,086.78	21	0
				extSDDiP-II-1	6.11	1,179.83	15	10,557
				extSDDiP-II-2	5.27	1,167.99	15	10,489
			6	CPX	19.28	1801	-	0
20	3.36×10^6	3.2×10^6	1	SDDiP	28.77	1,214.26	46	0
			2	extSDDiP-II-0	17.35	1,118.36	11	0
				extSDDiP-II-1	17.50	1,274.20	10	4,081
				extSDDiP-II-2	16.85	1,268.43	9	3,957
			3	extSDDiP-II-0	-	-	-	-
				extSDDiP-II-1	-	-	-	-
				extSDDiP-II-2	-	-	-	-
			6	CPX	94.24	1,801	-	0

Table 12 **Performance of Algorithm 1 at solving instances from Set 3 ($\Sigma = 8, R = 5$) of the SULS problem**

b	$\|\mathcal{V}\|$	$\|\mathcal{L}(1)\|$	G	Method	Gap	Time (s)	# ite	# VI
2	195312	78125	1	SDDiP	20.08	1662.00	91	0
				SDDiP-1	9.72	1,617.19	115	121
			2	extSDDiP-II-0	13.42	1,653.96	33	0
				extSDDiP-II-1	6.93	1,665.12	37	1,106
				extSDDiP-II-2	7.21	1,667.20	35	1,101
			4	extSDDiP-II-0	6.91	1,475.52	13	0
				extSDDiP-II-1	9.29	1,645.23	9	10,933
				extSDDiP-II-2	8.80	1,632.74	9	11,075
			8	CPX	43.81	1,802.40	-	0
5	488280	78125	1	SDDiP	13.14	2075.09	70	0
				SDDiP-1	5.71	2,025.49	100	1,204
			2	extSDDiP-II-0	7.76	1,780.95	29	0
				extSDDiP-II-1	3.66	1,730.56	35	4,815
				extSDDiP-II-2	3.53	1,703.66	37	4,831
			4	extSDDiP-II-0	6.41	1,629.26	10	0
				extSDDiP-II-1	24.18	1,864.58	4	20,673
				extSDDiP-II-2	26.63	1,862.54	4	18,764
			8	CPX	71.04	1,803.99	-	0

Table 13 **Performance of Algorithm 1 at solving instances from Set 4 ($\Sigma = 12, R = 3, b = 2$) of the SULS problem**

$\|\mathcal{V}\|$	$\|\mathcal{L}(1)\|$	G	Method	Gap	Time (s)	# ite	# VI
265720	177147	1	SDDiP	29.12	1,716.06	100	0
		2	extSDDiP-II-0	16.32	1,693.26	60	0
			extSDDiP-II-1	8.11	1,618.20	73	138
			extSDDiP-II-2	8.32	1,651.21	72	138
		3	extSDDiP-II-0	13.67	1,725.26	25	0
			extSDDiP-II-1	7.76	1,726.47	27	741
			extSDDiP-II-2	7.79	1,712.67	27	740
		4	extSDDiP-II-0	11.67	1,853.11	13	0
			extSDDiP-II-1	8.88	1,852.27	12	1,662
			extSDDiP-II-2	8.78	1,865.50	12	1,657
		6	extSDDiP-II-0	6.18	1,601.00	12	0
			extSDDiP-II-1	6.28	1,760.09	10	7,264
			extSDDiP-II-2	5.51	1,728.80	9	7,134
		12	CPX	53.78	1,803.29	-	0

Appendix E: Computational results on instances involving a large number of stages

This section provides the results of additional computational experiments carried out to assess the performance of the extSDDiP algorithm when using large values of G.

We used the procedure described in Subsection 5.1 of the manuscript to generate a new set of instances involving $\Sigma = 20$ stages, $b = 1$ period per stage and $R \in \{2, 3\}$ realizations per stage. We solve these instances with the mathematical solver CPLEX using the extensive MILP formulation, with the SDDiP algorithm proposed by Zou et al. (2019) and with the extSDDiP-I/II-2 algorithm for values of G in the set $\{2, 4, 5, 10\}$. The results are reported in Table 14.

As already discussed in Subsection 5.3, we first note that increasing G from 1 to 2, 4 or 5 leads to a significant improvement in the solution quality. However, further increasing G to 10 leads to a degradation of the algorithmic performance. This can be seen in particular for the instances corresponding to $R = 3$: on average, for these instances, the extSDDiP-I/II-2 algorithm with $G = 10$ provides solutions within a gap significantly larger than the one obtained using the SDDiP algorithm.

This might be explained as follows. When G is large, the extSDDiP algorithm decomposes the initial problem into a small number of large sub-problems. Solving each one of these sub-problems is computationally demanding so that the algorithm is able to perform only a limited number of iterations within the allotted time. Note e.g. how, on average, the extSDDiP-I/II-2 algorithm with $G = 10$ is able to carry out only 11 iterations when solving instances with $R = 3$ whereas the same algorithm solving the same instances with $G = 2$ carries out an average of 286 iterations. This reduction in the number of iterations negatively impacts the quality of the approximate expected cost-to-go functions built over the course of the algorithm, and as a consequence, the quality of the lower and upper bounds obtained by the algorithm.

Note that this phenomenon is also observed in the results reported in Tables 2 and 11 for the instances of Set 2 involving $R = 20$ realizations per stage. For these instances, increasing G from 2 to 3 negatively impacts the solution gap.

Table 14 **Performance of each method at solving instances with $\Sigma = 20$ and $b = 1$ of the SULS problem**

R	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{L}(1)\rvert$	G	Method	Gap	Time (s)	# ite	# VI
2	1,048,570	524,288	1	SDDiP	20.74	2,381.83	116	0
			2	extSDDiP-I/II-0	16.21	1,171.41	151	0
				extSDDiP-I/II-1	4.96	1,114.92	219	105
				extSDDiP-I/II-2	8.37	1,188.19	216	105
			4	extSDDiP-I/II-0	15.23	1,700.43	56	0
				extSDDiP-I/II-1	5.63	1,288.44	95	639
				extSDDiP-I/II-2	5.67	1,282.92	142	737
			5	extSDDiP-I/II-0	11.98	1,820.73	42	0
				extSDDiP-I/II-1	6.53	1,565.08	72	1,102
				extSDDiP-I/II-2	6.17	1,553.27	103	1,495
			10	extSDDiP-I/II-0	6.51	1,240.04	13	0
				extSDDiP-I/II-1	7.05	1,396.39	20	3,978
				extSDDiP-I/II-2	6.45	1,430.71	25	4,390
			20	CPX	88.58	1,805.53	-	-
3	$1.74 \cdot 10^9$	$1.16 \cdot 10^9$	1	SDDiP	24.23	2,212.64	93	0
			2	extSDDiP-I/II-0	26.51	1,128.94	108	0
				extSDDiP-I/II-1	9.52	1,013.80	229	276
				extSDDiP-I/II-2	11.21	1,061.60	286	275
			4	extSDDiP-I/II-0	18.45	1,862.04	53	0
				extSDDiP-I/II-1	9.23	1,855.41	96	2,719
				extSDDiP-I/II-2	8.03	1,605.79	127	5,697
			5	extSDDiP-I/II-0	13.54	1,910.29	36	0
				extSDDiP-I/II-1	7.05	1,861.89	62	4,304
				extSDDiP-I/II-2	9.57	1,829.56	60	4,201
			10	extSDDiP-I/II-0	75.37	1,787.26	10	0
				extSDDiP-I/II-1	71.60	1,877.45	12	0
				extSDDiP-I/II-2	61.54	1,773.12	11	0
			20	CPX	-	-	-	-

References

Barany I, Van Roy TJ, Wolsey LA (1984) Strong formulations for multi-item capacitated lot sizing. *Management Science* 30(10):1255–1261.

Guan Y, Ahmed S, Miller AJ, Nemhauser GL (2006a) On formulations of the stochastic uncapacitated lot-sizing problem. *Operations Research Letters* 34(3):241–250.

Guan Y, Ahmed S, Nemhauser GL (2009) Cutting planes for multistage stochastic integer programs. *Operations Research* 57(2):287–298.

Guan Y, Ahmed S, Nemhauser GL, Miller AJ (2006b) A branch-and-cut algorithm for the stochastic uncapacitated lot-sizing problem. *Mathematical Programming* 105(1):55–84.

Zou J, Ahmed S, Sun XA (2019) Stochastic dual dynamic integer programming. *Mathematical Programming* 175(1-2):461–502.